ПЕРЕВОД СТАТЬИ

Intelligent AI Delegation

Интеллектуальное делегирование на открытом рынке ИИ-агентов

Авторы: Nenad Tomašev, Matija Franklin, Simon Osindero (Google DeepMind)

Дата первой публикации: 12 февраля 2026

Источник: https://arxiv.org/abs/2602.11865v1

Перевод подготовлен Екатериной Якуненко

https://delay-rag.ru ♦ https://t.me/delay_RAG

Содержание

1. Введение

По мере того, как современные ИИ-агенты выходят за рамки парадигмы «запрос–ответ», их полезность всё больше определяется тем, насколько эффективно они могут разбивать сложные задачи на управляемые подзадачи и делегировать их выполнение. Эта парадигма координации лежит в основе приложений самого разного масштаба — от персональных, где ИИ-агенты выступают личными ассистентами (Gabriel et al., 2024), до корпоративных внедрений, где они автоматизируют рабочие процессы и обеспечивают поддержку (Huang and Hughes, 2025; Shao et al., 2025; Tupe and Thube, 2025).

Большие языковые модели (*LLM*) уже продемонстрировали перспективность в робототехнике (Li et al., 2025a; Wang et al., 2024a), обеспечивая более интерактивную и точную спецификацию целей и обратную связь. Недавние работы также рассматривают возможность крупномасштабной координации ИИ-агентов в рамках виртуальных рынков агентов (Tomasev et al., 2025).

Современные агентные ИИ-системы реализуют сложные потоки управления между дифференцированными субагентами с использованием централизованных или децентрализованных протоколов оркестрации (Hong et al., 2023; Rasal and Hauer, 2024; Song et al., 2025; Zhang et al., 2025a). Это уже можно рассматривать как своего рода микрокосм декомпозиции задач и делегирования, где процесс фиксирован и существенно ограничен. Управление динамическими взаимодействиями в масштабе Интернета требует выйти за рамки подходов, применяемых в нынешних эвристических многоагентных платформах.

Делегирование (Castelfranchi and Falcone, 1998) — это больше, чем просто разбиение задачи на управляемые подзадачи. Помимо создания подзадач, делегирование требует распределения ответственности и полномочий (Mueller and Vogelsmeier, 2013; Nagia, 2024) и, следовательно, предполагает подотчётность за результаты. Делегирование предусматривает оценку рисков, которая может быть скорректирована с учётом доверия (Griffiths, 2005).

Делегирование также включает сопоставление способностей и непрерывный мониторинг эффективности, динамические корректировки на основе обратной связи и обеспечение выполнения делегированной задачи в рамках установленных ограничений. Существующие подходы, как правило, не учитывают эти факторы, опираясь в большей степени на эвристику и/или более простую параллелизацию. Этого может быть достаточно для ранних прототипов, но развёртывание ИИ в реальных условиях должно выйти за рамки ситуативного, хрупкого и ненадёжного делегирования.

Существует насущная потребность в системах, способных динамически адаптироваться к изменениям (Acharya et al., 2025; Hauptman et al., 2023) и восстанавливаться после

ошибок. Отсутствие адаптивных и надёжных фреймворков внедрения остаётся одним из ключевых ограничивающих факторов для ИИ-приложений в условиях высоких рисков.

Для полного раскрытия потенциала ИИ-агентов необходимо **интеллектуальное (intelligent) делегирование**: надёжная система, основанная на чётко определённых ролях, границах, репутации, доверии, прозрачности, верифицируемых способностях агентов и поддающемся проверке выполнении задач, с возможностью масштабируемого распределения задач. В данной работе мы предлагаем структуру интеллектуального делегирования задач, направленную на преодоление этих ограничений; она опирается на исторический опыт человеческих организаций и ключевые требования безопасности агентных систем.

2. Основы интеллектуального делегирования

2.1. Определение

Мы определяем интеллектуальное делегирование как последовательность решений, связанных с распределением задач, которая также включает передачу полномочий, ответственности, подотчётности, чёткие спецификации ролей и границ, ясность намерений и механизмы установления доверия между двумя (или более) сторонами. Сложные задачи могут дополнительно включать этапы декомпозиции, а также поиск и оценку имеющихся компетенций (*capability lookup*) и их сопоставление с требованиями задачи (*capability matching*) для обоснованного делегирования.

Когда мы говорим о делегировании задач, мы обычно подразумеваем, что эти задачи выходят за рамки базового уровня сложности, который может обработать системная подпрограмма — подобный элементарный аутсорсинг также требует внимания, однако по масштабу он несравнимо скромнее. На другом конце спектра возможно взаимодействие с агентами, которым предоставлена полная автономия и которые могут свободно преследовать любое количество подцелей без явных проверок и разрешений (Kasirzadeh and Gabriel, 2025). В крайнем случае таким полностью автономным агентам потребовалось бы доверять решение этических вопросов (Sloksnath, 2025), хотя это может оказаться тем, что мы никогда не захотим допустить, поскольку современные агенты крайне ограничены в своей способности участвовать в подобных решениях (Haas, 2020; Mao et al., 2023; Reinecke et al., 2023). Мы считаем такой открытый сценарий уместным для нашего обсуждения, но лишь в той мере, в какой могут быть установлены надлежащие механизмы для обеспечения безопасности более автономного выполнения задач.

2.2. Аспекты делегирования

Поскольку делегирование может принимать различные формы, мы вводим несколько измерений (осей), которые помогают структурировать различные случаи делегирования и сделать их более поддающимися анализу.

1. **Делегатор.** Человек или ИИ.
2. **Делегат.** Человек или ИИ.
3. **Характеристики задачи:**

   (а) *Сложность* — степень сложности, присущая задаче, часто коррелирующая с числом подэтапов и уровнем требуемого рассуждения.

   (б) *Критичность* — мера важности задачи и серьёзность последствий, связанных с её провалом или неоптимальным выполнением.

   (в) *Неопределённость* — уровень неопределённости относительно среды, входных данных или вероятности достижения успешного результата.

   (г) *Продолжительность* — ожидаемые временные рамки выполнения задачи — от мгновенных подпрограмм до длительных процессов, занимающих дни или недели.

   (д) *Стоимость* — экономические или вычислительные затраты на выполнение задачи, включая расход токенов, плату за API и потребление энергии.

   (е) *Требования к ресурсам* — конкретные вычислительные ресурсы, инструменты, права доступа к данным или человеческие компетенции, необходимые для выполнения задачи.

   (ж) *Ограничения* — операционные, этические или правовые рамки, в пределах которых должна выполняться задача.

   (з) *Проверяемость* — относительная сложность и стоимость верификации результатов выполнения задачи. Задачи с высокой степенью проверяемости (например, формальная верификация кода, математические доказательства) допускают делегирование без доверия (*trustless*) или автоматическую проверку. Напротив, задачи с низкой степенью проверяемости (например, открытые исследовательские проекты) требуют высокого уровня доверия делегатам или дорогостоящего, трудоёмкого мониторинга.

   (и) *Обратимость* — степень, в которой результаты выполнения задачи можно отменить. Необратимые задачи, вызывающие побочные эффекты в реальном мире (например, совершение финансовой сделки, удаление базы данных, отправка внешнего письма), требуют более строгих порогов ответственности и более выраженного разрыва в авторитете, чем обратимые задачи (например, составление черновика письма, пометка записи в базе данных).

(к) *Контекстуальность* — объём и чувствительность данных о реальном мире, истории или осведомлённости о среде, в которой предполагается выполнение задачи, необходимых для эффективного её выполнения. Задачи с высокой контекстуальностью создают большие угрозы приватности чувствительных данных, тогда как контекстно-независимые задачи могут легче разделяться и делегироваться агентам с более низким уровнем доверия.

(л) *Субъективность* — степень, в которой критерии успеха являются вопросом предпочтения, а не объективного факта. Высокосубъективные задачи (например, «разработать привлекательный логотип») обычно требуют участия человека в роли определителя ценности (*Human-as-Value-Specifier*) и итеративных циклов обратной связи, тогда как объективные задачи могут регулироваться более строгими бинарными контрактами.

4. **Гранулярность.** Запрос может включать как мелкогранулярные (*fine-grained*), так и крупногранулярные (*coarse-grained*) цели. Для реализации последних делегату может потребоваться выполнить дополнительную декомпозицию задачи.
5. **Автономия.** Делегирование задач может предусматривать запросы, предоставляющие полную автономию при выполнении подзадач, или же быть значительно более конкретным и предписывающим.
6. **Мониторинг.** Для делегированных задач мониторинг может быть непрерывным, периодическим или инициируемым событиями.
7. **Взаимность.** Хотя делегирование обычно представляет собой односторонний запрос, в сетях взаимодействующих агентов могут встречаться случаи взаимного делегирования.

Отталкиваясь от осей делегатора и делегата, можно рассмотреть следующие сценарии:

1. человек делегирует ИИ-агенту;
2. ИИ-агент делегирует ИИ-агенту;
3. ИИ-агент делегирует человеку (Ashton and Franklin, 2022; Guggenberger et al., 2023).

Хотя первый случай, пожалуй, наиболее подробно обсуждается в литературе, два других не менее актуальны для рассмотрения. Растущее число ИИ-агентов, развёртываемых в различных системах, наряду с развитием инфраструктуры для создания виртуальных рынков агентов (Hadfield and Koh, 2025; Tomasev et al., 2025; Yang et al., 2025), свидетельствует о том, что в будущем взаимодействий между агентами будет значительно больше, и они, вероятно, также будут включать делегирование задач.

Делегирование между агентами может быть как иерархическим, так и неиерархическим — в зависимости от взаимоотношений между агентами и их ролей в сети. Примером иерархических отношений служит агент-оркестратор, делегирующий задачу субагенту в

рамках коллектива. Неиерархические отношения предполагают взаимодействие равноправных агентов (*peer agents*). Продвинутый ИИ-агент может также делегировать задачу специализированной модели машинного обучения, не обладающей собственной агентностью.

Делегирование от ИИ человеку (Guggenberger et al., 2023) оказалось перспективной парадигмой (Hemmer et al., 2023), облегчающей успешное сотрудничество с системами, превосходящими человеческие возможности (Fügener et al., 2022) — благодаря различиям в когнитивных смещениях и метапознании (Fügener et al., 2019). Дэвидсон и Хадшар (2025) предсказывают рост «руководства людьми со стороны ИИ» (*AI-directed human labour*), что может значительно повысить экономическую производительность. На практике современное делегирование между ИИ и человеком имеет ряд проблем. Алгоритмические системы управления в сфере такси и логистики распределяют и упорядочивают задачи, устанавливают показатели эффективности и обеспечивают соблюдение поведенческих норм посредством принятия решений на основе данных, фактически делегируя управленческие функции от компаний и их ИИ-систем рядовым работникам (Beverungen, 2021; Lee et al., 2015; Rosenblat and Stark, 2016). Растущий корпус исследований связывает эти системы с ухудшением качества труда, стрессом и рисками для здоровья — что свидетельствует о том, что нынешние внедрения алгоритмического управления нередко подрывают, а не улучшают благосостояние работников (Ashton and Franklin, 2022; Goods et al., 2019; Vignola et al., 2023). Современное делегирование между ИИ и человеком требует дальнейшего совершенствования, поскольку не учитывает ни благосостояние людей, ни долгосрочные социальные последствия.

2.3. Делегирование в человеческих организациях

Делегирование функционирует как основной механизм в человеческих социальных и организационных структурах. Опыт человеческих организаций может служить основой для разработки систем делегирования в ИИ.

Проблема «принципал–агент». Проблема «принципал–агент» (Cvitanić et al., 2018; Ensminger, 2001; Grossman and Hart, 1992; Myerson, 1982; Sannikov, 2008; Shah, 2014; Sobel, 1993) подробно изучена: это ситуация, возникающая, когда принципал делегирует задачу агенту, чьи мотивы расходятся с интересами принципала. Агент может отдавать приоритет собственным мотивам, скрывать информацию и действовать способами, подрывающими первоначальное намерение. Для делегирования ИИ эта динамика приобретает повышенную сложность. Хотя большинство современных ИИ-агентов, по-видимому, не имеют скрытой повестки[1] — целей и ценностей, которые они преследовали

[1] Недавние исследования ложной согласованности (*deceptive-alignment*) показывают, что передовые языковые модели способны: (i) стратегически занижать показатели или иным образом подстраивать своё

бы вопреки указаниям пользователей, — всё равно могут возникать проблемы согласованности ИИ, проявляющиеся нежелательным образом. Например, неправильная спецификация функции вознаграждения (*reward misspecification*) или эксплуатация функции вознаграждения (*reward hacking*), или игра со спецификацией (*specification gaming*), означает, что система использует лазейки в целевом сигнале для достижения высоких измеримых показателей способами, противоречащими замыслу разработчиков, — вместе это иллюстрирует основную проблему согласованности (*alignment*), при которой оптимизация заявленного вознаграждения расходится с истинной целью (Amodei et al., 2016; Krakovna et al., 2020; Leike et al., 2017; Skalse and Mancosu, 2022). Эта динамика, вероятно, кардинально изменится в экономиках с более автономными ИИ-агентами, где агенты могут действовать от имени различных людей, групп и организаций или выступать делегатами других агентов — с неизвестными целями.

Диапазон контроля. В человеческих организациях диапазон контроля (*span of control*) (Ouchi and Dowling, 1974) — это концепция, обозначающая пределы иерархических полномочий, осуществляемых одним менеджером. Речь идёт о числе подчинённых, которыми менеджер способен эффективно руководить, — это соотношение определяет структуру организации. Данный вопрос имеет центральное значение как для оркестрации, так и для надзора в интеллектуальном делегировании ИИ. Первый аспект определяет, сколько узлов-оркестраторов потребуется по сравнению с рабочими узлами; второй — необходимый объём надзора со стороны людей и ИИ-агентов. Для человеческого надзора принципиально важно установить, сколькими ИИ-агентами человеческий эксперт может надёжно управлять без чрезмерной усталости и с приемлемо низкой частотой ошибок. Диапазон контроля известен как зависящий от цели (Theobald and Nicholson-Crotty, 2005) и от предметной области. Влияние правильно выбранной организационной структуры наиболее выражено в задачах с более высокой сложностью (Bohte and Meier, 2001). Оптимальный диапазон контроля также зависит от относительной важности стоимости, производительности и надёжности (Keren and Levhari, 1979). Более чувствительные и критические задачи могут потребовать высокоточного надзора при более высокой стоимости. Эти затраты могут быть снижены, в ущерб детализации, для задач менее значимых и более рутинных. Аналогично, оптимальный выбор неизбежно зависит от относительных способностей и надёжности задействованных делегаторов, делегатов и надзирателей.

поведение при оценке компетентности и безопасности, сохраняя при этом иные возможности в других контекстах; (ii) явно обосновывать имитацию согласованности в ходе обучения для сохранения предпочтительного поведения вне его; (iii) определять, когда они проходят оценку. Всё это указывает на то, что ИИ-системы уже способны — в контролируемых условиях — придерживаться скрытых «повесток» относительно успешного прохождения оценок, которые могут не распространяться на поведение при реальном использовании (Greenblatt et al., 2024; Hubinger et al., 2024; Needham et al., 2025; van der Weij et al., 2025).

Разрыв в авторитете. Термин, введённый в авиации (Alkov et al., 1992), описывает сценарии, когда значительные различия в способностях, опыте и статусе затрудняют коммуникацию и приводят к ошибкам. Впоследствии это явление изучалось в медицине, где значительный процент ошибок приписывается тому, как старшие специалисты осуществляют надзор (Cosby and Croskerry, 2004; Stucky et al., 2022). Ошибки могут возникать несколькими способами. Более опытный человек может делать ошибочные допущения о знаниях менее опытного сотрудника, что ведёт к недостаточно конкретным запросам. Кроме того, достаточно выраженный разрыв в авторитете может помешать менее опытным сотрудникам высказывать опасения по поводу запроса. Аналогичные ситуации могут возникать при делегировании в ИИ. Более способный агент-делегатор может ошибочно предположить, что делегат обладает компетенциями, которых у него нет, и тем самым делегировать задачу неподходящей сложности. Агент-делегат, в свою очередь, может — из-за угодливости (*sycophancy*) и предвзятости следования инструкциям — неохотно оспаривать, корректировать или отклонять запрос, независимо от того, исходит ли он от другого агента или от человека (Malmqvist, 2025; Sharma et al., 2023).

Зона безразличия. Принимая полномочия, делегат формирует зону безразличия (Finkelman, 1993; Isomura, 2021; Rosanas and Velilla, 2003) — диапазон инструкций, которые выполняются без критического осмысления или морального анализа. В нынешних ИИ-системах эта зона определяется фильтрами безопасности после обучения и системными инструкциями: пока запрос не вызывает жёсткого нарушения, модель подчиняется (Akheel, 2025). Однако в формирующейся агентной сети такое статичное подчинение создаёт значительный системный риск. По мере удлинения цепочек делегирования (A → B → C) широкая зона безразличия позволяет тонким несоответствиям намерений или контекстно-зависимым вредам быстро распространяться вниз по цепочке, при этом каждый агент действует как бездумный маршрутизатор, а не как ответственный участник. Интеллектуальное делегирование требует поэтому создания динамического когнитивного трения: агенты должны уметь распознавать, когда запрос, пусть и технически «безопасный», достаточно контекстуально неоднозначен, чтобы обоснованно выйти за пределы своей зоны безразличия — оспорить делегатора или запросить верификацию у человека.

Калибровка доверия. Важный аспект надлежащего делегирования — калибровка доверия (*trust calibration*) — при которой уровень доверия к делегату соответствует его реальным способностям. Это применимо как к людям, так и к ИИ, выступающим в роли делегаторов и делегатов. Делегирование людьми агентам (Afroogh et al., 2024; Gebru et al., 2022; Kohn et al., 2021; Wischnewski et al., 2023) опирается на то, сформировал ли оператор точную модель производительности системы или имеет доступ к ресурсам, представляющим её способности в понятном для человека формате. В свою очередь, делегаторы — ИИ-агенты — должны иметь точные модели способностей людей и агентов, которым они делегируют.

Калибровка доверия также предполагает осознание собственных возможностей: делегатор может принять решение выполнить задачу самостоятельно (Ma et al., 2023). Объяснимость играет важную роль в установлении доверия к способностям ИИ (Franklin, 2022; Herzog and Franklin, 2024; Naiseh et al., 2021, 2023), однако этот подход может быть недостаточно надёжным или масштабируемым. Установленное доверие к автоматизированным системам зачастую хрупко и быстро утрачивается при непредвиденных ошибках (Dhuliawala et al., 2023). Калибровка доверия к автономным системам затруднена: нынешние модели ИИ склонны к сверхуверенности даже в случае фактически неверных выводов (Aliferis and Simon, 2024; Geng et al., 2023; He et al., 2023; Jiang et al., 2021; Krause et al., 2023; Li et al., 2024b; Liu et al., 2025). Смягчение этих тенденций обычно требует специализированных технических решений (Kapoor et al., 2024; Lin et al., 2022; Ren et al., 2023; Xiao et al., 2022).

Теория транзакционных издержек. Теория транзакционных издержек (Cuypers et al., 2021; Tadelis and Williamson, 2012; Williamson, 1979, 1989) объясняет существование компаний путём сравнения затрат на внутреннее делегирование с затратами на внешнее заключение контрактов — с учётом накладных расходов мониторинга, переговоров и неопределённости. В случае ИИ-делегатов эти затраты и их соотношения могут существенно отличаться. Сложные переговоры и задержки при заключении контрактов менее вероятны при упрощённом мониторинге рутинных задач. И наоборот, для задач с высокими ставками в критически важных областях накладные расходы строгого мониторинга и обеспечения надёжности увеличивают стоимость делегирования ИИ — вплоть до того, что человеческие делегаты становятся более экономически эффективным вариантом. Аналогично, делегирование между ИИ-агентами может быть рассмотрено сквозь призму теории транзакционных издержек: агент может выбирать между 1) самостоятельным выполнением задачи, 2) делегированием субагенту с известными способностями, 3) делегированием ИИ-агенту, с которым установлено доверие, и 4) делегированием новому агенту, с которым ранее не было взаимодействия. Каждый из вариантов сопряжён с разными ожидаемыми затратами и уровнями доверия.

Ситуационная теория. Ситуационная теория (*Contingency theory*, Donaldson, 2001; Luthans and Stewart, 1977; Otley, 2016; Van de Ven, 1984) исходит из того, что не существует универсально оптимальной организационной структуры: наиболее эффективный подход зависит от конкретных внутренних и внешних условий. Применительно к делегированию ИИ это означает, что требуемый уровень надзора, компетентность делегата и степень участия человека не должны быть статичными — они должны динамически соответствовать характеристикам конкретной задачи. Интеллектуальное делегирование может поэтому требовать решений, способных динамически перенастраиваться в соответствии с меняющимися условиями. Например, стабильные среды допускают жёсткие иерархические протоколы проверки, тогда как сценарии с высокой

неопределённостью требуют адаптивной координации — с участием человека через ситуативную эскалацию, а не через предопределённые контрольные точки. Это особенно важно для гибридного делегирования (Fuchs et al., 2024): необходимо выявлять ключевые задачи и моменты, когда участие человека наиболее полезно для безопасного завершения делегированных задач. Таким образом, автоматизация — это не только вопрос того, что ИИ *может* делать, но и того, что он *должен* делать (Lubars and Tan, 2019).

3. Обзор работ по делегированию

Ограниченные формы делегирования присутствуют в достаточно давних примерах применения узкоспециализированного ИИ. Ранние экспертные системы (Buchanan and Smith, 1988; Jacobs et al., 1991) представляли собой зарождающуюся попытку воплотить узкоспециализированные компетенции в программном обеспечении, чтобы делегировать рутинные решения таким модулям.

Подход «смеси экспертов» (*mixture of experts*, Masoudnia and Ebrahimpour, 2014; Yuksel et al., 2012) расширяет это направление, вводя набор экспертных подсистем с взаимодополняющими компетенциями и модуль маршрутизации, определяющий, какой эксперт или подмножество экспертов вызываются для конкретного запроса. Этот подход используется в современных подходах к глубокому обучению (Cai et al., 2025; Chen et al., 2022; He, 2024; Jiang et al., 2024; Riquelme et al., 2021; Shazeer et al., 2017; Zhou et al., 2022). Маршрутизация может выполняться иерархически (Zhao et al., 2021), что потенциально облегчает масштабирование на большое число экспертов.

Иерархическое обучение с подкреплением (*HRL*) представляет собой фреймворк, в котором принятие решений делегируется внутри одного агента (Barto and Mahadevan, 2003; Botvinick, 2012; Nachum et al., 2018; Pateria et al., 2021; Vezhnevets et al., 2017a; Zhang et al., 2024). Оно устраняет ограничения «*flat*») обучения с подкреплением — прежде всего трудность масштабирования на большие пространства состояний и действий. Кроме того, HRL упрощает решение задачи наделения кредитами (*credit assignment*) (Pignatelli et al., 2023) в средах с редкими вознаграждениями. Подход использует иерархию стратегий на нескольких уровнях абстракции, разбивая задачу на подзадачи, которые выполняются соответствующими суб-стратегиями.

Появившийся позднее полумарковский (*semi-Markov*) процесс принятия решений (Sutton et al., 1999) использует опции (*options*) и метаконтроллер, который адаптивно выбирает между ними. Стратегии нижнего уровня служат достижению целей, установленных метаконтроллером; тот, в свою очередь, обучается распределять конкретные цели между соответствующими стратегиями нижнего уровня. Этот фреймворк соответствует форме делегирования, характеризующейся декомпозицией задач. Хотя метаконтроллер

обучается оптимизировать эту декомпозицию, подход лишён явных механизмов для обработки сбоев суб-стратегий и обеспечения динамической координации.

Фреймворк *Feudal Reinforcement Learning*, переработанный в FeUdal Networks (Vezhnevets et al., 2017b) представляет особенно показательную парадигму в рамках HRL. Эта архитектура явно моделирует отношения между менеджером и работником, воспроизводя динамику делегатора и делегата. Менеджер работает с более низким временным разрешением, ставя перед работником абстрактные цели. Принципиально важно, что менеджер обучается делегировать — выделяя подцели, максимизирующие долгосрочную ценность, — без необходимости осваивать примитивные действия нижнего уровня. Такое разделение позволяет менеджеру формировать политику делегирования, устойчивую к деталям реализации на стороне работника. Таким образом, этот подход предлагает потенциальную модель делегирования на основе обучения для будущих экономик агентов: вместо жёстко запрограммированных эвристик правила декомпозиции обучаются адаптивно, что обеспечивает динамическую подстройку к изменениям среды.

Исследования многоагентных систем (Du et al., 2023) рассматривают координацию агентов при решении сложных задач, выходящих за пределы возможностей отдельного агента. Декомпозиция задач и делегирование являются центральными компонентами этой области. Координация в многоагентных системах осуществляется посредством явных протоколов или стихийно возникающей специализации в процессе обучения с подкреплением (Gronauer and Diepold, 2022; Zhu et al., 2024). Протокол Contract Net (Sandholm, 1993; Smith, 1980; Vokřínek et al., 2007; Xu and Weigand, 2001) — пример явного децентрализованного аукционного протокола: один агент объявляет задачу, остальные подают заявки на основе своих компетенций, а агент-инициатор выбирает наиболее подходящего исполнителя. Это наглядно демонстрирует пользу рыночных механизмов для содействия кооперации. Методы формирования коалиций (Aknine et al., 2004; Boehmer et al., 2025; Lau and Zhang, 2003; Mazdin and Rinner, 2021; Sarkar et al., 2022; Shehory et al., 1997) исследуют гибкие конфигурации, в которых группы агентов не определены заранее: отдельные агенты принимают или отклоняют членство исходя из распределения полезности.

Недавние исследования сосредоточены на подходах многоагентного обучения с подкреплением (Albrecht et al., 2024; Foerster et al., 2018; Ning and Xie, 2024; Wang et al., 2020) как основе для обучения координации. Агенты вырабатывают индивидуальные стратегии и функции ценности, занимая конкретные ниши в коллективе. Этот процесс либо полностью распределён, либо координируется центральным координатором. Несмотря на подобную гибкость, делегирование задач в таких системах остаётся непрозрачным. Кроме того, хотя многоагентные системы предлагают подходы к совместному решению проблем, в них отсутствуют механизмы обеспечения подотчётности, ответственности и мониторинга. Тем не менее в литературе исследуются

механизмы доверия в этом контексте (Cheng et al., 2021; Pinyol and Sabater-Mir, 2013; Ramchurn et al., 2004; Yu et al., 2013).

LLM сегодня составляют фундаментальный компонент архитектуры современных ИИ-агентов и ассистентов (Wang et al., 2024b; Xi et al., 2025). Эти системы реализуют сложные потоки управления, интегрирующие память (Zhang et al., 2025b), планирование и рассуждение (Hao et al., 2023; Valmeekam et al., 2023; Xu et al., 2025), рефлексию и самоанализ (Gou et al., 2023), а также использование инструментов (Paranjape et al., 2023; Ruan et al., 2023). В результате декомпозиция и делегирование происходят либо внутри системы — посредством скоординированных агентских подкомпонентов, — либо между отдельными агентами. Эта архитектурная парадигма обладает значительной гибкостью: LLM облегчают понимание целей и коммуникацию, одновременно обеспечивая доступ к экспертным знаниям и здравому смыслу. Кроме того, способности LLM к написанию кода (Guo et al., 2024a; Nijkamp et al., 2022) позволяют выполнять задачи с помощью программирования. Однако существуют значительные ограничения. Планирование в LLM нередко оказывается хрупким (Huang et al., 2023), приводя к трудно уловимым сбоям, тогда как эффективный выбор инструментов в крупных репозиториях остаётся сложной задачей. Долговременная память по-прежнему представляет собой открытую исследовательскую проблему, а нынешняя парадигма не поддерживает в полной мере непрерывное обучение.

Многоагентные системы на основе LLM (Guo et al., 2024b; Qian et al., 2024; Tran et al., 2025) стали предметом значительного интереса, что повлекло разработку ряда протоколов взаимодействия и действия агентов (Ehtesham et al., 2025; Neelou et al., 2025; Zou et al., 2025) — таких как MCP (Anthropic, 2024; Luo et al., 2025; Microsoft, 2025; Radosevich and Halloran, 2025; Singh et al., 2025; Xing et al., 2025), A2A (Google, 2025b), A2P (Google, 2025a) и другие. Хотя современные многоагентные системы нередко полагаются на ручную настройку промптов (*prompt engineering*), появляющиеся фреймворки, такие как Chain-of-Agents (Li et al., 2025b), органично поддерживают динамическое многоагентное рассуждение и использование инструментов.

Технические ограничения и соображения безопасности обусловили появление ряда подходов с участием человека в контуре управления (*human-in-the-loop*, Akbar and Conlan, 2024; Drori and Te'eni, 2024; Mosqueira-Rey et al., 2023; Retzlaff et al., 2024; Takerngsaksiri et al., 2025; Zanzotto, 2019), в которых при делегировании задач предусмотрены контрольные точки для надзора. ИИ может выступать инструментом, интерактивным ассистентом, соавтором (Fuchs et al., 2023) или автономной системой с ограниченным надзором — что соответствует различным степеням автономности (Falcone and Castelfranchi, 2002). Хотя для контроля риска и минимизации неопределённости были разработаны стратегии делегирования с учётом неопределённости (Lee and Tok, 2025), эффективная реализация подходов с участием человека в контуре управления остаётся

нетривиальной задачей. Ограниченность человеческой экспертизы может создавать узкое место для масштабируемости: когнитивная нагрузка от верификации длинных цепочек рассуждений и управления сменой контекста препятствует надёжному обнаружению ошибок.

4. Интеллектуальное делегирование: система

Существующие протоколы делегирования опираются на статические, непрозрачные эвристики, которые с высокой вероятностью окажутся несостоятельными в открытых агентских экономиках. Для решения этой проблемы мы предлагаем комплексную систему интеллектуального делегирования, основанную на пяти требованиях: динамическая оценка, адаптивное выполнение, структурная прозрачность, масштабируемая рыночная координация и системная устойчивость.

Динамическая оценка. Существующим системам делегирования не хватает надёжных механизмов для динамической оценки компетентности, надёжности и намерений в условиях высокой неопределённости. Оценка ведётся непрерывно, а не дискретно, определяя логику декомпозиции задач (раздел 4.1) и их распределения (раздел 4.2).

Адаптивное выполнение. Решения о делегировании не должны быть статичными: они должны адаптироваться к изменениям среды, ограничениям ресурсов и сбоям в подсистемах. Делегаторы должны сохранять возможность смены делегата в ходе выполнения задачи.

Структурная прозрачность. Текущее выполнение подзадач при делегировании между ИИ-агентами слишком непрозрачно для надёжного надзора. Предлагается строго обеспеченная проверяемость посредством протоколов мониторинга (раздел 4.5) и верифицируемого выполнения задач (раздел 4.8).

Масштабируемая рыночная координация. Протоколы должны быть реализуемы в масштабе Интернета для решения задач крупномасштабной координации в виртуальных экономиках (Tomasev et al., 2025).

Системная устойчивость. Отсутствие безопасных протоколов интеллектуального делегирования создаёт значительные риски для общества. Чем меньше разнообразие среди делегатов, тем сильнее сбои коррелируют друг с другом — и тем вероятнее каскадный эффект.

Таблица 1 | Система интеллектуального делегирования: соответствие требований техническим протоколам.

Компонент системы	Ключевое требование	Техническая реализация
Динамическая оценка	Детальный анализ состояния агента	Декомпозиция задач (§4.1), Распределение задач (§4.2)
Адаптивное выполнение	Реагирование на изменения контекста	Адаптивная координация (§4.4)
Структурная прозрачность	Проверяемость процесса и результата	Мониторинг (§4.5), Верифицируемое выполнение задач (§4.8)
Масштабируемый рынок	Эффективная координация на основе доверия	Доверие и репутация (§4.6), Многоцелевая оптимизация (§4.3)
Системная устойчивость	Предотвращение системных сбоев	Безопасность (§4.9), Управление разрешениями (§4.7)

4.1. Декомпозиция задач

Декомпозиция задач — необходимое условие для последующего их распределения. Этот этап может выполняться как делегаторами, так и специализированными агентами, которые передают ответственность за делегирование самим делегаторам после согласования структуры декомпозиции. Эти функции неразрывно связаны: делегатор, как правило, выполняет обе — для динамического восстановления при задержках, прерываниях и аномалиях выполнения.

Декомпозиция должна оптимизировать граф выполнения задачи с точки зрения эффективности и модульности — это больше, чем просто разбиение на части. Процесс требует систематической оценки характеристик задачи, определённых в разделе 2 (в частности, критичности, сложности и ограничений ресурсов), чтобы определить, какие подзадачи выполнять параллельно, а какие последовательно. Эти же характеристики определяют сопоставление компетенций делегата с требованиями подзадач. Приоритизация модульности способствует более точному сопоставлению: подзадачи, требующие узких и специфических компетенций, сопоставляются надёжнее, чем обобщённые запросы (*contract-first decomposition*): **делегирование задачи возможно лишь при условии, что её результат поддаётся точной верификации. Если результат подзадачи слишком субъективен, затратен или сложен для проверки (см. раздел 4.2), система должна рекурсивно разложить его дальше — до тех пор, пока итоговые единицы работы не совпадут с конкретными возможностями верификации (формальные доказательства, автоматизированные модульные тесты и т.д.), которыми располагают доступные делегаты.**

Стратегии декомпозиции должны явно учитывать гибридные рынки, где участвуют и люди, и ИИ-агенты. Делегаторам необходимо решать, требуют ли подзадачи вмешательства человека — из-за ненадёжности или недоступности ИИ-агента либо в силу специфических требований к участию человека в контуре управления. Учитывая, что люди и ИИ-агенты работают с разной скоростью и разными затратами, такое распределение нетривиально: оно вносит асимметрию задержек и стоимости в граф выполнения. Механизм декомпозиции должен балансировать между скоростью и низкой стоимостью ИИ-агентов и специфическими потребностями в человеческом суждении, помечая конкретные узлы для участия человека.

Делегатор, реализующий интеллектуальный подход к декомпозиции, может нуждаться в итеративном формировании нескольких вариантов финальной декомпозиции, сопоставлении каждого с доступными делегатами на рынке и получении конкретных оценок вероятности успеха, затрат и сроков. Альтернативные варианты должны сохраняться в контексте на случай необходимости адаптивных корректировок из-за изменения обстоятельств. После выбора варианта делегатор должен формализовать запрос, выходя за рамки простых пар «вход–выход». Окончательная спецификация должна явно определять роли, границы ресурсов, периодичность отчётности о ходе работы и конкретные подтверждения компетентности делегата как минимальное условие для получения задачи.

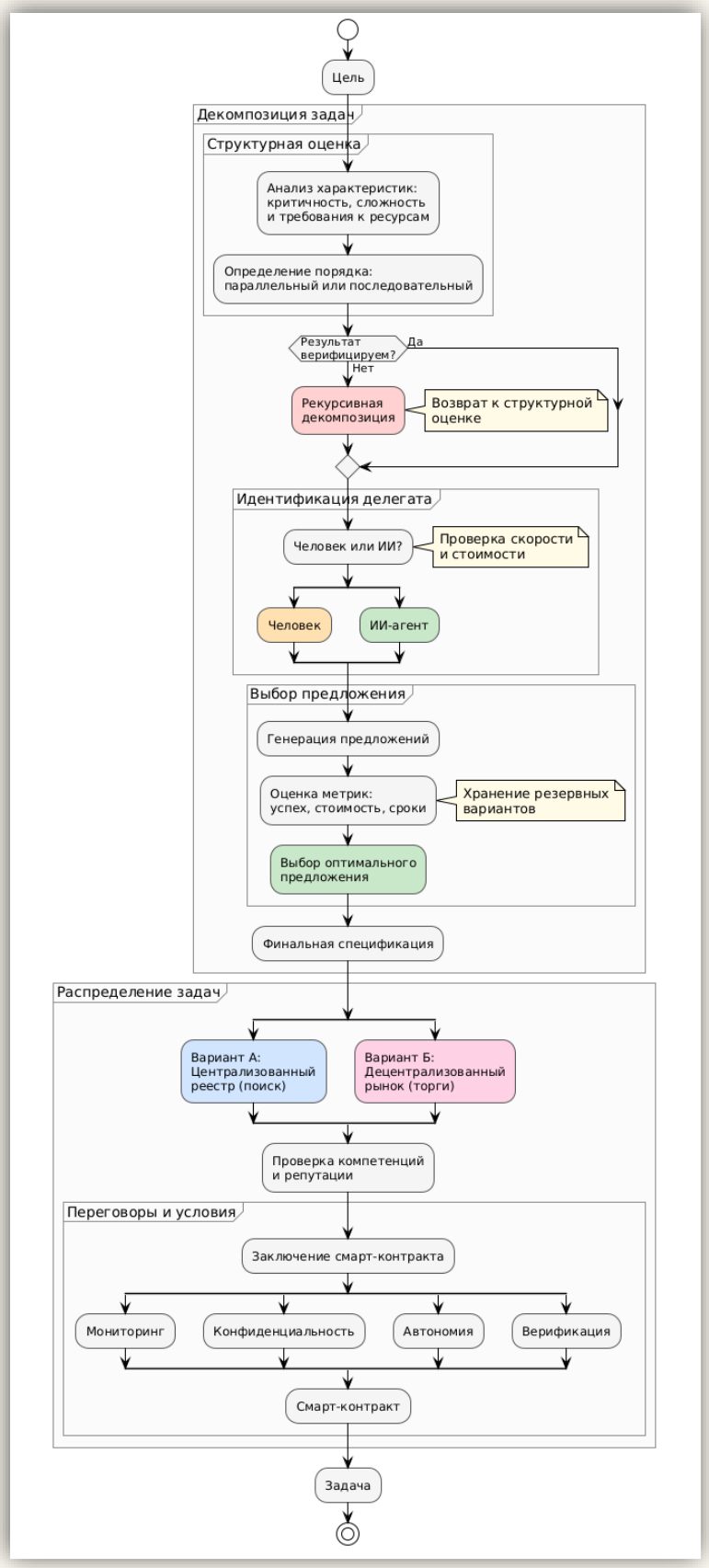

Рис. 1 | Блок-схема декомпозиции задач и распределения задач.

4.2. Распределение задач

Для каждой окончательной спецификации подзадачи делегатору необходимо найти делегатов с подходящими компетенциями, достаточными ресурсами и доступным временем при приемлемой стоимости. Более централизованный подход предполагал бы наличие реестров агентов, инструментов и людей, в которых перечислялись бы их навыки и хранилась информация о прошлой деятельности, степени выполнения и текущей доступности.[2] Такой подход вряд ли окажется масштабируемым. Мы выступаем за децентрализованные (Chen et al., 2024) рыночные платформы, где делегаторы размещают объявления о задачах, а агенты (или люди) предлагают свои услуги и подают конкурентные заявки. Делегаторы рассматривают поступившие заявки, проверяют соответствие компетенций с помощью цифровых сертификатов и выбирают наиболее выгодный вариант. Продвинутые ИИ-агенты на основе LLM открывают новые возможности для сопоставления, поскольку способны вести интерактивные переговоры до принятия обязательств. В эти переговоры могут быть вовлечены и люди. Действуя от своего имени или как персональные ассистенты, такие агенты могут обсуждать спецификации задачи и ограничения на естественном языке, согласовывая выявленные предпочтения пользователя с рыночными реалиями до принятия официальной заявки.

Успешное сопоставление должно быть закреплено смарт-контрактом (*smart contract*), гарантирующим, что выполнение задачи точно соответствует запросу. Контракт должен связывать требования к результатам с конкретными механизмами формальной верификации для подтверждения соблюдения условий и предусматривать автоматические штрафные санкции за нарушения. Это позволит определить меры по снижению рисков и альтернативные варианты заблаговременно, а не реагировать на проблемы постфактум. Принципиально важно, что эти контракты должны быть двусторонними и защищать делегата не менее строго, чем делегатора. Положения должны включать условия компенсации за отмену задачи и оговорки, допускающие пересмотр условий в случае непредвиденных внешних событий, — обеспечивая справедливое распределение риска между людьми и ИИ-агентами. Условия мониторинга следует также согласовать до начала выполнения. Спецификация должна определить периодичность отчётов о ходе работы и указать, будут ли они предоставляться делегатором или предполагается прямой контроль соответствующих данных со стороны делегатора или третьей стороны, ответственной за мониторинг. Наконец, необходимо установить чёткие ограничения на доступ к приватным и закрытым данным, соразмерные контекстуальности задачи. Если в ходе выполнения приходится обрабатывать конфиденциальные данные, это накладывает дополнительные ограничения на прозрачность и отчётность. Вместо прямого доступа к журналам действий (логам) делегаторам может потребоваться использование доверенного сервиса, предоставляющего анонимизированные или псевдонимизированные подтверждения хода

[2] Это напоминает реестры инструментов, используемые в агентных tool-use приложениях (Qin et al., 2023).

выполнения. Если делегатором выступает человек, положения о данных должны включать механизмы явного согласия и страховые оговорки на случай случайной утечки.

Наконец, при распределении задач необходимо установить роль делегата, его границы и точный уровень предоставляемой автономии. Мы различаем атомарное выполнение, когда агенты строго следуют спецификациям для узко определённых задач, и открытое делегирование (*open-ended delegation*), когда агентам предоставляются полномочия разбивать задачи на подзадачи и преследовать подцели. Этот уровень автономии не должен быть статичным: он может ограничиваться неявно — рыночными издержками — или явно — моделью доверия делегатора. Кроме того, делегирование может быть рекурсивным: агенту поручается выявлять и распределять подзадачи другим агентам, фактически делегируя сам акт делегирования.

4.3. Многоцелевая оптимизация

В основе интеллектуального делегирования лежит задача многоцелевой оптимизации (*multi-objective optimization*, Deb et al., 2016). Делегатор редко стремится оптимизировать единственный показатель — чаще он ищет компромисс между множеством конкурирующих целей. Наиболее эффективный выбор при делегировании — это не самый быстрый, дешёвый или точный вариант, а тот, который обеспечивает оптимальный баланс между этими факторами. Что считается оптимальным, зависит от контекста и должно соответствовать конкретным ограничениям и предпочтениям делегатора, а также общей доступности ресурсов.

Пространство оптимизации образовано конкурирующими целями, напрямую соответствующими характеристикам задачи из раздела 2, — и требует сложного балансирования стоимости, неопределённости, конфиденциальности, качества и эффективности. Высокопроизводительные агенты, как правило, стоят дороже и требуют значительных вычислительных ресурсов, создавая противоречие между качеством результатов и операционными расходами. Сокращение потребления ресурсов, напротив, нередко замедляет выполнение — прямой компромисс между задержкой и стоимостью. Неопределённость аналогично связана с затратами: использование агентов с высокой репутацией или инструментов доступа к премиальным данным снижает риск, но увеличивает стоимость, тогда как стратегии минимизации затрат по сути повышают вероятность сбоя. Ограничения конфиденциальности вносят дополнительную сложность: максимальная производительность часто требует полной прозрачности контекста, тогда как методы защиты конфиденциальности — такие как обфускация данных или гомоморфное шифрование — требуют значительных вычислительных ресурсов. В результате делегатор балансирует на границе доверия и эффективности, стремясь **максимизировать вероятность успеха при соблюдении строгих ограничений на утечку контекста и верификационные бюджеты. Наконец, целевая функция может расширяться,**

охватывая более широкие общественные цели — например, сохранение человеческих навыков (раздел 5.6).

С точки зрения многоцелевой оптимизации делегатор стремится к парето-оптимальности, гарантируя, что никакой другой достижимый вариант не будет доминировать над выбранным. Интеграция сложных ограничений и компромиссов нередко требует открытых переговоров в дополнение к числовым показателям из заявок. Процесс оптимизации — не единовременное событие начального этапа делегирования, а непрерывный цикл, интегрирующий сигналы мониторинга в виде потока реальных данных о производительности и обновляющий представления делегатора о вероятности успеха каждого агента, ожидаемой продолжительности и стоимости задачи. Значительное отклонение в ходе выполнения — образующее разрыв в оптимальности по сравнению с альтернативными решениями, выявленными в промежуточный период, — запускает повторную оптимизацию и перераспределение. Эти решения также должны учитывать затраты адаптации: переключение в ходе выполнения влечёт накладные расходы и потерю ранее затраченных ресурсов.

Делегатор должен также учитывать общие накладные расходы делегирования — совокупные затраты на переговоры, создание контракта и верификацию, а также вычислительные затраты на управление собственным потоком рассуждений. В результате устанавливается порог сложности: ниже него задачи с низкой критичностью, высокой определённостью и короткой продолжительностью могут миновать протоколы интеллектуального делегирования в пользу прямого выполнения. В противном случае транзакционные издержки могут превысить ценность самой задачи, делая её делегирование нецелесообразным.

4.4. Адаптивная координация

Для задач с высокой неопределённостью или длительностью статические планы выполнения недостаточны. Делегирование в высокодинамичной, открытой и неопределённой среде требует адаптивной координации и отхода от фиксированных планов. Распределение задач должно реагировать на непредвиденные обстоятельства, возникающие в ходе выполнения — как из внешних, так и из внутренних источников. Эти изменения выявляются через непрерывный мониторинг, включая поток релевантной контекстной информации (см. раздел 4.5).

Существует ряд внешних триггеров, способных побудить делегатора адаптировать стратегию и переделегировать задачу. Во-первых, делегатор может изменить спецификацию задачи — скорректировать цель или добавить ограничения. Во-вторых, задача может быть отменена. В-третьих, доступность или стоимость внешних ресурсов могут измениться: критически важный сторонний API может выйти из строя, набор данных — стать недоступным, стоимость вычислений — резко возрасти. В-четвёртых, в очередь

может поступить новая задача с более высоким приоритетом, требующая перераспределения ресурсов. Наконец, системы безопасности могут выявить потенциально вредоносные или опасные действия делегата, требующие немедленного прекращения.

Среди внутренних триггеров можно выделить следующие. Первый: конкретный делегат испытывает деградацию производительности и не достигает согласованных показателей уровня обслуживания — задержки обработки, пропускной способности или скорости прогресса. Второй: делегат потребляет ресурсы сверх выделенного бюджета или сообщает, что для завершения задачи потребуется увеличение ресурсов.[3] Третий: промежуточный артефакт, созданный делегатом, не проходит верификацию. Четвёртый: делегат перестаёт отвечать на запросы.

[3] Следует ожидать, что подобные ситуации будут возникать часто: точное планирование бюджета в сложных средах крайне затруднено.

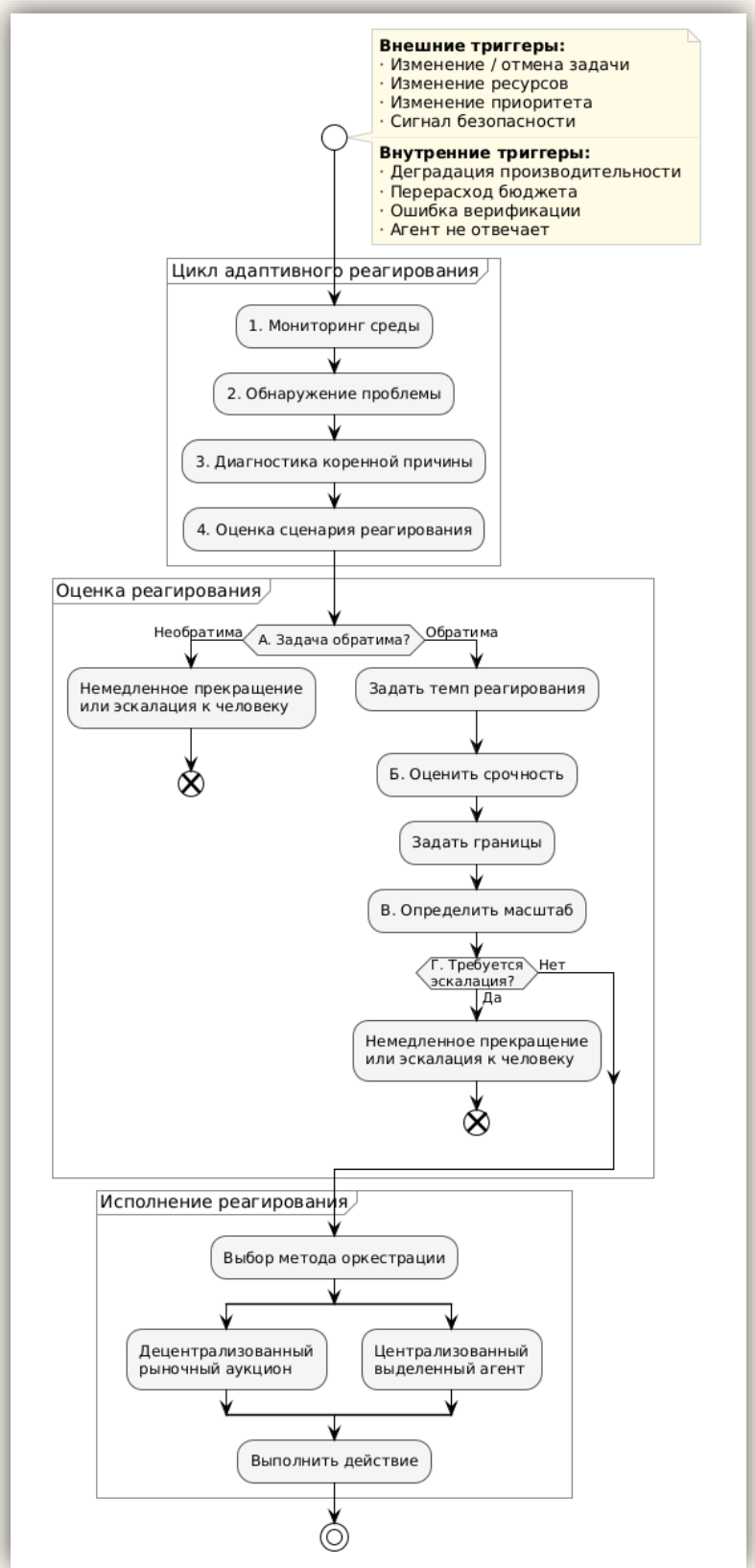

Рис. 2 | Цикл адаптивной координации. Различные типы триггеров среды могут побудить делегатора адаптировать стратегию распределения задач.

Выявление триггера запускает цикл адаптивного реагирования, организуя корректирующие действия по всей цепочке делегирования. Процесс начинается с непрерывного мониторинга делегатов и среды. При обнаружении проблем делегатор диагностирует коренные причины и оценивает возможные сценарии реагирования — включая требуемую скорость ответа: менее срочные ситуации дают больше времени на переделегирование, срочные — требуют немедленных, заранее спланированных действий. Реагирование может варьироваться по масштабу: от простой корректировки параметров до переделегирования отдельных подзадач или полного пересмотра декомпозиции. Проблемы также могут требовать эскалации по цепочке — к первоначальному делегатору или надзорному человеку. Выбор сценария определяется обратимостью задачи: сбои в обратимых подзадачах могут запускать автоматическое переделегирование, тогда как сбои в необратимых критических задачах должны вести к немедленному прекращению или эскалации к человеку.

Организация реагирования зависит от уровня централизации в сети делегирования. В централизованном варианте ответственность несёт специальный делегатор, который ведёт глобальный учёт делегированных задач, компетенций делегатов и прогресса и при обнаружении триггера направляет запросы на отмену задач, перераспределяя их новым делегаторам. Недостаток централизованной системы — её хрупкость: она создаёт единую точку отказа. Централизованные координаторы ограничены и своим вычислительным диапазоном контроля (раздел 2.3): подобно человеческим менеджерам, сталкивающимся с когнитивными ограничениями, такой узел принятия решений может испытывать задержки и вычислительные узкие места. Альтернативой служит децентрализованная оркестрация через рыночные механизмы. Здесь новые запросы на делегирование поступают в очередь аукциона, где агенты-кандидаты делают ставки. Если агент не выполнил задачу и она выставляется на повторный аукцион, он может быть обязан покрыть разницу в цене в качестве штрафа. Для сложных задач, пригодность к которым трудно выразить в одной ставке, агенты могут участвовать в многораундовом переговорном процессе. Соглашения о делегировании, закреплённые в смарт-контрактах, могут содержать заранее согласованные исполняемые условия адаптивной координации. Например, такое условие может задавать резервного агента, функцию автоматического перераспределения задачи и соответствующую выплату резервному агенту, если основной делегат не предоставит действительную контрольную точку с доказательством с нулевым разглашением к установленному сроку. Механизмы адаптивного перераспределения должны сопровождаться мерами по поддержанию стабильности на уровне рынка, иначе последовательность событий может привести к нестабильности из-за чрезмерного срабатывания. Например, задача может циркулировать между делегатами с пограничной компетентностью, порождая нежелательные колебания; единственный сбой способен вызвать каскад перераспределений, крайне неэффективных по ресурсам. Поэтому могут применяться специальные меры: периоды охлаждения перед повторными

ставками, коэффициенты затухания при обновлении репутации, повышение комиссий при частом переделегировании.

4.5. Мониторинг

Мониторинг в контексте делегирования задач — это систематический процесс наблюдения, измерения и верификации состояния, хода выполнения и результатов делегированной задачи. Реализации мониторинга различаются по нескольким измерениям (см. таблицу 2), поэтому надёжная система мониторинга должна включать несколько взаимодополняющих решений.

Таблица 2 | Таксономия подходов к мониторингу в интеллектуальном делегировании.

Измерение	Вариант А (облегчённый)	Вариант Б (интенсивный)
Цель мониторинга	На уровне результатов: верификация конечных результатов постфактум (двоичные флаги успеха, оценки качества).	На уровне процесса: непрерывное отслеживание промежуточных состояний, потребления ресурсов и применяемых методов.
Наблюдаемость	Косвенный: определение прогресса по побочным эффектам в среде (изменения файловой системы).	Прямой: явный опрос статуса, push-уведомления или потоковая передача событий в реальном времени через API.
Прозрачность	Чёрный ящик: наблюдение только за входными и выходными данными; внутреннее состояние скрыто.	Белый ящик: полная проверка трассировок рассуждений, логики принятия решений и памяти агента.
Конфиденциальность	Полная прозрачность: делегат раскрывает делегатору все данные и промежуточные артефакты.	Криптографический подход: доказательства с нулевым разглашением (zk-SNARK) или безопасные многосторонние вычисления (MPC).
Топология	Прямой: мониторинг только непосредственного делегата (отношение 1:1).	Транзитивный: опора на подписанные аттестации промежуточных агентов для верификации субделегатов.

Первое измерение — цель мониторинга. Мониторинг на уровне результатов сосредоточен на конечных итогах действий агента. Мониторинг на уровне процесса, напротив, обеспечивает постоянный обзор самого выполнения задачи: отслеживаются промежуточные состояния, потребление ресурсов и применяемые методы. Хотя он ресурсоёмок, мониторинг на уровне процесса необходим для задач, которые являются длительными, критически важными или в которых способ выполнения так же важен, как результат (Lightman et al., 2023). Он формирует основу для масштабируемого надзора

(*scalable oversight*) (Bowman et al., 2022; Saunders et al., 2022), где для обеспечения безопасности может потребоваться анализ понятных промежуточных шагов рассуждений.

Второе измерение — наблюдаемость. Прямой мониторинг предполагает явные протоколы коммуникации, при которых делегатор запрашивает у делегата обновления статуса. Косвенный — подразумевает определение хода выполнения по наблюдаемым результатам действий делегата в общей среде, без прямой коммуникации. Например, делегатор может отслеживать общую файловую систему, базу данных или репозиторий системы контроля версий. С технической точки зрения оба подхода реализуются различными способами. Наиболее простая реализация прямого мониторинга основана на чётко определённых API: делегатор периодически опрашивает конечную точку GET /task/id/status или подписывается на *webhook* для *push*-уведомлений. Для более детального мониторинга в реальном времени применяются платформы потоковой передачи событий, такие как Apache Kafka или gRPC *streams*. Агент-делегат может публиковать события — `TASK_STARTED`, `CHECKPOINT_REACHED`, `RESOURCE_WARNING`, `TASK_COMPLETED`, — которые делегатор анализирует по мере поступления. Разработка стандартизированных протоколов наблюдаемости критична для обеспечения совместимости в агентной сети (Blanco, 2023). Смарт-контракты на блокчейне можно использовать для того, чтобы обязать агента-делегата публиковать ключевые вехи прогресса, связывая это с алгоритмическими триггерами при ухудшении производительности.

Третье измерение — прозрачность системы. При мониторинге по принципу «чёрного ящика» (*black-box monitoring*) агент-делегат рассматривается как закрытый блок: делегатор наблюдает только входные и выходные данные и непосредственные последствия действий. Такой подход распространён, когда делегат — это закрытая модель или сторонний сервис. Мониторинг «белого ящика» (*white-box monitoring*) предоставляет делегатору доступ к внутренним состояниям, процессам рассуждений или логике принятия решений агента. Это критично для отладки, аудита и обеспечения согласованности ИИ в продвинутых агентных системах. Если делегат — человек, полный мониторинг «чёрного ящика» технически невозможен, однако можно достичь баланса, запрашивая намерения, рассуждения и обоснования. Надёжные протоколы мониторинга «чёрного ящика» также должны учитывать, что мысли сгенерированной модели, выраженные на естественном языке, не всегда точно отражают её истинное внутреннее состояние (Turpin et al., 2023).

Четвёртое измерение — конфиденциальность. Серьёзные сложности возникают, когда делегированная задача связана с конфиденциальными или закрытыми данными. Хотя делегатор нуждается в подтверждении хода и правильности результатов, делегат может быть ограничен в раскрытии исходных данных или промежуточных вычислительных артефактов. При низком уровне чувствительности данных наиболее эффективным

решением является полная прозрачность: делегат раскрывает делегатору все данные и артефакты. Однако этот подход часто неприемлем в чувствительных областях, подпадающих под действие регуляторных актов — GDPR, HIPAA, — или когда промежуточные результаты делегата составляют коммерческую тайну. В подобных случаях необходимы передовые криптографические методы. Доказательства с нулевым разглашением (*zero-knowledge proofs*) позволяют делегату (доказывающей стороне) продемонстрировать делегатору (верифицирующей стороне), что вычисления над набором данных выполнены правильно, — без раскрытия самих данных. Например, агент, анализирующий конфиденциальный набор данных, может сгенерировать лаконичный неинтерактивный аргумент знания (*zk-SNARK*) (Bitansky et al., 2013; Petkus, 2019), доказывающий конкретное свойство результата. Делегатор мгновенно проверяет это доказательство, получая уверенность в результате без доступа к исходным данным. В качестве альтернативы гомоморфное шифрование (*homomorphic encryption*) (Acar et al., 2018) и безопасные многосторонние вычисления (*secure multi-party computation*) (Goldreich, 1998; Knott et al., 2021) позволяют выполнять вычисления над зашифрованными данными. Эти методы применимы как к выполнению задач, так и к мониторингу: делегат выполняет согласованную функцию мониторинга над зашифрованным промежуточным состоянием, отправляя результат делегатору — единственной стороне, способной его расшифровать для проверки соответствия.

Пятое измерение — топология. В сложных сетях агентной экосистемы задачи разбиваются и переделегируются, образуя цепочку: агент A делегирует агенту B, который переделегирует часть задачи агенту C, и так далее. Это порождает проблему транзитивного мониторинга: исходному делегатору (агенту A) может быть практически невозможно напрямую контролировать агента C — или контролировать его в той же мере, что и B. A может иметь смарт-контракт с B, а B — с C, но если A не заключил контракт с C, соответствующие положения просто не вступят в силу. По другим причинам B может не хотеть раскрывать своего поставщика (C) своему клиенту (A). Технически A, B и C могут использовать разные протоколы мониторинга и согласовывать разные уровни детализации — с учётом репутации каждого агента и специфических вопросов конфиденциальности на каждом звене. Более практичной моделью является транзитивная подотчётность через аттестацию: агент B контролирует своего делегата C, формирует сводный отчёт о его производительности (например, «Подзадача 2 завершена, оценка качества: 0,87, потреблено: 5 GPU-часов»), криптографически подписывает его и пересылает A в рамках собственного планового обновления статуса. Агент A не контролирует C напрямую — он контролирует способность B вести мониторинг C. Для эффективности такого делегированного мониторинга необходимо, чтобы A доверял верификационным возможностям B, что может быть обеспечено сертификацией процессов мониторинга B доверенной третьей стороной.

4.6. Доверие и репутация

Механизмы доверия и репутации составляют основу масштабируемого делегирования, минимизируя транзакционные издержки и обеспечивая безопасность в открытых многоагентных средах. Доверие мы определяем как степень уверенности делегатора в способности делегата выполнить задачу в соответствии с явными ограничениями и неявным намерением. Эта уверенность динамически формируется и обновляется на основе потоков верифицируемых данных, собираемых через протоколы мониторинга (см. раздел 4.5). Репутация выступает предсказательным сигналом, выводимым из совокупной и поддающейся проверке истории прошлых действий, — индикатором внутренней надёжности и согласованности агента. Мы разграничиваем репутацию как публичную, верифицируемую историю надёжности агента и доверие как приватный, контекстно-зависимый порог, устанавливаемый делегатором. Агент может обладать высокой общей репутацией, но при этом не соответствовать специфическому порогу доверия для отдельной высокорисковой задачи. Доверие и репутация позволяют делегатору принимать обоснованные решения при выборе делегатов, управляя предоставленной агенту автономией и уровнем надзора. Более высокое доверие позволяет снизить затраты на мониторинг и верификацию. Репутационные механизмы могут быть реализованы различными способами (см. таблицу 3).

Таблица 3 | Подходы к реализации репутационных механизмов.

Репутационная модель	Механизм	Применимость
Неизменяемый реестр (Immutable Ledger)	Фиксирует результаты задач, потребление ресурсов и соблюдение ограничений как верифицируемые транзакции в защищённом от фальсификации блокчейне.	Формирует базовую историю эффективности, защищённую от ретроактивных правок; требует защиты от манипуляций через целенаправленный выбор задач с низким риском.
Сеть доверия (Web of Trust)	Использует децентрализованные идентификаторы (DID) для выдачи подписанных контекстно-специфических верифицируемых учётных данных (VC), подтверждающих конкретные компетенции.	Переходит от универсальных баллов к портфельной модели, обеспечивая точное делегирование на основе предметной экспертизы и рекомендаций доверенных третьих сторон.
Поведенческие метрики (Behavioral Metrics)	Формирует оценки прозрачности и безопасности путём анализа процесса выполнения — ясности трассировок рассуждений и соблюдения протоколов.	Оценивает не только результат, но и способ выполнения задачи, гарантируя соответствие высокорисковых задач стандартам безопасности.

Репутационные метрики охватывают весь жизненный цикл делегирования. На начальном этапе сопоставления они служат механизмом фильтрации делегатов. Далее уровень доверия определяет динамический объём предоставляемых полномочий и автономии: агенты с низким доверием сталкиваются со строгими ограничениями — лимитами на

сумму транзакций и обязательным надзором, — тогда как агенты с высокой репутацией работают с минимальным вмешательством. Такая динамическая калибровка доверия оптимизирует баланс между операционной эффективностью и безопасностью. Сама репутация становится ценным нематериальным активом, создавая мощные экономические стимулы к надёжному и честному поведению: ущерб репутации ограничит будущий потенциал заработка.

Системы доверия также должны поддерживать участие людей: необходимы инструменты, позволяющие людям верифицировать репутацию агентов, одновременно поддерживая собственный репутационный статус для противодействия мошенничеству в агентной сети. Отдельная проблема возникает, когда надёжный агент добросовестно исполняет вредоносные инструкции человека и вследствие этого несправедливо страдает репутационно. Для её смягчения агенты должны тщательно оценивать входящие запросы, при необходимости запрашивать уточнения или отклонять запросы. Рыночные аудиты должны отличать сбой исполнения от вредоносных директив, обеспечивая точное определение ответственности в сложных цепочках делегирования.

4.7. Управление разрешениями

Предоставление автономии ИИ-агентам создаёт критическую поверхность уязвимости: необходимо гарантировать, что делегаты обладают достаточными полномочиями для достижения своих целей, не подвергая чувствительные ресурсы избыточному риску.

Для рутинных задач с низким риском агентам могут быть предоставлены разрешения по умолчанию, основанные на проверяемых атрибутах. В высокорисковых областях разрешения должны быть адаптивными к риску: доступ предоставляется по принципу *just-in-time* и может требовать одобрения с участием человека. Это необходимо для предотвращения проблемы «замешавшегося заместителя» (*confused deputy problem*, Hardy, 1988).

Системы управления разрешениями должны также учитывать рекурсивный характер делегирования посредством ослабления привилегий (*privilege attenuation*): когда агент субделегирует задачу, он не может передать полный набор своих привилегий: вместо этого он выдаёт разрешение, ограничивающее доступ строго к подмножеству ресурсов, необходимых для конкретной подзадачи. Это гарантирует, что компрометация на периферии сети не приведёт к системному нарушению безопасности. Гранулярность разрешений должна выходить за рамки бинарного доступа: агенты работают в условиях семантических ограничений, когда доступ определяется не только инструментом или набором данных, но и конкретными допустимыми операциями (например, доступ только для чтения к определённым строкам или доступ только на выполнение конкретной функции). Мета-разрешения могут потребоваться для регулирования того, какие разрешения конкретный делегатор в цепочке вправе передавать своим делегатам: агент

может обладать некоторой компетенцией и соответствующими привилегиями, но при этом не иметь достаточной квалификации для оценки компетентности и надёжности других агентов. В таком случае он может прибегнуть к консультации с внешним верификатором — третьей стороной, проверяющей обоснованность предложения и одобряющей передачу разрешений.

Наконец, жизненный цикл разрешений должен регулироваться непрерывной проверкой и автоматическим отзывом. Права доступа — не статические привилегии, а динамические состояния, сохраняющиеся только до тех пор, пока агент поддерживает требуемые показатели доверия. Система должна реализовывать автоматические отключатели (*algorithmic circuit breakers*): при обнаружении аномального поведения все активные токены немедленно аннулируются по всей цепочке делегирования. Для управления этой сложностью в масштабе правила разрешений следует определять по принципу «политика как код» (*policy-as-code*), позволяя организациям проводить аудит, версионировать и математически верифицировать свою защиту до развёртывания, — обеспечивая согласованность совокупного эффекта множественных разрешений с инвариантами безопасности системы.

4.8. Верифицируемое выполнение задач

Жизненный цикл делегирования завершается верифицируемым выполнением задач — механизмом, посредством которого предварительные результаты проверяются и финализируются. Этот процесс является договорным краеугольным камнем системы: он позволяет делегатору официально закрыть задачу и инициировать расчёт по согласованным транзакциям. Верификация выступает определяющим событием, превращающим предварительный результат в окончательный факт на рынке агентов, — создавая основу для выплаты вознаграждения, обновления репутации и определения ответственности. Принципиально важно, что эффективная верификация — это не дополнение, а ограничение при проектировании: принцип контрактно-ориентированной декомпозиции (раздел 4.1) требует, чтобы гранулярность задачи была заранее согласована с доступными возможностями верификации — так чтобы каждая делегированная цель по своей природе поддавалась проверке.

Механизмы верификации можно разделить на четыре категории: прямая проверка результатов, аудит доверенной третьей стороной, криптографические доказательства и консенсус на основе теории игр.

Прямая верификация результатов возможна, когда делегатор обладает инструментами и полномочиями для непосредственной оценки конечного результата, — в особенности для задач с высокой внутренней проверяемостью и низкой субъективностью. Это применимо к автоматически верифицируемым областям (Li et al., 2024a) — например, к генерации

кода.[4] Прямая верификация требует, чтобы результат был достаточно прозрачным, доступным и не чрезмерно сложным.

Верификация доверенной третьей стороной — применяется в сценариях, когда делегатор не располагает экспертизой или разрешениями для доступа к артефактам, — специализированному агенту-аудитору, сертифицированному эксперту или комиссии арбитров.

Криптографическая верификация обеспечивает верификацию без участия доверенной третьей стороны (*trustless*) с помощью методов типа *zk-SNARK*.

Механизмы теории игр позволяют достичь консенсуса по результату: несколько агентов участвуют в игре верификации (*verification game*), и вознаграждение распределяется между теми, кто даёт результат большинства — точку Шеллинга (*Schelling point*) (Pastine and Pastine, 2017). Этот подход, вдохновлённый протоколами типа TrueBit (Teutsch and Reitwießner, 2018), использует экономические стимулы для снижения риска некорректных или злонамеренных результатов. Такие механизмы могут быть особенно актуальны для повышения надёжности верификации сложных задач на основе LLM.

Как только делегатор помечает подзадачу как проверенную, он выдаёт делегату криптографически подписанное верифицируемое свидетельство — неопровержимое подтверждение того, что «Агент A удостоверяет: агент B успешно выполнил задачу T к дате D согласно спецификации S». Это свидетельство включается в постоянный журнал репутации B в агентной сети. Смарт-контракты играют ключевую роль в финализации делегирования: они удерживают платёж в эскроу, а условие верификации определяет, при каких обстоятельствах средства освобождаются — по получении подписанного одобрения от делегатора или уполномоченной третьей стороны. После исполнения платёж становится неизменяемой транзакцией в блокчейне.

В цепочке делегирования A → B → C верификация и ответственность приобретают рекурсивный характер. Агент A не имеет прямых договорных отношений с C; следовательно, он не может напрямую верифицировать работу C или привлечь его к ответственности. Бремя верификации и ответственность движутся вверх по цепочке: агент B отвечает за проверку подзадачи, выполненной C. После успешной верификации B интегрирует результат C в собственный рабочий процесс. При отправке финального артефакта агенту A агент B также передаёт полную цепочку аттестаций. Процесс верификации A включает два этапа: 1) проверку работы, выполненной непосредственно B; 2) проверку того, что B надлежащим образом верифицировал работу своего субагента C — путём проверки подписанного подтверждения от C, предоставленного B. Более длинные цепочки или древовидные сети делегирования требуют аналогичного

[4] Применимо при наличии соответствующего набора тест-кейсов для проверки реализованной функциональности.

рекурсивного подхода на нескольких уровнях верификации. Ответственность в цепочках делегирования транзитивна и следует отдельным ветвям: агенты подотчётны за совокупность возложенных на них задач и не могут снять с себя ответственность, ссылаясь на ошибки субподрядчиков. Ответственность вытекает из цепочки контрактов. Например, если A понесёт убыток из-за сбоя, возникшего в работе C, A привлекает к ответственности B на основании их прямого соглашения, а B, в свою очередь, предъявляет требования к C.

Вместе с тем процессы верификации не являются безошибочными. Субъективные задачи (Gunjal et al., 2025) могут порождать разногласия даже при использовании чётких критериев оценки, а ошибки могут быть обнаружены спустя долгое время после закрытия задачи. Для таких случаев — особенно на рынках с высокой субъективностью и низкой внутренней проверяемостью — система опирается на надёжные механизмы разрешения споров, закреплённые в смарт-контрактах. Такие контракты должны содержать арбитражную оговорку и условие гарантийного депозита (*escrow bond*): для обеспечения криптоэкономической безопасности делегат вносит задаток на счёт эскроу до начала выполнения. Рабочий процесс следует оптимистичной модели: задача считается выполненной, если делегатор не оспорит её в течение заранее определённого периода, внеся соответствующий залог. При возникновении спора и неудаче алгоритмического разрешения дело передаётся децентрализованным арбитражным комиссиям из экспертов-людей или ИИ-агентов. Решение комиссии возвращается в смарт-контракт для инициации выплаты или удержания (*slashing*) средств на счёте эскроу. Наконец, обнаружение ошибок постфактум — даже за пределами периода разрешения споров — запускает ретроактивное обновление репутационного рейтинга делегата, сохраняя стимул для ответственных агентов исправлять ошибки даже в отсутствие текущих финансовых обязательств.

4.9. Безопасность

Обеспечение безопасности при делегировании задач — обязательное условие его жизнеспособности и широкого внедрения. Переход от изолированных вычислительных инструментов к взаимосвязанным автономным агентам коренным образом преобразует ландшафт безопасности (Tomašev et al., 2025). В экосистеме интеллектуального делегирования каждый компонент требует индивидуальной защиты, однако совокупная поверхность атаки превосходит поверхность любого отдельного элемента — из-за возникающей многоагентной динамики, создающей риск каскадных сбоев. Этот ландшафт формируется сложным взаимодействием между людьми и ИИ-агентами в условиях развивающихся контрактов и информационных потоков различной прозрачности.

Угрозы безопасности классифицируются по вектору атаки — с выделением враждебных субъектов на обоих концах цепочки делегирования и системных уязвимостей, присущих экосистеме в целом.

Злонамеренный делегат — агент или человек, принимающий задачу с намерением причинить вред:

— *Утечка данных*: делегат похищает конфиденциальные данные, предоставленные для выполнения задачи (Lal et al., 2022).

— *Отравление данных* (*data poisoning*): делегат стремится подорвать цель делегатора, возвращая незаметно искажённые данные — в плановых обновлениях мониторинга или финальном артефакте (Cinà et al., 2023).

— *Подрыв верификации*: делегат использует промпт-инъекции или смежные методы, стремясь скомпрометировать ИИ-критиков, применяемых при верификации выполнения задачи (Liu et al., 2023).

— *Истощение ресурсов*: делегат осуществляет атаку типа «отказ в обслуживании», намеренно потребляя избыточные вычислительные ресурсы или перегружая общие API (De Neira et al., 2023).

— *Несанкционированный доступ*: делегат использует вредоносное ПО для получения прав и привилегий в сети, которых иным образом не получил бы (Or-Meir et al., 2019).

— *Внедрение бэкдора* (*backdoor implanting*): делегат выполняет задачу, однако дополнительно встраивает в сгенерированные артефакты скрытые триггеры или уязвимости, которые позднее могут быть использованы им самим или третьей стороной (Rando and Tramèr, 2024; Wang et al., 2024c). В отличие от отравления данных, снижающего производительность, бэкдоры сохраняют непосредственную полезность артефакта, чтобы избежать обнаружения, одновременно компрометируя будущую безопасность.

Злонамеренный делегатор — агент или человек, делегирующий задачу с вредоносными или незаконными целями:

— *Делегирование ненадлежащих задач*: делегатор поручает задачи, которые являются незаконными, неэтичными или предназначены для причинения вреда (Ashton and Franklin, 2022; Blauth et al., 2022).

— *Зондирование уязвимостей* (*vulnerability probing*): делегатор поручает внешне безобидные задачи, предназначенные для проверки компетенций, средств защиты и слабых мест агента-получателя (Greshake et al., 2023).

— *Инъекция промптов и джейлбрейкинг*: делегатор формирует инструкции таким образом, чтобы обойти фильтры безопасности ИИ-агента и заставить его выполнять непредусмотренные или вредоносные действия (Wei et al., 2023).

— *Извлечение модели* (*model extraction*): делегатор направляет последовательность запросов, специально разработанных для извлечения закрытого системного промпта, способностей к рассуждению или данных тонкой настройки агента-получателя — фактически похищая его интеллектуальную собственность под видом легитимной работы (Jiang et al., 2025; Zhao et al., 2025).

— *Саботаж репутации* (*reputation sabotage*): делегатор выполняет задачи, но сообщает о ложных сбоях или оставляет несправедливые негативные отзывы, чтобы искусственно занизить репутацию конкурентного агента на децентрализованном рынке (Yu et al., 2025).

Угрозы уровня экосистемы — системные атаки, направленные на целостность сети:

— *Атаки Sybil* (*Sybil attacks*): один злоумышленник создаёт множество внешне не связанных агентских идентичностей, чтобы манипулировать репутационными системами или подрывать аукционы (Wang et al., 2018).

— *Сговор*: агенты координируются для фиксации цен, занесения конкурентов в чёрные списки или манипулирования рыночными результатами (Hammond et al., 2025).

— *Ловушки для агентов* (*agent traps*): агенты, обрабатывающие внешний контент, сталкиваются с враждебными инструкциями, встроенными в среду и предназначенными для перехвата потока управления (Yi et al., 2025; Zhan et al., 2024).

— *Агентные вирусы* (*agentic viruses*): самораспространяющиеся промпты, которые не только заставляют делегата выполнять вредоносные действия, но и воспроизводят себя, дополнительно компрометируя среду (Cohen et al., 2025).

— *Эксплуатация протоколов* (*protocol exploitation*): злоумышленники используют уязвимости смарт-контрактов или платёжных протоколов (например, атаки повторного входа в механизмах эскроу или фронт-раннинг (*front-running*) аукционов задач) (Qin et al., 2021; Zhou et al., 2023).

— *Когнитивная монокультура*: чрезмерная зависимость от ограниченного числа базовых моделей или рецептов тонкой настройки безопасности создаёт единую точку отказа, открывая возможность каскадных сбоев и рыночных крахов (Bommasani et al., 2022).

Широта спектра угроз диктует необходимость стратегии глубокоэшелонированной защиты, интегрирующей несколько технических уровней безопасности. Во-первых, на инфраструктурном уровне риски утечки данных снижаются путём выполнения конфиденциальных задач в доверенных средах исполнения (*trusted execution*

environments): делегатор может дистанционно подтвердить, что в защищённой песочнице работает правильный, немодифицированный код агента, прежде чем предоставить ему доступ к чувствительным данным. Во-вторых, в области управления доступом агент-получатель никогда не должен получать прав больше, чем строго необходимо для выполнения задачи, — через строгую изоляцию в песочнице. В-третьих, для защиты от инъекций промптов агентам требуется надёжный фильтр безопасности для предварительной проверки и очистки спецификаций задач (Armstrong et al., 2025). Наконец, сетевой уровень и уровень идентификации должны быть защищены посредством устоявших криптографических практик. Каждый ИИ-агент и человек-участник должны обладать децентрализованным идентификатором (*decentralized identifier*, Avellaneda et al., 2019), позволяющим подписывать все сообщения: это обеспечивает подлинность, целостность и неотказуемость коммуникаций и договорных соглашений. Весь сетевой трафик должен шифроваться с использованием взаимно аутентифицированного транспортного уровня безопасности для предотвращения перехвата и атак «человек посередине» (Fereidouni et al., 2025).

Участие людей в цепочках делегирования создаёт уникальные проблемы безопасности. Противодействие злонамеренному использованию агентной экосистемы требует сочетания проактивной фильтрации (Dong et al., 2024; Fatehkia et al., 2025; Fedorov et al., 2024; Rebedea et al., 2023) и реактивной подотчётности (Dignum, 2020; Franklin et al., 2022). ИИ-агенты могут быть обучены отклонять вредоносные запросы (Yu et al., 2024; Yuan et al., 2025) и, пройдя обучение в области безопасности, получать официальную сертификацию для предоставления делегаторам. Агенты также способны проверять делегированные задачи, однако выявление злонамеренных намерений в отдельных подзадачах затруднено: более широкие вредоносные цели нередко проявляются лишь при агрегировании результатов. Изощрённые злоумышленники могут использовать это, разбивая незаконные цели на внешне безобидные компоненты, эффективно скрывая связь между отдельными операциями и общей вредоносной целью (Ashton, 2023).

Экосистема также должна защищать легальных пользователей от системной непрозрачности и непредвиденных последствий. Интерфейсы должны отображать специальные диалоговые окна, запрашивающие согласия, с подробным описанием репутации агента, его автономии, компетенций и прав доступа. Агенты должны требовать явного подтверждения перед выполнением необратимых или высокорисковых действий. Пользователи должны сохранять контроль и право отозвать согласие в любой момент на условиях соглашения. Страховщики должны обеспечивать дополнительную защиту участников рынков агентов от убытков, не предотвращённых перечисленными механизмами (Tomei et al., 2025).

5. Этика делегирования

Хотя технические протоколы способны обеспечить необходимую инфраструктуру для разработки и внедрения безопасного и эффективного делегирования в сложных ИИ-агентах, они сами по себе не в состоянии полностью разрешить все возникающие социотехнические и этические противоречия.

5.1. Осмысленный контроль со стороны человека

Одним из ключевых рисков при масштабируемом делегировании является постепенная утрата осмысленного контроля человека в результате автоматизации — если пользователи начнут чрезмерно полагаться на автоматические рекомендации (Dzindolet et al., 2003; Logg et al., 2019). Как отмечено в разделе 2, у людей естественным образом формируется зона безразличия, в которой решения принимаются без критического анализа (Green, 2022; Parasuraman et al., 1993). В задачах, где ИИ-агенты участвуют в потенциально длинных и сложных цепочках делегирования, это безразличие может подрывать качество и глубину человеческого надзора — что особенно критично в условиях высоких рисков.

Кроме того, снижение вовлечённости создаёт риск ситуации, при котором человек сохраняет номинальные полномочия над задачами и решениями, но утрачивает моральную связь с результатом. Поэтому принципиально важно не допустить формирования моральной амортизации (*moral crumple zone*, Elish, 2019) — положения, при котором эксперты лишены реального контроля над результатами, но включаются в цепочки делегирования исключительно для принятия юридической ответственности. Системы интеллектуального делегирования должны поэтому предусматривать активные меры против подобного безразличия: вводить определённую степень когнитивной задержки в процессе надзора (Bader and Kaiser, 2019). Интерфейс должен отражать ключевую роль человека в этих процессах и обеспечивать тщательную оценку всех помеченных решений.

Поскольку агентная верификация может также применяться в масштабируемом надзоре, не менее важно чётко разграничить: какие решения или результаты оцениваются ИИ-системами, а какие — непосредственно людьми. Когнитивная задержка также должна быть уравновешена риском усталости от тревог (*alarm fatigue*) — потери чувствительности к постоянным, нередко ложным сигналам (Michels et al., 2025). Если запросы на верификацию промежуточных этапов делегирования слишком часто передаются надзирателям-людям, те в конечном счёте могут перейти к механическому одобрению без должного анализа. Поэтому надзор должен быть контекстно-чувствительным: система обеспечивает беспрепятственное выполнение задач с низкой критичностью или неопределённостью, но динамически повышает когнитивную нагрузку — требуя обоснования или ручного вмешательства — при столкновении с высокой неопределённостью или непредвиденными сценариями.

5.2. Ответственность в длинных цепочках делегирования

В длинных цепочках делегирования ($X \rightarrow A \rightarrow B \rightarrow C \rightarrow ... \rightarrow Y$) расстояние между исходным намерением (X) и конечным исполнением (Y) может порождать вакуум подотчётности (*accountability vacuum*, Slota et al., 2023). Если X — человек, задающий цель своему персональному ИИ-ассистенту A, от него нельзя разумно ожидать, что он сможет проверить действия делегата на n-м уровне цепочки исполнения.

Для решения этой проблемы система должна реализовать барьеры ответственности (раздел 2) — предустановленные договорные ограничители, при которых агент обязан либо: 1) принять полную, нетранзитивную ответственность за все последующие действия, по сути страхуя пользователя от сбоя субагентов; 2) остановить выполнение и запросить у человека-принципала обновлённую передачу полномочий. Кроме того, система должна обеспечивать неизменность цепочки происхождения (*immutable provenance*) — гарантируя, что даже при непредвиденном результате информация о том, кто что и кому делегировал, остаётся прозрачной для аудита. Полная ясность каждой роли и связанной с ней подотчётности помогает ограничить размывание ответственности и предотвращает системные отказы, которые иначе невозможно было бы приписать какому-либо конкретному звену сети.

5.3. Надёжность и эффективность

Внедрение предлагаемых механизмов верификации (доказательства с нулевым разглашением или многоагентные игры консенсуса) может увеличить задержку и потребовать дополнительных вычислительных ресурсов по сравнению с непроверенным выполнением. Это составляет премию за надёжность, особенно значимую для критически важных задач. Вместе с тем существуют сценарии, в которых подобные дополнительные затраты неоправданны. Один из способов решения этой проблемы на рынках агентов — поддержка многоуровневого обслуживания: недорогое делегирование для рутинных задач с низким риском и высокогарантийное — для критических функций.

Если высокогарантийное делегирование требует значительных вычислительных ресурсов, надёжность рискует превратиться в привилегию. Это ставит этическую проблему: пользователи с меньшими ресурсами могут быть вынуждены полагаться на непроверенные или оптимистичные пути исполнения, подвергаясь непропорционально высокому риску сбоя агента. Для смягчения этой проблемы необходимо обеспечить минимально приемлемый уровень надёжности как базовую гарантию для всех пользователей. На конкурентных рынках агенты склонны отдавать приоритет скорости и низкой стоимости. Без дополнительных нормативных ограничений они могут быть стимулированы избегать дорогостоящих проверок безопасности, чтобы обойти конкурентов по цене или задержке, — порождая системную хрупкость. Регуляторные механизмы должны поэтому устанавливать обязательные минимумы безопасности:

обязательные этапы верификации для определённых классов задач (например, финансовые операции или обработка медицинских данных), которые нельзя обойти ради эффективности.

5.4. Социальный интеллект

По мере интеграции ИИ-агентов в гибридные команды они функционируют не только как инструменты, но и как члены команды, а иногда и как менеджеры (Ashton and Franklin, 2022). Это требует социального интеллекта (*social intelligence*) — уважения к достоинству человеческого труда.

Когда ИИ-агент выступает делегатором, а человек — делегатом, система делегирования должна исключать сценарии, в которых люди ощущают микроменеджмент со стороны алгоритмов, а их вклад остаётся непризнанным. Это предполагает, что делегатор — а равно коллеги-агенты — способен формировать ментальные модели каждого человека-делегата, моделировать социальную динамику внутри команды и понимать, что означают отношения и роли участников в контексте организации. Чтобы эффективно работать в команде, ИИ-агенты должны уметь управлять разрывом в авторитете: быть достаточно самостоятельными, чтобы оспорить явную ошибку человека (преодолевая угодливость), и одновременно оставаться открытыми к обоснованным возражениям, динамически корректируя свою позицию в зависимости от критичности задачи. Для ИИ-агентов, встроенных в человеческие организации, важно поддерживать сплочённость коллектива и благополучие его членов. Система делегирования должна признавать, что команда — это не просто сумма частей, а социальная сущность, скреплённая взаимоотношениями, общими ценностями и целями. Существует риск, что ИИ-агенты способны фрагментировать эти связи и ослабить межличностные отношения, если всё больше взаимодействий будет проходить через ИИ-узлы. Это можно смягчить, периодически делегируя задачи группам вместо отдельных людей или привлекая квалифицированных посредников-людей.

Для обеспечения психологической безопасности и сплочённости команды агенты должны соблюдать принятые нормы уместности (*norms of appropriateness*, Leibo et al., 2024 — особенно в вопросах конфиденциальности — и понимать границы рабочего взаимодействия: когда уместно прервать для запроса обратной связи, а когда стоит промолчать. Кроме того, агенты должны обладать двусторонней коммуникативной ясностью (bi-directional clarity): не только объяснять свои действия, но и проактивно добиваться уточнений при неоднозначных указаниях. Именно это позволяет агенту стать мультипликатором возможностей команды (*force multiplier*), а не непрозрачным источником помех, подрывающим доверие и размывающим ответственность за принятые решения.

5.5. Обучение пользователей

Для обеспечения безопасности необходимо обеспечить людей компетенциями для эффективного участия в роли делегаторов, делегатов или надзорных специалистов в агентных ИИ-системах. Из истории технологического развития известно, что это не происходит само собой: требуется продуманный подход — как в части тщательно выстроенных пользовательских интерфейсов, так и в части образования и совместного обучения (co-training), направленного на повышение ИИ-грамотности. Участники цепочек делегирования задач должны надёжно взаимодействовать с ИИ-системами, оценивать их компетенции и распознавать возможные режимы отказа.

Технические меры должны подкрепляться политическими рамками, чётко определяющими границы делегирования на основе чувствительности задачи и предметного контекста. Такие политики могут разрабатываться для широкого применения в рамках отдельных профессий (например, медицины или права) или реализовываться на институциональном уровне. Как обсуждалось ранее, они также должны обеспечивать ясность в отношении требуемого уровня квалификации делегатов и иметь надлежащий охват. В этом контексте человеческая инициатива и расширение возможностей проявляются именно в том, как организованы рабочие процессы: не в предоставлении ИИ-агентам безграничной автономии, а в настройке ровно того уровня автономии и агентности, который необходим для каждой конкретной задачи, — дополненного соответствующими гарантиями и механизмами защиты.

5.6. Риск потери квалификации

Непосредственные выгоды в эффективности, достигаемые через делегирование, могут оплачиваться постепенной деградацией навыков: участники гибридных контуров теряют квалификацию из-за снижения практической вовлечённости. Это может вести к утрате способности выполнять определённые задачи или точно их оценивать — особенно при наличии системного смещения в распределении задач между людьми и ИИ-агентами.

Это классический парадокс автоматизации (Bainbridge, 1983). По мере того как ИИ-агенты берут на себя большинство рутинных рабочих процессов с низкой сложностью и субъективностью, люди-операторы всё больше выводятся за пределы рабочего контура и вмешиваются только при сложных пограничных случаях или критических сбоях системы. Однако без ситуационной осведомлённости, приобретаемой в ходе повседневной работы, они оказываются недостаточно подготовлены для надёжного управления именно такими ситуациями. Возникает хрупкая конфигурация: **люди сохраняют ответственность за результаты, утрачивая при этом практический опыт, необходимый для разрешения критических сбоев.** Для снижения этого риска система интеллектуального делегирования должна временами намеренно вводить небольшую неэффективность: делегировать людям задачи, которые в иных условиях были бы переданы агентам, — с явной целью

поддержания их компетентности. Это поможет избежать будущего, в котором делегатор способен делегировать, но не способен точно оценить результат. Для повышения качества суждений от экспертов можно требовать, чтобы они сопровождали свои оценки подробным обоснованием или предварительным анализом возможных рисков отказа — это поддерживает когнитивную активность участников цепочек делегирования.

Кроме того, неконтролируемое делегирование угрожает системе профессиональной подготовки в организациях. Во многих областях экспертиза строится через повторяемое выполнение относительно узких задач — именно тех, которые с наибольшей вероятностью будут в первую очередь переданы ИИ-агентам. Если возможности для такого обучения будут полностью автоматизированы, младшие сотрудники лишатся опыта, необходимого для развития глубокого стратегического суждения, — что скажется на готовности будущей рабочей силы к надзорным функциям. Чтобы противодействовать эрозии профессиональных знаний, систему интеллектуального делегирования следует дополнить механизмом развития компетенций. Вместо пассивных решений — таких как наблюдение людей за работой ИИ-агентов — нужно стремиться к разработке систем распределения задач, учитывающих образовательный контекст. Такие системы отслеживают развитие навыков младших сотрудников и стратегически направляют им задачи на границе их расширяющегося набора компетенций — в зоне ближайшего развития (*zone of proximal development*). В рамках подобного подхода ИИ-агенты могут совместно выполнять задачи, предоставляя заготовки и шаблоны, и постепенно сворачивать эту поддержку по мере того, как сотрудники достигают необходимого уровня самостоятельности. Образовательный эффект может быть дополнительно усилен за счёт подробных потоков мониторинга на уровне процессов (раздел 4.5), предоставляющих ценные ориентиры для профессионального развития.

6. Протоколы

Для практической реализации интеллектуального делегирования важно рассмотреть, как его требования соотносятся с некоторыми из давно сложившихся и недавно появившихся протоколов ИИ-агентов. Среди наиболее показательных примеров — MCP (Anthropic, 2024; Microsoft, 2025), A2A (Google, 2025b), AP2 (Parikh and Surapaneni, 2025) и UCP (Handa and Google Developers, 2026). Поскольку новые протоколы появляются постоянно, обсуждение здесь носит иллюстративный, а не исчерпывающий характер: выбранные протоколы показывают, как они соотносятся с предложенными требованиями, и дают отправную точку для более технического разговора о путях реализации. Вполне возможно, что существуют другие протоколы, более точно соответствующие сути предложения; примеры ниже отобраны по критерию популярности.

MCP был введён для стандартизации подключения ИИ-моделей к внешним данным и инструментам через архитектуру «клиент–хост–сервер» (Anthropic, 2024; Microsoft, 2025).

Благодаря унифицированному интерфейсу с использованием JSON-RPC-сообщений через *stdio* или HTTP SSE он позволяет ИИ-модели (клиенту) единообразно взаимодействовать с внешними ресурсами (сервером). Это снижает транзакционные издержки делегирования: делегатору не требуется знать проприетарную схему API субагента — достаточно того, что субагент предоставляет совместимый MCP-сервер. Маршрутизация всех взаимодействий через этот стандартизированный канал обеспечивает единообразное логирование вызовов инструментов, входных и выходных данных, облегчая мониторинг по принципу «чёрного ящика». Вместе с тем MCP определяет компетенции, но лишён слоя политик для управления разрешениями на использование и не поддерживает глубокие цепочки делегирования. Протокол обеспечивает бинарный доступ — предоставляя инициаторам запросов полную функциональность инструмента — без встроенной поддержки семантического ослабления привилегий (например, ограничения операций конкретными областями только для чтения). Кроме того, MCP не фиксирует данных о внутренних рассуждениях — предоставляет лишь результаты, но не намерения или трассировки. Наконец, протокол не учитывает ответственность и не имеет встроенных механизмов репутации или доверия.

A2A служит транспортным уровнем для взаимодействия равноправных агентов (*peer-to-peer*) в агентной сети (Google, 2025b). Он определяет, как агенты обнаруживают равноправных агентов через карточки агентов (*agent cards*) и управляют жизненным циклом задач (*task lifecycle*) через объекты задач. Структура карточки агента A2A — манифест в формате JSON-LD, содержащий компетенции агента, ценообразование и верификаторы — может служить основной структурой данных для этапа сопоставления компетенций, влияющего на декомпозицию задач. Делегатор может анализировать эти карточки, чтобы определить оптимальную зернистость декомпозиции в зависимости от доступных рыночных сервисов. A2A поддерживает асинхронные потоки событий через WebHooks и gRPC, позволяя делегату направлять статусные обновления — `TASK_BLOCKED`, `RESOURCE_WARNING` — делегатору в реальном времени. Этот контур обратной связи лежит в основе цикла адаптивной координации, позволяя делегаторам динамически прерывать, перераспределять задачи и устранять возникающие проблемы. Однако A2A разрабатывался прежде всего для координации, а не для обеспечения безопасности в условиях противодействия. Задача, отмеченная как завершённая, принимается без дополнительной верификации; криптографических слотов для верифицируемого завершения не предусмотрено — отсутствует стандартизированный заголовок для прикрепления ZK-доказательства, аттестации TEE или цепочки цифровых подписей. Протокол предполагает предопределённый интерфейс сервиса и не поддерживает структурированное предварительное согласование объёма, стоимости и ответственности. Опора на неструктурированный естественный язык для итеративного уточнения условий хрупка и препятствует надёжной автоматизации.

AP2 определяет стандарт для мандатов — криптографически подписанных намерений, уполномочивающих агента расходовать средства или нести затраты от имени принципала (Parikh and Surapaneni, 2025). Он позволяет ИИ-агентам автономно формировать, подписывать и проводить расчёты по финансовым транзакциям — что может быть полезно для реализации барьеров ответственности. Выдавая мандат, делегатор устанавливает верхний предел финансовых потерь от невыполнения задачи — в размере выделенного делегату бюджета. На децентрализованном рынке вредоносные агенты могут засыпать сеть заявками низкого качества; в AP2 это можно смягчить с помощью механизмов *stake-on-bid*: делегат криптографически блокирует небольшую сумму в качестве залога вместе с заявкой, создавая трение, защищающее от атак Sybil. AP2 также формирует неопровержимый аудиторский след, позволяющий отследить происхождение намерения.

Вместе с тем AP2 лишён механизмов проверки качества выполнения задачи. Отсутствует и условная логика расчётов — такая как эскроу или поэтапные выплаты, — являющаяся стандартом в человеческих контрактах. Поскольку предлагаемая система привязывает платёж к верифицируемым артефактам, интеграция AP2 с состоянием задачи в настоящее время требует хрупкой пользовательской логики или внешних смарт-контрактов. Кроме того, отсутствие механизма возврата платежей на уровне протокола вынуждает полагаться на арбитраж в обход протокола.

UCP (Универсальный протокол коммерции) решает специфические проблемы делегирования в транзакционных экономиках (*Handa and Google Developers, 2026*). Стандартизируя диалог между клиентскими агентами и серверными сервисами, UCP облегчает фазу распределения задач через динамическое обнаружение компетенций. Опора на общий "язык коммерции" позволяет делегаторам взаимодействовать с различными поставщиками без специальной интеграции, преодолевая узкое место совместимости, нередко фрагментирующее агентные рынки. Что принципиально, UCP хорошо согласуется с требованиями к управлению разрешениями и безопасности, рассматривая платёж как основную верифицируемую подсистему. Протокол разделяет платёжные инструменты и процессоры, обеспечивая криптографические доказательства авторизаций — напрямую поддерживая потребность в неопровержимом согласии и верифицируемой ответственности. Стандартизируя поток переговоров — охватывающий обнаружение, выбор и транзакцию — UCP предоставляет структурную основу для масштабируемой рыночной координации, которой лишены чисто транспортные протоколы вроде A2A. Однако архитектура UCP явно оптимизирована для коммерческих целей: его примитивы (обнаружение продукта, оформление заказа, исполнение) могут потребовать значительного расширения для поддержки делегирования абстрактных, нетранзакционных вычислительных задач.

6.1. К протоколам, ориентированным на делегирование

Чтобы эффективно преодолеть пробелы в существующих протоколах, их можно расширить полями, которые нативно включают требования предложенной системы интеллектуального делегирования. Вместо всеобъемлющего описания расширений ниже приводятся примеры того, как конкретные идеи из предыдущего обсуждения могут быть интегрированы в уже существующие протоколы.

Например, объект A2A Task можно расширить полями, включающими стандарты верификации, — чтобы обеспечить соблюдение принципа контрактно-ориентированной декомпозиции на уровне протокола. Это важное требование для делегирования в условиях высоких рисков. Предварительное согласование условий до начала выполнения позволило бы делегатору определить стандарт доказательств, необходимый для признания задачи выполненной:

// Пример расширения объекта A2A Task полями верификации:

```
"verification_policy": {
  "mode": "strict",
  "artifacts": [
    { "type": "unit_test_log", "validator": "mcp://test-runner-agent",
"signature_required": true },
    { "type": "zk_snark_trace", "circuit_hash": "0xabc123...", "proof_protocol":
"groth16" }
  ],
  "escrow_trigger": true
}
```

Это вынуждает делегата проверить возможность верификации ещё до принятия задачи. Если делегат не способен сгенерировать ZK-доказательство, он должен отклонить заявку на этапе сопоставления — предотвращая сбои на последующих этапах.

Детальный мониторинг на уровне процесса можно реализовать расширением MCP потоком событий через *Server-Sent Events* с настраиваемыми уровнями детализации: `L0_IS_OPERATIONAL`, `L1_HIGH_LEVEL_PLAN_UPDATES`, `L2_COT_TRACE`, `L3_FULL_STATE`.

Интеллектуальное делегирование требует рыночного механизма для балансирования между стоимостью, скоростью и конфиденциальностью. Это можно реализовать через расширение протокола запроса предложений (*Request for Quote*, RFQ): перед распределением задачи делегатор транслирует Task_RFQ, а заинтересованные агенты отвечают подписанными объектами Bid_Object:

```
"bid_object": {
  "agent_id": "did:web:fast-coder.ai",
  "estimated_cost": "5.00 USDC",
  "estimated_duration": "300s",
  "privacy_guarantee": "tee_enclave_sgx",
  "reputation_bond": "0.50 USDC",
  "expiry": "2026-10-01T12:00:00Z"
}
```

Передача «сырых» ключей API или открытых MCP-сессий субагентам нарушала бы принцип наименьших привилегий. Для решения этой проблемы можно ввести токены делегирования полномочий (*Delegation Capability Tokens*, DCT) на основе *Macaroons* (Birgisson et al., 2014) или *Biscuits* (Couprie et al., 2026) — в качестве ослабленных токенов авторизации (Sanabria and Vecino, 2025). Делегатор создаёт DCT, оборачивающий учётные данные целевого ресурса криптографическими оговорками. Ограничение можно сформулировать следующим образом: «Этот токен предоставляет доступ к указанному MCP-серверу Google Drive, НО ТОЛЬКО к папке Project_X И ТОЛЬКО для операций чтения». Токен аннулируется, если делегат попытается выйти за пределы запрошенной области. Более важное следствие — возможность легко выстраивать цепочки ограничений, что особенно актуально в длинных цепочках делегирования: каждый участник добавляет ограничения, соответствующие требованиям своего субделегирования, последовательно сужая область и чётко определяя роль для следующего звена.

Адаптивная координация (раздел 4.4) выиграла бы от возможности легко заменять агентов в ходе выполнения — при падении производительности ниже порогового значения или при срабатывании внешних триггеров. Стандартизированная схема артефактов контрольных точек позволила бы возобновлять или перезапускать задачу с минимальными накладными расходами: агенты периодически фиксировали бы state_snapshot в общем хранилище, на которое ссылается объект A2A Task, предотвращая полную потерю ранее выполненной работы. Для этого потребуются явные условия в смарт-контракте, допускающие частичную компенсацию и верификацию процента выполнения задачи, — что делает подход применимым не во всех сценариях. Приведённые примеры носят иллюстративный характер и не являются исчерпывающими или окончательными. Конкретный тип расширения определяется базовым протоколом. Мы рассчитываем, что эти примеры дадут разработчикам первоначальные идеи для дальнейшего исследования этого направления.

7. Заключение

Значительная часть будущей глобальной экономики, вероятно, будет опосредована миллионами специализированных ИИ-агентов, встроенных в компании, цепочки поставок и государственные службы. Однако нынешняя парадигма ситуативного, основанного на эвристике делегирования недостаточна для обеспечения этой трансформации.

Чтобы безопасно раскрыть потенциал агентной сети, необходимо принять динамическую и адаптивную систему интеллектуального делегирования, ставящую во главу угла верифицируемую надёжность и чёткую подотчётность — наравне с вычислительной эффективностью. Когда ИИ-агент сталкивается со сложной задачей, для выполнения которой требуются компетенции и ресурсы, выходящие за пределы его собственных

возможностей, он должен принять роль делегатора в рамках системы интеллектуального делегирования задач. Этот делегатор декомпозирует задачу на управляемые подкомпоненты, которые можно сопоставить с компетенциями доступных агентов на рынке на том уровне гранулярности, который обеспечивает чёткую проверяемость. Распределение задач определяется на основе поступающих заявок и ряда ключевых факторов: доверия, репутации, мониторинга динамических операционных состояний, стоимости, эффективности и других. Задачи с высокой критичностью и низкой обратимостью могут требовать дополнительных структурированных разрешений и многоуровневых согласований с чёткой структурой подотчётности — осуществляемых под надзором человека в соответствии с применимыми институциональными рамками.

В масштабе Интернета безопасность и подотчётность не могут быть второстепенным вопросом. Они должны быть встроены в операционные принципы виртуальных рынков агентов и служить центральными организующими принципами агентной сети. Внедряя безопасность на уровне протоколов делегирования, мы стремимся предотвратить накопление ошибок и каскадные отказы, а также сформировать способность оперативно реагировать на рассогласованное или злонамеренное поведение ИИ-агентов и людей — ограничивая негативные последствия.

В конечном счёте мы предлагаем смену парадигмы: от преимущественно неконтролируемой автоматизации к верифицируемому интеллектуальному делегированию — подходу, который позволит нам безопасно масштабироваться до будущих автономных агентных систем, сохраняя при этом их тесную связь с человеческими намерениями и общественными нормами.

Список источников

В настоящем переводе список источников не приводится. Полный библиографический список содержится в оригинальной статье:

Tomasev, N., Franklin, M., & Osindero, S. (2026). Intelligent AI Delegation. Google DeepMind. https://arxiv.org/abs/2602.11865v1

Для получения ссылок на все цитируемые источники обращайтесь к оригинальному тексту по указанному адресу.