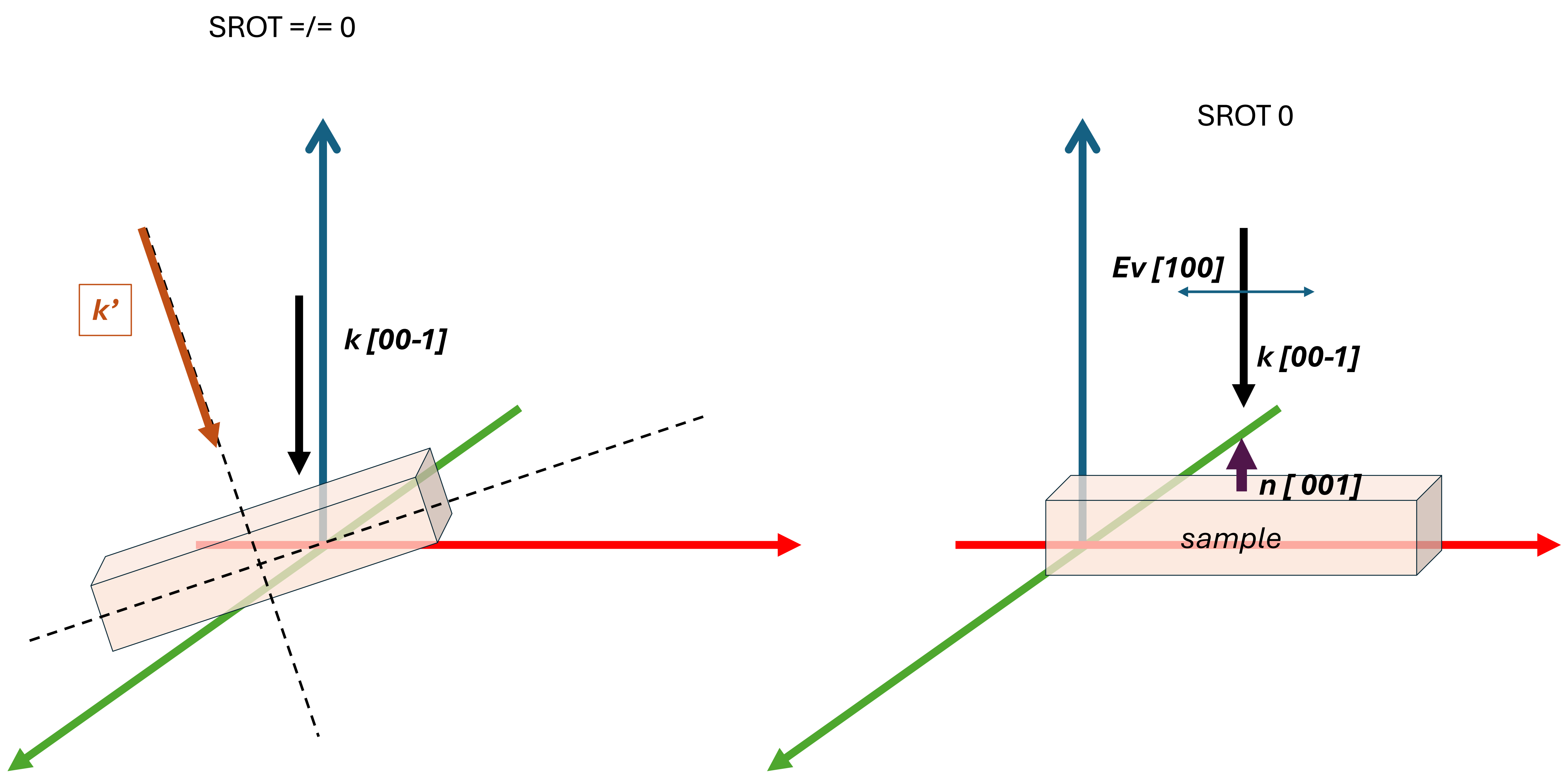
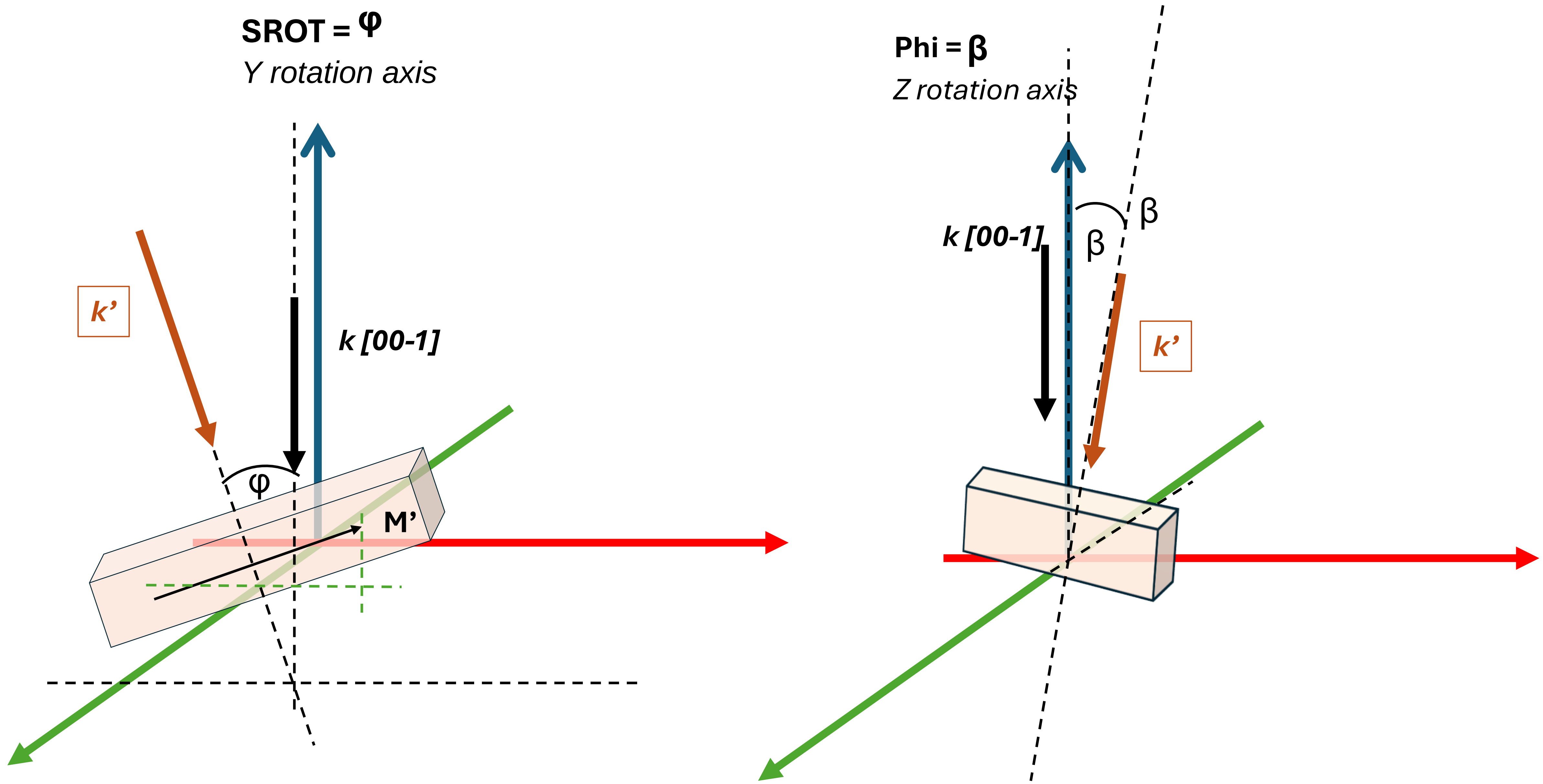
Matrix rotation for crispy

+ visualisation

Simple fast python script from ewa


SROT =/= 0
k'
k [00-1]
SROT 0
Ev [100]
k [00-1]
n [001]
sample



SROT = φ
Y rotation axis
k'
k [00-1]
φ
M'
Phi = β
Z rotation axis
β
β
k [00-1]
k'

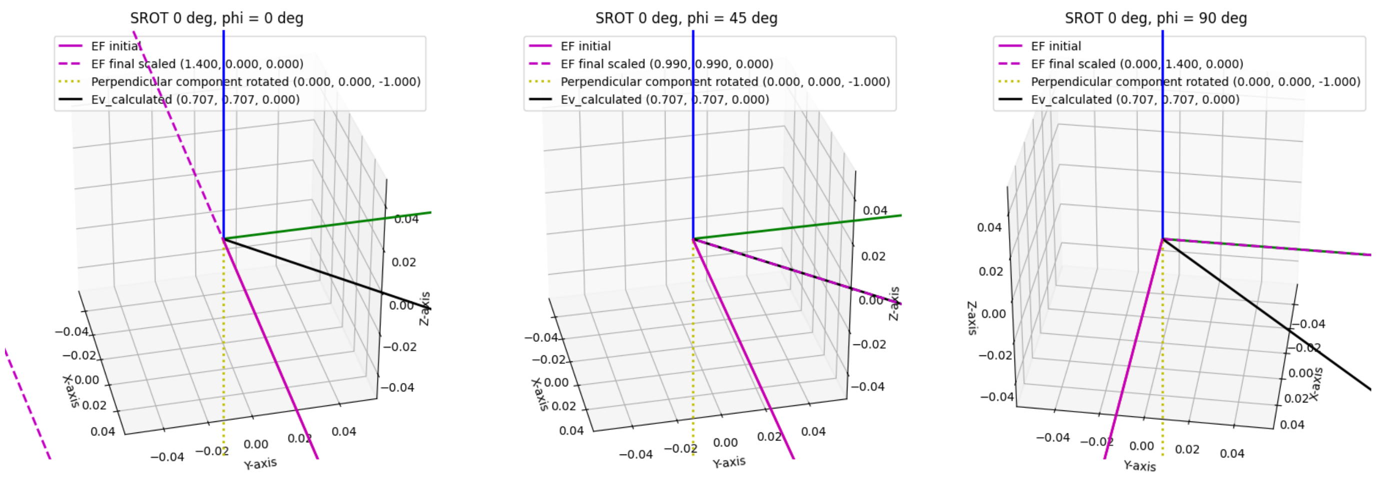
$$R = R_z(\alpha)\, R_y(\beta)\, R_x(\gamma) = \overset{\text{yaw}}{\begin{bmatrix} \cos\alpha & -\sin\alpha & 0 \\ \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix}} \overset{\text{pitch}}{\begin{bmatrix} \cos\beta & 0 & \sin\beta \\ 0 & 1 & 0 \\ -\sin\beta & 0 & \cos\beta \end{bmatrix}} \overset{\text{roll}}{\begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\gamma & -\sin\gamma \\ 0 & \sin\gamma & \cos\gamma \end{bmatrix}}$$

SROT rotation matrix

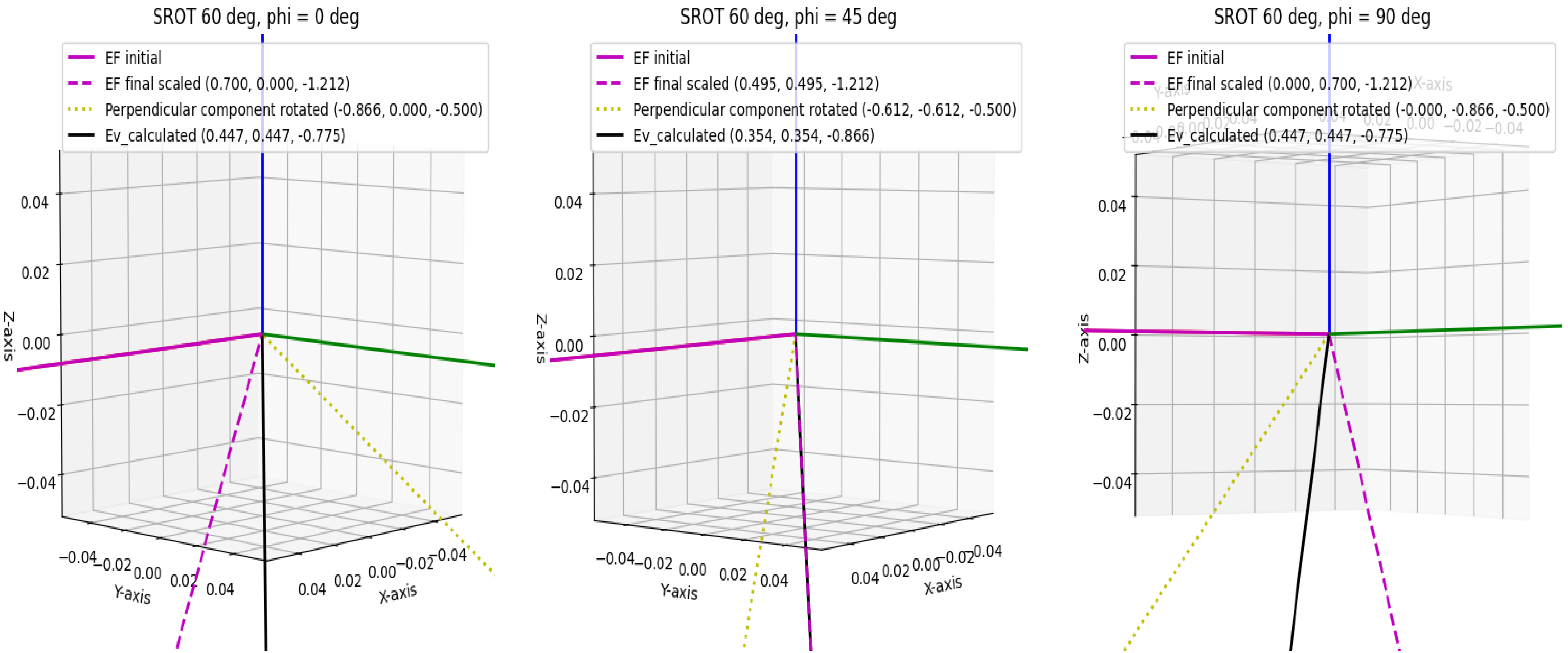
Phi rotation matrix

We want to transform M via rotation matrices: $M' = R(\varphi)R(\beta)\, M$

So for the EF SROT 0 deg, phi 0, 45, 90 deg we may obtain rotated EF:



Examplary srot 60, phi 0, M [100]



Projection of **k [00-1]** onto **rotated EF**

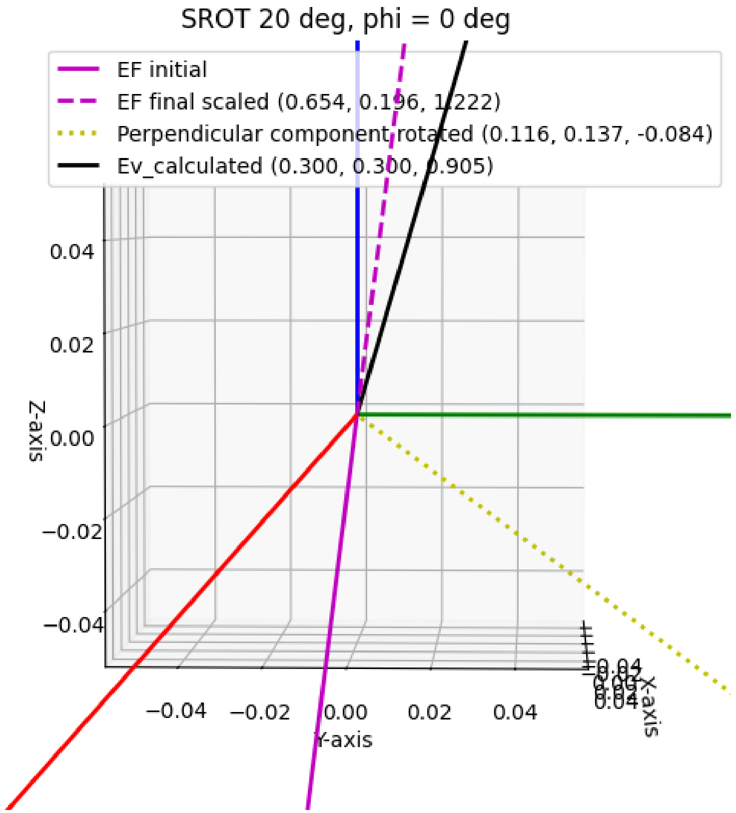
Extraction of **perpendicular k** from the projection (vector substraction simply)
Perpendicular k is the k vector used in crispy

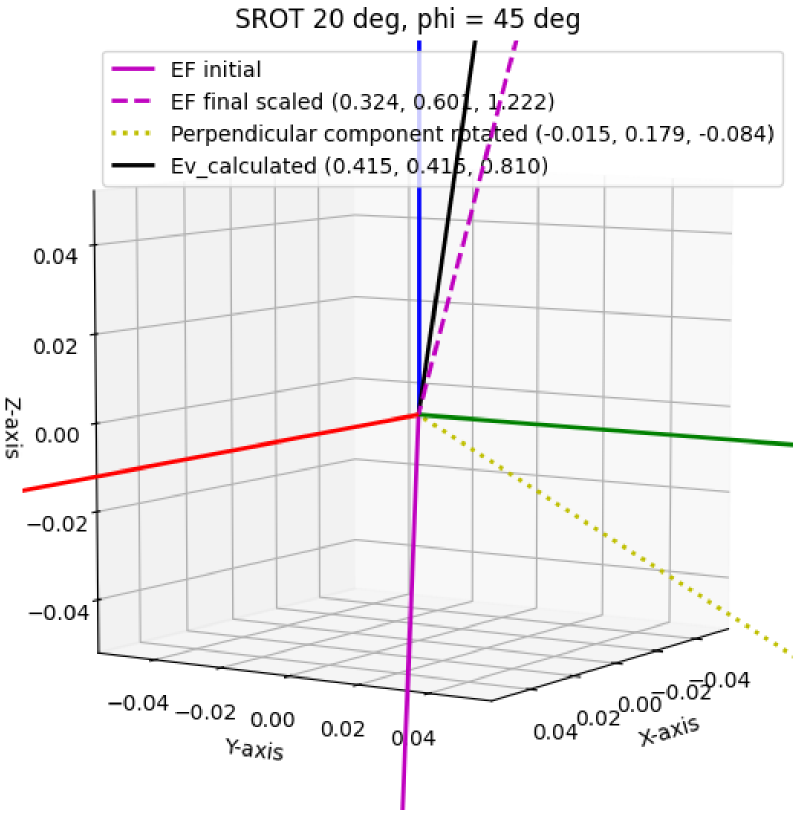
$$proj_k M = \frac{\boldsymbol{M} \cdot \boldsymbol{k}}{|k|^2} \cdot \boldsymbol{k} \longrightarrow \boldsymbol{k}_{\perp} = R(\varphi)R(\beta)\boldsymbol{k} - proj_k M$$

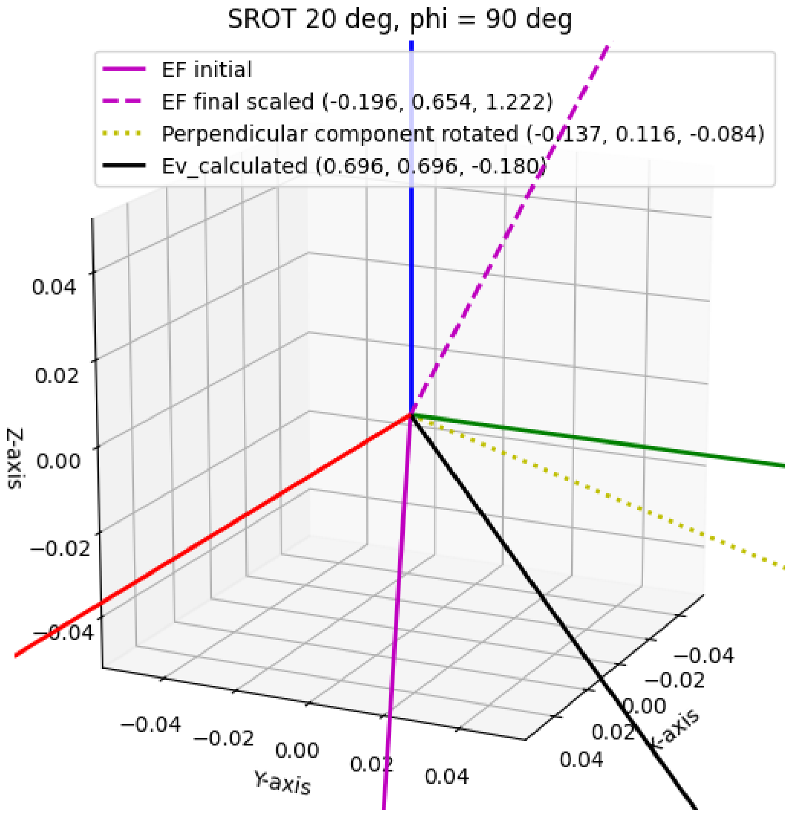
Ev calc: obviously from the vector multiplication, **BUT the cross product of z & k' is an expression not holding true when z of M is not equal to 0, as we need to be perpendiculary oriented in regards to sample's Surface (x, y)**

$$\hat{u}\ is\ [1, -1, 0], \qquad \widehat{E_v} = \boldsymbol{k}_{\perp} \times \hat{u}$$

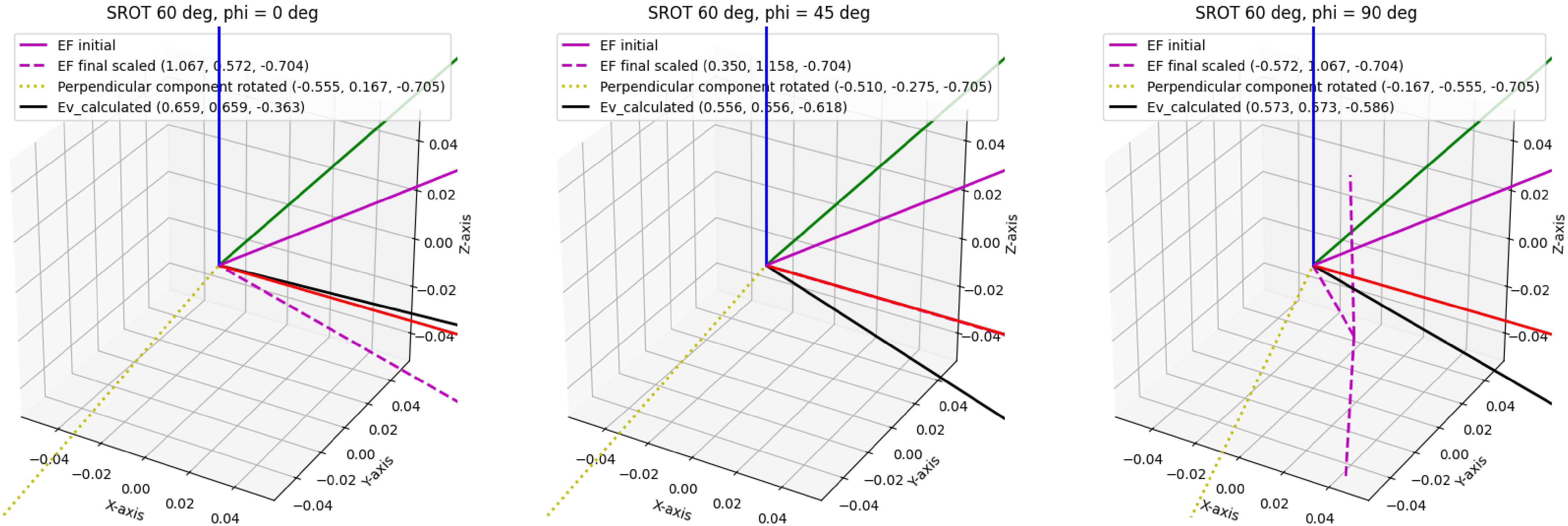
M(117) + SROT 20:


SROT 20 deg, phi = 0 deg
EF initial
EF final scaled (0.654, 0.196, 1.222)
Perpendicular component rotated (0.116, 0.137, -0.084)
Ev_calculated (0.300, 0.300, 0.905)
Z-axis
Y-axis
X-axis
0.04
0.02
0.00
−0.02
−0.04


SROT 20 deg, phi = 45 deg
EF initial
EF final scaled (0.324, 0.601, 1.222)
Perpendicular component rotated (-0.015, 0.179, -0.084)
Ev_calculated (0.415, 0.415, 0.810)
Z-axis
Y-axis
X-axis
0.04
0.02
0.00
−0.02
−0.04

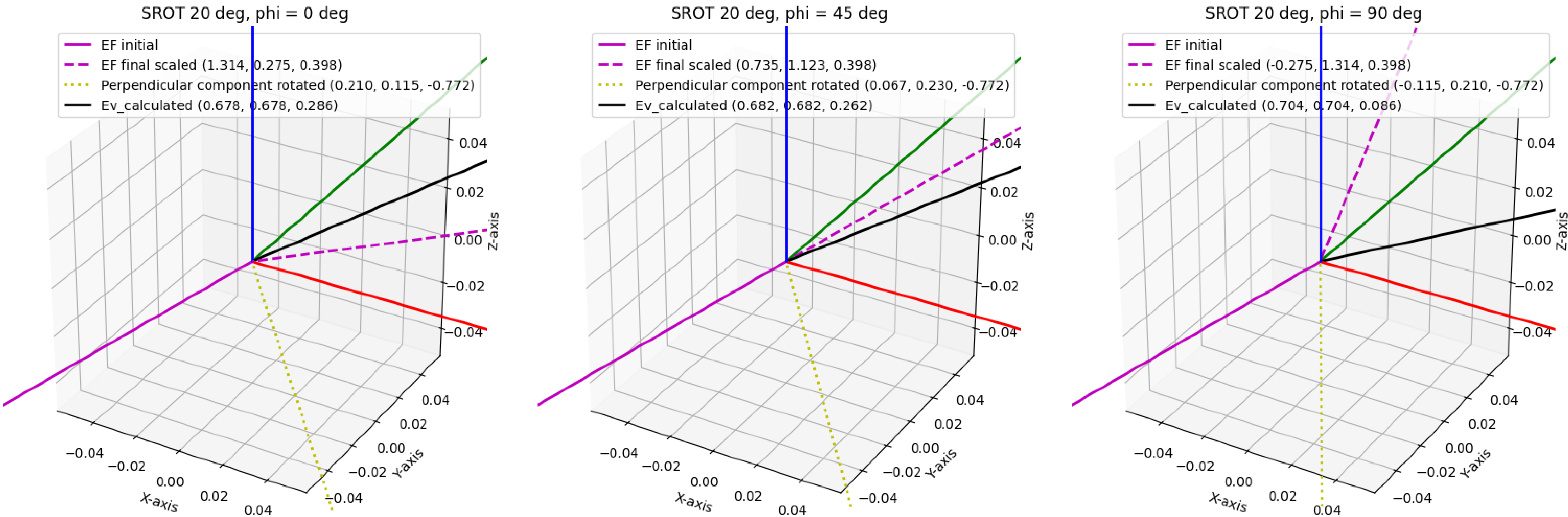

SROT 20 deg, phi = 90 deg
EF initial
EF final scaled (-0.196, 0.654, 1.222)
Perpendicular component rotated (-0.137, 0.116, -0.084)
Ev_calculated (0.696, 0.696, -0.180)
Z-axis
Y-axis
X-axis
0.04
0.02
0.00
−0.02
−0.04

M(211):

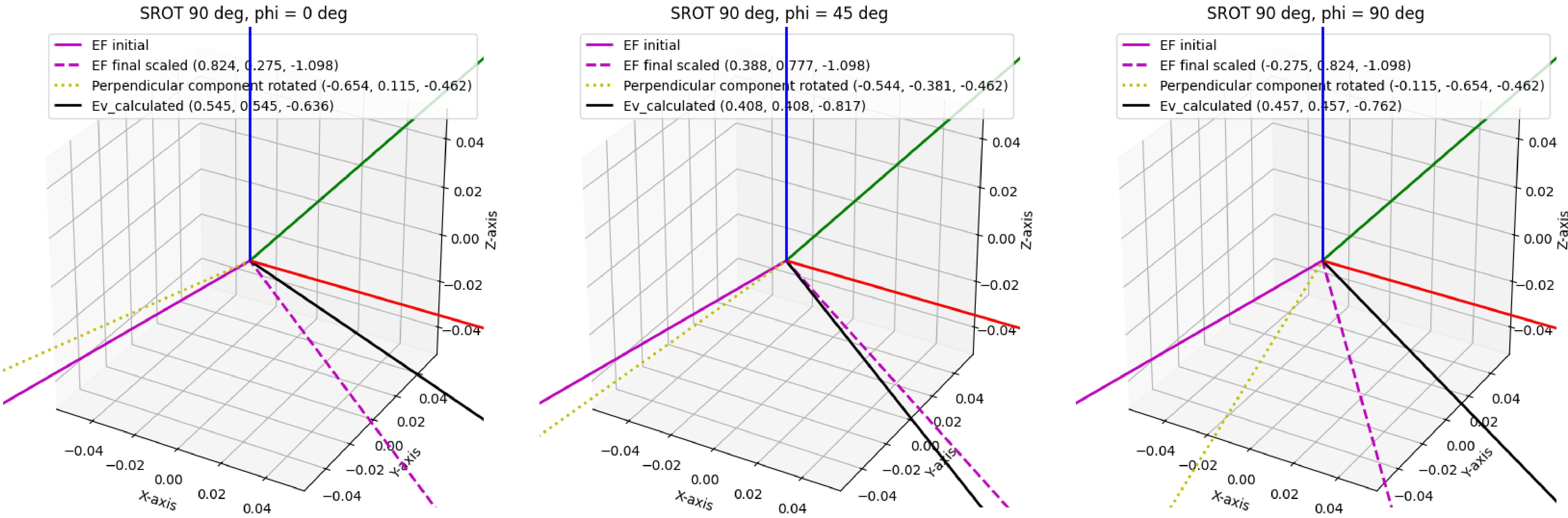


SROT 60 deg, phi = 0 deg
EF initial
EF final scaled (1.067, 0.572, -0.704)
Perpendicular component rotated (-0.555, 0.167, -0.705)
Ev_calculated (0.659, 0.659, -0.363)
SROT 60 deg, phi = 45 deg
EF initial
EF final scaled (0.350, 1.158, -0.704)
Perpendicular component rotated (-0.510, -0.275, -0.705)
Ev_calculated (0.556, 0.556, -0.618)
SROT 60 deg, phi = 90 deg
EF initial
EF final scaled (-0.572, 1.067, -0.704)
Perpendicular component rotated (-0.167, -0.555, -0.705)
Ev_calculated (0.573, 0.573, -0.586)
X-axis
Y-axis
Z-axis

M(413):


SROT 20 deg, phi = 0 deg
EF initial
EF final scaled (1.314, 0.275, 0.398)
Perpendicular component rotated (0.210, 0.115, -0.772)
Ev_calculated (0.678, 0.678, 0.286)
SROT 20 deg, phi = 45 deg
EF initial
EF final scaled (0.735, 1.123, 0.398)
Perpendicular component rotated (0.067, 0.230, -0.772)
Ev_calculated (0.682, 0.682, 0.262)
SROT 20 deg, phi = 90 deg
EF initial
EF final scaled (-0.275, 1.314, 0.398)
Perpendicular component rotated (-0.115, 0.210, -0.772)
Ev_calculated (0.704, 0.704, 0.086)
X-axis
Y-axis
Z-axis

M(413):


SROT 90 deg, phi = 0 deg
EF initial
EF final scaled (0.824, 0.275, -1.098)
Perpendicular component rotated (-0.654, 0.115, -0.462)
Ev_calculated (0.545, 0.545, -0.636)
SROT 90 deg, phi = 45 deg
EF initial
EF final scaled (0.388, 0.777, -1.098)
Perpendicular component rotated (-0.544, -0.381, -0.462)
Ev_calculated (0.408, 0.408, -0.817)
SROT 90 deg, phi = 90 deg
EF initial
EF final scaled (-0.275, 0.824, -1.098)
Perpendicular component rotated (-0.115, -0.654, -0.462)
Ev_calculated (0.457, 0.457, -0.762)
X-axis
Y-axis
Z-axis