In [1]:
```python
import numpy as np
from matplotlib import pyplot as plt
```

ЗАДАЧА С ОПРЕДЕЛИТЕЛЕМ

In [2]:
```python
matrix = np.genfromtxt('det/matrix.txt')
print(np.linalg.det(matrix))
```

110468830.00000018

Ускорение в определителе

In [22]:
```python
posl = np.genfromtxt('det/determinant_data_from_dim_posl.txt')
par_dim = np.genfromtxt('det/determinant_data_from_dim_parallel.txt
par = np.genfromtxt('det/determinant_data_from_threads.txt')

fig, ax = plt.subplots()

ax.plot(par[:, 0], (par[:, 1]/par[0, 1])**(-1), label='acceleration
ax.set_xlabel('threads')

ax.legend()
```
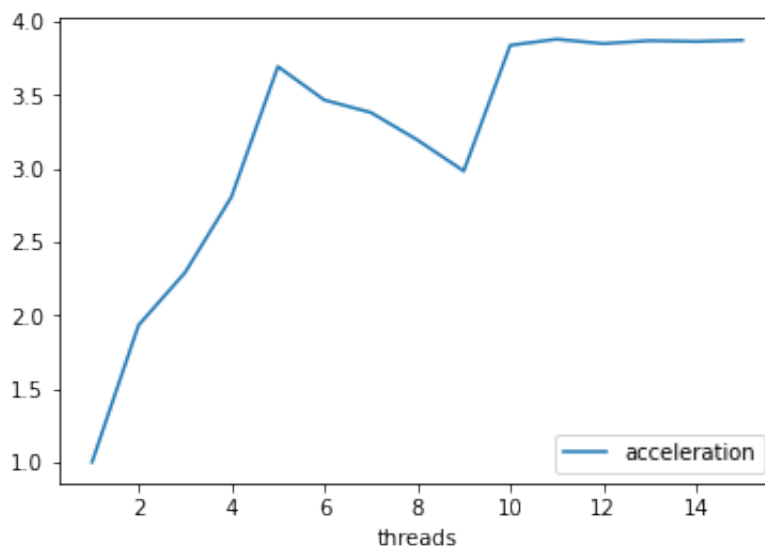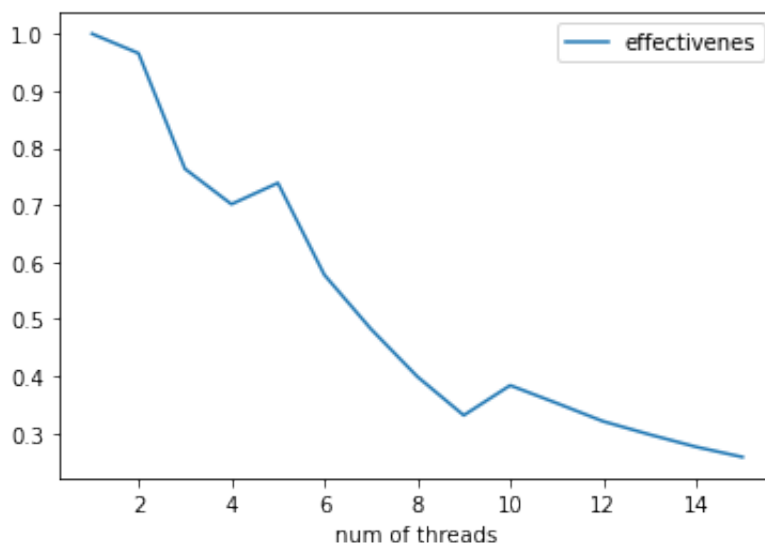
Out[22]: <matplotlib.legend.Legend at 0x7fb6fab85790>

Эффективность

In [23]:
```python
par = np.genfromtxt('det/determinant_data_from_threads.txt')

y = par[:, 1]/par[0, 1]

fig, ax = plt.subplots()

ax.plot(par[:, 0], y**-1/par[:, 0], label='effectivenes')
ax.set_xlabel('num of threads')
ax.legend()
```

Out[23]: <matplotlib.legend.Legend at 0x7fb6fab72610>

ЗАДЧА С СЛАУ

Проверка работы алгоритма

In [5]:
```python
A = np.genfromtxt('sys_of_eq/matrix.txt')
b = np.array([1, 1, 1])
print(b)
x = np.linalg.solve(A, b)

print(x)
```

```
[1 1 1]
[-0.00157233 -0.11792453  0.01100629]
```

In [6]:
```python
A[1] -= A[0] * A[1, 0] / A[0, 0]
A[2] -= A[0] * A[2, 0] / A[0, 0]

A[2] -= A[1] * A[2, 1] / A[1, 1]
print(A)
```

```
[[ 4.          -9.          -5.        ]
 [ 0.           5.5         13.5       ]
 [ 0.           0.          57.81818182]]
```

In [7]:
```python
b = np.array([1, 0.9, -6])
x = np.linalg.solve(A, b)
```

In [8]:
```python
print(x)
```

```
[ 1.06157804  0.41835334 -0.10377358]
```
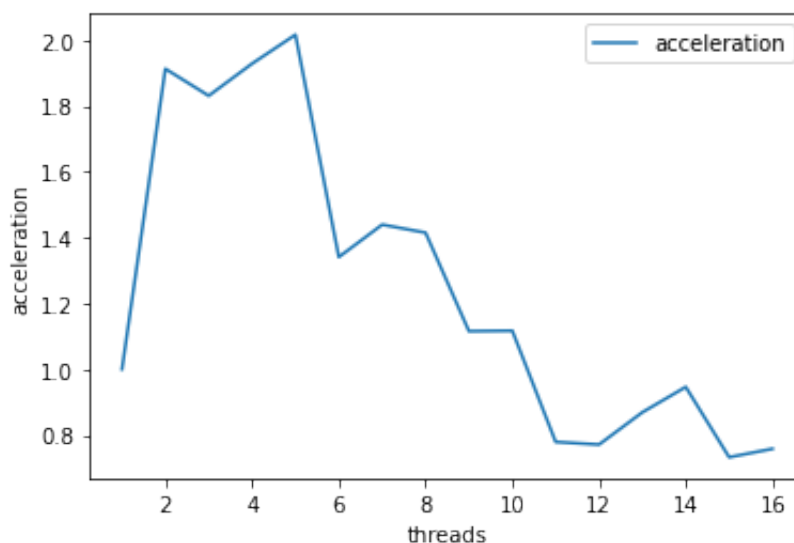
Графики эффективности и ускорения

In [9]:
```python
par_cores = np.genfromtxt('sys_of_eq/sys_cores_par.txt')
par_dim = np.genfromtxt('sys_of_eq/sys_dim_par.txt')
posl_dim = np.genfromtxt('sys_of_eq/sys_dim_posl.txt')
```

In [18]:
```python
fig, ax = plt.subplots()

ax.plot(par_cores[:, 0], (par_cores[:, 1]/par_cores[0, 1])**-1, lab
ax.set_xlabel('threads')
ax.set_ylabel('acceleration')
ax.legend()
```
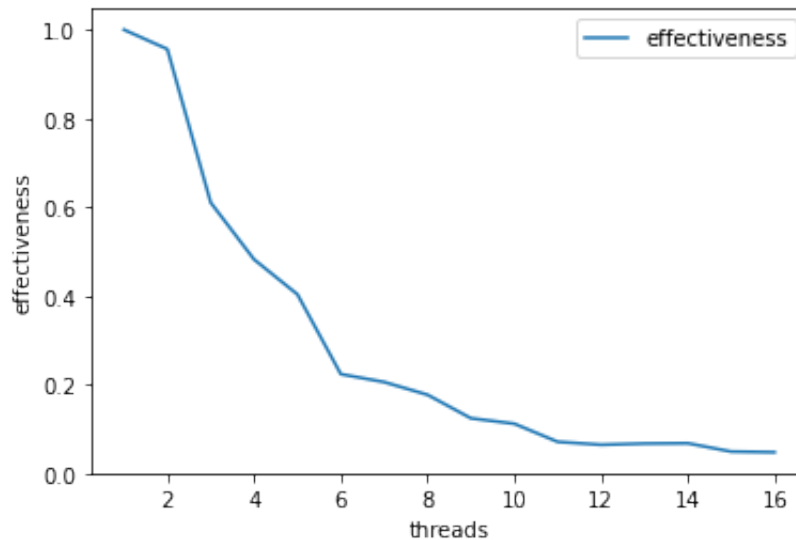
Out[18]: <matplotlib.legend.Legend at 0x7fb6fa590340>

In [16]:
```python
fig, ax = plt.subplots()

ax.plot(par_cores[:, 0], (par_cores[:, 1]/par_cores[0, 1])**(-1)/pa
ax.set_xlabel('threads')
ax.set_ylabel('effectiveness')
ax.legend()
```

Out[16]: <matplotlib.legend.Legend at 0x7fb6fa8865b0>

In [ ]: