

# ПОВЫШЕНИЕ ПРОИЗВОДИТЕЛЬНОСТИ МУЛЬТИАГЕНТНОЙ СИСТЕМЫ ПРЕДСТАВЛЕНИЯ И ОБРАБОТКИ ЗНАНИЙ

*Е. И. Зайцев*<sup>1</sup>, *Е. В. Нурматова*<sup>1,\*</sup>, *В. В. Гусев*<sup>2</sup>

<sup>1</sup> МИРЭА — Российский технологический университет, Москва

<sup>2</sup> Институт физики высоких энергий им. А. А. Логунова  
Национального исследовательского центра «Курчатовский институт»,  
Протвино, Россия

Рассматриваются методы повышения производительности мультиагентной системы представления и обработки знаний (МСПОЗ). Описан подход к разработке системного и прикладного программного обеспечения МСПОЗ. Представлена схема управления информационно-вычислительными ресурсами, смоделировано влияние фактического времени работы микросервисов системы, учитывающего скорость поступления и обработки запросов в очереди без ограничения на их количество, в условиях возможного масштабирования системы.

The paper deals with the methods of improving the performance of multi-agent knowledge representation and processing system (MAKRPS). The approach to the development of system and application software of MAKRPS is described. The scheme of management of information-computing resources is considered, and the influence of actual time of work of microservices of the system, taking into account speed of arrival and processing of queries in a queue without restriction on their quantity, in conditions of possible scaling of the system is modeled.

PACS: 89.20.Ff; 07.05.Tp

## ВВЕДЕНИЕ

Мультиагентная система представления и обработки знаний (МСПОЗ) — это распределенная система искусственного интеллекта, состоящая из взаимодействующих прикладных (интеллектуальных) и системных программных агентов [1–4]. Для решения сложных задач прикладные программные агенты используют распределенную базу знаний (РБЗ) и методы машинного обучения, которые позволяют агентам выбирать лучшие стратегии. Стратегия определяет действия агента в каждый момент времени в соответствии с воспринимаемым состоянием окружающей среды.

---

\* E-mail: nurmatova@mirea.ru

Интеллектуальные агенты обучаются и действуют рационально в условиях неполноты и нечеткости поступающей информации [5]. Для обучения прикладных программных агентов используется подход, ориентированный на целенаправленное обучение посредством взаимодействия. В данном подходе, называемом обучением с подкреплением, применяется поиск методом проб и ошибок, при котором прикладные программные агенты получают за свои действия отложенные вознаграждения от среды [6, 7].

В МСПОЗ реализованы собственные системные API и системные библиотеки (LibOS), которые ассоциированы с системными программными агентами. Определенные в LibOS специализированные программные модули являются более производительными и требуют меньше ресурсов по сравнению с системными программными модулями общего назначения. Системные программные агенты осуществляют мониторинг и, используя сервисы экзоядра операционной системы, эффективно управляют информационно-вычислительными ресурсами МСПОЗ.

В МСПОЗ используются событийно-управляемые микросервисы EDM (Event-Driven Microservices), которые реализованы как локальные процессы, создаваемые и контролируемые операционной системой вычислительного узла, а также как отдельные сервисы на платформе PAAS (Platform As A Service). Системные программные агенты и микросервисы взаимодействуют посредством событий, определенных в потоках событий.

## **МЕТОДЫ ПОВЫШЕНИЯ ПРОИЗВОДИТЕЛЬНОСТИ МСПОЗ**

Решение сложной проблемы в МСПОЗ осуществляется путем декомпозиции этой проблемы на подзадачи, которые параллельно решают прикладные (реактивные и когнитивные) программные агенты. Эти интеллектуальные агенты взаимодействуют между собой, а также с системными программными агентами, входящими в состав LibOS.

Прикладные программные агенты обучаются через серию поощрений и наказаний. Действия интеллектуальных агентов переводят окружающую среду в новое состояние. Среда возвращает агентам следующее состояние и вознаграждение. Цикл «состояние → действие → вознаграждение» повторяется до тех пор, пока проблема не будет решена. Для обучения прикладных программных агентов в МСПОЗ используется адаптированный вариант алгоритма Actor-Critic [7].

Высокая производительность МСПОЗ достигается за счет оптимизации логической структуры РБЗ и правильного распределения агентов по вычислительным узлам. РБЗ разбивается на кластеры с учетом ограничений на их размерность, на тип используемой системы хранения, на степень семантической близости логических записей, включаемых в кластеры [8]. Прикладные программные агенты каждого узла МСПОЗ

распределяются по группам на основе матриц совместимости и включения [4].

В системе используются микросервисы, которые с целью повышения производительности и надежности дублируются на разных узлах. Системные программные агенты управляют доступом к микросервисам и распределяют вычислительную нагрузку на основе данных, предоставляемых агентами-мониторами.

Система может быть масштабируема относительно ее размера. В любой момент может потребоваться добавить больше пользователей и ресурсов в систему без заметной потери ее производительности. В этой ситуации можно столкнуться с проблемой ограничения централизованных услуг, а именно сервер или их группа может стать узким местом, если потребуются обрабатывать все большее количество запросов. Ресурсные параметры микросервисов многоагентной системы, оказывающие влияние на ее производительность, могут быть формально проанализированы с использованием теории очередей, в соответствии с которой выполняются запросы/транзакции соответствующим микросервисом. Введем упрощающее предположение, что процесс, находящийся в состоянии готовности обработать запрос, извлекает его из очереди, выполняет обработку и выдает ответ. Также будем предполагать, что очередь бесконечна, а именно нет ограничений на количество запросов, принятых для дальнейшей обработки.

Предположим, что  $\omega$  — скорость поступления запросов (в секунду), а  $\tau$  — скорость обработки запросов, точнее, количество успешно завершенных запросов (в секунду). Тогда можно определить временное отношение  $\varphi_k$ , если в МСПОЗ находится  $k$  запросов:

$$\varphi_k = \left(1 - \frac{\omega}{\tau}\right) \left(\frac{\omega}{\tau}\right)^k. \quad (1)$$

Будем считать, что микросервис системы использован, если известно время  $Q$  занятости системы. Очевидно, что

$$Q = \sum_{k \geq 0} \varphi_k = 1 - \varphi_0 = \frac{\omega}{\tau} \implies \varphi_k = (1 - Q)Q^k. \quad (2)$$

Далее вычислим  $T^{\text{med}}$  — среднее количество запросов в МСПОЗ, выполнив несложные преобразования:

$$T^{\text{med}} = \sum_{k \geq 0} k \varphi_k = \sum_{k \geq 0} k (1 - Q) Q^k (1 - Q) \sum_{k \geq 0} k Q^k = \frac{(1 - Q)Q}{(1 - Q)^2}. \quad (3)$$

Имеем

$$T^{\text{med}} = \frac{Q}{1 - Q}. \quad (4)$$

В нашем случае важно вычислить время отклика (RSP) системы, т. е. время, за которое сервис обрабатает запрос, включая время ожидания

запуска транзакции. Для этого используется средняя пропускная способность  $V$ , при определении величины которой учитывается занятость сервера во время обработки запроса, а также пропускная способность запросов  $\tau$  в секунду в течение общего времени  $Q$ :

$$V = Q\tau + (1 - Q) \cdot 0 = \frac{\omega}{\tau} \tau = \omega. \quad (5)$$

Первое слагаемое определяет пропускную способность при работе сервера, второе — во время остановки сервера, поэтому скорость обработки услуги — нулевая.

Далее можем вычислить время отклика системы [9]:

$$\text{RSP} = \frac{T^{\text{serv}}}{1 - Q} \implies \frac{\text{RSP}}{T^{\text{serv}}} = \frac{1}{1 - Q}, \quad (6)$$

где  $T^{\text{serv}}$  — фактическое время работы сервиса системы.

Соотношение времени ответ-обслуживание, близкое к единице, означает, что запрос обработан мгновенно, на максимальной скорости. Но если время занятости системы становится ближе к единице, то соотношение быстро увеличивается до высоких значений, что говорит о возможной ее остановке. По данной простой модели можно заключить, что для повышения производительности системы сокращаем время фактического времени работы сервиса системы  $T^{\text{serv}}$ .

## ЗАКЛЮЧЕНИЕ

В данной работе рассматриваются методы повышения производительности мультиагентной системы представления и обработки знаний. Описан подход к разработке высокоэффективной МСПОЗ и реализации ее системного и прикладного программного обеспечения. Рассмотрена несложная модель, показывающая обратно пропорциональную зависимость времени отклика системы от фактического времени работы сервиса системы с учетом времени занятости сервиса, среднего количества принятых и обрабатываемых запросов.

## СПИСОК ЛИТЕРАТУРЫ

1. *Baranauskas R., Janaviciute A., Jasinevicius R., Jukavicius V.* On Multi-Agent Systems Intellectics // Inf. Technol. Control. 2015. V. 1. P. 112–121.
2. *Zaytsev E. I.* The Distributed Intelligent Learning System Based on Cognitive and Reactive Software Agents // 4th Intern. Conf. "Distributed Computing and Grid-Technologies in Science and Education". Dubna, 2010. P. 264–267.
3. *Zaytsev E. I., Khalabiya R. F., Stepanova I. V., Bunina L. V.* Multi-Agent System of Knowledge Representation and Processing // Proc. of the Fourth Intern. Sci. Conf. "Intelligent Information Technologies for Industry" (IITI'19). Springer, 2020. P. 131–141.

4. *Zaytsev E. I., Nurmatova E. V.* Approach to Knowledge Management and the Development of a Multi-Agent Knowledge Representation and Processing System // Russ. Technol. J. 2023. V. 11, No. 4. P. 16–25.
5. *Darweesh S., Shehata H.* Performance Evaluation of a Multi-Agent System Using Fuzzy Model // 1st Intern. Workshop on Deep and Representation Learning (IWDRL). Cairo, Egypt, 2018. P. 7–12.
6. *Graesser L., Keng W. L.* Foundations of Deep Reinforcement Learning. Addison-Wesley Professional, 2020. 416 p.
7. *Sutton R. S., Barto A. G.* Reinforcement Learning: An Introduction. Second ed. Cambridge, MA: MIT Press, 2018. 552 p.
8. *Nurmatova E. V., Gusev V. V., Kotliar V. V.* Analysis of the Features of the Optimal Logical Structure of Distributed Databases // Proc. of the 8th Intern. Conf. “Distributed Computing and Grid-Technologies in Science and Education”. Dubna, 2018. P. 167.
9. *Little J. D. C., Graves S. C.* Little’s Law // Building Intuition. 2008. V. 115. P. 81; doi: 10.1007/978-0-387-73699-0\_5.