УДК 517.9

САМОХИНА Наталья Станиславовна, кандидат технических наук, доцент кафедры высшей школы передовых производственных технологий, Поволжский государственный университет сервиса, г. Тольятти

ЕФРЕМОВ Алексей Станиславович, аспирант, Поволжский государственный университет сервиса, г. Тольятти

ОПТИМИЗАЦИЯ РАСПРЕДЕЛЕНИЯ ВЫЧИСЛИТЕЛЬНЫХ РЕСУРСОВ В ГЕТЕРОГЕННЫХ КЛАСТЕРАХ С ПРИМЕНЕНИЕМ ТЕХНОЛОГИИ ВИРТУАЛИЗАЦИИ HAMi

Целью исследования является разработка и экспериментальная проверка методики оптимального распределения вычислительных ресурсов графических процессоров (GPU) в гетерогенных Kubernetes-кластерах с применением middleware-платформы виртуализации HAMi. В работе рассматривается задача совместного использования физически разнородных GPU несколькими вычислительными нагрузками при жёстких ограничениях по видеопамяти (gpumem) и вычислительным ядрам (gpucores), а также оценивается устойчивость планировщика HAMi к отказам при исчерпании квот. Экспериментальная часть выполнена на кластере под управлением Kubernetes v1.29.15 с использованием профилей vGPU различного размера (small/medium/large). Установлено, что критический порог отказа планировщика по ядровой квоте находится в диапазоне 95–100% суммарной загрузки узла, при этом ограничивающим ресурсом в большинстве сценариев выступают вычислительные ядра, а не объём видеопамяти. Показано, что HAMi обеспечивает аппаратную изоляцию памяти на уровне заданной квоты (889/1000 MiB) независимо от модели физического GPU даже при 100%-й вычислительной нагрузке, однако зафиксирован единичный случай

перезапуска подконтрольного pod'a, что указывает на необходимость дополнительного исследования устойчивости системы под длительной нагрузкой. Результаты имеют практическую значимость для проектирования энергоэффективных и устойчивых политик планирования в гетерогенных AI-инфраструктурах.

Annotation

The aim of the research is to develop and experimentally validate a methodology for optimal allocation of GPU computing resources in heterogeneous Kubernetes clusters using the HAMi (Heterogeneous AI Computing Virtualization Middleware) virtualization platform. The paper addresses the problem of sharing physically heterogeneous GPUs among multiple workloads under hard constraints on GPU memory and compute cores, and evaluates the resilience of the HAMi scheduler to failures under quota exhaustion. The experimental part was carried out on a cluster running Kubernetes v1.29.15 using vGPU profiles of different sizes (small/medium/large). It was found that the critical failure threshold of the scheduler with respect to the core quota lies in the range of 95–100% of total node load, with compute cores — rather than memory volume — acting as the limiting resource in most scenarios. It is shown that HAMi enforces hard memory isolation at the level of the assigned quota (889/1000 MiB) regardless of the physical GPU model, even under 100% computational load; however, a single pod restart event was recorded, indicating the need for further investigation of system stability under sustained load. The results are of practical significance for designing energy-efficient and resilient scheduling policies in heterogeneous AI infrastructures.

Ключевые слова: гетерогенные вычислительные кластеры, виртуализация GPU, HAMi, Kubernetes, планирование ресурсов, устойчивость систем, энергоэффективность, vGPU, изоляция ресурсов.

Keywords: heterogeneous computing clusters, GPU virtualization, HAMi, Kubernetes, resource scheduling, system resilience, energy efficiency, vGPU, resource isolation.

Рост масштабов задач машинного обучения и высокопроизводительных вычислений привёл к резкому увеличению спроса на графические ускорители (GPU), при этом инфраструктура значительной части организаций формируется постепенно и включает физически разнородное оборудование — GPU различных поколений и производительности [9]. Такая гетерогенность создаёт проблему эффективного совместного использования дорогостоящих ускорителей: традиционный планировщик Kubernetes рассматривает GPU как неделимый бинарный ресурс, что приводит к низкой утилизации оборудования и невозможности гибкого распределения нагрузки между разными моделями карт [8].

Для решения этой проблемы разработан ряд middleware-платформ виртуализации GPU для Kubernetes, среди которых выделяется HAMi (Heterogeneous AI Computing Virtualization Middleware, ранее k8s-vGPU-scheduler) — проект, принятый в CNCF Sandbox в 2024 году и получивший статус Incubating в 2026 году [7]. HAMi предоставляет унифицированный интерфейс управления разнородными ускорителями (NVIDIA, AMD, Cambricon, Hygon, Ascend и др.), обеспечивая частичное распределение устройства по памяти и вычислительным ядрам, а также аппаратную изоляцию ресурсов на уровне контейнера [6].

Актуальность настоящего исследования обусловлена недостаточной изученностью поведения планировщика HAMi в условиях предельной загрузки гетерогенного кластера, в частности — вопросов устойчивости распределения ресурсов (resilience)[5] и оптимальности энергопотребления при совместном использовании физически различных GPU. Существующие работы преимущественно рассматривают общие принципы GPU-

ориентированного планирования в Kubernetes (bin packing, gang scheduling, topology-aware placement), однако количественные экспериментальные данные о поведении конкретно HAMi под стресс-нагрузкой на разнородном оборудовании остаются крайне ограниченными.

Целью работы является экспериментальное исследование механизмов распределения вычислительных ресурсов HAMi в гетерогенном GPU-кластере с акцентом на устойчивость планирования при исчерпании квот и на характеристики энергопотребления/производительности физически разных ускорителей при равных виртуальных профилях.

Задачи исследования:

1. Развернуть экспериментальный гетерогенный Kubernetes-кластер с HAMi и двумя моделями GPU (RTX 3060 Ti, RTX 4060 Ti).

2. Провести серию тестов с профилями vGPU разного размера для определения закономерностей распределения ресурсов и порогов отказа планировщика.

3. Оценить устойчивость системы под стресс-нагрузкой (gpu-burn) на обеих моделях карт.

4. Проанализировать соотношение энергопотребления, температуры и производительности при равных vGPU-квотах на разнородном оборудовании.

Методологическую основу работы составляет комбинация системного анализа архитектуры HAMi и экспериментального метода — контролируемого нагрузочного тестирования на реальном физическом кластере, что соответствует практике эмпирической апробации, принятой в аналогичных работах по оценке качества гетерогенных вычислительных систем [1].

Платформа HAMi состоит из четырёх компонентов: мутирующего вебхука (HAMi MutatingWebhook), который определяет, может ли задача обслуживаться HAMi, и назначает соответствующий scheduler-extender; расширения планировщика (HAMi scheduler-extender), отвечающего за назначение задач узлам и устройствам с учётом глобального состояния гетерогенных ресурсов; device-plugin, сопоставляющего результат планирования с конкретным физическим устройством; и внутриконтейнерного модуля контроля ресурсов (HAMi-Core), обеспечивающего аппаратную изоляцию памяти и ядер [3]. Ресурсы GPU задаются пользователем через три параметра Kubernetes-манифеста: `nvidia.com/gpu` (число физических GPU), `nvidia.com/gpumem` (лимит видеопамяти в MiB) и `nvidia.com/gpucores` (доля вычислительных ядер в процентах) [2][4].

Кластер развёрнут на базе Kubernetes v1.29.15 с container runtime containerd и включает control-plane (с taint, исключающим размещение нагрузки) и два worker-узла: worker1 с GPU NVIDIA GeForce RTX 3060 Ti и worker2 с GPU NVIDIA GeForce RTX 4060 Ti. После установки HAMi с параметром `device-split-count = 10` каждый физический GPU регистрируется в кластере как ресурс `nvidia.com/gpu: 10`, что соответствует десяти виртуальным слотам на каждой карте.

Исследование проведено в две сессии:

- Сессия 1 (профили распределения ресурсов).

  Тест-набор 1: последовательное развёртывание четырёх pod'ов с профилями vGPU разного размера — gpu-small-1 и gpu-small-2 (gpumem=1000 МиБ, gpucores=20%), gpu-medium-1 (gpumem=4000 МиБ, gpucores=50%), gpu-large-1 (gpumem=7000 МиБ, gpucores=90%).

Тест-набор 2: развёртывание двух дополнительных «стресс»-подов (gpumem=2000 МиБ, gpucores=20% каждый) при уже занятых узлах, направленное на выявление отказа планировщика.

Тест-набор 3: развёртывание low-core pod'a (gpumem=2000 МиБ, gpucores=5%) с целью уточнения критического порога отказа.

- Сессия 2 (нагрузочное тестирование gpu-burn).

  На каждый физический GPU назначался pod с равным профилем vGPU (`nvidia.com/gpu:1`, `gpumem:1000`, `gpucores:20`), выполняющий инструмент gpu-burn (oguzpastirmaci/gpu-burn) для создания 100%-й вычислительной нагрузки.

  Для второго теста worker2 временно переводился в режим cordon, чтобы гарантированно направить второй pod на worker1, после чего оба pod'a работали параллельно. Фиксировались метрики nvidia-smi (температура, энергопотребление, загрузка памяти, GPU-Util) и лог производительности gpu_burn (Gflop/s, число ошибок).

Такой дизайн позволяет разделить два аспекта оптимизации: корректность и предсказуемость распределения ресурсов планировщиком HAMi при масштабировании профилей vGPU и устойчивость и энергоэффективность аппаратной изоляции под реальной вычислительной нагрузкой на физически разных ускорителях.

РЕЗУЛЬТАТЫ ЭКСПЕРИМЕНТАЛЬНОГО ИССЛЕДОВАНИЯ

В тест-наборе 1 планировщик HAMi распределил нагрузку неравномерно между узлами в соответствии со свободной квотой: три малых и средних pod'a (gpu-small-1, gpu-small-2, gpu-medium-1) были размещены на worker1, суммарно заняв gpumem=6000 МиБ и gpucores=90%, тогда как крупный pod (gpu-large-1) был размещён на worker2 с загрузкой gpumem=7000 МиБ и gpucores=90%. Таблица 1 обобщает итоговое распределение ресурсов по узлам.

Таблица 1. Распределение ресурсов vGPU по узлам кластера (тест-набор 1)

Узел	GPU	Размещённые pod'ы	gpumem, МиБ	gpucores, %
worker1	RTX 3060 Ti	gpu-small-1, gpu-small-2, gpu-medium-1	6000	90
worker2	RTX 4060 Ti	gpu-large-1	7000	90

Показательно, что оба узла достигли одинаковой загрузки по ядрам (90%) при разных объёмах занятой памяти, что свидетельствует о независимом учёте планировщиком двух типов квот и о приоритете балансировки по свободному месту (`FilteringSucceed` с метриками score по узлам).

В тест-наборе 2 попытка развернуть два дополнительных стресс-pod'а (gpucores=20% каждый) была отклонена планировщиком с ошибкой `CardInsufficientCore` на обоих узлах, несмотря на достаточный объём свободной видеопамяти (на worker2 суммарная потребность 9000 МиБ из доступных 16380 МиБ). Это подтверждает, что при загрузке ядер 90% на обоих узлах любое дополнительное выделение 20% приводит к превышению лимита 100% и гарантированному отказу — ограничивающим ресурсом выступают вычислительные ядра, а не память.

В тест-наборе 2 попытка развернуть два дополнительных стресс-pod'а (gpucores=20% каждый) была отклонена планировщиком с ошибкой `CardInsufficientCore` на обоих узлах, несмотря на достаточный объём

свободной видеопамяти (на worker2 суммарная потребность 9000 МиБ из доступных 16380 МиБ). Это подтверждает, что при загрузке ядер 90% на обоих узлах любое дополнительное выделение 20% приводит к превышению лимита 100% и гарантированному отказу — ограничивающим ресурсом выступают вычислительные ядра, а не память.

Таблица 2. Динамика загрузки ядер worker1 (RTX 3060 Ti) при последовательном добавлении pod'ов

Pod	Запрошено gpucores, %	Суммарная загрузка узла, %	Результат планирования
gpu-small-1	20	20	Scheduled
gpu-small-2	20	40	Scheduled (после FailedScheduling "node locked")
gpu-medium-1	50	90	Scheduled (после повторных FailedScheduling)
gpu-stress-1/2	20	110 (расчётно)	Отказ — CardInsufficientCore
gpu-lowcore-1	5	95	Scheduled

При равном профиле vGPU (gpumem=1000 МиБ, gpucores=20%) оба физических GPU достигли 100% GPU-Util и близкого к паспортному TDP энергопотребления: RTX 4060 Ti — 164/165 Вт (99,4% TDP), RTX 3060 Ti — 199/200 Вт (99,5% TDP). При этом использование видеопамяти на обеих

картах составило 889/1000 МиБ, то есть заданный лимит HAMi не был превышен ни на одной модели GPU, что подтверждает корректность аппаратной изоляции памяти независимо от архитектуры устройства. Таблица 3 суммирует сравнительные показатели устойчивости и энергопотребления.

Таблица 3. Сравнительные показатели GPU под нагрузкой gpu-burn при равном профиле vGPU (gpumem=1000, gpucores=20)

Показатель	RTX 4060 Ti (worker2)	RTX 3060 Ti (worker1)
Температура, °C	85	68
Энергопотребление, Вт (% TDP)	164/165 (99,4%)	199/200 (99,5%)
Использование памяти, МиБ	889/1000	889/1000
GPU-Util, %	100	100
Производительность, Gflop/s	8500–8700	8300
Число ошибок вычислений	0	0
Число перезапусков pod	не зафиксировано	1

Зафиксированный единичный перезапуск pod'a gpu-burn-small-1 на worker1 примерно через 8 минут работы (без ошибок вычислений в логе) при отсутствии вычислительных ошибок причиной может быть как штатное завершение цикла теста с последующим автоматическим перезапуском по политике Pod, так и потенциальная нестабильность модуля hard isolation HAMi под длительной пиковой нагрузкой (например, конфликт OOM на уровне драйвера vGPU).

РЕЗУЛЬТАТЫ

Полученные результаты позволяют сформулировать три ключевых вывода об оптимизации распределения ресурсов в гетерогенных HAMi-кластерах. Во-первых, в исследованной конфигурации именно вычислительные ядра (gpucores), а не объём видеопамяти, выступают связывающим ограничением при плотной упаковке нагрузки, что согласуется с общими наблюдениями по GPU-ориентированному планированию в Kubernetes, где утилизация ресурсов существенно зависит от корректного «правильного подбора размера» (right-sizing) запросов. Практическое следствие — при проектировании политик распределения нагрузки в гетерогенных кластерах приоритет мониторинга и резервирования должен отдаваться именно ядровой квоте.

Во-вторых, аппаратная изоляция памяти HAMi (hard isolation) продемонстрировала полную независимость от модели физического GPU: оба ускорителя (RTX 3060 Ti и RTX 4060 Ti) при равном виртуальном профиле показали идентичное поведение по памяти (889/1000 МиБ) при существенно различном энергопотреблении, обусловленном разной архитектурой и TDP карт (200 Вт против 165 Вт). Это подтверждает корректность реализации HAMi-Core для гетерогенного оборудования и согласуется с описанной в документации проекта моделью виртуализации ресурсов через контроль на уровне контейнера .

В-третьих, оптимальное энергопотребление в гетерогенном кластере при использовании HAMi не гарантируется автоматически: одинаковая доля вычислительных ядер (20%) привела к практически полной загрузке TDP на обеих картах (99,4–99,5%), то есть виртуальная квота ядер транслируется в пропорциональную реальную вычислительную (и, следовательно, энергетическую) нагрузку без программного демпфирования. Для задач, где приоритетом является энергоэффективность, а не максимальная производительность, целесообразно комбинировать квотирование HAMi с

дополнительными политиками управления частотой/энергопотреблением на уровне драйвера, поскольку сам HAMi такого механизма не предоставляет .

Сопоставление с методологическими подходами, применяемыми при оценке качества сложных гетерогенных систем обработки данных, показывает, что предложенный экспериментальный протокол (пошаговое масштабирование профилей ресурсов + контролируемая стресс-нагрузка) обеспечивает воспроизводимость и объективность результатов, аналогично комбинированным методикам системного анализа и экспертной оценки, показавшим повышение индекса стабильности систем до 0,92 при пороговом значении 0,75 в смежных исследованиях качества гетерогенных информационных систем .

ЗАКЛЮЧЕНИЕ

Экспериментальное исследование распределения вычислительных ресурсов в гетерогенном GPU-кластере с применением технологии виртуализации HAMi позволило установить количественные закономерности, ранее не зафиксированные в открытых источниках. Показано, что критический порог отказа планировщика по ядровой квоте лежит в диапазоне 95–100% суммарной загрузки узла, а ограничивающим ресурсом в большинстве сценариев выступают вычислительные ядра, а не видеопамять. Подтверждена корректность аппаратной изоляции памяти HAMi-Core независимо от модели физического GPU (RTX 3060 Ti/RTX 4060 Ti) даже под 100%-й вычислительной нагрузкой, при этом виртуальная квота ядер транслируется в пропорциональную реальную нагрузку TDP без встроенного механизма энергетической оптимизации. Зафиксированный случай перезапуска pod'a под стресс-нагрузкой обозначает область для дальнейшего изучения устойчивости гетерогенной vGPU-инфраструктуры.

Практическая значимость исследования заключается в формулировании рекомендаций по проектированию политик планирования в гетерогенных

Kubernetes-кластерах: приоритетный мониторинг ядровой квоты, учёт различий TDP физических карт при формировании энергетического бюджета кластера, а также необходимость дополнительного контроля устойчивости при длительной пиковой нагрузке. Дальнейшие исследования планируется направить на количественное сравнение производительности (Gflop/s) для обеих моделей GPU при равных vGPU-профилях большего размера (gpumem=4000/gpucores=50, gpumem=7000/gpucores=90), а также на выявление первопричины зафиксированной аномалии перезапуска.

СПИСОК ЛИТЕРАТУРЫ

1. Самохина Н.С., Ефремов А.С. Алгоритмические методы событийно-прогнозного управления качеством сложных систем обработки данных: интеграция системного анализа и вычислительного моделирования // Computational Nanotechnology. 2025. Т. 12. № 2. С. 75–81. DOI: 10.33693/2313-223X-2025-12-2-75-81. EDN: QWZBHA.

2. Самохина Н.С., Ефремов А.С. Процедура агрегирования исходных данных о требуемом уровне качества сложных систем обработки данных // Computational Nanotechnology. 2025. Т. 12. № 4. С. 61–70. DOI: 10.33693/2313-223X-2025-12-4-61-70. EDN: FPYQLP.

3. Еременко В.Т., Логинов И.В., Фисун А.П., Рытов М.Ю. Управление перестройкой информационно-вычислительных платформ эволюционирующих киберфизических систем в условиях неопределённости // Вестник компьютерных и информационных технологий. 2023. № 2. С. 26–36. DOI: 10.14489/vkit.2023.02.pp.026-036. EDN: HOSPS.

4. Мельников А.В., Кобяков Н.С. Численный метод модификации моделей, разработанных на основе метода анализа иерархий, с использованием искусственной нейронной сети // Вестник ВГУ. Серия: Системный анализ и информационные технологии. 2025. № 4. С. 5–21. DOI: 10.17308/sait/1995-5499/2024/4/5-21. EDN: CENQIP.

5. Меньших В.В., Морозова В.О. Численный метод исследования динамических рядов с апериодическими аномалиями // Вестник ВГУ. Серия: Системный анализ и информационные технологии. 2023. № 2. С. 22–30. DOI: 10.17308/sait/1995-5499/2023/2/22-30. EDN: GVIANX.

6. Десницкий В.А., Новикова Е.С. Обнаружение неисправностей в промышленных изделиях с использованием малых обучающих наборов данных // Вестник ВГУ. Серия: Системный анализ и информационные

технологии. 2024. № 1. С. 49–61. DOI: 10.17308/sait/1995-5499/2024/1/49-61. EDN: DHNYCM.

7. Вересников Г.С., Голев А.В., Московцев А.М., Мартиросян М.П. Методы и алгоритмы для решения задачи ранней диагностики технических объектов с использованием методов интеллектуального анализа данных // Информационные технологии. 2022. № 9 (28). С. 475–484. DOI: 10.17587/it.28.475-484. EDN: UJWIRT.

8. Аршинский Л.В., Лебедев В.С. Объективизация баз знаний интеллектуальных систем на основе индуктивного вывода с использованием нестрогих вероятностей // Информационные и математические технологии в науке и управлении. 2022. № 4 (28). С. 190–200. DOI: 10.38028/ESI.2022.28.4.015. EDN: LGJXFH.

9. Меркелов, М. В. Влияние настроек слоя рецепторов модуля имитации сетчатки на возможность распознавания образов нейронной сетью / М. В. Меркелов, Д. А. Локтев // Биотехнические, медицинские и экологические системы, измерительные устройства и робототехнические комплексы - Биомедсистемы-2025 : Сборник трудов XXXVIII Всероссийской научно-технической конференции студентов, молодых ученых и специалистов, Рязань, 03–05 декабря 2025 года. – Рязань: Рязанский государственный радиотехнический университет им. В.Ф. Уткина, 2025. – С. 254-259. – EDN MQBSJL.