第7回：理論形態モデル

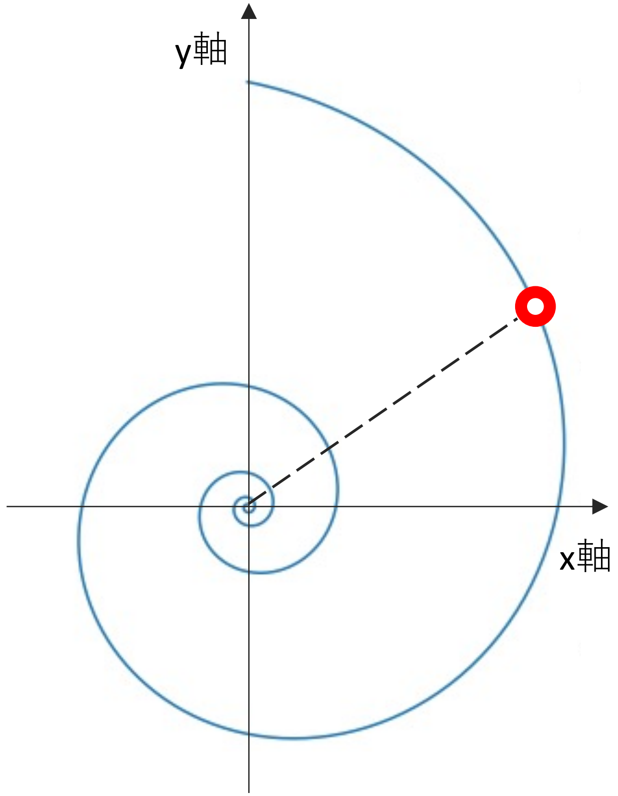
本日の目標

- Raupのモデル
- 回転行列
- 3Dプロット

対数らせんと可視化

対数らせん

$$\frac{dr}{d\theta} = a\theta$$ （aは定数）



$$r(\theta) = r_0 e^{a\theta}$$

指数増殖モデル

$$\frac{dx}{dt} = ax$$

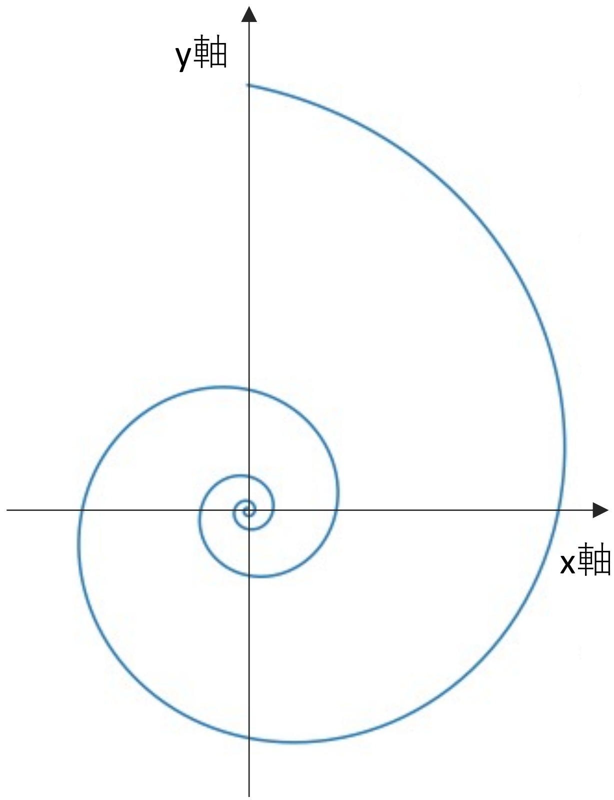
初期条件

$$x(0) = x_0$$

$$x(t) = x_0 e^{at}$$

対数らせん

$$\frac{dr}{d\theta} = a\theta$$ （aは定数）

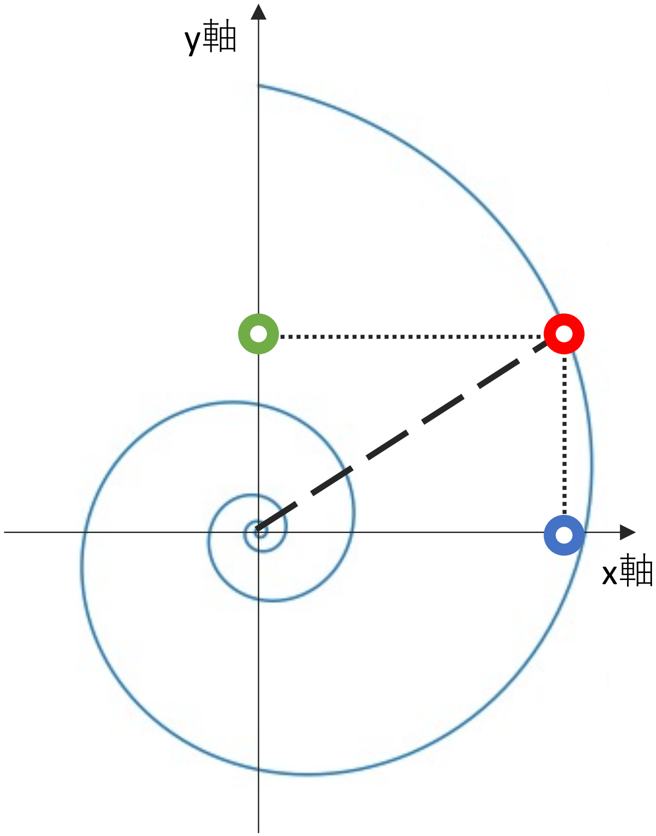


$$r(\theta) = r_0 e^{a\theta}$$

対数らせんで近似できる“巻き”パタン
オウムガイ

唐沢 與希 氏（三笠市立博物館）提供

対数らせん



$$r(\theta) = r_0 e^{a\theta}$$

```
#02-01. 対数らせん
import numpy as np
def logSpiral(a,r0,theta):

    """対数らせん
    対数らせんの座標値を返す関数
    Args:
        a：対数らせんの拡大率
        r0：動径の初期値
        theta：回転角

    Returns：
        x,y：対数螺旋状の座標値

    """
    r=r0*np.exp(a*theta)
    x=r*np.cos(theta)
    y=r*np.sin(theta)

    return(x,y)
```

対数らせんのプロット

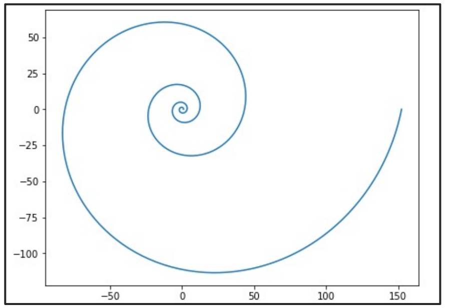
```
#02-02. 対数らせんのプロット
import matplotlib.pyplot as plt

#パラメータの設定
r0=1
a=0.2
theta=np.linspace(0,8*np.pi,1000)

#座標値の計算
x,y=logSpiral(a,r0,theta)

#プロット
plt.figure(figsize=(7,7))
plt.axes().set_aspect('equal')
plt.plot(x,y)
```

出力例



パラメータを変えてプロットしてみよう！

回転角θ〜8πまでプロット

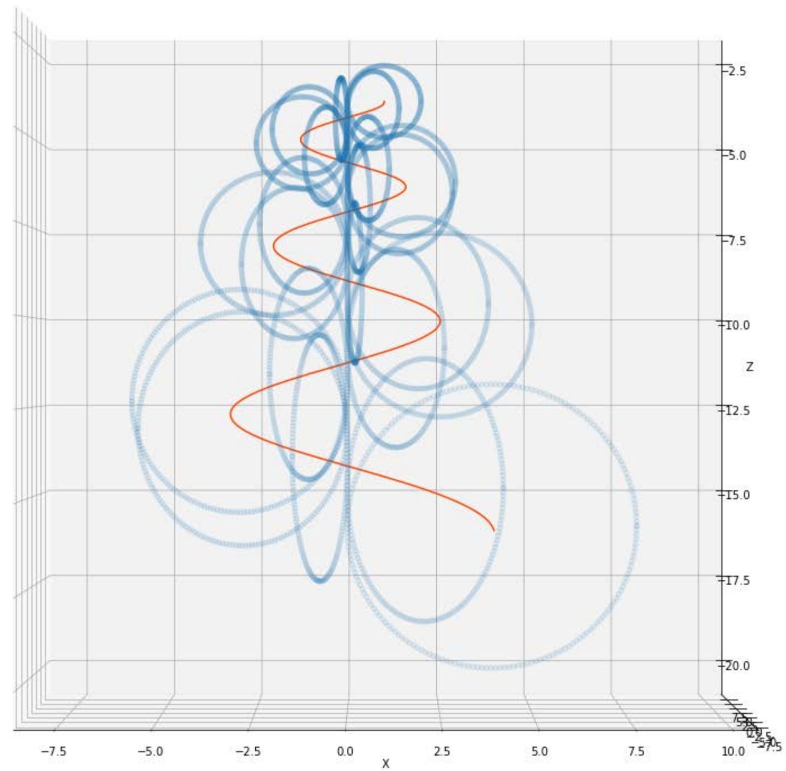
（さっき定義した）logSpiral関数を利用した計算，x，yにはそれぞれ座標値を記録したnumpy配列が代入される

x軸とy軸のスケールを同じにする

- matplotlib.pyplot
  - axes() figure環境にaxes(座標値, 様々な作図関連の要素を格納する入れ物)を追加する.
  - axes.Axes
    - set_aspect() axesのアスペクト比（縦/横）を設定する．自動（‘auto’），同じ（‘equal’），あるいは具体的な値を指定できる.

Raupのモデルと3Dプロット

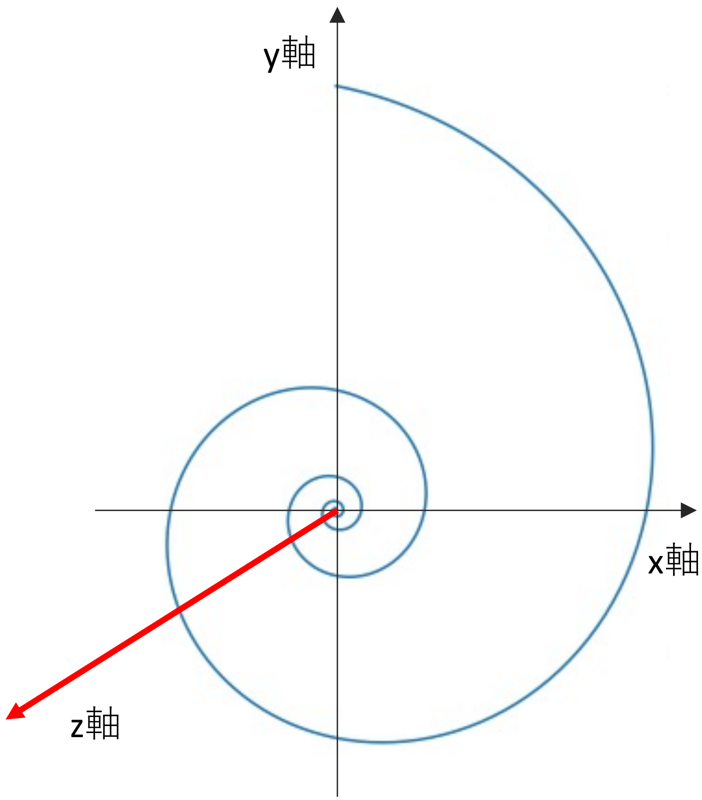
Raupのモデル*

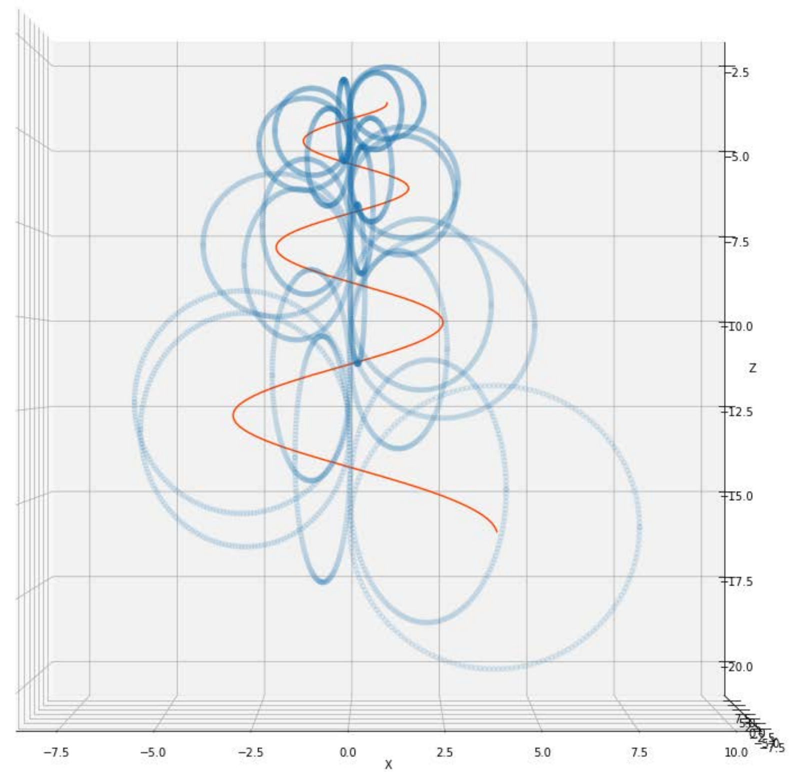


母曲線を巻き軸周りに回転させながら成長させることで
“巻き”のパタンを記述

*Raup(1962,1966),Raup&Michelson(1965)

Raupのモデル



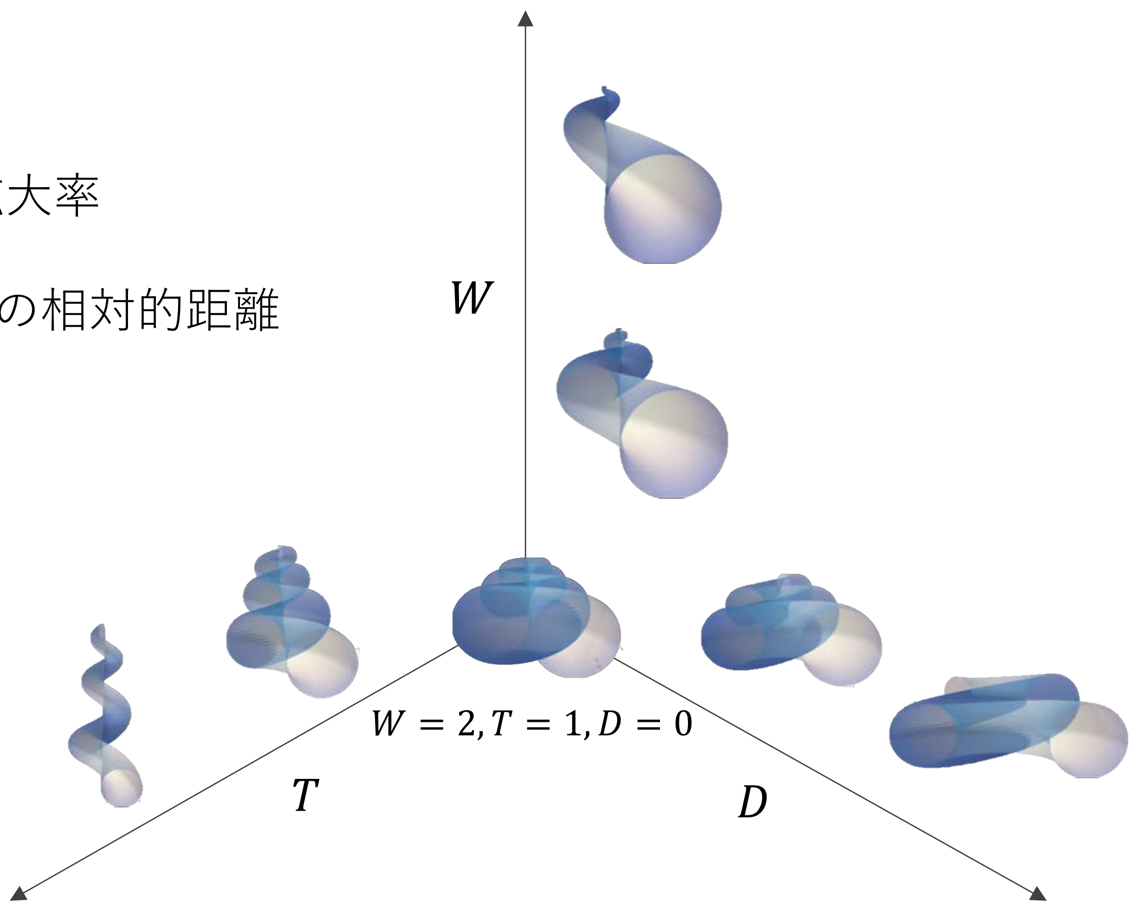


母曲線を巻き軸周りに回転させながら成長させることで
“巻き”のパタンを記述

Raupのモデル

パラメータを変えることで様々な巻パタンが表現できる

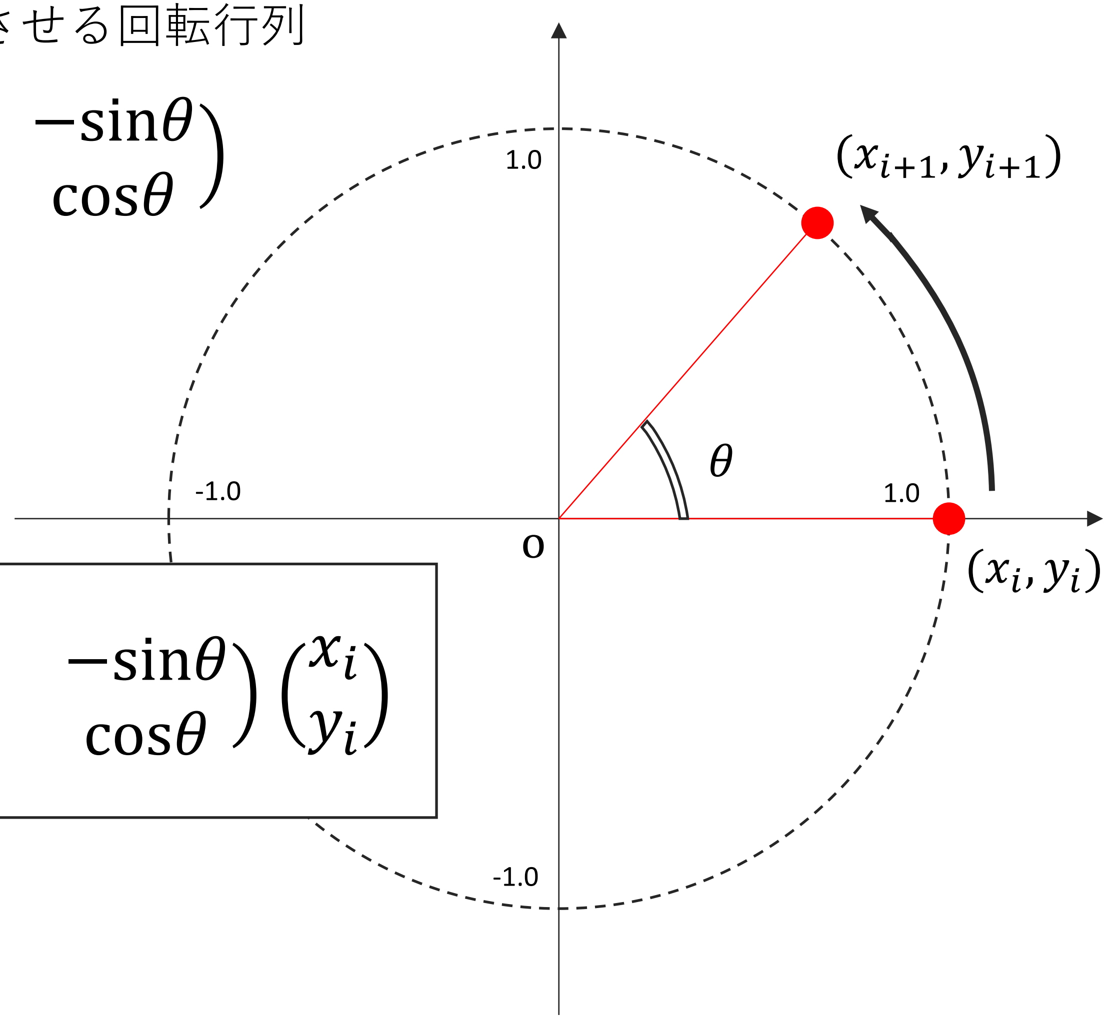
W：螺環の拡大率
T：転移率
D：巻軸からの相対的距離



回転行列 2次元

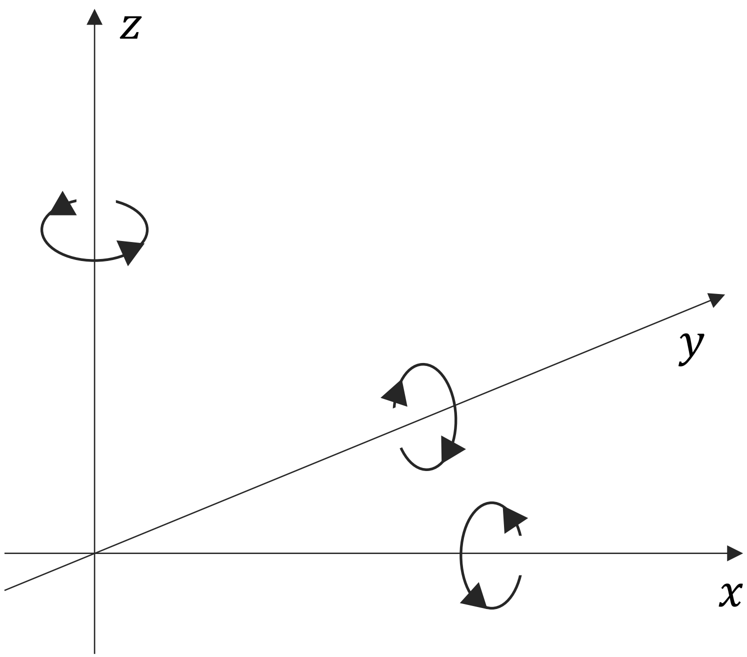
原点周りにθだけ回転させる回転行列

$$\mathbf{R}(\theta) = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix}$$



$$\boxed{\begin{pmatrix} x_{i+1} \\ y_{i+1} \end{pmatrix} = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} x_i \\ y_i \end{pmatrix}}$$

回転行列 3次元



右ねじ

x軸周り

$$\mathbf{R_x}(\theta) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta \\ 0 & \sin\theta & \cos\theta \end{pmatrix}$$

y軸周り

$$\mathbf{R_y}(\theta) = \begin{pmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{pmatrix}$$

z軸周り

$$\mathbf{R_z}(\theta) = \begin{pmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Raupのモデル

θ, ϕでパラメータ表示された母曲線の軌跡(曲面)で巻貝の殻形態を近似する

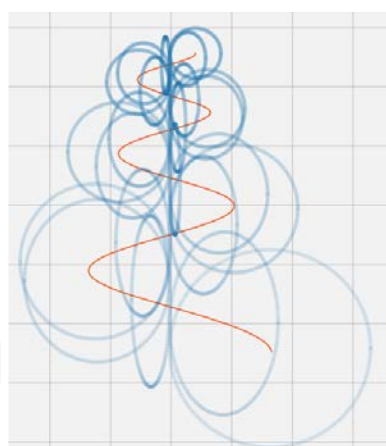
$$
\mathrm{r}(\theta, \phi | W, T, D) = \underbrace{W^{\frac{\theta}{2\pi}}}_{\textbf{管の拡大}} \underbrace{\begin{pmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{pmatrix}}_{\textbf{z軸周りの回転}} \left[\underbrace{\begin{pmatrix} \cos\phi \\ 0 \\ \sin\phi \end{pmatrix}}_{\textbf{母曲線}} + \underbrace{\begin{pmatrix} \dfrac{2D}{1-D} + 1 \\ 0 \\ 2T\left(\dfrac{D}{1-D} + 1\right) \end{pmatrix}}_{\textbf{初期位置}} \right]
$$

ただし，$W > 1, T \in \mathbf{R}, -1 < D < 1$とする

計算すると

$$
= \begin{pmatrix} W^{\frac{\theta}{2\pi}} \left(\dfrac{2D}{1-D} + 1 + \cos\phi \right) \cos\phi \\ W^{\frac{\theta}{2\pi}} \left(\dfrac{2D}{1-D} + 1 + \cos\phi \right) \sin\phi \\ W^{\frac{\theta}{2\pi}} \left(2T \left(\dfrac{D}{1-D} + 1 \right) + \sin\phi \right) \end{pmatrix}
$$

これに基づきプロットする



Raupモデルの定義

```python
#02-03. Raupのモデル
def raupModel(W, T, D, theta, phi):
	"""Raupのモデル
	Raupのモデルに基づき殻表面の座標（x, y, z）を計算する．
	Args:
		W: 螺層拡大率
		T: 転移率（殻の高さ）
		D: 巻軸からの相対的距離（臍の大きさ）
		theta: 成長に伴う回転角
		phi: 殻口に沿った回転角
	Returns:
		x, y, z: 殻表面のx座標，y座標，z座標の
		それぞれの座標値（の配列）
	"""
	w = W**(theta/(2*np.pi))
	x = w * (2*D/(1 - D) + 1 + np.cos(phi))*np.cos(theta)
	y = - w * (2*D/(1 - D) + 1 + np.cos(phi))*np.sin(theta)
	z = - w * (2*T*(D/(1 - D) + 1) + np.sin(phi))
	return (x, y, z)
```

$$\mathbf{R_x}(\theta) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta \\ 0 & \sin\theta & \cos\theta \end{pmatrix}$$

$$\mathbf{R_x}(\pi) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{pmatrix}$$

プロット時に見やすくするためにx軸周りで180°回転させている

Raupモデルのプロット

```
#02-04. Raupのモデルのプロット
import plotly.graph_objs as go

#Raupのモデルに基づく殻の表面座標の計算
W=10**0.2
T=1
D=0.2

thetaRange = np.linspace(0,8*np.pi, 3600 )
phiRange= np.linspace(0, 2*np.pi, 90)
theta, phi = np.meshgrid(thetaRange, phiRange)

x,y,z = raupModel(W,T,D,theta, phi)

fig=go.Figure(go.Surface(x=x,y=y,z=z,showscale=False))

fig.update_layout(scene={"aspectmode":"data"})

fig.show()
```

３次元プロットのためにプロット用パッケージ（plotly）のgraphi_objsをgoという略称でインポート

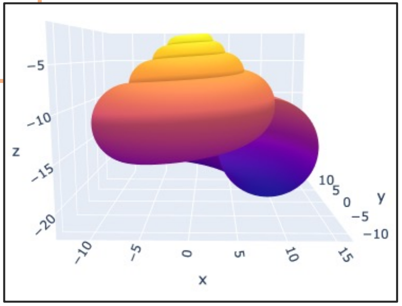
より詳しく知りたい人は公式ドキュメント参照　Plotly https://plotly.com/python/

numpy
- meshgrid(array1d_1,array1d_2)
  一次元配array1d_1,array1d_2に従い，それらのなす格子点の（座標ごとの）配列のリストを生成する．

３次元プロットのための殻両面の３次元座標値を計算

入力した点に張られる表面を作成

プロット時に各軸のスケールを揃える



meshgridの補足

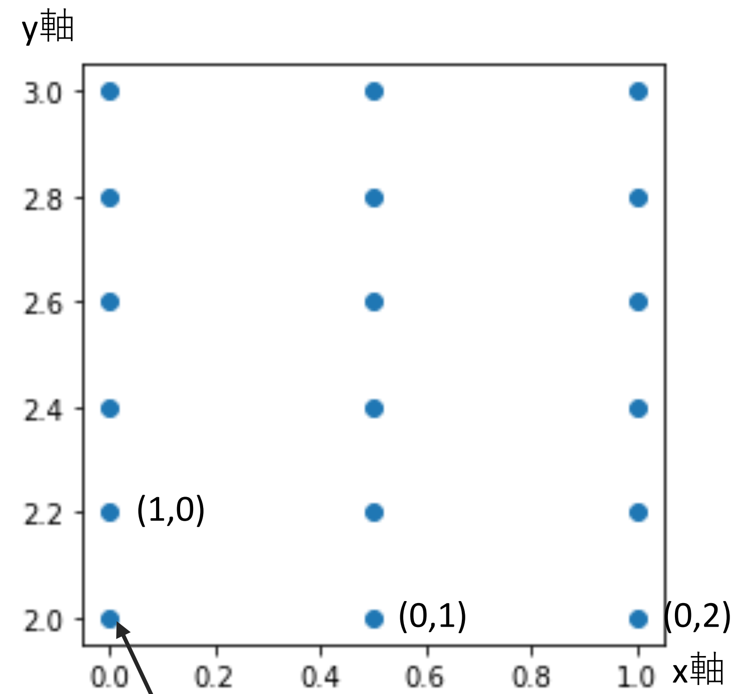
```
#meshgrid
import matplotlib.pyplot as plt
import numpy as np

a=np.linspace(0,1,3)
b=np.linspace(2,3,6)
mesh=np.meshgrid(a,b)
x,y=np.meshgrid(a,b)

print(mesh)

plt.axes().set_aspect("equal")
plt.scatter(x,y)
```

```
#出力
[array([[0. , 0.5, 1. ],
        [0. , 0.5, 1. ],
        [0. , 0.5, 1. ],
        [0. , 0.5, 1. ],
        [0. , 0.5, 1. ],
        [0. , 0.5, 1. ]]),
array([[2. , 2. , 2. ],
        [2.2, 2.2, 2.2],
        [2.4, 2.4, 2.4],
        [2.6, 2.6, 2.6],
        [2.8, 2.8, 2.8],
        [3. , 3. , 3. ]])]
```



座標値として(x[0,0], y[0,0])をもつ点

・plt.scatter(X,Y)：配列（やリスト）XとYを座標値とした散布図を描く

n(>2)個の配列からなる格子点の生成にも利用可能

本日の課題　ノーマル

1. Raupモデルのパラメータを変化させて，様々な“かたち”を描け（4つ程度）

選択 2. 1．で描いた“かたち”を巻貝の形態的なモデルとしよう．すると様々なかたちの中には現実の巻貝に存在する“かたち”と，現実には存在しない“かたち”が現れる．ではなぜ現実にはそうした“かたち”の巻貝が存在するのか，または存在しないのか，その要因について意見を述べよ．

選択 3. 現実の巻貝にはRaupモデルによって描けない“かたち”が存在する．そうした巻貝を探し出し，なぜRaupモデルでは描けないのかを考察せよ

4. 質問，意見，要望等をどうぞ．

課題をノートブック（.ipynbファイル）にまとめて，Moodleにて提出すること
ファイル名は[回数，01～15]_[難易度，ノーマルnかハードh].ipynb. （例）07_h.ipynb

本日の課題　ハード

ノーマル課題2，3のうち，ノーマルで選択しなかった方に取り組む

2. （ノーマルの）1．で描いた“かたち”を巻貝の形態的なモデルとしよう．すると様々なかたちの中には現実の巻貝に存在する“かたち”と，現実には存在しない“かたち”が現れる．ではなぜ現実にはそうした“かたち”の巻貝が存在するのか，または存在しないのか，その要因について意見を述べよ．
3. 現実の巻貝にはRaupモデルによって描けない“かたち”が存在する．そうした巻貝を探し出し，なぜRaupモデルでは描けないのかを考察せよ

課題をノートブック（.ipynbファイル）にまとめて，Moodleにて提出すること
ファイル名は[回数，01～15]_[難易度，ノーマルnかハードh].ipynb.（例）07_h.ipynb

次回予告
第8回：研究を始めるために
6月7日

研究内容
別のスライドへ