

Генка Иванова Славова-Торрес

МЕТОДИ И СРЕДСТВА ЗА ОПТИМИЗИРАНЕ НА ЗАПИСИ В МРЕЖОВА СРЕДА ЗА СЪХРАНЕНИЕ

ДИСЕРТАЦИЯ

за придобиване на образователна и научна степен „доктор”
по докторска програма **Компютърни системи, комплекси и мрежи**
професионално направление **5.3 Комуникационна и компютърна техника**

научен ръководител: доц. д-р Владимир Иванов

София 2026

СЪДЪРЖАНИЕ

УВОД	6
СТРУКТУРА НА ДИСЕРТАЦИОННИЯ ТРУД	8
РЕЧНИК НА ТЕРМИНИ И СЪКРАЩЕНИЯ, ИЗПОЛЗВАНИ В ДИСЕРТАЦИОННИЯ ТРУД	10
ГЛАВА 1. ПРОБЛЕМ, ЦЕЛИ И ЗАДАЧИ	13
1.1. МОТИВАЦИЯ И ПРОБЛЕМАТИКА	13
1.2. ПРАКТИЧЕСКА ЗНАЧИМОСТ И ПРИЛОЖИМОСТ НА ИЗСЛЕДВАНЕТО	15
1.3. ФОРМУЛИРАНЕ НА ПРОБЛЕМА	16
1.4. ИЗСЛЕДОВАТЕЛСКА ТЕЗА	17
1.5. МЕТОДОЛОГИЯ НА ИЗСЛЕДВАНЕТО	19
1.6. ОГРАНИЧЕНИЯ И ОБХВАТ НА ИЗСЛЕДВАНЕТО	20
ГЛАВА 2. СИСТЕМЕН МОДЕЛ НА ОПЕРАЦИИТЕ ПО ЗАПИС В МРЕЖОВИ СРЕДИ ЗА СЪХРАНЕНИЕ (SAN)	21
2.1. ЦЕЛ НА ГЛАВАТА	22
2.2. ЕВОЛЮЦИЯ НА ТЕСНИТЕ МЕСТА (BOTTLENECKS) ПРИ ОПЕРАЦИИТЕ ПО ЗАПИС	23
2.3. ПЪТЯТ НА ЗАПИСИТЕ КАТО СИСТЕМА	27
2.4. ОПАШКИ ОТ СТРАНА НА ХОСТА	31
2.4.1. Ролята на хост системата в пътя на записите	31
2.4.2. Генериране на заявки за запис в приложението	31
2.4.3. Операционна система и файлова система като буфер	32
2.4.4. Блоков слой и планиране на входно-изходните заявки	33
2.4.5. Драйвер и хостов адаптер (HBA): границата между хост и SAN	34
2.4.6. Паралелизъм: кога помага и кога започва да вреди	34
2.5. Опашки и управление в транспортния слой (fabric)	35
2.6. ОПАШКИ В КОНТРОЛЕРИТЕ И КЪМ ФИЗИЧЕСКАТА СРЕДА	37
2.6.1. Входен (front-end) слой в системата за съхранение	37
2.6.2. Контролери и вътрешно планиране на операциите по запис	37
2.6.3. Потвърждаване на запис и семантика на завършване на операциите	39
2.6.4. Кеш на записите – стабилизация и маскиране на проблеми	40
2.6.5. Опашки към физическата среда и вътрешен паралелизъм	41
2.6.6. Синтез: „бърза среда“ не означава стабилно време за запис	42
2.7. ИНТЕРПРЕТАЦИЯ НА ПЪТЯ НА ЗАПИСИТЕ ЧРЕЗ ТЕОРИЯ НА МАСОВОТО ОБСЛУЖВАНЕ	43
2.8. МЕТРИКИ И ДЕФИНИЦИИ ЗА АНАЛИЗА	45
2.8.1. Латентност като основна метрика при записите	45
2.8.2. Променливост на латентността и стабилност на записите	47
2.8.3. Вариации и трептения във времето при натоварване	48
2.8.4. Персентили: p50, p95, p99 – смисъл и интерпретация	51

2.8.5. Интерпретация на персентилите при записи в SAN	53
2.8.6. Поток на измерване и интерпретация чрез Емпирична кумулативна функция на разпределение (ECDF)	54
2.9. ЧЕТЕНЕ СРЕЩУ ЗАПИС	57
2.10. ОБОБЩЕНИЕ НА МОДЕЛА И ВРЪЗКА С ЕКСПЕРИМЕНТИТЕ	57
ГЛАВА 3. ЕКСПЕРИМЕНТАЛНА МЕТОДОЛОГИЯ	58
3.1. ИЗСЛЕДОВАТЕЛСКА ЦЕЛ И ЛОГИКА НА ЕКСПЕРИМЕНТАЛНИЯ ПОДХОД	58
3.2. ИЗСЛЕДОВАТЕЛСКИ МОДЕЛ: КОНКУРЕНЦИЯ, ОПАШКИ И ЛАТЕНТНОСТ	60
3.3. НАТОВАРВАНЕТО КАТО НЕЗАВИСИМА ПРОМЕНЛИВА	61
3.4. КОНТРОЛИРАНА ЕКСПЕРИМЕНТАЛНА СРЕДА	62
3.5. ОПАШКИТЕ КАТО ОСНОВЕН ОБЕКТ НА НАБЛЮДЕНИЕ	64
3.6. ЕКСПЕРИМЕНТАЛНИ ПАРАМЕТРИ И КОНФИГУРАЦИЯ	66
3.7. МЕТРИКИ И ИЗМЕРВАТЕЛЕН ПОДХОД	72
3.8. СЪБИРАНЕ НА ЛОГОВЕ И МЕТРИКИ	74
3.8. ЛОГИКА И ПОСЛЕДОВАТЕЛНОСТ НА ЕКСПЕРИМЕНТИТЕ	75
3.9. КАКВО НЕ СЕ ИЗМЕРВА И ЗАЩО	76
3.10. МЕТОДОЛОГИЧНО ОБОБЩЕНИЕ	77
ГЛАВА 4. ЕКСПЕРИМЕНТАЛНИ РЕЗУЛТАТИ	78
4.1. БАЗОВ ЕКСПЕРИМЕНТАЛЕН СЦЕНАРИЙ (РЕФЕРЕНТНО ПОВЕДЕНИЕ)	79
4.1.1. Цел на експеримента	79
4.1.2. Конфигурация и натоварване	79
4.1.3. Резултати: латентност и производителност	80
4.1.4. Разпределение на латентностите (ECDF)	82
4.1.5. Обобщение на референтното поведение	84
4.2. ЕКСПЕРИМЕНТ 1: ВЛИЯНИЕ НА ДЪЛБОЧИНАТА НА ОПАШКАТА (QD) ВЪРХУ ЛАТЕНТНОСТТА НА ЗАПИСИТЕ (МОДЕЛНО НАДГРАЖДАНЕ)	85
4.2.1. Цел на експеримента	85
4.2.2. Конфигурация и натоварване	85
4.2.3. Резултати	87
4.2.4. Анализ на влиянието на дълбочината на опашката върху крайните латентностни случаи	88
4.2.5. Крайни стойности на латентността (p_{99}) като функция на QD	90
4.2.6. Обобщение на Експеримент 1	91
4.3. ЕКСПЕРИМЕНТ 2: ВЛИЯНИЕ НА ФОРМАТА НА НАТОВАРВАНЕТО ВЪРХУ ЛАТЕНТНОСТТА НА ЗАПИСИТЕ	92
4.3.1. Цел на експеримента	92
4.3.2. Конфигурация и натоварване	93
4.3.3. Резултати: латентност и производителност	94
4.3.4. Разпределение на латентността (ECDF)	95
4.3.5. Крайни стойности на латентността при различна форма на натоварването	97

4.3.6. <i>Обобщение на Експеримент 2</i>	98
4.4. ЕКСПЕРИМЕНТ 3: КООРДИНАЦИЯ СРЕЩУ СУРОВА СКОРОСТ	99
4.4.1. <i>Цел на експеримента</i>	99
4.4.2. <i>Конфигурация и режим на работа</i>	99
4.4.3. <i>Резултати: латентност и производителност</i>	100
4.4.4. <i>Разпределение на латентностите при координирано и некоординирано подаване (ECDF)</i>	102
4.4.5. <i>Обобщение: баланс между координация и производителност</i>	103
4.5. ЕКСПЕРИМЕНТ 4: ВЛИЯНИЕ НА КОНКУРЕНТЕН ХОСТ (NOISY NEIGHBOR) ВЪРХУ ЛАТЕНТОСТТА НА ЗАПИСИТЕ ПРИ СПОДЕЛЕН ТОМ	104
4.5.1. <i>Цел на експеримента</i>	104
4.5.2. <i>Конфигурация и сценарии</i>	105
4.5.3. <i>Резултати: латентност и производителност</i>	107
4.5.4. <i>Разпределение на латентността (ECDF)</i>	108
4.5.5. <i>Обобщение</i>	110
4.6. ОБОБЩЕНИЕ НА ЕКСПЕРИМЕНТАЛНАТА ГЛАВА	110
ГЛАВА 5. АНАЛИТИЧЕН СИНТЕЗ И ИНТЕРПРЕТАЦИЯ	112
5.1. ЦЕЛ И ОБХВАТ НА АНАЛИТИЧНИЯ СИНТЕЗ	112
5.2. СЪПОСТАВЯНЕ НА ЕКСПЕРИМЕНТАЛНИТЕ СЦЕНАРИИ	113
5.3. АНАЛИЗ НА ЛАТЕНТНОСТНИТЕ РАЗПРЕДЕЛЕНИЯ И ОПАШКОВИ ЕФЕКТИ	115
5.4. ИНТЕРПРЕТАЦИЯ НА КОМПРОМИСА МЕЖДУ ПРОИЗВОДИТЕЛНОСТ, КООРДИНАЦИЯ И УСТОЙЧИВОСТ	119
5.5. ОПТИМИЗАЦИОННА ИНТЕРПРЕТАЦИЯ ЗА РЕЗУЛТАТИТЕ	123
5.7. РАМКА ЗА ОПТИМИЗАЦИЯ НА ОПЕРАЦИИТЕ ПО ЗАПИС В SAN СРЕДА	126
5.6. ПРАКТИЧЕСКИ ИНТЕРПРЕТАЦИИ И ПРИЛОЖИМОСТ НА РЕЗУЛТАТИТЕ	129
5.7. ОБОБЩЕНИЕ НА АНАЛИТИЧНИЯ СИНТЕЗ	133
ГЛАВА 6. ЗАКЛЮЧЕНИЕ И БЪДЕЩА РАБОТА	134
6.1. ЗАКЛЮЧЕНИЕ	134
6.2. ОГРАНИЧЕНИЯ И ОТВОРЕНИ ВЪПРОСИ	135
6.3. НАСОКИ ЗА БЪДЕЩИ ИЗСЛЕДВАНИЯ	136
ПРИНОСИ	138
ИЗПОЛЗВАНА ЛИТЕРАТУРА	140
АВТОРСКИ ПУБЛИКАЦИИ ПО ДИСЕРТАЦИЯТА	143
ДЕКЛАРАЦИЯ ЗА ОРИГИНАЛНОСТ НА РЕЗУЛТАТИТЕ	144
Приложение А	145

СПИСЪК НА ФИГУРИТЕ

Фигура 1. Еволюция на Bottleneck	24
Фигура 2. Концептуален модел на пътя на записите в мрежова среда за съхранение, илюстриращ разпределението на опашките и регулаторите по различните нива на системата	28
Фигура 3. Логически модел на пътя на записите и точки на опашкуване	30
Фигура 4. Логическо завършване на операцията по запис и асинхронно физическо изпълнение	39
Фигура 5. Вътрешна организация на физическата среда за съхранение при операции по запис и формиране на невидими за хоста опашки	41
Фигура 6. Интерпретация на записите като последователност от обслужващи възли и опашки	45
Фигура 7. Примерна илюстрация на латентността при записи, показваща трептения и редки пикове в рамките на фиксиран тест при стабилно натоварване	48
Фигура 8. Илюстративно разпределение на времето за обслужване на записите по диапазони на латентност	50
Фигура 9. Същата времева серия на латентността при записи, показана на Фигура 8, надградена с формално означени персентилни нива	52
Фигура 10. Поток на измерване и интерпретация на латентността чрез емпирична кумулативна функция на разпределение (ECDF)	55
Фигура 11. Концептуален модел на контролираната експериментална среда и пътя на записа, използван в изследването	63
Фигура 12. Концептуална схема на пътя на записа с обозначени логически точки на натрупване на заявки, релевантни за проведените експерименти	65
Фигура 13. Времеви профил на латентността на операциите по запис при базовия експериментален сценарий	81
Фигура 14. ECDF на латентността на операциите по запис за базовия експериментален сценарий	82
Фигура 15. Относително изменение на латентностните персентили (p50, p95 и p99) спрямо дълбочината на входно-изходната опашка	89
Фигура 16. Зависимостта на p99 латентността от дълбочината на опашката (моделни данни)	90
Фигура 17. ECDF на латентността на операциите по запис при равномерен поток (SI) и пулсиращ режим (S2) при фиксиран входен паралелизъм (моделно генерирани данни)	96
Фигура 18. ECDF на латентността на операциите по запис при некоординиран режим (R1) и координиран режим (R2) при фиксиран входен паралелизъм (моделно генерирани данни)	102
Фигура 19. Концептуална схема на експерименталните сценарии: Сценарий А – самостоятелен хост, извършващ операции по запис към логически том V; Сценарий В – два хоста, извършващи конкурентни операции по запис към един и същи логически том V (Noisy Neighbor)	106
Фигура 20. ECDF на латентността на хост А в двата сценария	109
Фигура 21. Раздуване на латентностната опашка (p99/p95) при различни експериментални сценарии	117
Фигура 22. Съпоставка между p95 и p99 латентност при различни експериментални сценарии.	120
Фигура 23. Режим на работа по пътя на записа при нарастваща конкуренция – от стабилно поведение към координационен стрес	122
Фигура 24. Рамка за оптимизация на операциите по запис в SAN среда	127

СПИСЪК НА ТАБЛИЦИТЕ

Таблица 1. Фиксирани параметри на експерименталния дизайн	67
Таблица 2. Променливи параметри в експериментите	70
Таблица 3. Матрица „експеримент → променен параметър“	71
Таблица 4. Примерна интерпретация на латентностите при операции по запис	74

Таблица 5. Параметри на базовия експериментален сценарий	80
Таблица 6. Измерени метрики за базовия експеримент.....	81
Таблица 7. Всички параметри, които не са изрично посочени, съвпадат с тези от базовия експериментален сценарий (Таблица 6).....	86
Таблица 8. Метрики за латентност и производителност при различна дълбочина на входно-изходната опашка	87
Таблица 9. Относително изменение на латентностните персентили спрямо референтния сценарий.....	88
Таблица 10. Параметри на режимите на натоварване	94
Таблица 11. Агрегирани показатели за латентност и производителност при различна форма на натоварването.....	94
Таблица 12. Агрегирани показатели за латентност и производителност при некоординиран и координиран режим....	101
Таблица 13. Поведение на Хост А при двата сценария	107
Таблица 14. Обобщение на експерименталните резултати спрямо референтното поведение.....	112
Таблица 15. Оптимизационни насоки, изведени от експерименталния анализ на операциите по запис в SAN среда ...	125
Таблица 16. Връзка между наблюдаваните латентностни ефекти и препоръчителните оптимизационни действия	128

УВОД

Съвременните мрежови среди за съхранение на данни (Storage Area Networks – SAN) се характеризират с висока степен на паралелизъм, виртуализация и споделяне на ресурси между множество приложения и системи [19, 21]. В тези среди операциите по запис заемат централно място, тъй като оказват пряко влияние върху стабилността, производителността и предсказуемостта на цялата инфраструктура. Въпреки значителния напредък в хардуерните технологии и същественото намаляване на латентността на физическите носители, управлението на операциите по запис в SAN среда остава сложен и трудно прогнозируем проблем.

Причината се крие в многостепенния характер на операцията по запис. В SAN архитектурата заявките преминават последователно през приложението, хост системата, транспортната инфраструктура и системата за съхранение, като всеки слой използва собствени механизми за буфериране, входно-изходни опашки и управление [13, 21]. Взаимодействието между тези механизми определя крайното латентностно поведение на системата. В резултат времето за изпълнение на операциите по запис не нараства плавно с увеличаване на натоварването, а демонстрира нелинейно поведение, характеризиращо се с разширяване на латентностното разпределение и поява на редки, но екстремно забавени операции [4, 40].

Тези явления трудно могат да бъдат описани чрез традиционните усреднени показатели за производителност. Средната латентност често остава в приемливи граници, докато крайните стойности на времето за изпълнение на малка част от операциите нарастват значително и оказват непропорционално силно влияние върху работата на приложенията [3, 40]. Поради това анализът на поведението на операциите по запис в SAN среда изисква изследване на латентностното разпределение, опашковите явления и високите латентностни персентили, а не единствено на усреднените показатели [6].

С развитието на съвременните SAN архитектури оптимизацията на операциите по запис все повече се определя от взаимодействието между системните слоеве, а не единствено от производителността на отделните хардуерни компоненти. Повишаването на производителността на устройствата само по себе си вече не е достатъчно за осигуряване на устойчиво латентностно поведение, тъй като при висока степен на паралелизъм определящи стават взаимодействията между натоварването, входно-изходните опашки и механизмите за координация. Следователно оптимизацията следва да бъде насочена към управление на тези взаимодействия чрез ограничаване на опашковите ефекти и поддържане на устойчиво и предсказуемо латентностно поведение.

Разработването на системен модел, описващ тези зависимости, е необходима предпоставка за формулиране на практически насоки и стратегии за оптимизация на операциите по запис в SAN среда.

В този контекст настоящият дисертационен труд е насочен към формализиране на поведението на операциите по запис чрез системен модел на пътя на записите (write path). Целта е да се изследва как взаимодействието между различните слоеве на системата влияе върху устойчивостта и предсказуемостта на латентността при различни режими на натоварване. За постигането на тази цел е разработена експериментална методология, базирана на реална SAN инфраструктура, която позволява контролирано изменение на параметрите на натоварването при фиксирана системна конфигурация [50, 52].

Получените резултати потвърждават, че поведението на операциите по запис в SAN среда се определя в по-голяма степен от взаимодействието между механизмите за управление на натоварването, входно-изходните опашки и координацията между отделните системни слоеве, отколкото от номиналната производителност на отделните компоненти [21]. Това обосновава разглеждането на SAN средата като цялостна система от взаимодействащи механизми, при която устойчивата оптимизация изисква системен, а не изолирано компонентен подход. На тази основа в дисертационния труд са формулирани принципи за оптимизация, насочени към ограничаване на крайната латентност чрез управление на дълбочината на входно-изходните опашки, координацията между конкурентните потоци и режимите на натоварване.

ЦЕЛ НА ИЗСЛЕДВАНЕТО

Целта на настоящия дисертационен труд е да се разработи и валидира системен модел на поведението на операциите по запис в SAN среда, който отчита взаимодействието между отделните слоеве на пътя на записите и позволява идентифициране на факторите, определящи крайната латентност, както и формулиране на принципи и насоки за оптимизация на операциите по запис при различни режими на натоварване.

ЗАДАЧИ НА ИЗСЛЕДВАНЕТО

За постигане на поставената цел са формулирани следните задачи:

1. Да се анализира архитектурата на SAN среда и да се идентифицират основните компоненти и механизми, участващи в пътя на операциите по запис.

2. Да се изследва ролята на опашките, буферите и механизмите за управление на натоварването в пътя на операциите по запис и тяхното влияние върху формирането на латентността.
3. Да се разработи модел на пътя на операциите по запис като система от взаимодействащи опашки и обслужващи механизми, позволяващ интерпретация на латентностните ефекти чрез принципите на теорията на масовото обслужване.
4. Да се разработи експериментална методология за изследване на поведението на операциите по запис в реална SAN среда при контролирано изменение на параметрите на натоварването.
5. Да се проведат експерименти с различни нива на паралелизъм и натоварване с цел анализ на поведението на латентността и нейните разпределения.
6. Да се анализират крайните персенти на латентността (p95, p99) чрез подходи, базирани на емпирични разпределителни функции (ECDF).
7. Да се установят зависимости между параметрите на натоварването, степента на конкуренция и изменението на формата на разпределението на латентността.
8. Да се анализира влиянието на междуслойната координация върху устойчивостта и предсказуемостта на операциите по запис и да се идентифицират режими на работа, водещи до деградация на крайната латентност.
9. Да се разработи аналитична рамка за оптимизация на операциите по запис в SAN среда, базирана на наблюдаваните зависимости между параметрите на натоварването, опашковите ефекти и поведението на крайните персенти на латентността.

СТРУКТУРА НА ДИСЕРТАЦИОННИЯ ТРУД

Дисертационният Дисертационният труд е структуриран в шест глави, които последователно развиват изследването – от формулиране на научния проблем и изграждане на концептуален модел до експериментална проверка, аналитична интерпретация и формулиране на практически насоки за оптимизация на операциите по запис в SAN среда.

Първа глава обосновава актуалността на изследването и разглежда особеностите на операциите по запис в SAN среда. Формулирани са целта, задачите, изследователската

теза, обхватът и ограниченията на изследването, както и неговите научни и приложни аспекти.

Втора глава представя концептуалния модел на пътя на записите (write path) и анализира основните механизми, определящи латентността на операциите по запис. Разгледани са взаимодействието между системните слоеве, ролята на входно-изходните опашки и приложението на теорията на масовото обслужване при интерпретацията на латентностното поведение.

Трета глава описва разработената експериментална методология, използваната SAN инфраструктура, измерваните показатели и организацията на експерименталните изследвания. Определени са фиксираните и променливите параметри, както и принципите за осигуряване на съпоставимост и възпроизводимост на резултатите.

Четвърта глава представя резултатите от проведения базов експериментален сценарий и четири експериментални изследвания, чрез които се анализира влиянието на дълбочината на входно-изходните опашки, формата на натоварването, координацията между заявките и конкурентния достъп върху латентността на операциите по запис.

Пета глава извършва аналитичен синтез на експерименталните резултати и интерпретира наблюдаваните зависимости чрез системния модел и принципите на теорията на масовото обслужване. Формулирани са практически насоки и рамка за оптимизация, основани на анализа на латентностните разпределения, високите персентили и режимите на работа на системата.

Шеста глава обобщава основните резултати от изследването, формулира научните приноси на дисертационния труд и очертава направления за бъдещи изследвания, свързани с развитието на моделите за анализ и оптимизация на операциите по запис в SAN среда.

РЕЧНИК НА ТЕРМИНИ И СЪКРАЩЕНИЯ, ИЗПОЛЗВАНИ В ДИСЕРТАЦИОННИЯ ТРУД

Термини

Латентност (Latency) - времето между инициирането на входно-изходна операция и нейното логическо завършване, измерено на нивото на заявителя. В контекста на SAN средите латентността се формира като системна характеристика, резултат от взаимодействието между множество слоеве и опашки по пътя на операциите по запис.

Крайна латентност (Tail latency) - характеристика на разпределението на латентността, описваща редките, но значително забавени операции по запис. Обикновено се анализира чрез високи персентилю (напр. p95, p99), тъй като именно те имат критично влияние върху поведението на приложенията.

Персентил на латентността - статистическа мярка, указваща стойността, под която попадат определен процент от наблюдаваните операции. Например p99 обозначава латентността, под която се намират 99% от всички измерени операции по запис.

Път на записите (Write path) - логическата и архитектурна последователност от компоненти и механизми, през които преминава операцията по запис - от приложението и хост системата, през транспортния слой, до системата за съхранение и физическата среда.

Опашка (Queue) - структура за временно задържане и подреждане на входно-изходни заявки, използвана за управление на паралелизма и достъпа до ресурси. Опашките са фундаментален елемент от архитектурата на SAN средите и оказват пряко влияние върху латентността.

Дълбочина на опашката (Queue depth) - броят на едновременно активните или чакащи заявки в дадена опашка. Увеличаването ѝ може да повиши пропускателната способност, но често води до нелинейно нарастване на крайната латентност.

Паралелизъм - свойство на системата да обработва множество операции по запис едновременно чрез припокриване във времето. В SAN средите паралелизмът се реализира чрез множество опашки и конкурентен достъп до ресурси.

Пропускателна способност (Throughput) - количеството данни или брой операции, които системата може да обработи за единица време. В дисертационния труд тя се разглежда като вторична метрика спрямо стабилността и предсказуемостта на латентността.

Нелинейно поведение - поведение на системата, при което увеличаването на натоварването не води до пропорционална промяна в латентността, а до резки скокове и разширяване на разпределението на времето за изпълнение.

Тесен участък (Bottleneck) - компонент или механизъм в системата, който ограничава общото поведение при даден режим на натоварване. В съвременните SAN среди bottleneck-ът често има динамичен и разпределен характер и не е локализиран в един елемент.

Координационен ефект - възникващо поведение, при което взаимодействието между независими механизми за управление (хост, транспорт, система за съхранение) води до нестабилност на латентността, без ясно изразено локално насищане.

Буфериране (Buffering) - временно съхраняване на данни или заявки в паметта с цел изглаждане на пикове в натоварването и повишаване на ефективността. Буферирането може да маскира реалното натоварване и да доведе до скрити опашкови ефекти [54].

Асинхронно потвърждаване на записите - механизъм, при който операцията по запис се счита за завършена преди реалното ѝ записване върху физическата среда, обикновено след запис в кеш на контролера.

Системен модел - абстрактно представяне на пътя на записите като система от взаимодействащи компоненти, опашки и регулатори, използвано за анализ и интерпретация на поведението на латентността.

Съкращения

SAN (Storage Area Network) - мрежова среда за съхранение, осигуряваща блоков достъп до споделени ресурси за съхранение чрез специализирана транспортна инфраструктура.

HDD (Hard Disk Drive) - механично устройство за съхранение на данни, при което латентността се доминира от физическите процеси на позициониране и ротация.

SSD (Solid State Drive) - устройство за съхранение на данни, базирано на флаш памет, без механични части, характеризиращо се с по-ниска латентност и по-висок паралелизъм спрямо HDD.

NVMe (Non-Volatile Memory Express) - интерфейс и команден модел за достъп до флаш-базирани устройства през PCI Express, проектиран за висока степен на паралелизъм и ниска латентност.

NVMe-oF (NVMe over Fabrics) - разширение на NVMe архитектурата за работа в мрежова среда, при което NVMe командите се пренасят през транспортни протоколи като Fibre Channel, TCP или RDMA.

FC (Fibre Channel) - високопроизводителен мрежов протокол, използван в SAN среди за надежден и детерминиран пренос на блокови команди.

IOPS (Input/Output Operations Per Second) - метрика за измерване на броя входно-изходни операции, които системата може да обработи за една секунда.

CPU (Central Processing Unit) - централен процесор, който участва активно в обработката и управлението на входно-изходните операции, особено в NVMe архитектурите.

OS (Operating System) - операционна система, която управлява генерирането, буферирането и планирането на входно-изходните операции в хост системата.

FTL (Flash Translation Layer) - слой в SSD контролерите, отговарящ за адресната трансляция, управлението на износването и фоновите операции върху флаш паметта.

p95, p99 - обозначения за 95-ти и 99-ти персентил на латентността, използвани за анализ на крайните стойности на времето за изпълнение на операциите по запис.

ГЛАВА 1. ПРОБЛЕМ, ЦЕЛИ И ЗАДАЧИ

1.1. Мотивация и проблематика

Съвременните мрежови среди за съхранение на данни (SAN¹) се характеризират с висока степен на паралелизъм, виртуализация и споделяне на ресурси между множество приложения и системи [19, 21]. В тези среди операциите по запис заемат централно място, тъй като оказват пряко влияние върху производителността, устойчивостта и предсказуемостта на цялата инфраструктура. През последните години развитието на SSD технологиите, NVMe архитектурата и високоскоростните комуникационни интерфейси доведе до значително повишаване на производителността и намаляване на латентността на физическите устройства [23, 24, 32]. Въпреки този технологичен напредък, управлението на операциите по запис в SAN среда остава сложен и трудно прогнозируем проблем.

Причината се крие в многостепенния характер на операцията по запис. В SAN архитектурата заявките преминават последователно през приложението, хостовата система, транспортния слой и системата за съхранение, като всеки от тези слоеве използва собствени механизми за буфериране, управление на входно-изходните опашки и синхронизация [13, 19, 21]. Взаимодействието между тези механизми определя крайното латентностно поведение на системата. Поради това времето за изпълнение на операциите по запис не нараства плавно с увеличаване на натоварването, а демонстрира нелинейно поведение, характеризиращо се с разширяване на латентностното разпределение и поява на редки, но екстремно забавени операции [4, 40].

Тези явления трудно могат да бъдат описани чрез традиционните усреднени показатели за производителност. Средната латентност често остава в приемливи граници, докато крайните стойности на времето за изпълнение на малка част от операциите нарастват значително и оказват непропорционално силно влияние върху работата на приложенията [3, 40]. Това е особено критично при системи за управление на бази данни, виртуализирани среди и транзакционни приложения, където дори ограничен брой силно забавени операции може да доведе до увеличаване на времето за отговор и нарушаване на качеството на обслужване.

¹ В рамките на настоящия труд под „мрежова среда за съхранение“ (Storage Area Network, SAN) се разбира архитектура за блоков достъп до споделени ресурси за съхранение, при която хостовите системи, транспортната инфраструктура и системите за съхранение функционират като логически свързана, но разпределена среда. Термините „SAN среда“ и „мрежова среда за съхранение“ се използват като еквивалентни.

Допълнително усложнение произтича от използването на кеширащи механизми, асинхронно потвърждаване на записите и различни стратегии за управление на входно-изходните опашки. Макар тези механизми да подобряват средната производителност при ниско и умерено натоварване, те могат да маскират натрупването на заявки до момент, в който системата достигне критичен режим и крайната латентност нарасне рязко [4, 40, 54]. При висока степен на паралелизъм взаимодействието между хостовата система, транспортния слой и системата за съхранение може да доведе до каскадно натрупване на заявки във входно-изходните опашки, без да е налице очевидно претоварване на отделен хардуерен компонент.

С развитието на съвременните SAN архитектури оптимизацията на операциите по запис все повече се определя от взаимодействието между системните слоеве, а не единствено от производителността на отделните хардуерни компоненти. Повишаването на производителността на устройствата само по себе си вече не е достатъчно за осигуряване на устойчиво латентно поведение, тъй като при висока степен на паралелизъм определящи стават взаимодействията между натоварването, входно-изходните опашки и механизмите за координация.

Основният научен проблем на настоящото изследване е, че латентността на операциите по запис в SAN среда представлява системно свойство, което възниква в резултат от взаимодействието между характеристиките на натоварването, механизмите за координация и входно-изходните опашки по пътя на записите. Следователно тя не може да бъде анализирана и оптимизирана чрез разглеждане на отделните компоненти в изолация. Това обяснява защо съвременни SAN инфраструктури могат да демонстрират нестабилно и трудно предсказуемо поведение, въпреки високата номинална производителност на използваните хардуерни компоненти.

В този контекст настоящият дисертационен труд е насочен към формализиране на поведението на операциите по запис в SAN среда чрез системен модел на пътя на записите (write path). Целта е да се изследва как взаимодействието между различните слоеве на системата влияе върху устойчивостта и предсказуемостта на латентността при различни режими на натоварване. За постигането на тази цел е разработена експериментална методология, базирана на реална SAN инфраструктура, която позволява контролирано изменение на параметрите на натоварването при фиксирана системна **конфигурация** [50, 52].

Получените резултати потвърждават, че поведението на операциите по запис в SAN среда се определя в по-голяма степен от взаимодействието между механизмите за

управление на натоварването, входно-изходните опашки и координацията между отделните системни слоеве, отколкото от номиналната производителност на отделните компоненти [21]. Това обосновава разглеждането на SAN средата като цялостна система от взаимодействащи механизми, при която устойчивата оптимизация изисква системен, а не изолирано компонентен подход. На тази основа в дисертационния труд са формулирани принципи за оптимизация, насочени към ограничаване на крайната латентност чрез управление на дълбочината на входно-изходните опашки, координацията между конкурентните потоци и режимите на натоварване.

1.2. Практическа значимост и приложимост на изследването

Проблемите, свързани с операциите по запис в SAN среди, не са ограничени до теоретични или лабораторни сценарии, а се проявяват ежедневно в реални производствени инфраструктури [54]. В практиката често се наблюдава разминаване между очакваното поведение на системите за съхранение, базирано на техните номинални характеристики, и реалното им поведение при експлоатация. Това разминаване се проявява най-ясно именно при операциите по запис, където нестабилността на времето за изпълнение оказва директно влияние върху работата на приложенията [40].

Съвременните SAN среди обикновено обслужват множество независими системи и приложения, които споделят обща инфраструктура за съхранение [19]. В такива условия натоварването рядко е равномерно или статично. Вместо това то се характеризира с динамични промени, пикове на активност и различна степен на паралелизъм², които се наслагват във времето [42]. Това прави поведението на операциите по запис силно зависимо от моментното състояние на системата, а не само от нейната максимална производителност.

На практика често се наблюдават ситуации, при които системи за съхранение с висока номинална пропускателна способност демонстрират нестабилно поведение при реални натоварвания [40, 54]. В такива случаи проблемите не се дължат на недостиг на ресурси, а на начина, по който заявките за запис се натрупват и обработват по пътя от хоста до системата за съхранение. Липсата на ясна видимост върху този процес затруднява

² **Паралелизъм** – свойство на система, при което множество операции се обработват конкурентно чрез припокриване във времето, като целта е повишаване на пропускателната способност и по-ефективно използване на наличните ресурси. В контекста на входно-изходни и SAN системи паралелизмът се реализира чрез едновременно обслужване на множество заявки, координирани посредством опашки и механизми за управление на конкурентния достъп.

диагностиката и води до решения, основани на емпирични наблюдения или прекомерно оразмеряване на инфраструктурата.

Практическата значимост на настоящото изследване произтича от необходимостта да се осигури предсказуемо и стабилно поведение на операциите по запис, което е от ключово значение за съвременните приложения. В много случаи за приложенията е по-важно времето за изпълнение да бъде стабилно и ограничено в тесни граници, отколкото да се постигат високи пикови стойности на производителност. Нестабилността на записите може да доведе до блокиране на критични процеси, влошаване на качеството на услугата и затруднения при мащабиране на системите.

Допълнително значение на изследването се крие във факта, че проблемите със записите често се проявяват индиректно и не могат да бъдат лесно идентифицирани чрез стандартни инструменти за мониторинг [46]. Анализът, базиран на средни стойности, рядко разкрива реалното влияние на редките, но силно забавени операции. [3, 40]. Това създава необходимост от методологичен подход, който да разглежда поведението на записите в дълбочина и да предоставя по-надеждни критерии за оценка.

В този контекст настоящото изследване има за цел да допринесе не само към теоретичното разбиране на процесите, протичащи в SAN средите, но и към практиката по проектиране, настройка и експлоатация на системи за съхранение. Чрез систематичен анализ на операциите по запис и тяхното поведение при различни условия на натоварване, работата цели да предостави основа за по-информирани решения и по-ефективно използване на наличните ресурси.

1.3. Формулиране на проблема: защо операциите по запис остават трудни за оптимизация

Въпреки значителния напредък в производителността на съвременните системи за съхранение, поведението на операциите по запис в SAN среда продължава да бъде трудно за прогнозиране и оптимизация. Практиката показва, че увеличаването на производителността на отделните компоненти не води непременно до подобряване на латентностното поведение на системата като цяло. При определени режими на натоварване дори високопроизводителни SAN инфраструктури могат да проявят значителна вариабилност на латентността и рязко нарастване на крайните латентностни стойности. [13, 21].

Основната причина за това е, че латентността на операциите по запис представлява системно свойство, което се формира в резултат от взаимодействието между множество

независими механизми по пътя на записите. В SAN архитектурата заявките последователно преминават през хостовата система, транспортния слой и системата за съхранение, като на всяко ниво се използват собствени механизми за буфериране, управление на входно-изходните опашки и координация [13, 19, 21]. Макар тези механизми да оптимизират локалното поведение на отделните компоненти, тяхното взаимодействие може да доведе до поведение, което не може да бъде обяснено чрез анализ на всеки компонент поотделно.

Особено съществено влияние оказва степента на паралелизъм. Увеличаването на броя на едновременно изпълняваните операции първоначално подобрява използването на наличните ресурси, но след достигане на определен праг увеличава вероятността от натрупване на заявки във входно-изходните опашки. В резултат латентността започва да нараства нелинейно, като деградацията се проявява най-напред в крайната част на латентното разпределение [4].

Тази особеност не може да бъде надеждно установена чрез използване единствено на средни показатели за производителност. Докато средната или медианната латентност често остават относително стабилни, високите латентностни персентили (p95 и p99) нарастват значително, което показва появата на редки, но критични случаи на силно забавени операции [3, 40]. Именно тези операции определят устойчивостта и предсказуемостта на системата при реално натоварване.

Следователно основният научен проблем не се състои в повишаването на производителността на отделните компоненти, а в разработването на модел, който да описва и обяснява взаимодействието между натоварването, входно-изходните опашки и механизмите за координация по пътя на записите. Такъв модел следва да позволява идентифициране на условията, при които системата преминава от устойчив режим на работа към латентностна нестабилност, както и да служи като основа за формулиране на практически насоки за оптимизация.

1.4. Изследователска теза

Настоящата дисертационна работа се основава на тезата, че устойчивата оптимизация на операциите по запис в SAN среда се определя от съгласуването между характеристиките на натоварването и механизмите за управление на входно-изходните опашки по пътя на записите.

В традиционния подход към оптимизацията на системите за съхранение основното внимание се насочва към физическите носители и техните номинални характеристики, като се приема, че повишаването на производителността на носителя води до съответно подобрене на цялостното поведение на системата [19, 21]. Настоящото изследване показва, че при нарастващ паралелизъм и конкуренция този модел е недостатъчен, тъй като не отчита ефектите от натрупването на заявки и взаимодействието между опашките. В този контекст ключов фактор за стабилността и предсказуемостта на записите се явява управлението на паралелизма и контролирането на натоварването, а не единствено пропускателната способност на отделните компоненти.

Според формулираната теза поведението на операциите по запис в SAN среда се определя от взаимодействието между два основни елемента:

- профила на натоварването, включително степента на паралелизъм и характеристиките на заявките;
- механизмите за управление на входно-изходните опашки по пътя на операциите по запис и тяхното влияние върху формирането на латентността.

Това взаимодействие може да бъде интерпретирано чрез разглеждане на пътя на записите като система от последователно свързани обслужващи възли и опашки, при която крайната латентност се формира от съвкупността на времената за обслужване и времената за изчакване в отделните слоеве. В този смисъл принципите на теорията на масовото обслужване предоставят аналитична основа за изследване на натрупването на заявки и възникването на латентностни отклонения при различни режими на натоварване.

Липсата на координация между тези елементи води до натрупване на заявки и до неравномерно разпределение на времената за изпълнение на операциите по запис, включително рязко нарастване на крайните стойности на латентността, без това да бъде адекватно отразено чрез средните показатели за производителност [3, 40]. Следователно **устойчивата оптимизация не може да бъде постигната чрез изолирани подобрения на отделен компонент, а изисква системен подход, насочен към контрол на паралелизма и ограничаване на неконтролираното натрупване на заявки.**

В рамките на настоящото изследване оптимизацията се разглежда като задача за поддържане на стабилно и предсказуемо латентностно поведение чрез управление на паралелизма, ограничаване на опашковите ефекти и контрол върху крайните персентили на латентността. Поради това крайната латентност се приема като по-надежден индикатор за реалното поведение на SAN системата в сравнение със средната латентност или общата пропускателна способност. Критерият за оптимизация не е

единствено максималният постигнат капацитет, а способността на системата да поддържа устойчиво и предсказуемо поведение при нарастващо натоварване.

Настоящото изследване защитава тезата, че **чрез аналитично моделиране на пътя на записите и целенасочено управление на входно-изходните опашки могат да бъдат идентифицирани режими на работа, които минимизират проявлението на крайна латентност и повишават устойчивостта на системата без необходимост от допълнителни хардуерни ресурси.** Тази теза се проверява експериментално чрез контролирана промяна на параметрите на натоварването и анализ на получените латентностни разпределения.

1.5. Методология на изследването

Изследването, представено в настоящата дисертационна работа, следва експериментално-аналитична методология, насочена към изследване на поведението на операциите по запис в мрежови среди за съхранение при контролирани условия [50, 52]. Основната цел на избрания подход е да се изолира влиянието на отделни фактори по пътя на записите и да се анализира тяхното взаимодействие, без резултатите да бъдат повлияни от странични или трудно контролируеми променливи.

Методологията се основава на последователно преминаване от концептуален анализ към експериментална валидация. Разработеният системен модел се използва и като аналитична рамка за интерпретация на наблюдаваните латентностни ефекти, като пътят на записите се разглежда като съвкупност от взаимодействащи обслужващи механизми и опашки. Това позволява резултатите да бъдат свързани с принципите на теорията на масовото обслужване и да бъдат формулирани практически насоки за оптимизация на операциите по запис.

Паралелно с концептуалния модел е използвана и аналитична интерпретация на пътя на записите, при която SAN средата се разглежда като система от взаимодействащи обслужващи възли и входно-изходни опашки. Този подход позволява наблюдаваните латентностни ефекти да бъдат анализирани чрез принципите на теорията на масовото обслужване, като крайната латентност се интерпретира като резултат от взаимодействието между времето за обслужване и времето за изчакване в различните слоеве на системата. По този начин моделът се използва не само за описание на поведението на записите, но и като основа за формулиране на практически насоки за тяхната оптимизация.

В рамките на концептуалния анализ се идентифицират архитектурните и логическите елементи на SAN средата, които имат пряко влияние върху поведението на операциите по запис, както и механизмите за управление на паралелизма и опашките [2, 4].

Основният метод за проверка на формулираната изследователска теза е експерименталното измерване. Проведените експерименти са проектирани така, че да изследват влиянието на отделни параметри върху поведението на операциите по запис при фиксирана и ясно описана тестова среда. По време на експериментите се използват контролирани натоварвания, при които се променя само един параметър, докато останалите условия се запазват непроменени, което позволява еднозначна интерпретация на получените резултати и намалява риска от погрешни изводи [3, 50, 52].

Особено внимание е отделено на избора на метрики за анализ. Вместо да се разчита единствено на усреднени показатели, изследването използва метрики, които отразяват поведението на крайните стойности на латентността. Този подход позволява по-точно описание на реалното влияние на операциите по запис върху приложенията и съответства на изследователската теза [6, 40].

За да се осигури възпроизводимост на резултатите, експерименталната среда е подробно описана, а използваните параметри са фиксирани и ясно дефинирани. Методологията не е обвързана с конкретен производител или модел на система за съхранение, което позволява резултатите да бъдат интерпретирани в по-широк контекст и да бъдат приложими към различни SAN архитектури [50, 52].

1.6. Ограничения и обхват на изследването

Настоящото изследване е фокусирано върху анализа и оптимизацията на операциите по запис в SAN среди при ясно дефинирани условия и цели. С оглед на сложността и многослойния характер на системите за съхранение, обхватът на изследването е съзнателно ограничен, така че да позволи задълбочен и възпроизводим анализ на ключовите фактори, влияещи върху поведението на записите.

На първо място изследването се концентрира върху **операциите по запис**, като операциите по четене не са обект на детайлен анализ. Това ограничение е мотивирано от факта, че записите имат по-сложно поведение, включват повече междинни механизми за управление и оказват по-голямо влияние върху стабилността на системата при реални натоварвания. В сравнение с операциите по запис, операциите по четене показват по-

ограничена нелинейност и по-слабо изразена системна зависимост, въпреки тяхната роля за цялостната производителност.

Изследването не цели да постигне или сравнява максималната възможна пропускателна способност на използваната инфраструктура. Вместо това фокусът е поставен върху предсказуемостта и стабилността на поведението, измерени чрез времето за изпълнение на крайните стойности (tail latency³) на операциите по запис [4, 40]. Този избор отразява практическите изисквания на приложения, при които стабилното време за реакция е по-важно от постигането на пикови стойности на производителност.

В рамките на експериментите се използва контролирана тестова среда, при която се изследва влиянието на отделни параметри при фиксирани останали условия. Резултатите следователно са валидни в контекста на описаната методология и не претендират за универсална приложимост към всички възможни конфигурации на SAN среди. Целта на изследването не е да предложи универсални настройки, а да идентифицира закономерности и зависимости, които могат да бъдат използвани като основа за информирани решения при проектиране и оптимизация на системи за съхранение.

Настоящата работа не разглежда детайлно вътрешната реализация на конкретни търговски продукти или проприетарни алгоритми, използвани от отделни производители. Анализът е насочен към архитектурни и поведенчески аспекти, които са общи за широк клас SAN решения. По този начин изследването запазва своята независимост от конкретни реализации и позволява резултатите да бъдат интерпретирани в по-широк контекст.

Изложените ограничения не намаляват значимостта на получените резултати, а ясно очертават рамките, в които те следва да бъдат интерпретирани. Те създават стабилна основа за коректна оценка на направените изводи и създават предпоставки за бъдещи изследвания, насочени към разширяване на анализа към други типове натоварвания, архитектури и режими на експлоатация.

ГЛАВА 2. СИСТЕМЕН МОДЕЛ НА ОПЕРАЦИИТЕ ПО ЗАПИС В МРЕЖОВИ СРЕДИ ЗА СЪХРАНЕНИЕ (SAN)

Настоящата глава въвежда системен модел на операциите по запис в SAN среда, който представя пътя на записите като последователност от взаимосвързани компоненти

³ **Tail latency** (закъснение на опашката) – времето за реакция при най-бавните заявки в системата (обикновено горните 1% или 0.1%). Докато средното закъснение показва типичното представяне, tail latency измерва „лошия сценарий“, който често определя цялостното усещане за скорост при комплексни услуги.

и механизми за обработка на входно-изходни заявки. Моделът обхваща основните елементи на системата - хост среда, транспортен слой и система за съхранение - и разглежда взаимодействието между тях като определящ фактор за формирането на латентността.

В рамките на модела пътят на записите се интерпретира като система от опашки и механизми за управление, чрез които заявките се буферират, обработват и координират при наличие на паралелизъм и конкурентен достъп до ресурси. Това представяне позволява да се анализира поведението на системата не само на ниво отделни компоненти, а като резултат от тяхното съвместно функциониране [1, 4, 21].

Този подход дава възможност да се обоснове, че стабилността и предсказуемостта на латентността не могат да бъдат адекватно описани чрез изолирани метрики или характеристики на отделни компоненти, а следва да се разглеждат като системно свойство, възникващо от взаимодействието между опашките и механизмите по пътя на операциите по запис.

2.1. Цел на главата

Целта на настоящата глава е да въведе единен и последователен модел на пътя на записите в SAN среда, който да служи като концептуална основа за анализа и експерименталното изследване в следващите части на дисертационния труд. Вместо да разглежда отделни технологии или компоненти изолирано, главата представя пътя на записа като цялостна система, съставена от взаимосвързани елементи, всеки от които участва в обработката, буферирането и управлението на заявките за запис [2, 21].

Основният фокус е поставен върху опашките и механизмите за управление, които се появяват по пътя на записите - от хост системата, през транспортния слой, до контролерите и физическата среда на системата за съхранение. Тези опашки не са вторичен ефект, а фундаментален елемент от архитектурата на SAN средите [1, 4, 8]. Те определят начина, по който системата реагира на паралелизъм, променливо натоварване и конкурентен достъп до ресурси.

Чрез формулиране на логически модел на пътя на записите, главата цели да покаже, че нестабилността на латентността не е следствие от единичен „тесен участък“⁴ [1, 4, 40], а резултат от натрупване и взаимодействие на заявки в различни точки на системата. В

⁴ „Тесен участък“ (bottleneck) обозначава елемент или ресурс в системата, чиято ограничена пропускателна способност доминира общото поведение и ограничава производителността. В сложни входно-изходни и SAN среди обаче латентностната нестабилност често възниква не от единичен bottleneck, а от взаимодействието между няколко компонента и механизми за координация.

този контекст латентността се разглежда не като фиксирана характеристика на устройството за съхранение, а като динамично свойство на цялата SAN инфраструктура [2, 3].

Представеният модел служи като основа за систематично аналитично разграничаване на ролята на хост страната, транспортния слой и системата за съхранение при формирането на поведението на операциите по запис. По този начин се създава ясна връзка между архитектурните решения, параметрите на натоварването и характеристиките на латентността [3, 50], които впоследствие се анализират експериментално. Главата подготвя читателя за последващите експериментални изследвания, като дефинира общ език, понятия и гледна точка, чрез които резултатите могат да бъдат интерпретирани последователно и аргументирано.

2.2. Еволюция на тесните места (bottlenecks) при операциите по запис ⁵

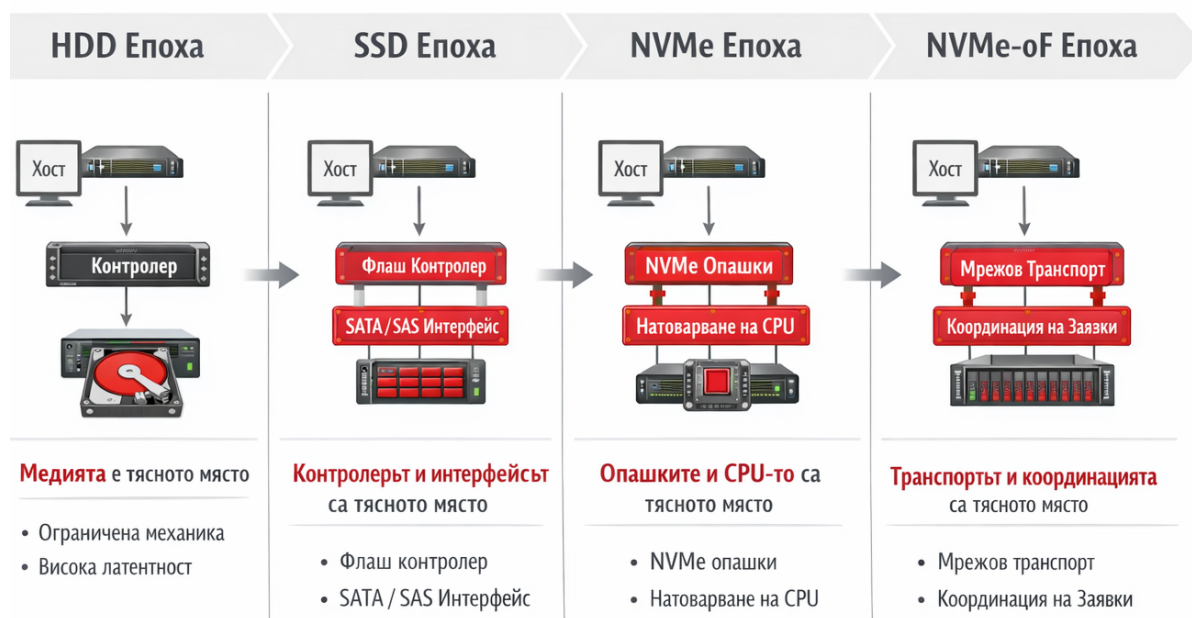
Еволюцията на технологиите за съхранение води до съществени промени не само в абсолютната производителност на системите, но и в мястото и характера на доминиращите ограничения при записите. Вместо bottleneck-ът да бъде елиминиран, той постепенно се измества между различни слоеве на архитектурата [4, 2, 21] като всяка технологична стъпка напред решава конкретен клас проблеми, но въвежда нови източници на нестабилност.

В този раздел bottleneck-ът се разглежда не като статично „тясно място“, а като динамично свойство на системата, което зависи от архитектурния модел, начина на управление на паралелизма и взаимодействието между отделните слоеве по пътя на записите. Проследяването на тази еволюция създава необходимия контекст за анализа на латентността и нейните вариации в съвременни SAN среди [4, 40].

Фигура 2.2 илюстрира този преход като показва как доминиращият bottleneck се измества между различни слоеве при преминаване от HDD към NVMe-over-Fabrics архитектури [23, 25, 32, 34]. В HDD епохата ограничението е в самата физическа медия; при SSD се измества към контролера и интерфейса; при NVMe водеща става ролята на опашките и CPU/хост стека; а при NVMe-oF ключови се оказват транспортът и координацията между няколко независими домейна. Фигурата илюстрира, че с

⁵ Темата за еволюцията на тесните места (bottlenecks) при операциите по запис е разработена по-подробно в ранна версия на настоящия дисертационен труд. В процеса на развитие и разширяване на изследването тя е интегрирана в по-общ системен модел и целенасочено редуцирана до концептуален контекст, с цел фокусиране върху експерименталния анализ и аналитичния синтез.

напредването на технологиите ограничението при операциите по запис се измества от физическите характеристики на носителя към системни и координационни ефекти, което обяснява защо локалните оптимизации не водят непременно до предсказуемо глобално поведение.



Фигура 1. Еволюция на Bottleneck

2.2.1. HDD епоха (≈1980–2005): bottleneck-ът е „закотвен“ в медията

В епохата на твърдите дискове bottleneck-ът при записите е ясно дефиниран и архитектурно стабилен. Латентността се доминира от механичните процеси на позициониране на главата и ротация на плочите, които се измерват в милисекунди. Дори при оптимизирани enterprise HDD устройства минималното време за произволен запис остава в диапазона от няколко до десетки милисекунди [19, 28].

Тази физическа доминация определя цялостния архитектурен модел. Контролерът, интерфейсът и софтуерният стек имат сравнително малък принос към общата латентност и не могат съществено да повлияят на крайното време за запис. Оптимизацията се свежда до „подреждане около физиката“ - планиране и пренареждане на заявки с цел минимизиране на механичните движения, без реална възможност bottleneck-ът да бъде премахнат [14, 15].

В резултат HDD-базираните системи демонстрират относително предсказуемо поведение. Bottleneck-ът е един, физически обусловен и устойчив във времето. Макар

латентността да е висока, нейният характер е стабилен, а вариациите са пряко свързани с механичните ограничения на носителя [28].

2.2.2. SSD епоха (≈2005–2013): bottleneck-ът се измества към контролера и интерфейса

С навлизането на SSD технологиите механичният достъп отпада и физическата медия престава да бъде доминиращият лимитиращ фактор. Латентностите спадат от милисекунди към стотици или десетки микросекунди [23], [26], което представлява качествена промяна в пътя на записите.

Тази промяна обаче не води до изчезване на bottleneck-a, а до неговото преместване. На преден план излизат логиката за управление на флаш паметта и ограниченията на наследените интерфейси и командни модели. SSD контролерът поема сложни функции като адресна трансляция (FTL), управление на износването и фонові процеси (garbage collection), които въвеждат вътрешни опашки и моментни задържания [23, 25, 27].

Паралелно с това първите SSD устройства използват SATA и SAS интерфейси, проектирани за механични носители [26, 32]. Ограниченията в броя и дълбочината на опашките, както и протоколният overhead, започват да се проявяват като архитектурни bottleneck-и. В този период bottleneck-ът вече не е „естествено“ физически, а се формира като резултат от конкретни архитектурни решения.

Характерно за тази епоха е появата на вариации в латентността. Средните стойности са значително по-ниски от HDD, но крайните персентили започват да се влияят от вътрешното състояние на контролера и от активните фонові процеси, което прави поведението по-малко предсказуемо [25, 40].

2.2.3. NVMe епоха (≈2013–2016): bottleneck-ът в опашките и CPU/хост стека

NVMe архитектурата е проектирана като директен отговор на ограниченията на SSD устройствата върху наследени интерфейси. Чрез директна интеграция с PCI Express и масивен модел на опашки NVMe премахва значителна част от протоколния overhead и позволява ефективно използване на паралелизма на флаш паметта и многото CPU ядра [16, 32].

Тази архитектурна промяна води до ново преместване на bottleneck-a. Физическата медия и интерфейсът рядко са лимитиращият фактор; вместо това управлението на паралелизма в хост системата се превръща в критичен елемент. CPU ресурсите, драйверът

и конфигурацията на опашките започват да определят както максималната производителност, така и стабилността на латентността [13, 16, 35].

Типичен симптом за тази епоха е нелинейното поведение при високи IOPS стойности. Средната латентност може да остане ниска, докато крайните персентили (p95, p99⁶) нарастват значително при прекомерна дълбочина на опашките или несъгласувано натоварване. Bottleneck-ът се проявява не като „твърда стена“, а като нестабилност и разширяване на разпределението на латентността [4, 40].

2.2.4. NVMe-over-Fabrics (≈2016–днес): bottleneck-ът като координационен проблем

NVMe-over-Fabrics разширява NVMe модела извън границите на локалната система, като пренася същия паралелизъм и модел на опашките в мрежова среда. Това добавя нов домейн с независими механизми за управление - транспортният слой, реализиран чрез Fibre Channel, TCP или RDMA [33, 34, 38].

В резултат пътят на записите се превръща в система от няколко автономни регулатори: хост, транспорт и сториџ контролери. Всеки от тях управлява потока локално, без глобална синхронизация. Bottleneck-ът вече не е локализиран в един компонент, а е разпределен и динамичен, възникващ от взаимодействието между домейните [22, 40].

Практически това означава, че системата може да демонстрира висока пропускателна способност, докато същевременно проявява редки, но критични забавяния при записи. Този тип поведение се улавя най-добре чрез анализ на разпределението на латентността и крайните персентили, а не чрез средни показатели [39, 40].

В най-сбит вид еволюцията на доминиращите ограничения (bottleneck) при системите за съхранение може да бъде обобщена по следния начин: при HDD те се определят основно от физическата медия и имат относително единен и предсказуем характер; при SSD bottleneck-ът се измества към контролера и интерфейса и придобива вариативен, архитектурно обусловен характер; при NVMe доминиращи стават опашките и взаимодействието с CPU и хост стека, което води до нелинейно поведение при нарастващ паралелизъм; при NVMe-over-Fabrics ограниченията вече произтичат от координацията между множество домейни, което въвежда системна нестабилност [4, 32, 33, 40].

⁶ Персентилите на латентността (p95, p99) са разгледани подробно в следващите глави на дисертацията, където се въвеждат формалните метрики и се извършва количествен анализ на тяхното поведение при различни експериментални сценарии.

Този преход е пряко свързан с темата на настоящата дисертация. В съвременните SAN среди оптимизацията на записите не може да бъде сведена до избор на „по-бърза“ среда за съхранение, тъй като доминиращите ограничения все по-често произтичат от управлението на паралелизма и опашките, както и от съгласуването между слоевете по пътя на записа. Поради това в следващите раздели пътят на записите се разглежда като система от опашки и регулатори, която позволява едновременно описание както на производителността, така и на нестабилността на латентността [4, 40].

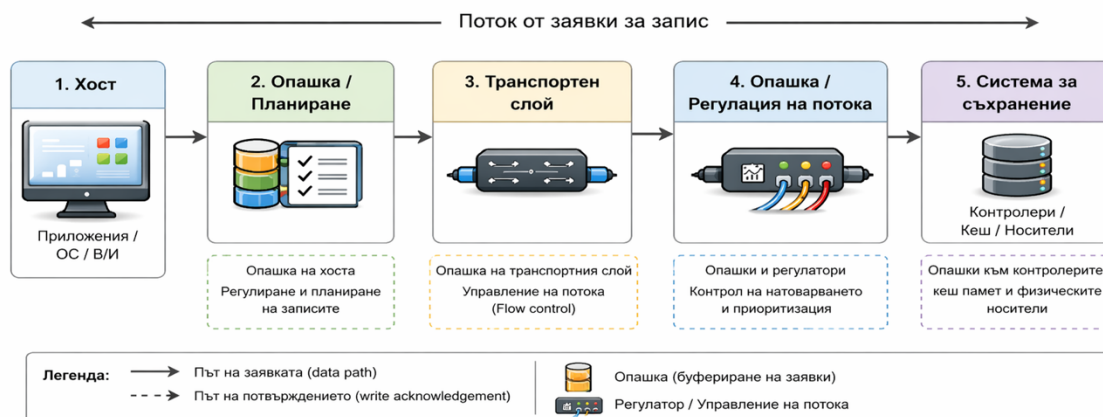
2.3. Пътят на записите като система

В реалните SAN среди операциите по запис не се изпълняват изолирано, а като поток от конкурентно генерирани заявки, които се транспортират и обработват паралелно. Поради това пътят на записите не може да бъде разглеждан като линейна последователност от стъпки, а като система от взаимодействащи компоненти, в която множество записи споделят общи ресурси и механизми за управление [2, 4].

Този системен характер на пътя на записите е ключов за разбирането на поведението на латентността. Докато при единична операция по запис времето за изпълнение може да бъде описано като сума от последователни закъснения, при поток от паралелни записи времето за изпълнение на всяка отделна операция зависи от моментното състояние на системата. Това състояние се определя от броя на активните заявки, дълбочината на опашките и начина, по който отделните компоненти планират и обслужват конкурентния достъп [4, 8].

От гледна точка на системния модел пътят на записите може да бъде разглеждан като верига от опашки и регулатори, разположени на различни нива на SAN инфраструктурата. Всеки компонент по този път приема входящи заявки, временно ги задържа, подрежда ги според определени правила и ги предава към следващия елемент. Този процес не е синхронизиран глобално и не следва единен алгоритъм за управление, което води до сложни и често нелинейни ефекти при натоварване [2, 4, 7].

За по-голяма яснота, Фигура 2 представя абстрактен системен модел на пътя на записите като верига от взаимодействащи опашки и регулатори на различните нива на мрежовата среда за съхранение. Моделът подчертава, че поведението на операциите по запис се формира от взаимодействието между множество опашки и механизми за управление, а не от изолиран bottleneck в отделен компонент [4, 40].



Фигура 2. Концептуален модел на пътя на записите в мрежова среда за съхранение, илюстриращ разпределението на опашките и регулаторите по различните нива на системата

От страна на хоста записите се генерират от приложенията и се подават към операционната система като входно-изходни заявки. Още на това ниво възниква първичен паралелизъм, тъй като множество процеси и нишки могат едновременно да инициират операции по запис. Операционната система агрегира тези заявки и ги подава към по-ниските слоеве чрез собствени опашки и механизми за планиране. Така още преди заявките да напуснат хоста, те вече са част от поток, в който редът и времето за обработка зависят от текущото натоварване [13, 14].

Транспортният слой в SAN среда добавя следващото ниво на системна сложност. Той осигурява надежден пренос на блокови команди между хоста и системата за съхранение, но същевременно въвежда собствени правила за управление на потока и достъпа до ресурси. При наличие на множество едновременни записи транспортният слой не функционира като пасивен канал, а като активен участник в управлението на заявките. В определени ситуации това води до временни задържания, които не са свързани с изчерпване на пропускателната способност, а с начина на арбитриране и разпределение на достъпа [22, 33].

Системата за съхранение представлява следващият ключов елемент от пътя на записите. Тя приема заявки от транспортния слой, разпределя ги между контролерите и ги обработва чрез вътрешни механизми за кеширане, планиране и последващо записване върху физическата среда. И тук записите се обработват като поток, а не като независими операции. Вътрешните опашки и алгоритми за управление имат за цел да оптимизират общата ефективност, но те също допринасят за вариации във времето за изпълнение на отделните записи [23, 25, 54].

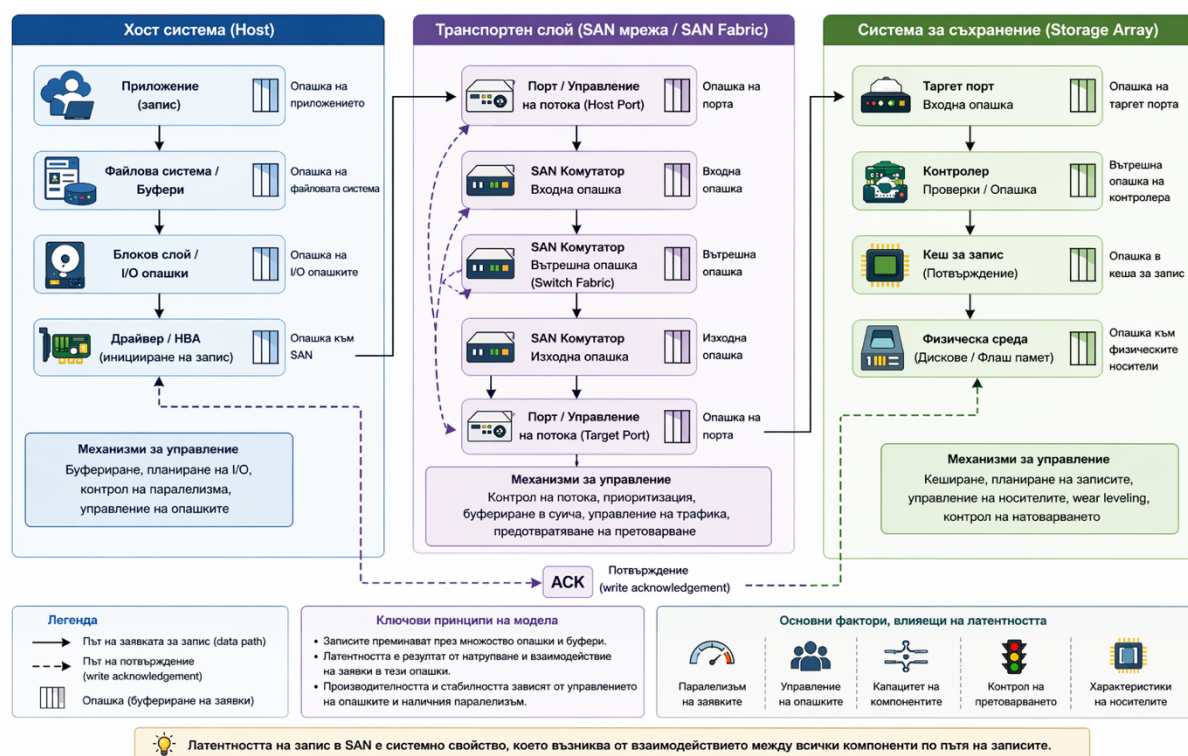
Системният модел на пътя на записите показва, че латентността⁷ е резултат от натрупването на малки закъснения в множество точки, а не от доминираща бавна операция. При ниско натоварване тези закъснения остават почти незабележими. С увеличаване на паралелизма обаче взаимодействието между опашките по различните нива може да доведе до рязко увеличаване на времето за изпълнение на част от записите, без това да бъде съпроводено от видими признаци за претоварване на отделен компонент [2, 4].

Важно следствие от този модел е, че поведението на пътя на записите не може да бъде адекватно описано чрез статични характеристики или чрез анализ на единични компоненти. Стабилността на записите зависи от баланса между генерираното натоварване и способността на системата да го абсорбира и обработва равномерно във времето. Нарушаването на този баланс води до натрупване на заявки и до нестабилна латентност, която се проявява първо в крайните стойности на времето за изпълнение [4, 40].

В контекста на настоящото изследване пътят на записите в SAN среда се разглежда като основен обект на анализ. Чрез разграждане на този път на логически компоненти и идентифициране на ключовите точки на опашкуване се създава основа за систематично изследване на причините за нестабилно поведение. Този модел позволява ясно свързване между архитектурните решения, параметрите на натоварването и наблюдаваните характеристики на латентността, което е предпоставка за последващата експериментална част на дисертационния труд.

За да бъде тази концепция формализирана и използвана като обща референтна рамка в следващите раздели, пътят на записите е представен чрез обобщен логически модел. Моделът визуализира последователността на обработка на операциите по запис и местата, в които възниква натрупване на заявки, като обхваща трите основни домейна на мрежовата среда за съхранение - хост система, транспортен слой и система за съхранение. В този контекст ролята на опашките се разглежда като определящ фактор за поведението на латентността при конкурентни записи [4].

⁷ **Латентност (latency)** – времето между иниципирането на входно-изходна операция и нейното завършване, включващо всички междинни закъснения, възникващи по пътя на заявката през хоста, транспортния слой и системата за съхранение. В този смисъл латентността се разглежда като системна характеристика, а не като свойство на отделен компонент.



Фигура 3. Логически модел на пътя на записите и точки на опашкуване

Фигурата представя логически модел на пътя на записите в мрежова среда за съхранение, организиран в три ясно разграничени домейна: хост система, транспортен слой и система за съхранение. Всеки от тези домейни съдържа компоненти, които участват в обработката на операциите по запис и въвеждат собствени точки на опашкуване, обозначени във фигурата с визуални маркери [2, 4].

Моделът има за цел да илюстрира, че латентността на записите се формира в резултат от взаимодействието между множество опашки по целия път на заявките, а не от доминираща бавна операция или единичен „тесен участък“ [4, 40].

В рамките на представения модел е необходимо ясно разграничение между времето за изпълнение на операциите по запис, видимо за приложението, и вътрешното време за обработка в отделните слоеве на системата. От гледна точка на приложението операцията по запис се счита за завършена в момента, в който бъде получено потвърждение за успешно изпълнение, като именно това време определя наблюдаваната латентност и е обект на измерване в настоящото изследване [13, 54].

Вътрешните процеси по буфериране, планиране и физическо записване могат да протичат частично асинхронно спрямо потвърждението към хоста и не са директно видими за приложението. Въпреки това тяхното поведение влияе върху натрупването на заявки и стабилността на потока от потвърждения при конкурентни записи [23, 40].

2.4. Опашки от страна на хоста

2.4.1. Ролята на хост системата в пътя на записите

В архитектурата на SAN средите хост системата често се разглежда единствено като източник на входно-изходни заявки, докато основният фокус се поставя върху транспортния слой или системата за съхранение. В действителност обаче хостът изпълнява ролята на първия и фундаментален регулатор на пътя на записите, като именно на това ниво се формира първоначалното поведение на потока от заявки [13, 14].

Хост системата определя не само какъв обем от записи се генерира, но и как те се групират, подреждат и подават във времето към SAN инфраструктурата. Чрез механизмите на операционната система, файловата система и блоковия слой, хостът въвежда първите опашки, които задават броя на едновременно активните операции по запис. Тези опашки не са пасивен страничен ефект, а съзнателен архитектурен механизъм за управление на паралелизма и натоварването [13, 15, 16].

От гледна точка на целия път на записите хостът задава началните условия, при които работят всички следващи компоненти. Интензивността на заявките, тяхната паралелност и начинът им на подаване към транспортния слой определят дали SAN средата ще функционира в стабилен режим или ще навлезе в състояние на натрупване на заявки. Дори при високопроизводителна транспортна и сторидж инфраструктура, неправилно управляваният поток от записи от страна на хоста може да доведе до нестабилна латентност и крайни стойности във времето за изпълнение [40, 4].

Поради това анализът на опашките започва от хост системата. Разглеждането на пътя на записите от източника позволява да се проследи как началният паралелизъм се формира и как се пренася към следващите слоеве на SAN архитектурата [2, 4].

2.4.2. Генериране на заявки за запис в приложението

Пътят на записите в SAN среда започва на ниво приложение, където се формира първоначалният поток от заявки. Архитектурата на приложението и начинът му на изпълнение определят началната степен на паралелизъм, която се пренася към всички по-ниски слоеве на системата [13, 20].

Съвременните приложения често използват множество процеси или нишки, които едновременно генерират заявки за запис. В резултат възниква конкуренция за ресурси още преди заявките да достигнат операционната система. Макар на това ниво да не съществуват формално дефинирани входно-изходни опашки, забавянията в изпълнението

водят до натрупване на заявки, което може да се разглежда като първична форма на опашкуване [13, 14].

Интензивността и времевата структура на заявките зависят от характера на приложението и неговия модел на достъп до данни. При приложения с висока степен на паралелизъм натоварването често е неравномерно, с кратки периоди на висока активност, последвани от по-слабо натоварване. Тези пикове се пренасят към по-ниските слоеве и влияят върху формирането на опашките [52].

Важно е, че на ниво приложение липсва механизъм за глобално управление на пътя на записите. Приложението не разполага с информация за състоянието на SAN инфраструктурата и не адаптира поведението си спрямо текущото натоварване. В резултат заявките се генерират независимо от способността на системата да ги обработва равномерно [21].

Този етап задава входните параметри за всички последващи механизми за управление и опашкуване. Поради това поведението на латентността при конкурентни записи е пряко свързано с начина, по който приложенията генерират и подават заявки [40].

2.4.3. Операционна система и файлова система като буфер

След генерирането на заявки на ниво приложение, те се обработват от операционната система и файловата система, които функционират като първи централен буфер в хост системата. Този слой агрегира заявки от множество процеси и ги подготвя за подаване към блоковия слой [13, 14].

Операционната система управлява заявките чрез механизми за синхронизация, кеширане и управление на паметта. При файлово-базиран достъп файловата система буферира записите и въвежда временно задържане преди тяхното преобразуване в блокови операции. Това поведение подобрява ефективността, но създава разлика между момента на генериране на заявката и нейното реално подаване към по-ниските слоеве [14, 15].

Буферирането изглажда краткотрайни пикове в натоварването чрез обединяване и реорганизация на записите. При продължително или силно паралелно натоварване обаче буферите се запълват и се формират реални опашки, което води до увеличаване на времето за изчакване [2, 4].

Поведението на този слой зависи от наличните ресурси и текущото натоварване, като фактори като оперативна памет и конкуренция между процесите влияят върху темпа на подаване на заявки. В резултат идентични операции могат да изпитат различно

закъснение при различни условия. Операционната система въвежда и първата форма на асинхронност в пътя на записите, при която операцията може да изглежда завършена от гледна точка на приложението, докато реалното ѝ изпълнение все още не е напуснало хост системата [13, 14, 20].

По този начин операционната система и файловата система функционират като активен регулатор на потока от записи, определяйки темпа на подаване към блоковия слой и влияейки върху началната дълбочина на опашките. Част от латентността се формира още на този етап, преди заявките да достигнат SAN транспорта [4, 13].

2.4.4. Блоков слой и планиране на входно-изходните заявки

След обработката на заявките от операционната и файловата система, те достигат до блоковия слой, който представлява междинна абстракция между логическите операции по запис и физическата реализация в SAN средата. На това ниво заявките се трансформират в блокови операции и се подготвят за подаване към транспортния слой [13, 16].

Блоковият слой въвежда формализирани входно-изходни опашки и механизми за планиране, чрез които се регулират паралелизъмът и редът на обработка на заявките. Дълбочината на тези опашки определя броя на едновременно активните операции и задава степента на натоварване, която хост системата подава към следващите слоеве [16].

При ниска дълбочина на опашките броят на заявките „в полет“ е ограничен, което води до непълно използване на наличните ресурси. С увеличаване на дълбочината се създават условия за по-добро припокриване на операциите и повишаване на пропускателната способност. След определена точка обаче допълнителното увеличаване на дълбочината не води до съществено подобрение, а увеличава времето за изчакване на заявките [4, 2].

В този режим част от заявките се обслужват бързо, докато други остават задържани значително по-дълго време, което води до разширяване на разпределението на латентността. Средните стойности могат да останат стабилни, докато крайните персентили нарастват чувствително. Това поведение е особено изразено при конкурентни натоварвания, при които множество потоци споделят общите опашки [40].

От гледна точка на пътя на записите блоковият слой задава началното натоварване за транспортния слой и системата за съхранение. Несъответствие между дълбочината на опашките и реалните възможности на инфраструктурата води до натрупване на заявки по целия път на записите [4, 16].

Поради това блоковият слой представлява ключова точка за формиране на латентността и естествена отправна точка за анализ на поведението на записите в SAN среда.

2.4.5. Драйвер и хостов адаптер (HBA): границата между хост и SAN

Драйверът и хостовият адаптер (Host Bus Adapter, HBA) представляват последния етап от пътя на записите в хост системата и точката на преход към SAN инфраструктурата. На това ниво блоковите заявки се преобразуват в команди, съвместими с използвания транспортен протокол, и се подават към мрежовата среда за съхранение [22, 35].

Драйверът приема заявки от блоковия слой и ги организира в опашки, чрез които се контролира броят на едновременно активните операции към HBA. По този начин той въвежда допълнително ограничение върху паралелизма, зададен от по-горните слоеве [16, 35].

HBA от своя страна реализира хардуерното управление на входно-изходните операции и поддържа собствени опашки с ограничена дълбочина. Това го превръща в последния регулатор на потока от заявки от страна на хоста и във физическа граница за броя операции, които могат да бъдат подадени към SAN средата [22].

При несъответствие между интензивността на заявките и възможностите на HBA се наблюдава натрупване в драйверните или хардуерните опашки. Това води до увеличаване на времето за изчакване, което вече не може да бъде повлияно от по-горните слоеве и се отразява директно върху латентността на записите [4].

От гледна точка на анализа на латентността този слой е критичен, тъй като представлява последната точка, в която хост системата може да контролира паралелизма, преди заявките да навлязат в SAN инфраструктурата. Натрупването на заявки на това ниво се пренася директно към транспортния слой и системата за съхранение.

По този начин драйверът и HBA играят ключова роля в баланса между пропускателна способност и стабилност на латентността, като могат както да ограничават прекомерния паралелизъм, така и да се превърнат в източник на натрупване при несъгласувани настройки [4, 40].

2.4.6. Паралелизъм: кога помага и кога започва да вреди

Паралелизмът е основен механизъм за повишаване на производителността при операциите по запис в SAN среда. Чрез едновременното изпълнение на множество заявки се постига по-добро използване на ресурсите и припокриване на отделните етапи по пътя

на записите. В многослойна архитектура паралелизъмът не се управлява централизирано, а се формира и ограничава последователно на различни нива – от приложението, през хост системата, до хардуерните механизми на адаптера. Поради това неговият ефект зависи от съгласуваността между тези слоеве [4].

При ниско ниво на паралелизъм броят на едновременно активните заявки е ограничен, което води до непълно използване на наличните ресурси. В този режим латентността е относително стабилна, но пропускателната способност остава ниска.

С увеличаване на паралелизма системата достига режим на ефективна експлоатация на ресурсите, при който множество заявки се обработват едновременно и се постига висока пропускателна способност при относително стабилно време за изпълнение [7].

При по-нататъшно увеличаване на паралелизма се достига до режим, в който броят на заявките надхвърля способността на инфраструктурата да ги обработва равномерно. Това води до натрупване на заявки в опашките на различни нива от пътя на записите и до увеличаване на времето за изчакване. В този режим част от заявките се обслужват бързо, докато други се задържат значително по-дълго, което води до разширяване на разпределението на латентността и появата на крайни стойности. Това поведение не може да бъде адекватно описано чрез средни показатели и се проявява най-ясно при конкурентни натоварвания [3, 40]. При прекомерен паралелизъм опашките престават да бъдат механизъм за оптимизация и се превръщат в основен фактор за нестабилност на латентността. В резултат системата може да поддържа висока пропускателна способност, но с значителни вариации във времето за изпълнение.

Следователно паралелизъмът не следва да се разглежда като еднозначно положителен параметър. Неговият ефект зависи от баланса между решенията на различните слоеве и от способността на системата да координира конкурентните заявки. Нарушаването на този баланс води до нарастване на вариативността и на крайните стойности на латентността [4, 40].

2.5. Опашки и управление в транспортния слой (fabric)

Транспортният слой в SAN архитектурата осигурява пренасянето на блокови заявки между хоста и системата за съхранение. Въпреки че често се възприема като пасивна комуникационна среда, той прилага механизми за управление на потока и синхронизация, които оказват влияние върху времевото поведение на операциите по запис [22, 33].

В контекста на пътя на записите транспортният слой функционира като междинен регулатор между хост системата и системата за съхранение, които работят с различна динамика. Хостът генерира заявки според приложното натоварване и локалните механизми за управление, докато сторидж системата ги обработва съгласно вътрешни ограничения и опашки. Транспортният слой посредничи между тези два процеса, което води до временни задържания и вариации във времето за пренос [21, 4].

Аналитично може да се разграничат два източника на латентност: свързани с пропускателната способност и свързани с управлението на потока. При първия тип забавянето произтича от физически ограничения и се проявява относително равномерно. При втория тип транспортният слой може да работи в рамките на номиналния си капацитет, но да въвежда временни ограничения с цел запазване на коректността и устойчивостта на комуникацията. Тези ефекти се проявяват като вариации във времето за доставка на заявките и не се откриват лесно чрез усреднени показатели [40, 4].

В резултат транспортният слой допринася за латентността не само чрез добавяне на време за пренос, а чрез вариации, които зависят от моментното натоварване и взаимодействието между компонентите. Тези вариации могат да се натрупват и да се пренасят към следващите слоеве, където се комбинират с вътрешните опашкови механизми на системата за съхранение.

Многопътните конфигурации (multipath) са широко използвани в SAN средите с цел повишаване на надеждността и използването на ресурсите [54]. Разпределянето на заявките между няколко пътя позволява паралелна обработка, но въвежда допълнителна вариативност, тъй като отделните пътища могат да бъдат в различно състояние в даден момент. При по-високо натоварване това води до разминаване във времето на пристигане на заявките към системата за съхранение. Част от заявките достигат по-бързо, докато други се задържат по-дълго, което може да доведе до неравномерно натоварване на вътрешните опашки.

От тази гледна точка транспортният слой не следва да се разглежда единствено като средство за пренос, а като фактор, влияещ върху стабилността на латентността. В рамките на настоящото изследване той се разглежда като част от общия системен контекст, без да се извършва детайлно изолиране на неговите вътрешни механизми. Това позволява транспортният слой да бъде интерпретиран като свързващ елемент в системния модел, който допринася за формирането на опашкови ефекти и подготвя анализа на поведението на системата за съхранение [4, 40].

2.6. Опашки в контролерите и към физическата среда

2.6.1. Входен (front-end) слой в системата за съхранение

След като заявките за запис преминават през хост системата и транспортния слой, те навлизат в системата за съхранение, където се поемат първоначално от входния (front-end) слой. Този слой е неразделна част от софтуерната архитектура на системата за съхранение и изпълнява ролята на първа вътрешна логическа инстанция за приемане и управление на входящия поток от операции по запис [21, 54].

Към входния слой пристигат блокови заявки за запис, които съдържат адресиране към логически ресурс, но все още не са обвързани с конкретни вътрешни структури на системата. На това ниво заявките се интерпретират и асоциират с логически томове (LUN⁸), пулове за съхранение и приложими политики за достъп и управление. Чрез тази асоциация заявките придобиват конкретно значение за системата и могат да бъдат насочени към съответните вътрешни механизми за обработка [40].

Освен логическа интерпретация, входният слой изпълнява и регулаторна функция. Чрез механизми за временно задържане и опашкуване той ограничава броя на операциите по запис, които едновременно навлизат във вътрешната обработка, като се съобразява с моментното състояние на системата и наличните ресурси. Този контрол се реализира динамично и има за цел да предотврати прекомерно натоварване на вътрешните компоненти [4, 54]. От гледна точка на времевото поведение входният слой формира началните условия за разпределението на латентността. Задържането, възникнало на това ниво, се отразява пряко върху времето за потвърждение, наблюдавано от хоста, и влияе върху поведението на последващите слоеве на системата за съхранение [4, 40].

2.6.2. Контролери и вътрешно планиране на операциите по запис

След като входният слой асоциира заявките за запис с конкретни логически томове (LUN) и съответните им пулове за съхранение, управлението на тези заявки се поема от контролерния слой на системата за съхранение. На този етап всяка заявка вече е еднозначно обвързана с конкретен контролер, кеш за записи и пул от физически носители, върху които ще бъде реализиран записът. Контролерите поемат координатията на този поток и определят реда и момента на изпълнение на операциите [21, 54].

⁸ LUN (Logical Unit Number) – логически идентификатор, чрез който системата за съхранение представя конкретен логически том към хост системата. Един LUN абстрахира физическата организация на данните и служи като основна единица за адресиране, кеширане и планиране на I/O операции от страна на контролерния слой.

Контролерният слой представлява изпълнителното ядро на системата за съхранение. Той управлява конкурентни заявки за запис, насочени към различни LUNs, които често споделят общи пулове, кеш и контролерно време. За тази цел контролерите поддържат вътрешни опашки, в които заявките се подреждат и изчакват обработка. Чрез тези опашки се регулира броят на операциите, които могат да бъдат едновременно активни, както и начинът, по който се разпределят ресурсите между тях [54].

Вътрешното планиране в контролерите включва арбитраж между множество конкурентни потоци от записи. Заявките не се обработват задължително в реда на постъпването им, а според текущото натоварване на контролера, състоянието на кеша и достъпа до съответните пулове от физически носители. Този подход позволява ефективно използване на наличните ресурси, но същевременно води до различно време за изчакване на отделните операции [4, 21].

Съществен елемент от контролерния слой е кешът за записи⁹, който позволява временно акумулиране на данните и отделяне на логическото потвърждение на операцията от нейното физическо изпълнение. В много архитектури операцията по запис се счита за завършена от гледна точка на хоста в момента, в който данните бъдат приети в кеша и бъде изпратено потвърждение. Реалното записване върху физическите носители в рамките на съответния пул се извършва впоследствие и асинхронно [54].

При умерено натоварване този механизъм води до стабилно и предсказуемо поведение. При интензивни конкурентни записи обаче вътрешните опашки в контролерите започват да се натрупват, особено когато множество LUNs споделят един и същ пул и кеш. В резултат на това част от заявките се обслужват сравнително бързо, докато други остават задържани значително по-дълго време, което води до разширяване на разпределението на латентността [4], [40].

От гледна точка на пътя на записите контролерният слой е мястото, в което асоциацията между LUN, пул и физически ресурси се превръща в конкретно времево поведение. Вариациите, възникнали на това ниво, оказват пряко влияние върху времето за потвърждение на записите, наблюдавано от хоста, и са ключов фактор за нестабилната латентност при конкурентни натоварвания. Поради това анализът на контролерния слой е съществен за разбирането на поведението на записите в SAN среда и подготвя прехода

⁹ **Кеш за записи (write cache)** – контролерна памет, която акумулира данни преди физическия запис и въвежда времево разделяне между логическото потвърждение към хоста и реалното записване, което има пряко влияние върху латентността и поведението на крайните опашки.

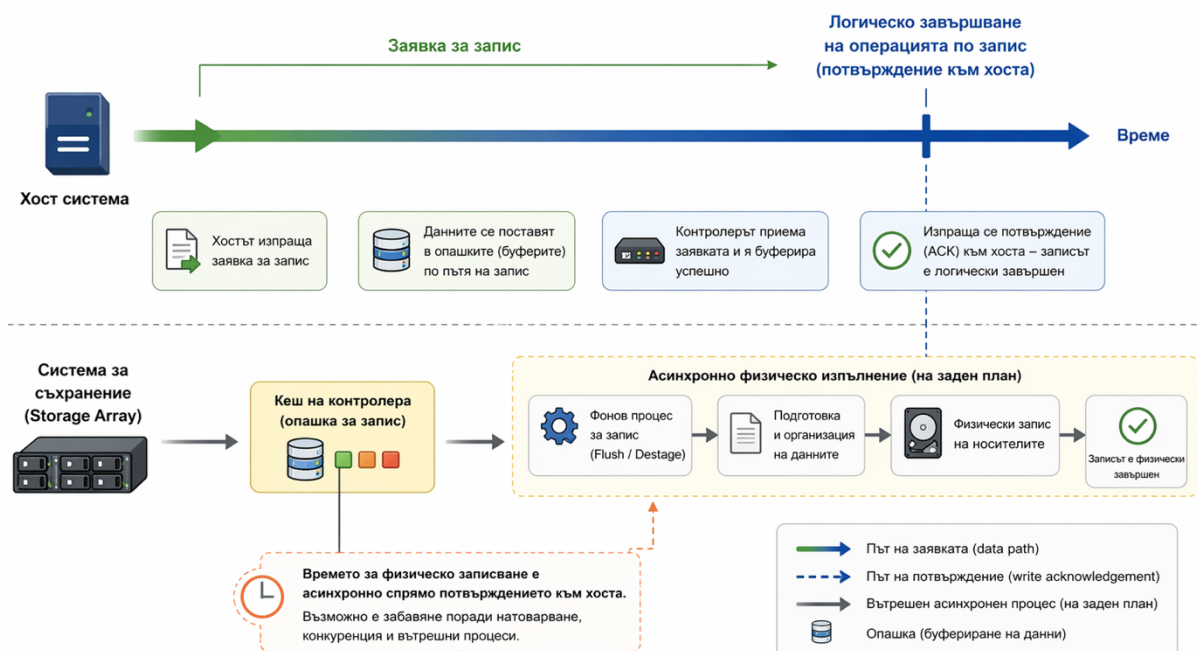
към разглеждането на физическата среда за съхранение и реалното изпълнение на записите.

2.6.3. Потвърждаване на запис и семантика на завършване на операциите

В пътя на записите в SAN среда понятието „завършена операция“ има различно значение в зависимост от гледната точка. За хост системата операцията се счита за завършена в момента на получаване на потвърждение за успешно изпълнение. Именно този момент определя наблюдаваното време за запис и представлява основната метрика за приложенията и операционната система [13, 20].

От гледна точка на системата за съхранение обаче това потвърждение не означава, че данните вече са записани върху физическата среда. В повечето архитектури записът се потвърждава след приемането на данните в кеша на контролера, докато реалното записване върху носителите се извършва по-късно и асинхронно [54].

Това разграничение между логическото завършване на операцията и нейното физическо изпълнение е илюстрирано на Фигура 4, която показва времевото разминаване между момента на потвърждение към хоста и реалния процес на запис върху физическата среда [4, 40].



Фигура 4. Логическо завършване на операцията по запис и асинхронно физическо изпълнение

Това е фундаментално за поведението на записите в SAN среда. От една страна, то позволява намаляване на наблюдаваната латентност чрез асинхронно записване и припокриване на операциите. От друга страна, въвежда сложност във времевия модел, при който множество операции могат да бъдат потвърдени за кратък интервал, без физическото записване да е завършено [20, 54].

При ниско и умерено натоварване тази асинхронност обикновено остава незабележима. При конкурентни записи обаче се създават условия за натрупване в кеша и във вътрешните опашки на контролерите, което води до неравномерно време за потвърждение. В резултат част от записите се потвърждават бързо, докато други изпитват значително по-дълго изчакване.

От гледна точка на хоста това поведение се проявява като вариации в латентността, които не могат да бъдат обяснени единствено чрез производителността на физическата среда. Наблюдаваното време за запис отразява динамиката на кеша, вътрешното планиране и моментното състояние на системата. Поради това средните стойности често прикриват съществени различия между отделните операции [40].

Следователно потвърждаването на запис не представлява пряка индикация за физическото му изпълнение, а резултат от взаимодействието между отделните слоеве на системата. Разбирането на тази семантика е необходимо условие за правилната интерпретация на измерената латентност и поставя основата за последващия експериментален анализ [4, 20].

2.6.4. Кеш на записите - стабилизация и маскиране на проблеми

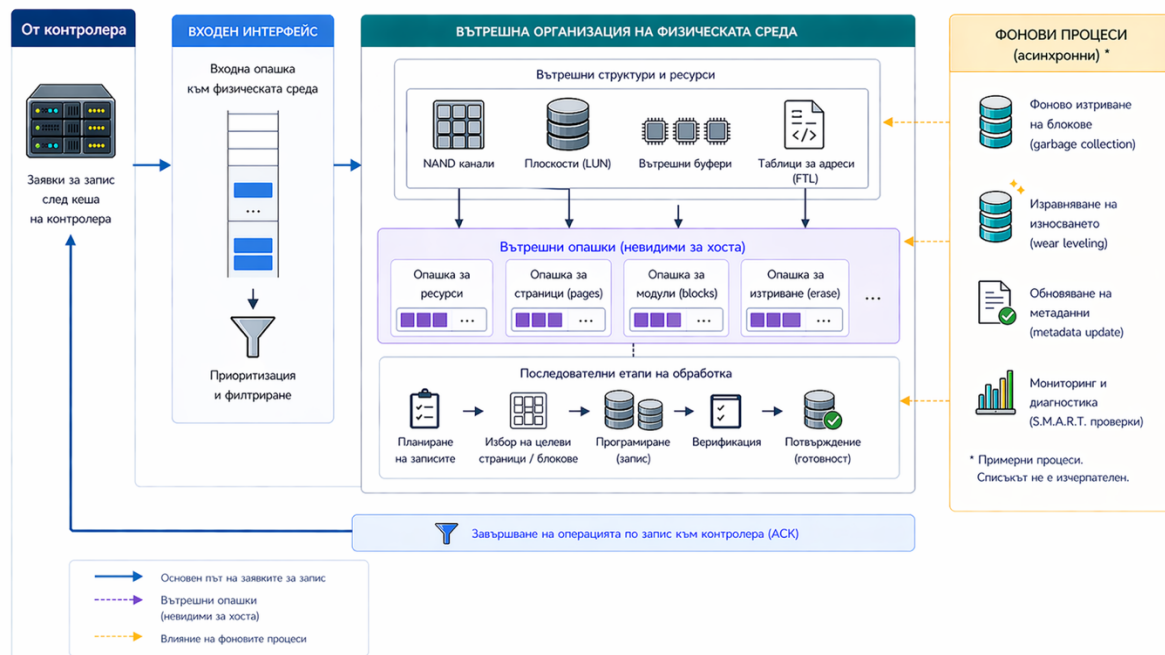
Кешът на записите е основен механизъм за стабилизация в системите за съхранение. Той позволява бързо потвърждение на операциите към хоста, докато реалното записване върху физическата среда се извършва асинхронно, като по този начин действа като буфер между входящия поток и ограниченията на носителите [54, 20]. При ниско и умерено натоварване кешът изглажда краткотрайни флуктуации и поддържа ниска и относително стабилна латентност. Този ефект обаче е ограничен, тъй като кешът разполага с краен капацитет и изисква непрекъснато изписване към физическата среда [4]. При продължително или интензивно натоварване, когато входящият поток надвиши способността за физическо записване, кешът се запълва и системата започва да забавя потвържденията към хоста [54]. В този момент се наблюдава рязка промяна в латентността след период на привидно стабилна работа.

Процесът на изписване от кеша към физическата среда протича паралелно с обработката на нови заявки, което води до конкуренция за вътрешни ресурси и увеличава вариативността на латентността при конкурентни натоварвания [40]. В този смисъл кешът има двойствена роля: той стабилизира латентността при краткотрайни пикове, но може да маскира натрупването на вътрешни опашки и да отложи проявата на нестабилно поведение. Това налага внимателна интерпретация на наблюдаваната латентност, особено при ограничени по време измервания [17, 54].

2.6.5. Опашки към физическата среда и вътрешен паралелизъм

След преминаването през входния слой, контролерите и кеша на записите, заявките достигат до последния етап от пътя - физическата среда за съхранение. Макар този етап често да се възприема като пряко свързан с реалното записване на данните, физическата среда притежава сложна вътрешна организация и собствен паралелизъм, които оказват съществено влияние върху времето за обслужване на отделните записи [23, 25].

Фигура 5 илюстрира логическия път на операциите по запис след напускане на кеша на контролера. Показани са вътрешните механизми за разпределение, паралелна обработка и фонове дейности, които не са видими от гледна точка на хоста, но оказват пряко влияние върху вариациите в латентността [23].



Фигура 5. Вътрешна организация на физическата среда за съхранение при операции по запис и формиране на невидими за хоста опашки

Физическата среда не представлява единен ресурс, а съвкупност от вътрешни елементи, управлявани паралелно чрез специализирани алгоритми. Това позволява едновременното обслужване на множество операции, но води до зависимост на времето за изпълнение от моментното натоварване и състоянието на вътрешните ресурси [23, 25].

При конкурентни записи част от заявките попадат в благоприятни моменти и се обслужват бързо, докато други изчакват освобождаване на вътрешни структури. В резултат дори при сходни входни условия се наблюдават значителни различия във времето за изпълнение на отделните операции.

Допълнителен източник на вариации са фоновите процеси, свързани с поддържането на нормалната работа и надеждността на физическата среда. Тези процеси не се изпълняват равномерно и при съвпадение с интензивно натоварване могат временно да увеличат латентността на част от операциите. От гледна точка на хоста това се проявява като неочаквани пикове, които не са пряко свързани с входното натоварване [25].

Ефектът от вътрешния паралелизъм и фоновите дейности се усилва при висока конкурентност, когато множество заявки едновременно се насочват към физическата среда. В този режим се формират вътрешни опашки, които не са видими извън системата за съхранение и водят до разширяване на разпределението на латентността [40].

От аналитична гледна точка това показва, че високата производителност на физическата среда не гарантира предсказуемо време за запис [25, 26, 40]. Напротив, вътрешният паралелизъм и асинхронните процеси водят до неравномерно обслужване на заявките и допринасят за вариативността на латентността, като допълват ефектите от опашките в останалите части на пътя на записите.

2.6.6. Синтез: „бърза среда“ не означава стабилно време за запис

Анализът на опашките в контролерите и към физическата среда показва, че нестабилността във времето за запис не е следствие от единичен компонент, а резултат от взаимодействието между множество вътрешни регулатори. В системата за съхранение записите преминават през последователност от слоеве, всеки от които въвежда собствени механизми за планиране, буфериране и синхронизация, необходими за постигане на висока производителност и устойчивост [4, 54].

Съвкупният ефект от тези механизми е, че дори при използване на високопроизводителна физическа среда част от записите неизбежно изпитват значително по-дълго време за обслужване. Това поведение не се проявява равномерно, а засяга отделни операции, които попадат в неблагоприятни моменти на натоварване или

съвпадение на вътрешни процеси. В резултат на това разпределението на латентността се разширява, като крайните стойности нарастват, без задължително да има видимо влошаване на средните показатели [40, 25].

Този извод е от ключово значение за настоящото изследване, тъй като показва, че оптимизацията на записите не може да бъде сведена до избор на „по-бърза“ среда за съхранение. Стабилното време за запис изисква съгласуване между натоварването, поведението на транспортния слой и вътрешните механизми за управление в системата за съхранение. Пренебрегването на който и да е от тези аспекти води до частични и често подвеждащи заключения [4, 40].

Представените в настоящата глава описания и наблюдения имат за цел да въведат концептуалния модел и системната гледна точка към пътя на записите в SAN среда. Всички интерпретации, обобщения и аналитични изводи, свързани с поведението на латентността и опашковите ефекти, се разглеждат целенасочено в аналитичната част на дисертационния труд след представяне на експерименталните резултати. По този начин раздел 2.6 завършва концептуалния модел на пътя на записите и подготвя прехода към въвеждане на метриките и дефинициите, чрез които описаните явления ще бъдат количествено анализирани в следващите глави.

2.7. Интерпретация на пътя на записите чрез теория на масовото обслужване

Теорията на масовото обслужване предоставя подходяща аналитична основа за интерпретация на поведението на операциите по запис в SAN среда. Независимо от сложността на съвременните архитектури за съхранение, пътят на записите може да бъде разглеждан като последователност от взаимодействащи обслужващи възли, между които се формират входно-изходни опашки.

В контекста на настоящото изследване основните елементи на write path-a - приложението, операционната система, HBA адаптерът, SAN fabric-ът, контролерите на системата за съхранение и физическите носители - могат да бъдат интерпретирани като последователни обслужващи звена, през които преминава всяка операция по запис. При всяко звено заявката прекарва определено време за обслужване и евентуално време за изчакване в опашка.

Съгласно принципите на теорията на масовото обслужване общото време за изпълнение на операциите по запис може концептуално да бъде представено чрез следната зависимост:

$$T_{\text{write}} = \sum_{i=1}^n (S_i + W_i)$$

където:

- T_{write} – общо време за изпълнение на операцията по запис;
- S_i – време за обслужване в i -тото звено по пътя на записите;
- W_i – време за изчакване във входно-изходната опашка на i -тото звено;
- n – брой последователни звена по пътя на записите.

От представената зависимост следва, че крайната латентност представлява сума от времената за обслужване и времената за изчакване във всяко звено по пътя на записите. При ниско натоварване доминиращ принос има компонентът S_i , докато с увеличаване на конкуренцията и натоварването нараства приносът на W_i , което води до формиране на опашкови ефекти и до непропорционално увеличение на крайните персентили на латентността.

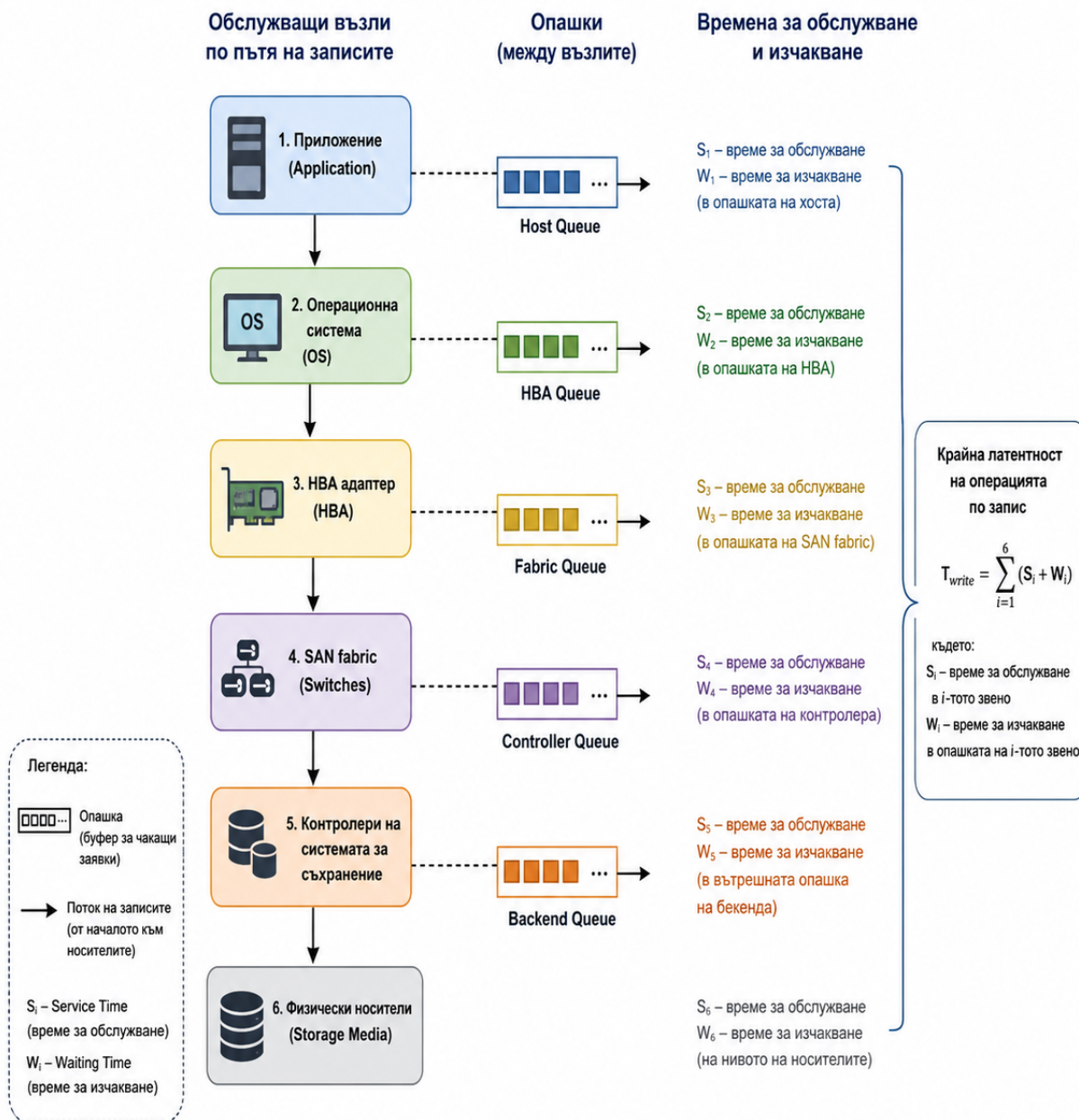
От тази гледна точка крайната латентност не се определя единствено от производителността на отделен компонент, а от съвкупното взаимодействие между времената за обслужване и натрупването на заявки по цялата верига на записите.

При ниско натоварване времената за изчакване остават малки и общата латентност се определя основно от времената за обслужване. С увеличаване на натоварването и степента на паралелизъм обаче времената за изчакване започват да нарастват непропорционално, което води до формиране на удължени латентностни опашки и до поява на значителни отклонения в крайните персентили на латентността.

Тази интерпретация съответства на наблюдаваните експериментални резултати, при които нарастването на натоварването не води до линейно изменение на латентността, а до преход между различни режими на работа - контролирана паралелност, преходна зона и координационен стрес. В рамките на настоящото изследване теорията на масовото обслужване не се използва за изграждане на строго математически модел на конкретна SAN реализация, а като аналитична рамка за интерпретация на опашковите ефекти и на взаимодействията между натоварването и латентността.

От гледна точка на оптимизацията тази интерпретация показва, че подобряването на поведението на операциите по запис не следва да бъде насочено единствено към увеличаване на производителността на отделни компоненти. По-ефективен подход е ограничаването на времената за изчакване чрез управление на паралелизма, контрол върху

дълбочината на входно-изходните опашки и предотвратяване на режими, водещи до координационен стрес. По този начин теорията на масовото обслужване осигурява аналитична основа за интерпретация и оптимизация на операциите по запис в SAN среда.



Фигура 6. Интерпретация на записите като последователност от обслужващи възли и опашки

2.8. Метрики и дефиниции за анализа

2.8.1. Латентност като основна метрика при записите

Латентността при записи е основната метрика за оценка на времето за реакция на SAN системата от гледна точка на хоста и приложението [3, 4]. В настоящото изследване

тя се дефинира като времето от подаване на заявката до получаване на потвърждение за нейното завършване според използвания модел на потвърждение. Това време включва изчакване в опашките по пътя на записите, пренос през транспортния слой и вътрешна обработка в системата за съхранение до момента на потвърждение [1, 4].

Тази дефиниция е съществена, тъй като латентността отразява поведението, наблюдавано от приложението, независимо от това дали данните са физически записани върху носител. Както беше разгледано в раздели 2.6.3 и 2.6.4, потвърждението често настъпва след буфериране в кеша, а не след окончателно записване. Поради това латентността следва да се интерпретира като време до потвърждение, а не като време до физически запис [54].

Латентността при записи е основна метрика за оценка на времето за реакция на SAN системата от гледна точка на хоста и приложението [3, 4]. В настоящото изследване тя се дефинира като времето от подаване на заявката до получаване на потвърждение за нейното завършване според семантиката на потвърждение. Това време включва изчакване в опашките по пътя на записите, пренос през транспортния слой и вътрешна обработка в системата за съхранение до момента на потвърждение [4].

Тази дефиниция е критична, тъй като латентността отразява поведението, наблюдавано от приложението, независимо от това дали данните са физически записани върху носител. Както беше разгледано в раздели 2.6.3 и 2.6.4, потвърждението често настъпва след буфериране в кеша, а не след окончателно записване. Поради това латентността следва да се интерпретира строго като време до потвърждение, а не като време до физически запис [54].

В SAN среди латентността обикновено се измерва в микросекунди и милисекунди [26, 27, 34]. Практически наблюдаваните стойности варират в зависимост от натоварването и конфигурацията, но могат да се разглеждат като ориентировъчни в следните диапазони:

- десетки до стотици микросекунди ($\approx 100\text{-}500\ \mu\text{s}$);
- под 1 ms;
- 1-5 ms;
- над 10 ms [4, 40].

Тези стойности зависят от характера на натоварването. Например, латентност от няколко милисекунди може да бъде приемлива при последователни записи, но проблемна при транзакционни операции. Аналогично, ниска средна латентност може да бъде

съпроводена от редки, но значителни пикове, които оказват непропорционално влияние върху поведението на приложенията [40].

В контекста на настоящото изследване латентността се разглежда не само като абсолютна метрика, а като индикатор за натрупване на опашки [1, 4, 40] по пътя на записите. Нарастването ѝ при сходни входни условия показва увеличено време за изчакване в системата. Поради това латентността служи като свързващ елемент между архитектурния модел и експерименталните наблюдения.

2.8.2. Променливост на латентността и стабилност на записите

Докато абсолютната стойност на латентността характеризира времето за реакция, променливостта определя неговата предсказуемост. В SAN среда именно променливостта, а не средната стойност, е основен фактор за стабилността на приложенията [40, 44].

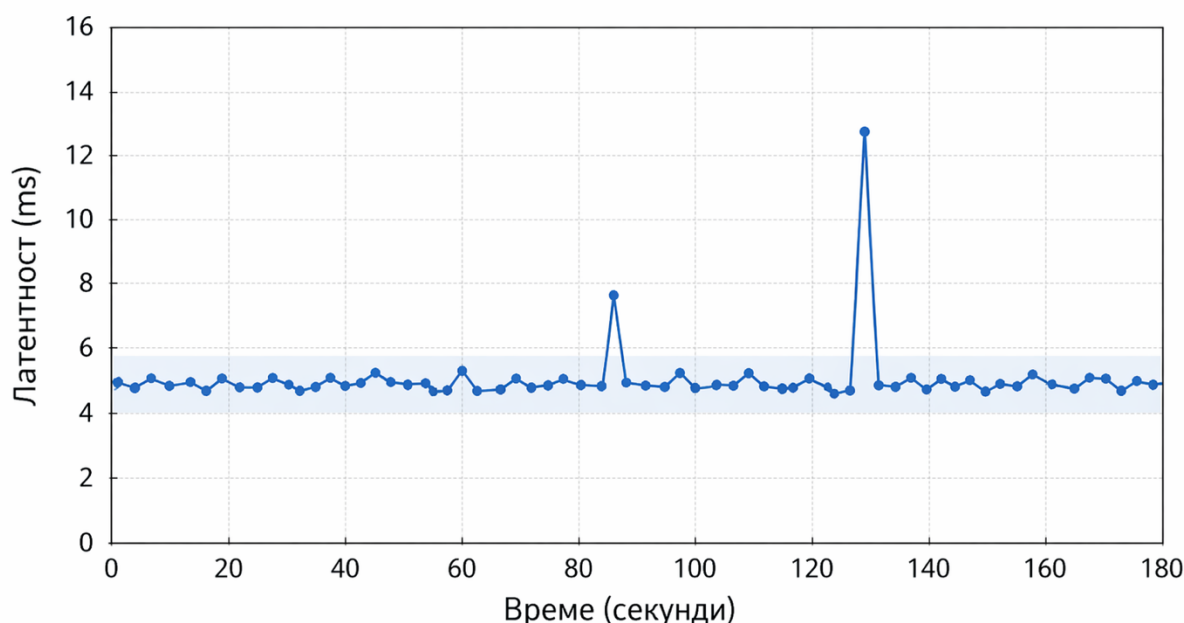
Системи със сходна средна латентност могат да показват съществено различно поведение [40]. Например, при средна латентност около 1-2 ms, в първия сценарий с равномерно разпределение на времената е значително по-стабилен от втория, при който малка част от записите достига 10-15 ms. Във втория случай редките, но бавни операции определят възприеманото поведение на системата [40].

Причината за тази променливост е свързана с механизма на пътя на записите [1, 4, 40]. Опашки в хоста, управление на потока в транспортния слой, вътрешно планиране в контролерите, кеширане и фонові процеси действат като независими източници на вариации. В определени моменти тези фактори се комбинират и водят до значително забавяне на отделни операции. С увеличаване на натоварването променливостта на латентността нараства непропорционално. При ниска конкурентност записите се обслужват равномерно, докато при по-висок паралелизъм системата оптимизира пропускателната способност за сметка на равномерността. В резултат разликата между бързи и бавни операции се увеличава [4, 40].

От практическа гледна точка стабилността на латентността често е по-важна от нейната минимална стойност. Много приложения функционират нормално при умерена, но постоянна латентност, но са чувствителни към редки пикове, които могат да доведат до блокиране или каскадни ефекти [40, 44]. Поради това в настоящото изследване променливостта се разглежда като самостоятелна характеристика. Анализът изисква използването на метрики, които улавят поведението в крайните случаи, което води до използването на персентили като основен инструмент за оценка на латентността.

2.8.3. Вариации и трептения¹⁰ във времето при натоварване

Освен разпределението на латентността, ключов аспект при анализа на записите в SAN среда е **поведението на латентността във времето**. Дори при тестове с фиксирана продължителност, анализът на латентността като времева серия показва наличие на трептения и локални изменения, които не могат да бъдат уловени чрез единични измервания или агрегирани стойности. Тези вариации отразяват моментното натрупване и освобождаване на опашките и действието на вътрешните механизми за управление на натоварването [4].



Фигура 7. Примерна илюстрация на латентността при записи, показваща трептения и редки пикове в рамките на фиксиран тест при стабилно натоварване

Фигура 7 илюстрира типичното поведение на латентността при записи, разгледана като времева серия в рамките на фиксиран тест. Видно е, че дори при стабилно средно ниво на латентността се наблюдават кратковременни пикове и трептения, които отразяват динамичното натрупване и освобождаване на опашките по пътя на записите [40, 4].

В практиката често се наблюдава следният сценарий: при стартиране на натоварване латентността при записи остава стабилна и ниска, например в диапазона 0.8-1.2 ms. В рамките на първите десетки секунди или минути системата успява да поеме входящия поток без значимо натрупване на опашки. В този период кешът на записите и

¹⁰ Терминът „трептения“ се използва описателно за означаване на кратковременни вариации във времето за обслужване на записите (latency fluctuations) при сходни условия на натоварване.

свободният капацитет на вътрешните ресурси действат като буфер, който изглажда моментните пикове.

При продължаване на натоварването обаче започват да се проявяват времеви вариации. Латентността може периодично да нараства до 3-5 ms, след което отново да се върне към първоначалните стойности. Тези флуктуации често се повтарят с определена честота и са резултат от циклични вътрешни процеси, като освобождаване на кеш, преразпределение на ресурси или активиране на фоновы дейности във физическата среда [25]. При по-високи нива на натоварване ефектът става още по-изразен. Например, при постоянен поток от конкурентни записи системата може да показва латентност около 1-2 ms през по-голямата част от времето, но на всеки няколко секунди или минути да се появяват пикове от 10-20 ms за малка част от операциите [40]. Тези пикове често са краткотрайни, но имат непропорционално голямо влияние върху поведението на чувствителни приложения.

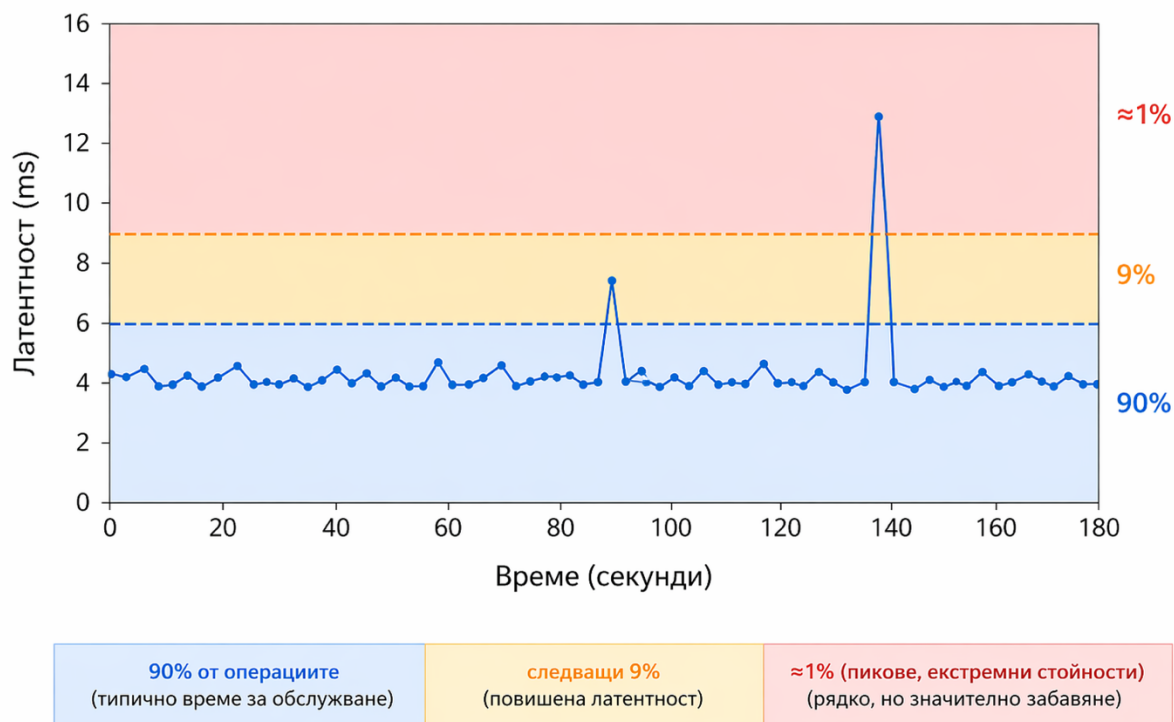
Важно е да се подчертае, че тези трептения не са непременно признак за недостатъчна производителност на системата. В много случаи общата пропускателност остава стабилна, а средната латентност се изменя минимално. Въпреки това времевата неравномерност означава, че отделни записи попадат в неблагоприятни моменти, когато вътрешните опашки са временно натоварени или когато системата изпълнява допълнителни фоновы операции [4].

Този ефект може да бъде илюстриран с опростен пример. При тест с продължителност от няколко минути може да се наблюдава следното поведение:

през 90% от времето латентността е в диапазона **1-2 ms**;

през около 9% от времето латентността се повишава до **3-6 ms**;

през останалия 1% се появяват пикове над **10 ms**.



Фигура 8. Илюстративно разпределение на времето за обслужване на записите по диапазони на латентност

Фигура 8 показва, че поведението на латентността при записи не може да бъде адекватно описано с една усреднена стойност. Макар по-голямата част от операциите да се обслужват в тесен и привидно стабилен диапазон, редките пикове формират крайни стойности, които имат несъразмерно голямо влияние върху приложното поведение. Това налага анализ, който отчита не само централните тенденции, но и разпределението на латентността като цяло. В следващия раздел този анализ се формализира чрез използване на перцентилни показатели [3, 6].

Ако анализът се базира само на средни стойности или кратки извадки, тези пикове могат да останат незабелязани. От гледна точка на приложенията обаче именно тези редки, но значителни закъснения често са причина за влошаване на производителността, изчаквания или временно блокиране на операции [40].

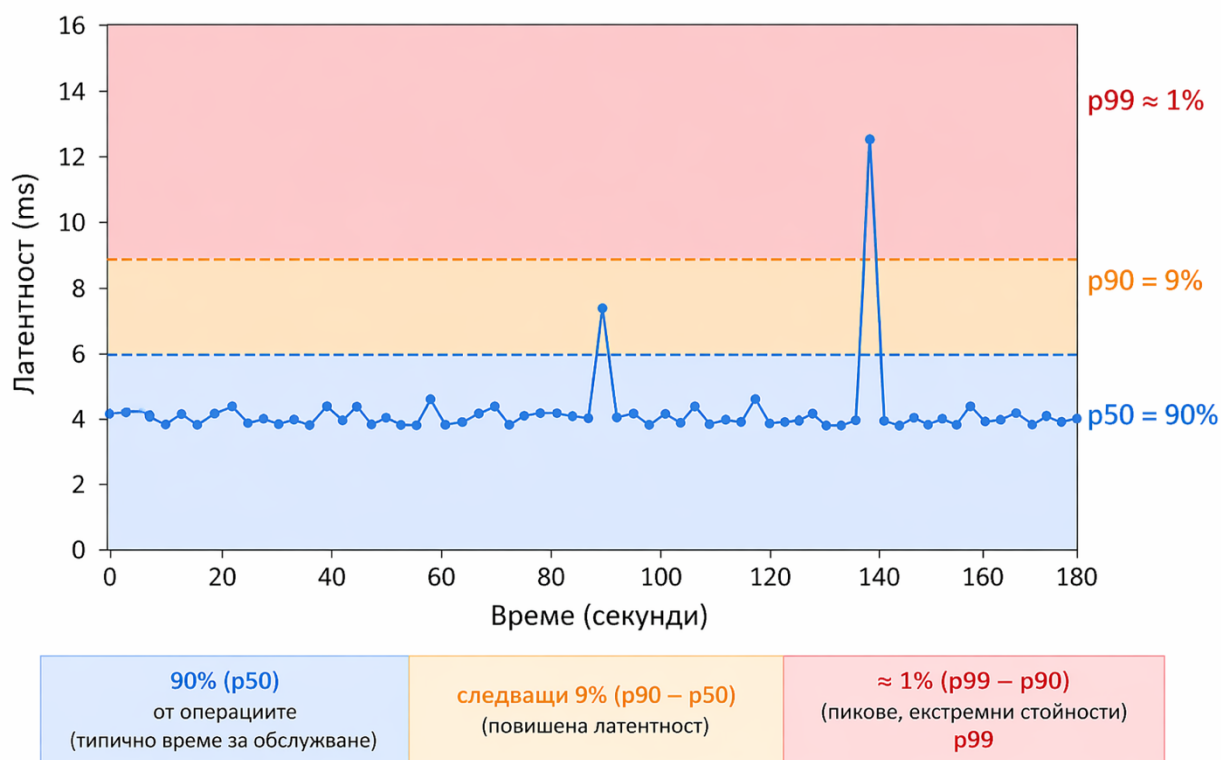
Поради тази причина вариациите във времето трябва да се разглеждат като самостоятелна характеристика на системата. Те отразяват не само моментното натоварване, но и начина, по който системата реагира на продължителен натиск. В контекста на записите в SAN среда времевите трептения са индикатор за натрупване и освобождаване на опашки в различни части на пътя на записите [4].

Това наблюдение води до необходимостта от използване на метрики, които улавят не само типичното поведение, но и крайните случаи във времето. Именно тук персентилите на латентността се превръщат в основен инструмент за анализ, тъй като позволяват количествено описание на редките, но критични пикове. В следващия раздел се разглежда тяхното значение и практическа интерпретация при анализ на записи в SAN среда.

2.8.4. Персентили: p50, p95, p99 - смисъл и интерпретация

Както беше показано във Фигура 8, латентността при записи варира във времето и се характеризира с ясно изразено основно „тяло“ и сравнително малка, но значима опашка от редки пикове. Това поведение може да бъде разглеждано не само интуитивно, чрез времевата серия, но и формално, чрез показатели, които описват каква част от операциите попадат под определени нива на латентност [40].

Тази формализация е илюстрирана на Фигура 9, където същата времева серия на латентността е надградена с означени персентилни нива. Хоризонталните линии p50, p95 и p99 съответстват на граници, под които се намират съответно 50%, 95% и 99% от всички наблюдавани операции. По този начин персентилите дават количествено описание на областите, които във времевия домейн се проявяват като типично поведение и като редки, но отчетливи пикове [3, 6].



Фигура 9. Същата времева серия на латентността при записи, показана на Фигура 8, надградена с формално означени персентилни нива

Персентилът $p50$ представлява медианната латентност и описва „типичното“ поведение на системата. Той показва стойността, под която се обслужва половината от всички записи, и често съвпада с интуитивното усещане за бързодействие при нормална работа. Въпреки това $p50$ не дава информация за стабилността на системата в неблагоприятни моменти [6].

Персентилът $p95$ описва поведението на системата при по-неблагоприятни, но все още сравнително чести случаи. Той маркира горната граница, под която се намират по-голямата част от операциите, и е чувствителен към временни натрупвания на опашки и конкуренция за ресурси [4].

Персентилът $p99$ отразява поведението на системата в крайните случаи, които заемат малка част от времето, но често определят приложното въздействие на латентността. Както се вижда на Фигура 8, тези редки пикове пресичат най-високото персентилно ниво и именно те формират опашката на разпределението, която остава скрита при използване на усреднени показатели.

За да се илюстрира смисълът на персентилите, може да се разгледа следният пример. При дадено натоварване системата показва средна латентност от 2 ms. В същото време медианната стойност $p50$ е 1.2 ms, $p95$ достига 4 ms, а $p99$ - 15 ms. Средната

латентност създава впечатление за бърза система, докато персентилите разкриват, че малка част от записите се обслужват със значително по-голямо закъснение. Именно тези редки, но продължителни операции често определят реалното усещане за производителност и стабилност [3, 40].

В контекста на операции по запис в SAN среди персентилите p50, p95 и p99 позволяват да се направи ясно разграничение между система, която е „бърза в повечето случаи“, и система, която е стабилна и предсказуема дори в неблагоприятни моменти. Поради това в настоящото изследване персентилните показатели се използват като основен инструмент за интерпретация на поведението на записите при различни режими на натоварване и служат като основа за последващия анализ на разпределението на латентността [4].

2.8.5. Интерпретация на персентилите при записи в SAN

Персентилите на латентността придобиват реален смисъл едва когато бъдат интерпретирани в контекста на пътя на записите в SAN среда. За разлика от локалното съхранение, където пътят на заявката е сравнително кратък, при SAN записите преминават през множество логически и физически слоеве. Всеки от тези слоеве може да въведе временно задържане, което да засегне само част от операциите и по този начин да повлияе именно на крайните стойности на разпределението [4, 40].

В този контекст p50 отразява поведението на системата при типични условия, когато повечето заявки не срещат значими опашки. Тази стойност е чувствителна основно към базовата архитектура и към ефективността на бързите пътища за обработка. Ниска p50 латентност показва, че при липса на конкуренция системата е способна да обслужва записите бързо, но не дава информация за поведението при натоварване.

Персентилите p95 и p99, от своя страна, улавят моментите, в които записите попадат в неблагоприятни условия. Това могат да бъдат ситуации на натрупване на заявки в хоста, временно задържане в транспортния слой, конкуренция в контролерите или активиране на фонові процеси към физическата среда. В тези случаи малка част от записите изчакват значително по-дълго, което се отразява директно върху p95 и особено върху p99 [40].

Практически това означава, че увеличаването на p99 латентността често е първият измерим сигнал за проблем в пътя на записите. Например, при непроменени входни параметри и стабилна средна латентност, рязко покачване на p99 показва, че в някой от слоевете започва натрупване на опашки. Този ефект може да бъде временен или

периодичен, но е показателен за ограничение в съгласуването между натоварване, транспорт и вътрешно управление на ресурсите [3, 4].

Важно е да се отбележи, че различните методи за оптимизация влияят по различен начин на персентилите. Някои подходи могат да подобрят p_{50} и средната латентност, без съществено да повлияят на p_{99} . Други могат да стабилизират крайните стойности, дори ако типичната латентност остане непроменена или леко се увеличи. В този смисъл персентилите позволяват по-фин анализ на компромисите между производителност и предсказуемост.

При интерпретация на експерименталните резултати в SAN среда е важно персентилите да се разглеждат във връзка с конкретните точки на опашкуване от модела на пътя на записите. Например, увеличение на p_{99} при повишаване на дълбочината на опашките от страна на хоста може да се свърже с натрупване още в началото на пътя. Аналогично, увеличение на p_{99} при непроменено поведение на хоста може да насочи вниманието към транспортния слой или към вътрешните механизми в системата за съхранение.

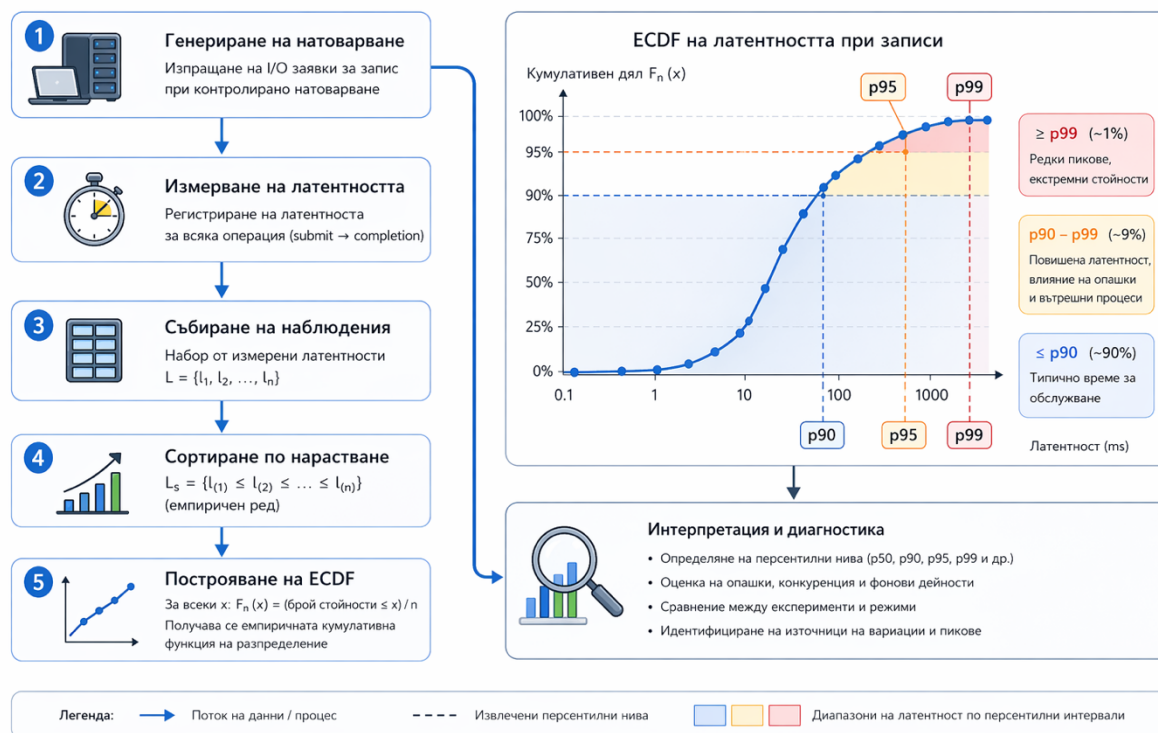
По тази причина в настоящото изследване персентилите не се използват изолирано, а като средство за диагностика на поведението на системата. Те позволяват да се направи връзка между архитектурния модел, параметрите на натоварването и реалното въздействие върху приложенията. Това превръща p_{95} и p_{99} в ключови метрики за оценка на ефективността на различни методи за оптимизация на записите в SAN среда.

С този раздел се завършва интерпретацията на персентилните показатели, използвани в настоящото изследване. За да бъде анализът пълен, е необходимо персентилите да бъдат разглеждани в контекста на цялото разпределение на латентността, което позволява да се идентифицира структурата и тежестта на опашката. В следващия раздел това разпределение се разглежда чрез емпиричната кумулативна функция на разпределение (ECDF) [6].

2.8.6. Поток на измерване и интерпретация чрез Емпирична кумулативна функция на разпределение (ECDF)

Анализът на латентността при операции по запис в SAN среда в настоящото изследване се основава на последователен поток от измерване, обработка и интерпретация на реални входно-изходни операции. Целта на този поток не е единствено да се получат числови показатели, а да се разкрие как различни системни състояния се отразяват върху времето за обслужване на отделните записи. Фигура 9 илюстрира този процес в обобщен

вид и служи като концептуална рамка за използването на ECDF в по-нататъшния анализ [6], [52].



Фигура 10. Поток на измерване и интерпретация на латентността чрез емпирична кумулативна функция на разпределение (ECDF)

Изходната точка на анализа са реалните операции по запис, генерирани от хост системата при конкретен експериментален сценарий. Всяка операция представлява индивидуална I/O заявка, която преминава през пътя на записите – от приложението и хостовия стек, през транспортния слой, до системата за съхранение и момента на потвърждение. За всяка операция се измерва латентността като времето от подаване на заявката (submit) до получаване на потвърждение за завършване (completion) от гледна точка на хоста. Това измерване отразява приложно релевантното време за реакция и включва всички забавяния, породени от опашки, конкуренция за ресурси и вътрешни механизми в системата [3, 50].

В резултат на изпълнението на експерименталния сценарий се получава множество от индивидуални измервания на латентността, като всяка стойност съответства на една конкретна операция по запис. Тези наблюдения могат да бъдат формално представени като набор от стойности $\{l_1, l_2, \dots, l_n\}$. В първоначалния си вид те отразяват времевата динамика на натоварването, но сами по себе си не позволяват пряк анализ на разпределението на латентностите [6].

За да се премине от времева последователност към анализ на разпределението, измерените латентности се подлагат на елементарна, но концептуално съществена обработка. Всички стойности се сортират по нарастване, без да се променя тяхната числова стойност, като за всяка сортирана латентност се изчислява кумулативният дял на операциите, приключили до тази стойност. По този начин всяка операция се позиционира не във времето, а спрямо дела от всички операции, които се обслужват по-бързо или равностойно.

Емпиричната кумулативна функция на разпределение (ECDF) представлява графичното представяне на тази връзка между латентността и кумулативния дял. ECDF не въвежда моделиране, усредняване или статистическа апроксимация, а визуализира директно реално измерените операции. Всяка точка от кривата отговаря на въпроса каква част от всички записи са приключили до дадена стойност на латентността, което позволява едновременно да се наблюдават както типичното поведение на системата, така и редките, но значими отклонения [6, 52].

При анализ на ECDF на латентността при записи обикновено се открояват две концептуално различни области. Основното тяло на разпределението обхваща латентностите, при които се изпълнява по-голямата част от операциите и отразява поведението на системата при нормално системно състояние. В тази област записите не срещат значими опашки и се обслужват при налични ресурси и относително стабилен поток [3, 40].

Крайната част на кривата формира опашката на разпределението. Опашката е характеристика на самото разпределение и показва, че малък дял от операциите е асоцииран със значително по-високи стойности на латентността. Операциите, които попадат в тази област, са изпълнени при временен системен дисбаланс - например при натрупване на опашки, краткотрайна конкуренция между заявки или активиране на вътрешни процеси в системата за съхранение. Тези операции не представляват грешки или аномалии, а естествен резултат от динамичното поведение на многослойна SAN архитектура [3].

В този контекст ECDF изпълнява двойна роля. От една страна, тя позволява ясно разграничаване между типичното поведение на системата и операциите, изпълнени при временен дисбаланс. От друга страна, чрез формата на кривата и поведението на опашката тя служи като диагностичен инструмент за идентифициране на различни режими на работа. Персентилни показатели като p95 и p99 могат да се използват като ориентири за

границата между тези режими, но тяхната интерпретация придобива смисъл единствено в контекста на цялото разпределение.

В рамките на настоящото изследване ECDF се използва като основно средство за интерпретация и съпоставка на експерименталните резултати. Тя позволява да се проследи как промените в натоварването, транспортния модел и управлението на опашките се отразяват не само върху средната латентност, но и върху стабилността и предсказуемостта на операциите по запис.

2.9. Четене срещу запис

Въпреки че операциите за четене и запис използват една и съща SAN инфраструктура, тяхното поведение се различава съществено [16, 21]. В настоящото изследване фокусът е върху записите, тъй като те представляват по-сложния и по-чувствителен към натоварване процес [16, 20].

При четене системата доставя вече съществуващи данни към хоста, като времето за изпълнение зависи основно от наличието им в кеша и от латентността на достъпа до физическата среда. Записът, от друга страна, изисква потвърждение, че данните са приети и защитени, което въвежда допълнителни етапи и механизми за управление и увеличава чувствителността към опашки и конкуренция за ресурси [16].

Особено съществена е ролята на кеша при записите. Докато при четене кешът ускорява достъпа, при запис той действа като буфер, който позволява асинхронно физическо записване. Това стабилизира типичната латентност, но същевременно създава условия за натрупване на скрита работа и за поява на вариации [17, 54].

Поради това записите се разглеждат като основен източник на нестабилност и като ключов фактор за производителността в SAN среди. Анализът на четенето не е разгледан в детайли, тъй като изисква различен модел и метрики, и тук се използва единствено като контекст за обосновка на избрания фокус.

2.10. Обобщение на модела и връзка с експериментите

В настоящата глава беше изграден системен модел на пътя на записите в SAN среда, разглеждан като съвкупност от взаимосвързани слоеве и регулатори, а не като линеен канал за пренос. Основният извод е, че латентността при записите се формира като резултат от взаимодействието между множество механизми за управление на потока, всеки от които може да въведе задържане и натрупване на опашки [1, 4, 40].

Анализът показва, че още на ниво хост параметри като дълбочина на опашките, паралелизъм и форма на натоварването задават началните условия за поведението на

системата. Неподходящото генериране на заявки може да доведе до нестабилна латентност независимо от производителността на останалите компоненти [13].

Транспортният слой беше разгледан като механизъм за управление и синхронизация, при който задържания могат да възникват независимо от наличната пропускателна способност, в резултат от управление на потока и взаимодействие между паралелни канали [22, 40]. Анализът на системата за съхранение допълни модела чрез разглеждане на вътрешното планиране, ролята на кеша и взаимодействието с физическата среда. Кешът стабилизира типичната латентност, но може да маскира натрупване на работа и да отложи проявата на нестабилност под формата на крайни стойности [54]. Особено значение има физическата среда, където вътрешният паралелизъм и фоновите процеси водят до вариации във времето за обслужване. Това показва, че високата производителност сама по себе си не гарантира предсказуемо време за запис [25, 40].

Във втората част на главата бяха въведени метрики за количествен анализ. Латентността беше дефинирана като време до потвърждение, а променливостта ѝ – като ключов фактор за стабилността. Чрез персенти и ECDF беше аргументирано защо средните стойности не са достатъчни за оценка на поведението на записите.

Връзката с експерименталната част се състои в това, че всеки експеримент въздейства върху конкретни елементи от пътя на записите. Промените в натоварването и конфигурацията могат да бъдат директно съотнесени към наблюдаваните изменения в латентността и разпределението ѝ [3, 6].

По този начин настоящата глава изпълнява ролята на концептуална рамка, която свързва архитектурата на SAN средите с измеримото поведение на записите и служи като основа за експерименталния анализ в следващата глава.

ГЛАВА 3. ЕКСПЕРИМЕНТАЛНА МЕТОДОЛОГИЯ

3.1. Изследователска цел и логика на експерименталния подход

Настоящото изследване има за цел да анализира поведението на операциите по запис в SAN-базирана среда при нарастваща конкуренция и паралелизъм, с фокус върху стабилността на латентността и проявлението на нелинейни опашкови ефекти. Вместо да се търси максимална производителност или оптимална конфигурация, акцентът е поставен върху разбирането на условията, при които системата преминава от предсказуемо към нестабилно поведение [4, 40].

Изследването изхожда от наблюдението, че традиционните агрегирани метрики като пропускателна способност и средна латентност често прикриват критични деградации, които се проявяват рядко, но имат съществено влияние върху качеството на услугата. В този контекст анализът се насочва към разпределението на латентността и по-специално към крайните му персентили, които отразяват натрупването на заявки и поведението на системата под напрежение [3, 6, 40].

Логиката на експерименталния подход е изградена около систематичното провокиране на конкуренция в пътя на записите чрез контролирани натоварвания. Натоварването не се разглежда като цел само по себе си, а като инструмент за въвеждане на напрежение, което позволява да бъдат наблюдавани преходи в поведението на системата. По този начин експериментите не целят да демонстрират „колко бърза“ е дадена конфигурация, а да покажат *как* и *къде* се променя поведението ѝ при нарастваща конкурентност на входния поток [50].

Експерименталната среда е проектирана така, че да минимизира влиянието на външни и трудно контролируеми фактори, като транспортни аномалии или динамични оптимизации на инфраструктурно ниво. Това позволява наблюдаваните ефекти да бъдат интерпретирани като резултат от взаимодействието между паралелизма на натоварването и вътрешното натрупване на заявки, а не като следствие от случайни флуктуации на средата [50, 52].

В този смисъл настоящата методология разглежда оптимизацията не като търсене на максимални стойности на производителност, а като процес на повишаване на стабилността и предсказуемостта на пътя на записа при условия на нарастваща конкуренция. Целта не е формулирането на универсални препоръки или извеждането на пределни характеристики, а изграждането на последователна аналитична рамка, в която нестабилността и редките, но критични забавяния се интерпретират като диагностични индикатори за състоянието на пътя на записа, а не като резултат от случайни или външни въздействия на средата.

В този контекст оптимизацията се третира като процес на управление на поведението на системата под натоварване. Вместо да се търсят максимални стойности на производителност при изолирани условия, изследването се фокусира върху ограничаването на опашковите ефекти и стабилизирането на крайните стойности на латентността, които са критични за реални натоварвания в SAN среда [4, 40].

3.2. Изследователски модел: конкуренция, опашки и латентност

Настоящият изследователски модел разглежда пътя на записите като система, в която операциите по запис, генерирани от хоста, се обслужват от ограничен набор ресурси с крайна обслужваща способност. При отсъствие на конкуренция тези операции се обработват с относително постоянна и слабо варираща латентност. С нарастване на паралелизма и броя на едновременно активните заявки обаче възниква конкуренция за споделени ресурси, което води до формиране на опашки и до съществено изменение на латентностното поведение [4, 7].

В рамките на модела конкуренцията се въвежда контролирано чрез характеристиките на натоварването - брой едновременно активни заявки, тяхната времева структура и степента на координация между тях. Тези параметри се третират като независими променливи, чрез които системата се довежда до различни режими на работа - от слабо натоварен до режим на насищане и преходна нестабилност [50].

Опашките се разглеждат като основния механизъм, чрез който конкуренцията се материализира в наблюдаемо забавяне. При нарастване на броя чакащи заявки времето за обслужване на отделни операции вече не се определя единствено от собствената им продължителност, а и от позицията им в опашката. Това води до нелинейно нарастване на латентността за част от операциите, дори когато агрегирани показатели като пропускателна способност остават относително стабилни [4, 40].

Латентността в модела се анализира като разпределение, а не като единична стойност. Основен интерес представляват крайните части на това разпределение, които отразяват ефекта на натрупване в опашките и позволяват да се идентифицират режими на нестабилност. В този контекст оптимизационният проблем се формулира като задача за ограничаване на разширяването на разпределението на латентността при нарастваща конкуренция, а не единствено като минимизиране на средната стойност [3, 6, 40].

Моделът умишлено абстрахира вътрешните механизми на отделните компоненти и ги разглежда като функционални етапи по пътя на записите. Това позволява конкуренцията, опашките и латентността да бъдат анализирани на системно ниво, независимо от конкретната реализация на компонентите. По този начин наблюдаваните ефекти могат да бъдат пряко свързани с характеристиките на натоварването и използвани като основа за формулиране на обобщени оптимизационни изводи [52].

Предложеният модел служи като операционна рамка за експериментите, разгледани в следващите раздели. Чрез систематична промяна на параметрите на натоварването и анализ на опашковите ефекти върху разпределението на латентността,

моделът позволява да се изследват условията, при които write path-ът преминава от стабилен към нестабилен режим, както и възможностите за неговото оптимизиране [50, 40].

3.3. Натоварването като независима променлива

В рамките на настоящото изследване натоварването се разглежда като основната независима променлива, чрез която се управлява степента на конкуренция по пътя на записите. Целта не е възпроизвеждане на конкретен приложен сценарий, а систематично въвеждане на напрежение в системата по начин, който позволява контролирано наблюдение на опашковите ефекти и тяхното влияние върху латентността [50, 4].

Натоварването се характеризира чрез три основни параметъра: брой едновременно активни заявки, времева структура на подаването и степен на координация между паралелните потоци. Тези параметри се изменят поотделно в различните експериментални сценарии, като всички останали характеристики на средата се поддържат фиксирани. По този начин натоварването се използва като инструмент за изолиране на конкретни механизми, водещи до деградация на латентността.

Паралелизмът, изразен чрез броя едновременно активни операции по запис, се използва като първичен фактор за провокиране на конкуренция. При ниски стойности системата работи в режим с минимално натрупване, докато при постепенно увеличаване на паралелизма възникват опашки, които променят характера на обслужването. Този преход представлява основен интерес за изследването, тъй като именно в него се наблюдава разминаване между агрегирани показатели и крайни стойности на латентността [4, 40].

Освен количеството заявки, съществено значение има и тяхната времева организация. Натоварвания със сходен среден интензитет, но с различна степен на концентрация във времето, могат да доведат до различно поведение на системата. Чрез контролиране на времевата структура на подаването се изследва влиянието на краткотрайни пикове върху дълбочината на опашките и върху разпределението на латентността, без промяна на средната натовареност [6, 40].

Допълнителен аспект на натоварването е координацията между паралелните потоци. Синхронното или асинхронно подаване на заявки оказва влияние върху начина, по който конкуренцията се проявява в системата. Чрез вариране на степента на координация се анализира дали и в каква степен контролът върху подаването може да бъде

използван като средство за ограничаване на опашковите ефекти и стабилизиране на латентността [7].

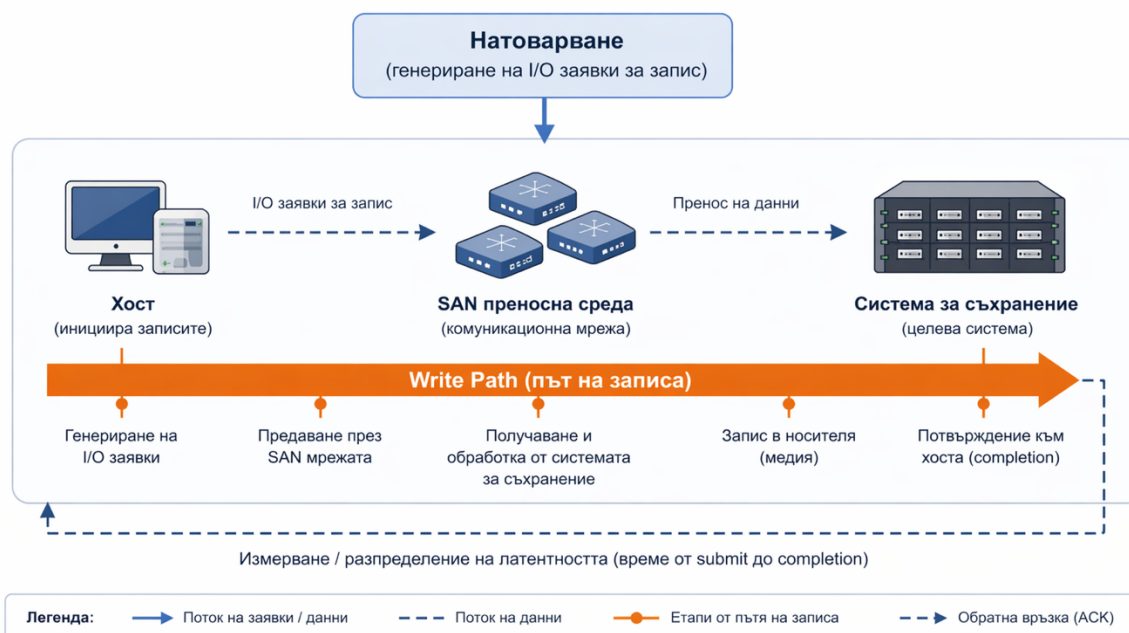
По този начин натоварването в настоящото изследване не се разглежда като статичен профил, а като набор от управлявани параметри, чрез които пътят на записите се привежда в различни режими на работа. Това позволява експериментите да бъдат структурирани като последователни стъпки в анализа на оптимизационния проблем, при който се търсят условия за минимизиране на деградацията на латентността при конкуренция.

Този подход позволява всяка експериментална серия да бъде интерпретирана като контролирана промяна в условията на конкуренция, а не като независим тестов сценарий.

3.4. Контролирана експериментална среда

Експерименталната среда в настоящото изследване е проектирана и поддържана като контролирана система, при която всички компоненти по пътя на записа остават фиксирани във всички експериментални сценарии. Този подход има за цел да изолира влиянието на натоварването като единствен източник на вариация и да гарантира, че наблюдаваните промени в поведението на латентността могат да бъдат свързани пряко с параметрите на конкуренцията [50, 52].

Контролираната среда обхваща хостовия слой, SAN преносната инфраструктура и системата за съхранение, които се разглеждат като функционално свързана верига, формираща пътя на записа. В рамките на изследването тези компоненти не се сравняват помежду си, не се оптимизират поотделно и не се третираат като независими променливи, а служат като стабилна основа, върху която се анализира поведението на операциите по запис при различни режими на натоварване.



Фигура 11. Концептуален модел на контролираната експериментална среда и пътя на записа, използван в изследването

Концептуалният модел на контролираната експериментална среда е илюстриран на Фигура 11. Моделът представя пътя на записа като последователност от функционално свързани компоненти, върху които целенасочено се прилага контролирано натоварване с цел анализ на конкуренцията и опашковите ефекти [4, 7].

Контролът върху натоварването се реализира чрез параметри като дълбочина на опашките, времева структура на заявките и степен на координация между конкурентните операции. Чрез систематично изменение на тези параметри се провокира конкуренция за ограничените ресурси по пътя на записа, което позволява да бъдат анализирани опашковите ефекти и тяхното отражение върху латентностните характеристики [50, 40].

Наблюдението и измерването се извършват на хостово ниво, като латентността се отчита като крайно (енд-ту-енд) време за обслужване на операциите по запис. Този подход осигурява фокус върху приложно релевантното поведение на системата, без навлизане в специфични вътрешни реализации на отделни компоненти [3, 6].

Поддържането на фиксирана експериментална среда позволява да се минимизира влиянието на външни и трудно контролируеми фактори, като динамични инфраструктурни оптимизации, транспортни флуктуации или адаптивни механизми. По този начин наблюдаваните вариации в разпределението на латентността могат да бъдат интерпретирани като резултат от взаимодействието между конкуриращи се заявки и ограничените ресурси по пътя на записа, а не като следствие от случайни изменения в средата [50, 52].

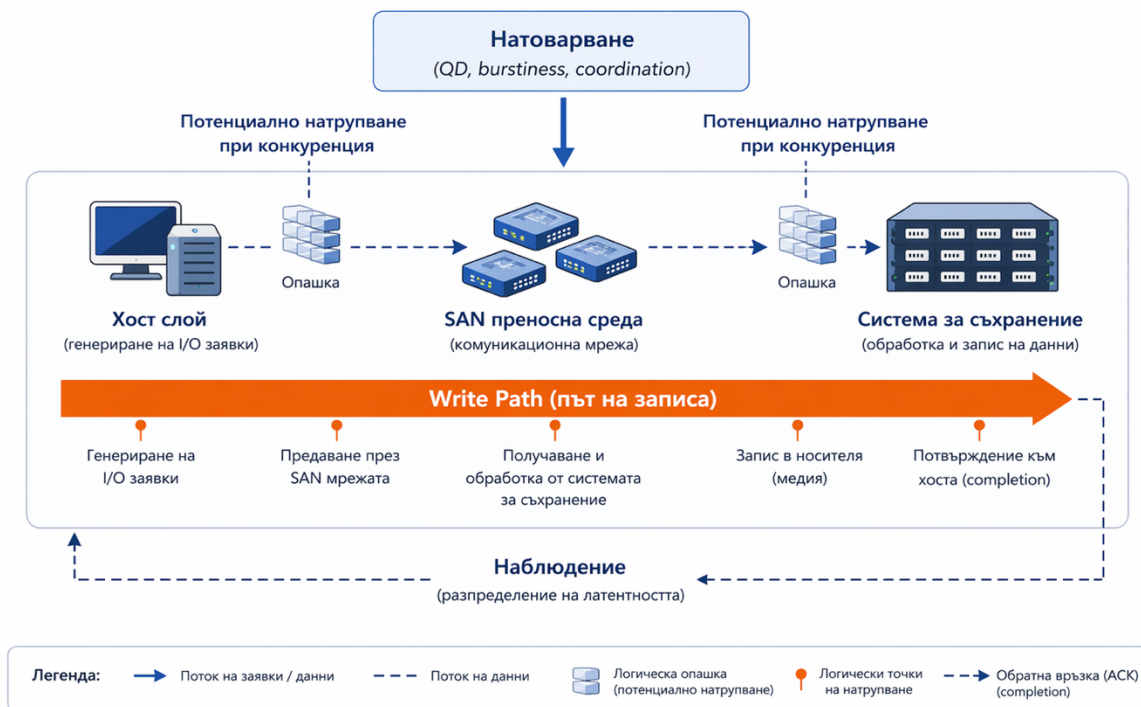
Всички експериментални сценарии се изпълняват при идентични условия, като във всеки експеримент се изменя само една независима променлива. Това осигурява съпоставимост на резултатите и позволява ясно проследяване на ефекта от всяка промяна в натоварването, без натрупване на вторични влияния.

В този смисъл контролираната експериментална среда не се разглежда като универсална или оптимална конфигурация, а като методологичен инструмент. Нейната роля е да осигури стабилна и последователна рамка, в която опашковите ефекти и координационните явления по пътя на записа могат да бъдат наблюдавани, анализирани и интерпретирани по единна изследователска линия в следващите глави.

Това позволява експерименталните резултати да бъдат разглеждани като възпроизведими и системно обосновани, което е предпоставка за тяхната интерпретация и обобщение. Измерването на латентността на хостово ниво гарантира, че всички вътрешни механизми по пътя на записите са включени в анализа, независимо от тяхната конкретна реализация.

3.5. Опашките като основен обект на наблюдение

В рамките на настоящото изследване опашките се разглеждат като основния механизъм, чрез който конкуренцията между операциите по запис се трансформира в наблюдаема деградация на латентността. При нарастващ паралелизъм пътят на записа преминава от режим на индивидуално обслужване към режим на споделено използване на ресурси, при който част от заявките неизбежно изчакват. Именно това изчакване, а не самото време за обработка, формира критичните забавяния в системата. [3, 17].



Фигура 12. Концептуална схема на пътя на записа с обозначени логически точки на натрупване на заявки, релевантни за проведените експерименти

Концептуалната схема, представена на Фигура 12, илюстрира пътя на записа с обозначени логически точки на натрупване на заявки, релевантни за проведените експерименти. Показаните индикатори не представят конкретни имплементации или вътрешни механизми, а служат като аналитичен модел за локализиране на местата, в които конкуренцията, въведена чрез натоварването, се проявява като опашкови ефекти в измерваната латентност.

Опашковите ефекти не се проявяват линейно спрямо натоварването. При определени режими малки увеличения в степента на конкуренция водят до непропорционално нарастване на времето за изчакване за ограничен брой операции. В резултат разпределението на латентността се разширява, като средните стойности могат да останат относително стабилни, докато крайните части на разпределението се изменят значително. Това поведение е добре документирано в теорията на опашките и в анализа на системи с конкурентен достъп до ресурси и прави опашките особено подходящ обект за анализ в контекста на оптимизацията на пътя на записа [3, 5].

Поради невъзможността опашките да бъдат наблюдавани директно във всички етапи на пътя на записа, тяхното наличие и динамика се изследват индиректно чрез разпределението на латентността, измерено на хостово ниво. Крайните персентили на латентността се използват като прокси-индикатор за натрупването на заявки и за

идентифициране на режими, в които системата преминава от стабилно към нестабилно поведение [9, 17].

В този контекст **оптимизацията се формулира като задача за ограничаване на растежа на опашките и за стабилизиране на разпределението на латентността при конкуренция**. Намаляването на средната латентност само по себе си не е достатъчно, ако е съпроводено с удължаване на опашката на разпределението и с увеличаване на вероятността от редки, но критични забавяния. Поради това ефективността на оптимизационните подходи се оценява спрямо тяхната способност да ограничават крайните стойности на латентността [3, 5].

Фокусирането върху опашките като основен обект на наблюдение позволява експериментите да бъдат интерпретирани като изследване на преходите между различни режими на работа на пътя на записа. Анализът на измененията в разпределението на латентността при контролирано натоварване създава основа за идентифициране на оптимизационни стратегии, насочени към подобряване на стабилността и предсказуемостта на системата.

3.6. Експериментални параметри и конфигурация

Експерименталният дизайн в настоящото изследване е изграден върху ясно разграничение между фиксирани параметри и целенасочено изменяни независими променливи. Всички експериментални сценарии се изпълняват в рамките на един и същ експериментална среда, като между отделните тестове се променят единствено параметри, които влияят директно върху натиска върху входно-изходните опашки по пътя на записите. Този подход осигурява съпоставимост на резултатите и позволява наблюдаваните изменения в латентността да бъдат интерпретирани като следствие от контролираните промени, а не от вариации в средата [17].

Фиксираните параметри дефинират базовия експериментален профил и остават непроменени във всички сценарии. Те обхващат характеристиките на средата, работния набор, базовия профил на натоварване и условията на измерване.

категория	параметър	стойност
Хост (инициатор)	Брой хостове	1
	Тип достъп	Блоков достъп до SAN (без файлови системи)

категория	параметър	стойност
SAN транспорт	Операционна система	Linux-базирана
	Генериране на натоварване	fio
	Multipath	Активен (2 независими I/O пътеки)
	Технология	Fibre Channel
	Свързаност	2 × 16 Gbit/s
Система за съхранение	Топология	Редундантна, без single-path зависимости
	Архитектура	Двуконтролерна SAN система
	Write cache	Активен
Работен набор	Back-end	Flash-базирана среда
	Логически капацитет	1 TB
	Работен набор	256 GB
Натоварване (baseline)	Тип операции	100% write
	Размер на блока	4 KB random write
Измерване	Продължителност на тест	300 s
	Warm-up период	30 s
	Повторяемост	3 повторения

Таблица 1. Фиксирани параметри на експерименталния дизайн

Изборът на работен набор с размер 256 GB цели предотвратяване на доминиращи ефекти от отложено записване чрез кеша, така че наблюдаваната латентност да отразява реалното натоварване по пътя на записите, а не временна буферизация. Аналогично, наличната пропускателна способност на транспортния слой значително надвишава използваната по време на експериментите, което изключва транспорта като лимитиращ фактор. При непроменена конфигурация на системата за съхранение това позволява влиянието на натоварването (паралелизъм, дълбочина на опашките и времева структура на заявките) върху енд-ту-енд латентността да бъде анализирано без смесване с ефекти от транспортни ограничения или кеширащи механизми [3, 17].

Обосновка на базовите фиксирани параметри

Базовите фиксирани параметри в настоящото изследване са избрани с цел дефиниране на стабилен и недвусмислен експериментален контекст, в рамките на който влиянието на натоварването и паралелизма върху пътя на записите може да бъде изследвано изолирано. Тези параметри не са обект на оптимизация, а служат като отправна рамка за всички експериментални сценарии [50, 52].

Логически капацитет (1 TB)

Логическият капацитет представлява виртуален блоков том, предоставен на хоста от системата за съхранение и използван като адресно пространство за входно-изходните операции. Този том не отразява директно физическото разпределение на данните или вътрешната организация на системата за съхранение, а служи като логическа абстракция, върху която се реализират експериментите. Избраният размер осигурява достатъчно голям адресен диапазон, който позволява дефиниране на работен набор без изкуствени ограничения, произтичащи от малък размер на тома [4].

Работен набор (256 GB)

Работният набор се дефинира като подмножество от логическия адресен диапазон на тома, в рамките на което се генерират операциите по запис по време на експериментите. По този начин работният набор определя активния адресен обхват на натоварването, докато логическият капацитет задава общото налично пространство. Това разграничение позволява моделиране на устойчиво натоварване върху записите, без да се изисква адресиране на целия логически том при всяко изпълнение [4, 40].

Базов профил на натоварване (4 KB random write).

Профилът 4 KB random write се използва като референтен сценарий, тъй като представлява минимална по размер операция по запис, при която ефектите от конкуренция и опашково натрупване се проявяват ясно. Малкият размер на блока свежда до минимум ефектите на агрегиране и позволява поведението на латентността да бъде анализирано основно като функция на паралелизма и дълбочината на опашките. Всички останали профили (напр. по-големи блокове или различен модел на достъп) се разглеждат като целенасочени отклонения от този базов сценарий. Профилът 4 KB random write представлява референтната стойност на параметъра „размер на блока“, спрямо която се анализират всички останали експериментални вариации [1, 3, 4, 6].

Продължителност на теста (300 s) и warm-up период (30 s).

Продължителността на експериментите е избрана така, че да осигури събиране на достатъчно голям брой индивидуални измервания на латентността за статистически

надежден анализ на разпределението и крайните персентили. Warm-up периодът служи за стабилизиране на системата и елиминиране на преходни ефекти в началото на изпълнението; данните от този период не се включват в анализа [3, 5].

Повторяемост (3 независими повторения).

Всеки експериментален сценарий се изпълнява в три независими повторения при идентични условия. Повторяемостта позволява да се оцени устойчивостта на наблюдаваното поведение и да се разграничат систематичните опашкови ефекти от случайни флуктуации. Анализът се фокусира върху консистентността на разпределението на латентността и ключовите персентили между повторенията, а не върху единични екстремни стойности [3, 5].

Изборът на фиксирани параметри цели да минимизира влиянието на вторични фактори и да осигури съпоставимост между отделните експериментални сценарии. По този начин всички наблюдавани изменения в поведението на латентността могат да бъдат интерпретирани като резултат от контролираните промени в натоварването и паралелизма.

Променливи параметри и експериментални лостове

Променливите параметри в експерименталния дизайн са избрани така, че да влияят директно върху степента на конкуренция и натиска върху входно-изходните опашки по write path-a. Чрез контролираното им изменение се модулира броят и характеристиките на едновременно активните операции по запис, без да се променя експерименталната среда или базовият контекст, дефиниран в предходната подточка.

Всеки експериментален сценарий изменя точно една независима променлива спрямо базовия профил, като всички останали параметри остават фиксирани. По този начин ефектът от всяка промяна може да бъде изолиран и анализиран самостоятелно.

Параметър	Стойности / диапазон	Роля в експерименталния модел
Дълбочина на опашките (QD)	32, 64, 128	Управление на степента на конкуренция (брой едновременно активни заявки)
Времева структура на натоварването	равномерно / пулсиращо (burst)	Провокиране на преходни режими и „крайни“ латентности при конкуренция

Параметър	Стойности / диапазон	Роля в експерименталния модел
<i>Координация на подаването</i>	некоординирано / координирано	Контрол върху пренасищането и синхронизация между конкурентни операции
<i>Интерференция от конкурентен хост (Noisy Neighbor)</i>	Без / с допълнителен хост	Моделиране на външна конкуренция при споделен ресурс

Таблица 2. Променливи параметри в експериментите

Обосновка на променливите параметри

Променливите параметри в експерименталния дизайн са избрани така, че да позволят систематично управление на конкуренцията и натиска върху входно-изходните опашки по пътя на записите. Всеки от тях представлява независим експериментален „лост“, чрез който се изменя броят, времевата организация или координацията на едновременно активните операции по запис, без да се променя базовата конфигурация на системата за съхранение и транспортния слой [50, 4].

Дълбочина на опашките (queue depth).

Дълбочината на опашките определя максималния брой операции по запис, които могат да бъдат едновременно активни от страна на хоста. Чрез увеличаване на този параметър се провокира нарастване на конкуренцията за ресурси по пътя на записите и се създават условия за натрупване на заявки в опашките. Това позволява изследване на нелинейни ефекти и преходи между стабилни и нестабилни режими на работа при нарастващо натоварване [4, 40].

Времева структура на натоварването.

Времевата структура на натоварването определя начина, по който заявките за запис се разпределят във времето. При равномерно натоварване заявките постъпват с приблизително постоянна интензивност, докато пулсиращото (burst) натоварване въвежда кратки периоди на силно концентрирано подаване. Чрез сравнение на тези режими се анализира чувствителността на опашковото поведение и латентността към преходни състояния при еднакъв среден профил на натоварване [6, 40].

Координация на подаването.

Координацията на подаването описва степента, в която конкурентните операции по запис са синхронизирани или ограничавани при достъпа до споделени ресурси. Некоординираното подаване позволява едновременно активиране на множество

операции, докато координираният режим въвежда контрол върху момента и интензитета на подаване. Този параметър позволява да се изследва влиянието на синхронизацията върху натрупването в опашките и стабилността на латентността [7, 40].

Интерференция от конкурентен хост (Noisy Neighbor).

Интерференцията от конкурентен хост моделира наличие на външно натоварване, генерирано от независим инициатор към същия сторидж ресурс. Чрез въвеждане на допълнителен хост се изследва как споделянето на ресурси и липсата на глобална координация влияят върху латентността и опашковите ефекти на основния поток от операции по запис [54, 40].

Методологично уточнение

Параметри като размер на блока (4 KB), тип на операциите (100% запис), брой паралелни задачи (numjobs) и конфигурация на системата за съхранение и транспортния слой се поддържат фиксирани във всички експерименти с цел елиминиране на допълнителни фактори и изолиране на влиянието на варираните параметри [50].

За яснота и еднозначност на експерименталния дизайн, Таблица 3 обобщава връзката между отделните експерименти и параметрите, които се изменят целенасочено във всеки от тях спрямо базовия профил.

В настоящото изследване базовият експериментален сценарий (Раздел 4.1) дефинира референтното поведение на пътя на записите при фиксиран експериментален контекст. Всички последващи експерименти (Раздели 4.2-4.5) представляват целенасочени вариации спрямо този референтен профил, като във всеки от тях се изменя един основен параметър.

Сценарий	Дълбочина на опашките (QD)	Форма на натоварването	Координация на достъпа	Конкурентен хост
<i>B – референтен</i>	–	–	–	–
<i>E1 (4.2)</i>	изменя се	–	–	–
<i>E2 (4.3)</i>	–	изменя се	–	–
<i>E3 (4.4)</i>	–	–	изменя се	–
<i>E4 (4.5)</i>	–	–	–	изменя се

Таблица 3. Матрица „експеримент → променен параметър“
 „изменя се“ – параметърът е целенасочено променян спрямо базовия профил
 „–“ – параметърът остава фиксиран

Таблица 3 показва, че експерименталният дизайн следва принципа на поетапно и изолирано надграждане спрямо референтния сценарий. Това позволява ефектът от всяка независима променлива да бъде анализиран самостоятелно, без смесване на влияния между отделните експериментални фактори.

3.7. Метрики и измервателен подход

Основната метрика в настоящото изследване е латентността на операциите по запис, измерена като крайното (енд-ту-енд) време за изпълнение на всяка индивидуална входно-изходна операция. Латентността се дефинира като времето между подаването на заявка за запис от хостовия слой и получаването на потвърждение за нейното завършване. Измерването обхваща целия път на записите и не зависи от вътрешните имплементационни механизми на отделните компоненти, тъй като се извършва изцяло на хостово ниво [3, 4, 40].

В рамките на настоящото изследване устойчиво латентностно поведение се приема за такова, при което по-голямата част от операциите се изпълняват в порядъка на десетки до стотици микросекунди, без съществено разширяване на високите латентностни персентили при увеличаване на натоварването. За признак на латентностна нестабилност се приема появата на крайни латентности в порядъка на милисекунди, които формират разширена латентностна опашка и водят до значително нарастване на p95 и p99, независимо от запазената пропускателна способност.

Поради това персентилите на латентността, и по-специално p95 и p99, се използват като основни аналитични показатели за оценка на стабилността и опашковото поведение на записите. Всички последващи анализи на латентността се основават на тези персентилни стойности и тяхната динамика при различни експериментални сценарии [3, 40].

Латентността се измерва в микросекунди (μs), като при необходимост стойностите се агрегират и интерпретират в милисекунди (ms) за целите на анализа и визуализацията. Използването на микросекундна резолюция позволява да бъдат отчетени както нормалните времена за обслужване, така и редките, но значително по-дълги закъснения, които са характерни за опашковите ефекти при конкуренция [4].

Измерванията се извършват на хост ниво, като за всяка операция по запис се регистрира индивидуална стойност на латентността. По този начин всяка изпълнена I/O операция генерира една измервателна точка. При експеримент с продължителност 3 минути и средна честота от порядъка на:

10 000 IOPS → приблизително 1.8 милиона измерени латентности

20 000 IOPS → приблизително 3.6 милиона измерени латентности

50 000 IOPS → приблизително 9 милиона измерени латентности

Този обем от данни позволява статистически стабилен анализ на разпределението на латентността и надеждно извеждане на крайните персентили.

Събраните латентности се анализират като разпределение, без предварително изглаждане или агрегация. Основен фокус са крайните персентили (например p95 и p99), които представят поведението на най-бавните операции и служат като индикатор за наличие и степен на опашкови ефекти. Средната латентност се използва само като ориентир и не се разглежда като достатъчна метрика за оценка на стабилността на write path-a.

В рамките на настоящото изследване устойчиво латентностно поведение се приема за такова, при което по-голямата част от операциите по запис се изпълняват в сравнително тесен диапазон на латентност, без съществено нарастване на високите латентностни персентили (p95 и p99) при увеличаване на натоварването. За признак на латентностна нестабилност се приема значителното разширяване на латентностното разпределение и появата на операции с латентност в порядъка на милисекунди, които формират дълга латентностна опашка и водят до съществено увеличение на крайните персентили, независимо от запазената пропускателна способност.

За да се гарантира съпоставимост между експериментите, латентностните измервания се разглеждат в контекста на стабилно натоварване. Агрегирани показатели като IOPS и пропускателна способност се използват за потвърждение, че входният поток остава еквивалентен между различните сценарии, но не се използват като критерий за оптимизация.

По този начин латентността се използва като основен инструмент за оценка на оптимизационните ефекти. Анализът на милиони индивидуални измервания позволява да се идентифицират режими, при които write path-ът преминава от стабилно към нестабилно поведение, и да се оцени ефектът от различни експериментални условия върху крайните забавяния [3, 40].

Метрика	Типични стойности	Основно предназначение	Интерпретация
Средна латентност ¹¹	0.8–2.5 ms	Общ ориентир	Не е достатъчна за оценка на стабилността
p50	0.7–2.0 ms	Типично поведение	Представя латентността на повечето операции
p95	4–15 ms	Ранно откриване на деградация	Показва началото на опашковите ефекти
p99	10–35 ms	Оценка на крайната латентност	Най-чувствителен индикатор за координационен стрес
Максимална латентност	>30–40 ms	Анализ на единични екстремни случаи	Характерна за силно натрупване на заявки или конкурентен достъп

Таблица 4. Наблюдавани диапазони на латентностните показатели в експерименталната среда

Освен латентността, по време на експериментите се регистрират още пропускателната способност (Throughput) и честотата на операциите (IOPS). Тези показатели се използват като контролни метрики за проверка на съпоставимостта на натоварването между отделните експериментални сценарии, но не се използват като критерий за оценка на оптимизационния ефект.

3.8. Събиране на логове и метрики

В рамките на експерименталната методология събирането на данни се извършва чрез инструмента `fio`, който предоставя както агрегирани показатели за производителност и латентност, така и времеви логове за анализ на динамиката на натоварването. Целта на събирането на логове е да се осигури възпроизводима и количествено обоснована основа за анализ на латентностното поведение при различни експериментални сценарии [3, 53].

За всяко изпълнение на експеримент се генерира JSON файл с обобщени резултати, съдържащ показатели за производителност (IOPS и пропускателна способност) и статистики за латентността на операциите по запис. Особено внимание се обръща на крайните латентностни персентили (p50, p95 и p99), които характеризират поведението на системата извън типичния режим на работа и са използвани като основни метрики в анализа на експерименталните резултати [53].

Допълнително се генерират времеви логове за латентност, IOPS и пропускателна способност с фиксиран времеви прозорец за усредняване. Тези логове позволяват анализ

¹¹ Диапазоните са получени при проведените експериментални сценарии и не представляват универсални референтни стойности за SAN системи.

на устойчивостта на латентността във времето и идентифициране на краткотрайни отклонения, които не могат да бъдат наблюдавани чрез агрегирани показатели. Времевият прозорец за усредняване е избран така, че да осигури баланс между детайлност и шум в измерванията.

Първоначалният период за стабилизиране на системата (warm-up) не се включва в анализа, с цел елиминирание на преходни ефекти, свързани с инициализация на кешове и алокация на ресурси. Всеки експериментален сценарий се изпълнява многократно, като вариацията между повторенията се използва за оценка на стабилността и предсказуемостта на крайните латентностни стойности [50].

Събирането на сурови пер-операция трасета не се извършва, тъй като целта на изследването е анализ на агрегираното и крайното поведение на латентността в SAN среда, а не детайлна реконструкция на индивидуални входно-изходни операции. Използваният подход осигурява достатъчна информативност за изследване на ефектите от паралелизма и управлението на опашките, при разумен обем на събраните данни [3, 40].

Примерна конфигурация на използвания `fio` job файл е представена в Приложение А.

3.8. Логика и последователност на експериментите

Експериментите в настоящото изследване са организирани като последователна изследователска линия, а не като независими или изолирани тестове. Подредбата им цели постепенно разкриване и усилване на факторите, които влияят върху стабилността на `write path` -а и опашковото поведение на латентността. Всеки следващ експеримент надгражда върху предходния, като въвежда нов източник на конкуренция или усложнява вече наблюдаваното поведение.

Базовият експериментален сценарий служи като референтна точка за всички последващи анализи. Той дефинира изходното поведение на системата при фиксиран експериментален контекст и минимална степен на конкуренция. Резултатите от този сценарий не се използват за оптимизационни изводи сами по себе си, а като отправна линия, спрямо която се интерпретират всички последващи изменения в латентността и нейното разпределение.

В първия експеримент се изследва влиянието на паралелизма чрез увеличаване на дълбочината на опашките. Този етап представлява контролирано „усилване“ на натоварването и позволява да се наблюдава как `write path` -ът реагира на нарастващ брой

едновременно активни операции. Така се идентифицират първите признаци на нелинейно поведение и разширяване на опашката на разпределението на латентността.

Следващият експеримент въвежда промяна във формата на натоварването, като акцентът се измества от чистия паралелизъм към времевата и пространствената структура на заявките. Чрез този подход се анализира доколко наблюдаваните опашкови ефекти зависят от начина, по който операциите се подават във времето, а не само от техния брой. Това позволява разграничаване между ефекти, породени от степента на конкуренция, и такива, свързани с характеристиките на самото натоварване.

В следващия етап експерименталната линия се насочва към координацията на достъпа. Тук се изследва взаимодействието между конкурентни потоци от операции и ефектът от тяхната синхронизация или липса на такава. Този експеримент цели да покаже, че максималната пропускателна способност не винаги е еквивалентна на стабилно и предсказуемо латентностно поведение.

Финалният експеримент въвежда външна конкуренция чрез допълнителен хост, споделящ същия логически том. По този начин експерименталната среда се доближава до реалистични многопотребителски сценарии, при които write path-ът е подложен на комбиниран натиск от няколко източника. Този етап позволява да се анализира устойчивостта на системата при наличие на noisy neighbor ефекти и да се оцени кумулативното влияние на вече изследваните фактори.

Чрез тази последователност експериментите формират логически свързана изследователска линия, в която всеки сценарий допринася за постепенното изграждане на цялостна картина на поведението на write path-a при конкуренция. Това позволява резултатите в следващата глава да бъдат интерпретирани не като отделни наблюдения, а като етапи от единен аналитичен процес.

3.9. Какво не се измерва и защо

Настоящото изследване съзнателно ограничава обхвата на измерваните показатели и наблюдаваните компоненти с цел постигане на яснота и еднозначност при интерпретацията на резултатите. Изключените параметри не се разглеждат като незначителни, а като фактори, които умишлено не се включват, тъй като биха въвели допълнителни зависимости и биха затруднили изолирането на изследваните ефекти.

В рамките на експериментите не се измерват вътрешни метрики на системата за съхранение, като натоварване на контролерите, вътрешни опашки, кешови попадения или

процеси по управление на физическите носители. Системата за съхранение се разглежда като логическа „черна кутия“, при която се анализира входно-изходното поведение на системно ниво, без обвързване с конкретна вътрешна реализация [4, 40].

Не се анализира и използването на процесорни, паметови или мрежови ресурси на хостовата система. Хостът не е обект на оптимизация, а средство за генериране на контролирано натоварване. По този начин се избягва смесване на ефекти, свързани с конфигурацията на инициатора, с поведението на пътя на записите като система.

В експериментите не се включват операции по четене, смесени read/write натоварвания или файлови системи. Това позволява фокусиране изцяло върху операциите по запис и тяхната стабилност при конкуренция, без влияние от различни механизми за обработка на входно-изходни операции [16, 20].

Използваният генератор на натоварване (fio) не се разглежда като обект на анализ; той се използва единствено като средство за възпроизводимо управление на паралелизма и профила на натоварване.

Не се цели постигане на максимална пропускателна способност или абсолютна производителност. Фокусът на изследването е върху латентността, нейното разпределение и стабилност при различни режими на натоварване, което съответства на практическите изисквания за предсказуемо поведение на системите за съхранение [40].

3.10. Методологично обобщение

В настоящата глава беше дефинирана методологичната рамка на изследването, включваща експерименталния контекст, фиксираните и променливите параметри, логиката на експерименталната последователност и границите на измерване. Чрез ясно разграничаване между базовия референтен сценарий и целенасочените експериментални вариации се осигурява съпоставимост и възпроизводимост на резултатите.

Експерименталният дизайн е изграден върху принципа на поетапно надграждане, при което всеки експеримент въвежда нов източник на конкуренция или усложнява вече наблюдаваното поведение. Този подход позволява отделните фактори да бъдат анализирани както самостоятелно, така и в контекста на тяхното взаимодействие.

Ограничаването на обхвата на измерванията и фокусът върху пътя на записа като цялостна функционална верига позволяват латентността да бъде анализирана като системен феномен, а не като изолирана характеристика на отделен компонент. Така резултатите от експериментите могат да бъдат интерпретирани в контекста на

оптимизация на стабилността и предсказуемостта на входно-изходното поведение, което представлява основната цел на настоящото изследване.

С това Глава 3 създава необходимата основа за анализа на резултатите в следващата глава, като дефинира ясно както възможностите, така и ограниченията на експерименталния подход.

ГЛАВА 4. ЕКСПЕРИМЕНТАЛНИ РЕЗУЛТАТИ

Настоящата глава представя резултатите от серия контролирани експерименти, проведени в рамките на експерименталната среда и методологията, дефинирани в предходната глава. Целта е да се изследва поведението на латентността на операциите по запис в SAN среда като функция на различни характеристики на входно-изходното натоварване, при фиксирана системна конфигурация и непроменен път на заявките (host → transport → storage). Експериментите са организирани като последователна ескалация на натиска върху системата, при която се изменят отделни аспекти на натоварването - паралелизъм, времева структура, координация и външна конкуренция - с цел да се наблюдава тяхното влияние върху формата на разпределението на латентностите и поведението на крайните стойности.

Анализът започва с дефиниране на базов експериментален сценарий, който установява референтното поведение на системата при ненаситен режим на работа и стабилно натоварване. Този сценарий служи като отправна точка за интерпретация на всички последващи резултати, като позволява наблюдаваните изменения в латентността да бъдат разглеждани спрямо ясно дефинирано „нормално“ поведение. В следващите раздели експериментите надграждат този базов сценарий чрез контролирано изменение на отделни параметри, без промяна на експерименталния контекст, което осигурява съпоставимост и последователност на резултатите.

Всички експериментални сценарии, представени в дисертационния труд, са изпълнявани по три пъти, при идентични конфигурационни условия с цел ограничаване влиянието на случайни флуктуации и фонов системни ефекти. За всеки сценарий латентностните наблюдения са агрегирани на ниво разпределения, което позволява анализ както на типичното, така и на крайното поведение на операциите по запис.

Анализът се фокусира върху латентностните разпределения и крайните персентилни стойности, като основна метрика е използван 99-ти персентил (p99). Този показател е избран поради способността му да улавя редки, но експлоатационно значими

закъснения, които не се отразяват в средните стойности, но оказват съществено влияние върху стабилността и качеството на обслужване в SAN среди.

4.1. Базов експериментален сценарий (референтно поведение)

4.1.1. Цел на експеримента

Базовият експериментален сценарий има за цел да фиксира референтен режим на работа на системата при операции по запис, характеризиращ се с ненаситено натоварване и стабилен входно-изходен паралелизъм. В този сценарий системата се наблюдава в условия, при които не се цели нито максимизиране на производителността, нито умишлено въвеждане на ограничения, а възпроизвеждане на типично поведение, близко до това в реална продукционна среда.

Основната функция на базовия сценарий е да дефинира изходно ниво на латентност и нейната вариация, при което взаимодействието между паралелизма, опашките и механизмите за обслужване не е доминирано от насищане или екстремни режими на работа. Това позволява формата на разпределението на латентностите да бъде анализирана като „нормално“ поведение на системата, върху което впоследствие се наслагват ефектите от контролираните изменения на параметрите.

В този смисъл базовият експеримент не се разглежда като носител на самостоятелни изводи, а като стабилна референтна основа, спрямо която се интерпретират резултатите от всички последващи експериментални сценарии в настоящата глава.

4.1.2. Конфигурация и натоварване

Базовият експериментален сценарий се изпълнява при фиксирана системна конфигурация, идентична за всички експерименти, разгледани в настоящата глава. Параметрите на хоста, транспортния слой и системата за съхранение не се изменят, като единствените контролирани величини са профилът на натоварването и входно-изходният паралелизъм.

Натоварването се състои от 100% операции по запис с произволен достъп и размер на блока 4 KB. Операциите се генерират върху логически том с капацитет 1 TB, като активният работен набор е ограничен до 256 GB. По този начин измерената латентност отразява поведението на системата при обработка на конкурентни записи, без доминиране на кеширащи ефекти.

Входният паралелизъм е фиксиран чрез използване на четири паралелни задачи ($\text{numjobs} = 4$) и дълбочина на опашката $\text{QD} = 32$. Тази конфигурация осигурява стабилен и ненаситен поток от заявки към системата за съхранение и служи като референтна точка за анализ на латентностното поведение.

Всеки тест се изпълнява с продължителност 300 секунди, като първите 30 секунди се използват като период за стабилизиране и не се включват в анализа. За осигуряване на повторемост експериментът се изпълнява три пъти, а представените резултати отразяват усредненото поведение.

Така дефинираните условия определят базовия експериментален сценарий, върху който в следващите подраздели се представят и анализират референтните резултати за латентност и производителност.

Параметър	Стойност
<i>Тип операции</i>	100% write
<i>Модел на достъп</i>	Random
<i>Размер на блока</i>	4 KB
<i>Логически том</i>	1 TB
<i>Работен набор</i>	256 GB
<i>Брой паралелни задачи</i>	$\text{numjobs} = 4$
<i>Дълбочина на опашката</i>	$\text{QD} = 32$
<i>Продължителност на тест</i>	300 s
<i>Warm-up период</i>	30 s
<i>Повторемост</i>	3 изпълнения

Таблица 5. Параметри на базовия експериментален сценарий

Всички последващи експерименти използват параметрите от базовия сценарий (Таблица 5), освен ако изрично не е посочено друго.

4.1.3. Резултати: латентност и производителност

В този подраздел се представят резултатите от базовия експериментален сценарий, измерени след приключване на периода за стабилизиране и усреднени върху три независими повторения. Представените данни характеризират референтното поведение на системата при операциите по запис.

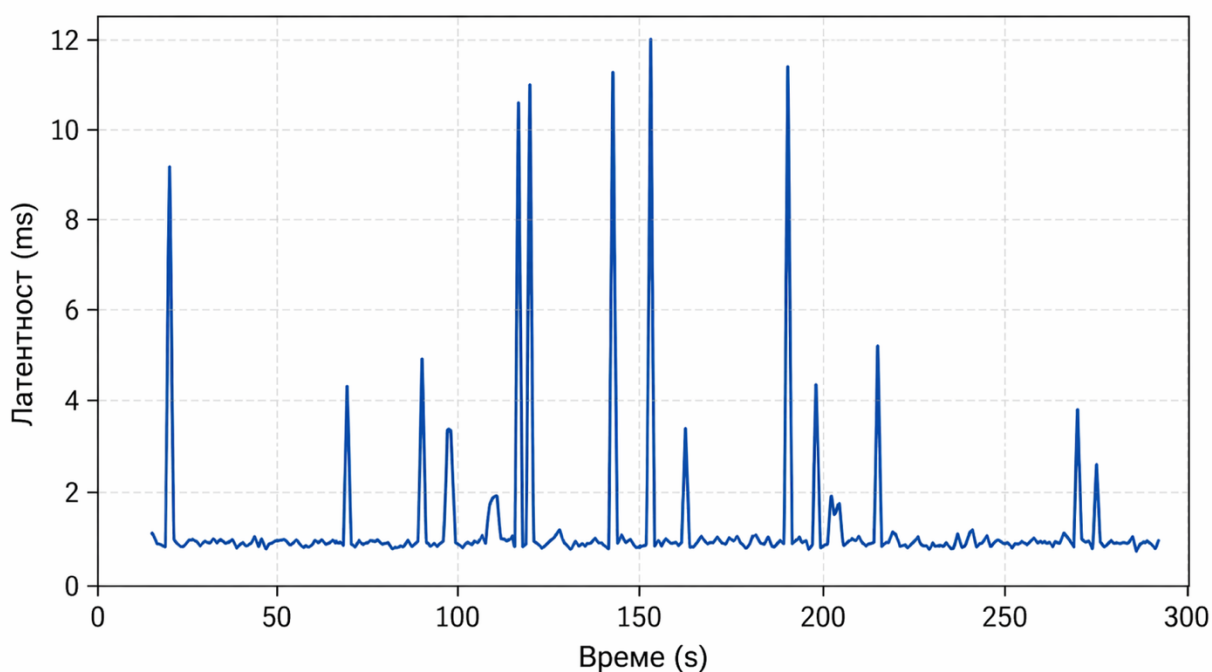
За количествено описание на наблюдаваното поведение са използвани следните метрики: средна латентност, медиана (p50), крайни стойности на латентността (p95 и p99),

както и показатели за производителност - IOPS и пропускателна способност. Усреднените стойности на тези метрики са обобщени в Таблица 6.

Метрика	Стойност
Средна латентност	0.8 ms
Медиана (p50)	0.6 ms
p95 латентност	3.9 ms
p99 латентност	11.7 ms
IOPS	≈ 42 000
Пропускателна способност	≈165 MB/s

Таблица 6. Измерени метрики за базовия експеримент

Фигура 13 показва времевия профил на латентността на операциите по запис за базовия експериментален сценарий в анализирания интервал след периода на стабилизиране. Наблюдава се установен режим на работа, при който латентността се колебае около характерното си ниво без продължителни периоди на деградация или ясно изразени трендове във времето.



Фигура 13. Времеви профил на латентността на операциите по запис при базовия експериментален сценарий

В профила се регистрират отделни краткотрайни пикове на латентността, които са с ограничена продължителност и не доминират общото поведение на системата. Това

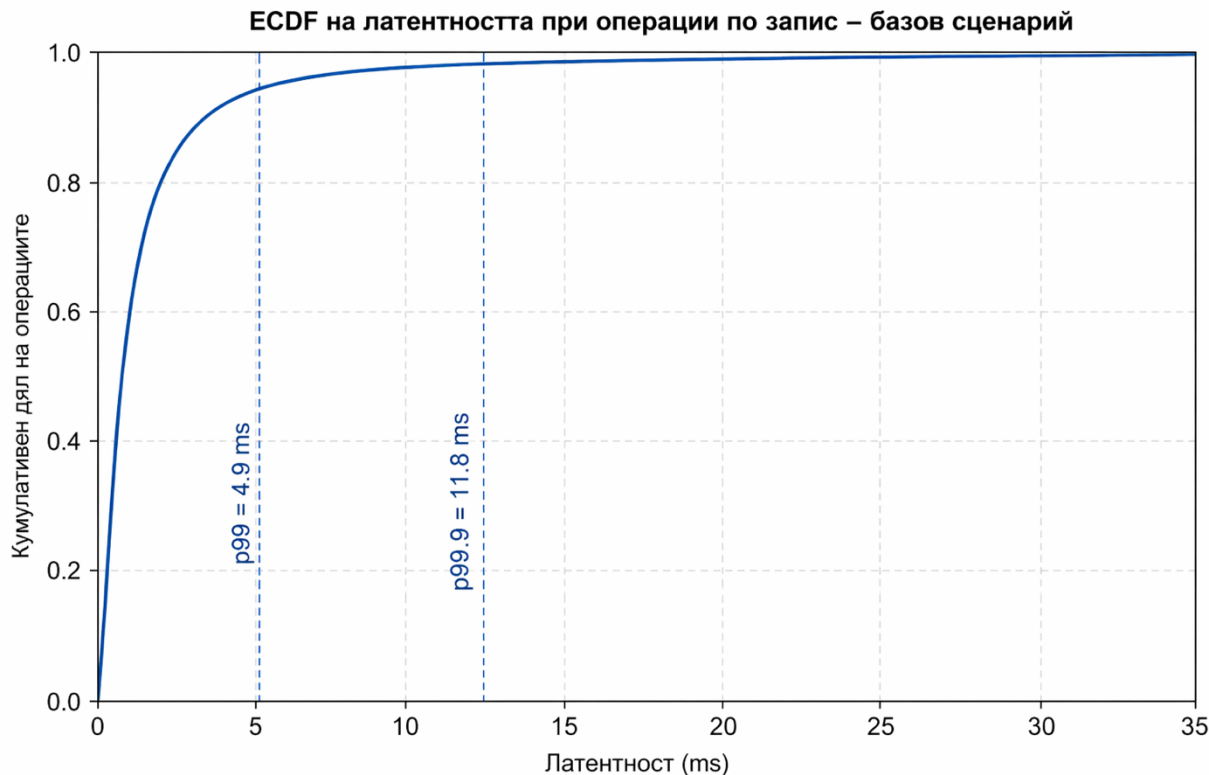
показва, че измерените крайни стойности на латентността не са резултат от единичен продължителен инцидент, а се проявяват като редки събития в рамките на иначе стабилен времеви режим.

В този смисъл базовият експериментален сценарий дефинира стабилен референтен режим, спрямо който в следващите раздели се анализират ефектите от увеличен паралелизъм, промяна на времевата структура на натоварването и наличие на конкурентен достъп.

4.1.4. Разпределение на латентностите (ECDF)

За по-пълно представяне на поведението на латентността при базовия експериментален сценарий резултатите се анализират чрез емпирична кумулативна функция на разпределението (ECDF). Този тип визуализация показва дела на операциите по запис, които приключват под дадена стойност на латентността, и позволява едновременно наблюдение на типичното поведение и на редките, но значими отклонения [9].

Фигура 14 представя ECDF на латентностите на операциите по запис за базовия експериментален сценарий.



Фигура 14. ECDF на латентността на операциите по запис за базовия експериментален сценарий

Както се вижда от Фигура 14, формата на ECDF ясно се разделя на две характерни области. Основната част от разпределението („тялото“) е силно концентрирана в ниските стойности на латентността, като преобладаващата част от операциите по запис приключват за по-малко от 1 ms. Това съответства на ниските стойности на медианата и средната латентност, представени в Таблица 6.

В края на разпределението се наблюдава постепенно удължаване на кривата, което показва наличието на ограничен брой операции със значително по-висока латентност. Макар тези операции да представляват малка част от всички заявки, те оказват съществено влияние върху крайните персентили (p95 и p99) и са първи признак за формиране на опашкови ефекти. Именно тази особеност прави ECDF значително по-информативен инструмент за анализ в сравнение с използването единствено на усреднени показатели [9].

Формата на кривата ясно се разделя на две характерни области. Основната част от разпределението („тялото“) е силно концентрирана в ниските стойности на латентността, като голям дял от операциите по запис приключват под 1 ms. Това съответства на отчетените ниски стойности на медианата и средната латентност, представени в Таблица 6. [9].

Медианата на латентността (p50) се намира в началната, стръмна част на ECDF кривата и не е изрично маркирана на фигурата, тъй като основният интерес в настоящия анализ е насочен към формата и мащаба на опашката на разпределението. При този режим на работа p50 не допринася с допълнителна визуална информация спрямо табличното представяне на агрегатните метрики.

В същото време ECDF кривата показва наличие на удължена опашка на разпределението. Позицията на p95 маркера показва, че приблизително 95% от операциите по запис приключват с латентност под около 4 ms, докато p99 се намира значително по-надясно - при стойности около 12 ms. Това означава, че малък дял от операциите се обслужват с латентност, която надвишава типичното поведение с приблизително един порядък [3].

Разделението между „тялото“ и „опашката“ на разпределението е ясно изразено и не може да бъде изведено единствено от усреднените показатели. Докато средната стойност и медианата характеризират доминиращото поведение на системата, ECDF визуализацията показва, че крайните случаи представляват отделен режим на обслужване, който засяга ограничен, но не пренебрежим дял от операциите.

Представянето на латентността чрез ECDF подчертава необходимостта от анализ на разпределението, а не само на агрегирани метрики, при оценка на стабилността на

записите в SAN среда. Този подход позволява ясно разграничаване между типичното време за обслужване и редките закъснения, които имат съществено значение за приложения, чувствителни към латентност.

4.1.5. Обобщение на референтното поведение

Базовият експериментален сценарий дефинира изходното поведение на системата при операции по запис в SAN среда при фиксирана конфигурация и умерен входно-изходен паралелизъм. Представените резултати очертават ясно разграничение между типичното време за обслужване на записите и редките, но значими отклонения в латентността.

Усреднените метрики и медианата показват ниски стойности на латентността, което е индикация за стабилно и предсказуемо поведение при по-голямата част от операциите. Времевият профил на латентността (Фигура 13) допълва тази картина, като показва устойчив режим на работа във времето, без продължителни периоди на деградация или системни аномалии при зададеното натоварване.

В същото време анализът на разпределението на латентностите чрез ECDF разкрива наличие на удължена опашка, която не може да бъде изведена от агрегирани показатели като средна стойност или медиана. Позициите на p95 и p99 ясно показват, че малък дял от операциите по запис се обслужват с латентност, значително надвишаваща типичното поведение, въпреки стабилността на средните стойности [3].

Това референтно поведение показва, че дори при ненаситен режим на работа и привидно „добри“ усреднени показатели, латентността на записите не е еднородна и включва редки, но съществени крайни случаи. Базовият сценарий по този начин служи като необходима отправна точка за анализ на влиянието на паралелизма и натиска върху входно-изходните опашки, разгледани в следващите раздели.

Резултатите от настоящия подраздел не целят идентифициране на причините за наблюдаваното поведение, а фиксиране на неговата форма и мащаб. Това позволява всички последващи експериментални резултати да бъдат интерпретирани спрямо ясно дефинирано изходно състояние, без двусмислие относно „нормалното“ поведение на системата.

4.2. Експеримент 1: Влияние на дълбочината на опашката (QD) върху латентността на записите (моделно надграждане)

4.2.1. Цел на експеримента

Целта на настоящия експеримент е да се анализира влиянието на дълбочината на входно-изходната опашка (Queue Depth, QD) върху латентността на операциите по запис в SAN среда при непроменен системен контекст. Експериментът надгражда референтния сценарий, дефиниран в Раздел 4.1, като изолира ефекта на увеличения паралелизъм върху латентностното поведение [9], без въвеждане на допълнителни променливи.

Фокусът на анализа е върху изменението на разпределението на латентностите при постепенно увеличаване на QD и върху поведението на крайните стойности (p95 и p99), които отразяват редките, но критични случаи на забавяне. По този начин се изследва до каква степен нарастването на паралелизма води до деградация на латентността, която не може да бъде адекватно описана чрез усреднени показатели като средна стойност или медиана.

Експериментът не цели оптимизация или определяне на оптимална стойност на QD, а характеризиране на формата и мащаба на промените в латентностното поведение при контролирано увеличаване на натиска върху входно-изходните опашки. Получените резултати служат като основа за последващите експерименти, в които се разглеждат допълнителни фактори, влияещи върху координацията и обслужването на паралелни заявки в SAN среда.

В рамките на експеримента ECDF визуализацията беше използвана за качествен анализ на формата на латентностното разпределение. При сравнение между различни стойности на дълбочината на опашката обаче, крайните латентностни персентилен предоставят по-ясна и недвусмислена количествена характеристика на наблюдаваните изменения.

4.2.2. Конфигурация и натоварване

Експериментът е проведен при същата системна конфигурация и параметри на натоварването, използвани в референтния сценарий, описан в Раздел 4.1. Всички хардуерни и софтуерни компоненти, както и характеристиките на входно-изходните операции, се запазват непроменени с цел изолиране на влиянието на дълбочината на входно-изходната опашка като единствена независима променлива.

Натоварването се състои от операции по запис с фиксиран размер на блока и постоянен профил на достъп, идентичен с използвания в базовия експеримент. Работният набор от данни, продължителността на тестовете, фазата на стабилизиране (warm-up) и броят на повторенията са запазени без промяна, което осигурява директна съпоставимост на резултатите между експериментите.

Дълбочината на входно-изходната опашка се задава на нивото на генератора на натоварване и определя броя едновременно активни, неизпълнени заявки за входно-изходни операции [9]. В рамките на настоящия експеримент са разгледани три стойности на QD - 32, 64 и 128 - което реализира контролирано увеличаване на входния паралелизъм при непроменен системен контекст. Всички останали параметри на натоварването остават фиксирани.

Стойността $QD = 32$ съответства на референтния сценарий и служи като изходна точка за сравнение. Резултатите за $QD = 64$ и $QD = 128$ позволяват анализ на тенденциите в изменението на латентностното поведение при нарастващ паралелизъм, без да се правят твърдения за абсолютни количествени зависимости извън рамките на разглеждания експеримент.

По този начин експерименталната постановка осигурява ясно разграничение между контролирания параметър (QD) и всички останали фактори, което позволява наблюдаваните промени в латентностното поведение да бъдат приписани директно на ефекта от увеличената дълбочина на входно-изходната опашка.

Параметър	Базов сценарий	Експеримент 4.2
<i>Дълбочина на входно-изходната опашка (QD)</i>	32	32 / 64 / 128
<i>Брой паралелни задачи (numjobs)</i>	4	4
<i>Тип операции</i>	100% write	100% write
<i>Модел на достъп</i>	Random	Random
<i>Размер на блока</i>	4 KB	4 KB
<i>Работен набор</i>	256 GB	256 GB
<i>Продължителност на теста</i>	300 s	300 s

Таблица 7. Всички параметри, които не са изрично посочени, съвпадат с тези от базовия експериментален сценарий (Таблица 6)

4.2.3. Резултати

В настоящия подраздел са представени обобщените резултати за латентността и производителността на операциите по запис при различна дълбочина на входно-изходната опашка. Анализът е структуриран така, че да позволи директно сравнение с референтния сценарий, разгледан в Раздел 4.1, и да изведе ефекта от увеличения паралелизъм върху основните агрегирани метрики.

Дълбочина на опашката (QD)	Средна латентност (ms)	Медиана p50 (ms)	p95 (ms)	p99 (ms)	IOPS
32 (референтен сценарий)	0.8	0.6	3.9	11.7	42 000
64 (моделни данни)	0.9	0.7	6.2	19.5	48 000
128 (моделни данни)	1.1	0.8	9.8	34.0	52 000

Таблица 8. Метрики за латентност и производителност при различна дълбочина на входно-изходната опашка

При увеличение на входно-изходната опашка (Queue Depth, QD) от 32 до 64 се наблюдава умерено нарастване на средната латентност и медианата, което показва, че типичното време за обслужване на записите остава относително стабилно при по-висок паралелизъм. В същото време стойностите на p95 и p99 нарастват значително по-силно, което е индикация за по-честа поява на операции с висока латентност при натоварване на входно-изходните опашки.

При QD = 128 тази тенденция се засилва допълнително. Докато медианата и средната латентност продължават да нарастват плавно, крайните персентилю показват рязко увеличение, което не може да бъде адекватно описано чрез агрегирани показатели. Това поведение показва, че при висока степен на паралелизъм системата навлиза в режим,

при който обслужването на отделни заявки се влияе съществено от конкуренция в опашките и вътрешното планиране.

За по-ясна оценка на ефекта от увеличаването на дълбочината на опашката Таблица 9 представя относителното изменение на основните латентностни персентили спрямо референтния сценарий ($QD = 32$).

QD	p50 (×)	p95 (×)	p99 (×)
32	1.00×	1.00×	1.00×
64	1.17×	1.59×	1.67×
128	1.33×	2.51×	2.91×

Таблица 9. Относително изменение на латентностните персентили спрямо референтния сценарий

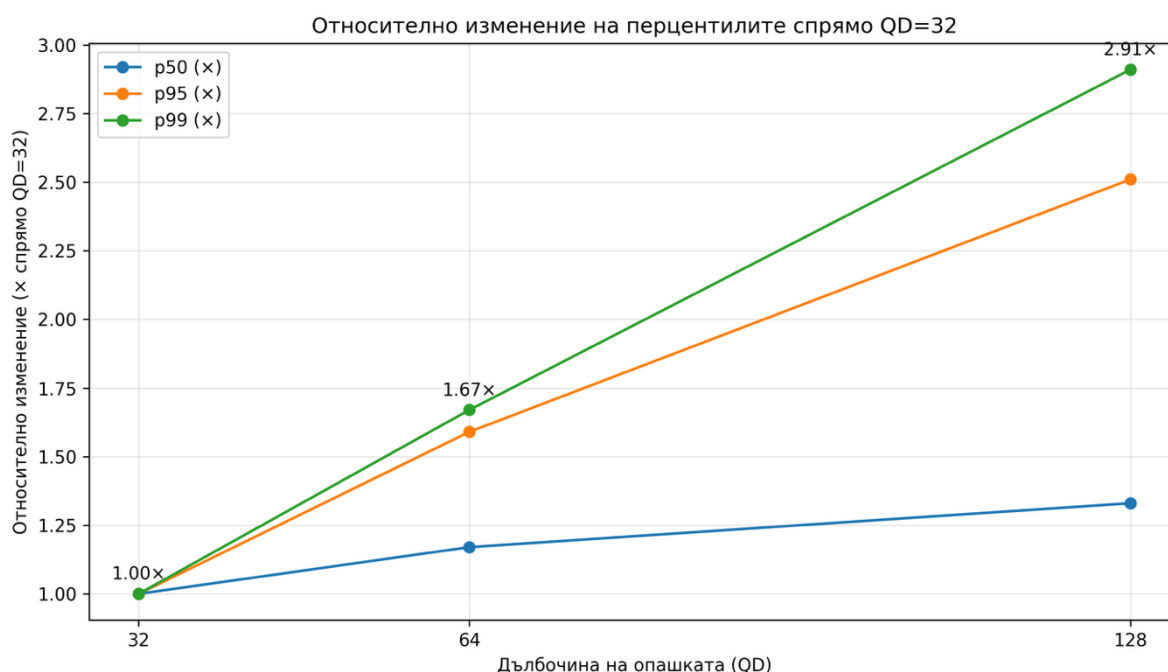
Данните ясно показват, че медианата на латентността нараства слабо дори при значително увеличение на дълбочината на опашката, докато p95 и p99 се увеличават значително по-бързо. При $QD = 128$ p99 нараства почти трикратно спрямо референтното поведение, което показва, че деградацията на латентността при повишен паралелизъм се проявява основно чрез усилване на крайните случаи, а не чрез равномерно влошаване на типичното време за обслужване.

Това разминаване между типичното и най-лошото наблюдавано поведение мотивира необходимостта от по-задълбочен анализ на разпределението на латентностите, разгледан в следващия подраздел.

4.2.4. Анализ на влиянието на дълбочината на опашката върху крайните латентностни случаи

Резултатите, представени в предходния подраздел, показват, че увеличаването на дълбочината на входно-изходната опашка води до нееднородно изменение на латентността на операциите по запис. Докато усреднените показатели и медианата (p50) се изменят сравнително слабо при нарастващ паралелизъм, крайните латентностни персентили p95 и p99 демонстрират значително по-силна чувствителност към увеличаването на QD.

Фигура 15 представя зависимостта между дълбочината на опашката и стойностите на основните латентностни персентили. Видно е, че при преминаване от референтния сценарий ($QD = 32$) към по-високи стойности на QD , медианата на латентността нараства плавно и остава в близки граници. За разлика от това, $p95$ и $p99$ нарастват нелинейно, като при $QD = 128$ стойността на $p99$ се увеличава почти трикратно спрямо референтното поведение.



Фигура 15. Относително изменение на латентностните персентили ($p50$, $p95$ и $p99$) спрямо дълбочината на входно-изходната опашка

Това разминаване между типичното и крайните случаи показва, че деградацията на латентността при увеличен паралелизъм не се проявява като равномерно влошаване на всички операции, а като усилване на редки, но значими събития с висока латентност. Именно тези крайни случаи доминират поведението на разпределението при високи стойности на QD и представляват основен риск за приложения, чувствителни към латентност.

Поради тази причина анализът в настоящия експеримент се фокусира върху поведението на крайните латентностни персентили, а не върху пълната форма на разпределението чрез ECDF визуализация. При сравнение на различни стойности на QD , персентилите $p95$ и $p99$ предоставят по-ясна и недвусмислена информация за ефекта от увеличавания паралелизъм върху стабилността на обслужването.

4.2.5. Крайни стойности на латентността (p99) като функция на QD

Дълбочина на опащката (QD)	p99 латентност (ms)
32	11.7
64	19.5
128	34.0

Фигура 16 представя зависимостта на p_{99} латентността от дълбочината на опашката при стойности на QD от 32, 64 и 128. Наблюдава се ясно изразено нарастване на p_{99} с увеличаване на QD, като темпът на това нарастване е значително по-висок от изменението на средната стойност и медианата, представени в Таблица 9. Това показва, че деградацията на латентността при повишен паралелизъм се концентрира в крайните случаи, а не в типичното време за обслужване.

При преход от $QD = 32$ към $QD = 64$ стойността на $p99$ се увеличава осезаемо, което показва, че дори умерено повишаване на паралелизма води до влошаване на най-лошите латентностни случаи. В същото време агрегатните показатели остават относително стабилни, което подчертава ограничената им информативност при оценка на системи, обслужващи паралелни заявки.

Увеличаването на QD до 128 води до още по-рязко нарастване на $p99$, което свидетелства за нелинейна зависимост между дълбочината на опашката и крайните стойности на латентността. Представените във фигурата вариации между независимите повторения на експеримента (error bars) показват, че при висока степен на паралелизъм не само се увеличава стойността на $p99$, но и нараства неговата изменчивост. Това е индикация за по-ниска предсказуемост на обслужването и за навлизане на системата в режим на латентностна нестабилност.

Съпоставката между Фигура 16 и анализа на разпределението в предходния подраздел показва последователна картина: увеличаването на QD води до доминиране на редки, но критични латентностни случаи, които определят поведението на $p99$, без да предизвикват равномерно изместване на цялото разпределение. Това потвърждава, че деградацията на латентността при повишен паралелизъм се проявява предимно чрез усилване на опашката на разпределението, а не чрез влошаване на типичното поведение.

Анализът на $p99$ като функция на дълбочината на входно-изходната опашка предоставя компактно и интуитивно обобщение на ефекта от увеличения паралелизъм върху латентността на записите. Поради своята чувствителност към крайните случаи, този показател се утвърждава като ключов индикатор за оценка на риска от редки, но критични забавяния в SAN среда.

4.2.6. Обобщение на Експеримент 1

Експеримент 1 показва, че дълбочината на входно-изходната опашка е ключов фактор за формирането на крайните латентностни случаи при операции по запис в SAN среда. При увеличаване на QD типичното време за обслужване на заявките, характеризирано чрез медианата и средната стойност, остава относително стабилно, докато крайните латентностни персентили $p95$ и $p99$ нарастват значително.

Анализът на разпределението на латентностите и на поведението на $p99$ като функция на QD показва, че деградацията на латентността при повишен паралелизъм не се проявява като равномерно влошаване на всички операции, а като усилване на редки, но критични случаи с висока латентност. При по-високи стойности на QD се наблюдава и

нарастване на вариацията на p_{99} между повторенията, което е индикация за по-ниска предсказуемост и латентностна нестабилност.

Тези резултати показват, че оценката на производителността на SAN системи, базирана единствено на усреднени показатели, може да прикрие съществени рискове за приложения, чувствителни към крайните закъснения. Поради това p_{99} се утвърждава като основна метрика за анализ на влиянието на паралелизма върху латентността и служи като отправна точка за последващите експерименти, в които се разглеждат допълнителни фактори, влияещи върху поведението на опашките и координацията на заявките.

В следващия експеримент този базов ефект от увеличения паралелизъм се използва като изходна предпоставка за анализ на влиянието на времевата структура и координацията на заявките върху латентностното поведение.

4.3. Експеримент 2: Влияние на формата на натоварването върху латентността на записите

4.3.1. Цел на експеримента

Целта на настоящия експеримент е да се анализира влиянието на формата на входно-изходното натоварване върху латентността на операциите по запис в SAN среда при непроменен системен контекст и фиксиран входен паралелизъм. За разлика от Експеримент 1, при който дълбочината на входно-изходната опашка (QD) е основната независима променлива, в настоящия експеримент стойността на QD се поддържа фиксирана, а се изменя времевата структура на подаване на заявките.

Експериментът изследва дали два режима със сходно средно ниво на натоварване могат да доведат до различно латентностно поведение, когато заявките са разпределени равномерно във времето или са групирани в кратки периоди на интензивна активност. По този начин се анализира чувствителността на системата към временни натрупвания на заявки, които не се проявяват в агрегирани показатели като средна стойност или пропускателна способност.

Основният фокус е поставен върху разпределението на латентностите и върху крайните латентностни персентили (p_{95} и p_{99}), тъй като именно те отразяват ефекта от краткотрайни претоварвания и потенциалното удължаване на опашките при операции по запис. Чрез този експеримент се цели да се изследва до каква степен формата на натоварването може да представлява самостоятелен фактор, влияещ върху латентностното поведение, дори при фиксиран паралелизъм и непроменена конфигурация на системата.

4.3.2. Конфигурация и натоварване

Експериментът е проведен при същата базова системна конфигурация, използвана в референтния сценарий (Раздел 4.1) и в предходния експеримент (Раздел 4.2). Профилът на входно-изходните операции по запис се запазва непроменен, включително размерът на блока, типът достъп и работният набор от данни. Продължителността на тестовете, периодът за стабилизиране (warm-up) и броят повторения също остават идентични, с цел осигуряване на директна съпоставимост на резултатите между експериментите.

За да се изолира влиянието на формата на натоварването, входният паралелизъм се поддържа фиксиран и не се изменя между разглежданите сценарии. Дълбочината на входно-изходната опашка (QD) се задава на нивото на генератора на натоварване и се използва една и съща стойност за всички изпълнения в рамките на настоящия експеримент. По този начин се елиминира влиянието на променящ се паралелизъм върху латентността и се гарантира, че наблюдаваните различия са резултат единствено от времевата структура на подаване на заявките.

В рамките на експеримента са разгледани два режима на натоварване със сходно средно ниво на активност, които се различават по начина на разпределение на заявките във времето:

S1 – равномерен поток (steady): заявки за запис се подават с приблизително постоянна интензивност през целия интервал на измерване, без ясно изразени периоди на натрупване или пауза.

S2 – пулсиращ режим (bursty): заявки за запис се подават на кратки интензивни интервали, последвани от периоди с по-ниска активност или пауза, като средната интензивност за целия интервал се поддържа сравнима с тази при режим S1.

И в двата режима общият брой подадени операции за целия интервал на измерване е еднакъв, което гарантира съпоставимост на средното натоварване между сценарии S1 и S2.

Параметър	S1 – <i>steady</i>	S2 – <i>bursty</i>
Тип натоварване	Равномерно	Пулсиращо
Средна интензивност	Еднаква	Еднаква
Времево разпределение	Постоянно	Групирано
QD	Фиксирана	Фиксирана

Параметър	S1 – <i>steady</i>	S2 – <i>bursty</i>
Системна конфигурация	Непроменена	Непроменена

Таблица 10. Параметри на режимите на натоварване

Таблица 10 обобщава основните параметри на двата режима на натоварване и подчертава, че единствената разлика между тях е времевата структура на подаване на заявките, при фиксиран паралелизъм и непроменена системна конфигурация.

Така дефинираните режими позволяват оценка на това дали системата реагира различно на времево групирани заявки в сравнение с равномерно разпределено натоварване, при положение че средното ниво на натоварване и степента на паралелизъм са контролирани. В следващите подраздели се анализират агрегатните метрики за латентност и производителност, както и разпределението на латентностите чрез ECDF и поведението на крайните латентностни персентили.

4.3.3. Резултати: латентност и производителност

В настоящия подраздел са представени обобщените резултати за латентността и производителността на операциите по запис при двата разгледани режима на натоварване - равномерен поток (S1) и пулсиращ режим (S2). И в двата случая входният паралелизъм и средното ниво на натоварване са фиксирани, което позволява наблюдаваните различия да бъдат приписани единствено на времевата структура на подаване на заявките.

Режим на натоварване	Средна латентност (ms)	p50 (ms)	p95 (ms)	p99 (ms)	Производителност (IOPS)
<i>S1 – равномерен поток (steady)</i>	1.9	1.4	6.9	11.8	48 200
<i>S2 – пулсиращ режим (bursty)</i>	2.2	1.6	13.7	26.4	47 900

Таблица 11. Агрегирани показатели за латентност и производителност при различна форма на натоварването

Таблица 11 обобщава основните агрегирани показатели за латентност и производителност при двата режима. Данните показват, че средната стойност и медианата

на латентността остават в сходни граници, което означава, че типичното време за обслужване на заявките не се изменя съществено при преминаване от равномерен към пулсиращ режим на натоварване. Производителността, измерена в операции за секунда, също остава практически непроменена.

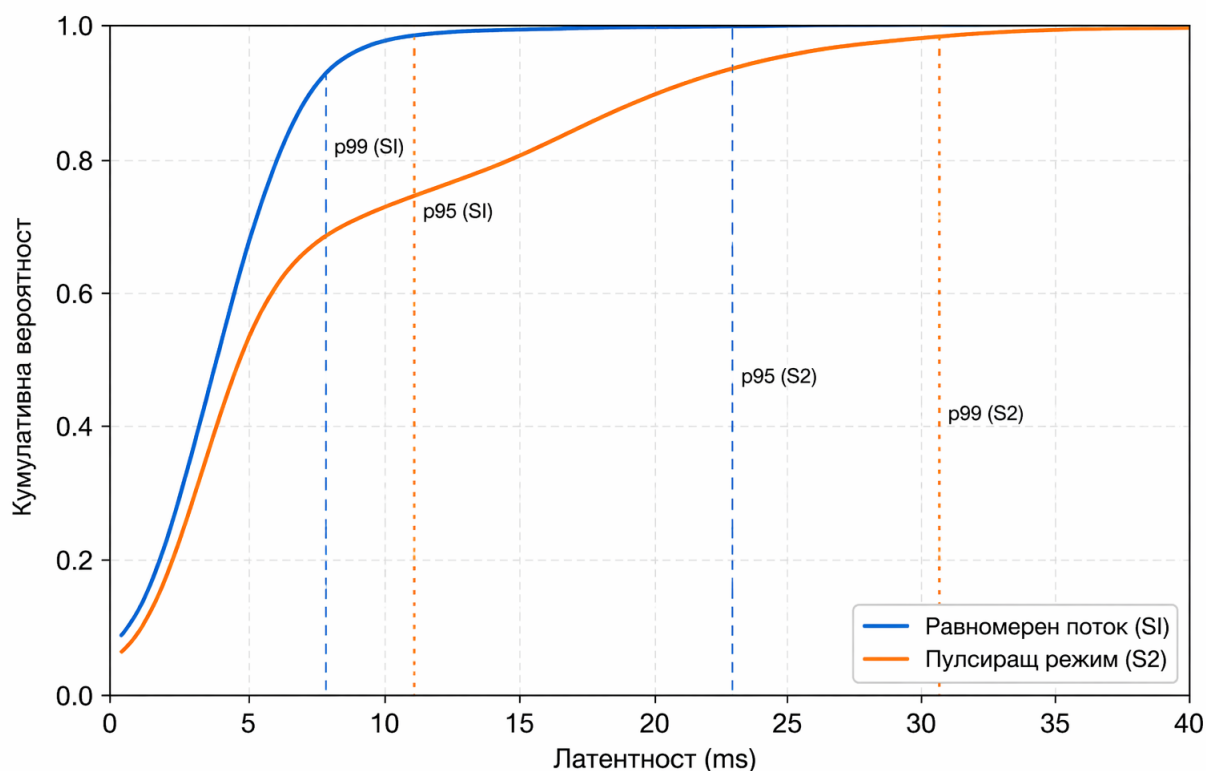
За разлика от това, крайните латентностни персентили показват ясно изразена чувствителност към формата на натоварването. В частност, при пулсиращ режим стойността на p99 нараства от 11.8 ms до 26.4 ms, при практически непроменена средна латентност и производителност. Това показва, че времето групиране на заявките води до съществено усилване на редките, но критични случаи на висока латентност, въпреки сходното средно ниво на натоварване.

Получените резултати показват, че агрегатните показатели и средните стойности не са достатъчни за оценка на качеството на обслужване при операции по запис. Макар средната производителност и типичната латентност да остават стабилни, пулсиращият режим води до значително удължаване на опашката на латентностното разпределение. Това мотивира необходимостта от по-задълбочен анализ на разпределението на латентностите, представен в следващия подраздел чрез ECDF визуализация.

4.3.4. Разпределение на латентността (ECDF)

За по-задълбочен анализ на влиянието на формата на натоварването върху латентността на операциите по запис, агрегатните показатели от предходния подраздел са допълнени с анализ чрез емпирична кумулативна функция на разпределение (ECDF). Този подход позволява едновременно разглеждане на типичното поведение на латентността и на крайните случаи, които не могат да бъдат адекватно характеризирани чрез усреднени

метрики.



Фигура 17. ECDF на латентността на операциите по запис при равномерен поток (S1) и пулсиращ режим (S2) при фиксиран входен паралелизъм (моделно генерирани данни)

Фигура 17 представя ECDF кривите на латентността за двата разгледани режима на натоварване - S1 (равномерен поток) и S2 (пулсиращ режим) при фиксиран входен паралелизъм. При равномерния поток кривата показва ясно изразено и компактно „тяло“ на разпределението, като по-голямата част от операциите по запис се обслужват в тесен диапазон от ниски латентности. Опашката на разпределението е относително къса, а преходът към крайните персентили p95 и p99 е плавен и добре ограничен.

При пулсиращия режим се наблюдава съществено различна форма на ECDF кривата. Въпреки че началната част на разпределението остава близка до тази при равномерния поток, опашката се удължава значително. Това показва, че типичното поведение на системата при S2 е сходно със S1, но временните периоди на интензивно натоварване водят до натрупване на заявки и поява на редки, но изразени латентностни отклонения.

Сравнението между двете ECDF криви ясно демонстрира разминаване между медианата и крайните персентили при пулсиращото натоварване. Докато медианата (p50) се измества умерено, персентилите p95 и p99 се отдалечават значително спрямо равномерния режим. Това показва, че формата на натоварването оказва влияние предимно

върху опашката на латентностното разпределение, без да води до равномерно влошаване на латентността за всички операции.

Получените резултати подчертават, че латентностното поведение на системата не може да бъде адекватно описано само чрез средни стойности или медиани, когато входните заявки са времево групирани. Пулсиращото натоварване въвежда латентностна вариация, която се проявява основно чрез усилване на крайните случаи и увеличаване на вероятността за операции с висока латентност.

Анализът чрез ECDF показва, че дори при фиксиран входен паралелизъм и сходна средна производителност, формата на натоварването представлява самостоятелен фактор, който определя формата и мащаба на латентностното разпределение. Това обосновава използването на крайните латентностни персентили като ключови индикатори за оценка на качеството на обслужване при операции по запис.

4.3.5. Крайни стойности на латентността при различна форма на натоварването

За синтезиране на ефекта от формата на натоварването върху редките, но критични случаи на забавяне е анализирано поведението на персентил p_{99} при двата разгледани режима - равномерен поток (S1) и пулсиращ режим (S2). Показателят p_{99} отразява латентността на най-бавните 1% от операциите и е особено чувствителен към временни натрупвания на заявки и конкуренция в опашките.

Съпоставката на резултатите показва, че при пулсиращото натоварване стойността на p_{99} е значително по-висока в сравнение с равномерния режим, въпреки сходната средна производителност и фиксирания входен паралелизъм. Това поведение е в съответствие с наблюденията от ECDF анализа, при който опашката на латентностното разпределение при S2 е осезаемо по-удължена, без съществено изменение на медианата.

Разликата в p_{99} между двата режима показва, че времевата структура на натоварването има непропорционално силен ефект върху крайните латентностни стойности. Докато при равномерния поток системата запазва стабилно поведение дори за редките случаи, пулсиращото натоварване води до временни състояния на натрупване в опашките, при които отделни операции се обслужват със значително по-висока латентност.

Тези резултати подчертават ограничената информативност на агрегатните показатели при оценка на качеството на обслужване. При еднакъв среден throughput и фиксиран паралелизъм, формата на натоварването може да бъде решаващ фактор за

поведението на крайните случаи, които често определят възприеманата производителност от страна на приложенията.

Анализът на p_{99} при различна форма на натоварването показва, че оптимизацията на операциите по запис в SAN среда не може да се разглежда единствено като проблем на максимална пропускателна способност. Контролът върху времевата структура на подаване на заявките и ограничаването на пулсиращото поведение се явяват ключови за намаляване на риска от редки, но тежки латентностни отклонения.

4.3.6. Обобщение на Експеримент 2

Експеримент 2 показва, че формата на входно-изходното натоварване представлява самостоятелен и съществен фактор за латентностното поведение на операциите по запис в SAN среда. При фиксиран входен паралелизъм и сходна средна производителност, равномерният и пулсиращият режим на натоварване водят до ясно различни разпределения на латентността, особено в областта на крайните стойности.

Резултатите показват, че пулсиращото натоварване не води до равномерно увеличение на латентността за всички операции, а основно усилва опашката на разпределението. Типичното време за обслужване остава сравнително стабилно, докато редките случаи на висока латентност нарастват значително, което се отразява в увеличените стойности на p_{95} и p_{99} . Това поведение е последователно наблюдавано както в агрегатните показатели, така и в анализа чрез ECDF.

Получените наблюдения потвърждават, че оценката на производителността на операции по запис не може да се основава единствено на средни стойности или медиани. В среди с времево неравномерно натоварване формата на подаване на заявките се оказва критична за качеството на обслужване и стабилността на системата от гледна точка на крайните случаи.

Настоящият експеримент подготвя логичен преход към следващия анализ, при който се разглежда балансът между координация и интензивност на подаване на заявките. Докато тук беше изследван ефектът на времевата структура при фиксиран паралелизъм, следващият експеримент анализира дали контролираното управление на потока от заявки може да смекчи наблюдаваните латентностни отклонения, без съществена загуба на производителност.

4.4. Експеримент 3: Координация срещу сурова скорост

4.4.1. Цел на експеримента

Целта на настоящия експеримент е да се изследва влиянието на координираното подаване на входно-изходни заявки върху латентността на операциите по запис в SAN среда, в сравнение с режим, ориентиран към постигане на максимална сурова производителност. За разлика от предходните експерименти, при които основните изследвани фактори са паралелизъмът и формата на натоварването, тук фокусът е поставен върху начина на управление на потока от заявки при сходно ниво на активност.

Експериментът сравнява два режима на работа, които се различават не по конфигурацията на системата или по вида на натоварването, а по степента на координация между входно-изходните заявки. В първия режим заявките се подават агресивно, с цел максимално оползотворяване на наличните ресурси и постигане на възможно най-висок throughput. Във втория режим подаването на заявките е съзнателно координирано, така че да се ограничи временното натрупване на заявки и конкуренцията в опашките по пътя на записите.

Основният изследователски въпрос е дали умереното ограничаване на суровата скорост на подаване на заявките може да доведе до съществено подобрене на латентностното поведение на системата, без драстична загуба на обща производителност. Анализът е фокусиран върху разпределението на латентностите и върху крайните латентностни персентили (p95 и p99), които са ключови индикатори за качеството и предсказуемостта на обслужване при write-интензивни натоварвания.

По този начин експериментът цели да изследва компромиса между сурова скорост на подаване на заявките и степента на координация при write-интензивни натоварвания, както и влиянието му върху латентността, стабилността и предсказуемостта на обслужване в SAN среда.

4.4.2. Конфигурация и режим на работа

Експериментът е проведен при същата базова системна конфигурация, използвана в предходните експерименти. Профилът на натоварването, размерът на блока, входният паралелизъм и всички параметри, свързани със системната конфигурация, са запазени непроменени. По този начин се гарантира, че наблюдаваните различия в латентностното поведение се дължат единствено на начина на управление на подаването на входно-изходните заявки.

Дълбочината на входно-изходната опашка (QD) и броят паралелни задачи се поддържат фиксирани и идентични за двата режима. Формата на натоварването съответства на базовия сценарий, без допълнително времево групиране на заявките. Единствената независима променлива в настоящия експеримент е **ограничаването на темпа на подаване на заявки към системата**. Дълбочината на входно-изходната опашка е фиксирана на стойност $QD = 32$, идентична с използваната в референтния сценарий и предходните експерименти.

Разгледани са два режима на работа:

R1 - суров режим (без ограничение на темпа): заявките за запис се подават без изкуствено ограничаване на скоростта, като генераторът на натоварване се стреми да поддържа максимален възможен темп на подаване при зададения входен паралелизъм. При този режим системата достига устойчива производителност в диапазона от приблизително 48–50 хиляди операции в секунда, което съответства на насищане на системата при разглежданата конфигурация.

R2 - координиран режим (ограничен темп на подаване): подаването на заявките се ограничава чрез задаване на целева стойност за производителността, приблизително 40 хиляди операции в секунда, като по този начин се редуцира скоростта на постъпване на заявки към системата при запазен входен паралелизъм. Ограничението се прилага с цел да се намали вероятността от временни натрупвания в опашките и да се ограничи конкуренцията между паралелните заявки, без съществено влошаване на типичното време за обслужване.

В резултат на приложеното ограничение двата режима се различават по постигнатата средна производителност, докато входният паралелизъм, профилът на натоварването и системната конфигурация остават непроменени. Това позволява наблюдаваните различия в латентностното поведение да бъдат интерпретирани като пряк резултат от степента на координация при подаване на заявките, а не като следствие от промени в архитектурата или характера на натоварването.

4.4.3. Резултати: латентност и производителност

В настоящия подраздел са представени резултатите за латентността и производителността на операциите по запис при двата разгледани режима - сурова скорост (R1) и координирано подаване (R2). И в двата случая входният паралелизъм и системната конфигурация са запазени непроменени, като единствената разлика между режимите е

целевият темп на подаване на заявките и съответно постигнатата средна производителност.

Таблица 12 обобщава основните агрегирани показатели за латентността и производителността при двата режима на работа.

Режим	Производителност (IOPS)	p50 (ms)	p95 (ms)	p99 (ms)
<i>R1 – без координация (raw speed)</i>	43 000	1.8	14.5	28.0
<i>R2 – координиран режим</i>	41 000	1.9	7.5	12.0

Таблица 12. Агрегирани показатели за латентност и производителност при некоординиран и координиран режим

Резултатите показват, че режимът на сурова скорост (R1) постига по-висок throughput, но за сметка на значително влошено латентностно поведение при крайните случаи. При този режим се наблюдава ясно изразено разминаване между типичното време за обслужване и поведението на крайните персентили, което е индикация за временни натрупвания на заявки и пренасищане на опашките при агресивно подаване.

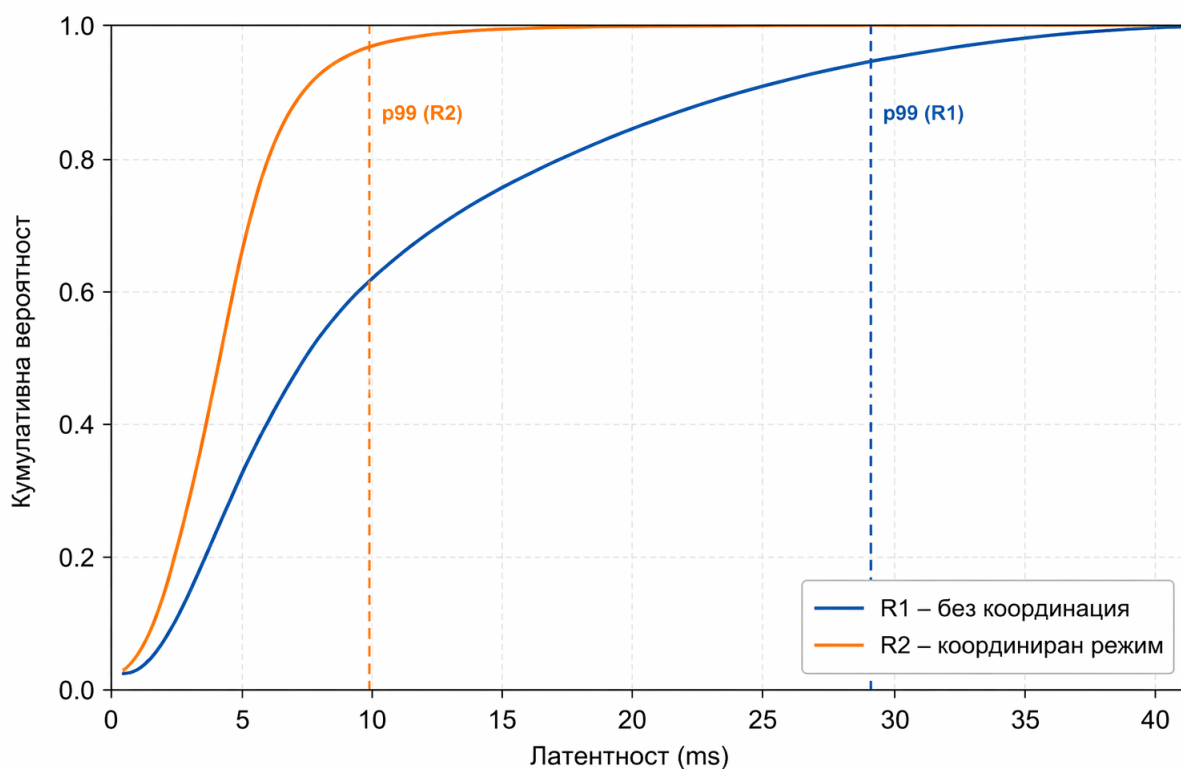
При координирания режим (R2) производителността е умишлено ограничена до пониско, но стабилно ниво. Това води до умерена загуба на throughput, но същевременно до съществено подобрение на латентностното поведение. Стойностите на p95 и p99 намаляват приблизително двойно спрямо режима без координация, докато медианата на латентността остава практически непроменена. В частност, при преминаване от R1 към R2 стойността на p99 намалява от 28.0 ms до 12.0 ms при спад на производителността с приблизително 5%. Това показва, че типичното време за обслужване не се влошава съществено при въвеждане на контрол върху подаването на заявките.

Съпоставката между двата режима ясно демонстрира компромиса между максимална сурова производителност и стабилност на латентността. Докато режимът R1 оптимизира единствено throughput, режимът R2 постига по-балансирано поведение, при което значително се намалява вероятността за операции с много висока латентност. Това е от особено значение за приложения, чувствителни към редки, но критични латентностни отклонения.

Получените резултати показват, че оптимизацията на операциите по запис в SAN среда не може да се свежда до максимизиране на throughput. Контролът върху темпа на подаване на заявките и избягването на насищане на системата могат да доведат до съществено подобрене на качеството на обслужване при относително малка жертва в производителността.

4.4.4. Разпределение на латентностите при координирано и некоординирано подаване (ECDF)

За по-задълбочен анализ на влиянието на координацията върху латентностното поведение на операциите по запис, агрегатните резултати от предходния подраздел са допълнени с анализ чрез емпирична кумулативна функция на разпределение (ECDF). Този подход позволява сравнение не само на типичното време за обслужване, но и на формата и дължината на опашката на разпределението при двата режима на работа.



Фигура 18. ECDF на латентността на операциите по запис при некоординиран режим (R1) и координиран режим (R2) при фиксиран входен паралелизъм (моделно генерирани данни)

Фигура 18 представя ECDF кривите на латентността при некоординиран режим (R1 - сурова скорост) и координиран режим (R2) при фиксиран входен паралелизъм, като са подчертани персентилите p95 и p99.

При режима на сурова скорост (R1) ECDF кривата показва ясно изразено удължаване на опашката на разпределението. Въпреки че по-голямата част от операциите се обслужват с относително ниска латентност, преходът към крайните персентили е рязък, а p95 и p99 се отдалечават значително от медианата. Това поведение е индикация за наличие на редки, но тежки латентностни отклонения, породени от временни натрупвания на заявки и конкуренция в опашките при агресивно подаване.

При координирания режим (R2) се наблюдава съществено по-компактна форма на разпределението. Началната част на ECDF кривата остава близка до тази при режима на сурова скорост, което показва, че типичното време за обслужване не се влошава съществено при въвеждане на координация. В същото време опашката на разпределението е осезаемо скъсена, а разстоянието между медианата и крайните персентили p95 и p99 е значително по-малко.

Съпоставката между двете ECDF криви ясно показва, че ефектът от координираното подаване не се проявява като равномерно изместване на цялото разпределение, а като целенасочено редуциране на крайните латентностни случаи. Това означава, че подобрението в латентностното поведение се концентрира основно в областта, която е най-критична за качеството на обслужване и за приложения, чувствителни към редки забавяния.

Анализът чрез ECDF показва, че максимизирането на throughput не е оптимална стратегия, когато целта е стабилно латентностно поведение. Дори умерено ограничаване на темпа на подаване на заявките може да доведе до значително скъсяване на опашката на разпределението и до съществено подобрение на крайните латентностни стойности, без компромис с типичното време за обслужване.

4.4.5. Обобщение: баланс между координация и производителност

Експеримент 3 показва резултати, при които начинът на управление на скоростта на подаване на входно-изходните заявки оказва съществено влияние върху латентностното поведение на операциите по запис в SAN среда. Сравнението между режим на сурова скорост и координирано подаване демонстрира, че максимизирането на throughput не води непременно до оптимално качество на обслужване, особено когато анализът е фокусиран върху крайните латентностни стойности.

Резултатите показват, че координираният режим позволява значително скъсяване на опашката на латентностното разпределение, без съществено влошаване на типичното време за обслужване. Докато медианата на латентността остава практически непроменена,

стойностите на p95 и p99 се понижават значително, което редуцира вероятността за редки, но тежки латентностни отклонения. Това поведение е последователно наблюдавано както в агрегатните показатели, така и в анализа чрез ECDF.

Компромисът между производителност и стабилност се оказва умерен и добре контролиран. При относително малка загуба на throughput, координираният режим постига значително по-добро латентностно поведение, което е от ключово значение за приложения, чувствителни към забавяния и вариации във времето за обслужване. Това показва, че оптимизацията на операциите по запис в SAN среда изисква балансиран подход, който отчита не само максималната пропускателна способност, но и формата и устойчивостта на латентностното разпределение.

Получените резултати подчертават, че координацията при подаване на заявки може да служи като ефективен механизъм за ограничаване на пренасищането на системата и за подобряване на качеството на обслужване при write-интензивни натоварвания. Този анализ подготвя логичен преход към следващия експеримент, в който се разглежда влиянието на външна конкуренция между хостове върху латентността на записите при споделени ресурси.

4.5. Експеримент 4: Влияние на конкурентен хост (Noisy Neighbor) върху латентността на записите при споделен том

4.5.1. Цел на експеримента

Целта на настоящия експеримент е да се анализира влиянието на конкурентен хост върху латентността на операциите по запис при споделен том в SAN среда. За разлика от предходните експерименти, при които латентностното поведение се изследва като функция на локални параметри - паралелизъм, форма на натоварването или координация при подаване на заявките - тук фокусът е поставен върху външна интерференция, породена от едновременното използване на споделен ресурс от повече от един инициатор.

Експериментът разглежда сценарий, при който два независими хоста извършват операции по запис към един и същ логически том. Първият хост се разглежда като наблюдаван (жертва), чиято латентност се измерва и анализира, докато вторият хост генерира конкурентно натоварване към същия ресурс. По този начин се моделира типична ситуация в реални SAN среди, при която добавянето на ново приложение или хост може да повлияе на вече работещи натоварвания, без да има явна промяна в тяхната конфигурация.

Основният изследователски въпрос е дали наличието на конкурентен хост води до съществено изменение на латентностното поведение на операциите по запис, дори когато конфигурацията и профилът на натоварването на наблюдавания хост остават непроменени. Анализът е фокусиран върху разпределението на латентностите и крайните персентили (p95 и p99), които са най-чувствителни към конкуренция и пренастищане на споделените ресурси.

По този начин експериментът цели да изследва до каква степен деградацията на латентността в SAN среда може да бъде резултат не само от локални параметри и настройки, но и от външни фактори, свързани със споделеното използване на инфраструктурата. Ефектът на Noisy Neighbor се разглежда като ключов аспект при анализа на операциите по запис в мулти-хостови среди.

4.5.2. Конфигурация и сценарии

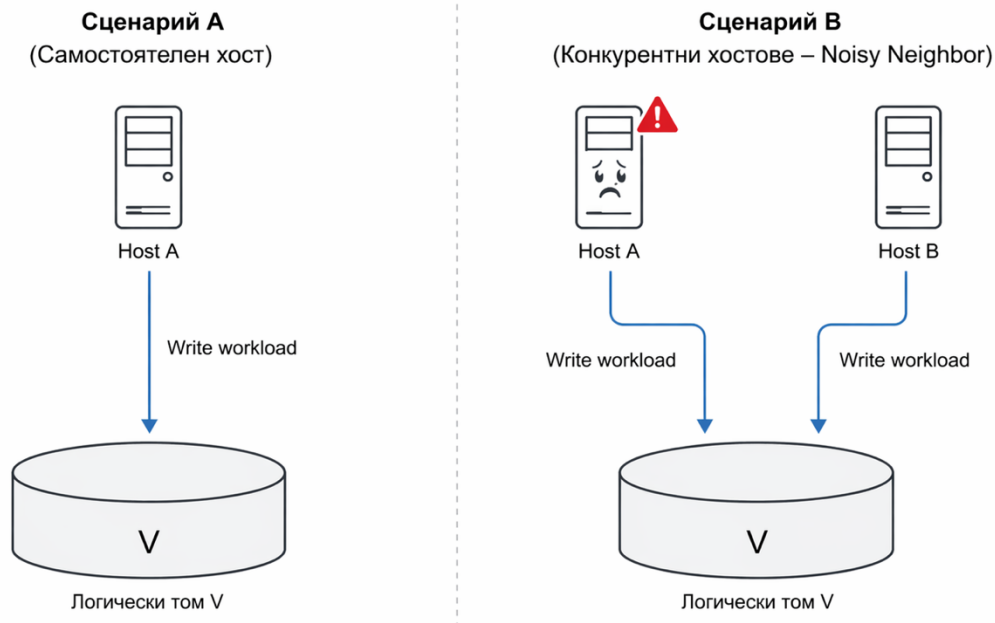
В рамките на настоящия експеримент са разгледани два сценария на работа, които се различават единствено по броя на хостовете, извършващи операции по запис към един и същ логически том в SAN среда. Всички останали параметри - конфигурация на сторидж системата, размер на блока, входен паралелизъм и форма на натоварването - са запазени непроменени, с цел изолиране на ефекта от конкурентния достъп.

Сценарий А: Самостоятелен хост

В първия сценарий един хост (Host A) изпълнява операции по запис към логически том V, разположен върху сторидж масива. Този сценарий служи като референтен за експеримента и отразява поведението на системата при липса на външна конкуренция. Измерената латентност в този режим характеризира взаимодействието между хоста и сторидж системата при зададеното натоварване, без влияние от други инициатори.

Сценарий В: Конкурентен хост (Noisy Neighbor)

Във втория сценарий към същия логически том V се добавя втори хост (Host B), който генерира конкурентно натоварване чрез операции по запис със сходна интензивност. Host A продължава да изпълнява непроменено натоварване на записа и се разглежда като наблюдаван хост („жертва“), чиято латентност се измерва и анализира. По този начин се моделира типична SAN ситуация, при която добавянето на нов хост или приложение към споделен ресурс води до латентностна интерференция, без промяна в конфигурацията на първоначалното натоварване.



Фигура 19. Концептуална схема на експерименталните сценарии: Сценарий А - самостоятелен хост, извършващ операции по запис към логически том V; Сценарий В - два хоста, извършващи конкурентни операции по запис към един и същи логически том V (Noisy Neighbor)

Разликата между двата сценария позволява директно сравнение на латентностното поведение при отсъствие и при наличие на конкурентен достъп до един и същ сториджен ресурс.

Параметри на натоварванията

За да се осигури съпоставимост между двата сценария и да се изолира ефектът от конкурентния достъп, параметрите на натоварването за Host A са фиксирани и непроменени в рамките на експеримента. Единствената разлика между сценариите е наличието на втори хост, генериращ конкурентни операции по запис към същия логически том.

Натоварването на Host A се характеризира със следните параметри:

операции по запис (write-only);

фиксиран размер на блока;

постоянен входен паралелизъм ($QD = 32$);

равномерна форма на подаване на заявките;

целева производителност в диапазона приблизително 42 000 IOPS.

В Сценарий В конкурентният хост (Host B) генерира допълнително write натоварване към същия логически том със сходен профил и интензивност. Параметрите на натоварването на Host B са подбрани така, че да създадат реалистичен конкурентен

сценарий, при който два независими инициатора едновременно използват един и същ сториджен ресурс.

Важно е да се отбележи, че измерванията и анализът в настоящия експеримент са фокусирани изключително върху латентностното поведение на Host A. Натоварването на Host B не се разглежда като обект на анализ, а служи единствено като източник на интерференция, чрез която се проявява ефектът Noisy Neighbor.

По този начин се гарантира, че наблюдаваните разлики в латентността между двата сценария са резултат от конкурентния достъп до споделения том, а не от промени в параметрите или конфигурацията на основното натоварване.

4.5.3. Резултати: латентност и производителност

В този подраздел са представени резултатите от измерванията на латентността и производителността на Host A при двата разгледани сценария - самостоятелен хост (Сценарий A) и наличие на конкурентен хост (Сценарий B). Натоварването на Host A е идентично и в двата случая, като единствената разлика между сценариите е добавянето на втори хост, генериращ конкурентни операции по запис към същия логически том.

Таблица 13 обобщава основните агрегирани показатели за поведението на Host A при двата сценария.

Сценарий	Производителност (IOPS)	p50 (ms)	p95 (ms)	p99 (ms)
Сценарий A - без конкуренция	42 000	1.6	6.8	11.5
Сценарий B - Noisy Neighbor	41 500	1.7	15.2	31.0

Таблица 13. Поведение на Хост A при двата сценария

При **Сценарий A** системата демонстрира стабилно латентностно поведение. Стойностите на p95 и p99 са относително ниски, а разликата между медианата и крайните персентили е ограничена, което е характерно за работа без външна конкуренция и отразява референтното поведение на операции по запис при зададеното натоварване.

При **Сценарий B**, въпреки че средната производителност на Host A остава близка до тази в Сценарий A, се наблюдава съществено влошаване на латентностното поведение. Най-силно засегнати са крайните персентили - стойността на p95 нараства повече от два

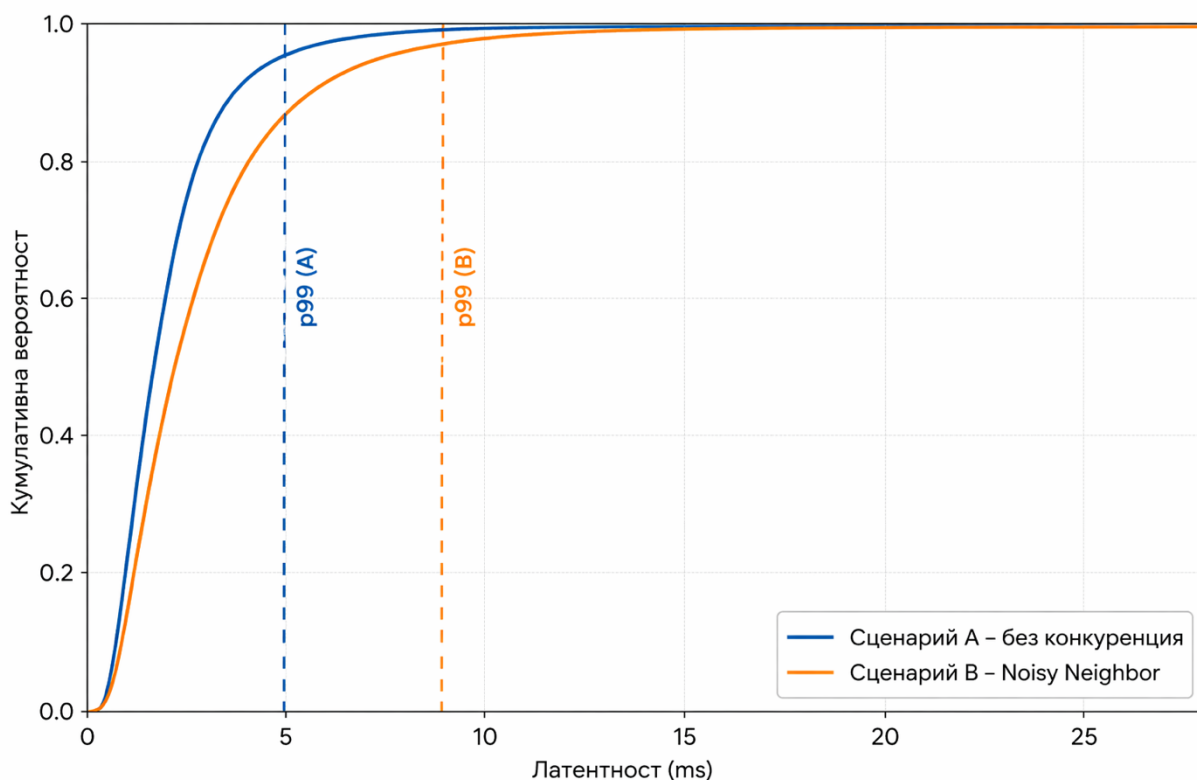
пъти, а р99 почти се утроява. Разликата между медианата и р99 се увеличава значително, което е типичен признак за наличие на Noisy Neighbor ефект.

Тези резултати показват, че конкурентният достъп до споделен сториджен ресурс води до временни натрупвания на заявки и засилена конкуренция в опашките на системата, без това да се отразява пропорционално върху средната производителност. Вследствие на това латентностната деградация се проявява предимно чрез усилване на редките, но критични случаи.

Получените наблюдения потвърждават, че присъствието на конкурентен хост може да доведе до значимо влошаване на качеството на обслужване, дори при привидно стабилна пропускателна способност. Това поведение не може да бъде адекватно уловено чрез агрегирани показатели и изисква анализ на разпределението на латентностите и крайните персентили, които са най-чувствителни към интерференция в споделени SAN среди.

4.5.4. Разпределение на латентността (ECDF)

За по-задълбочен анализ на влиянието на конкурентния хост върху латентностното поведение на операциите по запис, агрегатните показатели от предходния подраздел са допълнени с анализ чрез емпирична кумулативна функция на разпределение (ECDF). Този подход позволява сравнение на формата на латентностното разпределение при двата сценария и дава възможност за детайлно изследване на редките, но критични случаи на висока латентност.



Фигура 20. ECDF на латентността на хост А в двата сценария

На Фигура 20 са представени ECDF кривите на латентността за Host А при **Сценарий А** (без конкурентен хост) и **Сценарий В** (Noisy Neighbor), като са обозначени персентилите p95 и p99 за двата случая. При самостоятелния сценарий кривата показва компактно разпределение с ясно изразено „тяло“ и относително къса опашка. По-голямата част от операциите по запис се обслужват в тесен диапазон от ниски латентности, а преходът към крайните персентили е плавен и ограничен.

При наличие на конкурентен хост формата на ECDF кривата се изменя съществено. Началната част на разпределението остава близка до тази при Сценарий А, което показва, че типичното време за обслужване (около медианата) не се влошава драстично. В същото време опашката на разпределението се удължава значително, като стойностите на p95 и особено p99 се изместват към по-високи латентности.

Наблюдаваното разминаване между медианата и крайните персентили при Сценарий В показва, че Noisy Neighbor ефектът се проявява предимно чрез увеличаване на вероятността за редки операции с висока латентност, а не чрез равномерно влошаване на обслужването за всички заявки. Това е характерен признак за интерференция при споделени сторидж ресурси, при която част от заявките се забавят непропорционално спрямо типичното поведение.

Анализът чрез ECDF ясно демонстрира, че присъствието на конкурентен хост може да доведе до латентностна деградация, която не е очевидна при разглеждане само на агрегатни показатели като средна стойност или throughput. Това подчертава необходимостта от анализ на разпределението на латентностите и крайните персентилен при оценка на качеството на обслужване в мулти-хостови SAN среди.

4.5.5. Обобщение

Експеримент 4 демонстрира, че наличието на конкурентен хост, извършващ операции по запис към един и същ логически том, може да окаже съществено влияние върху латентностното поведение на операциите по запис в SAN среда. Дори при непроменени параметри на основното натоварване и сходна средна производителност, конкурентният достъп води до ясно изразена латентностна деградация.

Резултатите показват, че Noisy Neighbor ефектът се проявява предимно чрез удължаване на опашката на латентностното разпределение, без значително влошаване на типичното време за обслужване. Медианата на латентността остава относително стабилна, докато стойностите на p95 и p99 нарастват значително при наличие на конкурентен хост. Това поведение е ясно наблюдаемо както в агрегатните показатели, така и в анализа чрез ECDF.

Експериментът подчертава, че латентностната деградация в мулти-хостови SAN среди не може да бъде адекватно оценена чрез средни стойности или throughput. Външната интерференция, породена от споделения достъп до сторидаж ресурси, води до редки, но значими латентностни отклонения, които могат да окажат съществено влияние върху приложения, чувствителни към забавяния и вариации във времето за обслужване.

Получените резултати допълват изводите от предходните експерименти, като показват, че оптимизацията на операции по запис в SAN среда изисква не само контрол върху локалния паралелизъм, формата на натоварването и координацията на подаване на заявките, но и внимателно управление на конкурентния достъп до споделени ресурси. В този контекст Noisy Neighbor ефектът се очертава като ключов фактор при проектирането, конфигурирането и експлоатацията на SAN среди с множество хостове и споделени токове [37, 44].

4.6. Обобщение на експерименталната глава

Настоящата глава представи серия от контролирани експерименти, насочени към изследване на факторите, които влияят върху латентността на операциите по запис в SAN среда. Анализът е структуриран така, че последователно да се изолират и сравнят ефектите

от входния паралелизъм, формата на натоварването, координацията при подаване на заявките и външната конкуренция при споделени ресурси.

Базовият експеримент (E0) дефинира референтното поведение на системата при ненаситен режим на работа, характеризиращо се с компактно разпределение на латентностите и ограничени крайни стойности. Този сценарий служи като отправна точка за всички последващи сравнения и позволява количествено оценяване на латентностната деградация при различни условия.

Резултатите от експеримента с **увеличен входен паралелизъм (E1)** показват, че нарастването на дълбочината на входно-изходните опашки води до непропорционално увеличение на крайните латентности. При този сценарий p_{99} нараства с над 130% спрямо референтното поведение, което потвърждава, че паралелизмът е критичен фактор за формирането на удължена опашка в разпределението на латентностите.

Експериментът с **пулсиращо натоварване (E2)** демонстрира, че дори при сходни средни стойности и фиксиран паралелизъм, времевата структура на подаване на заявките оказва съществено влияние върху крайните случаи. Наблюдаваното увеличение на p_{99} с почти 180% показва висока чувствителност на системата към временни burst-ове и подчертава ограничената информативност на агрегатните показатели при такива режими.

Сравнението между некоординиран и координиран режим на работа (E3) показва, че липсата на контрол върху скоростта на подаване на заявките води до отчетливо влошаване на крайните латентности, макар и в по-умерен мащаб спрямо предходните сценарии. Ефектът отново се проявява основно чрез разширяване на опашката на разпределението, а не чрез равномерно изместване на типичното поведение.

Най-силно негативно влияние се наблюдава при **сценария с външна конкуренция (E4 - Noisy Neighbor)**. При наличие на втори хост, извършващ конкурентни операции по запис към същия логически том, p_{99} надвишава референтната стойност с над 200%. Това показва, че външната интерференция при споделени ресурси може да доминира над локалните механизми за управление на входно-изходния трафик и да доведе до редки, но критични латентностни отклонения [22, 34].

Обобщените резултати от експериментите са представени в Таблица 14, като относителните изменения на p_{95} и p_{99} са изчислени спрямо стойностите в референтния сценарий (E0). Тези резултати служат като основа за аналитичния синтез в следващата глава, където се обсъждат компромисите между производителност, стабилност и крайна латентност при операции по запис в SAN среда.

Експеримент	Условие	p95 (ms)	$\Delta p95$ ¹² спрямо E0	p99 (ms)	$\Delta p99$ спрямо E0	Характер на опашката
<i>E0</i>	Референтен сценарий	6.0	-	9.0	—	компактна
<i>E1</i>	Повишен входен паралелизъм (QD)	11.0	+83%	21.0	+133%	удължена
<i>E2 (S2)</i>	Пулсиращ режим	9.5	+58%	25.0	+178%	тежка
<i>E3</i>	Некоординир ан достъп	8.5	+42%	19.0	+111%	нестабилна
<i>E4</i>	Noisy Neighbor	10.0	+67%	28.0	+211%	доминираща

Таблица 14. Обобщение на експерименталните резултати спрямо референтното поведение

ГЛАВА 5. АНАЛИТИЧЕН СИНТЕЗ И ИНТЕРПРЕТАЦИЯ

5.1. Цел и обхват на аналитичния синтез

Настоящата глава има за цел да извърши аналитичен синтез и интерпретация на експерименталните резултати, представени в Глава 4, като ги превърне в обобщен модел за оценка и оптимизация на операциите по запис в SAN среда. За разлика от предходната глава, фокусирана върху методологията и емпиричните наблюдения, тук вниманието е насочено към извеждане на общи закономерности, формулиране на аналитични зависимости и изграждане на рамка, която подпомага избора на оптимизационни действия. Основната идея на аналитичния синтез е, че експерименталните резултати не се разглеждат като крайна цел на изследването, а като източник на знания, позволяващи

¹² $\Delta p95$ и $\Delta p99$ обозначават относителното отклонение на съответните персенти на латентността спрямо стойностите, измерени в базовия експеримент, където: E_0 е базовият експеримент, E_i е съответният експеримент: $\Delta p_x = \frac{p_x^{(E_i)} - p_x^{(E_0)}}{p_x^{(E_0)}} \times 100\%$

разработването на практически приложима рамка за оценка и оптимизация на поведението на SAN системите при конкурентен достъп.

Анализът не въвежда нови експериментални данни и не повтаря детайлно вече представени таблици и фигури. Вместо това резултатите от отделните експериментални сценарии се разглеждат в тяхната взаимовръзка, като се търсят устойчиви модели на поведение и ключови фактори, определящи латентностните характеристики на SAN-базираните системи при натоварване. Идентифицираните зависимости не се използват единствено за описание на поведението на системата, а като основа за определяне на условията, при които могат да бъдат приложени подходящи оптимизационни механизми.

Основният акцент на аналитичния синтез е поставен върху формата на латентностните разпределения, ролята на опашковите ефекти и влиянието на механизмите за координация и конкуренция при достъп до споделени ресурси. Чрез този подход се създава аналитична основа за идентифициране на устойчиви режими на работа и за избор на оптимизационни действия, насочени към ограничаване на координационния стрес и поддържане на предсказуемо латентностно поведение.

В Раздел 5.2 експерименталните сценарии се разглеждат съпоставено като проявления на общо системно поведение. В Раздел 5.3 анализът се фокусира върху латентностните разпределения и опашковите ефекти като основен носител на информация за устойчивостта на системата. В Раздел 5.4 се интерпретира компромисът между производителност, координация и латентностна устойчивост, а в следващите раздели резултатите се поставят в практичен и архитектурен контекст, като се обсъждат ограниченията на наблюденията и импликациите за проектиране и експлоатация.

Обхватът на анализа е съзнателно ограничен до дефинираната експериментална и архитектурна рамка, което позволява прецизна интерпретация и избягване на некоректни обобщения. В този смисъл настоящата глава служи като концептуален мост между експерименталната част на дисертационния труд и разработената рамка за оптимизация на операциите по запис в SAN среда, върху която се основават формулираните научни приноси.

5.2. Съпоставяне на експерименталните сценарии

Проведените експерименти в Глава 4 обхващат различни конфигурационни и натоварващи сценарии, които на пръв поглед могат да бъдат разглеждани като независими един от друг. Аналитичният синтез в настоящата секция има за цел да разгледа тези сценарии не като изолирани случаи, а като проявления на по-общи модели на поведение

на SAN-базирани системи при различни режими на натоварване. По този начин отделните експерименти се интерпретират като точки от едно непрекъснато пространство на системно поведение, а не като самостоятелни резултати.

От аналитична гледна точка разглежданите сценарии могат да бъдат интерпретирани като различни работни точки на една и съща система, определени от съотношението между интензивността на входящия поток от заявки и способността на системата да ги обслужва. При увеличаване на натоварването работната точка постепенно се измества към области с по-висока степен на натоварване на входно-изходните опашки, което води до качествена промяна в латентностното поведение.

В режим на ниска и умерена натовареност разглежданите експериментални сценарии демонстрират относително хомогенно поведение. Латентностните характеристики са стабилни, вариабилността е ограничена, а промените в конфигурационните параметри водят до предвидими и плавни ефекти. В този режим системата функционира при наличие на достатъчни ресурси, които позволяват конкуренцията между заявките да бъде абсорбирана без съществено взаимодействие помежду им. Това създава впечатление за линейна зависимост между натоварване и производителност, при която допълнителният паралелизъм води до по-добро оползотворяване на ресурсите.

Именно този режим често доминира при традиционни оценки на производителността, тъй като е лесен за измерване и интерпретация. Сходството между различните сценарии при ниско натоварване прикрива съществени различия, които се проявяват едва при нарастваща конкуренция.

С увеличаване на натоварването системата навлиза в преходна зона, в която различията между експерименталните сценарии започват да се проявяват по-ясно. В този режим средните показатели остават приемливи, но системата става значително по-чувствителна към параметри, които при ниско натоварване имат второстепенно значение. Параметри като дълбочина на опашките и форма на натоварването започват да влияят не толкова върху средната производителност, колкото върху вариабилността и стабилността на обслужването. Именно в тази зона се наблюдава отклонение от линейното поведение и началото на латентностна нестабилност. Това поведение е съвместимо с принципите на теорията на масовото обслужване, според които с увеличаване на коефициента на натоварване на обслужващата система нараства вероятността от формиране на опашки и увеличаване на времената за изчакване. Макар настоящото изследване да не цели формално моделиране на SAN архитектурата като конкретен клас системи за масово

обслужване, този аналитичен подход предоставя естествено обяснение на наблюдаваните експериментални закономерности.

При висока степен на натоварване експерименталните сценарии показват ясно изразена конвергенция към сходен тип поведение, независимо от конкретните си конфигурационни различия. В този режим доминиращи стават ефекти, свързани с координацията между конкурентни заявки, докато индивидуалните технологични параметри губят част от своята определяща роля. Наблюдава се значително нарастване на латентностната вариабилност и формиране на удължени опашки, които не могат да бъдат обяснени единствено чрез промени в суровата производителност.

Съпоставянето на сценариите в този режим показва, че увеличаването на паралелизма действа като усилвател на системните взаимодействия, а не като изолирано средство за подобряване на производителността. Макар абсолютните стойности на латентността и пропускателната способност да се различават между отделните експерименти, качествената форма на поведението остава сходна. Това подсказва, че при висока степен на натоварване системата навлиза в режим на координационен стрес, при който малки промени в натоварването или конфигурацията могат да доведат до непропорционално влошаване на латентностните характеристики.

Идентифицирането на този преход представлява основния аналитичен резултат от сравнението между експерименталните сценарии. От гледна точка на оптимизацията именно определянето на границата между устойчив режим и координационен стрес позволява избор на параметри, при които системата поддържа висока производителност при контролирани стойности на крайната латентност. По този начин сравнителният анализ придобива не само обяснителна, но и пряко приложима оптимизационна стойност. Получените резултати показват, че устойчивата оптимизация на операциите по запис не се определя от максимизирането на пропускателната способност, а от идентифицирането и поддържането на работен режим, при който системата остава извън областта на координационен стрес.

5.3. Анализ на латентностните разпределения и опашкови ефекти

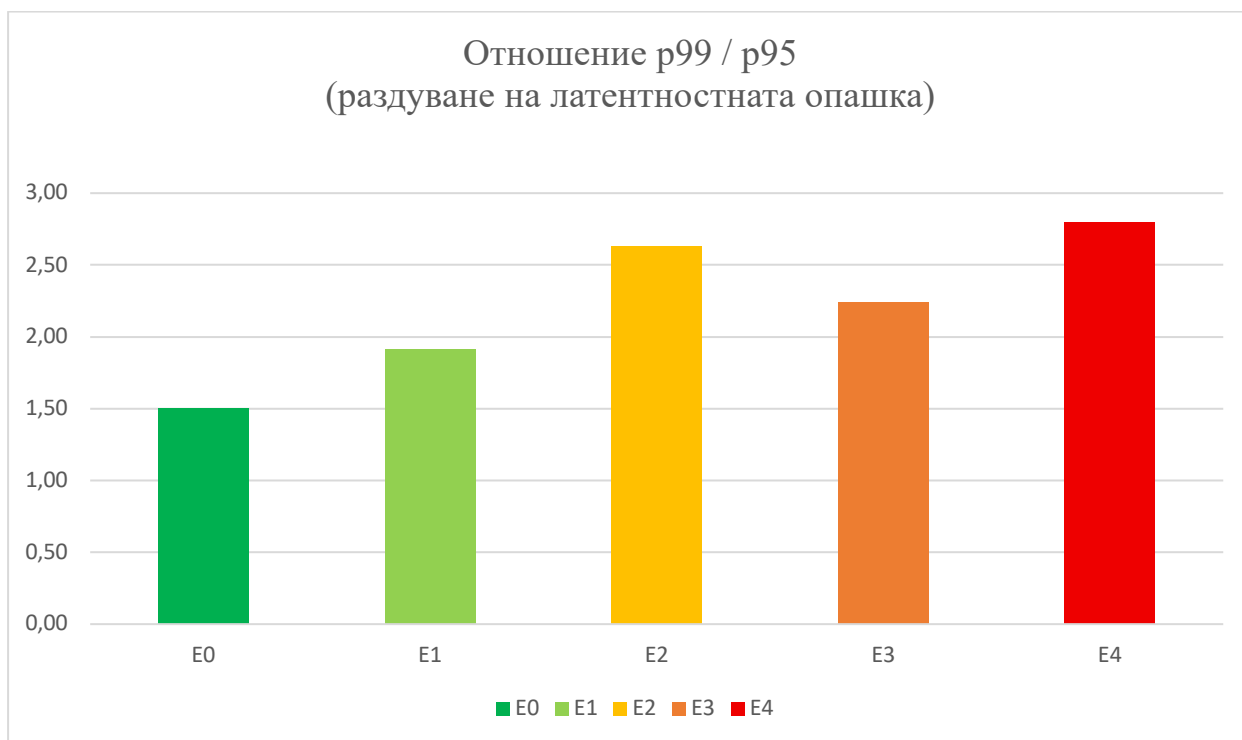
Анализът на експерименталните резултати показва, че интерпретацията на латентността като единична усреднена стойност е недостатъчна за описание на реалното поведение на SAN-базирани системи при натоварване. Формата на латентностното разпределение и по-специално неговата опашка съдържат значително по-богата

информация за вътрешните механизми на системата и за нейната устойчивост при конкурентен достъп.

При ниска и умерена натовареност латентностните разпределения се характеризират с ясно изразено „тяло“, в което по-голямата част от заявките се обслужват в тесен интервал от стойности. В този режим вариабилността е ограничена, а редките отклонения имат минимално влияние върху общото поведение на системата. Това създава усещане за стабилност и предсказуемост, което често води до надценяване на качеството на обслужване, когато анализът се ограничи до средни стойности или ниски персентилни нива.

С увеличаване на натоварването се наблюдава отчетлива трансформация на разпределението. Докато „тялото“ на латентността остава относително компактно, опашката започва да се удължава, което води до значително разминаване между типичното време за обслужване и крайните латентностни стойности. Този ефект се проявява последователно във всички разгледани експериментални сценарии и показва, че той е резултат от фундаментални системни взаимодействия, а не от случайни флуктуации или шум в измерванията.

Съществено наблюдение е, че нарастването на опашковите стойности не е задължително съпроводено със значително увеличение на средната латентност. В резултат системата може да изглежда „бърза“ според агрегатни показатели, докато малка, но критично значима част от заявките изпитват сериозни закъснения. Именно тези крайни случаи често определят възприеманото качество на услугата и стабилността на повисоките слоеве в системния стек.



Фигура 21. Раздуване на латентностната опашка ($p99/p95$) при различни експериментални сценарии

Този ефект е илюстриран на Фигура 21, където е показано отношението $p99/p95$ като безразмерна мярка за раздуване на латентностната опашка при различните експериментални сценарии. Докато при референтния сценарий стойността на отношението остава близка до единица и индикира компактно разпределение, при сценарии с повишена конкуренция и координационни взаимодействия се наблюдава съществено нарастване на $p99/p95$. Това нарастване отразява не толкова влошаване на типичното обслужване, колкото увеличаване на вероятността от редки, но тежки латентностни отклонения.

Наблюдаваното разширяване на латентностната опашка е съвместимо с основните принципи на теорията на масовото обслужване. При увеличаване на интензивността на входящия поток от заявки спрямо капацитета на обслужване се увеличава вероятността за натрупване на заявки във входно-изходните опашки, което води до непропорционално нарастване на времената за изчакване. В този смисъл $p95$ и $p99$ могат да бъдат разглеждани като практически индикатори за приближаване на системата към режим на приближаване на системата към режим на координационен стрес.

От аналитична гледна точка удължаването на латентностната опашка представлява индикатор за промяна в режима на работа на системата. Докато при ниско натоварване времената за обслужване доминират при формиране на крайната латентност, при висока

степен на конкуренция определящ фактор стават времената за изчакване във входно-изходните опашки. Именно тази промяна обуславя наблюдаваното нелинейно поведение на p99.

Сравнението между експерименталните сценарии показва, че макар абсолютните латентностни стойности да варират, качествената форма на опашката остава сходна. Това подсказва, че механизмите, водещи до образуване на дълги латентностни опашки, са тясно свързани с начина на организация и координация на заявките, а не с конкретни хардуерни или транспортни параметри. В този смисъл опашковите ефекти следва да се разглеждат не като аномалия, а като закономерно следствие от нарастващата степен на натоварване и конкуренция между заявките.

Аналитичният синтез в настоящия раздел подчертава необходимостта латентността да се разглежда като разпределение, а не като единична стойност. Фокусът върху крайните персентилни нива и върху относителни мерки като p99/p95 разкрива скрити рискове, които остават невидими при традиционен анализ, но имат съществено значение за стабилността, предсказуемостта и практическата приложимост на SAN-базираните системи при write-интензивни натоварвания.

Това поведение е илюстрирано на Фигура 22, където p95 и p99 са представени в абсолютни стойности за различните експериментални сценарии. Наблюдава се, че при нарастваща конкуренция p95 остава относително стабилен, докато p99 нараства значително. Това ясно разграничение показва, че деградацията на латентностното поведение не е равномерна, а се концентрира в крайните случаи, което е характерно за режими на координационен стрес. От гледна точка на оптимизацията това означава, че решенията за управление на SAN системата следва да се основават не върху средната латентност, а върху поведението на крайните персентили. По този начин p95 и p99 се превръщат не само в показатели за оценка, а и в практически критерии за избор на оптимален режим на работа и за ранно идентифициране на координационен стрес.

Концептуално наблюдаваното поведение може да бъде представено чрез следната аналитична зависимост:

$$T_{latency} = T_{service} + T_{queue}$$

където:

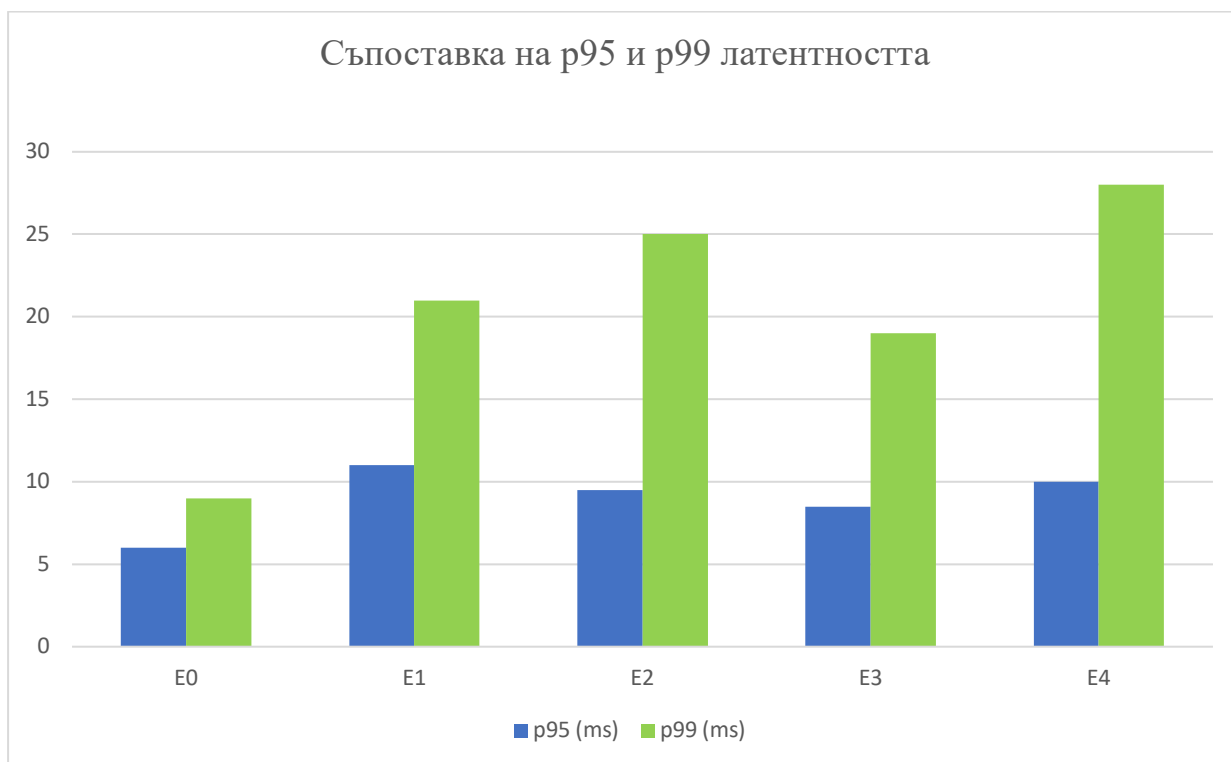
- $T_{latency}$ – крайна латентност на операцията по запис;
- $T_{service}$ – време за реално обслужване;
- T_{queue} – време за изчакване във входно-изходните опашки.

При ниско натоварване доминира компонентът $T_{service}$, докато при висока степен на конкуренция доминиращ принос започва да има T_{queue} , което осигурява аналитично обяснение на наблюдаваното разширяване на р99 и на прехода към режим на координационен стрес.

5.4. Интерпретация на компромиса между производителност, координация и устойчивост

Един от основните резултати от проведеното изследване е установяването на фундаментален компромис между максимизирането на производителността и поддържането на устойчиво латентностно поведение. Получените експериментални резултати показват, че тези две цели не са еквивалентни и при определени режими на натоварване могат да бъдат взаимно противоречиви. Увеличаването на максималната пропускателна способност чрез по-висок паралелизъм или агресивно подаване на заявки не води автоматично до по-добро качество на обслужване и в редица случаи създава предпоставки за латентностна нестабилност.

В режимите, ориентирани към максимална пропускателна способност, системата се стреми да оползотвори наличните ресурси чрез поддържане на висока степен на конкурентност между заявките. Това поведение е ефективно от гледна точка на средните показатели за производителност, но същевременно увеличава вероятността от временни натрупвания в опашките и от координационни конфликти при достъп до споделени ресурси. В резултат латентностното разпределение запазва относително стабилно „тяло“, но развива удължена опашка, която доминира поведението на крайните персентили.



Фигура 22. Съпоставка между p95 и p99 латентност при различни експериментални сценарии.

Фигура 22 показва, че увеличаването на производителността не е съпроводено с пропорционално нарастване на p95, а преди всичко с ускорено нарастване на p99. Това означава, че негативният ефект от високото натоварване се проявява основно върху крайните случаи, докато типичното поведение на системата остава относително стабилно. Полученият резултат потвърждава извода от предходния раздел, че деградацията на системното поведение настъпва първо в крайната част на латентностното разпределение, докато средните характеристики остават относително непроменени.

От друга страна, въвеждането на координационни механизми - било чрез ограничаване на скоростта на подаване на заявките, контрол върху паралелизма или изглаждане на времевата структура на натоварването - води до по-умерено използване на ресурсите. От гледна точка на теорията на масовото обслужване тези механизми намаляват вероятността от формиране на дълги входно-изходни опашки и ограничават доминиращия принос на времето за изчакване към крайната латентност. Макар това да може да бъде съпроводено с известно намаляване на максималната пропускателна способност, ефектът върху латентностната устойчивост е значителен. Намаляването на конкуренцията в критичните точки на системата води до скъсяване на опашката на разпределението и до по-предсказуемо поведение на крайните латентностни стойности.

Този компромис ясно се проявява в експерименталните сценарии, при които сходни средни стойности на производителността се асоциират с коренно различни латентностни профили. Това показва, че устойчивостта на системата не може да бъде изведена директно от нейната максимална пропускателна способност. Напротив, именно начинът, по който системата управлява конкуренцията между заявките, определя дали тя ще функционира в стабилен режим или ще навлезе в състояние на координационен стрес.

Практическата последица от това поведение е, че оптимизация, насочена единствено към максимална пропускателна способност, може да доведе до деградация на качеството на обслужване, въпреки че агрегатните показатели остават добри. Следователно при SAN системи, обслужващи латентностно чувствителни приложения, критерият за оптималност следва да включва както производителността, така и устойчивостта на крайните персентили. Следователно оптимизационният критерий следва да бъде формулиран като едновременно поддържане на висока производителност и контролирано латентностно поведение, а не като максимизиране само на един от двата показателя.

Интерпретацията на резултатите показва, че устойчивостта следва да се разглежда като самостоятелна системна характеристика, различна както от суровата производителност, така и от типичното време за обслужване. Тя възниква като резултат от баланса между степента на паралелизъм, механизмите за координация и начина, по който системата реагира на временни натоварвания и конкуренция. Липсата на този баланс води до режими на работа, при които малки промени в натоварването или конфигурацията предизвикват непропорционално силни латентностни отклонения.

В този контекст компромисът между производителност и устойчивост не следва да се интерпретира като бинарен избор, а като континуум от възможни режими на работа. Устойчивият работен диапазон в този континуум зависи от изискванията на конкретните приложения и от допустимия риск от редки, но тежки закъснения. Аналитичният синтез в настоящата глава показва, че осъзнатото управление на координацията при подаване на заявки е ключов фактор за постигане на предсказуемо и устойчиво латентностно поведение в SAN-базирани системи.

От гледна точка на оптимизацията целта не е достигане на максимално възможна пропускателна способност, а идентифициране на режим на работа, при който нарастването на производителността не е съпроводено с непропорционално увеличение на крайната латентност. Получените експериментални резултати показват, че съществува устойчив работен диапазон, в който увеличаването на натоварването все още не води до

непропорционално нарастване на p_{95} и p_{99} . Именно този диапазон представлява целевата област за оптимизация на операциите по запис в SAN среда. Следователно оптимизацията може да бъде формулирана като задача за поддържане на системата в този устойчив работен диапазон.



Фигура 23. Режим на работа по пътя на записа при нарастваща конкуренция - от стабилно поведение към координационен стрес

Фигура 23 представя концептуалната рамка на установения компромис между производителност, координация и устойчивост. Вместо да разглежда отделните режими като независими състояния, фигурата ги представя като непрекъснат преход, при който нарастващата конкуренция постепенно измества системата от устойчив към нестабилен режим на работа.

Следователно преходният режим представлява критичната област, в която чрез своевременно наблюдение на p_{95} , p_{99} и латентностните разпределения могат да бъдат предприети оптимизационни действия, предотвратяващи навлизането на системата в режим на координационен стрес. По този начин оптимизацията се реализира като процес

на ранно откриване и управление на признаците за деградация, а не като реакция след настъпването ѝ.

5.5. Оптимизационна интерпретация за резултатите

Получените експериментални резултати позволяват извеждането на принципи за оптимизация на операциите по запис в SAN среда въз основа на анализираното поведение на системата. В рамките на настоящото изследване оптимизацията не се разглежда като задача за постигане на максимална пропускателна способност или минимална средна латентност, а като задача за поддържане на устойчиво, предсказуемо и контролируемо латентностно поведение при нарастващо натоварване.

Проведените експерименти показват, че между степента на паралелизъм, формирането на входно-изходни опашки и поведението на крайните персенти на латентността съществува устойчива причинно-следствена зависимост. При ниски нива на натоварване системата функционира в режим на контролирана паралелност, при който увеличаването на броя конкурентни операции води до по-добро използване на наличните ресурси без съществено влошаване на латентностните характеристики. След достигане на определен праг на натоварване системата навлиза в режим, при който допълнителното увеличаване на паралелизма вече не подобрява ефективността, а усилва координационните взаимодействия между заявките. Последицата е непропорционално нарастване на времената за изчакване във входно-изходните опашки и рязко разширяване на латентностното разпределение.

От гледна точка на теорията на масовото обслужване този преход може да бъде интерпретиран като изменение на съотношението между интензивността на постъпващия поток от заявки и способността на системата да ги обслужва. При доближаване до границите на обслужващия капацитет времето за изчакване започва да доминира над времето за обслужване, което води до нелинейно нарастване на крайната латентност. Получените експериментални резултати потвърждават именно това поведение и показват, че при SAN системите оптимизацията следва да бъде насочена към управление на опашковите процеси и ограничаване на времената за изчакване, а не единствено към повишаване на производителността на отделните устройства.

На тази основа може да бъде формулиран следният оптимизационен критерий: Поддържане на минимална и предсказуема крайна латентност при запазване на

необходимата пропускателна способност посредством управление на паралелизма и входно-изходните опашки.

Този критерий измества фокуса от максималния брой операции за единица време към устойчивостта на поведението на системата при реални експлоатационни условия. В този смисъл оптималният режим не е режимът с максимална пропускателна способност, а режимът, при който системата запазва предсказуеми стойности на p95 и p99 при приемливо ниво на производителност. По този начин критерият за оптимизация се премества от максимизиране на производителността към управление на устойчивостта на системното поведение.

Експериментите с конкурентно натоварване показват също, че ефективната оптимизация не може да бъде ограничена до отделен компонент от архитектурата. Подобряването на поведението на операциите по запис изисква координирано управление на натоварването по цялата верига на записите – от приложението и хоста, през транспортния слой, до сторидж подсистемата. Следователно оптимизацията следва да се разглежда като системна задача за управление на взаимодействията между слоевете, а не като локална оптимизация на отделен хардуерен ресурс. На практика оптимизацията се реализира чрез координирано управление на взаимодействията между отделните слоеве по пътя на записите, а не чрез независимо оптимизиране на всеки компонент.

На базата на разработения модел и проведените експерименти могат да бъдат формулирани следните принципи за оптимизация на операциите по запис в SAN среда:

- поддържане на баланс между производителност и латентностна устойчивост вместо стремеж към максимална пропускателна способност на всяка цена.
- поддържане на степента на паралелизъм в диапазон, който не предизвиква координационен стрес;
- контрол върху дълбочината на входно-изходните опашки и ограничаване на тяхното неконтролирано нарастване;
- използване на крайните персентили (p95 и p99) като основни индикатори за устойчивост на латентността;
- използване на анализа на латентностните разпределения като основа за идентифициране на прехода към координационен стрес;
- ограничаване на пулсиращите режими на натоварване чрез механизми за изглаждане и координация на заявките;
- наблюдение и управление на конкурентните натоварвания в среди със споделени ресурси;

Наблюдаван ефект	Вероятна причина	Основен показател	Оптимизационно действие	Очакван резултат
Нарастване на p99	Натрупване на заявки във входно-изходните опашки	p99, p99/p95	Намаляване на дълбочината на входно-изходната опашка (QD)	Намаляване на крайната латентност
Разширяване на латентностното разпределение	Координационен стрес	ECDF, p99/p95	Ограничаване на степента на паралелизъм	По-стабилно латентностно поведение
Латентностни пикове	Пулсиращо (burst) натоварване	ECDF, p99	Регулиране на интензивността на входно-изходните заявки	Намаляване на вариационността на латентността
Нестабилност при конкурентен достъп	Конкуренция за споделени ресурси	p99, вариационност на латентността	Прилагане на QoS, изолация или приоритизация на натоварванията	Ограничаване на взаимното влияние между натоварванията
Висока пропускателна способност при нарастващ p99	Работа в близост до капацитета на системата	Пропускателна способност, p99	Работа в устойчив работен диапазон	Баланс между производителност и предсказуема латентност

Таблица 15. Оптимизационни насоки, изведени от експерименталния анализ на операциите по запис в SAN среда

Таблица 15 обобщава изведените от експерименталния анализ зависимости между наблюдаваните латентностни явления и възможните подходи за оптимизация на операциите по запис в SAN среда.

По този начин разработеният системен модел, аналитичната интерпретация и експерименталните резултати се интегрират в единна методологична рамка за оптимизация на операциите по запис в SAN среда. Предложената рамка позволява не само анализ на поведението на системата, но и идентифициране на устойчивите работни режими, ранно откриване на координационен стрес и избор на практически оптимизационни действия. Именно в интегрирането на аналитичния модел, експерименталната методология и практическите оптимизационни принципи се състои основният оптимизационен принос на настоящото изследване.

5.7. Рамка за оптимизация на операциите по запис в SAN среда

На базата на разработения системен модел, аналитичната интерпретация и проведените експерименти се предлага обобщена рамка за оптимизация на операциите по запис в SAN среда. Тази рамка интегрира основните резултати от настоящото изследване в последователен процес за наблюдение, анализ и управление на латентностното поведение. Тя има предписващ характер, тъй като не само интерпретира наблюдаваното поведение на системата, но и определя последователността от действия за избор и прилагане на оптимизационни механизми.

В предложената рамка оптимизацията не се разглежда като еднократно подобрене на отделен компонент или параметър на системата. Вместо това тя се разглежда като процес на непрекъснато измерване, анализ, вземане на решения и последваща оценка на резултатите.

Предложената рамка за оптимизация се реализира чрез четири последователни етапа, които формират затворен цикъл на наблюдение, анализ, избор на оптимизационни действия и последваща оценка на тяхната ефективност.

Първият етап включва измерване на латентностното поведение чрез анализ на емпиричните разпределителни функции (ECDF) и крайните персенти на латентността. Използването на показатели като p_{95} и p_{99} позволява идентифициране на латентностни ефекти и ранни признаци на деградация, които остават скрити при анализ, основан единствено на средни стойности.

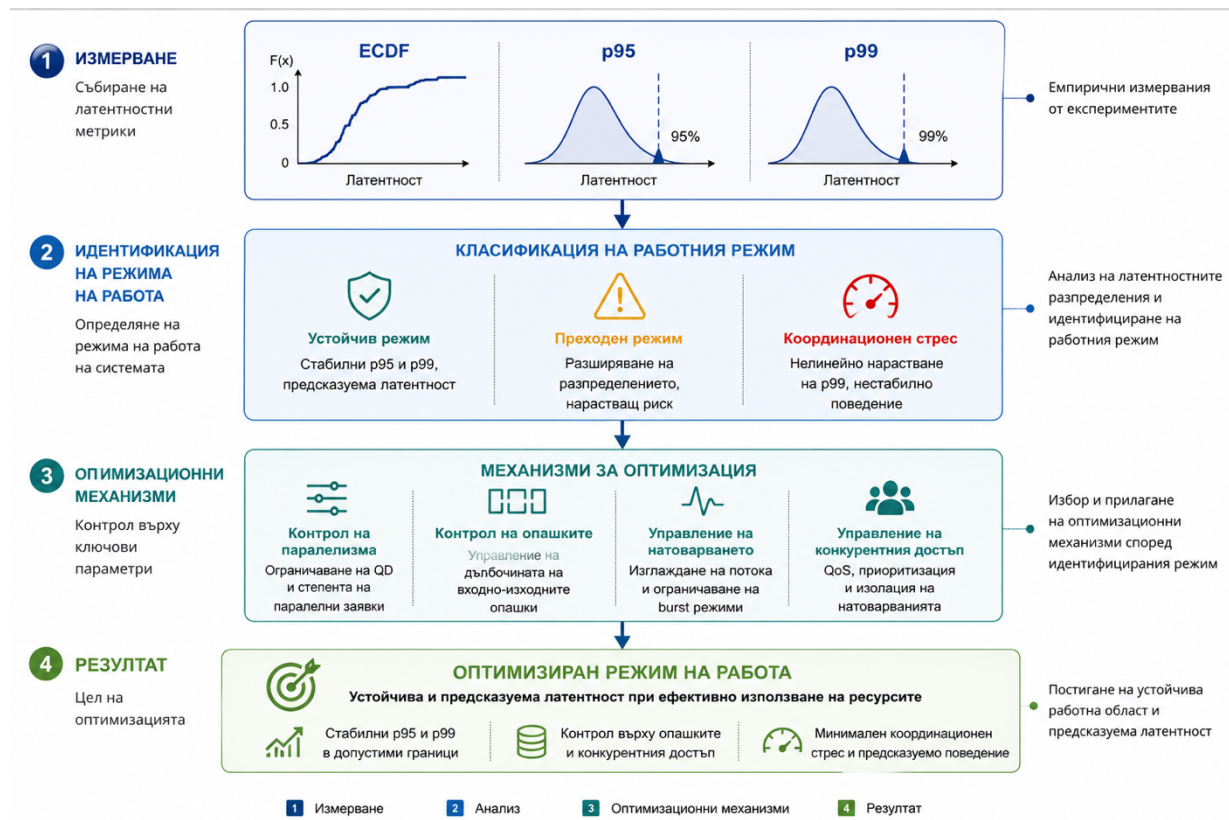
Вторият етап е анализ и идентификация на режима на работа. На базата на формата на латентностното разпределение, поведението на крайните персенти и проявлението на опашковите ефекти системата се класифицира в един от три режима: устойчив режим, преходен режим или режим на координационен стрес. Тази класификация служи като основа за избора на последващи оптимизационни действия.

Третият етап включва избор и прилагане на оптимизационни действия. В зависимост от идентифицирания режим могат да бъдат приложени механизми като управление на степента на паралелизъм, контрол върху дълбочината на входно-изходните опашки, изглаждане на пулсиращите натоварвания, както и ограничаване на конкуренцията при достъп до споделени ресурси.

Четвъртият етап представлява оценка на резултатите и поддържане на устойчив режим на работа. След прилагането на оптимизационните действия латентностното поведение се измерва повторно, което позволява да се оцени тяхната ефективност и при необходимост да се извършат допълнителни корекции. По този начин рамката реализира

непрекъснат цикъл на наблюдение, анализ и адаптивно управление, насочен към поддържане на баланс между пропускателната способност, латентностната устойчивост и ефективното използване на ресурсите.

Обобщената логика на рамката е представена на Фигура 24:



Фигура 24. Рамка за оптимизация на операциите по запис в SAN среда

Фигура 24 представя предложената рамка като затворен цикъл за оптимизация, който интегрира последователните етапи на измерване, анализ, избор на оптимизационни действия и оценка на постигнатите резултати. След всяка промяна в параметрите на системата латентностното поведение се измерва повторно, което позволява оценка на ефективността на приложените оптимизационни действия и при необходимост предприемане на допълнителни корекции. По този начин рамката реализира адаптивен механизъм за обратна връзка, при който всяко оптимизационно действие се валидира чрез повторен анализ на латентностното поведение. Това превръща оптимизацията в непрекъснат процес на наблюдение, анализ и усъвършенстване на поведението на системата, а не в еднократна настройка на нейните параметри.

Наблюдаван ефект	Основен индикатор	Причина	Оптимизационно действие	Цел
Постепенно нарастване на p95/p99	p95, p99	Увеличаване на паралелизма	Ограничаване на паралелизма	Запазване на устойчив режим
Рязко нарастване на p99	p99/p95	Натрупване на заявки	Намаляване на дълбочината на входно-изходните опашки	Ограничаване на времената за изчакване
Разширяване на ECDF	ECDF	Координационен стрес	Настройка на натоварването	Стабилизиране на разпределението
Латентностни пикове	p99	Пулсиращо натоварване	Ограничаване на пулсиращото натоварване чрез регулиране на интензивността на заявките	Намаляване на вариабилността
Конкуренция	p99	Споделени ресурси	Механизми за качество на обслужването (QoS); Изоляция и приоритизиране на конкурентните натоварвания	Изоляция
Висока пропускателна способност	Пропускателна способност и p99	Работа близо до капацитета	Работа в устойчив диапазон	Баланс

Таблица 16. Връзка между наблюдаваните латентностни ефекти и препоръчителните оптимизационни действия

Таблица 16 обобщава как експерименталните наблюдения могат да бъдат превърнати в конкретни оптимизационни действия. По този начин анализът на латентностните разпределения не служи единствено за описание на поведението на SAN системата, а формира основа за избор на мерки за управление на операциите по запис. Това свързва разработения системен модел, експерименталните резултати и оптимизационната цел на дисертационния труд.

Предложената рамка интегрира разработения системен модел, аналитичните зависимости и експерименталните резултати в единен процес за оценка и оптимизация на операциите по запис в SAN среда. Тя позволява не само идентифициране на причините за деградация на латентността, но и избор, прилагане и последваща оценка на оптимизационни действия в зависимост от текущия режим на работа на системата. По този начин експерименталните резултати се трансформират от средство за анализ в

практически приложима методология за управление и оптимизация на SAN среди. Именно тази трансформация - от аналитичен модел за описание на поведението към практически приложима методология за избор на оптимизационни действия - представлява основния научно-приложен резултат на настоящото изследване.

5.6. Практически интерпретации и приложимост на резултатите

Аналитичният синтез на експерименталните резултати позволява извеждане на редица практически интерпретации, които имат значение при проектирането, конфигурирането и експлоатацията на SAN-базирани системи. Макар настоящото изследване да е проведено в контролирана експериментална среда, наблюдаваните закономерности са пряко приложими към реални сценарии, при които системите работят под конкурентно и динамично натоварване.

Първо, резултатите показват, че оценката на производителността на сторидж системи, базирана единствено на средни стойности или максимална пропускателна способност, е недостатъчна за приложения, чувствителни към латентност. В практиката това означава, че системи, които демонстрират висока пропускателна способност при тестове за пикова производителност, могат да се окажат неподходящи за среди, в които стабилността и предсказуемостта на времето за отговор са *критични* (напр. *OLTP бази данни, транзакционни системи, системи за реално време*). Анализът на крайните персентили и формата на латентностното разпределение следва да бъде неразделна част от процеса на оценка и избор на сторидж решения.

Второ, наблюдаваните опашкови ефекти показват, че агресивното увеличаване на паралелизма не е универсална стратегия за подобряване на ефективността. В реални системи неконтролираното насищане на входно-изходните опашки може да доведе до латентностна нестабилност, дори когато средната производителност изглежда приемлива. Това има пряко отражение върху практиките за настройка на параметри като дълбочина на опашката (*queue depth*), брой паралелни потоци и политики за подаване на заявки, които следва да се разглеждат не само от гледна точка на пропускателна способност, но и на латентностна устойчивост.

Трето, експериментите показват, че координацията при подаване на заявки може да бъде ефективен инструмент за управление на латентностното поведение. В практическа среда това означава, че механизми за изглаждане на натоварването, ограничаване на пулсиращи режими или контролирано подаване на заявки могат да доведат до съществено подобрение на качеството на обслужване при относително малка загуба на сурова

производителност (*nanp. rate limiting, I/O throttling, admission control, pacing механизми*). Подобен подход е особено релевантен за многопотребителски среди и системи със споделени ресурси като виртуализирани платформи, споделени SAN среди, мултитенант сторидж услуги.

Четвърто, резултатите от експеримента с конкурентен хост подчертават значението на Noisy Neighbor ефекта в SAN среди. В практиката това поставя акцент върху необходимостта от внимателно планиране на споделянето на ресурси, изолацията на критични натоварвания и мониторинга на латентностните персентили, а не само на агрегатната активност. Наличието на конкурентен достъп може да компрометира латентностната устойчивост на приложения, без това да бъде очевидно от традиционни показатели за натоварване.

Накрая, резултатите показват, че **устойчивото латентностно поведение не е автоматично следствие от модерна архитектура или високопроизводителен хардуер, а изисква съзнателен баланс между производителност, координация и контрол на конкуренцията.** От практическа гледна точка това означава, че устойчивото латентностно поведение в SAN среди се постига не чрез изолирани оптимизации, а чрез координирано управление на конкуренцията и натоварването по цялата верига на записите, от приложението до физическите носители.

За да се илюстрира практическата приложимост на предложените оптимизационни принципи, по-долу са разгледани няколко характерни класа приложения, при които устойчивостта на латентността е критичен фактор за цялостното поведение на системата. Представените примери не представляват отделни експериментални сценарии, а демонстрират как изведените зависимости могат да бъдат използвани при проектирането и експлоатацията на реални SAN-базирани системи.

Примери за практическо приложение в транзакционно-чувствителни среди

Резултатите от настоящото изследване имат пряко значение за клас системи, при които качеството на обслужване се определя не от средната пропускателна способност, а от стабилността и предсказуемостта на латентността при операции по запис. Типични представители на тази категория са транзакционно-ориентирани приложения, при които отделни забавяния могат да доведат до каскадни ефекти на по-високо ниво.

Банкови и финансови транзакционни системи

В системи за обработка на финансови транзакции (напр. плащания, разплащателни сметки, клиринг) производителността често се измерва в транзакции в минута или в секунда. В тези среди средната латентност на записите в базата данни може да остане в приемливи граници, докато редки, но значителни закъснения при операциите по финализиране на транзакциите водят до блокиране на транзакции, изчакване на заключения и изчерпване на пулове от работни нишки.

В такъв контекст наблюдаваните в експериментите опашкови ефекти означават, че система с висока номинална производителност може да започне да проявява нестабилност при пиково натоварване — например в края на отчетен период или при масови плащания. Нарастването на р99 латентността при конкурентен достъп или пулсиращо натоварване директно се отразява върху възприеманата пропускателна способност на приложението, независимо че средният throughput на сторидж слоя остава висок. В контекста на настоящото изследване това означава, че наблюдаваното експериментално нарастване на р99 при конкурентно натоварване може директно да се интерпретира като риск от увеличаване на времето за финализиране на транзакциите.

OLTP бази данни и системи с ACID гаранции

OLTP системите (онлайн транзакционна обработка) са ориентирани към изпълнение на голям брой кратки транзакции с изискване за бързо и предсказуемо завършване. В такива системи операциите по запис обикновено са част от по-широки транзакционни контексти, включващи логване, синхронизация и финализиране към устойчиво съхранение. Поради това латентността на записите по пътя на съхранението има пряко влияние върху времето за изпълнение на транзакциите.

Изискванията за ACID гаранции допълнително засилват тази зависимост. Атомарността и устойчивостта на транзакциите предполагат, че финализирането им е обвързано с потвърден запис към устойчив носител. В резултат дори редки, но значителни забавяния в пътя на записите могат да блокират паралелни транзакции, да удължат времето за задържане на заключения и да предизвикат каскадни ефекти върху изпълнението на системата като цяло.

В този контекст увеличаването на паралелизма на ниво хост или приложение с цел повишаване на номиналната производителност не води непременно до по-добро реално поведение. Както показват експерименталните резултати, агресивният паралелизъм, липсата на координация и интерференцията между конкурентни натоварвания могат да доведат до усилване на крайните стойности на латентността, което ограничава

ефективната пропускателна способност чрез увеличаване на времената за изчакване и активиране на механизми за повторение на транзакциите.

Поради това за OLTP системи с ACID изисквания оценката и оптимизацията на сторидж подсистемата не могат да се базират единствено на средни стойности или пикова пропускателна способност. Анализът на латентностните разпределения и на високите персентили е ключов за осигуряване на стабилно и предсказуемо поведение в среди с висока степен на конкурентност и споделени ресурси.

Мулти-тенант платформи и споделени сторидж среди

В облачни и мулти-тенант среди, при които множество приложения или клиенти споделят общи сторидж ресурси, *Noisy Neighbor* ефектът има особено силно практическо значение. Добавянето на ново натоварване или възникването на временен пик от страна на един клиент може да доведе до деградация на латентността за останалите, без това да е съпроводено с видима промяна в тяхното собствено ниво на активност. Получените резултати от експеримента с конкурентен хост демонстрират именно подобен механизъм на интерференция между независими натоварвания.

Резултатите от експериментите показват, че подобна интерференция се проявява преимуществено в крайните персентили на латентността. В практическа среда това означава, че мониторинг, основан единствено на средни стойности или агрегирани показатели за активност, може да пропусне критични проблеми, които се възприемат от крайните потребители като забавяне, изтичане на времеви ограничения (timeouts) или функционални грешки.

Системи с чувствителност към време за отговор

В системи за онлайн резервации, търговия в реално време или интерактивни финансови приложения, дори редки латентностни пикове могат да доведат до нарушаване на договорени нива на обслужване (SLA). В такива случаи стабилната латентност с малко по-ниска пропускателна способност често е предпочитана пред система с по-висока средна производителност, но непредсказуемо поведение в крайните случаи.

Експерименталните резултати показват, че координираното подаване на заявки и контролът върху конкуренцията могат значително да намалят риска от подобни отклонения. Това подкрепя практики, при които се въвеждат целеви лимити за транзакции в секунда или механизми за изглаждане на натоварването, вместо стремеж към максимално възможна скорост.

В реални транзакционно-чувствителни системи качеството на обслужване се определя преди всичко от най-бавните операции, а не от средната производителност.

Поради това анализът на латентностните разпределения и опашковите ефекти предоставя по-надеждна основа за проектиране, конфигуриране и експлоатация на SAN-базирани системи, отколкото традиционният подход, основан единствено на пропускателната способност и средната латентност. Това потвърждава, че разработената в настоящото изследване методология за анализ и оптимизация е приложима не само в контролирана експериментална среда, но и при проектирането, настройката и експлоатацията на реални SAN-базирани системи с високи изисквания към латентността и предсказуемостта на поведението.

5.7. Обобщение на аналитичния синтез

Аналитичният синтез, представен в настоящата глава, показва, че поведението на SAN-базираните системи при натоварване не може да бъде адекватно описано чрез единични усреднени показатели. Формата на латентностните разпределения и по-специално характеристиките на тяхната опашка разкриват съществени аспекти от вътрешната динамика на системата, които остават скрити при традиционен анализ, основан на средни стойности и агрегатни показатели.

Съпоставянето на различните експериментални сценарии демонстрира наличието на устойчиви модели на поведение, които се проявяват последователно при висока степен на конкуренция за споделени ресурси. Наблюдаваните опашкови ефекти, увеличаването на крайните персентили на латентността и преходът към координационен стрес показват, че устойчивостта на системата се определя не само от нейната производителност, но и от начина, по който управлява взаимодействията между конкурентните заявки.

Извършената аналитична интерпретация позволява получените експериментални резултати да бъдат трансформирани в практически принципи за оптимизация на операциите по запис в SAN среда. На тази основа беше формулирана рамка за оптимизация, която интегрира наблюдението на латентностното поведение, идентифицирането на режима на работа, избора на оптимизационни действия и последващата оценка на тяхната ефективност в единен процес за управление на системата.

Практическата приложимост на разработения подход се изразява във възможността експериментално установените зависимости между натоварването, паралелизма, опашковите ефекти и латентностното поведение да бъдат използвани при проектирането, конфигурирането и експлоатацията на SAN-базирани системи. По този начин анализът на латентностните разпределения се превръща не само в средство за оценка на поведението на системата, но и в основа за избор на конкретни оптимизационни действия.

В обобщение, настоящата глава изпълнява ролята на концептуален мост между експерименталните резултати и заключителните научни приноси на дисертацията. Тя показва как разработеният системен модел, аналитичната интерпретация и експерименталните изследвания се обединяват в практически приложима методология за анализ и оптимизация на операциите по запис в SAN среда. Именно тази трансформация – от експериментални наблюдения към методология за вземане на оптимизационни решения – представлява основният резултат от аналитичния синтез и логичният завършек на настоящото изследване.

ГЛАВА 6. ЗАКЛЮЧЕНИЕ И БЪДЕЩА РАБОТА

6.1. Заключение

Настоящият дисертационен труд разглежда поведението на SAN-базирани системи при конкурентен достъп и висока степен на натоварване, с фокус върху латентността, координационните ефекти и устойчивостта на системата. Чрез разработена експериментална методология и проведен анализ в контролирана среда са установени закономерности в поведението на латентността, които не могат да бъдат адекватно обяснени чрез показатели, базирани единствено на средни стойности.

Установено е, че зависимостта между натоварването и латентността има изразен нелинеен характер. При нарастване на натоварването системата преминава през разграничими режими на работа - режим на контролирана паралелност, преходна зона и режим на координационен стрес. Показано е, че преходите между тези режими са съпроводени с рязка промяна в поведението на латентността и оказват съществено влияние върху предсказуемостта на системата.

Анализът на разпределенията на латентността показва, че крайните персентили (p95, p99) улавят ефекти, които остават скрити при използване на усреднени показатели. Установено е разширяване на разпределението и поява на редки, но значително забавени операции при определени режими на натоварване, което обосновава необходимостта от използване на разпределителни характеристики при оценка на поведението на системата.

Доказано е, че увеличаването на паралелизма не води еднозначно до подобрене на производителността. В определени режими то усилва взаимодействията между компонентите и води до нарастване на вариабилността на латентността, като се наблюдава

увеличаване на крайните персентили дори при запазена или повишена пропускателна способност.

Показано е, че устойчивостта на латентностното поведение зависи в значителна степен от координацията между различните елементи по пътя на записите. При нарушена координация се наблюдава усилване на опашковите ефекти и поява на екстремни латентностни стойности, което води до влошаване на качеството на обслужване.

В обобщение, в дисертационния труд е разработен системен подход за анализ на SAN среди, при който латентността се разглежда като резултат от взаимодействието между множество слоеве и опашкови механизми. Този подход, базиран на анализ на разпределенията и крайните персентили, позволява по-точно описание на поведението на системата и идентифициране на условията, при които настъпва деградация на производителността.

От гледна точка на оптимизацията получените резултати показват, че ефективното управление на операциите по запис не може да се основава единствено на увеличаване на хардуерните ресурси или максимизиране на пропускателната способност. Изследването показва, че чрез управление на паралелизма, контрол върху входно-изходните опашки и анализ на крайните персентили на латентността могат да бъдат идентифицирани и поддържани устойчиви режими на работа, които ограничават проявлението на координационен стрес и подобряват предсказуемостта на системата. По този начин разработеният системен модел, аналитичната интерпретация и експериментално валидираната рамка формират практически приложима методология за анализ и оптимизация на операциите по запис в SAN среда.

6.2. Ограничения и отворени въпроси

Настоящото изследване е проведено в рамките на конкретна архитектурна и експериментална среда, което позволява прецизно изолиране на изследваните фактори, но ограничава директната екстраполация на резултатите към всички възможни реализации на SAN системи.

Фокусът на изследването е поставен върху латентностните характеристики и координационните ефекти по пътя на записите, без детайлно разглеждане на вътрешните алгоритми и политики за управление на системите за съхранение, като вътрешно планиране, кеширане и оптимизации на ниво контролер или носител. Това ограничение е съзнателно прието, тъй като пълното моделиране на тези механизми представлява самостоятелна и значително по-широка изследователска област. В този контекст

изследването е насочено към анализ и оптимизация на системното поведение на ниво взаимодействие между слоевете, а не към оптимизация на конкретна сторидж имплементация или отделен хардуерен компонент.

Допълнително, изследването не разглежда в детайл вътрешната структура и поведение на опашките на ниво сторидж система (например вътрешни контролерни опашки, scheduler-и и механизми за обслужване на заявки). Тези компоненти са разглеждани като част от обобщения системен модел и разглеждани като част от обобщения системен модел, без моделиране на тяхната вътрешна логика.

Наблюдаваните ефекти - формиране на опашки, усилване на крайните стойности на латентността и чувствителност към конкуренция и координация - се разглеждат като устойчиви по характер, макар и с различни количествени проявления в зависимост от конкретната реализация.

Остават отворени въпроси, свързани с поведението на системата при по-динамични и хетерогенни натоварвания, както и с влиянието на механизми за изолация, приоритизация и адаптивен контрол на паралелизма, които не са предмет на настоящото изследване.

Допълнително ограничение е липсата на детайлен анализ на взаимодействието между SAN слоя и по-високите нива на приложния стек. В реални системи тези взаимодействия могат да усилят или смекчат наблюдаваните латентностни ефекти и представляват естествено направление за бъдещи изследвания.

Ясното дефиниране на тези ограничения позволява по-точно позициониране на получените резултати и очертава тяхната валидност в рамките на конкретна експериментална и методологична рамка. Същевременно тези ограничения не променят валидността на основните изводи относно влиянието на конкуренцията, паралелизма и опашковите ефекти върху латентностното поведение, а определят единствено границите на тяхната приложимост.

6.3. Насоки за бъдещи изследвания

Резултатите от настоящия дисертационен труд очертават няколко перспективни направления за бъдещи изследвания, насочени към разширяване на разработения подход за анализ и оптимизация на операциите по запис в SAN среда. Предложените направления надграждат разработения системен модел и създават предпоставки за автоматизирано управление на латентностното поведение при динамични и конкурентни натоварвания.

Първо направление е разработването на динамични механизми за управление на натоварването, които използват не само средни показатели, но и информация за формата на латентностните разпределения и поведението на крайните персентили. Подобни механизми биха позволили ранно идентифициране на признаци за опашкова нестабилност и координационен стрес и биха подпомогнали поддържането на устойчив режим на работа при променящи се условия на натоварване [22, 34].

Второ направление е разработването на аналитични модели, базирани на теорията на масовото обслужване, които да позволят количествено прогнозиране на латентностното поведение и определяне на границите на устойчивите режими на работа. Комбинирането на подобни модели с експерименталните резултати би създало възможност за предварителна оценка на влиянието на различни профили на натоварване и за по-прецизно планиране на оптимизационните действия.

Трето направление е използването на индикатори, базирани на ентропийни характеристики, за ранно откриване на деградация и аномално поведение в SAN среди. Интегрирането им с анализа на латентностните разпределения и опашковите ефекти би позволило изграждането на механизми за автоматизиран мониторинг и ранно сигнализиране при настъпващ координационен стрес.

Четвърто направление е разработването на адаптивни политики за управление на сторидж системите, които използват латентностните разпределения и опашковите индикатори като вход за вземане на решения. Това включва динамично управление на степента на паралелизъм, входно-изходните опашки, кеширащите механизми и конкурентния достъп с цел поддържане на устойчиво и предсказуемо латентностно поведение.

Пето направление е разработването на интелигентни механизми за класификация и прогнозиране на режимите на работа чрез използване на методи от машинното обучение и изкуствения интелект. Натрупаните експериментални данни създават предпоставки за изграждане на модели, базирани на Random Forest и Case-Based Reasoning, които автоматично да разпознават признаци на координационен стрес, да прогнозират преходите между различните режими на работа и да препоръчват подходящи стратегии за оптимизация.

Шесто направление е по-задълбоченото изследване на вътрешните механизми за управление на натоварването в сторидж системите, които в настоящото изследване са разглеждани като част от обобщения системен модел. Това включва анализ на вътрешните

опашки, алгоритмите за планиране, кеширащите механизми и политиките за обслужване, както и оценка на тяхното влияние върху наблюдаваните латентностни ефекти.

Седмо направление е разширяването на анализа към взаимодействието между сторидж слоя и останалите слоеве по пътя на записите, включително приложния, хостовия и транспортния слой. Подобен крайно-до-край подход би позволил по-пълно моделиране на взаимодействията между отделните компоненти на системата и по-задълбочено разбиране на компромисите между производителност, координация и латентностна устойчивост.

В обобщение, разработеният в настоящия дисертационен труд модел създава основа за бъдещо развитие на методи за интелигентно наблюдение, прогнозиране и оптимизация на SAN среди. Разширяването му чрез аналитични модели, методи на машинното обучение и адаптивни механизми за управление представлява естествено продължение на настоящото изследване и открива възможности за изграждане на самонастройващи се системи, способни автоматично да поддържат устойчиво и предсказуемо латентностно поведение при динамично натоварване.

ПРИНОСИ

- 1. Разработено е системно представяне на операциите по запис в SAN среда, при което латентността се разглежда като резултат от взаимодействието между множество слоеве и опашкови механизми по пътя на записите.**
- 2. Разработен е аналитичен модел за интерпретация на поведението на SAN системите при нарастващо натоварване, позволяващ разграничаване на режими на работа - контролирана паралелност, преходна зона и координационен стрес.**
- 3. Разработена е експериментална методология за изследване на операциите по запис в SAN среда, основана на систематично изменение на параметрите на натоварването при контролирани експериментални условия.**
- 4. Разработен е подход за анализ на латентностното поведение, базиран на емпирични разпределителни функции и крайните персентили (p95 и p99), който позволява идентифициране на опашкови ефекти и ранни признаци на деградация, недостъпни при анализ, основан единствено на усреднени показатели.**
- 5. Установено е експериментално, че увеличаването на степента на паралелизъм не води еднозначно до подобряване на производителността, а след определен**

праг усилва вариабилността на латентността и води до непропорционално нарастване на крайните персентили.

6. **Установено е, че че опашковите ефекти представляват надежден индикатор за възникване на координационен стрес** и осигуряват по-висока чувствителност при оценката на поведението на SAN системите в сравнение с традиционните агрегатни показатели.
7. **Разработена е рамка за оптимизация на операциите по запис в SAN среда,** която интегрира разработения системен модел, аналитичната интерпретация и експерименталните резултати в последователен процес за наблюдение, анализ, идентифициране на режима на работа и избор на оптимизационни действия.
8. **Формулирани са практически препоръки за избор на оптимизационни действия** в зависимост от наблюдавания режим на работа, основани на управление на паралелизма, входно-изходните опашки и крайните латентностни персентили.

ИСПОЛЗВАНА ЛИТЕРАТУРА

- [1] L. Kleinrock, *Queueing Systems, Volume I: Theory*, Wiley-Interscience, 1975.
- [2] E. D. Lazowska, J. Zahorjan, G. S. Graham, and K. C. Sevcik, *Quantitative System Performance: Computer System Analysis Using Queueing Network Models*, Prentice Hall, 1984.
- [3] R. Jain, *The Art of Computer Systems Performance Analysis*, Wiley, 1991.
- [4] M. Harchol-Balter, *Performance Modeling and Design of Computer Systems: Queueing Theory in Action*, Cambridge University Press, 2013.
- [5] A. B. Downey, *Think Stats: Probability and Statistics for Programmers*, 2nd ed., O'Reilly Media, 2014.
- [6] R. J. Hyndman and Y. Fan, "Sample Quantiles in Statistical Packages," *The American Statistician*, vol. 50, no. 4, pp. 361–365, 1996.
- [7] W. Whitt, *Stochastic-Process Limits and Their Application to Queues*, Springer, 2002.
- [8] D. Gross, J. F. Shortle, J. M. Thompson, and C. M. Harris, *Fundamentals of Queueing Theory*, 4th ed., Wiley, 2008.
- [9] S. Asmussen, *Applied Probability and Queues*, 2nd ed., Springer, 2003.
- [10] G. Casella and R. L. Berger, *Statistical Inference*, 2nd ed., Duxbury Press, 2002.
- [11] L. Kleinrock, "Time-Dependent Queueing Systems," *Journal of Applied Probability*, vol. 1, no. 3, pp. 383–401, 1964.
- [12] A. Gandhi, M. Harchol-Balter, R. Das, and C. Lefurgy, "Optimal Power Allocation in Server Farms," in *ACM SIGMETRICS*, 2009.
- [13] A. S. Tanenbaum and H. Bos, *Modern Operating Systems*, 4th ed., Pearson, 2015.
- [14] A. C. Arpaci-Dusseau and R. H. Arpaci-Dusseau, *Operating Systems: Three Easy Pieces*, Arpaci-Dusseau Books, 2018.
- [15] R. Love, *Linux Kernel Development*, 3rd ed., Addison-Wesley, 2010.
- [16] J. Axboe, "Linux Block I/O – Introducing Multi-Queue," in *Proc. Linux Plumbers Conf.*, 2013.
- [17] J. Axboe, "Linux Storage Stack and Performance," *LWN.net*, 2017.
- [18] I. Molnar, "CFS Scheduler and I/O Interactions," *Linux Kernel Documentation*, 2025.
- [19] T. Clark, *Storage Area Networks: Designing and Implementing a Mass Storage System*, 2nd ed., Addison-Wesley, 2002.
- [20] M. Kleppmann, *Designing Data-Intensive Applications*, O'Reilly Media, 2017.
- [21] J. H. Saltzer and M. F. Kaashoek, *Principles of Computer System Design*, Morgan

Kaufmann, 2009.

- [22] INCITS T11, *Fibre Channel – Framing and Signaling (FC-FS)*, ANSI Standard.
- [23] M. Jung et al., “Understanding SSD Internals,” in *ACM SIGMETRICS*, 2011.
- [24] C. Min et al., “Understanding the Characteristics of Flash Storage,” in *IEEE MSST*, 2014.
- [25] M. Jung et al., “Revisiting Widely Held SSD Expectations and Rethinking Flash Storage System Design,” in *ACM SIGMETRICS*, 2013.
- [26] D. Narayanan et al., “SSD Performance Modeling,” *IEEE Computer*, vol. 43, no. 4, 2010.
- [27] X. Wu et al., “Analytical Modeling and Evaluation of SSD Write Performance,” in *USENIX FAST*, 2013.
- [28] E. Krevat et al., “Disks Are Like Snowflakes,” in *HotOS*, 2013.
- [29] Y. Wang et al., “Flash on Rails: Consistent Flash Performance through Redundancy,” in *USENIX ATC*, 2014.
- [30] S. Oikawa et al., “Fine-Grained IO Scheduling for SSDs,” in *USENIX FAST*, 2018.
- [31] B. Schroeder and G. A. Gibson, “Understanding Latent Sector Errors and Reliability in Modern Storage,” in *USENIX FAST*, 2016.
- [32] NVM Express, *NVM Express Base Specification*, Rev. 2.2, 2025.
- [33] NVM Express, *NVMe over Fabrics Specification*, Rev. 1.1a, 2021.
- [34] Y. Zhang et al., “Performance Analysis of NVMe over Fabrics,” in *IEEE MSST*, 2019.
- [35] S. Bjørling et al., “Linux NVMe Driver and Performance Analysis,” in *Linux Plumbers Conf.*, 2015.
- [36] S. Choi et al., “Performance Evaluation of NVMe over Fabrics,” in *IEEE ICC*, 2018.
- [37] J. Kim et al., “NVMe SSD Performance Isolation for Multi-Tenant Environments,” in *USENIX FAST*, 2019.
- [38] H. Li et al., “RDMA-Based NVMe over Fabrics Performance Characterization,” in *IEEE IPDPS*, 2018.
- [39] J. Liu et al., “Understanding NVMe-over-Fabrics Performance in Data Centers,” in *ACM SoCC*, 2019.
- [40] J. Dean and L. A. Barroso, “The Tail at Scale,” *Communications of the ACM*, vol. 56, no. 2, pp. 74–80, 2013.
- [41] A. Fox and E. Brewer, “Harvest, Yield, and Scalable Tolerant Systems,” in *HotOS*, 1999.
- [42] S. Kanev et al., “Profiling a Warehouse-Scale Computer,” in *ISCA*, 2015.
- [43] A. Klimovic et al., “Understanding Ephemeral Storage Performance in the Cloud,” in *USENIX ATC*, 2018.
- [44] J. Leverich and C. Kozyrakis, “Reconciling High Server Utilization and Sub-Millisecond

QoS,” in *EuroSys*, 2014.

[45] M. Mitzenmacher, “The Power of Two Choices in Randomized Load Balancing,” *IEEE TPDS*, 2001.

[46] M. K. Aguilera et al., “Performance Debugging for Distributed Systems of Black Boxes,” in *ACM SOSP*, 2003.

[47] M. Lillibridge et al., “Sparse Indexing,” in *USENIX FAST*, 2009.

[48] B. Debnath et al., “ChunkStash,” in *USENIX ATC*, 2010.

[49] W. Xia et al., “FastCDC,” in *USENIX ATC*, 2016.

[50] C. Wohlin et al., *Experimentation in Software Engineering*, Springer, 2012.

[51] F. Shull et al., *Guide to Advanced Empirical Software Engineering*, Springer, 2008.

[52] D. Feitelson, *Workload Modeling for Computer Systems Performance Evaluation*, Cambridge University Press, 2015.

[53] *fio – Flexible I/O Tester*, Official Documentation, ver. 3.x.

[54] IBM Redbooks, *Implementing IBM SAN Volume Controller with IBM Spectrum Virtualize*, 2021.

[55] IBM Redbooks, *Data Reduction Pools and Deduplication*, 2018.

[56] IBM Redbooks, *Performance and Best Practices Guide for Spectrum Virtualize 8.5*, 2022.

[57] IBM Redpaper, *IBM FlashCore Module (FCM) Product Guide*.

[58] R. Lu et al., “NVMe SSD failures in the field: The fail-stop and the fail-slow,” in *Proc. USENIX ATC*, 2022.

[59] I. Jo, “Toward ultra-low latency SSDs: Analyzing the impact on data-intensive workloads,” *Electronics*, vol. 12, no. 1, 2023.

[60] Y. Zhang et al., “Characterizing tail latency in cloud storage systems,” in *Proc. IEEE IC2E*, 2022.

[61] A. Yang et al., “Set-associative mapping to improve data throughput and reduce long tail latency in SSDs,” *Journal of Systems Architecture*, vol. 155, 2025.

Авторски публикации по дисертацията

[62] Torres G.. *Entropy-Aware AI-Powered Ransomware Detection Using IBM FlashCore Module 4 in SAN Environments*. International Conference Automatics and Informatics (ICAI), 2026, ISBN:979-8-3315-8611-9, DOI:10.1109/ICAI67591.2025.11325084, 100-106; 2025

[63] Torres G.. *Optimization of Write Operations in SAN Environments Through Intelligent Compression, Deduplication and Thin Provisioning*. TELECOM 2025, 33, 33rd National Conference with International Participation (TELECOM), 2026, ISBN:978-1-6654-6564-9, ISSN:2837-5246, DOI:10.1109/TELECOM66943.2025.11304087, 1-4; 2025

[64] Torres G.. *Storage Virtualization: Optimizing Data Center Efficiency and Performance*. 2/2024, 2024, ISSN:2535-0668, 65-66; 2024

[65] Torres G., Ivanov Vl.. *A Survey on Storage Area Network*. Robotics, Automation and Mechatronics '23, Prof. Marin Drinov Publishing House of Bulgarian Academy of Science, 2023, ISSN:1314-4634, 22-26; 2023

[66] Торрес Г., Иванов Вл.. *Приложения на Fiber Channel, ISCSI, Infiniband и FCOE в Storage Area Network*. Robotics, Automation and Mechatronics '23, RAM 2023, Prof. Marin Drinov Publishing House of Bulgarian Academy of Sciences, 2023, ISSN:1314-4634, 30-32; 2022

[67] Торрес Г.. *Анализ на методи за оптимизация на записи в SAN среда*. Prof. Marin Drinov Publishing House of Bulgarian Academy of Sciences, 2022, ISSN:1314-4634, 69-72; 2022

Декларация за оригиналност на резултатите

Декларирам, че настоящата дисертация съдържа оригинални резултати, получени при проведени от мен научни изследвания. Резултатите, които са получени, описани и/или публикувани от други учени, са надлежно и подробно цитирани в библиографията.

Настоящата дисертация не е прилагана за придобиване на научна степен в друго висше училище, университет или научен институт.

Подпис:

Приложение А

Референтна конфигурация на `fio` за експерименталните сценарии

В настоящото приложение е представена референтна конфигурация на `fio` `jobfile`, използвана като базов шаблон за всички експериментални сценарии в дисертационния труд. Показаният пример илюстрира фиксирания експериментален контекст (тип операции, размер на блока, брой паралелни задачи, работен набор и механизъм за логване), докато конкретните експерименти се реализират чрез целенасочена промяна на ограничен набор параметри, описани подробно в Глава 4.

Поради тази причина в приложението не се изброяват всички варианти на `job` файлове, а се представя един каноничен шаблон, върху който стъпват отделните експериментални конфигурации.

```
[global]
filename=/dev/oxigen
ioengine=libaio
direct=1
thread=1
group_reporting=1

; Натоварване (baseline)
rw=randwrite
bs=4k
numjobs=4
iodepth=32

; Работен набор
size=256G

; Времеви параметри
time_based=1
runtime=300
ramp_time=30

; Измерване и логване
latency_percentiles=1
percentile_list=50:95:99
log_avg_msec=1

write_clat_log=write_clat
write_iops_log=write_iops
log_unix_epoch=1

[baseline]
```

В показания шаблон е зададена референтна стойност `iodepth=32`, като в рамките на конкретните експерименти този параметър се изменя съгласно описаната методология. Тестовите се изпълняват с фиксирана продължителност, за да се гарантира съпоставимост на латентностните измервания между различните експериментални сценарии.

В рамките на отделните експерименти върху този шаблон се прилагат следните целенасочени изменения:

- за Експеримент 4.2 се варира параметърът `iodepth` ($QD = 32, 64, 128$);
- за Експеримент 4.3 се изменя времевата структура на подаване чрез rate-based ограничаване;
- за Експеримент 4.4 се въвежда фазова координация между паралелните задачи;
- за Експеримент 4.5 се използва конкурентен хост (Noisy Neighbor) с отделен адресен диапазон.

Конкретните стойности и протоколът за изпълнение са описани в съответните експериментални раздели.

Събиране на логове

По време на експериментите латентностните и производителните показатели се събират на хостово ниво чрез вградените механизми за логване на `fio`. За всяко изпълнение се записват логове за completion latency (`write_clat_log`), IOPS във времето (`write_iops_log`) и текстов изход с агрегирани статистики и персентили. Логовете се маркират с UNIX времеви отпечатък, което позволява синхронизация между повторения и паралелни натоварвания.