

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
СХІДНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМ. В. ДАЛЯ  
ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ ТА ЕЛЕКТРОНІКИ  
КАФЕДРА КОМП'ЮТЕРНИХ НАУК ТА ІНЖЕНЕРІЇ

До захисту допускається  
Завідувач кафедри  
\_\_\_\_\_ Скарга-Бандурова І.С.  
« \_\_\_\_ » \_\_\_\_\_ 20\_\_ р.

**МАГІСТЕРСЬКА РОБОТА**

НА ТЕМУ:

КОМПОНЕНТИ ВІЗУАЛІЗАЦІЇ АРХІВНИХ ДАНИХ СИСТЕМИ ЛОКОМОТИВНОЇ  
БЕЗПЕКИ СЛБ-І

Освітньо-кваліфікаційний рівень “Магістр”  
Спеціальність 123 – Комп’ютерна інженерія (Спеціалізація – Системне програмування)

Науковий керівник роботи:

\_\_\_\_\_

(підпис)

Скарга-Бандурова І. С.

\_\_\_\_\_

(ініціали, прізвище)

Консультант з охорони праці:

\_\_\_\_\_

(підпис)

Критська Я. О.

\_\_\_\_\_

(ініціали, прізвище)

Студент:

\_\_\_\_\_

(підпис)

Грушка М. О.

\_\_\_\_\_

(ініціали, прізвище)

Група:

СП-16дм

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
СХІДНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ  
ІМЕНІ ВОЛОДИМИРА ДАЛЯ**

Факультет Інформаційних технологій та електроніки  
Кафедра Комп'ютерних наук та інженерії  
Освітньо-кваліфікаційний рівень магістр  
Напрямок підготовки \_\_\_\_\_  
(шифр і назва)

Спеціальність \_\_\_\_\_  
(шифр і назва)

**ЗАТВЕРДЖУЮ:**

Завідувач кафедри \_\_\_\_\_  
\_\_\_\_\_ І.С. Скарга-Бандурова  
« \_\_\_\_\_ » \_\_\_\_\_ 20 \_\_\_\_\_ р.

**З А В Д А Н Н Я  
НА ДИПЛОМНИЙ ПРОЕКТ (РОБОТУ) СТУДЕНТУ**

Грушка Михайлу Олександровичу  
(прізвище, ім'я, по батькові)

1. Тема проекту (роботи): Компоненти візуалізації архівних даних системи локомотивної безпеки СЛБ-І

керівник проекту (роботи) д.т.н., проф. Скарга-Бандурова І. С.,  
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом вищого навчального закладу від "\_\_\_" \_\_\_\_\_ 2017 року №

2. Строк подання студентом проекту (роботи) \_\_\_\_\_

3. Вихідні дані до проекту (роботи) матеріали науково-дослідної практики

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити) Дослідження методів візуалізації великих наборів даних за допомогою графіків, Аналіз існуючих алгоритмів спрощення полігональних ланцюгів, розробка комплексного алгоритму спрощення параметрів архівних даних, практична реалізація програмних компонентів для візуалізації параметрів архівних даних з використанням розробленого комплексного алгоритму.

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)

6. Консультанти розділів проекту (роботи)

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Охорона праці	Критська Я.О.		

7. Дата видачі завдання \_\_\_\_\_

**КАЛЕНДАРНИЙ ПЛАН**

№ з/п	Назва етапів дипломного проекту (роботи)	Строк виконання етапів проекту (роботи)	Примітка
1	Дослідження методів візуалізації великих наборів даних за допомогою графіків		
2	Аналіз існуючих алгоритмів спрощення параметрів архівних даних		
3	Розробка алгоритму для спрощення полігональних ланцюгів архівних даних		
4	Розробка компоненти для візуалізації архівних даних		
5	Оформлення пояснювальної записки та плакатів		

Студент \_\_\_\_\_ Грушка М.О.  
(підпис) (прізвище та ініціали)

Керівник проекту (роботи) \_\_\_\_\_  
(підпис) (прізвище та ініціали)

## АНОТАЦІЯ

Грушка М.О. Компоненти візуалізації архівних даних системи локомотивної безпеки СЛБ-І.

Розглянуто проблеми візуалізації послідовностей великих наборів даних. Задачу пошуку оптимального відображення даних сформульовано у вигляді задачі вибору алгоритму спрощення простих полігональних ланцюгів з мінімальною кількістю помилок. Виконано аналіз ефективності алгоритмів Рамера-Дугласа-Пекера, Реумана-Віткама, Опхейма, Ланга, алгоритму виключення за висотою трикутника та алгоритму радіального виключення точок при роботі з великими даними. Наводяться результати порівняння та оцінки ефективності їх використання для аналізу реальних даних, що збираються в системах локомотивної безпеки. Іл.: 28. Табл.: 20. Бібліогр.: 43 назв.

**Ключові слова:** візуалізація, великі набори даних, алгоритм спрощення, полігональний ланцюг, система локомотивної безпеки

## АНОТАЦИЯ

Грушка М.А. Компоненты визуализации архивных данных системы локомотивной безопасности СЛБ-И.

Рассмотрены проблемы визуализации последовательностей больших наборов данных. Задача поиска оптимального отображения данных сформулирована в виде задачи выбора алгоритма упрощения простых полигональных цепей с минимальным количеством ошибок. Выполнен анализ эффективности алгоритмов Рамера-Дугласа-Пэкера, Реумана-Виткама, Опхейма, Ланга, алгоритма исключения по высоте треугольника и алгоритма радиального исключения точек при работе с большими данными. Приводены результаты сравнения и оценки эффективности их использования для анализа реальных данных, собираемых в системах локомотивной безопасности. Ил. : 28. Табл. : 20. Библиогр. : 43 наим.

**Ключевые слова:** визуализация, большие наборы данных, алгоритм упрощения, полигональная цепь, система локомотивной безопасности.

## ABSTRACT

Grushka M.O. Components of visualization of archival data of the locomotive safety system SLB-I.

The problems of visualization of sequences of large data sets are considered. The task of finding an optimal data mapping is formulated in the form of selecting the best algorithm for simplification of simple polygonal circuits with a minimum number of errors. The analysis of the efficiency of variation of Ramer-Douglas-Packer algorithm, Reumann-Whitcom algorithm, Opheim algorithm, Lang simplification algorithm, the triangle exclusion algorithm and the radial exclusion algorithm for large data sets are performed. The results of comparison and estimation of their efficiency for the analysis of real data collected in systems of locomotive safety are presented.

**Keywords:** visualization, large data sets, simplification algorithm, polygonal chain, system of locomotive safety

## ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧОК І СКОРОЧЕНЬ .....	7
ВСТУП .....	8
1 ТЕОРЕТИЧНІ АСПЕКТИ ПРОБЛЕМИ ВІЗУАЛІЗАЦІЇ ВЕЛИКИХ ДАНИХ .....	11
1.1 Огляд проблем візуалізації великих даних .....	11
1.1.1 Візуальний шум .....	11
1.1.2 Сприйняття великих зображень .....	11
1.1.3 Спрощення даних .....	12
1.1.4 Математичні обмеження алгоритмів спрощення полігональних ланцюгів .....	14
1.1.5 Оцінка якості спрощення полігональних ланцюгів методом знаходження позиційних похибок. ....	14
1.2 Аналіз вимог щодо візуалізації даних системи локомотивної безпеки .....	15
1.3 Постановка наукової задачі та обґрунтування методики досліджень .....	16
1.4 Практичні задачі роботи: .....	17
1.5 Висновки .....	18
1.6 Література до розділу 1 .....	19
2 АНАЛІЗ АЛГОРИТМІВ СПРОЩЕННЯ ПОЛІГОНАЛЬНИХ ЛАНЦЮГІВ .....	20
2.1 Вибір алгоритмів спрощення полігональних ланцюгів.....	20
2.1.1 Алгоритм Дугласа-Рамера-Пекера .....	20
2.1.2 Алгоритм Реумана-Віткама .....	21
2.1.3 Алгоритм Опхейма.....	22
2.1.4 Алгоритм Ланга.....	23
2.1.5 Алгоритм виключення за висотою трикутника.....	24
2.1.6 Алгоритм радіального виключення.....	24
2.2 Вибір порогу спрощення .....	25
2.3 Порівняльний аналіз алгоритмів спрощення.....	25
2.2.1 Візуальний аналіз .....	26
2.3.1 Математичний аналіз .....	29
2.4 Висновки до розділу 2 .....	31
Література до розділу 2 .....	32
3 ПРАКТИЧНА РЕАЛІЗАЦІЯ ПРОГРАМНИХ КОМПОНЕНТІВ ВІЗУАЛІЗАЦІЇ ДАНИХ	33
3.1 Функціональне призначення ПЗ «Дешифратор» .....	33
3.2 Формалізація задачі програмного опису технологічних параметрів .....	34
3.2.1 Розробка програмного інтерфейсу для роботи із значеннями параметрів архівних даних	34
3.3 Розробка програмних інтерфейсів для описання алгоритмів вибору значень .....	35
3.3.1 Розробка інтерфейсу для опису алгоритму вибору значень параметрів, що архівуються БУ(Блоками управління).....	35

3.3.2	Розробка інтерфейсу для опису алгоритму вибору значень параметрів, що архівуються ДЛ(Дисплеями локомотивними).....	36
3.3.3	Розробка інтерфейсу для опису алгоритму вибору значень параметрів, що архівуються БФДС(Блоками формування дискретних сигналів).....	37
3.3.4	Розробка інтерфейсу для опису алгоритму вибору значень параметрів, що архівуються ДЛ(Дисплеями локомотивними) та ІнЛ(Індикатором локомотивним).....	38
3.4	Створення інтерфейсу, що відповідає за відображення даних параметру на графіку.	39
3.5	Створення кінцевого класу-нащадка, що репрезентує параметр архівних даних. ....	42
3.6	Відображення об'єктів електронної карти. ....	43
3.7	Формування груп параметрів.....	47
3.8	Компонента відображення графіків.....	49
3.9	Панель параметрів.....	51
3.10	Масштабування графіків та виведення значень параметрів за курсором.....	53
3.11	Висновки до розділу 3.....	58
3.12	Література до розділу 3.....	59
4	<b>ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ. ЕКОЛОГІЯ</b> ....	61
4.1	Загальні питання з охорони праці.....	61
4.1.1	Організаційно-технічні заходи з безпеки праці.....	62
4.2	Аналіз стану умов праці.....	63
4.2.1	Вимоги до приміщень.....	63
4.2.2	Вимоги до організації місця праці.....	63
4.2.3	Навантаження та напруженість процесу праці.....	65
4.3	Виробнича санітарія.....	65
4.3.1	Аналіз небезпечних та шкідливих факторів при виробництві (експлуатації) виробу 65	
4.3.2	Пожежна безпека.....	67
4.3.3	Електробезпека.....	68
4.4	Гігієнічні вимоги до параметрів виробничого середовища.....	68
4.4.1	Мікроклімат.....	68
4.4.2	Освітлення.....	69
4.5	Вентилювання.....	70
4.6	Заходи з організації виробничого середовища та попередження виникнення надзвичайних ситуацій.....	71
4.7	Охорона навколишнього природного середовища.....	73
4.7.1	Загальні дані з охорони навколишнього природного середовища.....	73
4.7.2	Визначення впливу та заходів щодо поводження з відходами ІТ галузі.....	74
4.8	Висновки до розділу 4.....	75
	Література до розділу 4.....	76
5	<b>ВИСНОВКИ</b> .....	77

## ПЕРЕЛІК УМОВНИХ ПОЗНАЧОК І СКОРОЧЕНЬ

АЛСБ	Автоматична лінійна сигналізація безперервного типу
БРЛ	Блок реєстрації локомотивний
БФДС	Блок формування дискретних сигналів
КР	Касета реєстрації
СЛБ	Система локомотивної безпеки
ТЗА	Технічні засоби автоматизації
LOD	Level of Detail
МВРС	Моторвагонний рухомий склад
PPI	Pixel per inch (Пікселів на дюйм)

## ВСТУП

**Актуальність теми.** Обробка великих даних зовсім не є тривіальним завданням, і для цього потрібні спеціальні методи та підходи. Графічне відображення - це дуже простий та ефективний спосіб візуалізації даних, що дозволяє полегшити сприйняття даних та надає достатньо інформації для аналізу і прийняття рішень. Але, у випадку великих даних, ефективність класичних методів візуалізації даних зменшується пропорційно збільшенню об'єму відображених даних.

До методів візуалізації даних належать: таблиці, гістограми, точкові діаграми, лінійні діаграми, кругові діаграми, стрижневі діаграми.

В рамках дослідження, приділяючи увагу властивостям наборів даних, можна виділити наступні проблеми візуалізації великих даних:

- Візуальний шум.
- Сприйняття великих зображень.
- Спрощення даних

Візуальний шум – втрата видимості даних внаслідок накладання графічних примітивів. У загальному випадку проблема візуального шуму може бути вирішена шляхом збільшення розмірів компоненти відображення, але у випадку візуалізації великих наборів даних такий підхід може призвести до появи проблема сприйняття великого зображення.

Всі способи візуалізації обмежуються роздільною здатністю пристрою, що відповідає за вивід візуалізації, тому існує межа кількості точок, які повинні відобразитися для кожної візуалізації. Звичайно, ми можемо замінити візуалізаційний пристрій більш сучасним або групою пристроїв для часткової візуалізації даних, що дозволяє нам представити більш детальне зображення з більшою кількістю точок даних, але навіть якщо ми зможемо повторити цей процес нескінченну кількість разів, ми зустрінемося з обмеженням людського сприйняття. Зі збільшенням об'ємів даних, що відображаються відразу, людина буде стикатися з труднощами в розумінні даних та їх аналізу. Тому можна стверджувати, що методи візуалізації даних обмежені не тільки співвідношенням сторін та роздільною здатністю пристрою, але й обмеженнями фізичного сприйняття.

Для вирішення проблеми сприйняття великих зображень вихідні дані піддають передобробці задля виділення меншої за обсягом підмножини даних, що якісно відображає поведінку вихідного набору великих даних. Задача відображення великих обсягів даних на точкових та лінійних діаграмах може бути розглянута як окремий випадок задачі спрощення полігональних ланцюгів.



У роботі буде розглянута проблема застосування алгоритмів спрощення полігональних ланцюгів до даних технологічних параметрів із урахуванням пріоритетів та дотримання необхідного рівня деталізації.

Актуальність теми обумовлюється необхідністю адаптації існуючих алгоритмів спрощення полігональних ланцюгів до обробки даних з різними пріоритетами.

*Об'єкт дослідження* – процеси формування та візуалізації великих наборів даних на прикладі системи локомотивної безпеки СЛБ-І.

*Предмет дослідження* – методи, моделі та алгоритми, що використовуються для спрощення полігональних ланцюгів та візуалізації великих наборів даних.

**Мета і завдання дослідження.** Метою дослідження є підвищення швидкості візуалізації великих наборів архівних даних для системи локомотивної безпеки за рахунок розробки методів спрощення полігональних ланцюгів із збереженням точності даних та з урахуванням їх пріоритетів.

Для досягнення мети дослідження необхідно вирішити такі **завдання**:

- аналіз методів і засобів спрощення полігональних ланцюгів;
- оцінка ефективності алгоритмів спрощення полігональних ланцюгів;
- розроблення метода динамічного спрощення в залежності від масштабу;
- розроблення програмних компонент для візуалізації даних технологічних параметрів.

**Методи дослідження.** Проведені в роботі дослідження основані на методах та алгоритмах спрощення ліній, що використовувались для узагальнення векторів; методах управління рівнями деталізації (Level of Detail (LOD) management), що використовувались для автоматизації візуальної обробки даних при зміні їх масштабу; методах об'єктно орієнтованого програмування що використовувались при розробці програмних компонентів для візуалізації архівних даних системи локомотивної безпеки СЛБ-І.

#### **Наукова новизна одержаних результатів:**

1) Запропоновано використання алгоритмів спрощення полігональних ланцюгів для вирішення проблеми візуалізації великих наборів даних технологічних параметрів.

2) Розроблено алгоритм динамічної зміни сили спрощення даних технологічних параметрів в залежності від масштабу.

#### **Практичне значення одержаних результатів**

Основні теоретичні результати роботи втілено у наступних практичних додатках і положеннях:

- Обґрунтовано і подано варіант алгоритму для спрощення полігональних ланцюгів з пріоритетними даними.
- Розроблено компонент візуалізації архівних даних системи локомотивної безпеки СЛБ-І, який реалізує запропонований алгоритм для відображення довгих поїздок.

**Особистий внесок здобувача** полягає у розробленні нових моделей, методів та інструментальних засобів, що дозволяють вирішити поставлені задачі. Усі основні результати

отримані автором особисто.

**Апробація матеріалів дисертації.** Основні положення, ідеї, висновки магістерської роботи доповідалися та обговорювалися на форумі «ІТ Ідея 2017».

**Зв'язок з науковими програмами, планами, темами.** Магістерська робота виконана у Східноукраїнському національному університеті ім. В. Даля у відповідності з державними програмами і планами НДР: Internet of Things: Emerging Curriculum for Industry and Human Applications (ALIOT), реєстраційний номер 573818-EPP-1-2016-1- UK-EPPKA2-SVNE-JP.

**Структура та обсяг роботи.** Магістерська робота складається зі вступу, трьох розділів, висновків і списку використаної літератури. Робота викладена на 77 сторінках машинописного тексту, містить 28 рисунків, 20 таблиць. Бібліографічний список містить 43 найменувань.

**Реалізація.** Результати магістерської роботи впроваджено:

– при виконанні міжнародного проекту ERASMUS+ ALIOT, “Internet of Things: Emerging Curriculum for Industry and Human Applications” - при розробленні курсів з аналізу даних (акт впровадження від 15.01.2018 р.).

**Ключові слова:** система локомотивної безпеки, візуалізація даних, алгоритм, спрощення, полігональний ланцюг, графік

# РОЗДІЛ 1

## ТЕОРЕТИЧНІ АСПЕКТИ ПРОБЛЕМИ ВІЗУАЛІЗАЦІЇ ВЕЛИКИХ ДАНИХ

### 1.1 Огляд проблем візуалізації великих даних

Стан будь-якого технологічного об'єкту, що оснащений датчиками з яких постійно знімається інформація може бути описаний послідовностями їх значень, які в свою чергу характеризуються і можуть бути представлені точками, лініями, кривими або поверхнями.

Графічне відображення - це дуже простий та дієвий спосіб візуалізації даних, що дозволяє полегшити сприйняття даних та надає достатньо інформації для аналізу і прийняття рішень. [1].

До методів візуалізації даних належать: таблиці, гістограми, точкові діаграми, лінійні діаграми, кругові діаграми, стрижневі діаграми.

В рамках дослідження, приділяючи увагу властивостям наборів даних, можна виділити наступні проблеми візуалізації:[2,3]

- Візуальний шум.
- Сприйняття великих зображень.
- Надмірне спрощення даних
- Математичні обмеження алгоритмів спрощення даних

#### 1.1.1 Візуальний шум

Звичайна візуалізація цілого ряду даних може створити повний безлад на екрані, і ми побачимо тільки одну велику пляму, що складається з точок, що представляють кожен рядок даних. Ця проблема пов'язана з тим, що більшість об'єктів в наборі даних, занадто пов'язані один з одним, і на екрані спостерігач не може розділити їх у вигляді окремих об'єктів. Так, іноді, аналізуючи складно отримати навіть трохи корисної інформації від всього набору візуалізованих даних без будь-якої додаткової обробки інформації.[2]

Візуальний шум – втрата видимості даних внаслідок накладання графічних примітивів. У загальному випадку проблема візуального шуму може бути вирішена шляхом збільшення розмірів компоненти відображення, але у випадку візуалізації великих наборів даних такий підхід може призвести до появи проблема сприйняття великого зображення.

#### 1.1.2 Сприйняття великих зображень

Однієї із основних проблем візуалізації великих наборів даних є обмеження сприйняття занадто великих зображень. Існує певний рівень сприйняття людини для різних візуалізацій

даних. Незважаючи на те, що цей рівень для візуалізації графічних даних набагато вищий, порівняно з візуалізацією табличних даних, він має свої обмеження. І після досягнення цього рівня сприйняття, людина просто втрачає здатність отримувати будь-яку корисну інформацію з переважаного перегляду даних.

Всі способи візуалізації обмежуються роздільною здатністю пристрою, що відповідає за вивід візуалізації, тому існує межа кількості точок, які повинні відобразитися для кожної візуалізації. Звичайно, ми можемо замінити візуалізаційний пристрій більш сучасним або групою пристроїв для часткової візуалізації даних, що дозволяє нам представити більш детальне зображення з більшою кількістю точок даних, але навіть якщо ми зможемо повторити цей процес нескінченну кількість разів, ми зустрінемося з обмеженням людського сприйняття. Зі збільшенням об'ємів даних, що відображаються відразу, людина буде стикатися з труднощами в розумінні даних та їх аналізу. Тому можна стверджувати, що методи візуалізації даних обмежені не тільки співвідношенням сторін та роздільною здатністю пристрою, але й обмеженнями фізичного сприйняття.

### **1.1.3 Спрощення даних**

Для вирішення проблеми сприйняття великих зображень вихідні дані піддають передобробці задля виділення меншої за обсягом підмножини даних, що якісно відображає поведінку вихідного набору великих даних. Задача відображення великих обсягів даних на точкових та лінійних діаграмах може бути розглянута як окремий випадок задачі спрощення полігональних ланцюгів.

#### **1.1.3.1 Спрощення полігональних ланцюгів**

Застосування методів спрощення полігональних ланцюгів є важливою проблемою в контексті візуалізації великих обсягів даних. Суть алгоритмів спрощення полягає у зменшенні кількості точок шляхом видалення тривіальних точок, але без порушення істотної форми вихідної лінії.

Перше застосування алгоритми спрощення полігональних ланцюгів набули у картографії. На сьогоднішній день більшість навігаційних додатків використовують алгоритми спрощення ліній для зменшення об'єму карт та поліпшення швидкості виконання таких операцій як масштабування та прокрутка.

Ключовим моментом у використанні алгоритмів спрощення полігональних ланцюгів є вибір порога спрощення, що визначає силу спрощення. Саме від цього параметру залежить кількість точок та вид отриманої лінії.

У різних алгоритмах поріг спрощення визначається по-різному. Найпоширеніші варіації порогу спрощення:

1. Відстань від точки до лінії, утвореної сусідніми або крайніми точками.
2. Радіус, в межах якого видаляються усі точки
3. Номер точки: видаляється кожна N точка.
4. Площа, в межах якої залишається лише зазначена кількість точок.

Чим більшим є поріг спрощення тим менше рівень деталізації отриманої лінії та тим менше точок складають отриману лінію.

**Висновок.** Для знаходження оптимального значення порогу спрощення необхідно дати визначення факторам, що дають змогу вирахувати оптимальну кінцеву кількість точок.

### 1.1.3.2 Фактори, що впливають на необхідну силу спрощення полігонального ланцюга

Рівень деталізації спрощеної лінії тісно пов'язаний з порогом спрощення алгоритму спрощення. Поріг асоціюється із силою спрощення. Для вимірювання сили спрощення зазвичай використовують алгоритм знаходження середньої відстані зсуву.

- 1) Розподільна здатність дисплея ( $H_r \times V_r$ ), Розмір дисплея ( $D_s$ )

Чим більшу розподільну здатність має дисплей тим менше повинна бути сила спрощення і навпаки. При однаковій розподільній здатності: чим більший розмір (діагональ) дисплея – ти більший розмір пікселя.

- 2) Кількість пікселів на дюйм (PPI)

PPI є показником щільності розташування пікселів на дисплеї. PPI можна розрахувати за допомогою наступної формули:

$$PPI = \frac{\sqrt{H_r^2 + V_r^2}}{D_s}$$

Фізичний розмір пікселю ( $D_p$ ) можна розрахувати за допомогою наступної формули:

$$D_p = \frac{1}{PPI} \text{ (дюймів)}$$

- 3) Просторова розподільна здатність пікселя ( $SR_p$ )

Просторова розподільна здатність - число незалежних пікселів значень на дюйм. Цей показник головним чином впливає на здатність розрізнити деталі на лінії. Формула для обчислення просторової розподільної здатності:

$$SR_p = \frac{D_p}{PPI * S}$$

### 1.1.3.3 Розрахунок сили спрощення полігонального ланцюга

Для отримання знаходження сили спрощення у класичному випадку використовується формула:

$$D_{simpl} = SR_p * N_e ,$$

де  $N$  – мінімальна кількість пікселів, що зможе розлічити людське око на певній відстані від монітору. Оскільки за технічним завданням має бути можливість зміни пропорцій графіків відносно один одного за допомогою компоненти під назвою «сплітер», то  $N$  помножимо на співвідношення поточної висоти компоненти відображення графіків до висоти графіка, для якого робимо обчислення. Формула прийме наступний вид:

$$D_{simpl} = SR_p * N_e * \frac{H}{H_{plot}}$$

### 1.1.4 Математичні обмеження алгоритмів спрощення полігональних ланцюгів

Проблема математичного обмеження алгоритмів спрощення полігональних ланцюгів є найбільш складною. Більшість алгоритмів спрощення передбачають наявність параметрів, що характеризують поріг спрощення. Різноманіття варіацій порогу спрощення призводить до необхідності детального аналізу характеру вхідних даних. Вибір алгоритму з невідповідним порогом спрощення може призвести до втрати значущих даних та нашкодити людині, що аналізує візуалізовані дані, до хибних висновків.

### 1.1.5 Оцінка якості спрощення полігональних ланцюгів методом знаходження позиційних похибок.

Спрощення полігонального ланцюга змінює його форму. Чим вище ступінь спрощення, тим більше спрощений ланцюг відрізняється від оригінального. Метод знаходження позиційних похибок - це спосіб вимірювання похибки спрощення, що полягає в знаходженні відстаней між вихідними точками, що були спрощені, та спрощеним ланцюгом.

**Висновок.** Для отримання найбільш збалансованого відношення рівня деталізації до кількості точок утвореної лінії необхідно розрахувати силу спрощення полігонального ланцюга. Розрахунки необхідно проводити з урахуванням характеристик дисплею . У рамках поставленої задачі баланс між деталізацією та мінімальністю точок був зсунутий у бік зменшення сили спрощення та збільшення деталізації. Для оцінки застосовності алгоритмів спрощення необхідно розрахувати позиційні помилки.

## 1.2 Аналіз вимог щодо візуалізації даних системи локомотивної безпеки

Система локомотивної безпеки СЛБ-І – це система, організована за допомогою сучасних високотехнологічних пристроїв, що призначена для забезпечення безпеки руху в поїзній і маневровій роботі тягового і моторвагонного рухомого складу (далі – МВРС). СЛБ-І може застосовуватися на локомотивах та МВРС з автономною тягою, електротягою постійного струму та змінного струму. СЛБ-І може застосовуватися на локомотивах та МВРС, що експлуатуються на залізницях із кодуванням рейкових кіл сигналами автоматичної локомотивної сигналізації числового коду безперервного типу.[4]

СЛБ-І складається з наступних технічних засобів автоматизації (далі – ТЗА):[4]

- датчики й устаткування для автостопного і службового гальмування;
- обладнання для прийому й обробки вхідної інформації та формування управляючих команд;
- операторське обладнання;
- сервісне та діагностичне обладнання;
- блоки живлення.

*Згідно технічної документації [4], до складу СЛБ-І входять наступні датчики і устаткування з якого виконується збір сигналів і які мають архівуватися:[4]*

- датчики тиску, призначені для вимірювання тиску у двох вирівнювальних резервуарах (встановлюються по одному датчику у кожній кабіні), гальмових циліндрах (два датчика, загальні для двох кабін), гальмовій магістралі (два датчика, загальні для двох кабін), живильній магістралі (один датчик, загальний для двох кабін); датчик тиску, що визначає спрацьовування локомотивного гальма (один датчик, загальний для двох кабін), один резервний датчик, максимальна кількість датчиків для однокабінної конфігурації – 8 шт., максимальна кількість датчиків для двохкабінної конфігурації – 9 шт.;
- котушки приймальні КП-И 684459.001 (далі – КП-И), призначені для прийому сигналів АЛСН числового коду (встановлюються по дві котушки для кожної кабіни);
- датчик шляху і швидкості ДПС-И 468165.001 або датчик кута повороту (далі для обох–ДКП) типу ДПС-У, призначені для визначення швидкості руху і пройденого шляху, в кількості 2 шт.;
- одна антена для прийому сигналів цифрового радіоканалу і одна антена для прийому сигналів супутникових навігаційних систем, які встановлюються на даху кабіни локомотива;
- електропневмоклапани типу ЭПК-153 (ЭПК-150), призначені для виконання автостопного гальмування (встановлюються по 1 шт. для кожної кабіни);

- електропневмовентилі типу ЭВ-58-07 або аналогічні, призначені для виконання автостопного гальмування при відключенні електропневмоклапану ключем (встановлюються по 1 шт. для кожної кабіни);
- пристрій перетворення і обробки інформації УПО-1 468222.027 або УПО-2 468222.028 (далі – УПО), до складу якого входять наступні блоки:
  1. три блоки прийому аналогових сигналів БПАС-1 468151.099 для прийому і обробки сигналів від датчиків тиску, котушок приймальних АЛСН;
  2. три блоки управління БУ-514 468332.367 для прийому дискретних сигналів, сигналів ДКП, обробки інформації, формування управляючих даних, зберігання локомотивних і поїзних характеристик;
  3. два (або чотири) блоки формування дискретних сигналів БФДС-10 468172.033. Формування вихідних сигналів в кожному блоці проводиться двома мікропроцесорними каналами за схемою «2 з 2» з обробкою управляючих даних у кожному мікропроцесорі за схемою «2 з 3Р»;
  4. блоки зв'язку БСв-255 467141.076 по цифровим послідовним магістралям CAN, RS-485, MVB і приймання сигналів супутникової навігаційної системи і сигналів, переданих по послідовному інтерфейсу радіостанції, перетворення сигналів ДКП у відповідний формат для видачі до мікропроцесорної системи управління і діагностики ;
- блок реєстрації локомотивний БРЛ-1 467239.018, який повинен забезпечувати запис у встановлену касету реєстрації КР-1 467379.001 параметрів руху локомотива та результатів діагностики. Додатково БРЛ-1 виконує запис у внутрішню енергонезалежну пам'ять аналогічної інформації, що дозволяє скопіювати архівну інформацію на сервісну КР-1 і виконати її розшифрування за відсутності КР-1 з архівною інформацією. Це, наприклад, полегшує доступ обслуговуючому персоналу до діагностичних даних для пошуку і встановлення причин збійних ситуацій і виконання ремонту;

Максимальний розмір даних архіву однієї поїздки на касеті реєстрації КР-1 може сягати 700 мегабайт.

### **1.3 Постановка наукової задачі та обґрунтування методики досліджень**

Результати проведеного аналізу нормативної бази, моделей, методів й інструментальних засобів візуалізації великих наборів даних, показали, що у відомих публікаціях і нормативно-технічних документах процеси формування та візуалізації великих наборів даних розглядаються як різномірні технічні задачі без загального математичного апарата.



Проте, комплексний вплив різномірних параметрів на працездатність системи локомотивної безпеки СЛБ-І (з одного боку); і необхідність комплексної оцінки їх впливу з метою пошуку можливих несправностей (порушень) СЛБ-І обумовлюють необхідність розроблення і впровадження ефективних технологій візуалізації великих наборів архівних даних, здатних використовувати спрощення полігональних ланцюгів із збереженням точності даних та з урахуванням їх пріоритетів.

Важливою, раніше не розглянутою задачею, є розробка метода вибору алгоритму спрощення полігональних ланцюгів та динамічного визначення сили спрощення в залежності від типу і характеру вхідних даних. У даній роботі запропоновано вирішити задачу вибору алгоритму за допомогою методу знаходження позиційних похибок.

Різноманітність шляхів визначення порогу спрощення у різних алгоритмах та наявність пріоритетних точок у архівних даних обумовлює необхідність порівняльного аналізу існуючих алгоритмів спрощення полігональних ланцюгів, та розробку нового алгоритму, на базі існуючих, що має враховувати наявність пріоритетних даних.

#### **1.4 Практичні задачі роботи:**

Компонента візуалізації даних має реалізувати наступні функції:

- відображення оператору архівної інформації в графічному вигляді;
- відображення графіка значень АЛСН;
- відображення порушень в архівних даних;
- відображення даних ЕК із прив'язкою до архівної інформації;

Для зручної можливості аналізу параметрів архівних даних для діагностики і визначення причин несправностей та аварійних ситуацій необхідно надати можливість відображення параметрів в залежності від типу на чотирьох графіках:

1. Дискретні параметри
2. Аналогові параметри
3. Параметри швидкості та обмеження
4. Сигнал світлофору та об'єкти з електронної карти

Для дискретних та аналогових параметрів необхідно надати можливість відображення у групах до 20 параметрів.

Максимально необхідна кількість відображених параметрів 50, серед яких:

1. Дискретних параметрів на графіку №1 – до 20.
2. Аналогових параметрів на графіку №2 – до 20.
3. Параметрів швидкості на графіку №3 – 3, та 6 параметрів обмеження.
4. Параметр сигналу світлофора.

По осі X відобразити дату і час з точністю до секунди та значення лінійної координати у кілометр-пікетах. По осі Y відобразити значення відповідного параметра у заданій величині.

Достовірні значення параметрів відобразити суцільною лінією.

У разі недостовірності даних часу – відобразити розрив на графіку та позначити початок розриву спеціальним символом.

У разі недостовірності значення параметрів – відобразити недостовірний проміжок пунктирною лінією і позначити початок проміжку недостовірних даних спеціальним символом.

Необхідно надати можливість перегляду даних за максимально можливою за тривалістю поїздки(3 доби).

Має бути реалізована можливість навігації по графікам шляхом прокрутки та масштабування.

На загальному виді (максимально за 3 доби) мають бути виділені відповідним символом всі місця, на які необхідно звернути увагу, такі як:

1. Збої по живленню.
2. Недостовірність або невідповідність значення дати і часу.
3. Недостовірні значення параметрів.
4. Аварійні ситуації

Методика досліджень розробляється виходячи з вимог до логіки послідовності рішення поставлених задач і адекватності обраного математичного апарату. З урахуванням цього методика досліджень включає наступні етапи:

- аналіз сформульованих задач і визначення їх логічного взаємозв'язку, виходячи із загальної задачі магістерської роботи і запланованих результатів;
- визначення послідовності рішення наукових і прикладних задач;
- обґрунтування і вибір математичного апарату для рішення задач, виходячи з їх особливостей;
- рішення задач, аналіз отриманих результатів на несуперечність, розроблення рекомендацій щодо їх провадження і подальшому розвитку.

Для проведення досліджень доцільно застосовувати методи системного аналізу, векторного аналізу, комбінаторного аналізу.

## **1.5 Висновки**

1. Максимальна тривалість поїздки та максимально необхідна кількість відображених параметрів обумовлюють необхідність аналізу існуючих методів зменшення кількості точок на

доцільність застосування до спрощення архівних даних та оцінки результатів спрощення методом позиційної похибки.

2. Сформульована загальна науково-технічна задача дослідження, як задача розроблення способу розбиття вхідних наборів даних, для спрощення, що задовольнить вимогам точності відображення даних з урахуванням пріоритетів.

## **1.6 Література до розділу 1**

[1] E. Gorodov, V. Gubarev, Analytical Review of Data Visualization Methods in Application to Big Data doi:10.1155/2013/969458

[2] Lidong Wang, Guanghui Wang, Cheryl Ann Alexander. Big Data and Visualization: Methods, Challenges and Technology Progress, Vol. 1, No. 1, 2015, pp 33-38. doi: 10.12691/dt-1-1-7

[3] Jen Underwood, Data Visualization. Best Practices.

[4] 421413.024 РЭ СИСТЕМА ЛОКОМОТИВНОЇ БЕЗПЕКИ СЛБ-И

## РОЗДІЛ 2

### АНАЛІЗ АЛГОРИТМІВ СПРОЩЕННЯ ПОЛІГОНАЛЬНИХ ЛАНЦЮГІВ

#### 2.1 Вибір алгоритмів спрощення полігональних ланцюгів

Алгоритми спрощення можна поділити на три категорії в залежності від способу визначення порогу спрощення:

1. Алгоритми вибору точок
2. Алгоритми локальної обробки
3. Алгоритми обробки секцій

Алгоритми вибору точок не враховують математичні відношення обраної точки із сусідніми точками. Прикладами алгоритмів спрощення, що можна віднести до цієї категорії є алгоритм випадкового вибору точок та алгоритм вибору кожної  $N$  точки. До порівняння не були обрані алгоритми з цієї категорії, бо очевидно, що дані алгоритми мають погану точність.

Алгоритми локальної обробки здійснюють вибір точок урахувуючи математичні відношення поточної точки із попередньою або двома сусідніми точками. До цієї категорії можна віднести алгоритм радіального виключення точок та алгоритм виключення за висотою трикутника.

Алгоритми обробки секції здійснюють вибір точок серед підмножини точок, урахувуючи математичні відношення точок у поточному сегменті.

Для аналізу були обрані наступні алгоритми:

1. Алгоритм Рамера-Дугласа-Пекера [1]
2. Алгоритм Реумана-Віткама [2]
3. Алгоритм Опхейма [3]
4. Алгоритм Ланга [4]
5. Алгоритм виключення за висотою трикутника [5]
6. Алгоритм радіального виключення точок [6]

##### 2.1.1 Алгоритм Дугласа-Рамера-Пекера

Алгоритм Рамера-Дугласа-Пекера - Алгоритм був незалежно відкритий Урсом Рамером в 1972 та Давидом Дугласом і Томасом Пекером в 1973. Алгоритм визначає розбіжність, яка обчислюється по максимальній відстані між вихідною і спрощеною кривими. Спрощена крива складається з підмножини точок, які визначаються з початкової кривої.[1]. Початкова крива являє собою упорядкований набір точок або ліній, і задану відстань  $\epsilon > 0$ .

Алгоритм рекурсивно ділить лінію. Входом алгоритму служать координати всіх точок між першою і останньою. Перша і остання точка зберігаються незмінними. Після чого алгоритм

знаходить точку, найбільш віддалену від відрізка, що з'єднає першу і останню. Якщо точка знаходиться на відстані, меншій  $\epsilon$ , то всі точки, які ще не були відзначені до збереження, можуть бути викинуті з набору і отримана пряма згладжує криву з точністю не нижче  $\epsilon$ .

Якщо ж відстань більше  $\epsilon$ , то алгоритм рекурсивно викликає себе на наборі від початкової до даної і від даної до кінцевої точках (що означає, що дана точка буде відзначена до збереження). По закінченню всіх рекурсивних викликів вихідна ламана будується тільки з тих точок, що були відзначені до збереження.[7] Схематична дія алгоритму зображена на рис 2.1.

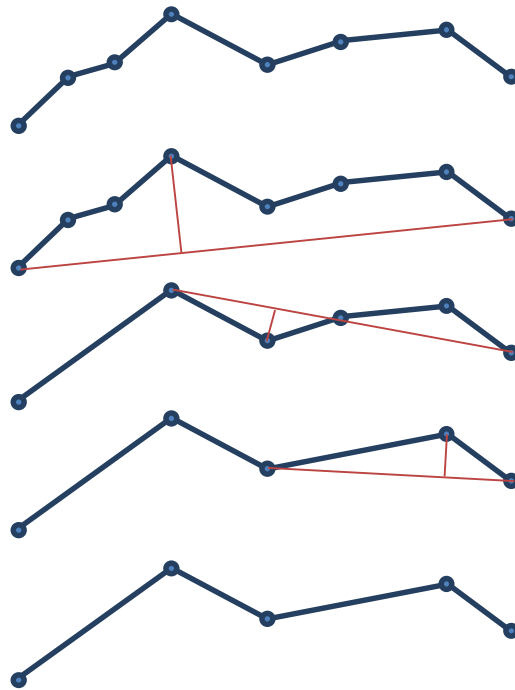


Рис 2.1 Згладжування кривої алгоритмом Рамера-Дугласа-Пекера

### 2.1.2 Алгоритм Реумана-Віткама

Алгоритм Реумана-Віткама (Reumann-Witkam) визначає пряму через перші дві точки ланцюга, остання із послідовних точок починаючи із другої, віддалених не більше ніж на  $\epsilon$  з'єднуються прямою, а всі проміжні точки виключаються. Алгоритм продовжується до тих пір, поки не буде досягнута остання точка. На рис 2.2 зображений принцип роботи алгоритма Реумаа-Віткама: пряма, що проходить через точки відображена лінією, що оточена границями, у які попадають вершини, що будуть спрощені, відображені пунктирними лініями. [8]

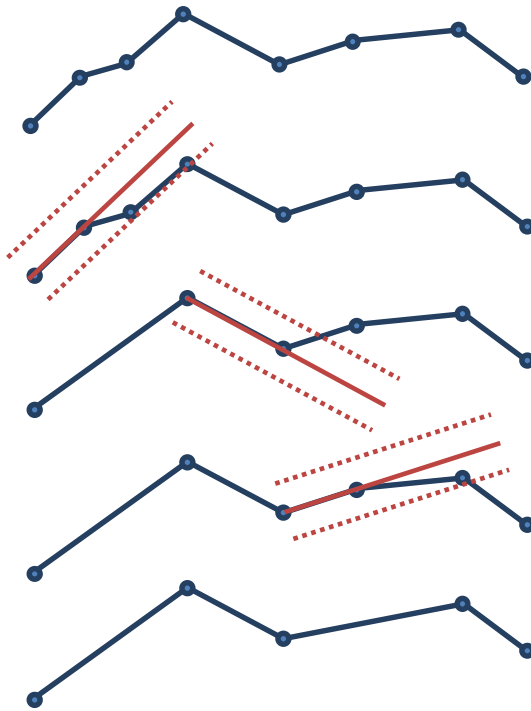


Рис 2.2 Принцип дії алгоритму Реумана-Віткама

### 2.1.3 Алгоритм Опхейма

Алгоритм Опхейма за своєю реалізацією схожий на алгоритм Реумана-Віткама [8]. Різниця полягає в тому, що в даному випадку ми розглядаємо всі вершини в радіусі  $\epsilon$  від першої, і будуємо промінь від поточної до останньої, що потрапила у радіус. Якщо таких точок немає – беремо наступну точку в якості поточної. Наступні вершини спрощуються до тих пір, поки відстань до променя більше ніж  $\epsilon$  і радіальна відстань до першої точки більше ніж  $\epsilon$ . Алгоритм продовжується, доти не буде досягнута остання точка. На рис 2.3 зображений принцип дії алгоритму Опхейма.

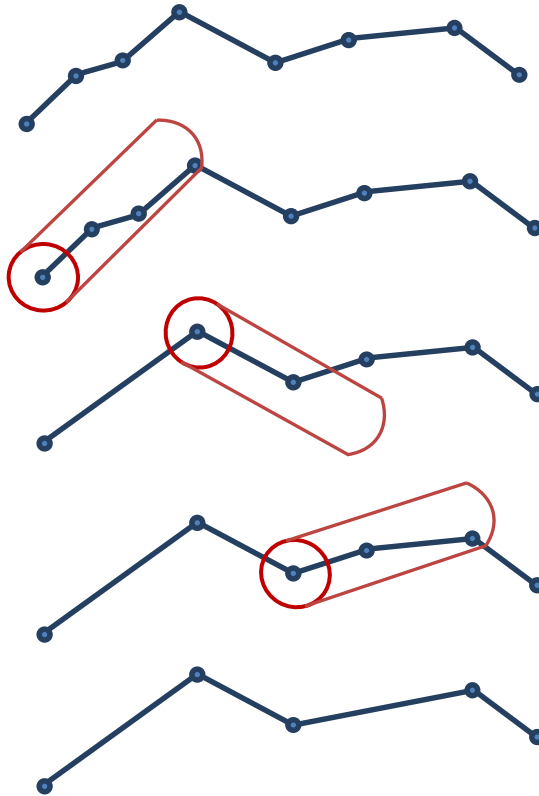


Рис 2.3 Принцип дії алгоритму Опхейма

#### 2.1.4 Алгоритм Ланга

Алгоритм Ланга визначає фіксований розмір пошукової області для спрощення [8]. Дві точки, що утворюють пошукову область, складають відрізок. Цей відрізок використовується для обчислення перпендикулярної відстані до кожної проміжної точки. Якщо розрахована відстань більше заданої  $\epsilon$ , або коли проміжні точки відсутні. Всі проміжні точки видаляються, а нова пошукова ділянка визначається, починаючи із останньої точки попередньої області. На рис 2.4 зображена схема дії алгоритму Ланга.[4]

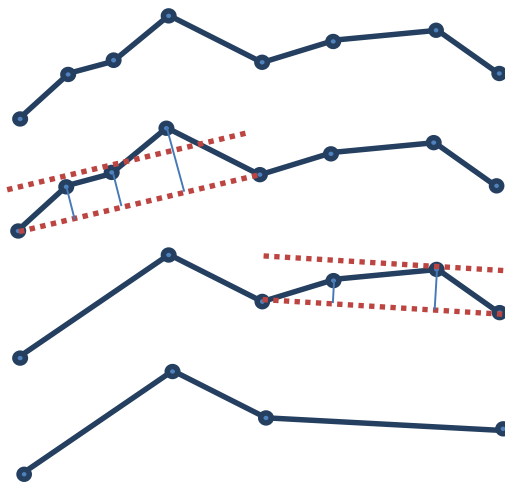


Рис 2.4 Принцип дії алгоритму Ланга

### 2.1.5 Алгоритм виключення за висотою трикутника

Для кожної точки  $V[i]$  розраховується відстань до лінії, утвореної точками  $V[i-1]$  і  $V[i+1]$ . Всі вершини, відстань яких менше ніж  $\epsilon$  видаляються. На рис 2.5 зображена схема дії алгоритму виключення за висотою трикутника.

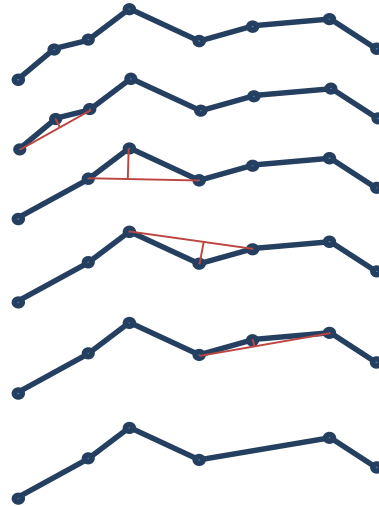


Рис 2.5 Принцип дії алгоритму виключення за висотою трикутника

### 2.1.6 Алгоритм радіального виключення

Алгоритм виключає точки що знаходяться в радіусі  $\epsilon$  від поточної точки. Схема дії алгоритму радіального виключення точок зображена на рис 2.6.

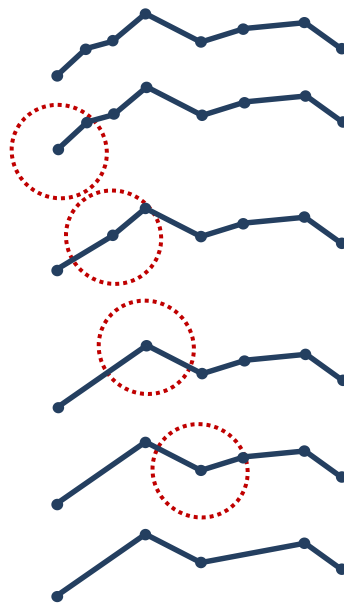


Рис 2.6 Принцип дії алгоритму радіального виключення точок



## 2.2 Вибір порогу спрощення

За умовами технічного завдання необхідно надати можливість масштабування компоненти відображення. Для ефективного спрощення даних при масштабуванні необхідно змінювати величину порога спрощення відповідно до масштабу. При виборі порога спрощення необхідно врахувати розміри графічної сцени в пікселях.

Для алгоритмів, що відносяться до категорії локальної обробки, поріг спрощення характеризується однією величиною  $t$  (tolerance), що є відношенням між поточною точкою та сусідніми. Поріг спрощення ( $t$ ) обрахуємо за формулою.

$$t = \sqrt{\left(\frac{|y_{max}-y_{min}|}{H}\right)^2 + \left(\frac{|x_{max}-x_{min}|}{W}\right)^2}, \text{ де}$$

$H$  – висота графічної сцени в пікселях

$W$  – ширина графічної сцени в пікселях

$y_{max}$  – верхня межа діапазону значень осі значень.

$y_{min}$  – нижня межа діапазону значень осі значень.

$x_{max}$  – верхня межа діапазону значень осі часу.

$x_{min}$  – нижня межа діапазону значень осі часу.

Для алгоритмів, що відносяться до категорії обробки секцій, поріг спрощення характеризується двома величинами: відношенням між поточною точкою та сегментом (tolerance), і довжиною сегмента.

Довжину сегмента обрахуємо за формулою

$$l = 0.05 * |x_{max} - x_{min}|$$

Tolerance обрахуємо за наступною формулою

$$t = \frac{|y_{max}-y_{min}|}{H}$$

## 2.3 Порівняльний аналіз алгоритмів спрощення

Порівняння ефективності алгоритмів спрощення складається із двох етапів: візуального аналізу та розрахунку відхилень. Візуальний аналіз дасть змогу оцінити на скільки спрощений ланцюг передає поведінку(форму) початкового ланцюга. Розрахунок відхилень дозволить обрати з поміж алгоритмів, що однаково добре пройшли візуальний аналіз, алгоритм з найменшим значенням позиційної похибки.

Для спрощення було обрано значення середньо-квадратичного відхилення сигналу АЛСБ[9]. Координати точок: по осі X – дата і час в мілісекундах з початку епохи(1970), по осі Y – значення середньоквадратичного відхилення сигналу АЛСБ.

### 2.2.1 Візуальний аналіз

На рисунках 2.8 – 2.13 наведено результат застосування алгоритмів спрощення. Синім кольором позначено початковий ланцюг. Зеленим кольором позначено спрощений ланцюг. Для більш наглядного порівняння розглядаємо один і той самий участок ланцюга даних тривалістю 10 секунд.

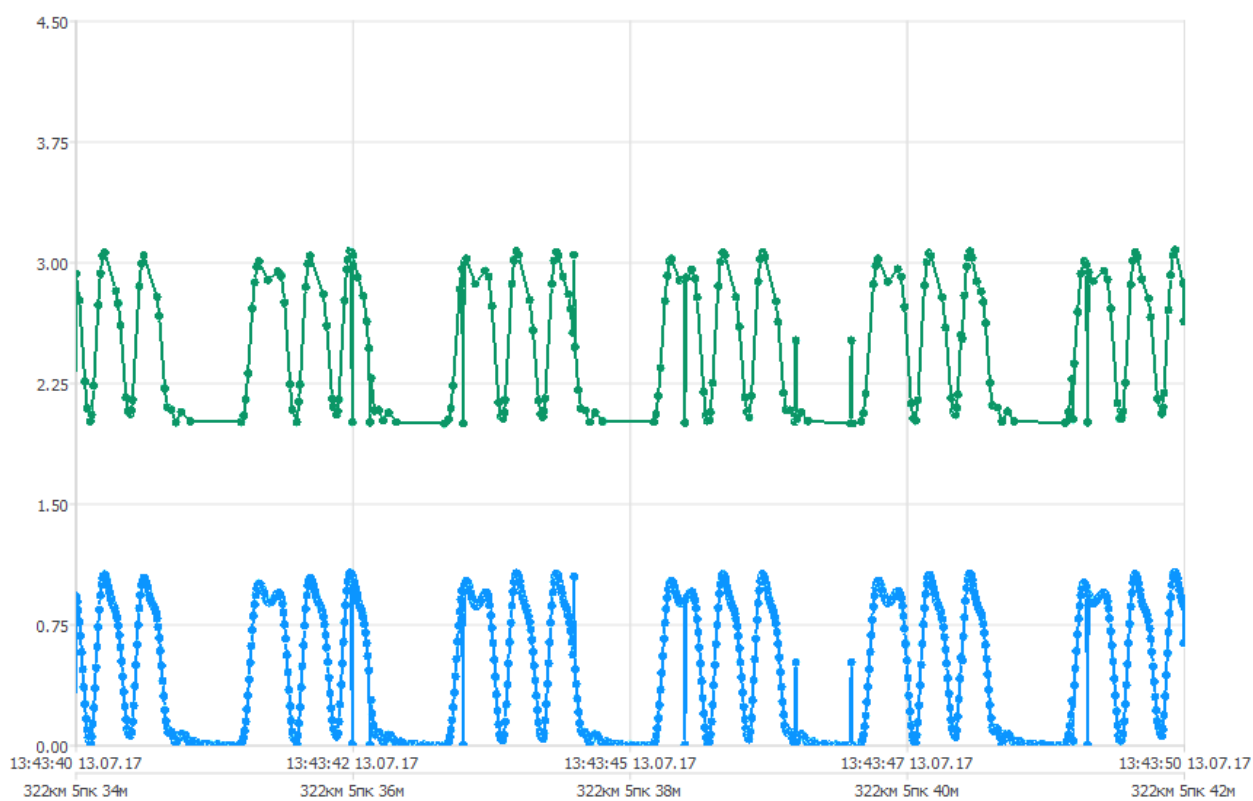


Рисунок 2.8 Результат спрощення алгоритмом Дугласа-Рамера-Пекера

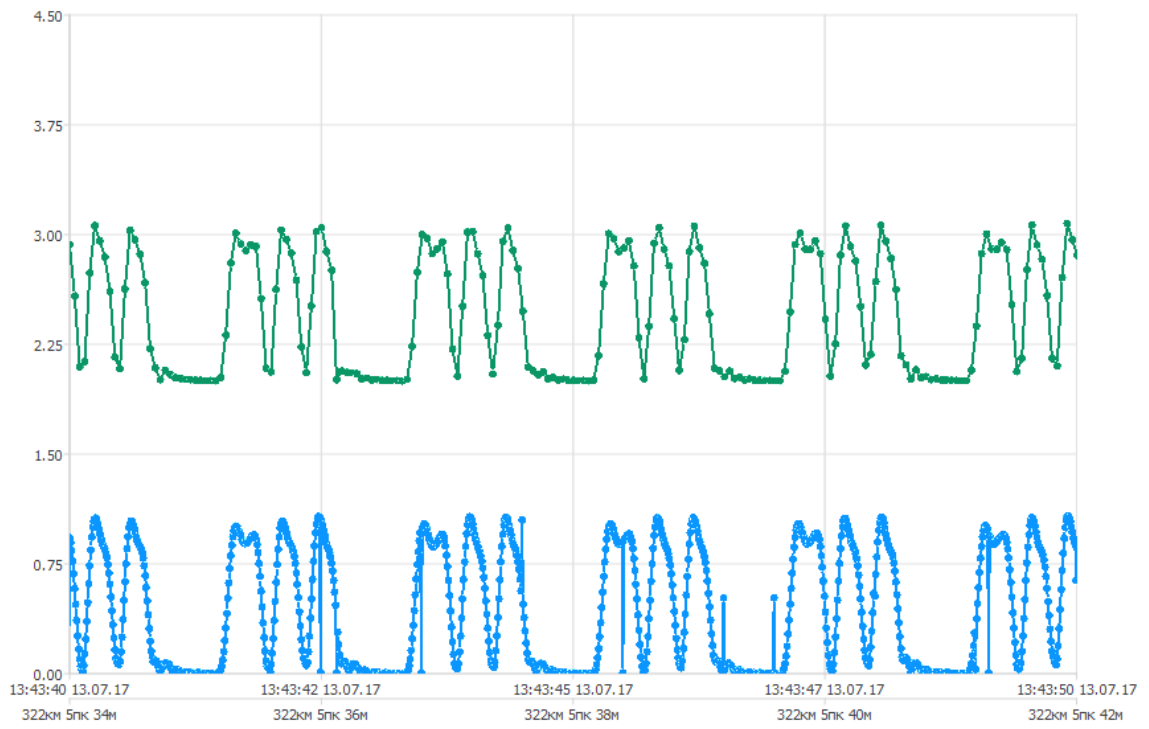


Рисунок 2.9 Результат спрощення алгоритмом радіального виключення точок

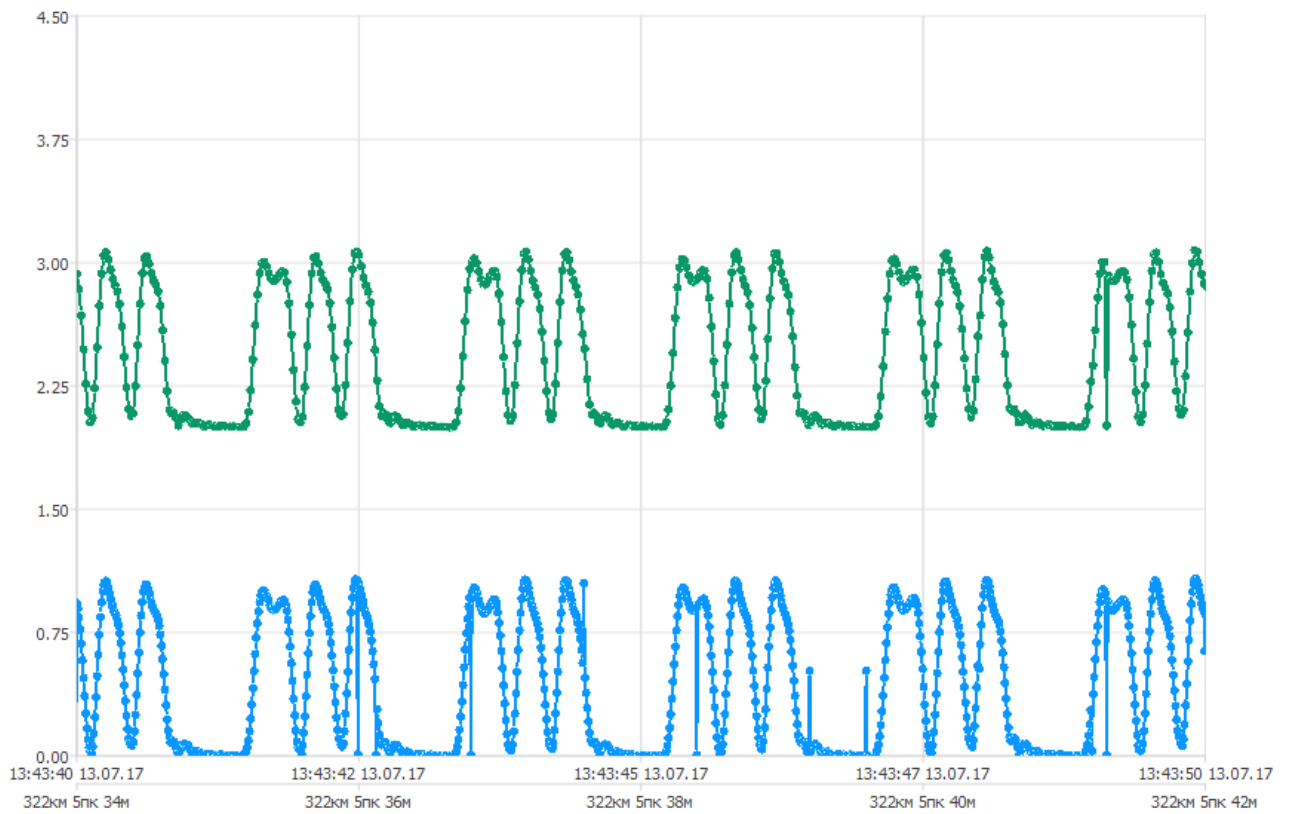


Рисунок 2.10 Результат спрощення алгоритмом виключення за висотою трикутника

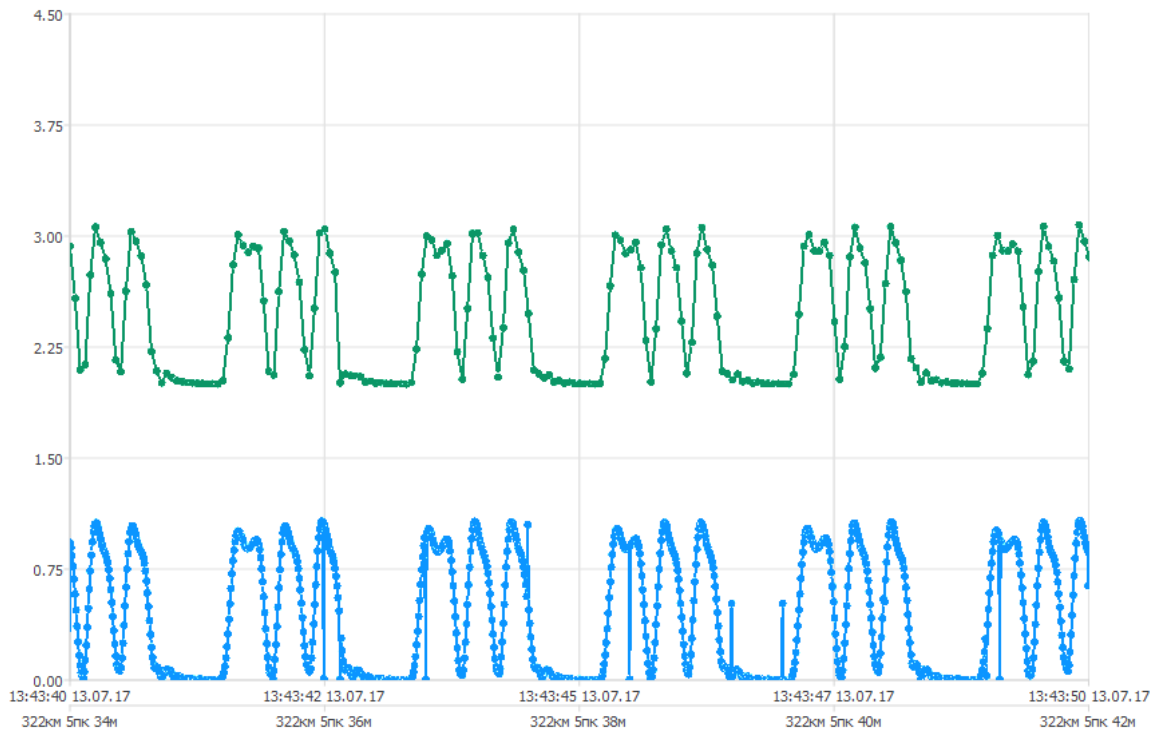


Рисунок 2.11 - Спрощення алгоритмом Ланга

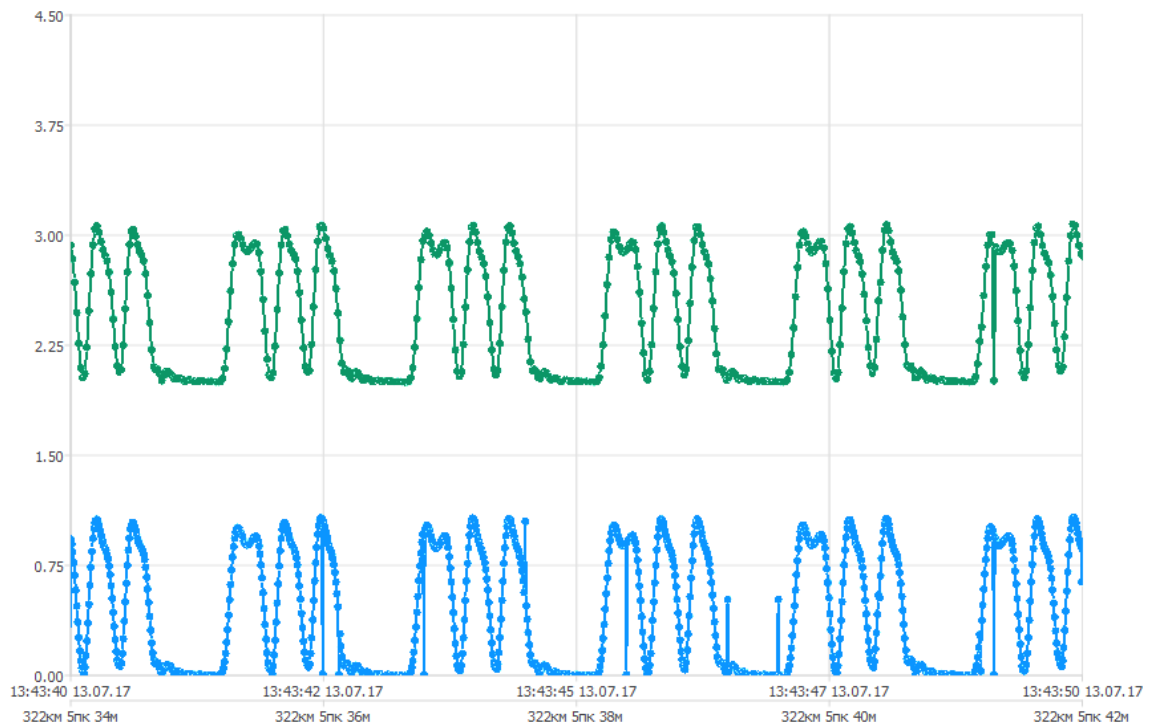


Рисунок 2.12 - Спрощення алгоритмом Опхейма

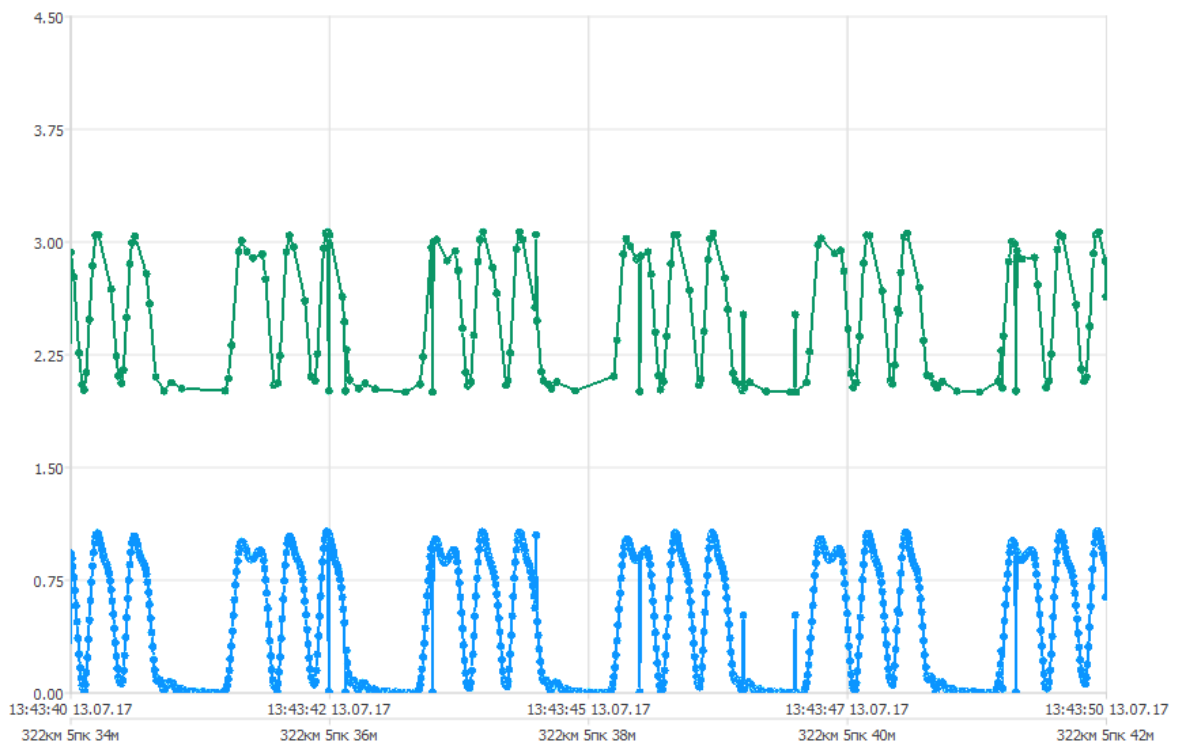


Рисунок 2.13 - Спрощення алгоритмом Реумана-Віткама

За результатами візуального аналізу для спрощення ланцюга сигналу АЛСН доцільно використовувати алгоритми: Реумана-Віткама, алгоритм Радіального виключення та алгоритм Дугласа-Рамера-Пекера. Обрані алгоритми найбільш точно відображають загальну поведінку ланцюга.

### 2.3.1 Математичний аналіз

Для порівняння обраних алгоритмів спрощення полігональних ланцюгів обчислимо позиційні відхилення.

Значення відхилення обчислюється окремо для кожної точки, що була спрощена і дорівнює відстані від точки що була спрощена до спрощеного ланцюга.

Зрозуміло, що спрощення полігонального ланцюга змінює його форму. Чим вище ступінь спрощення, тим більше спрощений ланцюг відрізняється від оригінального. Для оцінки застосовності алгоритмів спрощення до мети роботи необхідно розрахувати позиційні помилки. Спосіб вимірювання відхилення спрощення полягає в знаходженні відстаней між вихідними точками, що були спрощені, та спрощеним ланцюгом.

Найбільш часто використовуваними показниками для визначення якості спрощення є [10]:

- максимальне відхилення (максимальна відстань від оригінальної точки, що була спрощена, до спрощеної лінії)

$$\max_{i=1,N}(p_i - p_i^m); \quad (3)$$

- мінімальне відхилення (максимальна відстань від оригінальної точки, що була спрощена, до спрощеної лінії)

$$\min_{i=1,N}(p_i - p_i^m); \quad (4)$$

- середньоквадратичне відхилення

$$\sigma = \sqrt{\frac{\sum_{i=1}^N (p_i - p_i^m)^2}{N}}; \quad (5)$$

- сума відхилень (сума відстаней від оригінальних точок, що були спрощені, до спрощеної лінії)

$$S = \sum_{i=1}^N (p_i - p_i^m). \quad (6)$$

Для порівняльного аналізу було обрано ланцюг, що має 15425 точки. Результати обчислення позиційних відхилень наведено у таблиці 2.1.

Таблиця 2.1 Результати порівняння алгоритмів спрощення

Спрощення даних сигналу АЛСН						
	Дугласа-Рамера-Пекара	Спрощення методом радіального виключення	Спрощення за висотою трикутника	Ланга	Опхейма	Реумана-Віткама
Максимальне відхилення	0.0224553	1.06187	1.06422	1.06187	1.06422	0.126953
Середнє відхилення	0.00532108	0.0228263	0.00839531	0.0230799	0.00835633	0.0185176
Середньоквадратичне відхилення	0.00550363	0.0690616	0.0648543	0.069663	0.0666799	0.0198078
Сума відхилень	82.0777	352.096	129.498	356.007	128.896	285.634
Кількість точок після спрощення	3856	3090	7713	3086	7726	2867

## **2.4 Висновки до розділу 2**

Було проведено порівняльний аналіз алгоритмів спрощення полігональних ланцюгів. Порівняльний аналіз складався із двох етапів: візуальної оцінки схожості спрощеного ланцюга до початкового та розрахунку позиційних відхилень. Результати візуального аналізу найкращими алгоритмами виявились алгоритми Дугласа-Рамера-Пекера та Реумана-Віткама, бо їх результат спрощення найточніше відтворює поведінку початкового ланцюга.

## Література до розділу 2

- [1] Douglas, D. and Peucker, T. “Algorithms for the reduction of the number of points required to represent a digitized line or its caricature”, Canadian Cartographer, 10, 112–122.
- [2] Reumann, K. and Witkam, A. P. M.. ‘Optimizing curve segmentation in computer graphics’, in Proceedings of International Computing Symposium, pp. 467–472.
- [3] Opheim, H.. ‘Fast data reduction of a digitized curve’, GeoProcessing, 2, 33–40.
- [4] Lang, T. (1969) ‘Rules for robot draughtsmen’, Geographical Magazine, 42, 50–51.
- [5] Алгоритм радіального виключення точок [Електронний ресурс] // Режим доступу www. URL <http://psimpl.sourceforge.net/radial-distance.html>
- [6] Алгоритм радіального виключення точок [Електронний ресурс] // Режим доступу www. URL <http://psimpl.sourceforge.net/radial-distance.html>
- [7] Алгоритм Дугласа-Рамера-Пекера [Електронний ресурс] // Режим доступу www. URL [https://en.wikipedia.org/wiki/Ramer%E2%80%93Douglas%E2%80%93Peucker\\_algorithm](https://en.wikipedia.org/wiki/Ramer%E2%80%93Douglas%E2%80%93Peucker_algorithm)
- [8] Спрощення полігональних ланцюгів [Електронний ресурс] // Режим доступу www. URL [https://neerc.ifmo.ru/wiki/index.php?title=Упрощение\\_полигональной\\_цепи](https://neerc.ifmo.ru/wiki/index.php?title=Упрощение_полигональной_цепи)
- [9] Автоматична локомотивна сигналізація [Електронний ресурс] // Режим доступу www. URL [https://uk.wikipedia.org/wiki/Автоматична\\_локомотивна\\_сигналізація](https://uk.wikipedia.org/wiki/Автоматична_локомотивна_сигналізація)
- [10] Ekdemir S. “Efficient Implementation of Polyline Simplification for Large Datasets and Usability Evaluation”, [Електронний ресурс] // Режим доступу www. URL <http://www.diva-portal.org/smash/get/diva2:444686/FULLTEXT01.pdf>



## РОЗДІЛ 3

### ПРАКТИЧНА РЕАЛІЗАЦІЯ ПРОГРАМНИХ КОМПОНЕНТІВ ВІЗУАЛІЗАЦІЇ ДАНИХ

#### 3.1 Функціональне призначення ПЗ «Дешифратор»

Програмне забезпечення «Дешифратор» призначена для виконання наступних функцій:

- читання архівних даних з касети реєстрації;
- ведення архіву інформації, що зчитана з касети реєстрації (далі – архівні дані);
- відображення оператора архівної інформації в графічному та текстовому вигляді;
- відображення графіка значень автоматичної локомотивної сигналізації безперервної дії;
- відображення порушень в архівних даних;
- відображення даних електронної карти із прив'язкою до архівних даних;
- формування результуючого протоколу поїздки із встановленою виною машиніста та його друкування;
- ведення журналів поїздок (за типами журналів);
- автоматичного видалення поїздок і протоколів, строк зберігання яких минув;
- ведення бази даних машиністів;
- ведення бази даних маршрутів;
- ведення бази даних локомотивів;
- ведення бази даних депо.

Програмне забезпечення «Дешифратор» включає в себе наступні графічні компоненти:

- Компонента зчитування архівних даних з касети реєстрації і ведення архіву – виконує зчитування архівних даних з касети реєстрації
- Компонента візуалізації архівних даних – відображення інформації в графічному виді, відображення порушень в архівних даних, відображення даних електронної карти, відображення тижневих обмежень
- Компонента відображення графіка АЛСБ - відображення графіка значень автоматичної локомотивної сигналізації безперервної дії
- Компонента керування базою даних маршрутів
- Компонента керування даних машиністів
- Компонента керування базою даних локомотивів
- Компонента керування базою даних депо
- Компонента автоматичного видалення поїздок і протоколів

- Компонента формування результуючого протоколу і встановлення вини машиніста
- Компонента авторизації користувача
- Компонента відображення табличних значень параметрів
- Компонента налаштувань програмного додатку – встановлення директорій для журналів, архіву із зчитаними поїдками, баз даних, тощо

### 3.2 Формалізація задачі програмного опису технологічних параметрів

Архівні дані містять більше ніж 700 параметрів. В залежності від джерела даних, типу параметра та алгоритму вибору значень всі параметри можна поділити на окремі групи. Для вирішення задачі формалізації необхідно розробити ієрархію класів, що дозволить отримати уніфіковані програмні інтерфейси для роботи з параметрами, що за своїми ознаками входять до певної групи.

#### 3.2.1 Розробка програмного інтерфейсу для роботи із значеннями параметрів архівних даних

Для вирішення задачі створення програмного інтерфейсу для роботи із даними технологічних параметрів було розроблено базовий абстрактний клас BaseDataParameter. Опис полів та методів класу наведений таблиці:

Таблиця 3.1 – Опис інтерфейсу класу BaseDataParameter

Поле/Метод	Опис
QString _name;	Поле, у якому зберігається скорочена назва параметра
QString _full_name;	Поле, у якому зберігається повна назва параметра
BaseDataParameter(QString name, QString f_name);	Конструктор класу[1]. Приймає 2 параметри: скорочене та повне імена параметру, що необхідні для ідентифікації параметра у графічному інтерфейсі користувача.
virtual bool state(ArchiveDataList_t *list_element)=0;	Віртуальний метод[2], що описує інтерфейс для перевірки достовірності значення технологічного параметра у заданому півсекундному інтервалі

virtual qreal value(ArchiveDataList_t *list_element)=0;	Віртуальний метод[2], що описує інтерфейс для отримання значення технологічного параметра у заданому півсекундному інтервалі
QString name() const	Метод для отримання короткої назви параметра
QString fullName() const	Метод для отримання повної назви параметра

### 3.3 Розробка програмних інтерфейсів для описання алгоритмів вибору значень

За технічним завданням в залежності від типу параметра визначають 4 групи параметрів, дані яких дублюються декількома пристроями. Для таких параметрів необхідно надати можливість ручного вибору значень. Алгоритм вибору значення має 2 режими вибору: автоматичний та ручний. Для автоматичного вибору значення використовується алгоритм, що урахує достовірність та значення для кожного дубльованого параметра. У режимі ручного вибору оператору надається можливість переглядати значення з окремих пристроїв. Для реалізації функціоналу вибору алгоритму формування значення доцільно створити інтерфейси відповідно до кожної групи параметрів.

#### 3.3.1 Розробка інтерфейсу для опису алгоритму вибору значень параметрів, що архівуються БУ(Блоками управління).

Таблиця 3.2 – Опис полів та методів інтерфейсу BU\_Base

Поле/Метод	Опис
BU_Base()	Конструктор класу за замовчуванням, виконує початкову ініціалізацію.
void set_mode(QString mode);	Метод для встановлення алгоритму вибору значення
unsigned char major_d(unsigned char Xstep, unsigned char Xval, unsigned char Ystep, unsigned char Yval, unsigned char Zstep, unsigned char Zval);	Функція мажорювання для вибору значення із урахуванням достовірності та значення дискретних параметрів із трьох джерел.
enum mode{Auto,BU1,BU2,BU3};	Доступні алгоритми вибору значення.
mode current_mode=Auto;	Поле, у якому зберігається поточний алгоритм вибору значення. При початковій

	ініціалізації використовується алгоритм автоматичного вибору значення.
QMap<QString,mode> string_to_mode_map;	Хеш масив [3], у якому зберігається відповідність строкових репрезентацій імен алгоритмів вибору значень та відповідних полей enum mode. Ініціалізація відбувається у конструкторі за замовчуванням BU_Base()
QList<QString> mode_names;	Список[4], у якому зберігаються імена алгоритмів вибору значень.
bool get_state(ArchiveDataList_t* list_element,unsigned char bu1_Param_Reliability,unsigned char bu2_Param_Reliability,unsigned char bu3_Param_Reliability);	Метод, для отримання достовірності значення параметра у поточному півсекундному інтервалі. Використовує алгоритм, що встановлений у полі current_mode.
qreal get_value(ArchiveDataList_t* list_element,unsigned char bu1_Param_Reliability,unsigned char bu2_Param_Reliability,unsigned char bu3_Param_Reliability,unsigned char bu1_IsBU_Param,unsigned char bu2_IsBU_Param,unsigned char bu3_IsBU_Param);	Метод для отримання значення параметра у поточному півсекундному інтервалі. Використовує алгоритм, що встановлений у полі current_mode.

### 3.3.2 Розробка інтерфейсу для опису алгоритму вибору значень параметрів, що архівуються ДЛ(Дисплеями локомотивними)

Таблиця 3.3 – Опис полів та методів інтерфейсу DL1\_DL2\_Base

Поле/Метод	Опис
DL1_DL2_Base()	Конструктор класу за замовчуванням, виконує початкову ініціалізацію.
void set_mode(QString mode);	Метод для встановлення алгоритму вибору значення
enum mode{ Auto,Dl1,Dl2};	Доступні алгоритми вибору значення.
mode current_mode=Auto;	Поле, у якому зберігається поточний алгоритм вибору значення. При початковій

	ініціалізації використовується алгоритм автоматичного вибору значення.
QMap<QString,mode> string_to_mode_map;	Хеш масив, у якому зберігається відповідність строкових репрезентацій імен алгоритмів вибору значень та відповідних полей enum mode. Ініціалізація відбувається у конструкторі за замовчуванням DL1_DL2_Base().
QList<QString> mode_names;	Список, у якому зберігаються імена алгоритмів вибору значень.
bool get_state(ArchiveDataList_t* list_element,unsigned char DL1PressureNDataReliability,unsigned char DL2PressureNDataReliability);	Метод, для отримання достовірності значення параметра у поточному півсекундному інтервалі. Використовує алгоритм, що встановлений у полі current_mode.
qreal get_value(ArchiveDataList_t* list_element, unsigned char DL1PressureNDataReliability, unsigned char DL2PressureNDataReliability, unsigned char DL1PressureData, unsigned char DL2PressureData);	Метод для отримання значення параметра у поточному півсекундному інтервалі. Використовує алгоритм, що встановлений у полі current_mode.

### 3.3.3 Розробка інтерфейсу для опису алгоритму вибору значень параметрів, що архівуються БФДС(Блоками формування дискретних сигналів)

Таблиця 3.4 – Опис полів та методів інтерфейсу BFDS\_Base

Поле/Метод	Опис
BFDS_Base()	Конструктор класу за замовчуванням, виконує початкову ініціалізацію.
void set_mode(QString mode);	Метод для встановлення алгоритму вибору значення
enum mode{ Auto,BFDS1,BFDS2};	Доступні алгоритми вибору значення.
mode current_mode=Auto;	Поле, у якому зберігається поточний алгоритм вибору значення. При початковій ініціалізації використовується алгоритм автоматичного вибору значення.

QMap<QString,mode> string_to_mode_map;	Хеш масив, у якому зберігається відповідність строкових репрезентацій імен алгоритмів вибору значень та відповідних полей enum mode. Ініціалізація відбувається у конструкторі за замовчуванням BFDS_Base().
QList<QString> mode_names;	Список, у якому зберігаються імена алгоритмів вибору значень.
bool get_state(ArchiveDataList_t *list_element, unsigned char bfds1_IsParam_Present, unsigned char bfds2_IsParam_Present, unsigned char bfds1_IsParamError, unsigned char bfds2_IsParamError);	Метод, для отримання достовірності значення параметра у поточному пів секундному інтервалі. Використовує алгоритм, що встановлений у полі current_mode.
qreal get_value(ArchiveDataList_t* list_element, unsigned char bu1_Param_Reliability, unsigned char bu2_Param_Reliability, unsigned char bu3_Param_Reliability, unsigned char bu1_IsBU_Param, unsigned char bu2_IsBU_Param, unsigned char bu3_IsBU_Param);	Метод для отримання значення параметра у поточному пів секундному інтервалі. Використовує алгоритм, що встановлений у полі current_mode.

### 3.3.4 Розробка інтерфейсу для опису алгоритму вибору значень параметрів, що архівуються ДЛ(Дисплеми локомотивними) та ІнЛ(Індикатором локомотивним)

Таблиця 3.5 – Опис полів та методів інтерфейсу DL\_INL\_Base

Поле/Метод	Опис
DL_INL_Base()	Конструктор класу за замовчуванням, виконує початкову ініціалізацію.
void set_mode(QString mode);	Метод для встановлення алгоритму вибору значення
enum mode{ Auto,DL1,DL2,INL1,INL2};	Доступні алгоритми вибору значення.
mode current_mode=Auto;	Поле, у якому зберігається поточний алгоритм вибору значення. При початковій

	ініціалізації використовується алгоритм автоматичного вибору значення.
QMap<QString,mode> string_to_mode_map;	Хеш масив, у якому зберігається відповідність строкових репрезентацій імен алгоритмів вибору значень та відповідних полей enum mode. Ініціалізація відбувається у конструкторі за замовчуванням BFDS_Base().
QList<QString> mode_names;	Список, у якому зберігаються імена алгоритмів вибору значень.
bool get_state(ArchiveDataList_t *list_element, unsigned char DL1ParamDataReliability, unsigned char DL2ParamDataReliability, unsigned char INL1ParamDisplayedReliability, unsigned char INL2ParamDisplayedReliability);	Метод, для отримання достовірності значення параметра у поточному пів секундному інтервалі. Використовує алгоритм, що встановлений у полі current_mode.
qreal get_value(ArchiveDataList_t *list_element, unsigned char DL1ParamDataReliability, unsigned char DL2ParamDataReliability, unsigned char INL1ParamDisplayedReliability, unsigned char INL2ParamDisplayedReliability, unsigned char DL1MAJOR_Param, unsigned char DL2MAJOR_Param, unsigned char INL1ParamDisplayed, unsigned char INL2ParamDisplayed);	Метод для отримання значення параметра у поточному пів секундному інтервалі. Використовує алгоритм, що встановлений у полі current_mode.

### 3.4 Створення інтерфейсу, що відповідає за відображення даних параметру на графіку.

Інтерфейс базового класу BaseVisualParameter наведено у таблиці.

Таблиця 3.6 – Опис інтерфейсу класу BaseVisualParameter

Поле/Метод	Опис
DateTimeParameter* datetime_param;	Покажчик на змінну дати і часу

<code>QVector&lt;QLineSeries*&gt; normal_series;</code>	Вектор[5], що зберігає покажчики на лінії [6], що відображають достовірні проміжки значень параметру на графіку
<code>QVector&lt;QLineSeries*&gt; not_reliable_series;</code>	Вектор[5], що зберігає покажчики на лінії, що відображають не достовірні проміжки значень параметру на графіку.
<code>QPointer&lt;QValueAxis&gt; axisY;</code>	Покажчик [7] на вісь Y [8], до якої прив'язані значення параметра
<code>QPointer&lt;ParametersMenuItem&gt; menu_item;</code>	Покажчик [7] на віджет, що відображає інформацію про параметр та надає можливість вибрати алгоритм формування значень для параметрів, що передбачають такий вибір.
<code>QPointer&lt;QScatterSeries&gt; scatter_series;</code>	Покажчик [7] на екземпляр класу <code>QScatterSeries</code> [9], що зберігає точки, у яких є перехід від достовірного значення до не достовірного.
<code>QChart* m_chart;</code>	Покажчик на екземпляр класу графіку <code>QChart</code> [10] на якому буде відображений параметр.
<code>QList&lt;ArchiveDataList_t*&gt;* archive_data;</code>	Покажчик на список півсекундних інтервалів записів архівних даних.
<code>QDateTimeAxis* axisX;</code>	Покажчик на вісь X дати і часу [11], до якої прив'язані значення параметра
<code>QColor series_color;</code>	Поле, у якому зберігається поточний колір [12] ліній параметра.
<code>bool is_visible;</code>	Поле, у якому зберігається ознака видимості ліній параметра.
<code>void set_series_visible(bool visible);</code>	Метод для зміни видимості ліній параметра на графіку.
<code>void paint_analog_series();</code>	Метод, що реалізує логіку формування ліній значень для аналогових параметрів
<code>void paint_discrete_series();</code>	Метод, що реалізує логіку формування ліній значень для дискретних параметрів.



<code>void paint_speed_series();</code>	Метод, що реалізує логіку формування ліній значень для параметрів швидкості.
<code>void set_current_value(QString value);</code>	Метод для відображення поточного значення у елементі <code>menu_item</code> .
<code>void switch_mode(QString new_mode);</code>	Метод, що реалізує механізм вибору алгоритмів формування значення.
<code>BaseVisualParameter();</code>	Конструктор за замовчуванням
<code>void removeAllSeries();</code>	Метод для видалення ліній параметру з графіку
<code>void clear_connections();</code>	Метод для очищення коннектів
<code>void append_series(QLineSeries* series);</code>	Метод для занесення лінії, що репрезентує проміжок достовірних значень, до контейнеру <code>normal_series</code> .
<code>void append_not_reliable_series(QLineSeries* series);</code>	Метод для занесення лінії, що репрезентує проміжок не достовірних значень, до контейнеру <code>not_reliable_series</code> .
<code>void setRange(qreal min, qreal max);</code>	Метод, що встановлює видимі діапазони значень для вісі Y.
<code>void setName(QString name);</code>	Метод, що встановлює скорочене ім'я у <code>menu_item</code>
<code>void setFullName(QString full_name);</code>	Метод, що встановлює повне ім'я у <code>menu_item</code>
<code>void recreateMenuItem();</code>	Метод, що виконує реініціалізацію <code>menu_item</code>
<code>void createMenuItemForRestrictionParam();</code>	Метод, що виконує реініціалізацію <code>menu_item</code> у випадку, коли кінцевий клас нащадок є успадкованим від базового класу тимчасових обмежень.
<code>void setAxisY(QValueAxis* axis);</code>	Метод, що встановлює вісь Y для ліній параметру.
<code>bool visible() const</code>	Метод, що повертає ознаку видимості ліній параметру.
<code>QVector&lt;QLineSeries*&gt; get_normal_series() const;</code>	Метод, що повертає копію вектора <code>normal_series</code> .

ParametersMenuItem* menuItem() const;	Метод, що повертає покажчик на віджет меню параметра.
void set_datetime_param(DateTimeParameter* param);	Метод, що прив'язує параметр до параметра дати і часу
void setArchiveData(QList<ArchiveDataList_t*>* ad);	Метод, що встановлює покажчик на архівні дані.
void setAxisX(QDateTimeAxis* X);	Метод, що встановлює вісь X для параметра.
void setColor(QColor color);	Метод, що встановлює колір для ліній параметру на графіку.

Для того щоб програмно була можливість відрізнити параметри, що мають бути відображені на різних графіках було створено 3 класи, що успадковані від класу BaseVisualParameter:

- BaseDiscreteParameter
- BaseAnalogParameter
- BaseSpeedParameter

Створені класи і методи не мають своїх полів і методів. Натомість вони слугують засобом програмного контролю приналежності параметра до графіка.

### **3.5 Створення кінцевого класу-нащадка, що репрезентує параметр архівних даних.**

Усі класи, що репрезентують параметри архівних даних мають бути успадковані від BaseDataParameter. Якщо параметр потрібно виводити на графік – клас, що репрезентує параметр, має бути успадкований від одного із трьох класів (BaseDiscreteParameter, BaseAnalogParameter, BaseSpeedParameter), що реалізують логіку додавання на відповідний графік. Класи параметрів, що передбачають можливість вибору алгоритму формування значень мають бути успадковані від відповідного класу, що реалізує алгоритм вибору значення (BU\_Base, BFDS\_Base, DL1\_DL2\_Base, DL\_INL\_Base).

Реалізацію класу що репрезентує дискретний параметр «Гальмування» наведено на рис 3.1.

```

4   BrakingParameter::BrakingParameter():
5       BaseDataParameter(QObject::tr("Гальмування"),
6                           QObject::tr("Вхідний дискретний сигнал "Гальмування")),
7       BaseDiscreteParameter(),
8       BU_Base()
9   {
10      _modes.append(mode_names);
11      string_to_mode_map.insert(mode_names.at(0),Auto);
12      string_to_mode_map.insert(mode_names.at(1),BU1);
13      string_to_mode_map.insert(mode_names.at(2),BU2);
14      string_to_mode_map.insert(mode_names.at(3),BU3);
15      current_mode_string=_modes.at(0);
16  }
17
18  bool BrakingParameter::state(ArchiveDataList_t *list_element)
19  {
20      return get_state(list_element,
21                      list_element->entry->bu1.Torm_Reliability,
22                      list_element->entry->bu2.Torm_Reliability,
23                      list_element->entry->bu3.Torm_Reliability);
24  }
25
26  qreal BrakingParameter::value(ArchiveDataList_t *list_element)
27  {
28      return get_value(list_element,
29                      list_element->entry->bu1.Torm_Reliability,
30                      list_element->entry->bu2.Torm_Reliability,
31                      list_element->entry->bu3.Torm_Reliability,
32                      list_element->entry->bu1.IsBU_Torm,
33                      list_element->entry->bu2.IsBU_Torm,
34                      list_element->entry->bu3.IsBU_Torm);
35  }

```

Рисунок 3.1 – Реалізація методів класу BrakingParameter

### 3.6 Відображення об'єктів електронної карти.

Для опису об'єктів електронної карти були створено інтерфейс BaseEMObject. Кінцеві класи нащадки успадковують інтерфейс та перевизначають зображення об'єкту та його опис. Опис полів та методів інтерфейсу BaseEMObject наведено у таблиці.

Таблиця 3.7 Опис полів та методів інтерфейсу BaseEMObject

Поле/Метод	Опис
qreal linear_coordinate;	Поле, у якому зберігається значення лінійної координати об'єкту
QImage image;	Зображення об'єкту [13], що має бути відображене на графіку.
QString p_name;	Назва об'єкту
int p_speed;	Допустима швидкість проходження об'єкту
qlonglong p_slope;	Ухил
int p_path_number;	Номер колії, на якій розташований об'єкт
QString standard_info();	Метод, що повертає стандартну, для більшості об'єктів, інформацію: назва об'єкту, лінійну координату, швидкість походження.

explicit BaseEMObject(QString image_path,QString name,qlonglong coordinate,int speed,qlonglong slope,int path_number);	Конструктор класу, виконується у при створені кінцевих нащадків.Змінна image_path константно задається у класах-нащадках.
qreal linearCoordinate() const	Метод, що повертає значення лінійної координати.
QImage objectImage() const	Метод, що повертає значення поля image.
QString name() const	Метод, що повертає значення поля p_name.
int speed() const	Метод, що повертає значення поля p_speed.
qlonglong slope() const	Метод, що повертає значення поля p_slope.
int path_number() const	Метод, що повертає значення поля p_path_number.
virtual QString about()=0;	Віртуальний метод [2], має бути визначений у класах-нащадках. Використовується для тримання строкової репрезентації опису об'єкту електронної карти.

Для відображення об'єктів електронної карти на графіку необхідно створити новий клас, що має бути успадкований від класу QChartView [14], або його нащадка. Було створено клас TrafficChartView, що є нащадком QChartView через клас EnhancedChartView. Клас EnhancedChartView слугує для забезпечення взаємодії між віджетами управління і масштабування та графіком, використовуючи механізм сигналів та слотів. Інтерфейс класу наведено на рис. , опис інтерфейсу наведено у таблиці

```

80 class TrafficChartView: public EnhancedChartView
81 {
82     Q_OBJECT
83 public:
84     TrafficChartView(QDateTimeAxis *axisX, QWidget *parent = 0);
85     void addEMObject(qreal datetime, BaseEMObject* object);
86     void set_datetime_axis_range(QDateTime min, QDateTime max);
87     void update_datetime_axis_range(QDateTime min, QDateTime max);
88     QList<QRect> objects_geometry();
89     QRegion get_chart_region();
90     void clearEMObjects();
91 public slots:
92     void updateEM_coordinates();
93 private:
94     QMap<qreal, QPointer<QLabel>> EMObjects;
95     QImage multiple_object_icon;
96     QList<QLabel*> multiple_object_containers;
97 protected:
98     void resizeEvent(QResizeEvent *event);
99     void mouseReleaseEvent(QMouseEvent *event);
100 signals:
101     void object_position_updated();
102 };

```

Рисунок 3.2 - Інтерфейс класу TrafficChartView

Таблиця 3.8 – Опис полів і методів класу TrafficChartView

Поле/Метод	Опис
TrafficChartView(QDateTimeAxis *axisX, QWidget *parent = 0);	Конструктор класу.
void addEMObject(qreal datetime, BaseEMObject* object);	Метод, що додає об'єкт електронної карти та прив'язує його до відповідної точки часу по осі X на графіку.
void clearEMObjects();	Метод, що видаляє об'єкти електронної карти з графіку.
void updateEM_coordinates();	Слот[16], що встановлює позиції об'єктів електронної карти на графіку відповідно до поточного масштабу.
QMap<qreal, QPointer<QLabel>> EMObjects;	Контейнер[15], у якому зберігаються об'єкти електронної карти, що відповідають відображаємому маршруту.
QImage multiple_object_icon;	Зображення, що виводиться в разі наявності великої кількості об'єктів електронної карти на проміжку 30 пікселів графіку.

QList<QLabel*> multiple_object_containers;	Контейнер, у якому зберігаються зображення, що позначають декілька об'єктів електронної карти. Для відображення використовується клас мітки QLabel [17].
void object_position_updated();	Сигнал, що відправляється при оновленні позицій об'єктів електронної карти.

Діалогове вікно «Виберіть маршрути для прив'язки» призначене для прив'язки обраного маршруту до поточної поїздки. Залізничні об'єкти обраного маршруту будуть відображатися в області графічного відображення основного вікна програми «Дешифратор». Вид діалогового вікна «Виберіть маршрути для прив'язки» із зазначенням посилань на елементи управління наведений на рисунку 3.3.

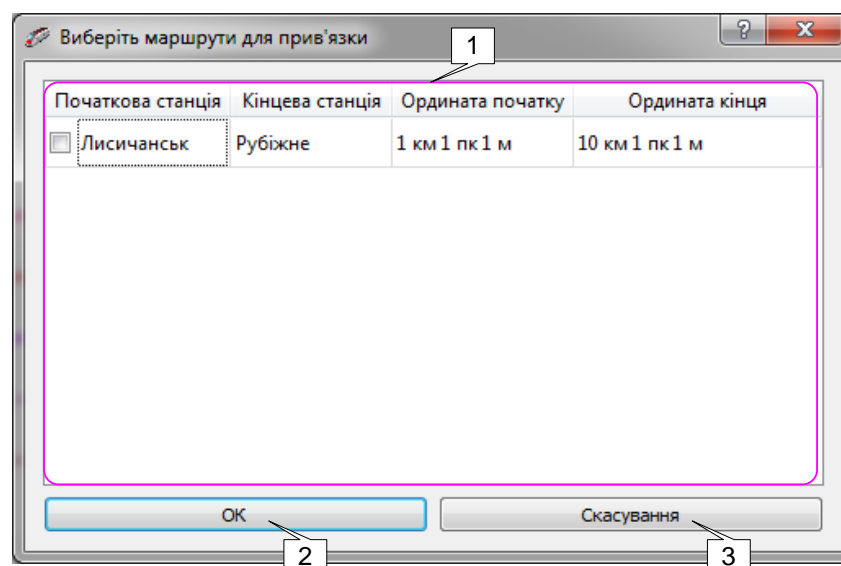


Рисунок 3.3 – Вид діалогового вікна «Виберіть маршрути для прив'язки» із зазначенням посилань на елементи управління

Опис елементів управління діалогового вікна «Виберіть маршрути для прив'язки» наведений в таблиці 3.9.

Таблиця 3.9 – Опис елементів управління діалогового вікна «Виберіть маршрути для прив'язки»

Номер елемента	Призначення елемента
1	Область для відображення інформації про маршрути
2	Кнопка «ОК» призначена для прив'язки обраного маршруту до поточної поїздки

При прив'язці електронної карти до поточної поїздки викликається метод ReadEMObjects, що приймає один аргумент – список структур об'єктів електронної карти. Логіка додавання об'єктів приведена на рис 3.4.

```
478     qreal last_coordinate=-1;
479     for(int i=0;i<archive_data2->size()-1;i++)
480     {
481         if(datetime_param->state(i) &&
482            linearcoordinate_param.data()->state(archive_data2->at(i)) &&
483            linearcoordinate_param.data()->value(archive_data2->at(i))>last_coordinate)
484         {
485             qreal startvalue=linearcoordinate_param.data()->value(archive_data2->at(i));
486             qreal speed=realspeed_param->value(archive_data2->at(i))/7.2;
487             qreal end_value;
488             if(speed-static_cast<qlonglong>(speed)>=0.5)
489                 end_value=startvalue+static_cast<qlonglong>(speed) +1;
490             else
491                 end_value=startvalue+static_cast<qlonglong>(speed);
492
493
494             for(;startvalue<=end_value;startvalue+=1)
495             {
496                 if(EMObjects.contains(startvalue))
497                 {
498                     for(auto it:EMObjects.values(startvalue))
499                     {
500                         if(pathnumber_param->value(archive_data2->at(i)) == it->path_number())
501                         {
502                             traffic_chartView->addEMObject(datetime_param.data()->value(i),it);
503                         }
504                     }
505                 }
506             }
507             last_coordinate=linearcoordinate_param.data()->value(archive_data2->at(i));
508         }
509     }
510     traffic_chartView->updateEM_coordinates();
```

Рисунок 3.4 – Програмна реалізація алгоритму вибору об'єктів електронної карти

### 3.7 Формування груп параметрів

Діалогове вікно «Управління групами параметрів» призначене для створення нової групи параметрів, редагування або видалення існуючої групи параметрів. Вид діалогового вікна «Управління групами параметрів» із зазначенням посилань на елементи управління наведений на рисунку 3.5.

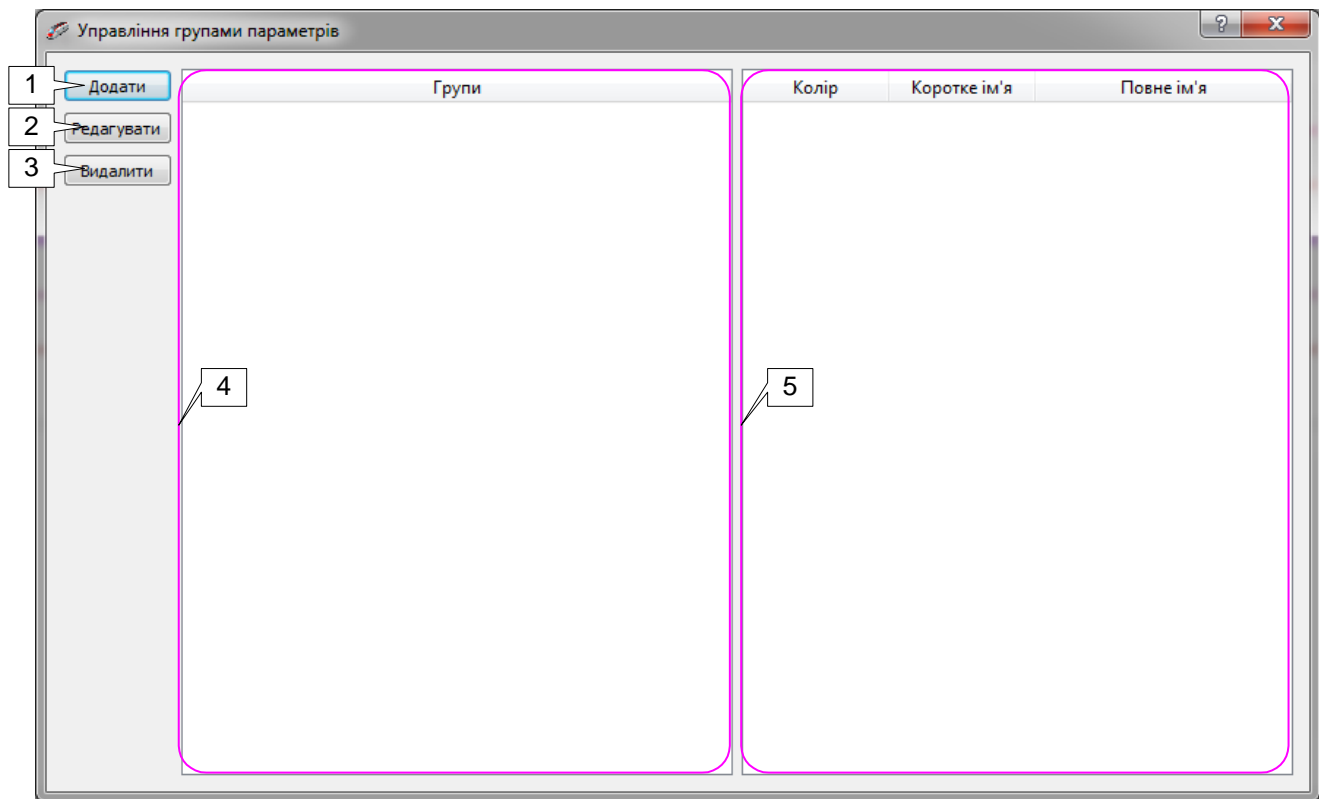


Рисунок 3.5 – Вид діалогового вікна «Управління групами параметрів» із зазначенням посилань  
 Опис елементів управління діалогового вікна «Управління групами параметрів» наведений в таблиці 3.10.

Таблиця 3.10 – Опис елементів управління діалогового вікна «Управління групами параметрів»

Номер елемента	Призначення елемента
1	Кнопка «Додати» призначена для створення нової групи параметрів. При натисканні на кнопку виконується перехід до діалогового вікна, вид якого наведений на <a href="#">рисунок 2.33</a>
2	Кнопка «Редагувати» призначена для редагування обраної групи параметрів. При натисканні на кнопку виконується перехід до діалогового вікна, вид якого наведений на <a href="#">рисунок 2.33</a>
3	Кнопка «Видалити» призначена для видалення обраної групи параметрів
4	Область для відображення груп параметрів
5	Область для відображення параметрів обраної групи параметрів

Вид діалогового вікна «Управління групами параметрів» при виконанні операцій із групами наведено на [рис 3.6](#).



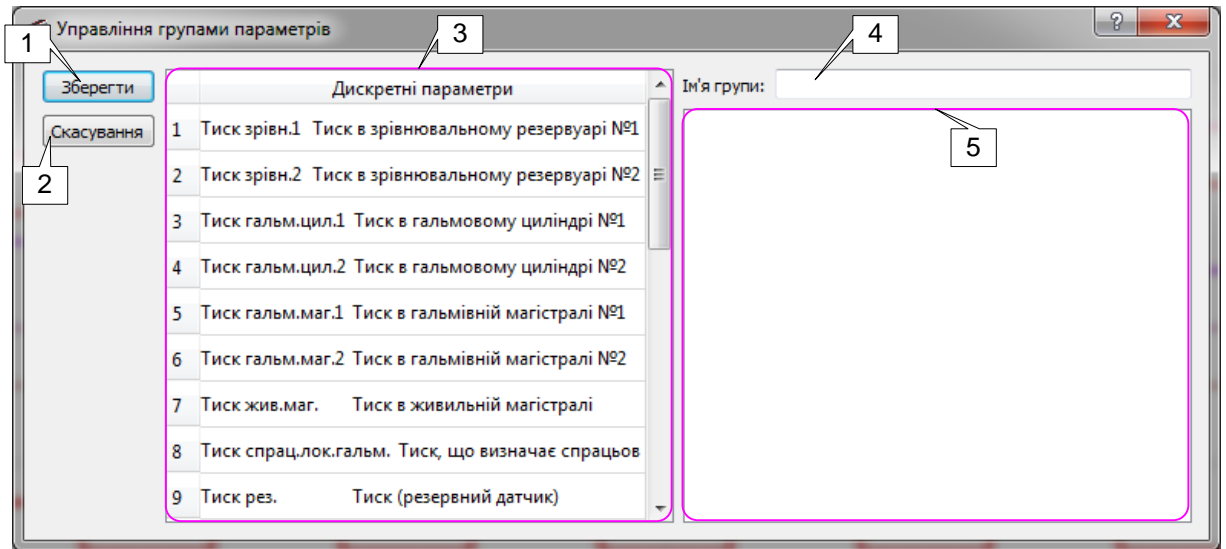


Рисунок 3.6 – Вид діалогового вікна «Управління групами параметрів» при створенні нової групи параметрів або редагуванні існуючої (із зазначенням посилань на елементи управління)

Опис елементів управління діалогового вікна «Управління групами параметрів» при створенні нової групи параметрів або редагуванні існуючої наведений в таблиці 3.11.

Таблиця 3.11 – Опис елементів управління діалогового вікна «Управління групами параметрів» при створенні нової групи параметрів або редагуванні існуючої

Номер елемента	Призначення елемента
1	Кнопка «Зберегти» призначена для збереження нової групи параметрів
2	Кнопка «Скасування» призначена для скасування виконаних дій і закриття діалогового вікна «Управління групами параметрів» при створенні нової групи параметрів або редагуванні існуючої
3	Область для відображення списку параметрів
4	Поле для введення або редагування найменування групи параметрів
5	Область для відображення списку параметрів, що входять до групи параметрів. Додавання параметрів в дану область виконується за допомогою «перетягування» їх зі списку параметрів (елемент 3)

### 3.8 Компонента відображення графіків

На рис 3.7 зображено вікно програми «Дешифратор» у режимі перегляду графічних значень технологічних параметрів.

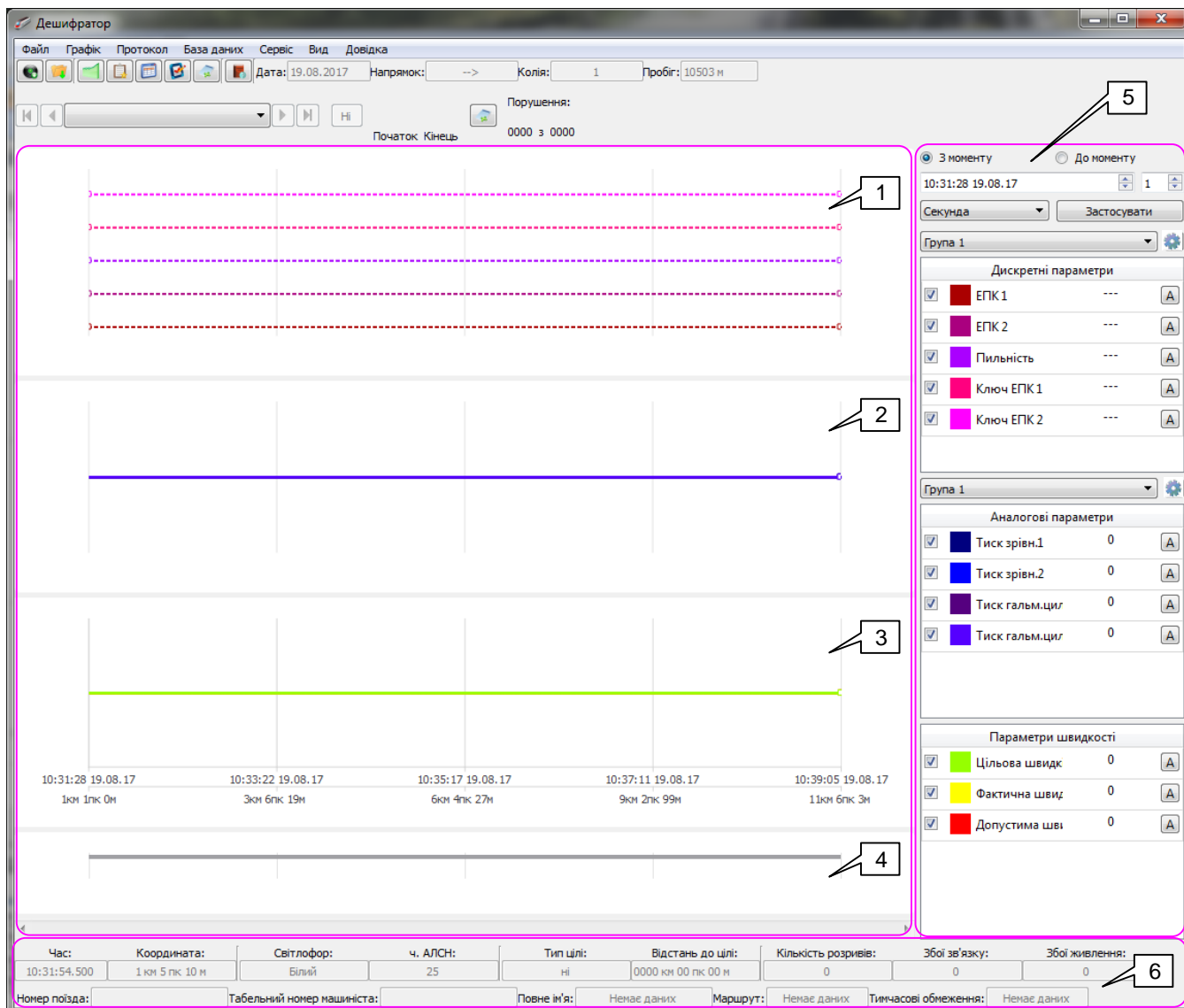


Рисунок 3.7 – Вид компоненти відображення параметрів архівних даних

Опис елементів компоненти відображення параметрів наведений в таблиці 3.12.

Таблиця 3.12

Номер елемента	Опис
1	Графік відображення дискретних параметрів
2	Графік відображення аналогових параметрів
3	Графік відображення параметрів швидкості та тимчасових обмежень
4	Графік відображення сигналу світлофора та об'єктів електронної карти
5	Панель параметрів
6	Панель основних параметрів

### 3.9 Панель параметрів

Панель параметрів призначена для перегляду поточних значень параметрів (дискретних, аналогових, швидкостей) та управління відображенням їх на графіку. Вид панелі параметрів із зазначенням посилань на елементи управління та динаміки наведений на рисунку 3.7.

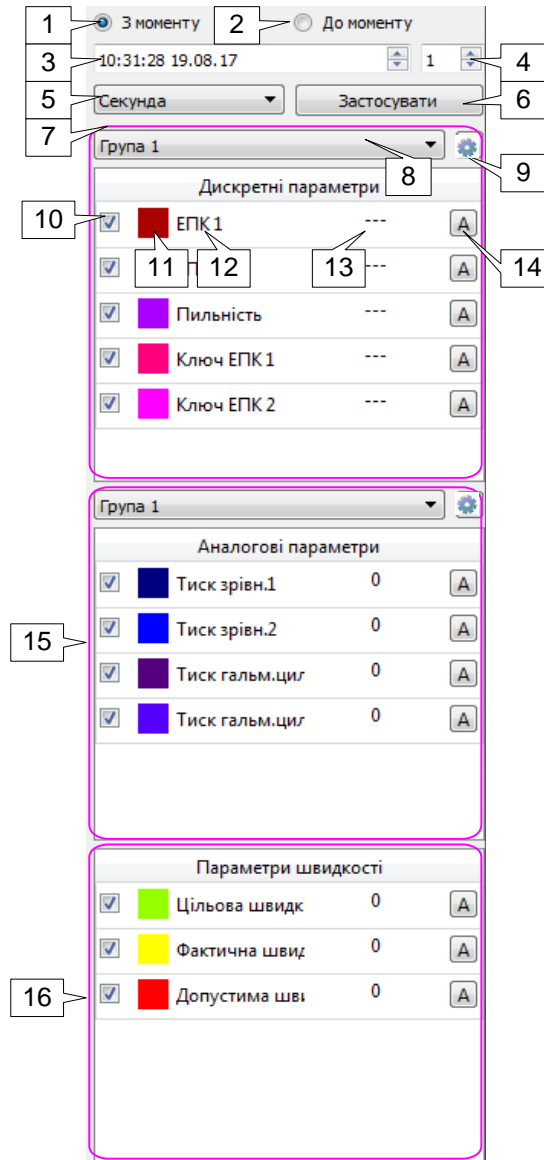





Рисунок 3.8 – Вид панелі параметрів із зазначенням посилань

Опис елементів управління та динаміки панелі параметрів наведений в таблиці 3.13.

Таблиця 3.13 – Опис елементів управління та динаміки панелі параметрів

Номер елемента	Опис
1	Елемент управління для зазначення періоду перегляду з моменту, встановленого в елементі 3

2	Елемент управління для зазначення періоду перегляду до моменту, встановленого в елементі 3
3	Поле для введення дати і часу початку перегляду параметрів при встановленому елементі 1 або кінця перегляду параметрів при встановленому елементі 2. Дата і час можуть вводитись за допомогою управляючих елементів  або введенням рядка в поле
4	Поле для введення кількості одиниць тривалості відображення параметрів. Кількість одиниць може вводитися за допомогою управляючих елементів  або введенням рядка в поле
5	Поле для вибору одиниць виміру тривалості відображення параметрів
6	Кнопка « <b>Застосувати</b> » призначена для підтвердження встановлених значень в елементах 1 - 5
7	Група елементів управління для відображення дискретних параметрів на графіку
8	Поле для вибору групи параметрів (примітка)
9	Кнопка «  » призначена для створення нової групи параметрів, редагування або видалення існуючої групи параметрів. При натисканні даної кнопки виконується перехід до діалогового вікна «Управління групами параметрів» (опис наведений в 3.6)
10	Елемент управління для встановлення ознаки відображення відповідного параметра на графіку
11	Поле для відображення кольору відповідного параметра на графіку
12	Поле для відображення найменування відповідного параметра
13	Поле для відображення поточного значення відповідного параметра
14	Кнопка « <b>A</b> » (або « <b>P</b> ») призначена для встановлення режиму вибору (автоматичного або ручного) значення сигналу відповідного параметра. Після натискання даної кнопки виконується перехід до діалогового вікна «Вибір режиму відображення». Назва кнопки змінюється в залежності від обраного режиму (« <b>A</b> » – автоматичний режим, « <b>P</b> » – ручний режим). У разі автоматичного вибору значення сигналу відповідного параметра, визначається перелік достовірних значень даного сигналу від блоків (пристроїв), а потім здійснюється вибір значення за алгоритмом «2 з 3» (у випадку трьох достовірних значень), або першого достовірного (у випадку двох або чотирьох значень). У разі ручного вибору значення сигналу відповідного параметра, здійснюється вибір значення сигналу від обраного користувачем блоку (пристрою)

15	Група елементів управління для відображення аналогових параметрів на графіку. Призначення елементів управління для аналогових параметрів аналогічне призначенню елементів управління для дискретних параметрів (елементи 8 - 14)
16	Група елементів управління для відображення швидкостей на графіку. Призначення елементів управління для швидкостей аналогічне призначенню елементів управління для дискретних параметрів (елементи 10 - 14)

### 3.10 Масштабування графіків та виведення значень параметрів за курсором.

Для реалізації функції масштабування та виводу значень за курсором було розроблено клас `EnhancedChartCursor`, що наслідує від `QFrame` і являє собою прозорий віджет, що завжди знаходиться поверх спліттера з графіками. Інтерфейс класу наведено у таблиці 3.14.

Таблиця 3.14

Поле/Метод	Опис
<code>explicit EnhancedChartCursor(QWidget *parent = nullptr);</code>	Конструктор класу.
<code>enum rubberBandType {Rectangle, Horizontal, Vertical};</code>	Перечислення, що визначає можливі режими масштабування
<code>void setRubberBandType(rubberBandType type);</code>	Метод, що встановлює режим масштабування
<code>rubberBandType getRubberBandType()const {return rubberband_type;}</code>	Метод, що повертає поточний режим масштабування
<code>QRubberBand* rubber_band;</code>	Показчик на екземпляр класу <code>QRubberBand</code> , що відповідає за візуалізацію дії масштабування.
<code>QPointer&lt;QLabel&gt; cursor_line;</code>	Показчик на курсор.
<code>bool pressed;</code>	Ознака, що встановлена в <code>true</code> , якщо ліва кнопка миші натиснута.
<code>QPoint clicked_pos;</code>	Точка, у якій відбулося натискання лівої кнопки миші
<code>QPoint origin;</code>	Точка, у якій почав свій рух курсор під час <code>Drag&amp;Drop</code>

<code>rubberBandType rubberband_type;</code>	Поле, у якому зберігається поточний режим масштабування.
<code>void mousePressEvent(QMouseEvent *event);</code>	Метод, що визивається при натисканні кнопки миші.
<code>void mouseReleaseEvent(QMouseEvent *event);</code>	Метод, що визивається при відпусканні кнопки миші.
<code>void resizeEvent(QResizeEvent *event);</code>	Метод, що відповідає за коректу зміну розміру курсору при зміні розміру головного вікна.
<code>void dragEnterEvent(QDragEnterEvent *event) override;</code>	Метод, що визивається при створенні об'єкту QDrag
<code>void dragMoveEvent(QDragMoveEvent *event) override;</code>	Метод, що визивається при переміщення мишею об'єкту QDrag.
<code>void dropEvent(QDropEvent *event) override;</code>	Метод, що визивається при відпусканні лівої кнопки миші та встановлення нової позиції для курсора
<code>void position_changed(int x);</code>	Сигнал, що посилається при зміні позиції курсору, у якості аргумента передає позицію верхнього лівого кута по осі X.
<code>void rubberband_rect_changed(QRectF geometry);</code>	Сигнал, що посилається при зміні масштабу. У якості аргумента передає прямокутник, що визначає новий масштаб
<code>void zoomOut();</code>	Слот, що визивається

Пересування курсору здійснено за допомогою технології Drag&Drop. Для цього було перевизначено метод `mousePressEvent` для створення дії `QDrag`. Було визначено методи, що реалізують механізм пересування Drag&Drop об'єктів:

- `dragEnterEvent`
- `dragMoveEvent`
- `dropEvent`

На рисунку 3.9 наведено програмну реалізацію частини перевизначеного метода `mousePressEvent`, що відповідає за створення об'єкту `QDrag` при натисканні на область курсору.

```

94     child->setCursor(QCursor(Qt::ClosedHandCursor));
95     QPixmap pixmap = *child->pixmap();
96
97     QByteArray itemData;
98     QDataStream dataStream(&itemData, QIODevice::WriteOnly);
99     dataStream << pixmap << QPoint(event->pos() - child->pos());
100
101     QMimeData *mimeData = new QMimeData;
102     mimeData->setData("application/x-dnditemdata", itemData);
103
104     QDrag *drag = new QDrag(this);
105     drag->setMimeData(mimeData);
106     drag->setPixmap(pixmap);
107     drag->setHotSpot(event->pos() - child->pos());
108
109
110     QPixmap tempPixmap = pixmap;
111     QPainter painter;
112     painter.begin(&tempPixmap);
113     painter.fillRect(pixmap.rect(), QColor(200, 200, 200, 200));
114     painter.end();
115
116     child->setPixmap(tempPixmap);
117
118     if (drag->exec(Qt::CopyAction | Qt::MoveAction, Qt::CopyAction) == Qt::MoveAction) {
119         child->close();
120     } else {
121         child->show();
122         child->setPixmap(pixmap);
123     }
124 }

```

Рисунок 3.9 Створення дії QDrag для пересування курсору у методі MousePressEvent

На рисунку 3.10 наведено перевизначений метод dragEnterEvent, роль якого полягає у встановленні дії за замовчуванням для об'єктів QDrag, що створені на графічній сцені класу EnhancedChartCursor.

```

220 void EnhancedChartCursor::dragEnterEvent(QDragEnterEvent *event)
221 {
222     if (event->mimeData()->hasFormat("application/x-dnditemdata")) {
223         if (event->source() == this) {
224             event->setDropAction(Qt::MoveAction);
225
226             event->accept();
227         } else {
228             event->acceptProposedAction();
229         }
230     } else {
231         event->ignore();
232     }
233 }

```

Рисунок 3.10 Перевизначений метод DragEnterEvent

На рисунку 3.11 наведено перевизначений метод dragMoveEvent, який визивається при зміні позиції курсору шляхом перетаскування за допомогою миші. Головне призначення

методу полягає у передачі нової позиції курсору для оновлення значень параметрів у панелі параметрів шляхом відправки сигналу `position_changed`.

```
235 void EnhancedChartCursor::dragMoveEvent(QDragMoveEvent *event)
236 {
237
238     if(event->pos().y() != pressed_y_position)
239         QCursor::setPos(QCursor::pos().x(), pressed_y_position);
240     if (event->mimeType() == "application/x-dnditemdata") {
241         if (event->source() == this) {
242             event->setDropAction(Qt::MoveAction);
243             event->accept();
244             emit position_changed(this->mapToGlobal(event->pos()).x());
245         } else {
246             event->acceptProposedAction();
247         }
248     } else {
249         event->ignore();
250     }
251 }
```

Рисунок 3.11 Перевизначений метод `DragMoveEvent`

На рисунку 3.12 наведено перевизначений метод `dropEvent`, що визивається при відпусканні лівої кнопки миші та фіксації курсору на позиції. Після фіксації нової позиції курсору відправляється сигнал `position_changed`.

```
253 void EnhancedChartCursor::dropEvent(QDropEvent *event)
254 {
255     if (event->mimeType() == "application/x-dnditemdata") {
256         QByteArray itemData = event->mimeTypeData()->data("application/x-dnditemdata");
257         QDataStream dataStream(&itemData, QIODevice::ReadOnly);
258
259         QPixmap pixmap;
260         QPoint offset;
261         dataStream >> pixmap >> offset;
262
263         cursor_line = new QLabel(this);
264         cursor_line->setPixmap(pixmap);
265         cursor_line->move(event->pos() - offset);
266         cursor_line->move(cursor_line->x(), this->y());
267         cursor_line->setCursor(Qt::OpenHandCursor);
268         cursor_line->show();
269         cursor_line->setAttribute(Qt::WA_DeleteOnClose);
270
271         if(is_multi_chart)
272         {
273             setMask(saved_region+cursor_line->geometry());
274         }else
275         {
276             setMask(saved_region);
277         }
278
279         if (event->source() == this) {
280             event->setDropAction(Qt::MoveAction);
281             event->accept();
282             emit position_changed(this->mapToGlobal(cursor_line->pos()).x());
283         } else {
284             event->acceptProposedAction();
285         }
286     } else {
287         event->ignore();
288     }
289 }
```

Рисунок 3.12 Перевизначений метод `dropEvent`



На рисунку 3.13 наведено частину перевизначеного методу `mousePressEvent`, яка відповідає за визначення початку операції масштабування. Виділення області виконується в залежності від обраного режиму масштабування:

- масштабування вибором прямокутної області.
- масштабування по вісі дати і часу.
- масштабування по вісі значень.

```
48     QLabel *child = static_cast<QLabel*>(childAt(event->pos()))
49     if (!child)
50     {
51         if(event->button() == Qt::LeftButton)
52         {
53             pressed=true;
54             clicked_pos=event->pos();
55             origin=event->pos();
56             QRect geometry=QRect(event->pos(),QSize());
57             switch (rubberband_type) {
58                 case Horizontal:
59                     origin.setY(0);
60                     geometry.setY(0);
61                     rubber_band->setGeometry(geometry);
62                     rubber_band->show();
63                     break;
64                 case Vertical:
65                     origin.setX(0);
66                     geometry.setX(0);
67                     rubber_band->setGeometry(geometry);
68                     rubber_band->show();
69                     break;
70
71                 case Rectangle:
72                     rubber_band->setGeometry(geometry);
73                     rubber_band->show();
74                     break;
75             }
76             saved_region=mask();
77             QRegion region;
78             region+=frameGeometry();
79             setMask(region);
80         }
81         return;
82     }
```

Рисунок 3.13 Візуалізація дії масштабування при виділенні бажаної області.

Схема взаємодії класів для виконання операції зміни масштабу наведена на рис 3.14

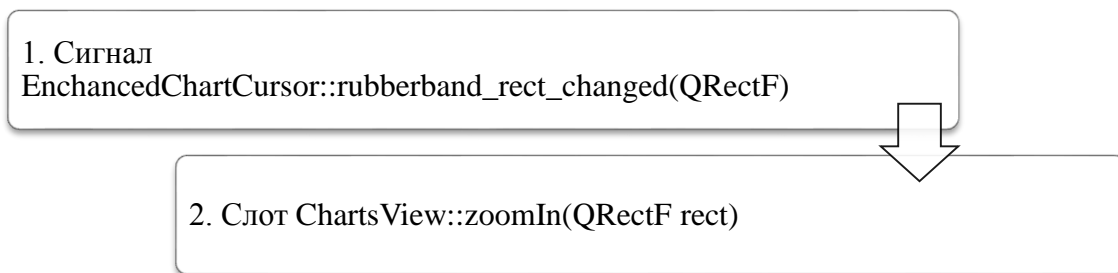


Рисунок 3.14 Схема взаємодії компонент при виконанні масштабування

Схема взаємодії класів для виведення значення за курсором наведена на рис 3.15

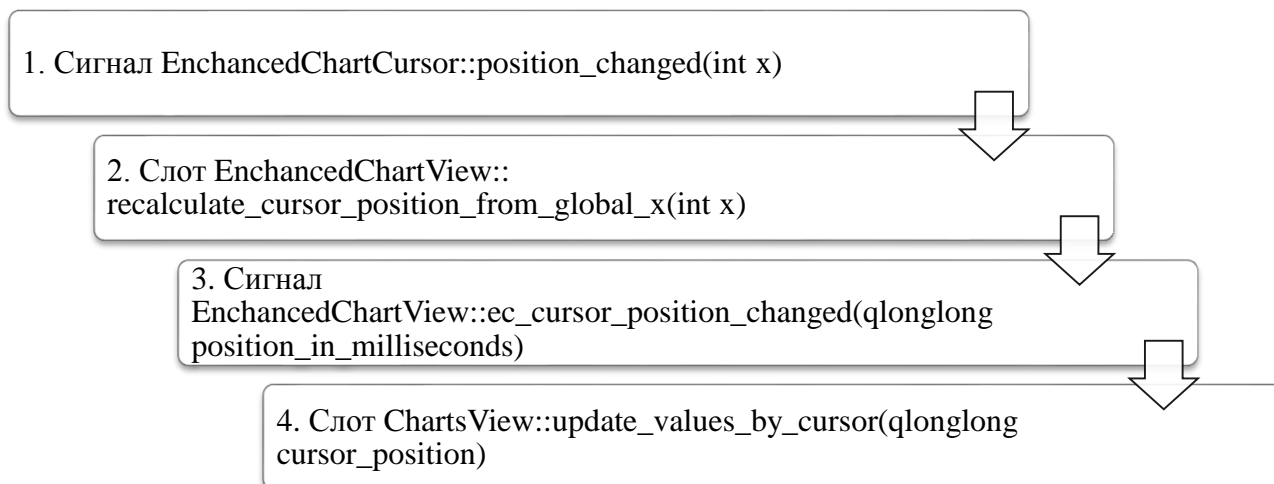


Рисунок 3.15 Схема взаємодії компонент при виведенні значень параметрів за курсором

При зміні позиції курсору посилається сигнал 1 `position_changed` до слота 2 `recalculate_cursor_position_from_global_x` для перерахунку позиць курсора на графіку із пікселей у мілісекунди на осі X. Позиція курсору у мілісекундах посилається у сигналі 3 `ec_cursor_position_changed` до слоту 4 `update_values_by_cursor`, де в циклі оновлюються значення для всіх відображених параметрів відповідно до встановленої позиції курсору.

### 3.11 Висновки до розділу 3

Було розроблено компоненту візуалізації архівних даних для ПЗ «Дешифратор» системи локомотивної безпеки СЛБ-І. Для програмного опису параметрів за допомогою засобів об'єктно-орієнтованого програмування було створено ієрархію абстрактних класів, що дозволяють об'єднати параметри за їх ознаками у групи.

Компонента повністю задовольняє поставленим функціональним вимогам та вимогам деталізації.

### 3.12 Література до розділу 3

- [1] Конструктор класу [Електронний ресурс] // Режим доступу [www. URL](http://www.URL) [https://uk.wikipedia.org/wiki/Конструктор\\_\(програмування\)](https://uk.wikipedia.org/wiki/Конструктор_(програмування))
- [2] Віртуальний метод [Електронний ресурс] // Режим доступу [www. URL](http://www.URL) [https://uk.wikipedia.org/wiki/Віртуальний\\_метод](https://uk.wikipedia.org/wiki/Віртуальний_метод)
- [3] Електронна документація Qt. Клас QMap [Електронний ресурс] // Режим доступу [www. URL](http://www.URL) <http://doc.qt.io/qt-5/qmap.html>
- [4] Електронна документація Qt. Клас QList [Електронний ресурс] // Режим доступу [www. URL](http://www.URL) <http://doc.qt.io/qt-5/qmap.html>
- [5] Електронна документація Qt. Клас QVector [Електронний ресурс] // Режим доступу [www. URL](http://www.URL) <http://doc.qt.io/qt-5/qvector.html>
- [6] Електронна документація Qt. Клас QLineSeries [Електронний ресурс] // Режим доступу [www. URL](http://www.URL) <https://doc.qt.io/qt-5.10/qlineseries.html>
- [7] Електронна документація Qt. Клас QPointer [Електронний ресурс] // Режим доступу [www. URL](http://www.URL) <http://doc.qt.io/qt-5/qpointer.html>
- [8] Електронна документація Qt. Клас QValueAxis [Електронний ресурс] // Режим доступу [www. URL](http://www.URL) <https://doc.qt.io/qt-5.10/qvalueaxis.html>
- [9] Електронна документація Qt. Клас QScatterSeries [Електронний ресурс] // Режим доступу [www. URL](http://www.URL) <https://doc.qt.io/qt-5.10/qscatterseries.html>
- [10] Електронна документація Qt. Клас QChart [Електронний ресурс] // Режим доступу [www. URL](http://www.URL) <https://doc.qt.io/qt-5.10/qchart.html>
- [11] Електронна документація Qt. Клас QDateTimeAxis [Електронний ресурс] // Режим доступу [www. URL](http://www.URL) <https://doc.qt.io/qt-5.10/qdatetimeaxis.html>
- [12] Електронна документація Qt. Клас QColor [Електронний ресурс] // Режим доступу [www. URL](http://www.URL) <http://doc.qt.io/qt-5/qcolor.html>
- [13] Електронна документація Qt. Клас QImage [Електронний ресурс] // Режим доступу [www. URL](http://www.URL) <http://doc.qt.io/qt-5/qimage.html>
- [14] Електронна документація Qt. Клас QChartView [Електронний ресурс] // Режим доступу [www. URL](http://www.URL) <https://doc.qt.io/qt-5.10/qchartview.html>
- [15] Електронна документація Qt. Клас QMapMultiMap [Електронний ресурс] // Режим доступу [www. URL](http://www.URL) <http://doc.qt.io/qt-5/qmultimap.html>
- [16] Електронна документація Qt. Механізм сигналів і слотів. [Електронний ресурс] // Режим доступу [www. URL](http://www.URL) <http://doc.qt.io/qt-5/signalsandslots.html>
- [17] Електронна документація Qt. Клас QLabel [Електронний ресурс] // Режим доступу

www. URL <http://doc.qt.io/qt-5/qlabel.html>

[18] Reumann, K. and Witkam, A. P. M.. ‘Optimizing curve segmentation in computer graphics’, in Proceedings of International Computing Symposium, pp. 467–472.

## РОЗДІЛ 4

### ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ. ЕКОЛОГІЯ

У даному розділі проведено аналіз потенційних небезпечних та шкідливих виробничих факторів, причин пожеж. Розглянуті заходи, які дозволяють забезпечити гігієну праці і виробничу санітарію. На підставі аналізу розроблені заходи з техніки безпеки та рекомендації з пожежної профілактики.

Завданням даної магістерської роботи було розробка компоненти візуалізації архівних даних системи локомотивної безпеки СЛБ-І, і як результат було створено програмний засіб логічного виводу інформаційної підтримки прийняття рішень з управління небезпечними об'єктами у критичних ситуаціях. За цим програмним засобом була розроблена реальна система, яка значно полегшить процес прийняття рішень щодо управління небезпечними об'єктами у критичних ситуаціях. Так як в процесі проектування використовувалося програмне забезпечення, то аналіз потенційно небезпечних і шкідливих виробничих чинників виконується для персонального комп'ютера.

#### 4.1 Загальні питання з охорони праці

Умови праці на робочому місці, безпека технологічних процесів, машин, механізмів, устаткування та інших засобів виробництва, стан засобів колективного та індивідуального захисту, що використовуються працівником, а також санітарно-побутові умови повинні відповідати вимогам нормативних актів про охорону праці. У законі України «Про охорону праці» визначається, що охорона праці - це система правових, соціально-економічних, організаційно-технічних, санітарно-гігієнічних і лікувально-профілактичних заходів та засобів, спрямованих на збереження життя, здоров'я і працездатності людини у процесі трудової діяльності.

При роботі з обчислювальною технікою змінюються фізичні і хімічні фактори навколишнього середовища: виникає статична електрика, електромагнітне випромінювання, змінюється температура і вологість, рівень вміст кисню і озону в повітрі. Повітря забруднюється шкідливими хімічними речовинами антропогенного походження за рахунок деструкції полімерних матеріалів, які використовуються для обробки приміщень та обладнання. Неправильна організація робочого місця сприяє загальному і локальній напрузі м'язів ший, тулуба, верхніх кінцівок, викривлення хребта і розвитку остеохондрозу. На всіх підприємствах, в установах, організаціях повинні створюватися безпечні і нешкідливі умови праці.

Забезпечення цих умов покладається на власника або уповноважений ним орган (далі роботодавець). Умови праці на робочому місці, безпека технологічних процесів, машин, механізмів, устаткування та інших засобів виробництва, стан засобів колективного та індивідуального захисту, що використовуються працівником, а також санітарно-побутові умови повинні відповідати вимогам нормативних актів про охорону праці. Роботодавець повинен впроваджувати сучасні засоби техніки безпеки, які запобігають виробничому травматизмові, і забезпечувати санітарно-гігієнічні умови, що запобігають виникненню професійних захворювань працівників. Він не має права вимагати від працівника виконання роботи, поєднаної з явною небезпекою для життя, а також в умовах, що не відповідають законодавству про охорону праці. Працівник має право відмовитися від дорученої роботи, якщо створилася виробнича ситуація, небезпечна для його життя чи здоров'я або людей, які його оточують, і навколишнього середовища.

#### **4.1.1 Організаційно-технічні заходи з безпеки праці**

У організації/підприємстві проводиться навчання і перевірка знань з питань охорони праці відповідно до вимог Типового положення про порядок проведення навчання і перевірки знань з питань охорони праці, затвердженого наказом Держнаглядохоронпраці України від 26.01.2005 N 15, зареєстрованого в Міністерстві юстиції України 15.02.2005 за N 231/10511 [1].

Також впроваджені організаційні заходи з пожежної безпеки - навчання і перевірку знань відповідно до вимог Типового положення про інструктажі в установах та організаціях України, затвердженого наказом Міністерства України з питань надзвичайних ситуацій.

Обов'язковими вимогами враховане наступне:

- не слід допускати до роботи осіб, що в установленому порядку не пройшли навчання, інструктаж та перевірку знань з охорони праці, пожежної безпеки та цих Правил.
- ознайомлення з правилами безпеки праці, одержання відповідних інструктажів засвідчується у журналі інструктажів.
- перед допуском до самостійної роботи кожен працівник проходить навчання з питань охорони праці:

1) *вступного*, який проводять працівники служби охорони праці об'єкта господарювання з усіма працівниками, яких приймають на роботу незалежно від їхньої освіти та стажу роботи за програмою, в якій подають загальні питання охорони праці із врахуванням її особливостей на об'єкті господарювання;

2) *первинного*, який проводять керівники структурних підрозділів на місці праці з кожним працівником до початку їхньої роботи на цьому робочому місці.

- обов'язкові організаційні заходи перед початком, під час і після завершення роботи

повинні включати перевірку (візуально) наявності і справності електрообладнання та його заземлення, а під час виконання роботи вимогу «не залишати без нагляду обладнання, яке працює». Після закінчення роботи - вимагається прибирання робочого місця, відключення всіх електроприладів від електромережі.

## **4.2 Аналіз стану умов праці**

Робота над створенням програмної компоненти візуалізації архівних даних системи локомотивної безпеки СЛБ-І буде проходити у приміщенні п'ятиповерхової будівлі на 2-ому поверсі. Для даної роботи достатньо однієї людини, для якої надано робоче місце зі стаціонарним комп'ютером.

### **4.2.1 Вимоги до приміщень**

Геометричні розміри приміщення зазначені у табл. 4.1.

Таблиця 4.1 – Розміри приміщення

Найменування	Значення
Довжина, м	12
Ширина, м	4
Висота, м	3
Площа, м <sup>2</sup>	48
Об'єм, м <sup>3</sup>	144

Згідно з [3] розмір площі для одного робочого місця оператора персонального комп'ютера має бути не менше 6 м<sup>2</sup>, а об'єм — не менше 20 м<sup>3</sup>. Отже, дане приміщення цілком відповідає зазначеним нормам. Для зручності спільної роботи з іншими працівниками у кімнаті є диван, стільці та стіл. Також робочий процес пов'язаний з багатьма документами, теками, журналами для чого приміщення облаштоване двома принтерами і трьома шафами для зручності. Задля дотримання визначеного рівня мікроклімату в будівлі встановлено систему опалення і кондиціонування. Для забезпечення потрібного рівного освітленості кімната має два вікна та систему загального рівномірного освітлення, що встановлена на стелі.

### **4.2.2 Вимоги до організації місця праці**

При порівнянні відповідності характеристик робочого місця (табл. 4.2) нормативним основні вимоги до організації робочого місця за [4] і відповідними фактичними значеннями для робочого місця, констатуємо повну відповідність.

Таблиця 4.2 - Характеристики робочого місця

Найменування параметра	Фактичне значення	Нормативне значення
Висота робочої поверхні, мм	740	680 ÷ 800
Висота простору для ніг, мм	720	не менше 600
Ширина простору для ніг, мм	820	не менше 500
Глибина простору для ніг, мм	710	не менше 650
Висота поверхні сидіння, мм	430	400 ÷ 500
Ширина сидіння, мм	410	не менше 400
Глибина сидіння, мм	500	не менше 400
Висота поверхні спинки, мм	600	не менше 300
Ширина опорної поверхні спинки, мм	500	не менше 380
Радіус кривини спинки в горизонтальній площині, мм	400	400
Відстань від очей до екрану дисплея, мм	750	700 ÷ 800

Робочий стіл на досліджуваному місці містить достатньо простору для ніг. Крісло, що використовується в якості робочого сидіння, є підйомно-поворотним, має підлокітники і можливість регулювання за висотою і кутом нахилу спинки, також воно м'яке і виконане з екологічної шкіри, що дає можливість працювати у комфорті. Екран монітору знаходиться на відстані 750 мм, клавіатура має можливість регулювання кута нахилу 5-15°. Отже, за всіма параметрами робоче місце відповідає нормативним вимогам.

Приміщення знаходиться на другому поверсі п'ятиповерхової будівлі і має об'єм 192 м<sup>3</sup>, площу – 48 м<sup>2</sup>. У цьому кабінеті обладнано сім місць праці, що укомплектовані ПК.

Температура в приміщенні протягом року коливається у межах 18–23°C, відносна вологість — близько 50%. Швидкість руху повітря не перевищує 0,2 м/с. Шум у приміщенні знаходиться на рівні 50 дБА. Система вентилявання приміщення — природна неорганізована, а опалення — централізоване.

Розміщення вікон забезпечує природне освітлення з коефіцієнтом природного освітлення не менше 1,5%, а загальне штучне освітлення, яке здійснюється за допомогою восьми люмінесцентних ламп, забезпечує рівень освітленості не менше 200 Лк.

У приміщенні є електрична мережа з напругою 220 В, яка створює небезпеку ураження електричним струмом. ПК може бути джерелом електромагнітних випромінювань, аерозолів та шкідливих речовин (часток тонеру, оксидів нітрогену та озону).

За ступенем пожежної безпеки приміщення належить до категорії В. Кабінет оснащений переносним вуглекислотним вогнегасником ВВК-5.

Наявна аптечка для надання долікарської допомоги, а також у кабінеті роблять вологе прибирання та щоденно провітрюють приміщення.



### **4.2.3 Навантаження та напруженість процесу праці**

Під час виконання магістерської роботи: за фізичним навантаженням робота відноситься до категорії легкі роботи (Ia), її виконують сидячи з періодичним ходінням. Щодо характеру організування виконання дипломної роботи, то він підпадає під нав'язаний режим, оскільки певні розділи роботи необхідно виконати у встановлені конкретні терміни. За ступенем нервово-психічної напруги виконання роботи можна віднести до II – III ступеня і кваліфікувати як помірно напружений – напружений за умови успішного виконання поставлених завдань.

Під час виконання робіт використовується ПК, що призводить до навантаження на окремі системи організму. Такі перекося у напруженні різних систем організму, що трапляються під час роботи з ПК, зокрема, значна напруженість зорового аналізатора і довготривале малорухоме положення перед екраном, не тільки не зменшують загального напруження, а навпаки, призводять до його посилення і появи стресових реакцій.

Найбільшому ризику виникнення різноманітних порушень піддаються: органи зору, м'язово скелетна система, нервово-психічна діяльність, репродуктивна функція у жінок.

Рекомендовано застосування екранних фільтрів, локальних світлофільтрів (засобів індивідуального захисту очей) та інших засобів захисту, а також інші профілактичні заходи наведені в [4].

Роботу за дипломним проектом визнано, таку, що займає 50% часу робочого дня та за восьмигодинної робочої зміни встановлено регламентовану перерву тривалістю 15 хв через кожну годину роботи.

## **4.3 Виробнича санітарія**

На підставі аналізу небезпечних та шкідливих факторів при виробництві (експлуатації), пожежної безпеки можуть бути надалі вирішені питання необхідності забезпечення працюючих достатньою кількістю освітлення, вентиляції повітря, організації заземлення, тощо.

### **4.3.1 Аналіз небезпечних та шкідливих факторів при виробництві (експлуатації) виробу**

Робота, пов'язана з ЕОП з ВДТ, у тому числі на тих, які мають робочі місця, обладнані ЕОМ з ВДТ і ПП, виконується із забезпеченням виконання [6], які встановлюють вимоги безпеки до обладнання робочих місць, до роботи із застосуванням ЕОМ з ВДТ і ПП.

Основними робочими характеристиками персонального комп'ютера є наступні:

- робоча напруга  $U = +220\text{В} \pm 5\%$ ;

- робочий струм  $I = 2A$ ;
- споживана потужність  $P = 350 \text{ Вт}$ .

Робочі місця відповідають вимогам Державних санітарних правил і нормам роботи з візуальними дисплейними терміналами електронно-обчислювальних машин, затверджених постановою Головного державного санітарного лікаря України від 10.12.98 N 7 [4].

За умов роботи з ПК виникають наступні небезпечні та шкідливі чинники: несприятливі мікрокліматичні умови, освітлення, електромагнітні випромінювання, електричний струм, електростатичне поле, напруженість трудового процесу та інше.

Аналіз небезпечних та шкідливих виробничих факторів виконується у табличній формі (табл. 4.3).

Таблиця 4.3 – Аналіз небезпечних і шкідливих виробничих факторів

Небезпечні і шкідливі виробничі фактори	Джерела факторів (види робіт)	Кількісна оцінка	Нормативні документи
<b><i>фізичні</i></b>			
підвищена температура поверхонь обладнання	експлуатація ЕОМ	2	[3]
підвищена або знижена вологість повітря	-	2	[3]
підвищений рівень напруги електричної мережі, замикання якої може відбутися через тіло людини	-	4	[6], [7]
підвищена напруженість електричного поля	-	2	[8]
недостатність природного світла	порушення умов праці (вимог до приміщень)	2	[9]
недостатнє освітлення робочої зони	порушення гігієнічних параметрів виробничого середовища	3	[9]
підвищена яскравість світла	порушення умов праці (організації місця праці- налагодження моніторів)	1	[10]
понижена контрастність	-	1	[10]
<b><i>психофізіологічні:</i></b>			
нервово-психічна перевантаження (розумове, перенапруження аналізаторів-зорових)	- пошук інформації для постановки теми; - пошук та аналіз аналогів і літератури; - пошук наявних технологій, моделювання та аналіз алгоритмів; - виконання роботи за темою диплома, тестування; - оформлення роботи	4	[5], [10]
фізичні (статичне – сидіння)	порушення умов праці (організації місця праці- сидіння користувача, ) та	2	[5], [10]

	організації робочого часу - безперервна робота)		
--	---	--	--

#### 4.3.2 Пожежна безпека

Небезпека розвитку пожежі на обчислювальному центрі обумовлюється застосуванням розгалужених систем електроживлення ЕОМ та вентиляції. Небезпека загоряння пов'язана з особливістю комп'ютерів - із значною кількістю щільно розташованих на монтажній платі і блоках електронних вузлів і схем, електричних і комутаційних кабелів, резисторів, конденсаторів, напівпровідникових діодів і транзисторів. Надійна робота окремих елементів і мікросхем в цілому забезпечується тільки в певних інтервалах температури, вологості і при заданих електричних параметрах. При відхиленні реальних умов експлуатації від розрахункових можуть виникнути пожежонебезпечні ситуації.

Для відводу теплоти від ЕОМ діє потужна система кондиціонування. Тому кисень, як окиснювач процесів горіння, є в будь-якій точці приміщень ВЦ.

Простори усередині приміщень в межах, яких можуть утворюватися або знаходиться пожежонебезпечні речовини і матеріали відповідно до [11] відносяться до пожежонебезпечної зони класу П-П4. Це обумовлено тим, що в приміщенні знаходяться тверді горючі та важкозаймісті речовини та матеріали. Приміщенню, у якому розташоване робоче місце, присвоюється II ступень вогнестійкості.

Потенційними джерелами запалювання можуть бути:

- 1) іскри і дуги короткого замикання;
- 2) електрична іскра при замиканні і розмиканні ланцюгів;
- 3) перегріву від тривалого перевантаження,
- 4) відкритий вогонь і продукти горіння,
- 5) наявність речовин, нагрітих вище за температуру самозаймання,
- 6) розрядна статична електрика.

Причинами можливого загоряння і пожежі можуть бути:

- 1) несправність електроустановки;
- 2) конструктивні недоліки устаткування;
- 3) коротке замикання в електричних мережах;
- 4) запалювання горючих матеріалів, що знаходяться в безпосередній близькості від електроустановки.

Для захисту персоналу від дії небезпечних і шкідливих чинників пожежі проектом передбачається застосування промислового протигазу, що фільтрує, з коробкою марки «В» із сірою відміткою забарвлення – захист від неорганічних газів (хлор, фтор, бром, сірководень, сірковуглець, хлорціан, галогени), а цей фільтр не захистить від СО (тобто від чадного газу).

Можливе також відповідне застосування фільтрувальної коробки з маркуванням «СО» із фіолетовим забарвленням на фільтрі означає, що він захищає від Чадного газу. Або фільтру для протигазу з літерним маркуванням «SX» із фіолетовим забарвленням захистить від спец речовин таких як (зарин, зоман та фосген).

### **4.3.3 Електробезпека**

На робочому місці виконуються наступні вимоги електробезпеки: ПК, периферійні пристрої та устаткування для обслуговування, електропроводи і кабелі за виконанням та ступенем захисту відповідають класу зони за ПУЕ (правила улаштування електроустановок), мають апаратуру захисту від струму короткого замикання та інших аварійних режимів. Лінія електромережі для живлення ПК виконана як окрема групова трипровідна мережа, шляхом прокладання фазового, нульового робочого та нульового захисного провідників. Нульовий захисний провідник використовується для заземлення (занулення) електроприймачів. Штепсельні з'єднання та електророзетки крім контактів фазового та нульового робочого провідників мають спеціальні контакти для підключення нульового захисного провідника. Електромережа штепсельних розеток для живлення персональних ПК, укладено по підлозі поруч зі стінами відповідно до затвердженого плану розміщення обладнання та технічних характеристик обладнання. Металеві труби та гнучкі металеві рукави заземлені. Захисне заземлення включає в себе заземлюючих пристроїв і провідник, який з'єднує заземлюючий пристрій з обладнанням, яке заземлюється - заземлюючий провідник.

## **4.4 Гігієнічні вимоги до параметрів виробничого середовища**

### **4.4.1 Мікроклімат**

Мікроклімат робочих приміщень – це клімат внутрішнього середовища цих приміщень, що визначається діючої на організм людини з'єднанням температури, вологості, швидкості переміщення повітря. У даному приміщенні проводяться роботи, що виконуються сидячи і не потребують динамічного фізичного напруження, тому для нього відповідає категорія робіт Іа. Отже оптимальні значення для температури, відносної вологості й рухливості повітря для зазначеного робочого місця відповідають [11] і наведені в табл. 4.4:

Таблиця 4.4 – Норми мікроклімату робочої зони об'єкту

Період	Категорія	Температура	Відносна	Швидкість руху
--------	-----------	-------------	----------	----------------

року	робіт	С <sup>0</sup>	вологість %	повітря, м/с
Холодна	легка-1 а	22 - 24	40 – 60	0,1
Тепла	легка-1 а	23 - 25	40 – 60	0,1

Дане приміщення обладнане системами опалення та системою кондиціонування повітря. У приміщенні на робочому місці забезпечуються оптимальні значення параметрів мікроклімату: температури, відносної вологості й рухливості повітря у відповідності до [3]. Рівні позитивних і негативних іонів у повітрі мають відповідати [3]. Для забезпечення оптимальних параметрів мікроклімату в приміщенні проводяться перерви у роботі співробітників, з метою його провітрювання.

#### 4.4.2 Освітлення

Освітленість приміщення має велике значення при роботі на ПЕОМ. Вона багато у чому визначається колірною і мережевий обстановкою. Для зменшеного поглинання світла стеля і стіни вище панелей (1,5-1,7м). Для забарвлення стіни панелей віддана перевага світлим фарбам.

Основний потік природного світла - зліва.

У проекті, що розробляється, використовувалося суміщене освітлення. У світлий час доби використовувалося природне освітлення приміщення через віконні отвори, в решту часу використовувалося штучне освітлення. Штучне освітлення створюється люмінесцентними лампами.

Джерелом природного освітлення є сонячне світло. Регулярно проводиться контроль освітленості, який підтверджує, що рівень освітленості задовольняє СНіП і для приміщення у світлий час доби достатньо природного освітлення.

*Розрахунок освітлення.*

Для будівель виробництв світловий коефіцієнт приймається в межах 1/6 - 1/10:

$$\sqrt{a^2 + b^2} \cdot S_b \quad (4.1)$$

де  $S_b$  – площа віконних прорізів, м<sup>2</sup>;

$S_n$  – площа підлоги, м<sup>2</sup>.

$$S_{\Pi} = a * b = 48 \text{ м}^2$$

$$S_{\text{Вік}} = \frac{1}{8} * 48 = 6 \text{ м}^2$$

Приймаємо 2 вікна площею  $S=3 \text{ м}^2$  кожне.

Світильники загального освітлення розташовуються над робочими поверхнями у рівномірно-прямокутному порядку. Для організації освітлення у темний час доби передбачається обладнати приміщення, довжина якого складає 12 м, ширина 4 м, світильниками ЛПО2П, оснащеними лампами типа ЛБ (дві по 80 Вт) з світловим потоком 5400 лм кожна.

Розрахунок штучного освітлення виробляється по коефіцієнтах використання світлового потоку, яким визначається потік, необхідний для створення заданої освітленості при загальному рівномірному освітленні. Розрахунок кількості світильників  $n$  виробляється по формулі (4.2):

$$n = \frac{E \cdot S \cdot Z \cdot K}{F \cdot U \cdot M} \quad (4.2)$$

де  $E$  – нормована освітленість робочої поверхні, визначається нормами – 300 лк;

$S$  – освітлювана площа, м<sup>2</sup>;  $S = 48$  м<sup>2</sup>;

$Z$  – поправочний коефіцієнт світильника ( $Z = 1,15$  для ламп розжарювання та ДРЛ;  $Z = 1,1$  для люмінесцентних ламп) приймаємо рівним 1,1;

$K$  – коефіцієнт запасу, що враховує зниження освітленості в процесі експлуатації – 1,5;

$U$  – коефіцієнт використання, залежний від типу світильника, показника індексу приміщення і т.п. – 0,575;

$M$  – число люмінесцентних ламп в світильнику – 2;

$F$  – світловий потік лампи – 5400лм (для ЛБ-80).

Підставивши числові значення у формулу (4.2), отримуємо  $n=3,82$ .

Приймаємо освітлювальну установку, яка складається із чотирьох світильників, кожен із яких складається з двох люмінесцентних ламп, загальною потужністю 640 Вт, напругою – 220 В.

#### 4.5 Вентилювання

У приміщенні, де знаходяться ЕОМ, повітрообмін реалізується за допомогою природної організованої вентиляції (вентиляційні шахти) і установки в віконному отворі автономного кондиціонера БК-2000. Цей метод забезпечує приток потрібної кількості свіжого повітря, що визначається в СНіП (30 м<sup>3</sup> на годину на одного працюючого).

Також має здійснюватися провітрювання приміщення, в залежності від погодних умов, тривалість повинна бути не менше 10 хв. Найкращий обмін повітря здійснюється при наскрізному провітрюванні.

#### 4.6 Заходи з організації виробничого середовища та попередження виникнення надзвичайних ситуацій

Відповідно до санітарно-гігієнічних нормативів та правил експлуатації обладнання наводимо приклади деяких заходів безпеки.

*Заходи безпеки під час експлуатації персонального комп'ютера передбачають:*

- правильне організування місця праці та дотримання оптимальних режимів праці та відпочинку під час роботи з ПК;
- експлуатацію сертифікованого обладнання;
- дотримання заходів електробезпеки;
- забезпечення оптимальних параметрів мікроклімату;
- забезпечення раціонального освітлення місця праці (освітленість робочого місця не перевищувала 2/3 нормальної освітленості приміщення);

*Розрахунок захисного заземлення (забезпечення електробезпеки будівлі).*

Загальний опір захисного заземлення визначається за формулою:

$$R_{ззп} = \frac{R_з \cdot R_n}{R_n \cdot n \cdot \eta_з + R_з \cdot \eta_n}, \quad (4.3)$$

де  $R_з$  - опір заземлення, якими когут бать труби, опори, кути і т.п., Ом;

$R_n$  - опір опори, яке з'єднує заземлювачі, Ом;

$n$  - кількість заземлювачів;

$\eta_з$  - коефіцієнт екранування заземлювача; приймається в межах  $0,2 \div 0,9$ ;  $\eta_з = 0,7$ ;

$\eta_n$  - коефіцієнт екранування сполучної стійки; приймається в межах  $0,1 \div 0,7$ ;  $\eta_n = 0,5$ .

Опір заземлення визначається за формулою:

$$R_з = \frac{\rho}{2\pi \cdot l} \cdot \left( \ln \frac{2 \cdot l}{d} + \frac{1}{2} \ln \frac{4 \cdot t + l}{4 \cdot t - l} \right), \quad (4.4)$$

де  $\rho$  - питомий опір ґрунту, залежить від типу ґрунту, Ом·м;

для піску -  $400 \div 700$  Ом·м; приймаємо  $\rho = 400$  Ом·м;

$l$  - довжина заземлювача, м; для труб - 2-3 м;  $l = 3$  м;

$d$  - діаметр заземлювача, м; для труб - 0,03-0,05 м;  $d = 0,05$  м;

$t$  - відстань від середини забитого в ґрунт заземлювача до рівня землі, м;  $t = 2$  м.

$$R_3 = \frac{400}{2 \cdot 3,14 \cdot 3} \left( \ln \frac{2 \cdot 3}{0,05} + \frac{1}{2} \ln \frac{4 \cdot 2 + 3}{4 \cdot 2 - 3} \right) = 110, \text{ Ом}$$

Опір смуги, що з'єднує заземлювачі, визначається за формулою:

$$R_{ш} = \frac{\rho}{2\pi \cdot L} \cdot \ln \frac{2 \cdot L^2}{b \cdot t^1}, \quad (4.5)$$

де  $L$  - довжина смуги, що з'єднує заземлювачі (м) і приблизно дорівнює периметру будівлі:  $P_{буд.} = 42 \cdot 2 + 38 \cdot 2 = 160$  м;  $L = 160$  м;

$b$  - ширина смуги, м;  $b = 0,03$  м;

$t_1$  - глибина заземлення від рівня землі, м;  $t_1 = 0,5$  м.

Опір смуги, що з'єднує заземлювачі дорівнює 5,99 Ом

Кількість заземлювачів захисного заземлення визначається за формулою:

$$n = \frac{2 \cdot R_3}{4 \cdot \eta_3}, \quad (4.6)$$

де 4 - допустимий загальний опір, Ом;

2 - коефіцієнт сезонності.

Кількість заземлювачів захисного заземлення дорівнює 79.

Визначаємо, що загальний опір захисного заземлення дорівнює 1,7 Ом

Висновок: дане захисне заземлення забезпечує електробезпеку будівлі, так як виконується умова:  $R_{ззп} < 4$  Ом.

3) При виникненню пожеж при роботі на ПЕОМ від таких можливими джерел запалювання як:

- іскри і дуги коротких замикань;
- перегрів провідників, резисторів та інших радіодеталей ПЕОМ, від тривалої перевантаження та наявності перехідного опору;
- іскри при розмиканні і розмиканні ланцюгів;
- розряди статичної електрики;
- необережному поводженню з вогнем, а також вибухи газо-повітряних і паро-повітряних сумішей.

Важливу увагу слід звернути на пожежну безпеку підприємства в цілому і окремих його приміщень. В приміщеннях не повинно накопичуватися сміття, непотрібний папір, мотлох та



ін. речі, які не використовуються у виробничому процесі. Наявний вільний аварійний вихід за межі приміщення в разі пожежі, бути передбачені вогнегасники. Вони повинні бути в робочому стані і перевірятися згідно з нормами. У приміщеннях повинна бути пожежна сигналізація, вогнегасник. У разі виникнення пожежі необхідно повідомити в найближчу пожежну частину, забезпечити інших працівників і по можливості прийняти кроки по запобіганню можливих наслідків та усуненню пожежі.

## **4.7 Охорона навколишнього природного середовища**

### **4.7.1 Загальні дані з охорони навколишнього природного середовища**

Діяльність за темою магістерської роботи, а саме: «Розробка компонент візуалізації архівних даних системи локомотивної безпеки СЛБ-І» в процесі її виконання впливає на навколишнє природне середовище і регламентується нормами діючого законодавства: Законом України «Про охорону навколишнього природного середовища», Законом України «Про забезпечення санітарного та епідемічного благополуччя населення», Законом України «Про відходи», Законом України «Про охорону атмосферного повітря», Законом України «Про захист населення і територій від надзвичайних ситуацій техногенного та природного характеру», Водний кодекс України.

Основним екологічним аспектом у процесі діяльності за даними спеціальностями є процеси впливу на атмосферне повітря та процеси поводження з відходами, які утворюються, збираються, розміщуються, передаються на знешкодження, утилізацію, тощо в ІТ галузі.

Вплив на атмосферне повітря при нормальних умовах праці не спричиняє, бо не має в приміщенні сканерів, принтерів та інших джерел викиду забруднюючих речовин в повітря робочої зони.

У процесі діяльності виникають процеси поводження з відходами ІТ галузі. Нижче надано перелік відходів, що утворюються в процесі роботи:

- батарейки та акумулятори (малі) - III клас небезпеки;
- макулатура - IV клас небезпеки;
- побутові відходи - IV клас небезпеки;
- відпрацьовані люмінесцентні лампи - I клас небезпеки.
- Акумулятор для джерел безперебійного живлення -III клас небезпеки
- Змінні носії інформації – IV клас небезпеки

#### 4.7.2 Визначення впливу та заходів щодо поводження з відходами ІТ галузі

З метою визначення та прогнозування впливу відходів на навколишнє середовище, своєчасного виявлення негативних наслідків, їх запобігання відповідно до Закону України «Про відходи» повинен здійснюватися моніторинг місць утворення, зберігання, і видалення відходів. Відомості про місце утворення та місце розташування відходів зазначаються на «План схемі місці розміщення відходів організації / виробництва» та наводяться у таблиці 4.5.

Таблиця 4.5 - Відомості про місце утворення та місце розташування відходів

№ № з/п	Код та найменування відходів за ДК -005-96	Технологічний процес або виробництво, де утворюються відходи/клас небезпеки	Місце розташування відходу, тара та її кількість, місткість, розміри у разі наявності майданчиків розташування відходів необхідно зазначити тип покриття та наявність даху)	№ на схемі (додається масштабна схема місць розміщення відходів)
11	7710.3.1.26 Лампи люмінесцентні	1	буд. 2, у приміщенні кладової S=10м <sup>2</sup> , в кількість 5 од.	2401-ТХ
22	7720.3.1.01 Відходи комунальні (міські) змішані, у т.ч. сміття з урн (Побутові відходи)	4	зовнішній майданчик зберігання побутових відходів біля буд. 2 S=6м <sup>2</sup> V= 3,1м <sup>3</sup> - 3од.	2401-ТХ
33	7710.3.1.01 Макулатура паперова та картонна (Макулатура)		буд. 2, 1 поверх кім. 6 S =5,0 м. <sup>2</sup>	2401-ТХ
44	Батарейки та акумулятори (малі)	3	буд. 2, кім. 112 V=0,0005 м <sup>3</sup>	2401-ТХ
55	Акумулятор для джерел безперебійного живлення	5	буд. 2, кім. 112 m=5,0 кг.	2401-ТХ

#### **4.8 Висновки до розділу 4**

У результаті проведеної роботи було зроблено аналіз умов праці, шкідливих та небезпечних чинників, з якими стикається робітник. Було визначено параметри і певні характеристики приміщення для роботи над запропонованим проектом, написаному в дипломній роботі, описано, які заходи потрібно зробити для того, щоб дане приміщення відповідало необхідним нормам і було комфортним і безпечним для робітника. Приведені рекомендації щодо організації робочого місця, а також важливу інформацію щодо пожежної та електробезпеки. Було наведено значення температури, вологості й рухливості повітря, необхідна кількість і потужність ламп та інші параметри, значення яких впливає на умови праці робітника, а також – наведені інструкції з охорони праці, техніки безпеки при роботі на комп'ютері. Визначені основні екологічні аспекти впливу на навколишнє природне середовище та зазначені заходи щодо поводження з ними.

#### Література до розділу 4

- [1] НПАОП 0.00-4.12-05 Типове положення про порядок проведення навчання і перевірки знань з питань охорони праці
- [2] НПАОП 0.00-4.15-98 Про розробку інструкцій з охорони праці
- [3] ДСН 3.3.6.042-99 «Санітарні норми мікроклімату виробничих приміщень»
- [4] ДСанПіН 3.3.2.007-98 «Правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин»
- [5] НПАОП 0.00.-1.28-10. Правила охорони праці під час експлуатації електронно-обчислювальних машин. Електробезпека.
- [6] ГОСТ 13109-97. Норми якості електричної енергії в системах електропостачання загального призначення.
- [7] ГОСТ 12.1.030-81. Захисне заземлення. Занулення.
- [8] ГОСТ 12.1.006-84. Електромагнітні поля радіочастот. Допустимі рівні на робочих місцях і вимоги до проведення контролю.
- [9] ДБН В.2.5-28:2015. Природне і штучне освітлення.
- [10] ДСанПіН 3.3.2-007-98. Державні санітарні правила і норми.
- [11] НАПБ Б.03.002-2007 Норми визначення категорій приміщень, будинків та зовнішніх установок за вибухопожежною та пожежною небезпекою.

## ВИСНОВКИ

У ході виконання магістерської атестаційної роботи було розроблено компоненту візуалізації архівних даних, що ввійшла до складу програмного забезпечення «Дешифратор» системи локомотивної безпеки СЛБ-І.

Для досягнення поставленої мети:

1. Розглянуті основні проблеми візуалізації великих даних. Запропоновано використовувати алгоритми спрощення полігональних ланцюгів для візуалізації даних технологічних параметрів системи локомотивної безпеки СЛБ-І.
2. Проведений аналіз алгоритмів спрощення полігональних ланцюгів. Оцінка якості спрощення проводилася шляхом розрахунку позиційних відхилень.
3. Формалізовано задачу програмного опису технологічних параметрів і як наслідок розроблено ієрархію класів, що описують властивості груп параметрів. Розроблено графічну компоненту Qt, що задовольняє усім функціональним вимогам та вимогам точності відображення даних.
4. Зроблено аналіз умов праці, шкідливих та небезпечних чинників, з якими стикається робітник. Приведені рекомендації щодо організації робочого місця, а також важливу інформацію щодо пожежної та електробезпеки. Були наведені інструкції з охорони праці, техніки безпеки при роботі на комп'ютері.