

Синтез устойчивой нейросетевой замкнутой системы управления существенно нелинейным объектом с оптимизацией времени регулирования

Д. Л. Хапкин, email: dima-hapkin@ya.ru¹
С. В. Феофилов, email: svfeofilov@mail.ru¹
А. В. Козырь, email: kozyr_a_v@mail.ru¹

¹ ФГБОУ ВО «Тульский государственный университет»

***Аннотация.** В данной работе рассматривается методика синтеза устойчивой замкнутой системы управления с нейросетевым регулятором, обученным стабилизировать объект за требуемое время. Для подтверждения выполнения условий устойчивости Ляпунова используется целочисленное программирование. «Исследование выполнено за счет гранта Российского научного фонда № 23-29-00609, <https://rscf.ru/project/23-29-00609/>»*

***Ключевые слова:** Нейронные сети, системы управления, нейросетевой регулятор, функция Ляпунова, целочисленное программирование.*

Введение

В настоящее время интенсивно развивается применение искусственного интеллекта в различных областях, в том числе и в области систем автоматического управления динамическими объектами. Часто в качестве обучаемой структуры используются нейронные сети. Однако использование обучаемых структур накладывает ограничение на применение таких систем в промышленности, так как нет теоретических гарантий их устойчивой работы в различных ситуациях и режимах, в отличие от классических регуляторов. Обученные нейросетевые регуляторы обычно проверяются на качество работы тестированием [1, 2], однако такой подход позволяет выявить некоторые ошибки, а не гарантирует их полное отсутствие.

Одним из основных свойств, которое необходимо нейросетевым системам управления является их устойчивость. Для обеспечения устойчивости предлагается использовать прямой метод Ляпунова. Уже имеются работы по формированию функции Ляпунова для замкнутой нейросетевой системы управления [3–6]. В данной работе рассматривается синтез нейросетевого регулятора для управления объектом с нелинейностью типа ограничения и гарантирующий

устойчивость замкнутой системы управления с помощью формирования функции Ляпунова, основанной на нейронной сети.

1. Система управления

В работе рассматривается замкнутая дискретная система управления. Структура представлена на рис. 1.

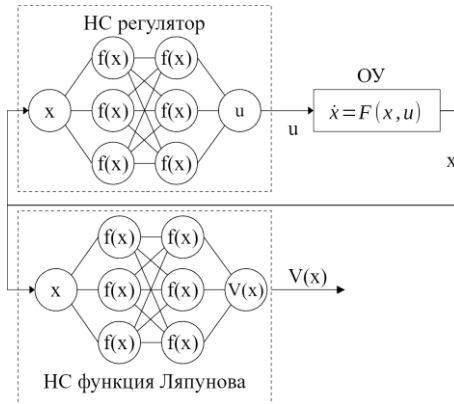


Рис. 1. Структура замкнутой системы управления

Регулятор основан на нейронной сети прямого распространения:

$$u_t(x_t) = \varphi_r(x_t) - \varphi_r(x^*) + u^*, \quad (1)$$

где φ_r – нейронная сеть прямого распространения, x_t – вектор состояния в момент времени t , x^* – точка устойчивости, u^* – сигнал управления в точке устойчивости.

Для гарантии устойчивости системы управления находится функция Ляпунова. Как известно, по второму методу Ляпунова система является асимптотически устойчивой в локальной области, если имеется такая функция, называемая функцией Ляпунова $v(x)$, которая в этой области является положительной, а ее производная отрицательной, кроме точки устойчивости, в которой она равна нулю.

$$V(x_t) > 0, \quad x_t \in S, \quad x_t \neq x^*; \quad (2)$$

$$V(x_{t+1}) - V(x_t) \leq -\varepsilon_2 V(x_t), \quad x_t \in S, \quad x_t \neq x^*; \quad (3)$$

$$V(x^*) = 0; \quad (4)$$

где s – область, в которой система должна быть устойчива, ε_1 – коэффициент от 0 до 1.

В качестве кандидата в функцию Ляпунова предлагается использовать функцию, основанную на нейронной сети прямого распространения. Вид данной функции (5):

$$v(x_t) = \phi_V(x_t) - \phi_V(x^*) + \|R(x_t - x^*)\|, \quad (5)$$

где ϕ_V – нейронная сеть, являющаяся частью функции Ляпунова, R – матрица с полным рангом столбца $\|R(x_t - x^*)\|$ – это 1-я норма вектора.

Уравнение (5) гарантирует, что $v(x^*) = 0$. Добавление $\|R(x_t - x^*)\|$ помогает выполнению условия (2).

В результате обучения на выполнение критериев устойчивости по Ляпунову (2-4) будут получены нейросетевые регулятор и функция Ляпунова, гарантирующая устойчивость системы.

Для обучения используется метод обратного распространения ошибки [7], который позволяет рассчитать градиент для каждого весового коэффициента нейронной сети. Поэтому для обучения нейрорегулятора необходимо создать структуру, имитирующую объект управления, по которой можно рассчитать градиент. Для этого предлагается использовать нейроимитатор объекта управления:

$$x_{t+1} = \phi_{\text{dyn}}(x_t, u_t), \quad (6)$$

где ϕ_{dyn} – нейронная сеть прямого распространения, обученная имитировать объект управления.

Для объектов с нелинейностью типа ограничение нужно правильно составить структуру нейроимитатора. Некоторые ограничения описаны в работе [8], которые, в общем случае, заключаются в том, что для имитации таких объектов в качестве функций активаций должны быть выбраны ReLU [9].

2. Контрпримеры

Для получения функции Ляпунова нужно подтвердить, что в требуемой области фазового пространства не нарушаются условия устойчивости. Для этого используется целочисленное программирование, с помощью которого в ограниченной области можно гарантированно найти оптимальное значение целевой функции. Для обеспечения выполнения условий Ляпунова будем использовать две целевые функции для проверки условий 2 и 3 соответственно:

$$\max_{x \in S} (\varepsilon_2 \left\| R(x_t - x^*) \right\|_1 - V(x_t)); \quad (7)$$

$$\max_{x_t \in S} (V(x_{t+1}) + (\varepsilon_2 - 1)V(x_t)). \quad (8)$$

Т.е. с помощью оптимизации целевых функций (7) и (8) находятся точки фазового пространства, контрпримеры, которые максимально не удовлетворяют условиям устойчивости. Далее эти точки используются для обучения функции Ляпунова и нейрорегулятора. Если найденные точки будут удовлетворять условиям устойчивости, то тогда система устойчива и функция Ляпунова сформирована.

Для того, чтобы применить целочисленное программирование, нужно все нейронные сети привести к системе неравенств. Для этого в нейронных сетях в качестве функции активации должна использоваться функция leakyReLU (9).

$$\text{leakyReLU}(x) = \begin{cases} x, & x \geq 0; \\ cx, & x < 0; \end{cases} \quad (9)$$

где c – значение коэффициента отрицательной части leakyReLU.

Описание leakyReLU в виде системы неравенств:

$$\begin{cases} z \geq y, & z \geq cy, \\ z \leq cy + (1 - c) \cdot y_{ub} \beta, \\ z \leq y + (1 - c) \cdot y_{lb} \cdot (\beta - 1), \\ \beta \in [0; 1], & z \in R, \end{cases} \quad (10)$$

где z – выход leakyReLU, β – бинарная переменная ($\beta = 0$ – для отрицательной, $\beta = 1$ – для положительной), y – значение входа leakyReLU, y_{ub} и y_{lb} – ограничения на входную переменную. Для расчета y_{ub} и y_{lb} для каждого нейрона используется интервальная арифметика [10].

Описывая каждый нейрон с функцией активацией leakyReLU, как (9), а нейроны без функций активации соответствующим выражением, то сформируется система равенств и неравенств, которую нужно использовать в задаче целочисленного программирования для поиска контрпримеров.

3. Обучение

На собранных контрпримерах выполняется обучение нейрорегулятора и нейросетевой функции Ляпунова, при этом веса

нейроимитатора не обновляются. Для этого найденные значения используются в качестве начальных условий и система, состоящая из указанных нейронных сетей, моделируется на один дискретный шаг. Полученные результаты используются для обучения на основе следующих функций потерь:

$$\eta_1 = \max(\varepsilon_1 \|R(x_t - x^*)\|_1 - V(x_t), 0); \quad (11)$$

$$\eta_2 = \max(V(x_{t+1}) + (\varepsilon_2 - 1) * V(x_t), 0). \quad (12)$$

После шага изменения весов процедура поиска контрпримеров повторяется, а найденные значения добавляются уже к имеющимся для расширения тренировочной выборки.

Однако, так как при обучении нет явного контроля времени регулирования, то предлагается добавить дополнительный критерий. Для его выполнения система моделируется не на один шаг, а на время, превышающее требуемое время регулирования. Во всех моментах, которые находятся после требуемого времени, вычисляется ошибка регулирования, которая используется для обучения нейрорегулятора. При этом используется метод обратного распространения ошибки сквозь время [11], как делается это в рекуррентных нейронных сетях. График формирования значений критерия времени регулирования представлен на рис. 2.

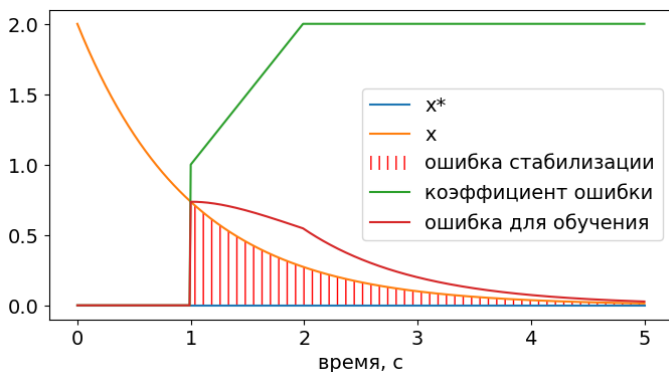


Рис. 2. График формирования ошибки для обучения нейрорегулятора на выполнение требований по времени регулирования (на графике 1 сек.)

Для каждого момента времени критерий выглядит так:

$$\eta_3 = \sum \left((x^* - x_t) \cdot k \right)^2. \quad (13)$$

Таким образом общая функция потерь выглядит так:

$$loss = \| \eta_1 \| + \| \eta_2 \| + \eta_3. \quad (14)$$

Общая схема обучения представлена на рис. 3.

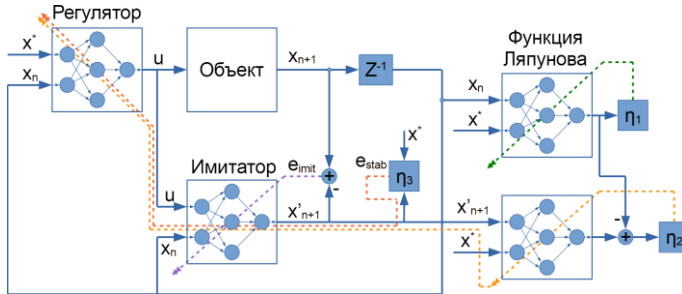


Рис. 3. Схема синтеза нейросетевого регулятора по предложенной методике

Однако при таком подходе необходимо адекватно выбирать требуемое время регулирования, так как при очень маленьком значении система может плохо обучиться. Это связано с тем, что при наличии больших ошибок от критерия η_3 значения ошибок по устойчивости (критерии η_1 и η_2) могут быть незначительными. Также для каждого критерия можно ввести коэффициенты учета их влияния на общую оценку по усмотрению разработчика нейрорегулятора.

4. Пример

В качестве примера рассмотрим упрощенную модель гидропривода (рис. 4).

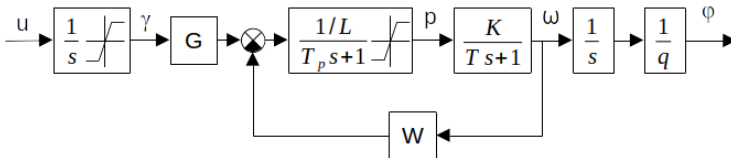


Рис. 4. Базовая модель гидропривода

На рис. 4 γ – угол поворота люльки гидронасоса, p – перепад давления в гидромоторе, ω – скорость вращения выходного вала

гидромотора, ϕ – угол поворота вала после редуктора, G – коэффициент усиления насоса регулируемой производительности по расходу, L – суммарный коэффициент утечек жидкости; $T_p = L * E / V$, где E – приведенное значение модуля объемной упругости жидкости, V – объём жидкости в магистрали высокого давления. В структурной схеме учитываются присущие гидроприводу ограничитель на угол поворота люльки насоса: $|\gamma| \leq D_1$ и ограничитель (в данной структурной схеме – безынерционный) перепада давления масла в силовых гидромагистралях: $|p| \leq D_2$.

В качестве нейросетевого имитатора выбрана следующая структура: на входе 5 нейронов (для сигнала управления объектом и вектора состояния), в первом скрытом слое 4 нейрона без функций активации, во втором слое 12 нейронов с функцией активации ReLU и с весами смещения, в выходном слое 4 нейрона.

В качестве структуры для нейрорегулятора используется нейронная сеть прямого распространения: на вход которого поступает вектор состояния объекта управления и вектор точки равновесия, получаемый из производных управляющего сигнала. В скрытом слое 12 нейронов, с функциями активации leakyReLU, на выходе один нейрон формирующий сигнал управления объектом.

Функция Ляпунова представлена 4-мя нейронами на входе, двумя скрытыми слоями с функцией активации leakyReLU и одним нейроном на выходе. Скрытые слои имеют по 12 нейронов в каждом.

Для стабильного обучения нейрорегулятора использовалось его предварительное обучение на LQR [12]. Этот фильтр является линейным, и был спроектирован для модели гидропривода без учета ограничений. Соответственно, качество управления при введении ограничений является не самым лучшим.

Затем выполняется обучение по предложенной методике. Требуемое время регулирования установлено в 1,5 сек. Оно было выбрано на основе параметров объекта управления и из-за ограничений, накладываемых на сигнал управления объектом и ограничителей в системе.

На рис. 5 показан переходный процесс на ступенчатый сигнал, по которому видно, что система с нейросетевым регулятором не имеет перерегулирования и стабилизируется за время, близкое к заданному. Система с LQR стабилизируется за тоже время, однако имеет перерегулирование.

На рис. 6 сравнивается система с нейросетевым регулятором и LQR. Можно заметить, что нейросетевой регулятор имеет меньшую ошибку.

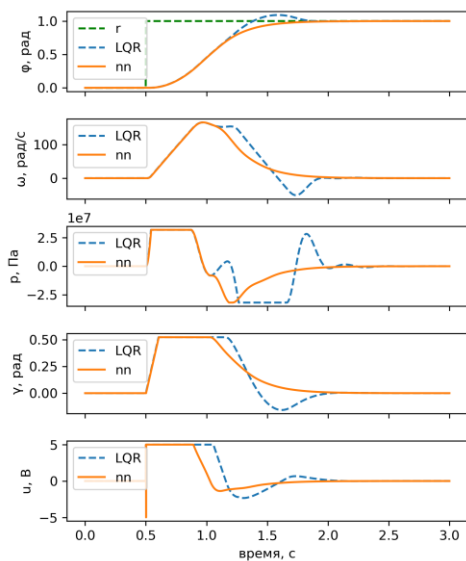


Рис. 5. Переходный процесс на ступенчатый сигнал

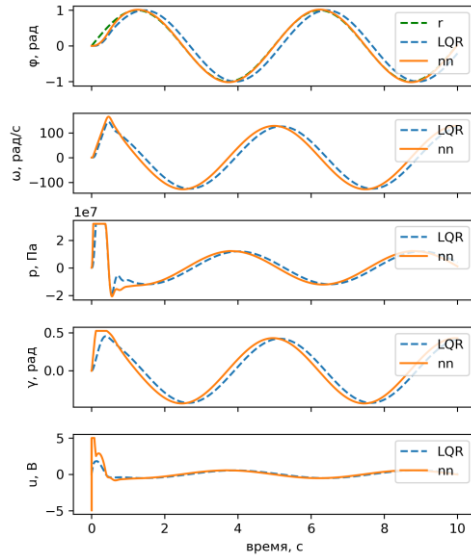


Рис. 6. Реакция системы на сигнал гармонический сигнал управления

Таким образом получаем регулятор, способный формировать нелинейный закон управления, позволяющий стабилизировать систему за определенное время регулирования, и функцию Ляпунова, гарантирующую асимптотическую устойчивость в локальной области.

Закключение

В работе предложена методика, позволяющая синтезировать нейросетевой регулятор, который обеспечивает выполнение требований по времени регулирования. При этом также формируется функция Ляпунова на основе нейронной сети, гарантирующая устойчивость замкнутой системы в локальной области путем подтверждения отсутствия точек фазового пространства, которые не удовлетворяют условиям устойчивости по Ляпунову.

Список литературы

1. Differentiable Convex Optimization Layers / A. Agrawal [et al.] // 33rd Conference on Neural Information Processing Systems (NeurIPS 2019). – Vancouver, Canada, 2019.
2. Verification of Binarized Neural Networks via Inter-neuron Factoring / С.-Н. Cheng [et al.] // Verified Software. Theories, Tools, and

Experiments : Lecture Notes in Computer Science / eds. R. Piskac, P. Rümmer. – Cham: Springer International Publishing, 2018. – P. 279-290.

3. Khapkin D.L. Study of neural network control stability based on mixed linear integer programming / D.L. Khapkin, S.V. Feofilov, A.V. Kozyr // 4th International Conference on Control Systems, Mathematical Modeling, Automation and Energy Efficiency (SUMMA). – Lipetsk: IEEE, 2022. – С. 326-329.

4. Counter-example guided synthesis of neural network Lyapunov functions for piecewise linear systems / H. Dai [и др.]. – 2020.

5. Lyapunov-stable neural-network control / H. Dai [и др.] arXiv:2109.14152 [cs, eess]. – arXiv, 2021.

6. Learning Lyapunov Functions for Hybrid Systems / S. Chen [и др.] // 2021 55th Annual Conference on Information Sciences and Systems (CISS) 2021 55th Annual Conference on Information Sciences and Systems (CISS). – Baltimore, MD, USA: IEEE, 2021.

7. Фефилов С.В. Анализ методов обучения искусственной нейронной сети / С.В. Фефилов, Д.Л. Хапкин // Сборник конференции XVIII Всероссийско научно-технической конференции студентов, магистрантов, аспирантов и молодых ученых «Техника XXI века глазами молодых ученых и специалистов». – Тула: изд. ТулГУ, 2020. – С. 88-96.

8. Фефилов С.В. Базовый синтез нейросетевых регуляторов для нелинейных следящих систем управления / С.В. Фефилов, Д.Л. Хапкин. – ООО «Вэлборн», 2021. – С. 1314-1322.

9. Maas A.L. Rectifier nonlinearities improve neural network acoustic models / A.L. Maas, A.Y. Hannun, A.Y. Ng // Proc. icml. – Atlanta, Georgia, USA, 2013. – Т. 30. – С. 3.

10. Wong E. Provable defenses against adversarial examples via the convex outer adversarial polytope / E. Wong, Z. Kolter // International Conference on Machine Learning. – PMLR, 2018. – С. 5286-5295.

11. Werbos P.J. Backpropagation through time: what it does and how to do it / P.J. Werbos // Proceedings of the IEEE. – 1990. – Т. 78. – № 10. – С. 1550-1560.

12. Anderson B.D.O. Linear Optimal Control / B.D.O. Anderson, J.B. Moore, B.P. Molinari // IEEE Transactions on Systems, Man, and Cybernetics. – 1972. – Т. SMC-2. – № 4. – 559 с.