Лабораторная работа № 7
Липатов Данила Вячеславович
МСМТ243

I Пункт.

Запрограммировать матричную динамическую систему 6.9 (полученную из 4.10) вращения Земли с использованием функции ss () state-space model Simulink. Взять параметры T = 300 (сек), f_c = 365/433 (1/сут), Q = 100.

Для этого, согласно приведенной системе:

$$\dot{\mathbf{m}} = \mathbf{G}\mathbf{m} + \mathbf{F}\varphi, \tag{6.9}$$

где $\mathbf{m} = \begin{pmatrix} m_1 \\ m_2 \end{pmatrix}$, $\varphi = \begin{pmatrix} \varphi_1 \\ \varphi_2 \end{pmatrix}$, а матрицы легко получить в виде

$$\mathbf{G} = \begin{pmatrix} -\beta & -\alpha \\ \alpha & -\beta \end{pmatrix} = -\mathbf{F},$$

Необходимо считать входные сигналы φ_1 и φ_2 из исходного файла. Исходный файл есть ААМ (согласно ЛР № 1)

Отрисуем два входных сигнала (рис. 1)

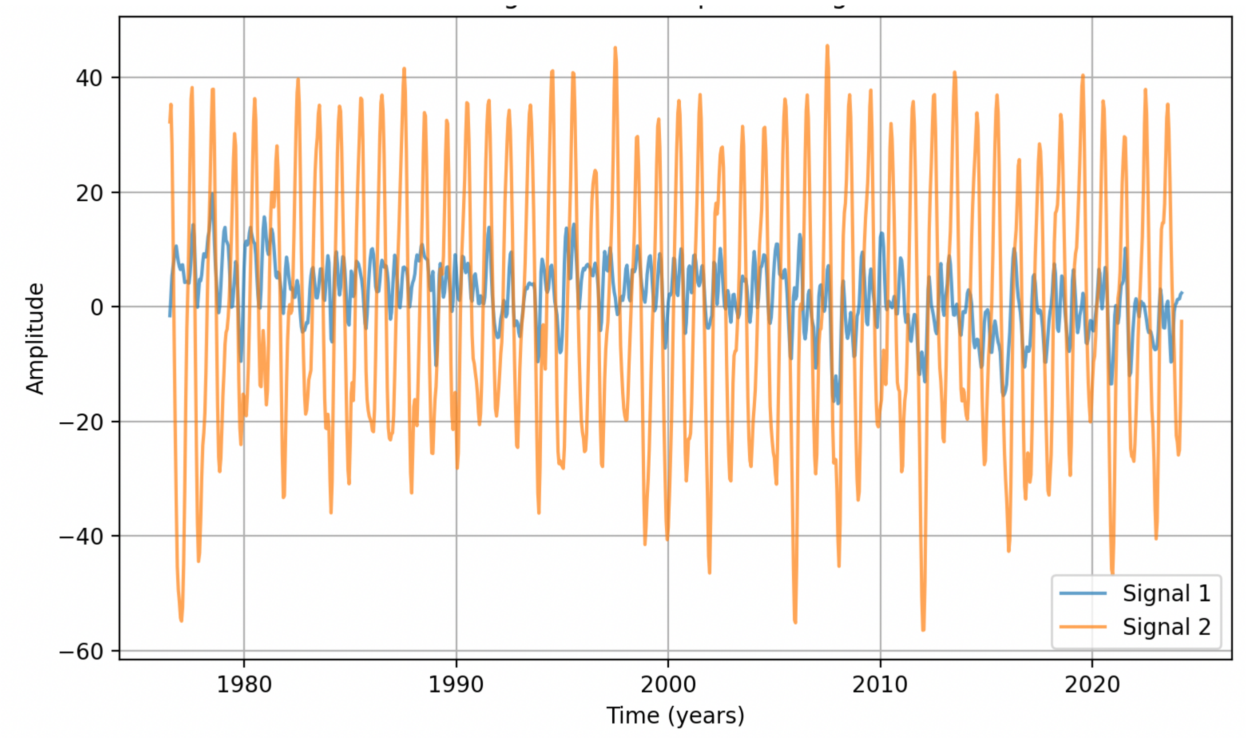


рис. 1

Далее необходимо использовать ss () state-space model Simulink для того, чтобы посмотреть на отклики для каждого сигнала (вектор m) (рис.2)

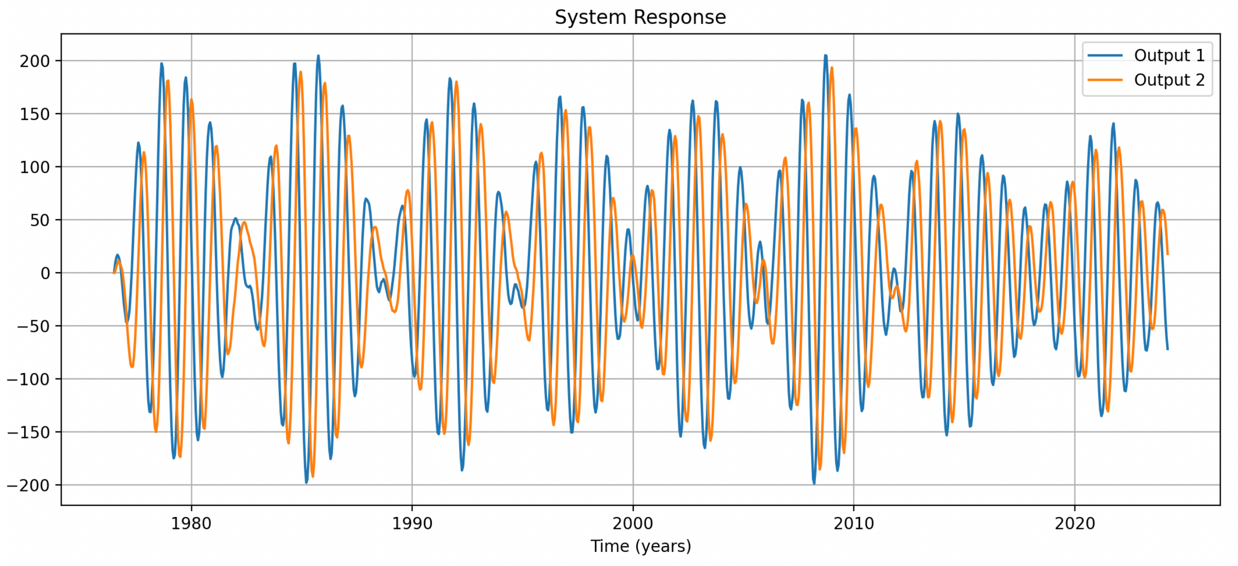


рис.2

Данные отклики были получены с учетом параметров из условия задачи, а точнее
Взять параметры T=300 сек,   f_c=365/433 сут$^{-1}$, Q=100.

Так же необходимо посмотреть единичный импульс и функцию Хевисайда (ступенчатая)

Единичный импульс позволяет оценить импульсную характеристику системы. Она показывает, как система реагирует на мгновенное изменение входного сигнала.

Функция Хевисайда используется для определения переходной характеристики **системы**. Это реакция системы на постоянное изменение входного сигнала.

Для этого можно воспользоваться аналогом встроенных функций Matlab на Python из библиотеки control: ctrl.step_response и ctrl.impulse_response. Причем, данные функции возвращают двумерный массив, то есть два столбца, а значит, рассматривать будем по два графика (рис. 3 – рис.6)

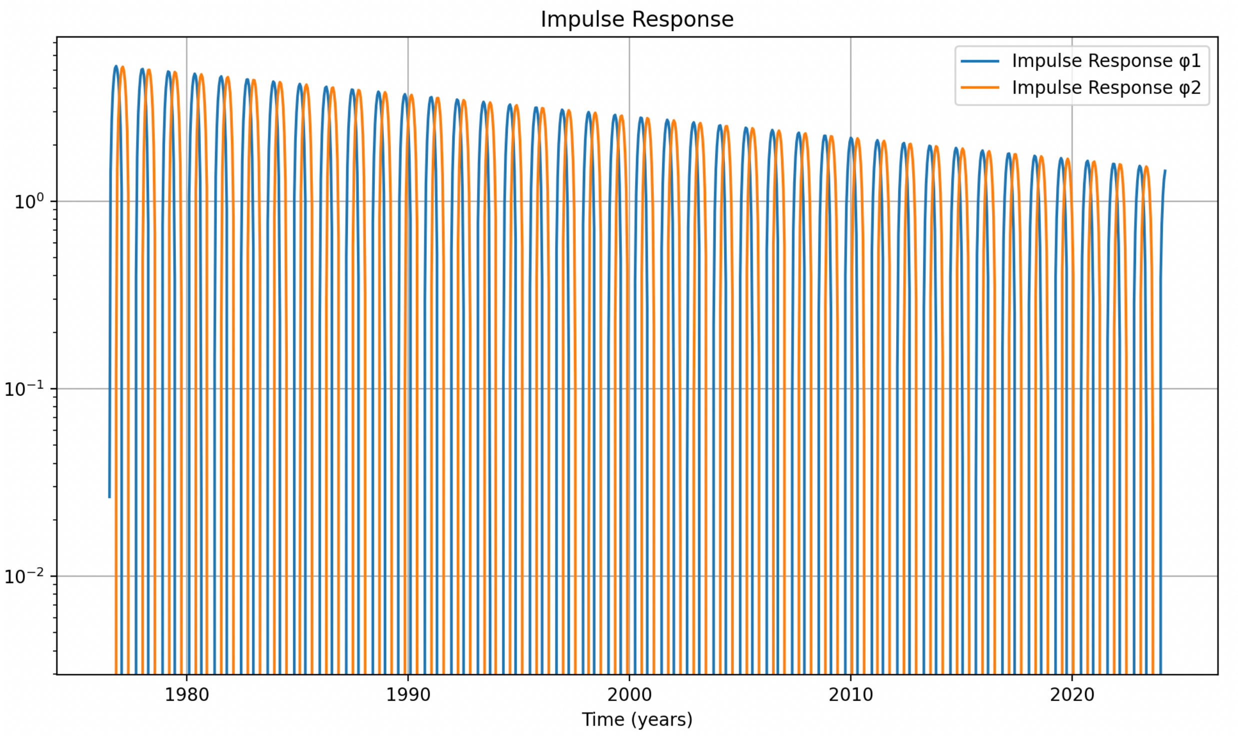


рис. 3

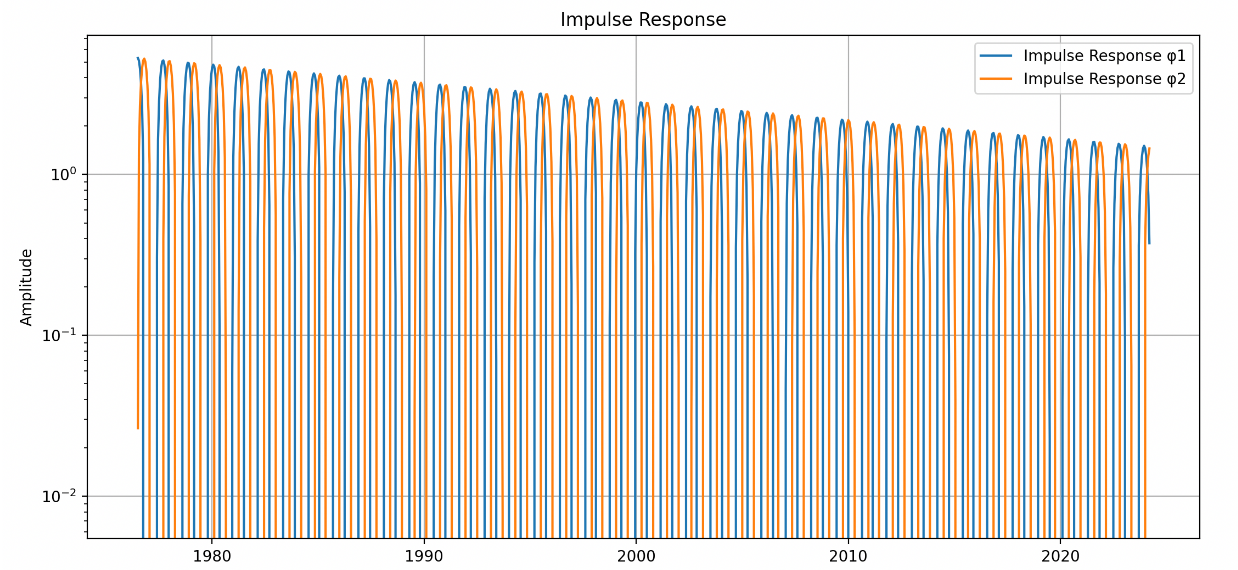


рис. 4

И аналогично рассмотрим функцию Хевисайда

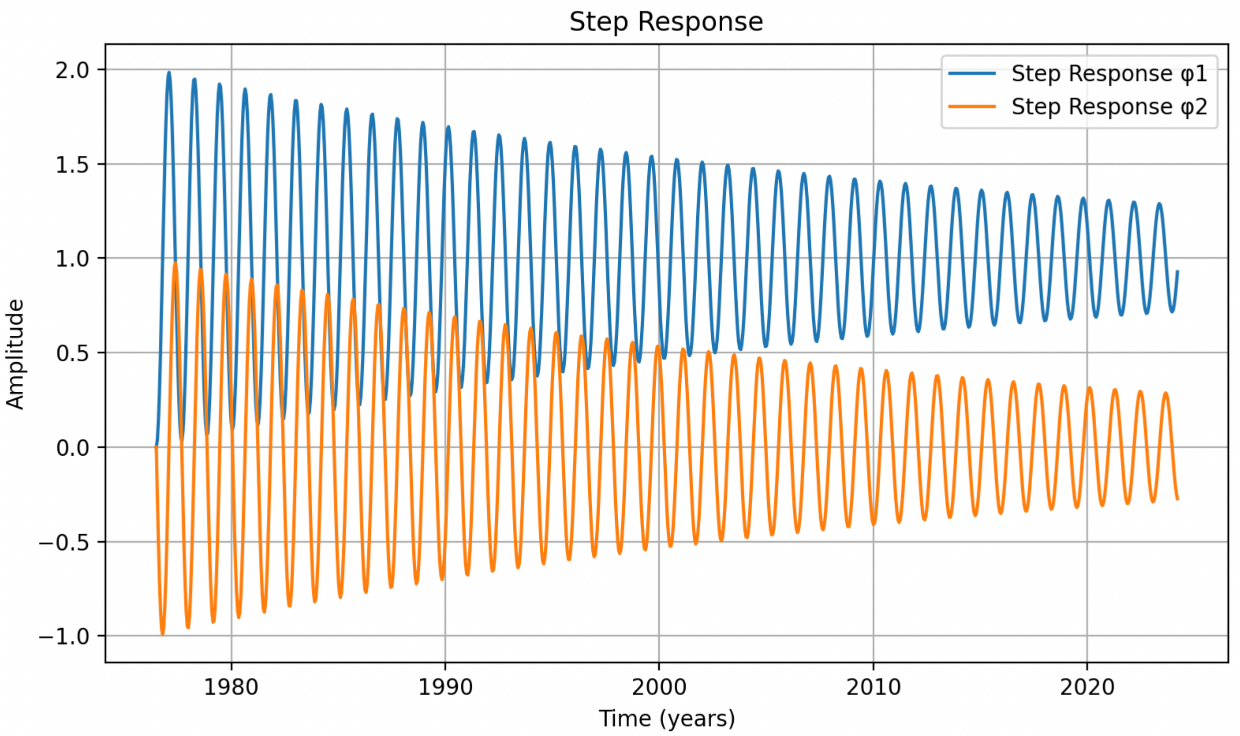


рис. 5

И для второго столбца:

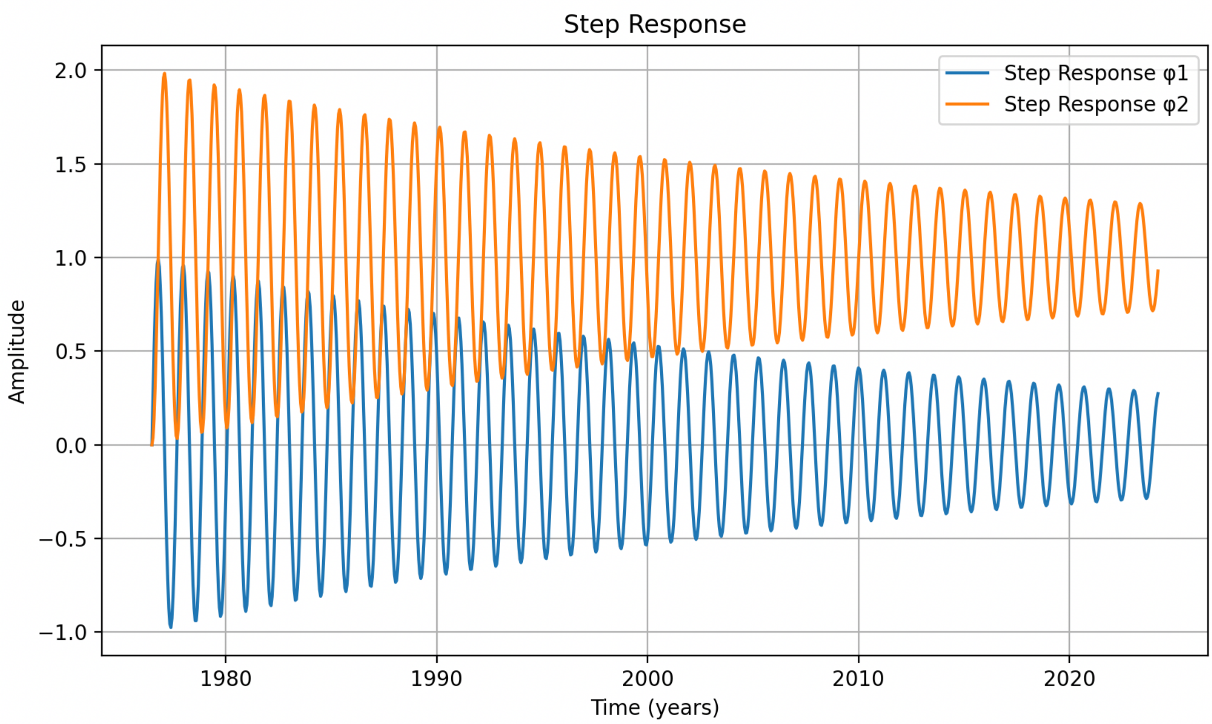


рис. 6

Аналогично Matlab в Python есть встроенная функция библиотеки control bode(sys)
Она выводит 4 АЧХ и 4 ФЧХ (амплитудно-частотная характеристика и фазовая-частотная характеристика).

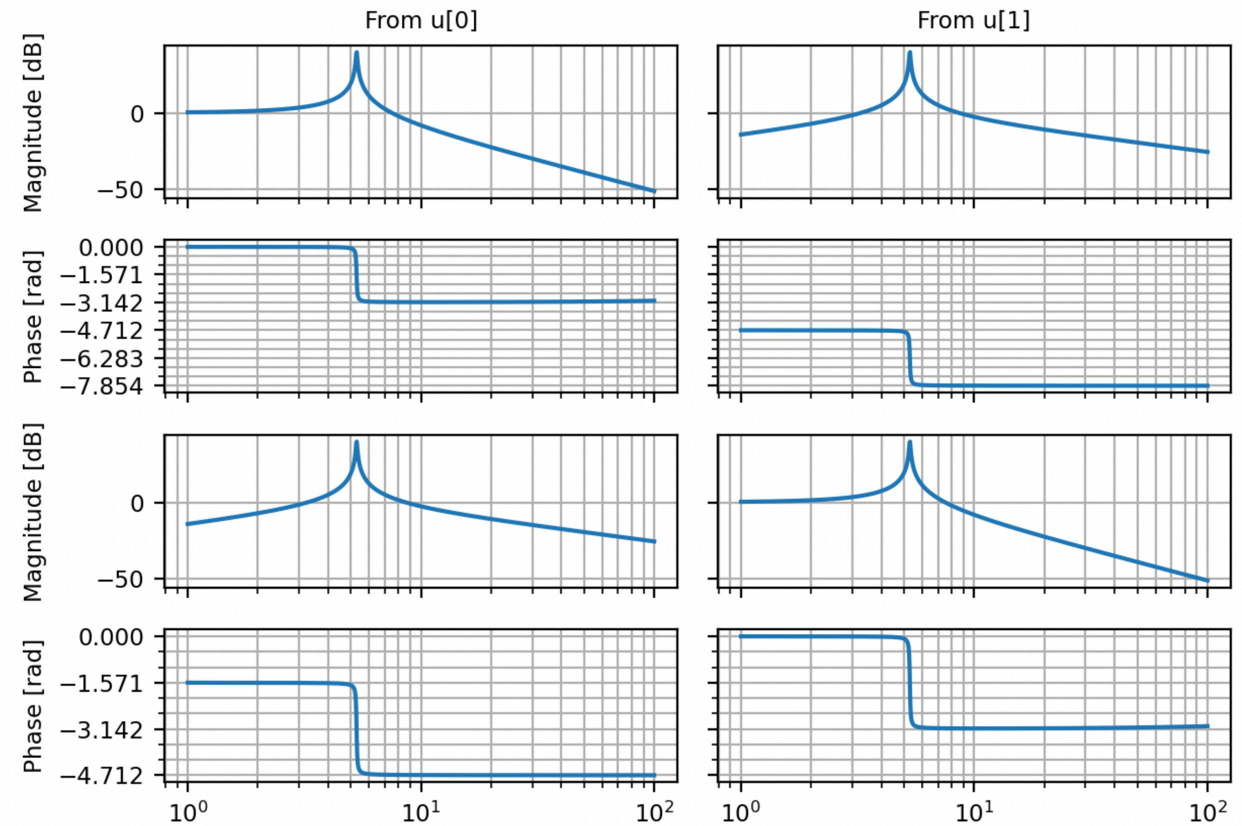


рис. 7

Все графики показывают условный резонанс и АЧХ и ФЧХ на каждом из графике отображают поведение друг друга.

На графиках ФЧХ происходит резкий переход фазы при резонансной частоте. Это является характерным поведением резонансных систем.

Роль параметров:
Q - Характеризует относительную ширину резонансного пика в системе.

Высокое значение Q означает узкий резонансный пик, то есть система более избирательно усиливает сигналы определённой частоты.
Низкое значение Q означает широкий резонансный пик, то есть система имеет меньшую избирательность.

fc — Центральная (резонансная) частота

Определяет частоту, на которой система имеет максимальное усиление

α=2πfc — Угловая частота

Представляет центральную частоту в угловых частотах (радиан/год)

β=πfc / Q — Коэффициент затухания

Описывает, как быстро колебания затухают в системе.

Теперь необходимо найти нулевое пространство (ядро матрицы G будет пустым или нет). Для этого воспользуемся аналогичными функциями Matlab в Python (библиотека numpy)

```
kernel = null_space(G)  # Функция для нахождения нулевого пространства
if kernel.size == 0:
    print("Ядро матрицы G пусто (нулевое пространство отсутствует).")
else:
    print("Ядро матрицы G (нулевое пространство):")
    print(kernel)
```

Получим следующий результат:

Ядро матрицы G пусто (нулевое пространство отсутствует).

Аналогично поступим и с другими функциями Matlab, а точнее: определитель и характеристическое уравнение, собственные значения

```
det(G)
np.poly(G)
np.roots(coefficients)
```

Получим следующие результаты:

Det(G): 28.053078704058112

Коэффициенты характеристического полинома матрицы G:
[1.   0.0529645 28.0530787]

Корни характеристического уравнения (собственные значения матрицы G):
[-0.02648225+5.29644951j -0.02648225-5.29644951j]

Матричная экспонента играет ключевую роль в решении линейных систем дифференциальных уравнений первого порядка

Определяет поведение системы во времени: устойчивость, затухание, резонанс и т.д.

Численное значение переходной матрицы expm(tG) при t = 1:
[[0.53700482 0.81242816]
 [-0.81242816 0.53700482]]

Символьное выражение переходной матрицы expm(tG):

$$\begin{bmatrix} 0.537004818476047 & 0.812428156926273 \\ -0.812428156926273 & 0.537004818476047 \end{bmatrix}$$

Найдем матричную передаточную функцию:

$$\mathbf{W}_{\mathbf{x}}(p) = (p\mathbf{I} - \mathbf{G})^{-1}$$

p –это переменная Лапласа

Достаточно вычислить эту обратную матрицу:

$$H\_11(p) = \frac{1.0 \cdot p + 0.0264822475418077}{1.0 \cdot p^{2} + 0.0529644950836154 \cdot p + 28.0530787040581}$$

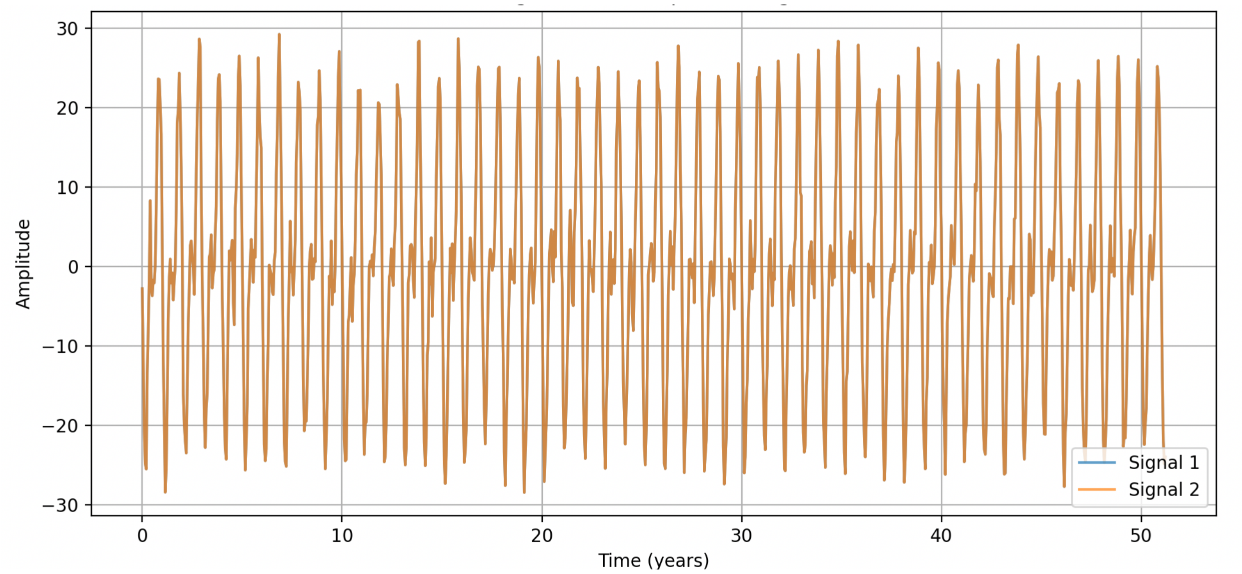
$$H\_12(p) = \frac{-5.29644950836154}{1.0 \cdot p^{2} + 0.0529644950836154 \cdot p + 28.0530787040581}$$

$$H\_21(p) = \frac{5.29644950836154}{1.0 \cdot p^{2} + 0.0529644950836154 \cdot p + 28.0530787040581}$$

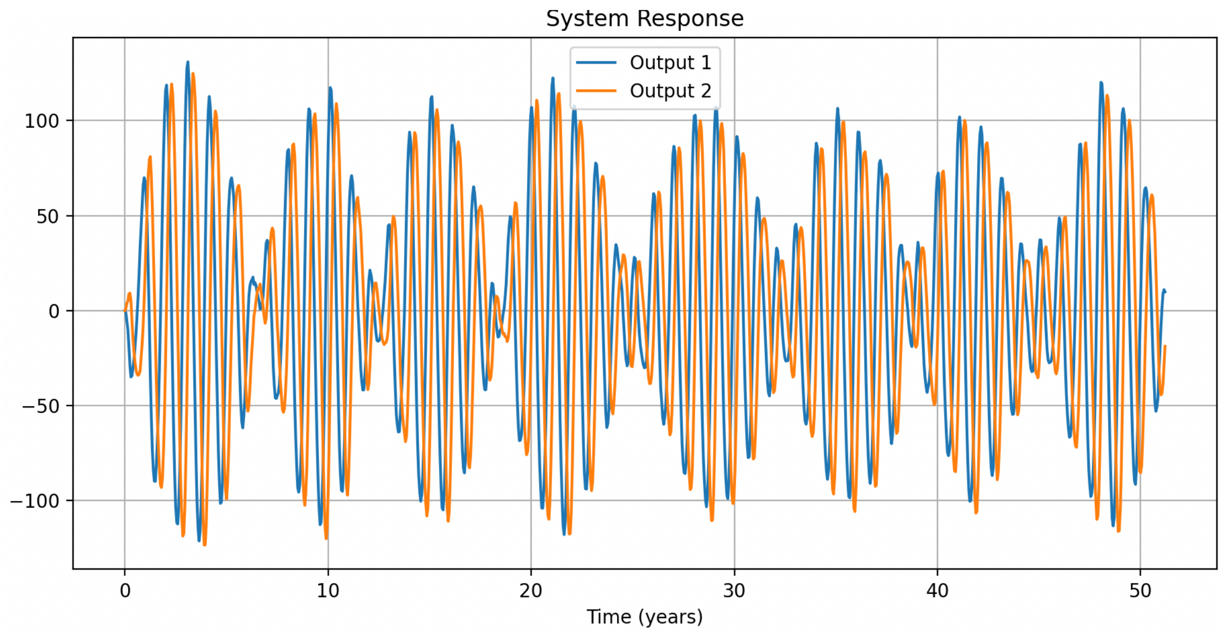
$$H\_22(p) = \frac{1.0 \cdot p + 0.0264822475418077}{1.0 \cdot p^{2} + 0.0529644950836154 \cdot p + 28.0530787040581}$$

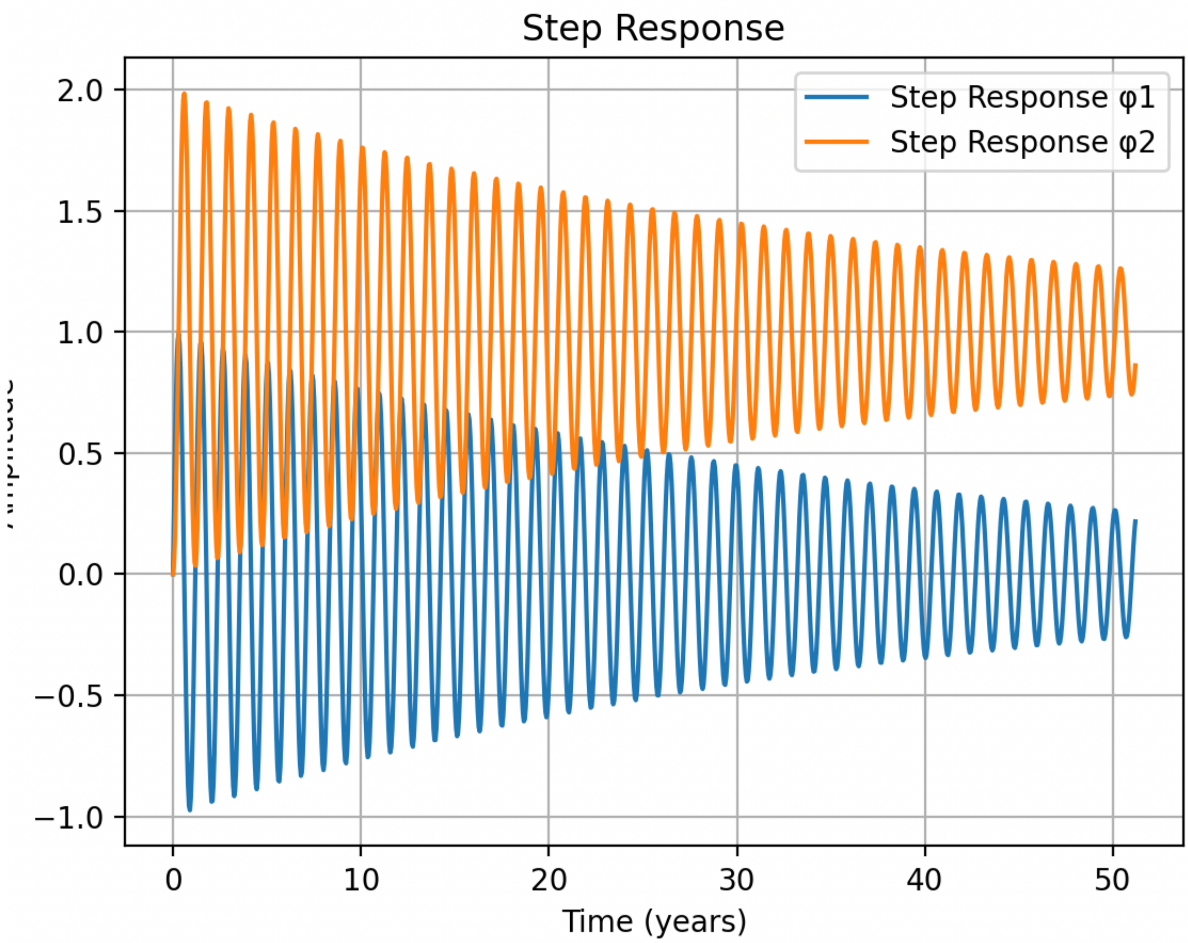
Если посмотреть на коэффициенты и собственные значения, то получили тоже самое.

Собственный сигнал:



Рассмотрим сразу отклики для наших входных сигналов:

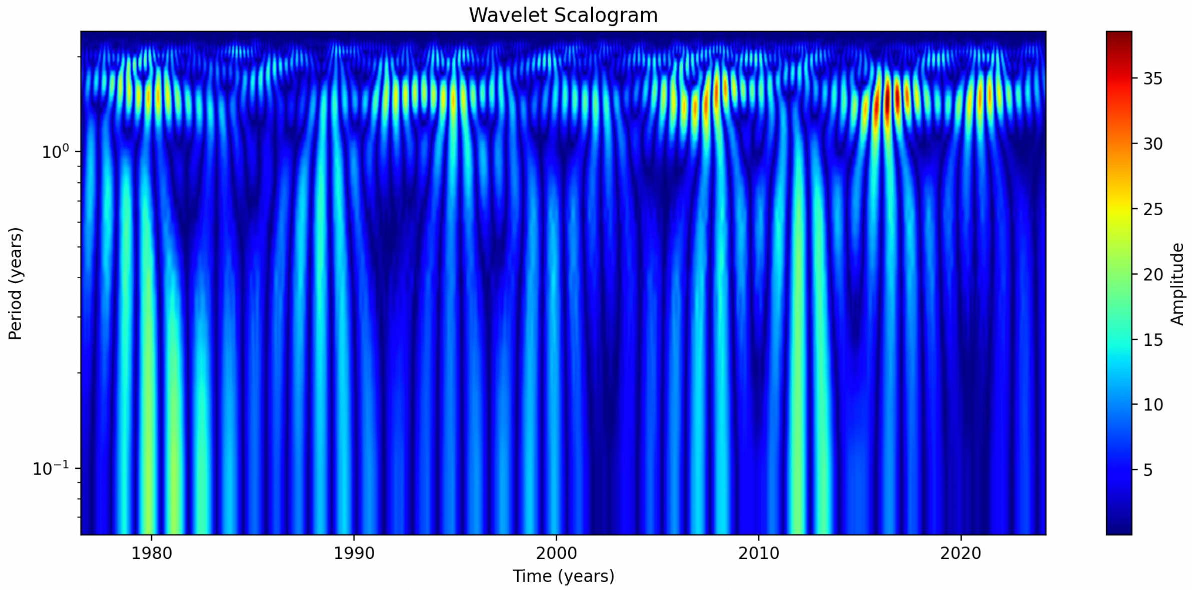




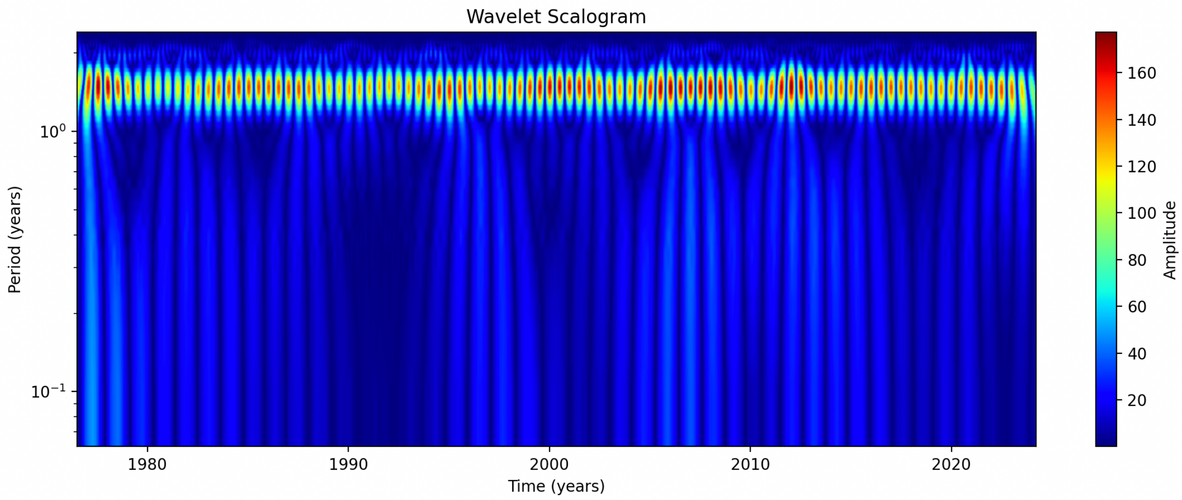
Для части 2 построим сначала скейлограмму:

Изначально построим для входного сигнала X (phi_1)

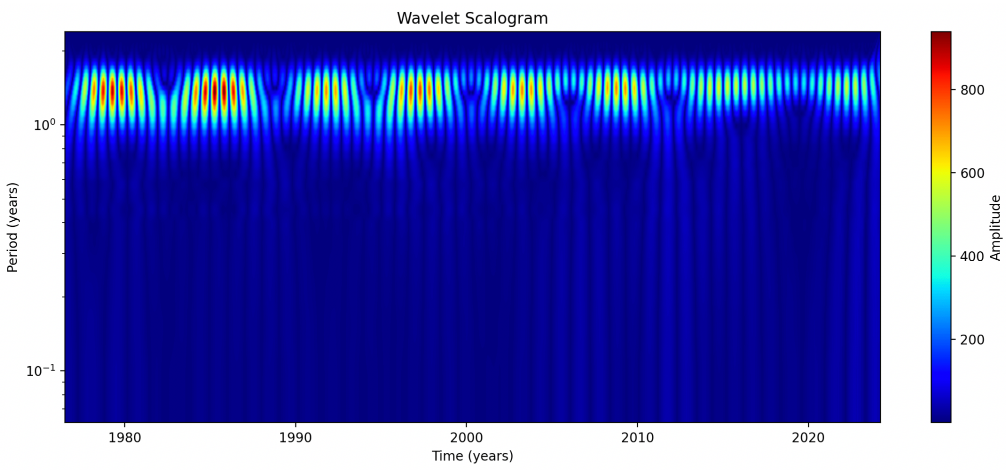
И, как следствие,

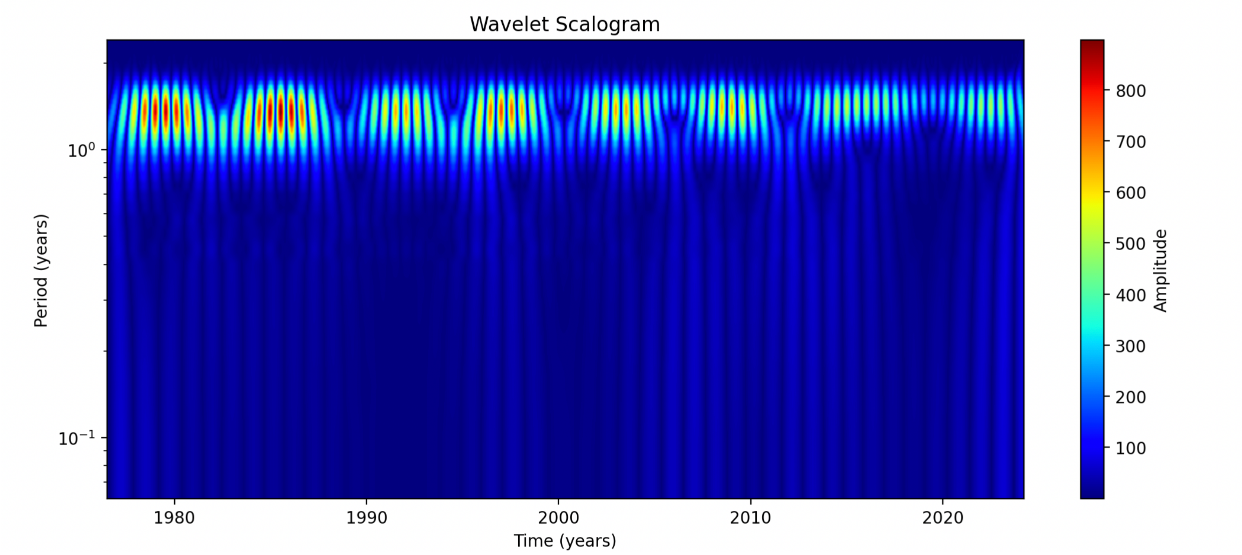


Для второго входного сигнала так же построим скейлограмму и посмотрим на амлпитуды (энергию)



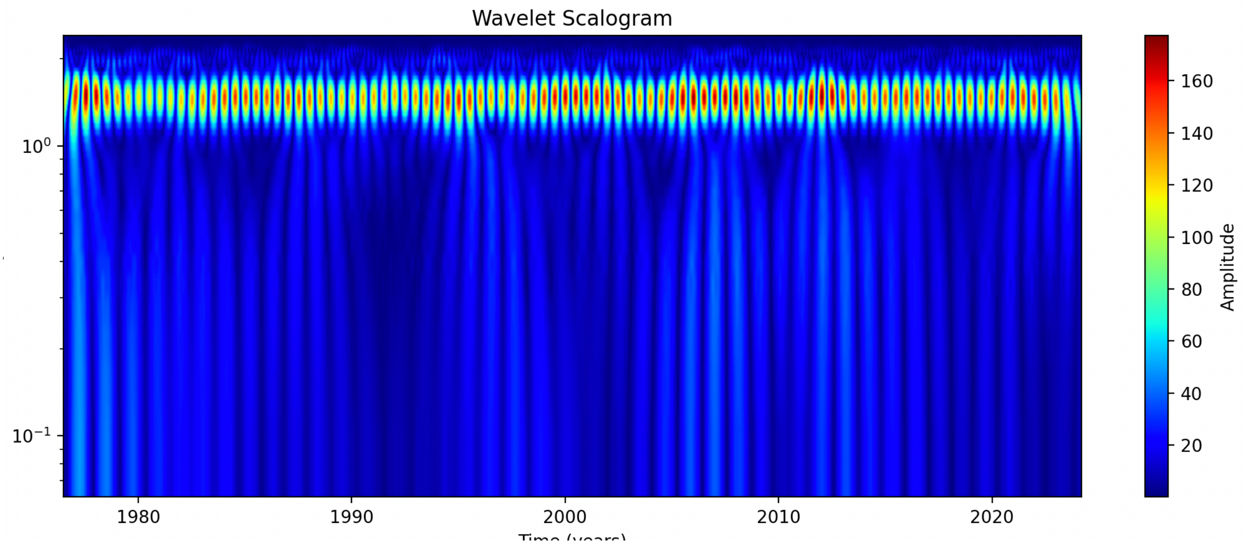
Теперь аналогично построим скейлограмму отклика для m1 и m2(ниже представлены скейлограммы для m1 и m2)



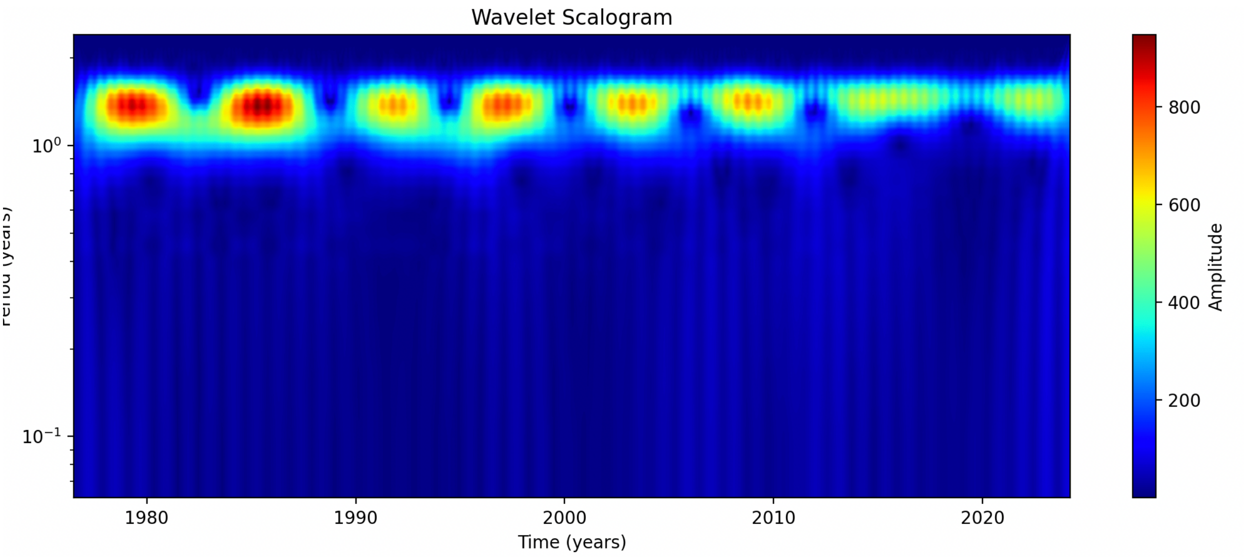


Учитывая тот факт, что графики откликов были получены +- одинаковые, довольно очевидно, что скейлограммы будут +- схожи.

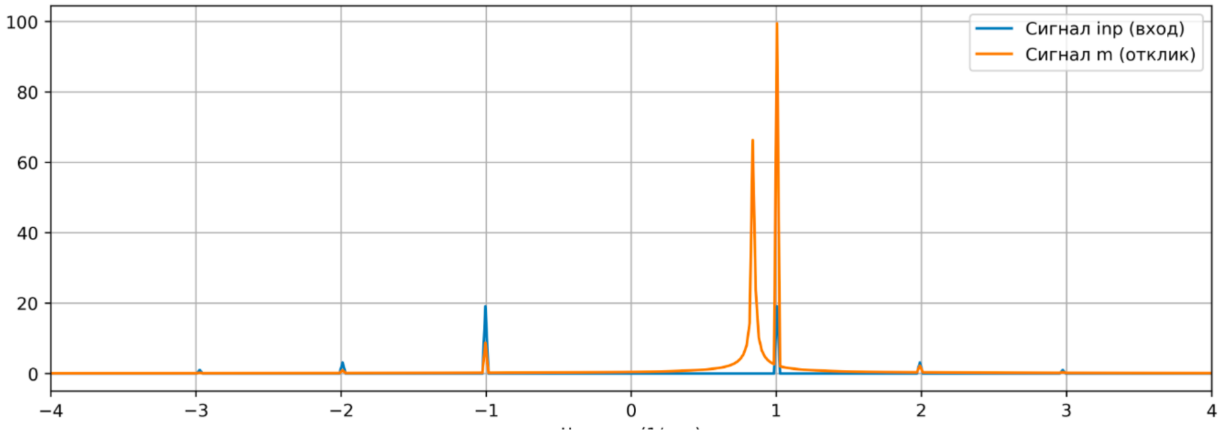
Так же построим комплексный сигнал:



И комплексный отклик:



А так же построим спектр для отклика и сигнала:



Далее необходимо записать импульсную характеристику для уравнений 6.9 и передаточную функцию такого уравнения - прямой оператор

Для системы 6.9 ранее приводилась матричная передаточная функция, а сама функция имеет вид:

$$\mathbf{W_y}(p) = \mathbf{C}\mathbf{W_x}(p)\mathbf{F} = \mathbf{C}(p\mathbf{I} - \mathbf{G})^{-1}\mathbf{F}.$$

Для системы 4.10 получим передаточную функцию:

$$W(p) = \frac{\sigma_c}{ip + \sigma_c}.$$

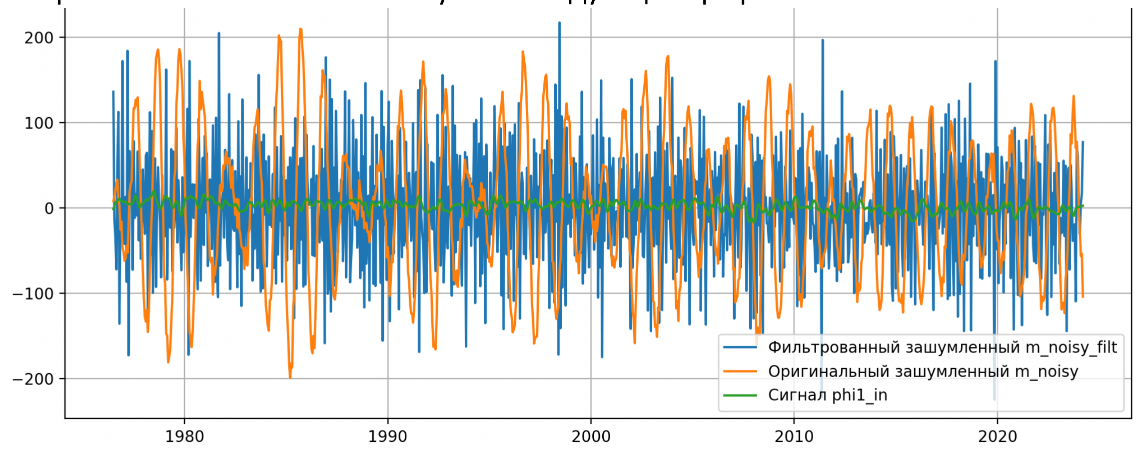
И импульсную характеристику

$$h(t) = -i\sigma_c e^{i\sigma_c t},$$

Необходимо применить к полученному в части 1 сигналу движения полюса (m_1, m_2) обратный оператор в частотной области (функцию. PMInversion)

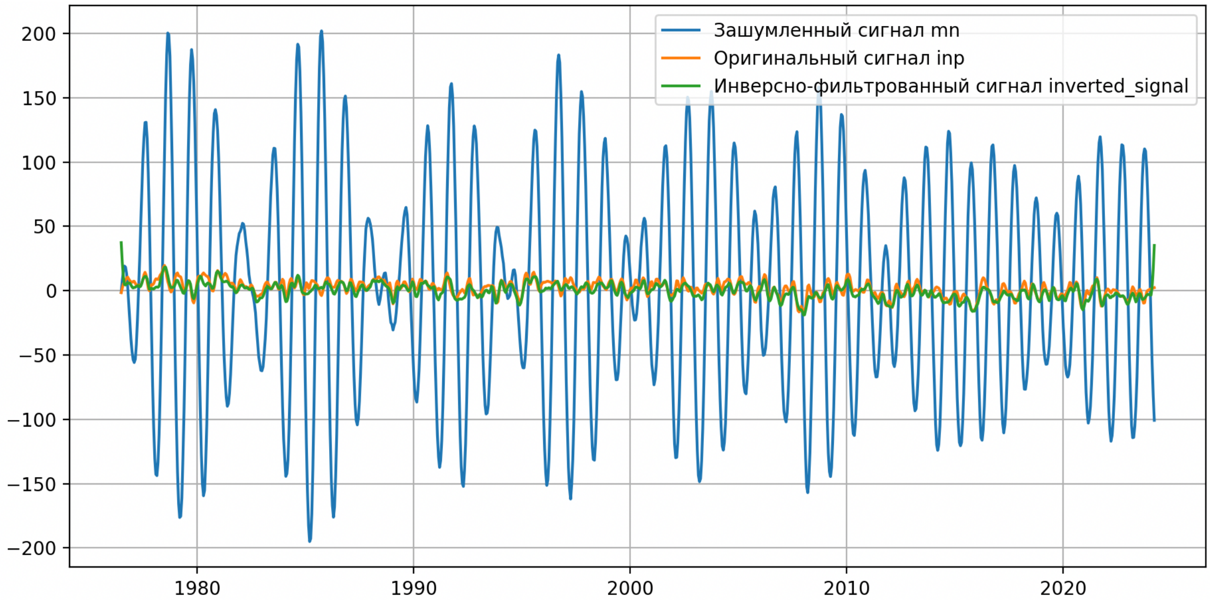
Обратный оператор применяется для восстановления исходного входного сигнала φ(t) из отклика системы m(t)

После применения PMInversion получили следующий график:



Применение обратного оператора. Оранжевым: Re(m + $noise$), синим: Re(PMInversion(m + $noise$)), зеленым: Re($\varphi 1$ + $i\varphi 2$)
Видно, что обратное решение далеко от входного сигнала.

Добавление шума так же не привело к каим то изменениям, воспользуемся фильтром Пантелеева , положим fc = 0 (как в ЛР6) и воспользуемся функцией ChandPantFreqFilter
И f0 = 3
После применения фильтра (функцией), получили следующее



Здесь оригинальный сигнал довольно хорошо сошелся с инверсно-отфлильтрованным