

ЦЕЛЕСООБРАЗНОСТЬ ПРИМЕНЕНИЯ ПРОГНОЗИРУЮЩЕГО НЕЙРОСЕТЕВОГО РЕГУЛЯТОРА В ЗАДАЧЕ ПОДДЕРЖАНИЯ УРОВНЯ ВОДЫ В РЕЗЕРВУАРЕ

Дуванов Е.С., Федянин Т.В.,

Липецкий Государственный Технический Университет, Липецк, Россия

evgenyduvanov@yandex.ru, castspin@mail.ru

Пащенко А.Ф.

Институт проблем управления им. В.А. Трапезникова РАН, Москва, Россия

paschenko_alex@mail.ru

Аннотация. В исследовании проведен сравнительный анализ результатов применения прогнозирующего нейро-регулятора, разработанного в среде MATLAB и использующего нейросеть для коррекции управляющего сигнала гидравлического лабораторного стенда, с классическим ПИ регулятором в задаче поддержания уровня воды в резервуаре. Также проведена оценка целесообразности применения данного подхода. Полученные результаты могут пролить свет на преимущества и ограничения использования нейро-регуляторов на примере гидравлического стенда.

Ключевые слова: MATLAB, Simulink, прогнозирующий нейро-регулятор, гидравлический объект управления.

Введение

Тенденция развития современных технологий управления склоняется к интеграции новейших научных разработок, что включает системы автоматического управления. В данном исследовании рассматривается перспективное направление развития систем автоматического управления, основанное на использовании нейронных сетей и систем прогнозирования.

Для более глубокого понимания и представления работы прогнозирующего нейро-регулятора необходимо изучить его основные компоненты, включая искусственные нейронные сети и системы прогнозирования. Искусственная нейронная сеть представляет собой математическую модель, построенную по аналогии с организацией и функционированием биологических нейронных сетей человеческого организма. После разработки алгоритмов обучения, такие модели находят применение в различных областях, включая прогнозирование, распознавание образов, управление и другие задачи.

Система прогнозирования основана на математической модели объекта управления, которая позволяет предсказывать изменения управляемой переменной на определенный временной интервал вперед и определять оптимальное управляющее воздействие для достижения оптимальной траектории управляемой переменной.

1. Описание объекта управления

Экспериментальные исследования проводились на лабораторном стенде «Гидравлический объект», как показано на рис. 1.

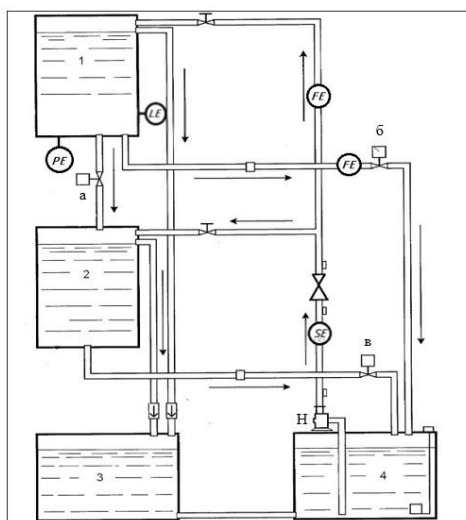


Рис. 1. Учебный стенд «Гидравлический объект»

Задачи данного объекта управления включают поддержание заданного уровня жидкости в экспериментальных резервуарах 1 и 2 путем регулирования клапанов, что достигается изменением частоты вращения вала частотного привода.

В структуре объекта управления включены следующие компоненты:

- PE – Датчик уровня давления;
- LE – Датчик кондуктометрический;
- FE – Датчик «Расходомер частотный 1»;
- FE – Датчик «Расходомер частотный 2»;
- SE – Датчик «Расходомер ультразвуковой»;
- а, б, в – задвижки, H – насос.

Описание технологического процесса объекта управления представляет собой следующее: насос (H) передает воду из резервуара 4, который соединен с резервуаром 3, в резервуар 1, где осуществляется контроль уровня воды. С помощью задвижки «б» возможно обеспечить слив воды обратно в резервуар 4. В резервуарах 1 и 2 имеются переливные каналы, направленные в резервуар 3.

Взаимодействие между управляющим сигналом, поступающим с персонального компьютера, и объектом управления осуществляется через OPC сервер, который предварительно создан на компьютере в программном обеспечении *TIA Portal*, и программируемый логический контроллер *Siemens S7-1200*.

2. Постановка задачи

Основной задачей в рамках данного исследования является поддержания определенного уровня воды в резервуаре средствами нейросетевого прогнозирующего и классического ПИ регуляторов с последующей оценкой эффективности данных методов управления при дополнительном воздействии на объект управления.

3. Описание прогнозирующего нейро-регулятора

Нейросетевое прогнозирующее управление (*Neural Network Predictive Control*) нацелено на минимизацию интегрального функционала ошибки, предсказанной на определенном горизонте прогнозирования.

$$N = \max(N_2, N_3), 0 \leq N_1 \leq N_2$$

тактов вперед

$$Q(k) = \sum_{i=N_1}^{N_2} e(k+i)^2 + \rho \sum_{i=0}^{N_2} (u(k+i) - u(k+i-1))^2.$$

Здесь e – представляет собой ошибку выхода системы, а ρ отражает влияние изменения управляющего сигнала на общий функционал стоимости Q . Для прогнозирования будущего поведения системы и вычисления ошибок используется прямой нейроэмулятор, обученный аналогично тому, как это делается при обратном распространении ошибки через прямой нейроэмулятор. Уникальность этого подхода заключается в отсутствии обучаемого нейроконтроллера. Его роль исполняет оптимизационный модуль, функционирующий в режиме реального времени и использующий методы, такие как симплекс-метод [2] или квази-Ньютоновский алгоритм [3].

Для оценки применимости нейро-регулятора в данном контексте предлагается использовать встроенный в *MATLAB* нейросетевой прогнозирующий контроллер, изображенный на рис. 2.

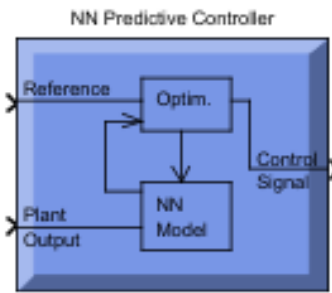


Рис. 2. Прогнозирующего нейро-регулятора

Прогнозирующая система регулирования основывается на методе уменьшения горизонта. Нейронная сеть моделирует ответ объекта на заданном временном горизонте. Прогнозы используются численным оптимизационным алгоритмом для определения управляющего сигнала, который минимизирует указанный критерий производительности в пределах заданного горизонта.

$$J = \sum_{j=N_1}^{N_2} (y_r(t+j) - y_m(t+j))^2 + \rho \sum_{j=1}^{N_u} (u'(t+j-2))^2,$$

где N_1 , N_2 и N_u определяют временные горизонты, на которых оцениваются ошибки отслеживания и шаг управления. Переменная u' представляет собой предварительный управляющий сигнал, y_r — желаемый выход, а y_m — выход сетевой модели. Параметр ρ определяет влияние, которое квадраты шага управления оказывают на индекс эффективности. Контроллер включает в себя нейросетевую модель процесса и блок оптимизации. Блок оптимизации определяет значения u' , минимизирующие J , после чего оптимальное значение u передается на вход процесса (рис. 3).

Для прогнозирования поведения объекта управления (его будущих реакций на возможные управляющие сигналы) используется нейронная сеть объекта. Нейронная сеть объекта обучается на основе расхождения между управляемой переменной (выходом объекта) y_o и выходом y_m его нейросетевой модели. Исходя из разницы между предсказанием y_m и желаемым значением y_r , а также учитывая историю управляющего воздействия u , регулятор оптимизирует управляющий сигнал u .

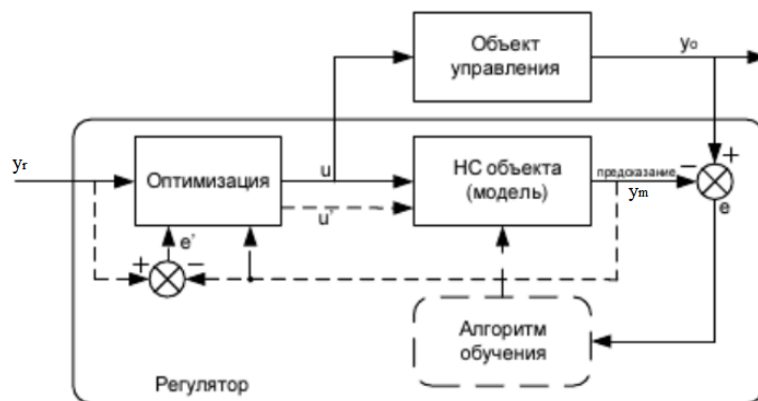


Рис. 3. Структура системы управления на основе нейро-регулятора с предсказанием

Нейронная сеть модели объекта управления использует предыдущие входные и выходные параметры объекта для прогнозирования будущих значений выхода объекта. Структура этой нейронной сети объекта управления показана на рис. 4.

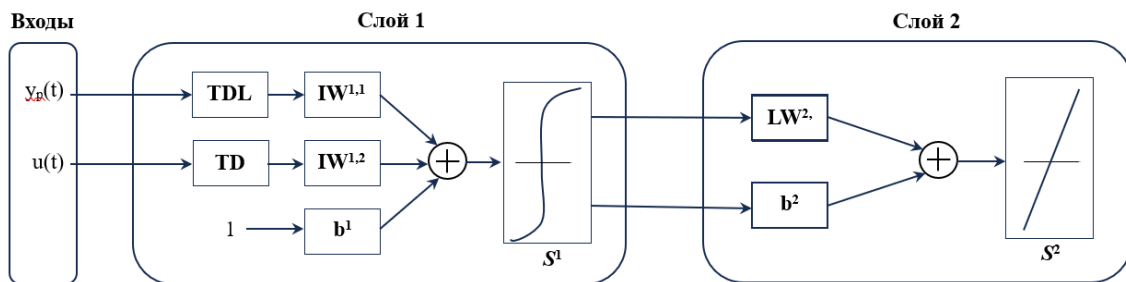


Рис. 4. Схема нейросети

Сигнал рассогласования между управляемой переменной (выходом объекта) $y_p(t)$ и входным сигналом $u(t)$ подаются на первый слой нейронной сети, где они используются для формирования табличных данных (TDL). Затем определяется матрица входных весов ($IW_{x,y}$), где x — номер слоя, а y — номер элемента слоя. Далее табличные данные подаются на входные веса и суммируются в нейроне, после чего полученные данные проходят через функцию активации.

$$a_1 = \tan \sin((IW_{x,y} y_p(t))(IW_{x,y} u(t)) + b_1).$$

Результаты с первого слоя передаются на последующие слои, где используется функция активации $\tan \sin$. Этот процесс повторяется для последующих слоев. На выходном слое данные определяются с использованием весов слоев ($LW_{x,y}$) и после выхода из нейрона активируются с помощью функции purelin .

$$a_2 = \text{purelin}(LW_{x,y} a_1 + b_2).$$

Эту сеть можно обучать автономно в пакетном режиме, используя данные, собранные в ходе работы установки.

4. Получение передаточной функции объекта

Для использования в качестве эталонной модели при регулировании и обучении нейронной сети, определим передаточную функцию объекта управления. Для этого устанавливается начальный объем воды в баке регулирования ($V = 0$ л) при отключенном насосе и применяем ступенчатое возмущение, равное максимальной частоте вращения насоса, до полного наполнения бака. На основе полученных данных получена кривая разгона наполнения резервуара водой (рис. 5).

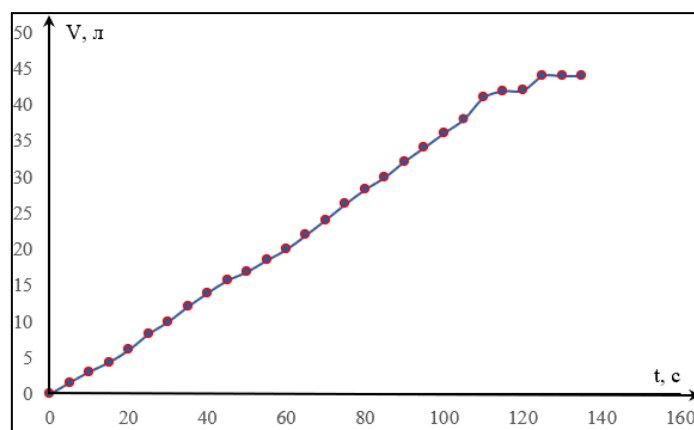


Рис. 5. Кривая разгона наполнения резервуара водой

В ходе исследования получена передаточная функции $W(s)$ с применением программного комплекса, основанного на методе площадей Симою [4, 5].

$$W(s) = \frac{44}{1426.2381s^2 + 63.6269s + 1}.$$

График переходной функции и ее аппроксимации представлена на рис. 6.

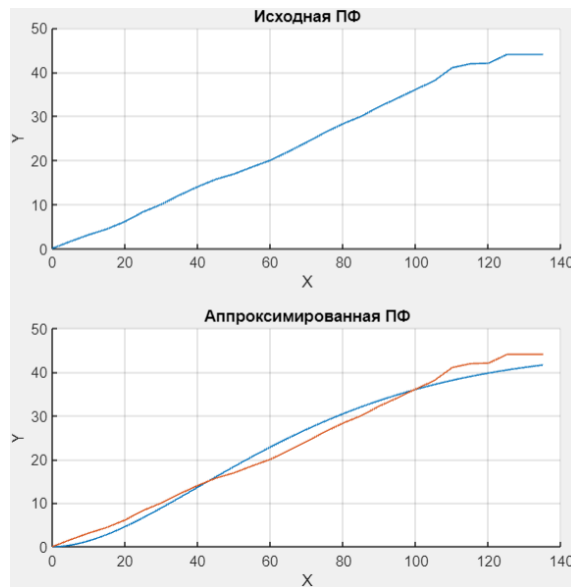


Рис. 6. График переходной функции и ее аппроксимации

5. Настройка блока прогнозирующего нейро-регулятора средствами MATLAB

Для конфигурации прогнозирующего нейро-регулятора применялось окно настройки прогнозирования, изображенное на рис. 7. В данном окне осуществляется настройка прогнозирования динамики объекта, а также задаются параметры структуры нейронной сети и параметры оптимизации.

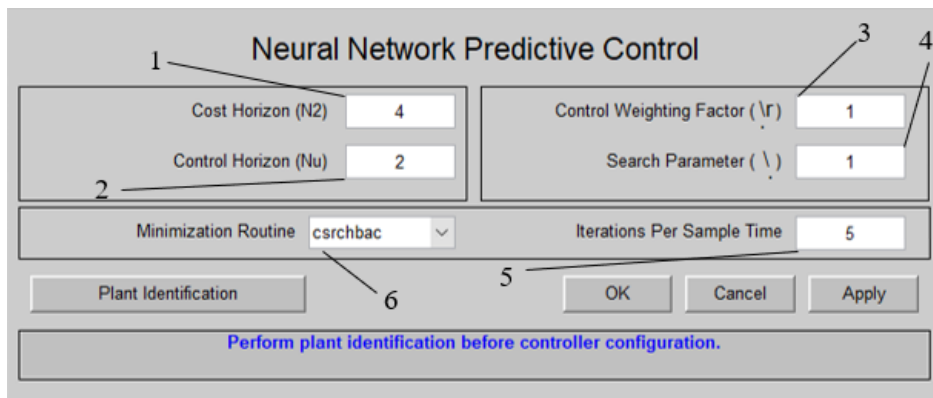


Рис. 7. Настройка NN Predictive Controller

Числовые обозначения указывают на следующее:

1. Количество временных интервалов минимизации ошибки;
2. Количество временных интервалов оценки мощности управления;
3. Коэффициент веса для составляющей мощности управления;
4. Порог одномерного поиска показателя качества;
5. Количество итераций алгоритма оптимизации за 1 такт;
6. Выбор процедуры одномерного поиска.

При нажатии кнопки *Plant identification* открывается окно идентификации согласно рис. 8, где доступна процедура настройки для обучения нейронной сети.

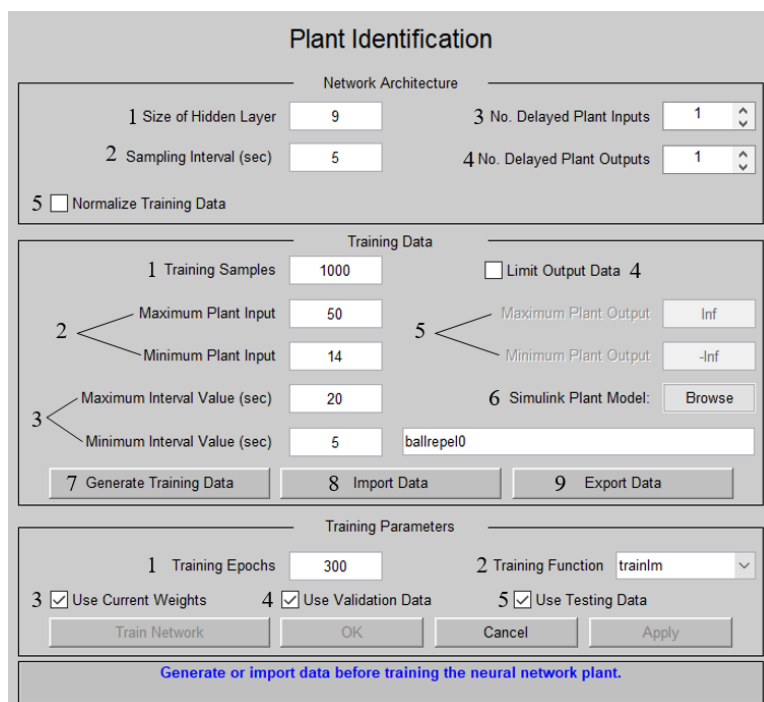


Рис. 8. Параметры управляющего воздействия на объект регулирования

В данном окне происходит разграничение на 3 области.

Первая область *Network Architecture* – задаются параметры нейронной сети для регулирования.

Числовые обозначения указывают на следующее:

1. Размер скрытого слоя (количество нейронов на первом слое нейронной сети);
2. Временной интервал сбора данных *Simulink* модели объекта;
3. Количество элементов запаздывания на входе;
4. Количество элементов запаздывания на выходе;
5. Нормирование обучающих к диапазону [0, 1].

Вторая область *Training Data* – задаются данные на обучение нейронной сети:

1. Длина выборки для обучения, контроля и тестирования;
2. Диапазон входного сигнала объекта, амплитуда формируется автоматически случайным образом;
3. Диапазон временных интервалов входного сигнала объекта, интервалы формируется автоматически случайным образом;
4. Ограничение объема входных данных;
5. Диапазон выходного сигнала во время обучения;
6. Выбор отдельно *mdl* или *slx* модели со своими портами ввода и вывода используемой для первичной обучения модели;
7. Кнопка запуска процесса обучения нервной системы;
8. Загрузка ранее созданной обучающей последовательности из workspace или mat файла данных;
9. Сохранение сгенерированных данных.

В третьей области *Training Parameters* задаются параметры обучения нейронной сети:

1. Количество итераций обучения объекта;
2. Выбор обучающей функции;
3. Использование текущих весов НС, в противном случае установятся случайным образом;
4. Использование 25% данных для контроля нейронной сети;
5. Использование 25% данных для тестирования нейронной сети во время обучения.

В *Simulink Plant Model* выбирается заранее подготовленная модель с передаточной функцией объекта, по которой нейросеть получит первичное представление о поведении объекта управления (рис. 9).

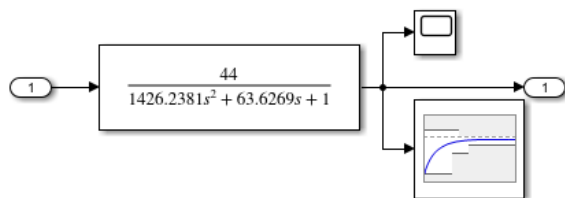


Рис. 9. Модель с ПФ для обучения нейросети

В дальнейшем, путем использования обучающей последовательности данных, осуществлялась калибровка весовых коэффициентов регулятора. Графическое отображение результатов обучения представлено на иллюстрации 10.

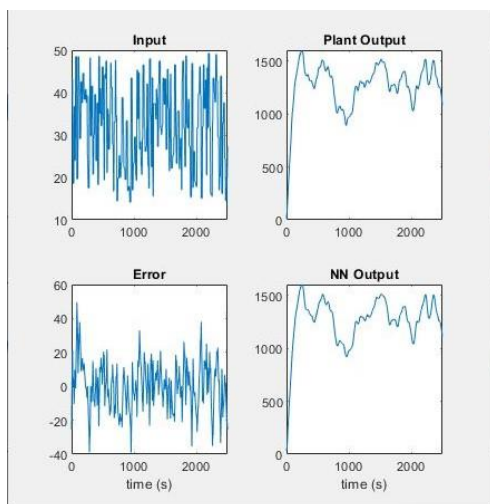


Рис. 10. Результат обучения нейросети

6. Подключение регулятора к объекту управления с OPC сервером

На рис. 11 представлена схема подключения регулятора с объектом управления средствами протокола связи OPC.

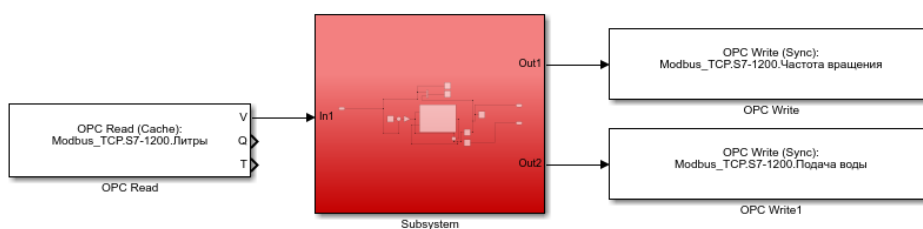


Рис. 11. Схема подключения к контроллеру с помощью OPC сервера

Схема подсистемы с регулированием *Subsystem* представлена на рис. 12.

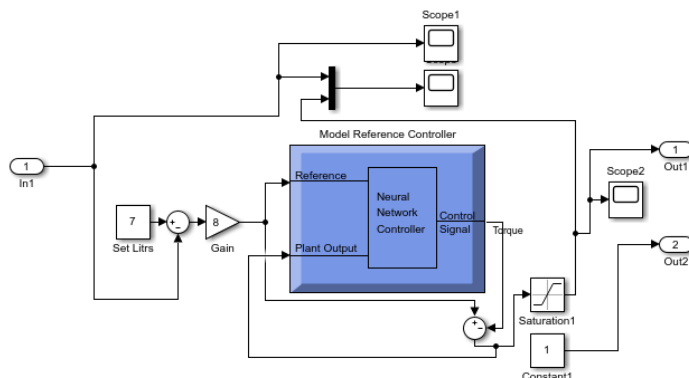


Рис. 12. Subsystem с прогнозирующим нейро-регулятором

7. Проведение активных экспериментов

Проведем серию исследований, установив значение уровня жидкости в резервуаре на уровне 7 литров и запустим программу. В результате данного эксперимента мы время переходного процесса в диапазоне от 45 до 65 секунд (с учетом физических ограничений). Дополнительные воздействия на регулятор не предусмотрены.

В окне *Scope* модели *Simulink* получим график переходного процесса регулирования объема воды в баке от зависимости изменения частоты вращения насоса от объема в резервуаре (рис. 13).

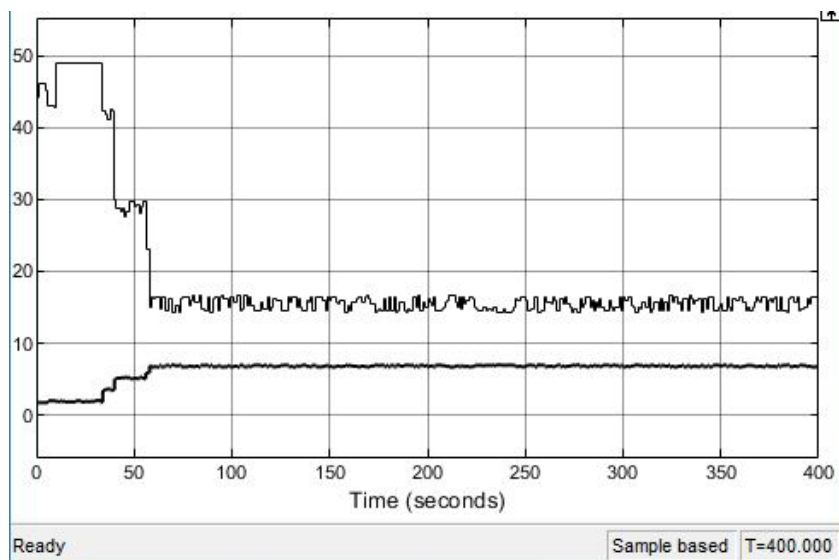


Рис. 13. Переходный процесс при регулировании с помощью прогнозирующего нейро-регулятора

Параметры качества переходного процесса занесены в таблицу 1.

Таблица 1. Результаты эксперимента

Время нарастания, с	Время переходного процесса, с	Перерегулирование, л	Характер устойчивости
54	57	0.04	Устойчива

Аналогичным образом создана схема управления ПИ регулятором. Коэффициенты регулятора рассчитаны с использованием блока оптимизации *Check Step Response Characteristics*. Значения коэффициентов $P=1.45$, $I=0.065$. На рис. 14 показана подсистема с ПИ регулятором.

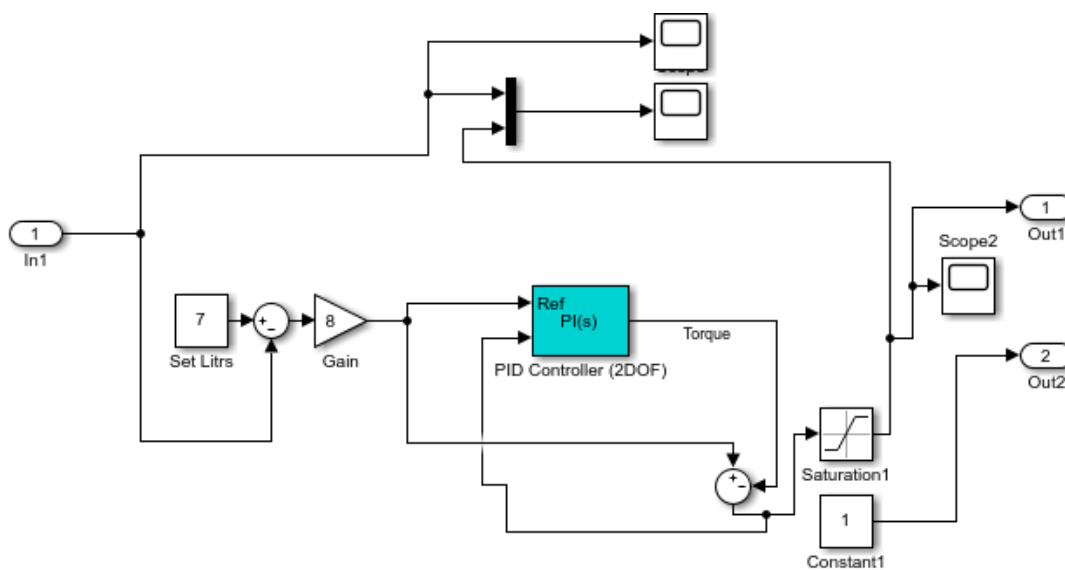


Рис. 14. Подсистема модели управления ПИ регулятора

Запустим систему. В окне *Scope* модели *Simulink* получим график переходного процесса регулирования объема воды в баке от зависимости изменения частоты вращения насоса от объема в резервуаре на основе ПИ управления (рис. 15).

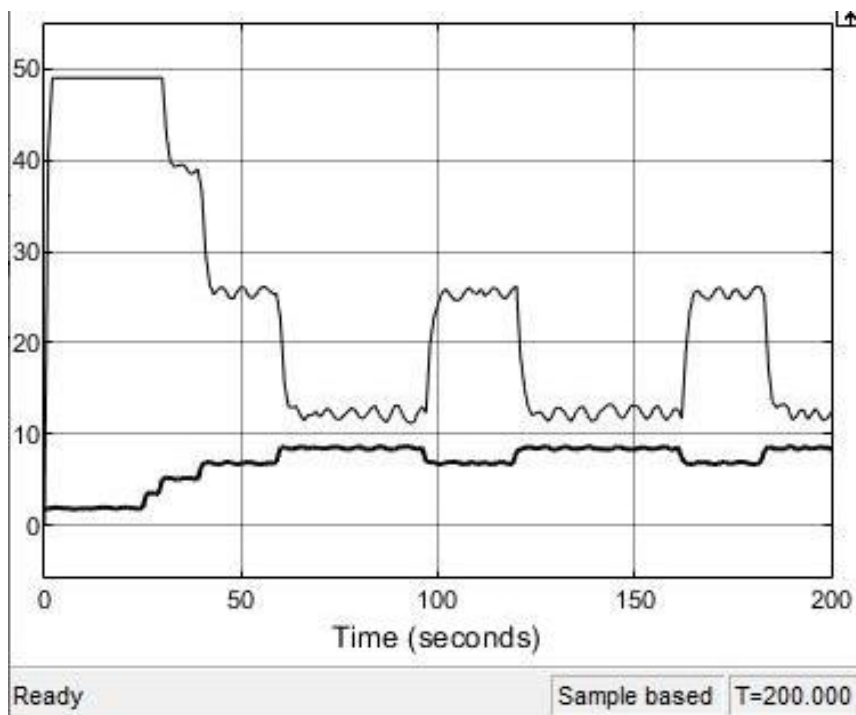


Рис. 15. Переходный процесс при регулировании с помощью ПИ регулятором

Параметры качества переходного процесса занесены в таблицу 2.

Таблица 2. Результаты эксперимента

Время нарастания, с	Время переходного процесса, с	Перерегулирование, л	Характер устойчивости
42	66	0.23	Не устойчива

Из приведенных выше графиков можно сделать вывод о неустойчивости системы управления поддержанием уровня воды в резервуаре при использовании классического ПИ регулятора. Это обусловлено тем, что классический ПИ регулятор не способен адекватно компенсировать даже минимальные накопления излишнего объема воды в резервуаре, приводящей к ложным реакциям на данные изменения.

Следующим этапом проведение эксперимента с дополнительным воздействием на объект управления. Для этого автоматически открываются задвижки «а» и «в» (рис. 1), что приводит к началу слива воды из резервуара под номером 2 в резервуар 4, откуда осуществляется дальнейшее пополнение, тем самым создавая возмущение на объект управления. Задвижки открываются на 100 секунде проведения активного эксперимента.

На рис. 16 представлен график переходного процесса при регулировании с помощью прогнозирующего нейро-регулятора и дополнительным возмущением на 100 секунде активного эксперимента.

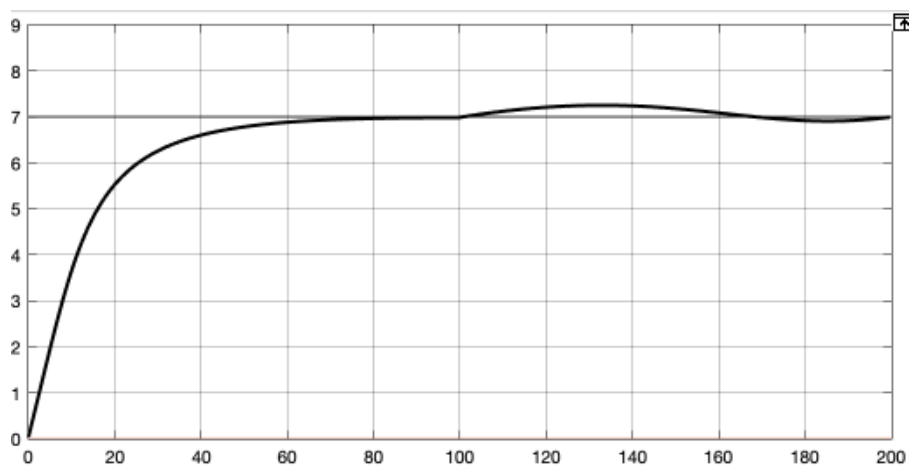


Рис. 16. Переходный процесс при регулировании с помощью прогнозирующего нейро-регулятора и дополнительного возмущения

Аналогичным образом повторим эксперимент, но с помощью классического ПИ регулятора и дополнительного возмущения на 100 секунде активного эксперимента.

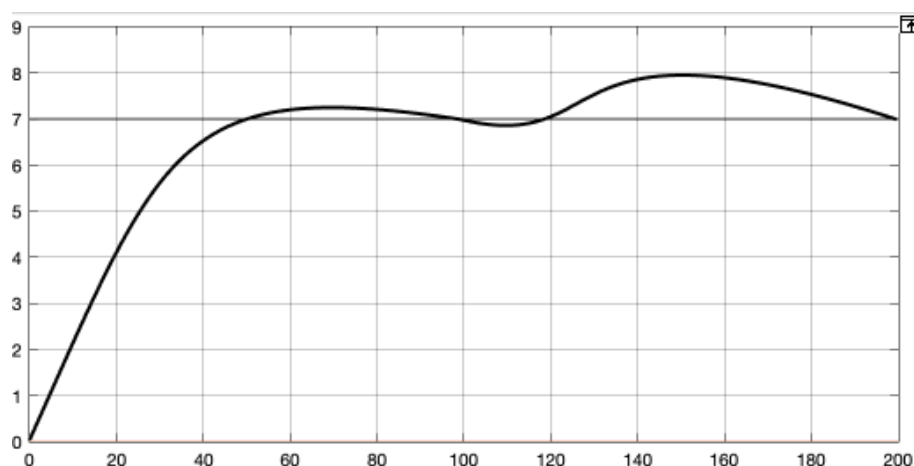


Рис. 17. Переходный процесс при регулировании с помощью классического ПИ регулятора и дополнительного возмущения

8. Заключение

В работе представлено применение прогнозирующего нейро-регулятора в задаче поддержания уровня воды в резервуаре. В ходе исследования выявлено, что применение такого алгоритма управления более эффективно в сравнении с классическим ПИ управлением, обладая большей чувствительностью и, следовательно, отзывчивостью к внешним воздействиям на систему в целом. Данное наблюдение подчеркивает перспективность применения прогнозирующих нейро-регуляторов в контексте сложных систем управления, где требуется высокая точность и быстрое реагирование на изменения внешних условий для обеспечения стабильности и оптимальной производительности объекта управления.

Однако стоит учитывать, что данный тип управления достаточно сложен в настройке и физическом внедрении в производстве, так как требует больших вычислительных ресурсов. Это связано с необходимостью обширной базы данных для обучения нейросети, а также с требованием к высокой вычислительной мощности для эффективного функционирования нейро-регулятора в реальном времени. Поэтому перед широким применением подобных методов в промышленности необходимо провести тщательный анализ затрат и преимуществ, чтобы оценить их целесообразность и эффективность в конкретных условиях производства.

Настройка данного контроллера возможна с помощью связки *MATLAB Simulink*, *OPC Toolkit* и интегрированной *SCADA* системы в объект управления.

Литература

1. Беркинблит М. Б. Нейронные сети. Экспериментальное учебное пособие. – Издательство Pubmix. com, 1993. – 254 с.
2. Takahashi Y. Adaptive predictive control of nonlinear time-varying systems using neural networks //IEEE International conference on neural networks. – IEEE, 1993. – P. 1464-1468.
3. Soloway D., Haley P. J. Neural generalized predictive control //Proceedings of the 1996 IEEE international symposium on intelligent control. – IEEE, 1998. – P. 277-282.
4. Duvanov E. S. et al. Analysis of the Technological Process of Egg Incubation and Formulation of the Control Problem //2021 3rd International Conference on Control Systems, Mathematical Modeling, Automation and Energy Efficiency (SUMMA). – IEEE, 2021. – P. 769-773.
5. Кудинов Ю. И. и др. Разработка и исследование моделей управления тепловым режимом работы инкубационного шкафа //Вестник Липецкого государственного технического университета. – 2019. – Т. 2. – №. 40. – С. 10.
6. Wang L. Model predictive control system design and implementation using MATLAB. – London: springer, 2009. – Т. 3.
7. Duvanov E. S. et al. Analysis and Synthesis of the Modified MRAC-MIT System and the MRAC-Lyapunov System //2020 2nd International Conference on Control Systems, Mathematical Modeling, Automation and Energy Efficiency (SUMMA). – IEEE, 2020. – P. 521-526.
8. Kudinov Y. I. et al. Construction and Analysis of Adaptive Fuzzy Linear Quadratic Regulator //Journal of Physics: Conference Series. – IOP Publishing, 2020. – Т. 1683. – №. 4. – С. 042065. Wang L. Model predictive control system design and implementation using MATLAB /L. Wang // Springer Science & Business Media, 2009.