

Лабораторно упражнение 1

КЛАСИЧЕСКИ АНАЛОГОВ ПИД РЕГУЛАТОР С ГРАФОАНАЛИТИЧНО НАСТРОЙВАНЕ

1. ЦЕЛ НА ИЗСЛЕДВАНИЯТА

Да се приложат различни методики за настройване на класически аналогов ПИД регулатор, които се базират на следните два подхода: първия, за обработване на преходния процес на управлявания обект и създаване на типов параметричен модел, и втория, за отчитане на критичните параметри наведена в режим на автоколебания САУ на управлявания обект. Да се съпоставят резултатите от функционирането на САУ с един и същ регулатор, настроен при различни методики.

2. МЕТОДИЧНИ УКАЗАНИЯ

2.1. Избор на непрекъснат ПИД закон на управление

За целите на изследванията се разглежда един класически ПИД регулатор с паралелна структура, който формира управление u при задаващо въздействие r и регулируема величина y в САУ в съответствие с описанието

$$u(s) = \left[K_p \left(1 + \frac{1}{T_i s} + T_d s \right) \right] e(s), \quad (1)$$

където $e(s) = r(s) - y(s)$ формира грешката в САУ като разлика между заданието и регулируемата величина, а параметрите на регулатора са съответно: коефициент на пропорционалност K_p и две времеконстанти - на интегриране T_i и на диференциране T_d . Тези коефициенти обикновено се определят (изчисляват, настройват) с графоаналитични зависимости, специфични за всеки от конкретно прилаганите методи за настройване на регулатори от вида (1).

2.2. Настройване на аналогов ПИД по зададен преходен процес на обекта

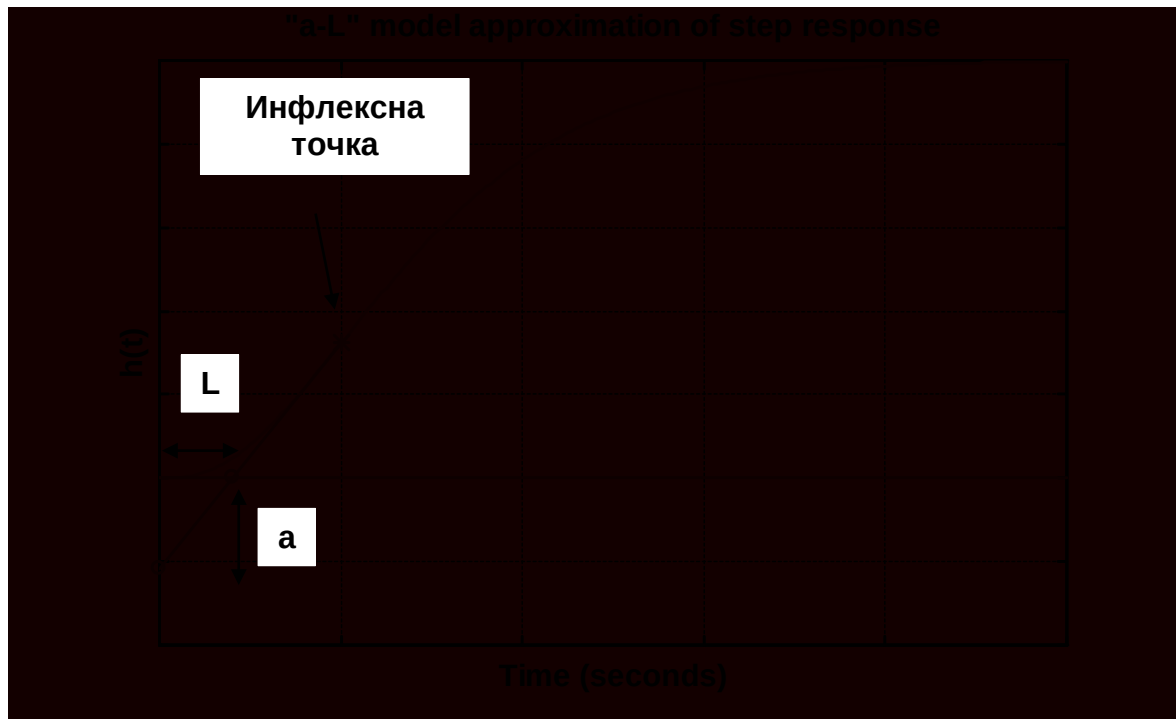
Методиките при този подход изискват: първо, да бъде известен преходният процес на обекта на управление, второ, да се извърши апроксимирането му чрез параметричен модел от определен тип, и трето, да се приложат изведени зависимости между коефициентите на регулатора и параметрите на така апроксимирания модел на обекта. По-долу са представени няколко класически подхода към така формулираната задача.

(а). Първи метод на Циглер-Никълс (Ziegler-Nichols)

Авторите предлагат преходният процес на обекта да се апроксимира с **двупараметричен модел** от вида

$$W(s) = \frac{a}{Ls} e^{-Ls}, \quad (2)$$

което да се осъществи чрез графоаналитичния подход, показан на фиг. 1.



Фигура 1. “a-L” апроксимация на преходната функция на обекта

Самото настройване на ПИД регулаторите се извършват в съответствие с формулите от табл. 1:

Таблица 1. Настройване по метод на Ziegler-Nichols 1

Регулатор	K_p	T_i	T_d
ПИД	$1.2/a$	$2L$	$L/2$

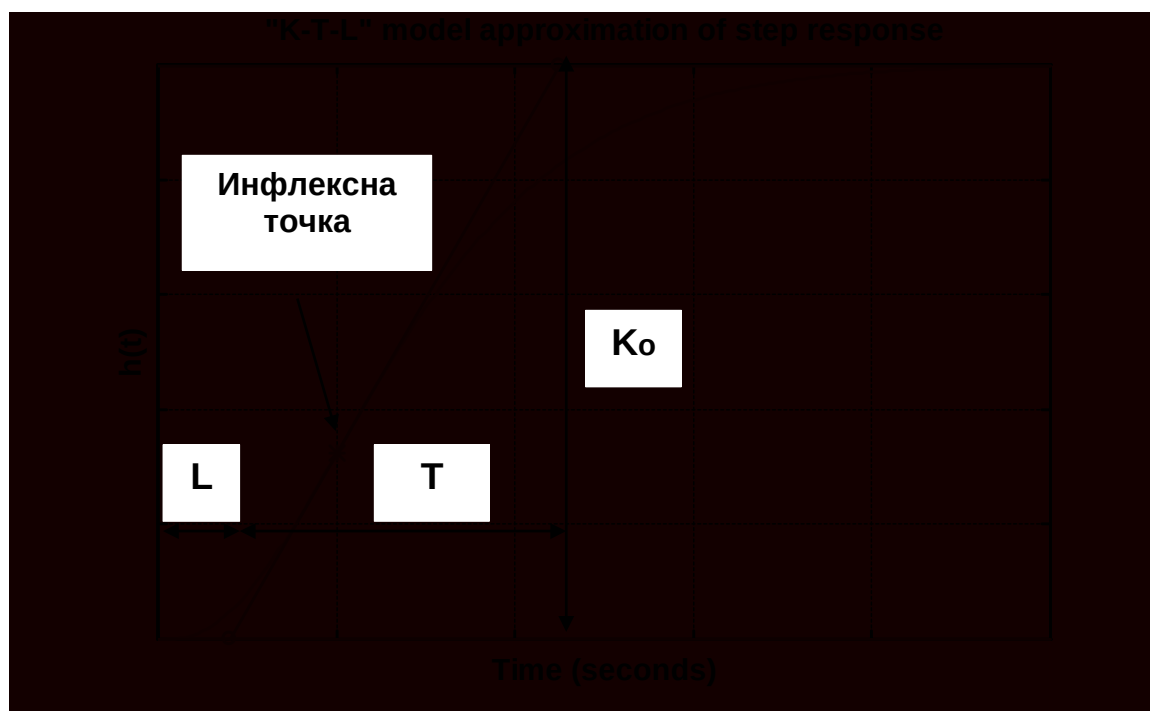
(б). Метод на Коен-Кун (Cohen-Coon)

Авторите предлагат преходният процес на обекта да се апроксимира с *три-параметричен модел* от вида

$$W(s) = \frac{K_o}{1+Ts} e^{-Ls} \quad (3)$$

по някой от различните варианти за това. Например:

Вариант 1. Метод на допирателната според графоаналитичния подход, показан на фиг. 2.



Фигура 2. “К-Т-Л” апроксимация на преходната функция на обекта

Вариант 2. **Метод на Стрейц** за апроксимиране с параболоа от втори ред на преходната характеристика на обекта в две точки - инфлексната точка (t_A, \bar{h}_A) и ($t_B, \bar{h}_B \geq 0.85$) и определяне на (3) според зависимостите:

$$L = \frac{t_A \ln(1 - \bar{h}_B) - t_B \ln(1 - \bar{h}_A)}{\ln(1 - \bar{h}_B) - \ln(1 - \bar{h}_A)}, \quad T = \frac{L - t_A}{\ln(1 - \bar{h}_A)} = \frac{L - t_B}{\ln(1 - \bar{h}_B)}. \quad (4)$$

Самото настройване на ПИД регулатора се извършват в съответствие с формулите от табл. 2 след предварително изчисляване на допълнителните променливи $\alpha = K_0 L / T$ и $\tau = L / (L + T)$:

Таблица 2. Настройване по метод на **Cohen-Coon**

Регулатор	K_p	T_i	T_d
ПИД	$\frac{1.35}{\alpha} \left(1 + \frac{0.18\tau}{1 - \tau} \right)$	$\frac{2.5 - 2.0\tau}{1 - 0.39\tau} L$	$\frac{0.37 - 0.37\tau}{1 - 0.81\tau} L$

(в). **Метод на Чиен, Хронес и Резуик (Chien, Hrones u Reswick)**

Авторите предлагат преходният процес на обекта да се апроксимира с **дву-параметричен модел** (2), но с определяне и на **доминиращата времеконстанта T** от трипараметричния модел (3). Изведените зависимости за настройване на ПИД регулатори зависят от определено допълнително изискване:

Допълнително изискване - вариант 1. Настройваният ПИД трябва основно да отработва **товарното смущение** в САУ. В този случай се използват формулите от табл.3.

Таблица 3. Настройване по метод на Chien, Hrones и Reswick по тов. смущение

Пререгулиране	0%			20%		
Регулатор	K_p	T_i	T_d	K_p	T_i	T_d
ПИД	$0.95/a$	$2.4L$	$0.42L$	$1.2/a$	$2L$	$0.42L$

Допълнително изискване - вариант 2. Настройваният ПИД трябва основно да отработва **заданието** в САУ. В този случай се използват формулите от табл.4.

Таблица 4. Настройване по метод на Chien, Hrones и Reswick по задание

Пререгулиране	0%			20%		
Регулатор	K_p	T_i	T_d	K_p	T_i	T_d
ПИД	$0.6/a$	T	$0.5L$	$0.95/a$	$1.4T$	$0.47L$

2.3. Настройване на аналогов ПИД по критичните параметри в САУ

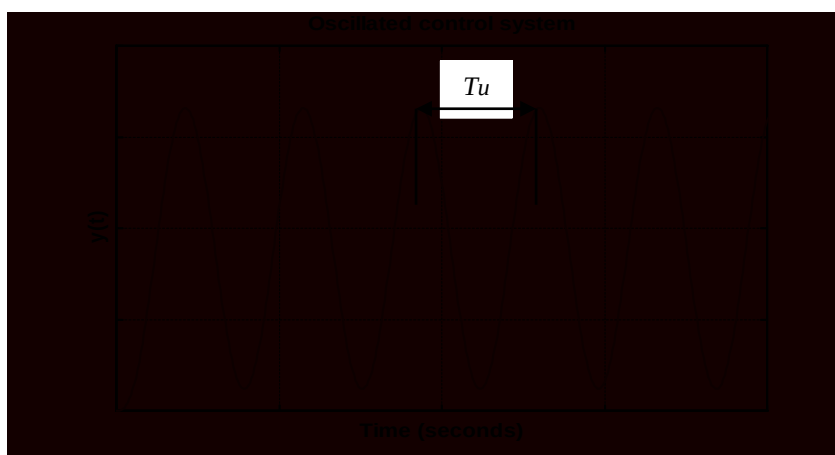
Методиките при този подход изискват: първо, САУ на обекта да бъде приведена в режим на автоколебания, второ, да бъдат определени критичните (граничните, т.е. на границата на устойчивостта) параметри на автоколебанията – коефициент на усилване и период на автоколебания, и трето, да бъдат аналитично изчислени настройките на регулатора.

Според **втория метод на Циглер-Никълс (Ziegler-Nichols)** се изгражда САУ на обекта с П регулатор, чрез който тя се довежда до границата на устойчивостта. Конкретната стойност на критичния коефициент на усилване на П регулатора е K_u , а периодът T_u се определя графично от възникналите автоколебания (фиг. 3).

Настройките се извършват в съответствие с формулите от табл. 5:

Таблица 5. Настройване по метод на Ziegler-Nichols 2

Регулатор	K_p	T_i	T_d
ПИД	$0.6 K_u$	$0.5 T_u$	$0.125 T_u$



Фигура 4. Режим на автоколебания

2.4. Оценяване на степента на приложимост на методите за настройване

За да има предварителна оценка за степента на приложимост на използваните методи за настройване на ПИД регулатор за управление на конкретен динамичен обект със закъснение L и доминираща времеконстанта T , е полезно да се изчисли показател на нормализирано чисто закъснение от вида

$$\tau = \frac{L}{L+T}, \quad 0 \leq \tau \leq 1. \quad (5)$$

Чрез него се отчита следната зависимост: ***С увеличаване на стойността на нормализираното чисто закъснение на обекта нарастват трудностите при управлението му с настроен по конкретен метод ПИД регулатор.***

2.5. Начални данни за провеждане на изследванията

Заимстват се условията за провеждане на Лабораторно упражнение (ЛУ) 1, но вместо определяне на настройваемите параметри чрез проби и грешки, се използват конкретни методики за това.

2.6. Особености на изпълнението на задачите

Използват се направените препоръки в ЛУ 1. Допълнително се изисква:

- (а). Да бъде анализирана степента на приложимост на всяка методика за функциониране на конкретна САУ.
- (б). Да бъдат съпоставени САУ с настроени ПИД регулатори от ЛУ 1 и ЛУ 2.
- (в). Да се осъществи финно донастройване на регулаторите.

3. ЗАДАЧИ ЗА ИЗПЪЛНЕНИЕ

3.1. Да се оценят коефициентите на модели (2) и (3) по експериментално снета преходна характеристика на обекта на управление

Стъпка 1. В SIMULINK се изгражда схема за снемане и визуализиране на преходна характеристика на обекта за експериментирание, която се приема за еталонна при по-нататъшните ѝ обработвания.

Стъпка 2. Избират се последователно дву- и три-параметрични модели и се оценяват техните коефициенти според съгласуваните с ръководителя на упражнението методики. Графо-аналитичното изследване на еталонната преходна функция, чиито стойности са включени в дефинирана променлива, се извършва пряко от екрана на монитора чрез функцията **ginput** или с помощта на потребителски програми в MATLAB, примерен вид на които са дадени в т.4.

Стъпка 3. Анализират се резултатите от идентификацията, като се сравняват преходните процеси на моделите с еталонната характеристика на обекта и се изчисли квадратичен показател от формираната грешка.

3.2. Да се определят критичните показатели на автоколебателен режим

Стъпка 1. В SIMULINK се изгражда САУ с П регулатор за обекта на експериментиране. Уточнява се интервалът на изследване.

Стъпка 2. Чрез промяна (ръчно или автоматично) на коефициента на пропорционалност на регулатора се привежда САУ в автоколебателен режим и се отчитат съответните критични показатели.

3.3. Да се изчислят настройките на непрекъснат ПИД регулатор

Стъпка 1. Използват се функции, дадени в т. 4 за изчисляване на параметрите на ПИД регулатор в зависимост от коефициентите на оценения модел или критичните показатели.

Стъпка 2. Определят се настройките на съответните ПИД регулатори.

3.4. Да се извърши съпоставка на САУ с различно настроени регулатори

Стъпка 1. Чрез позната от предишното упражнение SIMULINK блок-диаграма се организира функциониране на САУ с така настроените регулатори. Допълнително в схемата се включва товарно смущение.

Стъпка 2. Наблюдават се резултати от изследването на САУ, в които еднакви ПИД регулаторите имат различни настройки. Поведението на сигналите в САУ се наблюдават във вида, показан в Указанията за провеждане на лабораторно упражнение No 1.

Стъпка 3. SIMULINK схемите на САУ трябва да включват и блокове **To Workspace**, за да се запази в потребителски променливи необходимата при изчисляване на дадените показатели (6), (7), (8) и (9) информация за конкретни сигнали от функционирането на САУ при стъпално задание и моделирано товарно смущение.

4. ПРИМЕРНИ РЕЗУЛТАТИ

От създадената в MATLAB база от данни за снети характерни стойности от преходните процеси на обектите на управление се извършва тяхната идентификация чрез дву- и три-параметрични модели. Според избраната методика се настройват аналоговите ПИД регулатори. По-долу са дадени примерни функции за идентификация и настройка, които могат да служат за сравнение с разработените от студента програмни модули или функции. Подходящ модули за идентификация на обектите чрез дву-параметричен модел е

показан в табл. 6 за основна функция *h2aL* и в табл. 7 за спомагателна функция *infl_ind*.

Таблица 6. Функция *h2aL*

<pre> function [a,L]=h2aL(he) % % Апроксимиране на преходна характе- % ристика с дву-параметричен модел % % Определяне на инфлексната точка [n,m]=size(he); for i=1:n-2 ind=infl_ind(he(i,:),he(i+1,:),he(i+2,:)); if (ind==2) (ind==3) ipn=i; break; end end % Изчислява коефициентите на % допирателните y=coefi(1)*x+coefi(2) x1=[he(ipn-1,1) he(ipn,1) he(ipn+1,1)]; y1=[he(ipn-1,2) he(ipn,2) he(ipn+1,2)]; coef1=polyfit(x1,y1,1); x2=[he(ipn,1) he(ipn+1,1) he(ipn+2,1)]; y2=[he(ipn,2) he(ipn+1,2) he(ipn+2,2)]; coef2=polyfit(x2,y2,1); % Осредняване на двете допирателни </pre>	<pre> coef=(coef1+coef2)/2; % Координати на пресечната точка с % абцисата coor1(1)=-coef(2)/coef(1); coor1(2)=0; % Изчертава преходната характеристика plot(he(:,1),he(:,2),'b-'), hold on, grid % Отбелязва координатите на % инфлексната точки x_infl=(he(ipn,1)+he(ipn+1,1))/2; y_infl=(he(ipn,2)+he(ipn+1,2))/2; plot(x_infl,y_infl,'*'); % Изчертава допирателната line([0 x_infl],[coef(2) y_infl]); % Изчертава абцисната ос line([0 he(n,1)],[0 0]); % Отбелязва координатите на пресеч- % ните точки на допирателната plot(coor1(1),coor1(2),'o'); % (с абцисата) plot(0,coef(2),'o'); % (с ординатата) hold off a=abs(coef(2)); L=coor1(1); </pre>
--	--

Таблица 7. Функция *infl_ind*

<pre> function ind=infl_ind(xy1,xy2,xy3) % Извежда индекс за разположението % на три последователни точки % % ind=0-->когато точките имат нулеви % координати; % ind=1-->когато втората точка е под % правата между първа и трета; % ind=2-->когато втората точка е върху % правата между първа и трета; % ind=3-->когато втората точка е над % правата между първа и трета; </pre>	<pre> tgn1=(xy1(2)-xy2(2))/(xy1(1)-xy2(1)); tgn2=(xy1(2)-xy3(2))/(xy1(1)-xy3(1)); if tgn1==0 tgn2==0 ind=0; elseif tgn1<tgn2 ind=1; elseif tgn1==tgn2 & (tgn1~=0 tgn2~=0) ind=2; elseif tgn1>tgn2 ind=3; end </pre>
--	---

В табл. 8 и 9 са дадени примерни функции за преобразуване на преходни процеси в три-параметрични модели: *h2KTL* за оценяване на параметри на трипараметричен модел чрез метод на допирателната в инфлексната точка и *h2Strejc* за оценяване на параметрите на трипараметричен модел чрез метод на

Стрейц. Втората функция показва по-добро качество на извежданата предавателна функция на обекти със самоустановяване.

Таблица 8. Функция **h2KTL**

<pre> function [Ко,Т,L,Tmax]=h2KTL(he) % Предназначение % Апроксимиране на преходна % характеристика в три-параметричен % модел $W(p)=Ko/(1+Tp)*e^{(-Lp)}$ % чрез метод на допирателната % Входни аргументи: % he – преходна функция (вектор). % Изходни аргументи: % Ко – коефициент на усилване (скалар); % Т – времеконстанта (скалар); % L – чисто закъснение (скалар). % Tmax – максимална времеконстанта % (скалар); % Спомагателни авторски функции : % infl_ind % Определяне на явното чисто закъснение [n,m]=size(he); for i=1:n-1 if (he(i+1,2)>he(i,2)) Lapparent=i; break end end % Определяне на инфлексната точка for i=Lapparent:n-2 ind=infl_ind(he(i,:),he(i+1,:),he(i+2,:)); if (ind==2) (ind==3) ipn=i; break; end end % Изчислява коефициентите на допирателните $y=coefi(1)*x+coefi(2)$ x1=[he(ipn-1,1) he(ipn,1) he(ipn+1,1)]; y1=[he(ipn-1,2) he(ipn,2) he(ipn+1,2)]; coef1=polyfit(x1,y1,1); x2=[he(ipn,1) he(ipn+1,1) he(ipn+2,1)];y2=[he(ipn,2) he(ipn+1,2) he(ipn+2,2)]; coef2=polyfit(x2,y2,1); </pre>	<pre> % Осредняване на двете допирателни coef=(coef1+coef2)/2; % Координати на пресечната точка с абцисата coor1(1)=-coef(2)/coef(1); coor1(2)=0; % Координати на пресечната точка с установената стойност coor2(1)=(he(n,2)-coef(2))/coef(1); coor2(2)=he(n,2); % Изчертава преходната характеристика plot(he(:,1),he(:,2),'b-'), hold on, grid % Отбелязва координатите на инфлексната точка x_infl=(he(ipn,1)+he(ipn+1,1))/2; y_infl=(he(ipn,2)+he(ipn+1,2))/2; plot(x_infl,y_infl,'*'); % Изчертава допирателната line([coor1(1) coor2(1)],[coor1(2) coor2(2)]); % Изчертава установената стойност line([0 he(n,1)],[he(n,2) he(n,2)]); % Изчертава абцисната ос line([0 he(n,1)],[0 0]); % Отбелязва координатите на пресечните точки на допирателната plot(coor2(1),coor2(2),'o'); % (с установената стойност) plot(coor1(1),coor1(2),'o'); % (с абцисата) hold off % Определяне на параметрите на модела % Коефициент на усилване Ко=coor2(2); % Времеконстанта Т=table1([he(Lapparent+1:n,2) he(Lapparent+1:n,1)],0.63*Ко)-coor1(1); % Чисто закъснение L=coor1(1); % Максимална времеконстанта Тmax=coor2(1)-coor1(1); </pre>
--	--

Таблица 9. Функция *h2Strejc*

<pre> function [Ko,T,L]=h2Strejc(he,p) % Предназначение % Апроксимиране на преходна % характеристика в три-параметричен % модел $W(p)=Ko/(1+Tp)*e^{(-Lp)}$ чрез % графоаналитичен метод на Strejc % Входни аргументи: % he – преходна функция (вектор); % p – стойност в интервала (0.85 – 0.9) % (скалар). % Изходни аргументи: % Ko – коефициент на усилване (скалар); % T – времеконстанта (скалар); % L – чисто закъснение (скалар). % Спомагателни авторски функции : % infl_ind % Определяне на явното чисто закъснение [n,m]=size(he); for i=1:n-1 if (he(i+1,2)>he(i,2)) Lapparent=i; break end end % Определяне на инфлексната точка for i=Lapparent:n-2 ind=infl_ind(he(i,:),he(i+1,:),he(i+2,:)); </pre>	<pre> if (ind==2) (ind==3) ipn=i; break; end end % Изчертава преходната характеристика plot(he(:,1),he(:,2),'b-'), hold on, grid % Отбелязва координатите на инфлексната % точка A ta=(he(ipn,1)+he(ipn+1,1))/2; ha=(he(ipn,2)+he(ipn+1,2))/2; plot(ta,ha,'*'); % Определяне на параметрите на модела % Коефициент на усилване Ko=he(n,2); % Отбелязва координатите на точка B hb=p*Ko; tb=table1([he(Lapparent+1:n,2) he(Lapparent+1:n,1)],hb); plot(tb,hb,'*'); hold off % Междинни величини lb=log(1-hb); la=log(1-ha); % Чисто закъснение L=(ta*lb-tb*la)/(lb-la); % Времеконстанта T=(L-tb)/lb; </pre>
--	--

По-долу са показани примерни функции за настройване на класически ПИД регулатори: *PIDtun_ZN1* чрез първия метод на Циглер-Никълс (табл. 10), *PIDtun_CC* по метода на Коен-Кун (табл. 11), *PIDtun_CHRlo* (табл. 12) и *PIDtun_CHRre* (табл. 13) по метода на Chien- Hrones-Reswick за компенсиране, съответно, на товарното смущение и на заданието.

Таблица 10. Функция *PIDtun_ZN1*

<pre> function [K,Ti,Td]=PIDtun_ZN1(a,L,str) % Предназначение % Настройване на П, ПИ и ПИД регулатори % по метода на Ziegler-Nichols % чрез дву-параметричен модел на обекта % $W(p)=a/L*e^{(-Lp)}$ % Входни аргументи: % a – коефициент (скалар); % L – чисто закъснение (скалар); % str – символна променлива за типа на % регулатора (скалар), % str = 'П' – настройване на П регулатор; % str = 'ПИ' – настройване на ПИ регул.; % str = 'ПИД' – настройване на ПИД регул.; </pre>	<pre> % Изходни аргументи: % K – коеф. на усилване на регул.(скалар); % Ti – интегр. времеконстанта(скалар); % Td – дифер. времеконстанта (скалар). if nargin<3 error('Don"t miss the third input argument!'), return, end if strcmp(str,'P') K=1/a; elseif strcmp(str,'PI') K=0.9/a; Ti=3*L; elseif strcmp(str,'PID') K=1.2/a; Ti=2*L; Td=L/2; else error('Wrong name of the controller!') end </pre>
--	---

Таблица 11. Функция PIDtun_CC

<pre> function [K,Ti,Td]=PIDtun_CC(Ko,T,L,str) % Предназначение % Настройване на П, ПИ и ПИД регулатори % по метода на Cohen-Coon % чрез три-параметричен модел % $W(p)=Ko/(1+Tp)*e^{(-Lp)}$. % Входни аргументи: % Ko – коефициент на усилване (скалар); % T – времеконстанта (скалар); % L – чисто закъснение (скалар). % str – символна променлива за типа на % регулатора (скалар), % str = 'П' – настройване на П регулатор; % str = 'ПИ' – настройване на ПИ регул.; % str = 'ПИД' – настройване на ПИД регул.; % Изходни аргументи: % K – коеф. на усилване на регул.(скалар); % Ti – интегр. времеконстанта (скалар); </pre>	<pre> % Td – дифер. времеконстанта (скалар). if nargin<4 error('Don't miss the 4th input argument!'), return, end alfa=Ko*L/T; tau=L/(L+T); if strcmp(str,'P') K=(1+0.35*tau/(1-tau))/alfa; elseif strcmp(str,'PI') K=0.9*(1+0.92*tau/(1-tau))/alfa; Ti=L*(3.3-3*tau)/(1+1.2*tau); elseif strcmp(str,'PID') K=1.35*(1+0.18*tau/(1-tau))/alfa; Ti=L*(2.5-2*tau)/(1-0.39*tau); Td=L*(0.37-0.37*tau)/(1-0.81*tau); else error('Wrong name of the controller!') end </pre>
---	--

Таблица 12. Функция PIDtun_CHRlo

<pre> function [K,Ti,Td]=PIDtun_CHRlo(a,L,str,sig) % Предназначение % Настройване на П, ПИ и ПИД регулатори % по метода на Chien- Hrones-Reswick % за компенсирание на товарното смущение % чрез дву-параметричен модел на обекта % $W(p)=a/L*e^{(-Lp)}$. % Входни аргументи: % a – коефициент (скалар); % L – чисто закъснение (скалар); % str – символна променлива за типа на % регулатора (скалар), % str = 'П' – настройване на П регулатор; % str = 'ПИ' – настройване на ПИ регулатор; % str = 'ПИД' – настройване на ПИД регул.; % sig – желано пререгулиране в САУ (скалар), % sig = 0 – няма пререгулиране; % sig = ~ 0 – 20 процента пререгулиране. % Изходни аргументи: % K – коеф. на усилване на регул.(скалар); % Ti – интегр. времеконстанта(скалар); % Td – дифер. времеконстанта (скалар). if nargin<3 error('Don't miss the third input argument!'), return, end </pre>	<pre> if nargin<4 disp('You choose 0% overshoot!'), sigma=0, end if strcmp(str,'P') if(sigma==0) K=0.3/a; else K=0.7/a; end elseif strcmp(str,'PI') if(sigma==0) K=0.6/a; Ti=4*L; else K=0.7/a; Ti=2.3*L; end elseif strcmp(str,'PID') if(sigma==0) K=0.95/a; Ti=2.4*L; Td=0.42*L; else K=1.2/a; Ti=2*L; Td=0.42*L; end else error('Wrong name of the controller!') end </pre>
--	--

Таблица 13. Функция PIDtun_CHRre

<pre> function [K,Ti,Td]=PIDtun_CHRre (a,L,T,str,sig) % Предназначение % Настройване на П, ПИ и ПИД регулатори % по метода на Chien-Hrones-Reswick % за компенсиране на заданието чрез дву- % парам. модел на обекта $W(p)=a/L \cdot e^{(-Lp)}$ % и времеконстантата на три-парам. модел % $W(p)=K_o/(1+T_p) \cdot e^{(-Lp)}$. % Входни аргументи: % a – коефициент (скалар); % L – чисто закъснение (скалар); % T – времеконстанта (скалар); % str – символна променлива за типа на % регулатора (скалар), % str = 'П' – настройване на П регулатор; % str = 'ПИ' – настройване на ПИ регулатор; % str = 'ПИД' – настройване на ПИД регул.; % sig – желано пререгулиране в САУ (скалар), % sig = 0 – няма пререгулиране; % sig = ~ 0 – 20 процента пререгулиране. % Изходни аргументи: % K – коеф. на усилване на регул. (скалар); % Ti – интегр. времеконстанта (скалар); % Td – дифер. времеконстанта (скалар). if nargin<4 error('Don't miss the fourth input argument!'), </pre>	<pre> return, end if nargin<5 disp('You choose 0% overshoot!'), sigma=0, end if strcmp(str,'P') if(sigma==0) K=0.3/a; else K=0.7/a; end elseif strcmp(str,'PI') if(sigma==0) K=0.35/a; Ti=1.2*T; else K=0.6/a; Ti=T; end elseif strcmp(str,'PID') if(sigma==0) K=0.6/a; Ti=T; Td=0.5*L; else K=0.95/a; Ti=1.4*T; Td=0.47*L; end else error('Wrong name of the controller!') end </pre>
--	---

Методът на Циглер-Никълс за настройване е реализиран с *PIDtun_ZN2* (табл. 14), а една негова модификация, описана в учебника – с *PIDtun_ZN2m* (табл. 15).

Таблица 14. Функция PIDtun_ZN2

<pre> function [K,Ti,Td]=PIDtun_ZN2(Ku,Tu,str) % Предназначение % Настройване на П, ПИ и ПИД регулатори % по метода на Ziegler- Nichols % чрез критичните параметри на % автоколебанията в САУ. % Входни аргументи: % Ku – коеф. на автоколебания (скалар); % Tu – период на автоколебания (скалар); % str – символна променлива за типа на % регулатора (скалар), % str = 'П' – настройване на П регулатор; % str = 'ПИ' – настройване на ПИ регулатор; % str = 'ПИД' – настройване на ПИД регул.; </pre>	<pre> % Изходни аргументи: % K – коеф. на усилване на регул.(скалар); % Ti – интегр. времеконстанта (скалар); % Td – дифер. времеконстанта (скалар).if nargin<3 error('Don't miss the third input argument!'), return, end if strcmp(str,'P') K=0.5*Ku; elseif strcmp(str,'PI') K=0.4*Ku; Ti=0.8*Tu; elseif strcmp(str,'PID') K=0.6*Ku; Ti=0.5*Tu; Td=0.125*Tu; else error('Wrong name of the controller!') end </pre>
---	---

Таблица 15. Функция PIDtun_ZN2m

<pre> function [K,Ti,Td]=PIDtun_ZN2m(Ku,Tu,str,c) % Предназначение % Настройване на ПИ и ПИД регулатори по % метода на Ziegler-Nichols чрез критичните % параметри на автоколебанията в САУ % с модификация за преместваната точка в % комплексната равнина. % Входни аргументи: % Ku – коеф. на автоколебания (скалар); % Tu – период на автоколебания (скалар); % str – символна променлива за типа на % регулатора (скалар), % str = 'ПИ' – настройване на ПИ регулатор, % str = 'ПИД' – настройване на ПИД регул.; % c – позиция на преместваната точка в % комплексната равнина (комплексно число). % Изходни аргументи: % K – коеф. на усилване на регул.(скалар); % Ti – интегр. времеконстанта (скалар); % Td – дифер. времеконстанта (скалар). if nargin<3 error('Don't miss the third input argument!'), </pre>	<pre> return, end if nargin<4 disp('Standard ZN-2!'), if strcmp(str,'PI'), c = -0.4+0.08i; elseif strcmp(str,'PID'), c = -0.6-0.28i; else error('Wrong name of the controller!'), return end end % Изчисляване на полярните координати на % преместената точка rb=abs(c); fib=angle(c)+pi; % Изчисляване на настройките if strcmp(str,'PI') K=Ku*rb*cos(fib); Ti=-Tu/(2*pi*tan(fib)); elseif strcmp(str,'PID') K=Ku*rb*cos(fib); Ti=Tu/pi*(1+sin(fib))/cos(fib); Td=Ti/4; else error('Wrong name of the controller!') end </pre>
--	---