44-44-44-64-6-beta-10.85-twist-2

October 17, 2024

[16]:
```
%load_ext autoreload
%autoreload 2
from modules import read_and_write
from modules import polyakov
from modules import utility
from modules import fourier_surface
import pandas as pd
import os
import glob
```

```
The autoreload extension is already loaded. To reload it, use:
  %reload_ext autoreload
```

For reference with integration method the following surface tensions were computed

z_1: $\alpha_{o-o}/T^3(\beta = 10.85) = 1.2316804724774406$

z_2: $\alpha_{o-o}/T^3(\beta = 10.85) = 1.5433288477348852$

1 Load data

[69]:
```
folder_names = ["../data/output-measure-surface/su4-36-36-48-6",
                "../data/output-measure-surface/su4-44-44-64-6/beta-12-twist-2",
                "../data/output-measure-surface/su4-44-44-64-6/beta-10.
 ↪9-twist-2",
                "../data/output-measure-surface/su4-44-44-64-6/beta-10.
 ↪85-twist-1",
                "../data/output-measure-surface/su4-44-44-64-6/beta-10.
 ↪85-twist-2"]
choose_folder = 5
fourier_profiles = {}
folder = folder_names[choose_folder-1]
files = glob.glob(os.path.join(folder, "fourier_profile_*"))
for file in files:
    file_name = file.split("/")[-1]
    smearing_level = file_name.split("_")[-1]
    volume, fourier_profile = read_and_write.read_surface_data(folder,␣
 ↪file_name)
```

```
    fourier_profiles[smearing_level] = fourier_profile
fourier_profiles = dict(sorted(fourier_profiles.items(), key=lambda item:␣
 ↪int(item[0])))
```

[72]:
```
utility.display_markdown_title(folder)
```

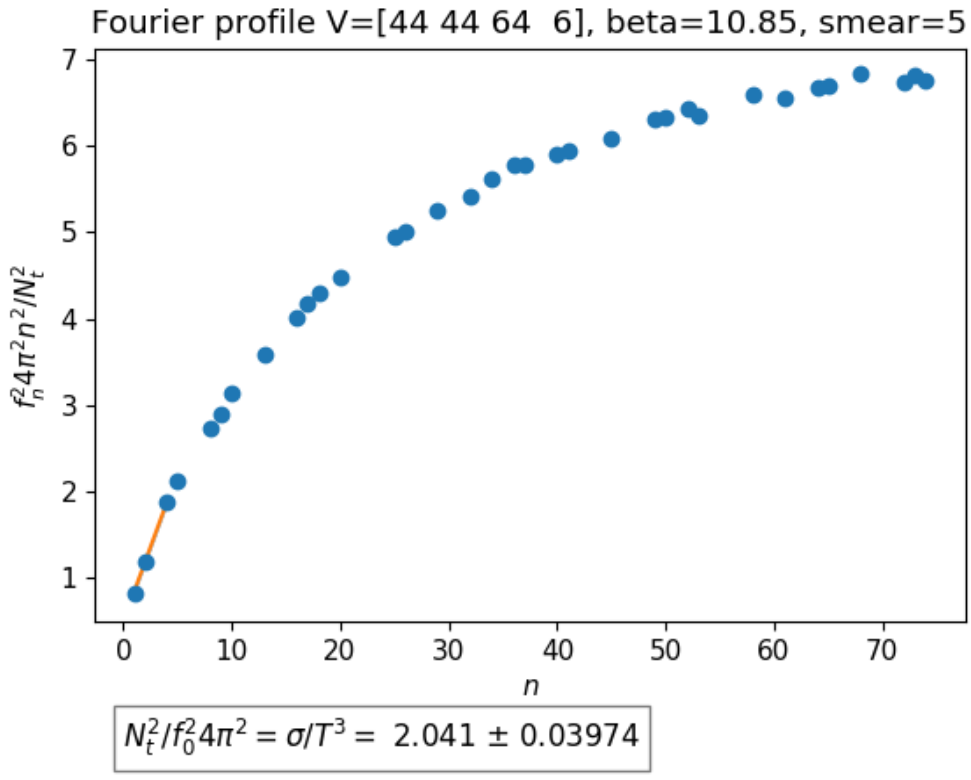
2 SU(4), V = ['44', '44', '64', '6'], β = 10.85, twist coeff = 2

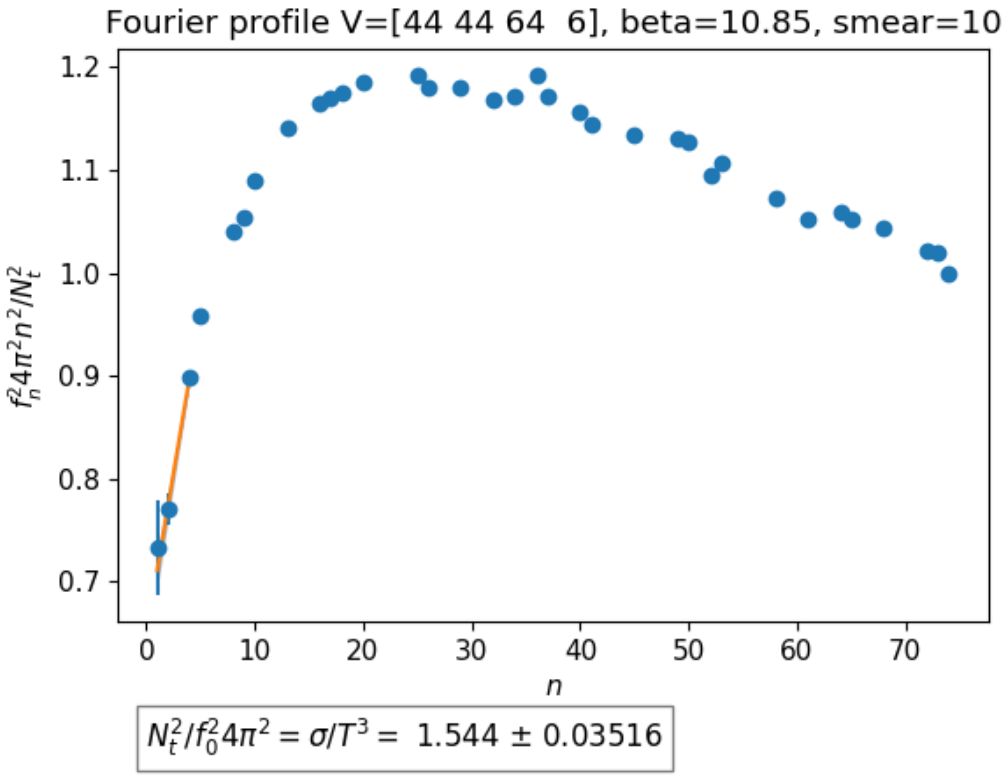
2.1 Perform post processing

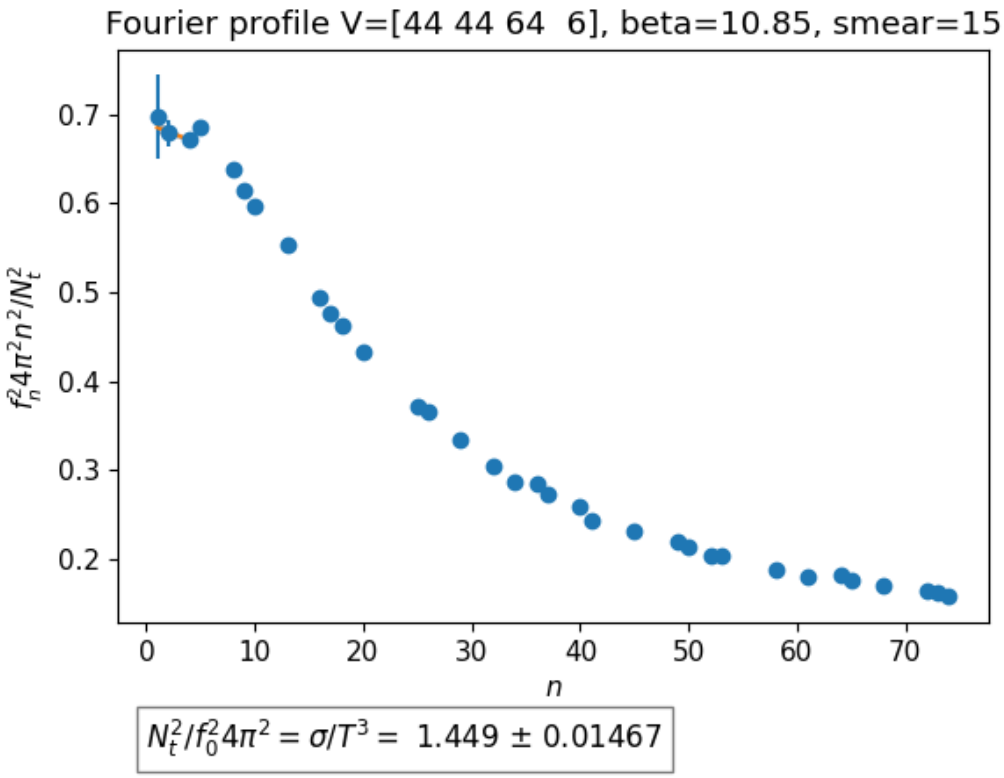
[70]:
```
n_2_list = []
f_n_list = []
errors_list = []
for smearing_level, profile in fourier_profiles.items():
    n_2, f_n, errors = utility.compute_with_aa_jackknife_fourier(profile, 10,␣
 ↪thermalization=100)
    n_2_list.append(n_2)
    f_n_list.append(f_n)
    errors_list.append(errors)
```

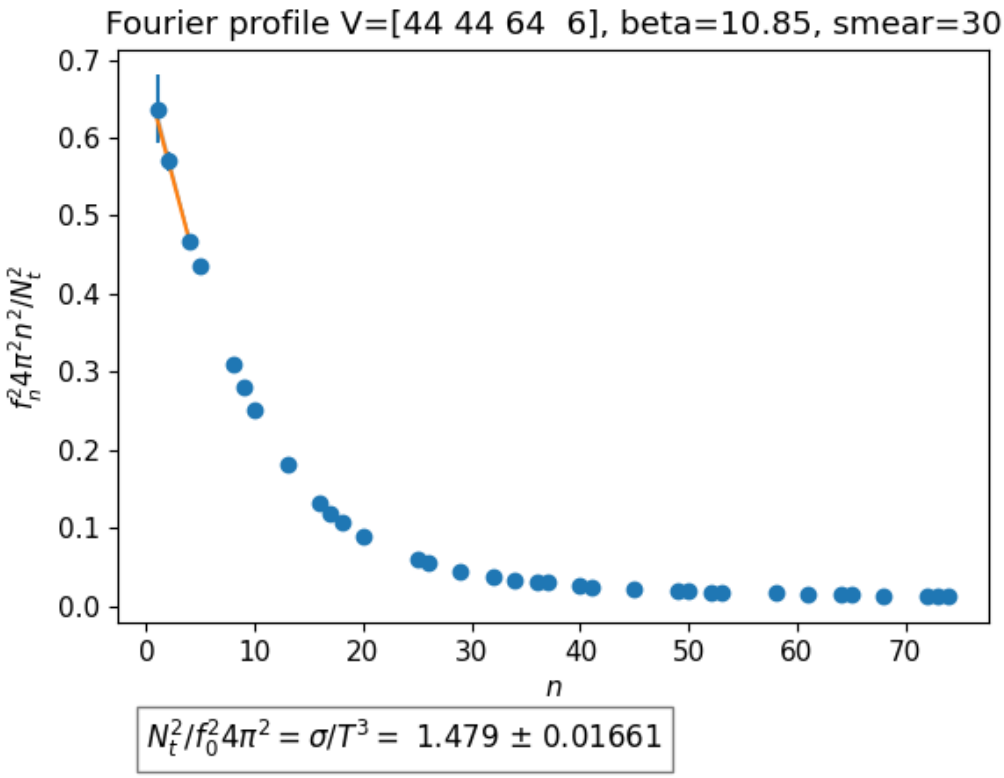
2.2 Plot Fourier modes for different smearing steps

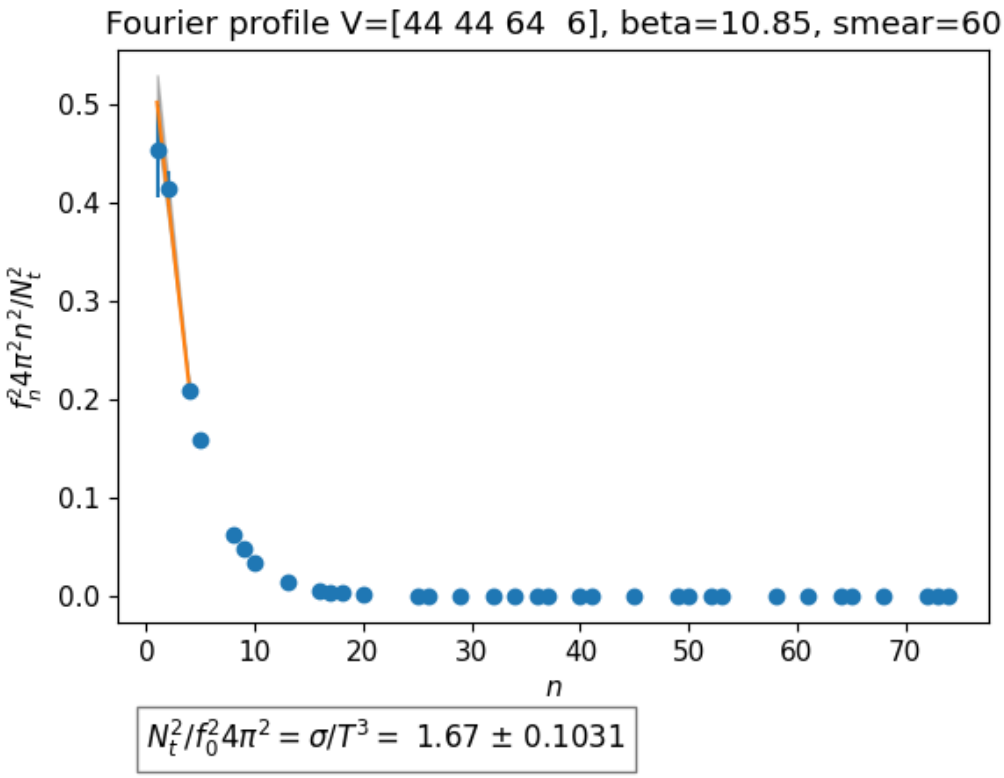
[71]:
```
%matplotlib widget
smearing_levels = list(fourier_profiles.keys())
show_plot = True
data = {
    "smearing": smearing_levels,
    "linear": [
        fourier_surface.compute_fourier_profile(
            n_2, f_n, volume, errors=error, beta=10.85, fit_range=3,␣
 ↪smearing=smear, show_plot=show_plot
        ) for n_2, f_n, error, smear in zip(n_2_list, f_n_list, errors_list,␣
 ↪smearing_levels)
    ],
    "exponential": [
        fourier_surface.compute_fourier_profile_exponential_fit(
            n_2, f_n, volume, errors=error, beta=10.85, smearing=smear,␣
 ↪show_plot=show_plot
        ) for n_2, f_n, error, smear in zip(n_2_list, f_n_list, errors_list,␣
 ↪smearing_levels)
    ]
}

df = pd.DataFrame(data)
utility.print_df_as_markdown_fourier_modes(df)
```

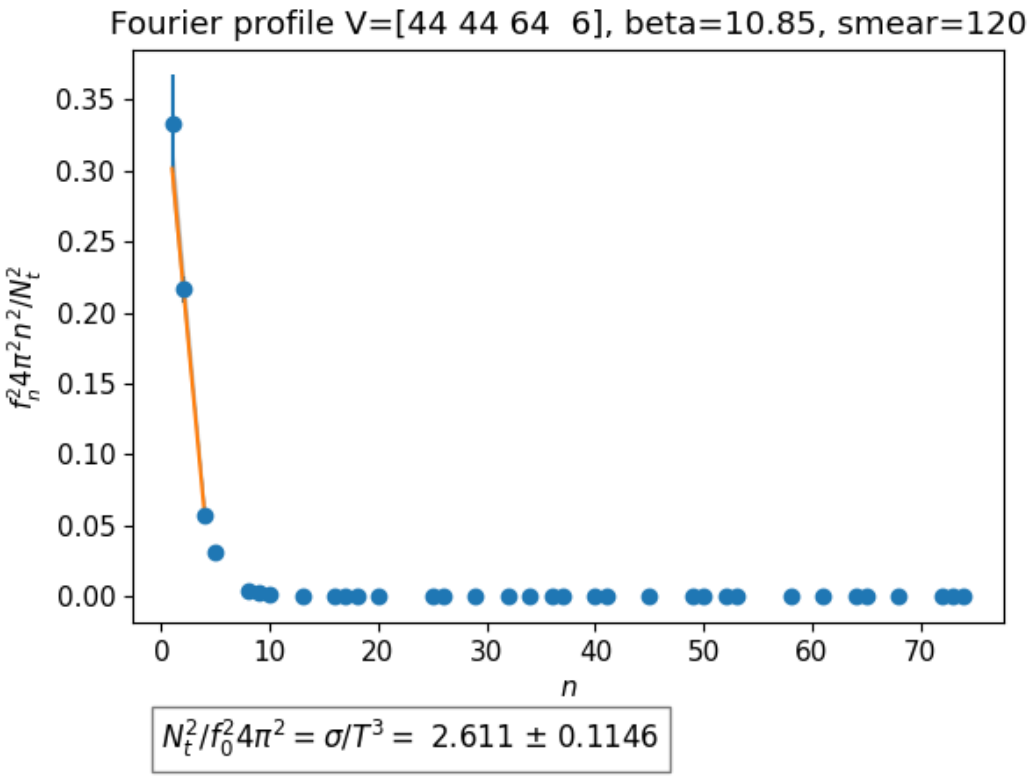


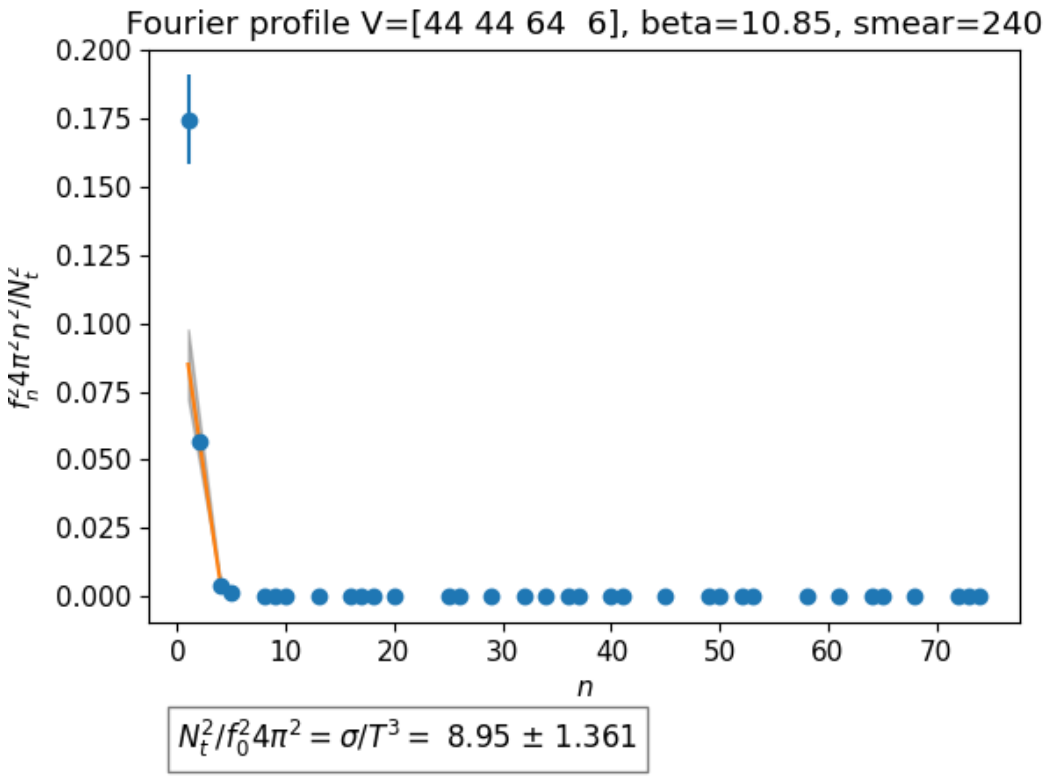

Fourier profile V=[44 44 64 6], beta=10.85, smear=10
$f_n^2 4\pi^2 n^2/N_t^2$
1.2
1.1
1.0
0.9
0.8
0.7
0
10
20
30
40
50
60
70
n
$N_t^2/f_0^2 4\pi^2 = \sigma/T^3 =$ 1.544 ± 0.03516

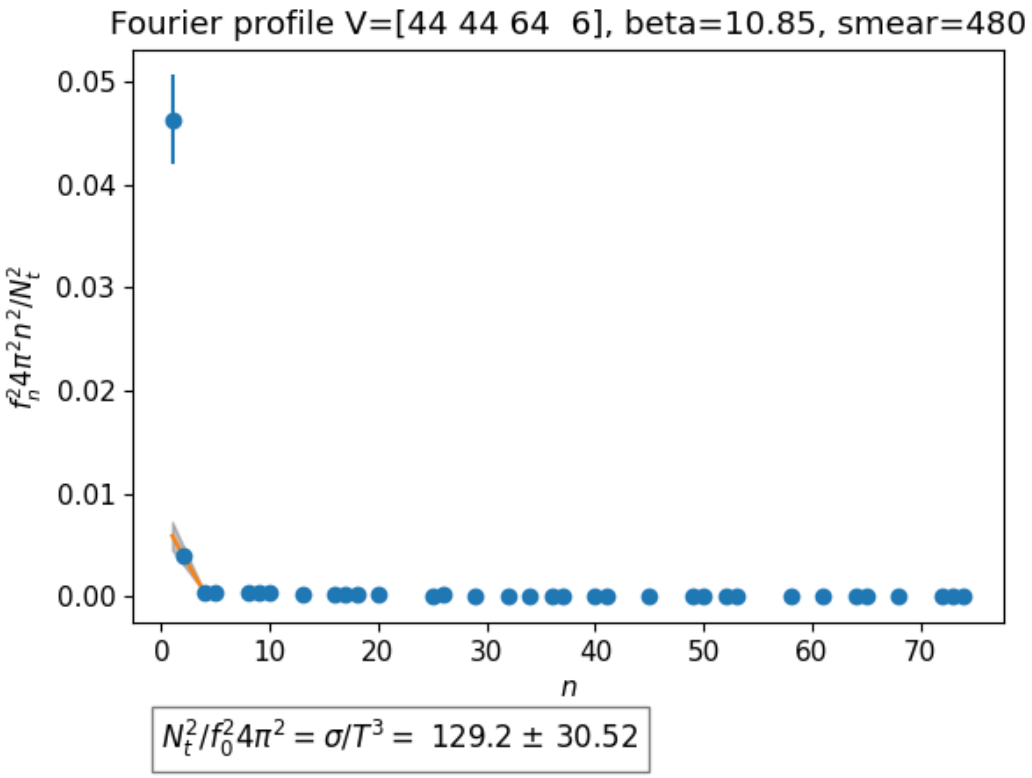

Fourier profile V=[44 44 64 6], beta=10.85, smear=15
$f_n^2 4\pi^2 n^2/N_t^2$
0.7
0.6
0.5
0.4
0.3
0.2
0
10
20
30
40
50
60
70
n
$N_t^2/f_0^2 4\pi^2 = \sigma/T^3 =$ 1.449 ± 0.01467


Fourier profile V=[44 44 64 6], beta=10.85, smear=30
$f_n^2 4\pi^2 n^2/N_t^2$
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0
10
20
30
40
50
60
70
n
$N_t^2/f_0^2 4\pi^2 = \sigma/T^3 =$ 1.479 ± 0.01661

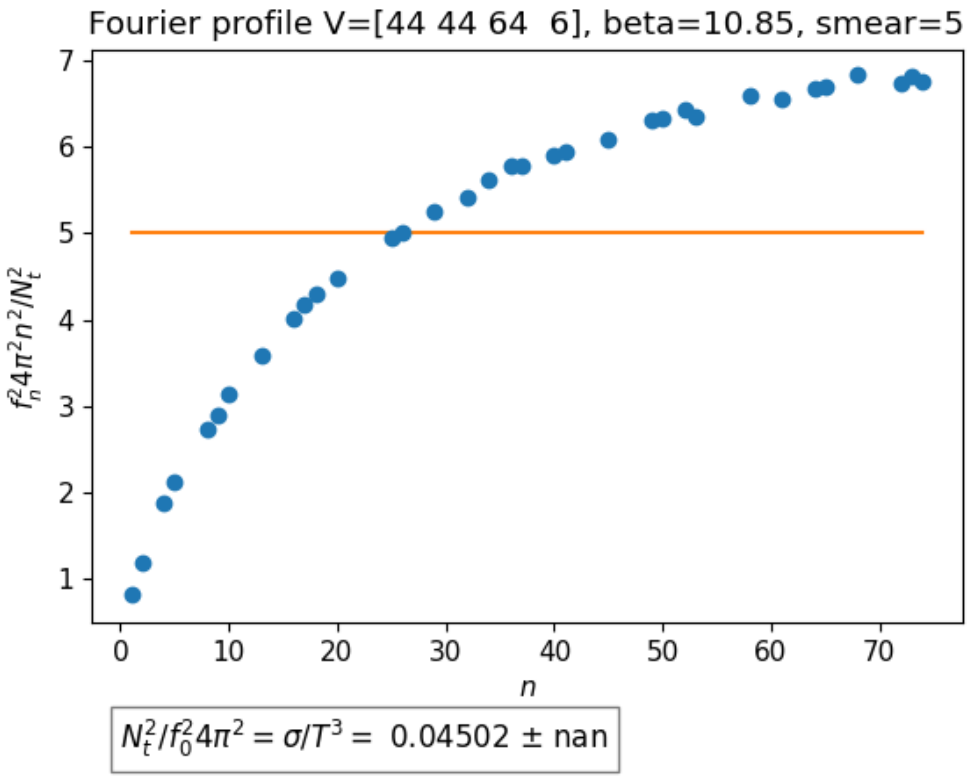

Fourier profile V=[44 44 64 6], beta=10.85, smear=60
$f_n^2 4\pi^2 n^2/N_t^2$
0.5
0.4
0.3
0.2
0.1
0.0
0
10
20
30
40
50
60
70
n
$N_t^2/f_0^2 4\pi^2 = \sigma/T^3 =$ 1.67 ± 0.1031

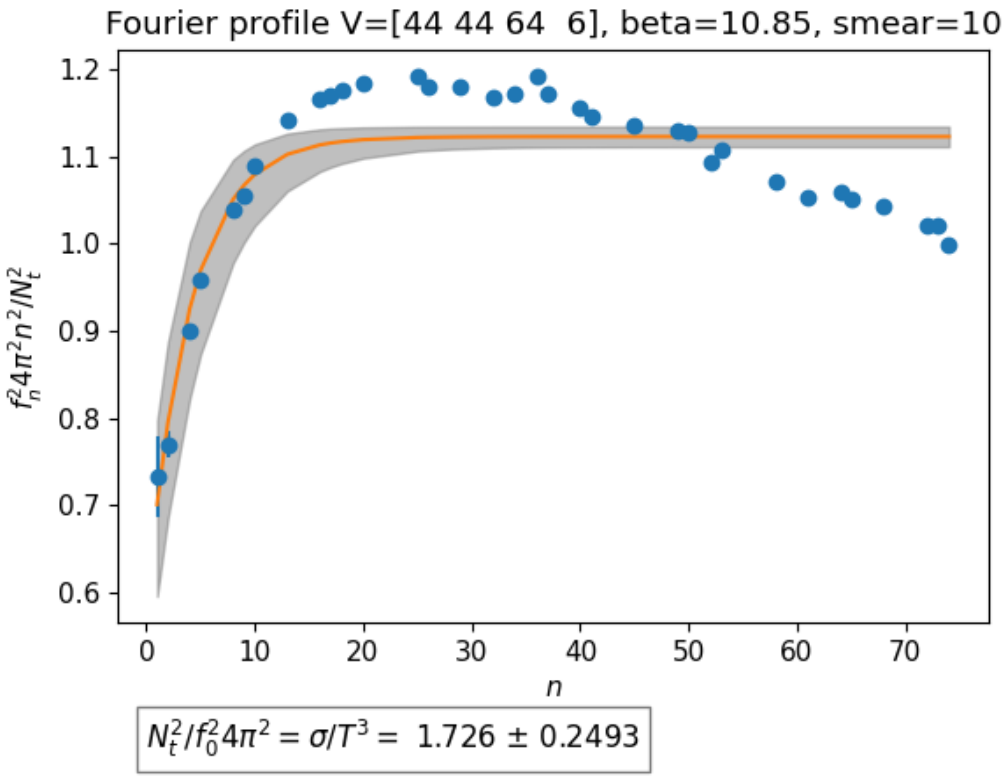

Fourier profile V=[44 44 64 6], beta=10.85, smear=120
$f_n^2 4\pi^2 n^2/N_t^2$
0.35
0.30
0.25
0.20
0.15
0.10
0.05
0.00
0
10
20
30
40
50
60
70
n
$N_t^2/f_0^2 4\pi^2 = \sigma/T^3 =$ 2.611 ± 0.1146

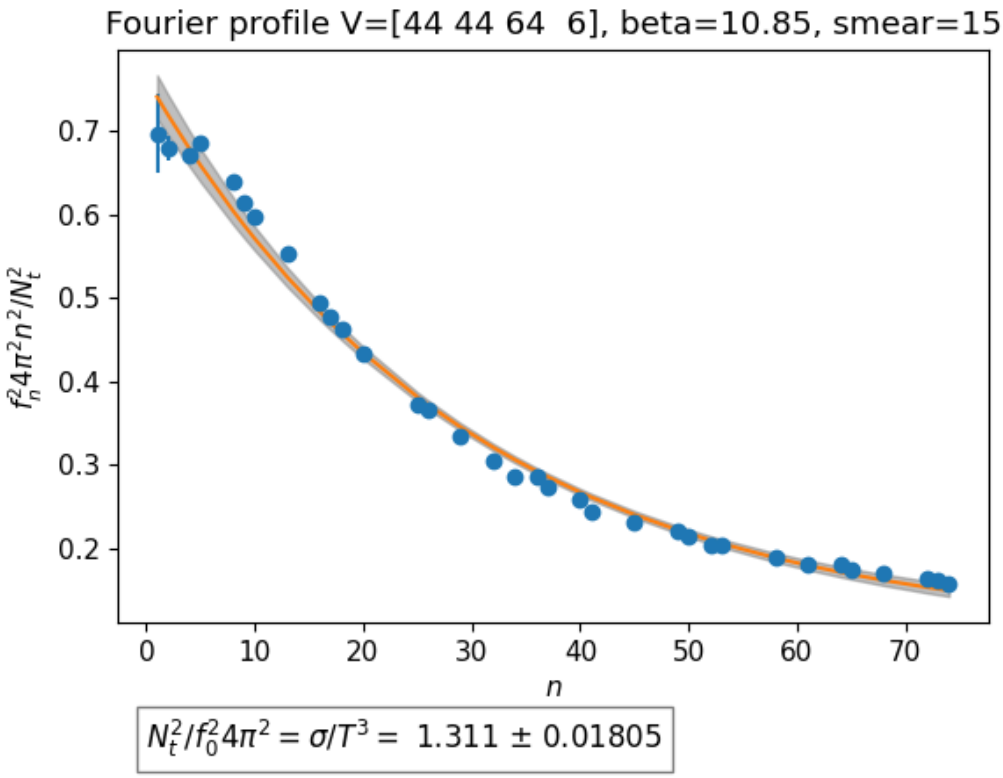

Fourier profile V=[44 44 64 6], beta=10.85, smear=240
0.200
0.175
0.150
0.125
0.100
0.075
0.050
0.025
0.000
$f_n^2 4\pi^2 n^2/N_t^2$
0
10
20
30
40
50
60
70
n
$N_t^2/f_0^2 4\pi^2 = \sigma/T^3 =$ 8.95 ± 1.361

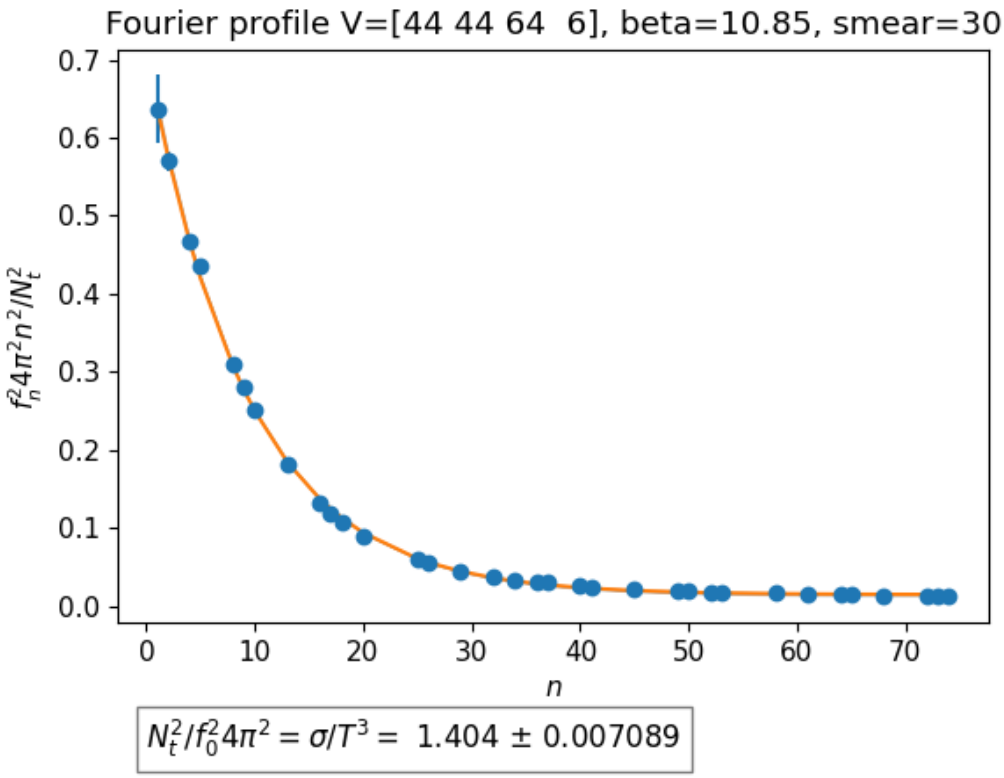


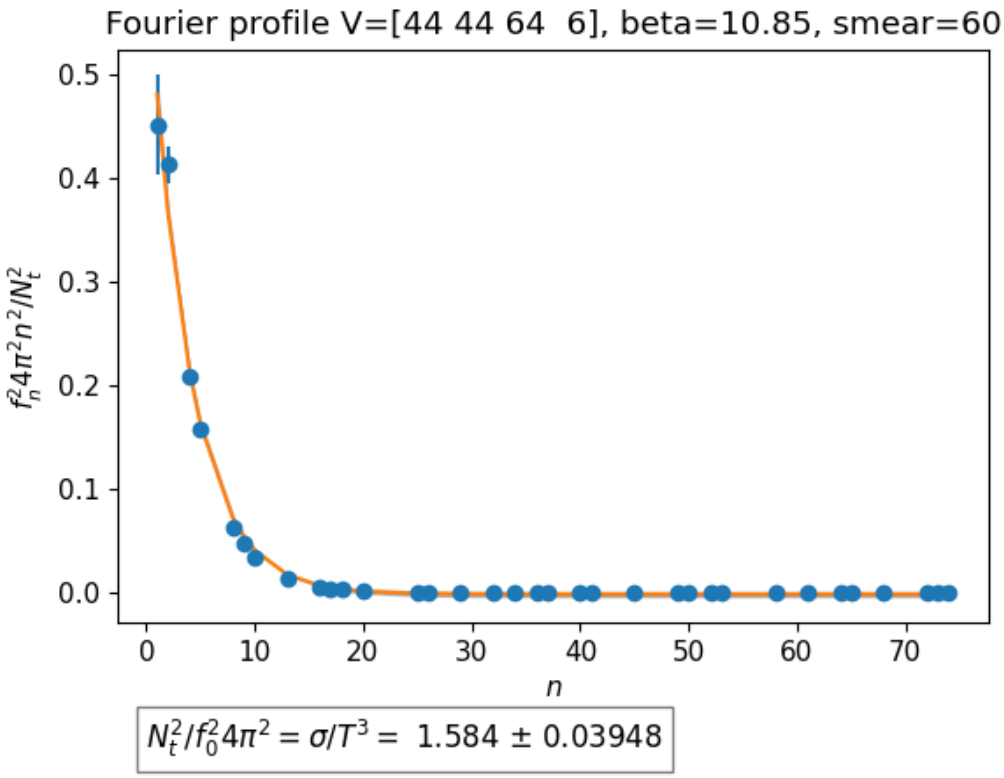

```
/home/haaaaron/SUN_twist_python_analysis/env/lib/python3.10/site-
packages/scipy/optimize/_minpack_py.py:1010: OptimizeWarning: Covariance of the
parameters could not be estimated
  warnings.warn('Covariance of the parameters could not be estimated',
/home/haaaaron/SUN_twist_python_analysis/modules/fourier_surface.py:93:
RuntimeWarning: invalid value encountered in multiply
  return a * np.exp(-b * x) + c
/home/haaaaron/SUN_twist_python_analysis/modules/fourier_surface.py:153:
RuntimeWarning: invalid value encountered in matmul
```

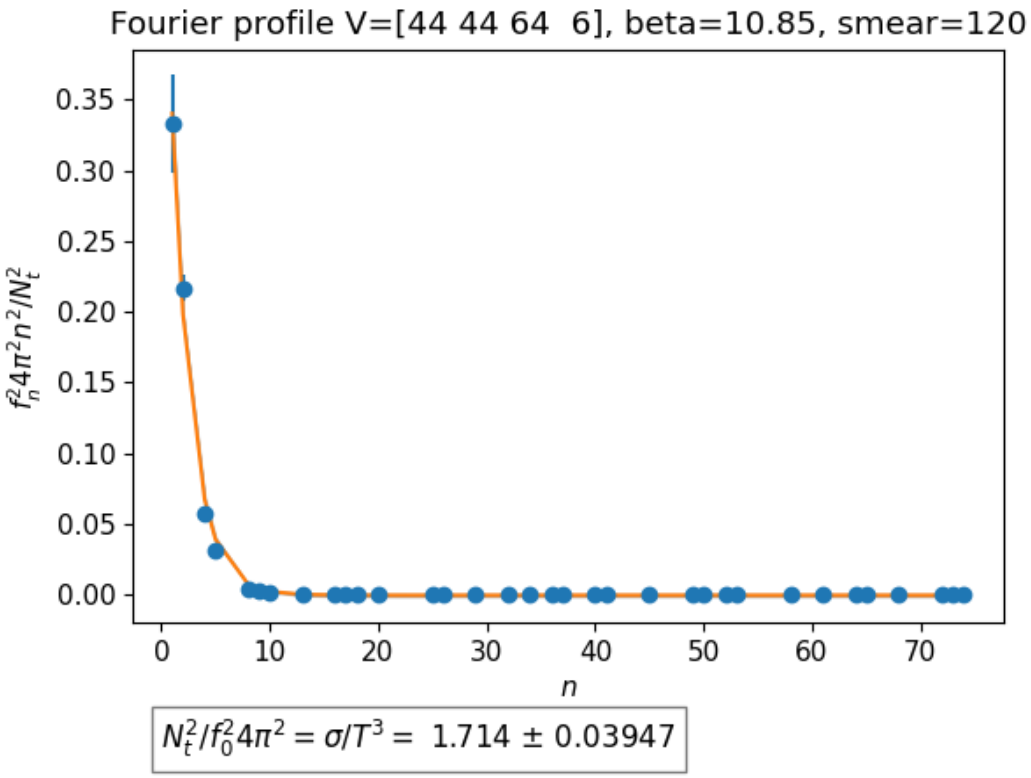

Fourier profile V=[44 44 64 6], beta=10.85, smear=5
$f_n^2 4\pi^2 n^2/N_t^2$
n
$N_t^2/f_0^2 4\pi^2 = \sigma/T^3 =$ 0.04502 ± nan

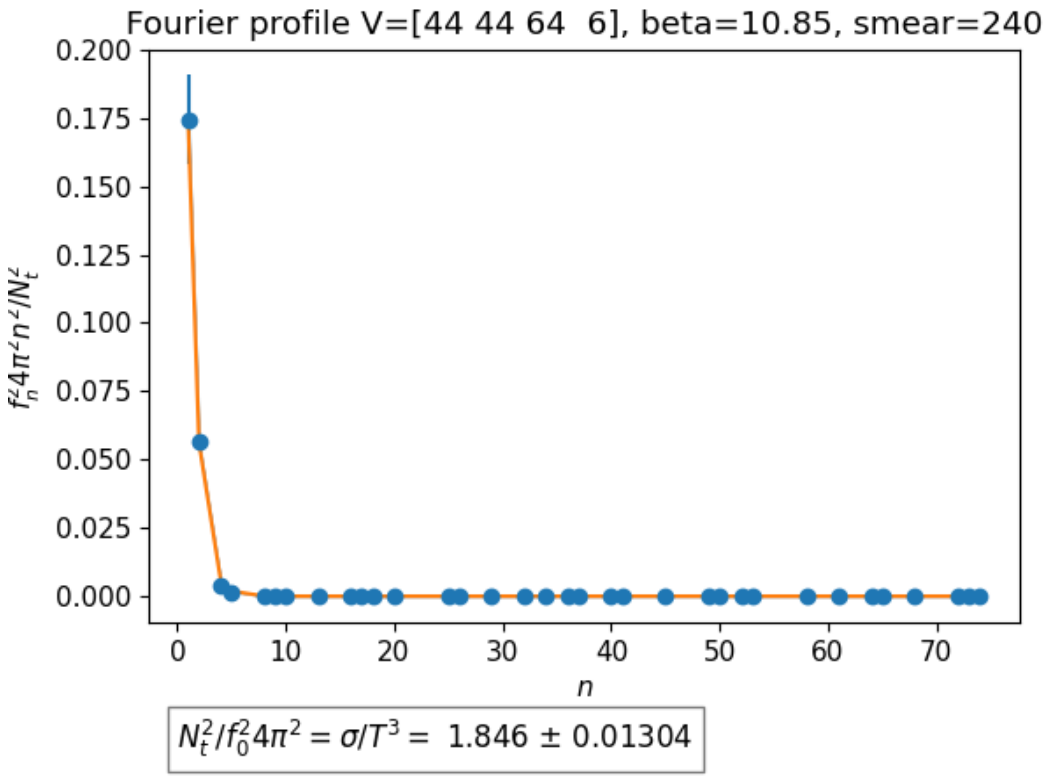

Fourier profile V=[44 44 64 6], beta=10.85, smear=10
$f_n^2 4\pi^2 n^2/N_t^2$
1.2
1.1
1.0
0.9
0.8
0.7
0.6
0
10
20
30
40
50
60
70
n
$N_t^2/f_0^2 4\pi^2 = \sigma/T^3 =$ 1.726 ± 0.2493

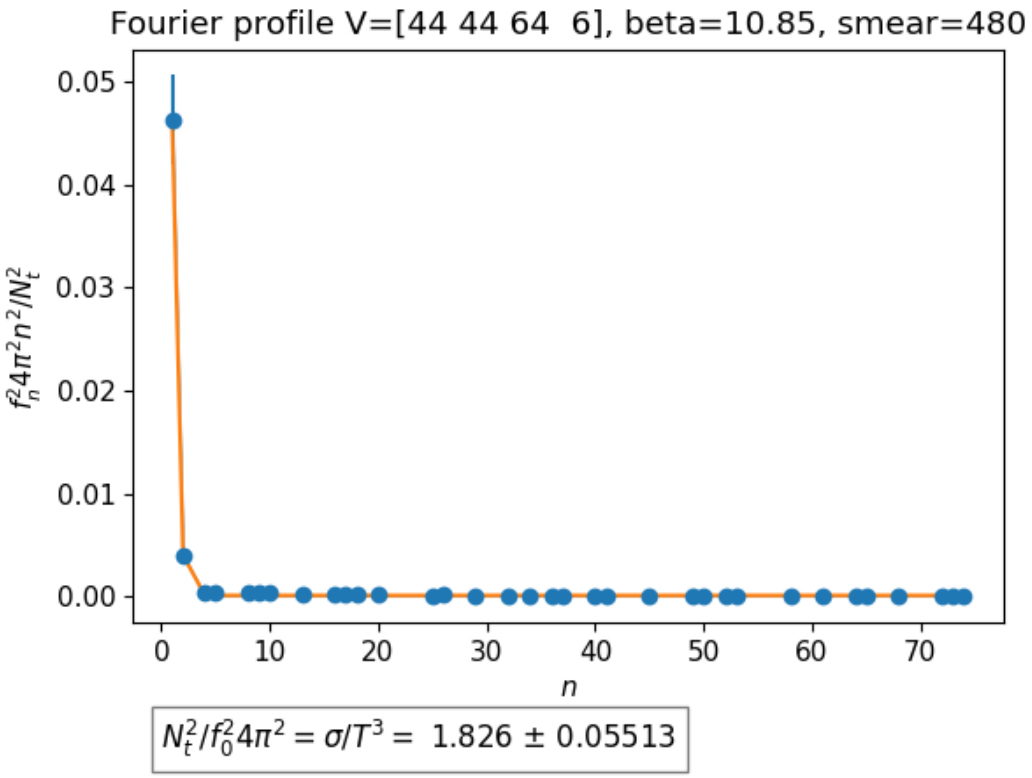

Fourier profile V=[44 44 64 6], beta=10.85, smear=15
$f_n^2 4\pi^2 n^2/N_t^2$
0.7
0.6
0.5
0.4
0.3
0.2
0
10
20
30
40
50
60
70
n
$N_t^2/f_0^2 4\pi^2 = \sigma/T^3 =$ 1.311 ± 0.01805


Fourier profile V=[44 44 64 6], beta=10.85, smear=30
$f_n^2 4\pi^2 n^2/N_t^2$
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0
10
20
30
40
50
60
70
n
$N_t^2/f_0^2 4\pi^2 = \sigma/T^3 =$ 1.404 ± 0.007089


Fourier profile V=[44 44 64 6], beta=10.85, smear=60
$f_n^2 4\pi^2 n^2/N_t^2$
0.5
0.4
0.3
0.2
0.1
0.0
0
10
20
30
40
50
60
70
n
$N_t^2/f_0^2 4\pi^2 = \sigma/T^3 =$ 1.584 ± 0.03948


Fourier profile V=[44 44 64 6], beta=10.85, smear=120
$f_n^2 4\pi^2 n^2/N_t^2$
0.35
0.30
0.25
0.20
0.15
0.10
0.05
0.00
0
10
20
30
40
50
60
70
n
$N_t^2/f_0^2 4\pi^2 = \sigma/T^3 =$ 1.714 ± 0.03947


Fourier profile V=[44 44 64 6], beta=10.85, smear=240
0.200
0.175
0.150
0.125
0.100
0.075
0.050
0.025
0.000
$f_n^2 4\pi^2 n^2/N_t^2$
0
10
20
30
40
50
60
70
n
$N_t^2/f_0^2 4\pi^2 = \sigma/T^3 =$ 1.846 ± 0.01304



smearing	Linear fit (σ/T^3)	Exponential fit (σ/T^3)
5	2.041 ± 0.03974	0.04502 ± nan
10	1.544 ± 0.03516	1.726 ± 0.2493
15	1.449 ± 0.01467	1.311 ± 0.01805
30	1.479 ± 0.01661	1.404 ± 0.007089
60	1.67 ± 0.1031	1.584 ± 0.03948
120	2.611 ± 0.1146	1.714 ± 0.03947
240	8.95 ± 1.361	1.846 ± 0.01304
480	129.2 ± 30.52	1.826 ± 0.05513

[45]:
```
import matplotlib.pyplot as plt

plt.close('all')
```

[ ]:
```

```