

Częstotliwościowe kryteria stabilności

Adrian Jałoszewski

21 marca 2017, godzina 12:30

1 Wstęp

Celem laboratorium było zapoznanie się z zastosowaniem częstotliwościowych kryteriów stabilności. Przykłady obejmowały kryterium Nyquista w zastosowaniu dla układu, który jest niestabilny bez sprzężenia zwrotnego, przypadku obiektu inercyjnego pierwszego rzędu z opóźnieniem oraz zadania dodatkowego, gdzie należy wyznaczyć dla jakiego wzmocnienia układ znajduje się na granicy stabilności.

2 Wykonanie

2.1 Zadanie 2.1

Transmitancja jest zadana jako:

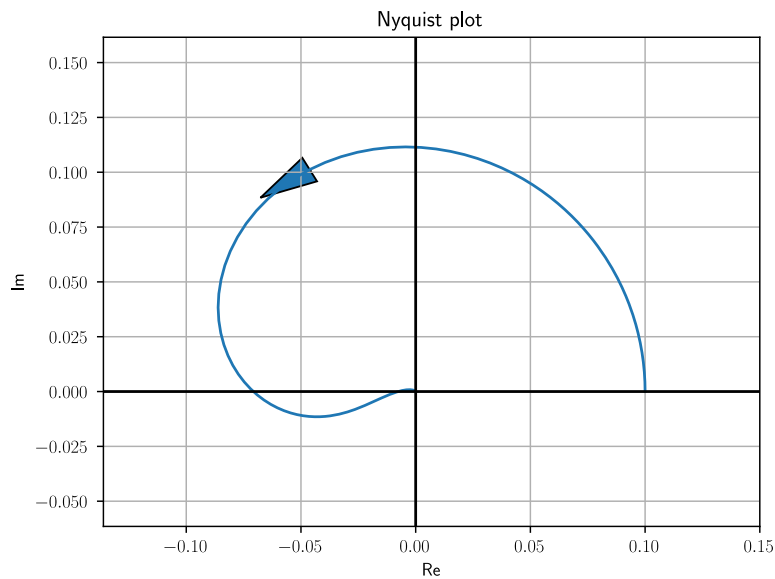
$$G_0(s) = \frac{s + 1}{0,01s^4 + 0,5s^3 + 3s^2 - 10s + 10}$$

Należy wyznaczyć krytyczne wartości współczynnika wzmocnienia K dla którego $K \cdot G_0(s)$ po zapieciu sprzężenia zwrotnego jest stabilne.

Najpierw sprawdzam pierwiastki wielomianu charakterystycznego układu aby sprawdzić ile z nich posiada dodatnią liczb rzeczywistą. Pierwiastki wielomianu:

$$M(s) = 0,01s^4 + 0,5s^3 + 3s^2 - 10s + 10$$

są w przybliżeniu następujące $s_0 \in \{-42,3443, -10,2412, 1,29278 \pm j \cdot 0,79665\}$. Dwa pierwiastki mają więc dodatnią część rzeczywistą, dlatego też przyrost $\Delta_{0 \leq \omega \leq \infty} \arg(1 + G_0(j\omega)) = 2\pi$.

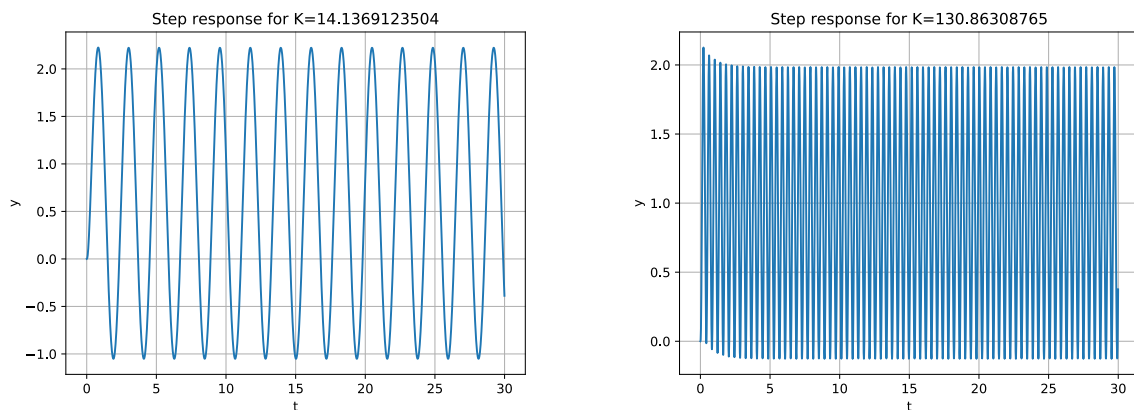


Rysunek 1: Wykres Nyquista dla $G_0(j\omega)$

Wykres ten oś rzeczywistą dla wartości rzeczywistych $-0,07074$ oraz $-0,007642$. Dlatego współczynnik K musi się znajdować w przedziale około:

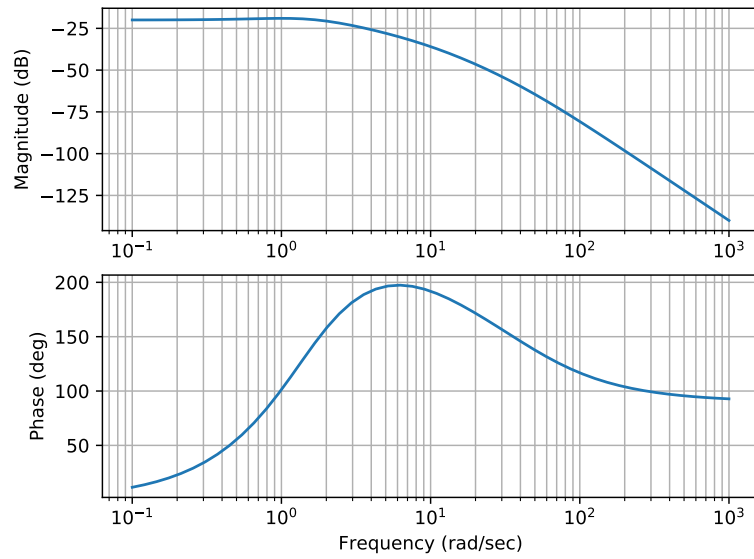
$$K \in [14.14, 130.86]$$

Dowodem tego są oscylacje nieliniowe dla dokładniej wyznaczonych wartości:



Rysunek 2: Odpowiedzi skokowe dla układu ze sprzężeniem zwrotnym

Charakterystyka amplitudowa Bodego ukazuje własności tłumiące układu otwartego. Na charakterystyce amplitudowej można zobaczyć dwa przegięcia, które następują dla dwóch pierwiastków o części rzeczywistej dodatniej. Występują również znaczne przesunięcia fazowe.



Rysunek 3: Wykres Bodego dla $G_0(j\omega)$

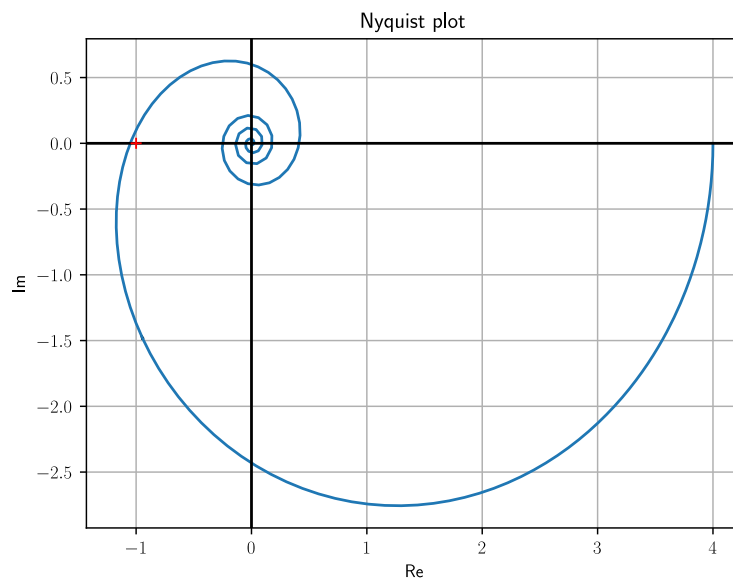
2.2 Zadanie 2.2

Transmitancja dana wzorem:

$$G(s) = \frac{4e^{-0.5s}}{s+1}$$

Podstawiając $s = j\omega$. I zapisując w postaci trygonometrycznej otrzymuję.

$$G(j\omega) = \frac{4}{1+\omega^2} \cdot ((\cos 0.5\omega - \omega \sin 0.5\omega) - j(\sin 0.5\omega + \omega \cos 0.5\omega))$$



Rysunek 4: Wykres Nyquista dla $G(j\omega)$

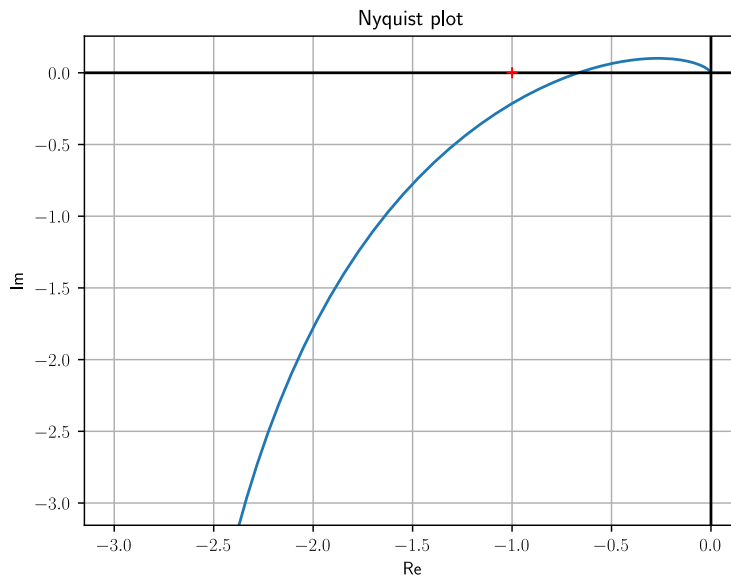
Mimo tego, że układ opóźniany sam z siebie jest stabilny, to dodanie opóźnienia sprawiło, że stał się niestabilny, gdyż przecina oś rzeczywistą na lewo od punktu $-1 + j0$. Można wyznaczyć to również analitycznie wyznaczając ω z $\sin 0.5\omega + \omega \cos 0.5\omega = 0$. Pierwszą dodatnią wartością dla jakiej to jest prawdziwe jest $\omega \approx 3.6732$. Część rzeczywista transmitancji dla tej wartości wynosi: $G(3.6732) \approx -1.0507$, więc punkt przecięcia znajduje się na lewo od punktu $-1 + j0$, co sprawia, że układ jest niestabilny.

2.3 Zadanie dodatkowe

Zadanie dodatkowe polegało na wyznaczeniu wzmocnienia krytycznego dla układu o transmitancji:

$$G_0(s) = \frac{1}{s(T_1s + 1)(T_2s + 1)}$$

Gdzie w szczególnym przypadku $T_1 = 1$, $T_2 = 2$. Więc wszystkie pierwiastki wielomianu charakterystycznego są niedodatnie, więc wykres nie może obejmować punktu $(-1, j0)$



Rysunek 5: Wykres Nyquista dla $G_0(j\omega)$

Punkt przecięcia wykresu z osią rzeczywistą wypada w $-\frac{2}{3}$, dlatego też wzmocnienie krytyczne wynosi $K = 1,5$.

Do wyznaczenia miejsc zerowych ujemnych posłużyła następująca funkcja.

```
def extract_from_poly(num, den):
    numerator = np.poly1d(num)
    num_re, num_im = real_imag_poly(numerator)
    num_re, num_im = np.poly1d(num_re), np.poly1d(num_im)
    denominator = np.poly1d(den)
    den_re, den_im = real_imag_poly(denominator)
    den_re, den_im = np.poly1d(den_re), np.poly1d(den_im)
    multiplier = den_re - den_im * 1j
    new_num = num_re + num_im * 1j
    new_num *= multiplier
    new_num_im = np.poly1d(np.imag(new_num.coefs))
    roots = np.roots(new_num_im)
    roots = [root for root in roots if root.real >= 0]
    print(roots)
    real_parts = np.polyval(new_num, roots) / \
        (np.polyval(den_re, roots) *
         np.polyval(den_re, roots) +
         np.polyval(den_im, roots) *
         np.polyval(den_im, roots))
    return np.array([x for x in real_parts.real if x < 0])
```

Posiłkowała się ona funkcją, która dzieli wielomian na jego część rzeczywistą i urojoną:

```
def real_imag_poly(polynomial: np.poly1d):
    pol = polynomial.coefs
    rev = [x for x in reversed(pol)]
    sign = 1
    real_ans = []
    imag_ans = []
    for i in range(0, len(rev)):
        if i % 2 == 0:
            real_ans.append(rev[i] * sign)
            imag_ans.append(0)
        else:
            real_ans.append(0)
            imag_ans.append(rev[i] * sign)
            sign *= -1
    real_ans = [x for x in reversed(real_ans)]
    imag_ans = [x for x in reversed(imag_ans)]
    return real_ans, imag_ans
```

3 Wnioski

Układy niestabilne da się ustabilizować dokładając sprzężenie zwrotne ujemne. O ile w przypadku układu stabilnego współczynnik wzmocnienia należał do przedziału zaczynającego się w zerze o tyle tu jest już to pewien przedział dobrany tak aby wykres Nyquista otoczył punkt $-1 + j$ odpowiednio wiele razy.

Opóźnienia potrafią doprowadzić do braku stabilności dla układów, które bez nich byłyby stabilne. Ma to znaczne znaczenie przy zdalnym sterowaniu oraz przy innych rodzajach sterowania, które wprowadzają opóźnienia (np. taśmociąg z węglem do normalnie stabilnego pieca).