kpp_vs_kppa

Unknown Author

December 15, 2014

In [7]:

```
kpp_file_1 = 'extended_box_model_Exa2Green/box_model.dat_OpenMP'
kpp_file_2 = 'extended_box_model_Exa2Green/kpp_input.dat'
```

In [8]:

```
%matplotlib inline
import re
from itertools import cycle
from pylab import *
from matplotlib.markers import MarkerStyle
import matplotlib.pyplot as plt

ATOL = 1.0e-2
RTOL = 1.0e-2
EPS = 2.220446049250313E-016
REGEX = re.compile('^([+\-]?)([0-9.]+)e?([+\-])([0-9.]+)$')
def convert(s):
    """
    Converts a number in Fortran E24.16 format to a Python float
    """
    m = re.search(REGEX, s)
    if m:
        s = ''.join([m.group(1), m.group(2), 'e', m.group(3), m.group(4)])
    try:
        fval = float(s)
    except ValueError:
        print '================> %s' % s
        fval = 0.0
    if fval < EPS:
        return 0.0
    else:
        return fval

def read_datfile(fname, tstart, cstart):
    """
    Read data from fname beginning on line tstart with concentration data beginning in
    Returns a tuple: (time, concentrations)
    Time data:
    [t0 t1 ... tN]
    Concentration data:
    [ [SPC_0(t0) SPC_1(t0) ... SPC_N(t0)]
    [SPC_0(t1) SPC_1(t1) ... SPC_N(t1)]
    : : :
    [SPC_0(tN) SPC_1(tN) ... SPC_N(tN)] ]
    """
    t = []
    c = []
    with open(fname, 'r') as f:
        while tstart:
            f.readline()
            tstart -= 1
        for line in f:
            parts = line.split()
            t.append(convert(parts[0]))
            c.append([convert(x) for x in parts[cstart:]])
    return t, c

def plot_dat(data, xlabel=’Time’, ylabel=’Conc’, names=None, titles=None):
    """
    Draw a plot of data read from read_datfile
    """
    lines = [’-’, ’--’, ’-.’, ’:’]
    markers = MarkerStyle.filled_markers
    linecycler = cycle(lines)
    markercycler = cycle(markers)
    datastyles = [’%s%s’ % (linecycler.next(), markercycler.next()) for _ in data]
    ndat = len(data)
    nspec = len(data[0][1][0])
    x = data[0][0]
    for i in xrange(0, nspec):
        fig, ax = plt.subplots()
        for j, dat in enumerate(data):
            t, c = dat
            y = [ct[i] for ct in c]
            style = datastyles[j]
            if names:
                label = ’%s’ % names[j]
            else:
                label = ’%d’ % j
            ax.plot(x, y, style, label=label)
        if ndat > 1:
            ax.legend(loc=2)
        ax.set_xlabel(xlabel)
        ax.set_ylabel(ylabel)
        if titles:
            ax.set_title(titles[i])
        else:
            ax.set_title(’Species %d’ % i)
        show()

def scaled_err(x, y):
    if x or y:
        return abs(x-y)/max(x, y)
    elif x == y:
        return 0.0
    else:
        return float(’inf’)

def calc_err(d0, d1):
    c0 = d0[1]
    c1 = d1[1]
    err = []
    nsteps = len(c0)
    nspec = len(c0[0])
    sigPow = 0.0
    errPow = 0.0
    errCount = 0.0
    for i in xrange(0, nsteps):
        e = []
        for j in xrange(0, nspec):
            x = c0[i][j]
            y = c1[i][j]
            sigPow += x*x
            errPow += (x-y)*(x-y)
            serr = scaled_err(x,y)
            if serr > RTOL:
                print ’%g > %g: %g, %g’ % (serr, RTOL, x, y)
                errCount += 1
            e.append(serr)
        err.append(e)
    if errPow > 0:
        snr = 20 * log10(sigPow / errPow)
    else:
        snr = float('inf')
    print 'SNR: %fdb' % snr
    if errCount:
        print '%d samples with relative error > %g' % (errCount, RTOL)
    return d1[0], err
```

In [9]:

```
kpp_dat_1 = read_datfile(kpp_file_1, 0, 1)
kpp_dat_2 = read_datfile(kpp_file_2, 0, 1)
```

In [10]:

```
err_dat = calc_err(kpp_dat_1, kpp_dat_2)
```

```
0.78462 > 0.01: 0.000414383, 8.925e-05
0.251905 > 0.01: 0.0309453, 0.02315
0.924368 > 0.01: 1.84164e-07, 2.435e-06
0.156043 > 0.01: 7.24621, 8.586
0.999908 > 0.01: 1.87512e-05, 0.2041
0.167375 > 0.01: 19.8529, 16.53
0.775835 > 0.01: 0.00521045, 0.001168
0.464558 > 0.01: 0.000545344, 0.000292
0.11317 > 0.01: 0.0234994, 0.02084
0.0305836 > 0.01: 0.639621, 0.6598
0.859789 > 0.01: 2.33171e-07, 1.663e-06
0.133437 > 0.01: 8.44119, 9.741
0.999948 > 0.01: 9.39141e-06, 0.1805
0.121625 > 0.01: 16.0979, 14.14
0.560113 > 0.01: 0.00352136, 0.001549
0.358367 > 0.01: 0.000768352, 0.000493
0.0213191 > 0.01: 0.0211305, 0.02068
0.0639684 > 0.01: 0.746663, 0.6989
0.874372 > 0.01: 1.62814e-07, 1.296e-06
0.0682571 > 0.01: 9.63422, 10.34
0.999939 > 0.01: 1.08867e-05, 0.1778
0.0621084 > 0.01: 13.7969, 12.94
0.507928 > 0.01: 0.00346494, 0.001705
0.268473 > 0.01: 0.000861759, 0.0006304
0.0405987 > 0.01: 0.0208985, 0.02005
0.0866803 > 0.01: 0.7868, 0.7186
0.886132 > 0.01: 1.29696e-07, 1.139e-06
0.0516856 > 0.01: 10.2513, 10.81
0.99994 > 0.01: 1.19995e-05, 0.1997
0.0542259 > 0.01: 12.614, 11.93
0.507633 > 0.01: 0.00333085, 0.00164
0.0759751 > 0.01: 0.000672783, 0.0007281
0.041029 > 0.01: 0.0202509, 0.01942
0.153526 > 0.01: 0.866181, 0.7332
0.734262 > 0.01: 4.15071e-06, 1.103e-06
0.078725 > 0.01: 10.6592, 11.57
0.594032 > 0.01: 0.68429, 0.2778
0.165078 > 0.01: 12.6718, 10.58
0.820622 > 0.01: 0.000248259, 0.001384
0.025192 > 0.01: 0.000725168, 0.0007069
0.0772677 > 0.01: 0.0196482, 0.01813
0.0808036 > 0.01: 0.760447, 0.699
```

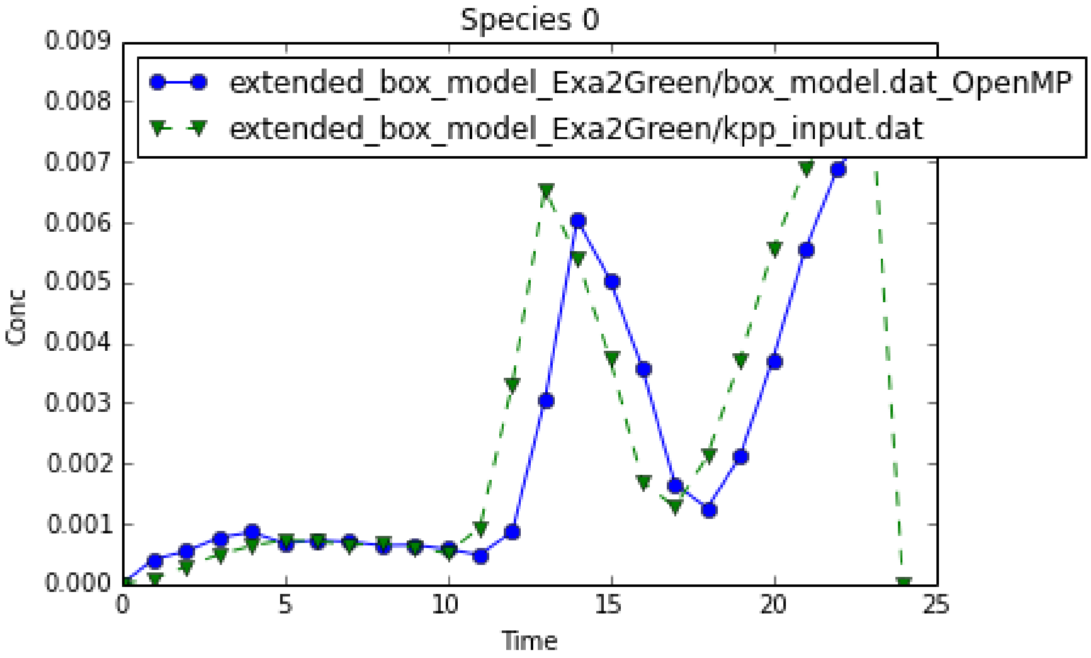
```
0.599182 > 0.01: 4.91745e-06, 1.971e-06
0.302527 > 0.01: 8.79514, 12.61
0.704002 > 0.01: 3.3784, 1
0.367857 > 0.01: 13.9921, 8.845
0.82985 > 0.01: 2.44335e-05, 0.0001436
0.0724426 > 0.01: 0.000699579, 0.0006489
0.116286 > 0.01: 0.0185015, 0.01635
0.715189 > 0.01: 1.05544e-05, 3.006e-06
0.419369 > 0.01: 7.43208, 12.8
0.066504 > 0.01: 6.11036, 5.704
0.458999 > 0.01: 14.2625, 7.716
0.163317 > 0.01: 1.15044e-05, 1.375e-05
0.0410451 > 0.01: 0.000636842, 0.0006641
0.236976 > 0.01: 0.0168094, 0.02203
0.0826735 > 0.01: 0.763307, 0.8321
0.658669 > 0.01: 1.73761e-05, 5.931e-06
0.30132 > 0.01: 7.87412, 11.27
0.158197 > 0.01: 10.4336, 8.783
0.155271 > 0.01: 13.0456, 11.02
0.401285 > 0.01: 6.73554e-06, 1.125e-05
0.0653132 > 0.01: 0.000645457, 0.0006033
0.0734441 > 0.01: 0.0221725, 0.02393
0.0605519 > 0.01: 0.862208, 0.81
0.53553 > 0.01: 2.54484e-05, 1.182e-05
0.142428 > 0.01: 8.08604, 9.429
0.311902 > 0.01: 11.6887, 8.043
0.0299222 > 0.01: 14.8034, 15.26
0.499448 > 0.01: 7.40817e-06, 1.48e-05
0.115319 > 0.01: 0.00057874, 0.000512
0.0771147 > 0.01: 0.0238383, 0.022
0.0986819 > 0.01: 0.839992, 0.7571
0.448474 > 0.01: 3.7913e-05, 2.091e-05
0.0476548 > 0.01: 7.39306, 7.763
0.375045 > 0.01: 9.70789, 6.067
0.109861 > 0.01: 18.0965, 20.33
0.535082 > 0.01: 1.04746e-05, 2.253e-05
0.479712 > 0.01: 0.000483816, 0.0009299
0.0748455 > 0.01: 0.0218059, 0.02357
0.0696923 > 0.01: 0.798768, 0.7431
0.455563 > 0.01: 5.43865e-05, 2.961e-05
0.0989979 > 0.01: 6.37819, 7.079
0.238729 > 0.01: 6.97912, 5.313
0.0162993 > 0.01: 22.7137, 23.09
0.412867 > 0.01: 1.62284e-05, 2.764e-05
0.737827 > 0.01: 0.000865957, 0.003303
0.284769 > 0.01: 0.0232665, 0.03253
0.104637 > 0.01: 0.807382, 0.7229
0.463999 > 0.01: 6.66976e-05, 3.575e-05
0.0780893 > 0.01: 5.96937, 6.475
0.226909 > 0.01: 5.85184, 4.524
0.388039 > 0.01: 2.08128e-05, 3.401e-05
0.53524 > 0.01: 0.00304092, 0.006543
0.298752 > 0.01: 0.0319489, 0.04556
```

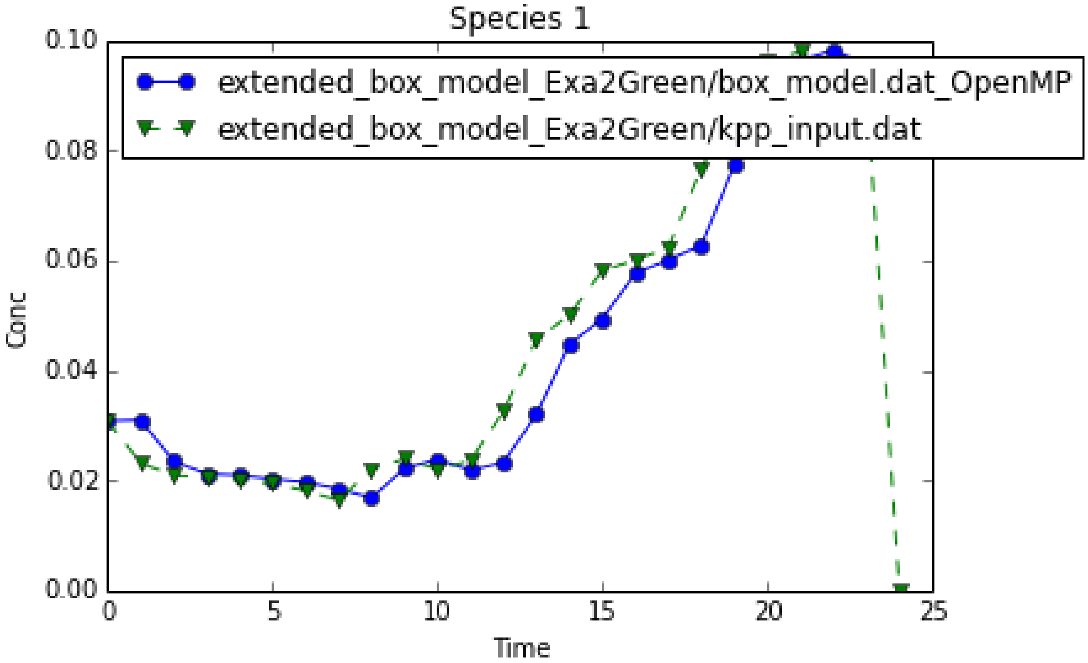
```
0.135193 > 0.01: 0.795207, 0.6877
0.523865 > 0.01: 7.11143e-05, 3.386e-05
0.0818664 > 0.01: 5.56205, 6.058
0.244983 > 0.01: 4.83698, 3.652
0.0434189 > 0.01: 27.4728, 26.28
0.405886 > 0.01: 2.57845e-05, 4.34e-05
0.106166 > 0.01: 0.00603356, 0.005393
0.107222 > 0.01: 0.0447192, 0.05009
0.115288 > 0.01: 0.75697, 0.6697
0.510517 > 0.01: 6.57429e-05, 3.218e-05
0.126566 > 0.01: 5.16811, 5.917
0.0773828 > 0.01: 4.00166, 3.692
0.120364 > 0.01: 28.3754, 24.96
0.250985 > 0.01: 3.00804e-05, 4.016e-05
0.258079 > 0.01: 0.00504636, 0.003744
0.149661 > 0.01: 0.0494047, 0.0581
0.0963756 > 0.01: 0.741016, 0.6696
0.526691 > 0.01: 4.77278e-05, 2.259e-05
0.19141 > 0.01: 5.36095, 6.63
0.016819 > 0.01: 3.79788, 3.734
0.131368 > 0.01: 26.5475, 23.06
0.256655 > 0.01: 3.09975e-05, 4.17e-05
0.525496 > 0.01: 0.00357426, 0.001696
0.0408642 > 0.01: 0.0576537, 0.06011
0.06446 > 0.01: 0.743421, 0.6955
0.460762 > 0.01: 2.93191e-05, 1.581e-05
0.223514 > 0.01: 6.26081, 8.063
0.213588 > 0.01: 3.74411, 4.761
0.162114 > 0.01: 24.2276, 20.3
0.0378427 > 0.01: 3.356e-05, 3.488e-05
0.226174 > 0.01: 0.00165153, 0.001278
0.0347653 > 0.01: 0.0601148, 0.06228
0.0672391 > 0.01: 0.779085, 0.7267
0.645739 > 0.01: 1.48393e-05, 5.257e-06
0.357131 > 0.01: 8.58872, 13.36
0.0439468 > 0.01: 3.94957, 3.776
0.334421 > 0.01: 20.3432, 13.54
0.272254 > 0.01: 3.69404e-05, 5.076e-05
0.406099 > 0.01: 0.00126145, 0.002124
0.182437 > 0.01: 0.0627397, 0.07674
0.0218839 > 0.01: 0.816161, 0.7983
0.49398 > 0.01: 6.70329e-06, 3.392e-06
0.251546 > 0.01: 13.1503, 17.57
0.125286 > 0.01: 3.76123, 3.29
0.319935 > 0.01: 14.1016, 9.59
0.232882 > 0.01: 4.08107e-05, 5.32e-05
0.431675 > 0.01: 0.0021136, 0.003719
0.118758 > 0.01: 0.0773026, 0.08772
0.0472864 > 0.01: 0.863349, 0.9062
0.028794 > 0.01: 2.13652e-06, 2.075e-06
0.157275 > 0.01: 19.349, 22.96
0.293913 > 0.01: 1.35781, 1.923
0.282963 > 0.01: 7.90755, 5.67
```

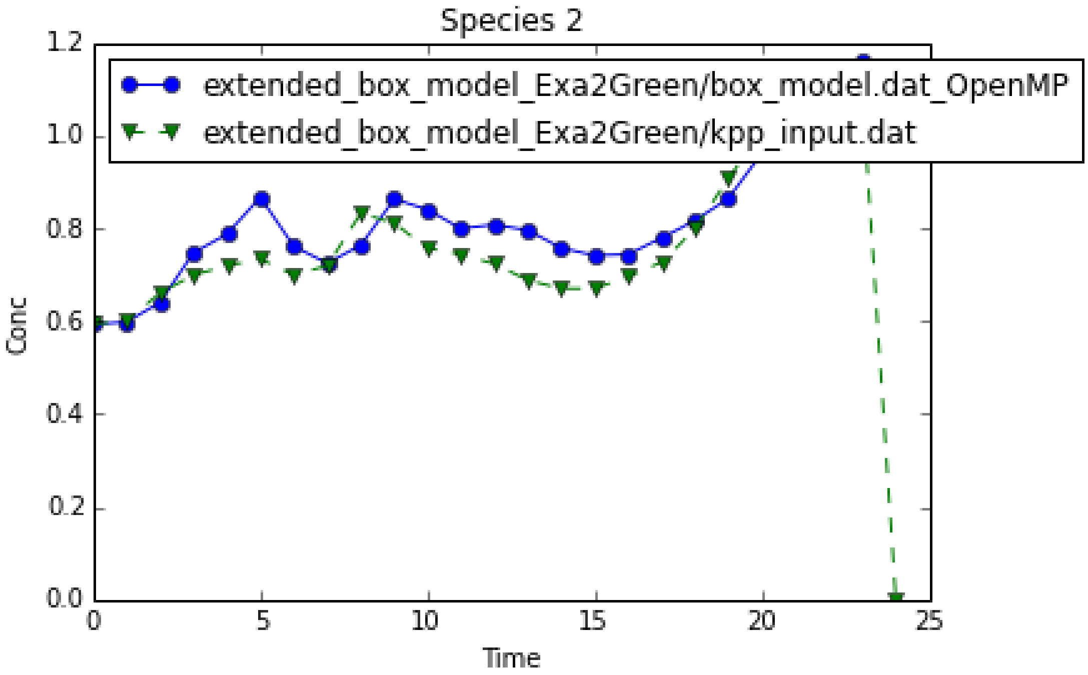
```
0.0847264 > 0.01: 8.68483e-05, 7.949e-05
0.328604 > 0.01: 0.00371819, 0.005538
0.0843831 > 0.01: 0.0880823, 0.0962
0.0458116 > 0.01: 0.960868, 1.007
0.819579 > 0.01: 3.08519e-07, 1.71e-06
0.0222349 > 0.01: 24.7277, 25.29
0.984764 > 0.01: 0.0225043, 1.477
0.0770713 > 0.01: 3.75817, 4.072
0.705702 > 0.01: 0.000331467, 9.755e-05
0.196335 > 0.01: 0.00554046, 0.006894
0.0153221 > 0.01: 0.0964394, 0.09794
0.0132493 > 0.01: 1.06174, 1.076
0.876951 > 0.01: 1.81005e-07, 1.471e-06
0.051018 > 0.01: 26.6496, 25.29
0.993414 > 0.01: 0.00815286, 1.238
0.301809 > 0.01: 2.58051, 3.696
0.593806 > 0.01: 0.000287301, 0.0001167
0.0817868 > 0.01: 0.00689578, 0.00751
0.0259813 > 0.01: 0.0981501, 0.0956
0.0221184 > 0.01: 1.13204, 1.107
0.883108 > 0.01: 1.61428e-07, 1.381e-06
0.0933045 > 0.01: 26.4367, 23.97
0.991008 > 0.01: 0.00920787, 1.024
0.399579 > 0.01: 2.45392, 4.087
0.390014 > 0.01: 0.000293777, 0.0001792
0.0991555 > 0.01: 0.00752025, 0.008348
0.034892 > 0.01: 0.0958027, 0.09246
0.058452 > 0.01: 1.15979, 1.092
0.888557 > 0.01: 1.43984e-07, 1.292e-06
0.0881512 > 0.01: 24.8835, 22.69
0.999152 > 0.01: 0.00069193, 0.8158
0.344046 > 0.01: 3.02461, 4.611
0.394162 > 0.01: 0.000451936, 0.0002738
1 > 0.01: 0.00837718, 0
1 > 0.01: 0.092658, 0
1 > 0.01: 1.15046, 0
1 > 0.01: 1.39525e-07, 0
1 > 0.01: 23.3724, 0
1 > 0.01: 1.90047e-05, 0
1 > 0.01: 3.735, 0
1 > 0.01: 0.000672563, 0
SNR: 22.480740db
189 samples with relative error > 0.01
```

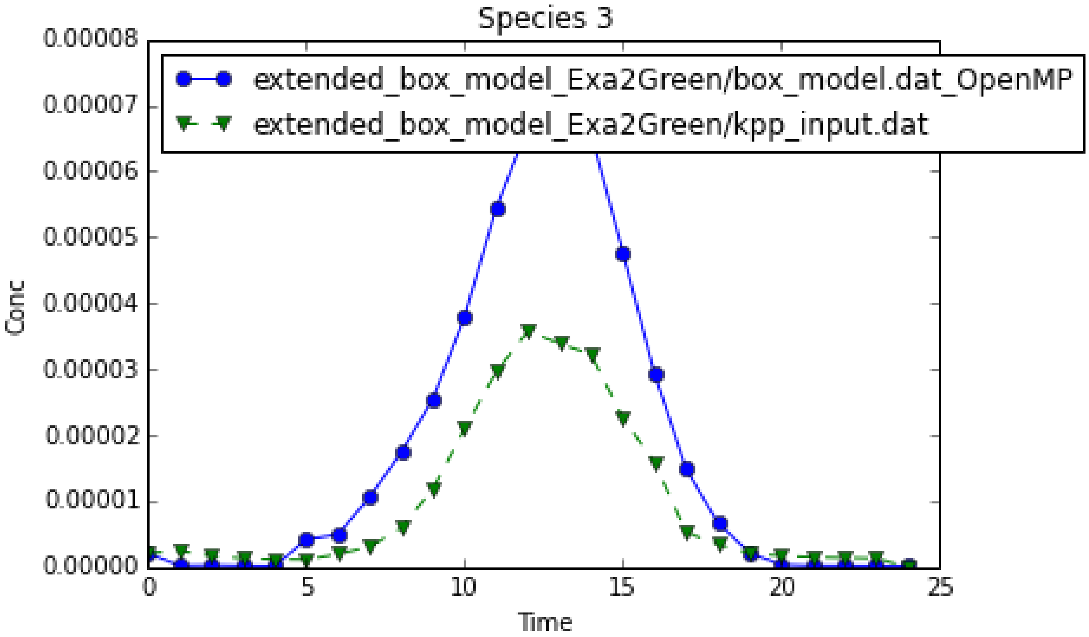
In [11]:

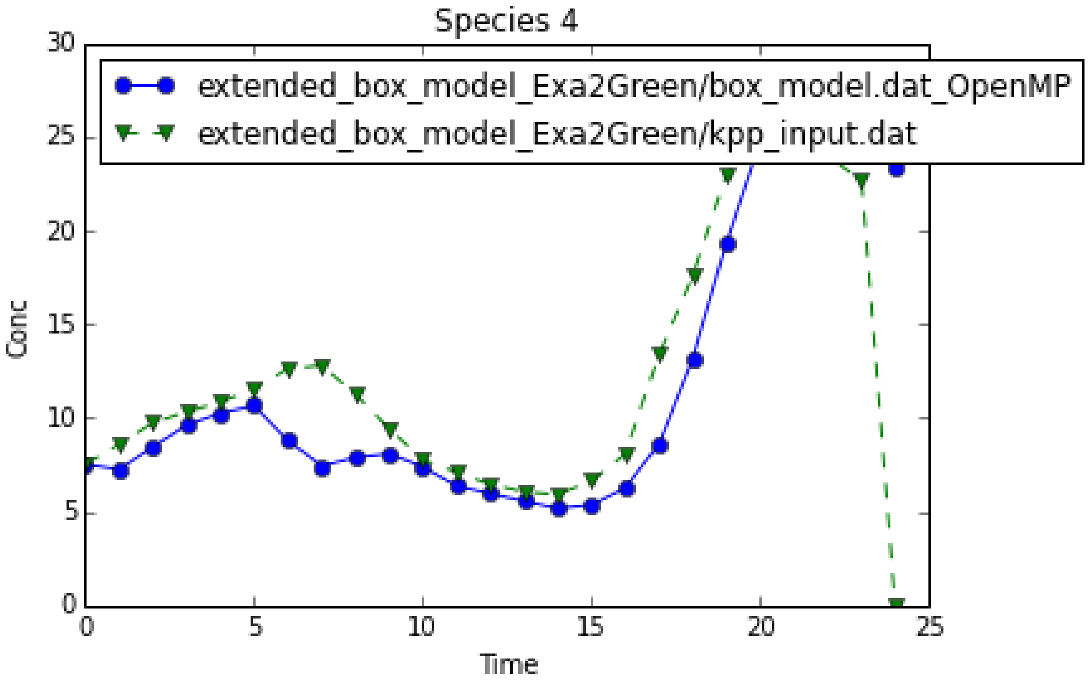
```
plot_dat([kpp_dat_1, kpp_dat_2], names=[kpp_file_1, kpp_file_2], titles=None)
```

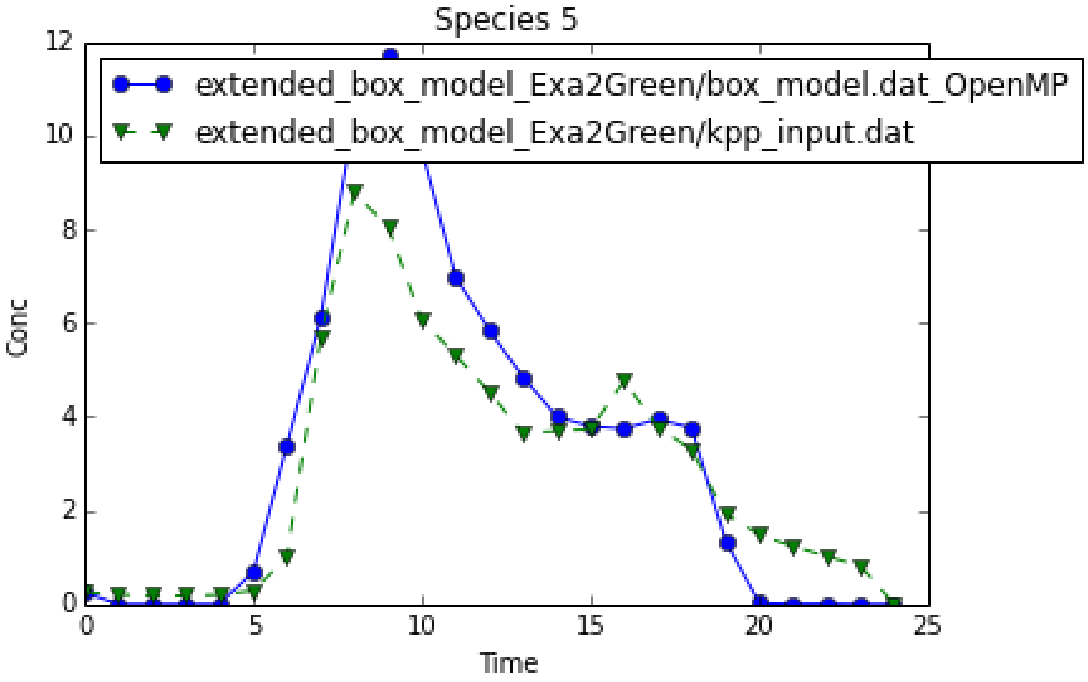

Species 0
extended_box_model_Exa2Green/box_model.dat_OpenMP
extended_box_model_Exa2Green/kpp_input.dat
Conc
0.009
0.008
0.007
0.006
0.005
0.004
0.003
0.002
0.001
0.000
0
5
10
15
20
25
Time

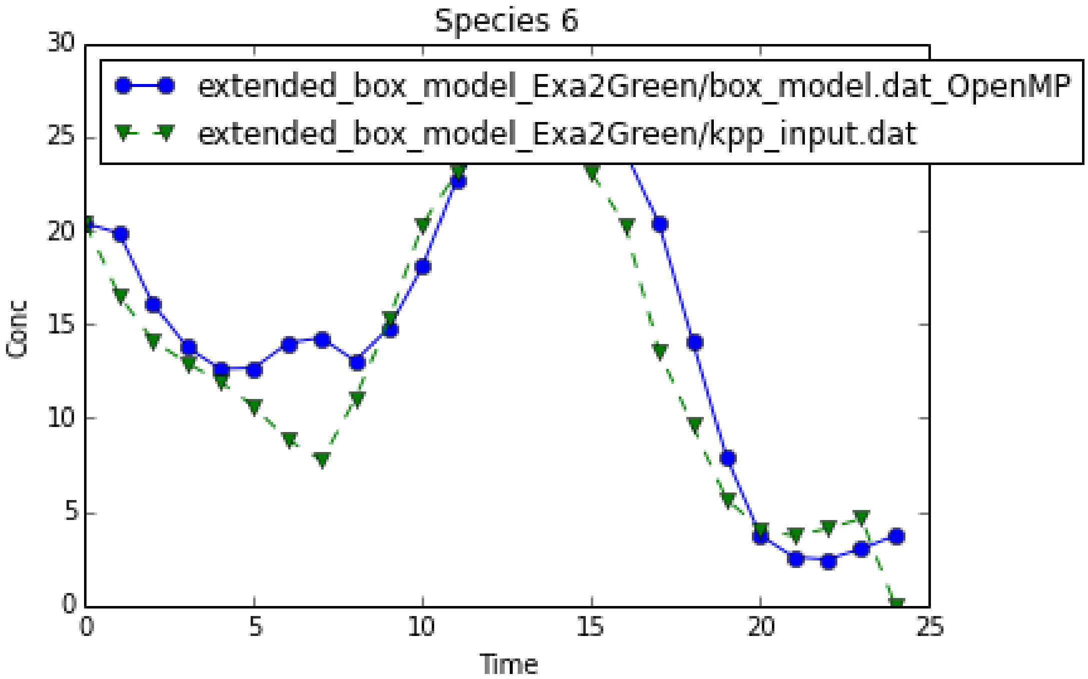

Species 1
extended_box_model_Exa2Green/box_model.dat_OpenMP
extended_box_model_Exa2Green/kpp_input.dat
Conc
0.10
0.08
0.06
0.04
0.02
0.00
0
5
10
15
20
25
Time

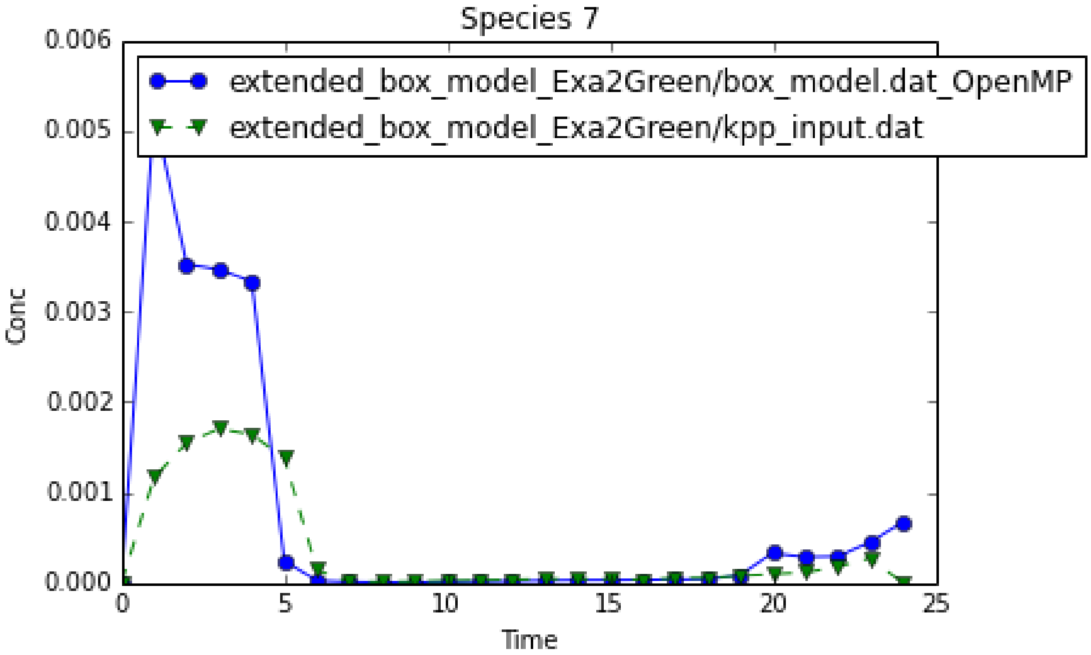

Species 2
extended_box_model_Exa2Green/box_model.dat_OpenMP
extended_box_model_Exa2Green/kpp_input.dat
Conc
1.2
1.0
0.8
0.6
0.4
0.2
0.0
0
5
10
15
20
25
Time


Species 3
extended_box_model_Exa2Green/box_model.dat_OpenMP
extended_box_model_Exa2Green/kpp_input.dat
Conc
0.00008
0.00007
0.00006
0.00005
0.00004
0.00003
0.00002
0.00001
0.00000
0
5
10
15
20
25
Time


Species 4
extended_box_model_Exa2Green/box_model.dat_OpenMP
extended_box_model_Exa2Green/kpp_input.dat
Conc
Time
0
5
10
15
20
25
30
0
5
10
15
20
25

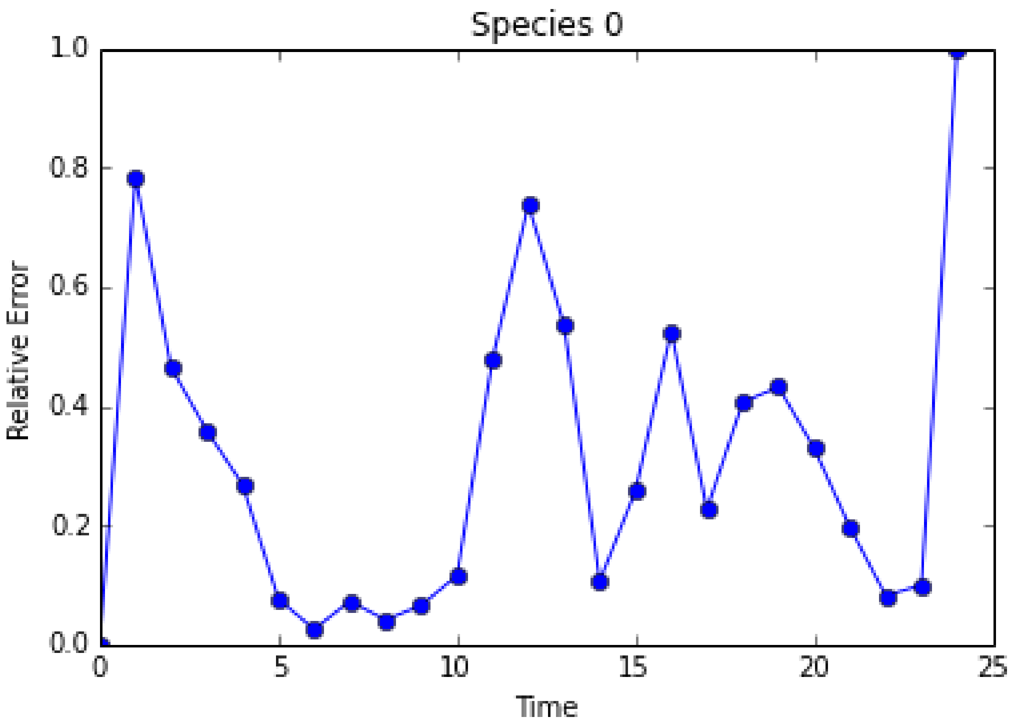

Species 5
extended_box_model_Exa2Green/box_model.dat_OpenMP
extended_box_model_Exa2Green/kpp_input.dat
Conc
Time
0
2
4
6
8
10
12
0
5
10
15
20
25

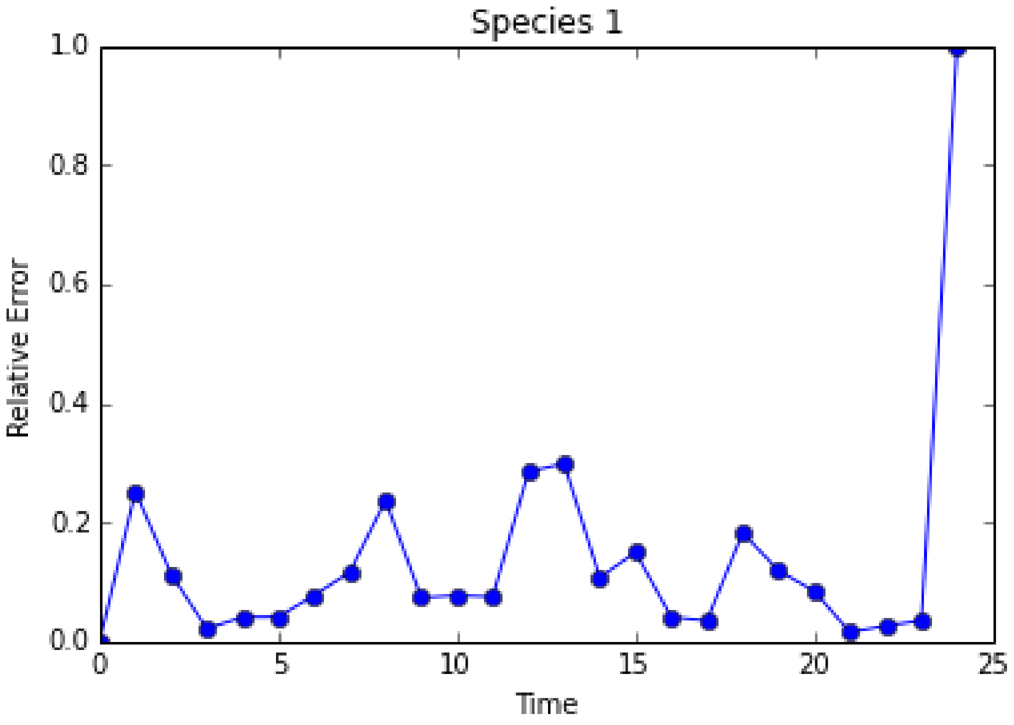


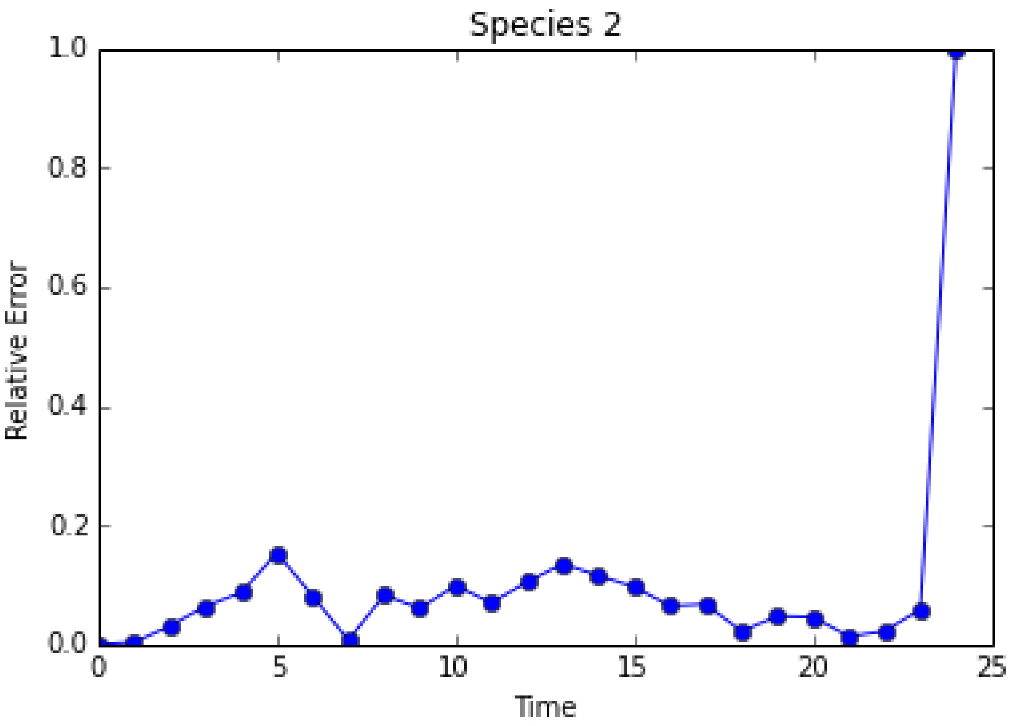


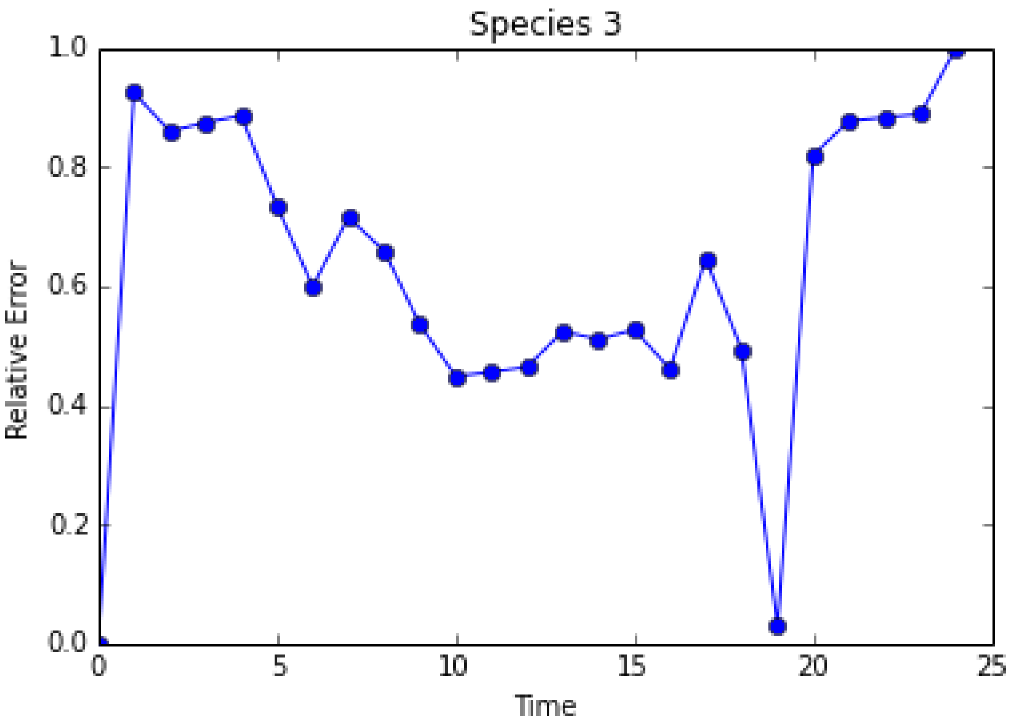
In [12]:

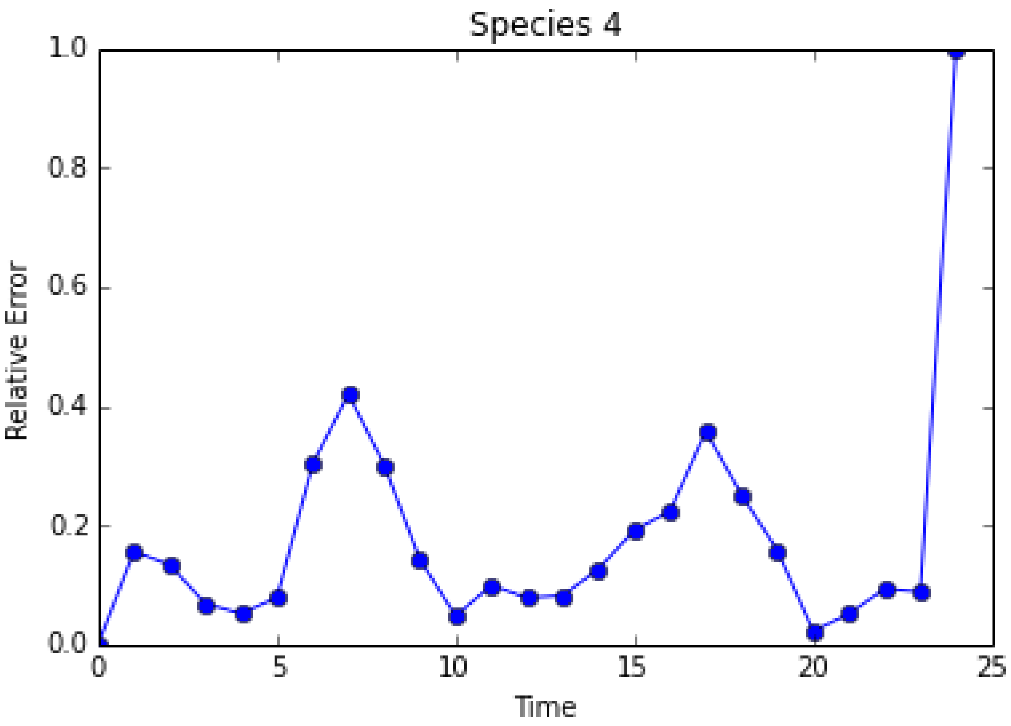
```
plot_dat([err_dat], ylabel='Relative Error')
```

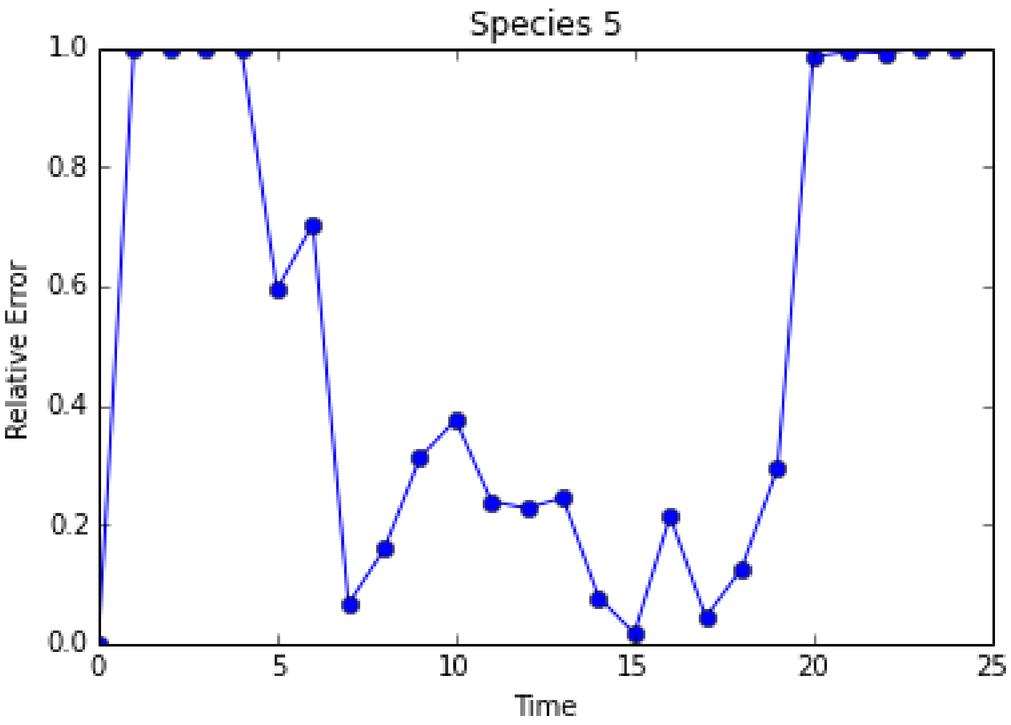

Species 0
Relative Error
1.0
0.8
0.6
0.4
0.2
0.0
0
5
10
15
20
25
Time

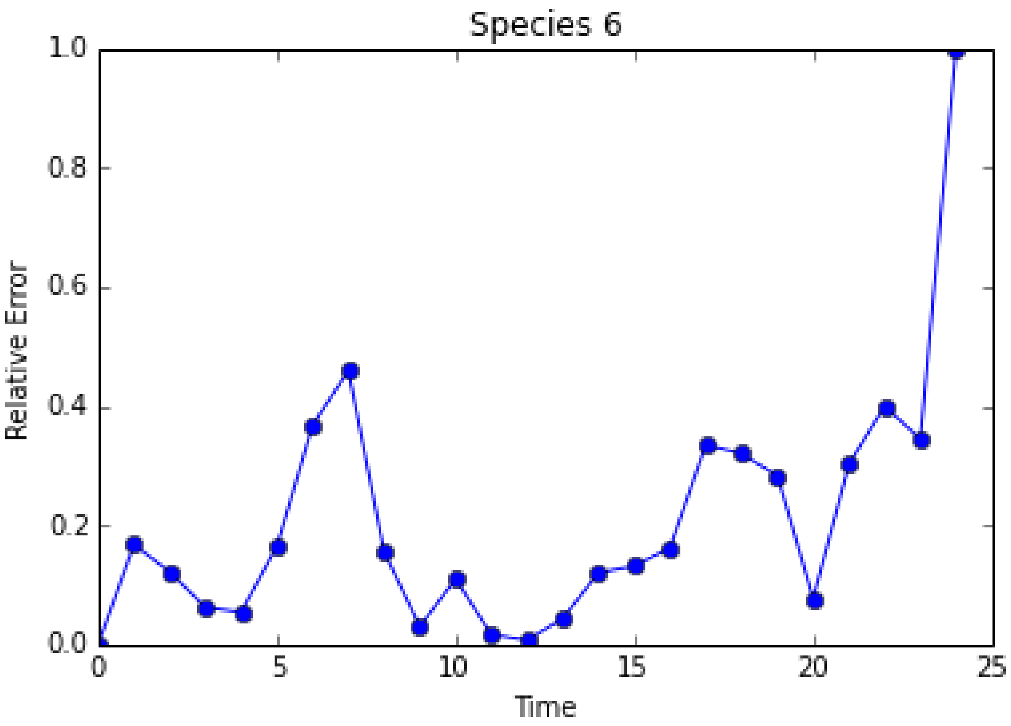

Species 1
Relative Error
1.0
0.8
0.6
0.4
0.2
0.0
0
5
10
15
20
25
Time


Species 2

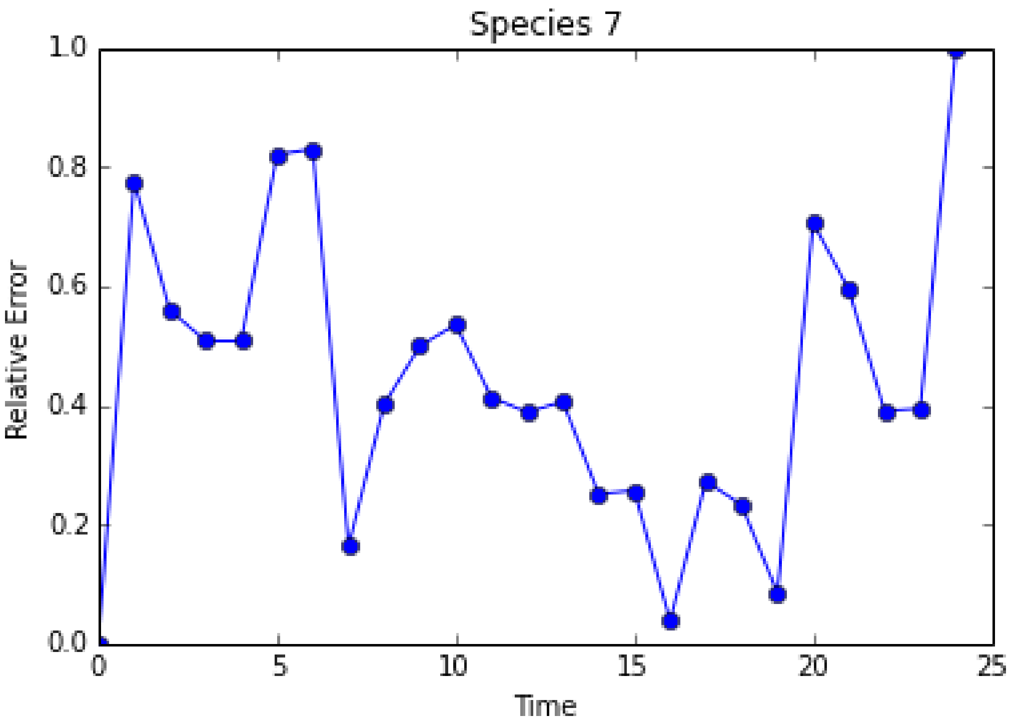

Species 3


Species 4
Relative Error
0.0
0.2
0.4
0.6
0.8
1.0
Time
0
5
10
15
20
25


Species 5
Relative Error
0.0
0.2
0.4
0.6
0.8
1.0
Time
0
5
10
15
20
25



Species 6



Species 7

In [12]: