Математическое моделирование процесса распределения заявок в системе-агрегаторе производственных услуг

М.А. Лаверычев

Московский государственный технологический университет «Станкин»

Аннотация: В статье рассматривается математическая модель процесса распределения производственных заявок на платформе-агрегаторе. Предложен формальный подход на основе конечного автомата и сетей Петри для описания состояний заказов, их переходов и ограничений. Разработана комплексная модель, учитывающая критерии выбора исполнителя, отбора оборудования и анализа производственного процесса с учетом времени, стоимости и доступности ресурсов. Представленные подходы обеспечивают точность и эффективность при управлении заказами на цифровых производственных платформах.
Ключевые слова: математическое моделирование, конечный автомат, сеть Петри, цифровое производство, агрегатор услуг, выбор исполнителя.

В условиях цифровизации производства и активного роста платформ-агрегаторов производственных услуг проблема оптимального распределения заявок становится особенно актуальной. Современные агрегаторы требуют эффективных методов, которые учитывают многочисленные критерии: сроки исполнения, стоимость, доступность оборудования, удаленность исполнителей и их репутацию. В настоящее время существует ряд моделей, демонстрирующие различные подходы к моделированию процессов распределения производственных заявок:

1. Линейное и нелинейное программирование:

Просветов Г.И. рассматривает математические методы оптимизации логистических задач, в частности, минимизацию суммарных затрат и сроков выполнения [1,2]. Эти модели эффективны при четко определенных условиях, однако имеют ограничения в адаптивности при изменении внешних факторов.

2. Методы комбинаторной оптимизации:

Подходы, предложенные Богдановой П.А., Сахаровым Д.М. и Васильевой Т.В., ориентированы на многокритериальную оптимизацию и принятие решений в условиях неопределенности [3]. Эти модели обеспечивают хорошие результаты, но их сложность существенно увеличивается при росте числа критериев и объектов.

3. Подходы на основе теории графов и сетей Петри:

Исследования Питерсона Дж.Л. и Jiang W., Zhou K-Q. показывают, что сети Петри эффективны для моделирования и анализа динамических процессов, таких как управление производственными заказами [4,5]. Однако существующие реализации зачастую сталкиваются с проблемами масштабируемости и обработки больших объемов данных.

4. Использование искусственного интеллекта и машинного обучения:

В работе Zhang C., Xie G., предлагается использовать адаптивные алгоритмы, основанные на машинном обучении для управления производственными задачами [6]. Такие методы отличаются высокой точностью, но требуют значительных вычислительных ресурсов и качественной подготовки данных.

5. Стохастические модели и имитационное моделирование:

Chowdhury S.N., Ray A., Dana S.K., Ghosh D. изучают применение стохастических моделей для прогнозирования поведения производственных систем при случайных воздействиях и отклонениях [7]. Эти модели позволяют учитывать неопределенность, однако требуют большого объема статистических данных.

Общий анализ существующих исследований показывает, что имеющиеся модели имеют ограничения по одному или нескольким параметрам: гибкости, масштабируемости, учету множественных критериев и неопределенности.

Научная проблема заключается в недостаточной эффективности существующих подходов к управлению распределением производственных заявок, которые не обеспечивают требуемую точность и оперативность в условиях динамических изменений цифровых производственных платформ.

Цель исследования – разработать усовершенствованную математическую модель распределения производственных заявок, способную учитывать множественные критерии оптимальности и динамические условия выполнения заказов с использованием формализма конечных автоматов и сетей Петри. Преимущества этих методов:

- высокая степень формализации процессов;
- удобство в моделировании состояний и их переходов;
- возможность наглядного представления и анализа динамических процессов;
- гибкость и адаптивность к изменению условий выполнения заказов.

Модель построена на основе конечного автомата [8,9]. Дополнительно используется сеть Петри для наглядной визуализации и анализа процессов распределения заявок. Конечный автомат – основа модели, которая явно описывает:

- Состояния заказа: создан, оценён, оплачен, в производстве, выполнен, отменён.
- Переходы между состояниями: через операции (создание, оценка, оплата, производство и т.д.).
- Правила переходов: например, оплата возможна только после оценки.

Расширенная формализация включает:

- Множества сущностей (пользователи, заказы, платежи).
- Функции и операции, изменяющие состояние системы (например, createOrder, processPayment).
- Ограничения и инварианты, гарантирующие корректность данных (например, сумма платежа должна соответствовать оценке стоимости).

Контекст применения: модель также можно отнести к "динамическим системам с дискретными событиями" [10], так как она описывает реакцию системы на действия участников (пользователей, отдела оценки, производителей). Определим основные элементы модели.

Основные множества:

- U={u1,u2,...,un} — множество пользователей.
- O={o1,o2,...,om} — множество заказов.
- D={d1,d2,...,dm} — множество деталей заказов.
- P={p1,p2,...,pk} — множество платежей.
- M={m1,m2,...,ml} — множество производителей.
- S={Создан, Оценён, Оплачен, Выбран исполнитель, В производстве, Выполнен, Отменён} — множество статусов заказа.

Атрибуты сущностей:

- Пользователь u∈U:
- name(u)∈String,
- email(u)∈String,
- password(u)∈String.
- Детали заказа d∈D:
- description(d)∈String,
- drawings(d)⊆File,
- deadline(d)∈Date.
- Заказ o∈O:
- orderId(o)∈N,
- details(o)∈D,
- costEstimate(o)∈R+,
- status(o)∈S.
- Платёж p∈P:

- $paymentId(p) \in N$,
- $amount(p)=costEstimate(o)$ (для связанного заказа оо),
- $paymentDate(p) \in Date$.

Функции и операции:

1. Создание заказа:

   $createOrder: U \times D \to O,\ (u,d) \mapsto o$, где $status(o)=$Создан.

2. Оценка заказа техническим отделом:

   $evaluateOrder: O \to O, o \mapsto o'$, где:

   - $costEstimate(o')=f(description(d),drawings(d),deadline(d))$
   - $status(o')=$Оценён.

3. Обработка платежа:

   $processPayment: P \times O \to O,(p,o) \mapsto o'$, где:

   - $status(o')=$Оплачен,
   - $paymentDate(p) \geq dateCreated(o)$.

4. Передача в производство:

   $produceOrder: M \times O \to O,(m,o) \mapsto o'$, где $status(o')=$В производстве.

5. Обновление статуса:

   $updateStatus: O \times S \to O,(o,s) \mapsto o'$, где $status(o')=s$.

Ограничения и инварианты:

1. Согласованность данных:

   - $\forall o \in O\ \exists! u \in U, d \in D: details(o)=d$.
   - $\forall p \in P\ \exists! o \in O: amount(p)=costEstimate(o)$.

2. Порядок операций:

   - $processPayment(p,o)$ возможен только при $status(o)=$Оценён.
   - $produceOrder(m,o)$ возможен только при $status(o)=$Оплачен.

3. Временные ограничения:

   - $\forall d \in D: deadline(d) \geq currentDate$ на момент создания заказа.

- $\forall p \in P: paymentDate(p) \geq dateCreated(o)$.

На основе разработанной математической модели можно построить сеть Петри для состояний заказа (рис.1).

Рис.1. – Сеть Петри для состояний заказа.

Формальная запись системы описывается кортежем:

$$\sum = \langle U, O, D, P, M, S, F \rangle,$$

где F={createOrder,evaluateOrder,processPayment,produceOrder,updateStatus}.

Состояние системы изменяется согласно:

$$\Sigma_{t+1} = F_j(\Sigma_t, \text{входные данные}).$$

Для формализации этапов процесса выполнения заказа, критериев принятия решений и зависимости между временем, стоимостью и ресурсами была создана модель, состоящая из следующих компонентов:

1. Входные данные. Запрос от Заказчика:

$$Q=(D, R),$$

где D – техническая документация,

R – требования (сроки, качество, объем).

2. Расчет стоимости и времени (Отдел Технологической Подготовки (ОТП)). Функция ОТП:

$$f_{ОТП}(D,R) \rightarrow (C_{ОТП}, T_{ОТП}),$$

где $C_{ОТП}$ – расчетная стоимость,

$T_{ОТП}$ – расчетное время.

3. Проверка доступности ресурсов производственного предприятия (ПП). Для каждого

$$ПП_i \in \{ПП_1, ПП_2, ..., ПП_n\}$$

условие доступности:

$$available_i = \begin{cases} 1, \text{если } \sum_{t=t_{current}}^{t_{current}+T_{\text{ОТП}}} R^i_{available}(t) \geq R_{required} \\ 0, \text{иначе.} \end{cases}.$$

Если $available_i = 1$, $ПП_i$ предоставляет C_i и T_i.

4. Выбор оптимального исполнителя. Целевая функция:

$$ПП^* = argmin\,(C_i + \alpha * T_i$$

где α – коэффициент значимости времени.

5. Согласование с Заказчиком. Вероятность подтверждения:

$$p(C^*, T^*),$$

где $C^* = C_{ОТП} + C_{ПП^*}$, $T^* = T_{ОТП} + T_{ПП^*}$.

6. Выполнение заказа. Фактическое время выполнения:

$$T_{execute} = T_{ПП^*} + \varepsilon,\ \varepsilon \sim N(0, \sigma^2),$$

где ε – случайная задержка.

7. Оплата. Общая стоимость для Заказчика:

$$C_{total} = C_{ОТП} + C_{ПП*} + C_{system},$$

где C_{system} – комиссия системы.

8. Временная модель. Общее время процесса:

$$T_{total} = t_{ОТП} + k \cdot (t_{request} + t_{response}) + t_{selection} + t_{approval} + T_{execute},$$

где k – количество проверенных ПП,

$t_{request}$, $t_{response}$ – время запроса и ответа.

Диаграмма состояний имеет следующий вид:

$$S = \{S_0, S_1, S_2, S_3, S_4, S_5, S_6\}:$$

где S_0 – получение запроса;

S_1 – расчет ОТП;

S_2 – опрос ПП (цикл);

S_3 – выбор исполнителя.

S_4 – согласование;

S_5 – выполнение заказа;

S_6 – завершение.

Переходы отражаются схемой

$$S_0 \rightarrow S_1 \rightarrow S_2 \rightarrow S_3 \rightarrow S_4 \rightarrow S_5 \rightarrow S_6.$$

Стохастические элементы:

1. Вероятность подтверждения:

$$p(C^*, T^*) = \frac{1}{1 + e^{-k(C_{ref} - C^* + \gamma(T_{ref} - T^*))}}$$

2. Задержка выполнения: $\varepsilon \sim \exp(\lambda)$. Это распределением означает, что случайное отклонение времени выполнения заказа от планового срока моделируется с помощью экспоненциального распределения. Это ключевой элемент стохастической модели, учитывающий неопределённость в производстве. $\lambda > 0$ — параметр интенсивности (число задержек в единицу времени).

Процессы отбора и выбора исполнителя описаны на рис.2.

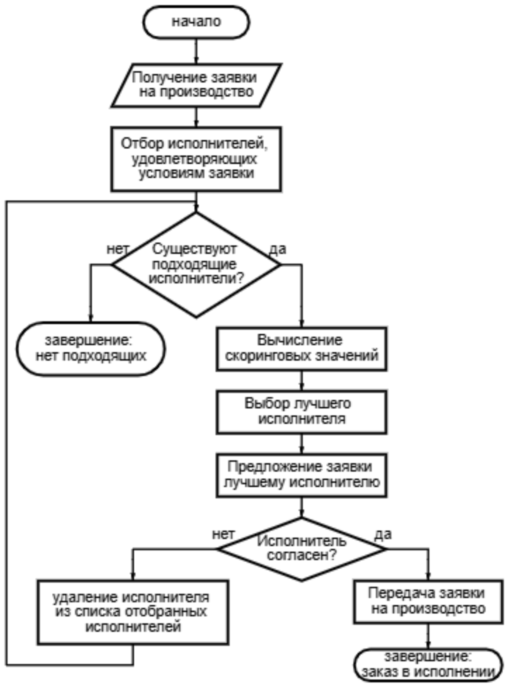

Рис.2 – Блок-схема процессов отбора и выбор исполнителей.

Математическое описание принципа отбора производственного оборудования по критерию соответствия требованиям технологической документации можно формализовать следующим образом:

1. Определение множеств и параметров:

- $E=\{E_1,E_2,...,E_k\}$ – множество доступного оборудования у исполнителя, где каждое оборудование E_i характеризуется набором параметров:

$$E_i=(e_{i1},e_{i2},...,e_{in}),$$

где e_{ij} – значение ij-го параметра оборудования E_i.

- $T=\{t1,t2,...,tm\}$ – множество требований технологической документации, где каждое требование tj задает ограничение для параметра оборудования. Ограничение может быть выражено:

- минимальным/максимальным значением: $t_j=[a_j,b_j]$,

- точечным значением: $t_j=c_j t_j=c_j$,

- логическим условием (например, наличие функции).

2. Функция соответствия

Для каждого требования t_j определим предикат соответствия $P_j(E_i)$, который принимает значение true, если оборудование E_i удовлетворяет требованию t_j, и false в противном случае:

$$\boldsymbol{P_j(E_i)} = \begin{cases} true, \text{ если } e_{ij} \in [a_j, b_j] \text{ (для диапазона)}, \\ true, \text{ если } e_{ij} = c_j \text{ (для точечного значения)}, \\ true, \text{ если параметр } e_{ij} \text{ существует и соответствует} \\ false, \text{ иначе} \end{cases}$$

3. Условие отбора оборудования

Оборудование E_i считается соответствующим технологической документации, если все требования T выполнены:

$$\boldsymbol{E}^* = \{E_i \in E | \forall_j \in \{1,2,\dots,m\}, P_j(E_i) = true\},$$

где E^* – подмножество оборудования, удовлетворяющего всем требованиям.

Математически принцип отбора сводится к проверке выполнения системы условий $P_j(E_i)$=true для всех j. Оборудование включается в итоговое множество E^* только при полном соответствии требованиям T.

Например, пусть для операции требуется оборудование со следующими параметрами:

- Мощность≥100 кВт (t_1),
- Точность≤0.1 мм (t_2),
- Наличие системы ЧПУ (t_3).

Тогда для оборудования

$$E_i=(150,\ 0.05,\ \text{ЧПУ}):$$

$$P_1(E_i)=\text{true},\ P_2(E_i)=\text{true},\ P_3(E_i)=\text{true} \Rightarrow E_i \in E^*.$$

Для выбора исполнителя на платформе-агрегаторе предлагается следующая формула, объединяющая три критерия: сроки выполнения, расстояние и рейтинг [8, 9].

$$Rate_i = \omega_T \cdot \frac{T_{min}}{T_i} + \omega_D \cdot \frac{D_{min}}{D_i} + \omega_R \cdot R_i,$$

где ω_T, ω_D, ω_R – веса критериев ($\omega_T + \omega_D + \omega_R = 1$). Определяют приоритет критериев;

$T_{min} = \min(T_i)$, $D_{min} = \min(D_i)$ – минимальные срок и расстояние среди исполнителей;

T_i – срок выполнения заказа исполнителем i;

D_i – расстояние между заказчиком и исполнителем;

R_i – рейтинг исполнителя i, рассчитываемый как:

$$R_i = \frac{S_i+\alpha}{S_i+F_i+\beta} \cdot \left(1 + \gamma \cdot \frac{L_i}{L_{max}}\right),$$

где S_i – успешные заказы, F_i – проваленные заказы;

L_i – продолжительность работы исполнителя в системе, $L_{max} = \max(L_i)$;

α, β – параметры регуляризации (например, α=1, β=2);

γ – коэффициент влияния продолжительности работы в системе (например, γ=0.1).

Рассмотрим основные критерии.

1. Срок выполнения (T_i):

- Нормализация: $\frac{T_{min}}{T_i}$. Исполнитель с минимальным сроком получает 1, остальные — пропорционально хуже.
- Чем меньше T_i, тем выше оценка.

2. Расстояние (Di):

- Нормализация: $\frac{D_{min}}{D_i}$. Ближайший исполнитель получает 1.
- Учитывает логистическую эффективность.

3. Рейтинг (Ri):

- Успешность: $\frac{S_i+\alpha}{S_i+F_i+\beta}$ – доля успешных заказов с регуляризацией (чтобы избежать крайних значений для новых исполнителей).
- Стаж: множитель $1+\gamma\cdot\frac{L_i}{L_{max}}$ повышает рейтинг опытных исполнителей.
- Пример: при γ=0.1, максимальный стаж увеличивает рейтинг на 10%.

Настройка параметров осуществляется следующим образом:

- Веса (wT, wD, wR) определяют приоритет критериев. Например, wT=0.5, wD=0.3, wR=0.2 – акцент на сроки.
- Регуляризация (α, β) задает априорную оценку для новых исполнителей. Например, α=1, β=2 соответствует априорной успешности 1/3.
- Влияние стажа (γ) контролирует, насколько стаж влияет на рейтинг.

Данная формула обладает гибкостью, учитывает баланс критериев и обладает устойчивостью.

К предложенной формуле можно применить критерии оптимизации, чтобы улучшить качество выбора исполнителей. Оптимизация может быть направлена на настройку параметров формулы: весов w_T, w_D, w_R и коэффициентов α, β, γ) или на адаптацию формулы под конкретные цели платформы.

Параметры формулы (веса и коэффициенты) можно оптимизировать, чтобы максимизировать ключевые метрики платформы. Для этого нужны исторические данные о заказах, исполнителях и их результатах.

Примеры целевых функций (критериев оптимизации) [10]:

- Максимизация успешности заказов:

$$Maximize:\ \sum_{\text{заказы}}(\text{заказ успешен})\ ,$$

где выбор исполнителя определяется формулой с текущими параметрами.

- Минимизация среднего времени выполнения:

$$Minimize:\ \frac{1}{N}\sum_{i=1}^{N} T_i,$$

где T_i– время выполнения i-го заказа.

- Максимизация удовлетворенности клиентов (на основе оценок заказчиков):

$$Maximize:\ \frac{1}{N}\sum_{i=1}^{N} Rate_i.$$

- Баланс между скоростью и качеством:

$$Maximize:\ \frac{1}{N}\sum_{i=1}^{N}(\lambda \cdot \frac{1}{T_i} + (1-\lambda)\cdot Rate_i),$$

где λ ∈ [0,1]— коэффициент важности скорости.

Методы оптимизации:

- Градиентный спуск: если формула дифференцируема по параметрам (например, весам wT, wD, wR).

- Генетические алгоритмы: для поиска оптимальных параметров в нелинейных и многомерных пространствах.

- A/B-тестирование: Сравнение разных наборов параметров на реальных данных.

Рассмотрим варианты оптимизации.

1. Многокритериальная оптимизация:

Если платформа стремится одновременно улучшить несколько метрик (например, скорость, качество, стоимость), можно использовать методы многокритериальной оптимизации:

- Парето-оптимальность: поиск компромиссных решений, где улучшение одного критерия не ухудшает другой.
- Взвешенная сумма критериев:

$$\text{Общий критерий} = \mu_1 \cdot \text{Успешность} + \mu_2 \cdot \text{Скорость} + \mu_3 \cdot \text{Стоимость},$$

где μ_1, μ_2, μ_3– приоритеты платформы.

2. Оптимизация для новых исполнителей:

Для новых исполнителей с малым количеством заказов ($S_i+F_i\approx 0$) формула рейтинга R_i может давать заниженные оценки. Чтобы это исправить можно ввести байесовский рейтинг:

$$Rate_i = \frac{S_i+\alpha}{S_i+F_i+\alpha} \cdot \left(1+\gamma \cdot \frac{L_i}{L_{max}}\right) + \delta,$$

где δ – бонус для новых исполнителей.

Предложенная в данной работе математическая модель основывается на использовании формализма конечных автоматов и сетей Петри, что позволяет комплексно и гибко подходить к решению задачи распределения заявок. В отличие от классических подходов, которые ориентированы на минимизацию издержек или времени исполнения задачи и зачастую статичны по отношению к изменяющимся условиям, предложенная модель характеризуется высокой адаптивностью к динамическим изменениям условий работы платформ-агрегаторов.

Основным преимуществом предлагаемой модели является интегральный подход, обеспечивающий одновременный учет множества критериев – временных, стоимостных, логистических и качественных. Благодаря детальной формализации процессов и состояний заказов с помощью конечных автоматов и визуализации с помощью сетей Петри, обеспечивается прозрачность и наглядность управления процессами. Это особенно важно в условиях неопределенности и высокой динамичности цифрового производства, где оперативность и точность принятия решений играют решающую роль в успешности работы платформы.

Кроме того, предлагаемая модель позволяет реализовать автоматизацию принятия решений, существенно снижая затраты времени и ресурсов на управление и обработку заявок. В условиях быстрого роста объема данных и задач это преимущество становится особенно важным для масштабируемости системы.

Перспективы дальнейших исследований по доработке математической модели связаны с:

1. Уточнением параметров модели: провести анализ чувствительности параметров модели (например, веса критериев и коэффициенты значимости). Это поможет выбрать оптимальные настройки и улучшить качество прогнозирования и принятия решений.
2. Расширением критериев: добавить дополнительные критерии, такие как риски задержек или поломок оборудования. Это повысит реалистичность и прикладную ценность модели.
3. Использованием исторических данных: внедрить возможность адаптивного изменения параметров модели на основе анализа исторических данных платформы. Это позволит модели самостоятельно оптимизировать параметры в процессе работы.
4. Стохастическим моделированием: более подробно интегрировать стохастические элементы (например, случайные задержки, колебания спроса). Это сделает модель более устойчивой к неопределенности реальных производственных условий.
5. Проверкой на реальных данных: провести апробацию модели на реальных данных существующих цифровых производственных платформ,

что позволит выявить скрытые недостатки и повысить практическую значимость результатов.

6. Усовершенствованием визуализации: улучшить визуализацию с помощью более совершенных инструментов и программных решений для наглядного представления состояний и процессов, что повысит удобство анализа и принятия решений пользователями модели.

Предлагаемая модель может быть эффективно применена на платформах-агрегаторах, которые занимаются координацией множества независимых производителей и заказчиков, обеспечивая выбор оптимального исполнителя и оборудования на основе заранее заданных критериев.

В результате проведенного исследования разработана усовершенствованная математическая модель для управления распределением производственных заявок на цифровых платформах-агрегаторах. Применение модели обеспечивает повышение эффективности, точности и оперативности управления заказами.

Литература

1. Просветов Г.И. Математические методы в логистике: задачи и решения. М: Альфа-Пресс, 2008. 302 с.

2. Имамкулиева Т. М., Шыхгулыева А. Х. Математические методы в анализе рисков и управлении рисками в логистике // Всемирный ученый. 2024. Т. 1. №28. С. 48-56.

3. Богданова П. А., Сахаров Д. М., Васильева Т. В. Обзор методов многокритериальной оптимизации в задачах принятия решений // Инновационные аспекты развития науки и техники. 2021. №6. С. 153-157.

4. Питерсон Дж.Л. Теория сетей Петри и моделирование систем. М: Мир, 1984. 265 с.

5. Jiang W., Zhou K-Q., Sarkheyli-Hägele A., Zain A.M. Modeling, reasoning, and application of fuzzy Petri net model: a survey // Artificial Intelligence Review. 2022. V.55. №8. pp. 6567-6605.

6. Zhang C., Xie G., Wang X. DynamicTuple: The dynamic adaptive tuple for high-performance packet classification // Computer Networks. 2022. V. 202. URL: doi.org/10.1016/j.comnet.2021.108630.

7. Chowdhury S. N., Ray A., Dana S.K., Ghosh D. Extreme events in dynamical systems and random walkers: A review // Physics Reports. 2022. V. 966. pp. 1-52.

8. Филатов В.А., Козырь О.Ф. Модель поведения автономного сценария в задачах управления распределенными информационными ресурсами // Инженерный вестник Дона, 2013, №3 URL: ivdon.ru/ru/magazine/archive/n3y2013/1771.

9. Хопкрофт Дж.Э., Ульман Дж.Д. Введение в теорию автоматов, языков и вычислений. М: Диалектика, 2019. 528 с.

10. Парийская Е. Ю. Сравнительный анализ математических моделей и подходов к моделированию и анализу непрерывно-дискретных систем // Дифференциальные уравнения и процессы управления. 1997. №1. С. 91-120.

References

1. Prosvetov G.I. Matematicheskie metody v logistike: zadachi i resheniya [Mathematical methods in logistics: problems and solutions]. M: Al'fa-Press, 2008. 302 p.

2. Imamkulieva T. M., Shykhgulyeva A. Kh. Vsemirnyy uchenyy. 2024. V. 1. №28. pp. 48-56.

3. Bogdanova P. A., Sakharov D. M., Vasil'eva T. V. Innovatsionnye aspekty razvitiya nauki i tekhniki. 2021. №6. pp. 153-157.

4. Piterson Dzh.L. Teoriya setey Petri i modelirovanie system [Theory of Petri nets and modeling of systems]. M: Mir, 1984. 265 p.

5. Jiang W., Zhou K-Q., Sarkheyli-Hägele A., Zain A.M. Artificial Intelligence Review. 2022. V.55. №8. pp. 6567-6605.

6. Zhang C., Xie G., Wang X. Computer Networks. 2022. V. 202. URL: doi.org/10.1016/j.comnet.2021.108630.

7. Chowdhury S. N., Ray A., Dana S.K., Ghosh D. Physics Reports. 2022. V. 966. pp. 1-52.

8. Filatov V.A., Kozyr' O.F. Inzhenernyj vestnik Dona, 2013, №3 URL: ivdon.ru/ru/magazine/archive/n3y2013/1771.

9. Khopkroft Dzh.E., Ul'man Dzh.D. Vvedenie v teoriyu avtomatov, yazykov i vychisleniy [Introduction to the theory of automata, languages and computations]. M: Dialektika, 2019. 528 p.

10. Pariyskaya E. Yu. Differentsial'nye uravneniya i protsessy upravleniya. 1997. №1. pp. 91-120.

Дата поступления: 6.06.2025

Дата публикации: 25.07.2025