4_1

March 28, 2016

In [25]:
```
__author__ = 'Security'
import numpy as np
import scipy.stats as stats
%matplotlib inline
import matplotlib.pyplot as plt
```

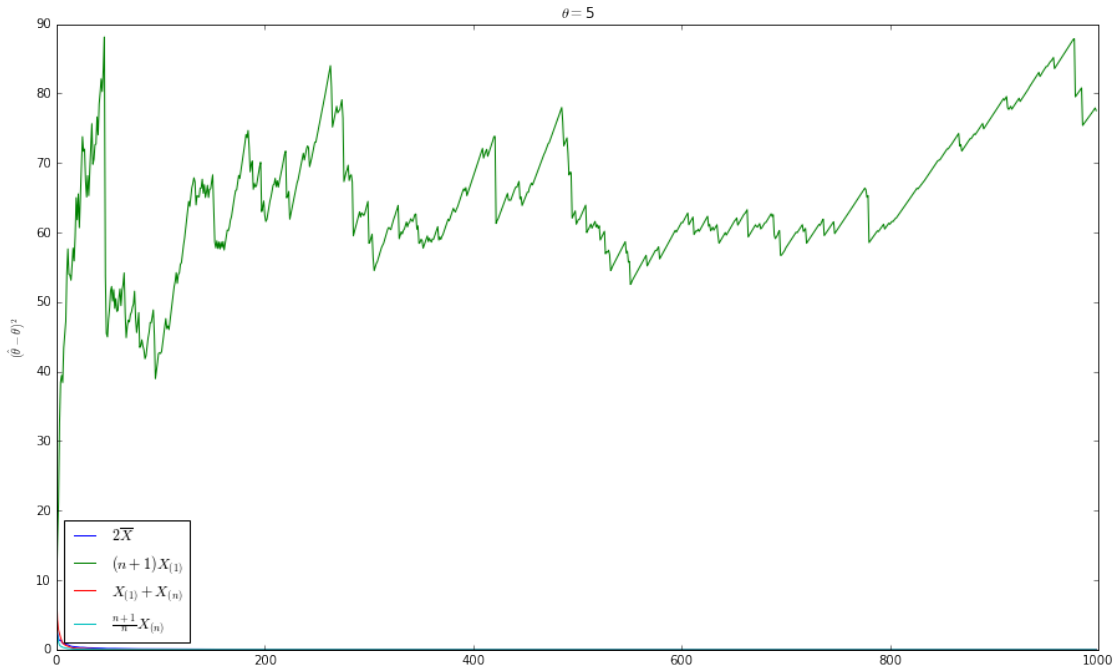
In [26]:
```
N = 1000
```

In [27]:
```
def getStat(arr, k):
    return np.partition(arr, k)[k]


def sqrDelta(estimation, theta):
    return (estimation - theta) ** 2


def estim1(arr, n):
    return sqrDelta(arr[:n+1].mean() * 2, theta)


def estim2(arr, n):
    return sqrDelta(float(n + 1) * getStat(arr[:n+1], 1), theta)


def estim3(arr, n):
    return sqrDelta(getStat(arr[:n+1], 1) + getStat(arr[:n+1], n), theta)


def estim4(arr, n):
    return sqrDelta((float(n + 1) / float(n)) * getStat(arr[:n+1], n), theta)
```

In [86]:
```
def task(M, theta):
    samples = [stats.uniform.rvs(size=N, scale=theta) for i in range(M)]
    estimations = [(estim1, r'$2\overline{X}$'), (estim2, r'$(n+1)X_{(1)}$'), (estim3, r'$X_{(1
    means = [[np.average([estimation[0](sample, n) for sample in samples]) for n in range(1, N)
    plt.figure(figsize=(15, 9))
    plt.title(r'$\theta=$' + str(theta))
    plt.ylabel(r'$(\hat{\theta} - \theta)^{2}$')
    for i in range(len(estimations)):
        plt.plot(means[i], label=estimations[i][1])
    plt.legend(loc='best')
    plt.show()

def taskNoBigEstimations(M, theta, ymin, ymax):
```

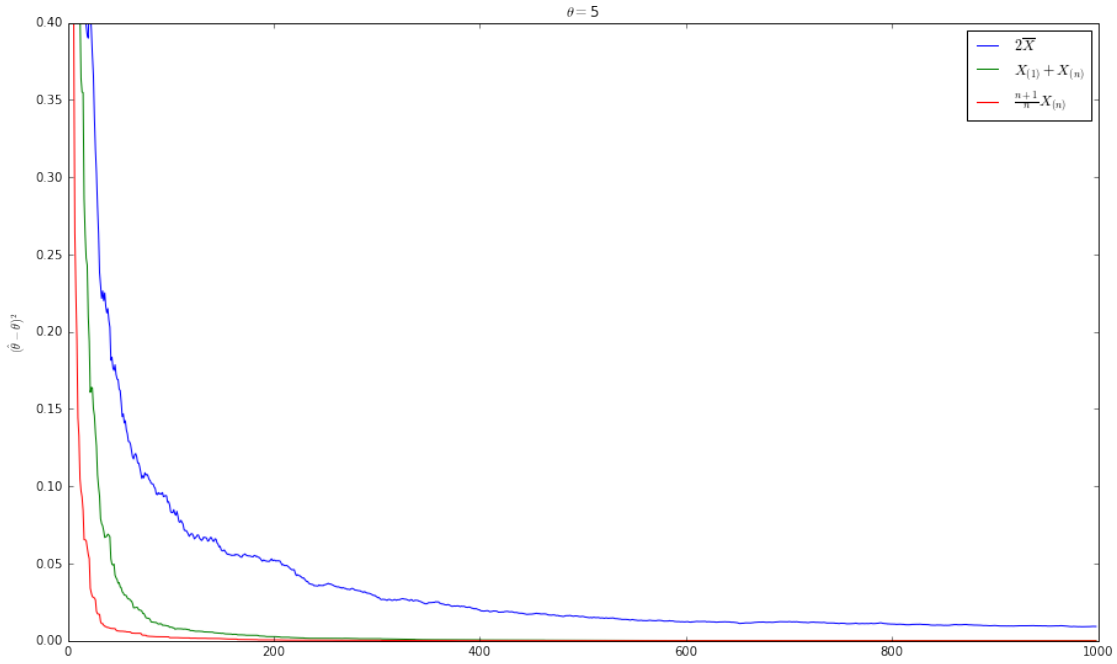
```
    samples = [stats.uniform.rvs(size=N, scale=theta) for i in range(M)]
    estimations = [(estim1, r'$2\overline{X}$'), (estim2, r'$(n+1)X_{(1)}$'), (estim3, r'$X_{(1
    means = [[np.average([estimation[0](sample, n) for sample in samples]) for n in range(1, N)
    plt.figure(figsize=(15, 9))
    plt.title(r'$\theta=$' + str(theta))
    plt.ylabel(r'$(\hat{\theta} - \theta)^{2}$')
    plt.ylim(ymin, ymax)
    for i in [0, 2, 3]:
        plt.plot(means[i], label=estimations[i][1])
    plt.legend(loc='best')
    plt.show()
```

In [79]: `task(100, 5)`

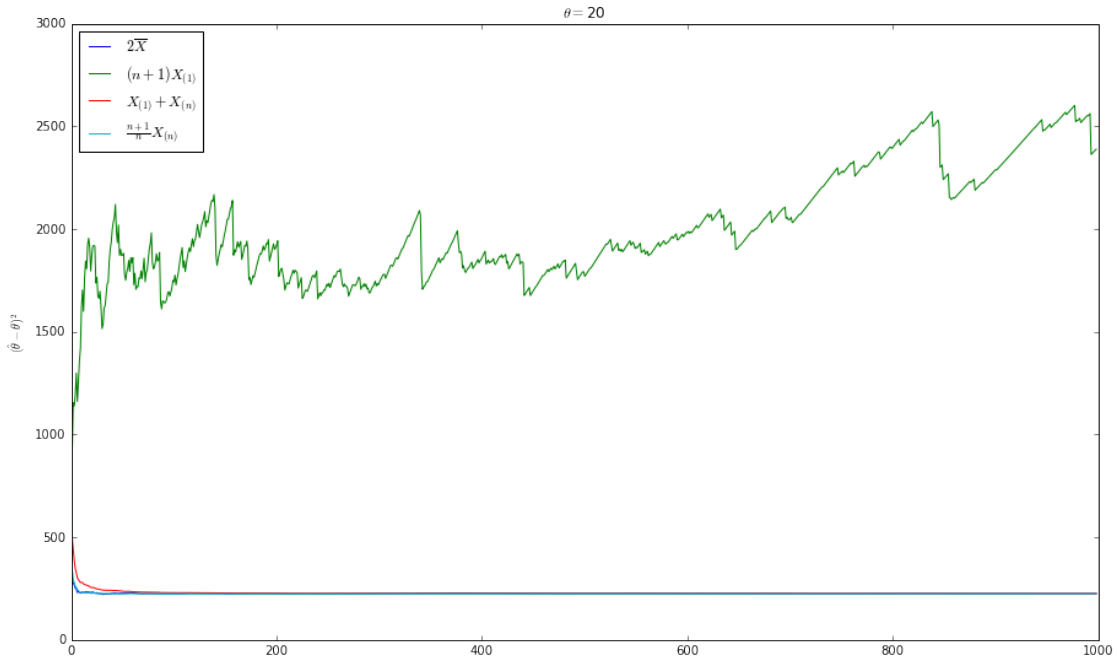


Для наглядности нарисуем график без $(n+1)X_{(1)}$

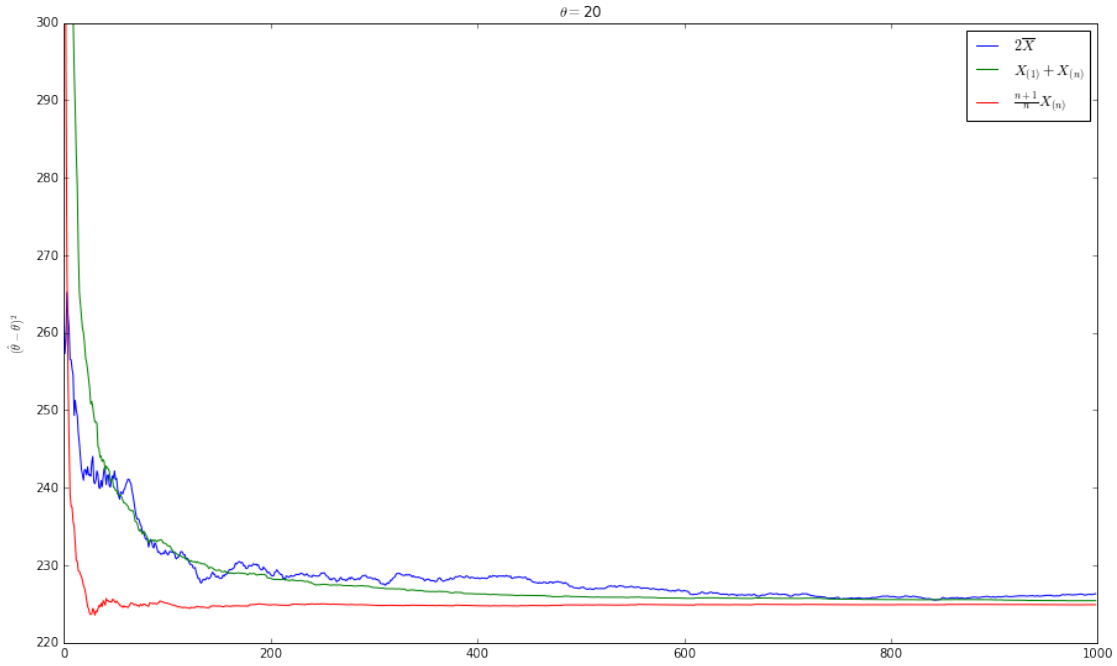
In [94]: `taskNoBigEstimations(100, 5, 0, 0.4)`



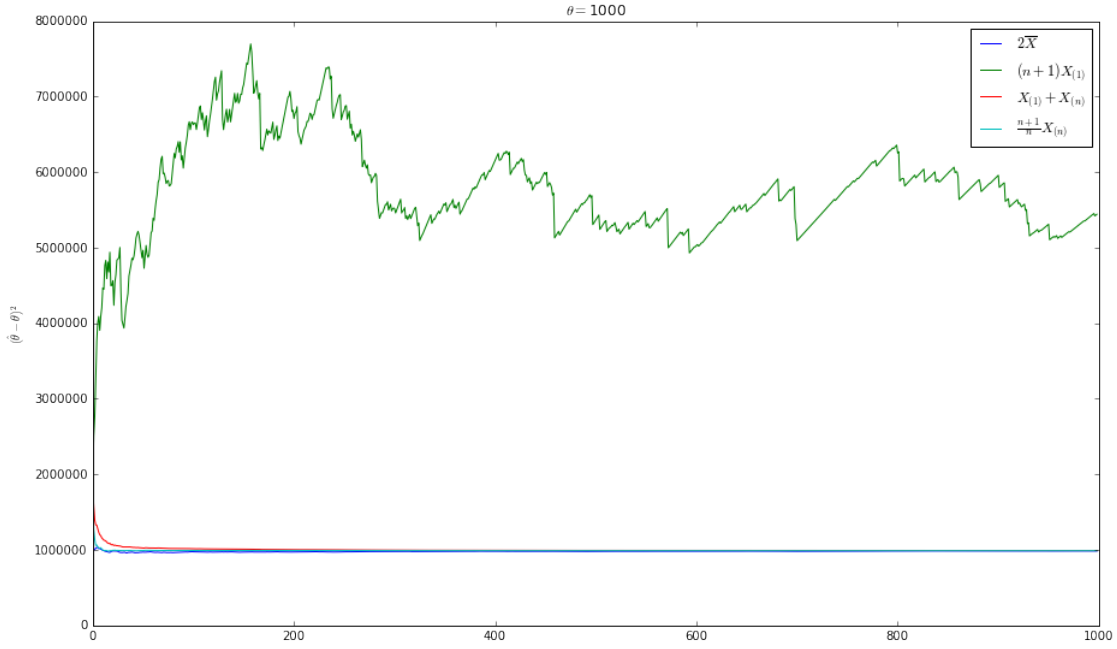
In [81]: `task(100, 20)`



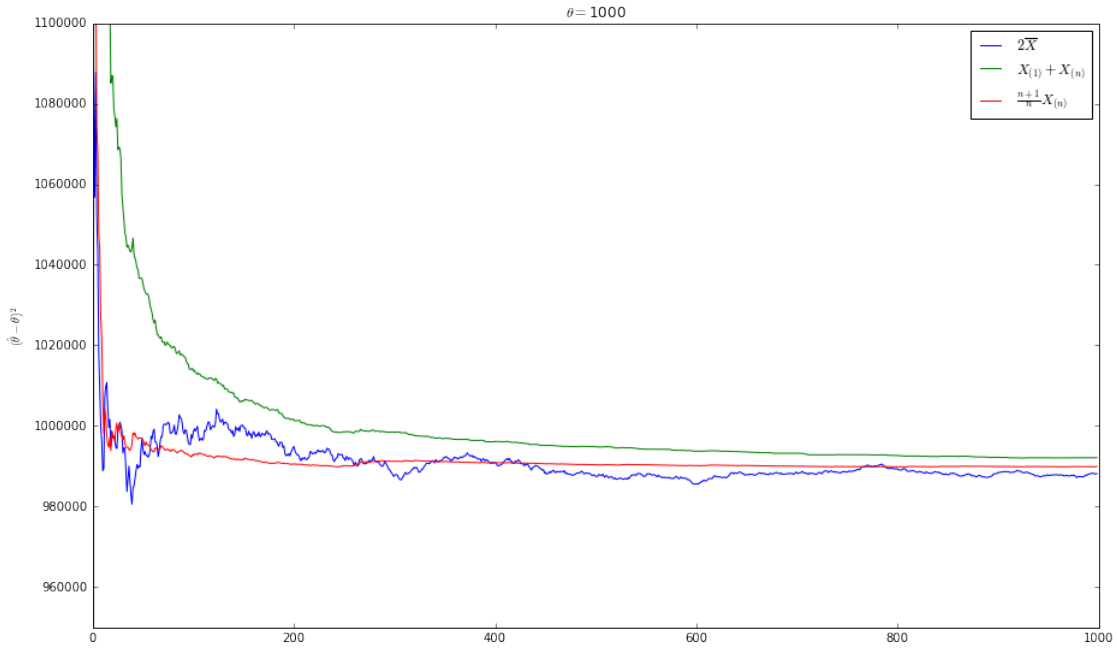
In [89]: `taskNoBigEstimations(100, 20, 220, 300)`



In [83]: task(100, 1000)



In [93]: taskNoBigEstimations(100, 1000, 950000, 1100000)


$\theta = 1000$
$2\overline{X}$
$X_{(1)} + X_{(n)}$
$\frac{n+1}{n} X_{(n)}$
$(\hat{\theta} - \theta)^2$
1100000
1080000
1060000
1040000
1020000
1000000
980000
960000
0
200
400
600
800
1000

In [ ]: