pulse2d_iter

March 22, 2019

In [1]:
```
# IMPORT PACKAGES

import pandas as pd
import numpy as np
import math
import pickle
```

In [2]:
```
#%% DEFINE FUNCTIONS


# FUNCTIONS
def time2node(x,Rt,Vw):
    t = (Rt*x)/Vw
    return t

def peakC(t,C0,A,Dx,Dy):
    Cmax = (C0*A)/(4*t*math.pi*math.sqrt(Dx*Dy)) ** 0.0001  # convert output value to pe
    return Cmax

def plumeDim(t,D):
    sigma3 = 3*math.sqrt(2*D*t)
    return sigma3
```

In [3]:
```
# read in input parameters
inputs = pd.read_csv('pulse2d_iterinput.txt',sep='\t')
inputs.head() # print out first few rows to console
```

Out[3]:

	Dx	Dy	Vw	C0	A	Rt
0	1.000000	0.100000	1.000000	10000.00000	10.000000	1.000000
1	1.184937	0.781501	1.197131	13135.64650	11.836978	1.306512
2	0.464198	0.459707	0.376031	13134.75488	11.781062	0.539950
3	0.589623	0.501242	0.572974	12551.96367	15.212112	0.476524
4	0.473860	0.627835	0.427449	13075.41568	14.650920	0.494132

In [4]:
```
# iteratively run the model for each system state initialization

datadict = {}
for row in range(inputs.shape[0]):
```

```
            Dx = inputs.loc[row,'Dx']
            Dy = inputs.loc[row,'Dy']
            Vw = inputs.loc[row,'Vw']
            C0 = inputs.loc[row,'C0']
            A = inputs.loc[row,'A']
            Rt = inputs.loc[row,'Rt']

            #Define model domain and interval size
            h = 5
            domain = np.linspace(h,100,20)

            # preallocate empty lists to store output
            t_out = []
            Cmax_out = []
            sigma3x_out = []
            sigma3y_out = []

            # Write for loop to iterate over model domain
            for x in domain:

                # Apply functions for each interation and store values
                t = time2node(x,Rt,Vw)  # calculate value
                t_out.append(t)  # store value
                Cmax = peakC(t,C0,A,Dx,Dy)
                Cmax_out.append(Cmax)
                sigma3x = plumeDim(t,Dx)
                sigma3x_out.append(sigma3x)
                sigma3y = plumeDim(t,Dy)
                sigma3y_out.append(sigma3y)

            # Write lists to dictionary then convert to dataframe...write to output file
            data = {'t': t_out,
                'Cmax': Cmax_out,
                'sigma3x': sigma3x_out,
                'sigma3y': sigma3y_out}
            df = pd.DataFrame.from_dict(data)
            datadict[row] = df

        pickle.dump(datadict, open( "pulse2d_iteroutput.pkl", "wb" ))
```

In [ ]: