Grids, Meshes and Surfaces

⊞— pGrid, pMesh and pSurf

⊟— pGrid

Grided Domains

The domain is given by a box and the number of subintervals or by the discrete points in two equally spaced vectors

$$pR\left(\begin{bmatrix}1 & 5\\3 & 8\end{bmatrix},\begin{bmatrix}2\\3\end{bmatrix}\right)=\begin{bmatrix}\begin{bmatrix}1\\3\\5\end{bmatrix} & \begin{bmatrix}3\\4.6667\\6.3333\\8\end{bmatrix}\end{bmatrix}\qquad U:=pR(1,3,2)=\begin{bmatrix}1\\2\\3\end{bmatrix}\qquad V:=pR(5,8,3)=\begin{bmatrix}5\\6\\7\\8\end{bmatrix}\qquad Box=\begin{bmatrix}x_1 & y_1\\x_2 & y_2\end{bmatrix}$$

$pGrid(f,B,N)$ or $pGrid(f,U,V)$ are used for evaluating the function over it's domain. The version with U, V vectors could be used for avoid discontinuites or another specific domain.

$pTGrid(f,B,N)$ with the same syntax, pTGrid returns the same grid and another with the middle points.

$$f(x,y):=x+0.1\cdot y$$
$$B:=\begin{bmatrix}1 & 5\\3 & 8\end{bmatrix}$$
$$N:=\begin{bmatrix}2 & 3\end{bmatrix}$$
$$G:=pGrid(f,B,N)$$
$$\begin{bmatrix}G & G'\end{bmatrix}:=pTGrid(f,B,N)$$

$$G=\begin{bmatrix}\begin{bmatrix}1.0\\3.0\\1.3\end{bmatrix} & \begin{bmatrix}1.0\\4.7\\1.5\end{bmatrix} & \begin{bmatrix}1.0\\6.3\\1.6\end{bmatrix} & \begin{bmatrix}1.0\\8.0\\1.8\end{bmatrix}\\ \begin{bmatrix}3.0\\3.0\\3.3\end{bmatrix} & \begin{bmatrix}3.0\\4.7\\3.5\end{bmatrix} & \begin{bmatrix}3.0\\6.3\\3.6\end{bmatrix} & \begin{bmatrix}3.0\\8.0\\3.8\end{bmatrix}\\ \begin{bmatrix}5.0\\3.0\\5.3\end{bmatrix} & \begin{bmatrix}5.0\\4.7\\5.5\end{bmatrix} & \begin{bmatrix}5.0\\6.3\\5.6\end{bmatrix} & \begin{bmatrix}5.0\\8.0\\5.8\end{bmatrix}\end{bmatrix}\qquad G'=\begin{bmatrix}\begin{bmatrix}2.0\\3.8\\2.4\end{bmatrix} & \begin{bmatrix}2.0\\5.5\\2.6\end{bmatrix} & \begin{bmatrix}2.0\\7.2\\2.7\end{bmatrix} & \begin{bmatrix}2.0\\8.0\\2.8\end{bmatrix}\\ \begin{bmatrix}4.0\\3.8\\4.4\end{bmatrix} & \begin{bmatrix}4.0\\5.5\\4.6\end{bmatrix} & \begin{bmatrix}4.0\\7.2\\4.7\end{bmatrix} & \begin{bmatrix}4.0\\8.0\\4.8\end{bmatrix}\\ \begin{bmatrix}5.0\\3.8\\5.4\end{bmatrix} & \begin{bmatrix}5.0\\5.5\\5.6\end{bmatrix} & \begin{bmatrix}5.0\\7.2\\5.7\end{bmatrix} & \begin{bmatrix}5.0\\8.0\\5.8\end{bmatrix}\end{bmatrix}$$

Vectorizing mat sorts the matrix values in a [X Y Z] matrix

$$XX:=pGridBox(G)=\begin{bmatrix}1 & 3 & 1.3\\5 & 8 & 5.8\end{bmatrix}$$

$$\vec{G}=\begin{bmatrix}\begin{bmatrix}1.0 & 1.0 & 1.0 & 1.0\\3.0 & 3.0 & 3.0 & 3.0\\5.0 & 5.0 & 5.0 & 5.0\end{bmatrix}\\ \begin{bmatrix}3.0 & 4.7 & 6.3 & 8.0\\3.0 & 4.7 & 6.3 & 8.0\\3.0 & 4.7 & 6.3 & 8.0\end{bmatrix}\\ \begin{bmatrix}1.3 & 1.5 & 1.6 & 1.8\\3.3 & 3.5 & 3.6 & 3.8\\5.3 & 5.5 & 5.6 & 5.8\end{bmatrix}\end{bmatrix}\qquad \vec{G'}=\begin{bmatrix}\begin{bmatrix}2.0 & 2.0 & 2.0 & 2.0\\4.0 & 4.0 & 4.0 & 4.0\\5.0 & 5.0 & 5.0 & 5.0\end{bmatrix}\\ \begin{bmatrix}3.8 & 5.5 & 7.2 & 8.0\\3.8 & 5.5 & 7.2 & 8.0\\3.8 & 5.5 & 7.2 & 8.0\end{bmatrix}\\ \begin{bmatrix}2.4 & 2.6 & 2.7 & 2.8\\4.4 & 4.6 & 4.7 & 4.8\\5.4 & 5.6 & 5.7 & 5.8\end{bmatrix}\end{bmatrix}$$

As utility, f could be the string of the function name or the expression in two unknows. For example

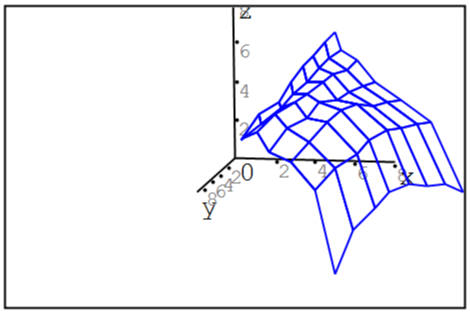
$$f(x,y):=x+\cos(x)\cdot y\qquad pGrid(\text{"f"},X,Y)=pGrid(x+\cos(x)\cdot y,X,Y)$$

For empirical data, pGrid sort it for use with pMesh and other functions.

In some sense, this is the most important example here

$$Y:=\begin{bmatrix}0.8 & 1 & 2.3 & 3.2 & 4.7 & 6.1 & 6.5\end{bmatrix}$$

$$X:=\begin{bmatrix}0\\1.1\\2.4\\3.4\\4.6\\5.6\\6.8\\7.9\end{bmatrix}\qquad Z:=\begin{bmatrix}0.5 & 0.5 & -1.1 & -4.4 & -7.4 & -11.6 & -17.3\\4.3 & 2.9 & 1.8 & -1 & -4.8 & -9.1 & -14\\7 & 6.1 & 4.6 & 1.2 & -1.6 & -6.9 & -11.5\\10.6 & 8.8 & 6.5 & 4 & 0.6 & -4.7 & -9.6\\14.3 & 11.7 & 9.3 & 5.5 & 2.3 & -2.7 & -8\\17 & 14.6 & 11.4 & 7.4 & 3.2 & -1.5 & -7.4\\20.7 & 17.7 & 13.4 & 9.3 & 4.8 & 0 & -6.2\\23.8 & 20 & 15.9 & 11.1 & 6.2 & 1.3 & -6\end{bmatrix}$$



$pMesh(EG)\cdot pView(30°,60°)$

$$EG:=pGrid(Z,X,Y)$$

$$EG=\begin{bmatrix}\begin{bmatrix}0\\0.8\\0.5\end{bmatrix} & \begin{bmatrix}0\\1\\0.5\end{bmatrix} & \begin{bmatrix}0\\2.3\\-1.1\end{bmatrix} & \begin{bmatrix}0\\3.2\\-4.4\end{bmatrix} & \begin{bmatrix}0\\4.7\\-7.4\end{bmatrix} & \begin{bmatrix}0\\6.1\\-11.6\end{bmatrix} & \begin{bmatrix}0\\6.5\\-17.3\end{bmatrix}\\ \begin{bmatrix}1.1\\0.8\\4.3\end{bmatrix} & \begin{bmatrix}1.1\\1\\2.9\end{bmatrix} & \begin{bmatrix}1.1\\2.3\\1.8\end{bmatrix} & \begin{bmatrix}1.1\\3.2\\-1\end{bmatrix} & \begin{bmatrix}1.1\\4.7\\-4.8\end{bmatrix} & \begin{bmatrix}1.1\\6.1\\-9.1\end{bmatrix} & \begin{bmatrix}1.1\\6.5\\-14\end{bmatrix}\\ \begin{bmatrix}2.4\\0.8\\7\end{bmatrix} & \begin{bmatrix}2.4\\1\\6.1\end{bmatrix} & \begin{bmatrix}2.4\\2.3\\4.6\end{bmatrix} & \begin{bmatrix}2.4\\3.2\\1.2\end{bmatrix} & \begin{bmatrix}2.4\\4.7\\-1.6\end{bmatrix} & \begin{bmatrix}2.4\\6.1\\-6.9\end{bmatrix} & \begin{bmatrix}2.4\\6.5\\-11.5\end{bmatrix}\\ \begin{bmatrix}3.4\\0.8\\10.6\end{bmatrix} & \begin{bmatrix}3.4\\1\\8.8\end{bmatrix} & \begin{bmatrix}3.4\\2.3\\6.5\end{bmatrix} & \begin{bmatrix}3.4\\3.2\\4\end{bmatrix} & \begin{bmatrix}3.4\\4.7\\0.6\end{bmatrix} & \begin{bmatrix}3.4\\6.1\\-4.7\end{bmatrix} & \begin{bmatrix}3.4\\6.5\\-9.6\end{bmatrix}\\ \begin{bmatrix}4.6\\0.8\\14.3\end{bmatrix} & \begin{bmatrix}4.6\\1\\11.7\end{bmatrix} & \begin{bmatrix}4.6\\2.3\\9.3\end{bmatrix} & \begin{bmatrix}4.6\\3.2\\5.5\end{bmatrix} & \begin{bmatrix}4.6\\4.7\\2.3\end{bmatrix} & \begin{bmatrix}4.6\\6.1\\-2.7\end{bmatrix} & \begin{bmatrix}4.6\\6.5\\-8\end{bmatrix}\\ \begin{bmatrix}5.6\\0.8\\17\end{bmatrix} & \begin{bmatrix}5.6\\1\\14.6\end{bmatrix} & \begin{bmatrix}5.6\\2.3\\11.4\end{bmatrix} & \begin{bmatrix}5.6\\3.2\\7.4\end{bmatrix} & \begin{bmatrix}5.6\\4.7\\3.2\end{bmatrix} & \begin{bmatrix}5.6\\6.1\\-1.5\end{bmatrix} & \begin{bmatrix}5.6\\6.5\\-7.4\end{bmatrix}\\ \begin{bmatrix}6.8\\0.8\\20.7\end{bmatrix} & \begin{bmatrix}6.8\\1\\17.7\end{bmatrix} & \begin{bmatrix}6.8\\2.3\\13.4\end{bmatrix} & \begin{bmatrix}6.8\\3.2\\9.3\end{bmatrix} & \begin{bmatrix}6.8\\4.7\\4.8\end{bmatrix} & \begin{bmatrix}6.8\\6.1\\0\end{bmatrix} & \begin{bmatrix}6.8\\6.5\\-6.2\end{bmatrix}\\ \begin{bmatrix}7.9\\0.8\\23.8\end{bmatrix} & \begin{bmatrix}7.9\\1\\20\end{bmatrix} & \begin{bmatrix}7.9\\2.3\\15.9\end{bmatrix} & \begin{bmatrix}7.9\\3.2\\11.1\end{bmatrix} & \begin{bmatrix}7.9\\4.7\\6.2\end{bmatrix} & \begin{bmatrix}7.9\\6.1\\1.3\end{bmatrix} & \begin{bmatrix}7.9\\6.5\\-6\end{bmatrix}\end{bmatrix}$$

⊟— Functions

Some functions and Grids for further examples

Matlab logo

$$m(x,y):=3\cdot\frac{(1-x)^2}{e^{x^2+(y+1)^2}}-10\cdot\frac{\frac{x}{5}-x^3-y^5}{e^{x^2+y^2}}-\frac{1}{3\cdot e^{(x+1)^2+y^2}}$$

$$Gm'':=\left(\begin{bmatrix} Gm & Gm' \end{bmatrix}:=pTGrid\left(\text{"m"},\begin{bmatrix}-3 & 3\\ -3 & 3\end{bmatrix},\begin{bmatrix}25\\25\end{bmatrix}\right)\right)$$

Up and down

$$b(x,y):=\frac{8}{e^{(x-0.5)^2+y^2}}-\frac{8}{e^{(x+0.5)^2+y^2}}$$

$$Gb'':=\left(\begin{bmatrix} Gb & Gb' \end{bmatrix}:=pTGrid\left(\text{"b"},\begin{bmatrix}-3 & 3\\ -3 & 3\end{bmatrix},\begin{bmatrix}25\\25\end{bmatrix}\right)\right)$$

Torus

$$\begin{cases}R:=2\\ r:=1.5\end{cases}\qquad torus(u,v):=\begin{bmatrix}(R+r\cdot\cos(u))\cdot\cos(v)\\ (R+r\cdot\cos(u))\cdot\sin(v)\\ r\cdot\sin(u)\end{bmatrix}$$

$$Gt'':=\left(\begin{bmatrix} Gt & Gt' \end{bmatrix}:=pTGrid\left(\text{"torus"},\begin{bmatrix}-60° & 120°\\ 0° & 220°\end{bmatrix},\begin{bmatrix}15\\20\end{bmatrix}\right)\right)$$

Ellipsoid

$$ellip(u,v):=\begin{bmatrix}1.5\cdot\cos(u)\cdot\sin(v)+1\\ 1.5\cdot\sin(u)\cdot\sin(v)+2\\ 2\cdot\cos(v)+0.5\end{bmatrix}$$

$$G\varepsilon'':=\left(\begin{bmatrix} G\varepsilon & G\varepsilon' \end{bmatrix}:=pTGrid\left(\text{"ellip"},\begin{bmatrix}0° & 360°\\ 0° & 180°\end{bmatrix},\begin{bmatrix}20\\20\end{bmatrix}\right)\right)$$

A wave

$$wave(r):=0.5\cdot\cos(4\cdot r)$$
$$wave(r,\theta):=cyl2xyz\left(\begin{bmatrix} r & \theta & wave(r)\end{bmatrix}\right)$$

$$Gw'':=\left(\begin{bmatrix} Gw & Gw' \end{bmatrix}:=pTGrid\left(\text{"wave"},\begin{bmatrix}0 & 5\\ 0° & 270°\end{bmatrix},\begin{bmatrix}20\\30\end{bmatrix}\right)\right)$$

This shows or hide the axis. $pShowAxis:=1$

Some colormaps and viewpoints

$CM_T:=pCMap(\begin{bmatrix}0 & 192 & 32\end{bmatrix},64,1)$ $\quad V_1:=pView2(120°,30°)$

$CM_R:=pCMap(\text{"R"},64,1)$ $\quad V_2:=pView2(30°,60°)$

$CM_{GS}:=pCMap(\text{"GS"},64,1)$ $\quad V_3:=pView(-37.5°,30°)$

$CM_J:=pCMap(\text{"Jet"},64,1)$ $\quad V:=V_1$

$CM:=CM_J$ $\quad V2:=V_{[1..3][1..2]}$

⊟—Mesh

The goal of a mesh is to have an only one matrix with 3 columns which could be manipulated by matrix algebra. But for showing it we can use also some directives, like the line and polygon utilities in XY Plot component. The idea is preserve it as pure numerical procedure, but with the flexibility of call the function pMesh with the function expression too.

Mesh Creation

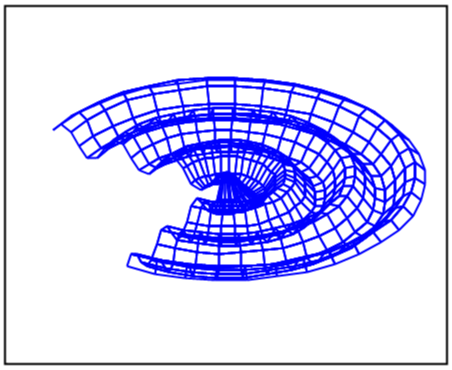
Mesh creation over a grid domain with a geometry given by a two rows matrix.

The left plots are 3D Smath plots, the middle 2D SMath plots and the right are XY Plots

$S_1 := pMesh\left(Gw, \begin{bmatrix} 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 \end{bmatrix}\right)$

$rows\left(S_1\right) = 3580$

Viacheslav original mesh

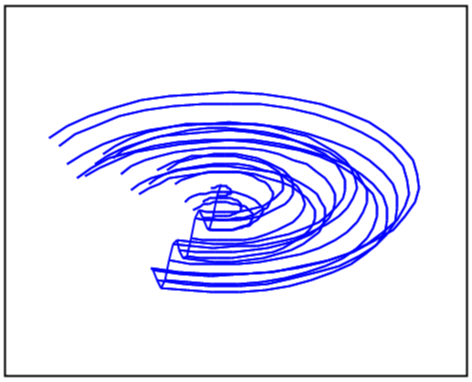


$S_1 \cdot pDiag(2, 2, 1.5) \cdot Y$

$S_2 := pMesh\left(Gw, \begin{bmatrix} 0 & 0 \\ 1 & 1 \end{bmatrix}\right)$

$rows\left(S_2\right) = 1780$

Waterfall along x

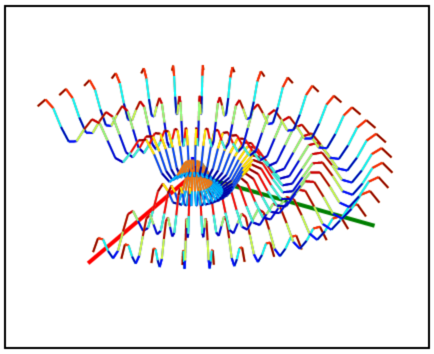


$S_2 \cdot Y2$

$S_3 := pMesh\left(Gw^T, \begin{bmatrix} 0 & 0 \\ 1 & 1 \end{bmatrix}\right)$

$rows\left(S_3\right) = 1770$

Waterfall along y

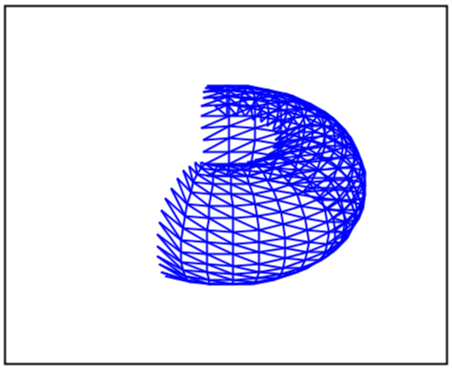


pMesh $\left(S_3, Y, CM\right)$

$S_1 := pMesh\left(Gt, \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}\right)$

$rows\left(S_1\right) = 885$

fake but fast trimesh

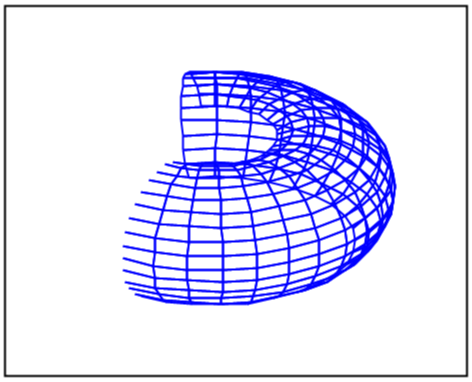


$S_1 \cdot pDiag(2, 2, 2) \cdot Y$

$S_2 := pMesh\left(Gt, \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix}\right)$

$rows\left(S_2\right) = 1185$

something else

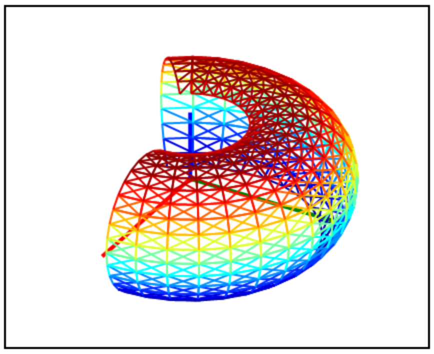


$S_2 \cdot Y2$

$S_3 := pMesh\left(Gt, \begin{bmatrix} 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 \end{bmatrix}\right)$

$rows\left(S_3\right) = 1785$

fake but nice trimesh



pMesh $\left(S_3, Y, CM\right)$

Trimesh

The first argument is now the grid of the domain points and the grid points in it's diagonal $G'' = [\,G \;\; G'\,]$

$S_1 := pMesh\left(Gw'', \begin{bmatrix} 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 \end{bmatrix}\right)$

$rows\left(S_1\right) = 4320$

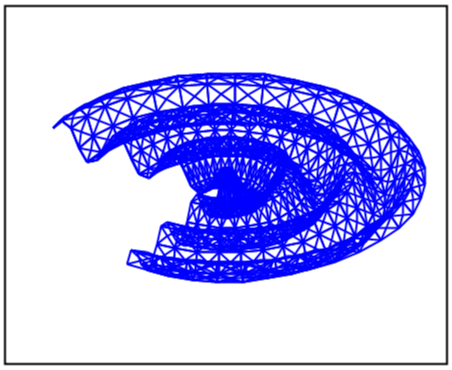
$S_2 := pMesh\left(Gw'', \begin{bmatrix} 0 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix}\right)$

$rows\left(S_2\right) = 3080$

$S_3 := pMesh\left(Gw'', \begin{bmatrix} 0 & 1 \\ 0 & 1 \end{bmatrix}\right)$

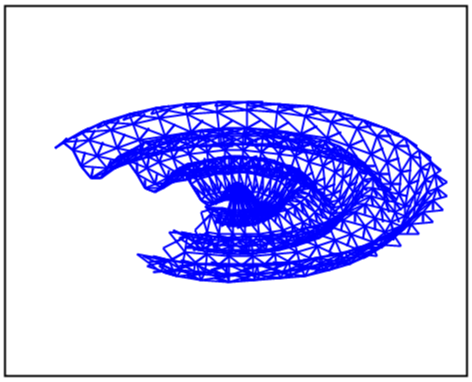
$rows\left(S_3\right) = 2460$

true tri mesh



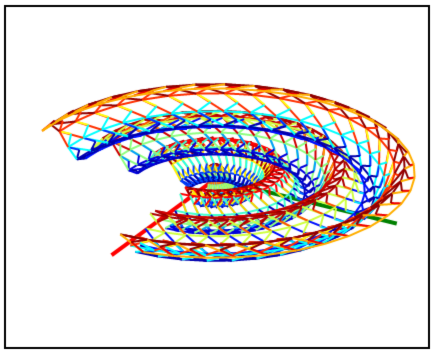
$S_1 \cdot pDiag(2,2,2) \cdot \gamma$

intermediate variation



$S_2 \cdot \gamma 2$

fast variation



$pMesh(S_3, \gamma, CM)$

Other notations

$pMesh(M, \gamma, CM)$	returns a color mesh for XY plots, like in the last example.
$pMesh(G)$	returns the Viacheslav's mesh using $h = \begin{bmatrix} 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 \end{bmatrix}$
$pTMesh(G)$	returns the tri mesh using $h = \begin{bmatrix} 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 \end{bmatrix}$

$pMesh(f, B, N)$ $pMesh(f, U, V)$

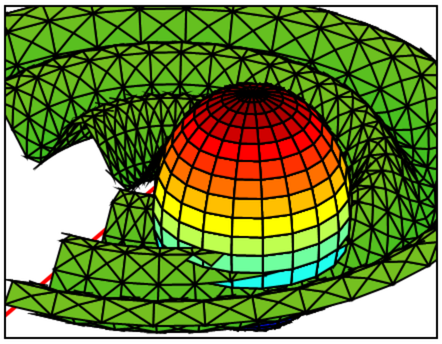
$pTMesh(f, B, N)$ $pTMesh(f, U, V)$

are just as shorthands for avoid griding and then meshing.

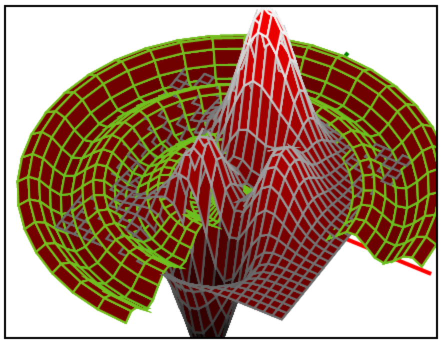
---

pSurf

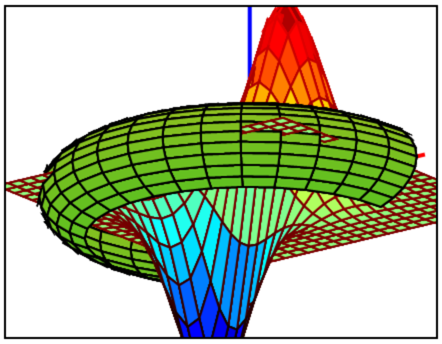
$pSurf(G, \gamma, CL, CS)$ Plots the grid G with the point view γ and colors CL for the lines and CS for surfaces. Also, can take a set of grids.



$pSurf\left(\begin{Bmatrix} G\varepsilon \\ Gw'' \end{Bmatrix}, \gamma_1, \text{"black"}, \begin{Bmatrix} CM_J \\ CM_T \end{Bmatrix}\right)$



$pSurf\left(\begin{Bmatrix} Gm \\ Gw \end{Bmatrix}, \gamma_2, \begin{Bmatrix} CM_{GS} \\ CM_T \end{Bmatrix}, CM_R\right)$



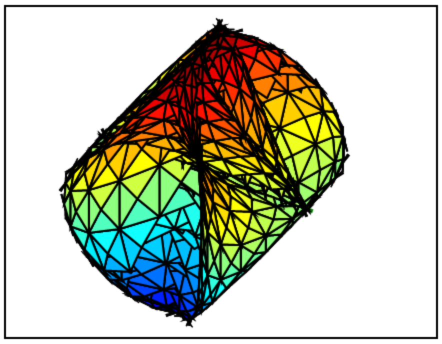
$pSurf\left(\begin{Bmatrix} Gb \\ Gt \end{Bmatrix}, \gamma_3, \begin{Bmatrix} CM_R \\ \text{"black"} \end{Bmatrix}, \begin{Bmatrix} CM_J \\ CM_T \end{Bmatrix}\right)$

$r := 2$

$$f(u,v) := \begin{bmatrix} \sin(2\cdot u)\cdot(\sin(v))^2 \\ \sin(u)\cdot\cos(2\cdot v) \\ \cos(u)\cdot\sin(2\cdot v) \end{bmatrix}$$

$$G := pTGrid\left(f, \begin{bmatrix} 0 & 2\cdot\pi \\ 0 & 2\cdot\pi \end{bmatrix}, \begin{bmatrix} 30 \\ 30 \end{bmatrix}\right)$$

Roman

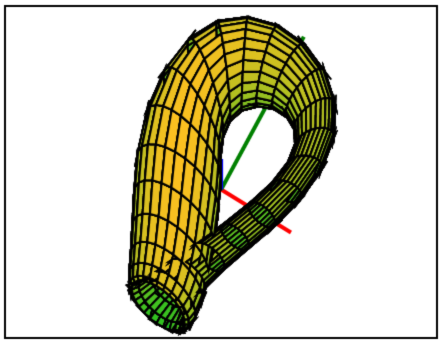


$pSurf(G, \gamma_1, \text{"black"}, CM_J)$

Klein Bottle

$$f(u,v) := \left| \begin{array}{l} bx := 6\cdot\cos(u)\cdot(1+\sin(u)) \\ by := 16\cdot\sin(u) \\ r := 4\cdot(1-0.5\cdot\cos(u)) \\ \text{if } \pi < u \\ \quad \text{stack}(bx + r\cdot\cos(v+\pi), by, r\cdot\sin(v)) \\ \text{else} \\ \quad \begin{bmatrix} bx \\ by \\ 0 \end{bmatrix} + sph2xyz\left(\begin{bmatrix} r \\ u \\ v \end{bmatrix}\right) \end{array} \right.$$

$$G := pGrid\left(f, \begin{bmatrix} 0 & 2\cdot\pi \\ 0 & 2\cdot\pi \end{bmatrix}, \begin{bmatrix} 25 \\ 25 \end{bmatrix}\right)$$



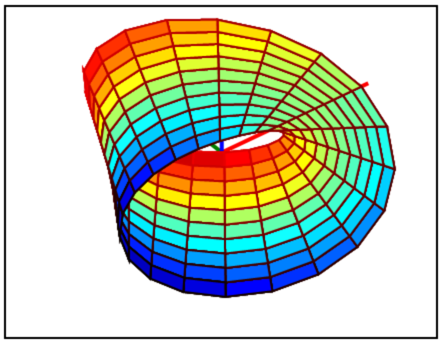
pSurf $(G, \gamma_2, \text{"black"}, CM_T)$

Möbius band

$$r := 2$$

$$f(t,s) := \begin{bmatrix} \left(r + s\cdot\cos\left(\frac{t}{2}\right)\right)\cdot\cos(t) \\ \left(r + s\cdot\cos\left(\frac{t}{2}\right)\right)\cdot\sin(t) \\ s\cdot\sin\left(\frac{t}{2}\right) \end{bmatrix}$$

$$G := pGrid\left(f, \begin{bmatrix} 0 & 2\cdot\pi \\ -1 & 1 \end{bmatrix}, \begin{bmatrix} 20 \\ 10 \end{bmatrix}\right)$$



pSurf $(G, \gamma_3, CM_R, CM_J)$

Algebraic Surface and a pseudosphere

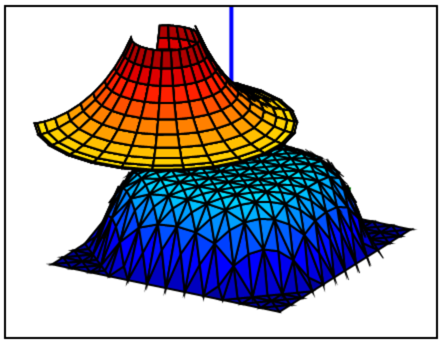
$$f(x,y) := \left| \max\left(\left[-1 \quad x^2\cdot y^2 - \left(x^2+y^2\right)^3\right]\right) \right.$$

$$G_1 := pTGrid\left(f, \begin{bmatrix} -1 & 1 \\ -1 & 1 \end{bmatrix}, \begin{bmatrix} 12 \\ 12 \end{bmatrix}\right)$$

$$a := 1$$

$$g(u,v) := \begin{bmatrix} a\cdot\frac{\cos(v)}{\cosh(u)} + 1 \\ a\cdot\frac{\sin(v)}{\cosh(u)} \\ a\cdot(u - \tanh(u)) + 1 \end{bmatrix}$$

$$G_2 := pGrid\left(\text{"g"}, \begin{bmatrix} 0 & 2 \\ -60° & 220° \end{bmatrix}, \begin{bmatrix} 10 \\ 20 \end{bmatrix}\right)$$



pSurf $\left(\left\{\begin{matrix} G_1 \\ G_2 \end{matrix}, \gamma_1, \text{"black"}, CM_J\right\}\right)$

Ruled Surfaces: Hyperboloid

The ruled surfaces of directrix b and director curve δ have the form

$$f(u,v) := b(u) + v\cdot\delta(u)$$

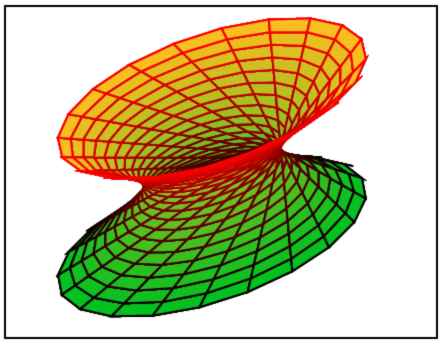
$$[a \;\; b \;\; c] := [1 \;\; 2 \;\; 3]$$

$$f(u,v) := \begin{bmatrix} a\cdot\cos(u) \\ b\cdot\sin(u) \\ 0 \end{bmatrix} + v\cdot\begin{bmatrix} -a\cdot\sin(u) \\ b\cdot\cos(u) \\ c \end{bmatrix}$$

$$U := pR(0°, 360°, 20)$$

$$V := pR(-2, 2, 20)$$

$$GH := pGrid(f, U, V)$$



pSurf $(GH, \gamma_2, CM_R, CM_T)$

Frenet Tube for a Knot

Frenet Tube of the path r with radious R

$$FrenetTube(r\#, R\#, t\#, \theta\#) := \begin{array}{l} \left[r'\# := \overrightarrow{\frac{d}{d\ tt\# := t\#} r\#} \quad r''\# := \overrightarrow{\frac{d}{d\ tt\# := t\#} r'\#} \right] \\ B\# := \frac{r''\# \times r'\#}{\mathrm{norme}(r''\# \times r'\#)} \\ N\# := \frac{B\# \times r'\#}{\mathrm{norme}(r'\#)} \\ r\# + N\# \cdot R\# \cdot \cos(\theta\#) + B\# \cdot R\# \cdot \sin(\theta\#) \end{array}$$

A tube for a knot

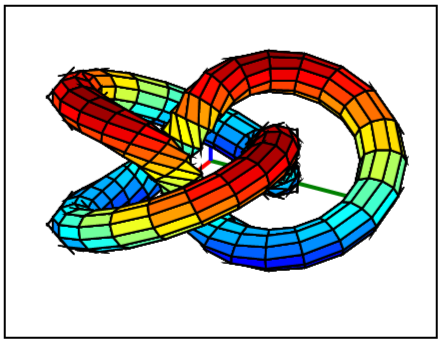
$$k2 := \begin{bmatrix} (2+\cos(3 \cdot t)) \cdot \cos(2 \cdot t) \\ (2+\cos(3 \cdot t)) \cdot \sin(2 \cdot t) \\ \sin(3 \cdot t) \end{bmatrix}$$

$$U := pR(0, 360°, 60)$$

$$V := pR(0, 360°, 10)$$

$$f(t, \theta) := FrenetTube(k2, 0.4, t, \theta)$$

$$GT := pGrid(f, U, V)$$



$pSurf(GT, \gamma_1, \text{"black"}, CM_J)$

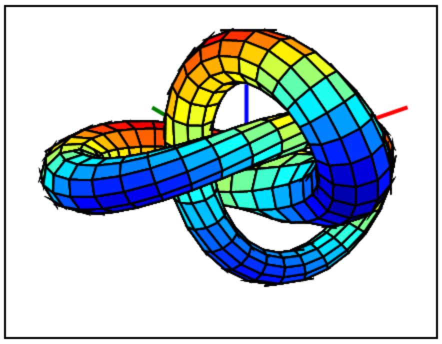
Untwisted version with variable radious

$$k2 := \begin{bmatrix} \sin(t) + 2 \cdot \sin(2 \cdot t) \\ \cos(t) - 2 \cdot \cos(2 \cdot t) \\ -\sin(3 \cdot t) \end{bmatrix}$$

$$R := 0.2 \cdot \cos(t) + 0.5$$

$$f(t, \theta) := FrenetTube(k2, R, t, \theta)$$

$$GT2 := pGrid(f, U, V)$$

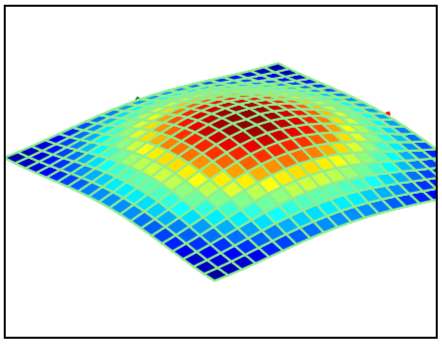


$pSurf(GT2, \gamma_3, \text{"black"}, CM_J)$

Discontinous function

$$f(x, y) := \frac{\sin\left(\sqrt{x^2 + y^2}\right)}{\sqrt{x^2 + y^2}}$$

$$G := pGrid\left(f, 3 \cdot \begin{bmatrix} -1 & 1 \\ -1 & 1 \end{bmatrix}, 20 \cdot \begin{bmatrix} 1 \\ 1 \end{bmatrix}\right)$$



$pSurf(G, \gamma_3, \text{"lightgreen"}, CM_J)$

---

pCMap

pCMap is for creating color maps. It take as arguments a SMath color string, or "Jet", "R", "G" and "B". or an array [r g b] with one of the r, g or b equals zero.

The other arguments are the number of the colors and the transparency as a number between 0 and 1.

$CM_T := pCMap([0\ 192\ 32], 64, 1)$ Makes a colormap looking at the zero value

from $rgb([0\ 192\ 32], 1) = CM_1$

to $rgb([255\ 192\ 32], 1) = CM_{32}$

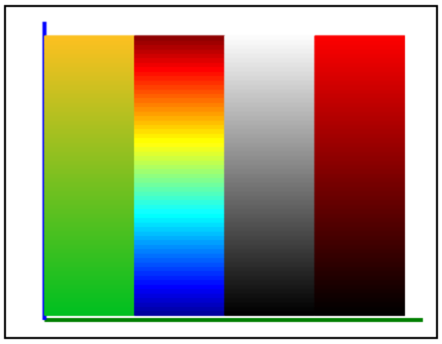
$CM_R := pCMap(\text{"R"}, 64, 1)$ Makes a colormap with a red scale. Similar for "G" and "B"

$pCMap(\text{"B"}, 9, 0) = [0\ 32\ 64\ 96\ 128\ 159\ 191\ 223\ 255]$

CM_{GS} := pCMap ("GS", 64, 1) Makes a colormap with 64 gray values

CM_J := pCMap ("Jet", 64, 1) The Jet color map, from NASA.

```
cm(x, y) := stack(0, x, y)
CM(n) := pGrid("cm", [-1..0] + n, [1..64])
cm := pSurf( {CM(1), CM(2), CM(3), CM(4)}, pView(90°, 0°), 0, {CM_T, CM_J, CM_GS, CM_R} )
```



cm

pView

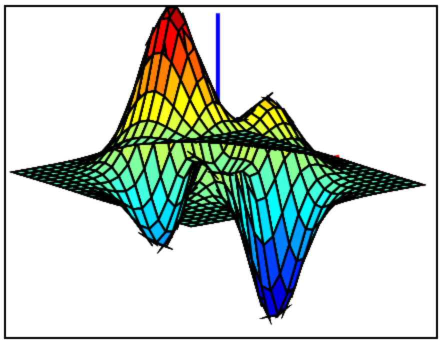
Examples of view

pView(az, el)

returns a rotation matrix for the given azimuth and elevation

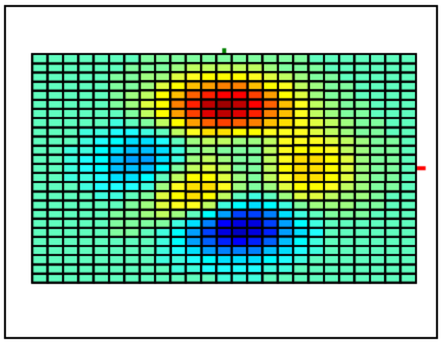
```
CM := pCMap("Jet", 32, 1)
view(az, el) := pSurf(Gm, pView(az, el), "black", CM)
```

Default Matlab view



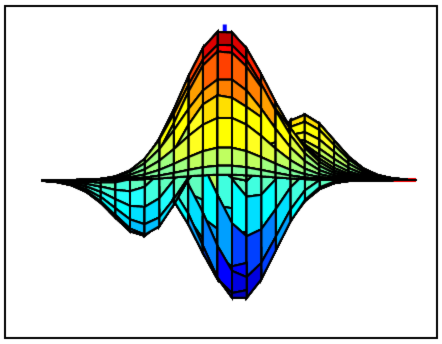
view(-37.5°, 30°)

2D view



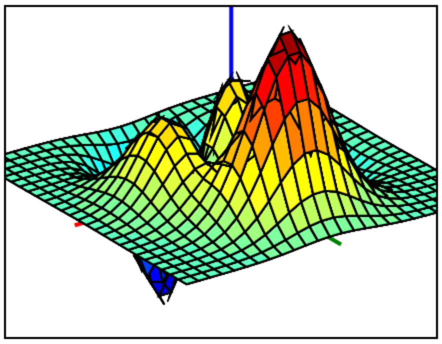
view(0°, 90°)

First column view



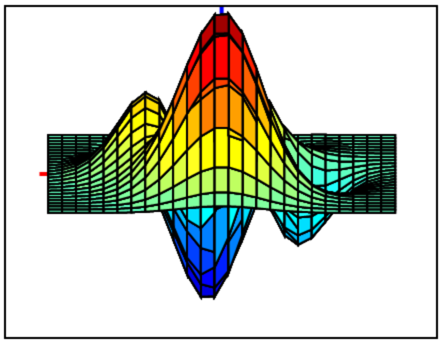
view(0°, 0°)

Fridel usual view



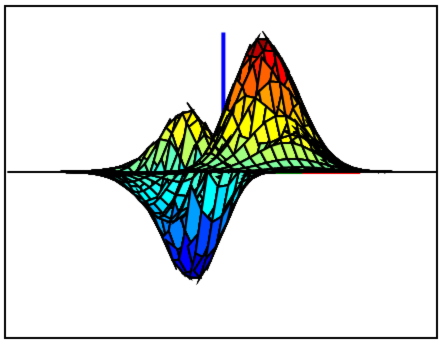
view(145°, 48°)

For az = 180 is behind the matrix



view(180°, 30°)

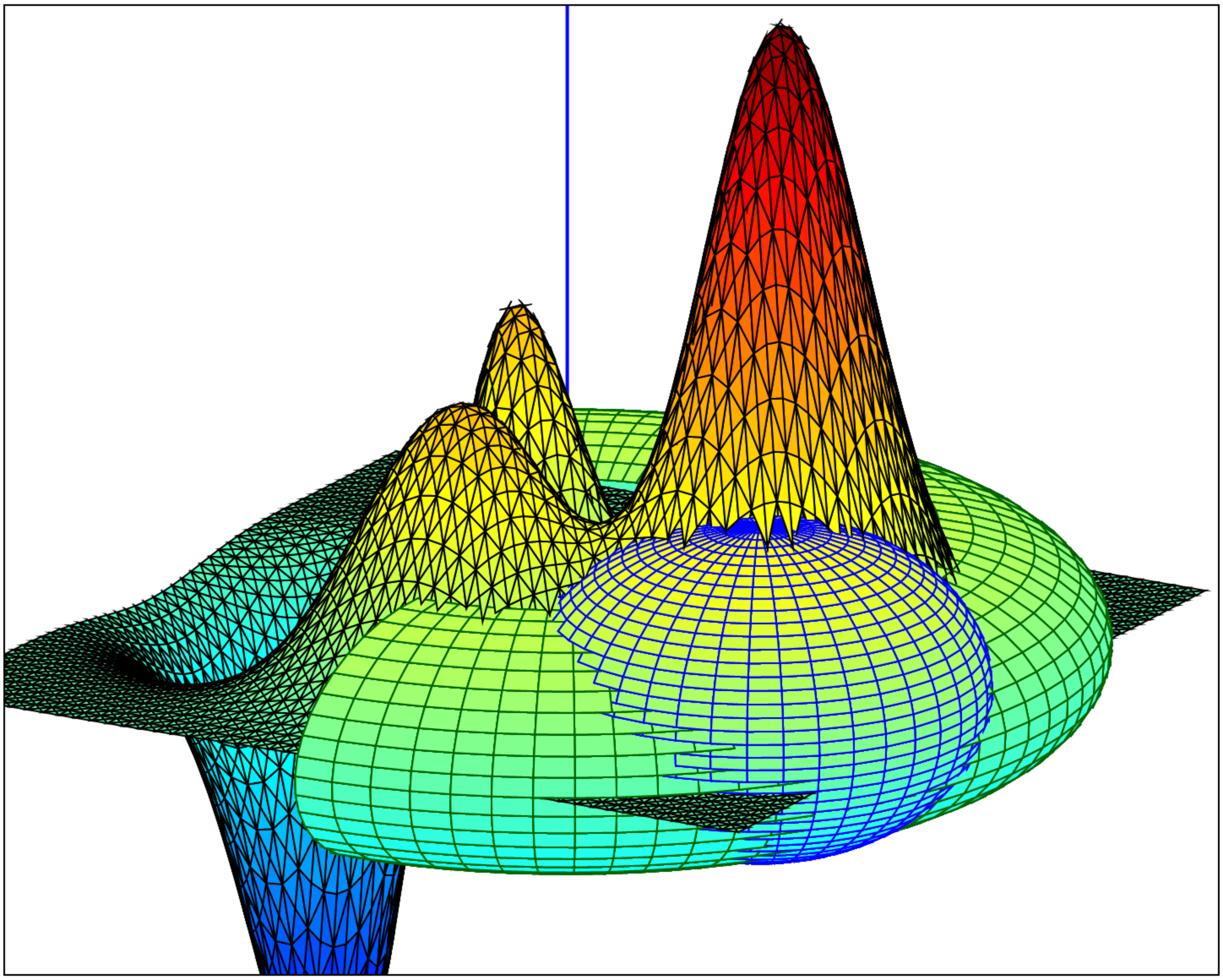
Rotated first column



view(30°, 0°)

Final example

$$\begin{cases} Gm'' := pTGrid\left("m", \begin{bmatrix} -3 & 3 \\ -3 & 3 \end{bmatrix}, \begin{bmatrix} 50 \\ 50 \end{bmatrix}\right) \\ Gt'' := pGrid\left("torus", \begin{bmatrix} -60° & 120° \\ 0° & 220° \end{bmatrix}, \begin{bmatrix} 20 \\ 50 \end{bmatrix}\right) \\ G\varepsilon'' := pGrid\left("ellip", \begin{bmatrix} 0° & 360° \\ 0° & 180° \end{bmatrix}, \begin{bmatrix} 40 \\ 40 \end{bmatrix}\right) \end{cases}$$



$$pSurf\left(\begin{cases} Gm'' \\ Gt'' \\ G\varepsilon'' \end{cases}, Y_1, \begin{cases} "black" \\ "darkgreen" \\ "blue" \end{cases}, CM_J\right)$$

Alvaro

appVersion(4) = "1.0.8348.30405"