In [64]:
```
import discretisedfield as df
import micromagneticmodel as mm
import oommfc as mc
import math
import numpy as np
import matplotlib.pyplot as plt
```

```
env: OOMMFTCL=/home/sellmyer/aullah8/.conda/envs/ubermag-2/opt/oommf/oommf.tcl
```

In [39]:
```
from scipy.spatial import cKDTree
import numpy as np

A = 1.3e-11
Ms = 8.6e5
n = 1
d = 30e-9

p1 = (-30e-9, -30e-9, -30e-9)
p2 = (30e-9, 30e-9, 30e-9)
cell = (2e-9, 2e-9, 2e-9)
region = df.Region(p1=p1, p2=p2)
mesh = df.Mesh(region=region, cell=cell)
```

In [40]:
```
# Create positions to be grain centres, and create a cKDTree to
# perform Voronoi Tesselation
np.random.seed(11)
Ngrains = 72
grain_centres = np.random.uniform(-n*d, n*d, (Ngrains, 3)) # create data between
grain_centres
```

Out[40]:
```
array([[-1.91838187e-08, -2.88314855e-08, -2.20688841e-09],
       [ 1.34960358e-08, -4.78778372e-09, -8.74374110e-10],
       [-2.92331511e-08, -7.57703561e-10,  2.65083991e-08],
       [ 2.10477054e-08,  1.37978682e-08, -2.34758357e-08],
       [ 2.36342502e-08,  2.14292548e-08, -2.00948029e-08],
       [ 7.94004083e-09, -2.87709832e-08, -2.29957639e-08],
       [-1.10179613e-08, -2.05252616e-08,  1.55387753e-08],
       [ 1.90965215e-08, -9.32253054e-09, -1.08720722e-08],
       [-2.33003261e-08, -2.49628114e-08,  1.27635561e-08],
       [ 5.97260378e-09, -2.66595792e-08, -1.21216310e-09],
       [-5.89941116e-09,  2.08787399e-08,  1.30709508e-08],
       [ 6.12384307e-09,  3.14302929e-09,  2.69461439e-08],
       [ 2.92003998e-08, -9.71675702e-09, -1.56075193e-08],
       [ 1.77861452e-08, -2.61788142e-08, -8.12306142e-09],
       [-2.57986318e-08, -1.08379376e-08, -2.57770443e-08],
       [-1.25841800e-08,  1.74060674e-08,  2.43240192e-08],
       [ 1.75572831e-08,  3.70912261e-09,  6.96110310e-09],
       [-8.31098747e-09, -1.98709621e-08, -3.82554398e-09],
       [ 1.39695202e-08, -2.62267426e-08, -2.87560213e-08],
       [ 1.62328840e-08, -1.20028794e-08,  1.20698565e-08],
       [ 1.40800633e-08,  2.59742776e-08, -5.98029402e-09],
       [-8.49372430e-09,  1.83940075e-08,  1.58694658e-08],
       [ 9.15688439e-09,  1.86579944e-08,  8.53289152e-09],
       [ 2.74466432e-08, -9.96753471e-09,  1.42951574e-08],
       [ 2.69899983e-08, -9.93815075e-09,  6.69791980e-09],
       [-8.05978995e-09, -2.30758252e-09, -2.54998804e-08],
       [-2.88393983e-08,  1.55789681e-08, -2.83445013e-09],
       [ 7.37004046e-09,  1.43970911e-08, -4.88019419e-09],
       [-7.94294250e-09, -1.98582714e-08,  1.76323444e-08],
       [ 1.99822445e-08,  1.40527792e-08,  2.25176748e-08],
       [ 8.88338652e-09,  1.17639705e-08, -1.84189152e-08],
       [ 1.60579341e-08, -9.95502634e-09, -3.72244764e-09],
       [-1.08677856e-08,  4.09738014e-09,  9.52154109e-09],
       [ 4.53351246e-09, -1.09087684e-08, -1.69197975e-08],
       [ 2.06963119e-08, -1.18701187e-08, -3.70675975e-09],
       [-2.45151248e-08, -1.19412498e-08, -2.48583116e-08],
       [-9.37720158e-09,  1.25580611e-08,  2.84380953e-08],
       [ 7.51349040e-09, -1.38417876e-08,  9.40252837e-09],
       [ 7.61986915e-09,  1.95391337e-08,  2.02163891e-09],
       [ 2.46672999e-08, -5.06977985e-09, -1.28373198e-08],
       [ 1.14046450e-09,  2.53470143e-08,  1.34346758e-08],
       [-3.65219932e-10, -1.50218835e-10,  8.92333190e-09],
       [-3.13373038e-09,  3.03836826e-09,  2.82705122e-08],
       [ 2.59448231e-08,  2.04411258e-08, -2.23771043e-08],
       [ 3.39126635e-09, -8.01436734e-09,  2.47077291e-08],
       [-1.11792508e-08,  2.58459553e-08, -1.56992339e-08],
       [-4.77655351e-09,  2.75977588e-08,  5.72920143e-09],
       [ 2.59289173e-08,  2.33101751e-09, -1.63177880e-08],
       [ 1.27777752e-08,  1.29440948e-09,  3.28390117e-10],
       [-2.18109292e-08, -2.13048323e-08,  2.57763069e-08],
```

```
[ 2.73634958e-08,  2.79558538e-08, -8.53264552e-09],
[-2.72477877e-08,  3.23563637e-09,  1.41785683e-08],
[ 8.88278839e-09, -2.68240291e-08, -1.60270300e-08],
[ 1.23530555e-08, -2.66949101e-08,  2.07898186e-08],
[ 2.93448697e-08, -1.51320215e-08, -2.67530817e-08],
[ 2.87815985e-08,  1.60329771e-08, -2.74813199e-09],
[-1.30460159e-09,  2.49301802e-08, -8.65019521e-09],
[-2.68092103e-08,  6.83821975e-09, -1.12473144e-08],
[ 1.08935426e-08,  2.68770018e-08, -8.36520730e-09],
[-1.51801108e-08, -2.68228879e-08,  1.98995964e-08],
[-2.52972115e-08,  2.71305288e-08,  1.23688577e-08],
[ 2.46919708e-08,  1.88664523e-08, -1.85567032e-08],
[-1.94457911e-08,  2.76832663e-08, -1.25377215e-08],
[ 1.74368116e-08, -1.97525911e-08, -1.20545862e-08],
[ 1.04094086e-08, -1.50948191e-08,  1.51120301e-08],
[ 2.27553638e-08, -1.07776530e-08, -1.20775274e-08],
[-2.83360404e-08, -4.64040152e-09, -2.60732050e-08],
[ 4.47935663e-09,  1.12761748e-08,  4.09834986e-09],
[ 6.22359284e-09,  5.97900625e-09,  2.41588457e-08],
[ 1.86176435e-09,  2.16835612e-08, -8.88170437e-09],
[-2.15624469e-08, -1.66873745e-08,  1.75969237e-08],
[-1.62189766e-08,  1.63522648e-08,  2.41702100e-08]])
```

In [41]:
```
voronoi_kdtree = cKDTree(grain_centres)


# Generate random anisotropy axes
axes = np.random.uniform(-1, 1, (Ngrains, 3))
axes
```

Out[41]:
```
array([[ 0.08877438, -0.71355468,  0.24699321],
       [-0.24100562,  0.81689589, -0.55272304],
       [-0.53142298, -0.03527781, -0.01289128],
       [-0.62446857, -0.25132886,  0.33634204],
       [-0.0589599 , -0.88056054, -0.16938497],
       [-0.16838089, -0.10479045, -0.95232039],
       [-0.45201522,  0.90082253,  0.76601354],
       [ 0.88141798,  0.51778021, -0.13201587],
       [ 0.87500837, -0.39867785,  0.18643172],
       [ 0.17486845, -0.11849203, -0.49926357],
       [-0.33903578,  0.28583854, -0.43195683],
       [-0.30760857, -0.29179474, -0.35038826],
       [ 0.65706035,  0.09157067, -0.96326021],
       [ 0.48315726,  0.00433957,  0.55778646],
       [-0.01087273, -0.37832449,  0.67857682],
       [-0.65330995, -0.44344686, -0.03170423],
       [ 0.71974237, -0.76957596, -0.32262306],
       [ 0.06448036,  0.35669912, -0.78961186],
       [ 0.51056727,  0.11384219,  0.0342888 ],
       [-0.23825908, -0.37070576,  0.78100924],
       [ 0.98755294, -0.64957027, -0.57699764],
       [ 0.33406819,  0.5291258 ,  0.64593161],
       [-0.39510892, -0.58179939, -0.85486151],
       [-0.24733547,  0.12496947,  0.7451027 ],
       [-0.93391117, -0.82068937,  0.30629025],
       [ 0.34264212, -0.75173702, -0.80726859],
       [-0.94404737,  0.29746815, -0.44028644],
       [-0.5180141 ,  0.32909799, -0.75799907],
       [ 0.07282651, -0.81815032, -0.86850118],
       [ 0.33706635,  0.58322896, -0.07514446],
       [-0.19570788,  0.4030976 , -0.84791724],
       [-0.44343425,  0.81053482,  0.89791161],
       [-0.92320782, -0.99387175,  0.53576824],
       [-0.09624345,  0.52486091,  0.28596612],
       [-0.10406722,  0.41996193,  0.20905448],
       [-0.96856788, -0.06775897,  0.7640093 ],
       [ 0.56056889,  0.30985624,  0.24315745],
       [ 0.63322472, -0.41731508, -0.21178972],
       [ 0.26131253,  0.64191156, -0.35722265],
       [-0.66304377, -0.45548744, -0.12204729],
       [-0.80307404,  0.25667102,  0.79296005],
       [-0.31560438,  0.91275788, -0.16074379],
       [ 0.07148759,  0.95346177,  0.42006699],
       [ 0.01101855,  0.50911603, -0.05338534],
       [ 0.44314796, -0.55530184, -0.8462089 ],
       [ 0.19831445,  0.04736808, -0.07862773],
       [-0.76504514,  0.86908793, -0.77652311],
       [ 0.93492991, -0.36376179, -0.43634009],
       [ 0.87997844,  0.92118734,  0.40132281],
       [ 0.91047008, -0.65004386,  0.71328332],
```

```
[ 0.23639749, -0.4088815 ,  0.19286522],
[-0.78141712,  0.5839796 ,  0.28375465],
[-0.78591847, -0.56068243, -0.41651731],
[-0.0525691 , -0.5387477 , -0.40488056],
[ 0.10296561, -0.70602874,  0.50663868],
[-0.09886328, -0.18579685,  0.14695957],
[-0.68463126,  0.78305233, -0.76968131],
[ 0.09423719,  0.92989042, -0.72161751],
[ 0.39762062,  0.95039122,  0.99165659],
[-0.30452558,  0.25152668,  0.88475192],
[ 0.2490758 ,  0.74563949,  0.39515105],
[-0.92619868, -0.3089444 ,  0.7876545 ],
[-0.06239888, -0.13206014, -0.39015968],
[-0.68052703,  0.33774654, -0.20931807],
[ 0.76421173, -0.91361968,  0.74869866],
[-0.81526135,  0.64703625, -0.08394987],
[ 0.48034988, -0.09822434, -0.68292842],
[ 0.89637932,  0.28495602, -0.57036607],
[ 0.1930285 , -0.58074097, -0.19612614],
[ 0.19044997,  0.22053605, -0.7841247 ],
[-0.61486618, -0.08350408,  0.03231501],
[ 0.71448855,  0.15346395,  0.62303687]])
```

In [42]:
```
# Weight them towards +z - assume grains oriented along field cooled direction
axes[:, 2] += 1.0
axes
```

Out[42]:
```
array([[ 0.08877438, -0.71355468,  1.24699321],
       [-0.24100562,  0.81689589,  0.44727696],
       [-0.53142298, -0.03527781,  0.98710872],
       [-0.62446857, -0.25132886,  1.33634204],
       [-0.0589599 , -0.88056054,  0.83061503],
       [-0.16838089, -0.10479045,  0.04767961],
       [-0.45201522,  0.90082253,  1.76601354],
       [ 0.88141798,  0.51778021,  0.86798413],
       [ 0.87500837, -0.39867785,  1.18643172],
       [ 0.17486845, -0.11849203,  0.50073643],
       [-0.33903578,  0.28583854,  0.56804317],
       [-0.30760857, -0.29179474,  0.64961174],
       [ 0.65706035,  0.09157067,  0.03673979],
       [ 0.48315726,  0.00433957,  1.55778646],
       [-0.01087273, -0.37832449,  1.67857682],
       [-0.65330995, -0.44344686,  0.96829577],
       [ 0.71974237, -0.76957596,  0.67737694],
       [ 0.06448036,  0.35669912,  0.21038814],
       [ 0.51056727,  0.11384219,  1.0342888 ],
       [-0.23825908, -0.37070576,  1.78100924],
       [ 0.98755294, -0.64957027,  0.42300236],
       [ 0.33406819,  0.5291258 ,  1.64593161],
       [-0.39510892, -0.58179939,  0.14513849],
       [-0.24733547,  0.12496947,  1.7451027 ],
       [-0.93391117, -0.82068937,  1.30629025],
       [ 0.34264212, -0.75173702,  0.19273141],
       [-0.94404737,  0.29746815,  0.55971356],
       [-0.5180141 ,  0.32909799,  0.24200093],
       [ 0.07282651, -0.81815032,  0.13149882],
       [ 0.33706635,  0.58322896,  0.92485554],
       [-0.19570788,  0.4030976 ,  0.15208276],
       [-0.44343425,  0.81053482,  1.89791161],
       [-0.92320782, -0.99387175,  1.53576824],
       [-0.09624345,  0.52486091,  1.28596612],
       [-0.10406722,  0.41996193,  1.20905448],
       [-0.96856788, -0.06775897,  1.7640093 ],
       [ 0.56056889,  0.30985624,  1.24315745],
       [ 0.63322472, -0.41731508,  0.78821028],
       [ 0.26131253,  0.64191156,  0.64277735],
       [-0.66304377, -0.45548744,  0.87795271],
       [-0.80307404,  0.25667102,  1.79296005],
       [-0.31560438,  0.91275788,  0.83925621],
       [ 0.07148759,  0.95346177,  1.42006699],
       [ 0.01101855,  0.50911603,  0.94661466],
       [ 0.44314796, -0.55530184,  0.1537911 ],
       [ 0.19831445,  0.04736808,  0.92137227],
       [-0.76504514,  0.86908793,  0.22347689],
       [ 0.93492991, -0.36376179,  0.56365991],
       [ 0.87997844,  0.92118734,  1.40132281],
       [ 0.91047008, -0.65004386,  1.71328332],
       [ 0.23639749, -0.4088815 ,  1.19286522],
       [-0.78141712,  0.5839796 ,  1.28375465],
       [-0.78591847, -0.56068243,  0.58348269],
```

```
[-0.0525691 , -0.5387477 ,  0.59511944],
[ 0.10296561, -0.70602874,  1.50663868],
[-0.09886328, -0.18579685,  1.14695957],
[-0.68463126,  0.78305233,  0.23031869],
[ 0.09423719,  0.92989042,  0.27838249],
[ 0.39762062,  0.95039122,  1.99165659],
[-0.30452558,  0.25152668,  1.88475192],
[ 0.2490758 ,  0.74563949,  1.39515105],
[-0.92619868, -0.3089444 ,  1.7876545 ],
[-0.06239888, -0.13206014,  0.60984032],
[-0.68052703,  0.33774654,  0.79068193],
[ 0.76421173, -0.91361968,  1.74869866],
[-0.81526135,  0.64703625,  0.91605013],
[ 0.48034988, -0.09822434,  0.31707158],
[ 0.89637932,  0.28495602,  0.42963393],
[ 0.1930285 , -0.58074097,  0.80387386],
[ 0.19044997,  0.22053605,  0.2158753 ],
[-0.61486618, -0.08350408,  1.03231501],
[ 0.71448855,  0.15346395,  1.62303687]])
```

In [43]:
```
# Normalise
axes /= np.linalg.norm(axes, axis=1)[:, np.newaxis]
axes
```

Out[43]:
```
array([[ 0.06167218, -0.49571139,  0.86629484],
       [-0.25052353,  0.84915711,  0.46494103],
       [-0.47379814, -0.03145246,  0.88007162],
       [-0.41733963, -0.16796601,  0.89309297],
       [-0.04864943, -0.72657456,  0.68536315],
       [-0.82549068, -0.51373726,  0.23375023],
       [-0.22229831,  0.44301899,  0.86851461],
       [ 0.6572641 ,  0.38610324,  0.64724662],
       [ 0.57296613, -0.26105911,  0.77688993],
       [ 0.32176465, -0.21802989,  0.92137423],
       [-0.47046485,  0.39664541,  0.78824821],
       [-0.39653893, -0.37615328,  0.83741602],
       [ 0.9889127 ,  0.137819  ,  0.05529544],
       [ 0.2962339 ,  0.00266068,  0.95511172],
       [-0.00631872, -0.21986442,  0.97550997],
       [-0.52288969, -0.35492156,  0.77499488],
       [ 0.57457878, -0.61436151,  0.54075795],
       [ 0.15384982,  0.85108237,  0.50198508],
       [ 0.44050571,  0.09822043,  0.89236062],
       [-0.12986153, -0.20205072,  0.9707273 ],
       [ 0.78661805, -0.51740386,  0.33693514],
       [ 0.18971759,  0.30049096,  0.93472585],
       [-0.5502148 , -0.81019337,  0.20211476],
       [-0.13997734,  0.07072538,  0.98762557],
       [-0.5178722 , -0.45508848,  0.72436376],
       [ 0.40390367, -0.88614133,  0.22719018],
       [-0.83022511,  0.2616029 ,  0.49222981],
       [-0.78522275,  0.49885752,  0.36683295],
       [ 0.08754821, -0.98353735,  0.15808097],
       [ 0.29459435,  0.50973929,  0.80831926],
       [-0.4135837 ,  0.85185427,  0.32139201],
       [-0.21007422,  0.38398584,  0.89912385],
       [-0.45055021, -0.48503611,  0.74949615],
       [-0.06912638,  0.37697874,  0.92363877],
       [-0.08104052,  0.32703797,  0.94152992],
       [-0.48102142, -0.03365124,  0.87606277],
       [ 0.40084744,  0.2215697 ,  0.8889478 ],
       [ 0.57892126, -0.38152739,  0.72061573],
       [ 0.27644808,  0.6790919 ,  0.68000783],
       [-0.55682649, -0.38251995,  0.73730777],
       [-0.40532869,  0.12954737,  0.90494538],
       [-0.246665  ,  0.71337864,  0.65593239],
       [ 0.04175787,  0.55694323,  0.82950016],
       [ 0.01025081,  0.47364245,  0.88065757],
       [ 0.60963607, -0.76392552,  0.21156952],
       [ 0.21015397,  0.05019599,  0.97637886],
       [-0.64877452,  0.73700501,  0.18951314],
       [ 0.81248227, -0.31611996,  0.48983745],
       [ 0.46465146,  0.48641083,  0.73993482],
       [ 0.4449609 , -0.31768655,  0.83730822],
       [ 0.18425894, -0.31870081,  0.92977332],
       [-0.484645  ,  0.3621917 ,  0.79620123],
       [-0.69670867, -0.49703923,  0.51725143],
```

```
[-0.06534586, -0.6696887 ,  0.73976142],
[ 0.06176535, -0.42352114,  0.90377812],
[-0.08478047, -0.15933057,  0.98357818],
[-0.64264441,  0.73502953,  0.21619378],
[ 0.09663073,  0.95350878,  0.28545315],
[ 0.1773248 ,  0.42384102,  0.88820871],
[-0.15813822,  0.13061623,  0.97873986],
[ 0.15553684,  0.46561895,  0.87121024],
[-0.45470788, -0.15167313,  0.87763091],
[-0.09950586, -0.21059285,  0.97249639],
[-0.62062138,  0.30801528,  0.72107953],
[ 0.36119047, -0.43180536,  0.82648991],
[-0.58798843,  0.46665996,  0.66068001],
[ 0.82268288, -0.1682263 ,  0.54304033],
[ 0.86685384,  0.27556997,  0.41548239],
[ 0.19105804, -0.57481267,  0.79566778],
[ 0.52517297,  0.60813648,  0.59528428],
[-0.51049402, -0.06932945,  0.85708184],
[ 0.40140479,  0.08621715,  0.91183375]])
```

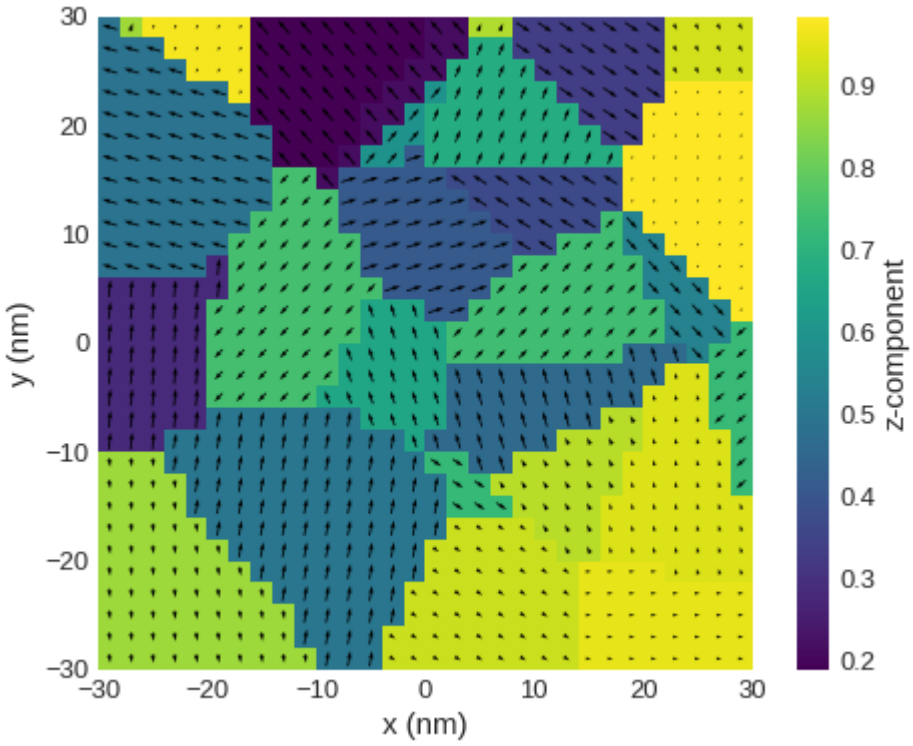
In [45]:
```
Ku = 1e5
# Generate a normal distribution of anisotropy strengths:
strengths = np.random.normal(Ku, Ku*0.2, Ngrains)
strengths
```

Out[45]:
```
array([125416.60244588, 113264.75803076,  62021.16371098,  80310.16009458,
        77285.48570018,  81800.71185825,  58395.35329679,  58965.43262684,
        85124.17441533,  88040.02267893, 110326.54604098,  93504.82797555,
       128428.80227099,  63461.893785  ,  72165.67034941, 104986.23044733,
        94518.62155559,  80203.44829353, 115357.12499158, 121737.5348981 ,
       126174.22357484,  91985.06567302,  99816.54921428, 104514.02768573,
       101105.16438771, 120365.3468904 ,  90136.12029707, 107359.4270245 ,
        55745.75217358, 115057.9921469 , 132748.78824114,  99424.4505686 ,
       120700.52608749,  66716.8558198 ,  93295.49099093,  69118.89100413,
       106339.70993317, 110415.87622442,  72545.88424885,  83027.66977328,
        88255.1419409 , 148417.22572717,  98502.41507686, 102373.91277767,
       107183.35671148, 105924.62185986,  95187.8367293 ,  73346.02711371,
        86380.54698092, 132659.84558423, 105095.0238786 ,  92524.38437615,
        72908.57995578,  98851.83198226, 100603.01957936,  86916.86094844,
       117808.50920675,  88194.16059653,  96603.99335486, 116827.25073186,
       104458.51068638, 122431.23081276,  95389.33619159,  91326.45326275,
       109339.26917193, 104272.37950638, 123694.58317776, 108105.80631909,
        99460.25693344, 108977.47574679, 113094.47182588,  77573.21933547])
```

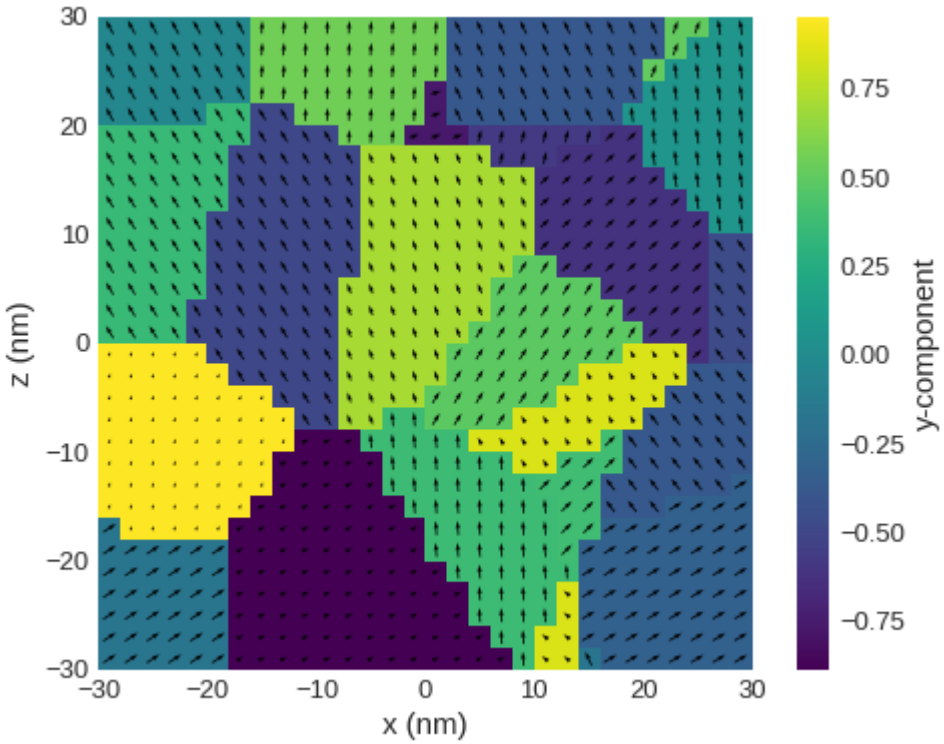
In [46]:
```
# We then use the cKDTree in two functions. We get the x, y position
# of each micromagnetic cell, and query the tree for the region that
# the cell sits in. The functions then return the axis and strength
# at that region index.

def K_axis(pos):
    x, y, z = pos
    _, test_point_regions = voronoi_kdtree.query(np.array([[x, y, z]]), k=1)
    region = test_point_regions[0]
    return axes[region]

def K_mag(pos):
    x, y, z = pos
    _, test_point_regions = voronoi_kdtree.query(np.array([[x, y, z]]), k=1)
    region = test_point_regions[0]
    return strengths[region]
```

In [56]:
```
K = 1e6
u = df.Field(mesh, dim=3, value=K_axis)

# uniaxial anisotropy axis
system = mm.System(name='grain3')
system.energy = mm.UniaxialAnisotropy(K=K, u=u)
```

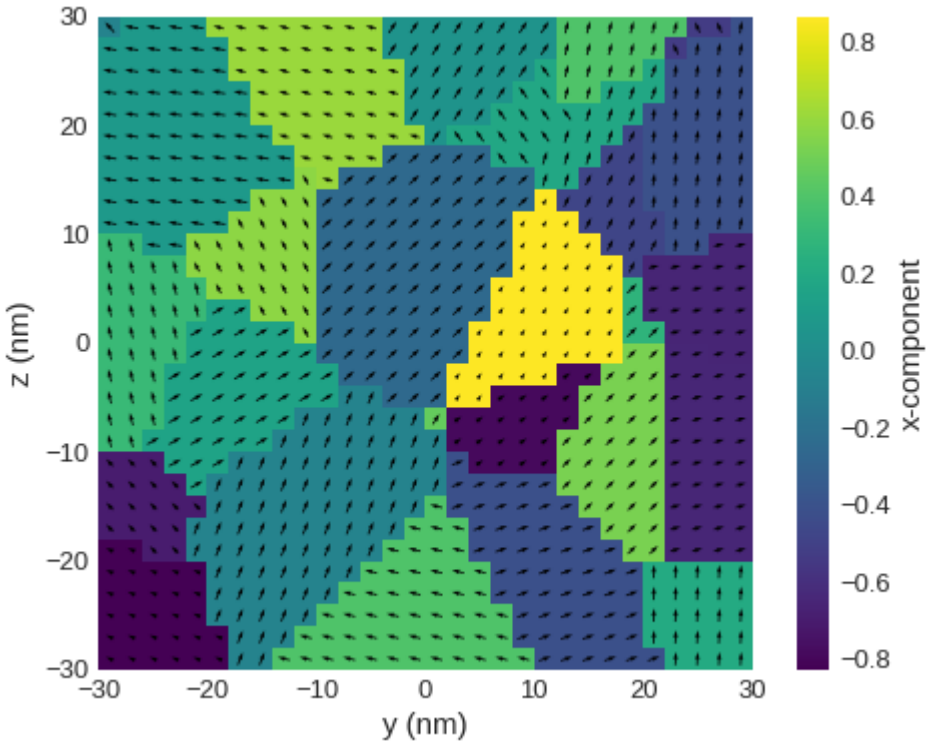
In [57]:
```
u.plane('z').mpl()
```



In [58]:
```
u.plane('y').mpl()
```



In [59]:
```
u.plane('x').mpl()
```



In [60]:
```
gamma0 = 0  # gyromagnetic ratio (m/As)
alpha = 0.2  # Gilbert damping
system.dynamics = mm.Precession(gamma0=gamma0) + mm.Damping(alpha=alpha)


system.energy = mm.Exchange(A=A) +  mm.UniaxialAnisotropy(K=K, u=u) + mm.Demag()
system.m = df.Field(mesh, dim=3, value=(0, 0, 1), norm=Ms)
Hmin = (0, 0, -3/mm.consts.mu0)
Hmax = (0, 0, 3/mm.consts.mu0)
n = 61
```

In [61]:
```
hd = mc.HysteresisDriver()
hd.drive(system, Hmin=Hmin, Hmax=Hmax, n=n)
```

```
Running OOMMF (TclOOMMFRunner) [2021/03/10 10:44]... (118.6 s)
```

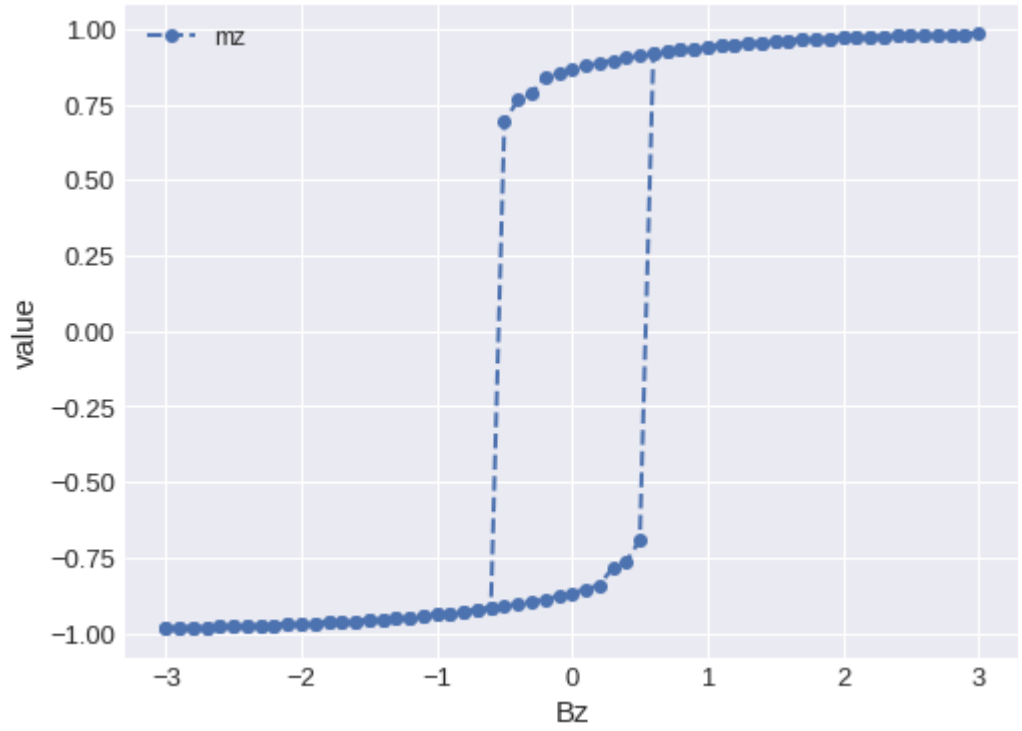
In [62]:
```
system.table.data.head()
```

Out[62]:

	max_mxHxm	E	delta_E	bracket_count	line_min_count	conjugate_cycle_count
0	0.084858	-4.383574e-16	1.479114e-31	250.0	50.0	177.0
1	0.092319	-4.201204e-16	2.465190e-31	296.0	95.0	178.0
2	0.096270	-4.019003e-16	0.000000e+00	343.0	141.0	179.0
3	0.098174	-3.836985e-16	-4.930381e-32	391.0	188.0	180.0
4	0.078269	-3.655165e-16	0.000000e+00	440.0	236.0	181.0

5 rows × 26 columns

In [63]:
```
system.table.mpl(x='Bz', y=['mz'], marker='o', linewidth=2, linestyle='dashed')
```



In [ ]:
```

```