

The practical value of the study lies in the comprehensive substantiation of a hybrid approach to spatial filtering, which combines high-precision lattice initialization by the robust RAIS-LCMV algorithm in a stationary mode with the subsequent transition to low-cost gradient methods in dynamics. The effectiveness of this approach for eliminating the effect of signal self-suppression in conditions of multicomponent electronic countermeasures is mathematically proved.

References

1. Zvonarev V. V., Pitrin A. V., Popov A. S. Metodika rascheta koehfficienta napravlennogo dejstviya adaptivnoj antennoj reshetki pri nalichii pomeh // Izv. vuzov. Priborostroenie, 2025. – V. 68. – No 3. – Pp. 198–206.
2. An Adaptive Five-Element Array Antenna Based on Miniaturised Polarisation-Insensitive Absorptive Metamaterial for Anti-Jamming Applications / G. Wang, X. Tang, F. Huang [et al.] // IET Microwaves Antennas & Propagation, 2025. – Vol. 19. – No. 1. – P. e70056.
3. Shima, K. Data-Aided SMI Algorithm Using Common Correlation Matrix for Adaptive Array Interference Suppression / K. Shima, K. Maruta, Ch. Ju. Ahn // IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences, 2021. – Vol. E104.A. – No 2. – Pp. 404–411.
4. Kashchenko I. E. Regularizaciya RLS-algoritma dlya adaptivnyh sistem vvoda cifrovyyh predyskazhenij / I. E. Kashchenko, A. P. Pavlov // Radiotekhnika, 2024. – V. 88. – No 1. – Pp. 141–148.
5. An efficient normalized LMS algorithm / A. Zerguine, Ja. Ahmad, M. Moinuddin [et al.] // Nonlinear Dynamics, 2022. – Vol. 110. – No. 4. – Pp. 3561–3579.
6. Praneet R. J., Kumar C. A., Rajendra N. A novel sign variable step size LMS (SiVSS-LMS) algorithm for adaptive beamforming // CSIT, 2020. – 8 (4). – Pp. 377–384.
7. Bojko I. A., Glushankov E. I., Lyalina A. Zh. Algoritmy adaptivnoj obrabotki signalov v geostacionarnyyh sistemah // Trudy uchebnyh zavedenij svyazi, 2025. – V. 11. – No 5. – Pp. 74–83.
8. Wnuk, M. Adaptive Antenna Array Control Algorithm in Radiocommunication Systems // Algorithms, 2024. – Vol. 17. – No 2. – P. 81.
9. Glushankov E. I., Carik I. V., Carik V. I. Prostranstvenno-raspredeleonnaya tehnologiya adaptivnogo formirovaniya diagrammy napravlenosti antennoj reshetki // Nauchnye izvestiya, 2022. – No 29. – Pp. 165–170.
10. Chistyakov V. A. Algoritmy adaptivnoj fil'tracii pomeh v cifrovyyh antennyh reshetkah sputnikovoy svyazi // Trudy MAI, 2019. – V.105. – Pp. 100–113.
11. Meng, Zh. Robust widely linear beamforming using estimation of extended covariance matrix and steering vector / Zh. Meng, W. Zhou, S. Gazor // Eurasip Journal on Wireless Communications and Networking, 2020. – Vol. 2020. – No 1. – Pp. 1–20.
12. Guo X., Chu L., Li B. Robust Adaptive LCMV Beamformer Based on an Iterative Suboptimal Solution // Radioengineering, 2015. – Vol. 24. – No 2. – P. 572–581.



ПРИНЦИПЫ ПОСТРОЕНИЯ ПАЙПЛАЙНОВ С ДИНАМИЧЕСКОЙ АРХИТЕКТУРОЙ И АГЕНТНЫМ УПРАВЛЕНИЕМ

Лукашенко В. И.¹, Изотов Д. Ю.²

DOI:10.21681/3034-4050-2026-3-21-28

Ключевые слова: динамические пайплайны, агентное управление, адаптивная архитектура, самоорганизация, масштабируемость, декомпозиция задач.

Аннотация

Цель работы: обосновать методологический подход к построению пайплайнов с динамической архитектурой, управляемых автономными программными агентами, способными адаптироваться к изменяющимся условиям выполнения задач.

Метод исследования: метод исследования сочетает принципы модульного проектирования, теорию много-агентных систем и методы имитационного моделирования для анализа поведения динамических пайплайнов.

Результаты исследования: представлен формализованный подход к описанию жизненного цикла агентов, их взаимодействия и эволюции топологии пайплайна, что позволяет осуществлять моделирование различных сценариев функционирования системы в условиях внешних возмущений. Сформулированы научно обоснованные принципы проектирования агентно-управляемых архитектур, включая критерии выбора granularity агентов, стратегии коммуникации и механизмы принятия решений. Эти аспекты будут положены в основу дальнейшего развития фреймворков для построения адаптивных систем обработки данных.

Научная новизна: обоснован принцип построения пайплайнов с динамической архитектурой и агентным управлением, рассматриваемые как взаимодополняющие механизмы адаптации.

Введение

В современном мире, где объемы обрабатываемых данных растут экспоненциально, а требования к скорости и качеству принятия решений ужесточаются, важность гибких и адаптивных вычислительных архитектур становится очевидной. Традиционные статические пайплайны обработки данных, характеризующиеся жестко заданной последовательностью операций, все чаще оказываются неспособными эффективно реагировать на изменения входных потоков, вариативность вычислительных ресурсов и динамические требования бизнес-процессов [1, 2].

Интеграция принципов динамической архитектуры и агентного управления представляет собой перспективное направление развития распределенных систем. Динамическая архитектура позволяет пайплайну изменять свою структуру в процессе выполнения: добавлять или исключать этапы обработки, перенаправлять потоки данных, масштабировать компоненты в зависимости от нагрузки [3, 4]. Агентное управление, в свою очередь, обеспечивает децентрализованное принятие решений, где автономные программные агенты координируют

свою деятельность для достижения общих целей системы^{3,4}[5].

Для обоснования эффективности проектируемых динамических пайплайнов применяется множество методов оценки. В основном анализ проводится на основе метрик производительности, задержек, использования ресурсов и отказоустойчивости, что позволяет использовать результаты при разработке промышленных решений. Ключевой характеристикой таких систем является адаптивность – способность изменять поведение и структуру в ответ на внешние и внутренние стимулы^{5,6}[6, 7]. Адаптивность косвенно определяет живучесть системы, т.е. способность сохранять работоспособность при частичных отказах компонентов или изменении условий функционирования.

В статье [8] рассматривается метод оркестрации микросервисов на основе правил и событий. В отличие от предыдущего примера, здесь предлагается алгоритм динамического перепланирования потоков данных, используя метрики загрузки узлов, приоритеты задач и ограничения по времени. Математические модели и программная реализация позволяют анализировать

1 Лукашенко Василий Ильич, младший научный сотрудник научно-исследовательского центра Военной академии связи. Санкт-Петербург, Россия. E-mail: lukasenokvasilij@gmail.com

2 Изотов Даниил Юрьевич, научный сотрудник научно-исследовательского центра Военной академии связи. Санкт-Петербург, Россия. E-mail: daniil.izotov.1999@mail.ru

3 Wooldridge M. An Introduction to Multiagent Systems // M. Wooldridge. 2nd ed. / Chichester: Wiley, 2009. – 488 p.

4 Weiss G. Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence / G. Weiss // Cambridge: MIT Press, 2000. – 640 p.

5 Huebscher M. C. A Survey of Autonomic Computing: Degrees, Models, and Applications / M. C. Huebscher, J. A. McCann // ACM Computing Surveys, 2008. – Vol. 40. – No 3. – Article 7. – DOI: 10.1145/1380584.1380585.

6 Kephart J. O. The Vision of Autonomic Computing / J. O. Kephart, D. M. Chess // Computer, 2003. – Vol. 36. – No 1. – Pp. 41–50. – DOI: 10.1109/MC.2003.1160055.

зависимость показателей эффективности от различных параметров, таких как частота поступления задач и доступность вычислительных ресурсов. Эти методы обеспечивают точную оценку производительности системы, но требуют значительных вычислительных ресурсов и сложных моделей для анализа.

Таким образом, оценка адаптивности и эффективности динамических пайплайнов с агентным управлением является важным параметром устойчивости и, как следствие – надежности системы, и в современных условиях создания и проектирования новых распределенных архитектур является актуальным направлением исследования. Гипотеза настоящего исследования заключается в предположении, что адаптивность динамических пайплайнов с агентным управлением может быть количественно оценена с использованием марковских моделей при допущении о стационарности вероятностей переходов между состояниями агентов, а предложенный подход к расчету позволяет выявлять критические точки структуры и прогнозировать поведение системы при внешних возмущениях с приемлемой для инженерной практики точностью. В качестве инструментария исследования выступает комплекс методов, включающий теорию многоагентных систем для описания поведения автономных сущностей, аппарат цепей Маркова для формализации переходов между состояниями, а также среду имитационного моделирования на платформе Python с использованием библиотек numpy и matplotlib для реализации вычислительных экспериментов и визуализации полученных зависимостей. Этому вопросу и посвящена данная статья.

Анализ предмета исследования и постановка задачи

Анализ существующих подходов к построению адаптивных пайплайнов показывает, что они в основном основаны на предопределенных правилах переконфигурации или реактивных механизмах масштабирования. Основным их недостатком является то, что они не учитывают семантическую связность между этапами обработки, т.е. насколько изменения в одном компоненте влияют на эффективность работы всей системы, и как адаптировать поведение агентов при изменении контекста выполнения задач.

Основополагающей частью работы является математическая модель оценки адаптивности сложных систем с децентрализованным управлением [9]:

$$S = \{A_1, A_2, \dots, A_n\}, n \in N, \quad (1)$$

где S – система, а каждый агент A_i характеризуется:

- вектором состояния $x_i(t) \in R^{d_i}$;
- вектором управления $u_i(t) \in R^{m_i}$;
- входным сигналом среды $y_i(t) \in R^{p_i}$;
- функцией полезности $J_i(x_i, u_i, y_i)$.

В данной статье предложена математическая модель построения пайплайнов, которая учитывает влияние динамических факторов на координацию между агентами системы. Особенностью модели является учет стохастического характера поступления задач и вариативности времени выполнения операций, что формализуется через случайный процесс поступления заявок с интенсивностью $\lambda(t)$ и случайное время обслуживания $\tau_i e(\mu_i)$ для каждого агента.

На основе предложенной формализации можно выделить два фундаментальных принципа построения динамических пайплайнов с агентным управлением. Первый принцип – декомпозиции на автономные агенты – заключается в разбиении вычислительного процесса на множество слабосвязанных сущностей (агентов), каждая из которых обладает собственным вектором состояния $x_i(t)$ и функцией полезности J_i . Это обеспечивает модульность, масштабируемость и возможность децентрализованной адаптации. Второй принцип – вероятностной связанности – утверждает, что взаимодействие между агентами не является детерминированным, а характеризуется вероятностями успешного перехода p_{ij} . Данный принцип позволяет учитывать стохастическую природу внешней среды (сбои, задержки, перегрузки) при оценке устойчивости всей цепочки обработки.

В условиях воздействия внешних возмущений на элементы вычислительной системы надежность узлов и каналов связи не является абсолютной: задержки могут происходить как из-за перегрузки ресурсов, так и под воздействием сетевых проблем. Поэтому каждый компонент пайплайна оценивается вероятностью его нормального функционирования в заданный момент времени $p_i(t) = P$ (агент A_i работает исправно в момент t).

Физически это означает вероятность того, насколько стабильно данный агент выполняет свои функции при текущей нагрузке. Вероятностная мера надежности может выражаться как вероятность завершения задачи в срок $P(T_i \leq T_{\text{дедлайн}})$ или другими показателями, имеющими иной физический смысл.

Использование равновесных состояний в поведении агентов позволило применить марковские модели, что значительно упростило процесс вычислений и повысило оперативность оценки

адаптивности пайплайна. Это стало возможным благодаря тому, что наш подход позволяет рассчитывать вероятность успешного завершения цепочки обработки, где каждый переход между состояниями имеет определенную вероятность. Формально, пусть $S = \{s_1, s_2, \dots, s_m\}$ – множества состояний пайплайна, а $P = [p_{ij}]$ – матрица переходных вероятностей, где p_{ij} – переход состояний. Тогда вектор стационарных вероятностей $\pi = \pi P$ при условии $\sum_i \pi_i = 1$. Формула полной вероятности используется для вычисления вероятности успешного исхода в серии зависимых событий, где каждый этап имеет два возможных результата: успешное завершение с вероятностью p_i или необходимость перепланирования с вероятностью $q_i = 1 - p_i$ [10]. Вероятность успешного прохождения всей цепочки из k этапов вычисляется как $P_{\text{успешн}} = \prod_i p_i$, а математическое ожидание числа попыток составляет $E[N] = \sum_{i=1}^k \frac{1}{p_i}$. Такой метод обеспечивает возможность отслеживания состояния пайплайна в реальном времени и предоставляет более простой и эффективный инструмент для анализа системы.

Поэтому в работе рассматривается обобщенное понятие адаптивной связности как взаимосвязи между агентами, образующими динамически перестраиваемую и самоорганизующуюся структуру пайплайна. Такой подход расчета адаптивности обусловлен необходимостью оперативного восстановления цепочек обработки в случае их нарушения на критических участках, вызванного внешними возмущениями или внутренними сбоями архитектуры [11, 12]. Данный способ позволяет не только вычислить адаптивность, т.е. способность к реконфигурации, но и оценить устойчивость восстановленных цепочек и вероятность успешной обработки информационных потоков. Кроме того, анализ адаптивной связности помогает выявлять узкие места системы, прогнозировать поведение при различных сценариях нагрузки и разрабатывать эффективные методы ее оптимизации [13, 14]. Таким образом, модель является многофункциональной, определяющей адаптивность, надежность, устойчивость и работоспособность сложных пайплайнов в зависимости от физического смысла показателя модели.

На платформе Python с использованием фреймворка симуляции агентов был разработан программный код, который позволяет рассчитывать адаптивность пайплайна для заданного числа агентов n и различных значений k , представляющих минимальное количество успешных переходов для завершения задачи. Программа также вычисляет среднее время выполнения для всех сценариев, которое меньше

или больше определенного значения t . Итоговый результат представляет собой график зависимости адаптивности от параметров системы, что обеспечивает наглядность и удобство анализа.

Таким образом, в работе предложен научный подход к анализу адаптивности пайплайнов с использованием марковских моделей и имитационного моделирования в программной среде Python. Этот подход позволяет упростить расчеты и повысить оперативность оценки адаптивности в условиях динамического поведения агентов.

В рамках данной работы анализируется вычисление адаптивности для динамически перестраиваемого и равновесного пайплайна. В таких условиях применима модель цепей Маркова, поскольку вероятности переходов между состояниями агентов считаются стационарными, а каждый элемент системы может взаимодействовать с другими через определенные протоколы.

Формула полной вероятности используется для вычисления вероятности успешного исхода в серии зависимых событий, где каждый этап имеет возможные результаты:

$$P(A) = \sum_{i=1}^n P(H_i) \cdot P(A|H_i), \quad (2)$$

где $P(A)$ – вероятность успешного завершения пайплайна; $P(H_i)$ – вероятность нахождения системы в состоянии i ; $P(A|H_i)$ – условная вероятность успеха при условии состояния i ; n – количество возможных состояний системы.

Вероятность перехода между состояниями можно вычислить по формуле:

$$P_{ij} = \frac{\lambda_{ij}}{\sum_k \lambda_{ik}}, \quad (3)$$

где λ_{ij} – интенсивность перехода из состояния i в состояние j .

Представленная математическая модель позволяет вычислить вероятность того, что из n последовательных этапов обработки будет достигнуто успешное завершение при вероятностях переходов P_{ij} , т.е. определить устойчивость цепочки агентов в отдельности. Для определения среднего значения адаптивности пайплайна будем использовать ее усредненное значение по множеству сценариев, тогда формализованную постановку задачи можно записать в следующем виде:

$$\dot{A} = \frac{1}{M} \sum_{m=1}^M \left(\prod_{t=1}^{T_m} P_{\text{success}}^{m,t} \right). \quad (4)$$

Ограничения и условия: $0 \leq P \leq 1$, $n > 0$, $0 < k \leq n$, T_m – длительность сценария m .

Полученная математическая модель позволяет вычислить усредненное значение адаптивности, если известна архитектура пайплайна и вероятностные характеристики переходов между состояниями агентов.

Анализ адаптивности пайплайнов

Анализ адаптивности пайплайнов с агентным управлением основан на использовании марковских моделей, что позволяет упростить расчеты в силу математической стройности модели, а следовательно, повысить оперативность оценки в условиях динамического поведения агентов.

Основным предположением данного подхода является наличие стационарных вероятностей переходов между состояниями агентов [15, 16]. Такое ограничение имеет преимущество в оценке адаптивности пайплайнов любой размерности, однако ограничивает рамки применимости модели, которые связаны с типом обрабатываемых данных, требованиями к задержкам и видам внешних возмущений. В таких условиях адаптивность пайплайна определяется как взаимосвязь между агентами, формирующая динамически перестраиваемую и самоорганизующуюся структуру обработки.

На платформе Python был разработан программный код, который позволяет рассчитывать адаптивность пайплайна для заданного числа агентов n и различных значений k , представляющих минимальное количество успешных переходов для завершения задачи. Программа также вычисляет среднее время выполнения для всех сценариев, которое меньше или больше определенного значения t . Итоговый результат представляет собой график зависимости адаптивности от параметров системы, что обеспечивает наглядность и удобство анализа.

Алгоритм расчета адаптивности пайплайна с агентным управлением приведен на рисунке 1. Он включает следующие процедуры:

1. Ввод параметров: запрос количества агентов n , минимального количества успешных переходов k , матрицы вероятностей переходов P_{ij} .
2. Расчет адаптивности: для каждого сценария нагрузки и каждого значения k вычисляется вероятность успешного завершения с использованием марковской модели. Далее определяются средние значения адаптивности для различных диапазонов времени выполнения.
3. Визуализация результатов: строится график зависимости средней адаптивности от вероятности успешных переходов для различных значений k . Затем линейный график зависимости адаптивности от количества агентов для различных стратегий координации.

Для реализации вышеописанного алгоритма были написаны основные фрагменты кода на Python с использованием библиотек `numpy` для матричных вычислений и `matplotlib` для визуализации. Этот алгоритм и код позволяют оперативно анализировать адаптивность пайплайнов, используя простые и понятные методы вычислений. Полученные результаты предоставляют значимые данные для оценки адаптивности, позволяя учесть стохастическую природу процессов обработки.

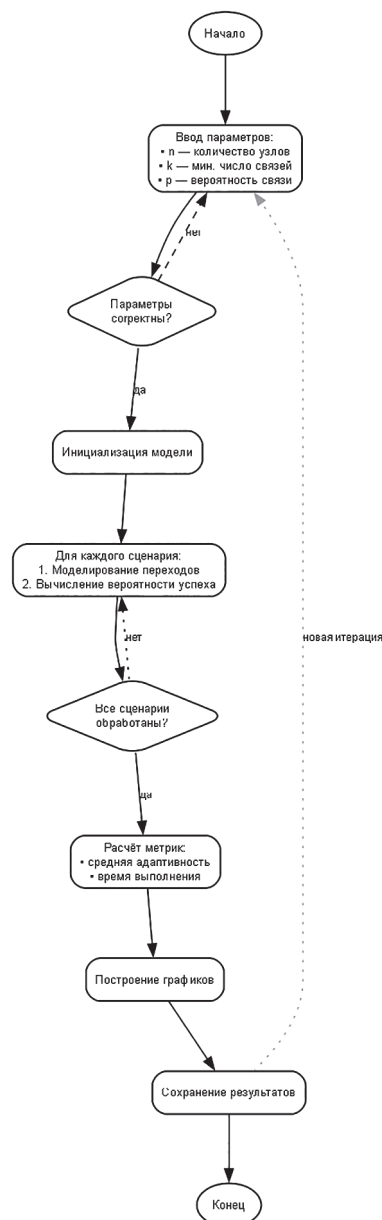


Рис. 1. Блок-схема алгоритма программы

Результаты исследований адаптивности пайплайна с десятью агентами от количества успешных переходов и значений вероятности

перехода между состояниями приведены на рисунке 2. Анализ полученных зависимостей показывает, что при вероятностном подходе к оценке показателя адаптивности фрагмента пайплайна его нормированное значение уменьшается по сравнению с оценкой по детерминированным методикам.

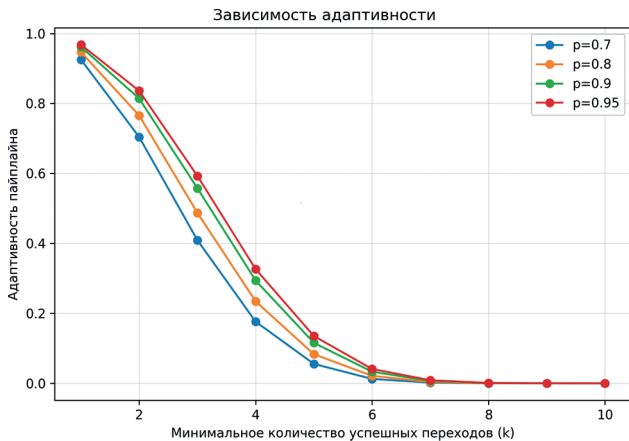


Рис. 2. График зависимости адаптивности от k для различных вероятностей переходов

Анализ полученных зависимостей показывает, что при вероятностном подходе к оценке показателя адаптивности фрагмента пайплайна его нормированное значение уменьшается по сравнению с оценкой по классическим методикам. Это связано с тем, что при наличии перехода между состояниями вес связи не равен единице, как в детерминированном подходе, а имеет вероятностный характер (меньше или равен единице), так как отражает степень воздействия внешних факторов на агентов пайплайна, что отражает реальную действительность.

С уменьшением значения вероятности успешного перехода между состояниями показатель адаптивности уменьшается, т.е. снижается устойчивость системы по причине внешних возмущений на ее элементы.

При увеличении количества агентов в пайплайне, но ограничении на число успешных переходов между ними (например, по причине ограничений пропускной способности или других факторов) даже при высокой вероятности переходов показатель адаптивности значительно снижается.

Также необходимо отметить, что в полностью связанной структуре пайплайна из 10 агентов показатель адаптивности уменьшился на 36 % по сравнению с детерминированной оценкой. Этот эффект связан с тем, что несмотря на практически гарантированные переходы в сложных системах с увеличением числа агентов возрастает

количество возможных путей отказа. Поэтому даже небольшие отклонения в вероятностях переходов оказывают существенное влияние на общую адаптивность. Этот результат доказывает тот факт, что чем сложнее структура пайплайна, тем большая надежность должна быть у каналов взаимодействия между агентами.

Полученные результаты демонстрируют важность учета надежности отдельных переходов в сложных системах. Анализ качества межагентных зависимостей позволит выявлять критически слабые места пайплайна, оптимизировать его структуру и на конечном этапе разрабатывать стратегии повышения устойчивости системы.

Вторая часть программы позволяет проводить анализ адаптивности пайплайна в зависимости от количества переходов, превышающих или недостающих до требуемого (заданного) значения. Этот этап особенно важен для аналитической работы, так как позволяет не только определить точную вероятность успешного завершения, но и рассчитать средние значения вероятностей для меньших и больших значений переходов. Такая детализация дает возможность формировать более точные и обоснованные выводы об архитектуре системы обработки. Полученные диаграммы адаптивности пайплайна на рисунке 3 предоставляют значимые данные для оценки, учитывая стохастическую природу процессов.

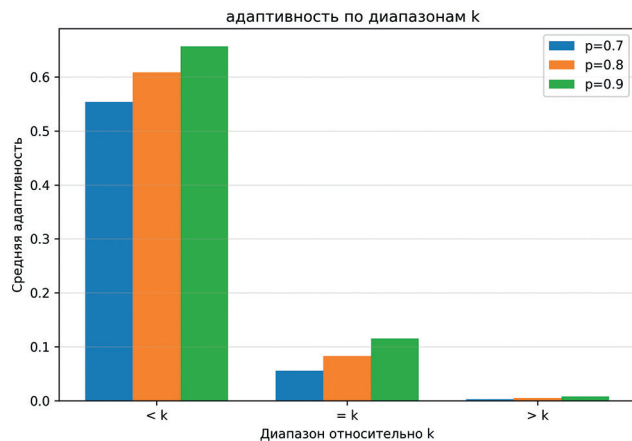


Рис. 3. Средняя адаптивность для менее, точно и более k успешных переходов

Анализ свидетельствует о том, что адаптивность не является фиксированным (детерминированным) значением, а представляет собой стохастический процесс, зависящий от множества факторов, таких как вероятность внешних возмущений, количество агентов и характер связей между ними. Для аналитической работы данная диаграмма более значима, так как

по ней определяется точная вероятность, а также средняя для меньших и больших значений. Это позволяет сделать более точное оценочное заключение по архитектуре системы обработки и способствует глубокому пониманию поведения сложных пайплайнов в различных условиях, что важно для проектирования устойчивых и эффективных структур.

Заключение

Разработанная математическая модель оценки и анализа адаптивности пайплайнов с динамической архитектурой и агентным управлением отражает значительный потенциал для оценки устойчивости и работоспособности систем в условиях внешних возмущений. В рамках настоящего исследования был предложен методологический подход, основанный на использовании марковских моделей, который позволяет упростить расчеты и повысить оперативность оценки адаптивности при наличии стационарных вероятностей переходов между состояниями агентов.

Основной целью исследования являлось создание простой, но адекватной модели оценки адаптивности и разработка программного обеспечения для её расчёта. Реализованная модель базируется на сформулированных в работе принципах: декомпозиции на автономные агенты, обеспечивающих необходимую гибкость архитектуры, и вероятностной связанности, позволяющей учитывать влияние внешних возмущений на надежность взаимодействий. Необходимо заметить, что предложенная методология оценки адаптивности пайплайна позволяет оценить устойчивость системы в целом в рамках способности к реконфигурации при внешних воздействиях на элементы, но она не дает информации о других свойствах системы, например, таких как семантическая согласованность данных, качество промежуточных результатов, энергоэффективность вычислений и т.п.

Очевидно, что в реальных условиях вероятности переходов между состояниями не могут быть одинаковыми из-за множества факторов, таких как различия в характеристиках оборудования, разнообразие во внешних условиях и типах возмущений. Однако принятое допущение о стационарных вероятностях переходов позволило разработать базовый алгоритм и программную реализацию, которая служит фундаментом для дальнейших исследований и расширений.

Полученные результаты исследований могут быть положены в основу научного обоснования и сравнительного анализа адаптивности пайплайнов различной сложности, в том числе распределенных, а также для проведения ситуационного анализа устойчивости систем к внешним воздействиям и оперативной оценки деформации цепочек обработки. Аналитическая работа с результатами оценки системных показателей позволит определять слабые звенья пайплайна, задавать научно обоснованные требования к вероятностным показателям элементов и формулировать точные оценочные заключения по адаптивности системы в различных условиях функционирования, что важно для проектирования устойчивых и эффективных архитектур.

Перспективами дальнейшего развития является совершенствование модели и программного обеспечения для оценки не только структурной, но и семантической адаптивности. Дальнейшая разработка принципов агентного управления, в частности, переход от стационарных к динамическим вероятностям переходов и внедрение механизмов обучения агентов, позволит учитывать более сложные сценарии функционирования пайплайнов. Это обеспечит более точные прогнозы производительности и надежности систем в реальных условиях эксплуатации, что имеет важное значение в условиях быстро меняющейся технологической среды и растущих требований к адаптивным системам обработки данных.

Литература

1. Вернер Х., Ламп Дж. Паттерны интеграции корпоративных приложений / Х. Вернер, Дж. Ламп. – М.: Вильямс, 2020. – 672 с.
2. Richardson C. Microservices Patterns: With examples in Java / C. Richardson // Shelter Island: Manning Publications, 2018. – P. 520.
3. Balalaie A. Microservices Architecture Enables DevOps: Migration to a Cloud-Native Architecture / A. Balalaie, A. Heydarnoori, P. Jamshidi // IEEE Software, 2016. – Vol. 33. – No 3. – Pp. 42–52. – DOI: 10.1109/MS.2016.64.
4. Soldani J. The Pains and Gains of Microservices: A Systematic Grey Literature Review / J. Soldani, D. A. Tamburri, W.-J. van den Heuvel // Journal of Systems and Software, 2018. – Vol. 146. – Pp. 215–232. – DOI: 10.1016/j.jss.2018.09.082.
5. Goonatilake S. T. Past, Present and Future Trends in Multi-Agent System Technology / S. T. Goonatilake, B. Hettige // Journal Européen des Systèmes Automatisés, 2022. – Vol. 55. – No. 6. – Pp. 723–739.
6. Smith E. LILLIE: Information extraction and database integration using linguistics and learning-based algorithms / E. Smith, D. Papadopoulos, M. Braschler, K. Stockinger // Information Systems, 2022. – Vol. 105. – DOI: 10.1016/j.is.2021.101938.

7. Голиков В. А. Исследование ключевых архитектурных паттернов микросервисов: теоретический анализ их роли в построении отказоустойчивой информационной системы / В. А. Голиков, М. В. Янаева, Н. Т. Рудник. – СПб: Информационно-управляющие системы, 2023. – № 4. – С. 28–37.
8. Абросимов Л. И. Методы проектирования и анализа производительности распределенных вычислительных сетей / Л. И. Абросимов. – М: Университетская книга, 2023. – 256 с.
9. Qiang S. The Nonlinear Mathematical Modeling and Optimization of Distributed Control in Complex Systems / S. Qiang // Journal of Advances in Engineering and Technology, 2025. – Vol. 2. – Pp. 34–42. – DOI: 10.62177/jaet.v2i1.152.
10. Wang Y., et al. A multi-level action coupling reinforcement learning approach for online two-stage flexible assembly flow shop scheduling // Journal of Manufacturing Systems, 2024. – Vol. 76. – Pp. 351–370.
11. Liu C. A Survey on Workflow Management and Scheduling in Cloud Computing / C. Liu, C. Yang, X. Fan [et al.] // IEEE Transactions on Services Computing, 2022. – Vol. 15. – No 4. – Pp. 2178–2197. – DOI: 10.1109/TSC.2020.3047462.
12. Calheiros R. N. CloudSim: A Toolkit for Modeling and Simulation of Cloud Computing Environments and Evaluation of Resource Provisioning Algorithms / R. N. Calheiros, R. Ranjan, A. Beloglazov [et al.] // Software: Practice and Experience, 2011. – Vol. 41. – No 1. – Pp. 23–50. – DOI: 10.1002/spe.995.
13. Sun S. Integrated data-driven topology reconstruction and risk-aware reconfiguration for resilient power distribution systems under incomplete observability / S. Sun, N. Li, L. Zhang, D. Zhao, D. Lun, L. Feng // Scientific Reports, 2025. – Vol. 15. – P. 30661. – DOI: 10.1038/s41598-025-15440-8.
14. Sahu A. ARM-IRL: Adaptive Resilience Metric Quantification Using Inverse Reinforcement Learning / A. Sahu, V. Venkataramanan, R. Macwan // AI, 2025. – Vol. 6. – No 5. – P. 103. – DOI: 10.3390/ai6050103.
15. Meylahn J. M. Quantifying the Likelihood of Learning Collusive Strategy Equilibria / J. M. Meylahn // Chaos: An Interdisciplinary Journal of Nonlinear Science, 2025. – Vol. 35. – No 8. – P. 083106. – DOI: 10.1063/5.0281443.
16. Шелухин О. И. Модели и методы управления в многоагентных системах / О. И. Шелухин, А. В. Осин. – М.: ФИЗМАТЛИТ, 2018. – 288 с.

PRINCIPLES OF BUILDING PIPELINES WITH DYNAMIC ARCHITECTURE AND AGENT-BASED MANAGEMENT

Lukashenok V. I.⁷, Izotov D. Yu.⁸

Keywords: dynamic pipelines, agent-based management, adaptive architecture, self-organization, scalability, task decomposition.

Abstract

The purpose of the work is to substantiate a methodological approach to building pipelines with dynamic architecture, controlled by autonomous software agents capable of adapting to changing conditions of task performance.

Research method: the research method combines the principles of modular design, the theory of multi-agent systems, and simulation modeling methods to analyze the behavior of dynamic pipelines.

Results of the study: a formalized approach to the description of the life cycle of agents, their interaction and evolution of the pipeline topology is presented, which makes it possible to simulate various scenarios of the system's functioning under conditions of external disturbances. Scientifically based principles for the design of agent-driven architecture have been formulated, including criteria for selecting agent granularity, communication strategies, and decision-making mechanisms. These aspects will form the basis for the further development of frameworks for building adaptive data processing systems.

Scientific novelty: the principle of building pipelines with dynamic architecture and agent-based control, considered as complementary adaptation mechanisms, is substantiated.

References

1. Verner H., Lamp Dzh. Patterny integracii korporativnyh prilozhenij / H. Verner, Dzh. Lamp // М.: Vil'yams, 2020. – P. 672.
2. Richardson C. Microservices Patterns: With examples in Java / C. Richardson // Shelter Island: Manning Publications, 2018. – P. 520.
3. Balalaie A. Microservices Architecture Enables DevOps: Migration to a Cloud-Native Architecture / A. Balalaie, A. Heydarnoori, P. Jamshidi // IEEE Software, 2016. – Vol. 33. – No 3. – Pp. 42–52. – DOI: 10.1109/MS.2016.64.
4. Soldani J. The Pains and Gains of Microservices: A Systematic Grey Literature Review / J. Soldani, D. A. Tamburri, W.-J. van den Heuvel // Journal of Systems and Software, 2018. – Vol. 146. – Pp. 215–232. – DOI: 10.1016/j.jss.2018.09.082.
5. Goonatilleke S. T. Past, Present and Future Trends in Multi-Agent System Technology / S. T. Goonatilleke, B. Hettige // Journal Européen des Systèmes Automatisés, 2022. – Vol. 55. – No. 6. – Pp. 723–739.

⁷ Vasily I. Lukashenok, Junior Researcher, Research Center, Military Academy of Communications. St. Petersburg, Russia. E-mail: lukashenokvasilij@gmail.com

⁸ Danil Yu. Izotov, Researcher, Research Center, Military Academy of Communications. St. Petersburg, Russia. E-mail: daniil.izotov.1999@mail.ru