

ПОСТРОЕНИЕ АСИМПТОТИЧЕСКИХ ЛАЧХ В СРЕДЕ MATLAB

Толочко О.И., Коцегуб П.Х., Федоряк Р.В.

Донецкий государственный технический университет, г.Донецк, Украина

e-mail: toi@elf.dgtu.donetsk.ua

При анализе систем автоматического управления (САУ) часто возникает задача построения логарифмических амплитудно-частотных характеристик (ЛАЧХ) $L(\lg \omega)$ по их передаточным функциям $W(p)$. Амплитуда L в децибелах (дБ) рассчитывается по формуле

$$L(\omega) = 20 \lg |W(j\omega)|.$$

Таким образом, для построения точной ЛАЧХ можно использовать любое программное средство, оперирующее с комплексными числами и имеющее развитый графический интерфейс.

Иначе обстоит дело с построением асимптотических ЛАЧХ (АЛАЧХ), т.к. автоматизация их расчета является более сложной задачей. Для ее решения необходимо выполнить следующую последовательность операций:

- 1) определить нули z и полюсы p системы

$$\mathbf{p} = [p_1, p_2, \dots, p_n], \quad \mathbf{z} = [z_1, z_2, \dots, z_m];$$

и отношение коэффициентов при старших степенях передаточной функции K_l .

- 2) рассчитать количество нулевых полюсов k_{po} и нулей k_{zo} и определить угол наклона начального участка АЛАЧХ

$$k_{n0} = k_{po} - k_{zo};$$

- 3) определить ненулевые частоты (амплитуды) нулей и полюсов и, если таковые существуют, объединить их в один вектор частот сопряжения участков асимптотической ЛАЧХ

$$\omega = [\omega_1, \omega_2, \dots, \omega_{n1}, \omega_{n1+1}, \omega_{n1+2}, \dots, \omega_{n1+m1}] = [\omega_1, \omega_2, \dots, \omega_k],$$

где $n_1 = n - k_{po}$, $m_1 = m - k_{zo}$, $k = n_1 + m_1 = n + m - k_{n0}$;

- 4) создать соответствующий массиву ω вектор изменений коэффициентов наклонов участков АЛАЧХ

$$\Delta \mathbf{k} = [-1, -1, \dots, -1, 1, 1, \dots, 1];$$

- 5) отсортировать массивы ω и $\Delta \mathbf{k}$ по возрастанию частот сопряжения;
- 6) дополнить отсортированный вектор ω слева начальной частотой ω_0 , а справа – конечной частотой ω_{end} , а отсортированный вектор $\Delta \mathbf{k}$ – слева и справа нулями;
- 7) сформировать из массива изменений наклонов ЛАЧХ вектор коэффициентов наклона участков

$$k_{ni} = \sum_{j=1}^i \Delta k_j, \quad i = 1, 2, \dots, k-1;$$

- 8) вычислить начальное значение АЛАЧХ: если, $k_{n0}=0$, то

$$L_0 = 20 \lg K ,$$

иначе –

$$L_0 = -20k_{n0} \lg \frac{\omega_{с.ф.}}{\omega_0},$$

где K – коэффициент усиления системы в установившемся режиме (отношение коэффициентов при свободных членах полиномов числителя и знаменателя передаточной функции),

$$\omega_{с.ф.} = \left[K_l \frac{\prod_{i=1}^{m_l} (-z_{li})}{\prod_{j=1}^{n_l} (-p_{lj})} \right]^{-1/k_{n0}} ;$$

9) вычислить значения АЛЧХ в остальных табличных точках

$$L_i = L_{i-1} + 20k_{ni-1} \lg \frac{\omega_i}{\omega_{i-1}}, \quad i = 1, 2, \dots, k + 1.$$

MATLAB-функция, составленная авторами, на основании этого алгоритма, для системы с одним входом и одним выходом (*SISO*) имеет вид:

```
function [La,wa]=bode_as(model,trace,w0,wk)

if (nargin==1), trace=0; end
if nargin>1 & isempty(trace), trace=0; end
if isstr(model), [a,b,c,d]=linmod(model); model=ss(a,b,c,d); end
                                                    % Преобразование в SS
% Преобразование LTI-объекта в форму ZPK (Zero-Pole-Gain):
if class(model)~='zp', syszp=zpk(model); else syszp=model; end
syszp1=minreal(syzsp,1e-10); % Сокращение одинаковых
                               % нулей и полюсов:
[z1,p1,K1v]=zpkdata(syzsp1,'v'); % Извлечение нулей и полюсов
                               % из объекта
[wp,ksip]=damp(p1); % Частоты и коэф-ты демпфирования
                               % полюсов
kp=length(p1); % Кол-во полюсов
[wz,ksiz]=damp(z1); % Частоты и коэф-ты демпфирования
                               % нулей
kz=length(z1); % Кол-во нулей
ip=find(wp==0); % Номера нулевых полюсов
iz=find(wz==0); % Номера нулевых нулей
kn0=length(iz)-length(ip); % Коэф-т наклона начального участка
                               % ЛАЧХ
                               % Расчет фиктивной частоты среза:
dknz= ones(kz,1); % Изменение наклона ЛАЧХ нулями
dknp=-ones(kp,1); % Изменение наклона ЛАЧХ полюсами
pf=[wz dknz; wp dknp]; % Матрица частот и изменений
                               %коэф-тов наклона ЛАЧХ
if (any(p1) & ~isempty(p1)) | (any(z1) & ~isempty(z1)) % Если есть
                               %ненулевые нули или полюсы
    pf([iz,kz+ip],:)=[]; % Исключение из матрицы нулевых
```

```

                                % полюсов и нулей
[ws,is]=sort(pf(:,1)); % Сортировка частот сопряжения по
                                % возрастанию
pf=pf(is,:); % Сортировка матрицы pf по
                                % возрастанию частот
else wsf=K1^(-1/kn0); pf(:,1)=wsf*ones(kz+kp,1);
if kn0>0, pf(1:kz,2)=wsf*zeros(kz,1);
else pf(kz+1:kz+kp,2)=wsf*zeros(kp,1); end
end
% Поиск начальной и конечной частоты и их логарифмов:
if nargin<3, wa0=pf(1,1); w0=wa0/5;
w0l=log10(w0); w0l=floor(w0l); w0=10^w0l;
else w0l=log10(w0); end
if nargin<4, wak=pf(end,1); wk=wak*5; wkl=log10(wk);
wkl=ceil(wkl); wk=10^wkl;
else wkl=log10(wk); end
% Присоединение к матрице начальной и конечной частот:
mfk=[w0 kn0; pf; wk,0];
% Замена во втором столбце изменений наклонов наклонами:
mfk(:,2)=cumsum(mfk(:,2));
La=zeros(size(mfk,1),1);
k=1;jj=[];
% Расчет начального значения асимпт. ЛАЧХ
if kn0==0, K=dcgain(syszp1); La(1)=20*log10(K);
else
z1(iz)=[]; p1(ip)=[];
if ~exist('wcf'),
prp=(prod(-p1)); prz=(prod(-z1));
wcf=(K1*prz/prp)^(-1/kn0);
end
La(1)=-20*kn0*log10(wcf/w0);
end
% Расчет точек ЛАЧХ
for i=2:length(La)
if mfk(i-1,1)==mfk(i,1); jj(k)=i-1; k=k+1; end
La(i)=La(i-1)+20*mfk(i-1,2)*log10(mfk(i,1)./mfk(i-1,1));
end
% Устранение точек с одинаковыми частотами
mfkp=mfk; mfk(jj,:)=[]; La(jj)=[]; pf=[mfk La];
wa=mfk(:,1);
% Вывод расчетных данных:
if trace~=0
syszp1
disp(blanks(1))
disp(' wa kn La')
disp(' -----')
knd=20*mfk(:,2); pf1=[wa knd La];
fprintf('%12.4g %4.0f %12.4g \n', pf1')
end
% Построение ЛЧХ:
if nargout==0
w=logspace(w0l,wkl,200);
[A,fi]=bode(syszp1,w); % Расчет параметров точных ЛЧХ

```


В качестве примера на рис.1 приведена Simulink-модель разомкнутого контура тока двухмассовой электромеханической системы, а на рис. 2 – ЛЧХ этой модели, полученные при выполнении команды `bode_as('dems2krt')`.

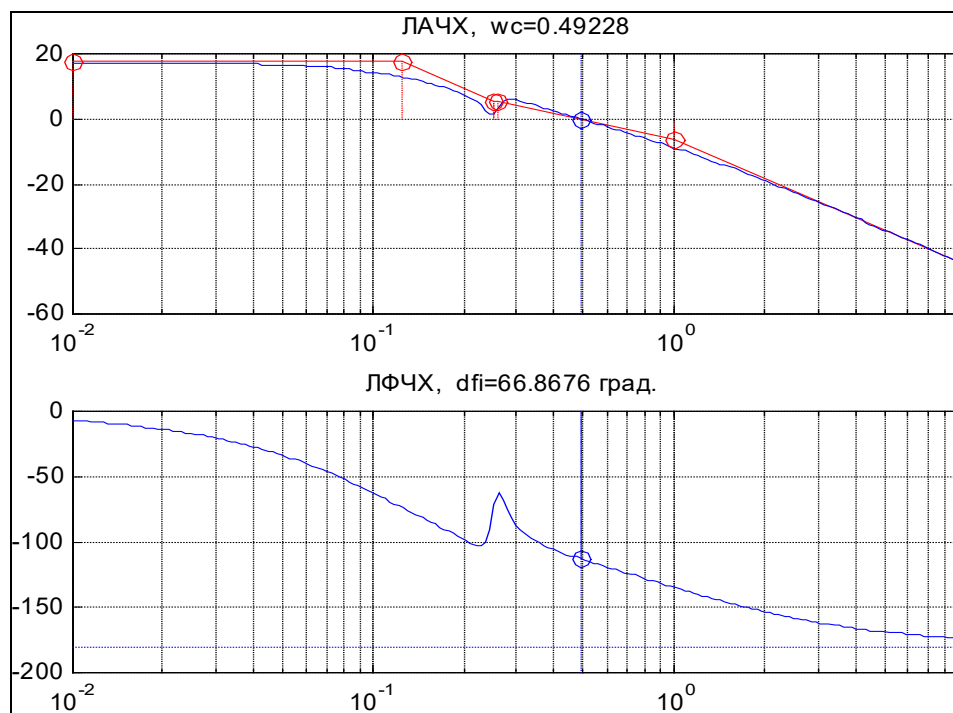


Рис. 2.

Если известна передаточная функция системы, например, $W(p) = \frac{0.08p + 1}{(0.032p)^2(0.02p + 1)}$, то для получения ее ЛЧХ нет необходимости в использовании Simulink. Достаточно только выполнить одну из команд

`bode_as(tf([0.08 1],[0.0032*0.02 0.0032 0 0]));`

или

`bode_as(zpk([1/0.08],[0 0 1/0.02], 0.08/(0.0032*0.02)));`

Правильность функционирования разработанной функции проверена на многочисленных примерах путем сопоставления точной и асимптотической ЛАЧХ.

Изображение точной и асимптотической ЛАЧХ в одной системе координат позволяет качественнее проанализировать исследуемую систему.

Литература

1. Медведев В.С., Потемкин В.Г. Control System Toolbox. М.: ДИАЛОГ-МИФИ, 1999. 287 с.