In [8]:

```
import scipy.stats as sps
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
%pylab inline
```

```
Populating the interactive namespace from numpy and matplotlib
```

θ = 1

In [13]:

```
n = 10000
theta = 1
```

In [84]:

```
sample = sps.uniform.rvs(loc = 0, scale = theta, size = n)
```

In [17]:

```
s = np.arange(1, n + 1)
means = sample.cumsum() / s
first = abs(means * 2 - theta)
```

In [18]:

```
second = np.zeros(n)
third = np.zeros(n)
fourth = np.zeros(n)
fifth = np.zeros(n)
```

In [87]:

```
max_in_array = sample[0]
min_in_array = sample[0]
for i in range(n):
    max_in_array = max(max_in_array, sample[i])
    min_in_array = min(min_in_array, sample[i])
    second[i] = abs(means[i] + max_in_array / 2 - theta)
    third[i] = abs((i + 2) * min_in_array - theta)
    fourth[i] = abs(max_in_array + min_in_array - theta)
    fifth[i] = abs((float) (i + 2) / (i + 1) * max_in_array - theta)
```

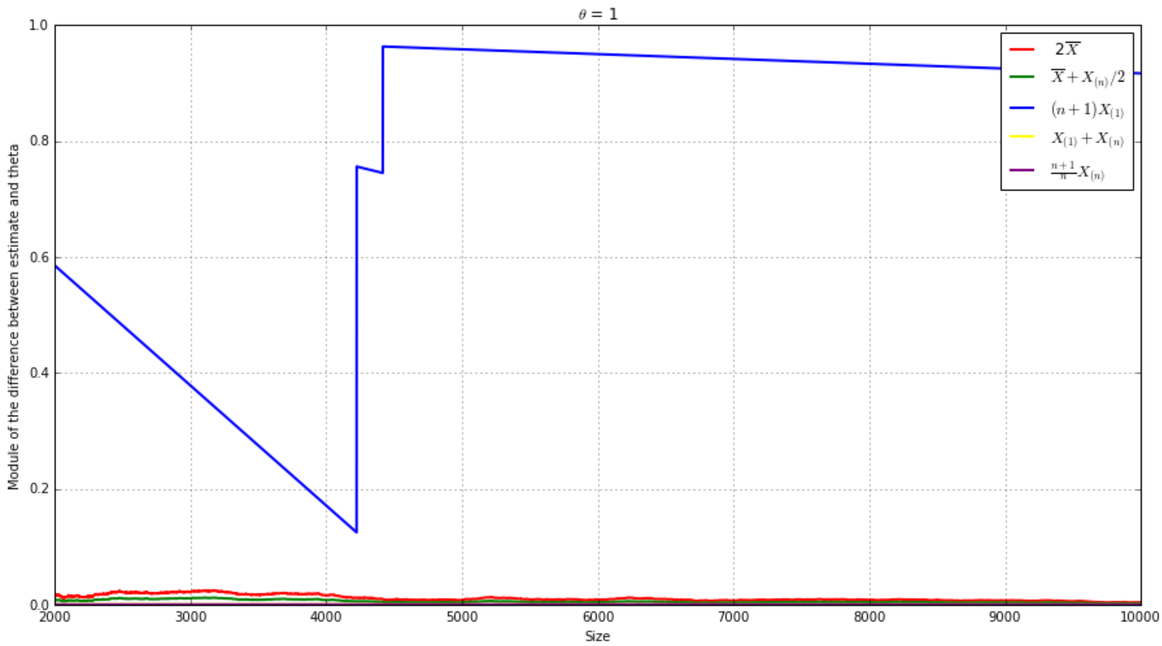
Построим график зависимости модуля разности оценки и истинного значения θ от размера выборки:

In [118]:

```
plt.figure(figsize=(15, 8))
plt.plot(s, first, color='red', linewidth=2, label=' 2$\overline{X}$')
plt.plot(s, second, color='green', linewidth=2, label='$\overline{X} + X_{(n)} / 2$')
plt.plot(s, third, color='blue', linewidth=2, label='${(n + 1)}X_{(1)}$')
plt.plot(s, fourth, color='yellow', linewidth=2, label='$ X_{(1)} + X_{(n)}$')
plt.plot(s, fifth, color='purple', linewidth=2, label=r'$\frac{n+1}{n}X_{(n)}$')

plt.legend()
plt.xlim((2000, n))
plt.ylim((0, 1))
plt.xlabel('Size')
plt.ylabel('Module of the difference between estimate and theta')
plt.title(r'$\theta$ = 1')
plt.grid()

plt.show()
```



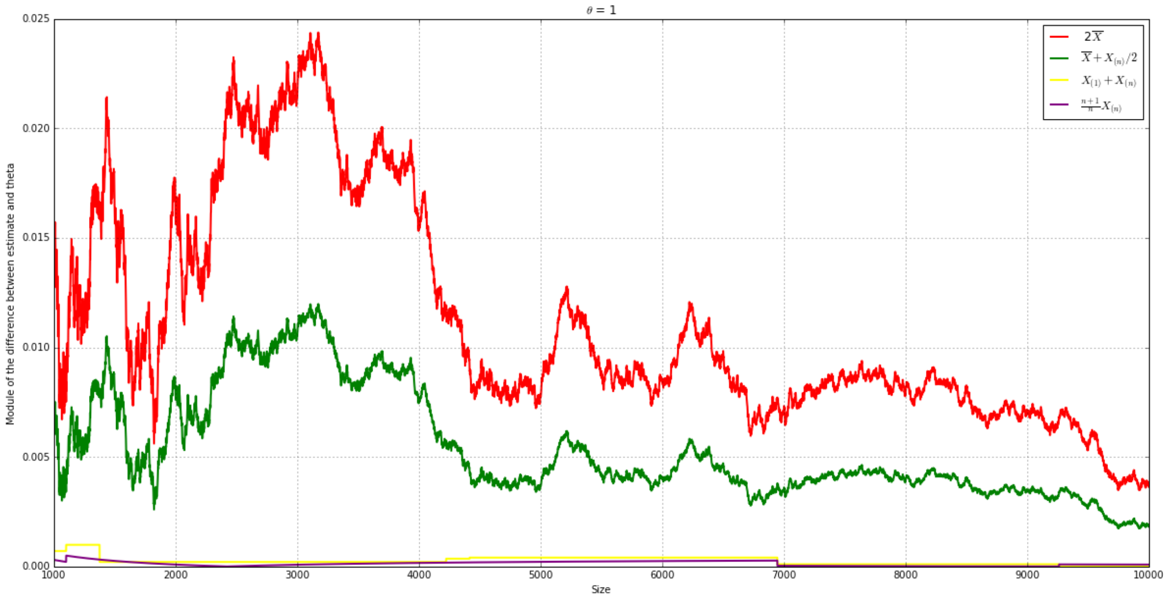
Для того чтобы понять, какая оценка является наиболее точной, исключим из рассмотрения оценку $(n+1)X_{(1)}$:

In [116]:

```
plt.figure(figsize=(20, 10))
plt.plot(s, first, color='red', linewidth=2, label=' 2$\overline{X}$')
plt.plot(s, second, color='green', linewidth=2, label='$\overline{X} + X_{(n)} / 2$')
plt.plot(s, fourth, color='yellow', linewidth=2, label='$ X_{(1)} + X_{(n)}$')
plt.plot(s, fifth, color='purple', linewidth=2, label=r'$\frac{n+1}{n}X_{(n)}$')

plt.legend()
plt.xlim((1000, n))
plt.ylim((0, 0.025))
plt.xlabel('Size')
plt.ylabel('Module of the difference between estimate and theta')
plt.title(r'$\theta$ = 1')
plt.grid()

plt.show()
```



θ = 10

In [14]:

```
theta = 10
```

In [15]:

```
sample = sps.uniform.rvs(loc = 0, scale = theta, size = n)
```

In [19]:

```

```

```
means = sample.cumsum() / s
first = abs(means * 2 - theta)
second = np.zeros(n)
third = np.zeros(n)
fourth = np.zeros(n)
fifth = np.zeros(n)
```

In [20]:

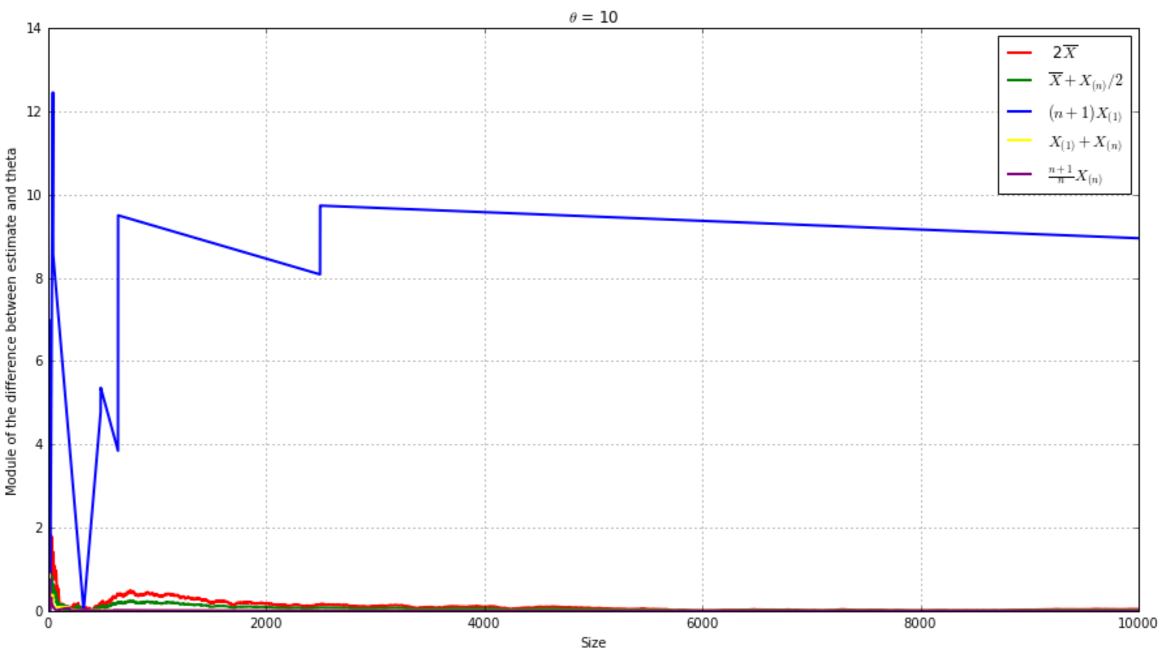
```
max_in_array = sample[0]
min_in_array = sample[0]
for i in range(n):
    max_in_array = max(max_in_array, sample[i])
    min_in_array = min(min_in_array, sample[i])
    second[i] = abs(means[i] + max_in_array / 2 - theta)
    third[i] = abs((i + 2) * min_in_array - theta)
    fourth[i] = abs(max_in_array + min_in_array - theta)
    fifth[i] = abs((float) (i + 2) / (i + 1) * max_in_array - theta)
```

In [27]:

```
plt.figure(figsize=(15, 8))
plt.plot(s, first, color='red', linewidth=2, label=' 2$\overline{X}$')
plt.plot(s, second, color='green', linewidth=2, label='$\overline{X} + X_{(n)} / 2$')
plt.plot(s, third, color='blue', linewidth=2, label='${(n + 1)}X_{(1)}$')
plt.plot(s, fourth, color='yellow', linewidth=2, label='$ X_{(1)} + X_{(n)}$')
plt.plot(s, fifth, color='purple', linewidth=2, label=r'$\frac{n+1}{n}X_{(n)}$')

plt.legend()
plt.xlim((1, n))
plt.ylim((0, 14))
plt.xlabel('Size')
plt.ylabel('Module of the difference between estimate and theta')
plt.title(r'$\theta$ = 10')
plt.grid()

plt.show()
```

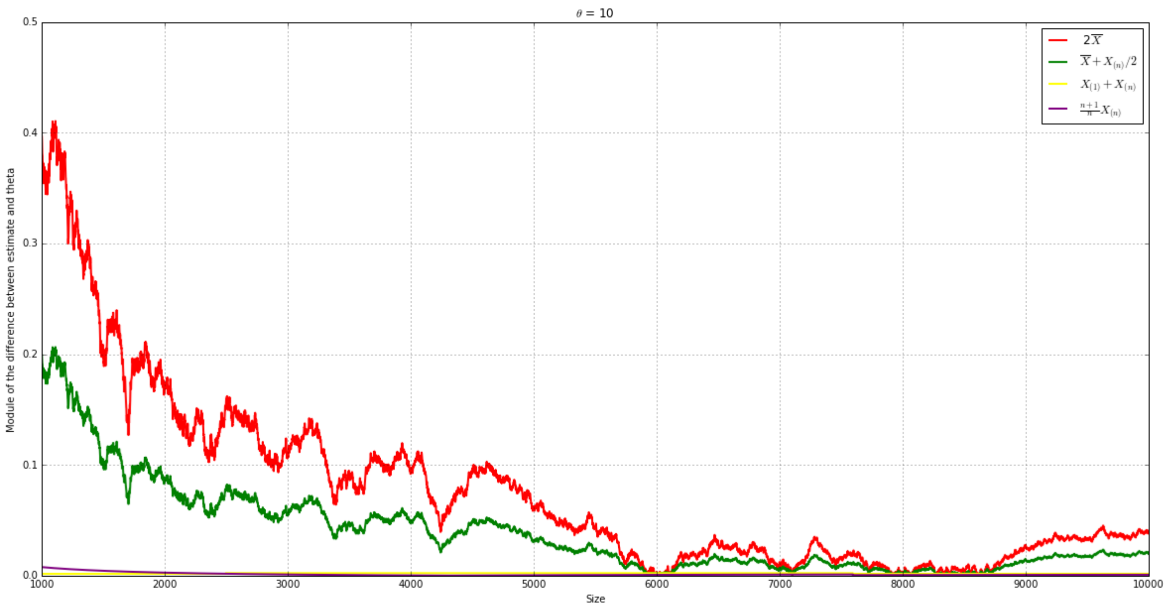


In [32]:

```
plt.figure(figsize=(20, 10))
plt.plot(s, first, color='red', linewidth=2, label=' 2$\overline{X}$')
plt.plot(s, second, color='green', linewidth=2, label='$\overline{X} + X_{(n)} / 2$')
plt.plot(s, fourth, color='yellow', linewidth=2, label='$ X_{(1)} + X_{(n)}$')
plt.plot(s, fifth, color='purple', linewidth=2, label=r'$\frac{n+1}{n}X_{(n)}$')

plt.legend()
plt.xlim((1000, n))
plt.ylim((0, 0.5))
plt.xlabel('Size')
plt.ylabel('Module of the difference between estimate and theta')
plt.title(r'$\theta$ = 10')
plt.grid()

plt.show()
```



θ = 100

In [33]:

```
theta = 100
```

In [34]:

```
sample = sps.uniform.rvs(loc = 0, scale = theta, size = n)
```

In [35]:

In [36]:

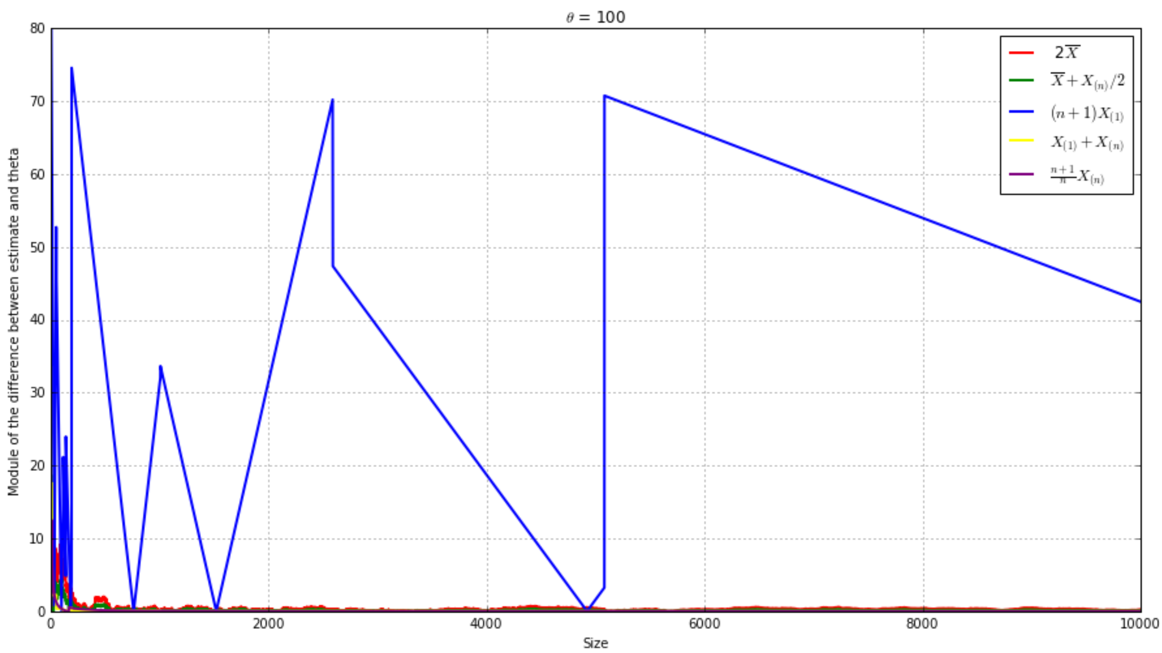
```
means = sample.cumsum() / s
first = abs(means * 2 - theta)
second = np.zeros(n)
third = np.zeros(n)
fourth = np.zeros(n)
fifth = np.zeros(n)
max_in_array = sample[0]
min_in_array = sample[0]
for i in range(n):
    max_in_array = max(max_in_array, sample[i])
    min_in_array = min(min_in_array, sample[i])
    second[i] = abs(means[i] + max_in_array / 2 - theta)
    third[i] = abs((i + 2) * min_in_array - theta)
    fourth[i] = abs(max_in_array + min_in_array - theta)
    fifth[i] = abs((float) (i + 2) / (i + 1) * max_in_array - theta)
```

In [41]:

```
plt.figure(figsize=(15, 8))
plt.plot(s, first, color='red', linewidth=2, label=' 2$\overline{X}$')
plt.plot(s, second, color='green', linewidth=2, label='$\overline{X} + X_{(n)} / 2$')
plt.plot(s, third, color='blue', linewidth=2, label='${(n + 1)}X_{(1)}$')
plt.plot(s, fourth, color='yellow', linewidth=2, label='$ X_{(1)} + X_{(n)}$')
plt.plot(s, fifth, color='purple', linewidth=2, label=r'$\frac{n+1}{n}X_{(n)}$')

plt.legend()
plt.xlim((1, n))
plt.ylim((0, 80))
plt.xlabel('Size')
plt.ylabel('Module of the difference between estimate and theta')
plt.title(r'$\theta$ = 100')
plt.grid()

plt.show()
```

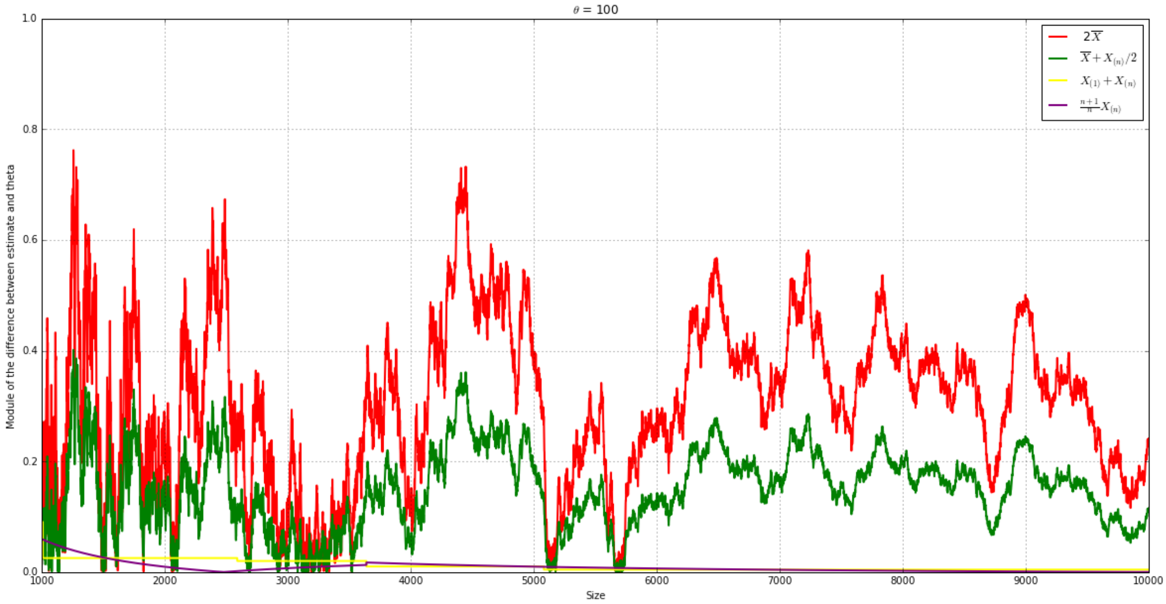


In [46]:

```
plt.figure(figsize=(20, 10))
plt.plot(s, first, color='red', linewidth=2, label=' 2$\overline{X}$')
plt.plot(s, second, color='green', linewidth=2, label='$\overline{X} + X_{(n)} / 2$')
plt.plot(s, fourth, color='yellow', linewidth=2, label='$ X_{(1)} + X_{(n)}$')
plt.plot(s, fifth, color='purple', linewidth=2, label=r'$\frac{n+1}{n}X_{(n)}$')

plt.legend()
plt.xlim((1000, n))
plt.ylim((0, 1))
plt.xlabel('Size')
plt.ylabel('Module of the difference between estimate and theta')
plt.title(r'$\theta$ = 100')
plt.grid()

plt.show()
```



Из графиков видно, что оценка $(n+1)X_{(1)}$ дает наибольшее отклонение от истинного значения θ, причем отклонение становится больше при увеличении θ. Она единственная из всех является несостоятельной. Наиболее точные оценки - $X_{(1)} + X_{(n)}$ и $\frac{n+1}{n}X_{(n)}$ - несмещённые и сильно состоятельные.