

МАРКОВСКИЕ МОДЕЛИ ПРИНЯТИЯ РЕШЕНИЙ ПРИ ОБОСНОВАНИИ РЕШЕНИЙ ПОСТРОЕНИЯ КЛАСТЕРА С КОНТЕЙНЕРНОЙ ВИРТУАЛИЗАЦИЕЙ

Чудинова А. А. (аспирант ИТМО)

Научный руководитель – доктор технических наук Богатырев В. А. (ИТМО)

Введение. Принятие решений в управлении проектами подразумевает неопределенность в большинстве случаев, особенно в динамических средах, таких как системы трафика запросов требуют ответов в реальном времени и должны сбалансировать несколько критериев производительности, таких как эффективность, стоимость и качество обслуживания. Марковские модели принятия решений предлагают надежную основу для оптимизации проектных решений путем включения вероятностных переходов и многокритериальной оценки [6].

В докладе рассматривается, как Марковские модели принятия решений могут применяться для принятия решений в компьютерных системах запросов, особенно в условиях неопределенности и многокритериальной оптимизации на примере контейнерной виртуализации. Контейнеризация стала стандартным подходом к развертыванию и управлению приложениями в облачных средах благодаря своей легковесности и масштабируемости. Марковские процессы принятия решений предоставляют структурированный способ моделирования проблем принятия решений, где результаты неопределенны и зависят от вероятностных правил [3]. Они особенно полезны для оптимизации распределения ограниченных ресурсов в сетях трафика, системах связи и сервисно-ориентированных инфраструктурах [1].

Основная часть. Марковские модели принятия решений (ММПР) — это подкласс марковских процессов, в котором лица, принимающие решения, контролируют эволюцию системы в течение дискретных временных шагов. Марковский процесс принятия решений определяется следующим образом:

- **Состояния (S)** – возможные состояния системы в любой момент времени;
- **Действия (A)** – набор возможных решений, доступных в каждом состоянии;
- **Вероятности перехода (P)** – вероятность перехода из одного состояния в другое при выполнении действия;
- **Вознаграждения (R)** – выгоды или издержки, связанные с выполнением определенных действий;
- **Политика (π)** – функция, которая сопоставляет каждое состояние с действием, направляя принятие решений.

Контейнеры позволяют приложениям работать в изолированных средах с общими системными ресурсами, что делает их идеальными для динамического распределения ресурсов [1]. Однако эффективное управление трафиком запросов в контейнерных средах создает такие проблемы, как:

- Динамические колебания рабочей нагрузки — трафик запросов может значительно меняться, требуя адаптивного распределения ресурсов;
- Ограничения качества обслуживания (QoS) — приложения должны соответствовать предопределенным показателям производительности, таким как низкая задержка и высокая доступность;
- Компромиссы между стоимостью и энергопотреблением — оптимизация для экономической эффективности при сохранении производительности является многокритериальной проблемой принятия решений.

ММПР предоставляют математическую основу для моделирования систем, решающих эти проблем, позволяя принимать динамические решения на основе вероятностных переходов и вознаграждений.

В управлении трафиком запросов Марковские модели принятия решений могут моделировать состояния системы, такие как уровни перегрузки, доступность сервера и шаблоны спроса пользователей [9]. Действия могут включать в себя приоритизацию запросов, выделение дополнительных ресурсов или перенаправление трафика. Вероятности перехода учитывают стохастические факторы, такие как колебания спроса или сбои в сети, в то время как награды измеряют показатели производительности, такие как сокращение задержки или качество обслуживания [8].

Моделирование проводилось с использованием системы оркестровки контейнеров на основе Kubernetes [7], где трафик запросов моделировался на основе трассировок реальной рабочей нагрузки. В моделировании на основе Kubernetes ключевые показатели производительности (KPI) в таблице 1 (время отклика, использование ресурсов, экономическая эффективность и соответствие QoS) были собраны с использованием различных инструментов мониторинга, механизмов регистрации и методов анализа производительности. Процесс сбора данных включал несколько уровней, включая сбор метрик, регистрацию и анализ. Следует реализовать подробное описание того, как собиралась каждая метрика в таблице 1:

1. Среднее время ответа реализовано через инструменты Prometheus [histogram_quantile(0.95, rate(http_request_duration_seconds_bucket[5m]))] и Проверки журналов входящих сообщений [kubectl logs -l app=my-app --tail=100 | grep "response_time"]
2. Использование ресурсов через проверку потребления ресурсов узлами (node) и модулями (pod) [kubectl top nodes kubectl top pods] и в Prometheus [rate(container_cpu_usage_seconds_total[5m])]
3. Сравнение выделенных и использованных ресурсов: $\text{sum}(\text{container_memory_usage_bytes}) / \text{sum}(\text{container_spec_memory_limit_bytes})$
4. Определение QoS SLA (время отклика < 200 мс). Скорость соответствия запроса:

$\text{count}(\text{http_requests_total}\{\text{status}="200"\} > 180\text{ms}) / \text{count}(\text{http_requests_total}) * 100$

В ходе моделирования сравнивались три стратегии планирования:

- Традиционное циклическое планирование — равномерно распределяет запросы по доступным контейнерам без учета состояния системы [4];
- Автомасштабирование на основе пороговых значений — контейнеры масштабируются на основе предопределенных пороговых значений ЦП и памяти;
- Адаптивное планирование на основе ММПР — решения принимаются динамически на основе изученной политики ММПР.

Для циклического планирования среднее время ответа составляет 320 миллисекунд, использование ресурсов — 65%, эффективность затрат — 1.0, а соответствие QoS — 88%. На основе порогового значения среднее время ответа уменьшается до 250 миллисекунд, использование ресурсов повышается до 78%, эффективность затрат составляет 0.85, а соответствие QoS — 92%. Адаптивный на основе ММПР тип планирования имеет среднее время ответа в 180 миллисекунд, использование ресурсов — 85%, эффективность затрат — 1.2, а соответствие QoS достигает 97%.

Результаты исследования. Подход на основе ММПР сократил среднее время отклика на 28% по сравнению с планированием на основе пороговых значений и на 44% по сравнению с циклическим планированием. Более высокое использование ресурсов за счет динамической корректировки распределения контейнеров, подход ММПР достиг 85% коэффициента использования ресурсов, оптимизируя вычислительную мощность. Улучшенная экономическая эффективность — политика ММПР сбалансировала энергопотребление и производительность, что привело к повышению экономической эффективности на 20% по сравнению с традиционными методами.

Лучшее соответствие QoS подтверждается тем, что система выполнила 97% требований QoS при меняющихся нагрузках запросов, превзойдя другие подходы.

Выводы. Модели принятия решений Маркова могут быть применены для оптимизации проектных решений в управлении трафиком запросов в условиях неопределенности и многокритериальности. Моделируя стохастические переходы и результаты решений, Марковские модели принятия решений позволяют использовать адаптивные и эффективные стратегии [5].

Проведение исследования демонстрирует эффективность моделей принятия решений Маркова в оптимизации проектных решений в контейнеризированных средах. Используя адаптивное планирование на основе ММПР, можно эффективно управлять трафиком запросов даже в условиях неопределенности и колеблющихся рабочих нагрузок. ММПР превосходят традиционные подходы к планированию в обработке динамического трафика запросов за счет оптимизации времени отклика, использования ресурсов и экономической эффективности. Многокритериальная оптимизация необходима для балансировки QoS, производительности и эксплуатационных расходов в облачных вычислительных средах.

Будущие исследования ориентированы на оптимизацию отказоустойчивых кластеров виртуальных машин на моделях контейнерной виртуализации [2] при должны изучить интеграцию глубокого обучения с мониторингом в реальном времени для дальнейшего улучшения принятия решений в облачных инфраструктурах.

Список использованных источников:

1. Фунг В. Задержки и надежность обслуживания запросов в виртуальном компьютерном кластере. / Фунг В., Богатырев В. А. // Вестник Томского государственного университета, 2024. – № 68. – С. 48–58.
2. Фунг В. Оценка вероятностно-временных характеристик компьютерной системы с контейнерной виртуализацией. / Фунг В., Богатырев В. А., Кармановский Н. С., Лэ В. // Научно-технический вестник информационных технологий, механики и оптики, 2024. – Т. 24. № 2. – С. 249–255.
3. Bellman R. E. Dynamic programming. – Princeton University Press, 1957. – 340 pp.
4. Bertsekas D. Reinforcement Learning and Optimal Control. – Athena Scientific, 2019. – 388 pp.
5. Bogatyrev V. A. Timeliness of the reserved maintenance by duplicated computers of heterogeneous delay-critical stream. / Bogatyrev V. A., Derkach A. N., Bogatyrev S. V. // Information Systems and Technologies in Modeling and Control, 2019. – С. 26-36.
6. Kaelbling L. Planning and acting in partially observable stochastic domains. / Kaelbling L. P., Littman L. M., Cassandra A. R. – Artificial Intelligence, 1998. – Vol. 101 – 99-134 pp.
7. Kubernetes. Режим доступа: <https://kubernetes.io/> (дата обращения: 09.02.2025).
8. Puterman M. L. Markov Decision Processes: Discrete Stochastic Dynamic Programming. – Wiley, 2014. – 684 pp.
9. Sutton R. S. Reinforcement Learning. Режим доступа: <https://web.stanford.edu/class/psych209/Readings/SuttonBartoIPRLBook2ndEd.pdf> (дата обращения: 09.02.2025).