The purpose of this notebook is to elucidate the two parts of projection and minimization.
Projection is analytical, whereas minimization is numerical and employs the Nminimize function.
INSTRUCTIONS: Run common_funs.nb, ES_FindSymGroups.nb, BC_ChooseTmat.nb, then this notebook
contact: Carl Tape (ctape@alaska.edu)

In[1246]:=
```
WantDetails = "WantDetails";
```

In[1247]:=
```
PaxesForLattice =
  Paxes[True, True, True, id, {0, 0, 0}, 1.35, 1, .17, .009, "Gray", ArrowColor];
```

choose a Tmat from BC_ChooseTmat.nb or define your own here
(This is the BB basis for Tmat; if you want Voigt, you can define C and then use TmatofCmat().)

In[1248]:=
```
Tmat = TmatIgel;
```

KEY FUNCTION: this will maximize (NMazimize) instead of minimize (NMinimize)
After running this function, {θ0,σ0,ϕ0} can be used to find the U associated with the maximum.

In[1249]:=
```
GetTempAndθ0σ0ϕ0Max[Tmat_, Σ_] := (Clear[θ, σ, ϕ];
  temp[Tmat, Σ] = NMaximize[{DistToVΣofU[Tmat, UsHat[{θ, σ, ϕ}], Σ],
     0 ≤ θ ≤ 2 π && -π ≤ σ ≤ π && 0 ≤ ϕ ≤ π}, {θ, σ, ϕ}, Method → "RandomSearch"];
  {θ0, σ0, ϕ0} = ({θ, σ, ϕ} /. temp[Tmat, Σ][[2]]);
  If[MemberQ[{XISO, MONO}, Σ], σ0 = 0];)  (* optional *)
```

Example elastic map

In[1250]:=
```
PrintVoigt[Tmat]
TISO = ProjToVSigOfU[Tmat, id, ISO];
```

The $[T]_{\mathbb{BB}}$ matrix is $\begin{pmatrix} 10. & 0.7 & 1.05 & 5.2 & -3.92598 & -3.10269 \\ 0.7 & 8. & -2. & -0.2 & -0.46188 & 0.326599 \\ 1.05 & -2. & 6. & -0.45 & -0.190526 & 0.318434 \\ 5.2 & -0.2 & -0.45 & 5.5 & 0 & -1.22474 \\ -3.92598 & -0.46188 & -0.190526 & 0 & 5.5 & 2.12132 \\ -3.10269 & 0.326599 & 0.318434 & -1.22474 & 2.12132 & 13. \end{pmatrix}$

The eigenvalues are: $\{17.7924, 10.1797, 9.22349, 5.83219, 4.30288, 0.669364\}$

The Voigt matrix is $\begin{pmatrix} 10. & 3.5 & 2.5 & -5. & 0.1 & 0.3 \\ 3.5 & 8. & 1.5 & 0.2 & -0.1 & -0.15 \\ 2.5 & 1.5 & 6. & 1. & 0.4 & 0.24 \\ -5. & 0.2 & 1. & 5. & 0.35 & 0.525 \\ 0.1 & -0.1 & 0.4 & 0.35 & 4. & -1. \\ 0.3 & -0.15 & 0.24 & 0.525 & -1. & 3. \end{pmatrix}$

3-19-2024 19:44

$\mathcal{V}_{\mathrm{ISO}}(U) = \mathcal{V}_{\mathrm{ISO}}(I)$, independent of U, so the function U
$\longrightarrow d(T, \mathcal{V}_{\mathrm{ISO}}(U)) = d(T, \mathcal{T}_{\mathrm{ISO}})$ is constant.

$d(T, \mathcal{T}_{\mathrm{ISO}}) = 12.0102$ (Distance from T to $\mathcal{T}_{\mathrm{ISO}}$)

$\beta_{\mathrm{ISO}}^{\square T} = 0.533 = 30.55^{\circ}$ (Angle between T and $\mathcal{T}_{\mathrm{ISO}}$)

$$P(T, \mathcal{V}_{\mathrm{ISO}}(I)) = \begin{pmatrix} 7. & 0. & 0. & 0. & 0. & 0. \\ 0. & 7. & 0. & 0. & 0. & 0. \\ 0. & 0. & 7. & 0. & 0. & 0. \\ 0. & 0. & 0. & 7. & 0. & 0. \\ 0. & 0. & 0. & 0. & 7. & 0. \\ 0. & 0. & 0. & 0. & 0. & 13. \end{pmatrix}$$

(The closest elastic map to T having symmetry ISO)

Lamé parameters for $P(T, \mathcal{V}_{\mathrm{ISO}}(I))$ are $\lambda = 2.$
and $\mu = 3.5$, bulk modulus $\kappa = 4.33333$, Poisson ratio $= 0.181818$

plot the (TRIV) map as a sphere

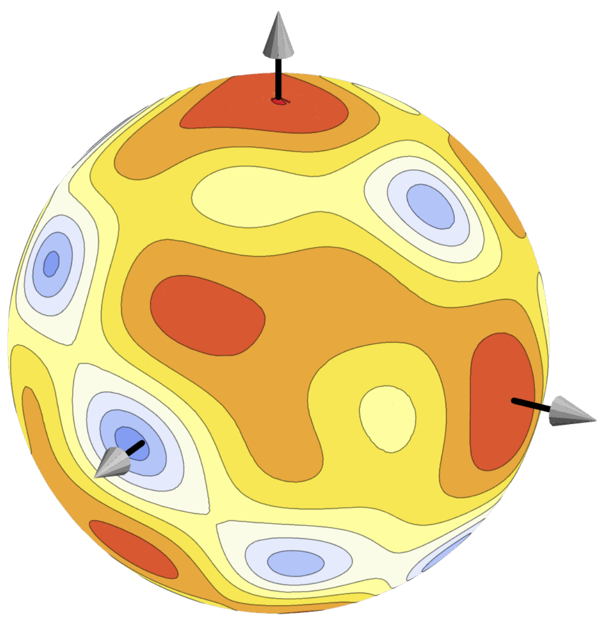
In[1252]:=

```
MatrixNote[Tmat]
Show[{cpMONO[Tmat, contoursMONO[Tmat], MaxForScaling[Tmat],
   plotpoints, contourstyle], Graphics3D[PaxesForLattice]}, options]
```

Out[1252]=

```
T is Igel
```

Out[1253]=



Closest to MONO maps

In[1254]:=

```
OutputFor[Tmat, MONO]
```

3-19-2024 19:44

NMinimize = {1.77381, {$\theta \to 6.2378$, $\sigma \to -1.04197$, $\phi \to 1.58615$}}

(minimizing the function $(\theta, \sigma, \phi) \longrightarrow d(T, \mathcal{V}_{MONO}(U))$, $U = \hat{U}_S(\theta,\sigma,\phi)$)

($(\theta_0, \sigma_0, \phi_0)$ in degrees are {357.4, 0., 90.9})

$$U_0 = \hat{U}_S(\theta_0,\sigma_0,\phi_0) = \begin{pmatrix} -0.0153358 & 0.0453676 & 0.998853 \\ 0.000696467 & 0.99897 & -0.0453622 \\ -0.999882 & 0 & -0.0153516 \end{pmatrix} \quad (A$$

MONO-minimizer for T. It minimizes the function $U \longrightarrow d(T, \mathcal{V}_{MONO}(U))$)

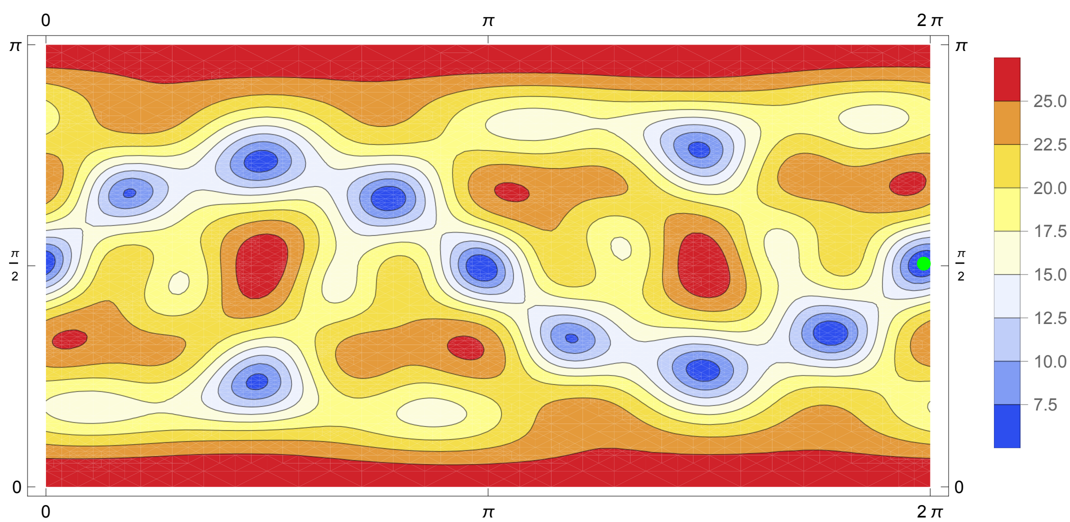
$\beta^{T}_{MONO} = 0.075 = 4.31°$ (Angle between T and $\mathcal{T}_{MONO}$)

$\mathcal{V}_{MONO}(U_0)$ contains a closest map to T in $\mathcal{T}_{MONO}$

$d(T, \mathcal{T}_{MONO}) = 1.774$ (Distance from T to $\mathcal{T}_{MONO}$)

Does the green point look like a correct
GLOBAL minimum? If not, the minimization has failed.

Uniqueness (or not) of the closest MONO map is usually obvious from the contour plot.



$$P(T, \mathcal{V}_{MONO}(U_0)) = \begin{pmatrix} 10.0483 & 0.312292 & 0.626973 & 5.21023 & -3.95328 & -3.07084 \\ 0.312292 & 7.96402 & -1.96673 & 0.37924 & -0.115108 & -0.213739 \\ 0.626973 & -1.96673 & 6.07884 & 0.0499996 & -0.114125 & -0.15424 \\ 5.21023 & 0.37924 & 0.0499996 & 5.39035 & 0.00319104 & -1.17338 \\ -3.95328 & -0.115108 & -0.114125 & 0.00319104 & 5.5185 & 2.10692 \\ -3.07084 & -0.213739 & -0.15424 & -1.17338 & 2.10692 & 13. \end{pmatrix}$$

(a closest elastic map to T having symmetry at least MONO)

plot the closest MONO map as a sphere

In[1255]:=

```
TMONO = Closest[Tmat, MONO];
UMONO = UT[TMONO, MONO];
OutputFor[TMONO, MONO]
```

3-19-2024 19:44

NMinimize = $\{7.6283\times10^{-7}, \{\theta\to 6.2378, \sigma\to -3.12923, \phi\to 1.58615\}\}$

(minimizing the function $(\theta, \sigma, \phi) \longrightarrow d(T, \mathcal{V}_{MONO}(U))$, $U = \hat{U}_S(\theta,\sigma,\phi)$)

($(\theta_0, \sigma_0, \phi_0)$ in degrees are {357.4, 0., 90.9})

$$U_0 = \hat{U}_S(\theta_0,\sigma_0,\phi_0) = \begin{pmatrix} -0.0153358 & 0.0453676 & 0.998853 \\ 0.000696467 & 0.99897 & -0.0453623 \\ -0.999882 & 0 & -0.0153516 \end{pmatrix}$$ (A

MONO-minimizer for T. It minimizes the function $U \longrightarrow d(T, \mathcal{V}_{MONO}(U))$)

$\beta^T_{MONO} = 0$ (Angle between T and $\mathcal{T}_{MONO}$. The initial $\beta^T_{MONO} = 3.24\times10^{-8}$ is here chopped to zero, since the chop threshold is set at 0.01°)

Since $\beta^T_{MONO} = 0$, T is assigned symmetry at least MONO.

$\mathcal{V}_{MONO}(U_0)$ contains T

$d(T, \mathcal{T}_{MONO}) = 7.6\times10^{-7}$ (Distance from T to $\mathcal{T}_{MONO}$)

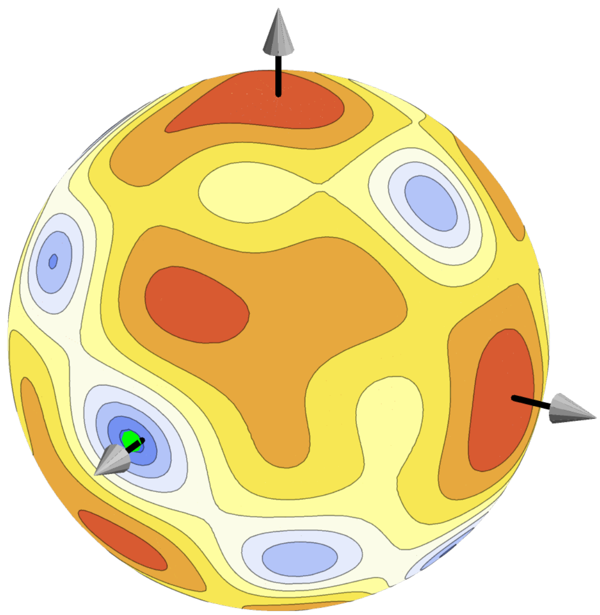
$$T_{MONO} = [\overline{U_0}]^T[T][\overline{U_0}] = \begin{pmatrix} 6.06321 & 2.00189 & 0 & 0 & 0 & 0 \\ 2.00189 & 7.94231 & 0 & 0 & 0 & 0 \\ 0 & 0 & 10.0733 & 0.802002 & -6.52726 & 3.07774 \\ 0 & 0 & 0.802002 & 5.48337 & -0.0517244 & 1.23396 \\ 0 & 0 & -6.52726 & -0.0517244 & 5.43786 & -2.07855 \\ 0 & 0 & 3.07774 & 1.23396 & -2.07855 & 13. \end{pmatrix}$$

(Should be a MONO-ref matrix for T. If not, consider decreasing ChopT.)

In[1258]:=

```
TwoFold = Graphics3D[
   TwoFoldGraphics[MONO, UMONO, 1.25 AngRadDisk, 3 ShiftToEye, eye, tkns, Green]];
Show[
 {cpMONO[TMONO, contoursMONO[Tmat], MaxForScaling[Tmat], plotpoints, contourstyle],
  TwoFold, Graphics3D[PaxesForLattice]}, options]
```

Out[1259]=



check

In[1260]:=

```
UforNM1 = UMONO;
TMONOx = ProjToVSigOfU[Tmat, UforNM1, MONO];
AngleMatrix[Tmat, TMONO] / Degree
AngleMatrix[Tmat, TMONOx] / Degree
AngleMatrix[TMONO, TMONOx] / Degree
```

Out[1262]=

4.30551

Out[1263]=

4.30551

Out[1264]=

1.47878×10^{-6}

What would our closest-to-MONO map look like for the default U (=Identity)?

In[1265]:=

```
UforNM1 = id;
TMONOx = ProjToVSigOfU[Tmat, UforNM1, MONO];
AngleMatrix[Tmat, TMONO] / Degree
AngleMatrix[Tmat, TMONOx] / Degree
AngleMatrix[TMONO, TMONOx] / Degree
```

Out[1267]=

```
4.30551
```

Out[1268]=

```
27.0034
```

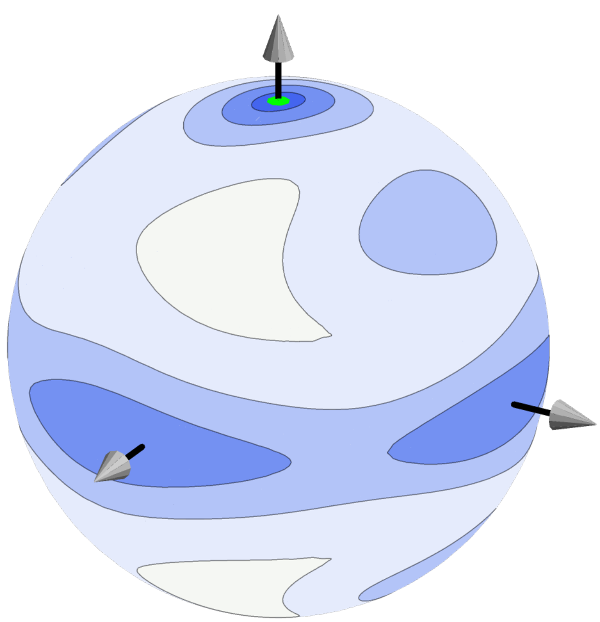
Out[1269]=

```
27.0269
```

In[1270]:=

```
TwoFold = Graphics3D[
   TwoFoldGraphics[MONO, UforNM1, 1.25 AngRadDisk, 3 ShiftToEye, eye, tkns, Green]];
Show[
 {cpMONO[TMONOx, contoursMONO[Tmat], MaxForScaling[Tmat], plotpoints, contourstyle],
  TwoFold, Graphics3D[PaxesForLattice]}, options]
```

Out[1271]=



Among the projected MONO maps, which one is farthest from our target map? Since $\mathcal{V}_{\text{ISO}}$ is contained in every $\mathcal{V}_{\Sigma}(U)$ and since your TMONOx is in

$\mathcal{V}_{\text{MONO}}$(UMONOmax), then all ISO maps are at least as far from Tmat as is TMONOx.

To get a truly (i.e., non ISO) MONO map that is farther from Tmat than TMONOx, write TMONOx in the form TMONOx = U $\mathcal{T}_{\text{MONO}}$ U^T, where $\mathcal{T}_{\text{MONO}}$ is a MONO ref matrix. (You already have U.)Then change a couple of entries of $\mathcal{T}_{\text{MONO}}$, but keeping it a MONO ref matrix. The new U $\mathcal{T}_{\text{MONO}}$ U^T will be farther from Tmat (since TMONOx was a projection).

In[1272]:=

```
GetTempAndθ0σ0ϕ0Max[Tmat, MONO]
```

In[1273]:=

```
UMONOmax = UsHat[{θ0, σ0, ϕ0}];
TMONOx = ProjToVSigOfU[Tmat, UMONOmax, MONO];
AngleMatrix[Tmat, TMONO] / Degree
AngleMatrix[Tmat, TMONOx] / Degree
AngleMatrix[TMONO, TMONOx] / Degree
```

Out[1275]=

```
4.30551
```

Out[1276]=

```
27.0575
```

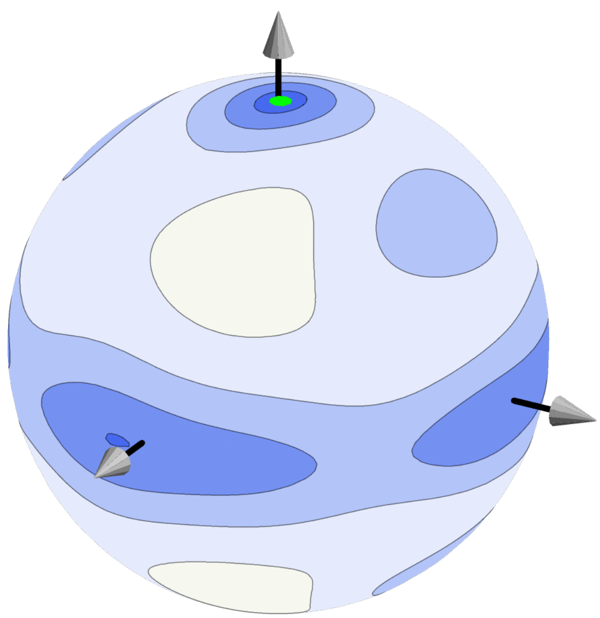
Out[1277]=

```
27.0448
```

In[1278]:=

```
TwoFold = Graphics3D[
   TwoFoldGraphics[MONO, UMONOmax, 1.25 AngRadDisk, 3 ShiftToEye, eye, tkns, Green]];
Show[
 {cpMONO[TMONOx, contoursMONO[Tmat], MaxForScaling[Tmat], plotpoints, contourstyle],
  TwoFold, Graphics3D[PaxesForLattice]}, options]
```

Out[1279]=

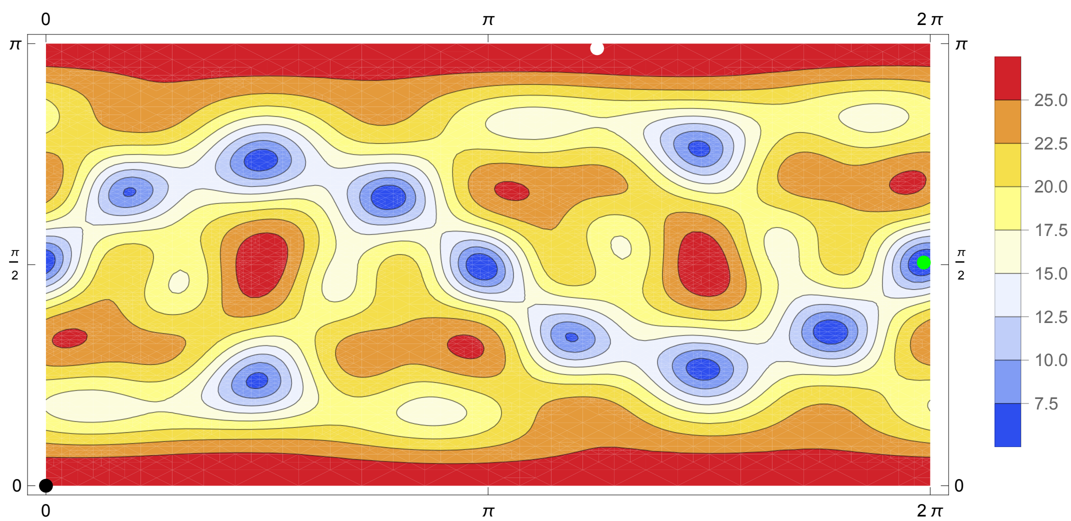


replot the map, but with the U=Id and U=Umax points added

In[1280]:=

```
Show[{ContourPlot[Alpha[Tmat, {θ, ϕ}, MONO] / Degree,
    {θ, 0, 2 π}, {ϕ, 0, π}, Contours → Automatic,
    ColorFunction → (ColorData["TemperatureMap"]), PlotLegends → Automatic],
   Graphics[
    {PointSize[.015], {Green, Point[TPxyz0to2pi[UT[Tmat, MONO].{0, 0, 1}]]}}],
   Graphics[{PointSize[.015], {White, Point[TPxyz0to2pi[UMONOmax.{0, 0, 1}]]}}],
   Graphics[{PointSize[.015], {Black, Point[{0, 0}]}}]
  }, FrameTicks → {Range[0, 2 π, π], Range[0, π, π / 2]},
  AspectRatio → Automatic, PlotRange → All, ImageSize → 500]
```

Out[1280]=



Closest to XISO maps

In[1281]:=

```
OutputFor[Tmat, XISO]
```

3-19-2024 19:46

NMinimize = {6.50485, {$\theta \to 1.50483$, $\sigma \to 3.07695$, $\phi \to 0.714334$}}
(minimizing the function $(\theta, \sigma, \phi) \longrightarrow d(T, \mathcal{V}_{XISO}(U))$, $U = \hat{U}_S(\theta,\sigma,\phi)$)

((θ_0, σ_0, ϕ_0) in degrees are {86.2, 0., 40.9})

$$U_0 = \hat{U}_S(\theta_0,\sigma_0,\phi_0) = \begin{pmatrix} 0.0498003 & -0.997825 & 0.0431815 \\ 0.753887 & 0.0659144 & 0.65369 \\ -0.655115 & 0 & 0.75553 \end{pmatrix}$$ (A XISO

-minimizer for T. It minimizes the function $U \longrightarrow d(T, \mathcal{V}_{XISO}(U))$)

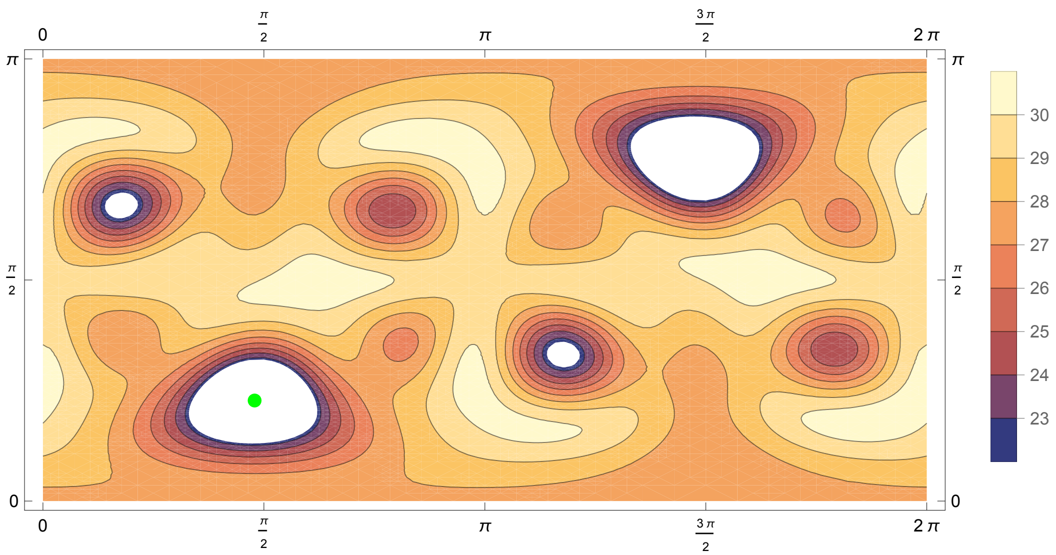
$\beta^{T}_{XISO} = 0.279 = 15.98^{o}$ (Angle between T and $\mathcal{T}_{XISO}$)

$\mathcal{V}_{XISO}(U_0)$ contains a closest map to T in $\mathcal{T}_{XISO}$

$d(T, \mathcal{T}_{XISO}) = 6.505$ (Distance from T to $\mathcal{T}_{XISO}$)

Does the green point look like a correct
GLOBAL minimum? If not, the minimization has failed.

Uniqueness (or not) of the closest XISO map is usually obvious from the contour plot.



$$P(T, \mathcal{V}_{XISO}(U_0)) = \begin{pmatrix} 12.0284 & 0.456493 & 0.397549 & 3.07421 & -2.91164 & -3.27377 \\ 0.456493 & 5.14803 & 0.181413 & 0.182489 & -0.192338 & -0.216259 \\ 0.397549 & 0.181413 & 5.09532 & 0.166797 & -0.150986 & -0.187109 \\ 3.07421 & 0.182489 & 0.166797 & 6.33018 & -1.13784 & -1.41006 \\ -2.91164 & -0.192338 & -0.150986 & -1.13784 & 6.39812 & 1.36334 \\ -3.27377 & -0.216259 & -0.187109 & -1.41006 & 1.36334 & 13. \end{pmatrix}$$

(a closest elastic map to T having symmetry at least XISO)

plot the closest XISO map as a sphere

In[1282]:=

```
TXISO = Closest[Tmat, XISO];
UXISO = UT[Tmat, XISO];
OutputFor[TXISO, XISO]
```

3-19-2024 19:47

NMinimize = $\{6.03787\times10^{-9}, \{\theta\to1.50483, \sigma\to1.41779, \phi\to0.714334\}\}$

(minimizing the function $(\theta, \sigma, \phi) \longrightarrow d(T, \mathcal{V}_{XISO}(U))$, $U = \hat{U}_S(\theta,\sigma,\phi)$)

($(\theta_0, \sigma_0, \phi_0)$ in degrees are $\{86.2, 0., 40.9\}$)

$$U_0 = \hat{U}_S(\theta_0,\sigma_0,\phi_0) = \begin{pmatrix} 0.0498003 & -0.997825 & 0.0431815 \\ 0.753887 & 0.0659144 & 0.65369 \\ -0.655115 & 0 & 0.75553 \end{pmatrix}$$ (A XISO

-minimizer for T. It minimizes the function $U \longrightarrow d(T, \mathcal{V}_{XISO}(U))$)

$\beta^T_{XISO} = 0$ (Angle between T and $\mathcal{T}_{XISO}$. The initial $\beta^T_{XISO} = 2.66\times10^{-10}$ is here chopped to zero, since the chop threshold is set at 0.01°)

Since $\beta^T_{XISO} = 0$, T is assigned symmetry at least XISO.

$\mathcal{V}_{XISO}(U_0)$ contains T

$d(T, \mathcal{T}_{XISO}) = 6.\times10^{-9}$ (Distance from T to $\mathcal{T}_{XISO}$)

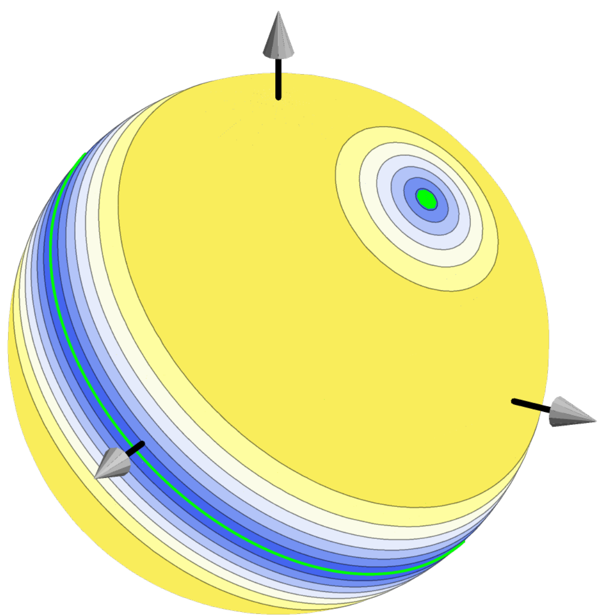
$$T_{XISO} = [\overline{U_0}]^T[T][\overline{U_0}] = \begin{pmatrix} 5.25329 & 0 & 0 & 0 & 0 & 0 \\ 0 & 5.25329 & 0 & 0 & 0 & 0 \\ 0 & 0 & 4.93777 & 0 & 0 & 0 \\ 0 & 0 & 0 & 4.93777 & 0 & 0 \\ 0 & 0 & 0 & 0 & 14.6179 & 3.82705 \\ 0 & 0 & 0 & 0 & 3.82705 & 13. \end{pmatrix}$$

(Should be a XISO-ref matrix for T. If not, consider decreasing ChopT.)

In[1285]:=

```
TwoFold = Graphics3D[
   TwoFoldGraphics[XISO, UXISO, 1.25 AngRadDisk, 3 ShiftToEye, eye, tkns, Green]];
Show[
 {cpMONO[TXISO, contoursMONO[Tmat], MaxForScaling[Tmat], plotpoints, contourstyle],
  TwoFold, Graphics3D[PaxesForLattice]}, options]
```

Out[1286]=



check

In[1287]:=

```
UforNM1 = UXISO;
TXISOx = ProjToVSigOfU[Tmat, UforNM1, XISO];
AngleMatrix[Tmat, TXISO] / Degree
AngleMatrix[Tmat, TXISOx] / Degree
AngleMatrix[TXISO, TXISOx] / Degree
```

Out[1289]=

15.9806

Out[1290]=

15.9806

Out[1291]=

0.

What would our closest-to-XISO map look like for the default U (=Identity)?

In[1292]:=

```
UforNM1 = id;
TXISOx = ProjToVSigOfU[Tmat, UforNM1, XISO];
AngleMatrix[Tmat, TXISO] / Degree
AngleMatrix[Tmat, TXISOx] / Degree
AngleMatrix[TXISO, TXISOx] / Degree
```

Out[1294]=

```
15.9806
```

Out[1295]=

```
27.855
```

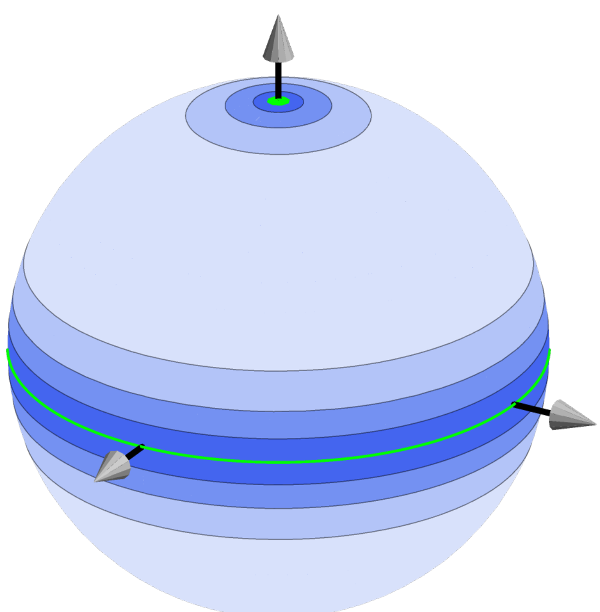
Out[1296]=

```
24.9357
```

In[1297]:=

```
TwoFold = Graphics3D[
   TwoFoldGraphics[XISO, UforNM1, 1.25 AngRadDisk, 3 ShiftToEye, eye, tkns, Green]];
Show[
 {cpMONO[TXISOx, contoursMONO[Tmat], MaxForScaling[Tmat], plotpoints, contourstyle],
  TwoFold, Graphics3D[PaxesForLattice]}, options]
```

Out[1298]=



Let’s how far from ISO it is.

In[1299]:=

```
AngleMatrix[TXISOx, TISO] / Degree
```

Out[1299]=

13.0875

In[1300]:=

```
PrintVoigt[TXISOx]
```

The $[T]_{\mathbb{BB}}$ matrix is $\begin{pmatrix} 9. & 0 & 0 & 0 & 0 & 0 \\ 0 & 9. & 0 & 0 & 0 & 0 \\ 0 & 0 & 5.75 & 0 & 0 & 0 \\ 0 & 0 & 0 & 5.75 & 0 & 0 \\ 0 & 0 & 0 & 0 & 5.5 & 2.12132 \\ 0 & 0 & 0 & 0 & 2.12132 & 13. \end{pmatrix}$

The eigenvalues are: {13.5584, 9., 9., 5.75, 5.75, 4.94158}

The Voigt matrix is $\begin{pmatrix} 9.125 & 3.375 & 2. & 0 & 0 & 0 \\ 3.375 & 9.125 & 2. & 0 & 0 & 0 \\ 2. & 2. & 6. & 0 & 0 & 0 \\ 0 & 0 & 0 & 4.5 & 0 & 0 \\ 0 & 0 & 0 & 0 & 4.5 & 0 \\ 0 & 0 & 0 & 0 & 0 & 2.875 \end{pmatrix}$

3-19-2024 19:47

$\mathcal{V}_{ISO}(U) = \mathcal{V}_{ISO}(I)$, independent of U, so the function U
$\longrightarrow d(T, \mathcal{V}_{ISO}(U)) = d(T, \mathcal{T}_{ISO})$ is constant.

$d(T, \mathcal{T}_{ISO}) = 4.73022$ (Distance from T to $\mathcal{T}_{ISO}$)

$\beta_{ISO}^{\square T} = 0.228 = 13.09°$ (Angle between T and $\mathcal{T}_{ISO}$)

$P(T, \mathcal{V}_{ISO}(I)) = \begin{pmatrix} 7. & 0. & 0. & 0. & 0. & 0. \\ 0. & 7. & 0. & 0. & 0. & 0. \\ 0. & 0. & 7. & 0. & 0. & 0. \\ 0. & 0. & 0. & 7. & 0. & 0. \\ 0. & 0. & 0. & 0. & 7. & 0. \\ 0. & 0. & 0. & 0. & 0. & 13. \end{pmatrix}$

(The closest elastic map to T having symmetry ISO)

Lamé parameters for $P(T, \mathcal{V}_{ISO}(I))$ are $\lambda = 2.$
and $\mu = 3.5$, bulk modulus $\kappa = 4.33333$, Poisson ratio = 0.181818

Among the projected XISO maps, which one is farthest from our target map?

In[1301]:=

```
GetTempAndθ0σ0ϕ0Max[Tmat, XISO]
```

In[1302]:=

```
UXISOmax = UsHat[{θ0, σ0, ϕ0}];
TXISOx = ProjToVSigOfU[Tmat, UXISOmax, XISO];
AngleMatrix[Tmat, TXISO] / Degree
AngleMatrix[Tmat, TXISOx] / Degree
AngleMatrix[TXISO, TXISOx] / Degree
```

Out[1304]=

```
15.9806
```

Out[1305]=

```
30.5468
```

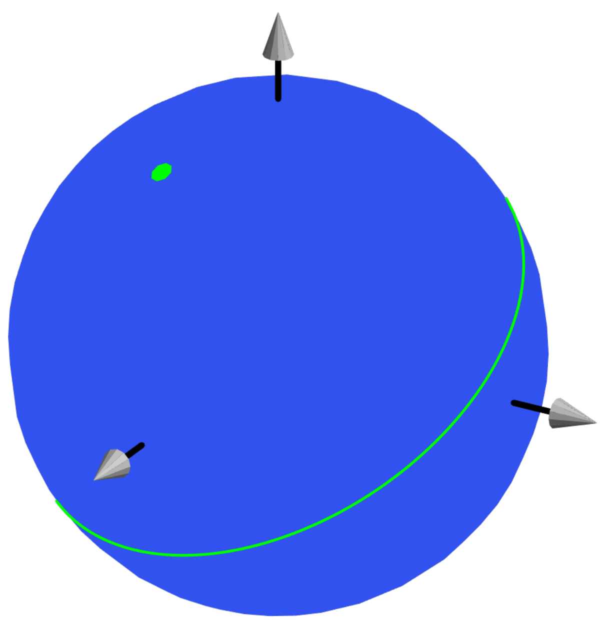
Out[1306]=

```
26.3551
```

In[1307]:=

```
TwoFold = Graphics3D[
   TwoFoldGraphics[XISO, UXISOmax, 1.25 AngRadDisk, 3 ShiftToEye, eye, tkns, Green]];
Show[
 {cpMONO[TXISOx, contoursMONO[Tmat], MaxForScaling[Tmat], plotpoints, contourstyle],
  TwoFold, Graphics3D[PaxesForLattice]}, options]
```

Out[1308]=



That looks ISO on this color scale. Let's how far from ISO it is.

In[1309]:=

```
AngleMatrix[TXISOx, TISO] / Degree
```

Out[1309]=

```
0.597797
```

In[1310]:=

PrintVoigt[TXISOx]

The $[T]_{\mathbb{BB}}$ matrix is

$$\begin{pmatrix} 7.01821 & 0.00448765 & 0.0418971 & -0.0240439 & -0.00134566 & -0.0399718 \\ 0.00448765 & 7.01005 & -0.0111118 & -0.0375894 & 0.00304252 & 0.0903755 \\ 0.0418971 & -0.0111118 & 6.96894 & -0.0021221 & -0.0209194 & -0.0228677 \\ -0.0240439 & -0.0375894 & -0.0021221 & 6.96935 & -0.0190231 & -0.0207948 \\ -0.00134566 & 0.00304252 & -0.0209194 & -0.0190231 & 7.03345 & -0.0733601 \\ -0.0399718 & 0.0903755 & -0.0228677 & -0.0207948 & -0.0733601 & 13. \end{pmatrix}$$

The eigenvalues are: {13.0027, 7.05164, 7.05164, 7.01436, 6.93983, 6.93983}

The Voigt matrix is

$$\begin{pmatrix} 8.98363 & 1.98632 & 2.00365 & -0.0046849 & 0.0565687 & -0.0143136 \\ 1.98632 & 8.92771 & 2.00864 & -0.0287288 & 0.0189792 & -0.0164357 \\ 2.00365 & 2.00864 & 9.09146 & -0.0155415 & 0.0351391 & 0.00274212 \\ -0.0046849 & -0.0287288 & -0.0155415 & 3.5091 & 0.00224383 & 0.0209485 \\ 0.0565687 & 0.0189792 & 0.0351391 & 0.00224383 & 3.50502 & -0.00555589 \\ -0.0143136 & -0.0164357 & 0.00274212 & 0.0209485 & -0.00555589 & 3.48447 \end{pmatrix}$$

3-19-2024 19:48

$\mathcal{V}_{\text{ISO}}(U) = \mathcal{V}_{\text{ISO}}(I)$, independent of U, so the function U
$\longrightarrow d(T, \mathcal{V}_{\text{ISO}}(U)) = d(T, \mathcal{T}_{\text{ISO}})$ is constant.

$d(T, \mathcal{T}_{\text{ISO}}) = 0.212298$ (Distance from T to $\mathcal{T}_{\text{ISO}}$)

$\beta^{\square T}_{\text{ISO}} = 0.0104 = 0.6^{o}$ (Angle between T and $\mathcal{T}_{\text{ISO}}$)

$$P(T,\mathcal{V}_{\text{ISO}}(I)) = \begin{pmatrix} 7. & 0. & 0. & 0. & 0. & 0. \\ 0. & 7. & 0. & 0. & 0. & 0. \\ 0. & 0. & 7. & 0. & 0. & 0. \\ 0. & 0. & 0. & 7. & 0. & 0. \\ 0. & 0. & 0. & 0. & 7. & 0. \\ 0. & 0. & 0. & 0. & 0. & 13. \end{pmatrix}$$

(The closest elastic map to T having symmetry ISO)

Lamé parameters for $P(T,\mathcal{V}_{\text{ISO}}(I))$ are $\lambda = 2.$
and $\mu = 3.5$, bulk modulus $\kappa = 4.33333$, Poisson ratio = 0.181818

replot the map, but with the U=Id (red) and U=Umax (blue) points added (U = Umin in green)

In[1311]:=

```
Show[{ContourPlot[Alpha[Tmat, {θ, ϕ}, XISO] / Degree, {θ, 0, 2 π}, {ϕ, 0, π},
    Contours → Automatic, ColorFunction → (ColorData["M10DefaultDensityGradient"]),
    PlotLegends → Automatic],
   Graphics[
    {PointSize[.015], {Green, Point[TPxyz0to2pi[UT[Tmat, XISO].{0, 0, 1}]]}}],
   Graphics[{PointSize[.015], {Blue, Point[TPxyz0to2pi[UXISOmax.{0, 0, 1}]]}}],
   (* Graphics[{PointSize[.015],{Red,Point[TPxyz0to2pi[UforNM1.{0,0,1}]]}}] *)
   Graphics[{PointSize[.015], {Red, Point[{0, 0}]}}]
  }, FrameTicks → {Range[0, 2 π, π], Range[0, π, π / 2]},
  AspectRatio → Automatic, PlotRange → All, ImageSize → 500]
```

Out[1311]=