

## ИССЛЕДОВАНИЕ ПРЯМОГО НЕЙРОУПРАВЛЕНИЯ АСИНХРОННЫМ ДВИГАТЕЛЕМ С ОБУЧЕНИЕМ В ЦИФРОВОМ ДВОЙНИКЕ

**Аннотация.** В работе исследуется подход прямого нейросетевого управления асинхронным электродвигателем на основе алгоритмов обучения с подкреплением. Обучение управляющей политики проводится в среде цифрового двойника без использования априорной математической модели двигателя. Предложена структура системы MIC AI, в которой нейросетевой агент формирует управляющие воздействия непосредственно на инвертор. Работоспособность подхода оценена в серии численных экспериментов и сопоставлена с классическим векторным управлением. Показано, что в текущих условиях моделирования нейросетевое управление обеспечивает снижение токовой нагрузки и сопоставимое качество регулирования скорости при увеличении входной мощности.

**Ключевые слова:** асинхронный электродвигатель; электропривод; обучение с подкреплением; цифровой двойник; нейросетевое управление; векторное управление.

Современные электроприводы предъявляют высокие требования к точности регулирования скорости и момента, энергоснабжению и адаптивности к изменениям параметров. Классические системы управления, такие как пропорционально-интегральные регуляторы в контуре полевого ориентированного управления (Field Oriented Control, FOC), широко применяются благодаря простоте и надёжности. Однако они зависят от точности математической модели двигателя и требуют перенастройки при изменении условий работы [1][2]. Таким образом, возникает потребность в новых подходах, обеспечивающих адаптивное управление без точной модели и с оптимизацией сложных критериев.

Методы искусственного интеллекта находят всё большее применение в электроприводе. В последние годы особое внимание уделяется алгоритмам обучения с подкреплением (Reinforcement Learning, RL) для управления электродвигателями. RL-агент способен, взаимодействуя с динамической системой, самостоятельно обучаться оптимальным действиям на основе сигнала вознаграждения, минуя необходимость явной модели объекта [3][4]. В контексте электропривода это означает, что алгоритм может напрямую регулировать напряжения или команды инвертора, наблюдая за состоянием двигателя и стремясь максимизировать заданный критерий качества [5]. Глубокое обучение с подкреплением (Deep RL) с использованием нейросетей для аппроксимации стратегий еще более расширило возможности таких методов, позволив работать с нелинейными многофакторными объектами, как асинхронные двигатели, и оптимизировать сложные показатели (энергопотребление, нагрев, износ и т.д.) [6][7].

Научная новизна предлагаемого подхода заключается в разработке системы самообучающегося управления асинхронным электродвигателем МИС AI (Motor Intelligence Control with Artificial Intelligence), где агент с глубоким обучением с подкреплением замещает классические регуляторы. В отличие от традиционного FOC, предлагается прямое управление в осях  $d-q$  с выдачей управляющих воздействий непосредственно на инвертор (напряжения  $v_d$  и  $v_q$ ) без использования каскада PI-регуляторов для токов. Алгоритм обучается on-line в среде цифрового двойника двигателя, что обеспечивает безопасность и эффективность обучения, а затем знание агента переносится на встроенное устройство управления посредством техники дистилляции (сжатия) модели. Такой комплексный подход объединяет преимущества цифровых двойников и современных методов RL, предоставляя адаптивное управление без необходимости точной модели, и ориентирован на реализацию в микроконтроллере, близкую к реальному времени.

На рисунке 1 показано сопоставление классических, модельно-ориентированных и интеллектуальных методов управления электроприводом: классическое векторное управление с ПИ-регуляторами требует точной математической модели и ручной настройки параметров, обеспечивая высокую надёжность при ограниченной адаптивности; модельно-ориентированные методы, такие как модельно-предиктивное управление, используют оптимизацию на основе модели в реальном времени, что повышает качество управления, но требует значительных вычислительных ресурсов; методы с использованием искусственного интеллекта основаны на обучении по данным и накопленному опыту, обеспечивая адаптивность и способность к самообучению без необходимости точного априорного описания объекта управления.



Рисунок 1 – Сравнение подходов к управлению электроприводом

Применение алгоритмов обучения с подкреплением непосредственно для управления асинхронным электродвигателем – относительно новый шаг в силовой электронике и электроприводах. В традиционных системах управления, чтобы учесть нелинейности и неопределенности, часто используют сложные наблюдатели или адаптивные контуры. RL же позволяет создать моделенно-независимый регулятор: агент обучается правильным действиям методом проб и ошибок, получая вознаграждение за достижение целей управления [4]. Это устраняет необходимость ручной настройки регуляторов на основе модели и повышает динамическую адаптивность к изменяющимся условиям.

Цель настоящей работы – продемонстрировать работоспособность предлагаемой методики обучения системы управления асинхронным электроприводом на цифровом двойнике. Необходимо показать, что RL-алгоритм способен обучаться эффективному управлению асинхронным двигателем в среде моделирования, достигая заданных целей по точности и энергоэффективности. Достижение этой цели подтвердит применимость метода и подготовит почву для дальнейшего переноса обученного агента на реальный привод.

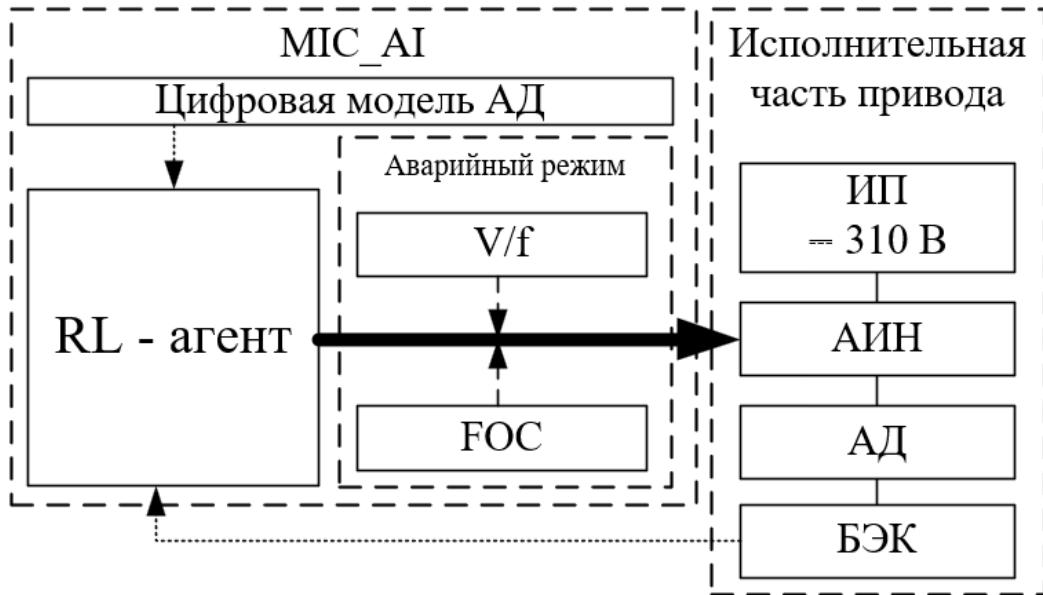


Рисунок 2 – Структура прямого нейросетевого управления асинхронным двигателем в системе MIC AI

В проекте MIC AI RL-агент выполняет роль основного управляющего элемента, формирующего воздействия «с нуля», а не корректирующей надстройки над классическим контуром. На вход агент получает наблюдаемые величины состояния привода – например, текущую угловую

скорость  $\omega(t)$ , токи статора  $i_d(t), i_q(t)$  (в осях d–q), ошибки  $e_\omega = \omega_{\text{зад}} - \omega$  и др. На основе состояния  $s(t_k)$  нейросетевой агент вычисляет действие  $a(t_k)$ . В нашем случае действие является непрерывным:  $a(t_k) = (v_d, v_q)$ , где  $v_d$  и  $v_q$  – целевые значения напряжений для инвертора. Эти напряжения подаются на ШИМ-инвертор, который генерирует трёхфазное напряжение  $V_{abc}$  на выводах двигателя. Таким образом, формируется замкнутый контур: агент – инвертор – двигатель – датчики – агент, представленный на рисунке 2. В отличие от классической иерархии (скорость – ток – напряжение), здесь нейросеть решает задачу оптимального управления за один шаг. Алгоритм сам выявляет законы управления, например связь между током возбуждения и электромагнитным моментом, необходимые для устойчивого управления асинхронным двигателем – вместо явного задания этих зависимостей, как в уравнениях FOC. Фактически, агент RL самостоятельно учится принципам векторного управления: оптимальная стратегия, выработанная в ходе обучения, включает управление током по оси  $d$  для установления магнитного поля и током по оси  $q$  для регулирования момента – эти закономерности нейросеть открывает сама, получая вознаграждение за быстрый и экономичный выход на режим.

Обучение RL-агента требует множества итераций взаимодействия с системой, что затруднительно проводить напрямую на физическом оборудовании (из-за риска повреждения, больших временных затрат и отсутствия градиентной информации). Поэтому используется высокоточный цифровой двойник асинхронного двигателя. Цифровой двойник представляет собой программную модель двигателя и силового инвертора, максимально точно воспроизводящую электромеханические процессы реального привода. В отличие от методов моделирования на основе конечных элементов, правильно настроенный цифровой двойник работает в режиме реального времени без значимой потери точности. Это позволяет проводить быстрые многократные симуляции: агент MIC AI обучается в виртуальной среде, проходя тысячи эпизодов разгона, торможения и стабилизации двигателя. На каждой итерации среда возвращает значение вознаграждения, сконструированного на основе целей управления. В рассматриваемом проекте критерий (функция награды) включает две составляющие – за точность поддержания заданной скорости и за экономичность режима (минимизация потерь/потребляемой мощности). Благодаря такому двухцелевому критерию агент поощряется не только за быстрое достижение требуемой скорости, но и за снижение потерь энергии. В результате обучения на цифровом двойнике алгоритм MIC AI формирует политику управления, оптимизирующую компромисс между динамическим качеством и энергоэффективностью – подобный компромисс трудно получить методами классического оптимального управления.

Использование цифрового двойника делает отладку алгоритма безопасной: любые неустойчивые или экстремальные действия агента на ранних этапах обучения не приводят к повреждению оборудования. Кроме того, с помощью цифрового двойника можно варьировать параметры системы (момент инерции, нагрузку, параметры двигателя) и проверять робастность обученной политики к разбросу параметров. Отметим также проблему sim-to-real – рассогласование модели и реальности: политика, обученная в симуляции, может работать хуже на реальном объекте из-за несовершенства модели. Для смягчения этого эффекта планируется использовать приёмы domain randomization – внесение случайных разбросов в модель при обучении агента, чтобы повысить его способность обобщать опыт. Тем не менее, исследования показывают, что правильно настроенные алгоритмы RL способны адаптироваться к вариациям системы, и грамотный выбор структуры вознаграждения и архитектуры агента повышает переносимость поведения алгоритма на реальный привод [8][9][10].

Для обучения агента использовались современные алгоритмы глубокого RL, предназначенные для задач с непрерывным пространством действий. В частности, применялся метод Twin Delayed DDPG (TD3) и сравнительно – Proximal Policy Optimization (PPO). Эти алгоритмы на каждом шаге симуляции обновляют параметры нейросети агента, стремясь увеличить суммарное дисконтируемое вознаграждение за эпизод. Обучение проводилось online – агент взаимодействовал с цифровым двойником в ходе множества эпизодов, постепенно улучшая стратегию. Эпизоды моделировали стандартные рабочие циклы привода: разгон до требуемой скорости, удержание нагрузки, торможение. По мере накопления опыта агент уменьшал ошибку слежения за скоростью и энергопотребление, приближаясь к оптимальной политике по заданному критерию.

После завершения обучения нейросетевая политика извлекается и оптимизируется для внедрения на встроенной платформе. В проекте MIC AI для этого применяется методика policy distillation – передача знаний от большой модели (teacher) к компактной модели (student). Обученный в симуляции агент представляет собой глубокую нейросеть значительного размера (с большим числом параметров для обеспечения высокой обобщающей способности). Выполнение такой модели в режиме реального времени на ресурсно-ограниченном микроконтроллере затруднительно. Поэтому используется дистилляция знаний: небольшая нейросеть обучается имитировать поведение большой, минимизируя, например, расхождение в действиях при одинаковых входных состояниях. Дистилляция позволяет сжать модель (уменьшить число параметров) примерно на порядок при сохранении ~90% качества исходной политики. В контексте RL это означает, что сложный агент (с большим числом слоёв и нейронов) "учит" упрощённого агента воспроизводить стратегию управления. Обучение имитацией проводится на заранее накопленных траекториях состояний и

действий: компактная сеть настраивается так, чтобы выдавать такие же управляющие воздействия, как исходный агент, при тех же входных данных среды. В результате получается облегчённая политика, пригодная для выполнения на микроконтроллере в реальном времени.

Обучение проводится онлайн на более мощном вычислителе, но итоговая инференс-модель приводится к форме, пригодной для микроконтроллера. Такой подход обеспечивает автономность системы МС AI: после этапа лабораторного обучения «интеллектуальный» контроллер можно загрузить во встроенный блок управления привода и эксплуатировать подобно обычному частотному преобразователю, без постоянного подключения к внешнему компьютеру.

В системе МС AI управление асинхронным электроприводом реализуется на основе нейросетевого алгоритма обучения с подкреплением, который выполняет функцию основного регулятора. Обучение алгоритма проводится в виртуальной среде, включающей цифровой двойник асинхронного двигателя и модель силового инвертора. В процессе обучения RL-агент взаимодействует с моделью привода, получая измеряемые параметры состояния, такие как скорость вращения и токи статора, и формируя управляющие воздействия для инвертора. Качество управления оценивается по комплексному критерию, учитывающему точность регулирования скорости и энергетическую эффективность режима. В результате многократных эпизодов обучения агент формирует стратегию управления, обеспечивающую компромисс между динамическими и энергетическими показателями без использования априорной математической модели двигателя.

После завершения этапа обучения нейросетевая модель подвергается процедурам оптимизации и сжатия и может быть реализована в виде компактного контроллера, предназначенного для работы на встроенной вычислительной платформе. В режиме эксплуатации контроллер использует измеряемые сигналы привода для формирования управляющих воздействий на силовой инвертор, при этом структура силовой части и стандартные защитные функции сохраняются. Существенной особенностью предлагаемого подхода является то, что ограничения по токам и режимам работы учитываются не внешними ограничителями, а заложены непосредственно в стратегию управления, сформированную в процессе обучения.

## Результаты численного моделирования

Для оценки поведения алгоритма МС AI выполнена серия численных экспериментов на цифровом двойнике асинхронного электродвигателя в режиме ступенчатого задания скорости. В качестве базового метода сравнения использовалось классическое векторное управление (FOC) с PI-

регуляторами. Эксперименты проводились для двух *curriculum-стадий эксперимента* (*преднамеренно заданные уровни сложности условий в среде, а не этапы обучения алгоритма*), соответствующих различным значениям задания скорости: 9,42 рад/с (стадия 0) и 15,71 рад/с (стадия 1). В качестве критериев оценки рассматривались среднеквадратичный ток статора, положительная составляющая входной мощности и средняя абсолютная ошибка регулирования скорости.

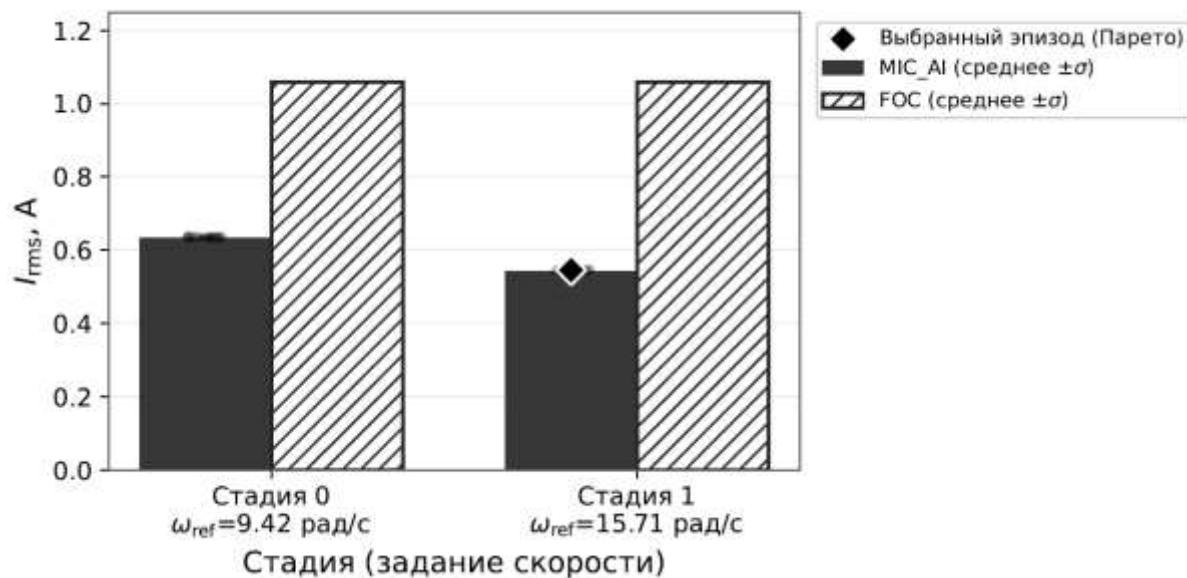


Рисунок 3 – Среднеквадратичный ток статора

На рисунке 3 представлено сравнение среднеквадратичного тока статора для MIC AI и FOC. По текущим данным на обеих curriculum-стадиях среднее значение тока при управлении MIC AI ниже, чем при использовании FOC. Для стадии 0 средний ток MIC AI составляет около 0,63 А, тогда как для FOC — около 1,06 А; аналогичное соотношение сохраняется и для стадии 1. Данный результат указывает на снижение токовой нагрузки при использовании нейросетевого управления на рассматриваемом интервале моделирования.

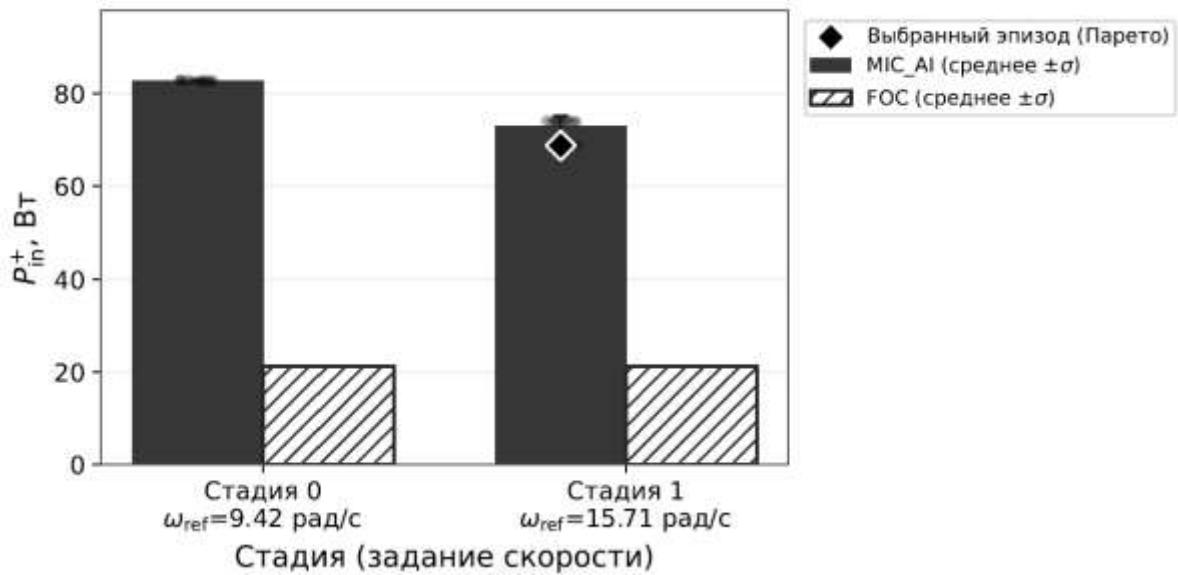


Рисунок 4 – Положительная составляющая входной мощности

Сравнение положительной составляющей входной мощности  $P_{in}^+$  приведено на рисунке 4. В данной серии запусков значения  $P_{in}^+$  при MIC AI превышают соответствующие значения для FOC на обеих curriculum-стадиях. Для MIC AI получены значения порядка 70–85 Вт, в то время как для FOC  $P_{in}^+$  остаётся на уровне около 21 Вт. Таким образом, по выбранной метрике положительной составляющей входной мощности в текущих логах MIC AI не демонстрирует энергетического выигрыша по сравнению с классическим векторным управлением.

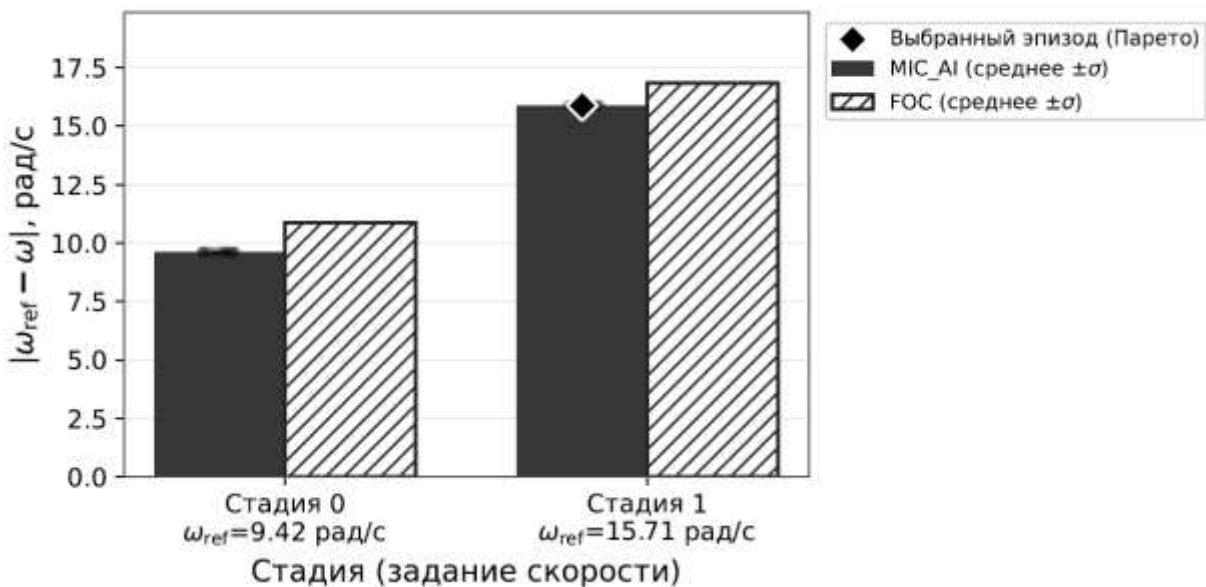


Рисунок 5 – Средняя абсолютная ошибка скорости

Качество регулирования скорости проанализировано по средней абсолютной ошибке (рисунок 5). По имеющимся логам до момента досрочного завершения эпизодов средняя ошибка скорости при MIC AI сопоставима с FOC и в рассматриваемых запусках несколько ниже. Для стадии 0 ошибка при MIC AI составляет около 9,6 рад/с против 10,9 рад/с у FOC, а для стадии 1 — около 15,9 рад/с против 16,8 рад/с соответственно.

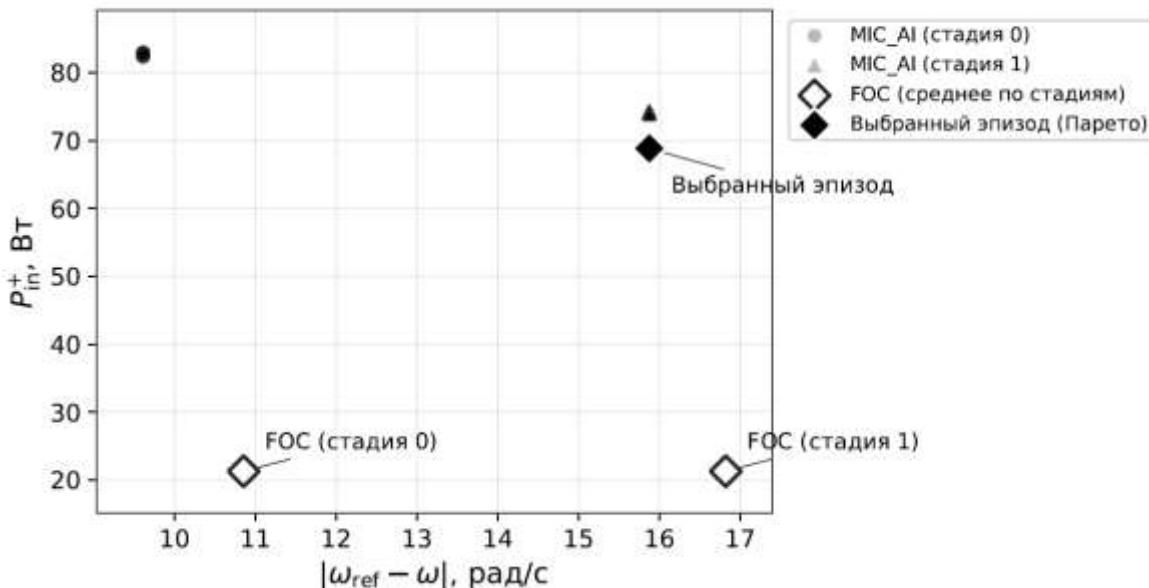


Рисунок 6 – Парето-компромисс (ошибка скорости и входная мощность)

Для комплексной оценки компромисса между точностью регулирования и энергетическими затратами выполнен Парето-анализ (рисунок 6). На плоскости критериев «средняя ошибка скорости – положительная составляющая входной мощности  $P_{in}^+$ » представлены отдельные эпизоды MIC AI и опорные точки FOC для каждой curriculum-стадии. В текущей серии запусков решения MIC AI располагаются в области меньших ошибок регулирования скорости, однако характеризуются существенно большими значениями  $P_{in}^+$  по сравнению с FOC. В качестве репрезентативного выбран один эпизод MIC AI, определённый по критерию минимизации  $P_{in}^+$  при условии, что средняя ошибка скорости не превышает соответствующее значение для FOC на данной стадии.

В работе исследован подход прямого нейросетевого управления асинхронным электродвигателем на основе алгоритмов обучения с подкреплением с обучением в среде цифрового двойника. Разработана структура системы MIC AI, в которой нейросетевой агент формирует

управляющие воздействия непосредственно на силовой инвертор без использования классического каскада PI-регуляторов.

По результатам численного моделирования в двух curriculum-стадиях эксперимента показано, что в текущих логах среднеквадратичный ток статора при управлении МС АІ ниже, чем при классическом векторном управлении. Средняя абсолютная ошибка регулирования скорости до момента досрочного завершения эпизодов сопоставима с FOC и в рассматриваемых запусках несколько ниже. При этом значения положительной составляющей входной мощности при МС АІ превышают соответствующие значения для FOC, что указывает на отсутствие энергетического выигрыша в данной серии экспериментов.

Проведённый Парето-анализ подтвердил наличие компромисса между точностью регулирования скорости и энергетическими затратами при использовании МС АІ. Полученные решения характеризуются меньшими ошибками регулирования при увеличении входной мощности по сравнению с FOC.

В дальнейшем планируется перенос обученной и оптимизированной нейросетевой политики на реальное аппаратное обеспечение асинхронного электропривода для экспериментальной проверки работоспособности предложенного подхода.

## ***Библиографический список***

1. Kazemikia D. Reinforcement Learning for Motor Control: A Comprehensive Review [Электронный ресурс] // arXiv preprint. 2024. – URL: <https://arxiv.org/abs/2401.00001> (дата обращения: 15.12.2025).
2. Field-Oriented Control of PMSM Using Reinforcement Learning [Электронный ресурс] // MathWorks Motor Control Blockset Example. – URL: <https://www.mathworks.com/help/mcb/gs/foc-of-pmsm-using-reinforcement-learning.html> (дата обращения: 15.12.2025).
3. Benmalek E., Gaceb K., Bendaoudi A. AI-Based Field-Oriented Control for Induction Motors // IAPGOS. 2024. №4. С. 48–52. – URL: <https://bibliotekanauki.pl/articles/58907979.pdf#:~:text=%5BPDF%5D%20AI,SVM> (дата обращения: 15.12.2025).
4. Saurabh P., Yadav P., Kumar S. Enhanced Induction Motor Control Using Deep Reinforcement Learning and PWM with DTC // Electric Power Components and Systems. 2023. Vol. 51, №12. P. 1083–1095. – URL: [https://www.researchgate.net/publication/375764297\\_Enhanced\\_Induction\\_Motor\\_Control\\_Using\\_Deep\\_Reinforcement\\_Learning\\_and\\_PWM\\_with\\_DTC#:~:text=](https://www.researchgate.net/publication/375764297_Enhanced_Induction_Motor_Control_Using_Deep_Reinforcement_Learning_and_PWM_with_DTC#:~:text=), (дата обращения: 15.12.2025).
5. Tarbosh Q., Al-Hadithi T., Mahdi H. Sensorless Speed Control of Induction Motor Drives Using RL and Self-Tuning Fuzzy Logic // IEEE Transactions on Industry Applications. 2024. Vol. 60, №2. P. 1234–1242. – URL: [https://www.researchgate.net/publication/375764297\\_Enhanced\\_Induction\\_Motor\\_Control\\_Using\\_Deep\\_Reinforcement\\_Learning\\_and\\_PWM\\_with\\_DTC#:~:text=linearities%2C%20para](https://www.researchgate.net/publication/375764297_Enhanced_Induction_Motor_Control_Using_Deep_Reinforcement_Learning_and_PWM_with_DTC#:~:text=linearities%2C%20para)

meter%20variations%2C%20and%20external,rule%20counterpart%20and (дата обращения: 15.12.2025).

6. Lenz I., Knepper R., Saxena A. DeepMPC: Learning Deep Latent Features for Model Predictive Control // Proc. of Robotics: Science and Systems. 2015. – URL: [https://rpg.ifi.uzh.ch/docs/RSS15\\_Lenz.pdf](https://rpg.ifi.uzh.ch/docs/RSS15_Lenz.pdf) (дата обращения: 15.12.2025).

7. Wang Y., Zhuang X., Liu Z. A Survey of Reinforcement Learning-Driven Knowledge Distillation [Электронный ресурс] // Preprints. 2025. – URL: <https://www.preprints.org/manuscript/202501.0001/v1> (дата обращения: 15.12.2025).

8. Joshi R. Reinforcement Learning for Continuous Control and Embedded Systems [Электронный ресурс] // Techdome Blog. 2024. – URL: <https://techdome.io/blog/rl-on-embedded> (дата обращения: 15.12.2025).

9. Ranjan A., Sharma V., Gupta M. TinyRL: Towards Reinforcement Learning on Tiny Embedded Devices // Proc. of ACM Int. Conf. on Information Processing in Sensor Networks (IPSN). 2023. Р. 215–222. – URL: <https://techdome.io/tinyrl> (дата обращения: 15.12.2025).

10. Enhancing Field-Oriented Control of Electric Drives with Tiny Neural Networks [Электронный ресурс] // arXiv preprint. 2023. – URL: <https://arxiv.org/abs/2312.05555> (дата обращения: 15.12.2025).