In [1]:
```
import numpy as np
from scipy.integrate import odeint
%matplotlib inline
import matplotlib.pyplot as plt
from matplotlib.backends.backend_pdf import PdfPages
import sys
```

In [2]:
```
t = np.linspace(0,50395, num=10080)
lightdata = np.transpose(np.delete(np.genfromtxt('AllLightnosmooth.csv', delimiter=',')


def func(t):
    z0 = [1,0,0,0,0,0]
    #mCherry2 = np.empty((len(lightdata[:,0]), len(t)))
    out =  np.empty((len(lightdata[:,0]),len(t),6))
    arrayvalues = np.asarray([])
    #for i in range(42):
    for i in range(len(lightdata[:,0])):
        def I(t):
            tindex = t/5
            if tindex > 10079:
                tindex = 10079
            return lightdata[i][int(tindex)]

        def model(z,t):
            d1 = 0.01829
            k1 = 0.08299
            d2 = 0.116691
            k2 = 0.001023
            Kd = 90.41029
            n = 0.964487
            k3 = 0.000432
            d3 = 0.000544
            k4 = 1.25
            d4 = 0.0000924
            k5 = 0.00144



            Pu = z[0]
            Pb = z[1]
            Pa = z[2]
            mRNA = z[3]
            mCherry1 = z[4]
            mCherry2 = z[5]

            dPudt = d1*Pb - k1*I(t)**n/(Kd**n+I(t)**n)*Pu
            dPbdt = k1*I(t)**n/(Kd**n+I(t)**n)*Pu + d2*Pa - d1*Pb - k2*Pb
            dPadt = k2*Pb - d2*Pa
            dmRNAdt = k3*Pa - d3*mRNA
            dmCherry1dt = k4*mRNA-(d4 + k5)*mCherry1
            dmCherry2dt = k5*mCherry1-d4*mCherry2
            return [dPudt,dPbdt,dPadt,dmRNAdt,dmCherry1dt,dmCherry2dt]

        z = odeint(model,z0,t)
        #mCherry2[i] = z[:,5]
        out[i] = z
    return out
```

In [3]:
```
model1 = np.asarray(func(t))
mCherry2 = model1[:,:,5]
Pu = model1[:,:,0]
Pb = model1[:,:,1]
Pa = model1[:,:,2]
mRNA = model1[:,:,3]

#total = Pu+Pb+Pa+Pi
#print(total)

#for i in range(42):
for i in range(len(lightdata[:,0])):
    last = mCherry2[i]
    ##total = Pi[i]+Pu[i]+Pb[i]+Pa[i]
    print(last[-1])
```

```
41.52102916687377
20.248554765093214
11.389581912765381
1.8871551847732995
0.0008445589381288593
48.93206360808423
26.001722652785872
15.161797197478716
0.001154020979539711
56.16402627386461
32.693429947049424
19.909975731782254
0.14832197297000765
0.01053327522564022
0.0015676049316839923
57.41491850921742
33.980666843491
20.87163934724879
0.0051628795654433055
0.0016548202226326042
35.494311520885724
12.15331773778011
6.07978526176911
2.0313077762943204
0.010069244008819278
40.50579331158043
13.902268656230902
6.954619647219236
2.3234130298728037
0.011516376825686266
0.16659117613819868
45.05303715485226
15.493082701154886
7.7503288940517185
2.5890520819370226
0.012832205295281585
0.012832193287178564
45.80782544578662
15.757486590374244
7.882578900882757
2.6331981006443836
0.6630126518329228
39.68059154012673
29.786700318726183
9.871463904690216
4.4833870791702415
1.176579630134472
0.04201448282737889
```

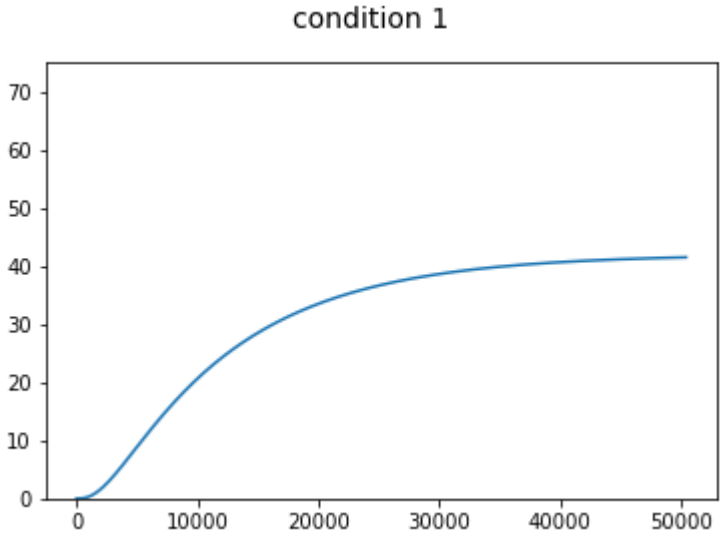
```
44.49656715098458
33.4026179670408
11.068967538830927
5.026900122839956
1.3191776123432195
0.04710647617494394
48.76697067925531
36.60902425334908
12.13069939313899
5.508724304787249
1.445584001327029
0.05162027056911451
49.467095759261674
37.134719049305076
12.304758339562895
5.587708133066559
1.4663047269912888
0.05236020533361611
49.13853517176222
28.883918677120203
13.527884015403043
3.5933613742191444
1.445451229080786
0.12834705154872958
54.87281351102796
32.2532258691551
15.104555148357768
4.012029690292364
1.6138619684611888
0.1433008735069185
59.92102579634878
35.21918506982529
16.492248245914315
4.38049332953662
1.7620775704672378
0.1564614755752918
60.74547925776723
35.703552535712646
16.718849634497214
4.4406591827713555
1.7862794420226475
0.15861042543992165
52.80648752181816
37.059953125307196
28.282560075083353
7.755658886669863
3.121711727898584
0.2771930458770494
58.90268232776609
41.33609153744877
31.544666261569336
8.649788770366513
3.4816025607002743
0.30914960883258175
64.25858557690182
45.09258252908244
34.41014061211318
9.4351351982198
3.797707077784863
0.3372182020920295
65.13232643643724
45.70536860617659
34.87755855260295
9.5632352287049
3.8492677870120424
```

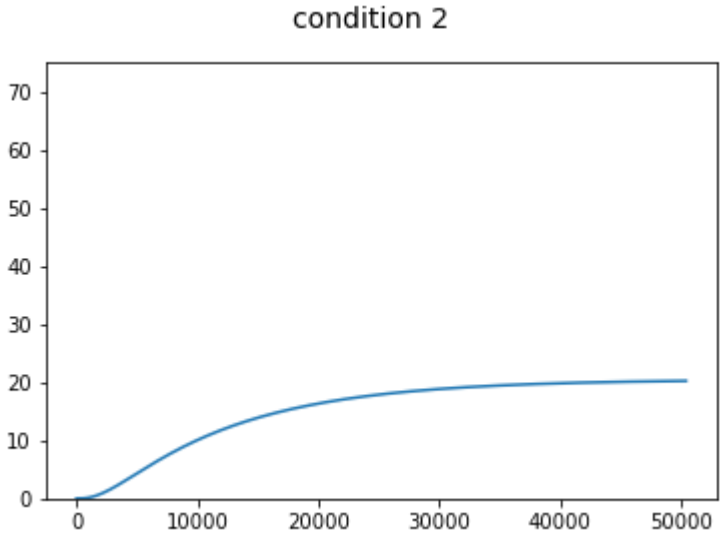
```
0.34179653017527944
56.1672534780314
62.583563270314805
68.20940350755855
69.1261823667569
```

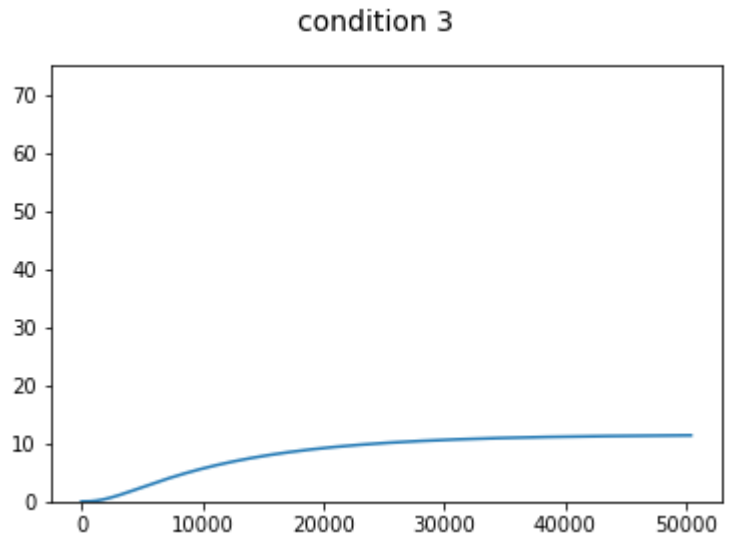
In [4]:

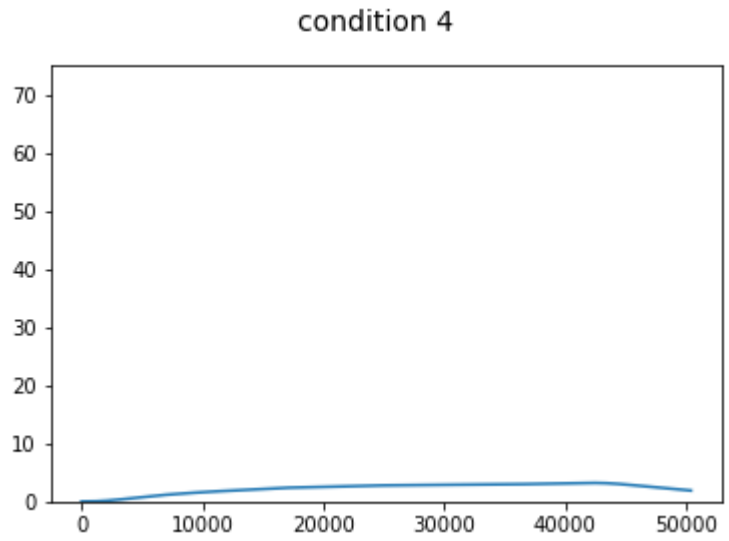
```
import csv
pp = PdfPages('multipage.pdf')

#for i in range(42):
for i in range(len(lightdata[:,0])):
#    mCherry2 = model1[:,:,6]
#    with open('dataout.csv','w') as csvfile:
#        csvwriter=csv.writer(csvfile)
#        for row in mCherry2:
#            csvwriter.writerow(row)
#    Pu = model1[:,:,0]
#    with open('Puout.csv','w') as csvfile:
#        csvwriter=csv.writer(csvfile)
#        for row in Pu:
#            csvwriter.writerow(row)
#    Pb = model1[:,:,1]
#    with open('Pbout.csv','w') as csvfile:
#        csvwriter=csv.writer(csvfile)
#        for row in Pb:
#            csvwriter.writerow(row)
#    Pi = model1[:,:,2]
#    with open('Piout.csv','w') as csvfile:
#        csvwriter=csv.writer(csvfile)
#        for row in Pi:
#            csvwriter.writerow(row)
#    Pa = model1[:,:,3]
#    with open('Paout.csv','w') as csvfile:
#        csvwriter=csv.writer(csvfile)
#        for row in Pa:
#            csvwriter.writerow(row)
#    mRNA = model1[:,:,4]
#    with open('mRNAout.csv','w') as csvfile:
#        csvwriter=csv.writer(csvfile)
#        for row in mRNA:
#            csvwriter.writerow(row)
    #U=U*100;
    #Pu=Pu*100;
    #Pb=Pb*100;
    #Pa=Pa*100;
    #t = np.linspace(0,50395, num=10080)
    #plt.plot(t,U[i])
    #plt.plot(t,Pu[i])
    #plt.plot(t,Pb[i])
    #plt.plot(t,Pa[i])
    #plt.legend(['U','Pu','Pb','Pa'])
    #plt.plot(t,lightdata[i])
    plt.plot(t,mCherry2[i])
    #plt.xlabel('Time [s]')
    #plt.ylabel('Fluoresence [a.u.]')
    plt. suptitle('condition '+str(i+1),fontsize = 14)
    plt.ylim((0,75))


    #def annot_max (t, mCherry2, ax=None):
    #    tmax = t[np.argmax(mCherry2[i])]
```

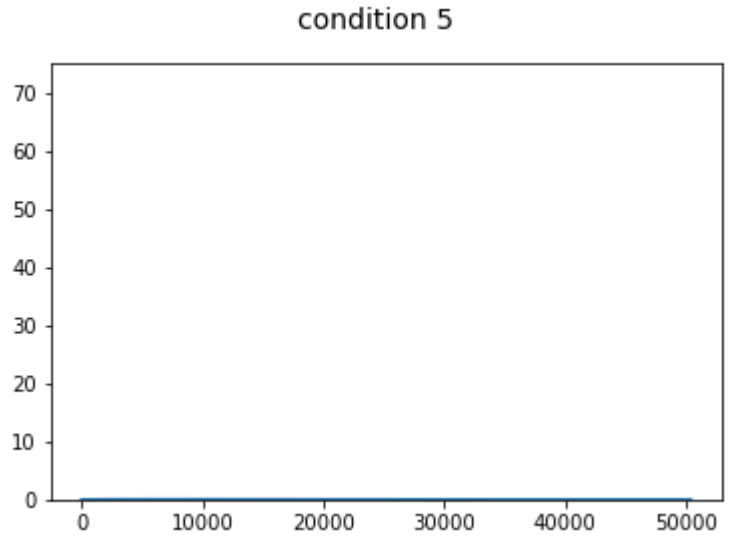
```
    #    ymax = mCherry2[i].max()
    #    text = "t={:.3f}, y={:.3f}".format(tmax,ymax)
    #    print(i+1, ymax)
    #    if not ax:
    #        ax=plt.gca()
    #    bbox_props = dict(boxstyle = "square,pad=0.3", fc="w", ec="k",lw=0.72)
    #    arrowprops = dict(arrowstyle="->", connectionstyle = "angle,angleA=0,angleB=60
    #    kw = dict(xycoords='data', textcoords="axes fraction", arrowprops=arrowprops,
    #    ax.annotate(text,xy=(tmax,ymax), xytext=(0.94,0.7), **kw)
    #annot_max(t,mCherry2)
    pp.savefig()
    plt.show()

pp.close()
```

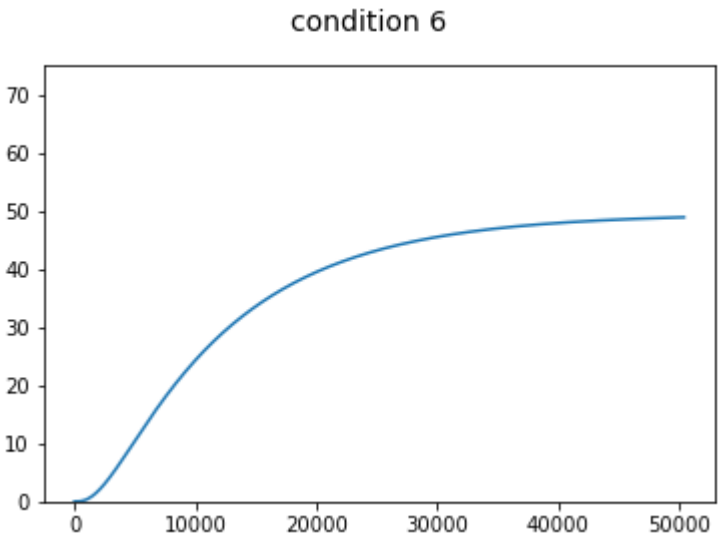


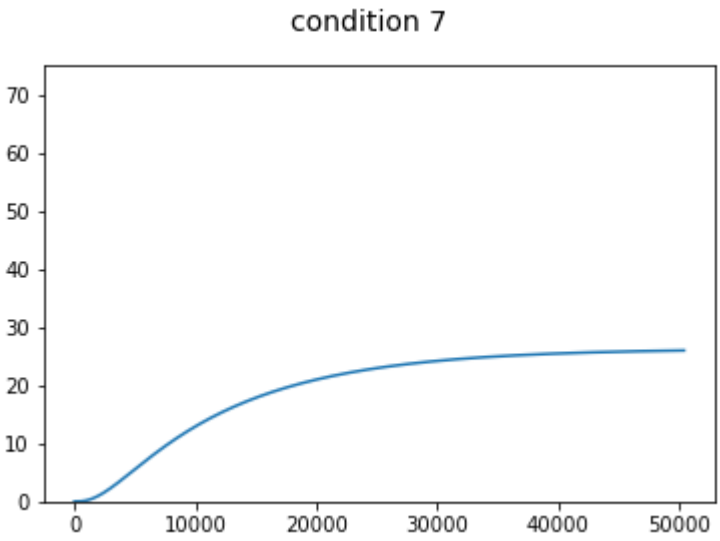


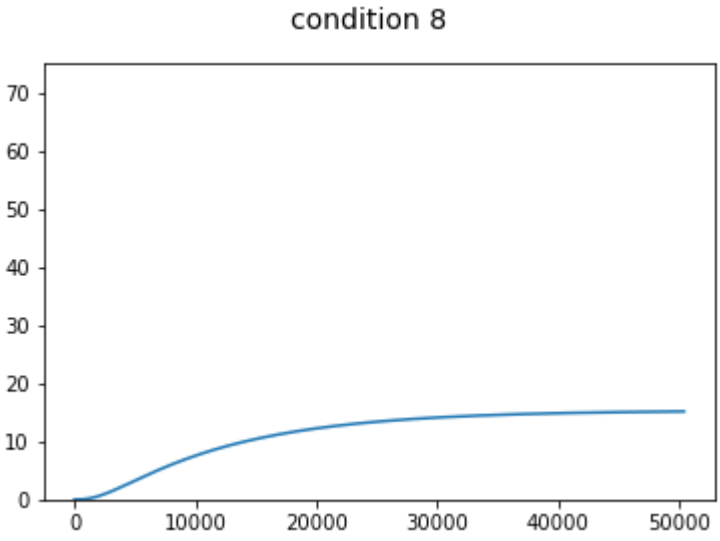

condition 3
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

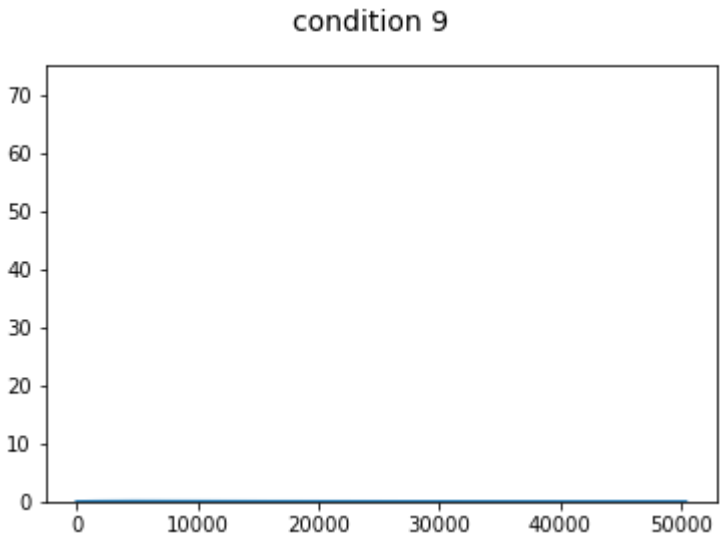

condition 4
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

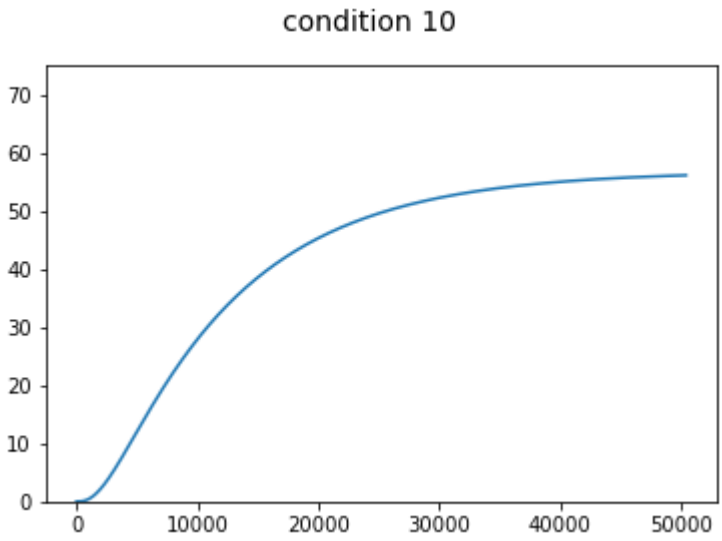

condition 5
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

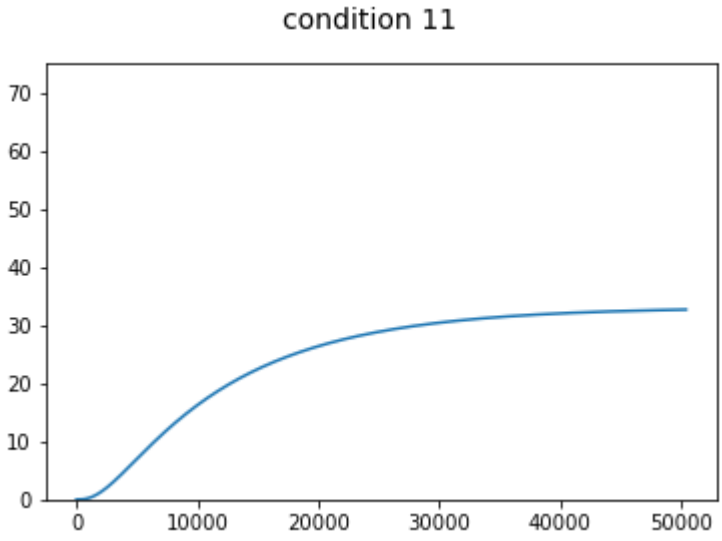

condition 6
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

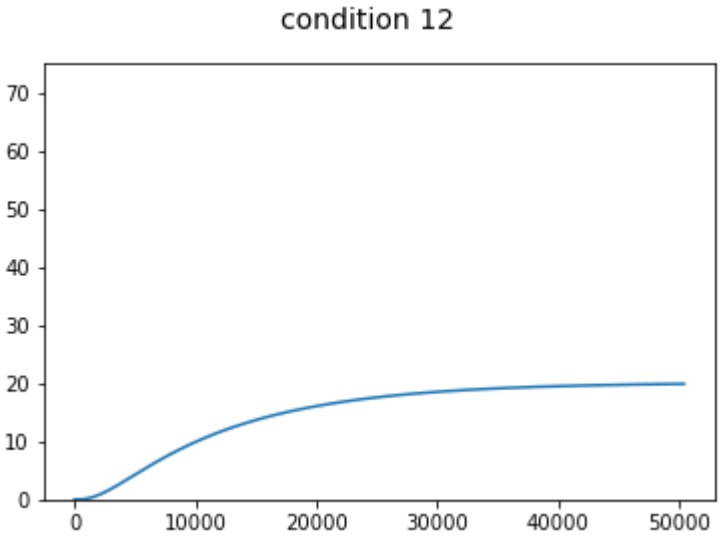

condition 7
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

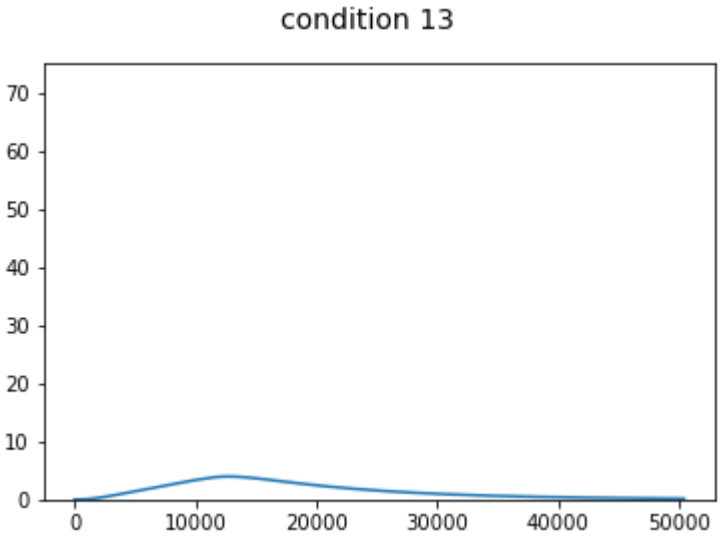

condition 8
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

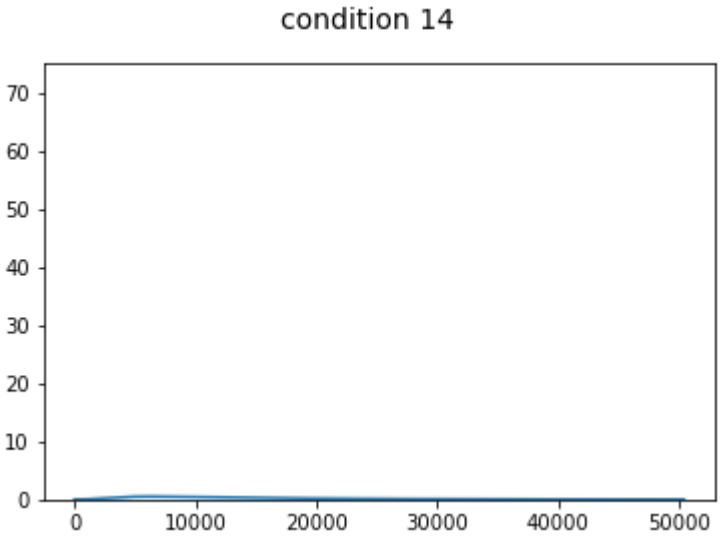

condition 9
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

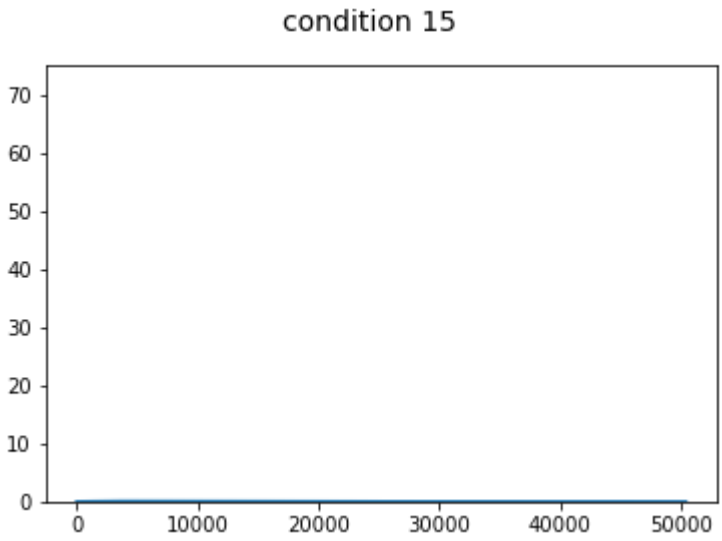

condition 10
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

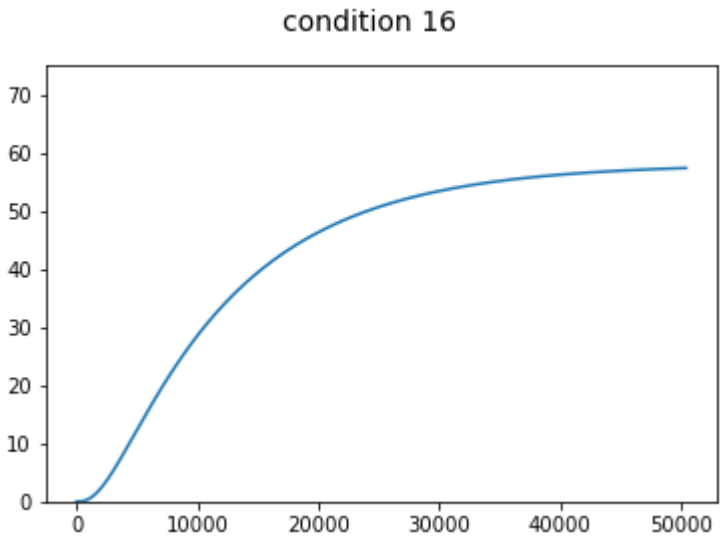

condition 11
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

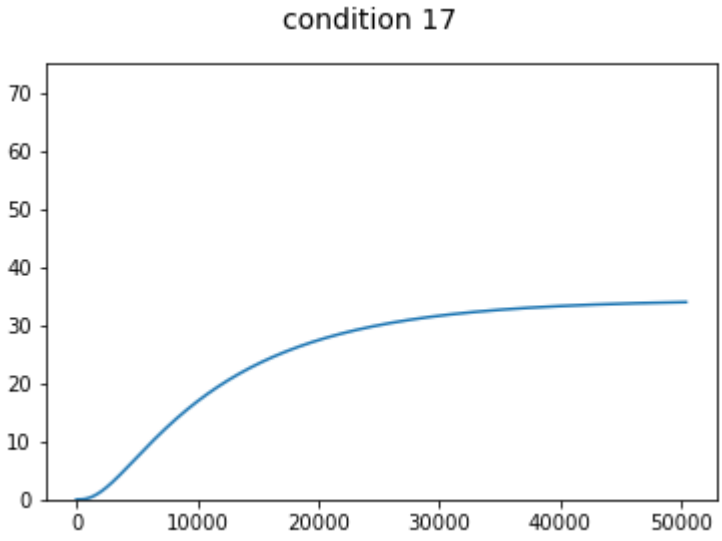

condition 12
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

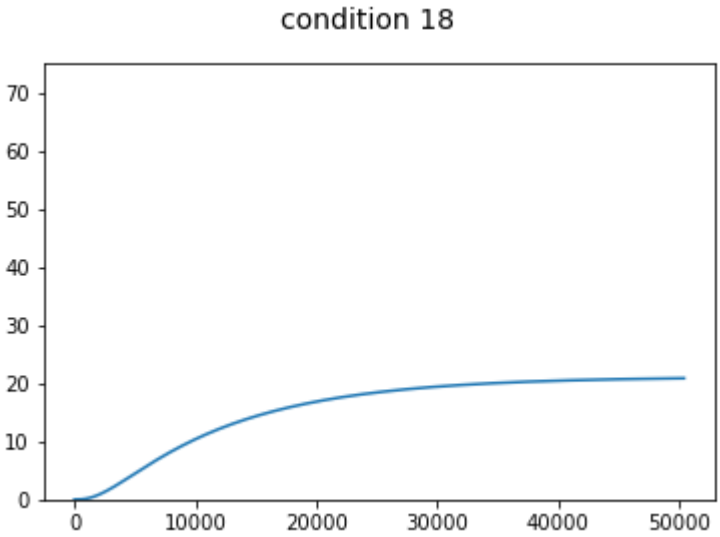

condition 13
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

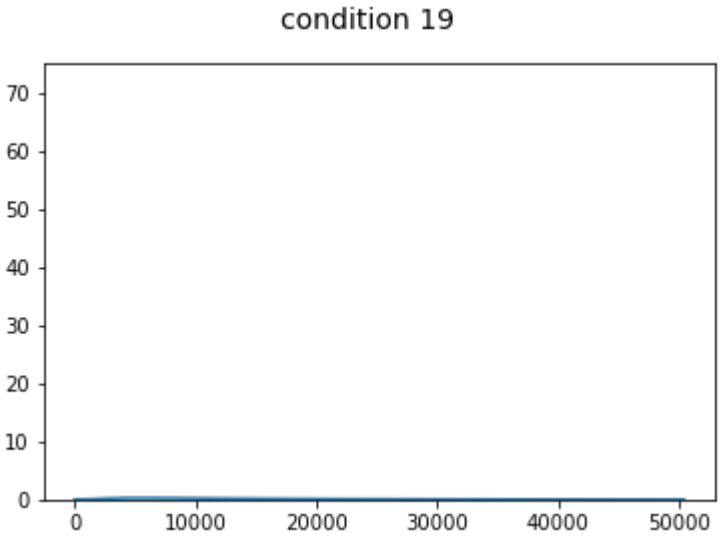

condition 14
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

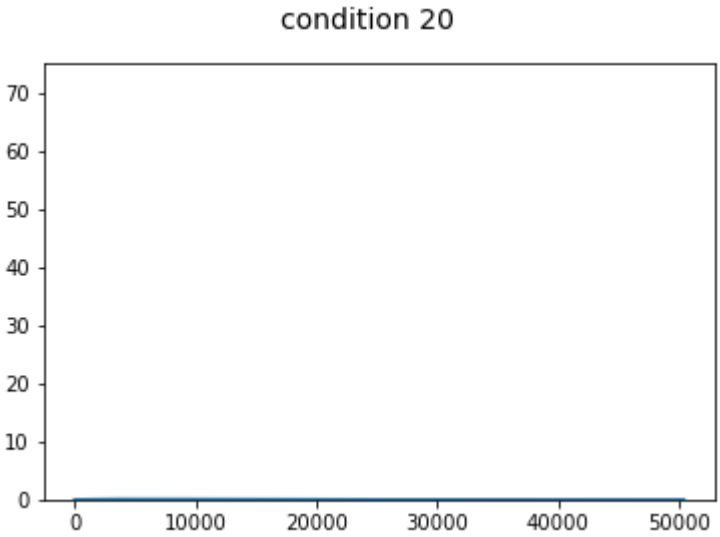

condition 15
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

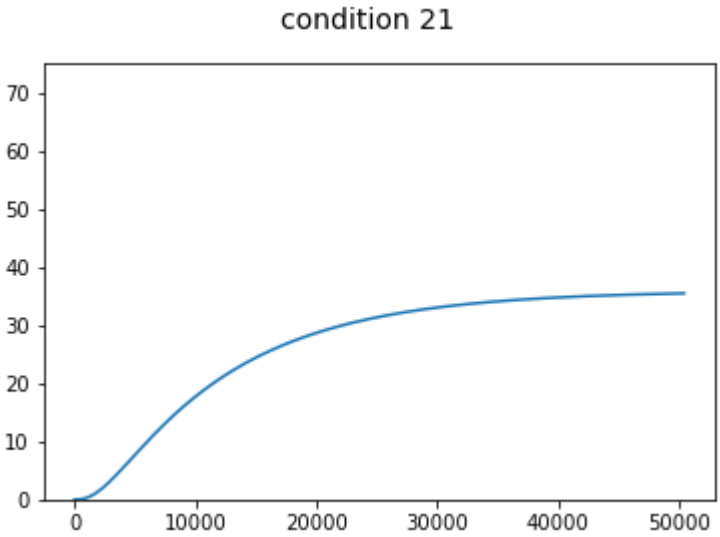

condition 16
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

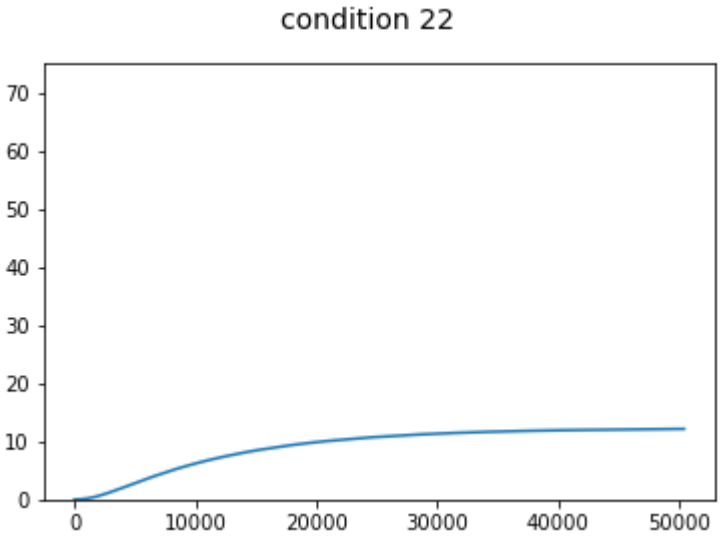

condition 17
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

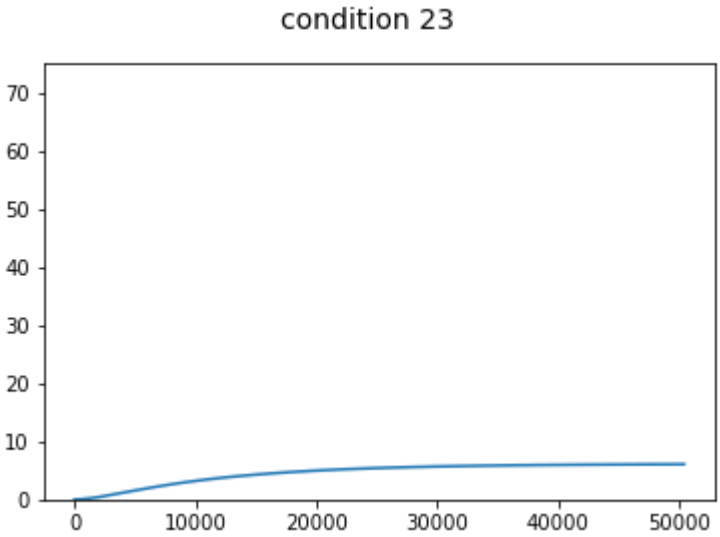

condition 18
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

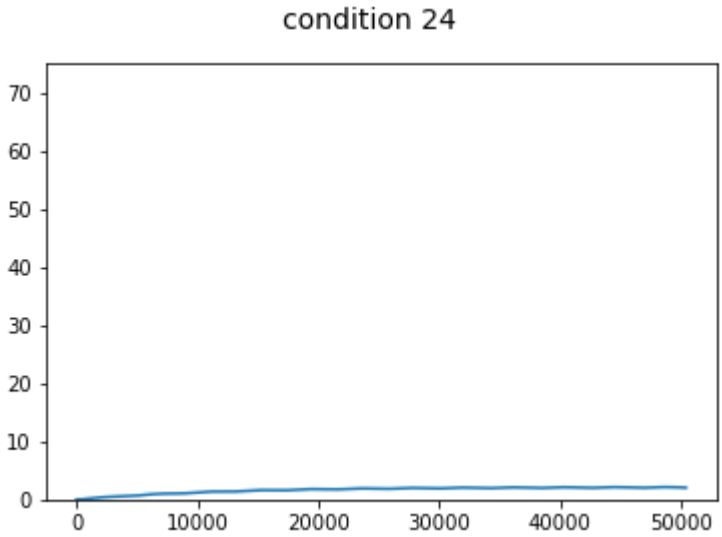

condition 19
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

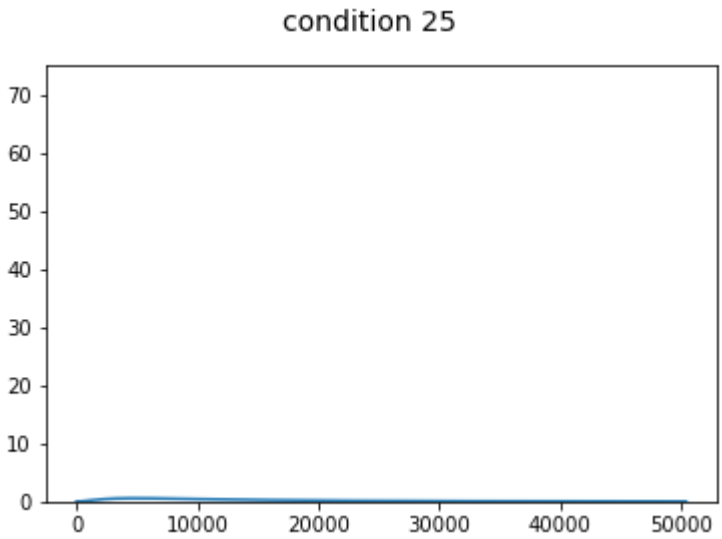

condition 20
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

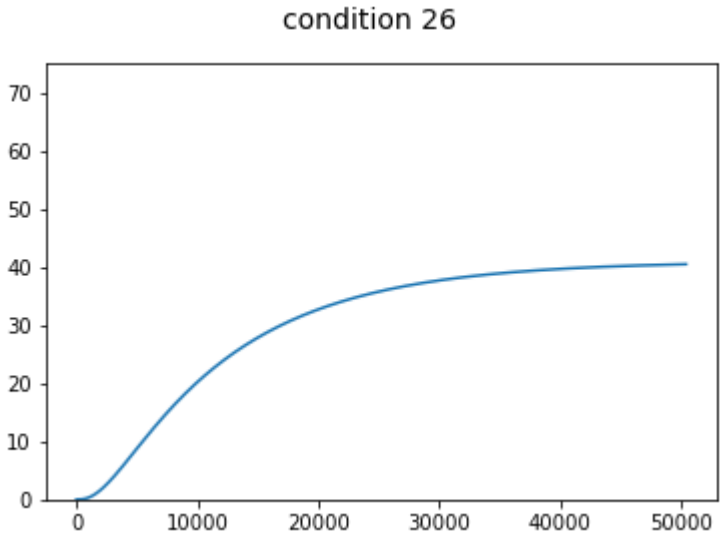

condition 21
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

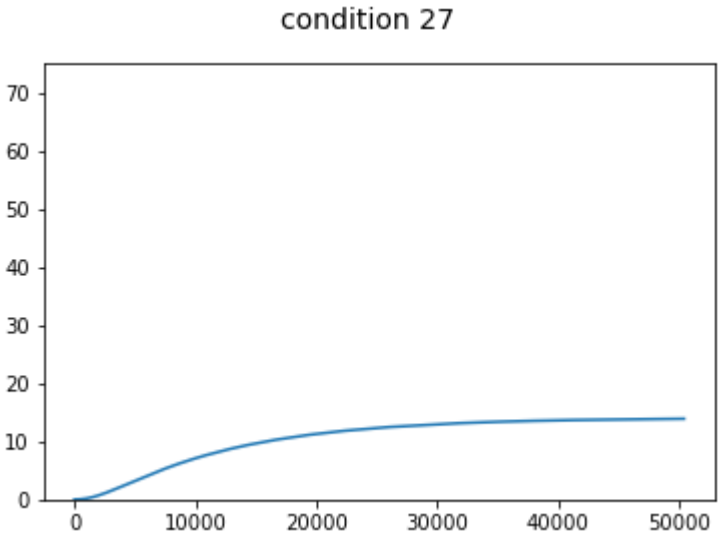

condition 22
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

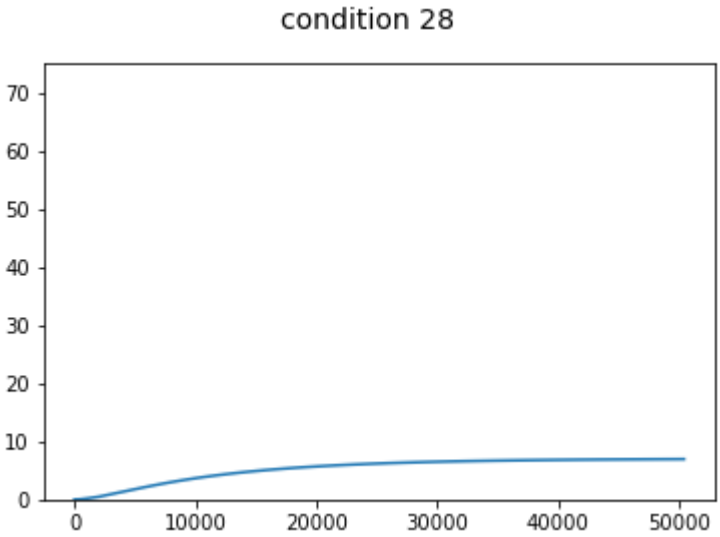

condition 23
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

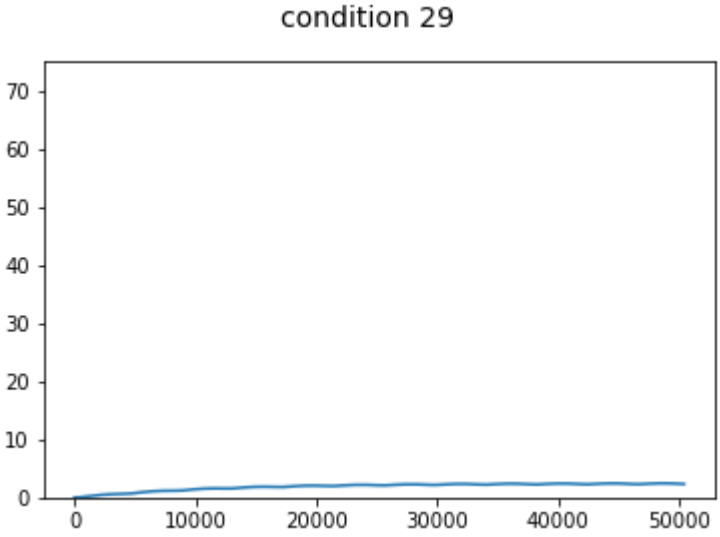

condition 24
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

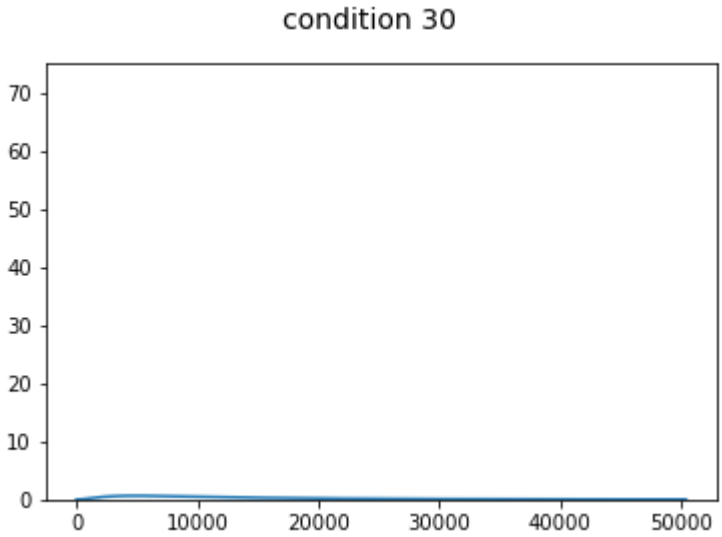

condition 25
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

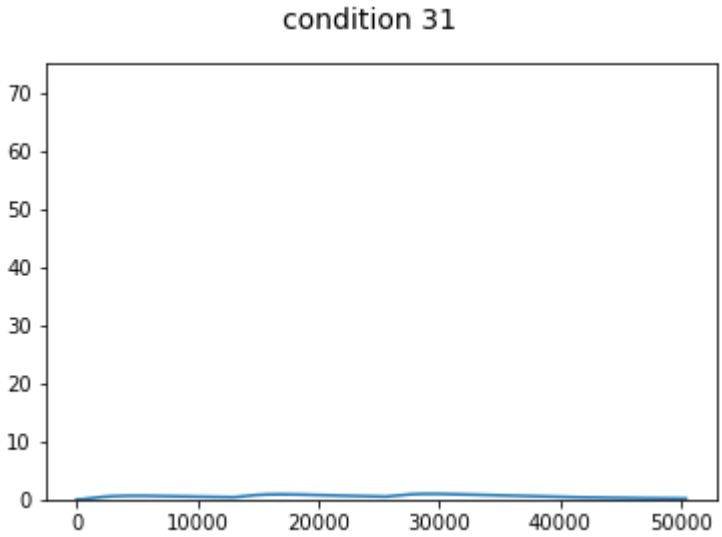

condition 26
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

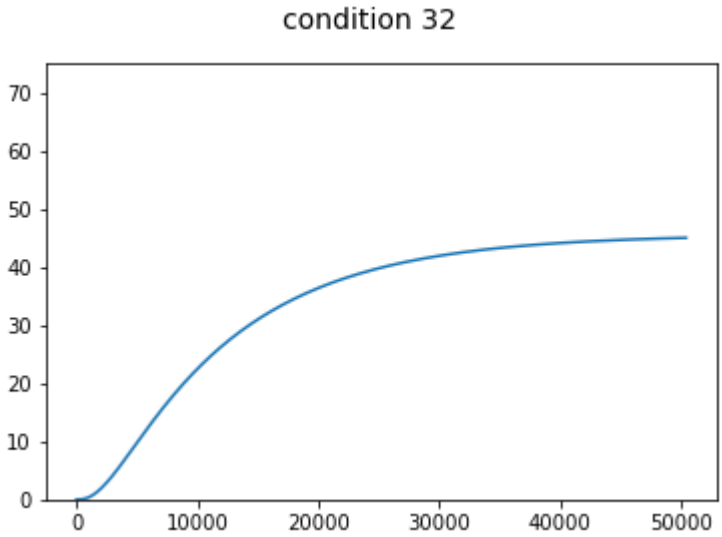

condition 27
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

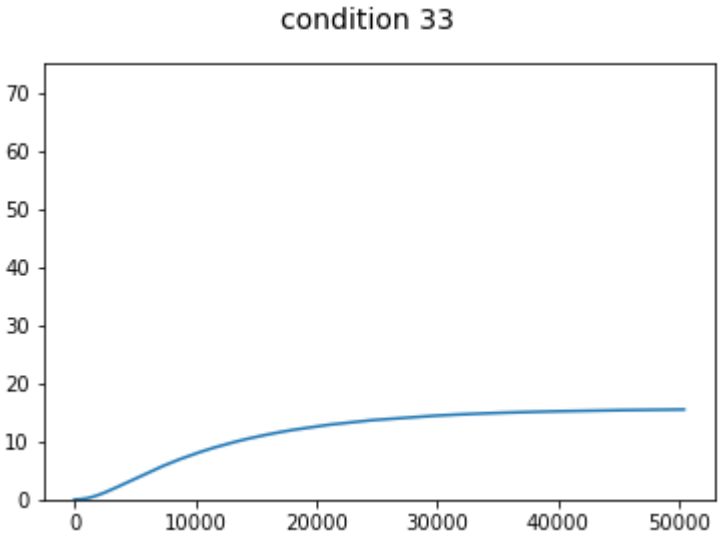

condition 28
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

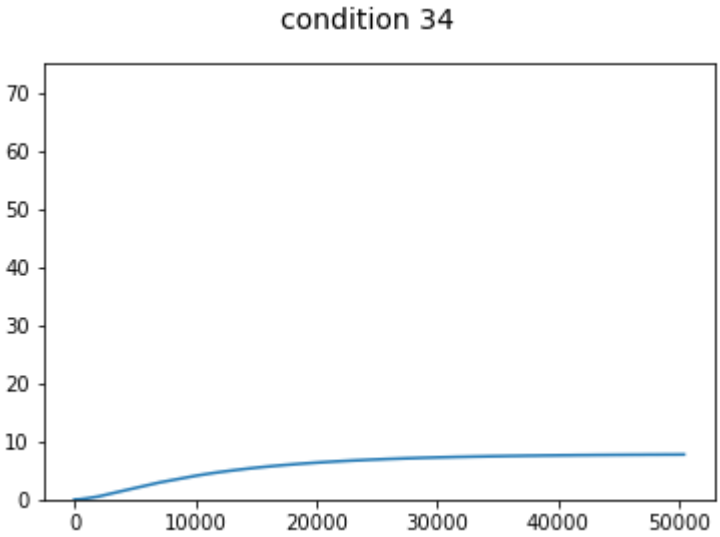

condition 29
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

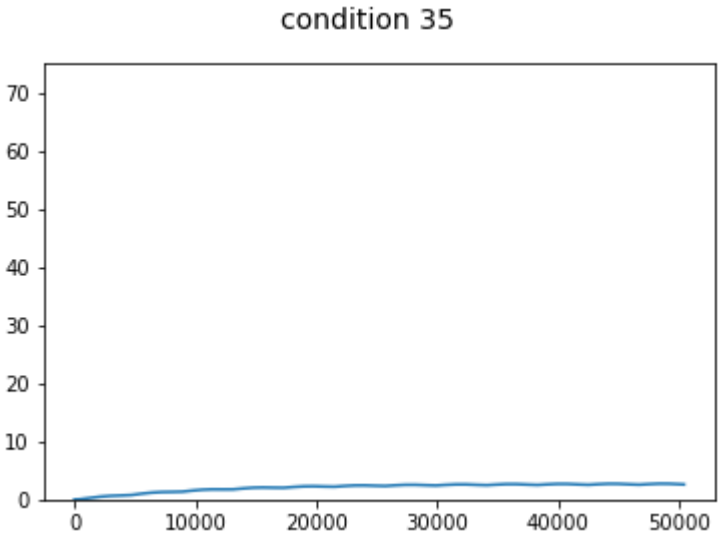

condition 30
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

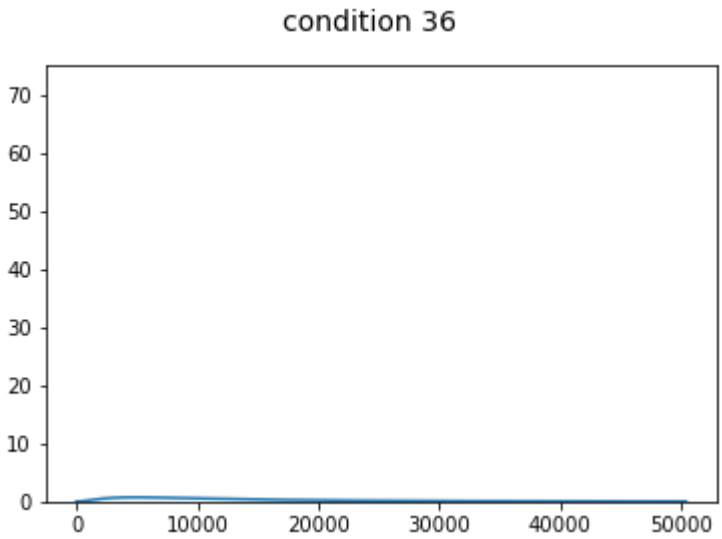

condition 31
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

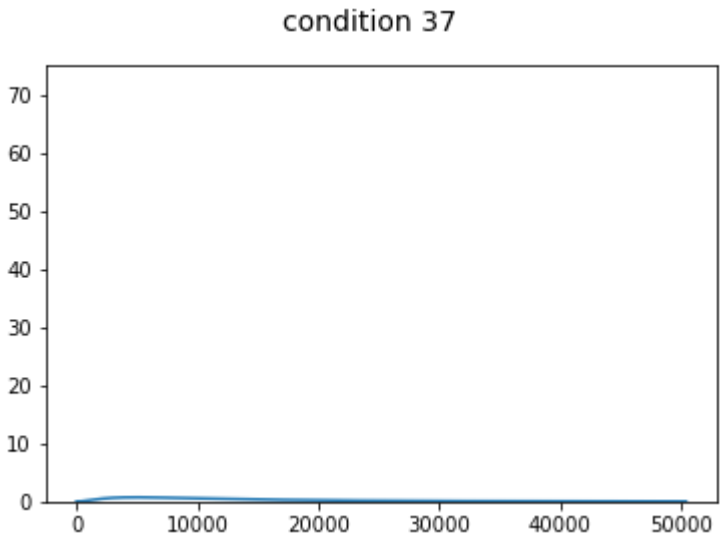

condition 32
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

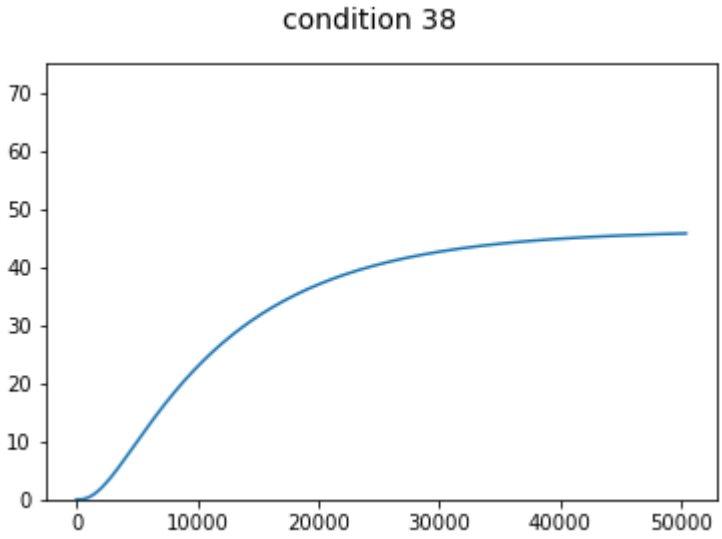

condition 33
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

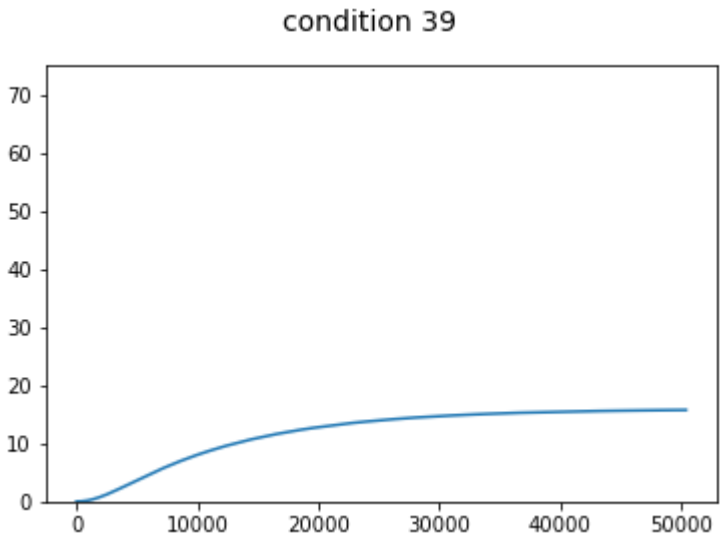

condition 34
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

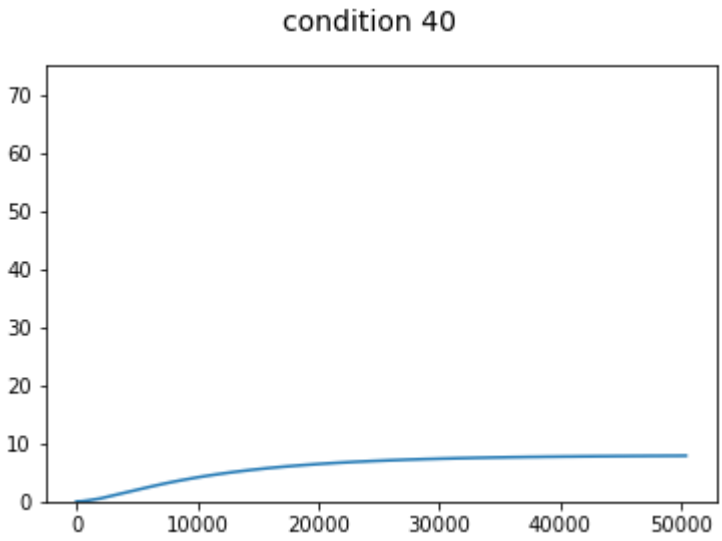

condition 35
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

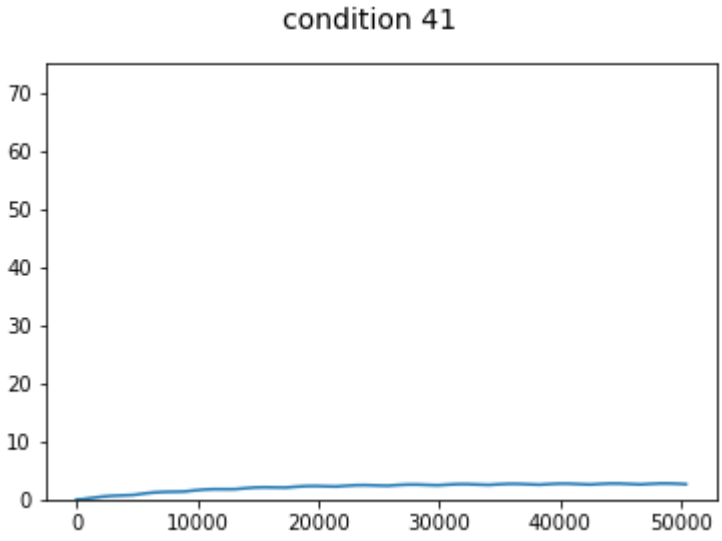

condition 36
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

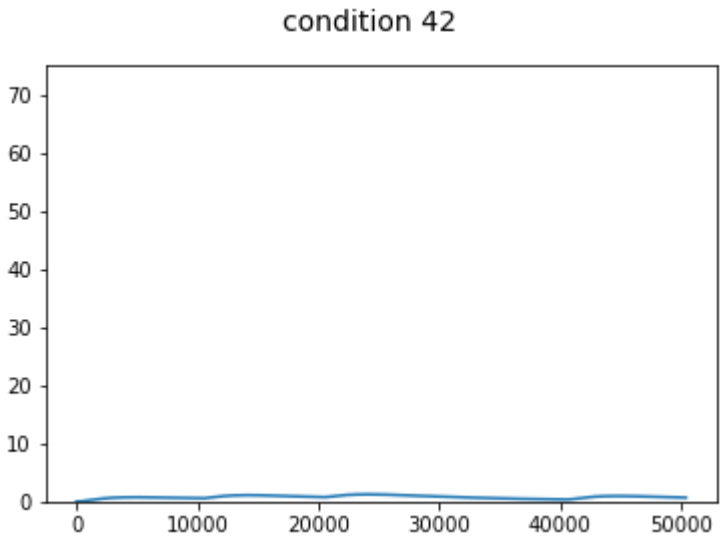

condition 37
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

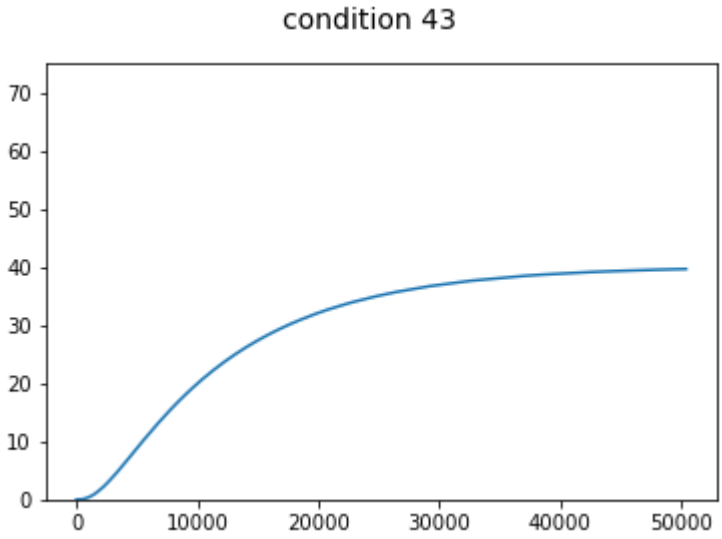

condition 38
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

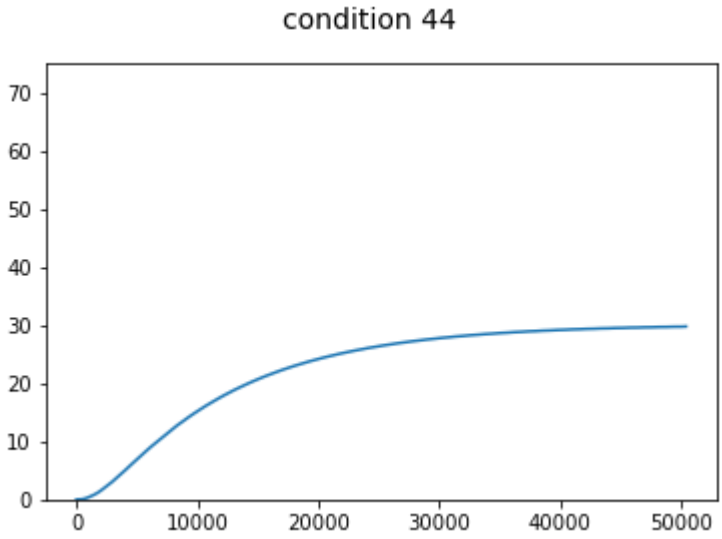

condition 39
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

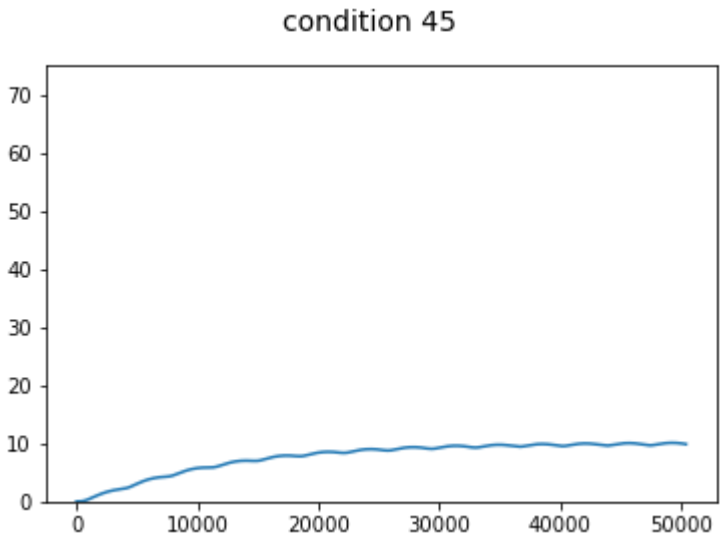

condition 40
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

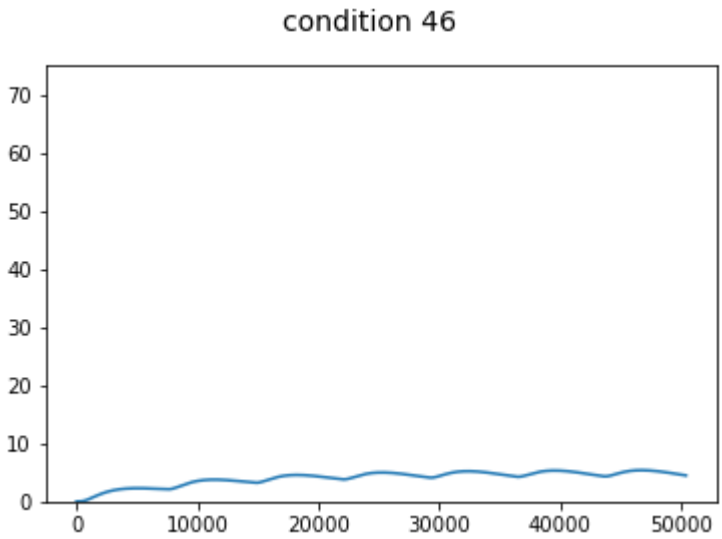

condition 41
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

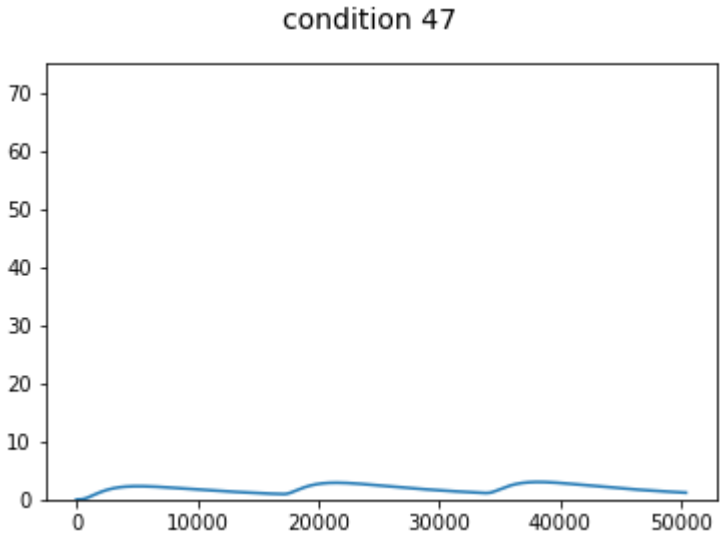

condition 42
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

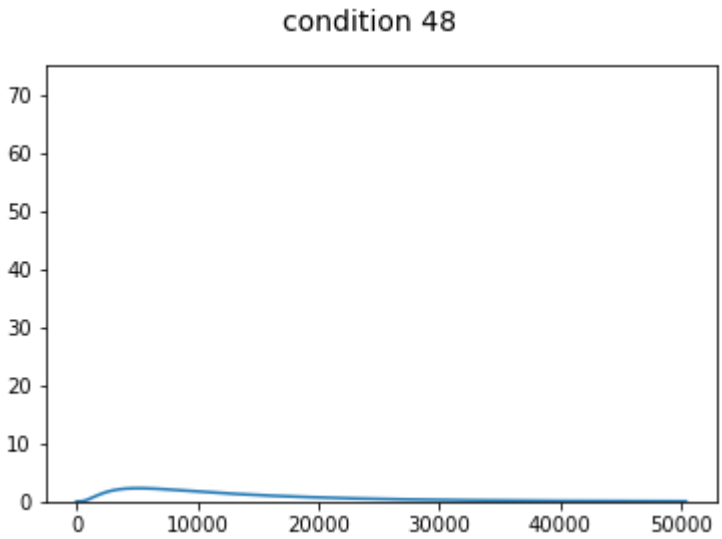

condition 43
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

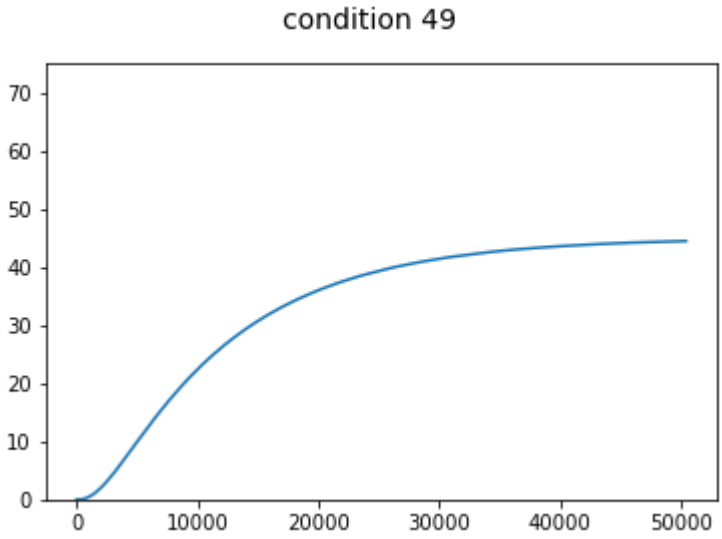

condition 44
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

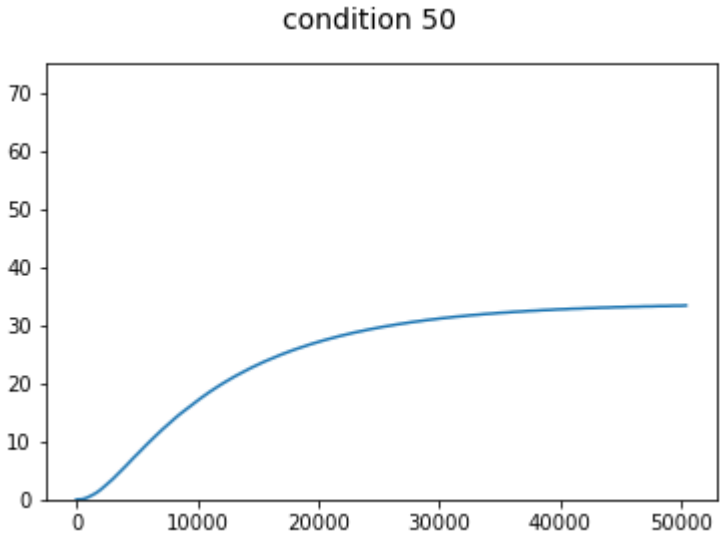

condition 45
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

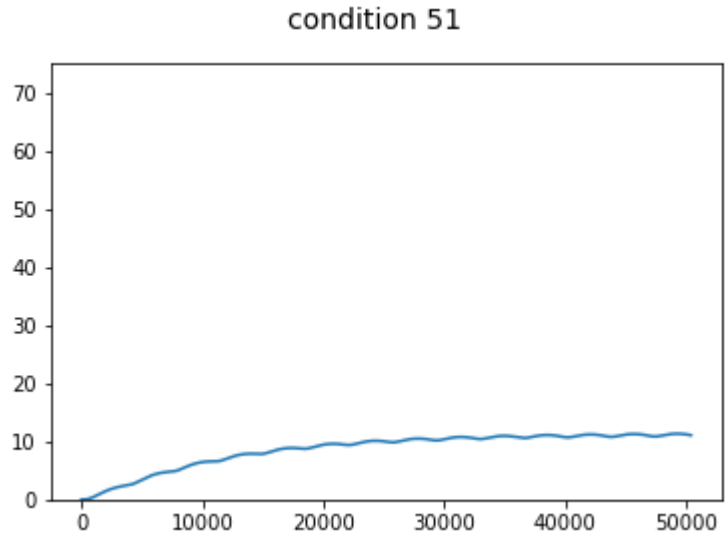

condition 46
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

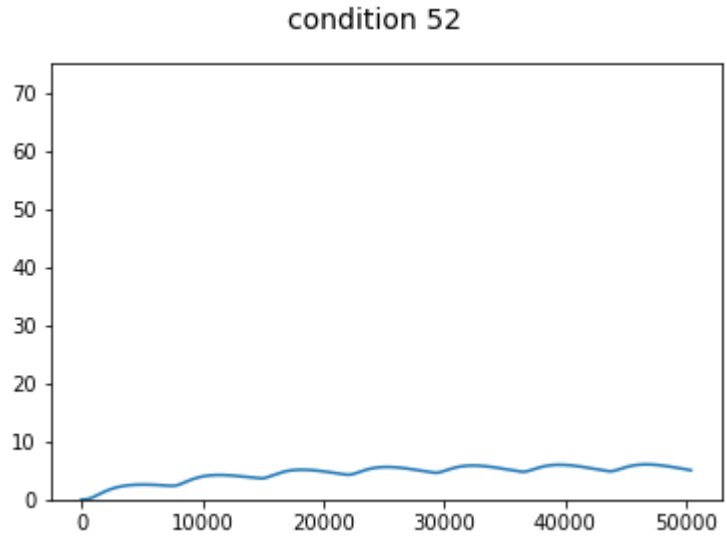

condition 47
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

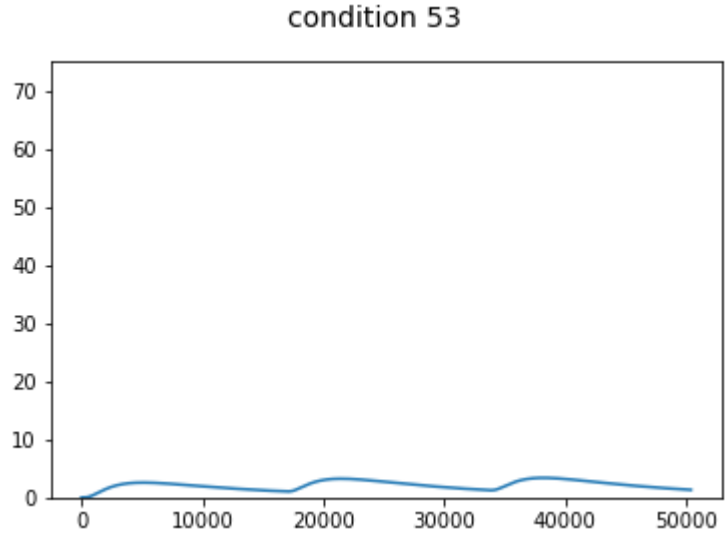

condition 48
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

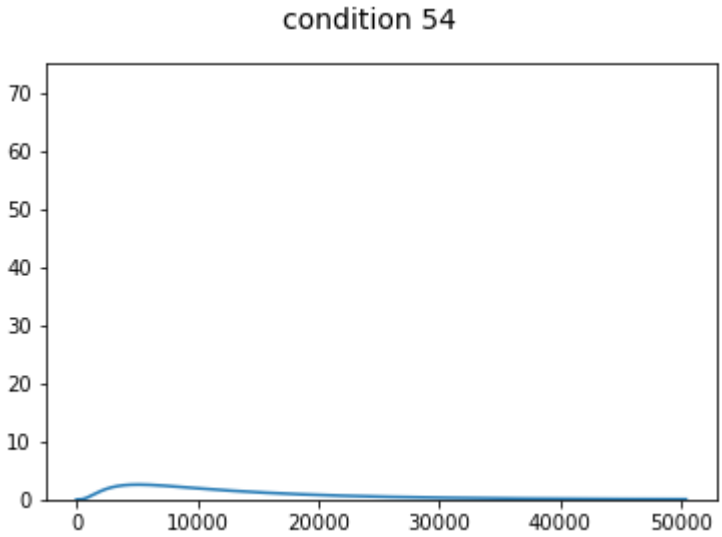

condition 49
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

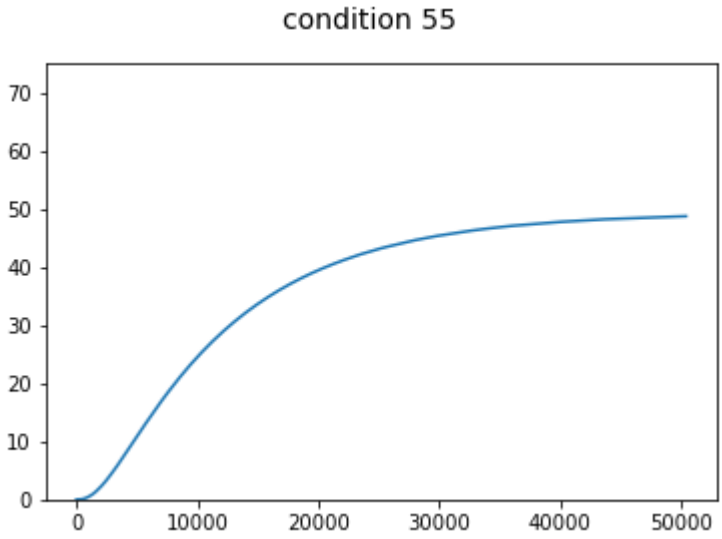

condition 50
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

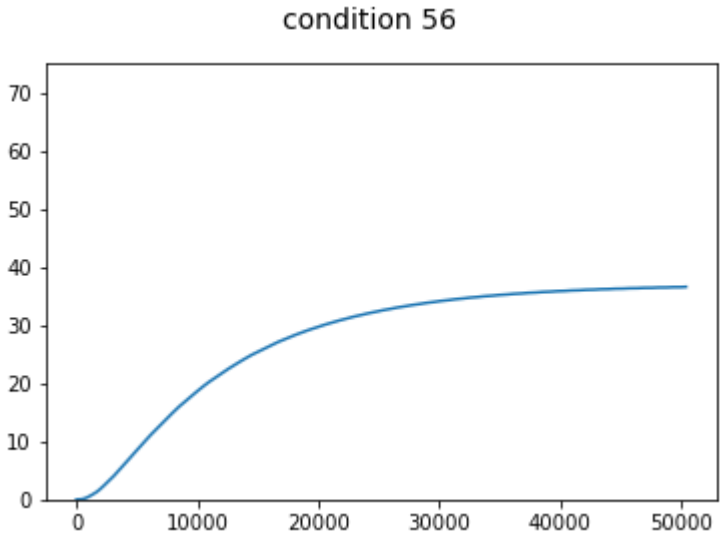

condition 51
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

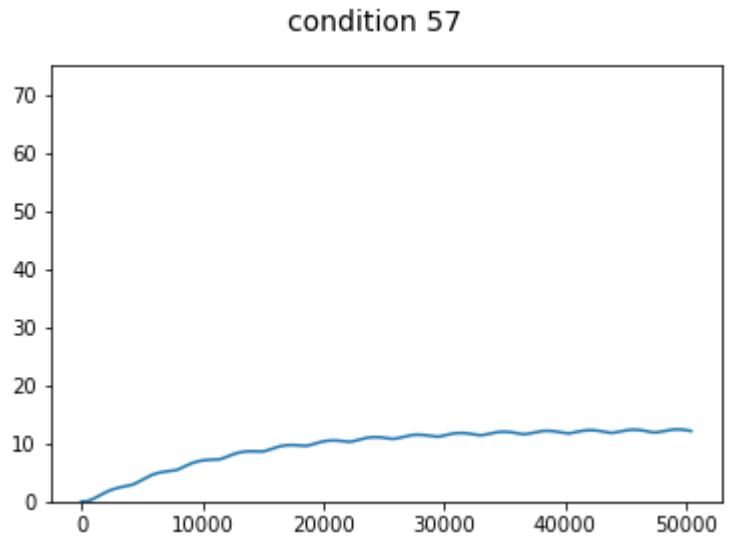

condition 52
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

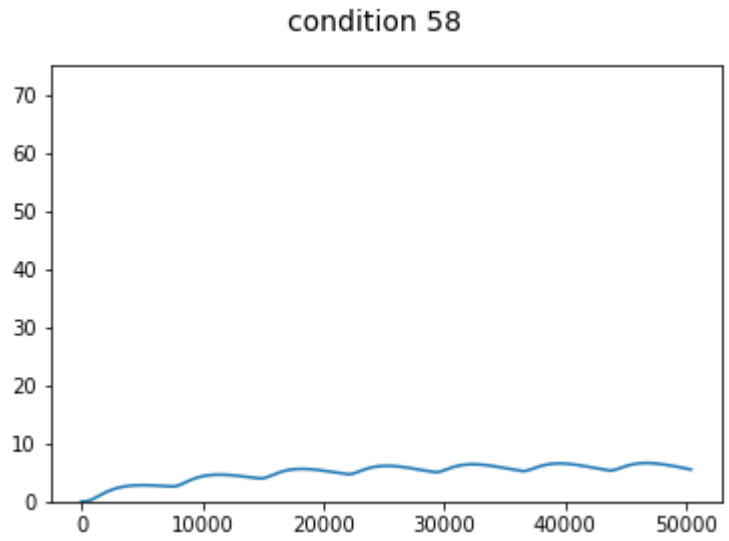

condition 53
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

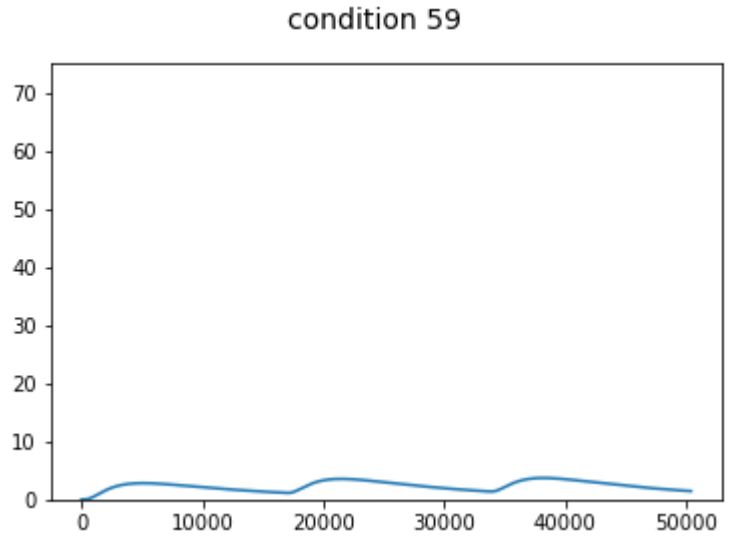

condition 54
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

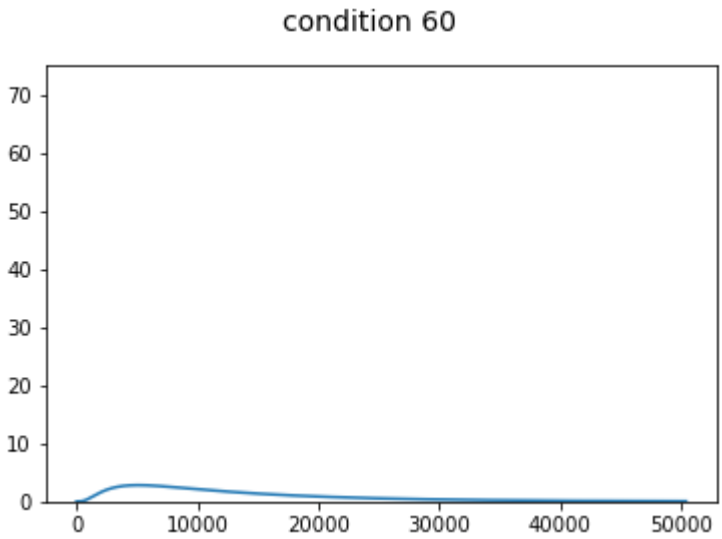

condition 55
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

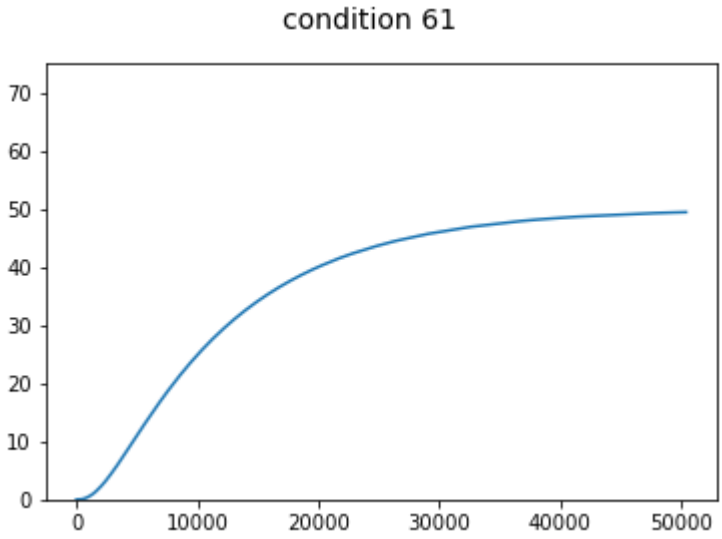

condition 56
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

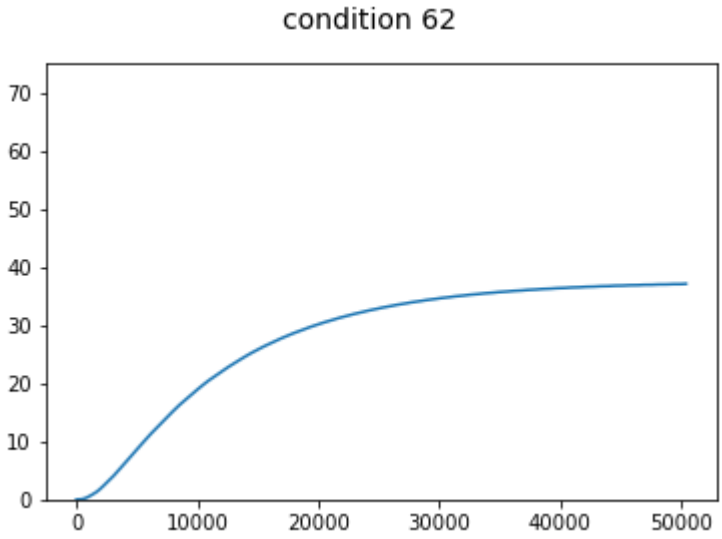

condition 57
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

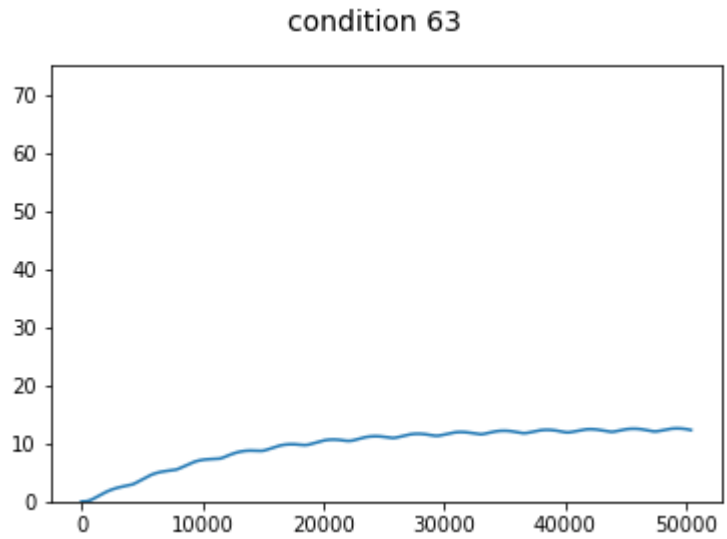

condition 58
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

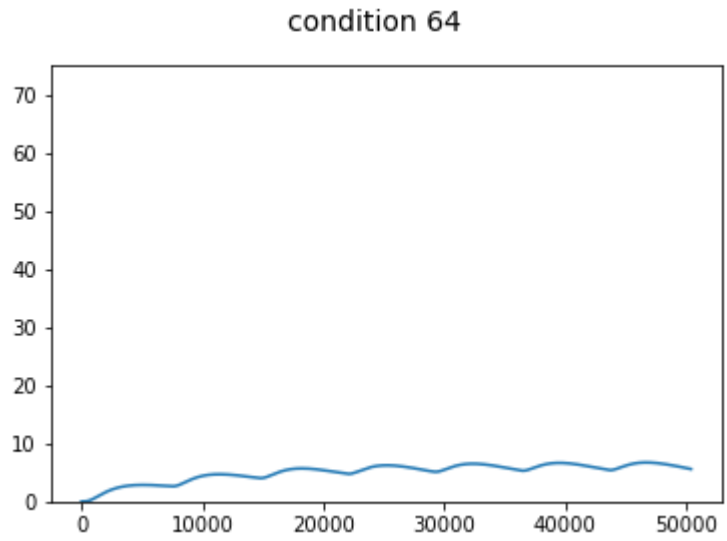

condition 59
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

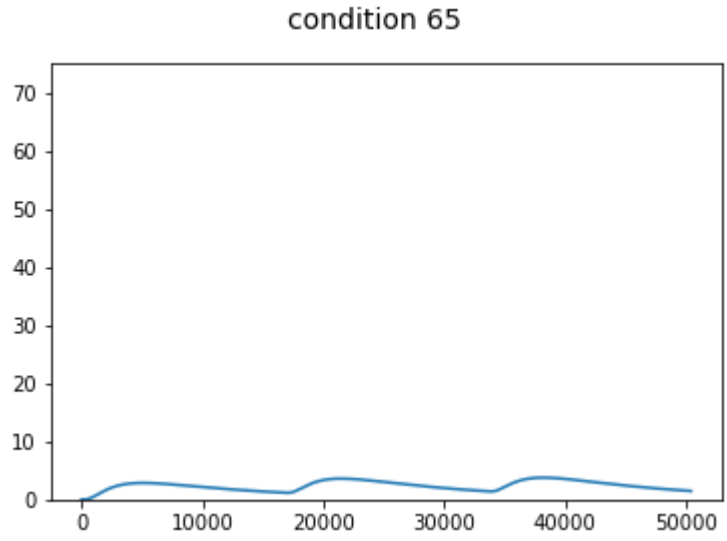

condition 60
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

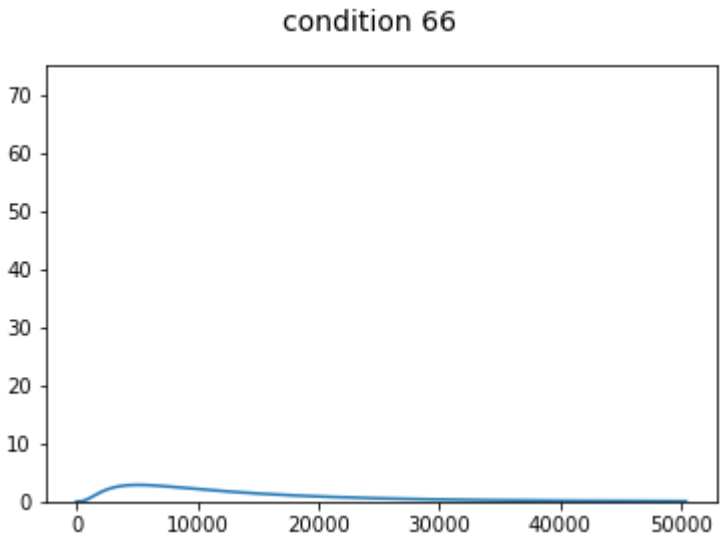

condition 61
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

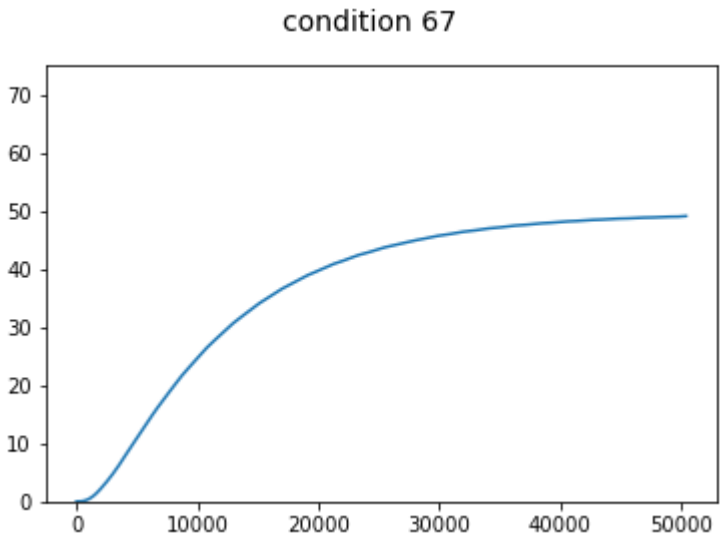

condition 62
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

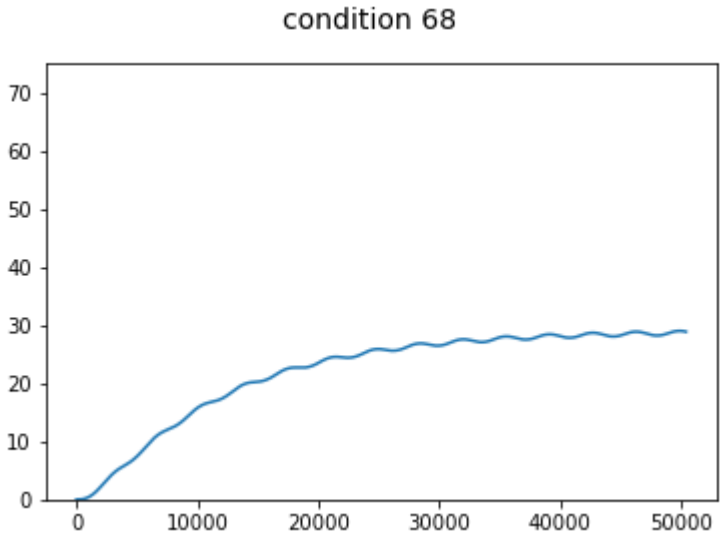

condition 63
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

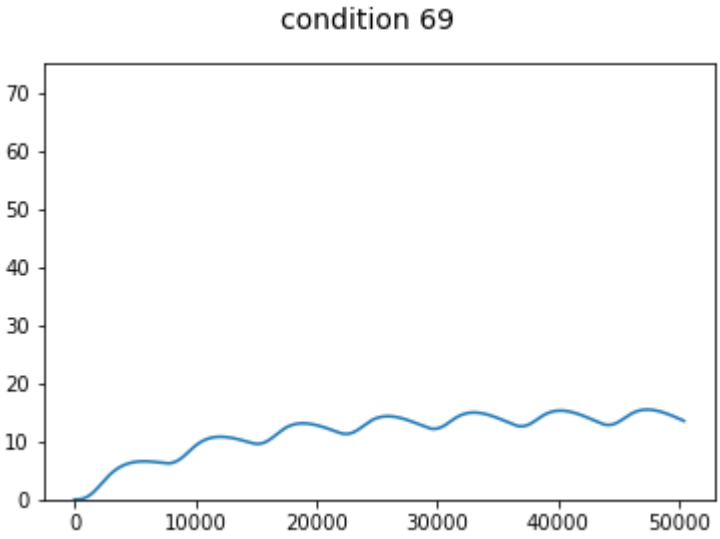

condition 64
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

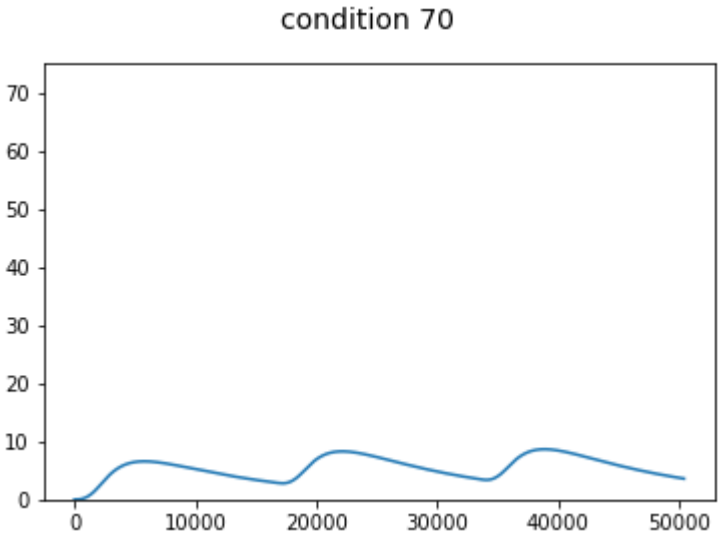

condition 65
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

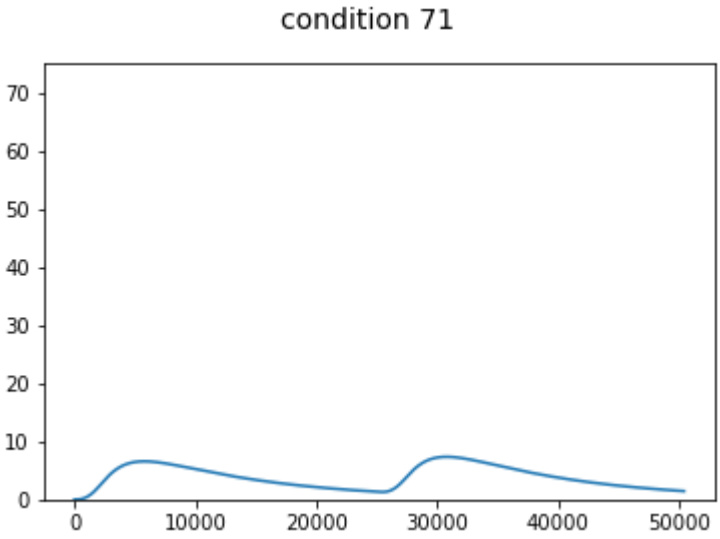

condition 66
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

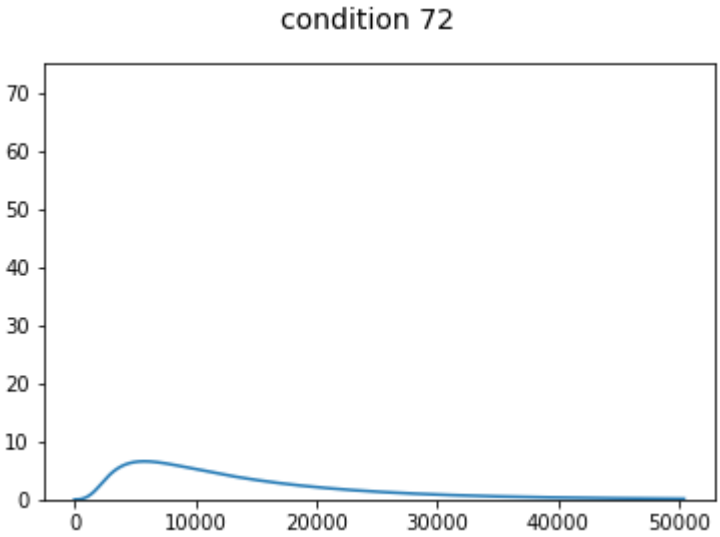

condition 67
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

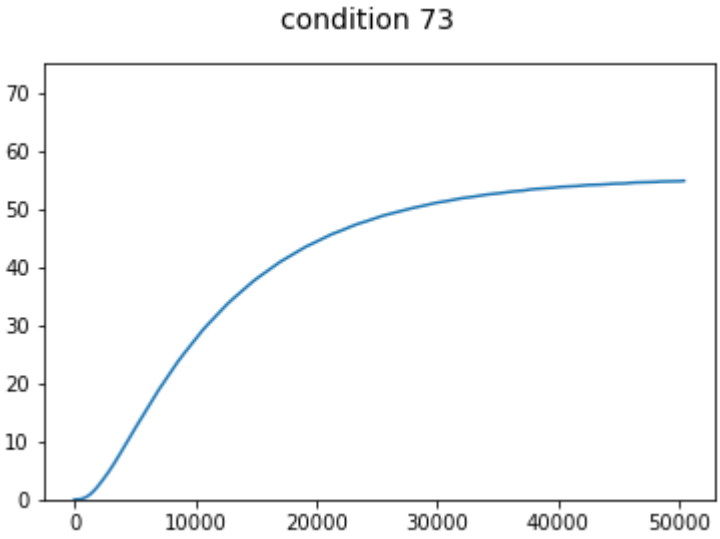

condition 68
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

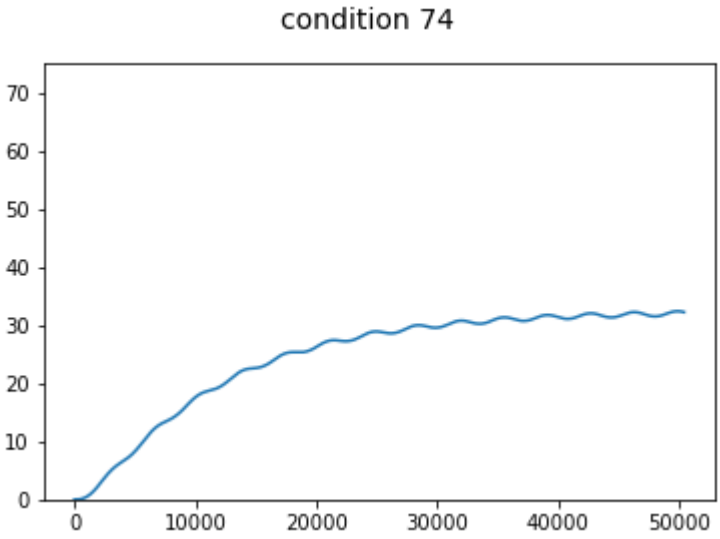

condition 69
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

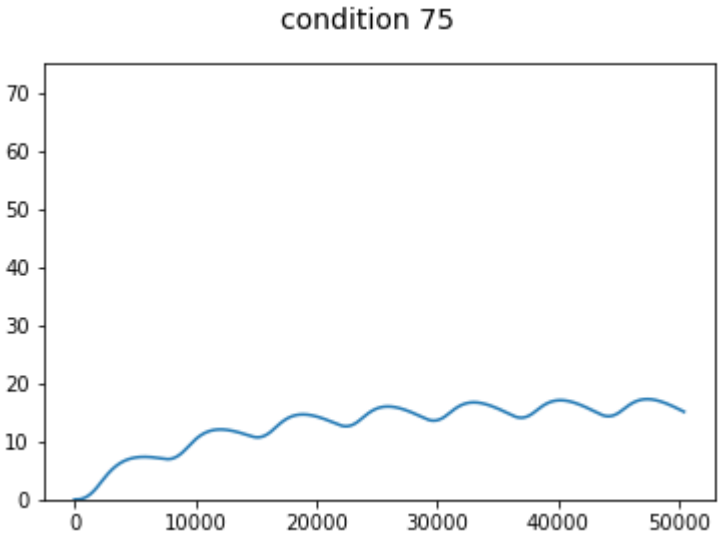

condition 70
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

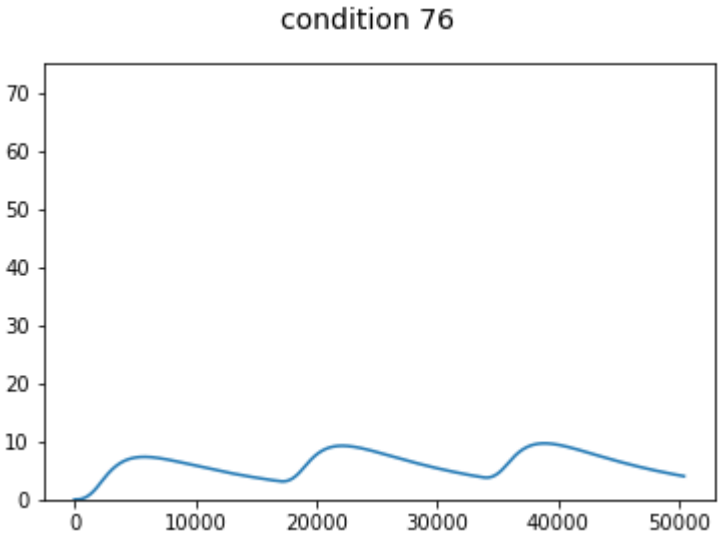

condition 71
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

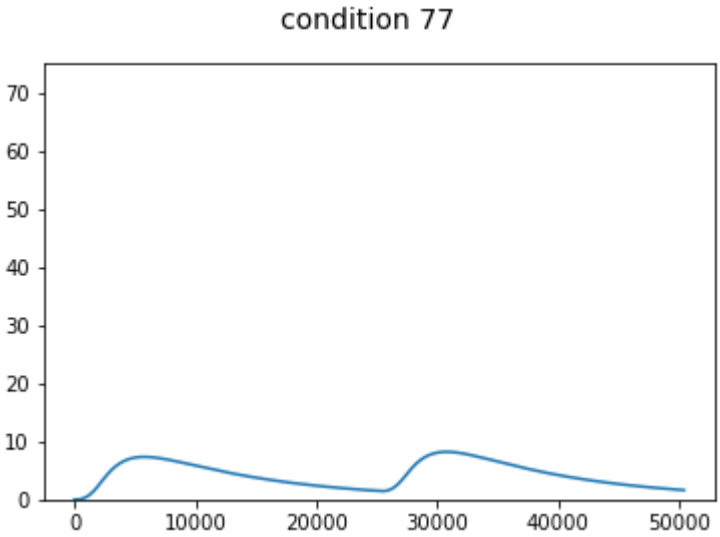

condition 72
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

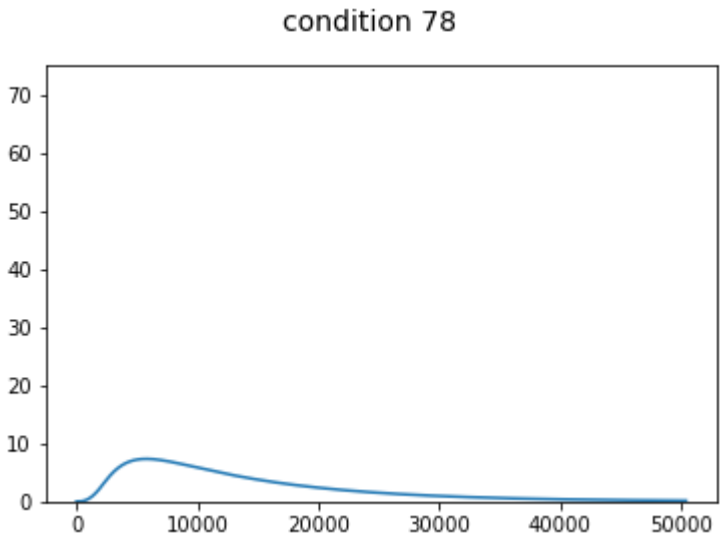

condition 73
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

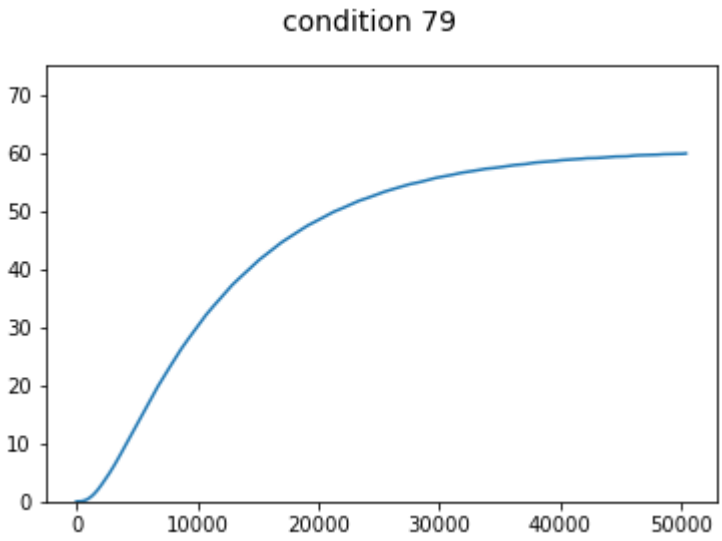

condition 74
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

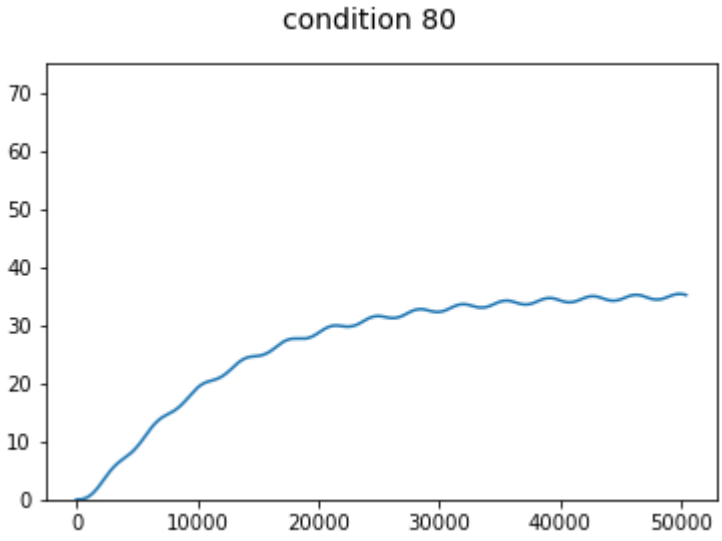

condition 75
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

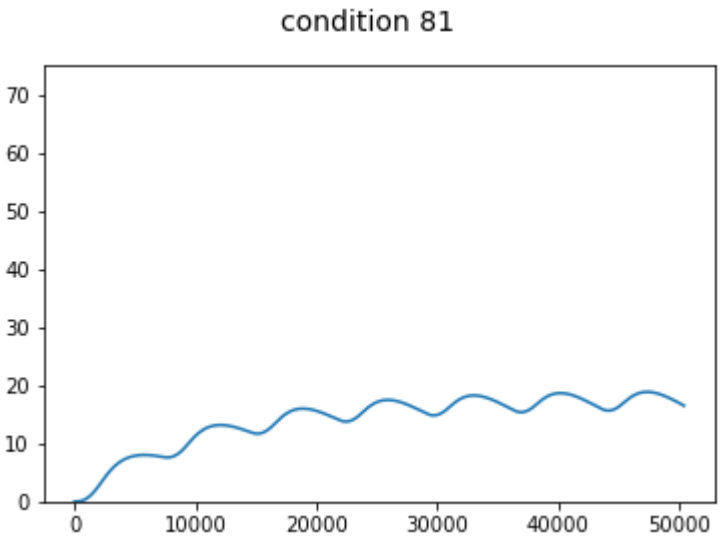

condition 76
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

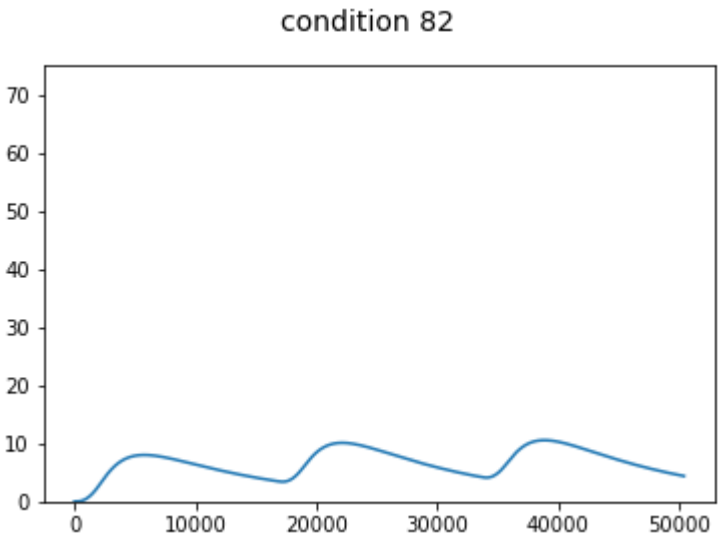

condition 77
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

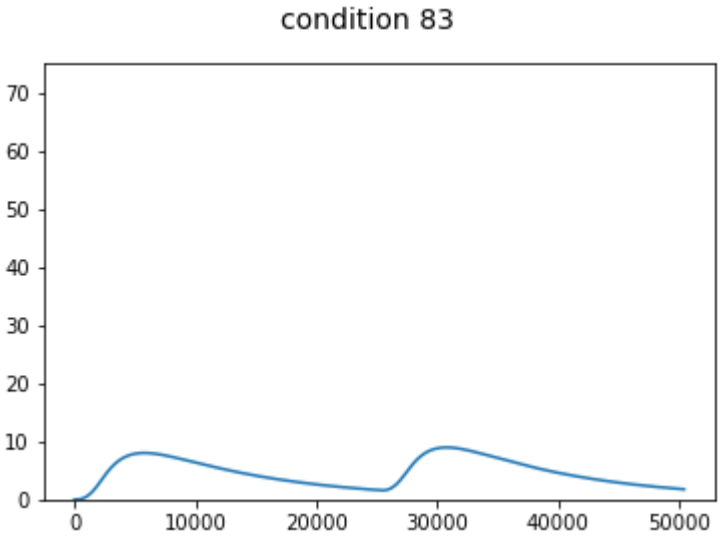

condition 78
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

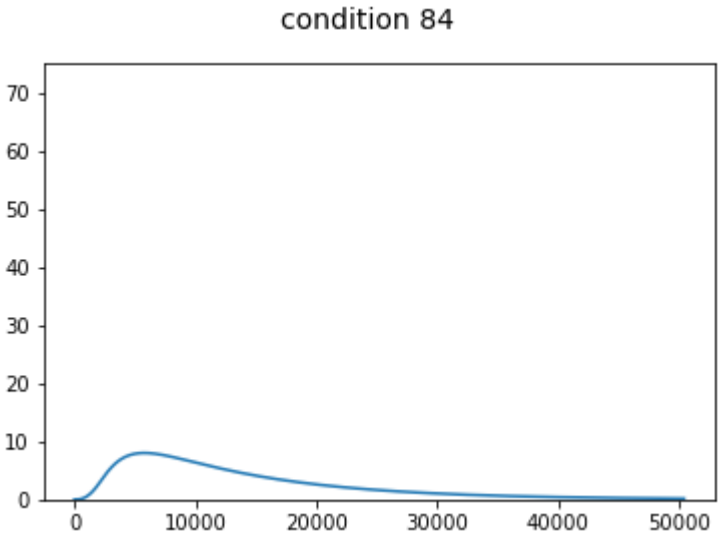

condition 79
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

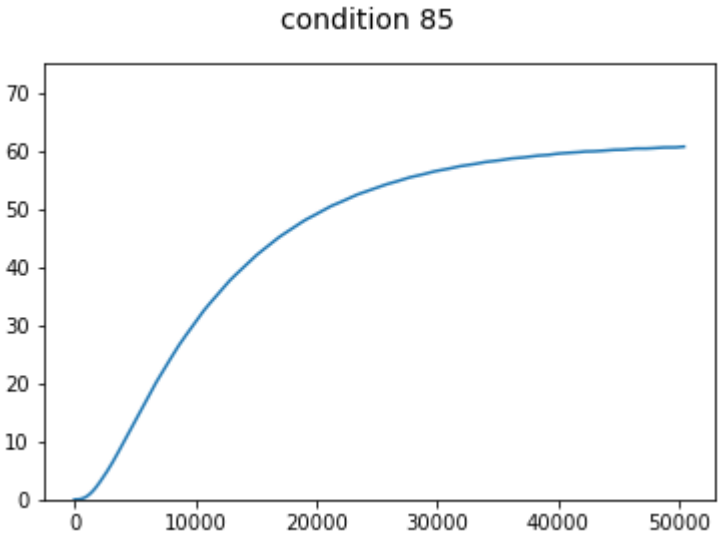

condition 80
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

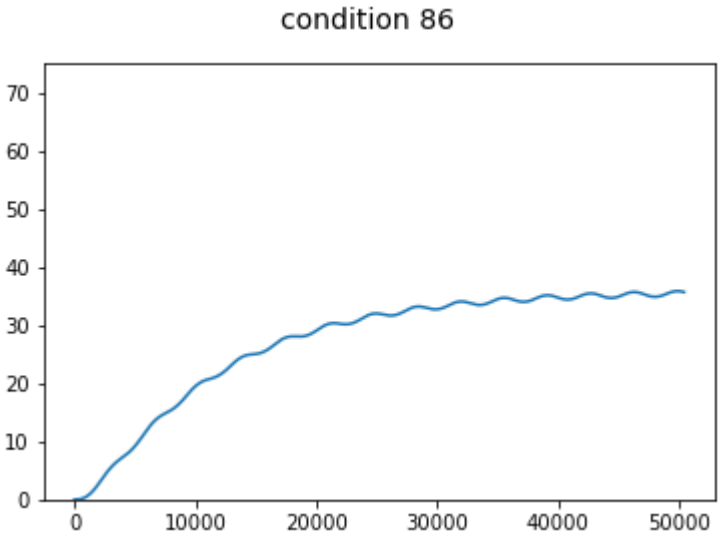

condition 81
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

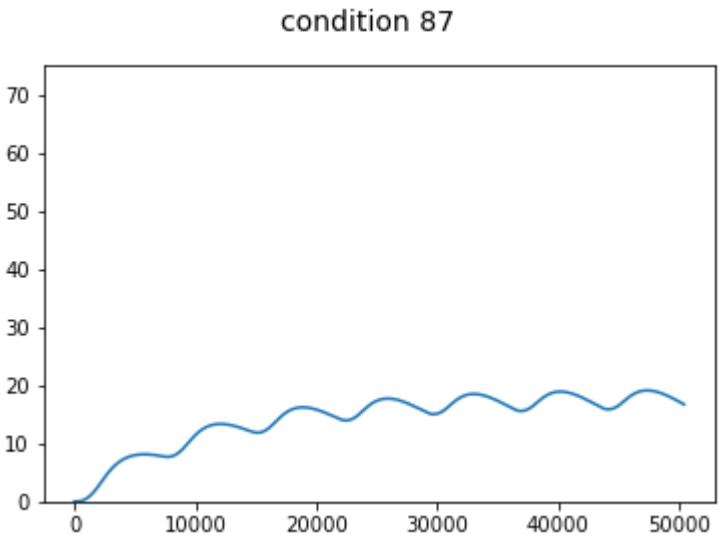

condition 82
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

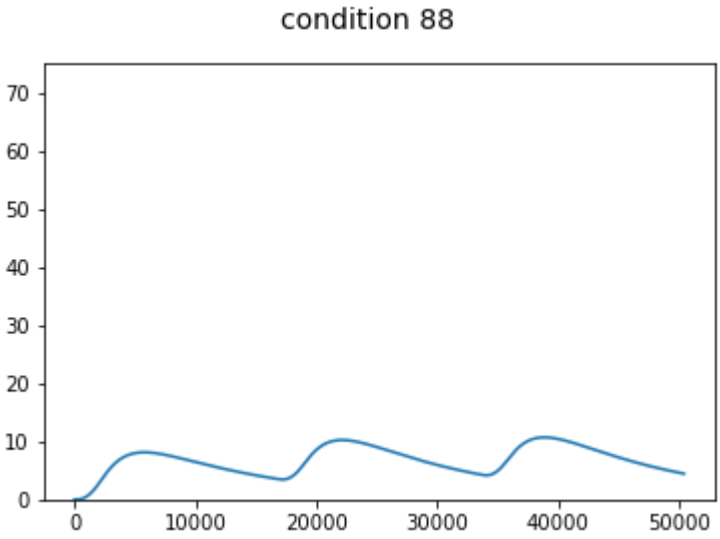

condition 83
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

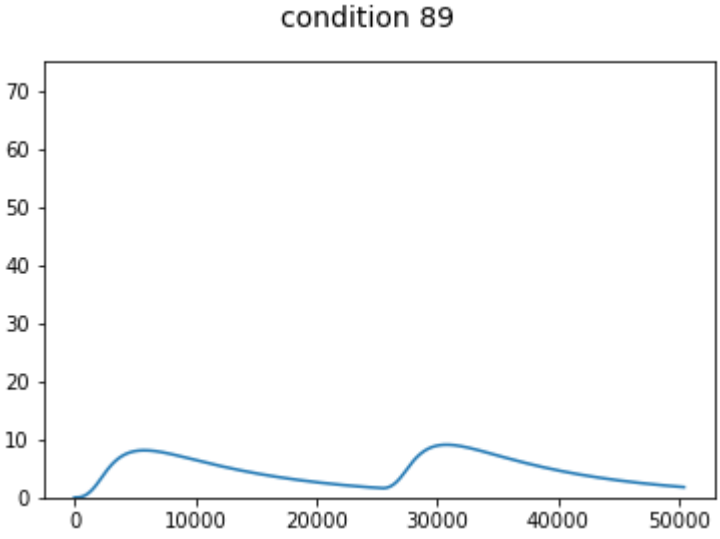

condition 84
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

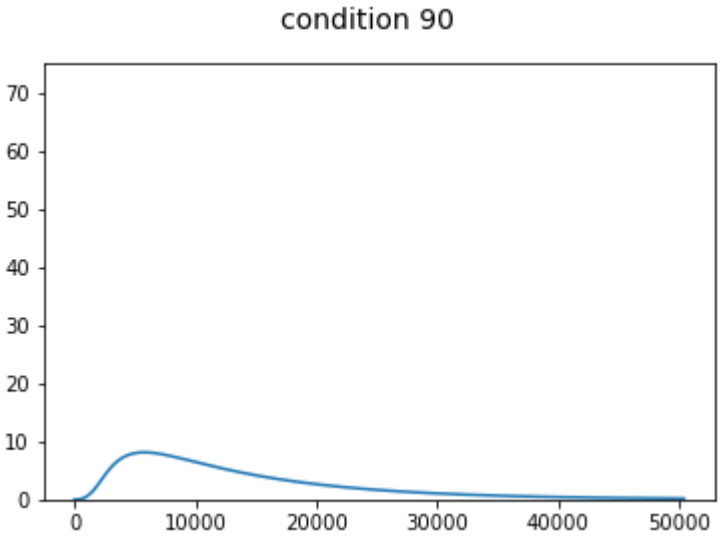

condition 85
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

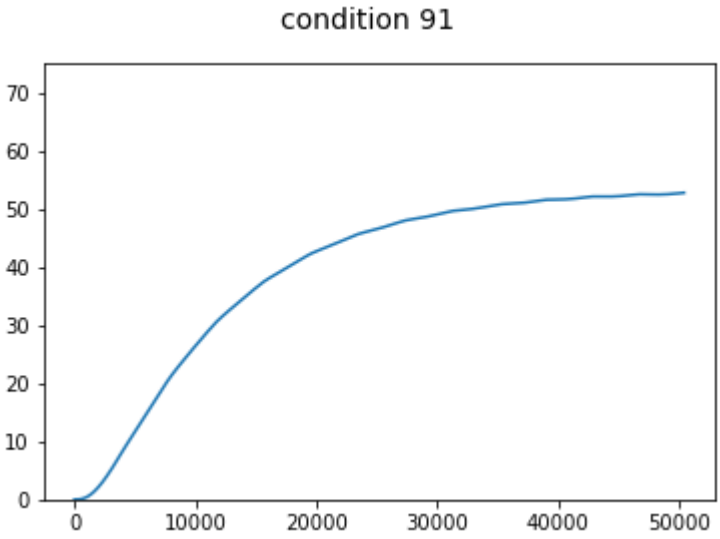

condition 86
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

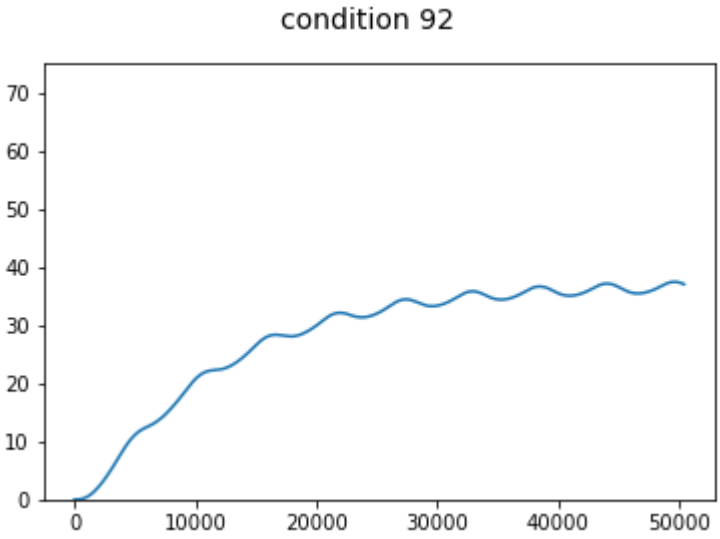

condition 87
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

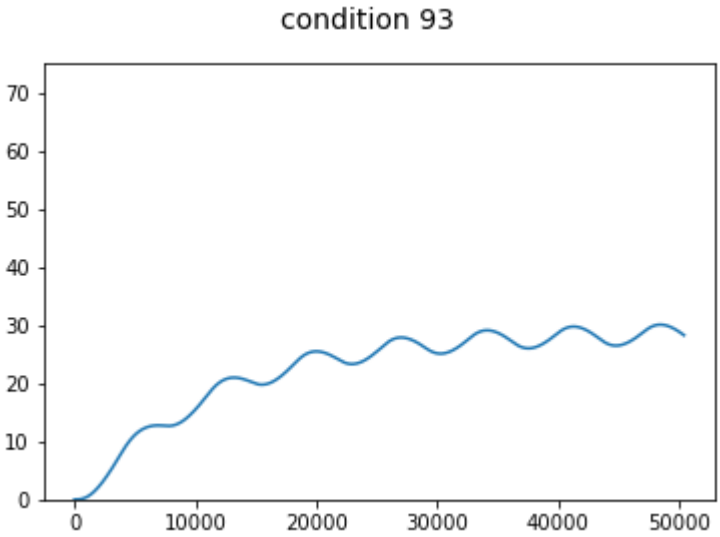

condition 88
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

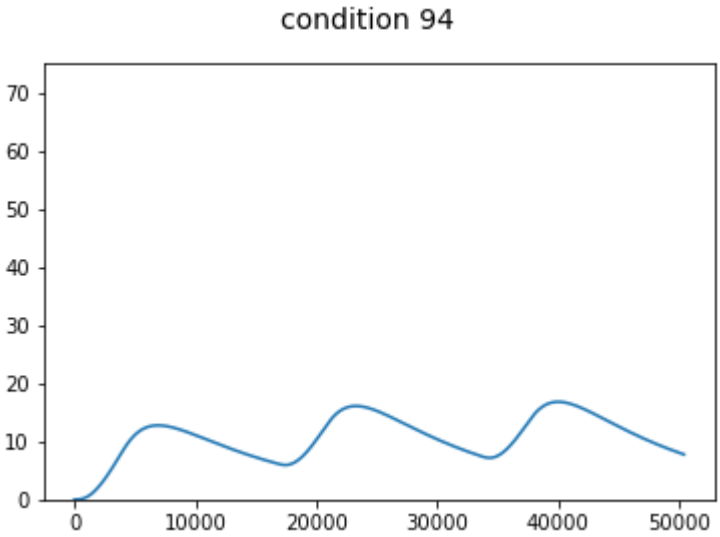

condition 89
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

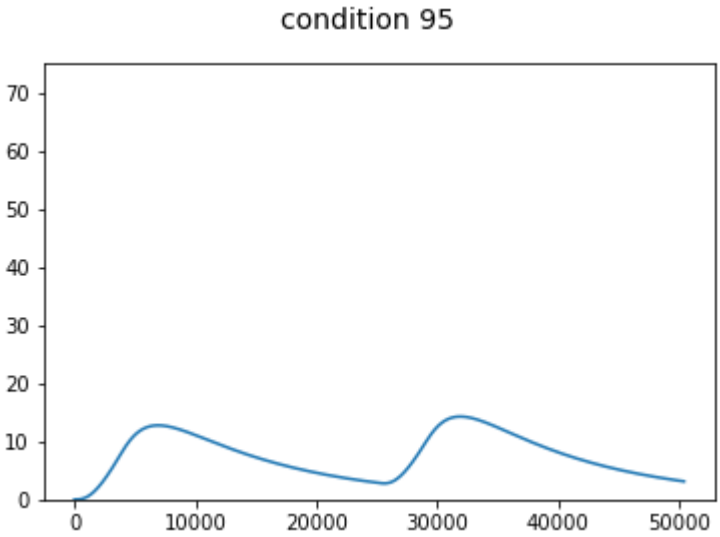

condition 90
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

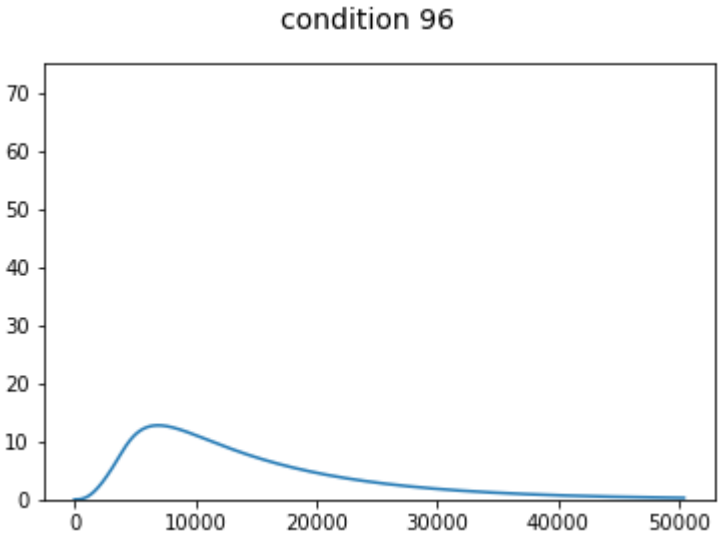

condition 91
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

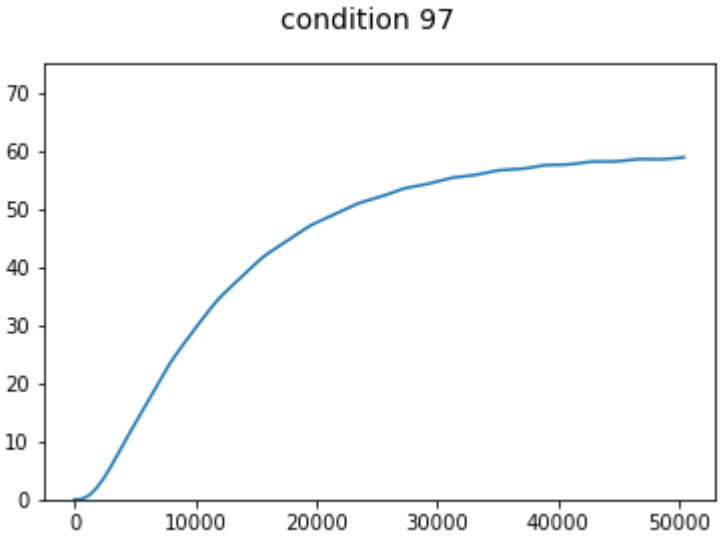

condition 92
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

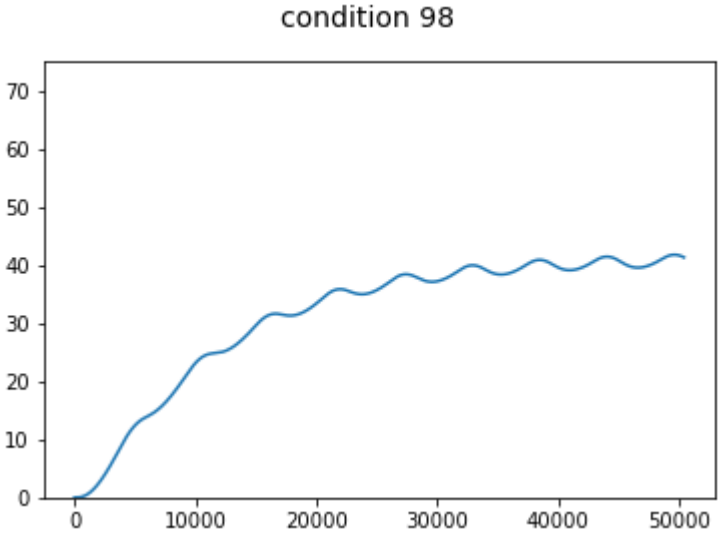

condition 93
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

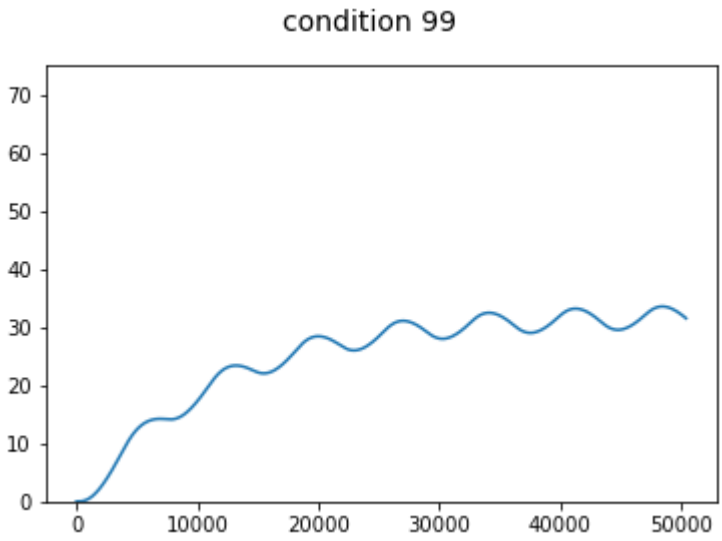

condition 94
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

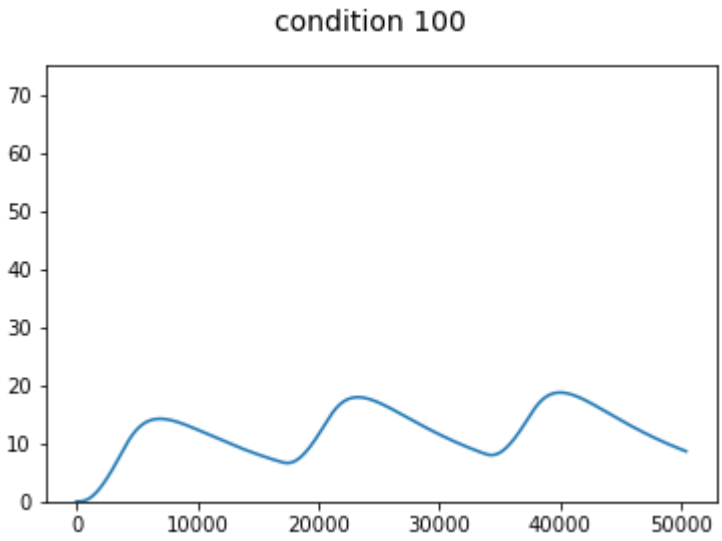

condition 95
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

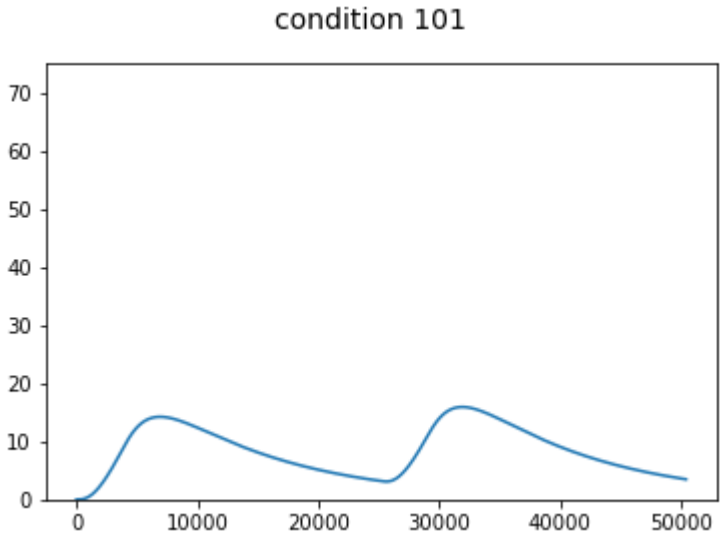

condition 96
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

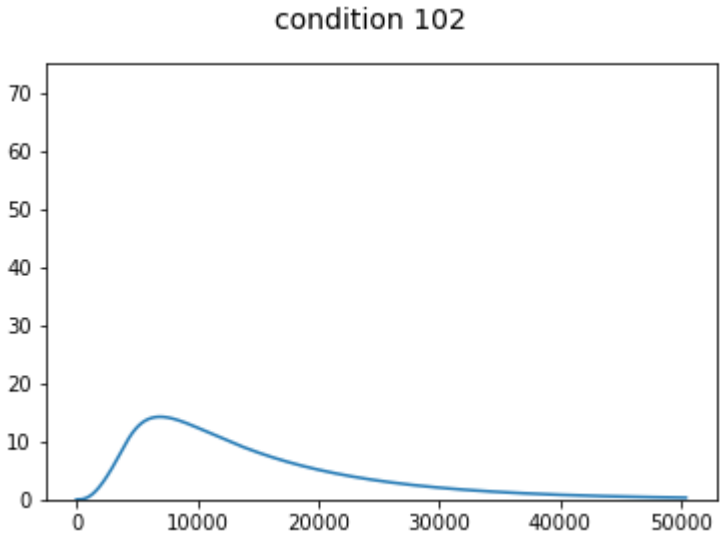

condition 97
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

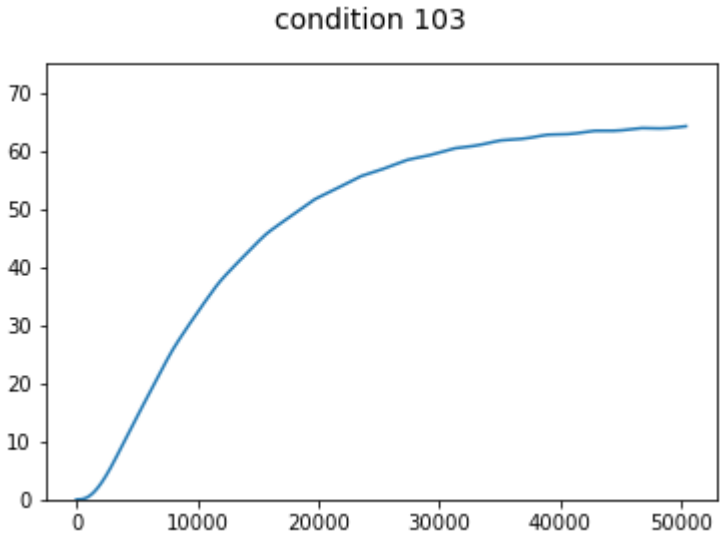

condition 98
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

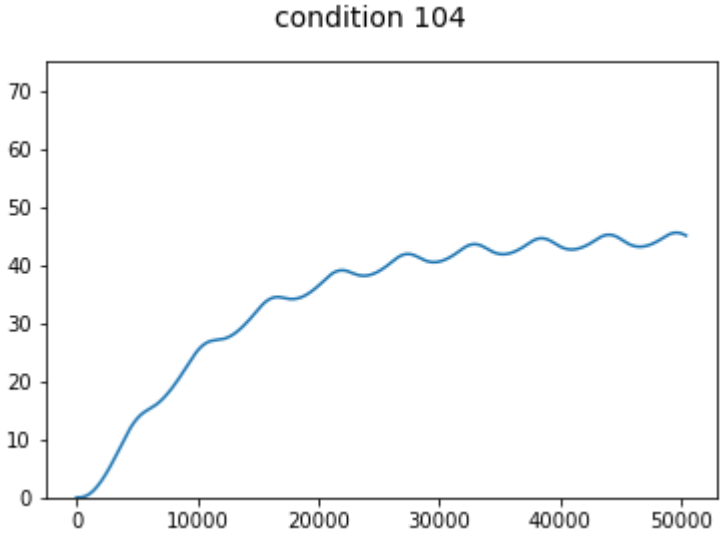

condition 99
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

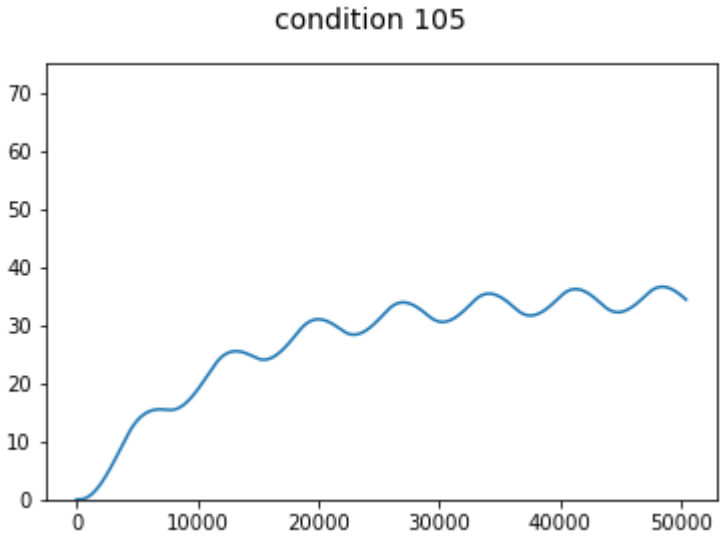

condition 100
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

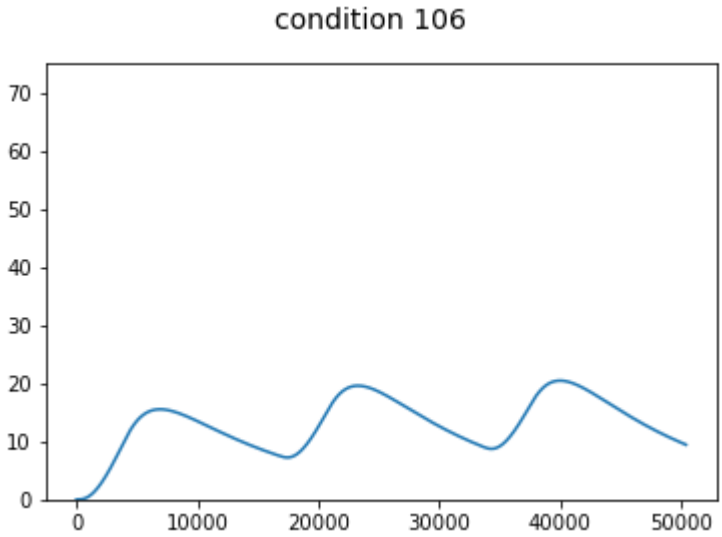

condition 101
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

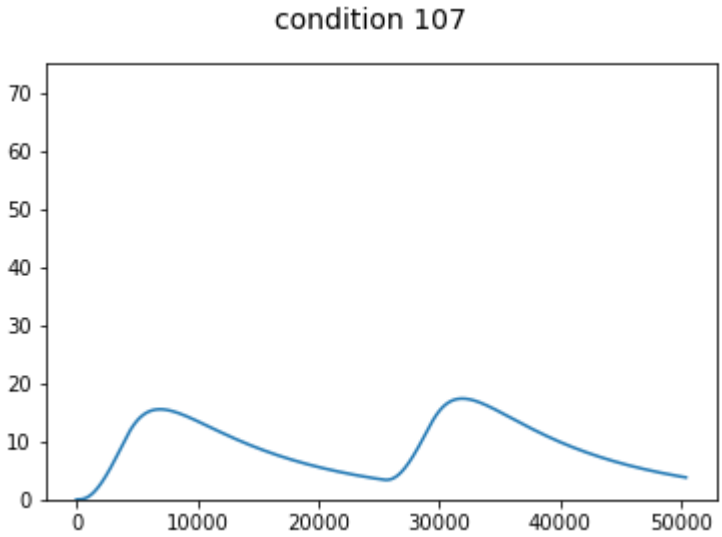

condition 102
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

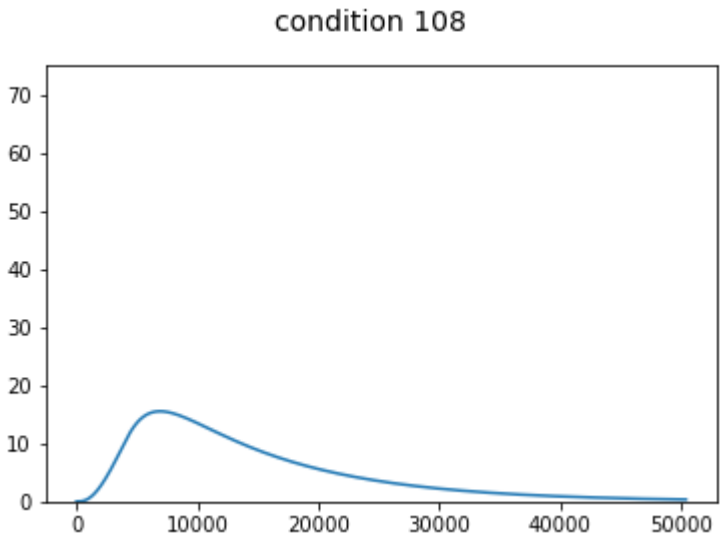

condition 103
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

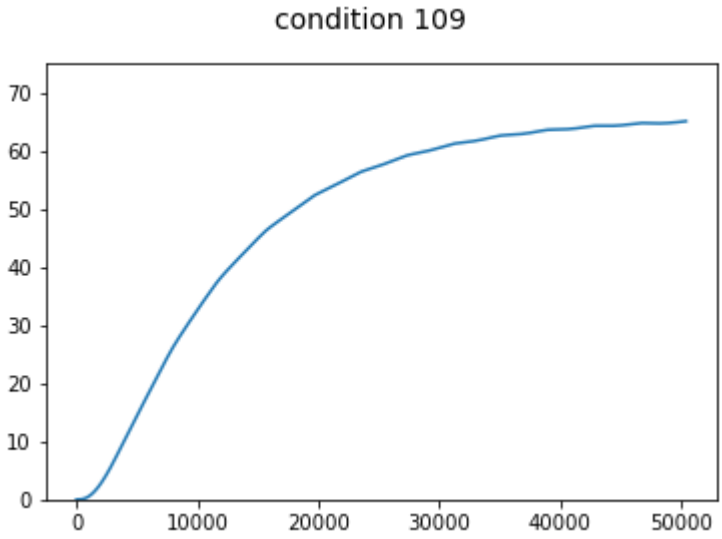

condition 104
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

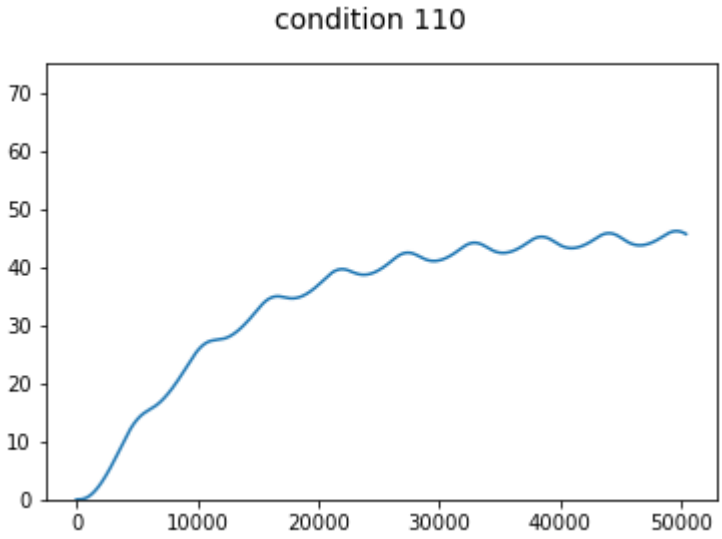

condition 105
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

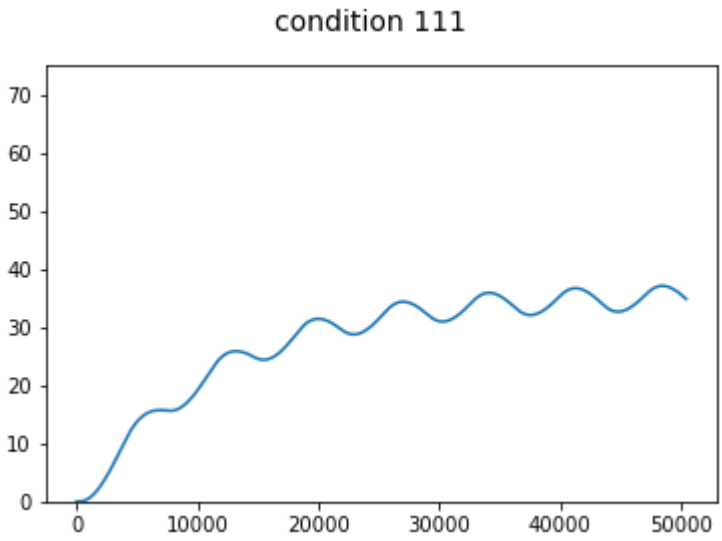

condition 106
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

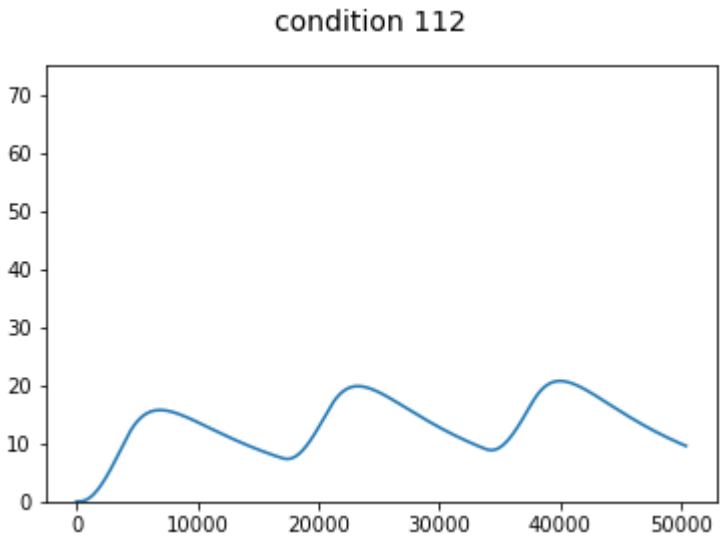

condition 107
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

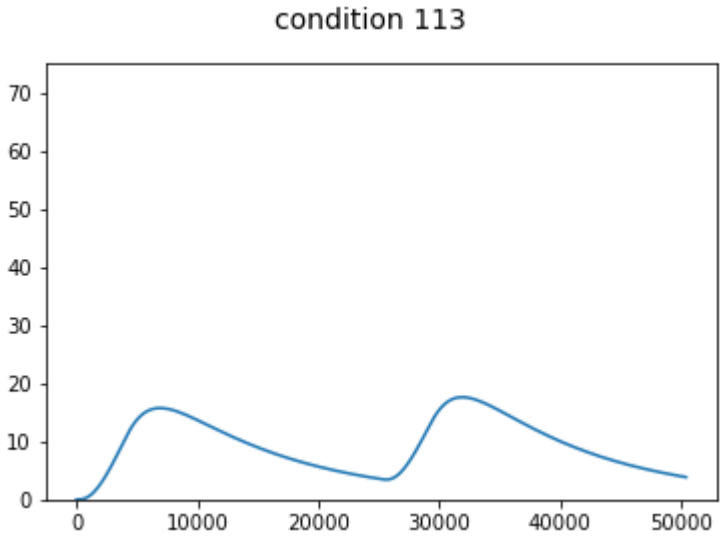

condition 108
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

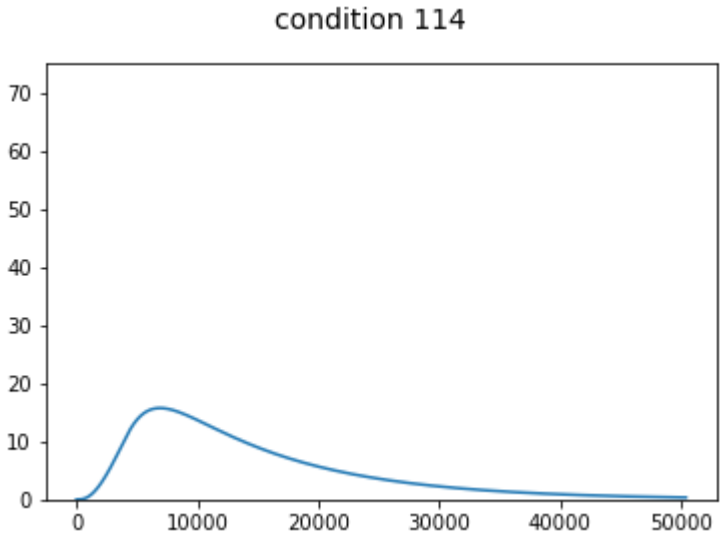

condition 109
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

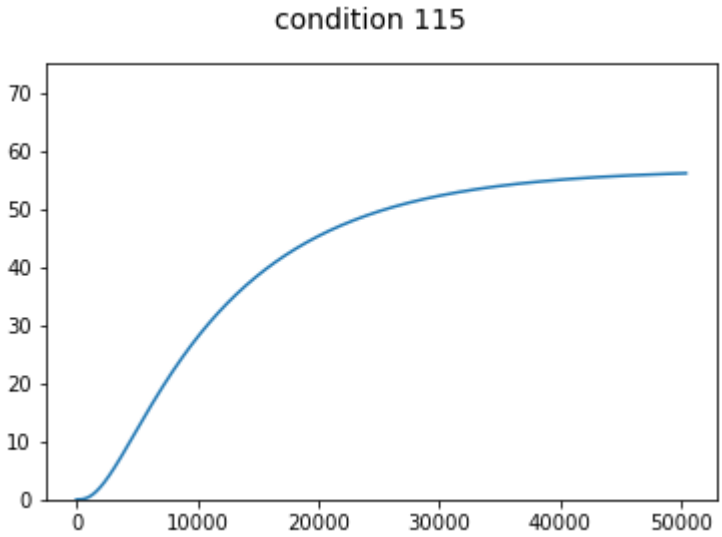

condition 110
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

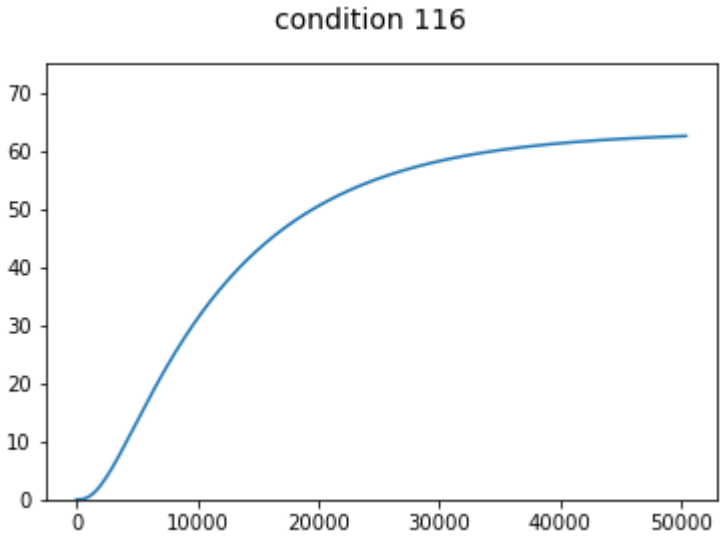

condition 111
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

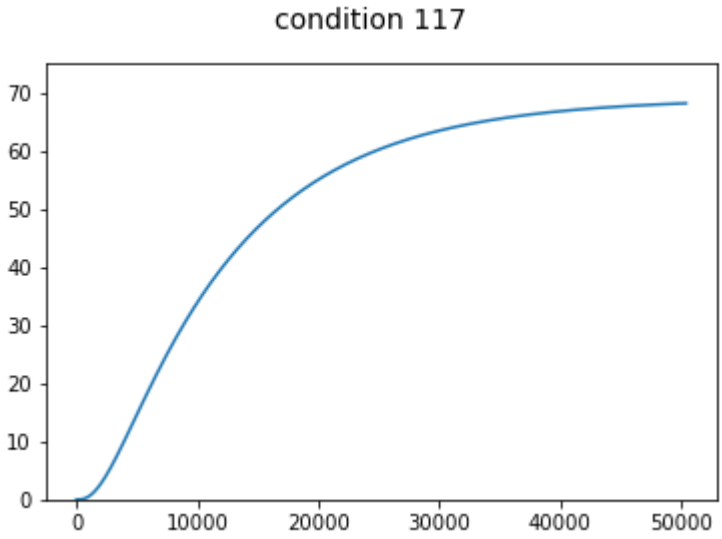

condition 112
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

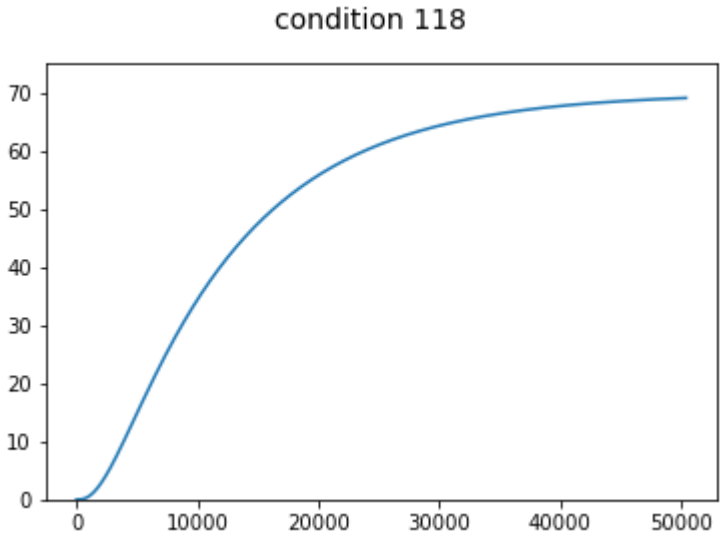

condition 113
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000


condition 114
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000


condition 115
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000


condition 116
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000


condition 117
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000


condition 118
70
60
50
40
30
20
10
0
0
10000
20000
30000
40000
50000

In [ ]:

In [ ]: