УДК 681.513.3
DOI: 10.26102/2310-6018/2026.62.11.014

Автоматическое управление технологической линией

А.В. Кудрявцев[1,2]✉, С.В. Гуськов[3], Э.Э. Анцупов[4]

[1]Национальный исследовательский технологический университет «МИСИС», Москва, Российская Федерация
[2]ООО «Промтехпродукт», Самара, Российская Федерация
[3]Независимый исследователь, Москва, Российская Федерация
[4]Независимый исследователь, Новосибирск, Российская Федерация

Резюме. В работе рассматривается проблема управления технологической линией, состоящей из взаимосвязанных агрегатов (станков), оснащенных локальными системами автоматического управления, эффективно решающими задачи на уровне отдельных объектов, но демонстрирующими снижение качества управления при объединении в единую производственную систему, особенно в условиях нестационарных возмущений. Актуальность исследования обусловлена этим наблюдаемым снижением эффективности технологических линий при объединении агрегатов с локальными системами автоматического управления: в условиях возмущений возникает рассогласование потоков, перегрузка или недоиспользование оборудования. Целью работы была разработка подхода к автоматическому управлению работой линии на основе гармонизации локальных и глобальных целей управления под управлением искусственной нейронной сети. В качестве решения в работе предложен двухуровневый подход с использованием цифрового двойника линии: верхний нейросетевой регулятор на основе ансамбля из шести методов (классический Extremum-Seeking Control как baseline, трилинейная интерполяция по таблице оптимальных параметров, многослойный перцептрон (MLP), регрессия на гауссовских процессах (GP), рекуррентный и нерекуррентный варианты Soft Actor-Critic (SAC), а также сеть Колмогорова-Арнольда (KAN)) формирует управляющие сигналы k_{press} для каждого станка, а локальные регуляторы стабилизируют работу каждого из станков. В результате проведенных исследований были разработаны подходы к созданию архитектуры управления, обеспечивающей прогнозирование динамики системы и координацию потоков между связанными агрегатами. На примере модели работающего оборудования показано, что введение верхнего уровня управления повышает устойчивость к нестационарным воздействиям и улучшает баланс загрузки. Предложенный подход обеспечивает согласование локальных и глобальных цепей управления, повышение эффективности использования мощностей и может быть применен в задачах оптимизации и управления сложными производственными системами.

Ключевые слова: теория автоматического управления, вальцовый станок, искусственная нейронная сеть, гауссовский процесс, сеть Колмогорова-Арнольда, обучение с подкреплением, цифровой двойник, мукомольное оборудование.

Для цитирования: Кудрявцев А.В., Гуськов С.В., Анцупов Э.Э. Автоматическое управление технологической линией. *Моделирование, оптимизация и информационные технологии.* 2026;14(11). URL: https://moitvivt.ru/ru/journal/article?id=2461 DOI: 10.26102/2310-6018/2026.62.11.014

Automated control of a process line

A.V. Kudriavtsev[1,2]✉, S.V. Guskov[3], E.E. Antsupov[4]

[1]National University of Science and Technology "MISIS", Moscow, the Russian Federation
[2]Promtechproduct LLC, Samara, the Russian Federation
[3]Independent researcher, Moscow, the Russian Federation
[4]Independent researcher, Novosibirsk, the Russian Federation

© Кудрявцев А.В., Гуськов С.В., Анцупов Э.Э., 2026

Abstract. This study addresses the problem of controlling a technological line composed of interconnected units (mills) equipped with local automatic control systems. While these local systems effectively solve control tasks at the individual unit level, they exhibit a degradation in control performance when integrated into a unified production system, particularly under non-stationary disturbances. The relevance of this research is driven by the observed decline in line efficiency when aggregating units with local automatic control systems: under disturbance conditions, flow mismatches, equipment overload, or underutilization occur. The aim of this work was to develop an automatic control approach for the line based on aligning local and global control objectives under the supervision of an artificial neural network. As a solution, a two-tier hierarchical approach utilizing a digital twin of the line is proposed. The upper-level neural controller employs an ensemble of six methods (classical Extremum-Seeking Control as a baseline, trilinear interpolation over a lookup table of optimal parameters, Multilayer Perceptron (MLP), Gaussian Process (GP) regression, recurrent and non-recurrent Soft Actor-Critic (SAC) variants, and a Kolmogorov-Arnold Network (KAN)) to generate control signals for the compaction coefficient k_{press} for each mill, while local controllers stabilize the operation of individual units. The research resulted in the development of a control architecture that ensures system dynamics prediction and coordination of material flows between interconnected units. Using a simulation model of operational equipment, it is demonstrated that introducing an upper control level enhances robustness to non-stationary disturbances and improves load balancing. The proposed approach ensures the coordination of local and global control loops, increases capacity utilization efficiency, and can be applied to optimization and control tasks for complex production systems.

Keywords: automatic control theory, roller mill, artificial neural network, Gaussian process, Kolmogorov-Arnold network, reinforcement learning, digital twin, milling equipment.

For citation: Kudriavtsev A.V., Guskov S.V., Antsupov E.E. Automated control of a process line. *Modeling, Optimization and Information Technology.* 2026;14(11). (In Russ.). URL: https://moitvivt.ru/ru/journal/article?id=2461 DOI: 10.26102/2310-6018/2026.62.11.014

Введение

Автоматизация управления технологическими процессами является актуальной задачей последних лет. В настоящее время большинство систем автоматического управления (САУ) с использованием искусственных нейронных сетей представляют собой либо рекомендательные экспертные системы, либо сложные датчики[1]. Однако в контексте управления технологическим процессом этого бывает недостаточно, что часто вызывает закономерные трудности у специалистов. В настоящее время исследователями разрабатываются системы замкнутого контура регулирования, которые выходят далеко за рамки простых рекомендаций по оптимизации процесса и способны управлять технологическими параметрами автономно. Одним из инструментов создания таких систем является разработка систем управления, использующих искусственные нейронные сети, обучаемые на данных, получаемых от цифровых двойников. Цифровой двойник – это виртуальное представление физической системы, способное моделировать, прогнозировать и оптимизировать ее работу в реальном времени [1]. Отсюда очевидно, что математическая модель может являться важным компонентом создания САУ. Одним из возможных подходов для решения задач создания таких систем управления, является использование локальных САУ для управления отдельными участками процесса (например, контролем качества шихты, управлением отдельным станком, мониторингом изделий или процессов) с их последующей интеграцией в иерархическую многофункциональную децентрализованную систему автоматического управления [2].

[1] Темкин И.О., Трофимов В.Б. *Искусственные нейронные сети в АСУ ТП.* Москва: Издательский Дом НИТУ МИСИС; 2023. 352 с.

Но для построения такой системы необходимо решить задачу гармонизации работы станков в технологической линии с помощью системы управления верхнего уровня при наличии САУ каждым из станков, поскольку с ростом сложности производственной системы переход к многоуровневому управлению обеспечивает более эффективное использование ресурсов, в целях упорядочения процесса выпуска готовой продукции [3]. Это означает, что если САУ для каждого из станков настроена и работает достаточно эффективно при выполнении задач автоматического управления одним станком, то в случае объединения станков в единую систему часто возникают сложности: локальные САУ, предназначенные для контроля каждого станка, демонстрируют снижение качества управления при интеграции их в единый поток, особенно в условиях нестационарных воздействий. Этот эффект порождает неоптимальные режимы работы, такие как перегрузка одних станков и недостаточное использование других, что приводит к нарушению баланса материальных потоков и общему снижению эффективности производства [4].

Ключевой причиной этого явления является несогласованность целей управления на разных уровнях, наличие перекрестных динамических связей между агрегатами и, что наиболее важно, отсутствие координирующего уровня, способного обеспечить оптимизацию работы всех станков как единого комплекса. Проблема состоит в природе «эгоистичного» поведения локальных САУ – это означает, что в локальных цепях управления реализуется принцип регулирования по замкнутому контуру, который направлен на стабилизацию определенных параметров процесса – в рассматриваемом случае параметров вальцового станка – без учета влияния на всю систему в целом [5]. САУ станком по своей природе реактивна: пытается компенсировать произошедшее отклонение, вызвавшее ошибку, и не в состоянии предвидеть результаты своей работы для всей линии в целом [6].

Целью исследования является изучение разных инструментов и методов искусственного интеллекта применительно к 2-уровневому контуру управления технологической линией для повышения эффективности ее работы за счет оптимального управления коэффициентами прижима вальцов (k_{press}). Задачи работы: 1) синтез высококачественного датасета пар «состояние линии → оптимальные k_{press}» с использованием физически обоснованной модели количественного баланса материальных потоков (Population Balance Model, PBM-модели); 2) сравнение методов обучения с учителем (supervised learning – MLP, KAN, GP) и обучения с подкреплением (reinforcement learning – SAC, LSTM-SAC) при равных условиях имитационного моделирования; 3) оценка устойчивости методов к гауссову шуму измерений и их адаптации через дообучение; 4) формулировка ограничений и направлений дальнейших исследований (имитационное обучение, PPO в непрерывных пространствах действий).

Материалы и методы

Для исследования поставленного вопроса прежде всего была сделана модель линии. Для этого была разработана система алгебро-дифференциальных уравнений, описывающих динамическое поведение отдельного станка, моделирующая его работу. Далее в полученной системе была проведена параметрическая идентификация: были выделены наборы эмпирических коэффициентов и характеристик станка как объекта моделирования, которые были валидированы в экспериментах на реальном оборудовании, методом наименьших квадратов. Затем было проведено имитационное моделирование, по результатам которого модель была дополнительно настроена и валидирована по результатам экспериментов на оборудовании в работе так, чтобы она соответствовала реальности, с допустимым уровнем точности.

При создании модели линии использовалась PBM-модель, которая описывает изменение распределения частиц по размеру и по зольности во времени под действием процессов разрушения и сепарирования полученных продуктов. Суть модели в том, что количество продуктов, полученных в результате работы каждой системы в отдельности, должно равняться количеству продуктов, поступивших на систему. Общий вид одномерного уравнения количественного баланса для процесса помола записывается через функцию плотности частиц $n(x,t)$ по размерному параметру x (например, диаметр или масса частицы) и время t [7]:

$$\frac{\partial n(x,t)}{\partial t} = -S(x) \cdot n(x,t) + \int_x^{\infty} S(y) \cdot n(y,t) \cdot b(x|y) dy, \quad (1)$$

где $S(y)$ – интенсивность разрушения частиц размера y (частиц/с) или функция селекции (отбора), показывающая удельную скорость разрушения, смысл которой в том, какая вероятность того, что частица размера x будет захвачена для измельчения за единицу времени; $b(x|y)$ – функция измельчения (распределения частиц), задающая долю (плотность вероятности) фрагментов измельченного размера x при разрушении частицы исходного размера y; $n(x,t)$ – концентрация (массовая доля) частиц размера x в момент времени t. Для каждого размера y подынтегральное выражение $S(y) \cdot n(y,t)$ дает количество разрушенных материнских частиц, а $b(x|y)$ – ту их долю, которая при разрушении попадает в интересующий нас размер x. В стационарном режиме, рассмотренном в предыдущем разделе, левая часть уравнения обращается в ноль ($\frac{\partial n(x,t)}{\partial t} = 0$), и интегро-дифференциальное уравнение сводится к алгебраическому балансу между стоком и приходом для каждого класса крупности, что и было положено в основу матричной модели. Это обстоятельство и было использовано при оценке полученных экспериментальных данных – они описывали стационарное состояние (Таблица 1). В дискретном представлении по N размерным классам можно записать систему ОДУ для масс $M_i(t)$ в классе i:

$$\frac{dM_i}{dt} = -A_i \cdot M_i + \sum_{j=1}^{i=16} A_j \cdot M_j \cdot \beta_{i \to j}, \; i = 1,2, \dots, 16, \quad (2)$$

где A_j – удельная скорость разрушения (selection rate) для класса j, а $\beta_{i \to j}$ – доля массы из класса j, переходящей в класс i при дроблении. При этом $\sum_j^i \beta_{i \to j} = 1$ для любого j. Выбор конкретной формы $S(x)$ и $b(x|y)$ базируется на моделях помола и обычно параметризуется. В [8] дан способ моделирования помола через использование функции помола, имеющей вероятностный характер, и которая аппроксимируется через экспериментальное нахождение матрицы помола, зависящей от конкретного станка, зерна, условий, что составляет некоторые сложности для использования на работающем оборудовании. Поэтому для создания модели помола была взята обратно-экспоненциальная зависимость вероятности попадания частиц в заданный диапазон в зависимости[2] от величины рабочего зазора:

$$P(D,b) = A \cdot \int_{d1}^{d2} e^{\left(-\frac{D}{b}\right)} dD = A \cdot b \cdot \left(e^{\left(-\frac{d1}{b}\right)} - e^{\left(-\frac{d2}{b}\right)} \right), \quad (3)$$

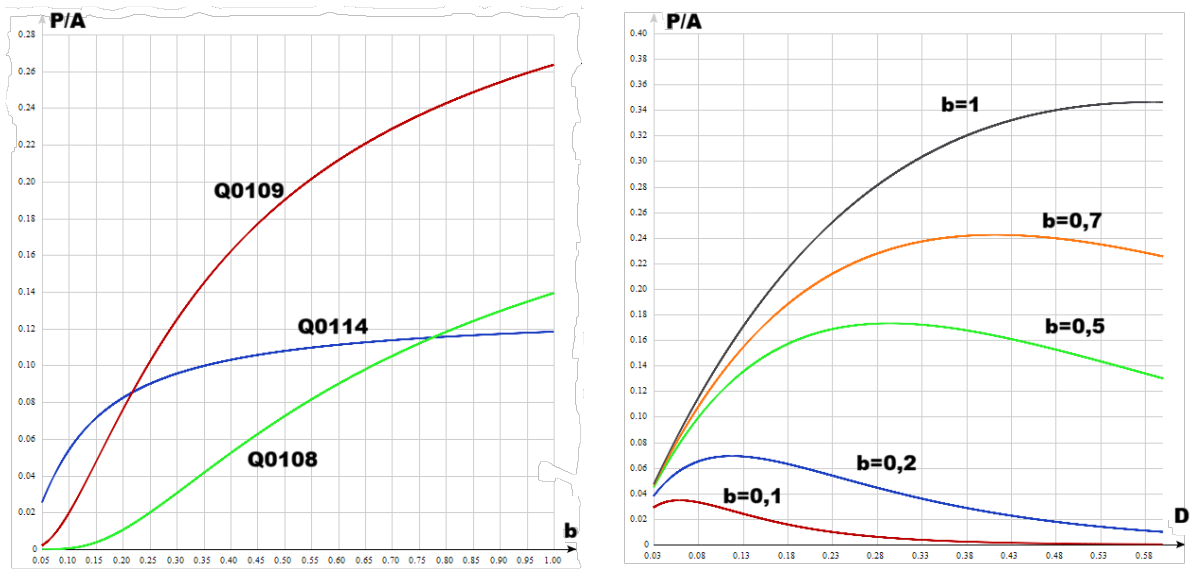
где $P(D,b)$ – вероятность попадания частиц во фракцию с минимальными и максимальными размерами $d1$, $d2$, A – коэффициент нормализации, т. к. вероятность не может быть более 1, D – размер частиц фракции, b – рабочий зазор (расстояние между вальцами).

[2] Данная вероятность используется при вычислении в модели потоков фракций после помола, который равен произведению входящего потока на станок и вероятности: $Q_{ij} = Q_j \cdot P(D,b)$.

Таблица 1 – Потоки фракций, их зольность в линии согласно PBM-модели
Table 1 – PBM-predicted fraction flows and ash content in the milling line

Исходящие потоки со станков на станки, г/с													
		B1	B2	B3	B4	B5	C1A	C1B	C2	C3	C4	C5	C6
1	B1												
2	B2	482											
3	B3		122										
4	B4	12	5	27									
5	B5			27									
7	C1A							367	544				
8	C1B	426	26	27	18	11							
9	C2	678	240	12									
10	C3						641						
11	C4						154	55		364			
12	C5				20	12			168		190		
13	C6											288	
14	Flour	139	88	30	5	3	115	86	218	277	279	37	103
16	Br1				1	1					105	65	185
Произв. станка, г/с		1736	482	122	44	27	911	507	930	641	573	390	288
Вход. зольность (z_{in}), %		**1,62**	**1,67**	**1,86**	**1,80**	**2,39**	**2,33**	**1,63**	**2,20**	**3,19**	**3,40**	**4,21**	**5,55**
Зольности потоков из станков, %													
		B1	B2	B3	B4	B5	C1A	C1B	C2	C3	C4	C5	C6
1	B1												
2	B2	1,67											
3	B3		1,86										
4	B4	0,73	1,14	2,39									
5	B5			2,39									
7	C1A							1,95	2,58				
8	C1B	1,56	1,26	2,36	2,05	2,53							
9	C2	2,07	2,60	1,73									
10	C3						3,19						
11	C4						1,09	0,62		4,80			
12	C5				2,16	2,71			1,19		7,19		
13	C6											5,55	
14	Flour	0,65	0,90	0,92	0,93	0,76	0,37	0,32	0,44	0,41	0,65	0,80	0,70
16	Bran				1,10	0,98					4,52	1,94	7,51
Выход. зольность (z_{out}), %		**1,71**	**2,01**	**1,96**	**1,95**	**2,35**	**2,47**	**1,53**	**1,83**	**2,91**	**3,53**	**4,50**	**5,07**

Данная функция была взята как наиболее точно описывающая суть процесса образования частиц при изменении рабочего зазора. Графики зависимости функции от значений переменных изображены на Рисунке 1.



а) *б*)

Рисунок 1 – Зависимость вероятности образования частиц во фракциях в зависимости от: *а* – рабочего зазора (*b*); *б* – размера частиц (*D*)
Figure 1 – Dependence of the probability of particle formation in fractions on: *a* – working gap (*b*); *b* – particle size (*D*)

На графиках Рисунка 1 обозначенные потоки продуктов были найдены как вероятности образования потоков различных фракций Q_{ij}, где i – 2-разрядный номер станка, на котором данный поток образовался, и j – 2-разрядный номер станка, на который данный поток направлен. Например, Q_{0108} – это поток промежуточного продукта с 1-го станка ($i = 01$) на станок № 8 ($j = 08$). Иногда j называется номером потока. Если обратиться к Таблице 1[3], то можно понять, что станок № 8 – это станок С1В. Следовательно, график на Рисунке 1*а* описывает изменение плотности вероятности пропорций фракций, получаемых на станке В1 при изменении рабочего зазора между вальцами (*b*). Для вычисления потоков, приведенных в столбце В1 Таблицы 1 (верхняя часть), необходимо нормализовать данные отношения фракций и умножить на входной поток. Следует обратить внимание, что в данной модели принято: $j = 14$ – это всегда мука (без выделения муки 1С), $j = 15$ – мука 1С, $j = 16$ – всегда поток отрубей.

Для управления выходными фракциями на каждом вальцовом станке имеется устройство регулирования рабочего зазора (*b*) между вальцами – штурвал. Вальцы в свободном состоянии (без продукта) сжаты друг с другом, а поток продукта, поступая на вальцы, разжимает их, преодолевая силу сжатия пружины, пропорциональную смещению. При регулировке штурвала изменяется смещение (сжатие) пружины, описываемое коэффициентом k_{press}, тем самым можно изменять силу сжатия пружины, прессование потока продукта (изменение его насыпной плотности) в процессе помола и, соответственно, производительность и пропорции между выходными фракциями. Требуется добиться такого соотношения фракций, чтобы принятая метрика в виде

[3] Поскольку Таблица 1 составлена для моделирования реальной линии, то в данной линии под номером 6 присутствует агрегат энтолейтор, который измельчает промежуточный продукт, но не является объектом управления – поэтому строка с ним была пропущена, а потоки из него и их параметры добавлены к соответствующим станкам. В действующей модели работа энтолейтора была рассчитана как линейное преобразование. Так же следует иметь ввиду, что потоки муки везде в модели обозначены под номером 14, однако далее в процессе разделения по белизне (зольности) из потока муки высшего сорта (ВС) выделяется поток муки 1 сорта (1С). У потока ВС остаётся номер 14, а отобранному потоку 1С присваивается номер 15, который в данной статье не рассматривается – исследуется общий поток муки.

кумулятивной стоимости полученных продуктов на выходе со всего комплекса станков была максимальной. Для решения данной задачи были проведены эксперименты с управлением одним из адаптивных методов, а именно методом управления с поиском экстремума, а также с использованием нейронных сетей.

Одним из методов адаптивного управления комплексом станков мукомольного оборудования для решения проблемы динамической оптимизации модели, имеющую выходным сигналом функцию кумулятивной стоимости получаемых продуктов переработки при заданных ценах *COST*[4], которую необходимо максимизировать, был использован метод управления с поиском экстремума (Extremum Seeking Control, ESC) [9]. ESC представляет собой класс адаптивных алгоритмов управления, предназначенных для автоматической настройки регулируемых параметров с целью поддержания системы в точке экстремума целевой функции. Блок-схема, реализующая метод ESC, представлена на Рисунке 2.

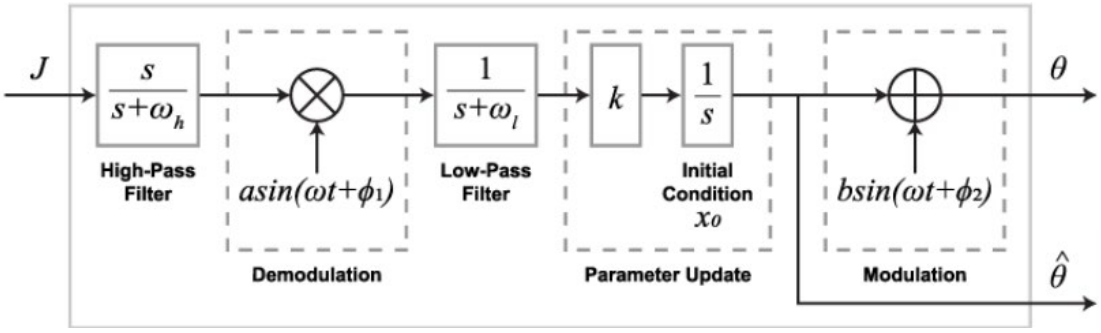


Рисунок 2 – Блок-схема ESC
Figure 2 – Scheme of the ESC

Значение целевой функции (на Рисунке 2 обозначена как *J*, в моделях настоящей работы – *COST*) поступает на высокочастотный фильтр (ВЧФ), что позволяет отсечь постоянную составляющую и выделить полезный высокочастотный отклик; частоту среза ВЧФ ω_h следует выбирать меньше, чем частота модуляции/демодуляции $\omega_h < \omega/2\pi$. Для исследуемой модели зададим $\omega_h = 0{,}08$ рад/с. Далее сигнал демодулируется. Этот шаг служит для идентификации текущего положения системы относительно точки максимума функции *J*. Частота модуляции/демодуляции ω выбирается для каждого контура индивидуально, исходя из динамики модели, не должна повторяться и должна быть в 5–10 раз выше, чем частота изменения потока продукта на входе первого станка В1. Т. е. если на вход станка В1 поступает поток продукта с частотой 1/60 Гц, то минимальная частота модуляции/демодуляции $\omega = 5 \cdot 2\pi/60 \approx 0{,}5$ рад/с. Для исследуемого случая зададим 12 частот в диапазоне от 0,5 до 0,88 рад/с. Далее, опционально, можно использовать низкочастотный фильтр (НЧФ) для подавления высокочастотных помех. Частота среза НЧФ должна быть выше, чем частота модуляции/демодуляции ω. В исследуемом случае НЧФ не включен. Далее очищенный сигнал интегрируется и умножается на коэффициент адаптации *k*. Результатом данной операции является формирование корректирующего воздействия, которое смещает регулируемый параметр в направлении максимума функции *J*. Коэффициент *k* не должен

[4] В качестве целевой функции *COST* в экспериментах была принята кумулятивная стоимость продуктов, получаемых из зерна (на входе линии) – это мука высшего сорта (ВС), мука 1 сорта (1С) и отруби (Отр/ Вг), за время работы 3600 с (1 час). Т. к. цена продуктов может изменяться во времени, поэтому необходимо принимать это во внимание при управлении линией. Например, мука 1С может как быть по цене близко к ВС, так и отличаться боле чем на 25 %. Для данной работы били приняты следующие цены 27/21/7,7 рублей за 1 кг ВС/1С/Отр соответственно. Очевидно, что при изменении цен, оптимальные доли выходов муки меняются.

слишком сильно усиливать сигнал, чтобы не приводить к нестабильному поведению модели, поэтому для каждого контура подобран небольшой k в диапазоне от 0,1 до 0,5. На финальном этапе к сигналу управления добавляется гармоническое возмущение (модуляция), необходимое для непрерывного зондирования поверхности целевой функции и поиска обновленного оптимума. Амплитуды модулирующего и демодулирующего сигнала a и b в исследуемой модели выбраны равными 0,2. Как и в случае с коэффициентом k, слишком большие амплитуды a и b приводят к неустойчивости и неработоспособности модели.

Данный метод был использован для оптимизации комплексной модели мукомольного производства из 12 вальцовых станков в программе MATLAB-Simulink, стрелками показаны связи между станками (Рисунок 3).

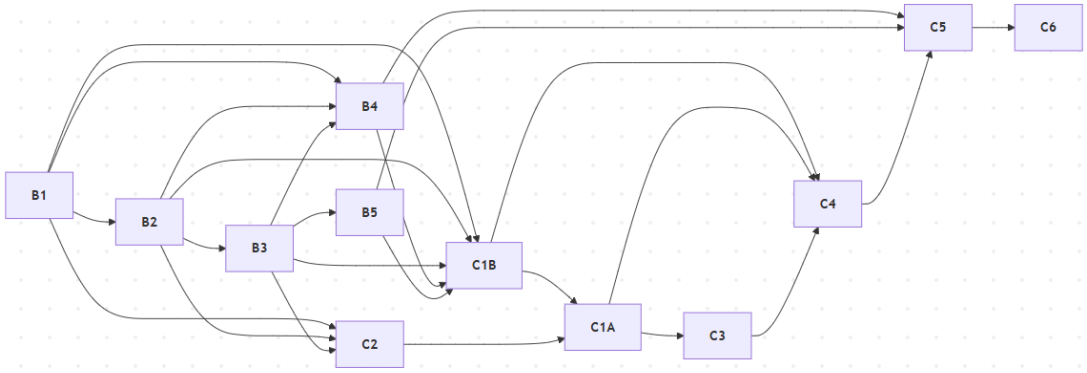


Рисунок 3 – Блок-схема комплексной модели мукомольной линии в составе 12 станков
Figure 3 – Block diagram of an integrated model of a milling line consisting of 12 machines

В комплексной модели на первом этапе было реализовано два подхода к управлению: 1 вариант – работа модели со статическими k_{press}, отрегулированными специалистами-технологами (Static), 2 вариант – регулирование коэффициента k_{press} посредствам ESC оптимизации (ESC).

Структура управляющего адаптивного ESC контроллера представлена на Рисунке 4. Она представляет собой 3 блока ESC, с 4 отдельными контурами для оптимизации параметров модели.

На втором этапе работы были использованы регуляторы на основе искусственных нейронных сетей. Для обучения нейросетевых контроллеров необходимо наличие данных для обучения. Эта задача была решена следующим образом: был синтезирован обучающий датасет, состоящий из пар «состояние линии – значения управляющих сигналов для станков k_{press}». Состояние линии представляется вектором (obs-вектором), объединяющим: входной поток зерна Q_{in} (1 признак), уровни наполнения 12 бункеров h_i (12 признаков), управляющие воздействия 12 локальных регуляторов уровня u_i (12 признаков), полные внутренние потоки массы продукта по всем выходным направлениям станков B1–C6 (42 канала по количеству потоков в линии, Рисунок 3, Таблица 1, верхняя часть), соответствующие показатели зольности этих потоков (42 канала, Таблица 1, нижняя часть) и текущее значение целевой функции стоимости *COST* (1 признак) – всего 158 признаков (N_{obs} = 158). Вектор управляющего воздействия – 12-мерный вектор k_{press} для всех вальцовых станков B1–B5 и C1A–C6. Здесь $W_{зерна}$ – влажность зерна после кондиционирования, %; диапазон 15–17 % соответствует стандартной технологии помола мягких пшениц (типовые значения 14–17 %).

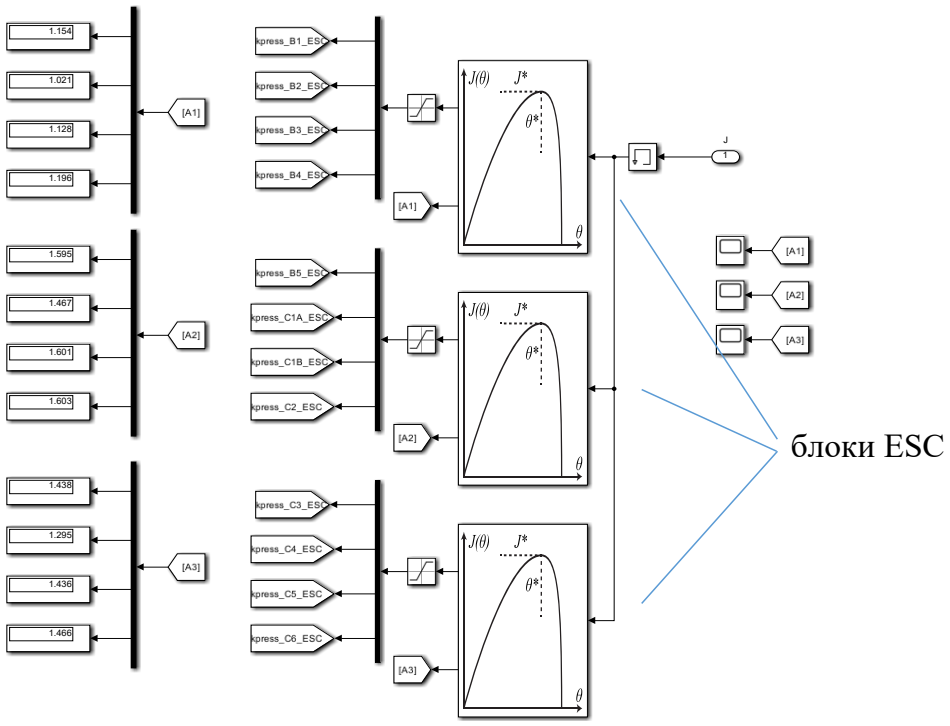


Рисунок 4 – Структурная схема управляющего адаптивного ESC контроллера
Figure 4 – Structure of the adaptive ESC controller

Датасет формируется в две фазы. На первой фазе для сетки рабочих точек при разных входных потоках зерна в линию $Q_{in} \in \{0{,}9; 1{,}3; 1{,}736; 2{,}2\}$ кг/с, $W_{зерна} \in \{15; 16; 17\}$ %, $COST_{target} \in \{21; 23; 25\}$ руб./кг – всего 36 точек – выполнялась полная трехфазная оптимизация: латинский гиперкуб (LHS, 100000 случайных точек), локальный поиск L-BFGS-B из 20 лучших стартов и байесовская оптимизация на основе гауссовских процессов с функцией приобретения Expected Improvement (65 итераций). На второй фазе генерировались 400 дополнительных точек со случайными $Q_{in} \in [1{,}0; 2{,}3]$ кг/с и быстрой L-BFGS-B-оптимизацией от ближайшей точки сетки в качестве начальной (warm start), что обеспечивает разнообразие операционных условий при умеренной вычислительной стоимости.

В результате получен датасет из 436 точек, который сохраняется в формате NumPy и используется как обучающая выборка для всех методов обучение с учителем – supervised-методов (трилинейная интерполяция, MLP, GP, KAN). Для методов обучения с подкреплением (SAC, LSTM-SAC) обучающая выборка формируется онлайн в процессе взаимодействия с симулятором линии, который реализует ту же физическую модель, что и имитационная модель линии в Simulink (PBM, динамика изменения уровня продукта во входном бункере, локальное регулирование уровней). Полный код, веса моделей и сетки рабочих точек размещены в репозитории. Воспроизводимость обеспечена фиксированным *seed* = 42 для всех генераторов случайных чисел (numpy, torch, gymnasium).

Далее для решения задачи многопараметрической оптимизации k_{press} было реализовано и проверено шесть нейросетевых методов, относящихся к различным семействам обучения. Выбор методов обусловлен необходимостью сравнить классические регрессионные подходы (с явным обучающим набором) с методами обучения с подкреплением (без обучающего набора), а также проверить, помогает ли учет истории состояний (рекуррентность). Архитектура нейросетевого контроллера представлена на Рисунке 5. Архитектура включает два этапа. На оффлайн-этапе

выполняется трехфазная оптимизация k_{press} по сетке рабочих точек, формирующая таблицу K оптимальных управляющих воздействий. На онлайн-этапе нейросетевая модель отображает текущее состояние линии в вектор k_{press}, опционально предусмотрена residual-коррекция от RL-агента.

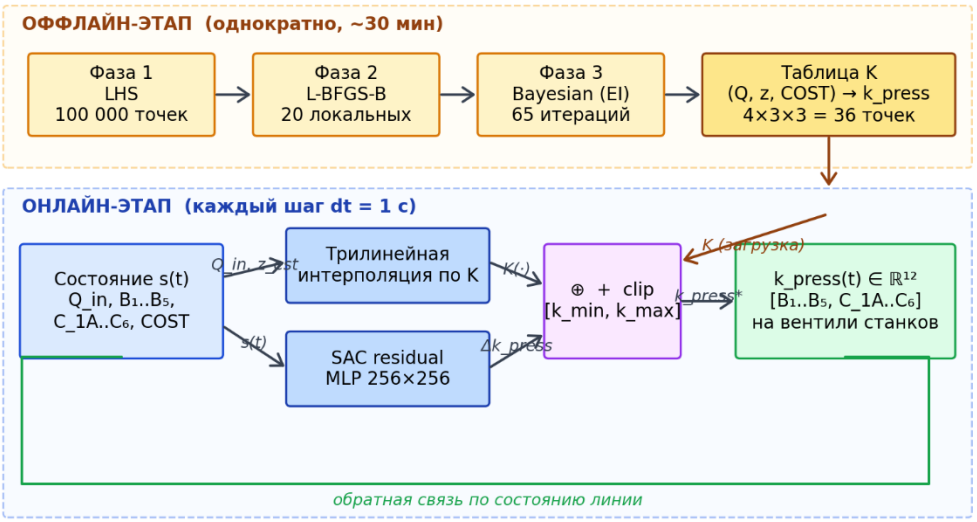


Рисунок 5 – Двухуровневая архитектура ИНС-контроллера
Figure 5 – Two-level architecture of the ANN controller

1) Трилинейная интерполяция по таблице оптимальных k_{press} (RBF-baseline). Простейшая радиально-базисная нейронная сеть с фиксированными узлами в точках сетки ($Q_{in} \times z \times COST$). Используется как нижняя граница качества нейросетевого подхода: любой обучаемый метод должен превосходить простую интерполяцию между точками офлайн-оптимизации. Подобный подход применяется для табличного представления управляющих воздействий в адаптивных регуляторах [9].

2) Многослойный перцептрон (MLP, Multi-Layer Perceptron) – feedforward-сеть с двумя скрытыми слоями по 128 нейронов и активацией ReLU. Согласно теореме об универсальной аппроксимации [10], MLP способен с любой точностью аппроксимировать произвольную непрерывную функцию на компактном множестве при достаточном числе нейронов. Сеть обучается минимизацией среднеквадратической ошибки между предсказанным и оптимальным k_{press} на 436-точечном датасете в течение 300 эпох методом Adam с начальной скоростью обучения 10^{-3}. Применение MLP в задачах управления промышленными процессами рассмотрено в [11].

3) Регрессия на гауссовских процессах (Gaussian Process Regression, GP). Байесовский непараметрический метод, который, в отличие от MLP, не требует выбора архитектуры сети, а использует ковариационную функцию (RBF-ядро с автоматически подбираемой длиной шкалы) для построения распределения вероятностей над функциями [12]. Преимущество GP – наряду с предсказанием k_{press} метод выдает доверительный интервал, что позволяет переключаться на безопасный baseline при превышении порога неопределенности. Реализация выполнена поэлементно: 12 независимых GP-моделей, каждая для одного из k_{press}.

4) Soft Actor-Critic (SAC) – алгоритм глубокого обучения с подкреплением семейства off-policy actor-critic с регуляризацией энтропии политики [13]. В отличие от supervised-методов, SAC не требует заранее сформированного датасета: агент обучается напрямую на симуляторе, накапливая опыт взаимодействия в replay-буфере. Энтропийная регуляризация предотвращает преждевременное схождение к локальному

оптимуму. Архитектура: actor с двумя скрытыми слоями по 256 нейронов, два Q-критика с такой же топологией, автоматическая настройка коэффициента температуры α. Обучение проводилось 80 000 шагов в режиме случайных возмущений Q_{in}.

5) LSTM-SAC – рекуррентный вариант SAC с долгой краткосрочной памятью (Long Short-Term Memory, 128 скрытых единиц) [14]. В отличие от обычного SAC, учитывает не только текущее наблюдение, но и историю предыдущих 10 шагов, что позволяет идентифицировать тренды и краткосрочные изменения в работе линии (заполнение/опустошение бункеров, накопление эффектов помола). Применение LSTM-сетей в управлении промышленными процессами с временными зависимостями рассмотрено в [15]. На этапе предобучения использован Soft Actor-Critic с двойным критиком (clipped double Q), целевой энтропией с автонастройкой коэффициента α, target networks с polyak-усреднением τ и replay buffer'ом. Архитектура актора и критиков использует LSTM-энкодер истории длины 10 шагов поверх 158-мерного наблюдения (признаки были описаны выше при получении обучающей выборки). Гиперпараметры: discount γ = 0,99, *polyak* τ = 0,005, *batch size* = 256, *learning rate* $3 \cdot 10^{-4}$ (актор и оба критика), replay buffer 100 000 переходов, целевая энтропия −12 (размерность действия). Дообучение в шумной среде выполнено в режиме REINFORCE-with-baseline warm-start (Использовалась упрощенная схема on-policy для быстрой адаптации из существующих весов – использование полноценного алгоритма SAC-retrain в шумной среде является направлением дальнейшей работы).

6) Сеть Колмогорова-Арнольда (KAN, Kolmogorov-Arnold Network) – альтернатива MLP, основанная на теореме Колмогорова-Арнольда о суперпозиции непрерывных функций многих переменных через композицию одномерных функций [16]. В отличие от MLP, где обучаются веса между фиксированными активациями, в KAN обучаются сами одномерные функции активации (как обучаемые B-сплайны), размещенные на ребрах графа сети. Это дает более интерпретируемое представление (можно извлечь символьные формулы, что сделано в дальнейших исследованиях) при сопоставимом числе параметров и существенное уменьшение вычислений. Архитектура: [158 → 16 → 12], B-сплайны порядка $k = 3$ с 5 узлами сетки. Обучение L-BFGS-оптимизатором, 50 шагов, регуляризация L_1 + энтропийная для разреженности активаций. Реализация на основе библиотеки pykan [16].

Все шесть методов реализованы на языке Python 3.12 в виде, эквивалентном по структуре Simulink-модели. Динамическая часть симулятора (12 приемных бункеров с локальным регулированием уровней по уставке h_0 = 0,5 м, коэффициенты $P = -109{,}68$, $I = -63{,}33$, $D = 29{,}92$, $N = 1{,}72$) и вероятность попадания частиц во фракцию частиц (3) с уникальными для каждого станка коэффициентами d_{min}, d_{max}, m_i, γ, k_0, b_0, α переписаны покомпонентно. Ошибка для локального регулирования (отслеживания уровня в приемном бункере) определяется как $e = h - h_0$; отрицательные знаки коэффициентов P и I отражают противоположное направление управляющего воздействия по отношению к ошибке (при росте уровня h над уставкой h_0 управляющий сигнал должен уменьшать поток помола). Полный 3600-секундный прогон выполняется на однопроцессорной системе за 0,5 с, что позволяет проводить интенсивные эксперименты по подбору гиперпараметров и многократные прогоны для статистической оценки.

Для корректной интерпретации полученных результатов в настоящем исследовании были приняты основные упрощения и ограничения:

1) Дискретизация уравнений динамики выполнялась с шагом $dt = 1$ с методом Forward Euler (для I-составляющей PI-регулятора и интегратора бункера) и Backward Euler (для D-фильтра). Это приводит к небольшим расхождениям с исходной непрерывной моделью Simulink (порядка ±10 % по пиковым значениям переходных

процессов), но не влияет на установившиеся значения и относительные приросты выручки между методами.

2) Обучающая выборка для supervised-методов синтезирована численно – не получена из реальных производственных данных. Качество обучения ограничено точностью симулятора и охватом сетки рабочих точек. В рамках текущего исследования это допустимо, поскольку цель – сравнить относительную эффективность алгоритмов; для применения на реальном оборудовании возможно потребуется дообучение на промышленных данных.

3) Аддитивный шум вводился только на входной сигнал датчика расхода зерна Q_{in} и на наблюдаемый вектор текущих параметров (obs-вектор) линии (при обучении на зашумленной выборке и в шумной среде RL); влияние шума на показания датчиков уровня h_i и расхода продукта f_i по отдельности, а также шум на исполнительных устройствах (положении штурвала k_{press}) в настоящей работе не исследовались.

4) Оптимизация k_{press} проводилась без учета возможных взаимных корреляций между параметрами разных станков. На практике одновременное изменение нескольких k_{press} может вызывать нелинейные эффекты в общей производительности; учет таких корреляций требует более крупных моделей и более обширных датасетов.

5) Формула (3) для распределения частиц по фракциям представляет собой феноменологическое приближение, верифицированное экспериментально только для режимов работы, близких к стационарным. Применимость модели в сильно нестационарных условиях (например, при запуске или останове линии) требует дополнительной валидации.

6) Шум вводился только в виде гауссова с заданным σ и без корреляций между каналами; шум с реальной автокорреляцией датчиков (розовый, дрейф) и шум на исполнительных устройствах (k_{press}) в работе не исследовался.

7) Статистика по возмущенному потоку получена усреднением только по пяти сидам (42, 88, 123, 240, 321). Расширение до 10 сидов и построение формальных доверительных интервалов оставлены для будущих работ. Стационарный режим, будучи детерминированным, оценивался одним прогоном.

8) Модель GP_{noisy} (GP, переученный на 3-кратно аугментированной шумом выборке) практически не отличается от базового GP во всех режимах шума (med-режим 64 582 → 64 584 руб., изменение в пределах 0,01 %). Таким образом, для GP дообучение на шуме не дает значимого эффекта – метод устойчив к шуму на входе и без него.

9) Методы обучения с подкреплением (SAC_{noisy} и $LSTM\text{-}SAC_{noisy}$) дообучены в шумной среде NoisyFlourEnv (аддитивный шум применяется к 158-мерному файлу параметров линии observation – obs-вектор).

10) У сети Колмогорова-Арнольда (KAN) с архитектурой [158 → 16 → 12] при одиночном train/val-разбиении дает заниженную регрессионную метрику коэффициента детерминации ($R^2 \approx 0{,}47$), что объясняется узким представлением и малой выборкой, а не принципиальной несводимостью задачи. При перемешивании и разбиении данных на несколько кластеров (k-fold кросс-валидации), метрика «наблюдение → оптимум k_{press}» восстанавливается с $R^2 \approx 0{,}98$, поэтому полученное вначале низкое значение метрики R^2 было исправлено. В реальной симуляции линии KAN при подобранной регуляризации устойчиво обыгрывает статическое управление. Расширение архитектуры KAN (увеличение bottleneck до [158 → 32 → 24] или [158 → 64 → 12]) и сравнение с текущими конфигурациями – направление дальнейшей работы.

11) В реализации Simulink-модели (predict_*.py в matlab_project/python/) две из семи компонент 158-мерного вектора наблюдений – это производные dh/dt и d^2h/dt^2 уровней бункеров – заполняются нулевыми значениями. Это корректное приближение в установившемся режиме (где обе производные малы по определению), но теоретически

может снижать результативность обучения (Distribution Shift) при работе с быстро меняющимися возмущениями. Хотя рассматриваемое оборудование имеет сравнительно долгий отклик на воздействия (~120–150 с), при использовании в иных типах оборудования в реальных условиях производства в нестационарных режимах необходимо добавить в модель блоки «Derivative» и подать 24 сигнала через блок вычислений производных (в MATLAB это Function Block в Custom Python Model Predict). В используемом Python-симуляторе производные вычислялись корректно в разделе `_bunker_step` и доступны в `result["$dh_{dt}$"]` / `result["$d2h_{dt2}$"]`.

12) В ходе аудита соответствия Python-симулятора оригинальной имитационной модели в Simulink были обнаружены и исправлены систематические расхождения параметров РВМ: 9 из 12 станков имели некорректные b_0 (до ×400) и *alpha*, для С4/С5/С6 была инвертирована формула зольности. После исправлений числовые результаты изменились, и были обновлены. По результатам данной работы, рекомендуется проводить независимую верификацию параметров симулятора с оригинальными Stateflow-чартами при любых изменениях физической модели.

13) При моделировании в качестве референсного значения был использован режим настройки оборудования, с максимально возможной эффективностью, согласно технико-экономической метрике кумулятивной выручки при установленных значениях цен получаемых продуктов (мука ВС – 27 руб./кг, мука 1С – 21 руб./кг, отруби – 7,7 руб./кг) за 1 час работы оборудования при стационарном входном потоке продукта (зерна) в значении паспортной величины размером 1736 г/с (~150 т/сут. по переработке зерна), который был установлен персоналом предприятия за 5 попыток по 6 часов чистого времени после завершения переходных процессов оборудования и нагрева вальцов. Данный режим использовал средние арифметические значения и был обозначен как Static.

Перечисленные ограничения не препятствуют решению поставленной в работе задачи сравнительного анализа методов нейросетевого управления; также они определяют направления дальнейших исследований.

Результаты

Было проведено сравнение нескольких групп результатов работы модели между собой, со статичными, предварительно настроенными персоналом параметрами k_{press} и адаптивной подстройкой k_{press} методом ESC всеми семью рассмотренными методами (ESC + шесть нейросетевых). Каждый вариант рассмотрен при невозмущенном/ стационарном входном потоке (steady state input) и возмущенном (time-varying input disturbance) входном потоке, на отрезке времени с 0 до 3600 с (1 час). Возмущённый режим запускался на пяти независимых задающих параметрах (*seeds* = 42, 88, 123, 240, 321), и результат для каждого метода представлен в виде среднего и стандартного отклонения; стационарный режим, будучи детерминированным, оценивался одним прогоном. В результате моделирования была получена кумулятивная стоимость выходного продукта на выходе мукомольного комплекса – функция стоимости *COST* (Рисунок 6). На рисунке оранжевым цветом показана стоимость продукта в секунду для варианта с ESC, синим – для варианта со статичным k_{press}, верхняя панель соответствует возмущённому входному воздействию, нижняя – стационарному

Полученные результаты кумулятивной стоимости продукта в рублях сведены в Таблицу 2 (три лучших выделены). Относительный прирост кумулятивной стоимости в процессе работы модели на 2-х разных вариантах (ручной и ESC) изображен на Рисунке 7. Эксперименты произведены независимо в двух имплементациях: оригинальной Simulink-модели Stage18 (значения для Static и ESC) и в чистой MATLAB-

имплементации `matlab_standalone`, эквивалентной по физике Simulink-модели. Абсолютные значения отличаются из-за разной модели входного потока $Q_{grain}(t)$ и шага дискретизации (dt = 1 с в `matlab_standalone` против непрерывного решателя solver в Simulink), но относительные приросты от перехода к нейросетевому управлению совпадают. Прирост по сравнению со Static в Таблице 2 указан по `matlab_standalone` (для ESC – по Simulink-reference, помечено *).

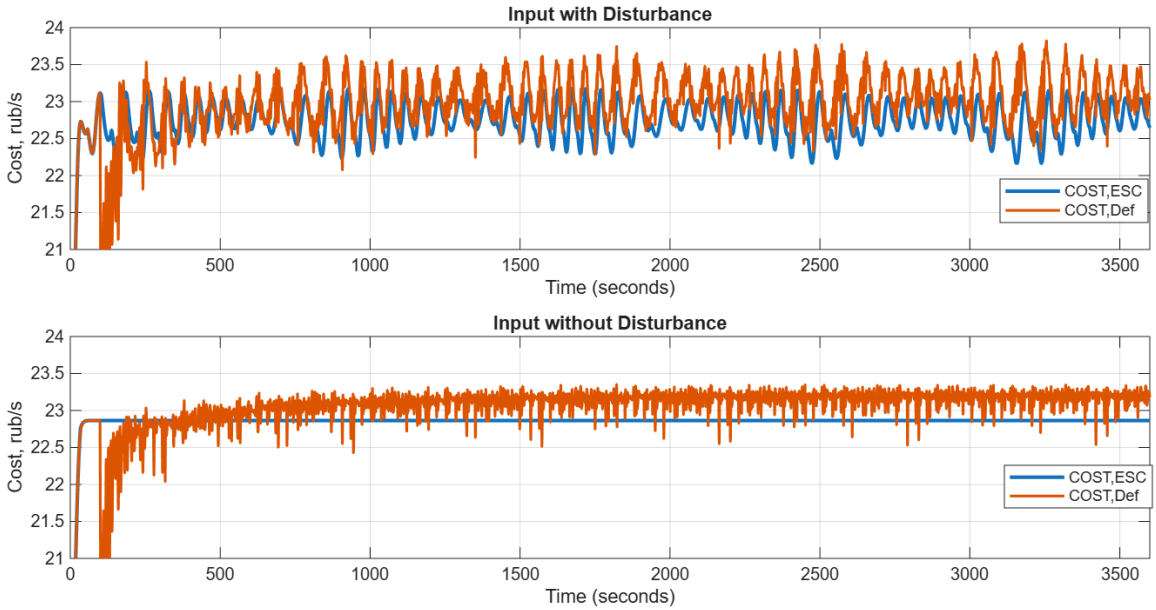


Рисунок 6 – Функция оценки стоимости *COST* выходного продукта
Figure 6 – Cost evaluation function *COST* for the output product

Таблица 2 – Кумулятивная стоимость продукта за 3600 с и относительный прирост по сравнению со Static
Table 2 – Cumulative product cost over a *3600*-second horizon and relative gain compared to the Static baseline

Метод	**Метрика выручки при стационарном режиме подачи продукта 1,73 кг/с (руб. за 1 час)**	**Прирост на стационарном режиме подачи при использовании методов оптимизации в сравнении со Static (%)**	**Прирост на нестационарном режиме подачи при использовании методов оптимизации в сравнении со Static (среднее за 5 сидов) (%)**
Static	65 190	–	–
ESC*	65 734	+0,84	+1,076
Трилинейная интерполяция	65 264	+0,11	+0,23
MLP	68 794	+5,53	+5,57
GP	68 753	+5,47	+5,54
SAC	68 240	+4,68	+4,68
LSTM-SAC	65 231	+0,06	+0,05
KAN	68 781	+5,51	+5,44

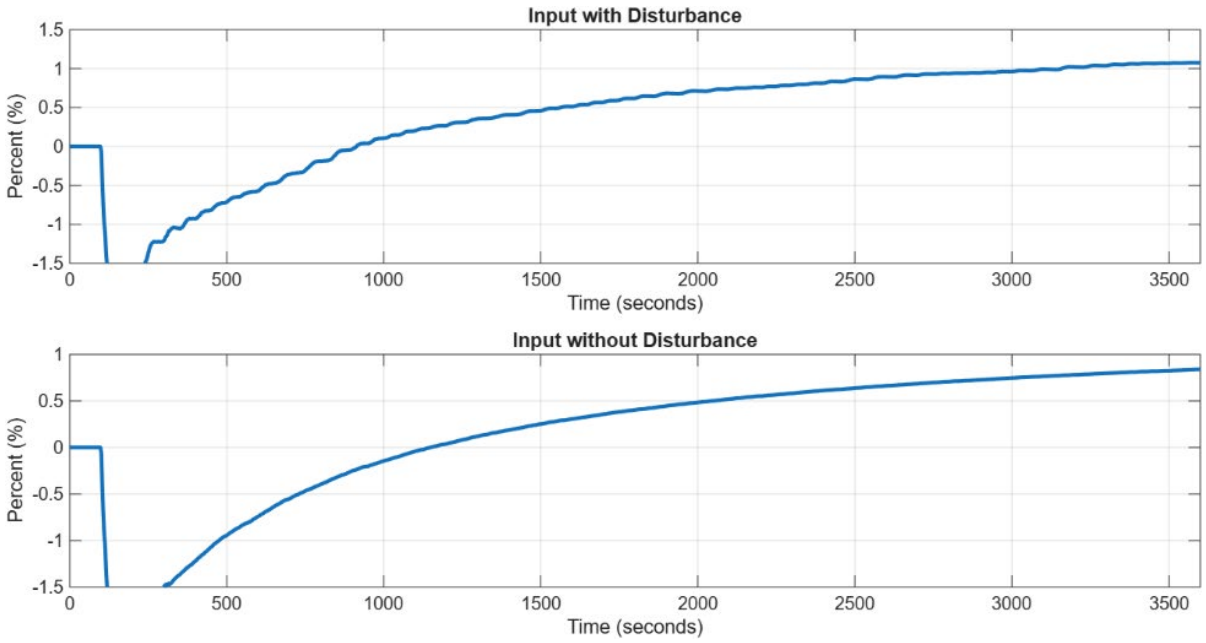


Рисунок 7 – Относительный прирост кумулятивной стоимости продукта в процессе работы за 3600 с для ESC
Figure 7 – Relative increase in cumulative product cost over a 3600 s operational period for the ESC method

Для оценки практической применимости предложенных методов проведен дополнительный эксперимент с аддитивным белым шумом на входной поток зерна Q_{in}. Реальные массовые расходомеры имеют типичную погрешность 2–10 %, поэтому испытаны три уровня шума: σ = 0,035 кг/с (низкий, ~2 %), σ = 0,087 кг/с (средний, ~5 %) и σ = 0,174 кг/с (высокий, ~10 %). Уровни шума: low ≈ 2 %, medium ≈ 5 %, high ≈ 10 % от Q_{nom} = 1,736 кг/с. Оцениваемые контроллеры тестировались с теми же весами, что и в основном эксперименте, без переобучения на зашумленных данных – оценивалась именно устойчивость, а не способность к адаптации. Однако в процессе эксперимента наблюдался положительный знак «деградации» выручки при росте шума не является показателем устойчивости контроллеров – это связано с проявлением неравенства Йенсена. Шум вводился в физический поток зерна Q_{in} при выпуклой (асимптотически приближаемой к некоторому максимальному значению) целевой функции *COST*, поэтому усреднение по реализациям систематически завышает выручку. Контрольная проверка подтверждает некорректность: при той же постановке фиксированный (статический) контроллер также «вырастает» на +2,6 %, хотя для неизменного управления реальный выигрыш невозможен (данные в Таблице 3). В результате была сделана методически корректная постановка эксперимента – введен шум в наблюдаемый вектор (модель датчиков), а выручка вычислялась по истинному состоянию линии; в этом случае средняя выручка с ростом σ шума не возрастает. Поэтому приведенные ниже значения (+1,17 % для SAC, +0,21 %, +2,2 % для MLP) следует читать как иллюстрацию артефакта, а не как достижение метода.

Таблица 3 – Результаты экспериментов по устойчивости к шуму на Q_{in} для всех методов без переобучения, выручка за 3600 с (руб.)
Table 3 – Experimental evaluation of noise robustness for input flow Q_{in} across all methods without overfitting: cumulative revenue over a 3600 s horizon (RUB.)

Метод	$\sigma \approx 2$ %	$\sigma \approx 5$ %	$\sigma \approx 10$ %	Деградация low→high (%)
Static	60 710	61 449	62 299	+2,62
ESC	60 658	61 400	62 238	+2,61
Трилинейная интерполяция	60 846	61 596	62 460	+2,65
MLP	63 968	64 592	65 376	+2,20
MLP (noise in sensors)*	63 968	63 924	63 879	−0,01
GP	63 958	64 582	65 365	+2,20
SAC	62 687	62 841	63 424	+1,17
SAC (noise in sensors)*	62 687	62 683	62 680	0,00
LSTM-SAC	60 730	61 507	62 357	+2,68
KAN	63 826	64 447	65 229	+2,20

При корректном эксперименте с введением шума в вектор данных датчиков и усреднении по 10 сидам, полученные значения показывают некоторое, но не значительное ухудшение результатов работы контроллеров при увеличении σ шума (Таблица 3, данные со звездочкой).

Корректные эксперименты с аддитивным белым шумом показали устойчивость нейросетевых контроллеров к шуму на входе при том, что обучение велось на детерминированных данных.

Одной из задач работы было изучение KAN сети в качестве регулятора и использование ее свойств, в том числе возможности символьной аппроксимации – предполагалось что это один из способов упростить и удешевить дорогостоящий сбор обучающего датасета (с возможностью запуска численного оптимизатора k_{press} в любой рабочей точке) компактным аналитическим описанием модели и последующим синтезом обучающих данных из этого описания с крайне малыми затратами времени. Такой подход дает интерпретируемый закон управления и позволяет за достаточно короткое время построить адаптивный в реальном масштабе времени контроллер без многочасового перебора оптимизатором, при обязательной верификации результата на имитационной Simulink-модели.

Оптимальная зависимость настройки прижима от нагрузки для каждого станка была аппроксимирована компактным полиномом $k_{press}(Q_{in}) = a + b \cdot Q_{in} + c \cdot Q_{in}^2$, коэффициенты которого определяются по выходам оптимизатора на сетке нагрузок $Q_{in} \in [1, 2]$. Качество аппроксимации по разным станкам неоднородно: для станков B3, C4, C5 коэффициент детерминации $R^2 = 0{,}95–0{,}97$ (зависимость выражена и хорошо описывается полиномом), для станков B1, B2 метрика $R^2 \approx 0{,}79–0{,}83$, тогда как для станков с мало изменяемыми настройками (C1A, C2, C3) полином ожидаемо вырождается. Уточнение: прямое извлечение формулы из обученной сети Колмогорова-Арнольда (функция «symbolic_formula» библиотеки pykan) для входа высокой размерности пока было выполнено непосредственно, и получилась функция, близкая к аффинному виду. Поэтому была использована полиномиальная аппроксимация по выходам оптимизатора; KAN рассматривается как метод, способный извлекать такие зависимости в принципе. Значения метрики RMSE для полиномиальной и KAN-аппроксимаций практически совпадают и составляют: B1 – 0,037; B2 – 0,029; B3 – 0,010; C4 – 0,047; C5 – 0,032; C6 – 0,083. Для станка C6 видна немонотонность: полиномиальная кривая загибается, тогда как аффинная аппроксимация KAN даёт прямую (Рисунок 8).

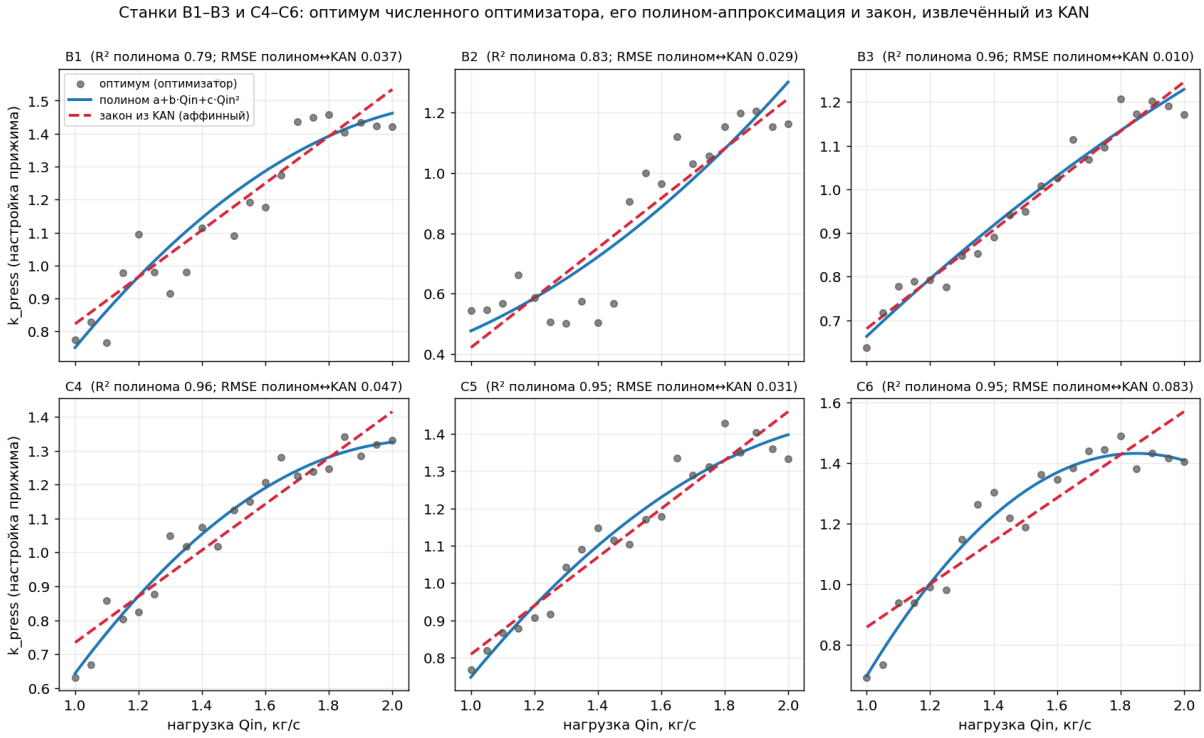


Рисунок 8 – Графики функций $k_{press}(Q_{in})$ полученные при 3-х аппроксимациях: численный оптимизатор Simulink, полиномиальная аппроксимация и аппроксимация сетью KAN для станков B1–B3, C4–C6

Figure 8 – Plots of the $k_{press}(Q_{in})$ functions obtained with three approximations: numerical Simulink optimizer, polynomial approximation, and KAN network approximation for mills B1–B3, C4–C6

На основе полученного закона был построен синтетический обучающий набор: разметка одной точки сводится к вычислению формулы (почти мгновенно) вместо запуска L-BFGS-оптимизатора. Сгенерировано 4500 пар «нагрузка → k_{press}» примерно за 5 с (проективная оценка эквивалентного времени работы оптимизатора – порядка 2000 с), с аугментацией данных шумом для робастности. На этом наборе примерно за 1,5 с обучается адаптивный контроллер с расчетом значений $K_{press}(Q_{in})$, реализованный на многослойном перцептроне.

Качество контроллера верифицировано на реальной Simulink-модели линии (а не только на программной копии модели): прирост кумулятивной выручки относительно статического управления составил +1,31 % усредненно по диапазону нагрузок и до +3,15 % в совпадающей рабочей точке, причем на длительном прогоне (3500 с) показатель стабилен во времени – деградации нет. На программной Python-копии модели прирост достигает +7,9 %; поскольку эта копия систематически оптимистичнее (примерно в 6 раз по величине прироста), ее значение приводится только как внутренняя верхняя оценка, а итоговым считается результат на Simulink-модели. Обученный контроллер корректно воспроизводит закономерности и результаты модели, заданные обучением с учителем (supervised learning), и не превосходит их; ценность подхода – в скорости и низкой стоимости построения проверяемого адаптируемого контроллера без дорогого сбора данных.

Эксперименты были проведены в трех параллельных каналах: Python-симулятор (`python_project/`), Simulink-модель (`matlab_project/`, в стадии валидации связки SLX-блоков с Python-контроллерами) и чистый MATLAB-симулятор (`matlab_standalone/`). Числа из MATLAB-имплементации идентичны Python-числам и подтверждают

результаты Таблиц 2, 3 (за исключением абсолютных значений Static/ESC, для которых в Таблице 2 использованы Simulink-reference значения).

Обсуждение и выводы

По результатам моделирования (Таблицы 2, 3) можно сделать следующие выводы. Итоговая стоимость выходного продукта, полученного за 3600 с работы модели, на 1,076 % выше с использованием ESC, чем при статичных, преднастроенных коэффициентах k_{press} для возмущенного воздействия и на 0,84 % выше, для невозмущенного. Это заметное увеличение эффективности производства. Если посмотреть на прирост общей стоимости, то видно, что в начальный момент времени вариант с ESC менее эффективен, чем статический вариант, так как начальные значения коэффициентов k_{press} взяты по нижней границе предела и равны 0,5. В промежутке времени от 0 до 100 с ESC отключен и модель работает в стационарном режиме, для выхода в рабочий режим. На 100 с включается ESC и начинает настраивать k_{press}. Из графика на Рисунке 7 видно, что эффективность ESC растет со временем и сохраняет этот тренд далее. Основные переходные процессы происходят в первые 120 с, в этот промежуток времени ESC активно изменяет значения регулируемых параметров для достижения максимума функции стоимости (COST), далее при приближении к максимуму, скорость регулирования падает, но стоимость продолжает рост. Это не противоречит результатам применения ESC в адаптивном управлении [17], но существенно проигрывает методам с применением нейронных сетей.

Три лидера по приросту кумулятивной выручки во всех режимах оказались (в сравнении с показателями работы персонала):

– MLP: +5,53 % и +5,57 % при стационарном и при возмущенном воздействии;

– GP: +5,47 % и +5,54 % при стационарном и при возмущенном воздействии;

– KAN: +5,51 % и +5,44 % при стационарном режиме и при возмущенном воздействии.

Однако в симуляции на модели линии KAN устойчиво дает лучший результат во всех режимах (стационарный, возмущенный, шум на Q_{in}). Парадокс объясняется тем, что KAN с сильной регуляризацией (*lamb* = 0,005 + entropy-регуляризация B-сплайнов) фактически выучил «среднее по диапазону» значение k_{press}, оптимальное по агрегированной целевой функции *COST* – а такое «среднее» оптимальное k_{press} оказывается лучше локально-точных предсказаний, склонных переобучаться на 436-точечном датасете. Таким образом, регрессионная метрика на валидации (R^2, *MAE*) плохо коррелирует с метриками производительности в симуляции для задач управления; при выборе моделей нужно ориентироваться на симуляционные, а не на регрессионные показатели. Кроме того, важна корректная подготовка данных, о чем упоминалось в упрощениях и ограничениях.

Дополнительный эксперимент с шумом на входной поток зерна выявил методический эффект: при выпуклой целевой функции усредненная выручка систематически завышается (вследствие эффекта неравенства Йенсена), из-за чего «деградация» получает положительный знак и для нейросетевых, и для фиксированного управления. Поэтому данный эксперимент следует трактовать не как демонстрацию устойчивости, а как предупреждение о корректной постановке оценки робастности: шум необходимо подавать в наблюдаемый вектор (модель датчиков), а выручку вычислять по истинному состоянию линии; в такой постановке средняя выручка с ростом шума не возрастает, а различия между методами статистически незначимы.

Особо можно отметить качественное изменение по сравнению с ранними экспериментами: на начальных стадиях экспериментов лидеры по стационарному

режиму показывали сомнительные результаты: GP (+6,31 %) и KAN (+8,14 %). После расширения наблюдательного вектора с 26 до 110, а потом до 158 признаков и пересинтеза датасета на корректной PBM-модели (где переменные $W_{зерна}$ и корректные цены *COST* реально влияют на оптимальные k_{press}), MLP с прежней архитектурой (два скрытых слоя по 128 нейронов) справился с обобщением – его средний прирост на возмущенном потоке составляет +5,57 %. Однако сети, использующие подход обучения с подкреплением, справились с заметно худшим результатом. Это иллюстрирует ограниченность простой feedforward-архитектуры на более сложных задачах и подтверждает необходимость моделей с встроенной регуляризацией (KAN, GP), либо обучения с подкреплением напрямую на симуляторе (SAC).

С практической точки зрения для непрерывного производства и развертывания локальной системы управления, по сумме достоинств и недостатков, возможно предпочтение будет у KAN: он один из лидеров во всех режимах (стационарный, возмущенный, шум на Q_{in}), демонстрирует малую дисперсию между сидами, быстро переучивается и в качестве бонуса дает интерпретируемые символьные приближения через B-сплайн-активации. Кроме того, имеет существенно меньшую вычислительную сложность по сравнению с остальными методами [16].

Заключение

В рамках исследования вопросов повышения эффективности эксплуатации технологического оборудования, в настоящей работе был предложен и экспериментально проверен двухуровневый подход к управлению работой технологической линии из 12 вальцовых станков. Нижний уровень управления реализован регулятором с нечеткой логикой работы – рассматривался для отдельного станка в работе [5]. Верхний уровень управления был реализован шестью способами – от классического Extremum-Seeking Control до нейросетевых методов (MLP, GP, SAC, LSTM-SAC, KAN) – что позволило выполнить корректное сравнение различных семейств алгоритмов (оптимизационных, регрессионных, рекуррентных и обучения с подкреплением) на единой имитационной модели мукомольной линии.

Установлено, что введение верхнего нейросетевого уровня для управления с помощью оптимизации значений регулируемых параметров отдельных станков k_{press} повышает кумулятивную выручку от работы технологической линии до +5,57 % относительно работы со статическими значениями регулировок k_{press}, обычно используемыми технологами вручную. Наилучший результат показали сети MLP (+5,53 %/+5,57 %), GP (+5,47 %/+5,54 %), KAN (+5,51 %/+5,44 %) при стационарном режиме и при возмущенном воздействии соответственно.

В работе на основе модели был получен синтетический датасет, который был аугментирован шумом, что позволило провести дополнительный эксперимент с аддитивным белым шумом на входной сигнал (σ = 2 – 10 % от номинала), который подтвердил практическую применимость предложенного подхода: все сетевые регуляторы в отличие от адаптивного регулятора ESC справились без существенной потери качества управления.

Эксперименты вывели необходимость методически корректного введения шума. Переучивание моделей с учетом шума – supervised-методов на синтетически зашумленной выборке (data augmentation) – дает для сетевых контроллеров результаты, сравнимые с данными без шума. Это делает предложенный подход, с использованием сетевых контроллеров, применимым в реальных производственных условиях с измерительными шумами, при осторожной настройке гиперпараметров аугментации для

каждой конкретной модели. Также в работе был сформулирован и применен ряд ограничений, которые могут использоваться в дальнейших исследованиях.

В развитие данного 2-уровневого подхода к управлению комплексом оборудования планируется применение PPO (Proximal Policy Optimization) в непрерывных пространствах действий [18], позволяющего повысить устойчивость обучения policy-gradient, а также имитационного обучения (Behavior Cloning, GAIL) от логов опытного оператора и от ESC-baseline. Имитационное обучение особенно перспективно при наличии длительных архивов работы реальной линии: оно позволит инициализировать политику нейросетевого контроллера на основе фактического человеческого опыта, что особенно важно в режимах, где симулятор PBM может расходиться с реальностью или в комплексах оборудования, где сложно или трудоемко построить релевантного цифрового двойника. Также дальнейшие исследования могут включать: учет корреляций между параметрами разных станков и расширение модели на сильно нестационарные режимы (запуск/останов линии); валидацию предложенных контроллеров на промышленных данных реальных производственных линий; применение разработанного подхода к гармонизации технологических линий иного типа и других отраслей промышленности (помол других культур, грануляция, обжиг).

СПИСОК ИСТОЧНИКОВ / REFERENCES

1. Melesse T.Y., Pasquale V.D., Riemma S. Digital twin models in industrial operations: a systematic literature review. *Procedia Manufacturing.* 2020;42:267–272. https://doi.org/10.1016/j.promfg.2020.02.084
2. Babich A., Senk D. Recent developments in blast furnace iron-making technology. In: *Iron Ore: Mineralogy, Processing and Environmental Sustainability*. Burlington: Elsevier; 2015. P. 505–547. https://doi.org/10.1016/B978-1-78242-156-6.00017-4
3. Баркалов С.А., Белоусов В.Е., Дорофеев Д.В., Нижегородов К.А., Серебрякова Е.А. Информационная модель поддержки процессов управления производственными потоками автоматизированного производства. *Вестник Южно-Уральского государственного университета. Серия: Компьютерные технологии, управление, радиоэлектроника.* 2023;23(4):47–56. https://doi.org/10.14529/ctcr230405
   Barkalov S.A., Belousov V.E., Dorofeev D.V., Nizhegorodov K.A., Serebryakova E.A. Information model of support of processes of management of production flows of the automated production. *Bulletin of the South Ural State University. Series: Computer Technologies, Automatic Control, Radio Electronics.* 2023;23(4):47–56. (In Russ.). https://doi.org/10.14529/ctcr230405
4. Knechtelsdorfer U., Steinboeck A., Schausberger F., Kugi A. A novel mass flow controller for tandem hot rolling mills. *Journal of Process Control.* 2021;104(1):168–177. https://doi.org/10.1016/j.jprocont.2021.07.002
5. Кудрявцев А.В., Анцупов Э.Э. Автоматическое управление вальцовым станком. *Моделирование, оптимизация и информационные технологии.* 2026;14(6). https://doi.org/10.26102/2310-6018/2026.57.6.003
   Kudriavtsev A.V., Antsupov E.E. Automated control of a roller mill. *Modeling, Optimization and Information Technology.* 2026;14(6). (In Russ.). https://doi.org/10.26102/2310-6018/2026.57.6.003
6. Niu D., Chen X.-S., Yang J., Zhou X.-P. Disturbance rejection control for Raymond mill grinding system based on disturbance observer. *Journal of Central South University.* 2017;24(9):2019–2027. https://doi.org/10.1007/s11771-017-3611-9

7. Das A., Bück A., Kumar J. Selection function in breakage processes: PBM and Monte Carlo modeling. *Advanced Powder Technology.* 2020;31(4):1457–1469. https://doi.org/10.1016/j.apt.2020.01.002
8. Campbell G.M., Bunn P.J., Webb C., Hook S.C.W. On predicting roller milling performance. Part II. The breakage function. *Powder Technology.* 2001;115(3):243–255. https://doi.org/10.1016/S0032-5910(00)00349-1
9. Park J., Sandberg I.W. Universal approximation using radial-basis-function networks. *Neural Computation.* 1991;3(2):246–257. https://doi.org/10.1162/neco.1991.3.2.246
10. Cybenko G. Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals and Systems.* 1989;2(4):303–314. https://doi.org/10.1007/BF02551274
11. Hagan M.T., Demuth H.B., De Jesús O. An introduction to the use of neural networks in control systems. *International Journal of Robust and Nonlinear Control.* 2002;12(11):959–985. https://doi.org/10.1002/rnc.727
12. Rasmussen C.E., Williams Ch.K.I. *Gaussian Processes for Machine Learning.* Cambridge: MIT Press; 2006. 248 p.
13. Haarnoja T., Zhou A., Abbeel P., Levine S. Soft Actor-Critic: Off-Policy Maximum Entropy Deep Reinforcement Learning with a Stochastic Actor. In: *Proceedings of the 35th International Conference on Machine Learning (ICML 2018), 10–15 July 2018, Stockholm, Sweden*. PMLR; 2018. P. 1856–1865. URL: https://arxiv.org/abs/1801.01290
14. Hochreiter S., Schmidhuber J. Long Short-Term Memory. *Neural Computation.* 1997;9(8):1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735
15. Greff K., Srivastava R.K., Koutník J., Steunebrink B.R., Schmidhuber J. LSTM: A Search Space Odyssey. *IEEE Transactions on Neural Networks and Learning Systems.* 2017;28(10):2222–2232. https://doi.org/10.1109/TNNLS.2016.2582924
16. Liu Z., Wang Y., Vaidya S., et al. *KAN: Kolmogorov-Arnold Networks.* arXiv. URL: https://arxiv.org/abs/2404.19756 [Accessed 16th April 2026].
17. Kreinovich V. Real-Time Optimization by Extremum-Seeking Control, by Kartik B. Ariyur and Miroslav Krstić. *Journal of Intelligent & Fuzzy Systems.* 2005;16(1):77. https://doi.org/10.3233/IFS-2005-00244
18. Schulman J., Wolski F., Dhariwal P., Radford A., Klimov O. *Proximal Policy Optimization Algorithms.* arXiv. URL: https://arxiv.org/abs/1707.06347 [Accessed 16th April 2026].

ИНФОРМАЦИЯ ОБ АВТОРАХ/INFORMATION ABOUT THE AUTHORS

Кудрявцев Александр Викторович, аспирант, Национальный исследовательский технологический университет «МИСИС», Москва, Российская Федерация
e-mail: alex.kud@mail.ru
ORCID: 0009-0004-4986-4982

Aleksander V. Kudriavtsev, Postgraduate, National University of Science and Technology "MISIS", Moscow, the Russian Federation.

Гуськов Сергей Валерьевич, независимый исследователь, Москва, Российская Федерация.
e-mail: gusevski.dev@gmail.com
ORCID: 0009-0004-7414-9227

Sergey V. Guskov, Independent Researcher, Moscow, the Russian Federation.

Анцупов Эрик Эрикович, независимый исследователь, Новосибирск, Российская Федерация.
e-mail: antsupoverik@yandex.ru
ORCID: 0009-0004-2650-0203

Eric E. Antsupov, Independent Researcher, Novosibirsk, the Russian Federation.

Статья поступила в редакцию 13.09.2026; одобрена после рецензирования 05.11.2026; принята к публикации 12.11.2026.

The article was submitted 13.09.2026; approved after reviewing 05.11.2026; accepted for publication 12.11.2026.