



**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«ДОНСКОЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ»
(ДГТУ)**

**ВАРИАНТЫ ЗАДАНИЙ КОНТРОЛЬНОЙ РАБОТЫ ПО
ДИСЦИПЛИНЕ «ИНТЕЛЛЕКТУАЛЬНЫЙ ПРИВОД В МЕХАТРОННЫХ И
РОБОТОТЕХНИЧЕСКИХ СИСТЕМАХ» И МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ
ИХ ВЫПОЛНЕНИЯ
2 СЕМЕСТР**

Учебно-методическое пособие

Предназначено для обучающихся 1-го курса заочной формы обучения по направлению 15.04.06 – Мехатроника и робототехника программа - Интеллектуальные методы обработки сенсорной информации и принятия решений в робототехнике

Ростов-на-Дону

2024

1

Составитель: кандидат технических наук, доцент Филимонов М.Н.

Приведены варианты заданий контрольной работы для обучающихся заочной формы обучения по основным темам, соответствующие базовому уровню изучения дисциплины «Интеллектуальный привод в мехатронных и робототехнических системах» во 2-м семестре. Приведены образцы решения всех заданий, снабжённые необходимыми теоретическими сведениями.

ВВЕДЕНИЕ

Учебно-методическое пособие содержит индивидуальные задания двух контрольных работ, выполняемых студентами заочной формы обучения. Тематика заданий охватывает все основные разделы дисциплины «Интеллектуальный привод в мехатронных и робототехнических системах». Представлены основные теоретические положения и понятия, соответствующие базовому уровню изучения дисциплины, и подробное решение всех заданий. Выбор тематики осуществлялся на основе анализа ФГОС3++ в базовой подготовке магистров направления 15.04.06. Также приведен список теоретических вопросов для подготовки к экзамену и рекомендуемая литература.

1 Выбор варианта задания

Задание имеет 30 вариантов, правило выбора варианта приведено перед заданиями контрольных работ.

Для выполнения контрольной работы необходимо выполнить практическую работу из 8 заданий. Номер варианта определяется двумя последними цифрами шифра зачетной книжки, по которой,

воспользовавшись следующей таблицей, следует определить по вертикали вариант задания в каждой из практических работ (одинаковый для всех практических работ).

Определение вариантов контрольной работы

Шифр	00	01	02	03	04	05	06	07	08	09
	30	31	32	33	34	35	36	37	38	39
	60	61	62	63	64	65	66	67	68	69
	90	91	92	93	94	95	96	97	98	99
Вариант практических работ	1	2	3	4	5	6	7	8	9	10
	10	11	12	13	14	15	16	17	18	19
	40	41	42	43	44	45	46	47	48	49
	70	71	72	73	74	75	76	77	78	79
Вариант практических работ	11	12	13	14	15	16	17	18	19	20
	20	21	22	23	24	25	26	27	28	29
	50	51	52	53	54	55	56	57	58	59
	80	81	82	83	84	85	86	87	88	89
Вариант практических работ	21	22	23	24	25	26	27	28	29	30

2 Задание на контрольную работу

В данном разделе приводятся задания, которые необходимо выполнить в контрольной работе. По практическим задачам даны краткие методические рекомендации, а также ссылки на литературные источники, в которых подробно освещены соответствующие темы заданий.

Для выполнения работы необходимо:

1) Выбрать в соответствии со своим вариантом модель двигателя постоянного тока и выполнить 8 практических заданий.

2) Перед решением практических заданий внимательно изучить примеры их выполнения.

В целом оформление работы произвольно с применением пакета MS Office любой версии, однако непременным требованием является наличие всех поясняющих рисунков, надписей, схем, графиков и аналитических выкладок, а сами расчёты выполнять в среде Matlab+Simulink.

3 Список литературы

1. Анучин А.С. Системы управления электроприводов: учебник для вузов. М.: Издательский дом МЭИ, 2015. – 373. с.
2. Шрейнер Р.Т. Математическое моделирование электроприводов переменного тока с полупроводниковыми преобразователями частоты. Екатеринбург. УРО РАН, 2000. – 654 с.
3. Денисенко В.В. Компьютерное управление технологическим процессом, экспериментом, оборудованием. — М.: Горячая линия-Телеком, 2009. — 608 с.
4. А. Г. Александров, М. В. Паленов, Состояние и перспективы развития адаптивных ПИД-регуляторов, Автомат. и телемех., 2014, выпуск 2, 16–30.

5. Aidan O'Dwyer. Handbook of PI and PID controller tuning rules, 3rd Edition. - London: Imperial College Press, 2009. – 608 p.
6. Справочник по электрическим машинам: В 2 т./ Под общ. ред. И.П. Копылова и Б.К. Клокова. Т. 1. – М.: Энергоатомиздат, 1988. – 456 с.
7. Using the NN Predictive Controller Block. url: <http://matlab.izmiran.ru/help/toolbox/nnet/controlba.html> (дата обращения 31.03.2024).
8. Герман-Галкин С. Г. Виртуальные лаборатории полупроводниковых систем в среде Matlab-Simulink: Учебник.— СПб.: Издательство «Лань», 2022.— 448 с.
9. Никитенко Г. В. Электропривод производственных механизмов: Учебное пособие. — 2-е изд., испр. и доп. — СПб.: Издательство «Лань», 2022. — 224 с.
10. Кацман М. М. Справочник по электрическим машинам: Учеб, пособие для студ. образовав учреждений сред. проф. образования / Марк Михайлович Кацман. — М.: Издательский центр «Академия», 2005. — 480 с.

4 Варианты практических заданий для контрольной работы

Для выбранного по номеру варианта двигателя постоянного тока:

1. Рассчитать параметры звеньев динамической модели электропривода постоянного тока с независимым возбуждением по двухконтурной схеме (контур тока и контур скорости) подчинённого регулирования.
2. В Matlab построить динамическую модель электропривода постоянного тока с независимым возбуждением по двухконтурной схеме (контур тока и контур скорости) подчинённого регулирования.
3. Исследовать динамические свойства двигателя постоянного тока с независимым возбуждением (ДПТ НВ) как объекта управления. Для этого определить характеристики переходного процесса (время регулирования, перерегулирование, колебательность, коэффициент усиления) по изменению тока якоря и угловой скорости якоря ДПТ при ступенчатом изменении напряжения якоря.
4. В Matlab выполнить синтез пропорционально-интегрально-дифференциальных регуляторов тока и угловой скорости якоря по критерию минимума динамической ошибки регулирования. Определить характеристики переходного процесса (время регулирования, перерегулирование, колебательность, коэффициент усиления) по изменению тока якоря и угловой скорости якоря ДПТ при ступенчатом изменении задания угловой скорости якоря. Сравнить с характеристиками разомкнутой системы регулирования.
5. В Matlab выполнить синтез нечёткого регулятора скорости. Определить характеристики переходного процесса (время регулирования, перерегулирование, колебательность, коэффициент усиления) по изменению угловой скорости якоря ДПТ при ступенчатом изменении задания угловой скорости якоря.
6. В Matlab выполнить синтез регулятора скорости на основе искусственной нейронной сети прямого распространения. Определить характеристики

- переходного процесса (время регулирования, перерегулирование, колебательность, коэффициент усиления) по изменению угловой скорости якоря ДПТ при ступенчатом изменении задания угловой скорости якоря.
7. В Matlab выполнить синтез предиктивного регулятора скорости на основе искусственной рекуррентной нейронной сети. Определить характеристики переходного процесса (время регулирования, перерегулирование, колебательность, коэффициент усиления) по изменению угловой скорости якоря ДПТ при ступенчатом изменении задания угловой скорости якоря.
 8. Сравнить характеристики переходных процессов по изменению угловой скорости якоря ДПТ при применении разных регуляторов. Сделать выводы по целесообразности применения того или иного регулятора в системе управления электропривода.

Варианты заданий

№ п/п	Мощность, кВт	Напряжение, В	Частота вращения, об/мин		кпд, %	Сопротивление обмотки при 15 °С, Ом			Индуктивность цепи якоря, мГн	Момент инерции якоря, кг· м ²
			номинальная	максимальная		якоря	добавочных полюсов	возбуждения		
Тип 2ПН90МУХЛ4, 2ПН90МГУХЛ4										
1.	0,17	110	750	3000	47,5	5,84	4,40	610	128	0,004
		220	750	1500	48,5	27,2	16,2	162	514	
2.	0,25	110	1060	4000	56	3,99	2,55	610	78,7	
		220	1120	2000	57	15,47	11,2	162	297	
3.	0,37	110	1500	3000	61,5	2,52	1,47	610	48	
		220	1500	2250	61,5	10,61	6,66	162	190	
4.	0,71	110	2360	4000	69,5	1	0,54	470	18	
		220	2360	3540	70	3,99	2,55	123	79	
5.	1	110	3000	4000	71,5	0,6	0,35	365	12	
		220	3000	4000	72,5	2,52	1,47	92	48	
Тип 2ПН90ЛУХЛ4, 2ПН90ЛГУХЛ4										
6.	0,2	110	750	3000	54	4,51	2,99	555	106	0,005
		220	800	1500	54,5	17,5	13,13	145	430	
7.	0,34	110	1060	4000	60	2,85	1,731	555	64,5	
		220	1000	2000	60	12,2	7,96	145	264	
8.	0,55	110	1500	4000	67,5	1,3	0,932	432	33	
		220	1500	4300	67,5	5,44	3,89	112	132	
9.	0,9	110	2000	4000	73	0,644	0,4	340	14,6	
		220	2120	4000	73	2,85	1,731	87	64	
10.	1,3	110	3150	4000	76	0,355	0,257	340	8,2	
		220	3150	4000	78	1,3	0,932	87	33	
Тип 2ПБ90МУХЛ4, 2ПБ90МГУХЛ4										
11.	0,13	110	800	3000	49,5	7,48	4,79	810	173	0,004
		220	750	1500	47,5	34,0	20,37	222	694	
12.	0,18	110	1000	4000	54,5	5,41	3,47	810	122	
		220	1000	2000	55,5	21,77	13,31	222	490	
13.		110	1600	4000	63,5	2,69	1,85	810	62	
		220	1500	3000	63,5	11,7	7,35	222	267	
14.		110	2360	4000	69,5	1,46	0,87	810	34	
		220	2200	4000	58,5	6,84	4,40	222	147	
		110	3000	4000	71,0	1	0,58	810	21	

15.		220	3000	4000	71,0	3,99	2,55	222	82	
Тип 2ПБ90ЛУХЛ4, 2ПБ90ЛГУХЛ4										
16.		110	750	3000	53	5,44	3,89	720	150	0,005
		220	750	1500	53	23,55	15	192	610	
17.		110	1000	4000	58,5	4,01	2,3	720	96	
		220	1120	2000	61,5	13,25	8,65	192	360	
18.		110	1500	4000	66	2,1	1,297	720	54	
		220	1500	3000	67,5	7,74	5,17	192	199	
19.		110	2200	4000	72	1,13	0,687	720	28	
		220	2240	4000	73	4,26	2,84	192	108	
20.		110	3000	4000	78,5	0,644	0,4	720	17	
		220	3150	4000	77	2,28	1,609	192	63	
Тип 2ПН100МУХЛ4, 2ПН100МГУХЛ4										
21.		110	750	3000	60	2,69	1,62	470	89	0,011
		220	750	1500	59,5	11,78	6,7	120	343	
22.		110	1000	4000	65	1,79	0,93	470	53	
		220	1000	2000	66	7,05	4,62	120	222	
23.		110	1500	4000	71	0,805	0,57	359	26	
		220	1500	4300	71,5	3,4	2,05	103	104	
24.		110	2120	4000	75	0,436	0,355	359	14	
		220	2200	4000	76,5	1,792	0,93	103	53	
25.		110	3000	4000	78,5	0,201	0,135	265	6,6	
		220	3000	4000	79	0,805	0,57	73	26	
Тип 2ПН100ЛУХЛ4, 2ПН100ЛГУХЛ4										
26.		110	750	3000	60,5	2,05	1,28	397	71	0,012
		220	750	1500	61,5	7,87	6,2	114	286	
27.		110	1000	4000	67	1,17	0,853	397	42,3	
		220	1060	2000	67	4,92	3,52	114	169	
28.		110	1500	4000	72,5	0,52	0,51	295	21	
		220	1500	4300	74	2,2	1,57	81	83	
29.		110	2200	4000	77	0,294	0,235	295	10,6	
		220	2200	4000	78	1,17	0,853	81	42	
30.		110	3000	4000	79,5	0,16	0,146	295	5,7	
		220	3150	4000	81	0,52	0,51	81	21	

$k_{пр.с}$, $k_{ир.с}$, $k_{др.с}$ – пропорциональный, интегральный и дифференциальный коэффициенты ПИД-регулятора скорости соответственно.

Для расчёта параметров структурной схемы можно воспользоваться следующими известными соотношениями:

$$C = \frac{U_{ян} - I_{ян}R_{яц}}{\omega_n},$$

$$T_я = \frac{L_{яц}}{R_{яц}}, T_м = \frac{JR_{яц}}{(k\Phi)^2}, T_\mu = T_п, T_п = \frac{1}{f_k},$$

где - $U_{ян}$ – номинальное напряжения питания цепи якоря ДПТ,

$L_{яц}$ – индуктивность якорной цепи, обычно $L_{яц} \approx L_я$, где $L_я$ – индуктивность якоря;

$R_{яц}$ – сопротивление якорной цепи, $R_{яц} = R_я + R_{дп}$, где $R_я$ – сопротивление якоря,

$R_{дп}$ – сопротивление добавочных полюсов;

ω_n – номинальная угловая скорость якоря, $\omega_n = \frac{2\pi n_n}{60}$, где n_n – номинальная частота вращения якоря,

$I_{ян}$ – номинальный ток якоря, $I_{ян} = \frac{P_n}{\eta_n U_{ян}}$, где P_n – номинальная мощность ДПТ, η_n – номинальный КПД ДПТ;

J – приведенный к якорю момент инерции нагрузки, $J \approx J_я$, где $J_я$ – момент инерции якоря;

$T_п$ – постоянная времени широтно-импульсного преобразователя;

f_k – частота коммутации широтно-импульсного преобразователя, обычно составляет от 250 до 3000 Гц.

Рассчитаем параметры динамической модели на рисунке 1 для ДПТ модели 2ПН90МУХЛ4 мощностью $P_n = 170$ Вт.

Номинальная угловая скорость якоря

$$\omega_n = \frac{2\pi \cdot 750}{60} = 78,5 \frac{\text{рад}}{\text{с}}.$$

Номинальный момент двигателя

$$M_n = \frac{P_n}{\omega_n} = \frac{170}{78,5} = 2,16 \text{ Н} \cdot \text{м}.$$

Сопротивление якорной цепи $R_{яц} = 27,2 + 16,2 = 43,4$ Ом.

Номинальный ток якоря

$$I_{ян} = \frac{170}{0,485 \cdot 220} = 1,6 \text{ А.}$$

Постоянная ДПТ

$$C = \frac{220 - 1,6 \cdot 43,4}{78,5} = 1,92 \text{ В} \cdot \text{с.}$$

Постоянные времени

$$T_{я} = \frac{0,514}{43,4} = 0,0118 \text{ с;}$$

$$T_{м} = \frac{0,004 \cdot 43,4}{(1,92)^2} = 0,0471 \text{ с;}$$

$$T_{\mu} = T_{п} = \frac{1}{1000} = 0,001 \text{ с.}$$

Коэффициенты регуляторов тока и регулятора скорости

$$k_{пр.т.} = \frac{L_{яц}}{2U_{DC}T_{\mu}} = \frac{0,514}{2 \cdot 220 \cdot 0,001} = 1,168;$$

$$k_{ир.т.} = \frac{R_{яц}}{2U_{DC}T_{\mu}} = \frac{43,4}{2 \cdot 220 \cdot 0,001} = 98,6;$$

$$k_{др.с.} = \frac{k\Phi T_{я} T_{м}}{2U_{DC}T_{\mu}} = \frac{1,92 \cdot 0,0118 \cdot 0,0471}{2 \cdot 220 \cdot 0,001} = 0,00243;$$

$$k_{пр.с.} = \frac{k\Phi T_{м}}{2U_{DC}T_{\mu}} = \frac{1,92 \cdot 0,0471}{2 \cdot 220 \cdot 0,001} = 0,206;$$

$$k_{ир.с.} = \frac{k\Phi}{2U_{DC}T_{\mu}} = \frac{1,92}{2 \cdot 220 \cdot 0,001} = 4,364.$$

Динамическая модель электропривода в Matlab представлена на рисунке 2.

Блок Filter – фильтр, который удаляет из сигнала ошибки по угловой скорости якоря средне и высокочастотные составляющие для исключения колебательного режима работы электропривода. Частота фильтрации выбирается из диапазона 150-250 Гц.

Блоки Sat1 и Sat2 ограничивают сигналы управления на выходе регулятора скорости и регулятора тока соответственно. Нижний и верхний пределы ограничений блока Sat1 задают ограничения по току и равны соответственно «минус» $(2\div 4)I_{\text{ян}}$ и «плюс» $(2\div 4)I_{\text{ян}}$. Нижний и верхний пределы ограничений блока Sat2 соответствуют минимальному и максимальному значениям скважности γ широтно-импульсного преобразователя, которые равны «минус»1 и «плюс» 1 соответственно.

Настройки блоков Sat1 и Sat2 показаны на рисунке 3.

Часть модели, изображённая на рисунке 4 предназначена для задания момента нагрузки M_c и реализует систему уравнений:

$$M_c = \begin{cases} -M_c \cdot \text{sign}(\omega), & \text{если } \omega \neq 0, \\ -M_d, & \text{если } \omega = 0, \text{ и } |M_d| \leq M_c, \\ -M_c \cdot \text{sign}(\omega), & \text{если } \omega = 0, \text{ и } |M_d| > M_c, \end{cases}$$

где M_c – приведенный к якорю суммарный момент сопротивления вращению;
 M_d – электромагнитный момент ДПТ.

Такое представление момента сопротивления необходимо для получения результатов моделирования, адекватных реальным физическим процессам, протекающим в электроприводе. В противном случае при пуске двигателя при нулевых начальных условиях якорь начнёт вращаться в обратную сторону под действием момента сопротивления, который при пуске превышает электромагнитный момент двигателя.

Блоки PC и PT – регуляторы угловой скорости и тока якоря соответственно.

Блок SHIP – широтно-импульсный преобразователь напряжения.

Блоки W1(p) и W2(p) реализуют электромагнитные и электро-механические процессы в ДПТ.

Блок C реализует постоянная ДПТ.

Блок Check Step Response Characteristics необходим для поиска оптимальных настроек регуляторов.

Блок simout нужен для передачи ошибки регулирования угловой скорости якоря, её производной и интеграла в Workspace системы Matlab.

Для анализа процессов в разомкнутой системе управления электропривода, преобразуем модель на рисунке 2, убрав обратные связи по току и скорости и регуляторы тока и скорости. Полученная модель прямого пуска ДПТ изображена на рисунке 4. Блок Γ задает ступенчатое изменение скважности γ от 0 до 1.

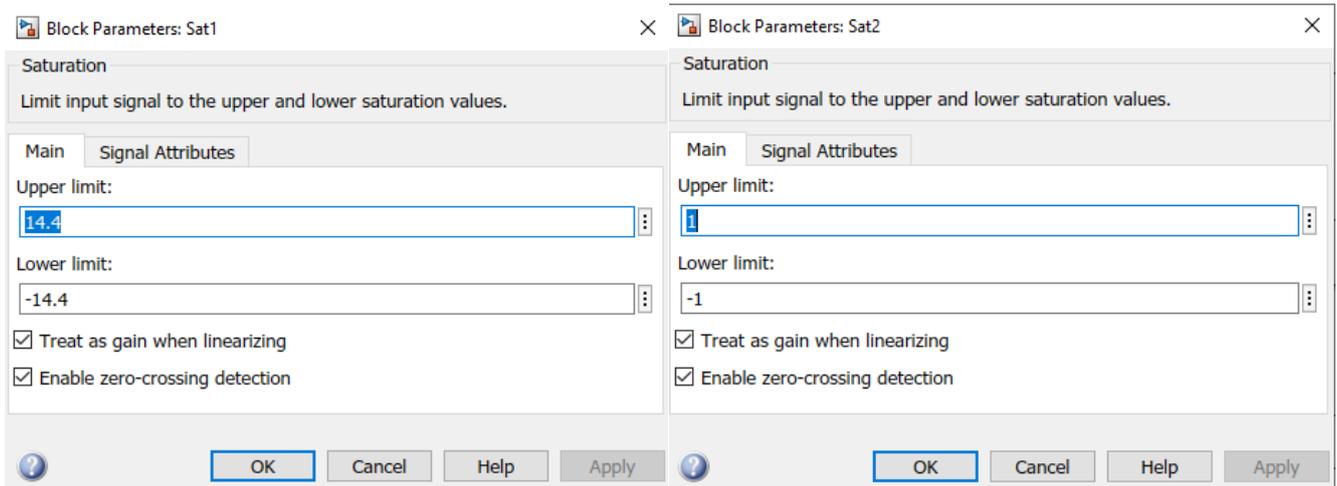


Рисунок 3 – Настройки блоков Sat1 и Sat2

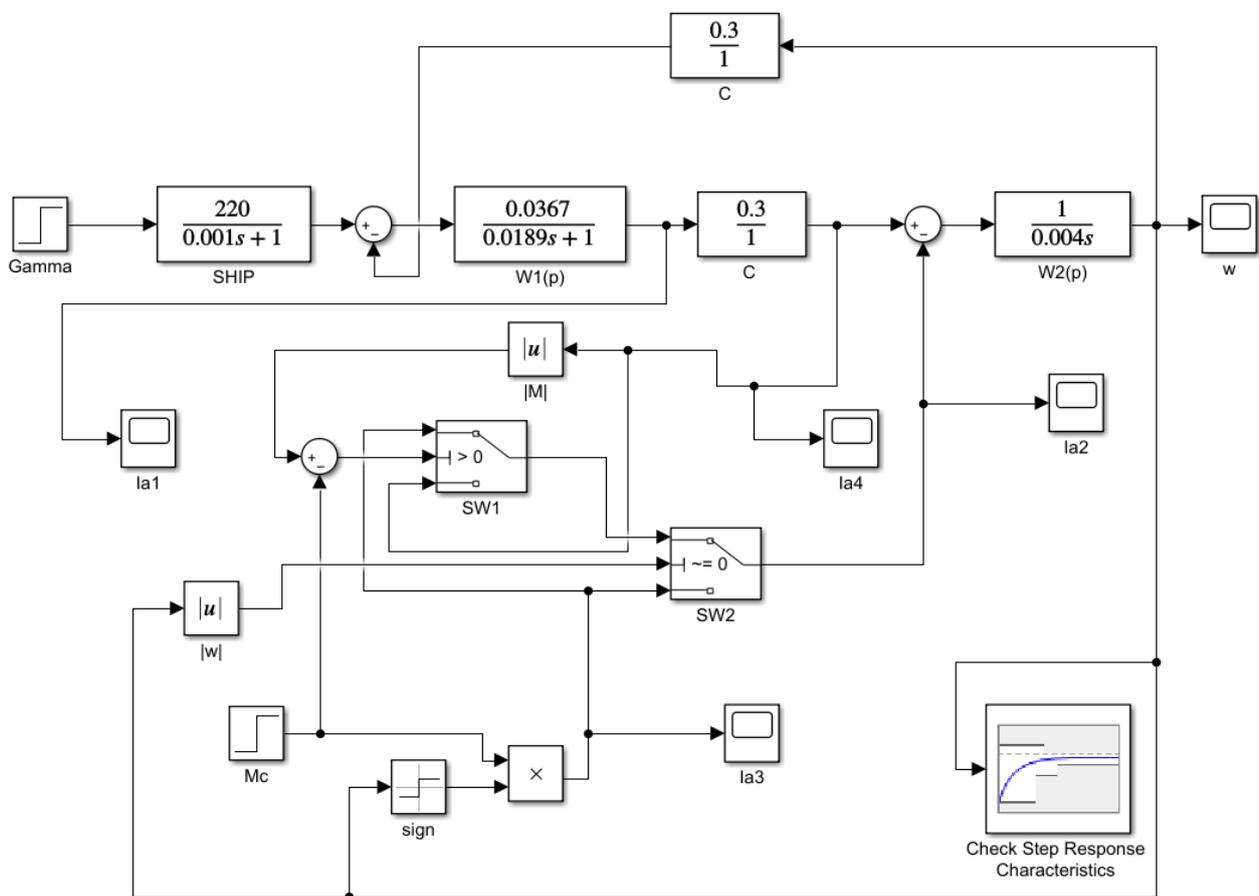


Рисунок 4 – Модель разомкнутой системы электропривода с ДПТ (прямого пуска ДПТ)

Результаты моделирования тока якоря и угловой скорости якоря представлены на рисунке 5.

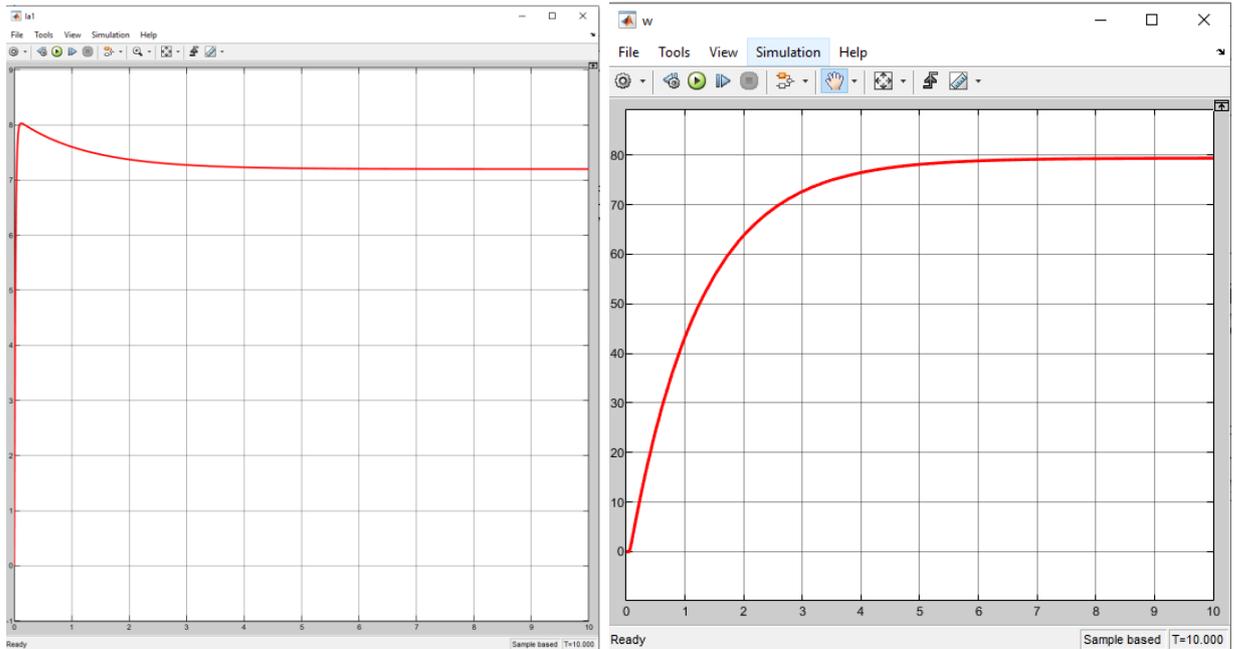


Рисунок 5 – Изменение тока якоря и угловой скорости якоря при прямом пуске ДПТ

Видно, что разгон до номинальной угловой скорости происходит примерно 5 с, а ток якоря при пуске превышает номинальное значение в 5 раз.

Попробуем улучшить пуск двигателя путем последовательной настройки двухконтурной замкнутой системы подчинённого регулирования с электропривода с ДПТ для этого преобразуем модель на рисунке 2 в модель, представленную на рисунке 4.

Если в качестве регулятора тока РТ на рисунке 4 использовать классический ПИД-регулятор с передаточной функцией

$$W_{РТ}(p) = k_{pi} + k_{di}p + \frac{k_{ii}}{p} = \frac{k_{di}p^2 + k_{pi}p + k_{ii}}{p},$$

где k_{pi} , k_{vi} , k_{ii} – коэффициенты пропорциональной, дифференциальной и интегральной составляющих ПИД-регулятора тока, то такой регулятор физически нереализуем, так как порядок полинома числителя в нём больше порядка полинома знаменателя. А для возможности физической реализации динамического звена необходимо хотя бы равенство порядков полинома числителя и знаменателя передаточной функции динамического звена. Поэтому одним из способов,

позволяющих реализовать ПИД-регулятор в системах управления, является способ добавления к его передаточной функции звена с передаточной функцией $W_N(p)$ вида

$$W_N(p) = \frac{1}{\frac{1}{N}p + 1} = \frac{1}{T_N p + 1},$$

которое не изменяет статический коэффициент усиления ПИД-регулятора и не искажает его низкочастотную часть логарифмической амплитудно-частотной характеристики, которая в основном и определяет динамические свойства регулятора. Обычно для электромеханических систем для выполнения последнего условия достаточно использовать в качестве N число, большее или равное 100.

Таким образом получим исправленную передаточную функцию ПИД-регулятора тока

$$W_{рти}(p) = W_{рт}(p) \cdot W_N(p) = \frac{k_{di}p^2 + k_{pi}p + k_{ii}}{p} \cdot \frac{1}{T_N p + 1} = \frac{k_{di}p^2 + k_{pi}p + k_{ii}}{p(T_N p + 1)}.$$

Зададим $N=1000$ ($T_N=0,001$) начальные настройки регулятора тока

$$k_{pi} = 0, k_{di} = 0, k_{ii} = 1,$$

и промоделируем изменение скорости двигателя при ступенчатом изменении тока якоря от 0 до номинального значения.

График изменения угловой скорости якоря при начальных настройках регулятора тока представлен на рисунке 6. Время разгона составляет примерно 5 секунд. Попробуем улучшить разгон, используя поисковый алгоритм в Matlab по минимуму динамической ошибки, реализуемой блоком Check Step Response Characteristics.

Для этого подадим на вход этого блока угловую скорость якоря и выполним настройку границ оптимального графика изменения угловой скорости якоря, задав соответствующие параметры, как показано на рисунке 7:

Step time (seconds) – начало переходного процесса в секундах;

Initial value – начальное значение целевой переменной (в данном случае – угловой скорости якоря);

Final value – конечное значение целевой переменной;

Rise time (seconds) – время, за которое целевая переменная достигнет нижней границы, задаваемое параметром %Rise;

% Rise – нижняя граница значения целевой переменной в процентах;

Settling time (seconds) – время регулирования (время, за которое значение целевой переменной войдёт в трубку регулирования и больше из неё не выйдет);

% Settling – величина трубки регулирования в процентах;

% Overshoot – перерегулирование в процентах;

% Undershoot – нижняя граница в процентах, меньше которой целевая переменная не может принимать значения.

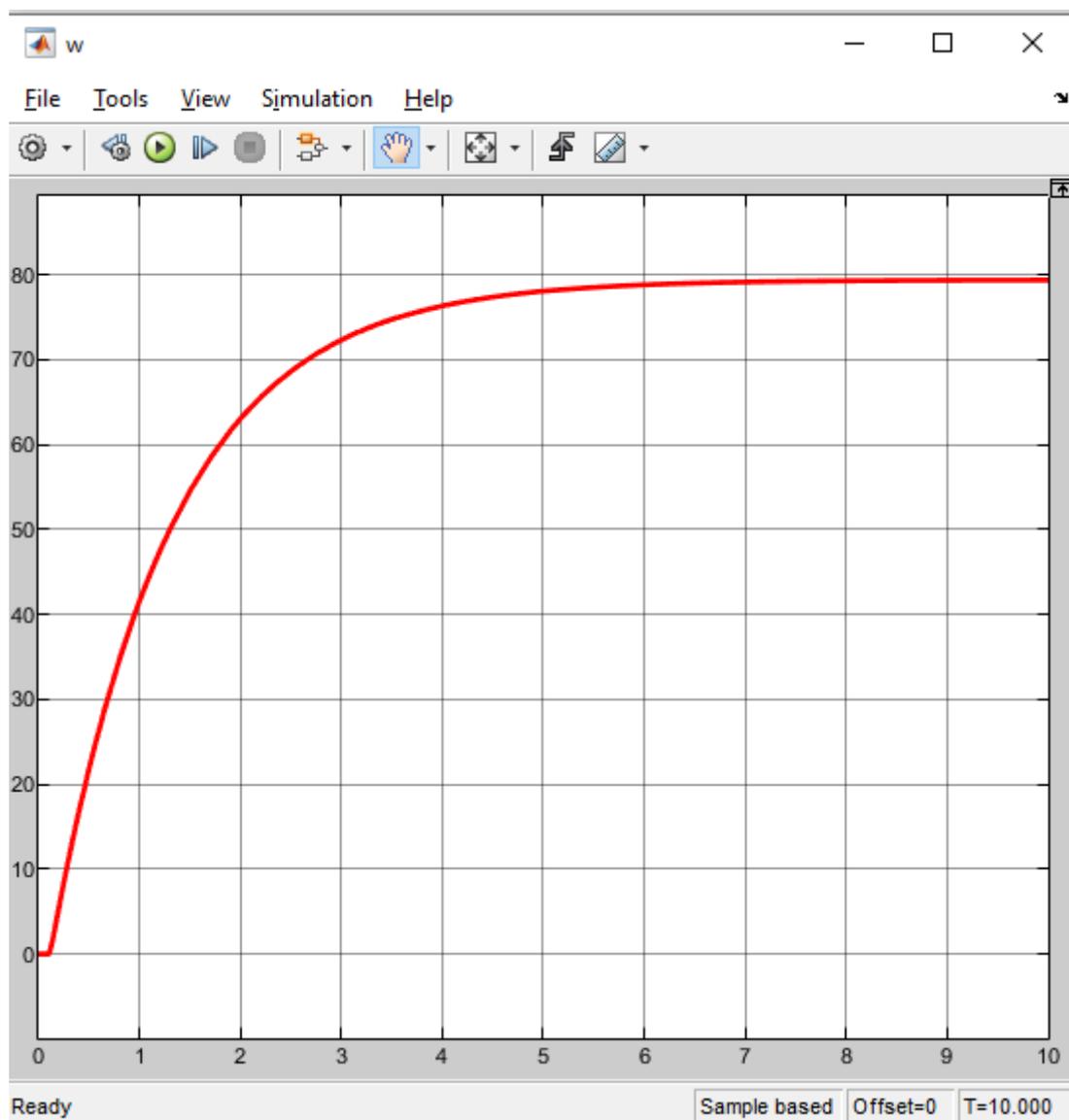


Рисунок 6 - График изменения угловой скорости якоря при начальных настройках регулятора тока

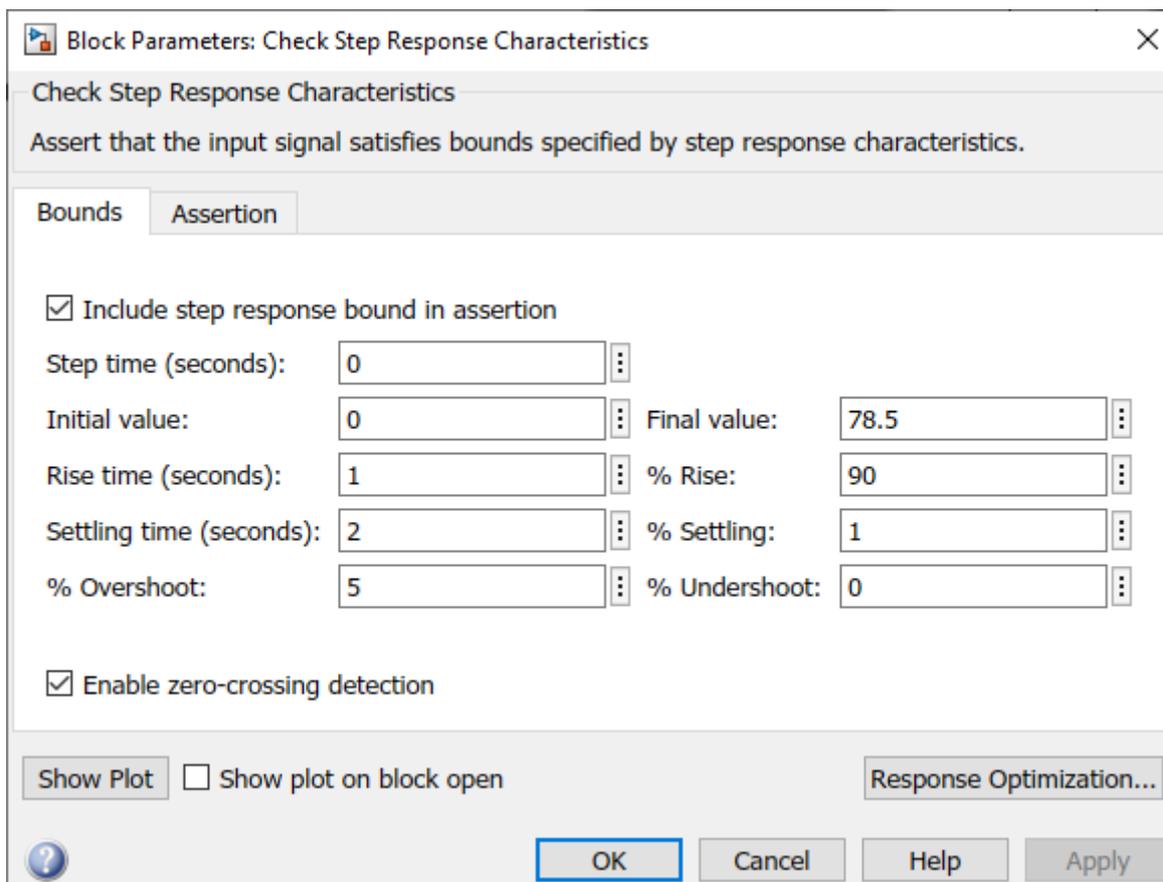


Рисунок 7 – Параметры блока Check Step Response Characteristics

После настройки параметров блока Check Step Response Characteristics, нужно нажать кнопку Response Optimization. Появится окно (рисунок 8), в котором отображаются границы для целевой переменной, в соответствии с ранее заданными параметрами блока Check Step Response Characteristics и для настройки параметров процесса оптимизации.

Для настройки параметров оптимизации выбираем значение параметра Design Variables Set: New, а затем в открывшемся окне Create Design Variables Set (рисунок 9) перетягиваем в левую пустую область параметры ПИД-регулятора тока k_{pi} , k_{di} , k_{ii} , которые будем искать в процессе оптимизации и указываем в качестве нижней границы интервала поиска значений 0 вместо «минус»Inf. После выбора оптимизируемых параметров и задания границ поиска закрываем окно Create

Design Variables Set, нажав кнопку Ok. Для запуска процесса оптимизации, нажимаем на зелёный треугольник под параметром Optimize в окне Response Optimization.

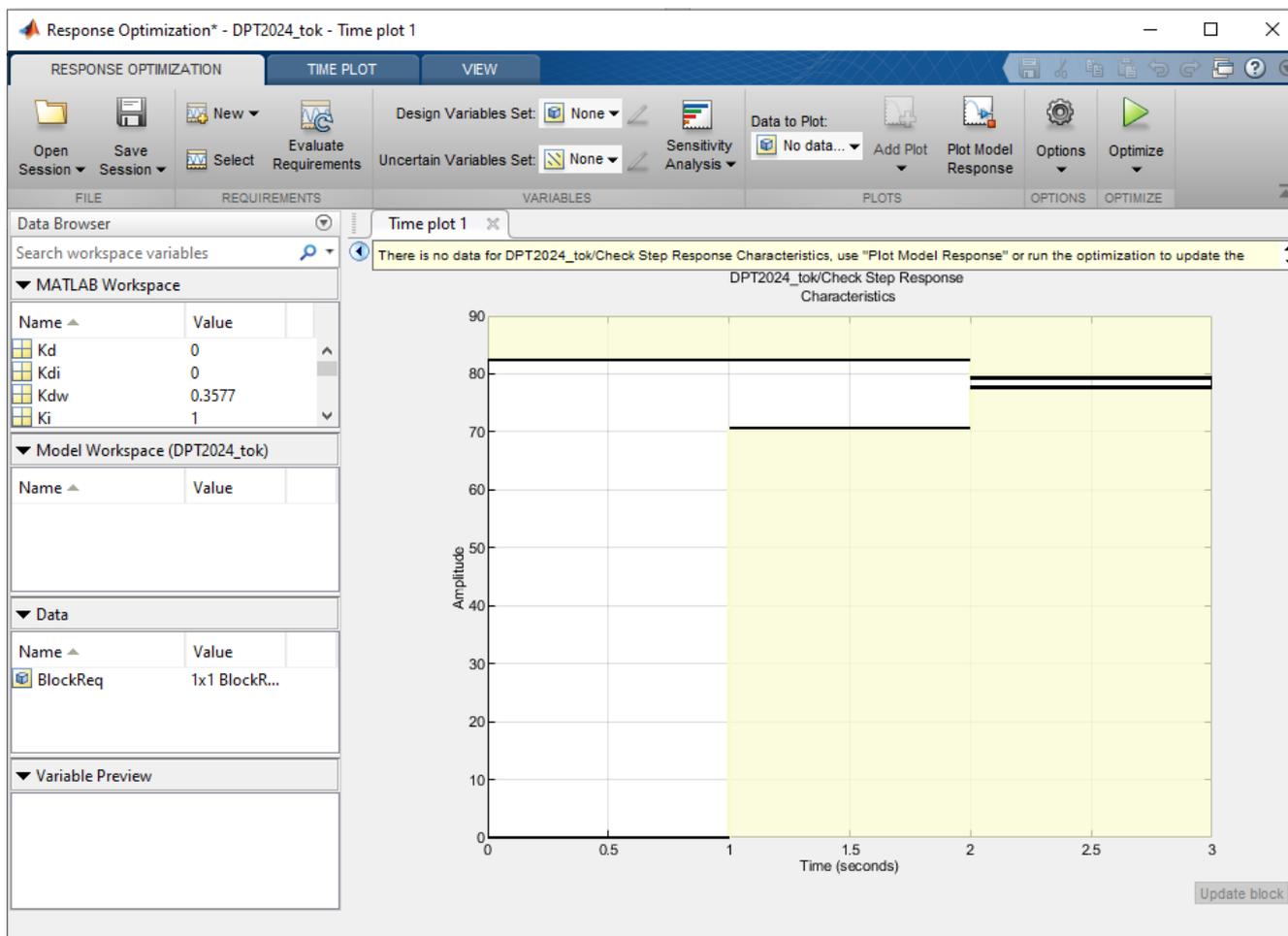


Рисунок 8 – Окно Response Optimization

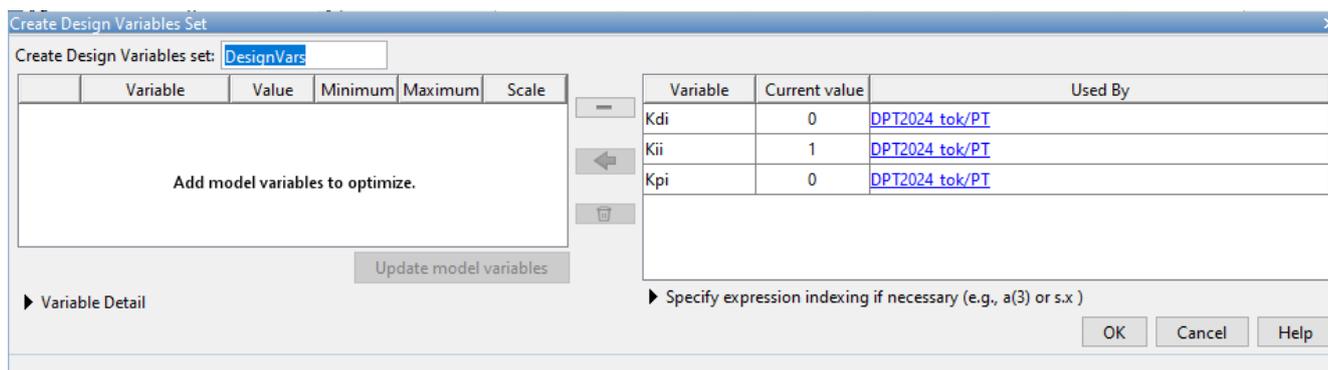


Рисунок 9 – Выбор оптимизируемых параметров ПИД-регулятора тока

В открывшемся окне Optimization Progress Report (рисунок 10) будут отображаться итерации процесса оптимизации параметров ПИД-регулятора, а в окне Response

Optimization в области границ изменения целевой переменной будут строиться графики изменения угловой скорости вращения якоря для найденных параметров ПИД-регулятора тока для текущей итерации (рисунок 11).

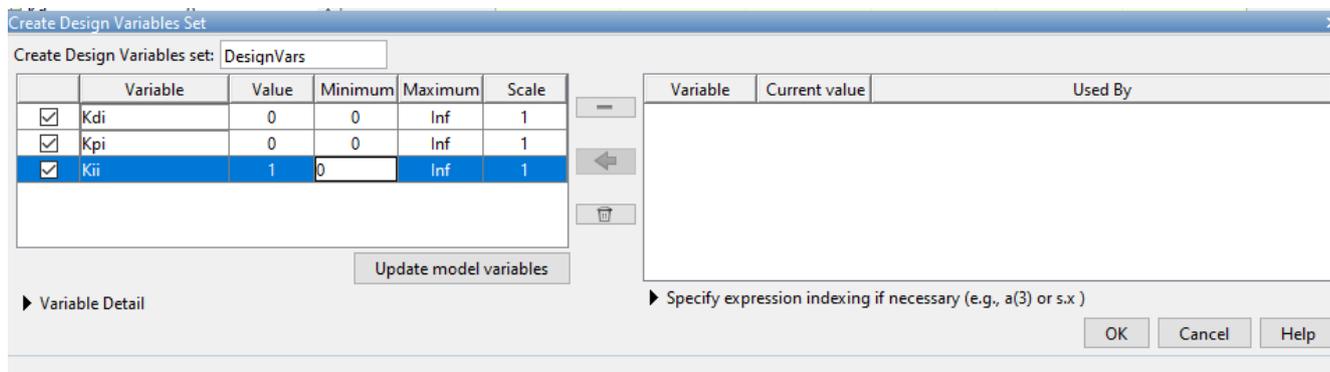


Рисунок 9 – Выбор границ интервала поиска оптимизируемых параметров ПИД-регулятора тока

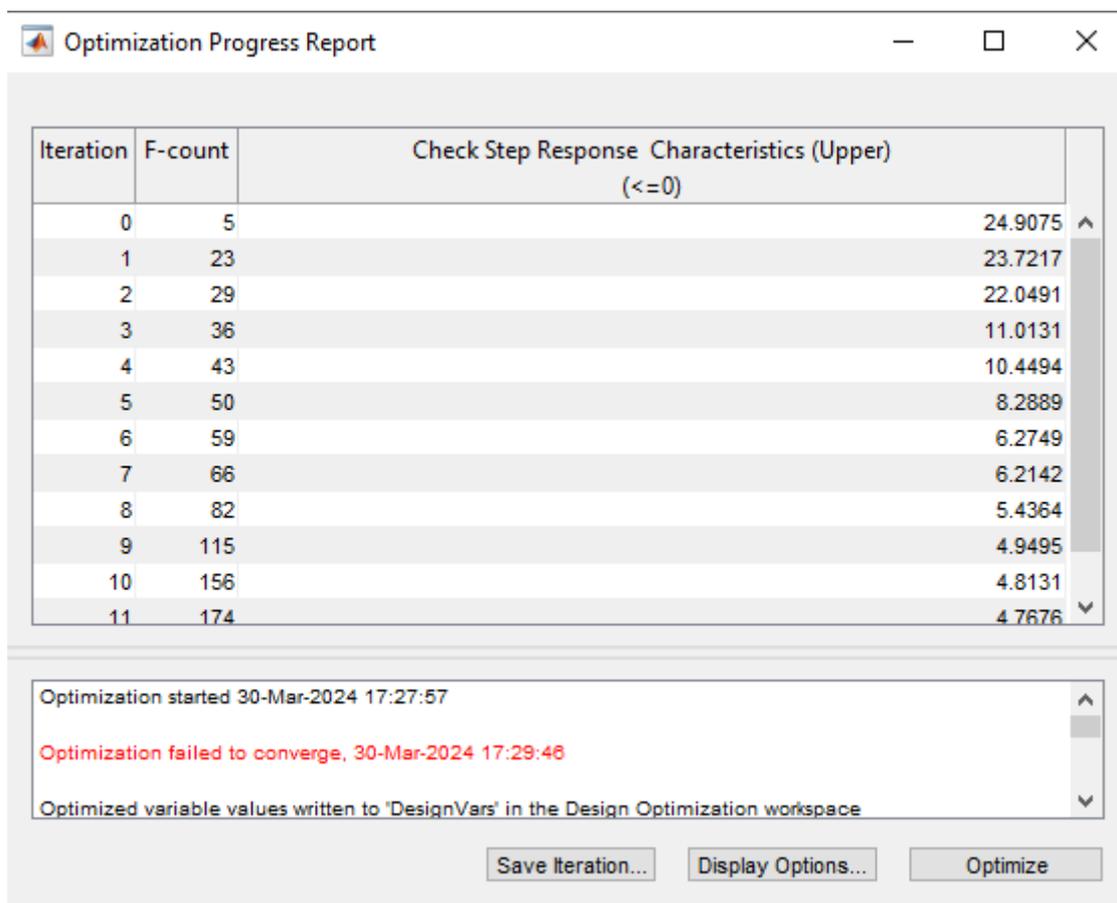


Рисунок 10 – Окно Optimization Progress Report

После окончания процесса оптимизации необходимо закрыть окна Optimization Progress Report и Response Optimization и запустить модель с найденными параметрами ПИД-регулятора тока (рисунок 12).

Найденные значения оптимальных настроек ПИД-регулятора тока можно в WorkSpace:

$$k_{pi} = 0,2618; k_{di} = 0,1437; k_{ii} = 2,0429.$$

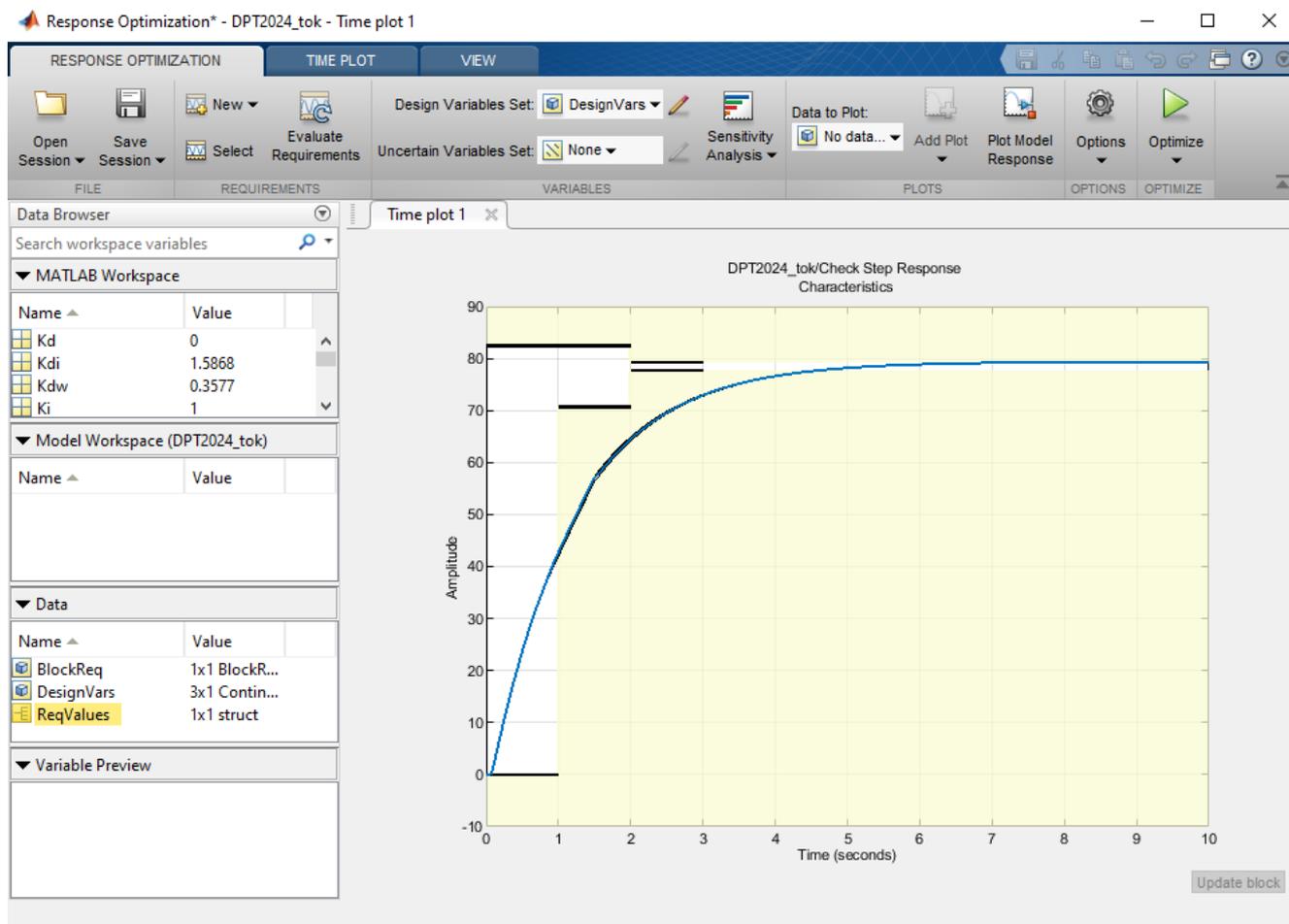


Рисунок 11 - Окно Response Optimization в процессе оптимизации

Далее выполним процедуру настройки на оптимум по минимуму динамической ошибки параметров ПИД-регулятора угловой скорости якоря. Для этого вернёмся к динамической модели электропривода, представленной на рисунке 2.

Зададим начальные настройки ПИД-регулятора скорости (блок РС на рисунке2):

$$k_{pw} = 0, k_{dw} = 0, k_{iw} = 1,$$

и промоделируем изменение скорости двигателя при ступенчатом изменении тока якоря от 0 до номинального значения.

График изменения угловой скорости якоря при начальных настройках ПИД-регулятора скорости представлен на рисунке 13. Время разгона составляет примерно 4 секунды. Попробуем улучшить разгон, используя поисковый алгоритм в Matlab по минимуму динамической ошибки, реализуемой блоком Check Step Response Characteristics, как это было выполнено для контура тока при настройке ПИД-регулятора тока ранее.

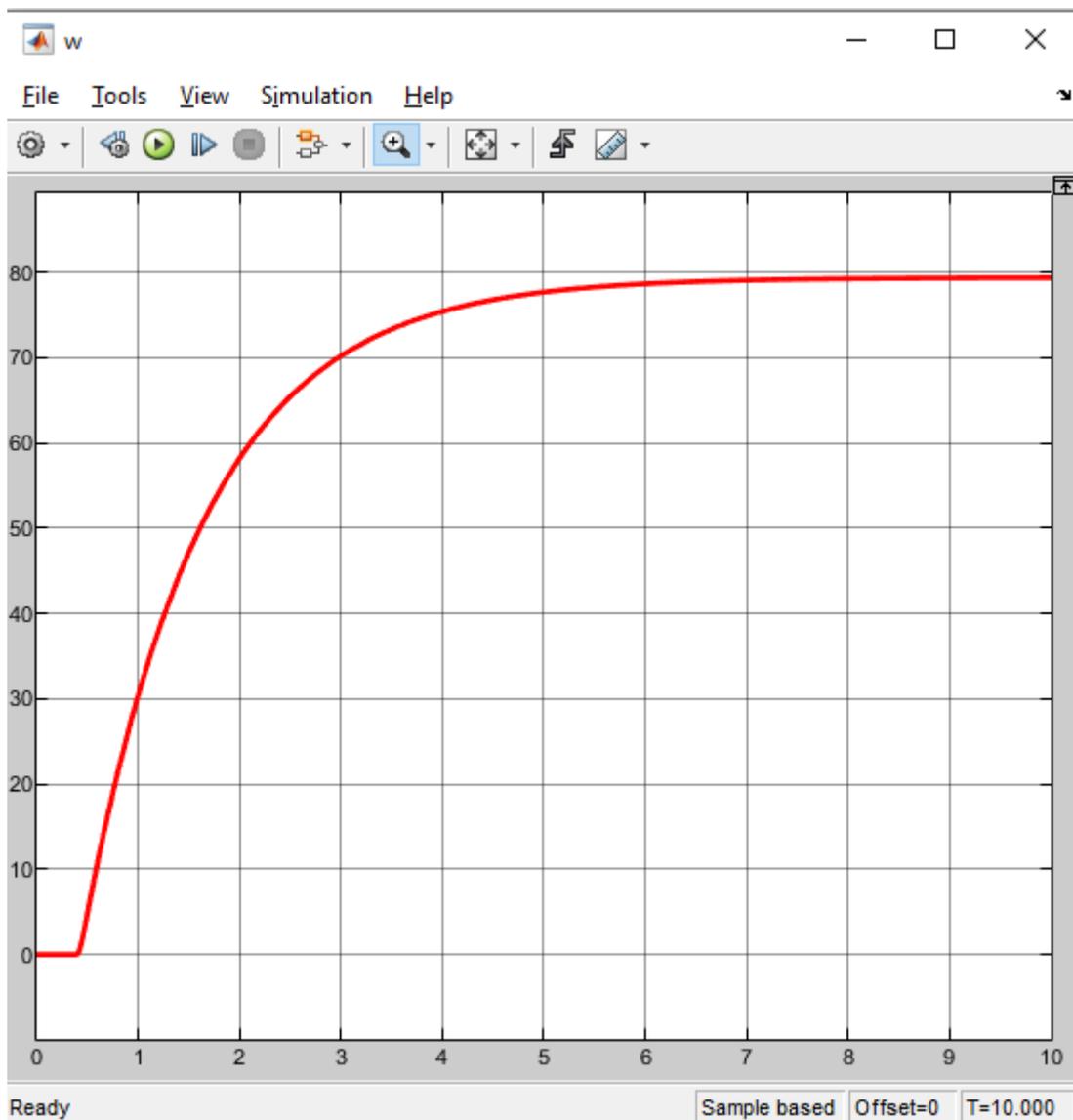


Рисунок 12 – Изменение угловой скорости якоря при пуске с оптимальными настройками ПИД-регулятора тока

На рисунках 14, 15, 15, 17, 18 показаны окна Check Step Response Characteristics, Response Optimization, Create Design Variables Set, Optimization Progress Report и Окно Response Optimization в процессе оптимизации. Целевое значение угловой скорости якоря (параметр Final Value блока Check Step Response Characteristics) был установлен на 50 рад/с.

Найденные значения оптимальных настроек ПИД-регулятора тока можно в Workspace:

$k_{pw} = 7,1699$; $k_{dw} = 0,3577$; $k_{iw} = 0,0097$.

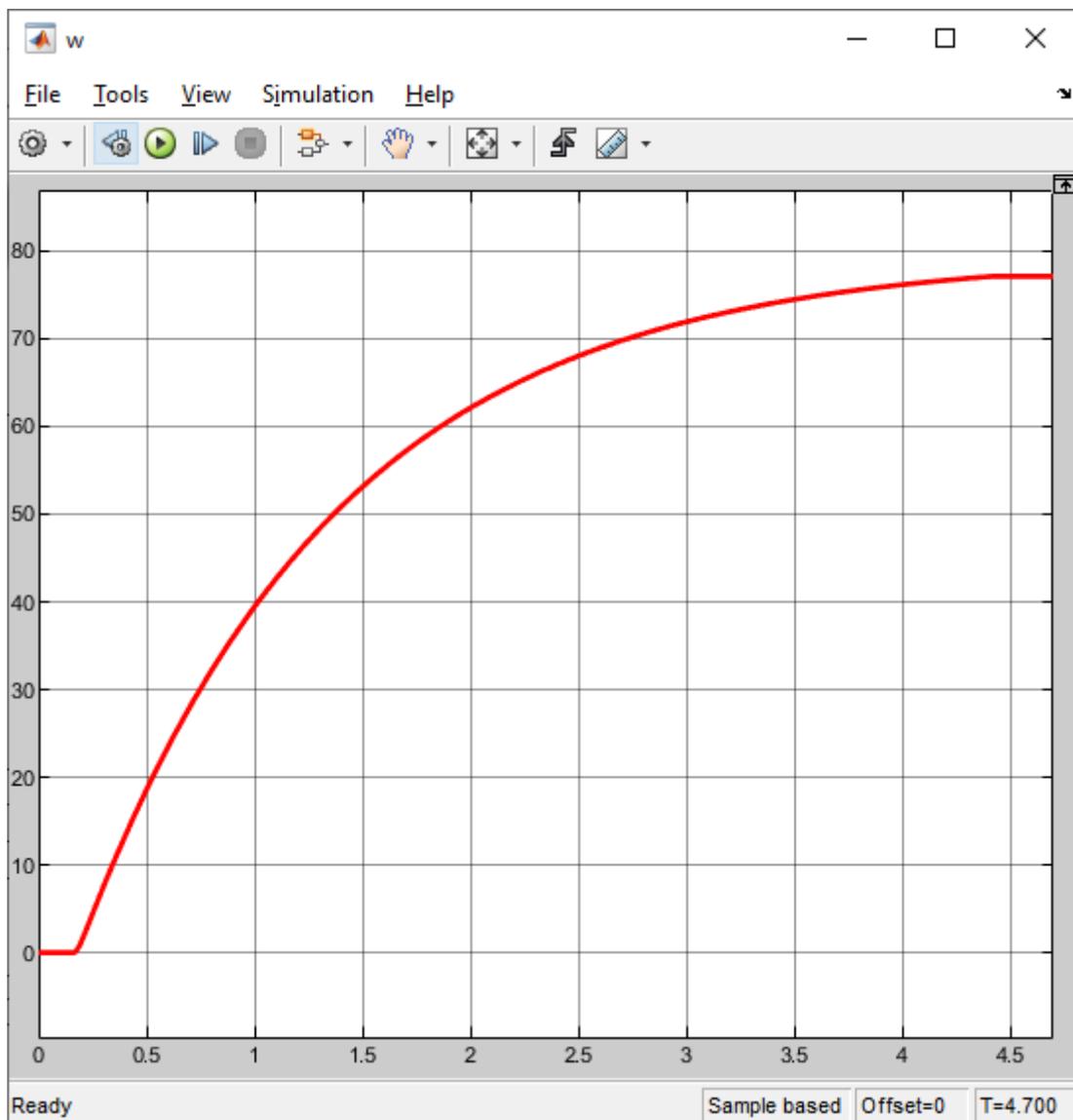


Рисунок 13 - График изменения угловой скорости якоря при начальных настройках ПИД-регулятора скорости

На рисунке 19 представлен график изменения угловой скорости якоря в двухконтурной системе подчиненного регулирования электропривода на базе ДПТ с оптимальными настройками ПИД-регуляторов тока и угловой скорости якоря. Как видно на рисунке 19, время регулирования составляет примерно 1,25 с, а перерегулирование отсутствует при разгоне ДПТ под номинальной нагрузкой от 0 до 50 рад/с. Т.е. в замкнутой системе процесс регулирования угловой скорости якоря ДПТ удалось улучшить по сравнению с разомкнутой системой.

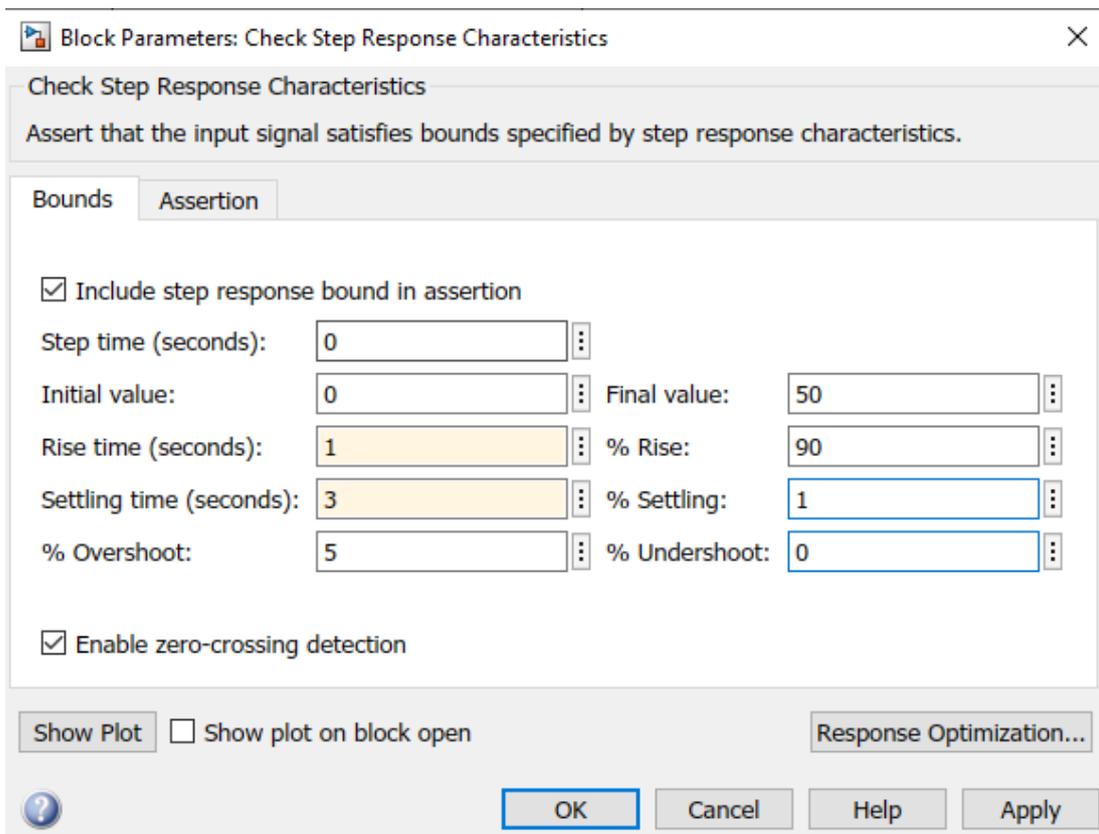


Рисунок 14 – Параметры блока Check Step Response Characteristics

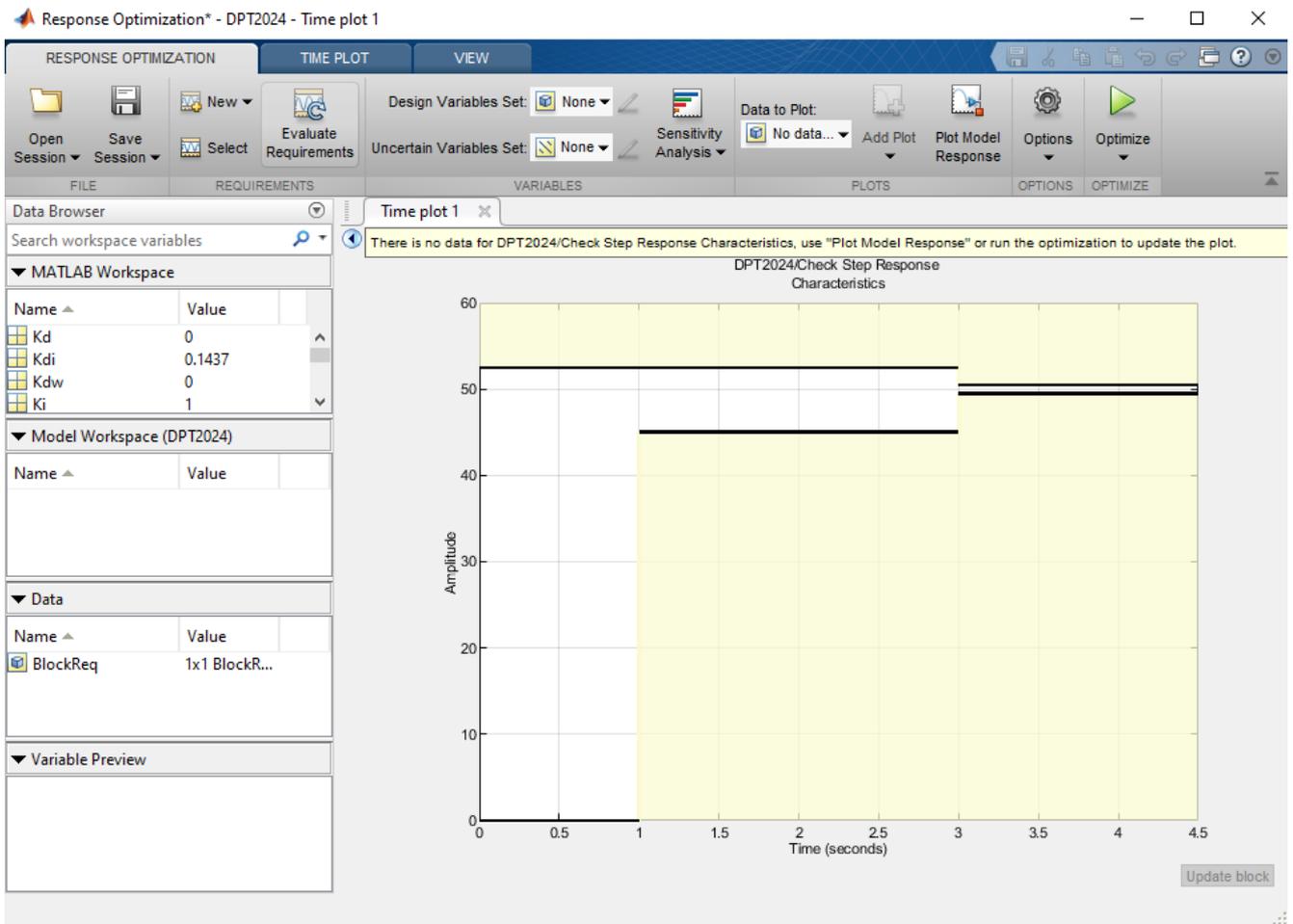


Рисунок 15 – Окно Response Optimization

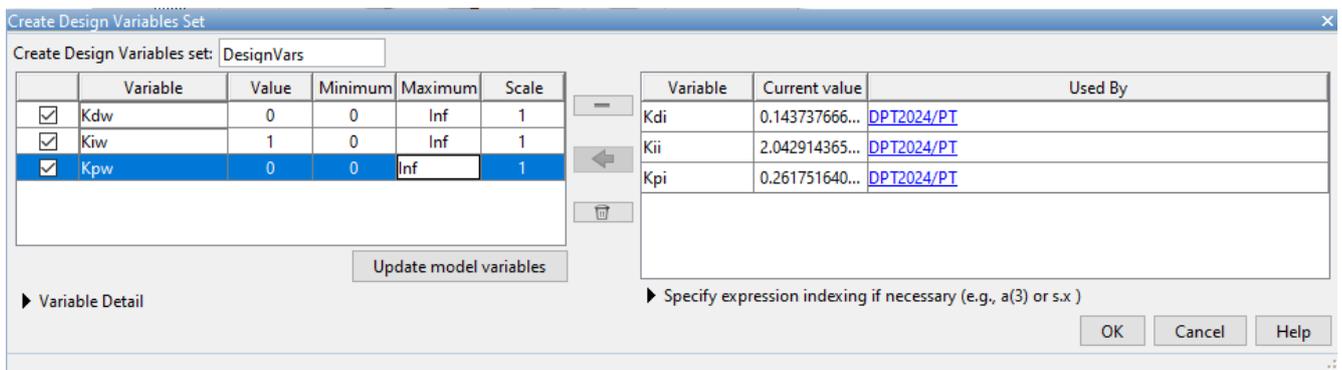


Рисунок 16 – Выбор оптимизируемых параметров ПИД-регулятора тока

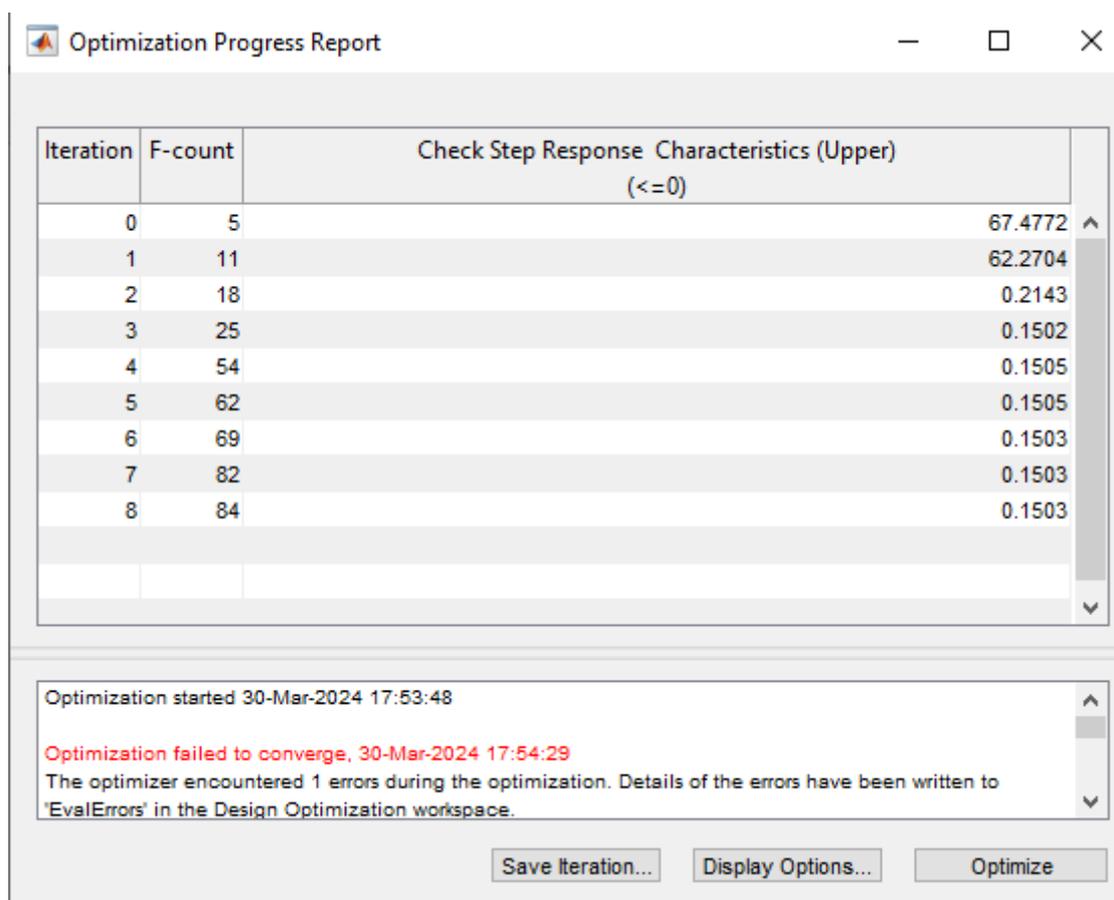


Рисунок 17 – Окно Optimization Progress Report

Для синтеза нечёткого регулятора скорости преобразуем динамическую модель на рисунке 2 в модель с нечётким регулятором, как показано на рисунке 20, добавив к модели блок Fuzzy Logic Controller и ключи SW1 и SW2 для переключения режимов работы модели:

- разомкнутого регулирования бех контура угловой скорости якоря (SW1 и SW2 в верхнем положении);
- замкнутого двухконтурного подчиненного регулирования с классическим ПИД-регулятором угловой скорости якоря (SW1 в нижнем, а SW2 в верхнем положении);
- замкнутого двухконтурного подчиненного регулирования с нечётким ПИД-регулятором угловой скорости якоря (SW1 в нижнем, а SW2 в верхнем положении);

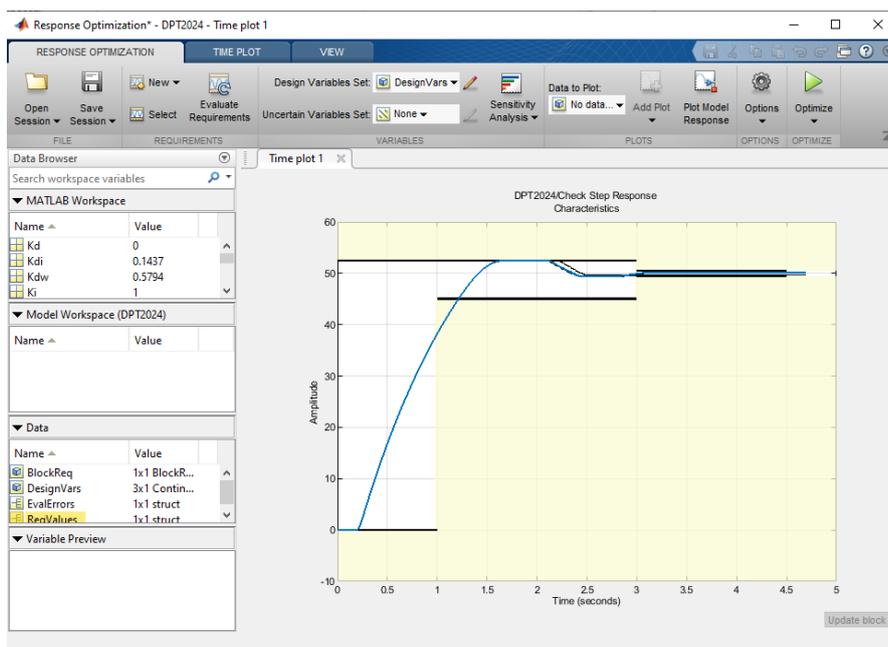


Рисунок 18 - Окно Response Optimization в процессе оптимизации

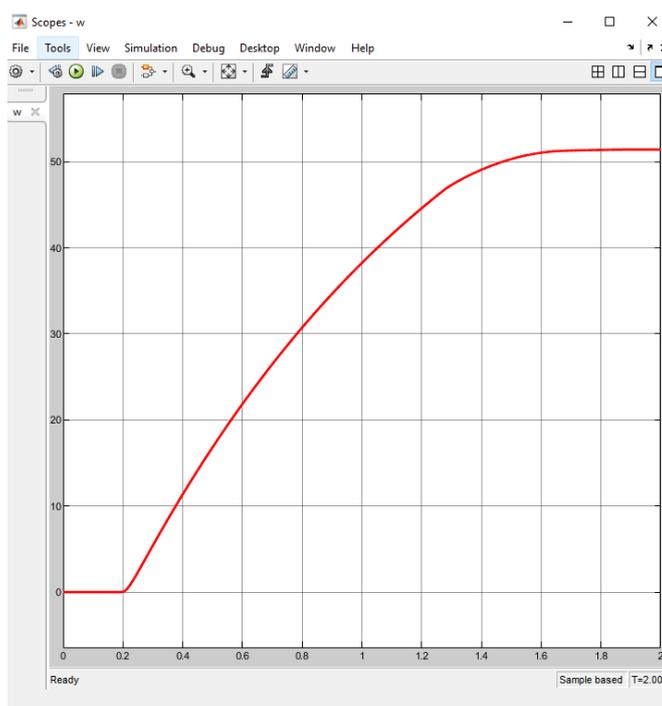


Рисунок 19 - График изменения угловой скорости якоря в двухконтурной системе подчиненного регулирования электропривода на базе ДПТ с оптимальными настройками ПИД-регуляторов тока и угловой скорости якоря

Для настройки нечёткого регулятора выведем график изменения ошибки угловой скорости якоря, ее производной и интеграла, а также сигнала управления с выхода ПИД-регулятора скорости (рисунок 21).

Для этого можно воспользоваться следующим скриптом в Matlab:

```
subplot(4, 1, 1);  
plot(simout.Time,simout.Data(:,1));  
subplot(4, 1, 2);  
plot(simout.Time,simout.Data(:,2));  
subplot(4, 1, 3);  
plot(simout.Time,simout.Data(:,3));  
subplot(4, 1, 4);  
plot(simout.Time,simout.Data(:,4));
```

Настройку нечёткого регулятора выполним с помощью мастера Fuzzy Logic Designer, который в среде Matlab запускается командой fuzzy (рисунок 22).

В мастере Fuzzy Logic Designer с помощью команды в меню Edit -> Add Variable -> Input создадим 3 входные переменные нечёткого регулятора:

Error – соответствует ошибке угловой скорости якоря;

VError – соответствует скорости изменения ошибки угловой скорости якоря;

SError – соответствует интегралу ошибки угловой скорости якоря.

Также в мастере Fuzzy Logic Designer с помощью команды в меню Edit -> Add Variable -> Output создадим одну выходную переменную нечёткого регулятора:

u - сигнала управления с выхода нечёткого регулятора скорости.

После указанных действий окно мастера Fuzzy Logic Designer должно выглядеть, как показано на рисунке 23.

Теперь произведём фаззификацию непрерывных переменных, изображённых на рисунке 21, преобразовав их в нечёткие переменные и внеся их значения в соответствующие входные и выходные переменные в мастере Fuzzy Logic Designer.

Выполним фаззификацию ошибки угловой скорости якоря и внесения значений в переменную Error. Для этого дважды мышью щёлкаем на входной

переменной Error в мастере Fuzzy Logic Designer (рисунок 23), откроется окно настройки нечёткой переменной Error (рисунок 24).

В поле Range вводим [-1.5 50], в поле Display Range также вводим [-1.5 50].

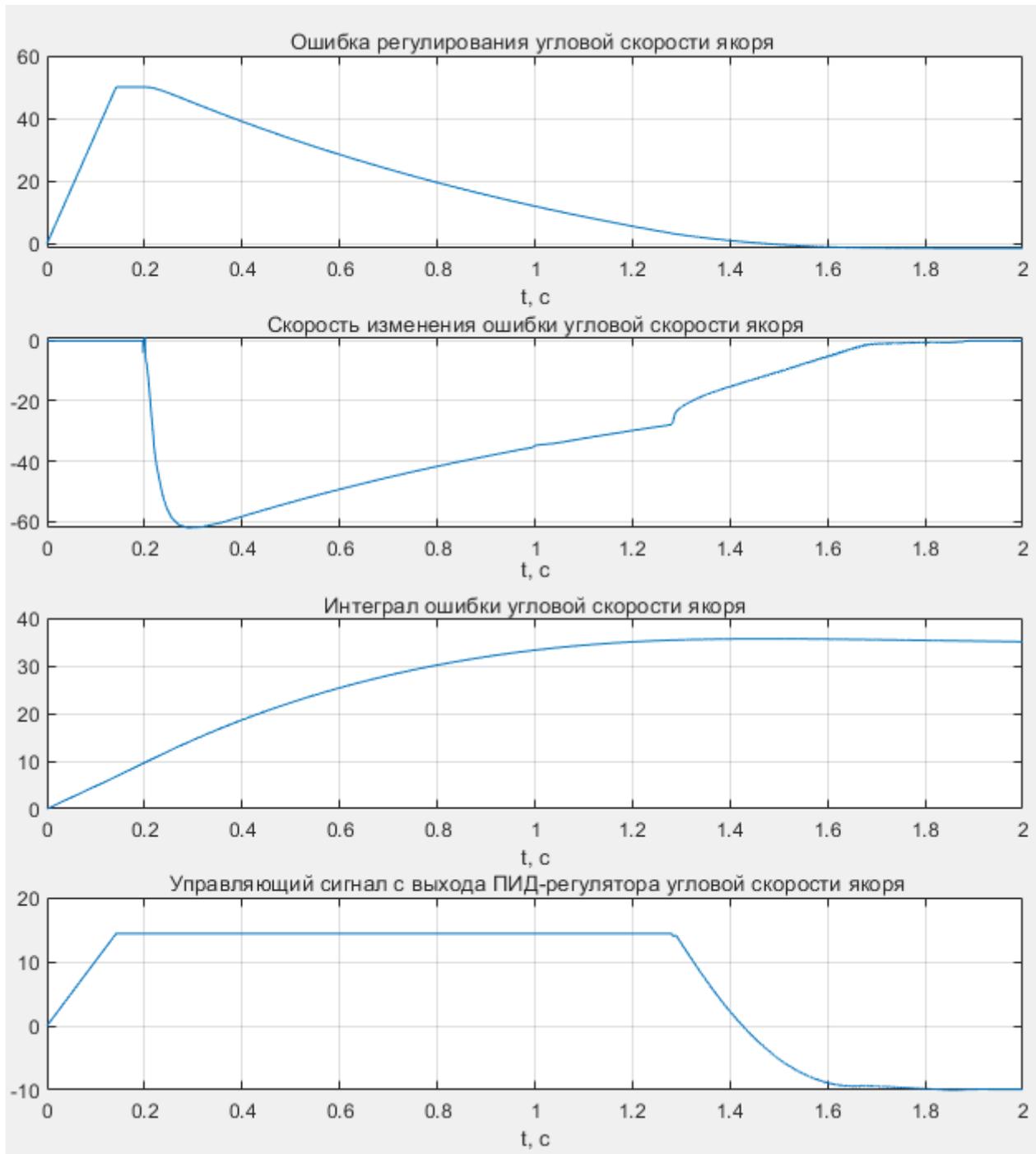


Рисунок 21 - Графики изменения ошибки угловой скорости якоря, ее производной, интеграла и сигнала управления с выхода ПИД-регулятора скорости

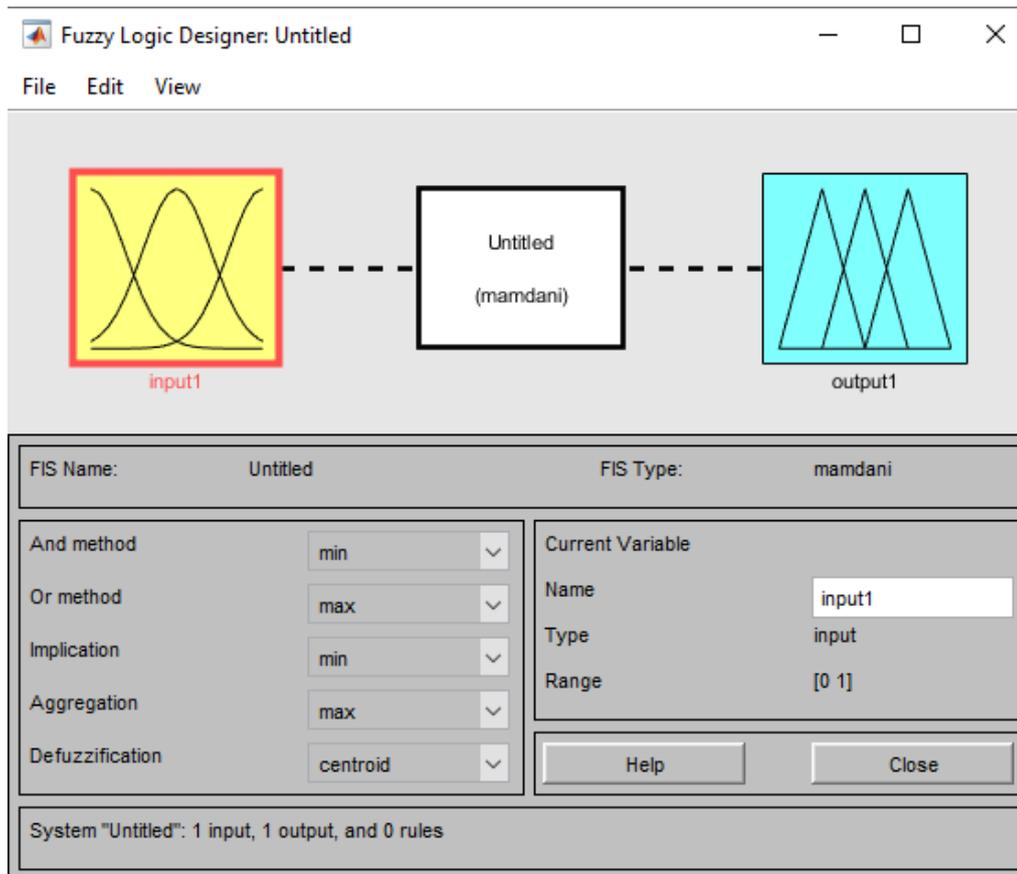


Рисунок 22 - Окно мастера Fuzzy Logic Designer

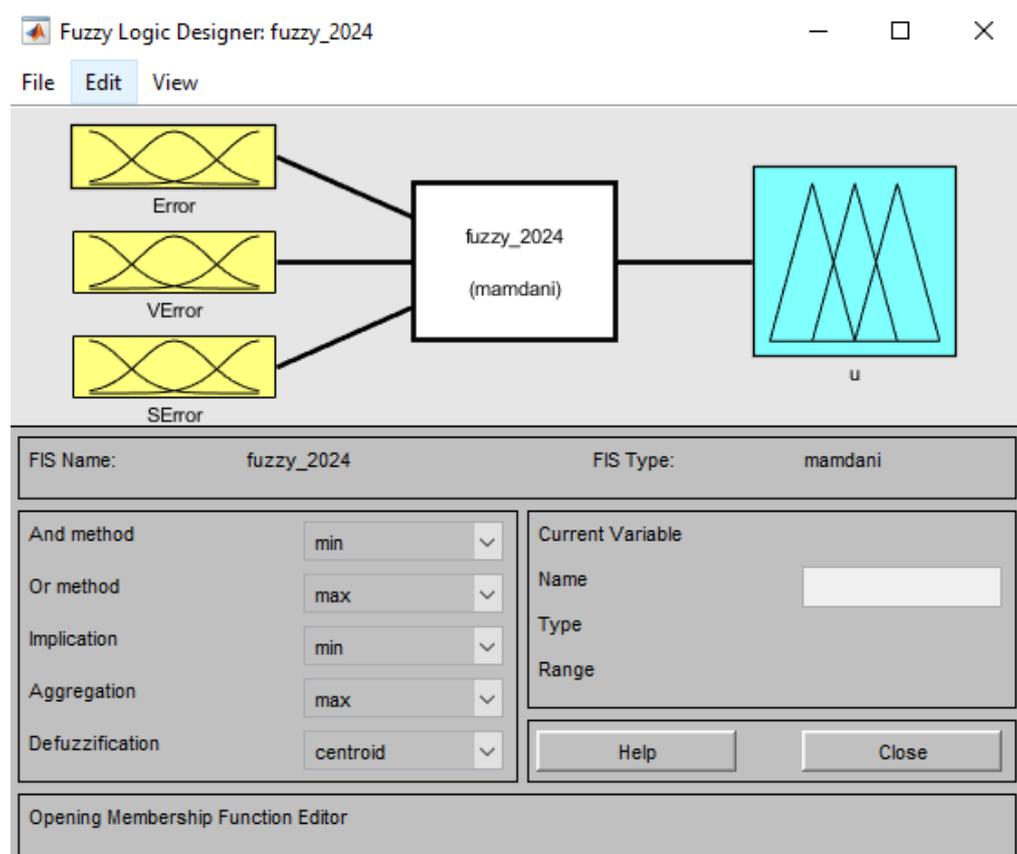


Рисунок 23 - Окно мастера Fuzzy Logic Designer после создания входных и выходных нечётких переменных

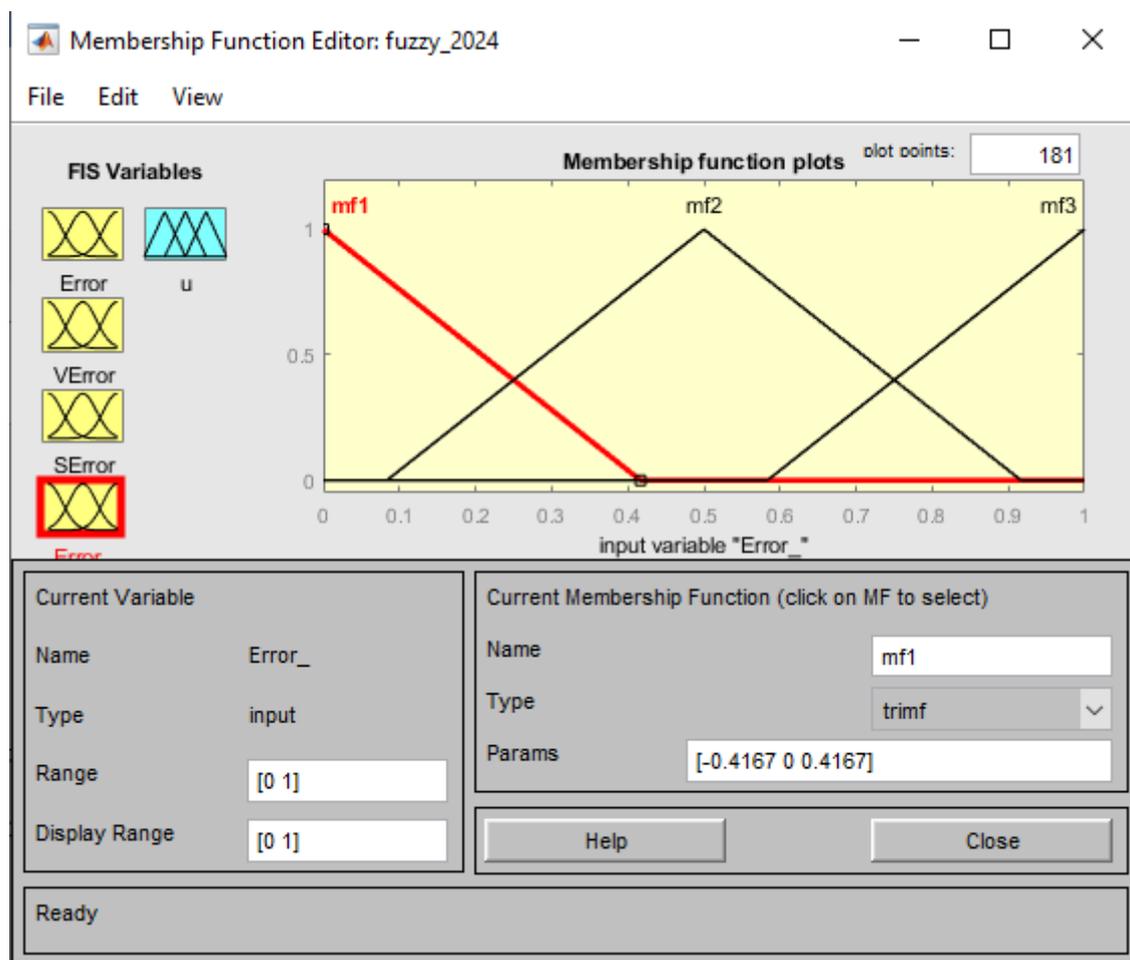


Рисунок 24 – Окно настройки значений нечёткой переменной Error

Вбираем линию mf1, чтобы она была красной. Для неё устанавливаем следующие значения параметров:

Name – NL;

Type – gbellmf;

Params – [3.744 2.63 -3.12] (определяются по графику ошибки на рисунке 21).

Для линии mf2 устанавливаем следующие значения параметров:

Name – Z;

Type – gbellmf;

Params – [1.29 2.5 1.598] (в соответствии с графиком ошибки на рисунке 21).

Для линии mf3 устанавливаем следующие значения параметров:

Name – PL;

Type – gbellmf;

Params – [45.93 50.4 48.7] (в соответствии с графиком ошибки на рисунке 21).

В результате получим нечёткую переменную ошибки угловой скорости якоря, как показано на рисунке 25.

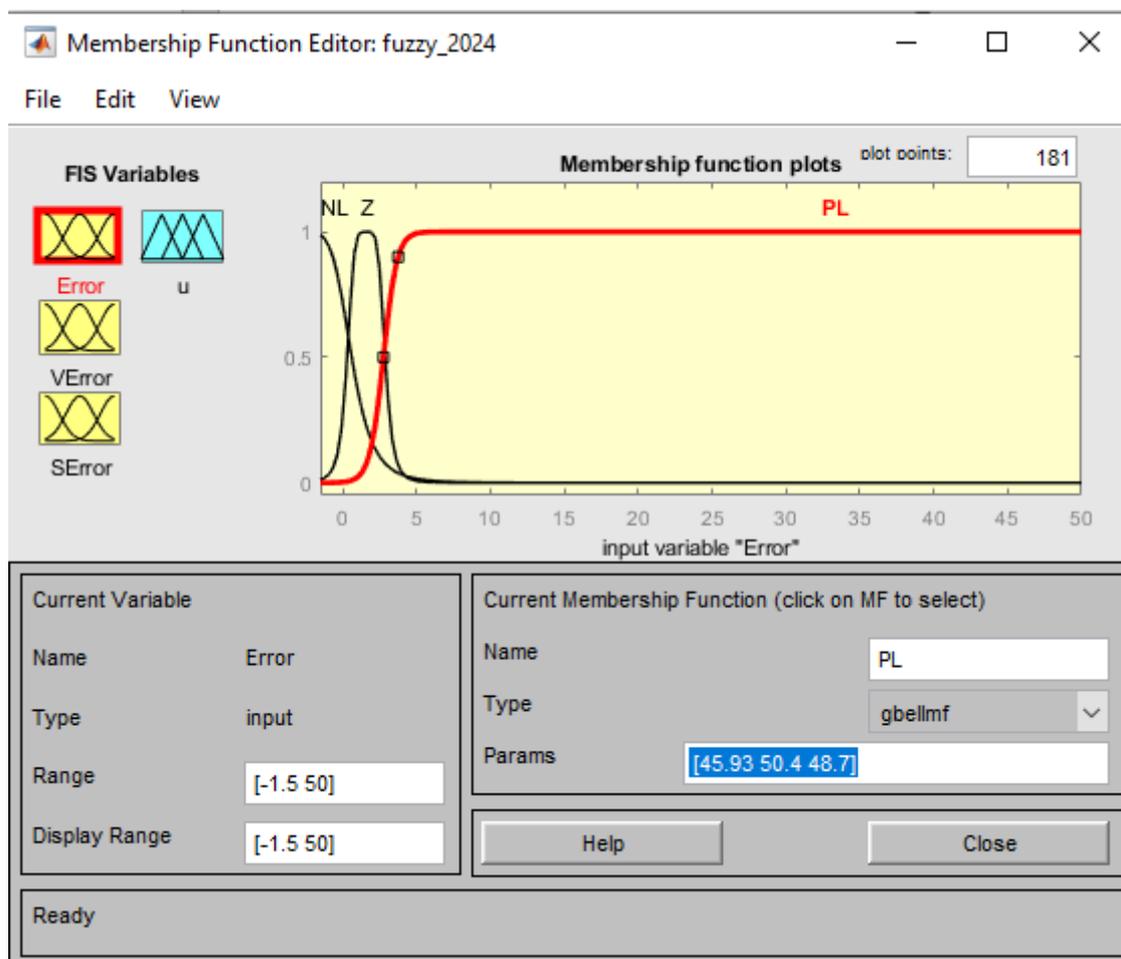


Рисунок 25 -Нечёткая переменная ошибки угловой скорости якоря – Error

Нажимаем кнопку Close и в мастере Fuzzy Logic Designer переходим к следующей входной переменной VError, соответствующей скорости изменения ошибки угловой скорости якоря.

В поле Range вводим [-22.5 25], в поле Display Range также вводим [-22.5 25].

Для линии mf1 устанавливаем следующие значения параметров:

Name – NLV;

Type – gbellmf;

Params – [6.41 3.63 -22.63] (в соответствии с графиком скорости изменения ошибки на рисунке 21).

Для линии mf2 устанавливаем следующие значения параметров:

Name – ZV;

Type – gbellmf;

Params – [4.28 3.91 -10.42] (в соответствии с графиком скорости изменения ошибки на рисунке 21).

Для линии mf3 устанавливаем следующие значения параметров:

Name – PLV;

Type – gbellmf;

Params – [19.3 10.17 13.3] (в соответствии с графиком скорости изменения ошибки на рисунке 21).

В результате получим нечёткую переменную скорости изменения ошибки угловой скорости якоря, как показано на рисунке 26.

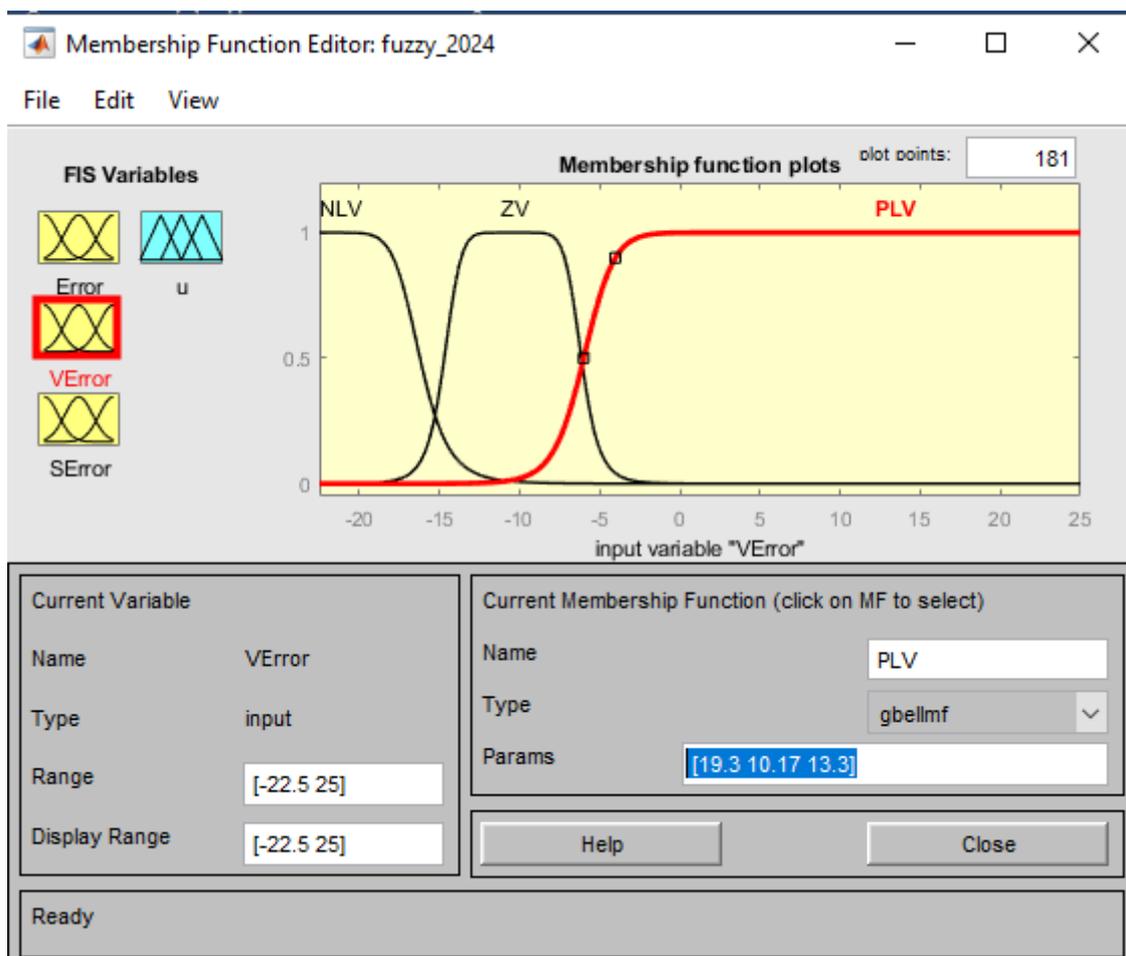


Рисунок 26 – Нечёткая переменная скорости изменения ошибки угловой скорости якоря – VError

Нажимаем кнопку Close и в мастере Fuzzy Logic Designer переходим к следующей входной переменной SError, соответствующей интегралу ошибки угловой скорости якоря.

В поле Range вводим [0 35.7], в поле Display Range также вводим [0 35.7].

Для линии mf1 устанавливаем следующие значения параметров:

Name – NLS;

Type – gbellmf;

Params – [25.68 43.8 8.19] (в соответствии с графиком интеграла ошибки на рисунке 21).

Для линии mf2 устанавливаем следующие значения параметров:

Name – ZS;

Type – gbellmf;

Params – [0.861 2.48 34.17] (в соответствии с графиком интеграла ошибки на рисунке 21).

Для линии mf3 устанавливаем следующие значения параметров:

Name – PLS;

Type – gbellmf;

Params – [1.307 1.82 35.7] (в соответствии с графиком интеграла ошибки на рисунке 21).

В результате получим нечёткую переменную интеграла ошибки угловой скорости якоря, как показано на рисунке 27.

Нажимаем кнопку Close и в мастере Fuzzy Logic Designer переходим к следующей выходной переменной u , соответствующей сигналу управления с выхода нечёткого регулятора скорости.

В меню выбираем команду Edit -> Remove All MFs (удаляем все функции принадлежности нечёткой переменной, которые были по умолчанию). Затем в меню выбираем команду Edit -> Add MFs. В открывшемся окне Membership Functions устанавливаем значение параметра MF type – gbellmf, а значение параметра Number of MFs – 5. Нажимаем кнопку ОК. В результате получим начальное значение переменной u , как показано на рисунке 29.

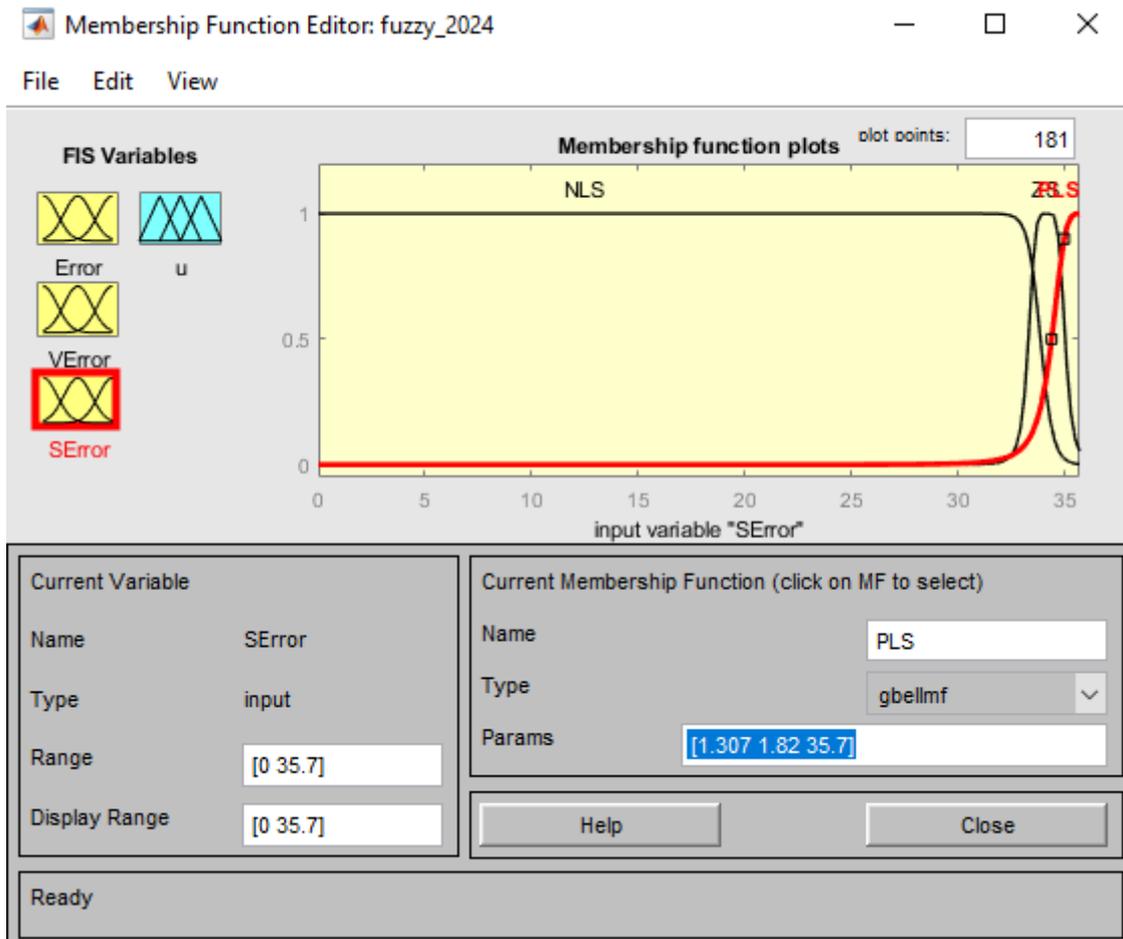


Рисунок 27 – Нечёткая переменная интеграла ошибки угловой скорости якоря – SError

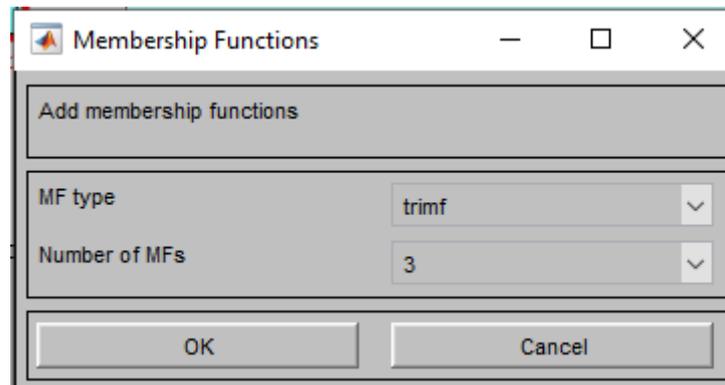


Рисунок 28 – Окно Membership Functions

Далее настраиваем параметры функций принадлежности переменной u .

В поле Range вводим $[-14.4 \ 14.4]$, в поле Display Range также вводим $[-14.4 \ 14.4]$.

Для линии mf1 устанавливаем следующие значения параметров:

Name – NLU;

Type – gbellmf;

Params – [4.619 3.5 -14.9] (в соответствии с графиком сигнала управления с выхода регулятора скорости на рисунке 21).

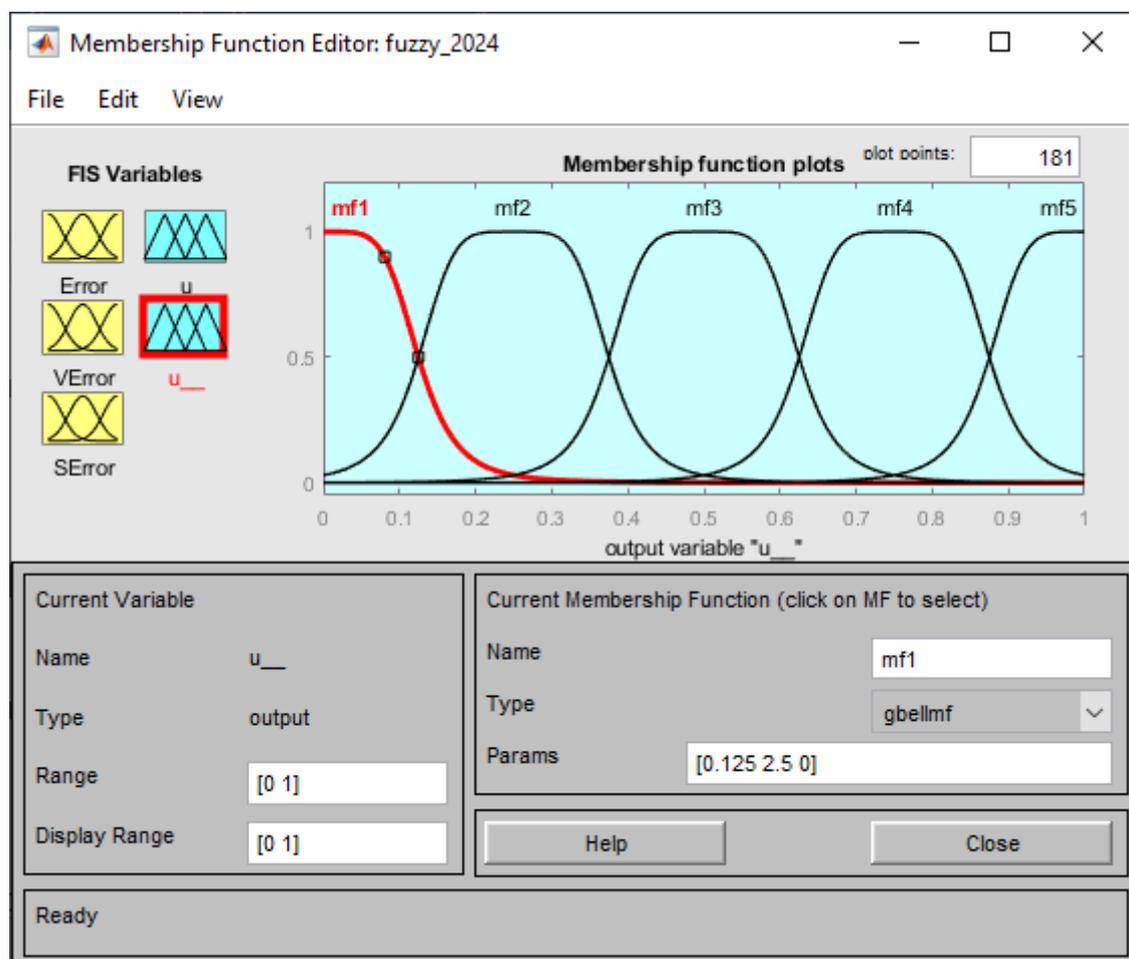


Рисунок 29 – Начальные значения нечёткой переменной u

Для линии mf2 устанавливаем следующие значения параметров:

Name – NMU;

Type – gbellmf;

Params – [4.091 5.01 -7.2] (в соответствии с графиком сигнала управления с выхода регулятора скорости на рисунке 21).

Для линии mf3 устанавливаем следующие значения параметров:

Name – ZU;

Type – gbellmf;

Params – [4.46 5.58 0.644] (в соответствии с графиком сигнала управления с выхода регулятора скорости на рисунке 21).

Для линии mf4 устанавливаем следующие значения параметров:

Name – PMU;

Type – gbellmf;

Params – [3.762 5.49 8.98] (в соответствии с графиком сигнала управления с выхода регулятора скорости на рисунке 21).

Для линии mf5 устанавливаем следующие значения параметров:

Name – PLU;

Type – gbellmf;

Params – [1.65 4.593 14.4] (в соответствии с графиком сигнала управления с выхода регулятора скорости на рисунке 21).

В результате получим нечёткую переменную интеграла ошибки угловой скорости якоря, как показано на рисунке 30.

Нажимаем кнопку Close и в мастере Fuzzy Logic Designer выбираем команду Edit -> Rules..., после чего откроется окно Rule Editor мастера настройки правил нечёткого вывода (рисунок 30). В мастере создаем следующие правила:

1. If (Error is PL) or (SError is NLS) then (u is PLU) (1)
2. If (Error is PL) or (SError is ZS) then (u is PLU) (1)
3. If (Error is NL) and (VError is ZV) and (SError is PLS) then (u is NMU) (1)
4. If (Error is NL) and (VError is NLV) and (SError is PLS) then (u is NLU) (1)
5. If (Error is NL) and (VError is PLV) and (SError is PLS) then (u is NLU) (1)
6. If (Error is Z) and (VError is ZV) and (SError is PLS) then (u is ZU) (1)
7. If (Error is Z) and (VError is NLV) and (SError is PLS) then (u is PMU) (1)
8. If (Error is Z) and (VError is PLV) and (SError is PLS) then (u is PLU) (1)

Правила составляются в соответствии с рисунком 21.

После завершения создания правил нечёткого вывода, закрываем окно мастера Rule Editor, нажав на кнопку Close.

Теперь в окне мастера Fuzzy Logic Designer можно командой View -> Surface вывести график нечёткой переменной u в функции любых двух нечётких входных

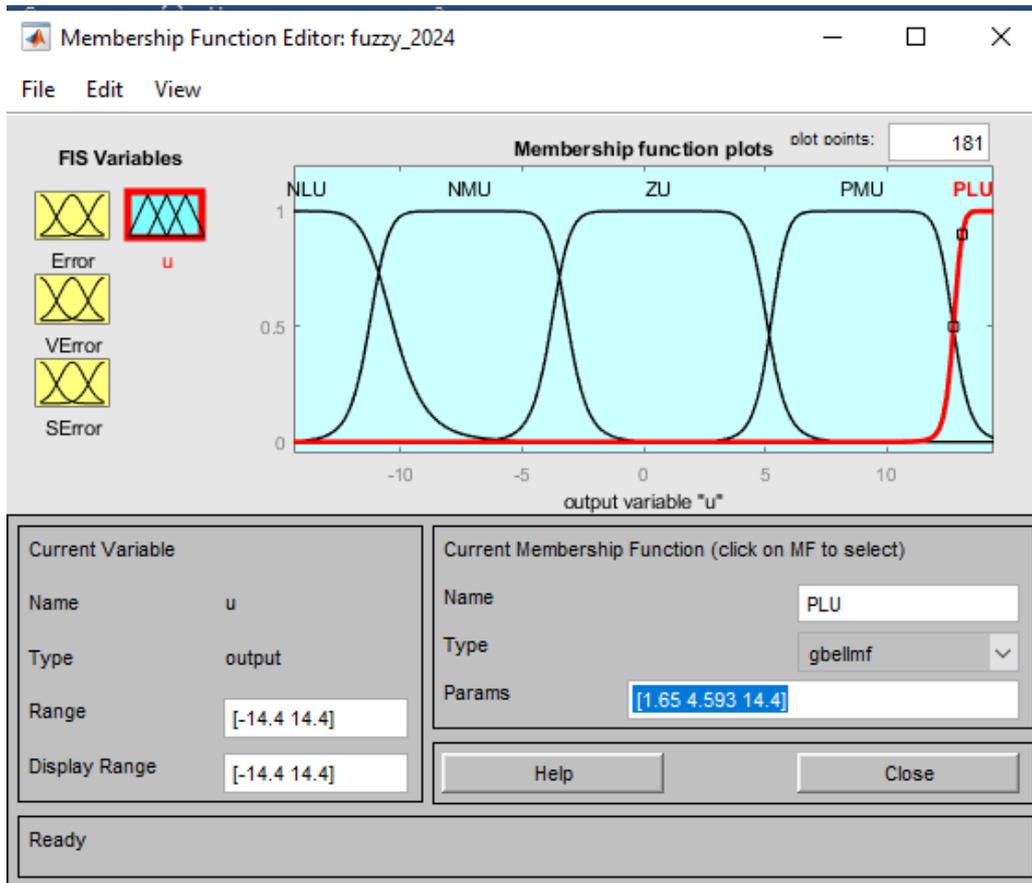


Рисунок 29 – Нечёткая переменная сигнала управления с выхода регулятора скорости – u

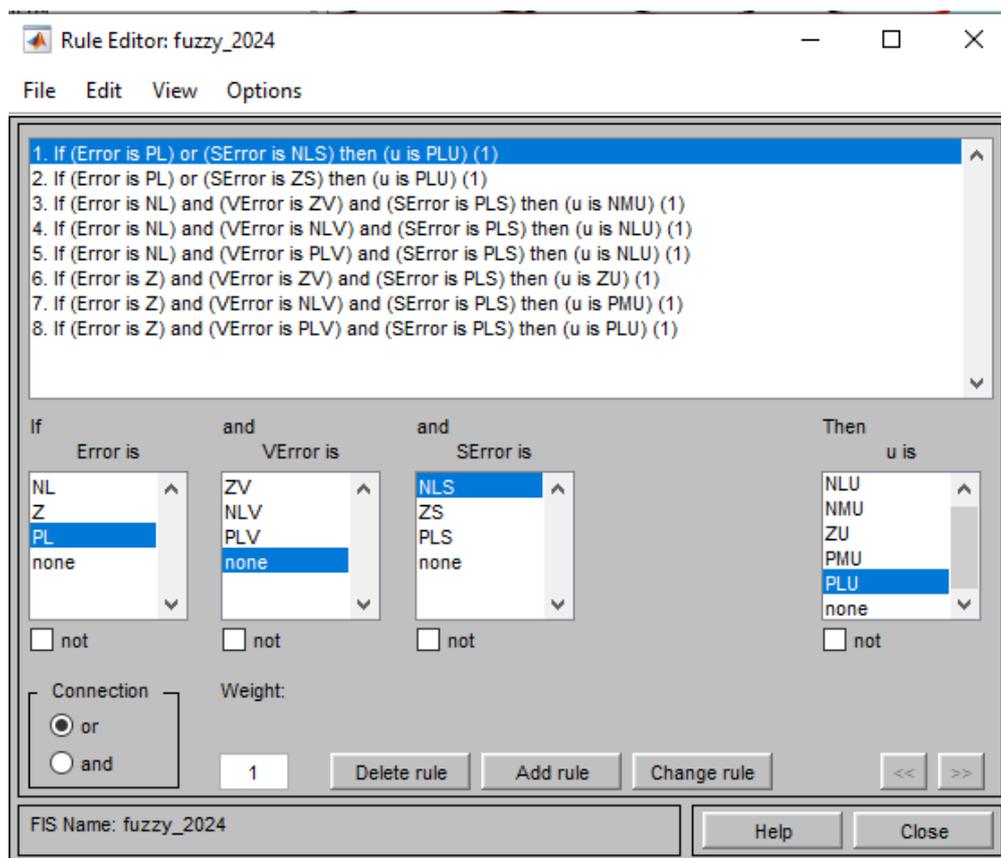


Рисунок 30 – Окно Rule Editor мастера настройки правил нечёткого вывода переменных (рисунок 31).

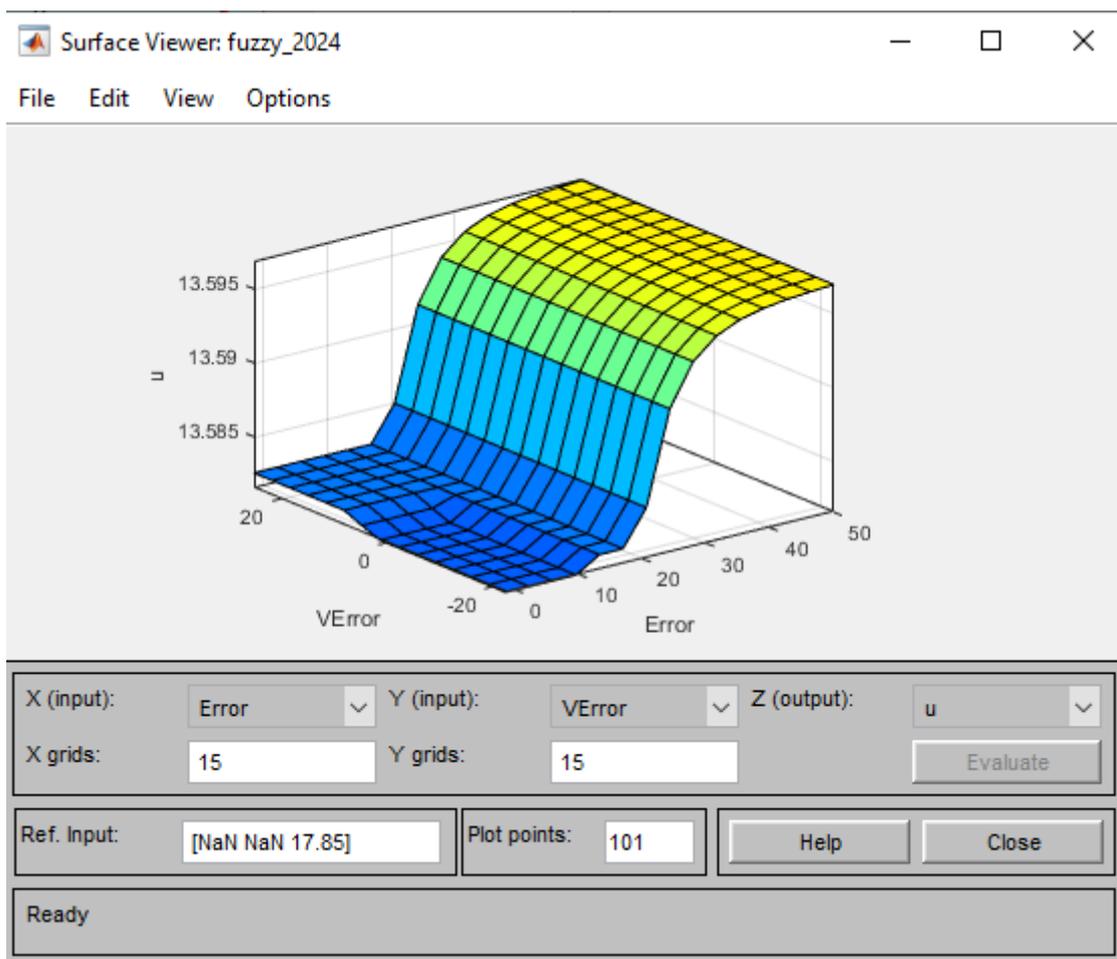


Рисунок 31 – График нечёткой переменной u в функции от нечётких переменных Error и VError

После просмотра графика, нажимаем кнопку Close и возвращаемся в окно мастера Fuzzy Logic Designer. Теперь выполнив команду File -> Export -> To File... сохраняем полученный нечёткий регулятор, например под именем fuzzy_22024.fis.

Теперь открываем модель, представленную на рисунке 20 и в блоке Fuzzy Logic Controller устанавливаем значение параметра FIS name - 'fuzzy_2024.fis'. Ставим ключи SW1 и SW2 в нижнее положение и запускаем модель электропривода с нечётким регулятором угловой скорости якоря.

На рисунке 32 изображен график изменения угловой скорости якоря при применении нечёткого ПИД-регулятора. Как видно на рисунке 32, перерегулирование отсутствует, время регулирования составляет примерно 1,33

секунды, что чуть больше, чем с классическим ПИД-регулятором, но при этом практически отсутствует установившаяся ошибка регулирования.

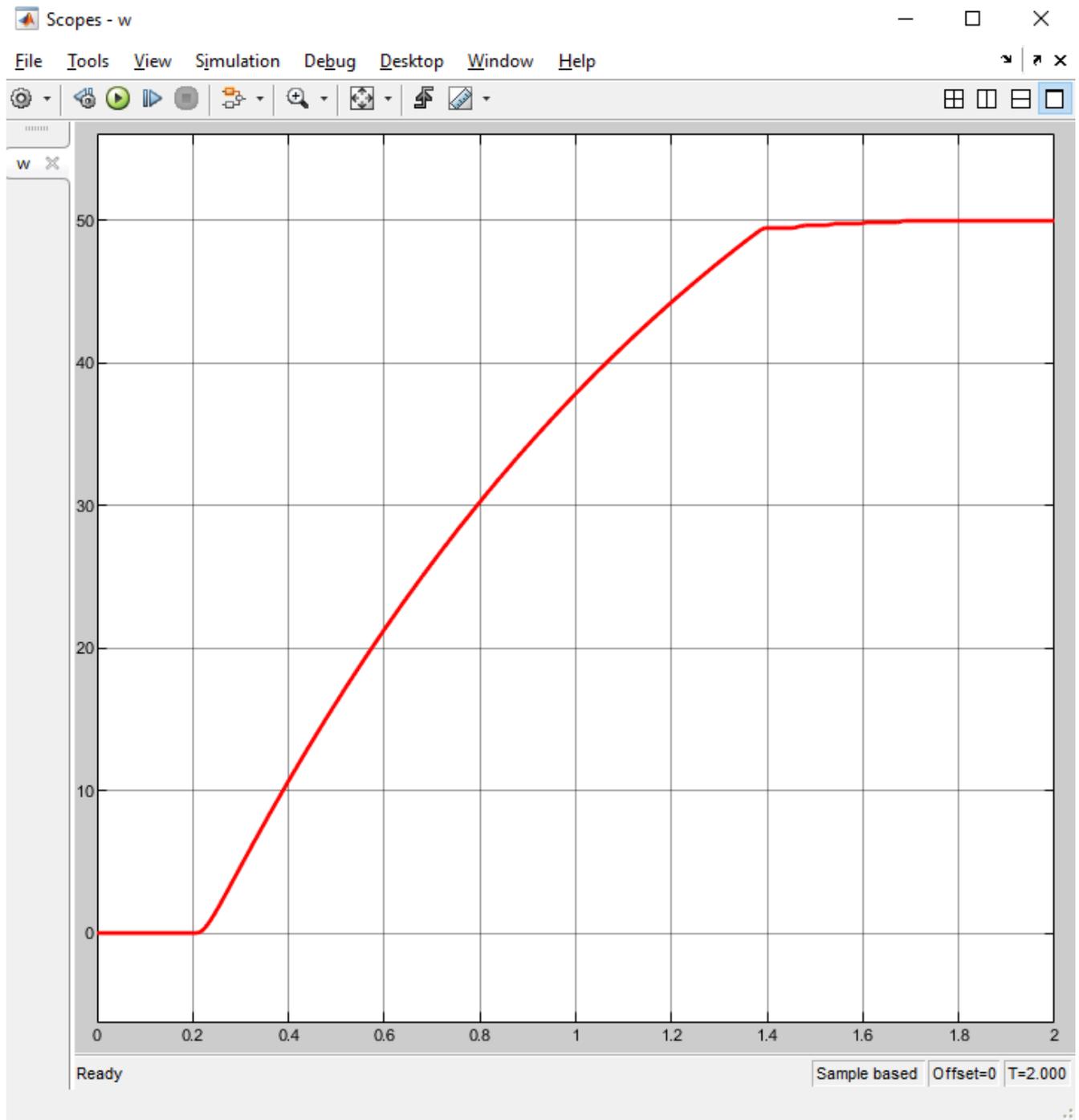


Рисунок 32 – График изменения угловой скорости якоря в двухконтурной системе подчиненного регулирования электропривода на базе ДПТ с нечётким регулятором угловой скорости якоря

Теперь попробуем применить в качестве ПИД-регулятора нейросетевой регулятор на основе искусственной нейронной сети прямого распространения. Для этого сначала подготовим входные и выходные данные для обучения нейронной сети. В качестве входных данных будем использовать матрицу, столбцы которой будут соответствовать ошибке угловой скорости якоря, её скорости изменения и интегралу. Создадим матрицу E выполнив в Matlab команду:

```
E= [simout.Data(:,1) simout.Data(:,2) simout.Data(:,3)].
```

В качестве выходных данных будем использовать вектор-столбец U значений сигнала управления с выхода PID-регулятора угловой скорости якоря. Для создания вектора-столбца U выполним в Matlab команду:

```
U= [simout.Data(:,4)].
```

В результате в *WorkSpace* появятся матрица E и вектор U , как показано на рисунке 33.

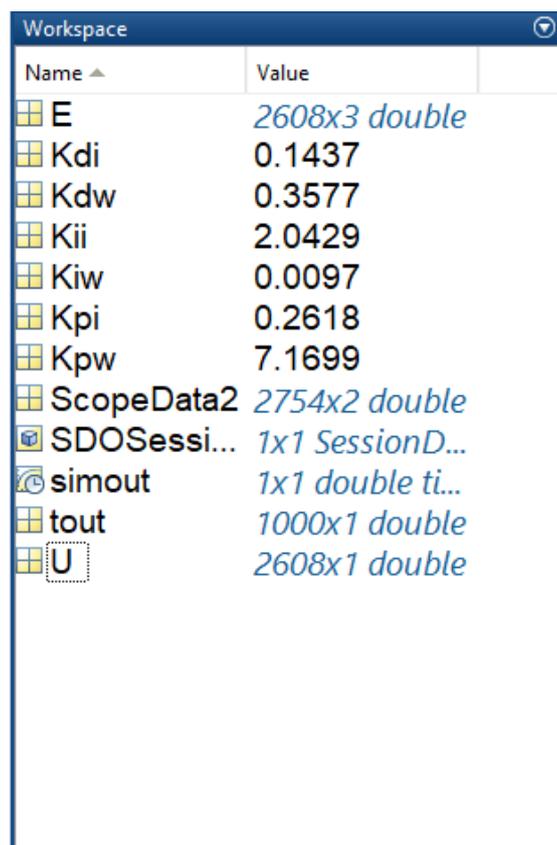


Рисунок 33 – Матрица E и вектор U

Далее в Matlab выполняем команду `nnstart` для запуска мастера создания нейронных сетей (рисунок 34).

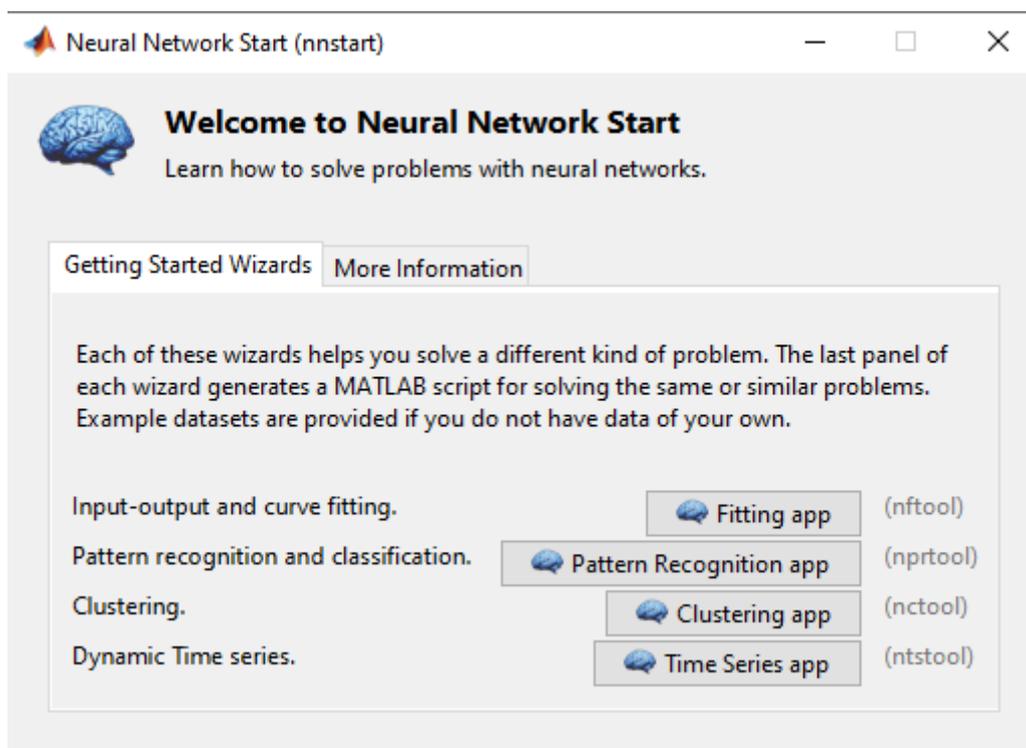


Рисунок 34 – Окно Neural Network Start (nnstart) мастера создания искусственных нейронных сетей

Для создания искусственной нейронной сети (ИНС) прямого распространения нажимаем на кнопку `Fitting app (nftool)`, откроется окно `Neural Fitting (nftool)` (рисунок 35). Нажимаем кнопку `Next` и переходим к окну `Neural Fitting (nftool) -> Select Data` выбора данных для обучения ИНС.

В качестве значения параметра `Inputs` выбираем матрицу E , в качестве значения параметра `Targets` выбираем вектор-столбец U , значение параметра `Samples are` устанавливаем равным `Matrix rows`, что позволяет выполнить автоматическое транспонирование входных и выходных данных (рисунок 36). Далее нажимаем кнопку `Next` для перехода к окну `Neural Fitting (nftool) -> Validation and Test Data` (рисунок 37). Устанавливаем значение параметра `Validation` – 5%, а параметра `Testing` – 5% (процент входных и выходных данных, которые будут использоваться для проверки правильности обучения ИНС и тестировании обученной ИНС). После

установки значений параметров по нажатию на кнопку Next переходим к окну Neural Fitting (nftool) -> Network Architecture (рисунок 38), где задаём значение параметра Number of Hidden Neurons (количество скрытых нейронов) равным 25.

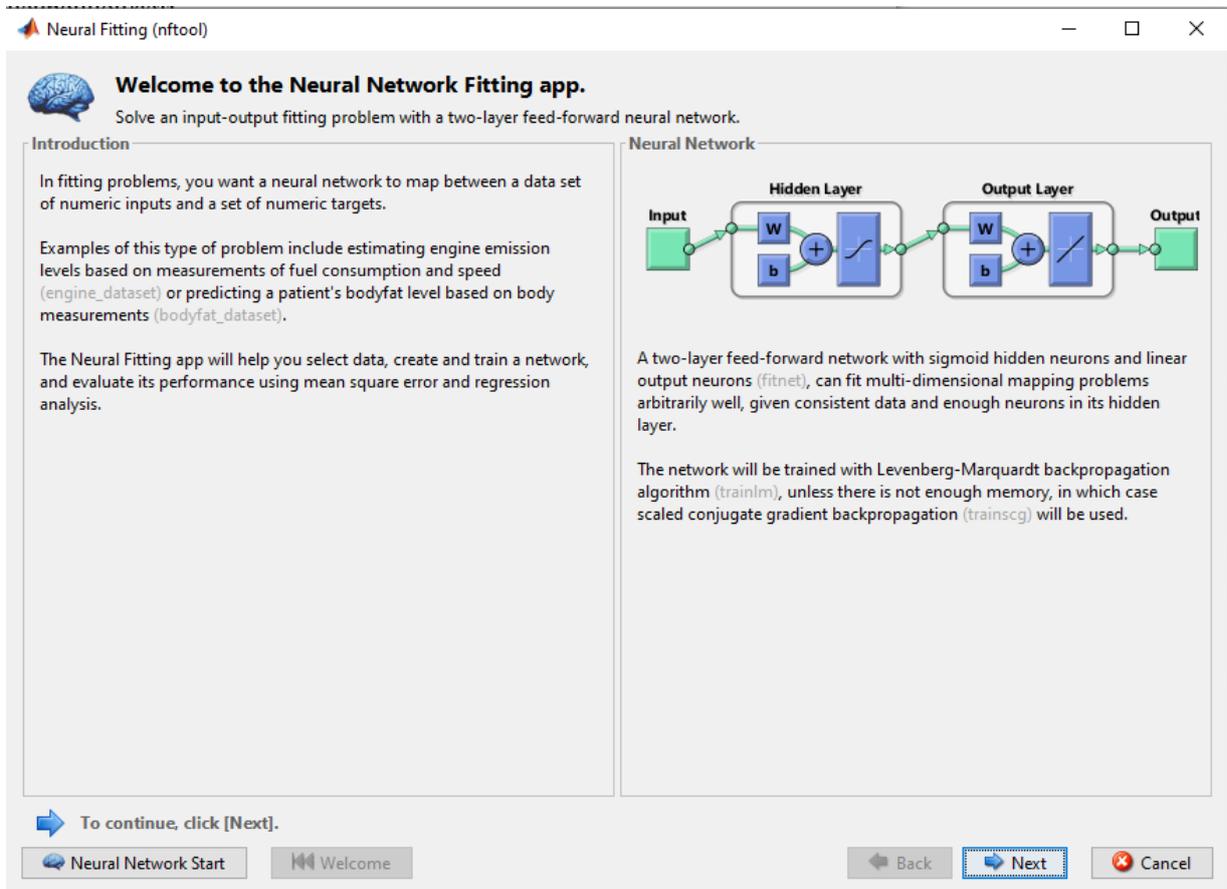


Рисунок 35 – Окно Neural Fitting (nftool)

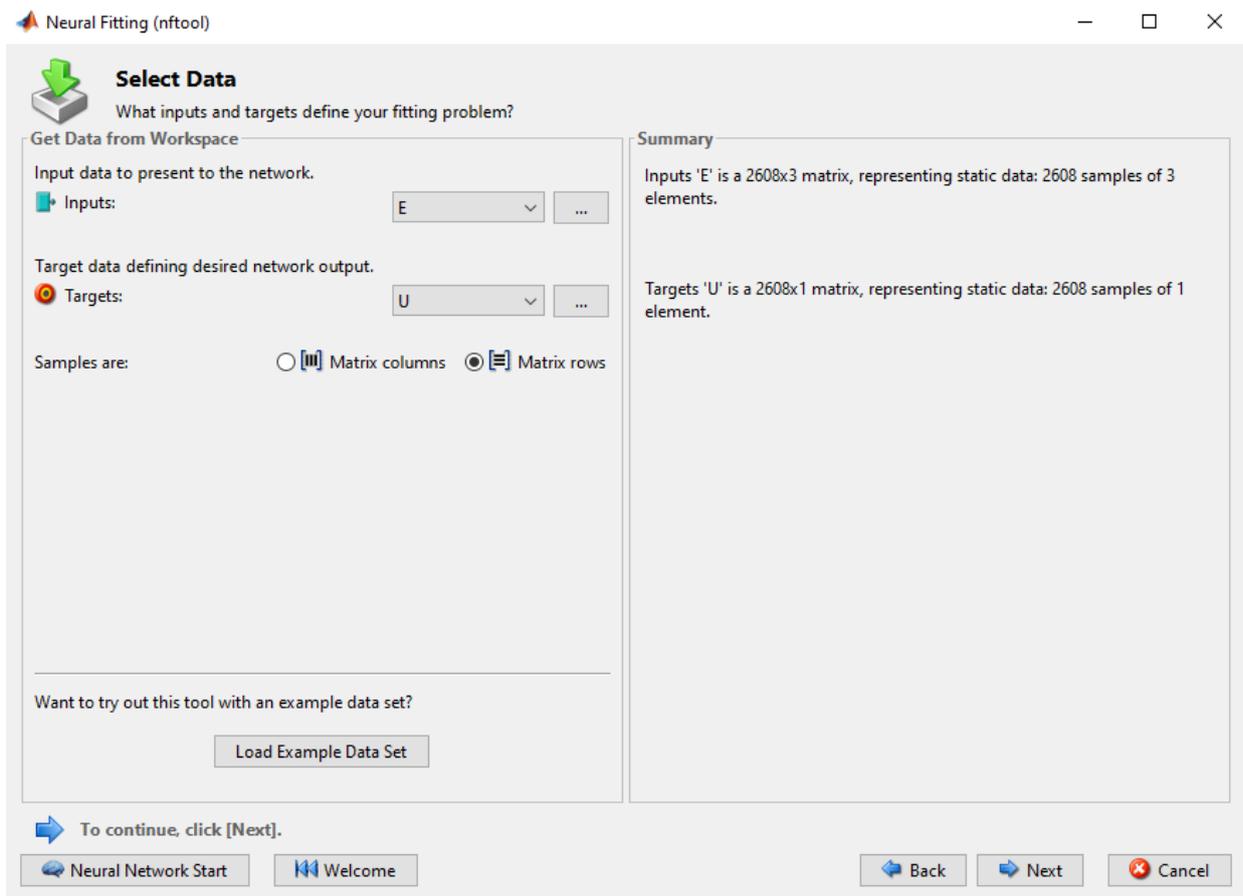


Рисунок 36 – Окно Neural Fitting (nftool) -> Select Data

Далее снова нажимаем кнопку Next и переходим к окну Neural Fitting (nftool) -> Train Network (рисунок 39). Устанавливаем следующие значения параметров:

Choose a training algorithm: Levenberg-Marquardt (алгоритм обучения ИНС).

Нажимаем кнопку Train и в открывшемся окне Neural Network Training (nntraintool) (рисунок 40) смотрим процесс обучения ИНС. Для оценки качества обучения нажимаем на кнопку Regression, и в открывшемся окне Neural Network Training Regression (plotregression) (рисунок 41) появятся графики регрессии выхода нейронной сети на обучающей выборке, проверочной выборке, тестовой выборке и полным входным данным к исходным выходным данным и значения коэффициента корреляции R для каждого графика.

При неудовлетворительных результатах обучения сеть можно переобучить, нажав на кнопку Retrain а окне Neural Fitting (nftool) -> Train Network, не меняя никаких параметров или предварительно поменяв количество скрытых нейронов и размеры обучающей и тестовой выборок.

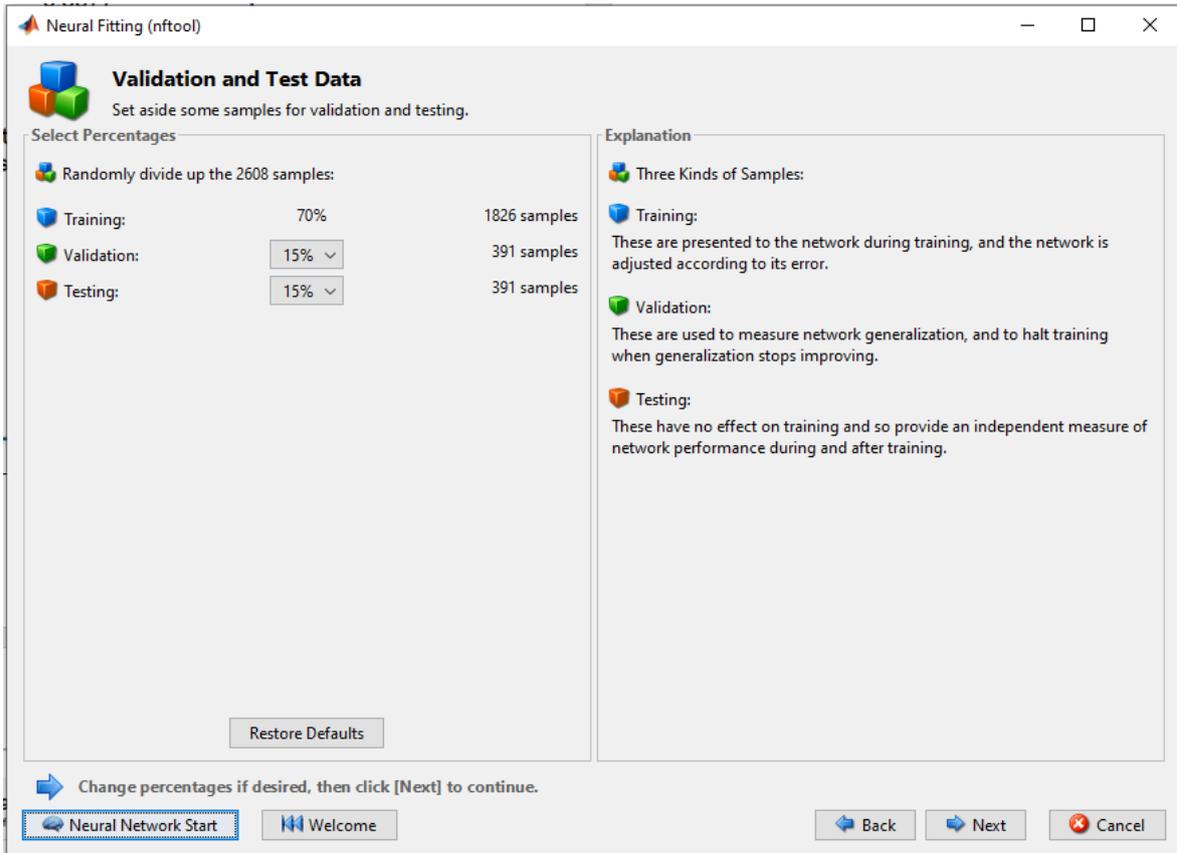


Рисунок 37 – Окно Neural Fitting (nftool) -> Validation and Test Data

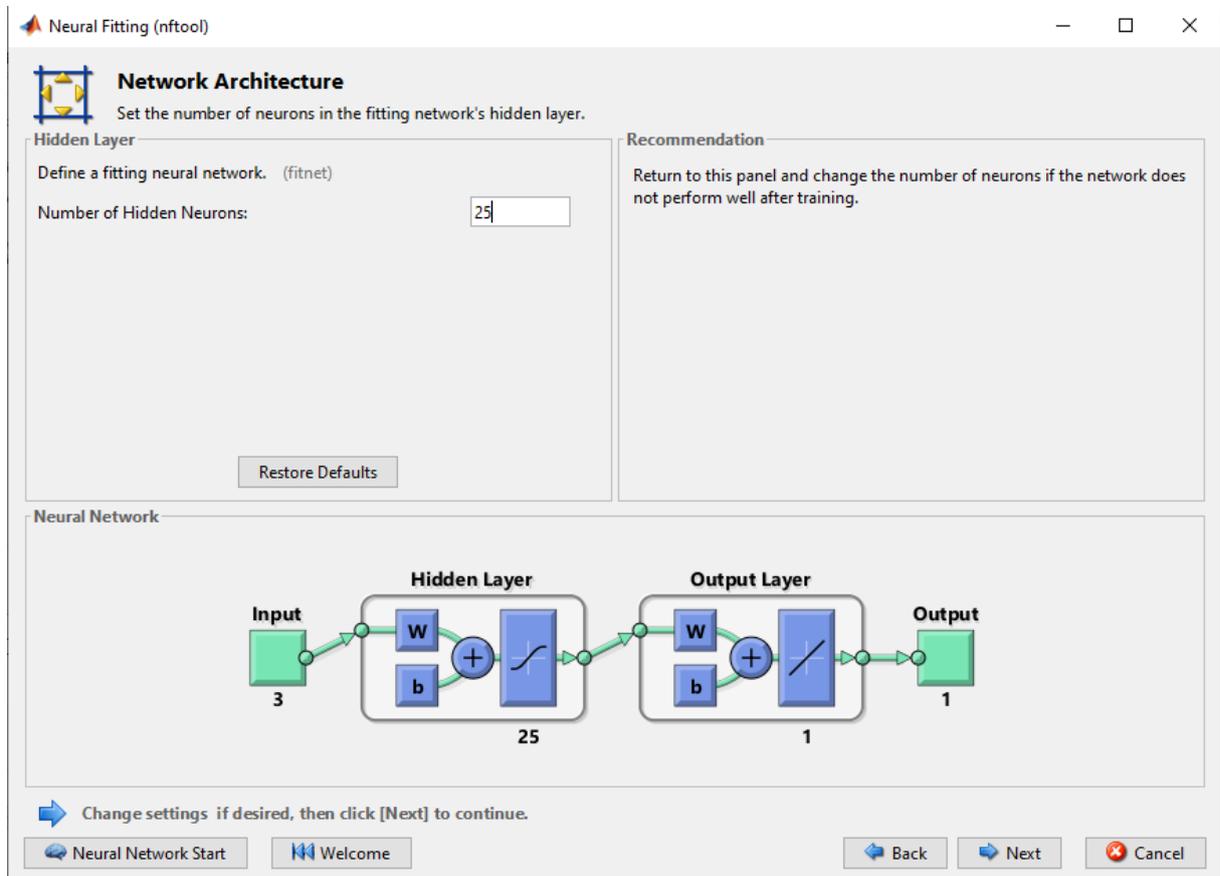


Рисунок 38 – Окно Neural Fitting (nftool) -> Network Architecture

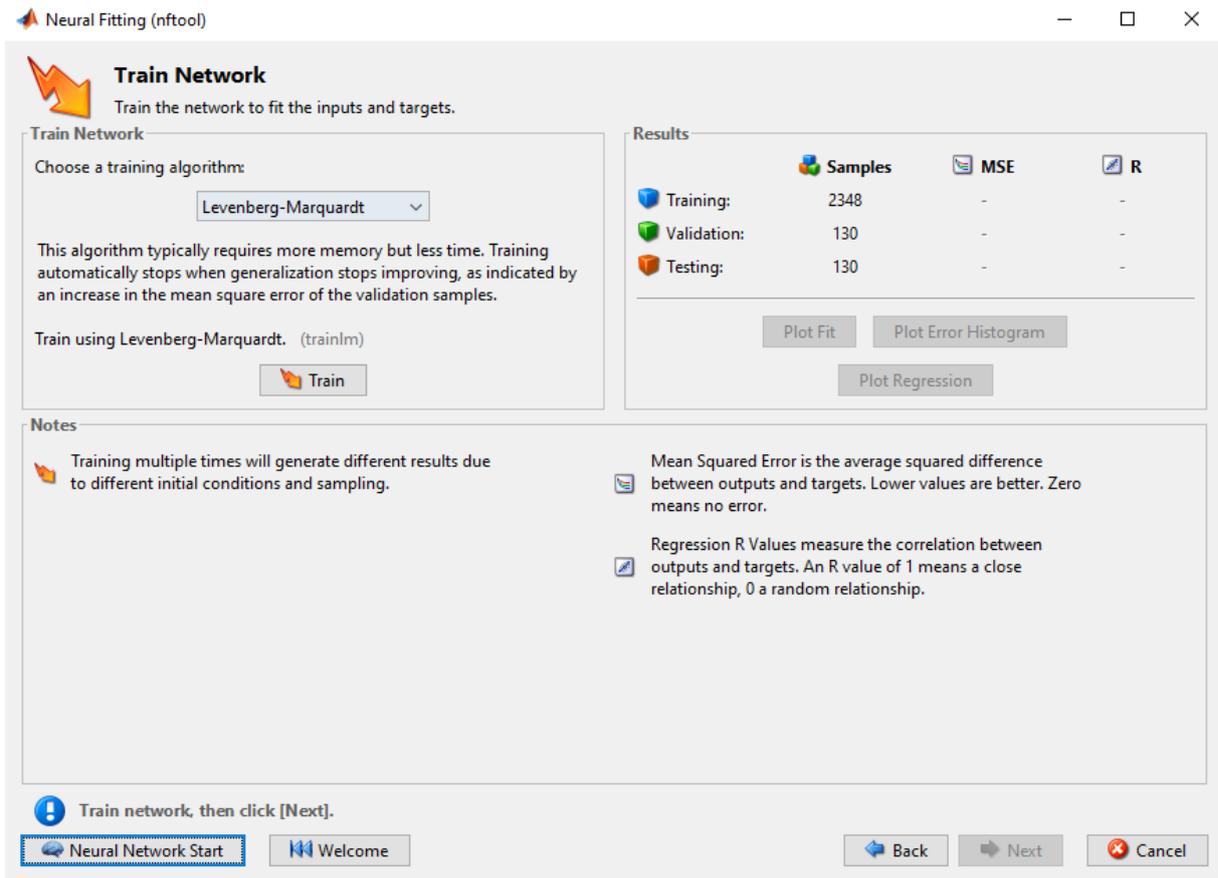


Рисунок 39 – Окно Neural Fitting (nftool) -> Train Network

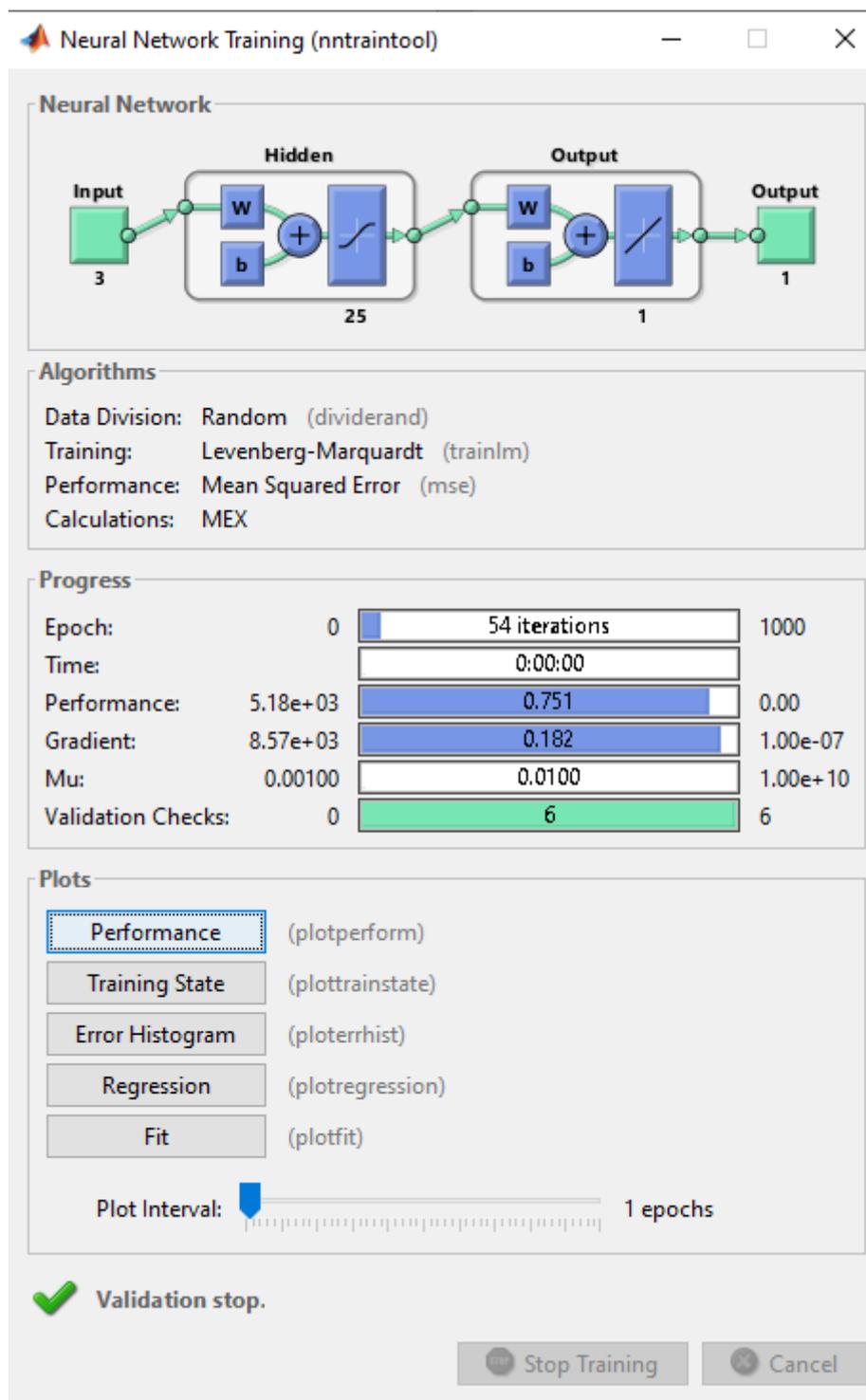


Рисунок 40 – Окно Neural Network Training (nntraintool)

После получения удовлетворительных результатов обучения в окне Neural Fitting (nftool) -> Train Network (рисунок 39) нажимаем кнопку Next и переходим в окно Neural Fitting (nftool) -> Evaluate Network (рисунок 42). В этом окне ещё раз

нажимаем кнопку Next и переходим к окну Neural Fitting (nftool) -> Deploy Solution (рисунок 43), где можно выбрать способ сохранения обученной ИНС.

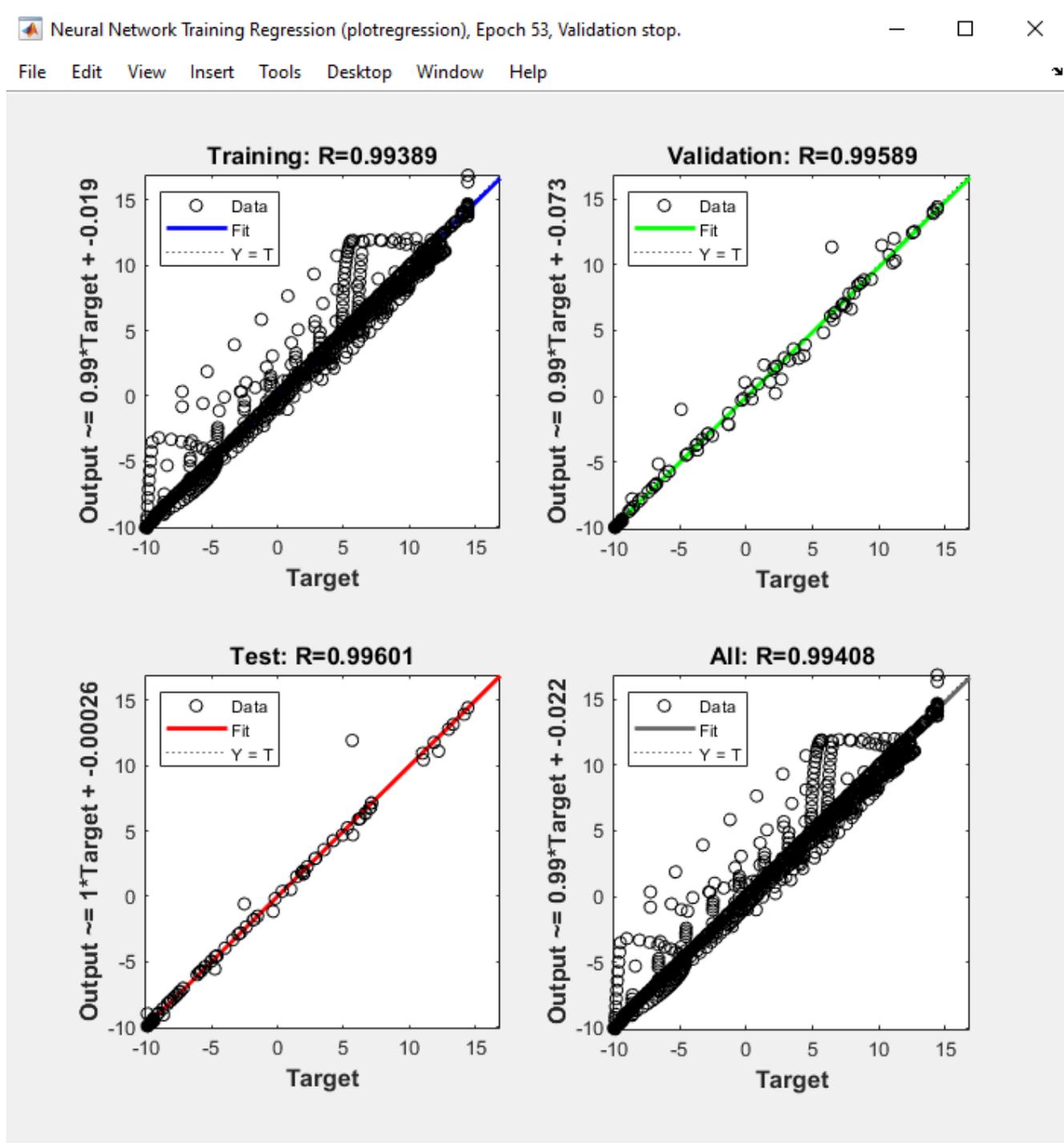


Рисунок 41 – Окно Neural Network Training Regression (plotregression)

Выбираем параметр Generate a Simulink diagram (gensim) -> Simulink Diagram.

В открывшемся окне untitled* - Simulink (рисунок 44) копируем синий блок

Function Fitting Neural Network и вставляем его в модель, изображённую на рисунке 20.

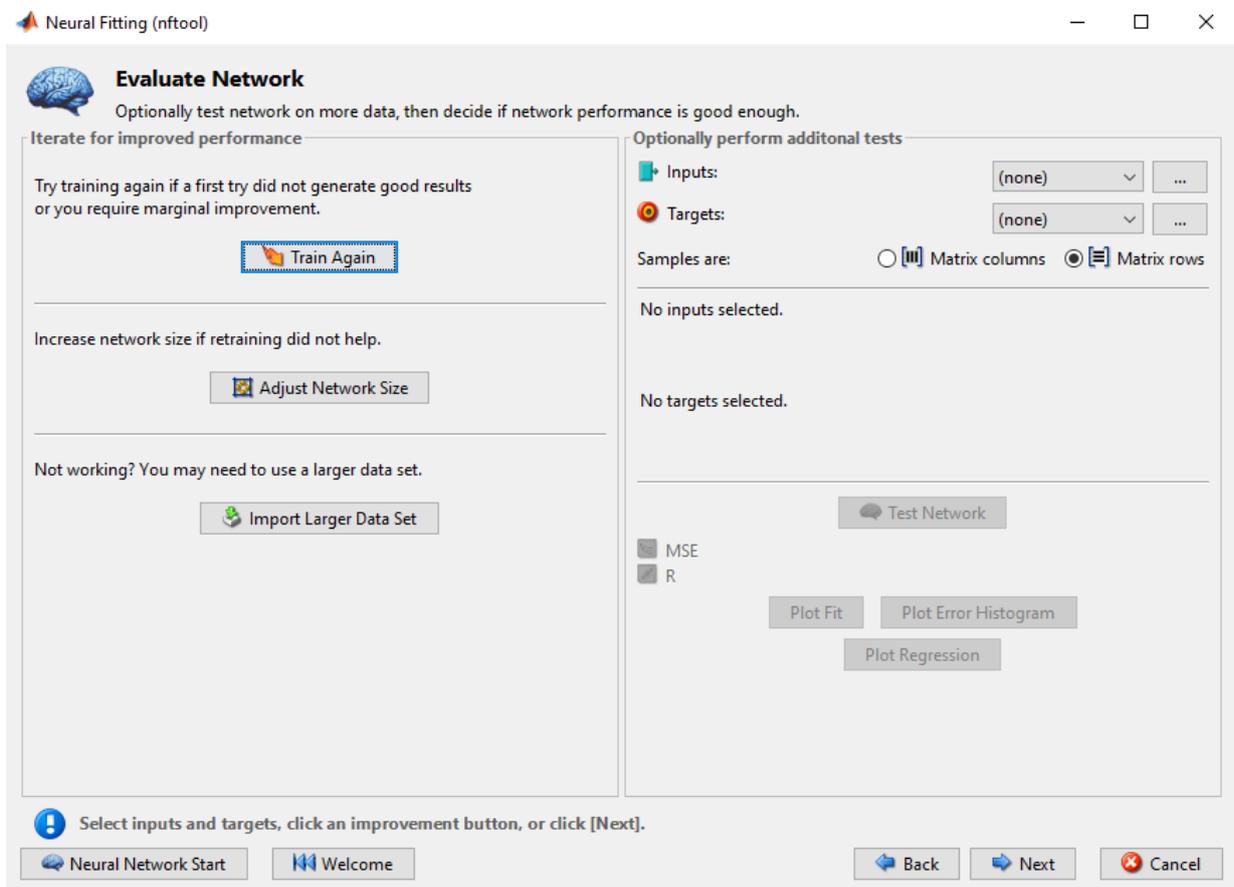


Рисунок 42 – Окно Neural Fitting (nftool) -> Evaluate Network

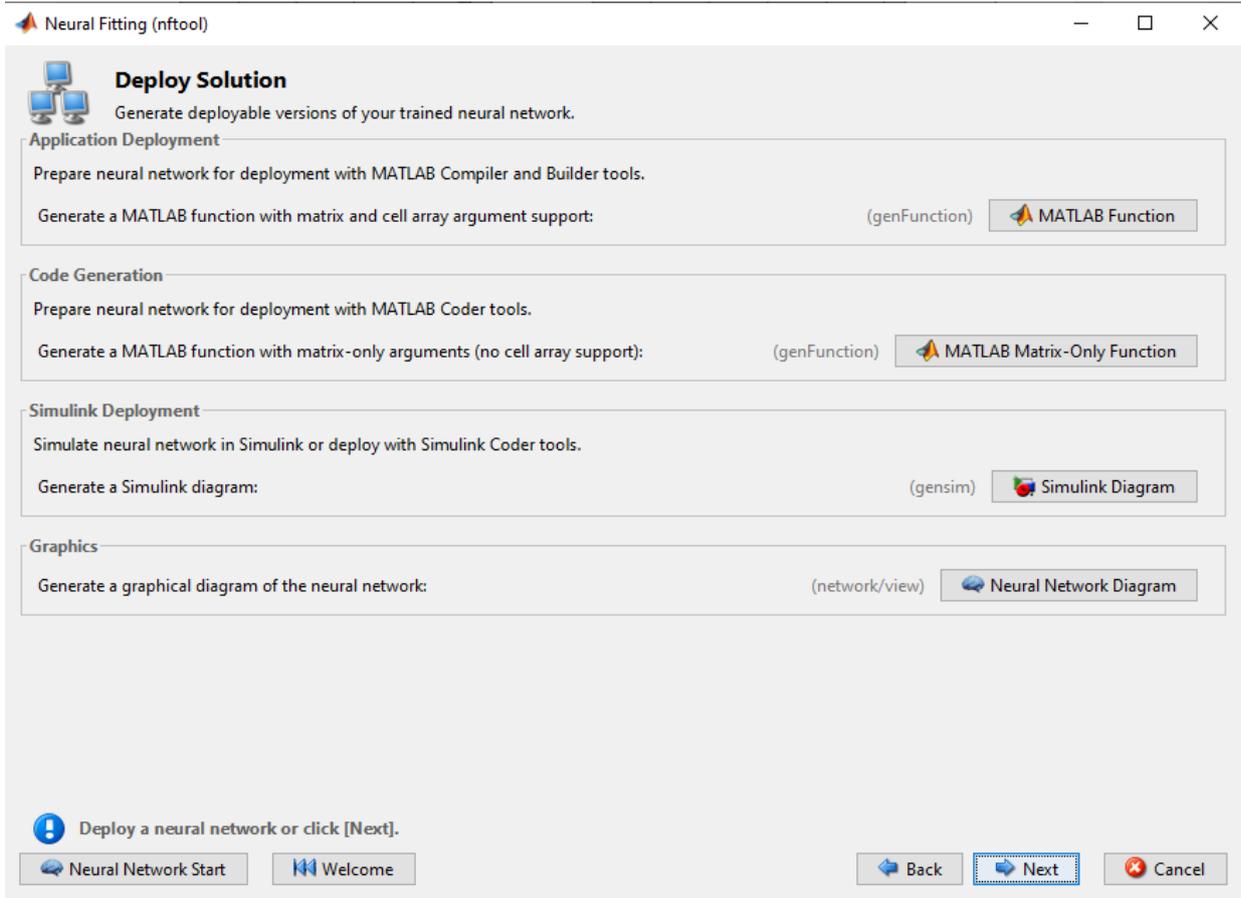


Рисунок 43 – Окно Neural Fitting (nftool) -> Deploy Solution

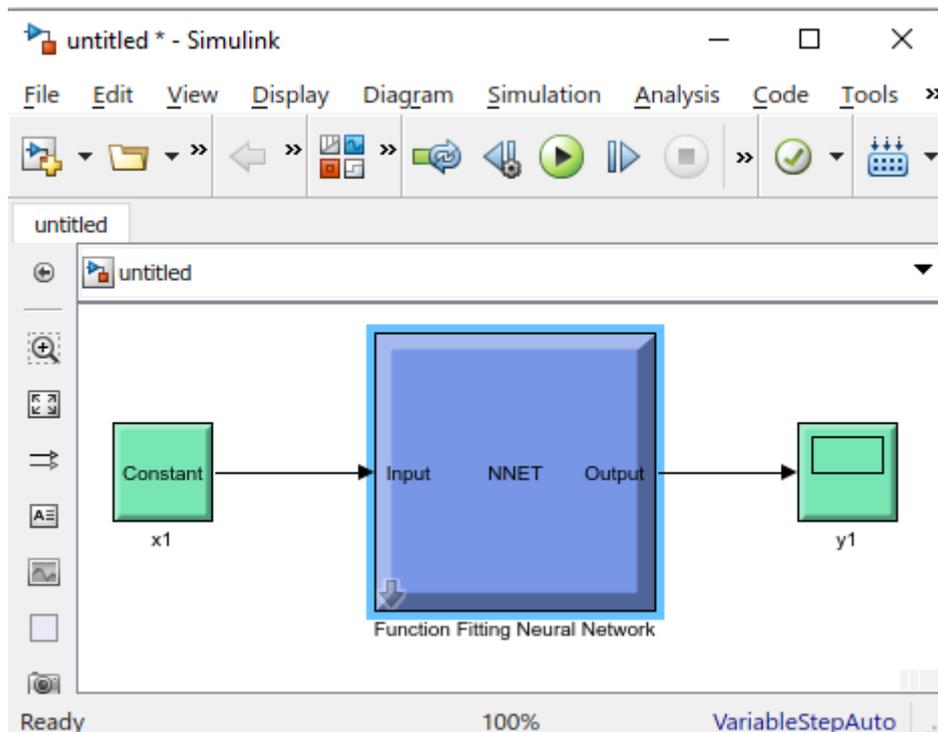


Рисунок 44 – Окно untitled* - Simulink

Дорабатываем модель на рисунке 20, добавляя ключ SW3, как показано на рисунке 45. Теперь, устанавливая ключ SW3 в верхнее положение, можно подключать вместо классического ПИД-регулятора угловой скорости якоря обученный нейросетевой регулятор на основе обученной ИНС прямого распространения.

На рисунке 46 изображен график изменения угловой скорости якоря при применении нейросетевого ПИД-регулятора. Как видно на рисунке 46, перегулирование составляет 34%, время регулирования составляет примерно 3,25 секунды, что почти в 3 раза больше, чем с классическим ПИД-регулятором, а установившаяся ошибка регулирования равна 3%.

Таким образом, нейросетевой регулятор показал из всех предыдущих самые худшие показатели качества переходного процесса. Возможно это связано с тем, что в процессе создания структуры ИНС и её обучения не были достигнуты оптимальные параметры и процесс обучения ИНС нужно повторить для новой структуры ИНС.

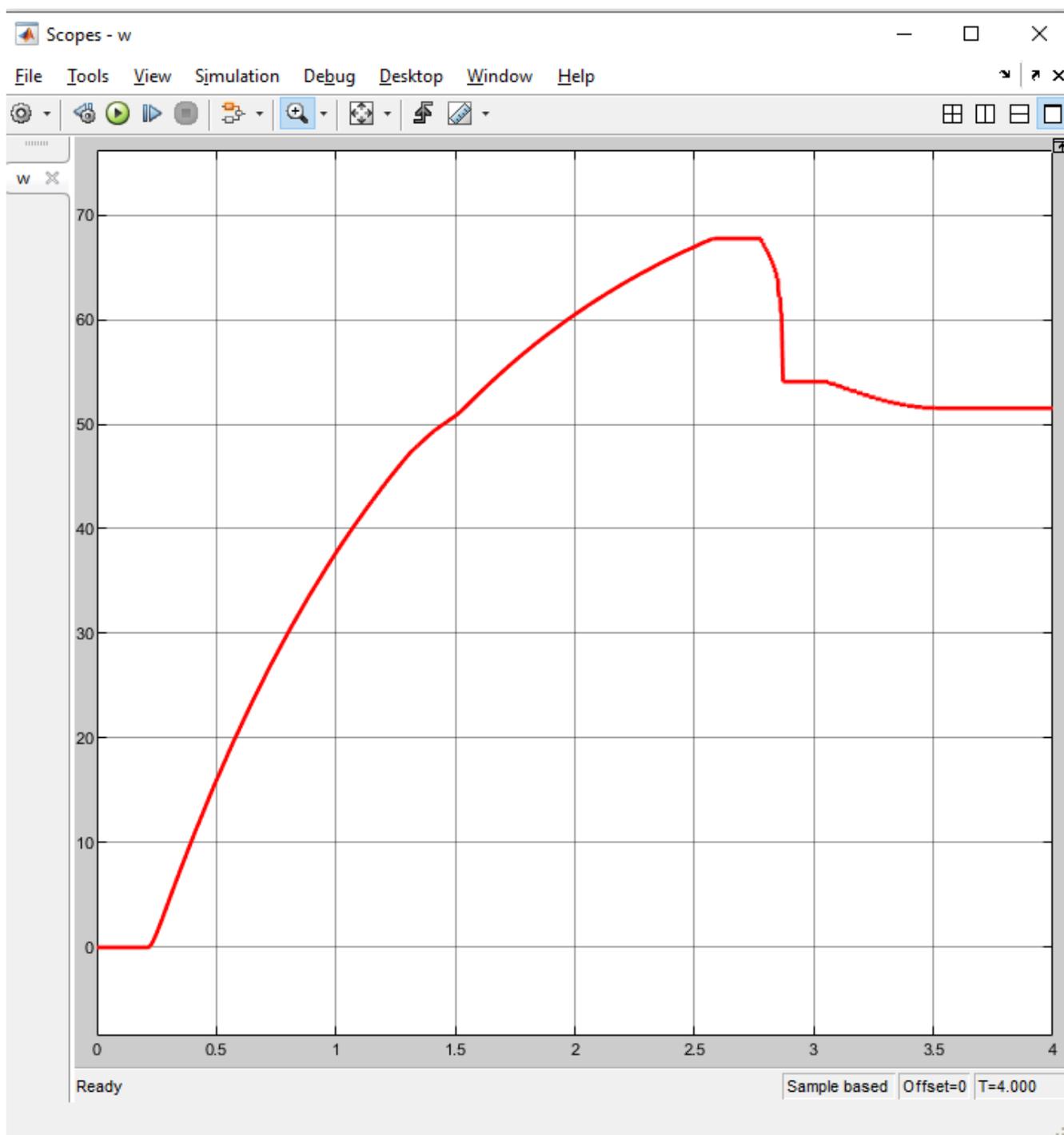


Рисунок 46 – График изменения угловой скорости якоря в двухконтурной системе подчиненного регулирования электропривода на базе ДПТ с нейросетевым регулятором угловой скорости якоря

Для синтеза предиктивного регулятора преобразуем модель электропривода как показано на рисунке 47, сделав из неё подсистему (subsystem) и запишем её в отдельный файл Simulink.

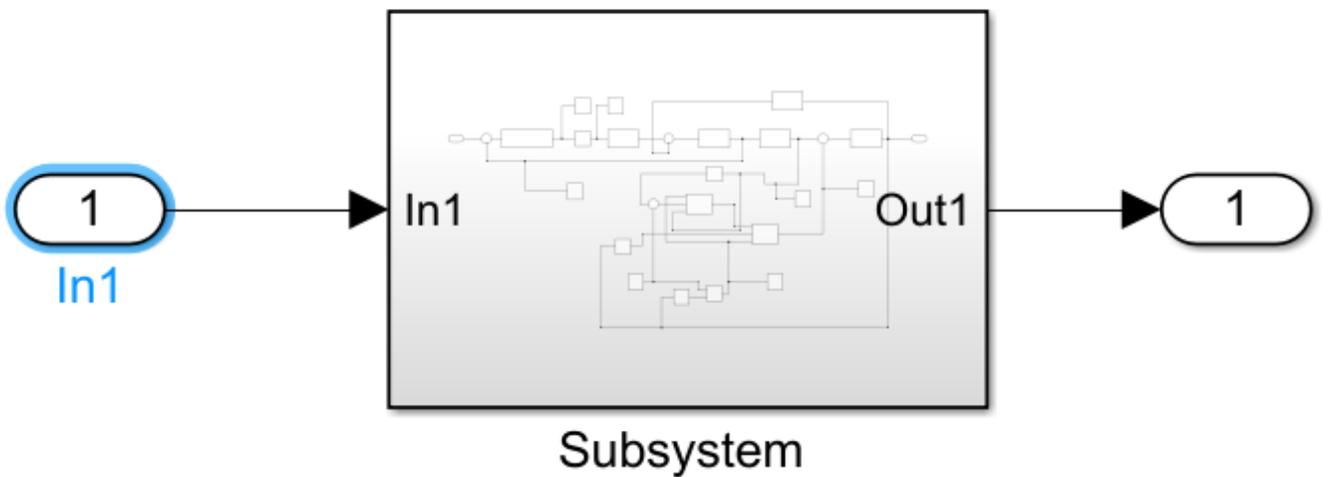
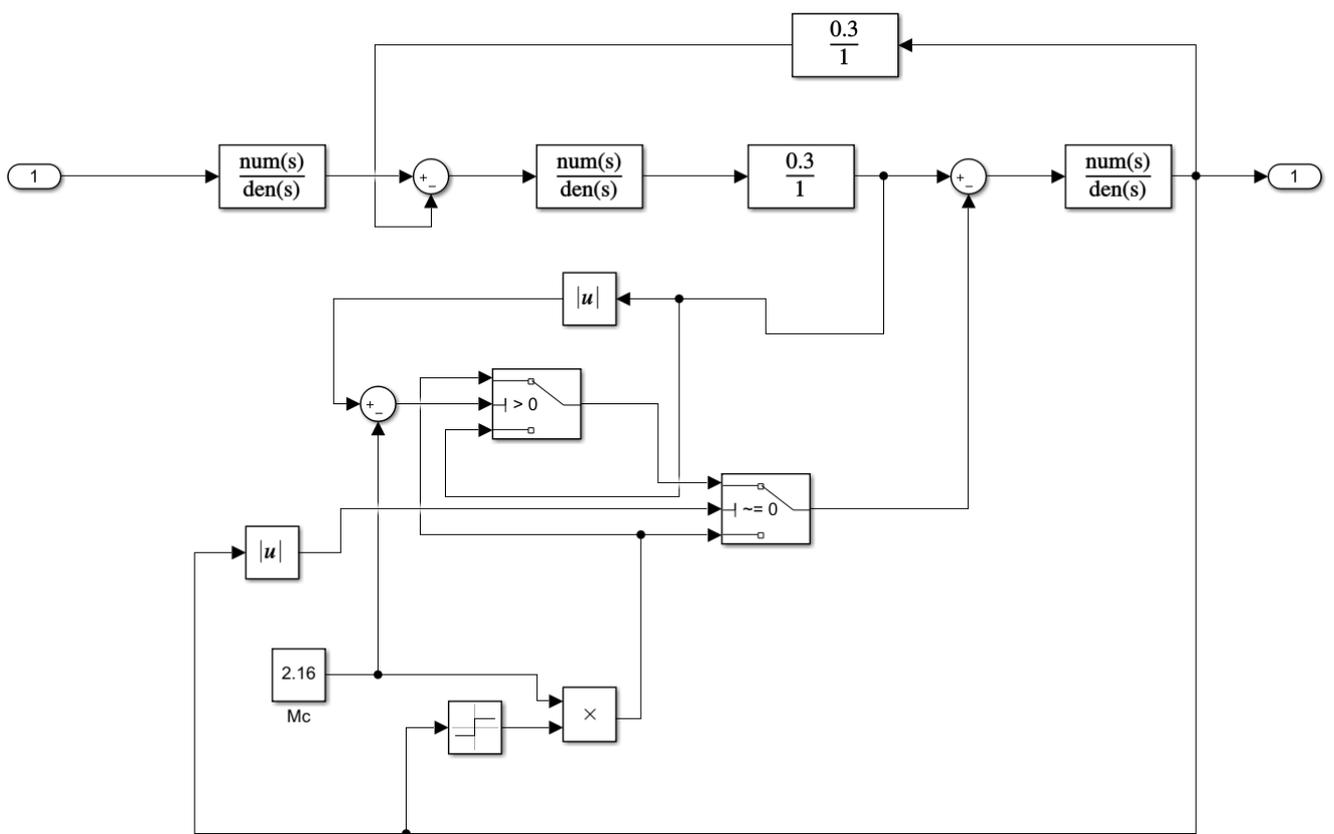


Рисунок 47 – Модель электропривода для синтеза предиктивного регулятора

На рисунке 48 представлена модель электропривода с предиктивным регулятором. Блок NN Predictive Controller – предиктивный регулятор, блок Subsystem – модель электропривода без обратной связи и регулятора угловой скорости якоря.

Для настройки предиктивного регулятора NN Predictive Controller зададим его основные параметры (рисунок 49).

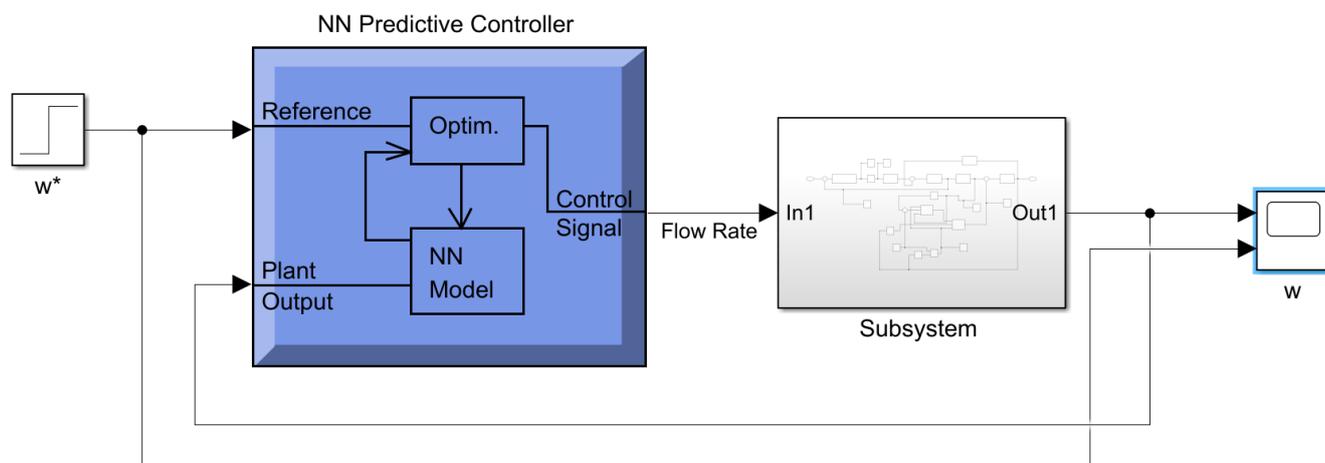


Рисунок 48 – Модель электропривода с предиктивным регулятором

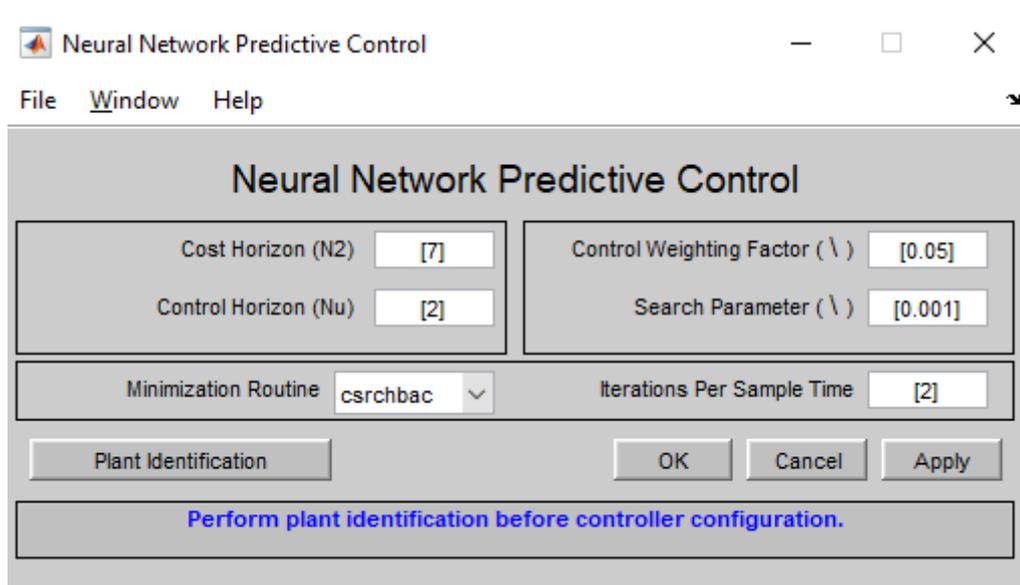


Рисунок 49 – Основные параметры настройки предиктивного регулятора

Зададим следующие значения параметров:

Cost Horizon (N2) – 7 (количество значений выходного сигнала, для которых будет оцениваться ошибка управления);

Control Horizon (Nu) – 2 (количество значений управляющего сигнала, для которых будет оцениваться ошибка управления);

Control Weighting Factor (ρ) – 0.05 (весовой коэффициент ошибки по управлению по отношению к ошибке отклонения выходного сигнала от заданного значения);

Search Parameter (α) – 0,001 (параметр, определяющий, когда запускается линейный поисковый алгоритм);

Minimization Routine – csrchbas (линейный поисковый алгоритм, используемый в алгоритме оптимизации ошибки регулирования);

Iterations Per Sample Time – 2 (количество итераций алгоритма оптимизации, выполняемых за один временной интервал моделирования).

Далее нажимаем кнопку Plant Identification для идентификации модели электропривода (рисунок 50).

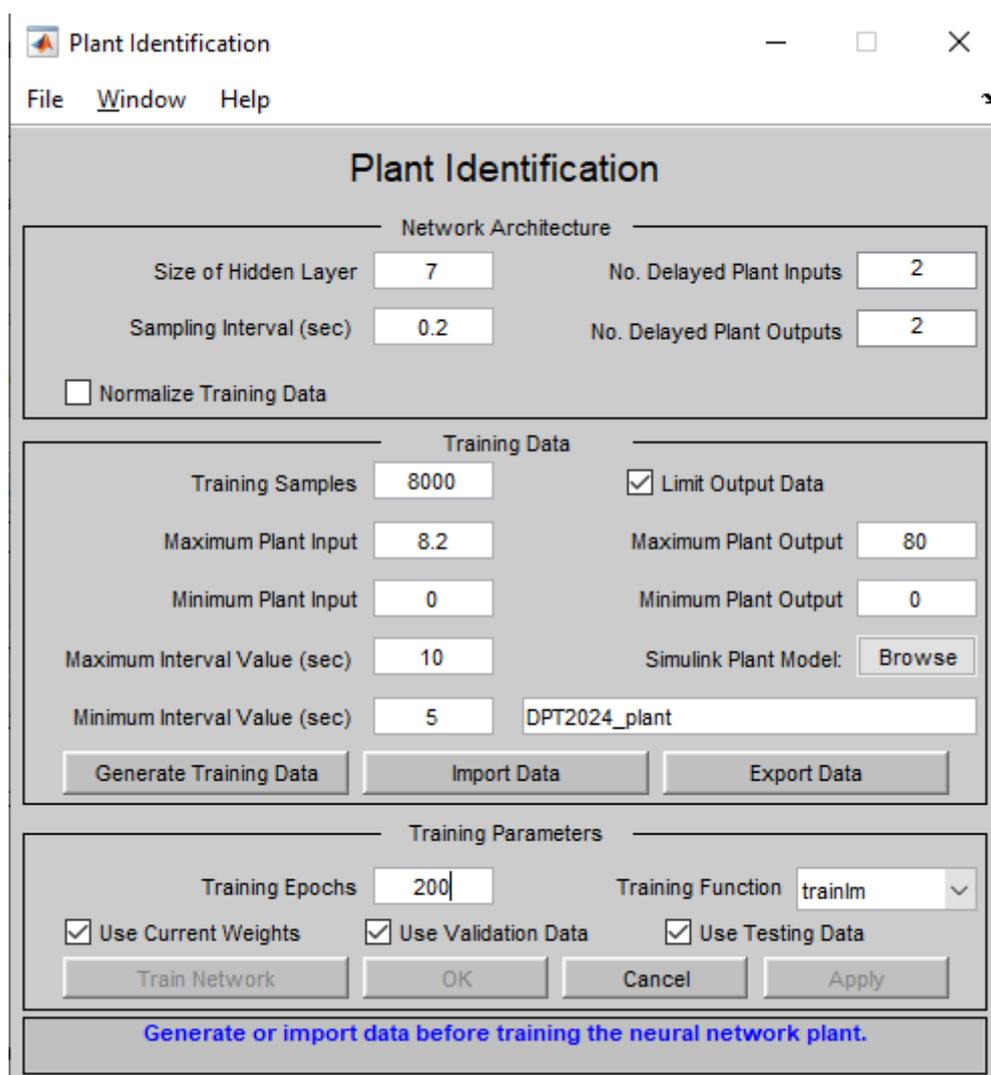


Рисунок 50 – Окно Plant Identification

Зададим следующие значения параметров:

Size of Hidden Layer – 7 (количество нейронов в первом слое нейросетевой модели электропривода);

Sampling Interval (sec) – 0.2 (интервал времени, за который программа получает данные из модели электропривода);

No. Delayed Plant Inputs – 2 (количество входов, поступающих в модель электропривода с задержкой);

No. Delayed Plant Outputs – 2 (количество выходов, поступающих в модель электропривода с задержкой);

Training Samples – 8000 (количество значений данных для обучения, проверки и тестирования ИНС);

Maximum Plant Input – 8.2 (максимальное значение входного сигнала электропривода);

Minimum Plant Input – 0 (минимальное значение входного сигнала электропривода);

Maximum Interval Value (sec) 10 – (максимальная длина интервала, на котором входной сигнал электропривода не меняет значение)

Minimum Interval Value (sec) – 5 (минимальная длина интервала, на котором входной сигнал электропривода не меняет значение);

Limit Output Data – ограничить выходной сигнал электропривода (угловую скорость якоря);

Maximum Plant output – 80 (максимальное значение выходного сигнала электропривода);

Minimum Plant output – 0 (минимальное значение выходного сигнала электропривода);

Simulink Plant Model – путь к файлу с моделью электропривода;

Generate Training Data – кнопка для генерации данных для обучения ИНС;

Training Epochs – 200 (количество эпох обучения ИНС);

Training Function – trainlm (метод обучения ИНС).

После задания параметров для генерирования входных данных и параметров обучения ИНС, окно Plant Identification должно выглядеть, как на рисунке 50. Далее нажимаем кнопку Generate Training Data. Открывается окно Plant Input-Output Data.

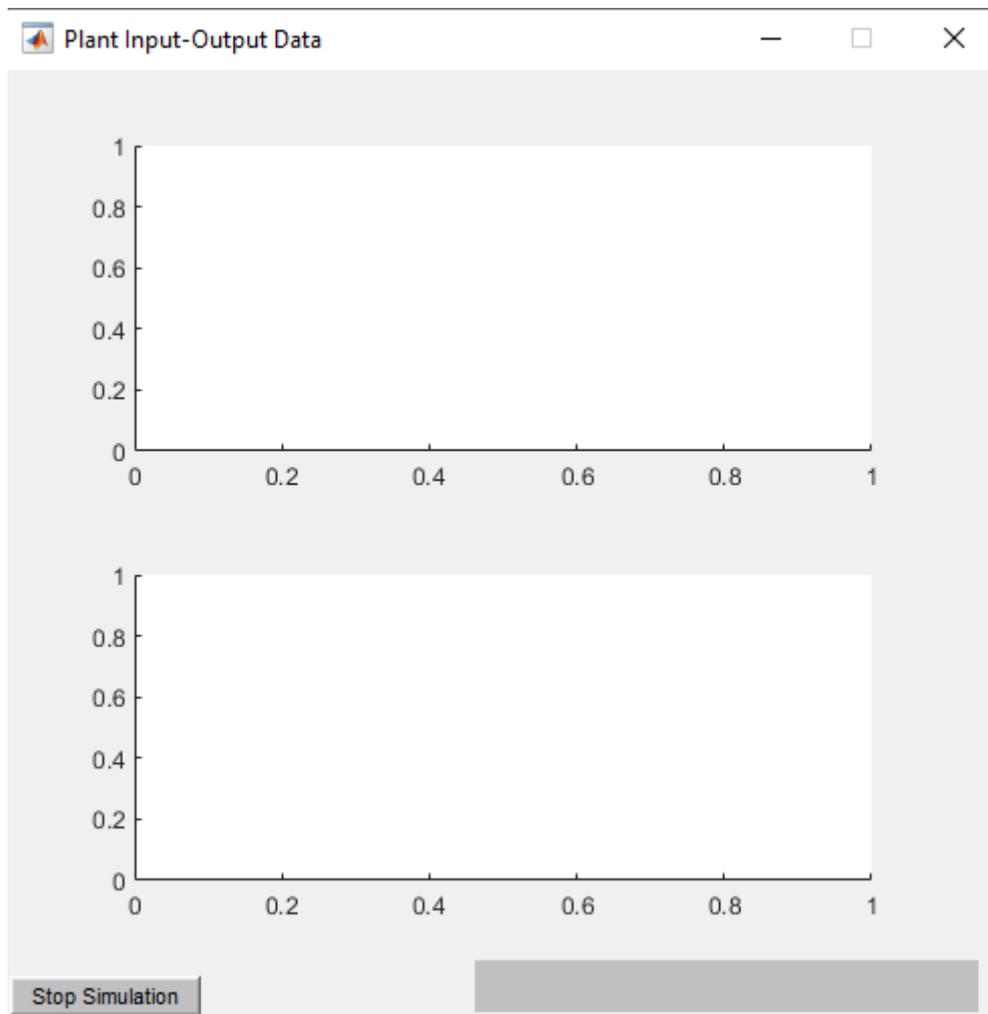


Рисунок 51 – Окно Plant Input-Output Data

После окончания процесса генерации данных для обучения ИНС в окне Plant Identification нажимаем кнопку Train Network, запуская процесс обучения ИНС (рисунок 58). Если этого не произошло в результате какой-то ошибки (красные предупреждения в командной строке WorkSpace), то данные для обучения ИНС можно подготовить вручную. Для этого из модели в файле 'DPT2024_plant.slx' сделаем изменения, как показано на рисунке 52.

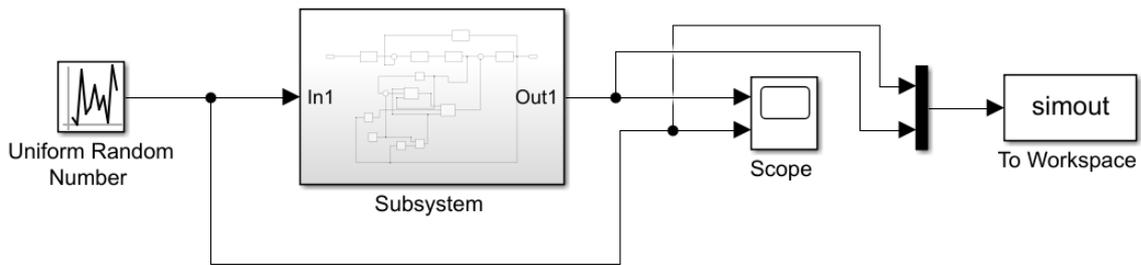


Рисунок 52 – Модель для генерирования данных для обучения ИНС

Блок Uniform Random Number генерирует случайную последовательность управляющих сигналов, подаваемых на вход разомкнутой системы электропривода в диапазоне от «минус»1 до 1 (скважность γ) через каждые 5 секунд в течение времени моделирования, равном 1000 секунд. Настройки блока приведены на рисунке 53.

Блок simout служит для передачи управляющих сигналов с выхода блока Uniform Random Number и с выхода электропривода в WorkSpace. Настройки блока приведены на рисунке 54.

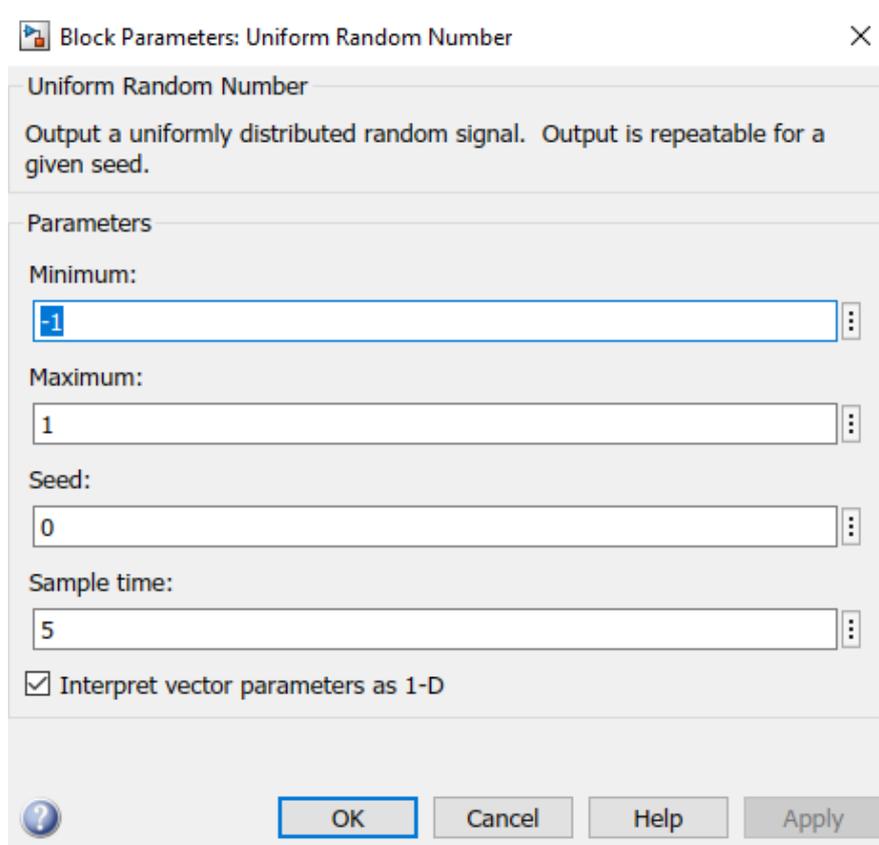


Рисунок 53 – Настройки блока Uniform Random Number

После окончания процесса выполнения модели на рисунке 52, в WorkSpace появится объект Out формата Timeseries с результатами для моделирования (рисунок 55). Для создания массивов для обучения ИНС можно в Matlab выполнить скрипт:

```
U= [simout.Data(:,1)];
```

```
Y= [simout.Data(:,2)]
```

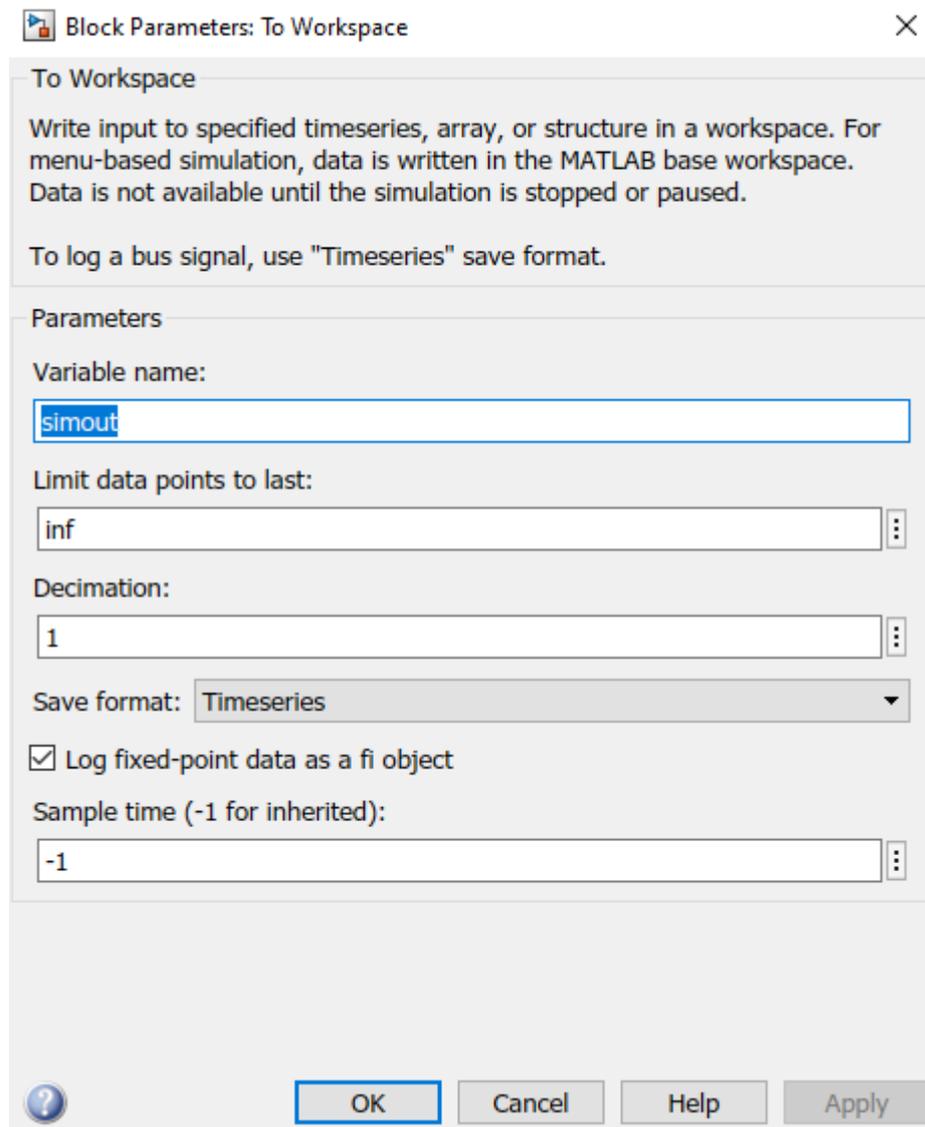


Рисунок 54 – Настройки блока simout

Name	Value
out	1x1 Simulati...
U	3023862x1 d...
Y	3023862x1 d...

Рисунок 55 – Подготовка данных для обучения ИНС

В результате после выполнения скрипта в WorkSpace появятся два массива входных данных U и выходных данных Y для обучения ИНС (рисунок 55).

Возвращаемся снова в окно Plant Identification (рисунок 50) и нажимаем кнопку Import Data. Откроется окно Import Data (рисунок 56). Устанавливаем параметр Type of Data -> Arrays и в появившемся в среднем поле списке перемещаем стрелкой массив U в поле Input Array, а массив Y в поле Output Array. После этого нажимаем кнопку ОК, после чего открывается окно Plant Input-Output Data (рисунок 57), в котором отображаются во времени графики изменения входного и выходного сигнала. Нажимаем на кнопку Accept Data и возвращаемся в окно Plant Identification (рисунок 50). Нажимаем на кнопку Train Network и запускаем окно Neural Network Training (ntraintool) (рисунок 58), в котором отображается прогресс обучения ИНС.

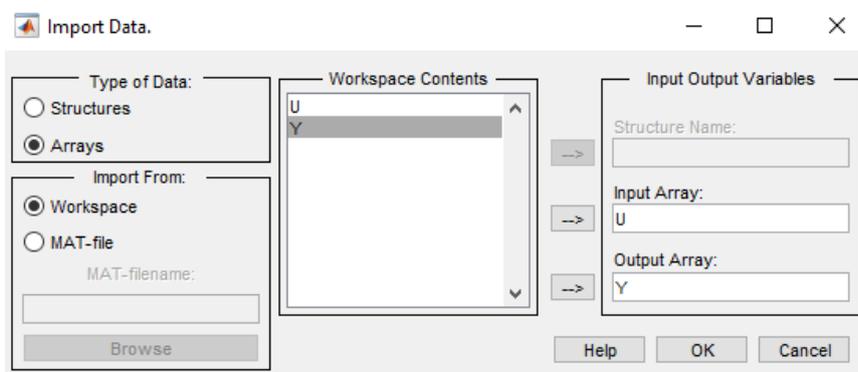


Рисунок 56 – Окно Import Data

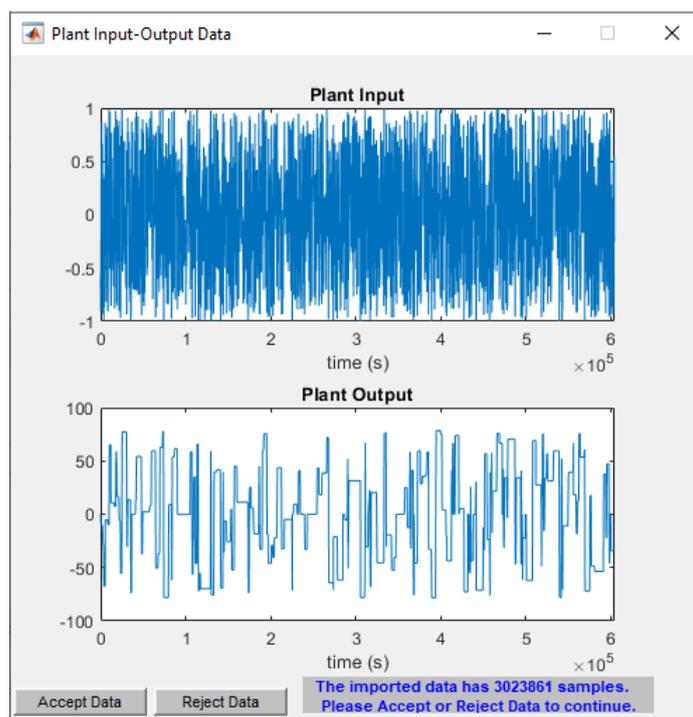


Рисунок 57 – Окно Plant Input-Output Data

По завершении процесса обучения ИНС появляются окна Training data for NN Predictive Control (рисунок 59), Validation data for NN Predictive Control (рисунок 60), Testing data for NN Predictive Control (рисунок 61), содержащие ошибки нейронной сети для тренировочной, проверочной и тестовой выборок данных соответственно. Эти графики позволяют оценить качество обучения ИНС. Если данные являются удовлетворительными, то в окне Plant Identification (рисунок 50) нажимаем кнопку Apply, а потом кнопку ОК. После этого в окне Neural Network Predictive Control (рисунок 49) нажимаем кнопку ОК.

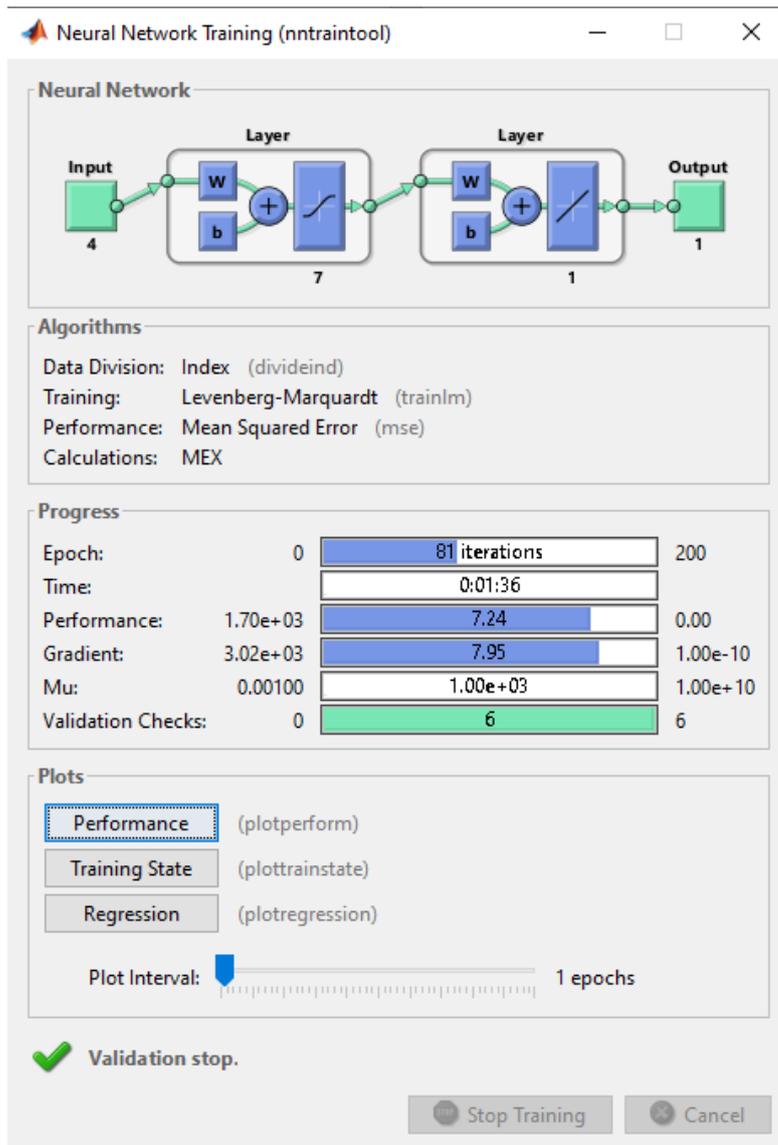


Рисунок 58 – Neural Network Training (nntraintool)

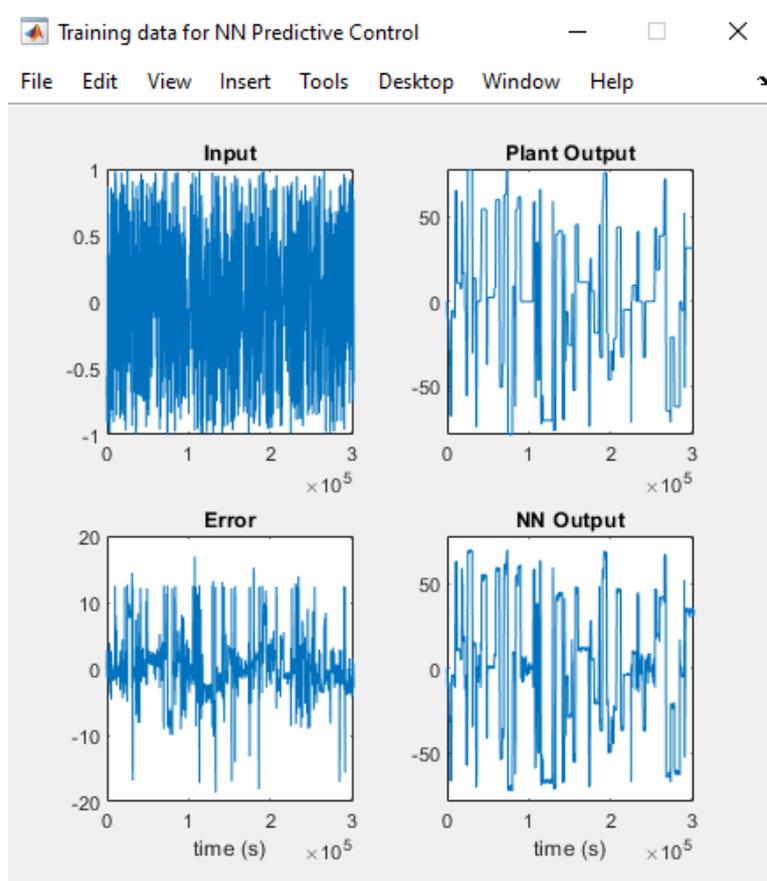


Рисунок 59 - Окно Training data for NN Predictive Control

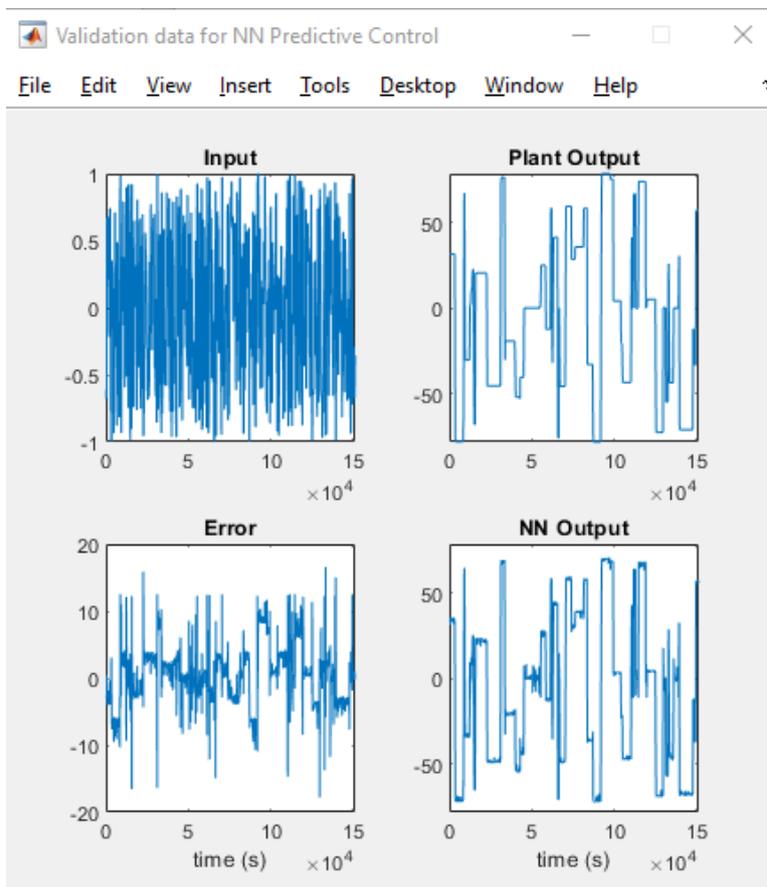


Рисунок 60 - Окно Validation data for NN Predictive Control

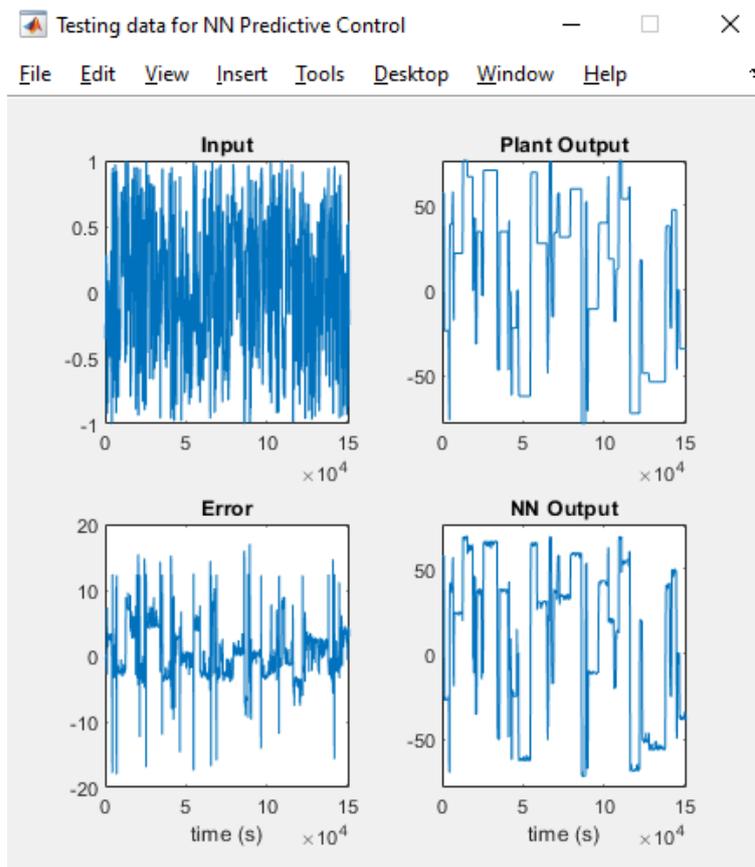


Рисунок 61 - Окно Testing data for NN Predictive Control

После обучения предиктивного регулятора NN Predictive Controller, запускаем модель, представленную на рисунке 48.

Результаты моделирования разгона электропривода до скорости вращения якоря от 0 до 50 рад/с представлены на рисунке 62.

Как видно на рисунке 62, предиктивный регулятор даже за 10 секунд не справился с задачей разогнать ДПТ электропривода до угловой скорости якоря 50 рад/с. Кроме того, в процессе разгона осуществлялся реверс двигателя, что недопустимо.

На рисунке 63 представлены результаты моделирования разгона ДПТ электропривода от 0 до 50 рад/с с различными типами регуляторов. Как видно, наилучшее быстродействие показала система управления электроприводом с классическим ПИД-регулятором (синяя кривая), но уступила по величине

перерегулирования системе с нечётким регулятором (красная кривая). Наихудшие результаты показала система с предиктивным регулятором (фиолетовая кривая), что вероятнее всего произошло из-за плохой настройки регулятора из-за недостаточно большой выборки данных для обучения ИНС, а также неоптимальной структуры и параметров ИНС. Нейросетевой регулятор (оранжевая) также нелучшим образом обеспечил управление: большое перерегулирование (34%) и нестабильная установившаяся ошибка (около 18%).

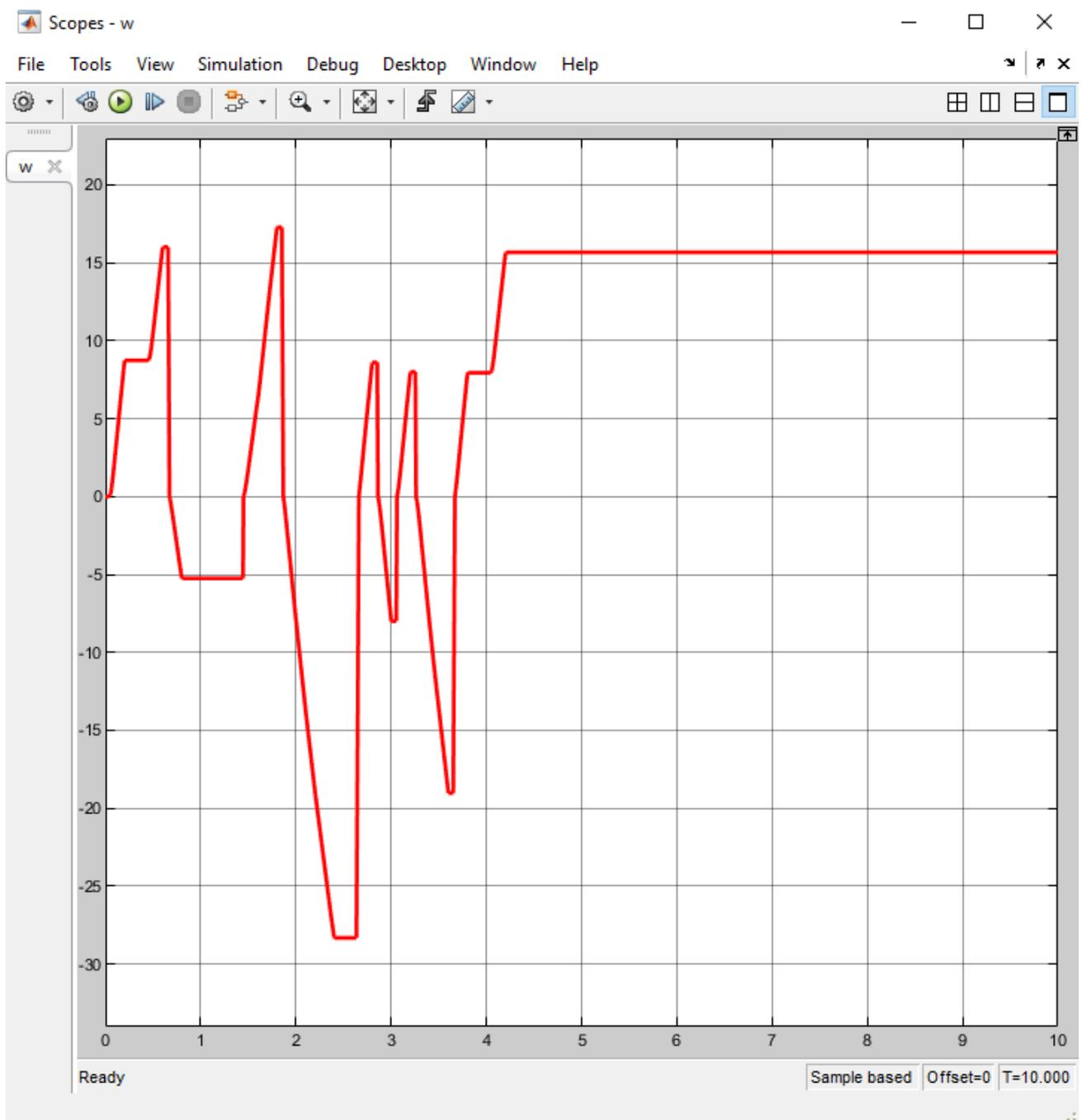


Рисунок 62 – График изменения угловой скорости якоря в двухконтурной системе подчиненного регулирования электропривода на базе ДПТ с предиктивным регулятором угловой скорости якоря

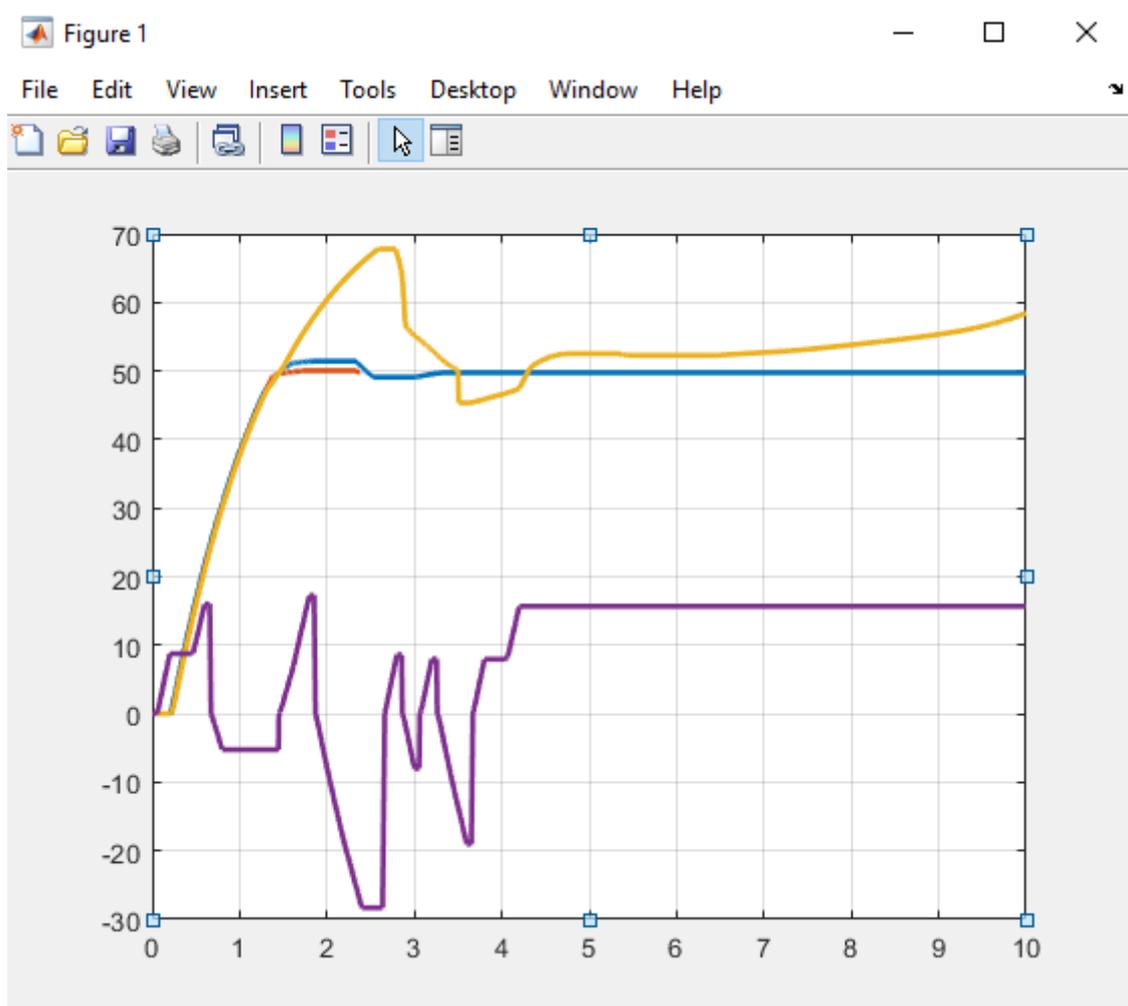


Рисунок 63 – Результаты моделирования разгона ДПТ электропривода с системой регулирования угловой скорости якоря с различными типами регуляторов: синяя кривая – классический ПИД-регулятор, красная – нечёткий, оранжевая – нейросетевой, фиолетовая – предиктивный