efficiency

January 17, 2018

In [1]:
```
%matplotlib inline
```

In [2]:
```
import matplotlib.pyplot as plt
import numpy as np
from pylab import *
import matplotlib as mpl
mpl.rcParams['figure.dpi']= 200

data10M = []
with open('/home/nicole/Data Science/HPC/Firstday/eff10M.txt') as datafile10M:
        for line in datafile10M:
            data10M.append(float(line))

data1G = []
with open('/home/nicole/Data Science/HPC/Firstday/eff1G.txt') as datafile1G:
        for line in datafile1G:
            data1G.append(float(line))

data100M = []
with open('/home/nicole/Data Science/HPC/Firstday/eff100M.txt') as datafile100M:
        for line in datafile100M:
            data100M.append(float(line))


x = np.array([1,2,4,8,16,20])
t = arange(0.0, 20.0, 1)
y = x
data10M
```

Out[2]:
```
[1.0,
 0.523203169215,
 0.264445080093,
 0.125953678474,
 0.0583648989899,
 0.0456543209877]
```

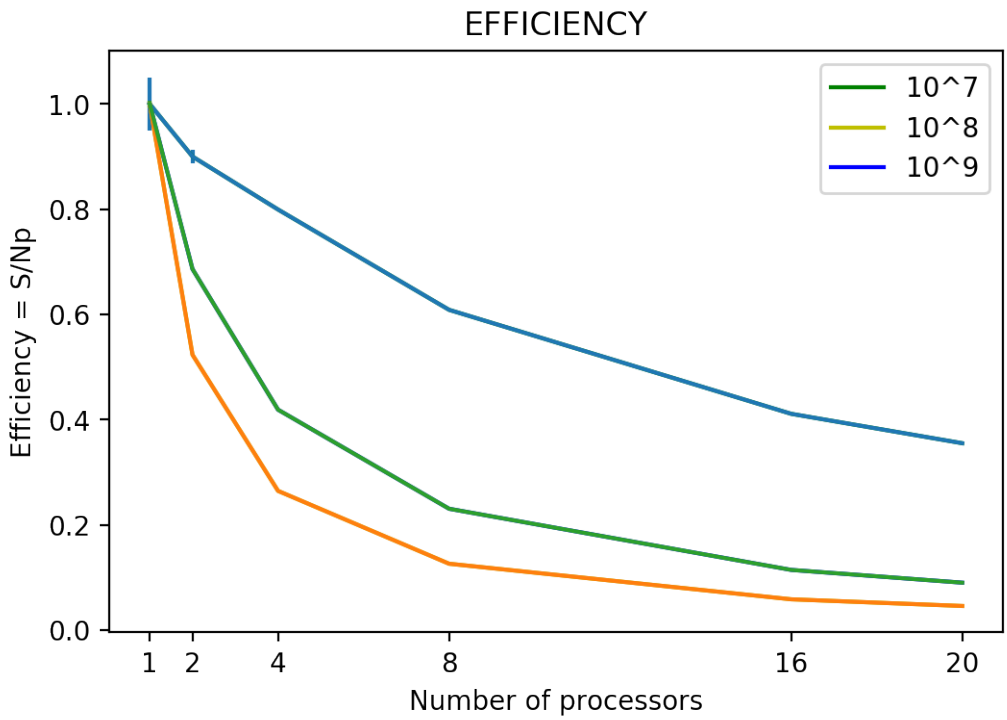
In [6]:
```
#errors
```

```
err10M = []
with open('eff10M_err.txt') as datafile10M:
        for line in datafile10M:
            err10M.append(float(line))

err1G = []
with open('eff1G_err.txt') as datafile1G:
        for line in datafile1G:
            err1G.append(float(line))

err100M = []
with open('eff100M_err.txt') as datafile100M:
        for line in datafile100M:
            err100M.append(float(line))
```

In [9]:
```
plt.legend(bbox_to_anchor=(0.5, 1.02, 1., .102), loc=3,
           ncol=2, mode="expand", borderaxespad=0.)
plt.ylabel('Efficiency = S/Np')
plt.xlabel('Number of processors')
plt.plot(x,data1G, 'g',label='10^7')
plt.errorbar(x,data1G, yerr = err1G)
plt.plot(x,data10M, 'y', label='10^8')
plt.errorbar(x,data10M, yerr = err10M)
plt.plot(x,data100M, 'b', label='10^9')
plt.errorbar(x,data100M, yerr = err100M)
#plt.plot(x,data10M, 'yo', 'y')
#plt.plot(x,data1G, 'go', 'g')
#plt.plot(x,data100M, 'bo', 'b')
plt.xticks(x,[1,2,4,8,16,20])
plt.title('EFFICIENCY')
plt.legend()
plt.show()
```


EFFICIENCY
10^7
10^8
10^9
1.0
0.8
0.6
0.4
0.2
0.0
Efficiency = S/Np
1
2
4
8
16
20
Number of processors