60-60-86-6-beta-10.83-twist-1

October 31, 2024

[1]:
```
%load_ext autoreload
%autoreload 2
from modules import read_and_write
from modules import polyakov
from modules import utility
from modules import fourier_surface
from modules import surface_amplitudes as sf

import pandas as pd
import numpy as np
import os
import glob
indices = None
```

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

[2]:
```
from modules.globals import folder_names

choose_folder = 4
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, modes, fourier_profile = read_and_write.
 ↪read_fourier_profile(folder,file_name=file_name)
    fourier_profiles[smearing_level] = fourier_profile
fourier_profiles = dict(sorted(fourier_profiles.items(), key=lambda item:␣
 ↪int(item[0])))
utility.display_markdown_title(folder)
```

2 SU(4), V = ['60', '60', '86', '6'], β = 10.85, twist coeff = 1

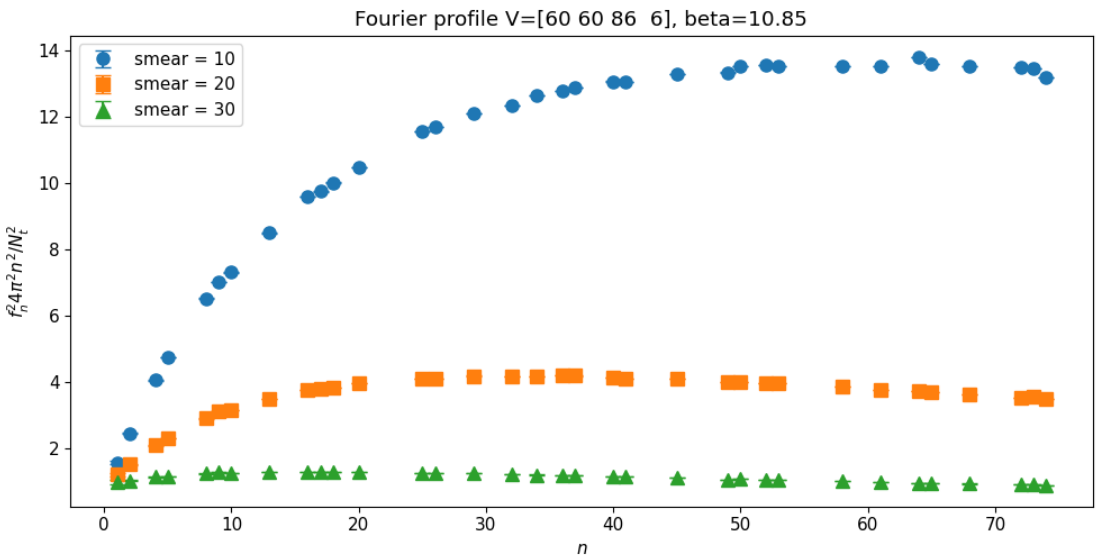
2.1 Perform post processing

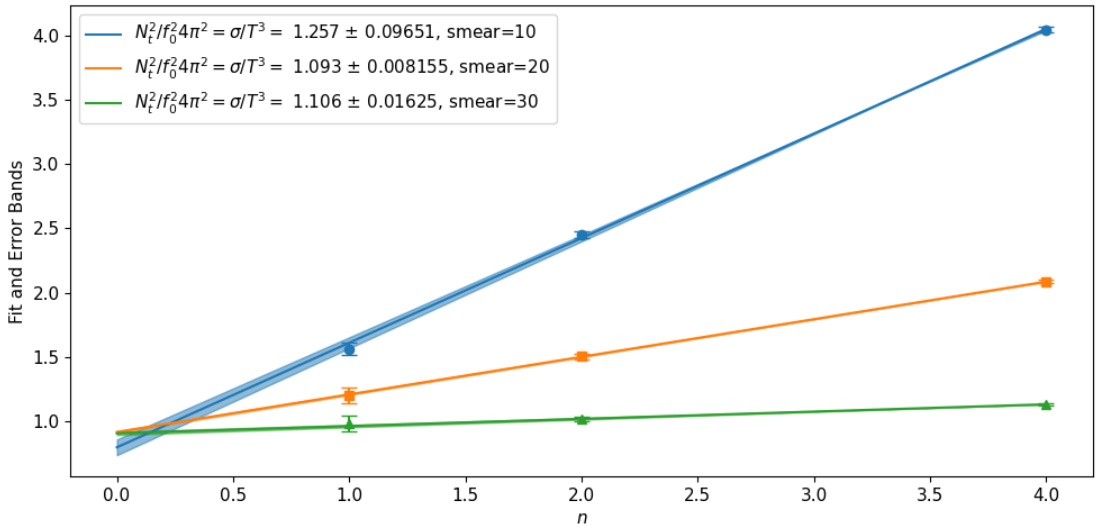
[3]:
```
f_n_list = []
errors_list = []
for smearing_level, profile in fourier_profiles.items():
    if indices is not None:
        sample_size = len(profile)
        indices_set = indices[smearing_level]
        profile = np.delete(profile, list(indices_set), axis=0)
        print(f"Dropped {sample_size-len(profile)} samples")
    f_n, errors = utility.compute_with_aa_jackknife_fourier(profile, 10,␣
 ↪thermalization=1000)
    f_n_list.append(f_n)
    errors_list.append(errors)
```

2.2 Plot Fourier modes for different smearing steps

[4]:
```
%matplotlib widget
smearing_levels = list(fourier_profiles.keys())
show_plot = True
data = {
    "smearing": smearing_levels,
    "linear": [
        fourier_surface.compute_fourier_profile(
            modes, f_n, volume, errors=error, beta=10.85, fit_range=3,␣
 ↪smearing=smear, show_plot=show_plot
        ) for f_n, error, smear in zip( 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_surface.global_fig = None
```

smearing	Linear fit (σ/T^3)
10	1.257 ± 0.09651
20	1.093 ± 0.008155
30	1.106 ± 0.01625


Fourier profile V=[60 60 86 6], beta=10.85
smear = 10
smear = 20
smear = 30
$f_n^2 4\pi^2 n^2/N_t^2$
n


$N_t^2/f_0^2 4\pi^2 = \sigma/T^3 =$ 1.257 ± 0.09651, smear=10
$N_t^2/f_0^2 4\pi^2 = \sigma/T^3 =$ 1.093 ± 0.008155, smear=20
$N_t^2/f_0^2 4\pi^2 = \sigma/T^3 =$ 1.106 ± 0.01625, smear=30
Fit and Error Bands
n

[ ]:

[ ]: