

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет _____ *Інфокомунікацій* _____
(повна назва)
Кафедра _____ *Інформаційно-мережної інженерії* _____
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА
Пояснювальна записка

рівень вищої освіти _____ *другий (магістерський)* _____

_____ *Дослідження механізму контролю якості реконструйованих* _____
_____ *відеоданих у реальному часі* _____
(тема)

Виконав:
студент 2 курсу, групи _____ *ІММ-21-1* _____
_____ *Рогаль В.В.* _____
(прізвище, ініціали)

Спеціальність _____ *172 Телекомунікації та* _____
_____ *радіотехніка* _____
(код і повна назва спеціальності)

Тип програми _____ *освітньо-професійна* _____
(освітньо-професійна або освітньо-наукова)

Освітня програма _____ *Інформаційно-мережна* _____
_____ *інженерія* _____
(повна назва освітньої програми)

Керівник _____ *ст. викл. Твердохліб В.В.* _____
(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри _____ *Безрук В.М.* _____
(підпис) (прізвище, ініціали)

2022 р.

Не містить відомостей, заборонених
до відкритого публікування

Керівник _____ /*В.В.Твердохліб*

Студент _____ / *В.В.Рогаль*

Харківський національний університет радіоелектроніки

Факультет Інфокомунікацій
Кафедра Інформаційно-мережної інженерії
Рівень вищої освіти другий (магістерський)
Спеціальність 172. Телекомунікації та радіотехніка
(код і повна назва)
Тип програми Освітньо-професійна
(освітньо-професійна або освітньо-наукова)
Освітня програма Інформаційно-мережна інженерія
(повна назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри _____
(підпис)

« 21 » жовтня 2022 р.

ЗАВДАННЯ

НА КВАЛІФІКАЦІЙНУ РОБОТУ

студентові Рогалю Владиславу Віталійовичу

(прізвище, ім'я, по батькові)

1. Тема роботи Дослідження механізму контролю якості реконструйованих відеоданих у реальному часі

затверджена наказом університету від 21 жовтня 2022 р. № 1376 Ст

2. Термін подання студентом роботи до екзаменаційної комісії грудня 2022 р.

3. Вихідні дані до роботи Обґрунтувати необхідність контролю якості реконструйованих відеоданих у реальному часі. Виявити механізми у складі технології кодуюння, що, по-перше, можуть впливати на рівень якості та інформаційної інтенсивності, а по-друге – можуть налаштовуватися. Дослідити можливість інтеграції методів контролю рівня якості відеоінформації в існуючі системи передачі. Розробити схеми контролю якості та інтенсивності відеоінформації у реальному часі.

4. Перелік питань, що потрібно опрацювати в роботі Вступ

1. Ключові стратегії забезпечення контролю рівня якості відновленої відеоінформації

2. Принципи контролю інформаційної інтенсивності потоку кодованих відеокадрів у рамках чинних систем передачі.

3. Механізми зміни інформаційної інтенсивності кодованого відеоінформаційного потоку з урахуванням вимог до якості реконструйованих кадрів

4. Додаткові заходи забезпечення якості відеоінформації у мережі

Висновки

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (п.5 включається до завдання за рішенням випускової кафедри) _____
слайди презентації в форматі Power Point (назва та мета роботи, чинники, що визначають необхідність контролю якості відео, підходи до забезпечення контролю рівня якості відео, принцип контролю інформаційної інтенсивності потоку кодованих відеокадрів, висновки)

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Терміни виконання етапів роботи	Примітка
1	Вступ		
2	Ключові стратегії забезпечення контролю рівня якості відновленої відеоінформації		
3	Принцип контролю інформаційної інтенсивності потоку кодованих відеокадрів у рамках чинних систем передачі		
4	Механізми зміни інформаційної інтенсивності кодованого відеоінформаційного потоку з урахуванням вимог до якості реконструйованих кадрів		
5	Додаткові заходи забезпечення якості відеоінформації у мережі		
6	Висновки		
7	Оформлення пояснювальної записки		

Дата видачі завдання 21 жовтня 2022 р.

Студент _____
(підпис)

Керівник роботи _____ ст. викл. Твердохліб В.В.
(підпис) (посада, прізвище, ініціали)

РЕФЕРАТ

Пояснювальна записка: 68 с., 19 рис., 19 джерел, 1 додаток

ПВСШ, MPEG, КВАНТУВАННЯ, ДКП, H.264, ЯСКРАВИСНО-ХРОМАТИЧНИЙ ОПИС, БУФЕР ПЕРЕДАВАЧА

Об'єкт дослідження – методи контролю рівня якості відеоінформаційного потоку у ході його трансляції мережею.

Мета роботи – дослідження існуючих підходів до побудови механізмів забезпечення сталої передачі відеотрафіку з забезпеченням вимог до рівня якості.

Виконується огляд проблематики, пов'язаної з кодуванням відеоконтенту. Обґрунтовується необхідність узгодження інформаційної інтенсивності відеопотоку з динамікою пропускної спроможності мережі. Розглядаються варіанти реалізації методу контролю неузгодженості інформаційної інтенсивності відео з пропускною спроможністю мережі. Досліджується схема контролю рівня інтенсивності відео з можливістю забезпечення показників якості.

THE ABSTRACT

Explanatory note: 68p., 19 fig., 19 sources, 1 app.

PSNR, MPEG, QUANTIZATION, DCT, H.264, BRIGHTNESS AND CHROMATIC DESCRIPTION, TRANSMITTER BUFFER

The object of the research is the methods of controlling the quality level of the video information stream during its broadcast over the network.

The purpose of the work is to study the existing approaches to the construction of mechanisms for ensuring the stable transmission of video traffic with quality requirements.

An overview of the issues related to the encoding of video content is performed. The necessity of harmonizing the information intensity of the video stream with the dynamics of network bandwidth is substantiated. We consider options for the implementation of the method of controlling the inconsistency of the information intensity of the video with the bandwidth of the network. A video intensity level control scheme with the possibility of ensuring quality indicators is being studied.

ЗМІСТ

	С.
ПЕРЕЛІК СКОРОЧЕНЬ.....	9
ВСТУП.....	10
1 КЛЮЧОВІ СТРАТЕГІЇ ЗАБЕЗПЕЧЕННЯ КОНТРОЛЮ РІВНЯ ЯКОСТІ ВІДНОВЛЕНОЇ ВІДЕОІНФОРМАЦІЇ	12
1.1 Фактори, що визначають проблематику трансляції мережевого трафіку	12
1.2 Міра якості реконструйованих відеоданих	17
1.3 Концепція забезпечення якості відеоінформації	19
1.4 Механізми реалізації концепції забезпечення якості відеоінформації	20
1.4.1 Аналіз базисної моделі MPEG для виявлення механізмів впливу на показник скорочення бітового об'єму відео	22
1.5 Попередні висновки	26
2 ПРИНЦИП КОНТРОЛЮ ІНФОРМАЦІЙНОЇ ІНТЕНСИВНОСТІ ПОТОКУ КОДОВАНИХ ВІДЕОКАДРІВ У РАМКАХ ЧИННИХ СИСТЕМ ПЕРЕДАЧІ	27
2.1 Обґрунтування неефективності застосування методів кодування відео з фіксованим коефіцієнтом стиснення	27
2.2 Взаємозв'язок основної технології стиснення відеоінформації з алгоритмами контролю інформаційної інтенсивності	28
2.3 Непряма оцінка поточного рівня пропускної спроможності з використанням величини RTT	29
2.4 Оцінка характеру зміни поточного рівня пропускної здатності мережі відносно контрольного моменту часу	36
2.5 Використання буферу обмеженого робочого об'єму у процесі контролю рівня інформаційної інтенсивності та якості відеоінформаційного потоку	40
3 МЕХАНІЗМИ ЗМІНИ ІНФОРМАЦІЙНОЇ ІНТЕНСИВНОСТІ КОДОВАНОГО ВІДЕОІНФОРМАЦІЙНОГО ПОТОКУ З УРАХУВАННЯМ ВИМОГ ДО ЯКОСТІ РЕКОНСТРУЙОВАНИХ КАДРІВ	42
3.1 Існуючі парадигми маніпулювання інформаційною інтенсивністю відео потоку	42

3.1.1 Контроль рівня інформаційної інтенсивності відео шляхом зміни роздільної здатності кадрів на прикладі технології ABR	42
3.1.2 Контроль рівня інформаційної інтенсивності та якості відео контенту шляхом зміни кількості структурних одиниць для опису відеокадрів чи їх окремих фрагментів	45
3.1.3 Механізм контролю якості відео на базі виявлених інструментів у складі MPEG, здатних впливати на рівень інформаційної інтенсивності потоку	50
4 ДОДАТКОВІ ЗАХОДИ ЗАБЕЗПЕЧЕННЯ ЯКОСТІ ВІДЕОІНФОРМАЦІЇ У МЕРЕЖІ	56
4.1 Обґрунтування необхідності залучення допоміжних інструментів забезпечення якості відеоінформації у ході трансляції	56
4.2 Використання завадостійкого кодування на базі алгоритму Хемінга для підвищення якості відеоданих	56
4.2.1 Загальні засади побудови методу Хемінга	56
4.2.2 Лістинг коду програми	57
4.2.3 Опис програмних одиниць	60
4.3 Результати кодування	64
ВИСНОВКИ.....	65
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ.....	67
ДОДАТОК А СЛАЙДИ ПРЕЗЕНТАЦІЇ	69

ПЕРЕЛІК СКОРОЧЕНЬ

MPEG – сімейство стандартів відео кодування;

JPEG – технологія кодування зображень;

ДКП – дискретно-косинусне перетворення;

H.264 — технологія високоефективного кодування відео;

ПВСШ – пікове відношення сигнал/шум;

СКВ – середньоквадратичне відхилення;

VoD – (Video on Demand) – сервіси «відео на запит»;

UHD – (Ultra-High Definition) – надвисока роздільна здатність;

LQ – (Low Quality) – формат низької роздільної здатності відео;

QoS – (Quality of Service) – система забезпечення якості мережевих сервісів;

FullHD – формат високої роздільної здатності.

ВСТУП

Важливість проблематики, пов'язаної з необхідністю підтримки необхідного рівня реконструйованого відео, сьогодні вадко переоцінити.

У першу чергу, це зумовлюється надзвичайно широкою поширеністю відео контенту як мережею, так і локальними сховищами. При цьому, частка відео у масі усього трафіку на сьогодні складає не менш, ніж 83-84%. До того ж, даний показник буде збільшуватися, як стверджують дані дослідження, що проводиться щорічно [1].

Така залежність у свою чергу, не є випадковою, що пояснюється наступним:

- сервіси відеохостингу, «відео на запит» (VoD), поступу до потокового мовлення реального часу, відеоконференцій та ін. набули широкої застосовуваності та продовжують розвиватися;

- розвиток технологічного базису за останні роки, з одного боку, спричинив здешевлення кінцевого обладнання для виходу у мережу (та сприяв зростанню кількості користувачів), а з іншого боку – розширив можливості користувачів з застосування усього діапазону можливих мережевих сервісів, у т.ч. тих, що прямо чи опосередковано пов'язані з передаванням відеотрафіку;

- постійно зростає відсоток мережевих сервісів, що використовують відео як одну зі своїх компонент.

Разом з цим слід зазначити, що на тлі цього спостерігається зростання вимог щодо рівня якості відео контенту, зокрема:

- ряд форматів відео (у першу чергу, це стосується надвисоких роздільних здатностей) орієнтуються на підтримку частоти слідування кадрів щонайменше удвічі більшу, ніж сьогодні є традиційною;

- зменшується відсоток LQ-відео, поступаючись формату UHD (відсоток відео формату HD на сьогодні є незмінним).

Окрім цього, існує ряд сервісів відеомовлення за класифікацією QoS, що самі собою вимагають забезпечення відносно жорстких вимог до якості (наприклад, інтерактивне відео).

Також обов'язково слід згадати інтелектуальні системи, що не лише фіксують відеоряд, але також на базі нього розпізнають образи і т.д, що

справедливо для відео мереж Розумних міст, систем комп'ютерного зору тощо. Для них, та подібних систем, підтримка якості відеоряду є тотожною забезпеченню самої їх функціональності.

Отже за таких умов усі питання, пов'язані з дослідженням, розробкою та впровадженням механізмів контролю якості реконструйованого відео, безумовно, є актуальними.

1. КЛЮЧОВІ СТРАТЕГІЇ ЗАБЕЗПЕЧЕННЯ КОНТРОЛЮ РІВНЯ ЯКОСТІ ВІДНОВЛЕНОЇ ВІДЕОІНФОРМАЦІЇ

1.1 Фактори, що визначають проблематику трансляції мережевого трафіку

Сучасні інфокомунікаційні мережі у загальному випадку функціонують в умовах [1]:

- у середньому високих показників інформаційної інтенсивності потоків даних, що надсилаються мережевим середовищем;
- суттєвого рівня складності, неоднорідності та специфічної структурованості трафіку.

У свою чергу, високі рівні інформаційної інтенсивності потоків даних, є наслідком (рисунок 1.1) [2]:

- значного обсягу додатків, сервісів та мережевих служб, які функціонують у межах загального інформаційного середовища;
- суттєва кількість користувачів майже кожного мережевого сервісу/дodatку;
- існування великої кількості клієнтських терміналів, на базі яких виконуються звернення до мережевих послуг та ресурсів (у середньому на одного користувача за різними оцінками приходиться від 3 до 5 терміналів).
- ріст обсягу бітових потоків від окремих джерел.

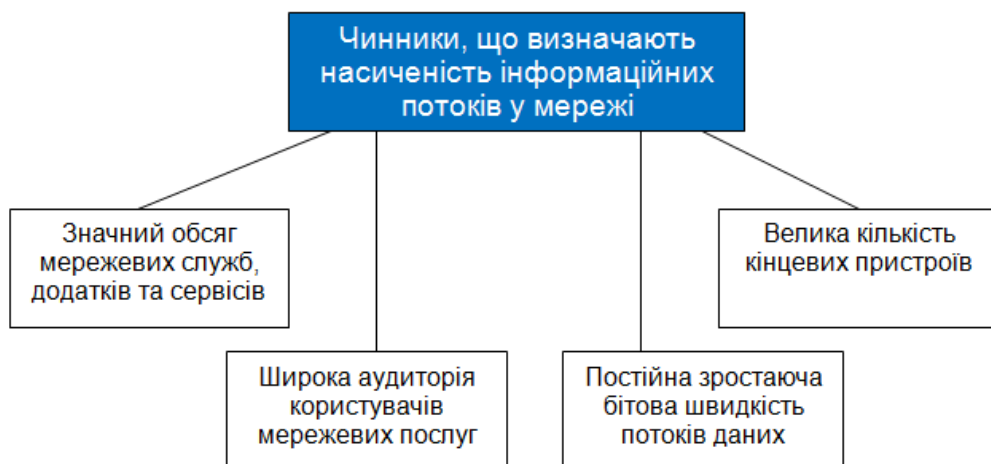


Рисунок 1.1 - Фактори, що впливають на ріст інформаційної інтенсивності потоків даних у мережі

При цьому, специфічність структури трафіку, його неоднорідність та високий рівень складності викликані такими чинниками, як:

- збільшення випадків взаємної інтеграції сервісів на базі мережі, для яких характерними є значні відмінності типів даних, які ними породжуються та значні відмінності інформаційних інтенсивностей таких даних;
- присутність у мережі великої кількості сервісів, що генерують трафік різного рівня інтенсивності;
- використання методів інтелектуальної обробки трафіку на кінцевих та мережевих вузлах;
- надсилання єдиним середовищем даних відкритого типу, шифрованої та/або маскованої інформації (рис.1.2).

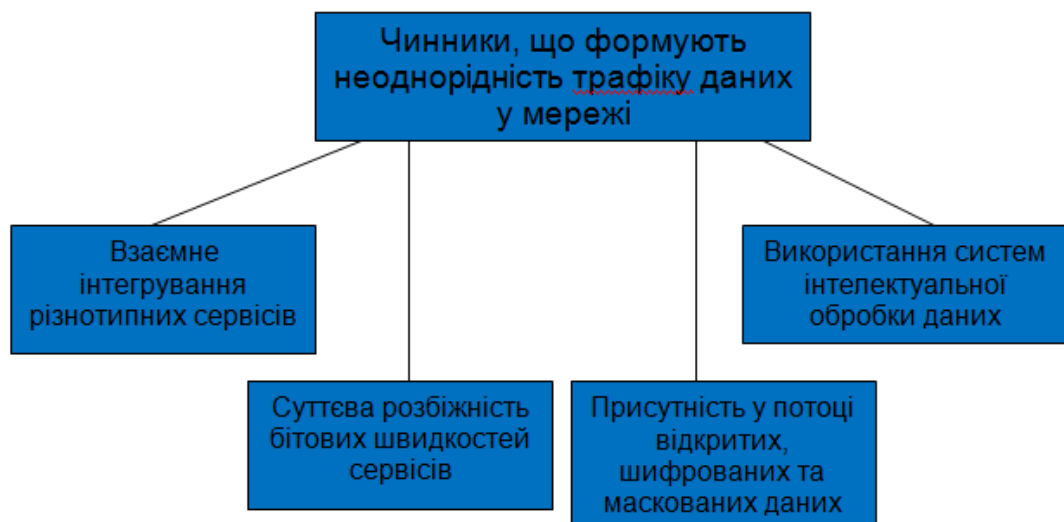


Рисунок 1.2 – Фактори, які визначають складність та неоднорідність структури трафіку даних у мережі

Разом з тим, судячи з результатів періодичних досліджень [1], які стосуються закономірностей розвитку мереж, останні роки спостерігається невпинне та прогресуюче зростання відсотку трафіку «важких» типів, тобто, якому відповідають найбільші величини інформаційної інтенсивності.

На тлі цього, максимальна частка «важкого» трафіку належить відео.

Це пояснюється тим, що:

1) Спостерігається збільшення у мережі відсотку відео високих роздільних здатностей.

Зазначений чинник, при цьому, зумовлений тим, що [1-4]:

- постійно зростає частка мережевих додатків та сервісів, де тим чи іншим чином застосовується інтелектуальний аналіз даних, що стосується також відеоінформації. При цьому, методи інтелектуальної обробки відеоінформації зможуть бути ефективними лише за умови, якщо роздільна здатність відеокадру буде значно вищою, ніж 2К (FullHD) з рівнем ПВСШ не менше, ніж 30-35 дБ [2, 3];

- роздільна здатність пристроїв відображення перевищує фактичні можливості мережевих каналів стосовно трансляції відео контенту, що, у свою чергу, зумовлює жорсткішання вимог, які кінцеві користувачі висувають до якості надання відеоінформаційних сервісів.

2) Постійне зростання переліку мережевих сервісів, складовою компонентою яких тією чи іншою мірою є відео. Наприклад:

- блогосфери, орієнтовані на відеоконтент;
- відео як головний елемент, або складова частина матеріалів презентацій;
- комунікації;
- сервіси потокового відео мовлення, на кшталт VoD;
- сервіси та системи відеоспостереження у реальному масштабі часу;
- сервіси відеохостингу;
- інфопродукти, інтерактивне навчання тощо.

На сьогодні трафіку відео у мережі відповідає не менш, ніж 83% сумарного обсягу мережевих даних.

В абсолютних величинах це приблизно 250 ексабайт щомісяця, як показано рис. 1.3.

При цьому, судячи з аналізу даного рисунку, за останні 5 років загальний об'єм відеоінформації, яка надсилається мережею, зріс приблизно у 4,5 разів.

Водночас, на тлі росту обсягів відеоданих темпи збільшення пропускної спроможності мережі у середньому значно запізнюється, що ілюструє рисунок 1.4.

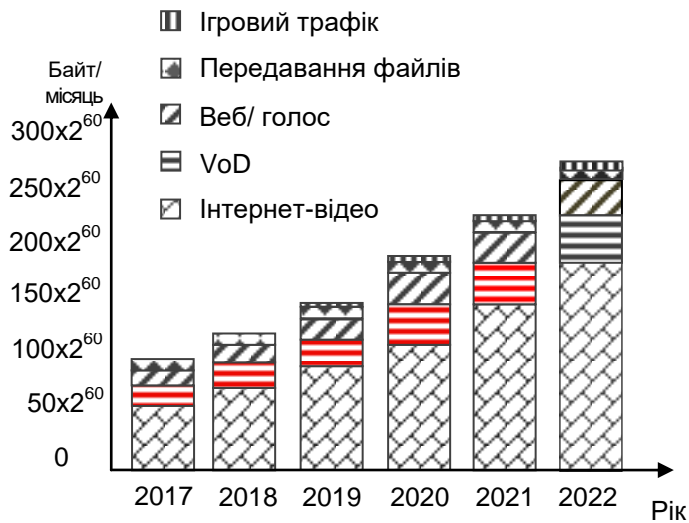


Рисунок 1.3 – Темпи росту у мережі обсягу різномітної інформації за період 2017-2022 років

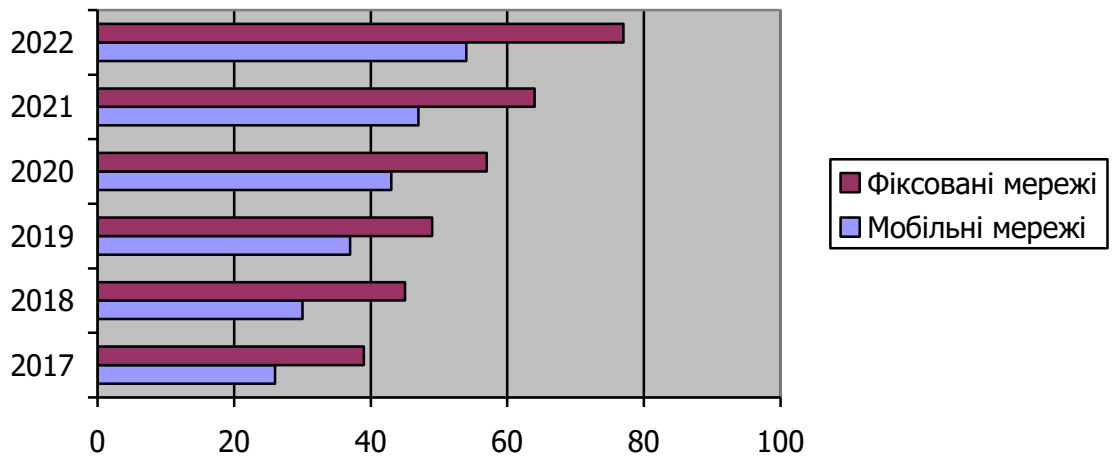


Рисунок 1.4 – Ріст середньої пропускної спроможності мереж мобільного та фіксованого доступу за період 2017-2022 років

Тобто, судячи з даних, розміщених на рис.1.3 та 1.4, можемо дійти висновку про те, що обсяг мережевої відеоінформації протягом зазначеного періоду збільшився у 5 рази, тоді як середній показник пропускної спроможності – не більш, ніж 2 рази [1].

Отже, у зазначених умовах складається тенденція для:

- неможливості гарантування якості відеоінформаційних сервісів згідно з нормативами QoS;

- у ряді випадків – неможливості повноцінного функціонування мережесервісів будь-яких типів узагалі.

Зазначене вище є наслідками:

- дисбалансу темпів росту відеоінформації та пропускну́ї спроможності, який, при цьому, має чітку тенденцію;
- складності обробки трафіку з неоднорідною структурою на мережесервісах, що спричинює ріст затримки передавання.

Результатами впливу вищезазначеного є:

- часткове чи повне руйнування кадру або масиву кадрів (рис.1.5);
- завмирання процесу трансляції;
- повна втрата одного чи масиву кадрів [4].

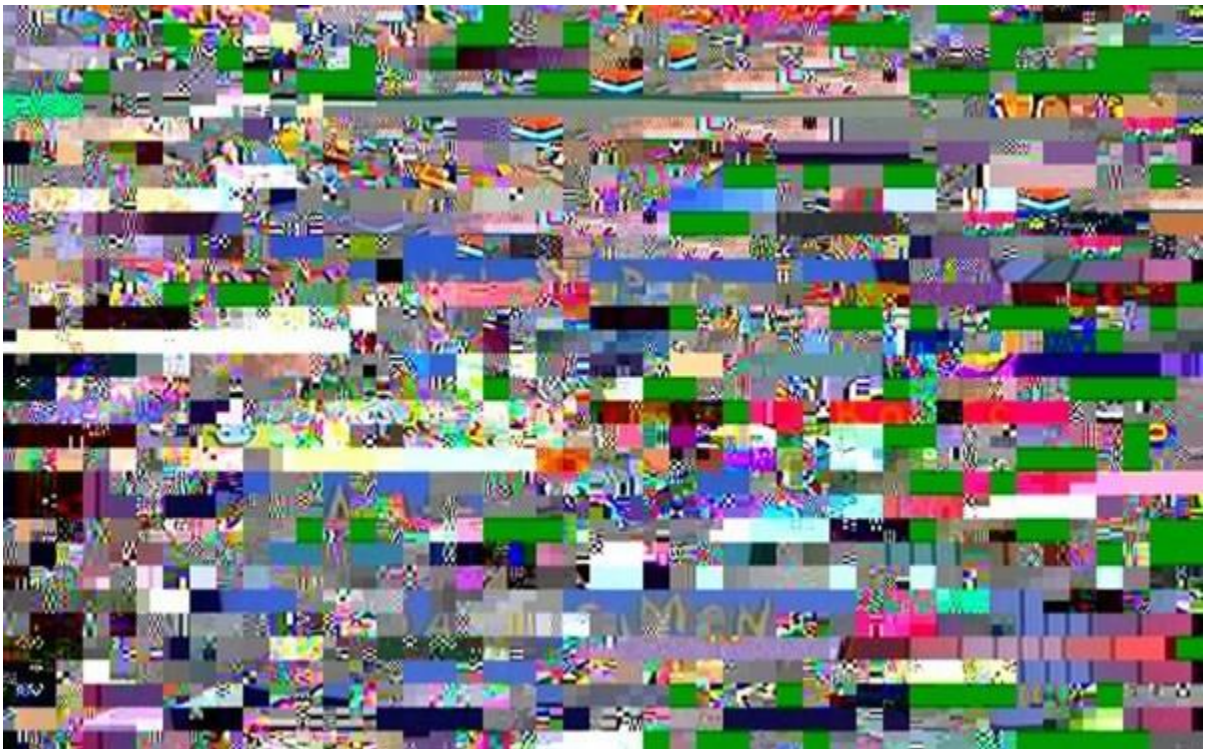


Рисунок 1.5 – Приклад руйнування відеокадру

Для усунення зазначених чинників, що негативно впливають на якість мережесервісів відео, у загальному випадку може бути задіяно такі підходи, як:

- резервування мережесервісів для відеотрафіку;
- скорочення у мережі кількості пакетів що належать відеоінформаційним сервісам.

Разом з тим, виходячи з інформації, яку зазначено на рис. 1.3 та 1.4 стає очевидним, що перший підхід не може бути реалізовано. Тому зараз широко використовуються технології кодування, які сприяють суттєвому скороченню інформаційної інтенсивності відео а відтак – зменшують у мережі кількість пакетів, що належать відеоінформаційним сервісам.

При цьому, у будь-якому разі, за виключенням невеликого відсотку випадків застосування технологій кодування без втрат, вихідна відеоінформація тією чи іншою мірою спотворюється.

Виходячи з цього, до процесу кодування висуваються вимоги щодо забезпечення необхідного рівня коефіцієнта K стиснення, та рівня Q якості реконструйованої відеоінформації.

Тут K визначається як [5]:

$$K = \frac{\Phi}{\Phi_{\text{comp}}}, \quad (1.1)$$

де Φ - вихідний рівень інформаційної інтенсивності відео;

Φ_{comp} - рівень інформаційної інтенсивності відео після виконання процедури стиснення (кодування).

У свою чергу, для оцінки якості Q реконструйованого відео контенту може застосовуватися досить широкий ряд показників, який розглянемо далі.

1.2 Міра якості реконструйованих відеоданих

У будь-якому випадку поняття якості відео вказує на ступінь відмінності між вихідним відеорядом та відео, що одержане за результатами кодування.

Так, у рамках покадрової оцінки якості відео, традиційними показниками, при цьому, можна вважати пікове відношення сигнал/шум (ПВСШ) та середньоквадратичне відхилення (СКВ). Тут величина ПВСШ визначається як:

$$P = 10 \log_{10} \frac{255^2 HW}{\sum_{x,y} (\eta(x,y) - \eta'(x,y))}, \quad (1.2)$$

де $\eta(x,y)$ та $\eta'(x,y)$ - піксель на позиції (x,y) початкового та відновленого відеокадру F роздільної здатності $H \times W$ відповідно.

У свою чергу, для оцінки рівня СКВ може бути використано наступний вираз:

$$M = \frac{1}{HW} \sqrt{\sum_{x=1,y=1}^{H,W} (\eta(x,y) \times \eta'(x,y))^2}. \quad (1.3)$$

Попри простоту реалізації кожного з показників для оцінки якості відео потоку, головним недоліком кожного з них є те, що для жодного з них не гарантується однозначна відповідність візуальній оцінці залежно від особливостей змісту кадру, який досліджується.

Інакше кажучи, метрики як СКВ, так і ПВСШ у ряді випадків можуть свідчити про повне руйнування кадру, хоча візуальних відмінностей при цьому порівняно з вихідним кадром може взагалі не бути. І навпаки.

Даного недоліку позбавлений ряд інших показників, серед яких, наприклад, вірність зображення або IFR (image fidelity rank), який визначається за виразом:

$$IFR = \frac{\sum_{x=1,y=1}^{H,W} (\eta(x,y) - \eta'(x,y))}{\sum_{x=1,y=1}^{H,W} (\eta(x,y))^2}. \quad (1.4)$$

При цьому, рівень Q якості відновленого відео незмінно пов'язаний з показником Φ його інформаційної інтенсивності прямо пропорційною залежністю, тобто:

$$Q \uparrow | \Phi \uparrow \quad (1.5)$$

1.3 Концепція забезпечення якості відеоінформації

У загальному випадку, формально, питання контролю рівня якості відео контенту, виходячи з виразу (2.4), можна сприймати як питання максимізації рівня Φ , що не є вірним, бо при цьому [6]:

- процедура кодування апріорі має забезпечити скорочення величини Φ ;

- той чи інший допустимий рівень Φ обмежений обсягами файлоосховища (якщо відео файл буде зберігатися локально), або рівнем $\xi(t)$ пропускної спроможності мережі, якщо відео контент генерується у ході відеоконференції чи якщо має місце потокова трансляція.

Далі питання контролю якості будемо розглядати з найгіршого випадку, тобто, коли відеопотік надсилається мережею.

Отже, враховуючи вищезазначене, можемо сформулювати попередні вимоги щодо забезпечення якості відновленого відео наступним чином:

$$Q \geq Q_{\text{set}} \mid \Phi \rightarrow \xi(t), \quad (1.6)$$

де Q_{set} - попередньо заданий рівень якості або відповідно до QoS, або узгоджений між провайдером сервісу та кінцевим користувачем.

Разом з тим, вираз (2.5) являє собою т.з. жорсткі вимоги, виконання яких не завжди може бути гарантовано. Тому має сенс розглянути альтернативні вимоги, що можуть бути продемонстровані наступним виразом:

$$Q \rightarrow Q'_{\text{max}} \mid \Phi \leq \xi(t), \quad (1.7)$$

де Q'_{max} - рівень якості, максимально можливий за поточних умов.

Таким чином, вимога (1.6) одностайно вказує на необхідність забезпечення рівня якості не нижче заданого. При цьому, згідно вимог (1.7) у процесі передавання кодованого відеопотоку має забезпечуватися стійка трансляція (на це вказує вимога $\Phi \leq \xi(t)$), та максимально можливий показник якості.

У загальному випадку вимоги (1.6) та (1.7) можуть застосовуватися спільно та послідовно – тобто, за неможливості забезпечення якості відео на рівні Q_{set} , далі пріоритетною стає вимога щодо максимально можливої якості за наявних умов.

При цьому, завдання забезпечення якості відео у процесі трансляції мережею ускладнюється наступними факторами:

1. Величина $\xi(t)$ пропускної спроможності є нестабільною у часі, тобто, $\xi(t-1) \neq \xi(t) \neq \dots \neq \xi(t+n)$.

2. Рівень інформаційної інтенсивності відео суттєвим чином визначається впливом:

- особливостей змісту кодованих кадрів;
- специфікою MPEG-обробки, яка передбачає формування потоку відео на базі кадрів трьох класів – I (ключові), P (передбачені) та B (двоспрямовано передбачені); для кадрів різних класів у загальному випадку справедливою є залежність $\Phi_I > \Phi_P > \Phi_B$.

Отже, як пропускна спроможність мережі, так і інформаційна інтенсивність відео є величинами, що змінюються у часі. До того ж, кожна з цих величин формується незалежно одна від одної.

Тому, по-перше, надалі величину інформаційної інтенсивності відео розглядатимемо у прив'язці до параметру часу, тобто, $\Phi(t)$.

По-друге, процес забезпечення вимог (1.6) та (1.7) слід розглядати як процес приведення величини $\Phi(t)$ у відповідність до $\xi(t)$ у динамічному режимі, керуючись при цьому вимогами до якості.

Таким чином, у процесі трансляції необхідно забезпечити оперативну зміну $\Phi(t)$ з огляду на $\xi(t)$ та необхідність забезпечення якості

1.4 Механізми реалізації концепції забезпечення якості відеоінформації

Визначимо спочатку, яким саме чином можливо впливати на зміну рівня $\Phi(t)$. Для цього попередньо слід розглянути, яким чином реалізується процес кодування потоку кадрів на рівні джерела та головні показники характеристики відео, що впливають на величину $\Phi(t)$.

У цілому, відеоряд може розглядатися як масив з ψ окремих кадрів F , що змінюються у часі зі швидкістю ω [6].

Кожен вихідний кадр при цьому характеризується роздільною здатністю $M \times N$ та глибиною d колірною опису одного пікселя, що зазвичай відповідає RGB-моделі та займає 24 біти.

Таким чином, наближено виконати оцінку кількості біт, необхідних для представлення вихідного кадру, можемо на базі виразу:

$$\Phi_F = M \times N \times d, \quad (1.8)$$

а кількість біт, які надсилатимуться за одиницю часу – наступним чином:

$$\Phi(t)_F = M \times N \times d \times \omega. \quad (1.9)$$

При цьому, на випадок кадру F' , утвореного після процедури кодування за MPEG-схемою, показник d не береться до уваги. Замість цього ураховується коефіцієнт скорочення бітового об'єму σ , який залежить від кодеку та режиму його роботи. Тоді для об'єму кодованого відео за одиницю часу маємо:

$$\Phi_F = \delta \times \sigma \times \omega, \quad (1.10)$$

де δ - параметр, який використовується для переведення роздільної здатності у мегапікселі, та визначається за виразом:

$$\delta = M \times N \times 10^{-6} \quad (1.11)$$

Тут σ варіюється у деякому діапазоні. Так, для кодеку H.264 умовно маємо $\sigma = \overline{0,03; 0,09}$, де найнижче значення відповідає випадку обробки відеоряду високої динаміки а найвище – низької.

З аналізу виразів (1.10) та (1.11) попередньо можемо зробити такі висновки:

1. На величину $\Phi(t)$ бітової інтенсивності у цілому можна здійснювати вплив маніпулюванням значеннями M та N , що є недоцільним, так як у цьому випадку якість відеоряду за будь-якою з метрик оцінювання забезпечено не буде.

2. На рівні потоку контроль значення $\Phi(t)$ може бути реалізовано за рахунок зміни частоти ω слідування кадрів.

3. На рівні окремого кадру варіювання значення $\Phi(t)$ може бути досягнуто шляхом змін параметру δ , для чого необхідно дослідити механізми кодеку, що впливають на його величину.

Таким чином, далі необхідно виконати аналіз схеми кодування на засадах MPEG.

1.4.1 Аналіз базисної моделі MPEG для виявлення механізмів впливу на показник скорочення бітового об'єму відео

У розрізі обробки потоку на рівні окремих кадрів процес MPEG-кодування передбачає послідовне виконання ряду технологічних етапів, як показано рис. 1.6.

У загальному випадку вони з деяким наближенням є аналогічними технології JPEG.

Отже, довільний кадр F , початково поданий у RGB-просторі, на початку каскаду перетворень конвертується до яскравісно-хроматичного (колірно-різницевого) опису, де замість базових колірних каналів утворюються масиви $M \times N$ яскравості Y , хроматичний синій C_b та хроматичний червоний C_r .

Далі кожен з масивів підлягає процедурі сегментації, у ході якої виконується їх поділ на фрагменти $\theta(\text{param})_{m,n}$ розміром 8×8 .

Тут $\text{param}=1$ для яскравісних фрагментів, 2 - хроматичних синіх а 3 - хроматичних червоних відповідно [4-7].

На рисунку (1.6) подальші перетворення розглядаються на прикладі яскравісного $\theta(1)_{m,n}$ фрагменту.

Етап сегментації, як і передуючий йому етап зміни колірної моделі не вносять жодних змін у величину $\Phi(t)$. Їх завданнями є виокремлення компонент яскравості та колірності, та зменшення навантаження на кодек – далі обробка виконується не кадру у цілому, кодується кожен фрагмент окремо.

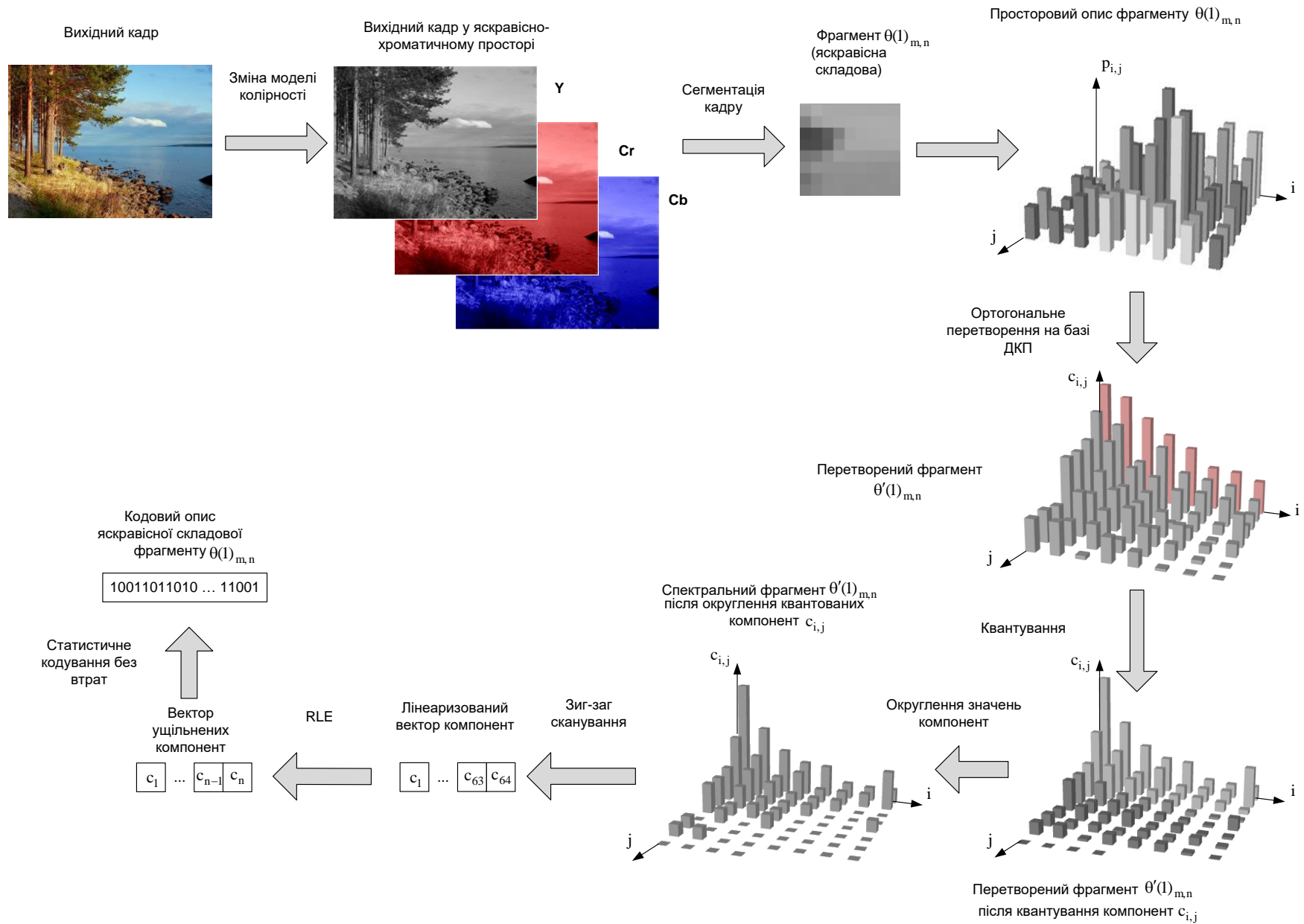


Рисунок 1.6 – Узагальнена схема обробки окремого кадру

У сутності, зазначені вище етапи є сервісним по відношенню до наступного технологічного етапу пертворень, а саме – проріджування масивів S_b та S_r .

Архітектуру даного етапу побудовано з тих міркувань, що зір людини майже не здатен відрізнити варіації відтінків колірності, проте, суттєвим чином сприймає градації яскравості [5, 6].

Таким чином, це дає можливість зменшити розмірність масивів S_b та S_r , проріджуючи їх з різними коефіцієнтами та у різних пропорціях, без внесення помітних викривлень. При цьому, якщо у вихідному вигляді, на початку даного етапу, 4 компонентам яскравості відповідали по 4 хроматичних (модель 4:4:4), надалі, залежно від режиму проріджування можуть утворюватися різні колірні моделі, що відповідають різній кількості хроматичних компонент при незмінній кількості яскравісних, а саме - 4:2:2, 4:2:1, 4:1:0, 4:1:1 та ін.

Отже, на цьому етапі кодування, по-перше, існує механізм зміни величини $\Phi(t)$ а по-друге, даний механізм – варіювання кроком та режимом проріджування масивів S_b та S_r , є керованим, тобто, його може бути використано для впливу на підсумкове значення інформаційної інтенсивності відео.

Далі фрагмент $\theta(1)_{m,n}$, представлений масивом 8×8 просторових компонент $p_{i,j}$ (що утворилися замість пікселів $\eta_{i,j}$), підлягає процедурі конвертації на базі дискретного косинусного перетворення, переходячи до спектрального формату представлення. Після цього перетворений $\theta'(1)_{m,n}$ фрагмент розглядається як матриця спектральних компонент $c_{i,j}$ такого ж розміру [5-8, 10].

На цьому етапі також відсутні механізми впливу на величину $\Phi(t)$ відео, так як його ключове завдання – сепарація компонент $c_{i,j}$ у різні частотні групи – ВЧ, СЧ та НЧ для подальшого виконання операції квантування.

Сама процедура квантування – це процес поелементного ділення матриці $\theta'(1)_{m,n}$ на матрицю \mathcal{Q} квантування.

При цьому, сенс виокремлення ВЧ, СЧ та НЧ компонент пояснює рис.1.7.

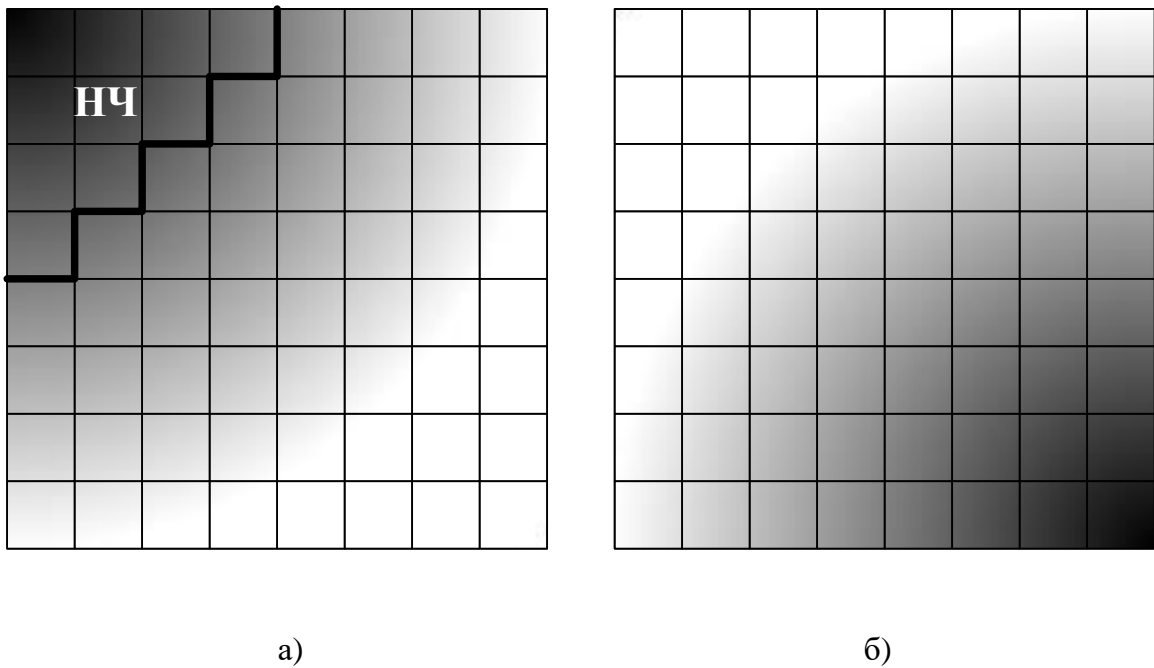


Рисунок 1.7 – Характер розподілу числових величин для перетвореного фрагменту а) та для типової матриці квантування б)

На рис. 1.7 градієнт зміни темного відтінку до світлого відповідає зменшенню числових величин, що знаходяться у відповідних чарунках матриці $\theta'(1)_{m,n}$ та матриці квантування.

У даному випадку найбільші числові значення на рис. 1.7 а) – зона низьких частот, які несуть у собі максимум потужності сигналу та є найбільш інформативні у межах фрагменту $\theta'(1)_{m,n}$. Відповідно, за напрямком руху до правого нижнього кута фрагменту величини компонент $c_{i,j}$ будуть зменшуватися.

Водночас, для матриці \mathcal{Q} , як видно з рис.1.7 б), існує зворотня залежність – тобто, коефіцієнти квантування є найбільшими для ВЧ-компонент (які при цьому найменш інформативні) і мінімальні – для НЧ-області фрагменту $\theta'(1)_{m,n}$. Тобто, компоненти $c_{i,j}$ зони нижніх частот пригнічуватимуться найменше.

Процедура квантування здійснює найбільший внесок у загальну величину δ , інакше кажучи, на цьому етапі можливо забезпечити найбільший вплив на рівень $\Phi(t)$.

При цьому, формуючи різні матриці \mathcal{Q} квантування, ступінь впливу на підсумкову величину $\Phi(t)$ можна змінювати [9-11].

Далі, у ході процедури округлення, прирівнюються до 0 усі компоненти $c_{i,j}$ (з них переважна більшість - ВЧ), величина яких після квантування є нижчою попередньо встановленою.

Після цього компоненти $c_{i,j}$ матриці $\theta'(1)_{m,n}$ лінеаризуються шляхом виконання зиг-заг обходу і далі – обробляються за алгоритмом RLE та кодуються без втрат на базі арифметичного алгоритму чи за Хафманом.

Жоден з перелічених тут етапів перетворень на дозволяє вносити контрольованих змін у підсумкову величину інформаційної інтенсивності.

1.5 Попередні висновки

Розглянувши процес кодування кадру відео на загальних принципах MPEG виявлено, що:

- конвеєр MPEG перетворень містить у собі технологічні етапи, які вносять зміни у підсумкову величину інформаційної інтенсивності відео і етапи, які є сервісними по відношенню до зазначених;

- серед технологічних етапів, які вносять зміни у рівень інформаційної інтенсивності, є такі, що можуть налаштовуватися на неконтрольовані;

- до технологічних етапів, які можливо налаштовувати у процесі кодування можна віднести квантування та проріджування матриць хроматичних складових (на рівні) кадру і окремо – зміна показника частоти надходження кадрів (на рівні потоку).

Разом з тим, беручи до уваги те, що як інформаційна інтенсивність відео, так і пропускна здатність мережі є динамічними величинами, далі необхідно:

- дослідити, яким чином може бути реалізовано механізми контролю інформаційної інтенсивності та оцінки рівня дисбалансу даної величини відносно пропускної спроможності;

- розглянути підходи інтеграції механізмів контролю в існуючі системи передачі.

2 ПРИНЦИП КОНТРОЛЮ ІНФОРМАЦІЙНОЇ ІНТЕНСИВНОСТІ ПОТОКУ КОДОВАНИХ ВІДЕОКАДРІВ У РАМКАХ ЧИННИХ СИСТЕМ ПЕРЕДАЧІ

2.1 Обґрунтування неефективності застосування методів кодування відео з фіксованим коефіцієнтом стиснення

Раніше було зазначено, що інформаційна інтенсивність $\Phi(t)$ відео не є сталою величиною. Водночас, значення $\xi(t)$ пропускної здатності мережі також змінюється у часі.

При цьому, для $\Phi(t)$ справедлива наступна залежність [10]:

$$\Phi(t) = f(\Theta; \zeta_1; \zeta_2), \quad (2.1)$$

де Θ - множина особливостей змісту кодованого відеоряду;

ζ_1 - сукупність технологічних концепцій, застосованих у рамках базового відеокодеку;

ζ_2 - множина налаштувань режимів функціонування базового відеокодека.

У свою чергу, величина $\xi(t)$ пропускної спроможності мережі у довільний момент часу t у загальному випадку визначається виходячи з наступної функціональної залежності:

$$\xi(t) = f(\chi_1; \chi_2; \chi_3; \chi_4), \quad (2.2)$$

де χ_1 - особливості базової технології для побудови мережі;

χ_2 - продуктивність мережевого обладнання;

χ_3 - топологічні особливості мережі;

χ_4 - рівень завантаженості;

χ_5 - ступінь складності трафіку, що надсилається мережею.

З аналізу виразів (2.1) та (2.2) бачимо, що:

- величини $\xi(t)$ та $\Phi(t)$ є незалежними між собою;

- стиснення відео з фіксованим коефіцієнтом у даних умовах є неефективним.

Все зазначене підтверджує висновок про необхідність узгодження величин $\xi(t)$ та $\Phi(t)$.

Отже, значення коефіцієнту стиснення необхідно обирати у динамічному режимі виходячи з величин $\xi(t)$ та чинних вимог до якості.

У зв'язку з цим попередньо розглянемо існуючий взаємозв'язок технології кодування з алгоритмами контролю інформаційної інтенсивності.

2.2 Взаємозв'язок основної технології стиснення відеоінформації з алгоритмами контролю інформаційної інтенсивності

У загальному випадку алгоритми контролю рівня інформаційної інтенсивності потоку кодованих відеокадрів, концептуально являють собою зовнішній модуль щодо базисної технології стиснення відео. Такий взаємозв'язок схематично може бути проілюстровано рис.2.1.

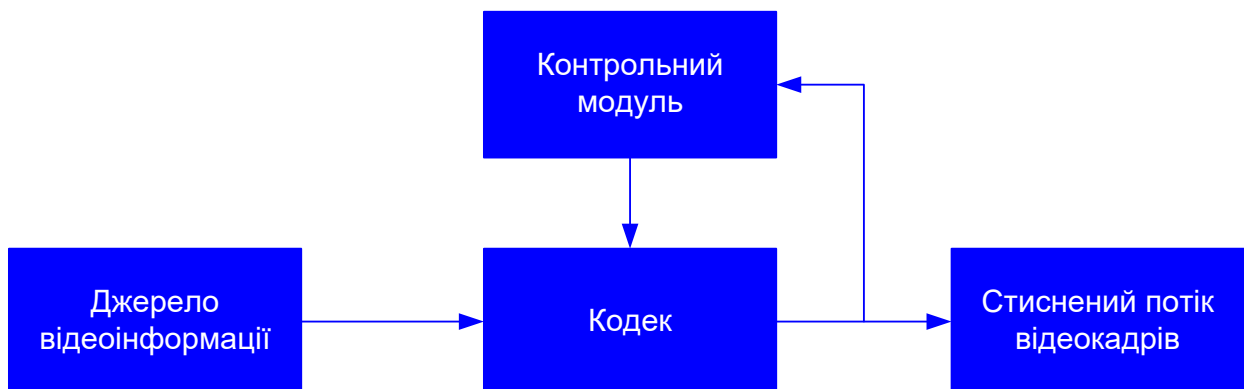


Рисунок 2.1 – Архітектура взаємозв'язку алгоритмів контролю інформаційної інтенсивності відеоінформації та базисної технології стиснення відеоінформації

Відповідно до схеми 4.1, контрольний модуль має одержувати відомості щодо величин пропускнуєї спроможності $\xi(t)$, а також Φ для кожного відліку часу t .

У свою чергу, з урахуванням значень $\xi(t)$ та Φ , контрольний модуль далі вносить зміни у налаштування кодеку таким чином, щоб теоретично мати змогу гарантувати виконання умов ефективної передачі відеоконтенту, у відповідності до формул (1.6) та (1.7).

Разом з тим, спільне використання алгоритмів контролю інформаційної інтенсивності відеоінформації з відеокодеком та, відповідно, системою передачі даних, потребує спочатку рішення завдання оцінки поточного рівня $\xi(t)$ пропускної спроможності мережі. Таким чином, у зворотній канал, що поєднує вихід кодеку, та контрольний модуль, (рис.2.1), має бути включено модуль розрахунку розбіжності (MPP) інформаційної інтенсивності потоку відео та поточного значення $\xi(t)$.

Це, у свою чергу, може бути реалізовано на базі таких підходів, як:

- непрямий розрахунок актуальної величини $\xi(t)$ у момент t , беручи до уваги значення RTT;
- оцінка закономірностей зміни величини $\xi(t)$ пропускної спроможності за рівнем завантаженості буферу передавача.

Виконаємо огляд кожного з підходів окремо.

2.3 Непряма оцінка поточного рівня пропускної спроможності з використанням величини RTT

Розглядуваний підхід передбачає, що величина $\xi(t)$ з деяким наближенням може бути одержана у процесі обміну передавача та приймача тестовими пакетами відомої довжини.

Так, для цього джерело даних спочатку генерує тестовий пакет довжиною Φ_p біт. Після генерування, відповідно, такий пакет надсилаються розподіленим середовищем приймачеві.

Далі, після того, як пакет-запит буде одержано приймачем, у відповідь формується підтверджуючий пакет. При цьому, його розмір дорівнює розміру отриманого пакета-запиту [12].

У свою чергу, сумарна тривалість τ_{trn} процесу надсилання пакету-запиту разом з часом τ_{rec} надходження підтверджуючого пакету, являє собою т.з. час RTT (round trip time), або, іншими словами, час двосторонньої затримки:

$$\text{RTT} \approx \tau_{\text{trn}} + \tau_{\text{rec}}. \quad (2.3)$$

У загальному випадку припускається, що величина RTT наближено буде дорівнювати $RTT \approx 2\tau_{tm}$, де τ_{tm} , відповідно тривалість процесу надходження пакету-запиту до приймача.

Також ураховуючи той факт, що довжина Φ_p пакету-запиту від початку є відомою, це дозволяє виконати розрахунок діючого рівня $\xi(t)$ пропускної спроможності мережі для довільного часу t . Для цього використовується наступний вираз:

$$\xi(t) = \frac{\Phi_p}{RTT} = \frac{\Phi_p}{2\tau_{tm}} \quad (2.4)$$

До переваг розглянутого підходу, безумовно, слід віднести простоту побудови алгоритму та його низьку ресурсоемність.

Проте, застосовуваність цього підходу обмежується рядом недоліків, які йому притаманні, а саме [12, 13]:

1. Часткове ускладнення структури трафіку.

При цьому, за кожну елементарну одиницю часу (умовно - t) структура потоку відеоінформації, що надсилається приймачеві. Буде такою, як зазначає рис.4.2.

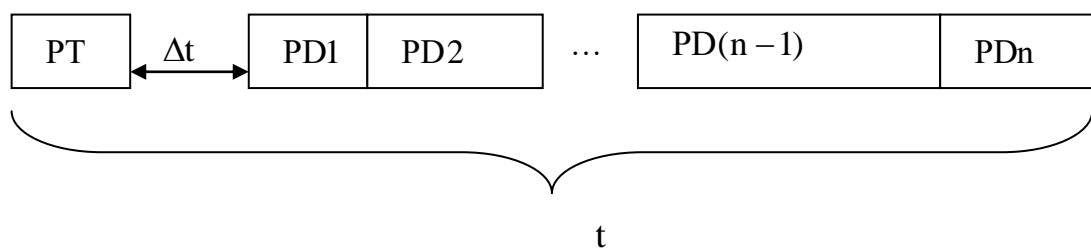


Рисунок 2.2 – Структура вихідного потоку даних за умов оцінки рівня пропускної спроможності мережі з використанням тестових пакетів

Так, на рис. 2.2 PT – сам тестовий пакет. Після його надсилання приймачеві виконуються такі технологічні операції, як:

- отримання пакету-підтвердження;
- обчислення діючого рівня $\xi(t)$;
- формування множини опцій для кодеку, які ймовірно можуть забезпечити виконання вимог (1.6) та (1.7);

- формування фрагменту потоку відео на базі знайденої множини опцій кодеку (пакети PD1 - PDn).

Відповідно, між умовним часом t надсилання тестового пакету та надсиланням пакетів даних буде мати зміщення Δt , що, у свою чергу буде дорівнювати:

$$\Delta t = RTT + \tau_{clc} + \tau_{of} + \tau_{oa}, \quad (2.5)$$

де τ_{clc} - час розрахунку рівня пропускнуої спроможності з використанням RTT;

τ_{of} - час формування множини потенційно прийнятних опцій для кодеку;

τ_{oa} - час формування фрагменту потоку на базі обраних попередньо опцій.

Отже, залежно від конкретних умов, даний спосіб оцінки величини $\xi(t)$ може бути неприйнятним, так як:

- час зміщення Δt передачі може суттєво впливати на загальну затримку надсилання пакетів приймачеві;

- теоретично у ряді випадків величина RT може суттєво перевищувати усю множину PD.

2. Обчислення діючого рівня $\xi(t)$ у будь-якому випадку виконуватиметься з тією чи іншою погрішністю. Така закономірність, у свою чергу, є наслідком того, що, у загальному випадку, має місце справедливість наступного виразу:

$$\tau_{trn} \neq \tau_{rec}. \quad (2.6)$$

Звідси виходить, що час, за який тестовий пакет надійде приймачеві, та час, за який до передавача буде доставлено пакет-підтвердження, у загальному випадку не є однаковими, як наслідок впливу ряду чинників, серед яких ключовими є наступні:

- висока ймовірність того що тестовий пакет до приймача та пакет-відповідь надсилатимуться різними шляхами, (рис. 2.3) що, у свою чергу, є невідомими;

- у виразі (4.1) вказується, що величина RTT є лише наближено рівною сумі τ_{trn} та τ_{rec} ; це пояснюється тим, що навіть коли $\tau_{trn} = \tau_{rec}$, а отже - вираз (2.6) не буде актуальним, справедливою є рівність:

$$\tau_{trn} + \tau_{rec} = RTT - \tau_{hnd}, \quad (2.7)$$

де τ_{hnd} - час обробки запиту на віддаленому вузлі-приймачеві, що при цьому розраховується на базі виразу:

$$\tau_{hnd} = \tau_{ar} + \tau_g + \tau_{bt}, \quad (2.8)$$

де τ_{ar} - час, що витрачається приймачем на прийом та наступний аналізу одержаного пакету;

τ_g - час, необхідний для формування пакету-підтвердження;

τ_{bt} - час, за який сформований пакет-відповідь буде розміщено у вихідний буфер передаючого пристрою та далі відправлено.

При цьому очевидно, що ніякий зі складників, перелічених у виразі (2.6), не є фіксованою величиною, та визначається впливом таких факторів, як:

- ступінь заповнення вихідного буферу прийомопередавача у момент t ;
- загального рівня навантаження на вузол, беручи до уваги зовнішні та внутрішні джерела впливу;
- кількості доступних обчислювальних ресурсів у поточний момент часу.

Варто зазначити також, що як τ_{trn} так і τ_{rec} , у свою чергу, суттєвим чином є залежними від величини часу τ_{line} їх транслявання мережевим середовищем, та від часу τ_{net} обробки пакету на транзитних мережевих пристроях, інакше кажучи [13, 14]:

$$\tau_{trn}, \tau_{rec} = \tau_{line} + \tau_{net}. \quad (2.9)$$

Водночас, якщо мова йде про магістральні мережі, чи мережі доступу, побудовані на базі високошвидкісних технологій (наприклад, WiMAX чи

ІМТ2020), тоді значення часу τ_{line} з великою ймовірністю на тлі величини τ_{net} буде несуттєвим.

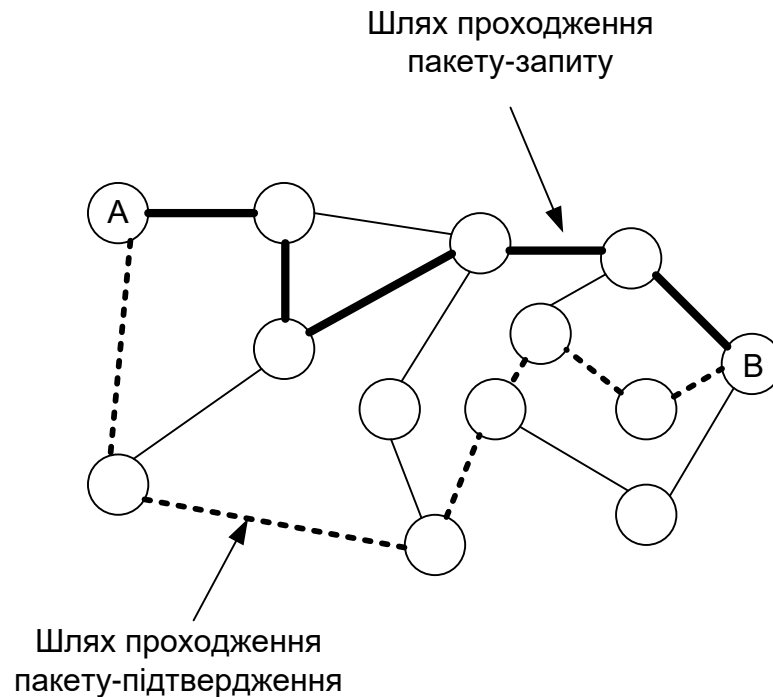


Рисунок 2.3 – Приклад надходження тестового пакету та пакету-відповіді між вузлами A та B різними шляхами

При цьому, очевидно, що за таких умов складником τ_{line} з виразу (2.9) можемо знехтувати. Відтак, (4.7) може бути переписано у такому вигляді:

$$\tau_{\text{net}} = \sum_{i=1}^n \varphi_{\text{hn}}(\alpha_q^{(i)}; \alpha_r^{(i)}; \alpha_{\text{osi}}^{(i)}; \gamma) \times \gamma_{\text{nold}}, \quad (2.10)$$

де $\varphi_{\text{net}}^{(i)}$ - функціонал, який встановлює порядок та закономірність обробки пакетів у мережевому пристрої з використанням існуючих налаштувань обробки;

$\alpha_q^{(i)}$ - доступна множина технологічних концепцій, на базі яких реалізується опрацювання черг вхідних пакетів для будь-якого i -го мережевого вузла;

$\alpha_r^{(i)}$ - множина алгоритмів маршрутизації, разом з наборами налаштувань для кожного з них, що можуть застосуватися на рівні i -го вузла для маршрутизації пакетів;

$\alpha_{osi}^{(i)}$ - множина перетворень у межах OSI, які передбачено для обробки пакетів на рівні окремого i -го вузла;

γ_{nold} - ступінь навантаження, яке чинить вплив на i -й мережевий вузол; дана величина, при цьому, може бути розглянутою як деяка функціональна залежність від кількості пакетів ρ_j , що надходять до вузла, а також показника Θ структурованості (складності) трафіку у мережі, отже виходить:

$$\gamma_{nold} = f\left(\sum_{j=1}^J \rho_j; \Theta\right). \quad (2.11)$$

Разом з тим, не може бути відомо заздалегідь, які саме алгоритми $\alpha_r^{(i)}$, що застосовуються для налагодження маршрутизації та керування чергами $\alpha_q^{(i)}$, застосовуватимуться на тому чи іншому вузлі під час аналізу та подальшого опрацювання тиз чи інших пакетів. У такій же мірі це справедливо щодо сукупності $\alpha_{osi}^{(i)}$ перетворень, що виконуються відносно пакетів на вузлах мережі, та наявної специфіки кожного перетворення за умови впливу навантаження γ_{nold} на мережевий вузол [12].

Отже, виходить, що час τ_{net} , необхідний для успішної обробки пакету на транзитних мережевих пристроях, є невідомою величиною. Відтак, характер впливу якої на загальну систему буде не прогнозованим та випадковим. Таким чином, у цих умовах справедливоє є залежність:

$$(\tau - 1)_{net} \neq \tau_{net} \neq (\tau + 1)_{net}. \quad (2.12)$$

3. Так як пропускна спроможність мережі не є величиною, стабільною у часі, а отже - $(t - 1) \neq t \neq (t + 1) \dots \neq (t + m)$, виходить, що значення $\xi(t)$ слід фіксувати для кожного елементарного фрагменту часу, за який у теорії може виконуватися надсилання пакетів кодованого відео.

Інакше говорячи, трансляцію тестових пакетів слід виконувати протягом кожного елементарного часового відрізка t у діапазоні $(t+n; t+n+1)$.

У решті решт, діаграма надсилання відеоданих на рівні передавача у загальному випадку матиме такий вигляд, як зазначено на рис. 2.4.

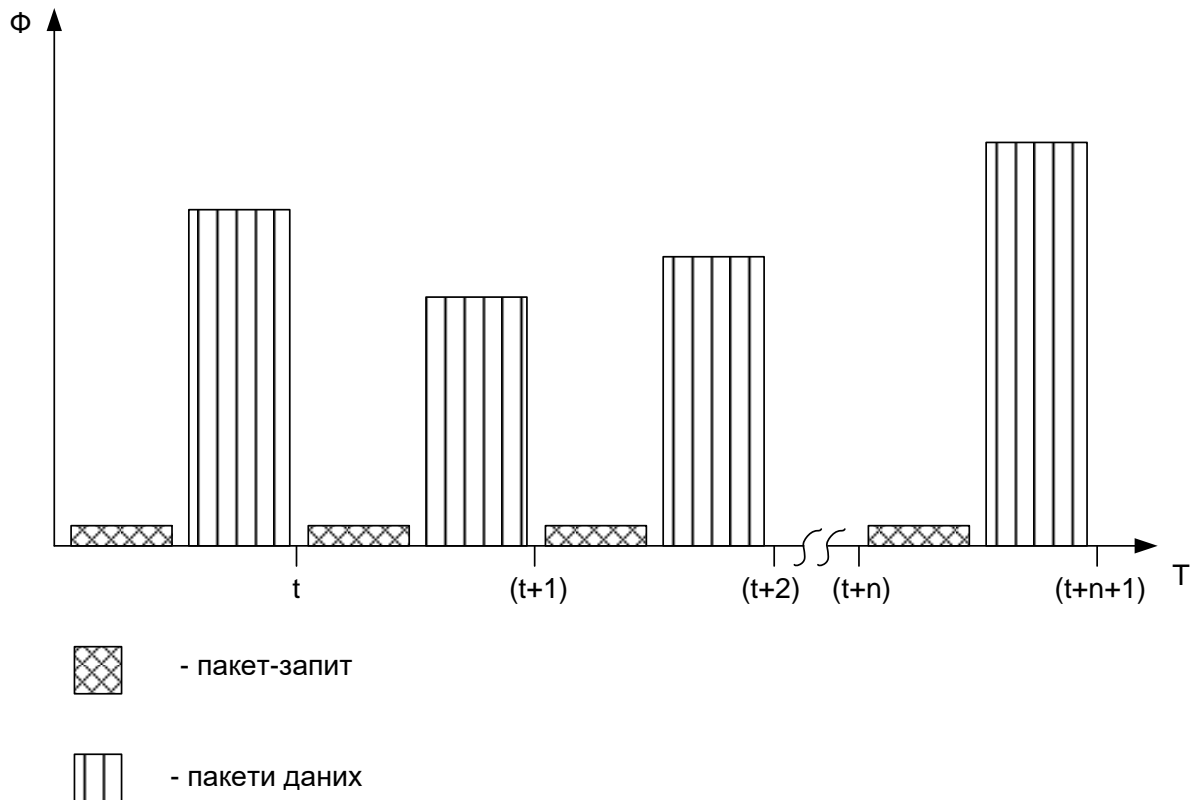


Рисунок 2.4 – Приклад діаграми надсилання пакетів приймачеві у ході розрахунку пропускної спроможності з використанням відомостей щодо RTT

У свою чергу, як на випадок одного часового інтервалу (рис.2.4), так і для потоку кодованих кадрів у цілому, це, по-перше, спричинює додаткове ускладнення структури даних, а по-друге - вимагає збільшення загальної швидкодії апаратної складової системи.

У цілому, компромісним рішенням щодо зменшення навантаження та обмеженого ускладнення структури даних може бути збільшення тестового інтервалу τ_{test} . При цьому, розширення значення τ_{test} до $\tau_{\text{test}} > |(t+n+1) - (t+n)|$, з іншого боку, веде до втрати достовірності розрахунку значення діючого рівня $\xi(t)$ пропускної спроможності мережі.

У підсумку, тут значення τ_{test} повинно бути рівним $|(t+n+1)-(t+n)|$. Іншими словами, частота ω слідування тестових пакетів повинна дорівнювати 1 на відрізок часу $(t+n; t+n+1)$.

4. Має гарантуватися забезпечення таких умов, у яких підсумковий час, необхідний для надсилання тестового пакету приймачеві, на прийом пакету-підтвердження та далі на обробку певної кількості відеоконтенту, який необхідно надіслати мережею протягом одиниці часу, у загальному випадку був не більше, ніж $|(t-1)-t|$:

$$|(\tau-1)-\tau| \leq \tau_{\text{tm}} + \tau_{\text{rec}} + \tau_{\text{hnd}}. \quad (2.13)$$

Приймаючи до уваги ряд недоліків, розглянутих вище, можемо стверджувати, що розрахунок пропускної спроможності мережі на базі вимірювання часу двосторонньої затримки є нераціональною та не доцільною для застосування.

Тут важливим також є те, що характер помилки, що виникає у процесі обчислення рівня $\xi(t)$, та її величина, фактично неможливо урахувати. Відтак, досліджений підхід апріорі забезпечує лише наближену оцінку $\xi(t)$.

Далі пропонується розглянути алгоритм, що базується не на вимірюванні фактичної величини $\xi(t)$, а є орієнтованим на виявлення змін її значення у часі відносно поточного моменту t .

2.4 Оцінка характеру зміни поточного рівня пропускної здатності мережі відносно контрольного моменту часу

Увесь цикл технологічних операцій з перетворення відеоданих на рівні самого джерела умовно зводиться до [8-14]:

- формування вихідної інформації;
- виконання процедури кодування попередньо утворених даних;
- перетворення кодованих даних до формату базової технології передавання (процедура пакетування);
- надсилання деякого числа сформованих пакетів даних у буфер передаючого пристрою;
- надсилання пакетів буферу передавача безпосередньо до мережі.

Однією з ключових характеристик буферу є його номінальний об'єм V_{buf} .

При цьому, протягом будь-якого довільного (кожного) відрізка $(t + n; t + n + 1)$ у ході взаємодії кодер-буфер виконується:

- прийом даних від кодеру;
- надсилання даних до мережі.

Водночас, за зазначений проміжок $(t + n; t + n + 1)$ часу кодер може надіслати у буфер $\Phi \leq V_{\text{buf}}$ біт попередньо кодованої відеоінформації. Інакше кажучи, у будь-якому разі у буфер може бути розміщено не більш, ніж V_{buf} біт. Після цього пакети, що містять у собі деякий об'єм кодованої відеопослідовності, що займає Φ біт, спрямовуються у мережевий канал.

У даному випадку, так як розглядається процес кодування потоку відеокадрів у парадигмі MPEG, мають ураховуватися її ключові особливості. Наприклад, те, що у будь-якому випадку кодер формує навантаження на буфер, яке має нерівномірний у часі характер, а отже:

$$\Phi(t) \neq \Phi(t+1) \neq \dots \Phi(t+n) \neq \Phi(t+n+1). \quad (2.14)$$

Закономірність, показана виразом (2.14), є, у свою чергу, наслідком існування ряду факторів, серед яких наступні:

- кадри, що формують відео потік, потребують для їх опису різну кількість біт, що викликано різними особливостями змісту кожного з них;
- структура потоку MPEG передбачає обов'язкову наявність кадрів I, P та B-типів, при цьому кожному з типів у середньому відповідає різний обсяг біт.

У будь-якому випадку є очевидним той факт, що за будь-яких обставин протягом якого завгодно часового відрізка $(t + n; t + n + 1)$ до мережі буфер може спрямувати не більше, ніж $\Phi = V_{\text{buf}}$ біт.

Далі розглянемо випадок, коли для довільного інтервалу $(t + n; t + n + 1)$ часу складуться умови, за яких значення $\xi(t)$ пропускну здатності мережі опуститься до величини, що є меншою, ніж V_{buf} .

Відтак, у цьому разі буфер зможе спрямувати до мережі кількість біт, що не перевищуватиме значення $\xi(t)$. Водночас, певну кількість $\Delta\Phi$ біт, яка визначається як $\Delta\Phi = V_{\text{buf}} - \xi(t)$, тобто, різницю між кількістю біт у буфері

та пропускною спроможністю, до мережі не буде надіслано. Далі виконаємо огляд зазначено процесу більш детально.

Припустимо, що у будь-який довільний часовий інтервал $(t + n; t + n + 1)$ у вихідному буфері знаходиться $\Phi'((t + n); (t + n + 1))$ біт. При цьому, конкретна кількість даних у буфері, буде першочергово залежати від ряду факторів, головними з яких є:

- діючий рівень $\xi((t + n); (t + n + 1))$ пропускної здатності мережі в інтервал часу $(t + n; t + n + 1)$;

- інтенсивністю Φ_{load} надходження біт з виходу кодера до буфера.

Отже, сумарну кількість $\Phi'((t + n); (t + n + 1))$ біт у буфері, яка відповідає часовому інтервалу $(t + n; t + n + 1)$, може бути розраховано за наступною формулою:

$$\Phi'((t + n); (t + n + 1)) = \Phi_{load}((t + n); (t + n + 1)) - \xi((t + n); (t + n + 1)), \quad (2.15)$$

де $\Phi_{load}((t + n); (t + n + 1))$ - інтенсивність надходження даних від кодера протягом інтервалу часу $(t + n; t + n + 1)$.

У свою чергу, зрозуміло, що при цьому обчислення значення величини $\Phi_{load}((t + n); (t + n + 1))$ з виразу (4.14) може бути виконано за наведеним далі виразом:

$$\Phi_{load}((t + n); (t + n + 1)) = \Phi'((t + n - 1); (t + n)) + \Phi((t + n); (t + n + 1)), \quad (2.16)$$

де $\Phi'((t + n - 1); (t + n))$ - підсумкова кількість біт, локалізованих у межах буферу передавача за попередній інтервал часу - $((t + n - 1); (t + n))$;

$\Phi((t + n); (t + n + 1))$ - обсяг біт, що надсилаються кодером у буфер за інтервал $((t + n); (t + n + 1))$.

Отже, виходячи з вищезазначеного, вираз (2.14), на базі якого може бути визначено сумарне число біт у буфері передавача, має сенс привести до наступного вигляду:

$$\begin{aligned} \Phi'((t + n); (t + n + 1)) = \Phi'((t + n - 1); (t + n)) + \Phi((t + n); (t + n + 1)) - \\ - \xi((t + n); (t + n + 1)). \end{aligned} \quad (2.17)$$

Далі, беручи за основу рівність (2.17), діючи величину пропускнуої спроможності для довільного інтервалу $((t + n); (t + n + 1))$ можемо обчислити наступним чином:

$$\xi((\tau + n); (\tau + n + 1)) = \Phi'((\tau + n - 1); (\tau + n)) + \Phi((\tau + n); (\tau + n + 1)) - \Phi'((\tau + n); (\tau + n + 1)). \quad (2.18)$$

Водночас, згадаємо про те, що достатньою умовою поступового заповнення буферу є справедливість нерівності:

$$\Phi'((t + n - 1); (t + n)) > \xi((t + n); (t + n + 1)). \quad (2.19)$$

Також може мати місце переповнення буферу. Воно може виникати тоді, коли на протязі того чи іншого часового відрізка за кожен окремий часовий інтервал кодер надсилатиме у буфер $\Phi((t + n); (t + n + 1))$ біт. При цьому дана величина перевищуватиме різницю величин $\Phi_{load}((t + n); (t + n + 1))$ та $\xi((t + n); (t + n + 1))$. Відтак, формальний опис умов переповнення буферу буде наступним:

$$\Phi((t + n); (t + n + 1)) \geq \Phi_{load}((t + n); (t + n + 1)) - \xi((t + n); (t + n + 1)). \quad (2.20)$$

У свою чергу, в умовах переповнення буферу передавача реалізація процесу надсилання кодованого потоку відеокадрів є неможливою.

Це зумовлено чинними механізмами розвантаження буферу. А саме, при виконанні умови (2.20) зміст буферу підлягає обнулінню, відповідно, поміщені до нього дані буде втрачено.

Виходячи з одержаних взаємозв'язків між величинами заповнення буферу, пропускнуою спроможністю мережі та інтенсивністю надходження даних від кодеку, доцільним для використання є підхід на базі буферу обмеженого робочого об'єму.

2.5 Використання буферу обмеженого робочого об'єму у процесі контролю рівня інформаційної інтенсивності та якості відеоінформаційного потоку

Нехай номінальний об'єм буферу дорівнює V_{buf} . У рамках зазначеного підходу умовимося, що робочий об'єм буферу рівнятиметься [12-15]:

$$V_{\text{buf}}^{(\text{wrk})} = \delta V_{\text{buf}}, \quad \delta \in [0,5; 0,7]. \quad (2.21)$$

Схему такого розподілу об'єму буферу показано на рис. 2.7. Сенс даного обмеження полягає у тому, що на випадок виникнення заповнення буферу вище рівня $V_{\text{buf}}^{(\text{wrk})}$, кодек отримає команду про необхідність зменшення інформаційної інтенсивності потоку кодованих відеоданих. Для цього буде змінено перелік опцій кодування.

При цьому, різниця між V_{buf} та $V_{\text{buf}}^{(\text{wrk})}$ забезпечує деякий резервний час Δt , протягом якого система намагатиметься знайти налаштування кодеку, що, по-перше, створять умови для розвантаження буферу (тобто, рівень Φ буде нижчим, ніж діюча величина пропускнуої здатності), а по-друге - теоретично можуть задовольнити виконання вимог (1.6) та (1.7).

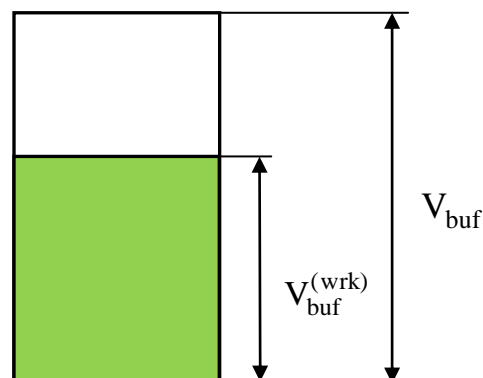


Рисунок 2.5 – Приклад лімітування робочого об'єму буферу передавача для резервування часу реакції

Таким чином, якщо виконуватиметься умова (2.19), це може викликати заповнення буферу до рівня:

$$V_{\text{buf}}^{(\text{wrk})} \geq \delta V_{\text{buf}} . \quad (2.22)$$

Відтак, вираз (2.22) є, у сутності, умовою для застосування механізму зміни інформаційної інтенсивності відео шляхом маніпулювання набором налаштувань кодеку.

3. МЕХАНІЗМИ ЗМІНИ ІНФОРМАЦІЙНОЇ ІНТЕНСИВНОСТІ КОДОВАНОГО ВІДЕОІНФОРМАЦІЙНОГО ПОТОКУ З УРАХУВАННЯМ ВИМОГ ДО ЯКОСТІ РЕКОНСТРУЙОВАНИХ КАДРІВ

3.1 Існуючі парадигми маніпулювання інформаційною інтенсивністю відео потоку

На сьогодні серед стандартизованих напрямків до реалізації контролю інтенсивності кодового відео можна виділити ряд ключових, а саме [7, 12, 14, 15]:

- зміна роздільної здатності потоку кадрів у реальному часі;
- зміна параметрів кодування відео у часовій та/або просторовій площинах;
- зміна кількості структурних одиниць для опису відеокадрів чи їх окремих фрагментів [16].

Розглянемо окремо кожен напрямок.

3.1.1 Контроль рівня інформаційної інтенсивності відео шляхом зміни роздільної здатності кадрів на прикладі технології ABR

Технологія адаптивної бітової швидкості (Adaptive Bit Rate) розроблена для використання у сервісах потокового відеомовлення – наприклад, у системах VoD та для численних відеохостингів, та використовується на базі протоколу HTTPS

У рамках цього передбачається, що:

1. Користувач (адміністратор) розміщує кодований відео файл на одному з ресурсів, що підтримують ABR.
2. Внутрішній обробник на базі первинного відео файлу з роздільною здатністю $M \times N$ та середньою інтенсивністю $\bar{\Phi}$ генерує деяку кількість Γ файлів зменшеної роздільної здатності. У рамках ABR кожен таких файл має назву прошарку (layer). Відтак, кожному такому γ -му прошарку відповідатиме деяка середня інформаційна інтенсивність $\bar{\Phi}_\gamma$. При цьому, виконується така залежність:

$$\bar{\Phi} > \bar{\Phi}_1 > \bar{\Phi}_2 \dots > \bar{\Phi}_\gamma \dots > \bar{\Phi}_\Gamma \quad (3.1)$$

3. Кожен з утворених таким чином окремих Γ прошарків умовно поділяється на T часових фрагментів τ_i , тривалістю 1-2 секунди кожен (часова мітка).

Результуюча інформаційна структура, що відповідає початковому файлу на відео сервері, буде такою, як зображено рис. 3.1 та рис.3.2.

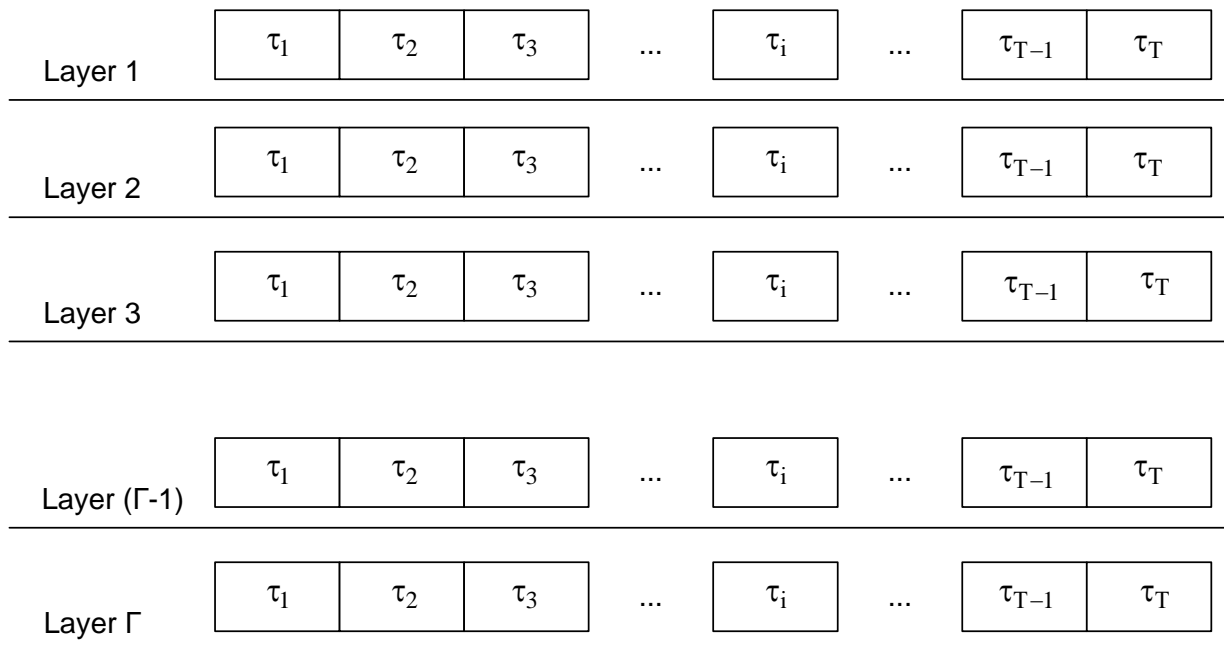


Рисунок 3.1 – Інформаційна структура відео файлу у рамках ABR

У випадку звернення користувача до того чи іншого файлу на відео сервері, у момент часу ($t=0$), зазвичай, користувачеві початково транлюється або потік Layer1, або той прошарок, який було задіяно у попередньому сеансі.

У процесі потокової трансляції відео як $\Phi \neq \text{const}$ так і $\xi(t) \neq \text{const}$. Таким чином, можуть скластися умови для виникнення ситуації, коли $\Phi \geq \xi(t)$. Відповідно, трансляція відеоконтенту за даного випадку не може виконуватися.

У даному разі, якщо умови $\Phi \geq \xi(t)$ мали місце під час трансляції потоку з Layer γ , виконується синхронний перехід до Layer ($\gamma+1$) за принципом:

$$\Phi(F(\text{Layer } \gamma, \tau_i)) \geq \xi(t) \rightarrow F := F(\text{Layer } (\gamma + 1), \tau_{i+1}), \quad (3.2)$$

тобто, якщо потоку кадрів F відповідає інтенсивність, яка перевищує величину пропускної здатності у момент часу τ , для наступного часового відрізка, $(\tau+1)$, обирається для трансляції часовий сегмент з прошарку $\text{Layer } (\gamma+1)$, тобто, такий, якому відповідає апіорі нижчий рівень інформаційної інтенсивності.

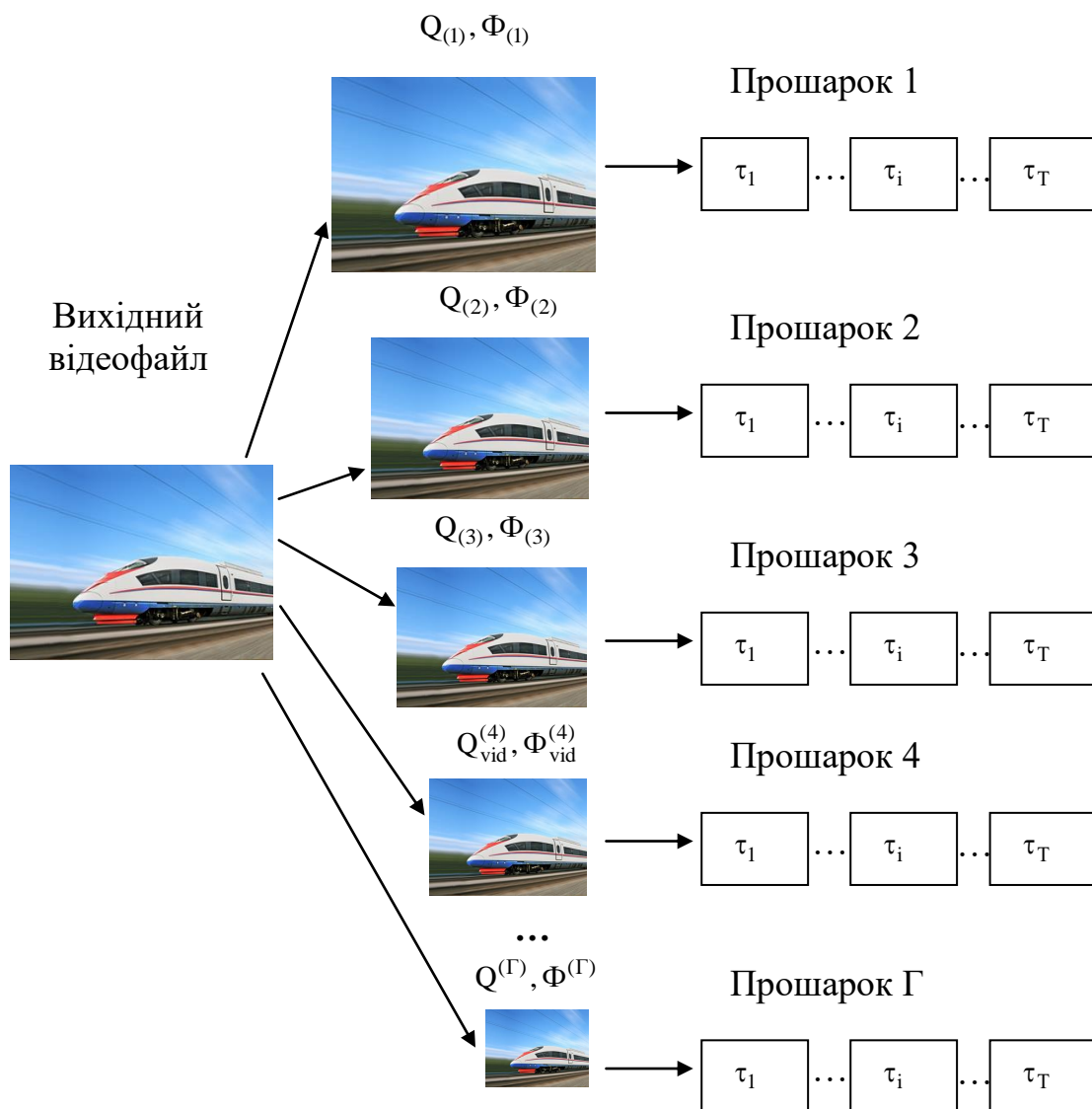


Рисунок 3.2 – Приклади прошарків відеоданих інформаційної структури на базі ABR, та рівні якості та інформаційної інтенсивності, що відповідають кожному з них виходячи з роздільної здатності відео

Далі, якщо перехід до прошарку $\text{Layer } (\gamma+1)$ все ще супроводжується перевищенням інформаційної інтенсивності рівня пропускної здатності, для моменту часу $(\tau+1)$ обирається для трансляції прошарок $\text{Layer } (\gamma+2)$ з мітки

$(\tau+2)$ і т.д. до того моменту, поки буде забезпечено виконання нерівності $\Phi < \xi(t)$.

Після цього, у момент $(\gamma+2)$ робиться спроба переходу до прошарку Layer $(\gamma+1)$. Якщо така спроба є успішною, далі виконується спроба переходу до наступного прошарку і т.д.

До очевидних переваг технології, при цьому можемо віднести:

- простоту алгоритму;
- гарантування сталої трансляції відео інформаційного потоку.

При цьому, недоліками ABR є:

- відсутність будь-яких механізмів, спрямованих на збільшення якості реконструйованого відео контенту;
- повна неможливість реалізації ABR-підходів для випадків трансляції інтерактивного відео контенту;
- відсутність стандартизованої сітки роздільних здатностей для формування множини прошарків роздільних здатностей;
- чим більшою буде сітка роздільних здатностей, тим більшим буде необхідний обсяг дискового простору для її зберігання.

Таким чином, урахувавши виявлені недоліки, головним чином – відсутність механізмів підтримки якості реконструйованого відео, можемо зробити висновок про те, що технологія ABR не може розглядатися у якості уніфікованого інструментарію контролю інформаційної інтенсивності відео.

Далі розглянемо напрямок контролю інформаційної інтенсивності відео та рівня якості, який базується на маніпуляціях відносно кількості структурних одиниць, що використовуються для опису відеокадрів чи їх окремих фрагментів.

3.1.2 Контроль рівня інформаційної інтенсивності та якості відео контенту шляхом зміни кількості структурних одиниць для опису відеокадрів чи їх окремих фрагментів

Розглянемо напрямок контролю інформаційної інтенсивності/якості відеоконтенту, який на сьогодні може бути реалізовано на базі кодування кадрів потоку у базисі JPEG2000 [16].

У цьому випадку кожен окремий кадр F після попередньої конвертації кольорового простору з RGB до YUV, а також сегментації (не є обов'язковою процедурою) далі підлягає процедурі дискретного вейвлетного перетворення

(ДВП) – часткового випадку ортогонального перетворення. На відміну від ДКП, процедура ДВП передбачає виконання 4 етапів вейвлетної фільтрації відносно кожного з сегментів $S(F)_{x,y}$ кадру F , або кадру цілком за наступним сценарієм:

- фільтрація рядків та стовпців вейвлетними НЧ-фільтрами (утворюється суб-сегмент LL);
- фільтрація рядків вейвлетними НЧ-фільтрами а стовпців - ВЧ-фільтрами (утворюється суб-сегмент LH);
- фільтрація рядків вейвлетними ВЧ-фільтрами а стовпців - НЧ-фільтрами (утворюється суб-сегмент HL);
- застосування для рядків та стовпців вейвлетних ВЧ-фільтрів (формується суб-сегмент HH).

Формально даний процес може бути зображено рис. 3.3. Важливим тут є те, що наслідком одного такту ДВП-перетворення є утворення 4 суб-

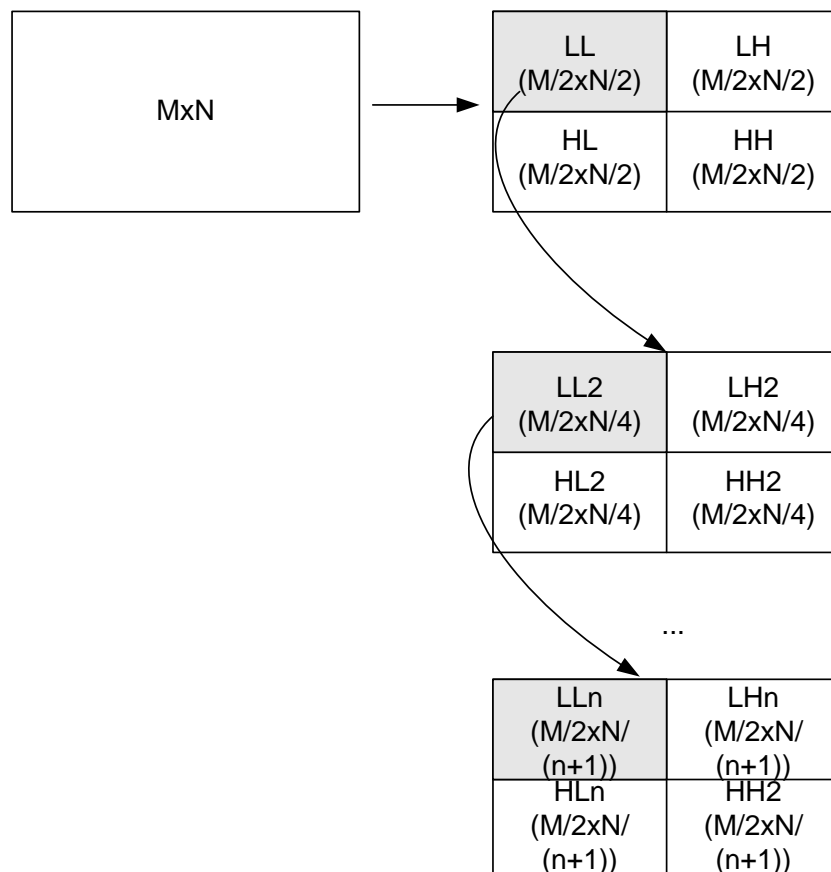


Рисунок 3.3 – Результуючі роздільні здатності сегментів на різних рівнях ДВП-перетворення

сегментів C_{sub} роздільної здатності $(M/2 \times N/2)$ на першому етапі перетворення.

У подальшому, відносно вихідного кадру/сегменту за потреби може бути примінено ряд етапів ДВП-перетворення. При цьому, для кожного такого i -го етапу використовується суб-сегмент LL, ориманий на етапі $(i-1)$. Це пояснюється тим, що саме LL суб-сегмент є найбільш інформативним, тобто, несе у собі максимум потужності сигналу.

Таким чином, на кожному черговому i -му етапові ДВП-перетворення (рис.4.3) утворюється множина сегментів/кадрів потоку відео, яка буде у 2 рази меншою порівняно з утвореною на $(i-1)$ -му етапові за горизонтальною та вертикальною роздільними здатностями.

Отже, кожному суб-сегменту $C_{\text{sub}}^{(i)}$, урахувуючи зазначену закономірність зменшення роздільної здатності кадру відповідно до збільшення кількості етапів ДВП-перетворення, відповідатиме свій деякий середній рівень Φ_i інформаційної інтенсивності, що як i у випадку співвідношення (4.1) буде априорі нижчим, ніж Φ_{i-1} , що відповідає суб-сегменту $C_{\text{sub}}^{(i-1)}$.

Звідси виходить, що використання багаторівневої ДВП-трансформації здатне на базі вихідного кадру $M \times N$ утворити сітку роздільних здатностей. Проте, на відміну від ABR, така сітка формується динамічно, при цьому немає потреби утворювати та зберігати кілька файлів різної роздільної здатності.

Разом з тим, у рамках кодування кадрів потоку відео на засадах jpeg2000, головним механізмом контролю інформаційної інтенсивності відео та його рівня якості є EBCOT (Embedded Block Coding with Optimal Truncation) – метод внутрішньо кадрового кодування з оптимальним усіченням.

Сутність механізму EBCOT полягає у наступному [16].

Після того, як над кадром F виконано процедуру ДВП-трансформації, відносно кожного з його суб-сегментів – LL, LH, HL та HH реалізується процедура зміни формату опису, за результатом чого довільний суб-сегмент $C_{\text{sub}10}^{(i)}$ десяткового формату опису буде переведено до двійкового простору представлення. Для цього спочатку суб-сегмент розглянемо, як матрицю $M \times N$ компонент, утворених після виконання процедури ДВП над пікселями початкового сегменту, а саме:

$$C_{\text{sub}10}^{(i)} = \bigcup_{x=1}^M \bigcup_{y=1}^N C(x, y)_{\text{sub}10}^{(i)}, \quad (3.3)$$

де $C(x, y)_{\text{sub}10}^{(i)}$ - компонента суб-сегменту $C_{\text{sub}10}^{(i)}$ десятичного формату представлення на позиції (x, y) .

Після цього виконується процедура бінаризації матриці $C_{\text{sub}10}^{(i)}$, тобто $C_{\text{sub}10}^{(i)} \rightarrow C_{\text{sub}2}^{(i)}$. У наслідок даної операції кожна з компонент $C(x, y)_{\text{sub}10}^{(i)}$ десятичного формату надалі розглядається як множина бінарних елементів, тобто:

$$C_{\text{sub}2}^{(i)} = \{b_1, b_2, \dots, b_n, \dots, b_N\} \quad (3.4)$$

При цьому, за рахунок об'єднання біт i -го розряду, може бути утворено бінарну матрицю i -го рівня відповідно до виразу:

$$C(n)_{\text{sub}2}^{(i)} = \bigcup_{x=1}^X \bigcup_{y=1}^Y b(x, y)_v, \quad (3.5)$$

де $b(x, y)_v$ - біт v -го розряду на позиції (x, y) .

У наслідок цього на базі виразу (3.6) утворюється 8 бінарних матриць $C(n)_{\text{sub}2}^{(i)}$. При цьому, кожна i -та матриця утворюється поєднанням біт i -го розряду.

Таким чином, замість однієї матриці $C_{\text{sub}10}^{(i)}$ маємо 8 матриць $C_{\text{sub}2}^{(i)}$.

У наслідок цього отримується можливість впливу на рівень $\Phi(t)$ шляхом маніпулювання кількістю двійкових матриць $C_{\text{sub}2}^{(i)}$ (рис.4.4), на базі яких будуватиметься структура матриці $C_{\text{sub}10}^{(i)}$.

Таким чином, отримується можливість контролю інформаційної інтенсивності, ураховуючи при цьому вимоги до рівня якості. У даному разі передбачається, що, у загальному випадку справедливим є твердження про те, що:

$$(i \rightarrow 8) \rightarrow (\Phi(t) \rightarrow \Phi(t)_{\max}), \quad (3.6)$$

де $\Phi(t)_{\max}$ - рівень інтенсивності, який відповідає випадку використання для опису суб-сегменту $C_{\text{sub}10}^{(i)}$ усіх 8 двійкових матриць $C_{\text{sub}2}^{(i)}$.

Таким чином, варіюючи індекс i , можна змінювати рівень $\Phi(t)$.

З іншого боку, ураховуючи взаємозв'язок $\Phi(t)$ та Q очевидно, що, зменшуючи i у виразі (4.3), отримуватимемо як різні значення $\Phi(t)$, так і Q .

При цьому, відповідно до ідеології ЕВСОТ, кожна з матриць $C_{\text{sub}2}^{(i)}$ додає свій пропорційний внесок до величини $\Phi(t)$ суб-сегменту $C_{\text{sub}10}^{(i)}$, величина даного внеску залежить від індексу i . У цьому випадку, максимальний внесок у величину як Q , так і $\Phi(t)$, чиниться при $i \rightarrow 8$. І навпаки – при $i \rightarrow 1$ рівень внеску суттєво скорочується.

У свою чергу, для того, щоб встановити взаємозв'язок між Q та $\Phi(t)$, у базовому варіанті технології використано метод множників Лагранжа, що може бути проілюстровано виразом:

$$Q = \lambda \Phi(t), \quad (3.7)$$

де λ - множник Лагранжа, що встановлює відповідність між значеннями величин Q та $\Phi(t)$ при однаковій кількості i . Фізична сутність множника λ - кут нахилу кривої якості/інтенсивності на її окремій елементарній ділянці (рис.4.4).

Разом з тим, залежність між Q та $\Phi(t)$, наведена рис. 3.4 є актуальною виключно за умови, що кожна з матриць $C_{\text{sub}2}^{(i)}$ є значимою – тобто, містить ненульові елементи, яких при цьому, буде переважна більшість.

В інших випадках модель, зображена рис. 3.4, не буде описувати реальну залежність між рівнями інтенсивності відео та підсумковим рівнем якості, так як зміст матриць $C_{\text{sub}2}^{(i)}$ у реальності визначається випадковим розподілом.

Отже, математичний апарат ЕВСОТ, реалізований у рамках механізму контролю інформаційної інтенсивності та якості відео, не дозволяє

застосовувати дану технологію, як універсальний засіб. Про це також зазначається, зокрема, у офіційній специфікації стандарту jpeg2000 [160, 17].

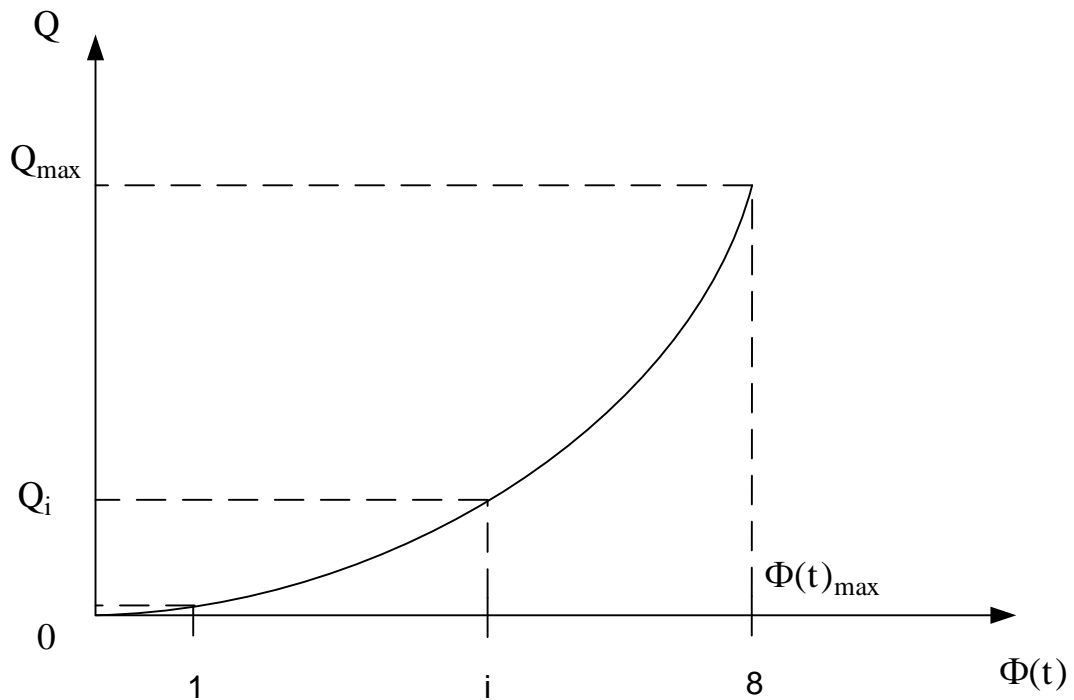


Рисунок 3.4 – Характер залежності між значеннями Q та $\Phi(t)$ згідно загальних засад EVCOT

3.1.3 Механізм контролю якості відео на базі виявлених інструментів у складі MPEG, здатних впливати на рівень інформаційної інтенсивності потоку

Отже, як раніше було визначено у ході аналізу внутрішньо кадрового алгоритму кодування, характерного для базисного MPEG, впливати на рівень підсумкового значення $\Phi(t)$ можуть такі механізми зі складу технології, як:

- формат та опції проріджування колірно-різницевих матриць;
- обрані параметри квантування.

Додатковим механізмом може також розглядатися маніпулювання частотою слідування кадрів у потоці.

Таким чином, для побудови гнучкого та продуктивного механізму контролю якості відео передбачається спільне використання кожного з зазначених механізмів у черговості, що відповідає ступеню внесеної помилки

у реконструйоване відео. Беручи до уваги вищезазначене, а також ураховуючи взаємозв'язок величин Q та $\Phi(t)$, розглянемо спочатку алгоритм контролю рівня інформаційної інтенсивності кодованого відеопотоку (рис.3.5). $\varphi > \varphi_{\min}$

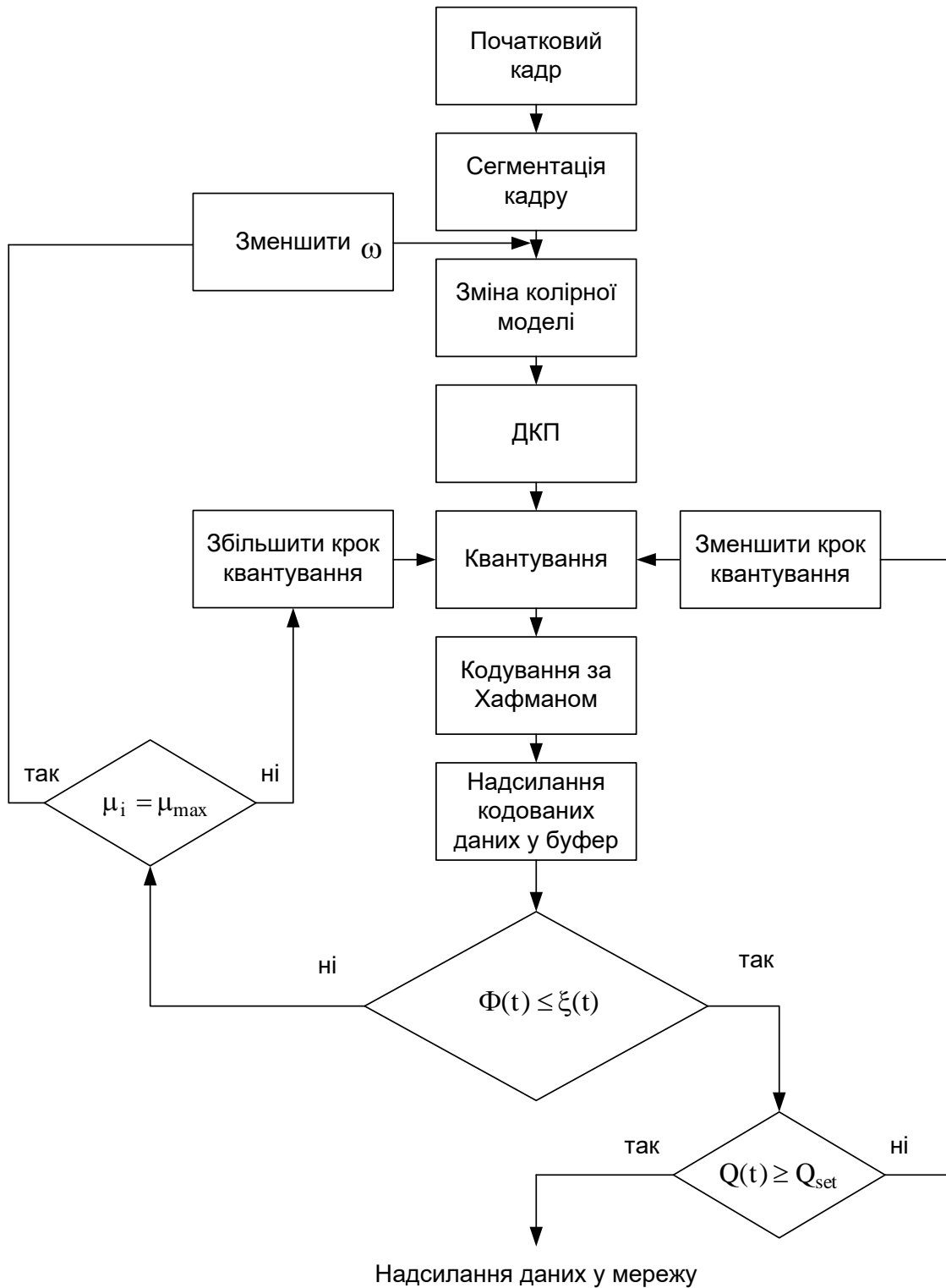


Рисунок 3.5 - Структурна схема управління інтенсивністю відеотрафіку

Схема, наведена на рис. 3.5 передбачає, що ключовим механізмом контролю рівня якості/інформаційної інтенсивності є маніпуляція кроком квантування.

На рис.3.5 початково передбачається, що обробка кадру виконується з використанням набору опцій кодексу за замовчуванням.

Далі, якщо за умови використання даних опцій у момент часу t величина $\Phi(t)$ перевищує значення $\xi(t)$, для наступного моменту $(t + 1)$ виконується збільшення кроку μ квантування.

Якщо при цьому буде досягнуто значення μ_{\max} , проте зменшення величини $\Phi(t)$ до значення $\Phi(t) \leq \xi(t)$ не буде забезпечено, далі приймається рішення щодо необхідності скорочення частоти ω слідування кадрів.

При цьому, у контурі контролю якості головним механізмом управління її величиною також залишається маніпулювання кроком квантування.

Очевидно, що наведена схема не може гарантувати достатньої гнучкості, так як вибір величин $\Phi(t)$ здійснюється з дискретного простору, що залежить, за великим рахунком, від доступної кількості кроків квантування.

У зазначених умовах збільшення кроку вибору $\Phi(t)$ та, як наслідок, можливості впливу на величину Q якості, досягається включенням у схему обробки механізму вибору режиму проріджування хроматичних складових фрагменту відеокадру.

Схема, яка включає у себе також механізм вибору режиму проріджування хроматичних складових фрагменту відеокадру, наводиться рис. 3.6.

У даному випадку на перших кроках контролю рівня інтенсивності/якості відеопотоку виконуються маніпуляції щодо застосовуваної моделі колірної моделі.

При цьому, передбачається, що спочатку кодування здійснюється у моделі 4:4:4, проте, одночасно з цим формується сітка моделей ϕ , де збільшення індексу ϕ вказує на підвищення ступеню проріджування матриць S_b та S_r фрагментів а отже – зменшення кількості біт, необхідних для їх опису [9, 11, 14].

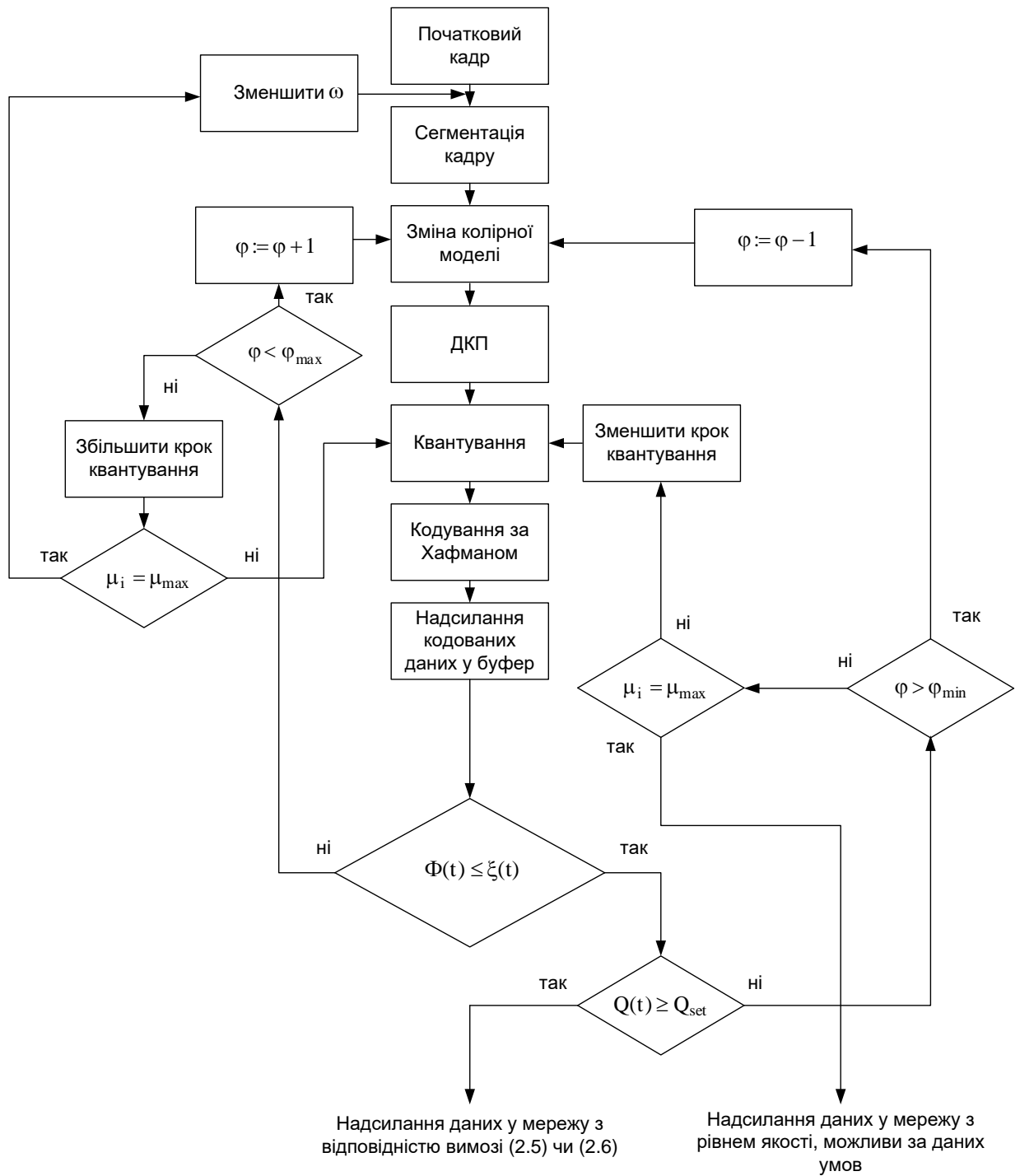


Рисунок 3.6 – Розширена структурна схема механізму управління інтенсивністю відеотрафіку з контролем рівня якості

Так, у випадку використання схеми 3.6 також початково необхідно гарантувати надсилання потоку стиснених кадрів у мережу, тобто, тут слід спершу забезпечити виконання умови $\Phi(t) \leq \xi(t)$. Для забезпечення гнучкого кроку зміни $\Phi(t)$ у рамках даної схеми спочатку використовується механізм

вибору режиму проріджування хроматичних складових фрагменту відеокадру.

При цьому, даний механізм використовується доти, поки не досягнуто φ_{\max} , тобто, доки не обрано режим максимального проріджування хроматичних складових. Доцільність застосування такого механізму на першому етапі контролю Q та $\Phi(t)$ полягає у тому, що крок зміни даних величин є значно меншим, ніж за умови маніпулювання кроком μ квантування.

У свою чергу, якщо тут:

$$\varphi = \varphi_{\max} \mid \Phi(t) > \xi(t), \quad (3.7)$$

далі буде застосовано механізм зміни кроку μ квантування до тих пір, або поки буде забезпечено справедливість нерівності $\Phi(t) \leq \xi(t)$, або поки буде досягнуто значення μ_{\max} .

Далі, в умовах:

$$(\varphi = \varphi_{\max}) \& (\mu = \mu_{\max}) \mid \Phi(t) > \xi(t) \quad (3.8)$$

для забезпечення гарантованої можливості надсилання кодованого відео у мережу додатково застосовуватиметься механізм зменшення частоти слідування кадрів.

Разом з тим, у схемі (4.6) контур контролю рівня якості розширено.

Так, в умовах:

$$(\Phi(t) > \xi(t)) \& (Q(t) < Q_{\text{set}}) \mid (\varphi > \varphi_{\min}), \quad (3.9)$$

виконується поступове зменшення значення φ , або до виконання умов $Q(t) \geq Q_{\text{set}}$, або до тих пір, поки величина φ не стане рівною φ_{\min} . Після цього, відповідно, у контурі контролю якості відеоінформації використовуватиметься механізм зміни кроку квантування, тобто, умови для його застосування формально будуть наступними:

$$(\Phi(t) > \xi(t)) \& (Q(t) < Q_{\text{set}}) \& (\varphi = \varphi_{\min}) \mid (\mu < \mu_{\max}) \quad (3.10)$$

Якщо таким чином буде забезпечено відповідність величини $Q(t)$ вимозі (3.5) або (3.6), отримані таким чином дані надалі спрямовуються до мережі.

В інакшому випадку, далі, якщо у підсумку буде досягнуто величини $\mu = \mu_{\min}$, але, при цьому, рівень якості ні за вимогою (3.5), ні за вимогою (3.6) забезпечити у поточних умовах неможливо, до мережі завантажуватиметься відеоконтент, для якого виконуватиметься лише нерівність $\Phi(t) \leq \xi(t)$, що, тим паче, гарантує можливість проходження його мережею.

4 ДОДАТКОВІ ЗАХОДИ ЗАБЕЗПЕЧЕННЯ ЯКОСТІ ВІДЕОІНФОРМАЦІЇ У МЕРЕЖІ

4.1 Обґрунтування необхідності залучення допоміжних інструментів забезпечення якості відеоінформації у ході трансляції

Розглянемо відеодані як інформаційний потік узагалі. Тобто, не беручи до уваги специфіки відеоінформації.

У цьому випадку відеодані будуть інтерпретуватися як бінарна послідовність даних, що надходить у канал з інтенсивністю, рівною $\Phi(t)$.

Так як відеоінформація, як і будь-який інший тип даних, розповсюджується каналами мережі, де має місце вплив сторонніх завад різної природи та потужності, це, у свою чергу, зумовлює виникнення помилок на рівні пакетів [12, 17].

У свою чергу, це викликає:

- падіння якості відновленого відео за рахунок присутності помилок;
- збільшення часових затримок у ході передавання, якщо у ролі транспорту виступає один з протоколів гарантованої доставки – TCP, RTSP, RTP та їм подібні, як наслідок виконання повторних запитів доставки пакетів, прийнятих з помилками.

Отже, необхідно забезпечити умови для хоча б часткової компенсації помилок, що можуть виникати у ході надсилання відеоінформації мережею. У цьому випадку пропонується розглянути приклад використання завадостійкого кодування для виправлення помилок прийому.

4.2 Використання завадостійкого кодування на базі алгоритму Хемінга для підвищення якості відеоданих

4.2.1 Загальні засади побудови методу Хемінга

В основу методу кодування за Хемінгом покладено перевірку парності ряду одиничних символів. При цьому, до кодованої послідовності біт повинен бути доданий такий контрольний біт, щоб сума всіх одиничних символів у повідомленні була парною, тобто сума за модулем 2 дорівнювала нулю [18-19].

У свою чергу, формування коду Хемінгу здійснюється за ряд технологічних етапів, серед яких:

1. Обчислення кількості контрольних біт.
2. Поміщення контрольних біт на позиції у повідомленні, індекс яких відповідає ступеням двійки.
3. Обчислюється контрольна сума біт: починаючи з позиції n контрольного біта, підсумовуються кожен n біт, потім пропускаються n біт і знову підраховується сума.
4. Кожна контрольна сума береться за модулем 2 і записується у відповідний контрольний біт.

Далі розглянемо приклад програмної реалізації коду Хемінга на базі мови програмування Python.

4.2.2 Лістинг коду програми

Розглянемо приклад реалізації коду Хемінга за умови, що кодованою послідовністю буде масив біт, утворений генератором псевдовипадкових чисел, а саме – 0100010000111101.

Код програми на базі Python наводиться далі:

```
import copy
import math
# обчислює кількість контрольних біт та формує повідомлення
# з місцями під контрольні біти
def raw_control(word):
    len_word = len(word)
    k = 2 # у контрольних біт
    word[0:0] = [0]
    word[0:0] = [0]
    for i in range(len_word - 1):
        x = pow(2, k) - 1 # Обчислення позиції наступного контр. біта
```

```

if x >= len(word): break # передчасний вихід, якщо контр. біт уже зайвий
word[x:x] = [0] # додавання контр. біта
k += 1 # оновлення лічильника контр. біт
return word, k

# обчислення контрольних бітів
def control_counting(raw_coded_word, num):
    len_raw = len(raw_coded_word)
    for i in range(num): # лічильник контрольного біта
        x = pow(2, i)-1 # індекс контрольного біта
        for j in range(x, len_raw, pow(2, i+1)): лічильник початку ділянки контрольної суми
            for k in range(pow(2, i)): # лічильник проходу
                if j+k >= len_raw: break # обробка виходу за межу слова
                raw_coded_word[x] += raw_coded_word[j+k] # підрахунок контрольної суми
                raw_coded_word[x] = raw_coded_word[x] % 2 # установка контрольного біта
    return raw_coded_word

# перетворити список на повідомлення
def to_message(word):
    return "".join(map(str, word))

# Перевірка на наявність помилки
def is_fault(message):
    fault = 0
    num = math.log2(len(message)) # підрахунок числа контрольних біт
    if num % 1 > 0: num = int(num)+1
    raw_message = copy.deepcopy(message) # отримання повідомлення з нульовими контр. бітами
    for i in range(num): # обнулення контр. біт
        raw_message[pow(2, i)-1] = 0
    raw_message = control_counting(raw_message, num) # перерахунок контр. біт
    for i in range(num): # зміна fault у разі розбіжності

```

```

t = pow(2, i)-1 # біт message та raw_message
if raw_message[t] != message[t]: fault += t+1
if fault: return True, fault
else: return False, None

# введення інформаційного коду
def word_input():
word = list(input("Вихідне повідомлення:\t\t"))
return list(map(int, word))

# введення закодованого повідомлення
def message_input():
message = list(input("Прийняте повідомлення:\t\t"))
return list(map(int, message))

# висновок про помилку
def result_output(accepted_message):
result, bit = is_fault(accepted_message)
if result:
print("Виправлене повідомлення:\t\t", to_message(fault_fix(accepted_message,
bit))))
print("Помилка в біті номер:\t\t", bit)
else: print("Повідомлення було передано без помилок")

# виправлення помилки у fault_message на місці place
def fault_fix(fault_message, place):
if fault_message[place-1]: fault_message[place-1] = 0
else: fault_message[place-1] = 1
return fault_message

word = word_input() # введення вихідного повідомлення
raw_coded_word, num = raw_control(word) # додавання в вих. повідомлення
місця під контр. біти
message = to_message(control_counting(raw_coded_word, num)) # підрахунок
контр. біт і перетворення на повідомлення

```

```
print("Закодоване повідомлення:\t\t", message) # висновок Закодоване
повідомлення
accepted_message = message_input() # введення прийнятого повідомлення
result_output(accepted_message) # аналіз та виправлення помилки за наявності
```

4.2.3 Опис програмних одиниць

1. Функції:

- word_input – введення інформаційного коду.

Вхідні дані:

відсутні.

Вихідні дані:

інформаційне повідомлення у вигляді списку.

Алгоритм роботи:

- 1) На введення потрібно вихідне повідомлення у форматі рядка;
- 2) Рядок перетворюється на список;
- 3) За допомогою функції map перетворюємо рядкові елементи списку на цілі числа;
- 4) Повертаємо з функції результат функції map, перетворений на список.
 - raw_control - обчислює кількість контрольних біт та формує повідомлення з місцями під контрольні біти.

Вхідні дані:

word – вихідне повідомлення у вигляді списку.

Вихідні дані:

word - результуюче повідомлення з інформаційних та контрольних біт, останні встановлені в 0.

k – кількість контрольних біт.

Алгоритм роботи:

- 1) Обчислюємо довжину вихідного повідомлення;

- 2) Ініціалізуємо змінну $k=2$, відповідальна за кількість контрольних біт;
- 3) Додаємо у вихідне повідомлення два обов'язкові контрольні біти поза циклом;
- 4) У циклі послідовно додаємо відсутні контрольні біти на позиції, відповідні ступеням двійки;
- 5) Обновлюємо лічильник контрольних біт;
- 6) Повертаємо змінене повідомлення `word` та кількість контрольних біт `k`.

- `control_counting` – обчислення контрольних бітів.

Вхідні дані:

`raw_coded_word` – повідомлення з нульовими контрольними бітами.

`num` – кількість контрольних біт.

Вихідні дані:

`raw_coded_word` – результуюче повідомлення

- 1) Обчислюємо довжину перетвореного повідомлення;
- 2) У циклі за кількістю контрольних біт i обчислюємо x - ступінь двійки – індекс контрольного біта;
- 3) У вкладеному циклі проходимо від індексу x ділянкою розміром x з відступом між елементами розміру x і підсумовуємо елементи ділянки;
- 4) Обчислюємо модуль суми та записуємо в контрольний біт;
- 5) Повертаємо результуюче повідомлення із порахованими контрольними бітами.

- `to_message` - перетворює список повідомлення.

Вхідні дані:

`word` – повідомлення.

Вихідні дані:

результуюче повідомлення у вигляді рядка.

Алгоритм роботи:

1) За допомогою методу join рядка та функції map перетворюємо список цілих чисел у рядок та повертаємо його.

- message_input – введення закодованого повідомлення.

Вхідні дані:

відсутні.

Вихідні дані:

інформаційне повідомлення у вигляді списку.

Алгоритм роботи:

1) На введення запитується прийняте повідомлення у форматі рядка;

2) Рядок перетворюється на список;

3) За допомогою функції map перетворюємо рядкові елементи списку на цілі числа;

4) Повертаємо з функції результат функції map, перетворений на список.

- fault_fix – виправлення помилки.

Вхідні дані:

fault_message – повідомлення з помилкою.

place – позиція помилки.

Вихідні дані:

fault_message – повідомлення із виправленою помилкою.

Алгоритм роботи:

1) Якщо позиції place списку fault_message стоїть одиниця – змінюємо її на 0;

2) Інакше – міняємо нуль на 1;

3) Повертаємо прийняте повідомлення fault_message із виправленою помилкою.

- is_fault – перевірка на наявність помилки.

Вхідні дані:

message - прийняте повідомлення у вигляді списку.

Вихідні дані:

True/False – прапор помилки.

fault/None – біт помилки або відсутність помилки

Алгоритм роботи:

- 1) Ініціалізуємо нулем змінну fault - біт помилки;
- 2) Ініціалізуємо змінну num значенням контрольних біт прийнятого

повідомлення за формулою:

$$num = \log_2(\text{длина сообщения}) + 1;$$

- 3) Копіюємо в змінну raw_message прийняте повідомлення message та обнулюємо контрольні біти для перерахунку;
- 4) Проводимо перерахунок контрольних біт у списку raw_message за допомогою вищеописаної функції control_counting, подавши як аргумент message;
- 5) У циклі по num порівнюємо контрольні біти message та raw_message, при розбіжності додаємо у fault індекс біта t і одиницю;
- 6) Якщо змінна fault змінилася з ініціалізації (тобто не дорівнює нулю), то повертаємо True (помилка є) і номер біта з помилкою fault;
- 7) Інакше – повертаємо False (помилки немає) та відсутність помилки.

- result_output

Вхідні дані:

accepted_message – прийняте повідомлення як списку.

Вихідні дані:

відсутні.

Алгоритм роботи:

- 1) Визначаємо наявність помилки та її біт (за наявності) за допомогою вищеописаної функції is_fault, подавши як аргумент accepted_message, виведення функції приймаємо в змінні result і bit;

2) Якщо `result` виявився рівним `True`, то виводимо в консоль виправлене повідомлення передавши у функцію `to_message` результат перетворення функції `fault_fix` з аргументами `accepted_message` і `bit` і номер біта з помилкою;

3) Інакше виводимо у консоль повідомлення про те, що повідомлення було передано без помилок.

2. Змінні основні функції

- `word` – вихідне повідомлення;
- `raw_coded_word` - вихідне повідомлення з доданими нульовими контрольними бітами;

- `num` – кількість доданих контрольних біт;
- `message` - результуюче повідомлення;
- `accepted_message` – прийняте повідомлення

4.3 Результати кодування

Одержані результати кодування наведено на рис.4.1

```

Вихідне повідомлення:      0100010000111101
Закодоване повідомлення:   100110000100001011101
Прийняте повідомлення:     100110100100001011101
Виправлене повідомлення:   100110000100001011101
Помилка у біті номер:      7

```

Рисунок 4.1 – Результати кодування за Хемінгом тестової послідовності біт

Як бачимо з рис. 4.1, програмна реалізація коду Хемінга забезпечує виявлення а далі – виправлення помилково прийнятих біт повідомлення.

ВИСНОВКИ

У чіткій відповідності до технічного завдання, під час виконання атестаційної роботи було здійснено:

1. Аналіз специфіки передавання відеоінформації мережею в умовах загального росту мережевого трафіку та ускладнення його структури.
2. Дослідження підходів, потенційно здатних забезпечити сталу трансляцію відеоінформації у мережі.

Показано, що створенню умов для ефективного функціонування як відеосервісів на базі мережі, так і будь-яких мережевих сервісів узагалі сприяє зменшення кількості пакетів відеоданих, що надходять до мережі.

При цьому також доведено, що:

1. Кодувати відео потік, який транслюється, зі сталим набором опцій кодеку, який, у підсумку, формує коефіцієнт стиснення, не має сенсу, так як інтенсивність відео та рівень пропускної спроможності є динамічними процесами, що також взаємно незалежні.

2. Коефіцієнт стиснення, та, відповідно, набір опцій кодеку, що його утворюють, має динамічно змінюватися, ураховуючи зміни пропускної спроможності мережі.

Окрім цього, досліджено механізм контролю інформаційної інтенсивності відео з використанням лімітованої ємності вихідного буферу.

Разом з тим, досліджено ряд механізмів контролю відео потоку на базі ABR та механізму EBCOT. Доведено, що їх використання на практиці обмежується:

- відсутністю механізмів контролю якості (ABR);
- незавершеністю математичного апарату (EBCOT).

Водночас, за результатами схеми внутрішньо кадрового кодування MPEG, визначено, що на рівень інформаційної інтенсивності відео потоку впливають такі технологічні етапи перетворення, як:

- вибір режиму проріджування хроматичних складових;
- квантування.

При цьому, зазначені етапи є керованими.

У якості додаткового механізму розглянуто використання зміни частоти слідування кадрів у мережу.

Далі користуючись виявленими механізмами, побудовано загальну схему контролю інформаційної інтенсивності відео з можливістю контролю рівня якості реконструйованих відеоданих.

Таким чином, усі завдання виконано у повній мірі.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. VNI Forecast Highlights Tool [Електронний ресурс] – Режим доступу: https://www.cisco.com/c/m/en_us/solutions/service-provider/vni-forecast-highlights.html
2. Video quality testing in mobile networks | Rohde & Schwarz [Електронний ресурс] – Режим доступу: https://www.rohde-schwarz.com/solutions/test-and-measurement/mobile-network-testing/expertise/video-quality-testing/video-quality-testing_232054.html?change_c=true
3. Качество обслуживания в операторских сетях [Електронний ресурс] – Режим доступу: https://www.opennet.ru/docs/RUS/qos_oper/
4. New Open Industry Standard Introduced for Connecting Next-Generation VR Headsets to PCs, Other Devices [Електронний ресурс] – Режим доступу: <https://www.globenewswire.com/news-release/2018/07/17/1538342/0/en/New-Open-Industry-Standard-Introduced-for-Connecting-Next-Generation-VR-Headsets-to-PCs-Other-Devices.html>
5. Гонсалес Р. Цифровая обработка изображений / Р. Гонсалес, Р. Вудс. – М. : Техносфера, 2005. – 1073 с.
6. Миано Дж. Форматы и алгоритмы сжатия изображений в действии: учебное пособие / Дж. Миано. пер. с англ. – М. : Триумф, 2003. – 336 с.
7. Ричардсон Ян. H.264 and MPEG-4 Video Compression: Video Coding for Next-Generation Multimedia / Ян Ричардсон. – Город. : Издательство, 2005. – 368 с.
8. Чемпен Н., Чемпен Д. Цифровые технологии мультимедиа. – М.: Вильямс, 2006. – 624 с.
9. Красильников Н.Н. Цифровая обработка изображений. – М.: Вузовская книга, 2011. – 320 с.
10. Айфичер Эммануил С. Цифровая обработка сигналов: практический поход / Эммануил С. Айфичер, Барри У. Джервис. – 2-е изд. – М. : Вильямс, 2008. – 992 с
11. Ватолин Д. Методы сжатия данных. Устройство архиваторов, сжатие изображений и видео / Д. Ватолин, А. Ратушняк, М. Смирнов, В. Юкин. – М. : ДИАЛОГ – МИФИ, 2003. – 384 с
12. Олифер В. Г. Компьютерные сети. Принципы, технологии, протоколы : учебник для вузов / В. Г. Олифер, Н. А. Олифер. – 3-е изд. – СПб. : Питер, 2006. – 958 с.

13. Video encoding in the cloud | Telestream Cloud [Електронний ресурс] – Режим доступу: <https://www.telestream.net/telestream-cloud/adaptive-bitrate-encoding.htm>
14. Shi, Yun Q. Image and video compression for multimedia engineering: fundamentals, algorithms, and standards / Yun Q Shi, Huifang Sun.
15. Сэломон Д. Сжатие данных, изображений и звука / Д. Сэломон. – М.: Техносфера, 2004. – 368 с.
16. Official JPEG2000 Homepage. [Електронний ресурс] – Режим доступу: <http://www.jpeg.org/jpeg2000.html>
17. Беляев Е.А, Тюрликов А.М. Управление скоростью и ошибкой кодирования в системе сжатия и передачи видеoinформации с ограничениями на память передающего и принимающего устройств // Компьютерная оптика. 2007. - Т.31. №2. С.69-76.
18. Романюк М.І., Власюк Г.Г. Основи теорії інформації та кодування [Електронний ресурс] : навч. посіб. для студ. спеціальності 171 «Електроніка», спеціалізації «Електронні та інформаційні технології кінематографії та аудіовізуальних систем» / М. І. Романюк, Г. Г. Власюк; КПІ ім. Ігоря Сікорського. – Електронні текстові данні (1 файл: 2,09 Мбайт). – Київ : КПІ ім. Ігоря Сікорського, 2018.
19. Жураковський Ю.П., Полторак В.П. Теорія інформації та кодування: Підручник. – К.: Вища шк., 2001. – 225 с.