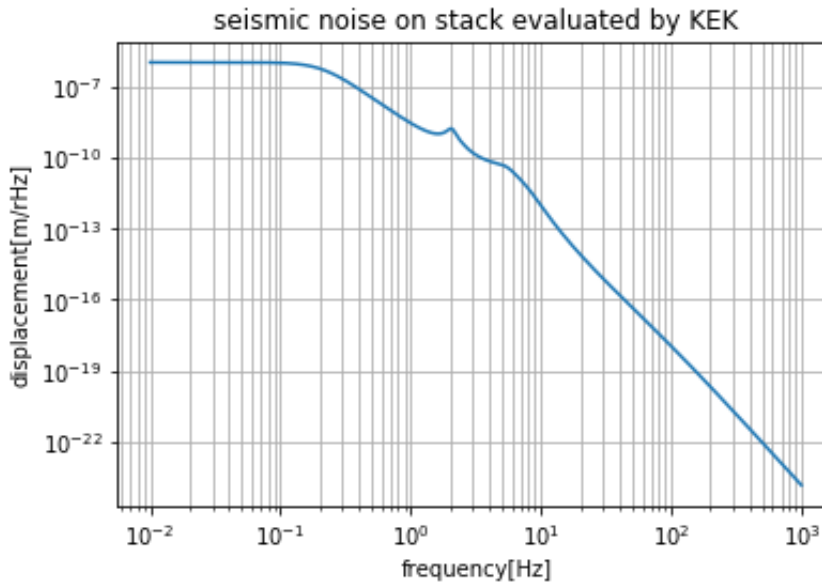


```
In [1]: import numpy as np
import matplotlib.pyplot as plt
freq1, mprHz1 = np.loadtxt("./stackKEK.dat", unpack=True)
plt.xscale("log")
plt.yscale("log")
plt.grid(which="both")
plt.xlabel(r"frequency[Hz]") # x-axis
plt.ylabel(r"displacement[m/rHz]") # y-axis
plt.title(r"seismic noise on stack evaluated by KEK")#title
plt.plot(freq1, mprHz1)
plt.show()
```



```
In [ ]: [Rough estimation of the mirror angular motion
due to extra static magnetic feature of OSEM]
```

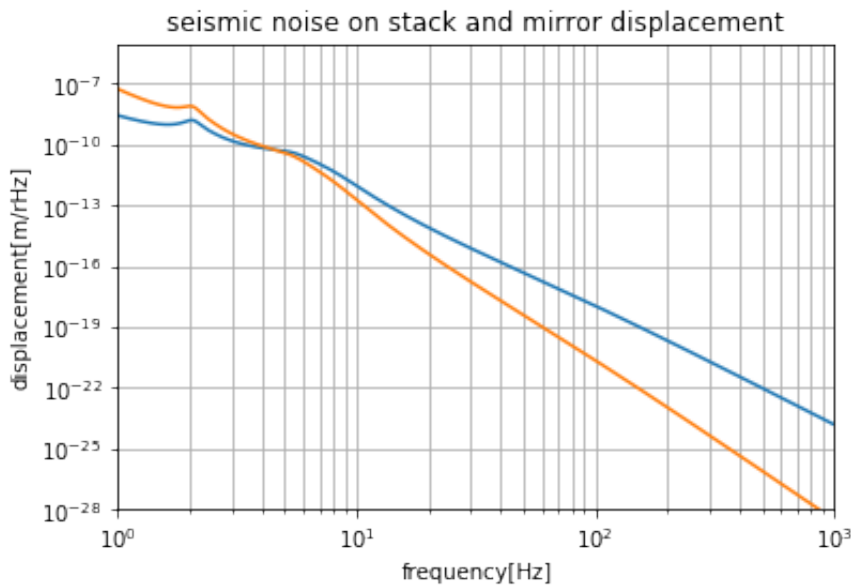
While the mirror **is** suspended by Type-C double pendulum, OSEM **is** fixed to its suspension frame. Then the relative displacement between the OSEM **and** the magnets on the mirror **is** about the same **as** the seismic motion on stack measured by KEK, **if** the mirror **is** isolated well, **in** particular **in** high frequency range.

Because the OSEM has a kind of extra magnetic feature, the mirror **is** accelerated due to this relative motion, about $k_1=0.2\text{N/mm}$ per 1 magnet; $L_5\phi^2$. There are 4 magnets on the mirror, so the coupling coefficient **is**, $k_4=0.8\text{[N/mm]}=800\text{[N/m]}$.

The mirror displacement by this force **is** about $m_a=k_4\text{[N/m]}*x_{\text{stack}}\text{[m/rHz]}$.

Mirror mass **is** about $m=1\text{kg}$, then $a=800\text{[s}^2*\text{m/rHz]}$. To make this into displacement x , divide the value by $(2\pi*f)^2$, $x_{\text{ex}}=x_{\text{stack}}*800/(2\pi*f)^2\text{[m/rHz]}$

```
In [2]: plt.xscale("log")
plt.yscale("log")
plt.grid(which="both")
plt.xlabel(r"frequency[Hz]") # x-axis
plt.ylabel(r"displacement[m/rHz]") # y-axis
plt.title(r"seismic noise on stack and mirror displacement")#title
plt.plot(freq1, mprHz1)
mprHz2=mprHz1*800/(2*np.pi*freq1)/(2*np.pi*freq1)
plt.plot(freq1, mprHz2)
plt.xlim([1,1000])
plt.ylim([1e-28, 1e-5])
plt.show()
```



Blue line shows the original stack motion.

Orange line shows excited motion by magnetic feature.

At low frequency region, the assumption of the equation is not valid.

```
In [ ]: If there is 10% imbalance of the mirror horizontal motion x_stack,
and convert it into angular motion by dividing it by the mirror diameter:
d=10cm, angular motion noise due to extra static magnetic feature,
x_a is evaluated as
x_a=x_stack[m/rHz]*0.1(assumed coupling)/0.1[m]=x_stack[rad/rHz]
```

```
In [ ]: Other way to estimate the angular motion is to take moment of inertia
into account.
```

$$I_z = M/12 * (3r^2 + L^2)$$

$$= 1[\text{kg}] / 12 * (3 * 5[\text{cm}^2] + 6[\text{cm}^2]) = 9.25[\text{kg} \cdot \text{cm}^2]$$

$$r * F = 5[\text{cm}] * k_1[\text{N/mm}] * 4 * x_{\text{stack}} * 0.1(\text{assumed imbalance})$$

$$= 5 * 0.2 * 4 * 0.1 * 10 * s_{\text{stack}} = 4 * x_{\text{stack}}[\text{N} \cdot \text{m/rHz}]$$

$$\text{radprHz} = r * F / I_z / (2\pi * f)^2$$

$$= 4 * x_{\text{stack}} / 9.25 / (2\pi * f)^2$$

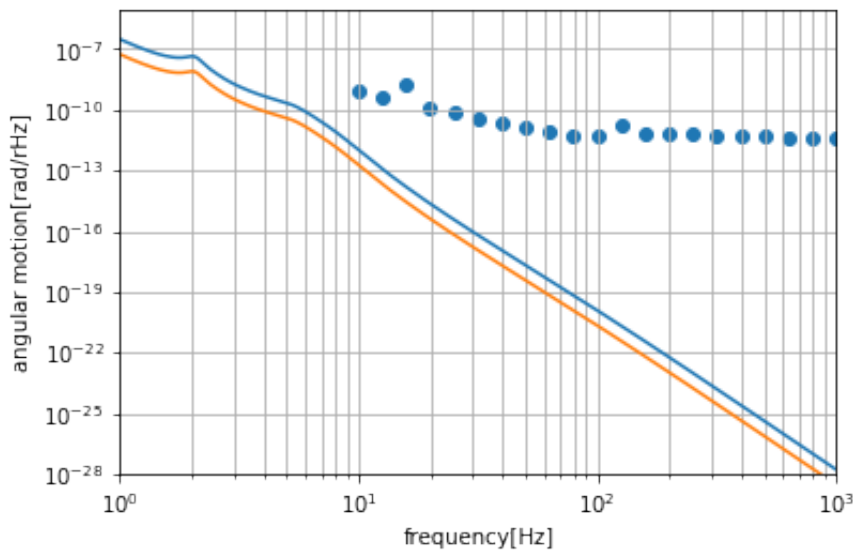
$$= 4444 * x_{\text{stack}} / (2\pi * f)^2 [\text{rad/rHz}]$$

```

In [3]: radprHz=mprHz1*4444/(2*np.pi*freq1)/(2*np.pi*freq1)
plt.xscale("log")
plt.yscale("log")
plt.grid(which="both")
plt.xlabel(r"frequency[Hz]") # x-axis
plt.ylabel(r"angular motion[rad/rHz]") # y-axis
plt.scatter(freq1, radprHz)
hor_req, ver_req, freq_req = np.loadtxt("./typeCreq.dat", unpack=True,
skiprows=1)

plt.plot(freq1, radprHz)
plt.plot(freq1, mprHz2)
plt.scatter(freq_req, ver_req)
plt.xlim([1, 1000])
plt.ylim([1e-28, 1e-5])
plt.show()

```



Blue dots show required upper limit of the mirror angular motion calculated by Somiya-san.
Blue line shows the angular motion estimated by using moment of inertia.
Orange line shows the angular motion estimated from 10% coupling from horizontal motion.
Both estimations are well below the requirement over 10Hz.