



УДК 658.512.22

© 2016 г. **Ю.И. Кудинов**, д-р техн. наук,
В.А. Колесников

(Липецкий государственный технический университет),

И.С. Дургарян, канд. техн. наук,

А.Ф. Пащенко, канд. техн. наук,

О.Н. Белова

(Институт проблем управления им. В.А.Трапезникова РАН, Москва)

ПОСТРОЕНИЕ И НАСТРОЙКА НЕЧЕТКОГО АДАПТИВНОГО ПИД-РЕГУЛЯТОРА*

Рассматриваются вопросы построения и настройки адаптивных нечетких ПИД-регуляторов, которые широко применяются в системах автоматического управления технологическими процессами в энергетике. Предложен метод нечеткой адаптации параметров настройки регулятора, учитывающий нелинейные свойства объекта. Описана процедура настройки нечеткого регулятора с использованием программных средств SIMULINK и MATLAB.

Ключевые слова: ПИД-регулятор, нечеткий ПИД-регулятор, нелинейный объект, блок нечеткой адаптации, параметры регулятора, качество регулирования, переходный процесс.

DOI: 10.22250/isu.2016.49.86-96

Введение

Применение линейных пропорционально-интегрально-дифференциальных (ПИД) регуляторов в системах управления нелинейными объектами зачастую приводит к низкому качеству процесса регулирования, которое характеризуется большими значениями перерегулирования, статической ошибкой и/или временем переходного процесса. В настоящей работе предлагается нечеткая адаптация параметров настройки ПИД-регулятора, позволяющая учитывать нелинейные свойства объекта и обеспечивать требуемое качество регулирования при использовании поисковых алгоритмов программного продукта SIMULINK.

Нечеткий адаптивный ПИД-регулятор состоит из линейного ПИД-регулятора (рис. 1), описываемого соотношением

$$u(t) = K_p e(t) + K_i \int_0^t e(t) + K_d \dot{e}(t), \quad (1)$$

и блока нечеткой адаптации (БНА) пропорционального K_p , интегрального K_i и

* Работа выполнена при поддержке РФФИ, проект № 16-08-01279.

дифференциального K_d коэффициентов, содержащего операции фазификации входов (ошибка $e = y^0 - y$ и ее скорость \dot{e}), нечеткие правила, нечеткий вывод и операцию дефазификации трех выходов, т.е. определения и уточнения K_p, K_i, K_d .

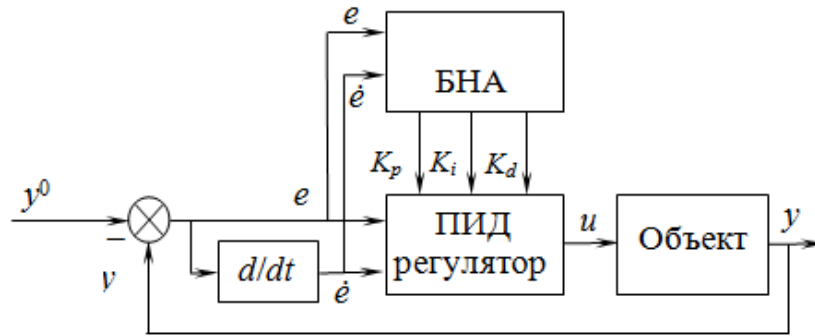


Рис. 1. Схема нечеткого адаптивного ПИД-управления.

Переменные e и $\dot{e} = de/dt$ изменяются в диапазоне $[-3, 3]$, коэффициенты: K_p – в диапазоне $[-0.3, 0.3]$; K_i – в диапазоне $[-0.06, 0.06]$ и принимают 7 лингвистических значений $NB, NM, NS, ZE, PS, PM, PB$. Уб. Ниже рассматривается построение модели БНА, структуры ПИД-регулятора и процедуры параметрической оптимизации, обеспечивающей требуемое качество регулирования.

Построение модели БНА

Начнем с основных принципов построения трех групп нечетких правил, связывающих ошибку e и ее производную \dot{e} с тремя коэффициентами K_p, K_i, K_d [1, 2]:

1) если e относительно велико, то для снижения e следует увеличить K_p и уменьшить K_d , а также исключить интегральное воздействие;

2) если e и \dot{e} допустимы, то следует уменьшить K_p до некоторой требуемой величины для снижения перерегулирования и влияния на систему;

3) если e очень мало, надо увеличить K_p и K_i , чтобы системы обладала устойчивостью. Требуется так изменить K_d , чтобы избежать колебания в системе. Если e мало, то увеличиваем K_d , если \dot{e} велико, уменьшаем K_d .

Составим таблицы управляющих правил согласно рассмотренным правилам.

Запишем нечеткие правила базы правил в табл. 1, с помощью которых уточняются: пропорциональный коэффициент K_p

R_p^1 : если e есть NB , \dot{e} есть NB , то K_p есть PB , ...,

R_p^{49} : если e есть PB , \dot{e} есть PB , то K_p есть NB ; (2)

Таблица 1

\dot{e}	e						
	NB	NM	NS	ZE	PS	PM	PB
NB	PB	PB	PM	PM	PS	ZE	ZE
NM	PB	PB	PM	PS	PS	ZE	NS
NS	PM	PM	PM	PS	ZE	NS	NS
ZE	PM	PM	PS	ZE	NS	NM	NM
PS	PS	PS	ZE	NS	NS	NM	NM
PM	PS	ZE	NS	NM	NM	NM	NB
PB	ZE	ZE	NM	NM	NM	NB	NB

дифференциальный коэффициент K_d (табл. 2)

R_d^1 : если e есть NB , \dot{e} есть NB , то K_d есть NB , ...,

R_d^{49} : если e есть PB , \dot{e} есть PB , то K_d есть PB ; (3)

Таблица 2

\dot{e}	e						
	NB	NM	NS	ZE	PS	PM	PB
NB	NB	NB	NM	NM	NS	ZE	ZE
NM	NB	NB	NM	NS	NS	ZE	ZE
NS	NB	NM	NS	NS	ZE	PS	PS
ZE	NM	NM	NS	ZE	PS	PM	PM
PS	NM	NS	ZE	PS	PS	PM	PB
PM	ZE	ZE	PS	PS	PM	PB	PB
PB	ZE	ZE	PS	PM	PM	PB	PB

коэффициент K_i (табл. 3)

R_i^1 : если e есть NB , \dot{e} есть NB , то K_i есть PS , ...,

R_i^{49} : если e есть PB , \dot{e} есть PB , то K_i есть PB . (4)

Таблица 3

\dot{e}	e						
	NB	NM	NS	ZE	PS	PM	PB
NB	PS	NS	NB	NB	NB	NM	PS
NM	PS	NS	NB	NM	NM	NS	ZE
NS	ZE	NS	NM	NM	NS	NS	ZE
ZE	ZE	NS	NS	NS	NS	NS	ZE
PS	ZE	ZE	ZE	ZE	ZE	ZE	ZE
PM	PB	NS	PS	PS	PS	PS	PB
PB	PB	PM	PM	PS	PS	PS	PB

При построении классических регуляторов сегодня широко используются средства MATLAB [3]. Для определения параметров нечеткого регулятора будут использованы и программные средства SIMULINK.

Опишем процедуру построения замкнутой цифровой системы управления, содержащей БНА, настраиваемый ПИД-регулятор, а также объект управления с передаточной функцией

$$W_o(s) = \frac{1}{s^3 + 3s^2 + 3s + 1}. \quad (5)$$

Для создания модели цифровой системы управления в пакете динамического моделирования Simulink в окне MATLAB нажмем на вкладку Simulink Library, откроется окно элементов Simulink Library Browser. Командой **File, New, Model** откроем окно модели “Untitled”.

В окне “Simulink Library Browser” (рис. 2) найдем категорию Simulink, в ней выделим подкатеорию Sources, а в правой части окна – блок Step, вырабатывающий единичную ступенчатую функцию. Удерживая нажатой левую кнопку мыши, перетащим значок блока Step в окно моделирования “Untitled”.

Двойным щелчком на блоке Step откроем окно параметров, в котором уста-

новим момент возникновения единичного скачка Step time: 0 с, его начальную величину Initial value: 0, конечную величину Final value: 1 и интервал дискретности Sample time: 0.1, который по умолчанию установим в других диалоговых окнах.

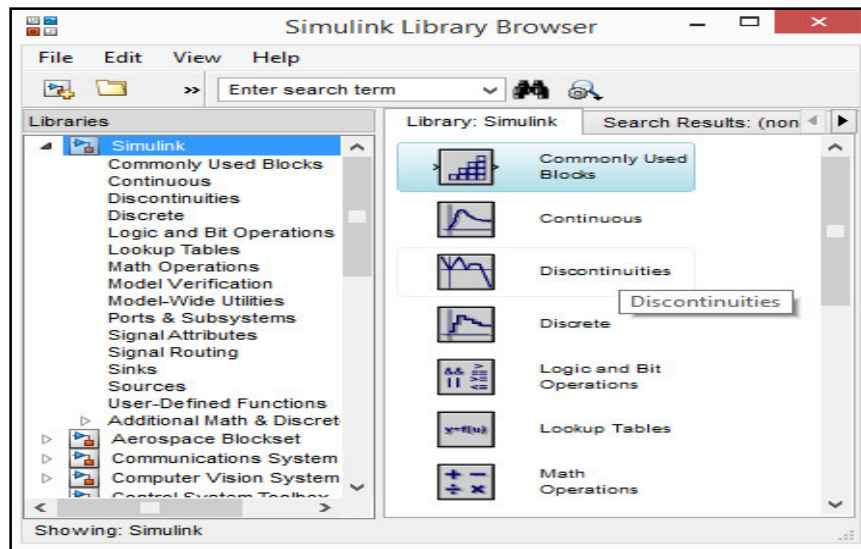


Рис. 2. Окно Simulink Library Browser.

Аналогичным образом из подкатегории Math Operations перетащим блок сравнения Sum в окно моделирования (рис. 3). Двойным щелчком на элементе сравнения Sum, откроем окно параметров, в поле List of sign (список знаков) правый знак + заменим на – и нажмем ОК. При нажатой левой клавише мыши соединим блоки Step и Sum.

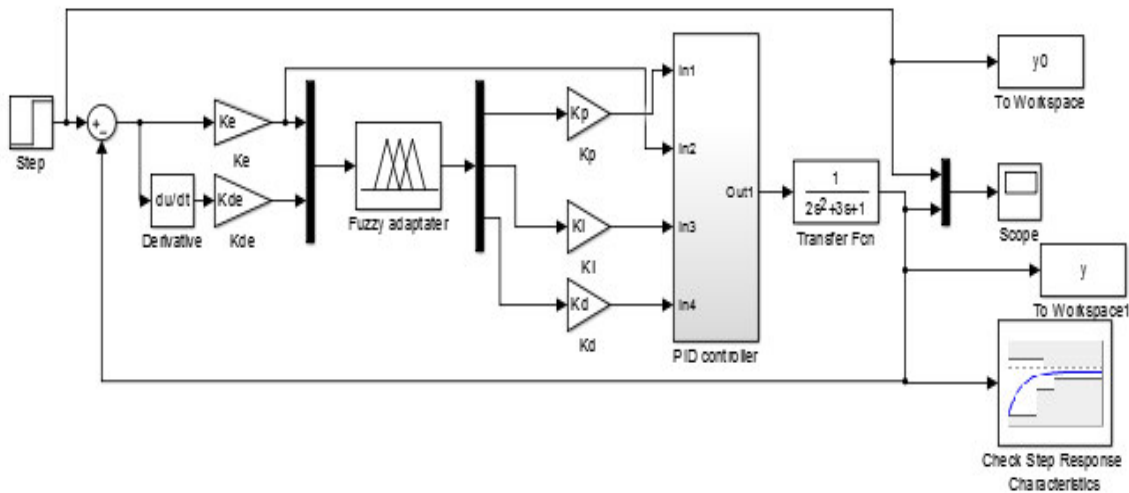


Рис. 3. Схема нечеткого адаптивного ПИД-управления.

В модель цифровой замкнутой системы управления из категории Simulink и подкатегории Math Operations перетащим два блока Gain в окно моделирования. Двойным щелчком на блоке Gain (или Gain1) откроем окно параметров и зададим Ke (или Ku для блока Gain1) в поле Gain, нажмем кнопку ОК. Из подкатегории Simulink Design Optimization перейдем в подкатегию Signal Constraints и переместим в модель блок Check Step Response Characteristic.

Двойным щелчком на категории Fuzzy Logic Toolbox выделим блок нечеткого регулятора Fuzzy Logic Controller with Ruleviewer и перетащим его в окно модели,

обозначив его как Fuzzy adapter.

Приступим к построению БНА, именуемому как Fuzzy adapter, с помощью редактора системы нечеткого вывода FIS Editor. Заменяя $\hat{e} = K_e e$ на e , $\hat{de} = K_{de} \dot{e}$ на de , K_p на K_r , K_d на K_d и K_i на K_i , запишем правила (1) – (3) БНА в более удобном для программирования виде:

$$R_p^1 : \text{если } e \text{ есть NB, } de \text{ есть NB, то } K_r \text{ есть PB, } \dots, \\ R_p^{49} : \text{если } e \text{ есть PB, } de \text{ есть PB, то } K_r \text{ есть NB;} \quad (6)$$

$$R_d^1 : \text{если } e \text{ есть NB, } de \text{ есть NB, то } K_d \text{ есть NB, } \dots, \\ R_d^{49} \text{ если } e \text{ есть PB, } de \text{ есть PB, то } K_d \text{ есть PB;} \quad (7)$$

$$R_i^1 \text{ если } e \text{ есть NB, } de \text{ есть NB, то } K_i \text{ есть PS, } \dots, \\ R_i^{49} : \text{если } e \text{ есть PB, } de \text{ есть PB, то } K_i \text{ есть PB.} \quad (8)$$

Для запуска FIS-редактора в командной строке MATLAB введем команду `fuzzy`, нажмем Enter, и откроется окно “FIS Editor Untitled”. Командой **File, New FIS, Mamdani** выберем тип нечеткого вывода Mamdani. Командой **Edit, Add Variable, Input** добавим еще один вход. Щелчком мыши выделим блок входной переменной `input1`, в поле Name введем `e` и нажмем Enter. Так же выделим блок `input2` и в поле Name введем `de` и нажмем Enter. Командой **Edit, Add Variable, Output** добавим еще два выхода. После чего выделим блок выходной переменной `output1` и в поле Name введем `K_r`, в блоке `output2` в поле Name введем `K_i`, в блоке `output3` в поле Name введем `K_d`. В поле Defuzzification выберем метод `bisector`. Верхняя часть FIS-редактора будет иметь вид, показанный на рис. 4.

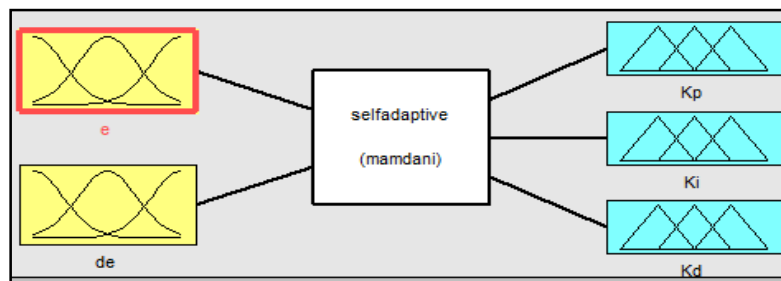


Рис. 4. Верхняя часть окна FIS-редактора.

Двойным щелчком по блоку входной переменной `e` откроем окно “Membership Function Editor:Untitled”. Щелчком мыши выделим первую функцию принадлежности (ФП) `mf1`. В поле Range установим диапазон `[-3 3]` и подтвердим его щелчком мыши в поле `Display Range`. Далее в поле Name введем имя ФП `NB` и нажмем Enter, в поле Type выберем ФП `zmf`, зададим значение Params `[-3 -1]` и нажмем Enter. Выделим вторую ФП `mf2`, в поле Name введем имя `NM` нажмем Enter, проверив в Type наличие треугольной ФП `trimf`, зададим значение Params `[-3 -2 0]` и нажмем Enter. Выделим третью ФП `mf3`, в поле Name введем `NS`, в Type `trimf`, зададим значение Params `[-3 -1 1]` и нажмем Enter. Аналогичным образом построим остальные ФП. ФП для входной переменной `e` будут иметь вид, изображенный на рис. 5. Повторим аналогичные действия с входным блоком `de`. ФП будут иметь вид как на рис. 6.

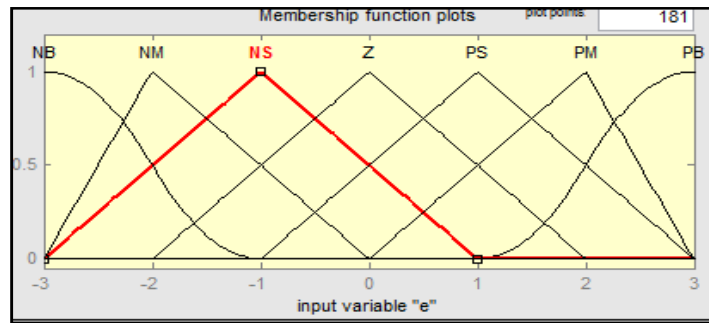


Рис. 5. ФП для входа e.

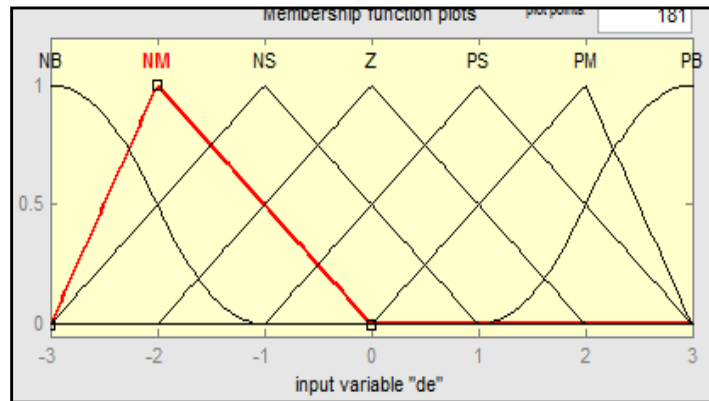


Рис. 6. ФП для входа de.

В окне “Membership Function Editor:Untitled” щелчком мыши нажмем на блоке u1 и перейдем к построению ФП выходной переменной u1. Командой **Edit, Add MFs** добавим четыре функции trimf. Определим пределы Range и Display Range, равные $[-0.3 \ 0.3]$, после чего первой mf1 зададим имя NB, тип – zmf, значение Params $[-0.3 \ -0.1]$; второй mf2 – имя NM, значение Params $[-0.3 \ -0.2 \ 0]$; третьей mf3 – имя NS, значение Params $[-0.3 \ -0.1 \ 0.1]$; четвертой mf4 – имя Z, значение Params $[-0.2 \ 0 \ 0.2]$; пятой mf5 – имя PS, значение Params $[-0.1 \ 0.1 \ 0.3]$; шестой mf5 – имя PM, значение Params $[0 \ 0.2 \ 0.3]$; седьмой mf5 – имя PB, тип – zmf, значение Params $[0.1 \ 0.3]$. ФП для выходной переменной Kp имеют вид, показанный на рис. 7.

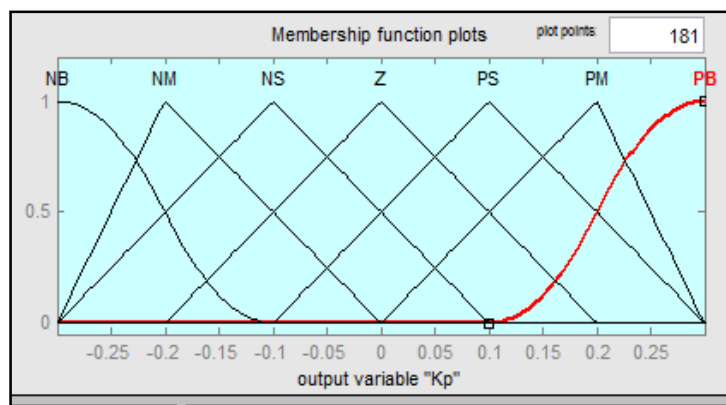


Рис. 7. ФП для коэффициента Kp.

Повторим аналогичные действия с выходными блоками Ki и Kd . ФП будут иметь вид, представленный на рис. 8 и рис. 9.

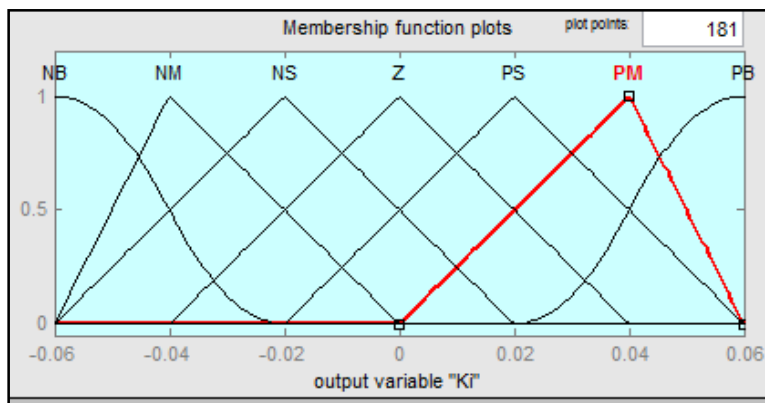


Рис. 8. ФП для коэффициента K_i .

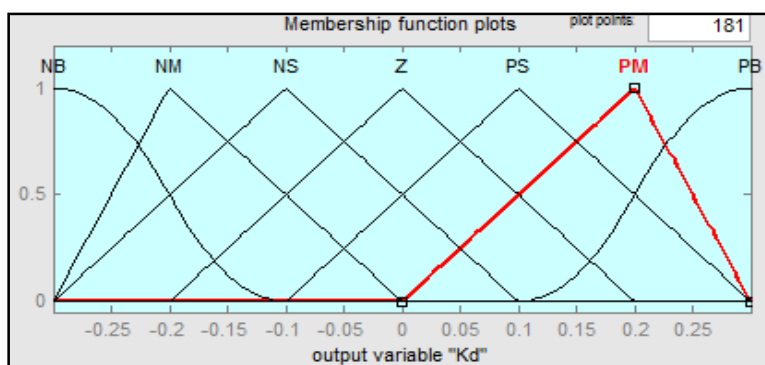


Рис. 9. ФП для коэффициента K_d .

В окне “FIS Editor Untitled” командой **Edit, Rules** откроем окно редактора правил “Rule Editor: Untitled”. Щелчком мыши в левом поле If e is выделим функцию принадлежности NB, в поле de выделим ФП NB, а в правом поле Then Kp is – PB, Ki is – PS, Kd is – NB, после чего нажмем кнопку Add rule. Аналогичным образом добавим остальные правила (6) – (8). В верхнем поле появятся $49 \cdot 3 = 147$ нечетких правил (рис. 10).

```

35. If (e is PS) and (de is PB) then (Kp is NM)(Ki is PS)(Kd is PM) (1)
36. If (e is PM) and (de is NB) then (Kp is Z)(Ki is NM)(Kd is Z) (1)
37. If (e is PM) and (de is NM) then (Kp is Z)(Ki is NS)(Kd is Z) (1)
38. If (e is PM) and (de is NS) then (Kp is NS)(Ki is NS)(Kd is PS) (1)
39. If (e is PM) and (de is Z) then (Kp is NM)(Ki is NS)(Kd is PM) (1)
40. If (e is PM) and (de is PS) then (Kp is NM)(Ki is Z)(Kd is PM) (1)
41. If (e is PM) and (de is PM) then (Kp is NM)(Ki is PS)(Kd is PB) (1)
42. If (e is PB) and (de is NB) then (Kp is Z)(Ki is PS)(Kd is Z) (1)
43. If (e is PB) and (de is NM) then (Kp is NS)(Ki is Z)(Kd is Z) (1)
44. If (e is PB) and (de is NS) then (Kp is NS)(Ki is Z)(Kd is PS) (1)
45. If (e is PB) and (de is Z) then (Kp is NM)(Ki is Z)(Kd is PM) (1)
46. If (e is PB) and (de is PS) then (Kp is NM)(Ki is Z)(Kd is PB) (1)
47. If (e is PB) and (de is PB) then (Kp is NB)(Ki is PB)(Kd is PB) (1)
48. If (e is PB) and (de is PM) then (Kp is NB)(Ki is PB)(Kd is PB) (1)
49. If (e is PM) and (de is PB) then (Kp is NB)(Ki is PS)(Kd is PB) (1)

```

Рис. 10. Фрагмент базы правил для коэффициентов K_p , K_i , K_d .

Для выхода из окна “Rule Editor: Untitled” нажмем кнопку Close. Аналогично закроем окно “Membership Function Editor: Untitled”. В окне “FIS Editor Untitled” сохраним FIS модель БНА командой **File, Export, To Disk** под именем Fuzzy_adapter.fis. Поместим FIS модель в рабочую область MATLAB. Для этого

командой **File, Export, To Workspace** откроем окно, в поле **Workspace variable** введем **Fuzzy_adapter** и нажмем **ОК**, чтобы воспользоваться БНА в Simulink. Свернем окно FIS редактора.

Структура настраиваемого ПИД-регулятора

Определим структуру ПИД-регулятора, представленного на рис. 3, в виде блока с именем **PID controller**.

В окно модели из категории **Simulink** и подкатегории **Commonly Used Block** перетащим три блока **Gain**, задающие начальные значения коэффициентов $K_p=4$, $K_i=3$, $K_d=1.6$, два блока произведения **Product**, а также интегрирующий блок **Integrator**. Из подкатегории **Continuous** переместим блок дифференцирования **Derivative**, а из подкатегории **Math Operation** – четыре блока суммирования **Add**. Соединяя эти элементы согласно рис. 11, получим схему ПИД-регулятора, имеющего четыре входа **In** и один выход **Out**.

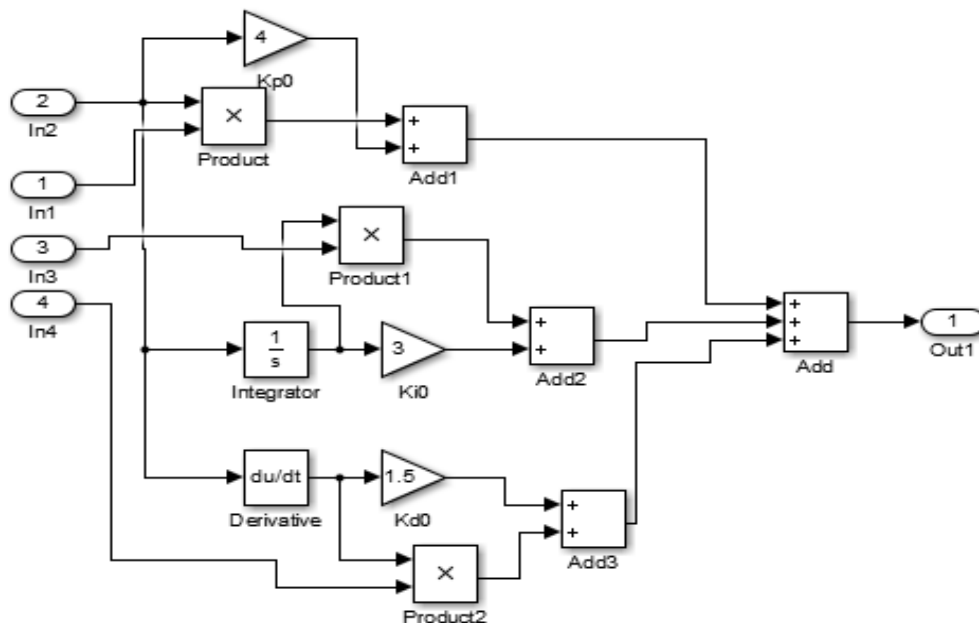


Рис. 11. Схема настраиваемого ПИД-регулятора.

На вход **In2** подается ошибка регулирования, которая затем поступает в пропорциональную (блок K_p), интегральную (блок **Integrator**) и дифференциальную (блок **Derivative**) части ПИД-регулятора, которые в сумматорах **Add1** – **Add3** складываются с соответствующими П, И, Д составляющими. Последние образованы умножением ошибки регулирования на K_p , интеграла ошибки на K_i и производной от ошибки на K_d , причем K_p , K_i , K_d рассчитываются в РНА и поступают на входы **In1**, **In3**, **In4** настраиваемого ПИД-регулятора.

Из подкатегории **Continuous** перетащим блок передаточной функции объекта **Transfer Fcn** в окно модели. Дважды щелкнем на блоке **Transfer Fcn** и откроем окно передаточной функции с параметрами числителя (**Numerator**) и знаменателя (**Denominator**) для ввода передаточной функции объекта

Согласно выражению передаточной функции объекта (5) введем следующие значения для **Numerator**: [1] и **Denominator**: [1 3 3 1], нажмем **ОК**. Соединим блоки

Sum, Gain, Zero-Order Hold, Fuzzy Logic Controller with Ruleviewer, Gain2, Zero-Order Hold1, Transfer Fcn.

В окно модели из категории Simulink и подкатегории Sink перетащим два блока To Workspace, подкатегории Commonly Used Block – блок мультиплексора сигналов Mux и блок осциллографа Scope с двумя входами. Двойным щелчком по блоку Scope откроем его окно и введем ряд параметров. Соединим блок Transfer Fcn с блоком Mux, а последний – с блоком Scope. Аналогичным образом проведем линию отрицательной обратной связи, соединяющей блоки Transfer Fcn и Mux, до блока Sum.

В результате получим модель замкнутой цифровой системы управления (рис. 3), содержащей блок нечеткой адаптации (Fuzzy adapter) коэффициентов K_p , K_i , K_d ПИД-регулятора (PID-controller) и объект управления с передаточной функцией (5).

Оптимизация параметров настройки БНА и ПИД-регулятора

Перейдем к модели в Simulink, двойным щелчком мыши откроем окно параметров блока Fuzzy Logic Controller with Ruleviewer и в поле FIS matrix вместо Fis введем Fuzzy_adapter. В командной строке MATLAB зададим значения коэффициентов $K_e=1$, $K_{de}=1$, $K_p=1$, $K_d=1$, $K_i=1$ и запустим процесс моделирования системы управления нажатием кнопки Start simulation. В блоках Display отразятся текущие значения коэффициентов K_e , K_{de} , K_p , K_d и K_i . В результате оптимизации получим график переходного процесса в окне Scope (рис. 12).

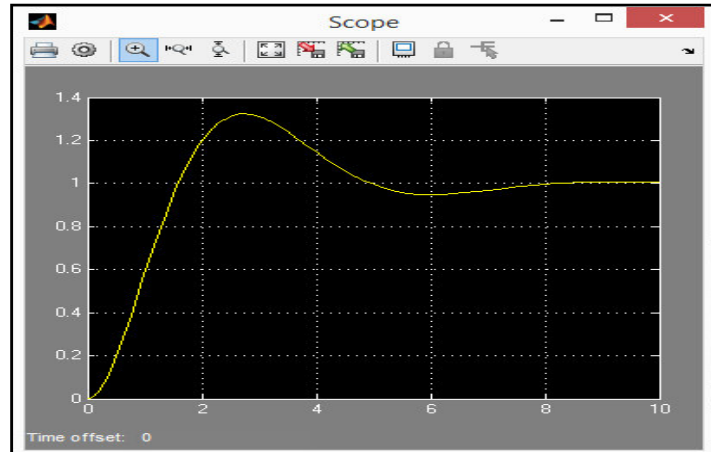


Рис. 12. Переходный процесс для исходных значений коэффициентов.

Переходный процесс на рис. 12 имеет следующими показателями качества: время переходного процесса 9с и перерегулирование 13%, которые не удовлетворяют принятым и заданным в окне свойств “Check Step Response Characteristics” (рис. 13). Здесь: Rise time (время нарастания) – не более 5с; Settling time (время переходного процесса) – не более 7с; Overshoot (перерегулирование) – не более 10%.

Запустим Response Optimization для настройки нечеткого ПИД-регулятора. Перетащим мышью в поле Model Workspace коэффициенты K_d , K_{de} , K_e , K_i , K_p , подлежащие оптимизации. После чего зададим переменные, за счет оптимизации которых будет улучшено качество переходного процесса.

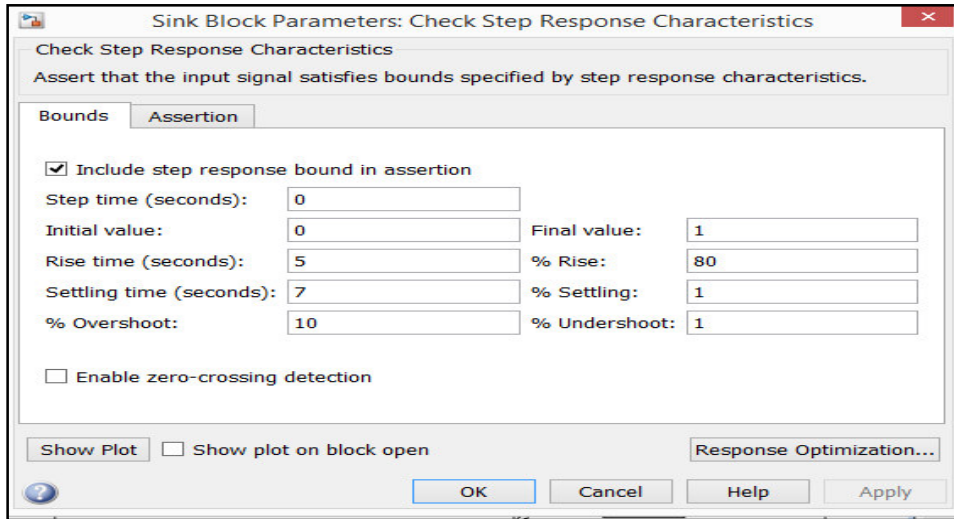


Рис. 13. Установка ограничений переходного процесса.

Для этого на вкладке Response Optimization в поле Design Variables Set выберем пункт New. Откроется окно “Create Design Variables Set”, в правой части которого выберем настраиваемые переменные Kd, Kde, Ke, Ki, Kp и с помощью стрелки перенесем их в левую часть окна, как изображено на рис. 14.

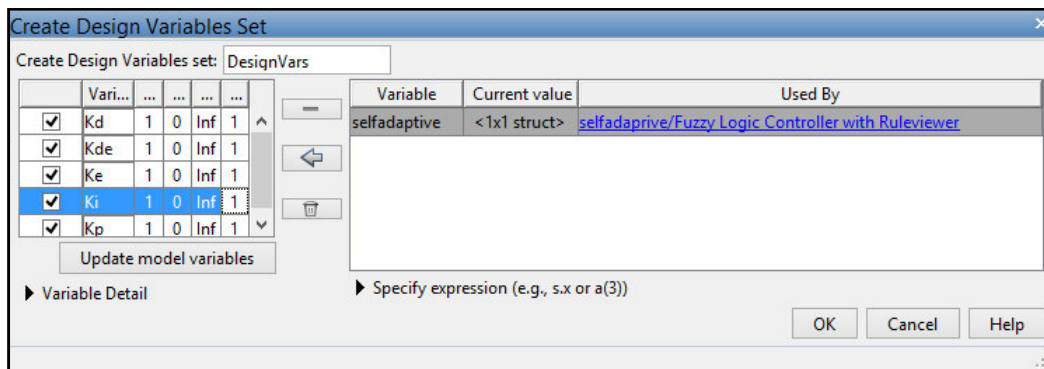


Рис. 14. Установка граничных значений коэффициентов.

В окне “Design Optimization” запустим процесс оптимизации параметров регулятора, нажав кнопку Optimize. В ходе процесса оптимизации в поле Method окна “Response Optimization Options” (рис. 15) можно устанавливать и использовать последовательно несколько поисковых алгоритмов.

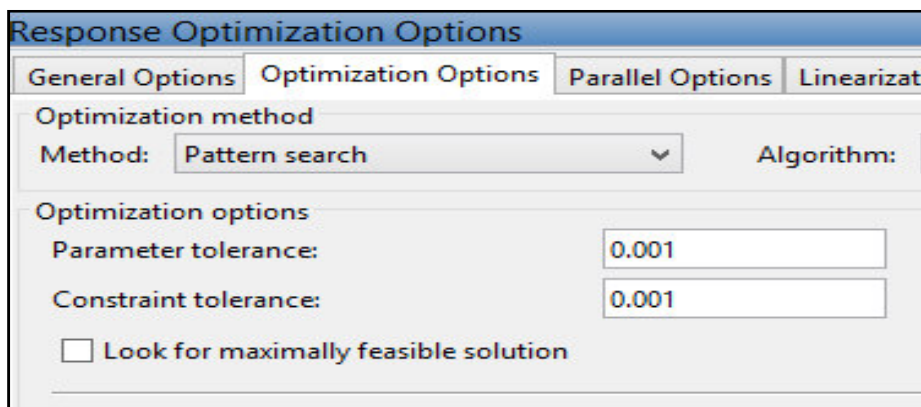


Рис. 15. Задание поискового алгоритма.

После завершения оптимизации в окне “Design Optimization” выводится график оптимального переходного процесса (рис. 16).

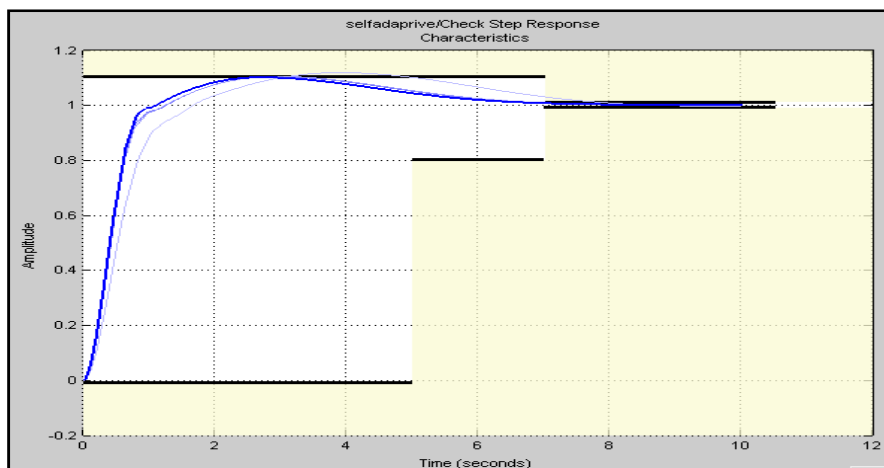


Рис. 16. Оптимальный переходный процесс.

Из рис. 16 видно, что полученный переходный удовлетворяет принятым ограничениям при следующих значениях коэффициентов: $K_e = 11.36$; $K_{de} = 16.77$; $K_p = 16.99$; $K_d = 6.2$; $K_i = 68.33$.

Заключение

В работе рассмотрены вопросы построения и настройки нечеткого адаптивного ПИД-регулятора, состоящего из блока нечеткой адаптации нечетких коэффициентов линейного ПИД-регулятора.

Средствами MATLAB-SIMULINK определены оптимальные значения параметров ПИД-регулятора и коэффициентов блока нечеткой адаптации, обеспечивающие требуемое качество переходного процесса в замкнутой системе с нечетким адаптивным регулятором.

ЛИТЕРАТУРА

1. Jin J., Huang H., Sun J., Pang Y. Study on Fuzzy Self-Adaptive PID Control System of Biomass Boiler Drum Water // Journal of Sustainable Bioenergy Systems. – 2013. – V. 3. – P.93-98.
2. Кудинов Ю.И., Келина А.Ю. Упрощенный метод определения параметров нечетких ПИД-регуляторов // Мехатроника, автоматизация, управление. – 2013. – № 1. – С.12-22.
3. Pashchenko F.F., Pashchenko A.F., Durgaryan I.S., Kudinov Y.I., Kelina A.Y. On Neuro-Fuzzy Prediction in Matlab // IEEE (ICEA).10-th Conference Industrial Electronics and Applications. – Auckland, New Zeland. – 2015. – P.1539-1542.

Статья представлена к публикации членом редколлегии Е.А. Ереминым.

E-mail:

Кудинов Юрий Иванович – kui_kiu@lipetsk.ru;

Колесников Валентин Анатольевич – valentinn48@ya.ru;

Дургарян Ирина Суреновна – pif-70@yandex.ru;

Александр Федорович – paschenko_alex@mail.ru;

Белова Ольга Николаевна – pif-70@yandex.ru.