Міністерство освіти і науки України
Національний технічний університет
«Дніпровська політехніка»

Інститут електроенергетики
Факультет інформаційних технологій
Кафедра безпеки інформації та телекомунікацій

ПОЯСНЮВАЛЬНА ЗАПИСКА
кваліфікаційної роботи ступеня магістра

студента *Ліпкіна Микити Олексійовича*

академічної групи *125м-22-2*

спеціальності *125 Кібербезпека*

спеціалізації[1]

за освітньо-професійною програмою *Кібербезпека*

на тему *Розробка акустичного генератора шуму з мовоподібною завадою*

Керівники	Прізвище, ініціали	Оцінка за шкалою		Підпис
		рейтинговою	інституційною	
кваліфікаційної роботи	к.т.н., доц. Мацюк С.М.			
розділів:				
спеціальний	к.т.н., доц. Мацюк С.М.			
економічний	к.е.н., доц. Пілова Д.П.			
Рецензент				
Нормоконтролер	ст. викл. Мєшков В.І.			

Дніпро
2023

ЗАТВЕРДЖЕНО:
завідувач кафедри
безпеки інформації та телекомунікацій
_____ д.т.н., проф. Корнієнко В.І.

«_____»_____20___року

ЗАВДАННЯ
на кваліфікаційну роботу
ступеня магістра

студенту *Ліпкіну Микиті Олексійовичу* **академічної групи** *125м-22-2*
(прізвище ім'я по-батькові) (шифр)

спеціальності *125 Кібербезпека*

за освітньо-професійною програмою *Кібербезпека*

на тему *Розробка акустичного генератора шуму з мовоподібною завадою*

затверджену наказом ректора НТУ «Дніпровська політехніка» від _____ № ____

Розділ	Зміст	Термін виконання
Розділ 1	Методи захисту мовної інформації	02.11.2023
Розділ 2	Спеціальна частина	16.11.2023
Розділ 3	Економічна частина	30.11.2023

Завдання видано _____ Мацюк С.М.
(підпис керівника) (прізвище, ініціали)

Дата видачі: _____

Дата подання до екзаменаційної комісії: _____

Прийнято до виконання _____ Ліпкін М.О.
(підпис студента) (прізвище, ініціали)

РЕФЕРАТ

Пояснювальна записка містить: 105 с., 31 рис., 6 табл., 10 додатків, 12 джерел.

Об’єкт дослідження – процес активного захисту від витоку акустичної мовної інформації.

Метою кваліфікаційної роботи є підвищення ефективності активного захисту акустичної мовної інформації.

Методи дослідження: системний підхід, методи індукції, аналізу, порівняння і синтезу, статистики, імітаційне моделювання.

У спеціальній частині дана характеристика методів та засобів протидії витоку мовної інформації акустичним каналом. Проведено порівняльний аналіз акустичних генераторів шуму. Запропоновано новий алгоритм генерації мовоподібної завади. Виконане імітаційне моделювання згідно з розробленою структурною схемою та оцінка ефективності сформованої завади.

В економічному розділі визначено капітальні витрати на програмні розробки.

Практичне значення роботи полягає у розробці нових принципів побудови генератора мовоподібної завади.

Результати проведених у кваліфікаційній роботі досліджень можуть бути використані при розробці акустичних генераторів шуму, у прикладних задачах криптології та теорії інформації.

Наукова новизна дослідження полягає у розробці генератора мовоподібної завади для української мови.

У подальших дослідженнях необхідно приділити увагу розробці алгоритму швидкої сегментації мови диктора на основі нейронних мереж, збільшенню кількості літер у n-грамах.

МОВОПОДІБНА ЗАВАДА, ГЕНЕРАТОР ЗАВАДИ, АКУСТИЧНИЙ КАНАЛ ВИТОКУ, АКУСТИЧНА МОВНА ІНФОРМАЦІЯ, СИНТЕЗ МОВИ, АНАЛІЗ ТЕКСТУ, ГЕНЕРАЦІЯ ТЕКСТУ, ОЦІНКА РОЗБІРЛИВОСТІ

ABSTRACT

The explanatory note consists of: 105 p., 31 fig., 6 tables, 10 appendices, 12 sources.

The object of study is the process of active preventing the leakage of acoustic speech information.

The aim of the qualification work is to improve the efficiency of active protection acoustic speech information.

Methods: systematic approach, methods of induction, analysis, statistics, comparison and synthesis, and simulation.

The special part describes the methods and means of combating loss of vocal acoustic information channel. The comparative analysis of acoustic noise generators has been done. A new generation algorithm of speech-like masking signal have been proposed and justified. Simulation modelling and the structural scheme have been accomplished. The efficiency of the existing signal has been evaluated.

In the economic section the capital costs of establishing developed programs have been calculated.

The practical significance of the work is to create a block diagram of the generator, which is suitable for implementation in production. Results of the research can be used to produce acoustic noise generators, in applicable problems in cryptology and information theory.

The scientific novelty of the research is to develop a speech-like masking signal generator for the Ukrainian language.

In further research it is necessary to pay attention to develop rapid segmentation algorithm language speaker based on neural networks, to increase the dimension of the n-grams.

SPEECH-LIKE MASKING SIGNAL, NOISE GENERATOR, LEAKAGE OF ACOUSTIC INFORMATION, ACOUSTIC SPEECH INFORMATION, SPEECH SYNTHESIS, TEXT ANALYSIS, TEXT GENERATION, ARTICULATION SCORE

СПИСОК УМОВНИХ СКОРОЧЕНЬ

АС	–	автоматизована система;
ІБ	–	інформаційна безпека;
ІТ	–	інформаційні технології;
ІКС	–	інформаційно-комунікаційна система;
КЗЗ	–	комплекс засобів захисту;
ЗІ	–	захист інформації;
КСЗІ	–	комплексна система захисту інформації;
НСД	–	несанкціонований доступ;
НД ТЗІ	–	нормативний документ технічного захисту інформації;
ОС	–	обчислювальна система;
ПБ	–	політика безпеки;
ПЗ	–	програмне забезпечення.

ЗМІСТ

ВСТУП

Людська мова є найбільш давнім і розповсюдженим способом обміну інформацією. В даний час, незважаючи на появу і розвиток різноманітних засобів і способів обміну інформацією, частка мовної інформації складає близько 80%. Інтерес до перехоплення мовної інформації з часом не те що не пропадає, а навіть збільшується.

У зв'язку з цим одним із найважливіших, що не втрачає своєї актуальності, завдань захисту інформації є протидія витоку мовної інформації акустичним каналом.

Попередження витоку акустичної інформації акустичними каналами ведеться пасивними та активними мірами. Зважаючи на труднощі організації та виконання пасивних методів, особливо у випадку орендованого приміщення або виїзних засідань, їх застосування на практиці обмежене. Активні методи протидії засновані на використанні генераторів, що за видом завади поділяються на два класи: шумової та мовоподібної завади.

Широко розповсюджені генератори «білого» шуму генерують сигнал, що принципово відрізняється від приховуваного мовного, тому значно перевищує сигнал у частині спектру. Це доставляє незручності учасникам переговорів та сприяє їх підвищеній втомлюваності.

У рамках удосконалення апаратури протидії витоку інформації, розширення функціональних можливостей і зменшення негативного впливу були розроблені алгоритми генерації мовоподібної завади. Відмінність даного типу полягає у подібності спектру завадового сигналу до природної мови, через що відбувається рівномірне покриття сигналу завадою і, як наслідок, відсутність необхідності збиткової потужності та зменшення значення розбірливості при однаковому відношенні сигнал/шум у порівнянні з шумовими.

На сьогодні в Україні не випускають генератори мовоподібних завад, а також у публічному доступі відсутні розробки у цьому напрямі.

Таким чином, актуальним науковим завданням, що має теоретичне і практичне значення, є розробка нових, удосконалення та адаптація існуючих алгоритмів генерації мовоподібних завад.

Метою кваліфікаційної роботи є підвищення ефективності активного захисту акустичної мовної інформації.

Для досягнення зазначеної мети кваліфікаційної роботи поставлені окремі завдання:

- провести аналіз сучасного ринку генераторів акустичного шуму та існуючих алгоритмів генерації мовоподібної завади;
- визначити вимоги до розроблюваного генератора;
- розробити структурну схему пропонованого генератора;
- на основі запропонованих рішень провести моделювання для оцінки якісних характеристик сформованої завади.

Об'єкт дослідження – процес активного захисту від витоку акустичної мовної інформації.

Предмет досліджень – алгоритми формування мовоподібної завади, що забезпечують більш ефективне приховання мовного сигналу у порівнянні з генераторами «білого» шуму.

При вирішенні поставлених завдань у кваліфікаційній роботі використані методи наукової абстракції, індукції, аналізу, порівняння і синтезу (при розкритті теоретичних положень та розроблені структурної схеми); імітаційне моделювання (при оцінці ефективності завади); методи статистики (при підрахунку відносних частот повторюваності); генерацію на основі n-грам та метод оцінювання мовленнєвої розбірливості.

Наукова новизна одержаних результатів:

- розроблено структурну схему генератора на основі фонемного клонера для української мови;
- проведено аналіз частот повторюваності монограм, біграм, триграм для тексту українською мовою обсягом понад 100 млн. слів;

- розроблено генератор, що враховує частотний розподіл літер природніх текстів.

Практична цінність роботи полягає в наступному:

- розробці структурної схеми генератора мовоподібної завади, що дозволяє створити простий і компактний генератор завади;

- створенні корпусу української мови об'ємом понад 800 Мб, що може бути використаний у прикладних задачах криптографії, лінгвістики та теорії інформації;

- аналізі відносних частот повторюваності мови на основі великого корпусу, що можна використати у практичних додатках криптографії та теорії інформації;

- розробці програми генерації псевдотексту, що дозволяє створити унікальні тексти змінної довжини, з характеристиками наближеними до природної мови.

РОЗДІЛ 1. МЕТОДИ ЗАХИСТУ МОВНОЇ ІНФОРМАЦІЇ

1.1 Структура акустичного каналу витоку інформації

Звуком називаються механічні коливання частинок пружного середовища (повітря, води, металу і т.д.), які викликані певним джерелом звуку та суб'єктивно сприймаються органом слуху. Вважається, що людське вухо розпізнає коливання середовища в смузі частот від 16 Гц до 20 кГц. Процес поширення коливального руху в середовищі називається звуковою хвилею.

Однією з найважливіших характеристик акустичного сигналу є звуковий тиск. Звуковий тиск - це змінний тиск в середовищі, обумовлений розповсюдженням в ньому звукових хвиль.

Поріг чутності – найбільш тихий звук, який ще здатна чути людина на частоті 1000 Гц, що відповідає звуковому тиску 2×10^{-5} Па.

Рівень звукового тиску вимірюється в децибелах і обчислюється за формулою (1.1).

$$L_{дБ} = 20 \lg \frac{P_c}{P_0}. \qquad (1.1)$$

де P_c – значення виміряного звукового тиску;

P_0 - поріг чутності.

Діапазон основних звукових частот мовлення лежить в межах від 70 до 1500 Гц. З урахуванням обертонів мовної діапазон розширюється до 5000-8000 Гц. Гучність мови при звичайній розмові близько 60 дБ.

В акустичному каналі витоку носієм інформації від джерела до несанкціонованого одержувачу є звукова хвиля в газоподібному, рідкому або твердому середовищі, де на неї впливають навмисно створені, природні, випадкові та інші засоби.

Джерелами звукового сигналу можуть бути: диктор, що веде розмову конфіденційного характеру, технічні засоби звуковідтворення, механічні вузли обладнання, які при роботі формують акустичні хвилі. Технічними засобами знімання інформації з цього каналу є акустичні приймачі, такі як мікрофони, стетофони, геофони, гідрофони тощо.

В подальшому сигнал може передаватися навмисно або ненавмисно до зловмисника засобами телефонного зв'язку, лініями електропередач, радіоканалом, оптичним каналом тощо, або записуватися на носій. Спрощена схема акустичного каналу витоку представлена на рисунку 1.1.

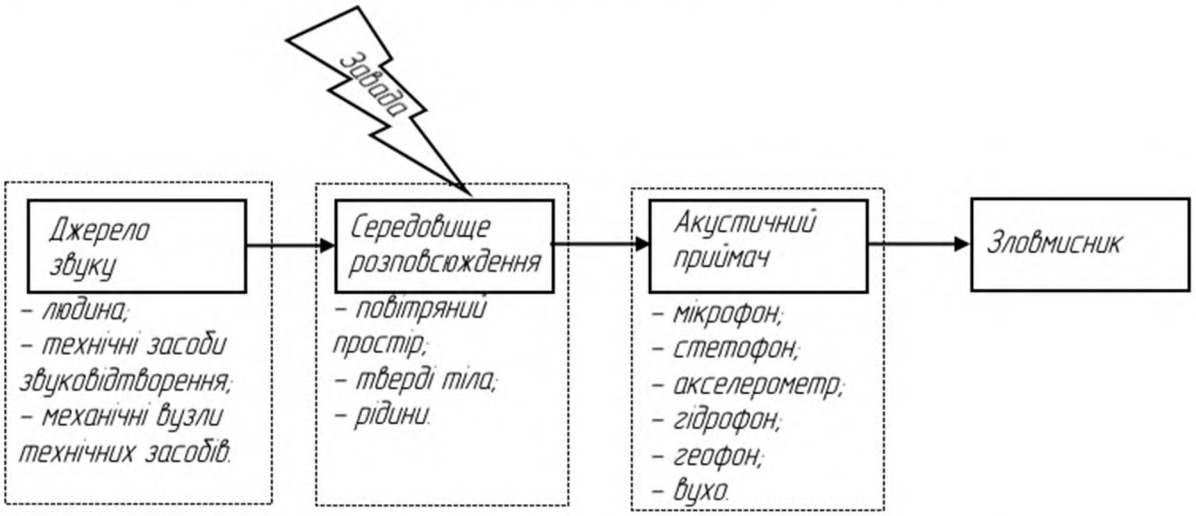

Рисунок 1.1 – Структура акустичного каналу витоку інформації

1.2 Аналітичний огляд методів і засобів захисту акустичної інформації

Завданням заходів з технічного захисту інформації є або ліквідація каналів витоку інформації, або зниження якості отримуваної зловмисником інформації. Основним показником якості мовної інформації вважається розбірливість - складова, словесна, фразова і ін. Прийнято вважати, що якість акустичної інформації достатня, якщо забезпечується біля 40% складової розбірливості.

Протидія витоку інформації акустичними каналами зводиться до пасивних і активних способів захисту. Як наслідок, всі засоби і методи захисту інформації можна розділити на три великих класи – пасивні, активні,

а також комбіновані. Найбільш повна класифікація методів захисту представлена на рисунку 1.2.

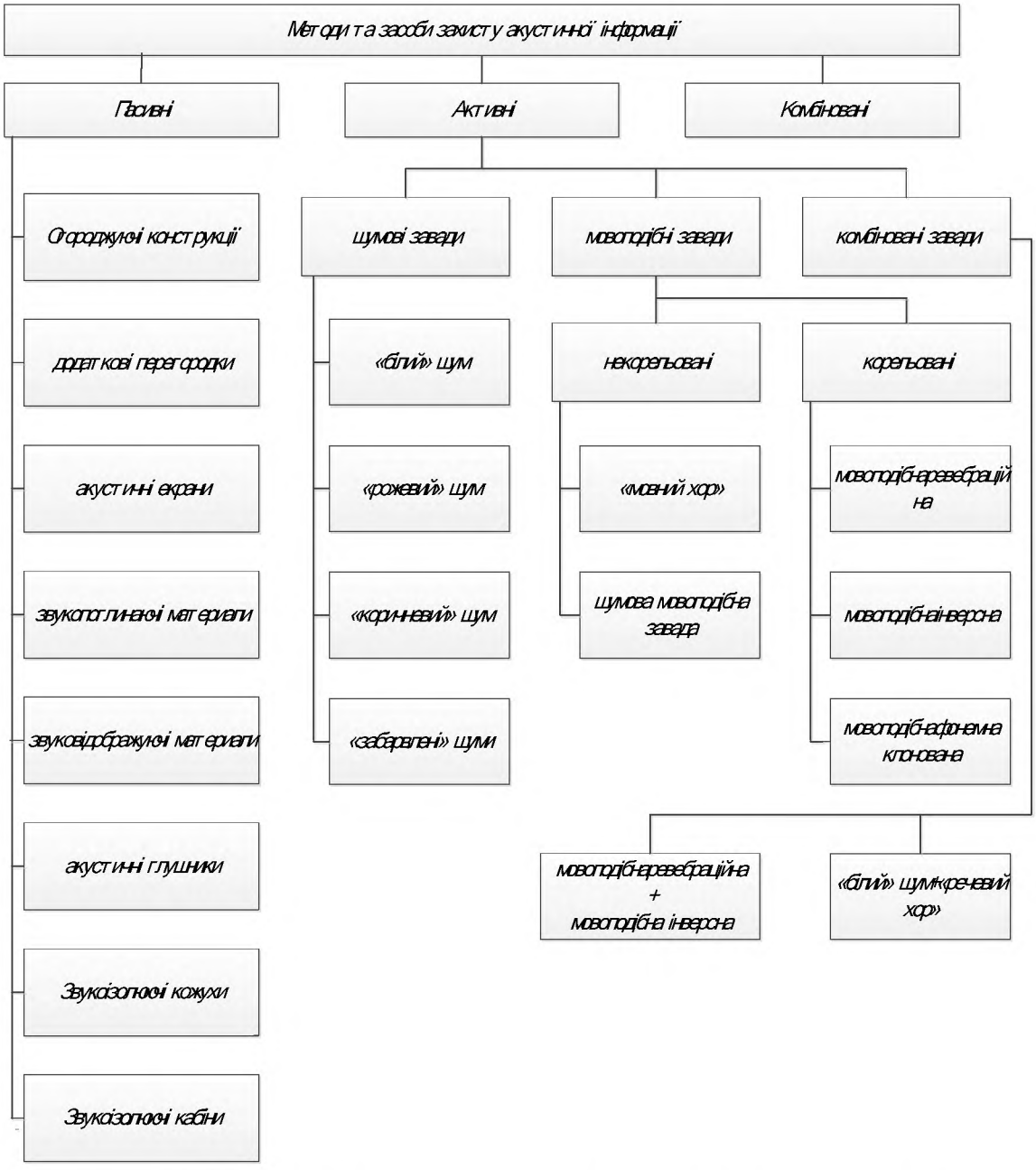

Рисунок 1.2 – Класифікація методів та засобів захисту акустичної інформації

1.2.1 Пасивні методи

Метою пасивних методів є послаблення акустичних сигналів в середовищі поширення сигналу до величин, що забезпечують неможливість

їхнього виділення засобом розвідки на фоні природних шумів на границі контрольованої зони. Іншими словами, пасивні методи спрямовані на погіршення розбірливості шляхом максимального ослаблення сигналу.

Пасивні методи реалізуються комплексом проєктних та будівельно-монтажних заходів, спрямованих на:

- доопрацювання огороджувальних конструкцій (стін, підлоги, стелі, вікон, дверей) виділеного приміщення за допомогою звукопоглинаючих і звуковідображуючих матеріалів;
- звукоізоляцію систем інженерного забезпечення (припливно-витяжної вентиляції, опалення, кондиціонування) із застосуванням акустичних глушників і звукоізолюючих корпусів;
- поділ виділеного приміщення на зони за допомогою спеціальних ізолюючих перегородок або установкою переговорних кабін;
- іншими методами.

Перелік методів і матеріалів, що застосовуються дуже великий і можна домогтися зниження рівня «небезпечного» сигналу до прийнятного, але всі вони мають істотний недолік – їх реалізація в повному обсязі можлива тільки на стадії будівництва або реконструкції виділеного приміщення. Це неможливо у разі орендованого приміщення або виїзної наради. Також до недоліків можна віднести зміну параметрів захищеності об'єкта при виконанні будівельних робіт поза межами приміщення (наприклад, при появі отворів у стіні з суміжним приміщенням).

1.2.2 Активні методи

Активні методи захисту мовної інформації від витоку акустичним і віброакустичним каналами полягають в зниженні розбірливості акустичного сигналу до прийнятного рівня на кордонах контрольованої зони шляхом підвищення рівня шуму. Їх метою є створення завад, які під час перехоплення і подальшої обробки суміші мовного сигналу з шумом, не дозволять відновити вихідний сигнал.

Активний технічний засіб захисту – пристрій, що забезпечує створення маскувальних активних завади (або імітують їх) для засобів технічної розвідки або порушують нормальне функціонування засобів негласного знімання інформації.

На практиці найбільш широке застосування знайшли генератори шумових коливань, які поділяються за типом генерованих сигналів на шумові, мовоподібної завади, а також комбіновані.

Шумові генератори працюють на принципі постановки випадкової за частотою та амплітудою завади, яку складно відфільтрувати, в тій же смузі частот, що і приховуваний сигнал.

Формування завад типу «білий», «рожевий» і «коричневий» шум отримують шляхом модуляції щодо високочастотного несучого сигналу флуктуаційними шумами, створюваними електровакуумними, газорозрядними, напівпровідниковими радіоелементами або за допомогою алгоритмів генерації псевдовипадкових послідовностей.

«Білий» шум має постійну спектральну щільність у всій смузі мовних частот, «рожевий» - шум зі спадом 3 дБ на октаву убік високих частот, «коричневий» - 6 дБ на октаву. Спрощено спектральні характеристики шумових сигналів з уніфікованим спектром мовного сигналу показані на рисунку 1.3.

Загальним недоліком описаних типів завад є надлишкова спектральна потужність сигналу в окремих смугах частот (особливо для «білого» шуму), що створює додаткову шумове навантаження та, як наслідок, дискомфорт учасникам переговорів, бажання «перекричати» заваду. Також відміна спектрів шумових сигналів від мовного дає можливість його компенсації при недостатньому рівні шумової завади.

Інші пофарбовані шуми рідко застосовуються при захисті мовної інформації.

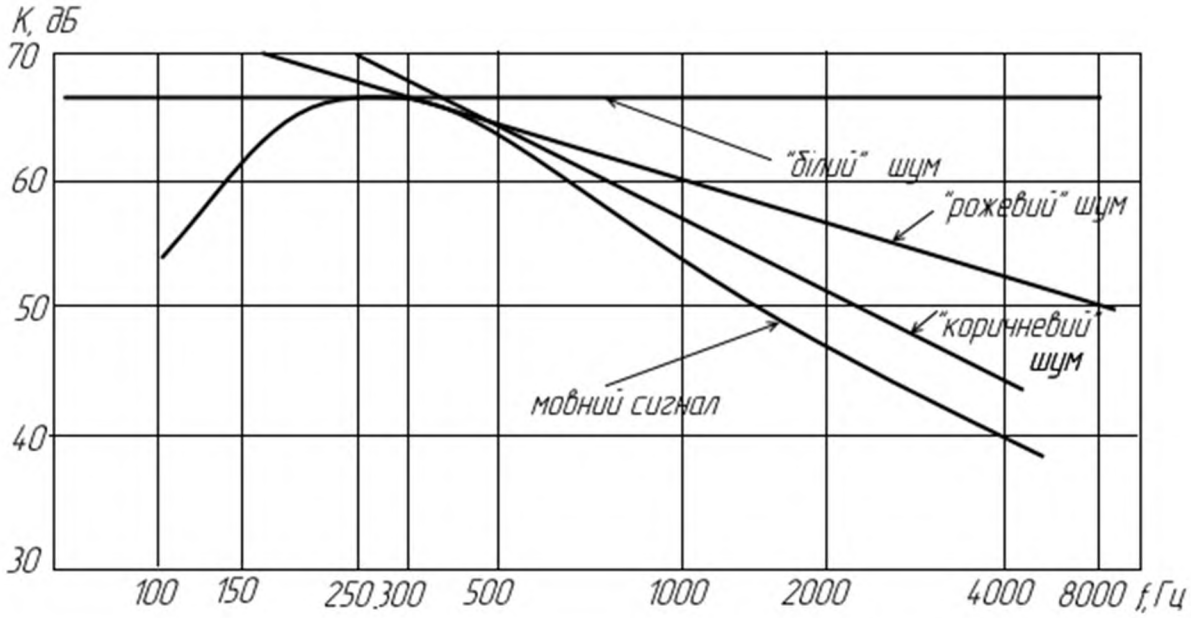

Рисунок 1.3 – Спектральні характеристики завад та мови

1.2.3 Мовоподібні завади

Недоліків «фарбованих» шумів не мають так звані «мовоподібні» завади.

Суть методу формування мовоподібної шумової завади полягає в заповненні обвідної (або середньої обвідної) мовного сигналу завадою типу «білий» шум в режимі реального часу як показано на рисунку 1.4.

Даний метод вимагає надзвичайної швидкодії і має велику обчислювальну складність, тому мало використовується на практиці.

Завада типу «мовний хор» формується з фрагментів кількох різних поєднань відрізків мовних сигналів і музичних фрагментів мовних радіостанцій. Метод дозволяє домогтися спектральної характеристики, подібної до приховуваного сигналу і, як наслідок, рівномірного покриття спектра приховуваного сигналу без значної надмірності, але заваду можливо компенсувати при розкритті радіостанцій, що використовуються.

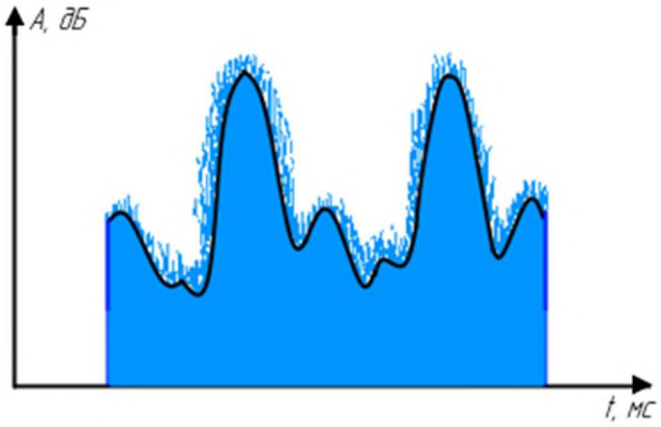

Рисунок 1.4 – Осцилограма мовного сигналу, зашумленого мовоподібною шумовою завадою

Із приховуваного мовного сигналу формують три типи завад: мовоподібну реверберaційну, мовоподібну інверсивну і заваду із застосуванням фонемного клонера.

Реверберaційні завади формуються з фрагментів приховуваного мовного сигналу шляхом багаторазового їх накладення з різними рівнями. На практиці це здійснюється за рахунок приймання постановником завади акустичного сигналу в приміщенні, вибірки з нього відрізків за заданим алгоритмом і випромінювання в ефір. Алгоритм повторюється циклічно. Відображення сигналу відбувається при його перевідбитті від огороджувальних конструкцій і предметів в приміщенні, а також за рахунок часткового поглинання енергії хвилі різноманітними предметами у приміщені.

Мовоподібна інверсійна завада формується з приховуваного мовного сигналу шляхом складної інверсії його спектра. Прийнятий сигнал передається в блок обробки, де відбувається множення і ділення його частотних складових. Отримана в результаті цього процесу завада озвучується акустичними пристроями мововідтворення. Процес відбувається циклічно.

Крім високої надійності у таких генераторів наявна суттєва перевага - вони працюють тільки тоді, коли ведеться бесіда (коли в приміщенні тихо, то і шуми не створюються).

Фонемні клонери призначені для синтезу мовоподібних завад, оптимізованих для захисту мовної інформації, озвученої конкретними дикторами. Формування завадового сигналу відбувається в два етапи - на першому етапі за допомогою спеціального програмного забезпечення з запису розмови диктора синтезується «псевдомова» шляхом клонування основних фонемний складових. На другому етапі синтезатор завади, в пам'яті якого міститься «псевдомова», за випадковим законом бере з цієї послідовності сигналів випадкові сегменти, які надходять на вхід тракту постановника завадового сигналу.

В результаті аналізу виявлено, що головним недоліком шумових завад є суттєва відмінність сигналу завади у порівнянні з приховуваним сигналом. Це призводить до нерівномірного покриття спектра, і, як наслідок, надмірності енергії шуму, що поставляється, в частині спектру. Цього недоліку позбавлені мовоподібні корельовані завади, так як вони мають найбільш близький до природної мови людини спектр. Виходячи з цього можна припустити, що вони володіють найкращою якістю акустичної маскування.

При аналізі матеріалів з даної тематики дане припущення підтверджує ряд досліджень.

Так, при аналізі формантної розбірливості показано, що найбільш ефективними є завади типу «рожевий» шум і шумова мовоподібна завада. В одній з робіт показано, що значення словесної розбірливості, визначеної інструментально-розрахунковим способом, набагато нижче при використанні мовоподібних комбінованих (реверберацийних і інверсійних) завад у порівнянні з шумовими.

Результати аналізу можливості відновлення завади сигналу методом сонограм та при обчисленні розбірливості показали значну перевагу мовоподібних сигналів над іншими типами шумів.

1.3 Аналітичний огляд ринку генераторів акустичного шуму

У «Перелік засобів технічного захисту інформації, дозволених для забезпечення технічного захисту державних інформаційних ресурсів та інформації, вимога щодо захисту якої встановлена законом» увійшли два генератора акустичного шуму: «Марс - ТЗО-4-2» і «ОЦЗІ-ВА».

Також дія сертифікатів ряду генераторів акустичного шуму закінчилась на момент аналізу: «СКЕЛЯ-2Г», «DNG-2300», «Топаз ГША-4», «PIAC-2C», «Базальт-4ГА». Всі зазначені генератори формують заваду типу «білий» шум. Порівняльна характеристика генераторів приведена в таблиці 1.1.

Таблиця 1.1 – Порівняльна характеристика генераторів шуму

Генератор	Експлуатаційні параметри				
	діапазон робочих частот, Гц	кількість каналів, шт.	глибина регулювання, не менше, дБ	вихідна потужність, Вт	спосіб регулювання
ОЦЗІ-ВА	180..5600	2	40	70	еквалайзер
Марс -ТЗО	170..5700	2	20	-	еквалайзер
СКЕЛЯ-2Г	170..5700	2	8	10	еквалайзер
DNG-2300	250..6500	3	24	10	еквалайзер
Топаз ГША	170..5700	2	24	4	еквалайзер
PIAC-2C	180..5600	1	20	10	еквалайзер
Базальт-4ГА	170...5700	2	25	-	еквалайзер

Широко розповсюджені генератори «білого» шуму, хоча і мають регулювання спектра, як правило, в 5 октавних смугах, але її глибина не

дозволяє впоратися з надмірністю звукового тиску. Середній діапазон частот, що покриваються - 170..5700 Гц.

Також з'ясовано, що випуск генераторів мовоподібної завади не налагоджений в Україні. Доступні на ринку України постачальники генерують заваду при використанні комбінації реверберації і інверсії спектра.

Відзначимо також, що серед доступних на сьогоднішній день відсутні генератори, що працюють на основі фонемного клонера.

1.4 Аналіз існуючих алгоритмів генерації мовоподібної завади

На даний момент розроблено кілька алгоритмів генерації завад з мовоподібними характеристиками, деякі з яких розглянемо нижче.

Відомим і найбільш вдалим представником генераторів мовоподібної завад є генератор, розроблений Хорєвим А.А. у співавторстві. Метод формування базується на комбінації псевдореверберації і інверсії вихідного мовного сигналу. Структурна схема генератора наведена на рисунку 1.5.

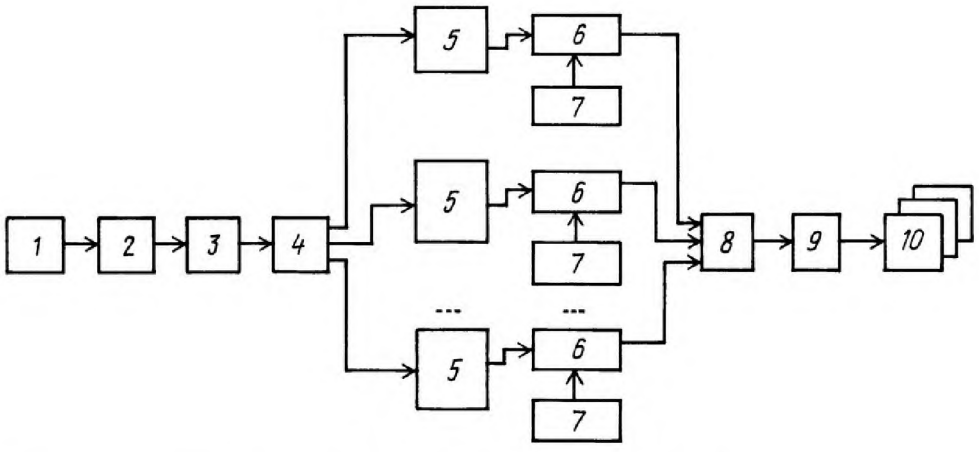

Рисунок 1.5 – Структурна схема генератора завад

Робота генератора відбувається наступним чином. Приховуваний акустичний сигнал перетворюється мікрофоном 1 в електричний і подається на мікрофонний підсилювач 2. Посилений сигнал з мікрофонного підсилювача 2 надходить на логарифмічний підсилювач 3, а потім через

розгалужувач 4 - на n смугових фільтрів 5. З виходу кожного смугового фільтра 5 сигнал подається на відповідний аналоговий скремблер 6, який виробляє інверсію спектра сигналу. Управління роботою аналогових скремблерів 6 здійснюється генераторами випадкових імпульсів 7, при цьому частота точки інверсії кожного скремблера 6 змінюється за випадковим законом в межах частот від f_{n1} до f_{n2} через інтервал часу t_u. Далі сигнали з аналогових скремблерів 6 подаються на суматор 8. На суматорі 8 n шумових сигналів складаються і надходять на підсилювач потужності 9, який підсилює їх до заданого рівня. Далі шумовий сигнал надходить на випромінювачі 10 (звукові колонки або вібровипромінювачі).

Зміни точки інверсії за випадковим законом не дозволяє виділити приховуваний сигнал із завадового. Алгоритм формування мовоподібної завади базується на принципі фонемного клонера і включає в себе верифікацію, сегментацію і класифікацію мови з подальшою компіляцією ділянок записів мови дикторів на відповідних мовах. Верифікація мови в алгоритмі застосовується для визначення зміни диктора на основі його індивідуальних характеристик мови. Перед використанням в базу даних генератора заносяться індивідуальні характеристики мови, а також визначається мова для подальшого формування акустичної завади. В цей же момент формується індивідуальна база алофонів, з якої в подальшому вибираються відрізки за випадковим законом і формується мовоподібний сигнал (рисунок 1.6).

В подальшому алгоритм був модифікований шляхом включення в структуру генератора "білого шуму" і суматора. Структурна схема представлена на рисунку 1.7.

Рисунок 1.6 – Схема алгоритму синтезу мовоподібних завад

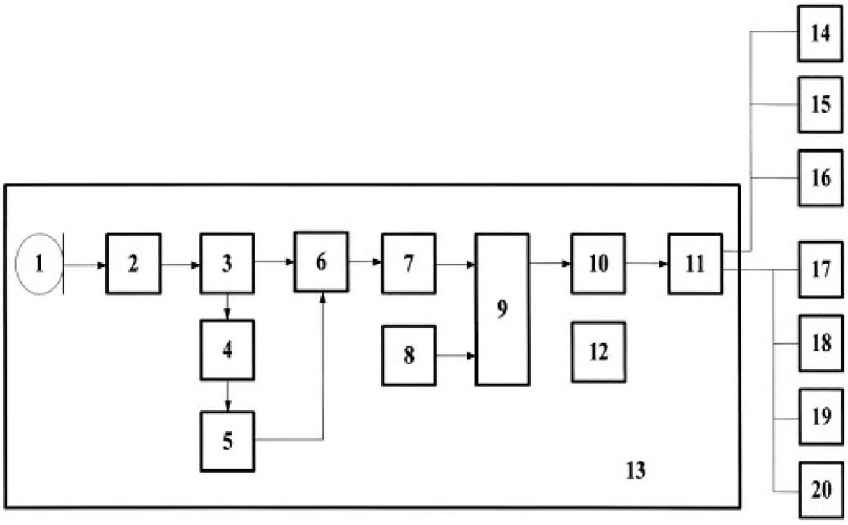

Рисунок 1.7 - Структурна схема постановника комбінованої завади

Формування завади відбувається наступним чином. Мікрофон 1 вловлює зміни акустичної обстановки і передає їх в блок детектування мови 2, який передає керуючий сигнал в блок верифікації диктора за голосом 3 при наявності мови. Блок верифікації передає сигнал в базу алофонів 6 у разі підтвердження особи і в блок сегментації мови 4, якщо особистість не

підтверджена. Після цього мова розбивається на алофони, які класифікуються в блоці класифікації мови 5 і надходять до бази алофонів дикторів 6. На основі бази алофонів генератор мовоподібних сигналів 7 формує сигнали, які надходять на суматор сигналів 9 і складаються з сигналами, які надходять з генератора «білого» шуму 8, в пропорціях за середньоквадратичними значеннями, рівними від 3 до 15 дБ. З виходу суматора 9 складний маскуючий сигнал надходить до блоку управління 10, для регулювання рівня та у підсилювач потужності 11. До виходу підсилювача потужності підключені акустичні 14, 15, 16, і вібраційні перетворювачі 17, 18, 19, 20.

Альтернативний спосіб створення завади полягає в формуванні багатоголосої завади з одноголосих мовних завад у вигляді послідовно записаних голосів з різних тембром в діапазоні 50-15000 Гц. Структурна схема представлена на рисунку 1.8.

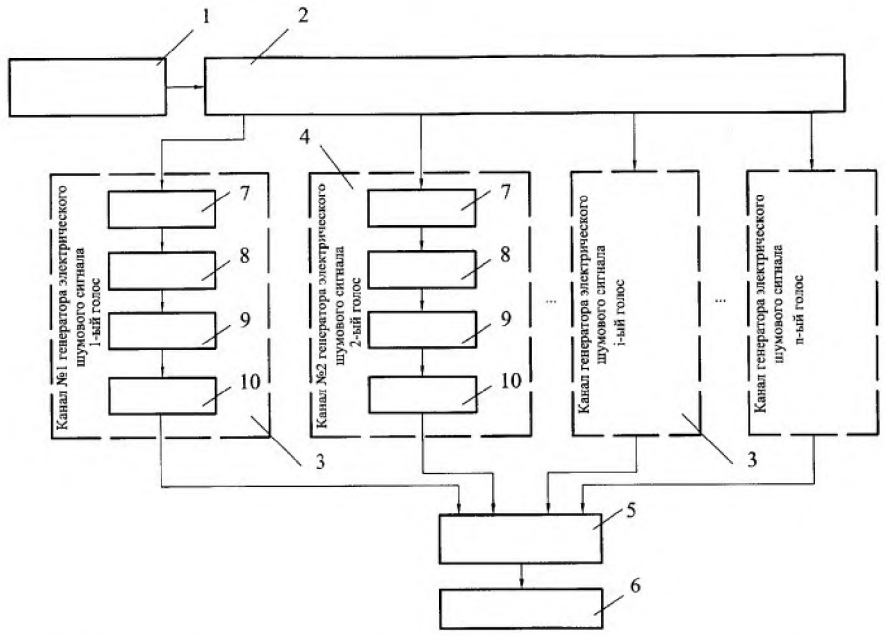

Рисунок 1.8 - Структурна схема генератора багатоголосої завади

Послідовність формування завади наступна. Генератор 1 тактових імпульсів формує послідовність імпульсів, які через розподільник 2

потрапляють на входи відповідних каналів 3, 4 генератора електричного шумового сигналу. Тактові сигнали на вході кожного з каналів 3, 4 запускають блок 7 формування випадкових чисел, який виробляє випадкове дробове число N, що має рівномірний розподіл в інтервалі від 0 до 1, і подає його на вхід блоку 8 розподілу ймовірностей вживання звуків мови. У блоці 8 на основі розподілу ймовірностей вживання звуків мови зіставляє випадкове дробове число N і тип звуку з можливих для сучасного вимови.

Тип звуку, визначений блоком 8, передається в з'єднаний послідовно з ним блок 9 зберігання і вибірки звуків мови, в якому виконується пошук і порівняння заданого типу звуку з зберігаються в базі фонограм. В результаті порівняння в блоці 9 визначається фонограма, яка відтворюється в блоці 10 відтворення фонограм елементарних звуків мови.

Мікшуючий підсилювач потужності 5 перетворює одноголосі мовоподібні завади в багатоголосу мовну заваду за допомогою накладення відтворюваних випадкових послідовностей звуків і підсилює її. Підключений до нього випромінювач 6 відтворює отриманий сигнал в приміщенні.

При цьому заміна наборів фонограм звуків мови 9 і заміна розподілу ймовірностей вживання звуків мови 8 дозволяє адаптувати формується мовоподібну заваду до особливостей вимови слів мови.

Перший алгоритм генерації, є достатньо простим і обґрунтованим для застосування в системах захисту інформації. Однак використання самого приховуваного сигналу при формуванні завади може виявитися «небезпечним» в наступному через стрімкий розвиток можливостей обчислювальної техніки. Також завдання компенсації реверберaційної складової активно розробляється в рамках сучасних математичних методів в області ехолокації і не виключено, що результати досліджень не будуть застосовані для компенсації цієї шумовий завади.

Другий алгоритм має дуже громіздку структуру, що вимагає значних обчислювальних витрат та призведе до удорожчання виробу, що реалізує даний алгоритм. Також на даний момент не розроблено алгоритмів швидкого

автоматичного створення повного набору алофонів, а їх сегментація оператором займе велику кількість часу. Введення в структуру генератора «білого» шуму нівелює основну перевагу даного типу генераторів привносячи надмірність в покриття спектра вихідного сигналу.

Третій алгоритм не прив'язаний до голосових характеристик диктора. Його структура надмірно ускладнена, не враховує формантного розподілу натуральної мови та мало відрізняється від генераторів на основі мікшування дикторських радіостанцій.

1.5 Висновки до розділу 1 та постановка задачі

В ході аналізу були виявлені недоліки шумових завад, що широко застосовуються і перевагу в якості акустичної маскування мовоподібних завад. При аналізі ринку акустичних генераторів шуму відзначено відсутність серійного виробництва постановників мовоподібної завади в Україні. Всі доступні на ринку моделі подібних пристроїв працюють на основі реверберації і її комбінацій. На даний час відсутні у публічному доступі та у патентних базах придатні до практичної реалізації алгоритми встановлення завади на основі фонемного клонера.

Виходячи з цього, доцільна розробка алгоритму генерації за принципом фонемного клонера, зважаючи на просту і компактну реалізацію подібних алгоритмів. Згенерована завада повинна мати максимальне наближення до природної мови в приміщенні, її спектральні характеристики повинні забезпечувати енергетичний виграш в порівнянні з «білим» шумом не менше 1.5-2 дБ при однаковому приховуванні мовного сигналу. Спираючись на результати аналізу, смуга робочих частот розроблювального пристрою повинна бути не менша 170..5700 Гц.

РОЗДІЛ 2. СПЕЦІАЛЬНА ЧАСТИНА

2.1 Розробка структурної схеми генератора мовоподібної завади

2.1.1 Обґрунтування структурної схеми генератора мовоподібної завади

На основі аналізу алгоритмів генерації, проведеного у розділі 1, запропонована структурна схема генератора, що об'єднала у собі переваги алгоритмів.

Структурна схема генератора надана на рисунку 2.1.

Генератор працює за принципом синтезатора та має модульну структуру.

Ідея алгоритму полягає в використанні забарвленого голосом диктора мовоподібного сигнала, що не несе смислового навантаження, статистичні характеристики якого можна порівняти з характеристиками мови, якою ведуться переговори. Для цього формується база одиниць мови для синтезу.

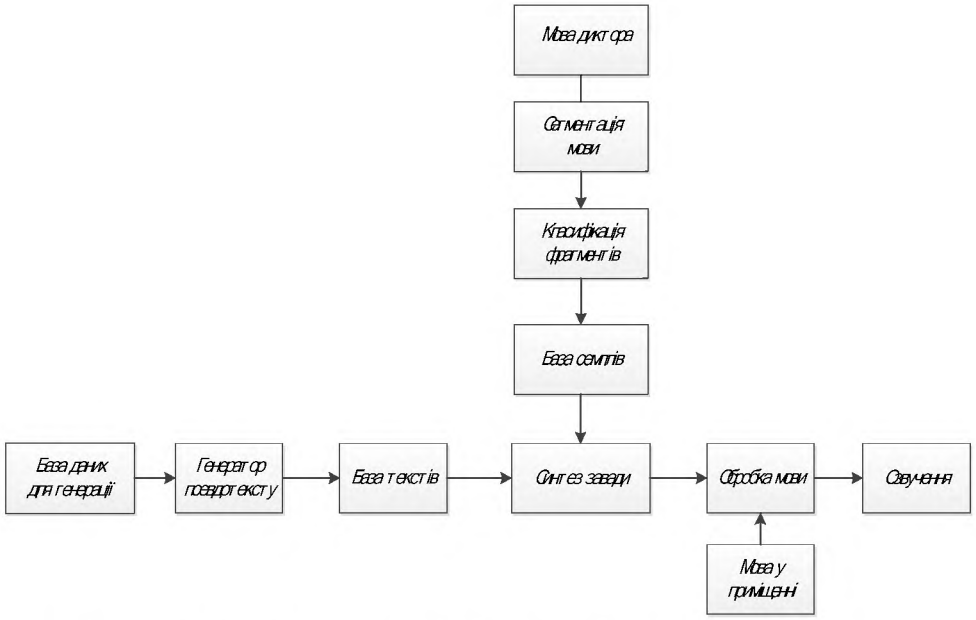

Рисунок 2.1 – Структурна схема генератора завади

Формування бази відбувається в кілька етапів. На першому етапі мова диктора проходить попередню фільтрацію для видалення фонового шуму на записі. Після цього відбувається розбиття сигналу на кадри, відповідні

фонемам і проводиться класифікація отриманих сегментів і формування наборів «фонемна структура-елемент», які записуються в базу даних для їх подальшого використання при синтезі. Це дозволяє попередньо сформувати записи сегментів і поповнити базу даних постановника завади.

Для формування завади використовується генератор псевдотексту. Генератор повинен забезпечувати генерацію тексту зі статистичними характеристиками, подібними до натуральної мови. Після цього текст заноситься в базу даних для подальшого його озвучування синтезатором. Текст також може бути сформований заздалегідь і внесений в базу текстів.

Після формування баз даних синтезатор формує мовний сигнал, озвучений голосом диктора. Подальша обробка сигналу полягає в підборі гучності звучання та швидкості мовлення для якісного приховування сигналу.

Для вибору алгоритму генерації псевдотексту, сегментації мови та синтезу завади проведений аналіз існуючих методів.

2.1.2 Вибір метода генерації текстів

Псевдотекст – набір одиниць письмової мови (символів, складів, слів тощо) та знаків пунктуації, що має подібну до тексту структуру, проте може зовсім не мати змісту. В загальному випадку слова, з яких складається псевдотекст, можуть не існувати.

На даний час розроблено безліч алгоритмів генерації, що дозволяють отримати псевдотекст, характеристики якого наближені до природної мови. Найбільш складні алгоритми створюють тексти, які складно відрізнити від авторських. Найбільш розповсюджені методи генерації, що працюють на основі слів та символів. Розглянемо деякі з них.

2.1.2.1 Генератори, що оперують над вхідним текстом

До даного типу належать рандомайзери, синономайзери, алгоритми розмноження.

Рандомайзери зчитують послівно вхідний текст і кожне слово або вибірково замінюють на слова з словника. При цьому словники можуть бути сформовані як для кожної окремої позиції окремо, так і з одного загального. Найбільш досконалі алгоритми замінюють однакові частини мови з узгодженням закінчень.

Синономайзери працюють за подібним алгоритмом, проте слова замінюються синонімами. У порівнянні з рандомізатором кількість отриманих вихідних текстів з одного вхідного значно менша.

Алгоритм розмноження тексту схожий на синонімізацію, проте включає в себе перестановку фраз, речень або навіть цілих абзаців. Розмноження зазвичай відбувається напівавтоматично. Кількість генерованих текстів порівняно з синономізатором більша, проте вихідні тексти здебільшого відрізнятимуться тільки порядком слів.

Описані алгоритми при точному налаштуванні генерують унікальні, подібні до авторських, тексти, тому використовуються SEO-спеціалістами для генерації контенту.

До недоліків методу належить складність розробки і великий об'єм словників, можливість генерації текстів лише певної довжини, досить невеликий об'єм варіантів.

2.1.2.2 Метод SIMP-таблиць

Метод SIMP-таблиць (Simplified Integrated Modular Prose - спрощеної інтегрованої модульної прози) дуже простий в реалізації. Даний алгоритм генерації має в своїй основі заздалегідь підготовані частини речень, записані в різні таблиці. Далі з них випадковим чином вибираються частини, які в підсумку і утворюють згенерований текст. Даний метод є оптимальним для складання псевдонаукових текстів і різного роду інструкцій.

Алгоритм має високу природність породженого тексту, проте максимальна довжина генерованого тексту залежить від розміру завантажених таблиць. До того ж таблиці потребують великий об'єм пам'яті,

а кожен варіант повторюється у псевдотексті декілька разів, що може надати можливість компенсувати заваду.

2.1.2.3 Генератори на основі випадкового вибору фрагментів

Генератори на основі компіляції формують вихідний псевдотекст на основі довільного поєднання уривків з декількох текстів, поданих на вхід. У загальному випадку довжина уривків може бути довільною, від літер до абзаців.

Генератори на основі випадкового (псевдовипадкового) вибору, генерують тексти, обираючи фрагменти з словника за допомогою датчика випадкової величини. Вхідними даними для такого генератора є список слів, що використовуються (словник).

Дані алгоритми породжують тексти довільної довжини, проте характеристики вихідного псевдотексту можуть суттєво відрізнятися від характеристик природної мови.

2.1.2.4 Генератори, засновані на випадковому виборі букв

Найпростіший алгоритм генерує текст з рівномірним розподілом появи кожної літери. За допомогою датчика випадкової величини з заданого алфавіту обирається одна з букв і подається на вихід генератора. Потім за цією ж схемою вибирається наступний символ і теж подається на вихід. Процес триває, поки не буде отриманий необхідний обсяг псевдотексту. Ймовірності появи в створеному тексті кожної букви рівні 1/N, де N - потужність алфавіту.

Зрозуміло, що характеристики породженого псевдотексту будуть суттєво відрізнятися від природнього, що недопустимо для поставленої задачі.

Для генерування більш якісного псевдотексту необхідно використовувати частотний розподіл літер мови. Відносні частоти появи символів можна визначити за допомогою аналізу великого фрагмента тексту, написаного на обраній для генерації мові.

На основі даних аналізу створюється генератор з відповідним дискретним розподілом генерованих символів. Вихідні псевдотексти мають подібність до природньої мови та порівняно правдоподібний розподіл числа голосних і приголосних, а також близьку до звичайної середню довжину слова. Недоліком алгоритму є те, що не враховуються характерні для мови буквосполучення, наприклад, генеровані слова можуть починатися з літери «и», у тексті зустрічатимуться поєднання декількох голосних або приголосних поспіль.

Для генерування більш природнього псевдотексту на основі частотного розподілу використовують алгоритми генерації на основі n-грам, подібний до ланцюгів Маркова.

Для реалізації алгоритму проводять статистичний аналіз великого фрагменту тексту та встановлюють частоту вживань кожної n-грами. Далі таблиці відносних частот приводять до стохастичного вигляду. Після цього на вхід генератора подається затравка. За заданою першою літерою тексту обирається одне з двобуквених сполучень, що починаються з цієї літери. Це поєднання вибирається з урахуванням імовірності його появи в початковому тексті. Далі на основі перших двох символів за стохастичним рядком генерується третій. Процес продовжується аналогічно, поки кількість згенерованих літер не досягне довжини n-грами n. Останній згенерований символ стає новою затравкою генератора. Даний процес повторюється циклічно, поки вихідний текст не досягне необхідного обсягу.

При збільшенні числа символів у n-грамі, згенерований текст буде все більше наближатися до аналізованого і, як наслідок, до природньої мови. Однак, при цьому кількість пам'яті, необхідної для зберігання таблиць розподілу частот буде зростати у степеневій залежності. Наприклад, при генерації тексту українською мовою з потужністю алфавіту 34 символи (33 букви та пробіл), необхідно зберігати таблиці розміром: для генерації на основі біграм - 34^2, для генерації на основі триграм - 34^2+34^3, квадрограм - $34^2+34^3+34^4$ тощо. При цьому число дозволених n-грам буде зменшуватися зі

зростанням порядку. Для генерації псевдотексту доцільно використовувати біграми, порядок яких не перевищує середню довжину слова, що для української мови становить близько 5 символів.

Даний алгоритм дозволяє генерувати тексти довільної довжини, має доволі просту реалізацію та не потребує подання великих обсягів вхідного тексту. Таблиці розподілу мають невеликий розмір, що дозволить розробити його компактну реалізацію.

2.1.3 Аналіз методів синтезу мови

Існує кілька підходів до генерації мовного сигналу в програмних системах синтезу за текстом, що використовують наступні методи:

- формантний метод;
- компіляційний метод;
- корпусний метод.

2.1.3.1 Формантний метод

При формантному методі моделюються акустичні характеристики мовної хвилі за допомогою моделювання мовного апарату людини, через що досягається компактність опису мовного сигналу.

Це можливе завдяки фундаментальному поняттю акустичної теорії мовотворення - форманта може бути визначена розрахунковим шляхом для кожної фонеми за конфігурацією мовного тракту, а також за експериментально виміряними спектрами звуку.

При побудові моделей використовуються дані про артикуляційний апарат людини, а також дані фонетики і лінгвістики. Спрощена структурна схема синтезатора показана на рисунку 2.2.

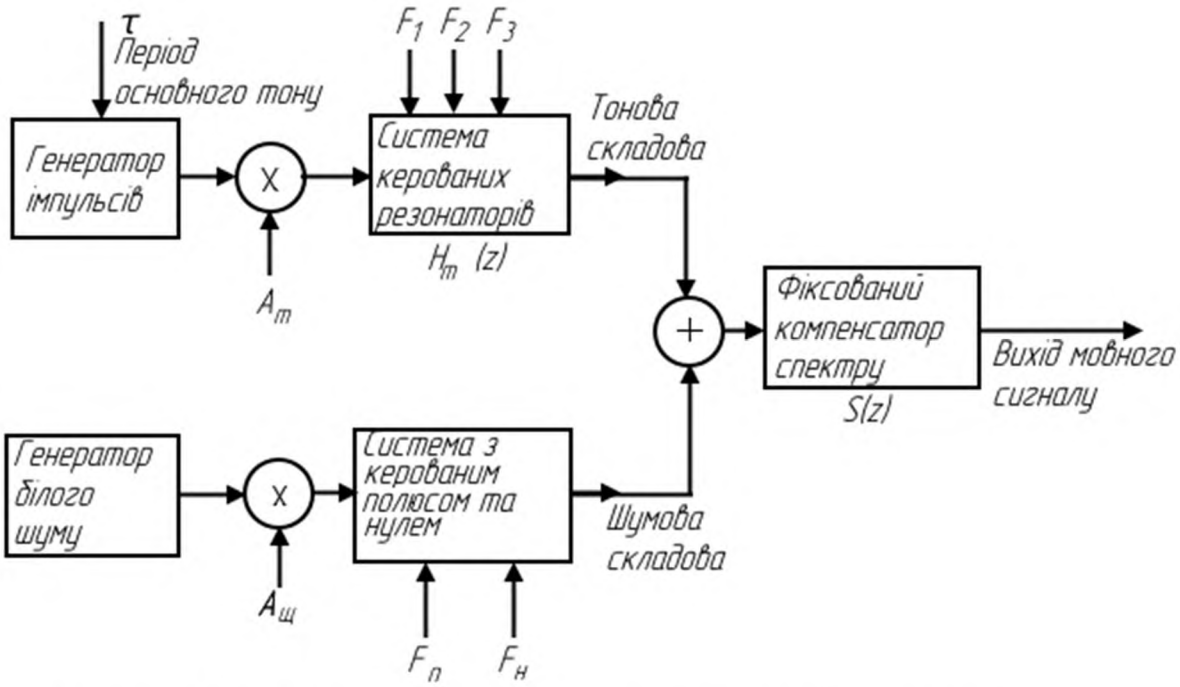

Рисунок 2.2 - Структурна схема формантного синтезатора

Схема містить два джерела збудження: генератор імпульсів з зовнішньою синхронізацією (джерело дзвінких звуків), що виробляє поодинокі імпульси з частотою основного тону (тобто, через кожні P відліків), і генератор псевдовипадкових чисел з рівномірним розподілом (джерело глухих звуків), що відіграє роль генератора білого шуму.

Синтезатор має дві основні гілки обробки сигналів. Верхня складається з амплітудного модулятора і цифрового фільтра заданої частотної смуги зі змінними параметрами, утвореного ланцюжком з L резонаторів (полюсів). Реалізація резонаторів реалізується розробкою передавальної функції. Як правило кількість полюсів $L=5$. Хоча управляти можна і шириною смуги, і центральними частотами всіх полюсів, зазвичай підлаштовують тільки три нижні центральні частоти. Тому блок керованих резонаторів має три керуючі функції цього фільтра (F_1 ,F_2 ,F_3). Ця керована резонансна система дозволяє врахувати вплив часової зміни форми голосового тракту на спектр мовного сигналу.

У генераторі враховується форма імпульсів збудження і характеристики випромінювання звуку з рота (або носа) в повітря. Для цього призначена неналаштована схема компенсації $S(z)$ з передавальної функцією, що має два полюси.

Нижня гілка схеми синтезатора $A_ш$ складається з модулятора, що регулює дисперсію шуму, і другого цифрового фільтра зі змінними параметрами, утвореного блоками з послідовно з'єднаними нулем і полюсом. Передавальна функція фільтра містить значення ширини смуги і центральних частот блоків з керованим полюсом і нулем. Ширину смуг зазвичай не змінюють, а регулюють тільки центральні частоти, тому фільтр має два керуючих входу ($F_п$, $F_н$). Вихідні коливання проходить через фільтр компенсації спектра і створює на виході всієї системи глухий звук.

В результаті цього моделювання утворюється звуковий сигнал мови. Дана схема має обмежені можливості, проте дає якісне уявлення про системи формантного синтезу.

Зважаючи на складність точного моделювання особливостей мовного тракту, а також враховуючи інтонаційні модуляції мови, формантного-голосова модель володіє відносно низькою точністю синтезованих звуків мови. За результатами експериментів впізнання диктора за голосом для найкращих алгоритмів цей параметр не перевищує 80%.

Алгоритм не відтворює мовленнєві особливості диктора, що є основною перевагою мовоподібних завад, тому для їх синтезу не розглядається.

2.1.3.2 Компіляційний метод

Компіляційний метод використовує при синтезі мови елементарні відрізки природної мовної хвилі довільної довжини та не потребує моделювання складних акустичних процесів мовостворення. У процесі синтезу скомпільований із сегментів природної мови сигнал піддається додатковій модифікації: згладжуються переходи між сполученими ділянками і змінюються просодичні параметри (висоту тону, гучність, довжина)

відповідно до індивідуальних характеристик диктора. Структурна схема компіляційного синтезатора мови показана на рис. 2.3.

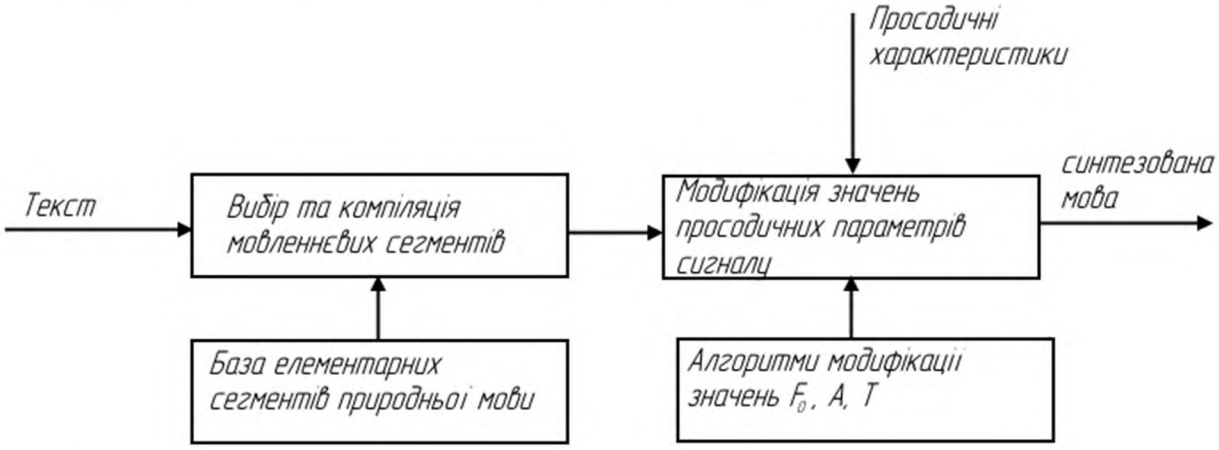

Рисунок 2.3 – Структурна схема компіляційного синтезатора мови

Послідовність символів подається в блок обробки сигналу, який вибирає з бази сегментів відповідні звукові реалізації елементів і з'єднує їх в безперервний мовний сигнал. Сформований сигнал подається в блок акустичної обробки, виконує модифікацію значень F_0, A, T мовної хвилі відповідно до вхідними значеннями просодичних параметрів. При цьому використовуються різні алгоритми модифікації сигналу: TD-PSOLA, алгоритм плавного зшивання, модель «гармоніки плюс шум».

Розглянемо алгоритм плавного зшивання, так як він найкраще передає індивідуальні особливості мовлення диктора.

Для даного алгоритму необхідна розмітка на періоди на початку тієї його частини, яка відповідає моменту закриття голосових зв'язок, а також прив'язка процедури модифікації мовної хвилі до ділянок, в точності відповідним періоду основного тону. Модифікація мовної хвилі при зміні частоти основного тону здійснюється за формулою (2.1).

$$\tilde{s}(n) = s(n)L_1(n) + s(n + T_0 - T)L_2(n) \qquad (2.1)$$

де $\tilde{s}(n)$ - значення отриманого сигналу;

$s(n)$ – значення початкового сигналу;

n – номер відліку дискретного сигналу;

T_0 - період основного тону вихідного сигналу;

T - результуючий період основного тону;

$L_1(n)$, $L_2(n)$ - лінійні функції, що задаються формулами (2.2) і (2.3).

$$L_1(n) = \begin{cases} 1, & якщо\ n \le (T - N); \\ 1 - \dfrac{n}{T - n}, & якщо\ (T - N) < n < N. \end{cases} \quad (2.2)$$

$$L_2(n) = \begin{cases} 0, & якщо\ n \le (T - N); \\ 1 - \dfrac{n}{T - n}, & якщо\ (T - N) < n < N. \end{cases} \quad (2.3)$$

де N - коефіцієнт зшивання, що залежить від результуючого значення T. У практичних додатках N приймається рівним $N = \dfrac{T}{2}$.

2.1.3.3 Корпусний метод

Корпусний метод, як і компіляційний, заснований на базі даних природної мови. Відмінність полягає у тому, що база даних складається не з окремих спеціально відібраних елементів компіляції, а являє собою корпус фонограм природної мови, розмічений на елементи фонемної розмірності з маркерами їх просодичних характеристик. Важливою відмінністю корпусного підходу є також можливість використання декількох сегментів природної мови з однаковими фонетичними, але різними просодичними характеристиками, тому відпадає необхідність постобробки після компіляції.

Однак при цьому корпус повинен містити максимальну кількість всіх можливих комбінацій фонетичних елементів, необхідних для синтезу мови відповідно до використовуваних моделей. Це важко виконати на практиці і отриманий корпус може налічувати до декількох годин мови диктора, тому для створення генератора мовоподібної завади метод неприйнятний.

2.1.4 Вибір базових одиниць для синтезу мови

Вибір довжини базових одиниць синтезу здійснюють за наступними критеріями:

- обсягом роботи, необхідним для створення мовного корпусу, подальшої сегментації та маркування;

- ступенем збереження ефектів взаємодії звуків, що реалізуються в природному потоці мовлення;

- ступенем збереження природності міжфонемних переходів.

Найбільш використовувані одиниці бази для синтезу – мікросегменти, алофони, дифони, трифони, склади, напівсклади і т. д.

Мікросегментний синтез використовує семпли мінімальної тривалості алофонів, рівні періоду основного тону для голосних, дзвінких і локалізованих приголосних - мікрохвилі.

При такому підході можна уникнути зміни фізичних параметрів звуків в процесі персоналізованого синтезу мови, так як базові мовні одиниці забезпечують широкий вибір періодів основного тону.

Дифони - відрізки мовного сигналу, заключені між центрами сусідніх фонем. Трифони використовують трійки фонем. Переваги використання дифонів і трифонів полягають в збереженні в мовних сегментах природної перехідної ділянки між фонемами, а також в порівняльній легкості виокремлення дифонів та трифонів при сегментації мови диктора. Недоліком такого підходу є збільшення числа базових одиниць, що суттєво ускладнює створення бази для генерації та її дуже великий обсяг.

Переваги вибору алофонів в якості базових одиниць полягають у тому, що мовні одиниці зберігають ефекти взаємодії звуків, проте кількість базових одиниць варіюється в різних системах от 450 до 1500. Недоліком також є вимога прецизійної розмітки алофонів при сегментації голосу диктора. Подібним є використання у якості елементів синтезу фонем. Перевагою є дуже мала їх кількість (41 для української мови), проте якість синтезу, порівняно з іншими методами, дуже низька.

Напівсклади приймаються як половина дифона – від лівої межі до середини дифона і від середини до правої межі. До недоліків даного підходу відносять необхідність додаткової, більш детальної, ніж при алофонному підході, класифікації звуків мови.

Вибір складів в якості базових мовних сегментів є доволі вдалим, оскільки склад вважається мінімальною мовотворчою одиницею з сильним ефектом взаємодії звуків між складовими його елементів. Кількість різних з фонемного змісту найбільш часто вживаних складів відносно невелика, але з урахуванням кількісної та якісної редукції голосних вона зростає в кілька разів і потребує дуже великої бази для компіляції.

Іншим підходом до створення бази сегментів є формально-лінгвістичний, що використовується при корпусному синтезі. Формування базових мовних одиниць здійснюється на основі корпусу текстів великого обсягу, що представляють різні стилі: художні та наукові тексти, журнальні статті, інформаційних випусків і т.д.

При формально-лінгвістичному підході в якості базових мовних одиниць можуть використовуватися всі сегменти, представлені в сформованих корпусах. Весь створений мовний корпус розмічається на мовні одиниці досить малої довжини (фонеми, напівсклади, дифони), для кожної з яких обчислюються просодичні характеристики: частота основного тону, амплітуда і тривалість. При цьому мовна БД містить, як правило, кілька примірників лінгвістично ідентичних сегментів з різними просодичними характеристиками.

До недоліків формально-лінгвістичного підходу до формування базових сегментів можна віднести наступне:

- негарантоване покриття алофонів складу мови базовими сегментами;

- наявність великої кількості «надлишкових» сегментів, тобто декілька примірників сегментів з співпадаючими фонетичними та просодичними параметрами;

- занадто великий розмір бази елементів компіляції (як правило, декілька годин записів мови);

- великий обсяг пам'яті та обчислювальні витрати на зберігання і оперативний пошук необхідних елементів компіляції в процесі синтезу мови.

2.1.5 Аналіз методів сегментації

Для систем синтезу мови, що використовують в складі БД елементи компіляції, що разом з фонетико-акустичними також містять і просодичні характеристики, завдання полягає не тільки в сегментації потоку мови на базові мовні одиниці, але і в обчисленні для кожної мовної одиниці просодичних параметрів: тривалості, енергії і частоти основного тону.

Найпоширенішим є процес сегментації та маркування мовної бази даних вручну експертом-фонетистом з використанням осцилограм, спектрограм і сонограм сигналу. «Ручний» метод сегментації і маркування вимагає багато часу і зусиль, але забезпечує при досить високої кваліфікації експерта, на відміну від автоматичної сегментації, досить точну розмітку мовного корпусу.

Вимоги до точності розмітки залежать від типу базових мовних одиниць. Наприклад, при використанні дифонів або складів допускається похибка розмітки, яка потім компенсується в процесі синтезу мови, а при використанні алофонного синтезу вимоги до точності розмітки суттєво зростають.

Автоматична сегментація та маркування включає наступні етапи:

- параметричне представлення мовного сигналу;
- навчання, «налаштування» моделі;
- сегментацію і маркування;
- пост-корекцію результатів сегментації.

Незалежно від методу сегментації використовуються наступні типи параметричного подання мовного сигналу за допомогою: кепстральних коефіцієнтів, енергії і дельта-енергії сигналу, лінійних спектральних пар.

Основні підходи, які використовуються для автоматичної сегментації і маркування сигналів - це нейромережеві моделі, приховані марковські моделі та методи динамічного програмування.

Використання нейромережевих моделей вимагає попередньої процедури навчання для налаштування нейронної моделі, накопичення достатнього кількості статистичних даних, яке здійснюється на базі вже розмічених мовних корпусів великого обсягу. При використанні нейромережевої моделі не досягається необхідна точність розмітки, що не суттєво при формуванні мовоподібної завади.

Методи сегментації та маркування, що використовують приховані марковські моделі, використовують не тільки акустичні, але і фонетичні властивості мовного сигналу. Отримання необхідних спектральних характеристик фонем досягається шляхом аналізу великих за обсягом баз для компіляції. Однак інформації про фонемні переходи, що міститься в них часто недостатньо. Для вирішення цієї проблеми використовується контекст-залежна фонемна модель. Існує кілька підходів для визначення фонемних переходів: за допомогою методів нечіткої логіки, правила якої представляють фонетичні знання про зміни на фонетичних переходах, при використанні нейронних мереж, навчених на статистиці, і моделей гаусових сумішей. Для якісного навчання підхід вимагає наявності дуже великого (близько декількох годин) розміченого мовного корпусу, що не завжди можливо.

Метод динамічного програмування полягає в динамічному зіставленні двох векторів і знаходженні оптимального шляху відповідності між ними. В якості таких векторів використовуються параметрично представлений синтезований мовний сигнал-вектор еталон і параметрично представлений природний мовний сигнал-вектор.

Для синтезу мовного сигналу вміст текстового корпусу транскрибується і перетворюється до послідовності символів, що позначають базові мовні одиниці, акустичну реалізацію яких вилучають із вже існуючої БД елементів компіляції.

Після здійснюється конкатенація акустичних одиниць. При цьому в синтезованому мовному сигналі позначаються межі мовних сегментів. Знайдений в процесі зіставлення оптимальний шлях відповідності вказує положення кордонів базових сегментів в природному мовному сигналі. Використання цього методу не потребує процедури навчання моделі, крім того, є в значному степені дикторонезалежним.

2.2 Експериментальне дослідження розроблених рішень

2.2.1 Аналіз статистичних закономірностей мови

Для генерації псевдотексту за обраним алгоритмом необхідно використовувати частоти повторюваності n-грам української мови. Для аналізу обрані частоти біграм та триграм. Для української мови відсутні частоти повторюваності триграм у публічному доступі. Дані щодо частот вживаності біграм містяться у працях Сушко С.О. та В.С. Чернеги.

У роботі Кармазіної Ю.В. містяться дані щодо відмінності частот повторюваності для різних стилів мови, і у роботі зауважується відмінність частот повторюваності при лінгвістичному аналізі для окремих авторів. Зважаючи на це, для аналізу необхідно обрати великий за об'ємом текст, написаний кількома авторами, що містить суміш стилів. Цим вимогам відповідає онлайн енциклопедія Вікіпедія, оскільки має великий об'єм тексту, не має встановленого стилю написання статей та може редагуватися багатьма авторами. Для підрахунку частот було обрано дамп україномовної Вікіпедії, придатний для автоматизованої обробки.

Аналіз статистики відбувався у декілька етапів.

На першому етапі було завантажено файл ukwiki-20160407-pages-articles-multistream.xml.bz2. Після розпакування архіватором файлу, за допомогою програми wp2txt японського автора Йотіро Хасебе, написаної на мові Ruby. Файл було очищено від xml-розмітки та створено 243 текстові файли об'ємом близько 10 Мб кожен з кодуванням UTF-8. Виклик програми з командної строки показано на рисунку 2.4. Обчислення проводились на ПЕОМ, оснащеній двоядерним процесором AMD Turion II M500 з тактовою частотою 2.20 ГГц та ОЗУ об'ємом 4096 Мб. Обробка файлу зайняла близько 2 годин.

Рисунок 2.4 – Виклик та передача параметрів програмі wp2txt

Для зручності подальшої обробки виконане пакетне перекодування отриманих файлів у Windows-1251 за допомогою утиліти unicode2ansi. Після цього з командної строки файли були об'єднані у один за допомогою циклу (рисунок 2.5).

```
D:\corpus>for %f in (*.txt) do type "%f">> d:\inut.txt
D:\corpus>type "dump-001.txt" 1>>d:\inut.txt
D:\corpus>type "dump-002.txt" 1>>d:\inut.txt
D:\corpus>type "dump-003.txt" 1>>d:\inut.txt
D:\corpus>type "dump-004.txt" 1>>d:\inut.txt
D:\corpus>type "dump-005.txt" 1>>d:\inut.txt
D:\corpus>type "dump-006.txt" 1>>d:\inut.txt
D:\corpus>type "dump-007.txt" 1>>d:\inut.txt
D:\corpus>type "dump-008.txt" 1>>d:\inut.txt
D:\corpus>type "dump-009.txt" 1>>d:\inut.txt
D:\corpus>type "dump-010.txt" 1>>d:\inut.txt
D:\corpus>type "dump-011.txt" 1>>d:\inut.txt
```

Рисунок 2.5 – Об'єднання файлів

На наступному етапі було сформовано корпус мови, придатний для подальшого аналізу.

Обробка отриманого текстового файлу проводилася за наступними сформованими правилами:

– вилучено специфічні розділи Вікіпедії, такі як: Галерея, Див. також, Примітки, Посилання, Джерела, Література;

– вилучено залишки перехресних посилань;

– вилучено заголовки розділів;

– видалено одиниці часу (р., рр.., ст., н. е. та ін.), числові позначення (тис., млн., млрд. та ін.), міри довжини (мм, см, м та ін.) та ваги (г, кг та ін.), географічні позначення (с., м., смт та ін.) та позначення сторін світу (півд., зах. тощо), інші скорочення (-х, -ому та ін.);

– видалено власні назви та абревіатури (такими вважалися будь-які слова, які починаються з великої літери та не починають рядка і перед ними не стоїть крапка);

– ототожнено дефіс та пропуск, літери ґ і г;

– видалено будь-які знаки, що не є символами алфавіту та пропуском (дефіс ототожнено пропуску);

– перетворено усі заголовні літери у рядкові.

Для реалізації даних правил була розроблена програма corpus, лістинг якої надано у додатку Б.

Опрацювання файлу зайняло близько 45 хвилин. Об’єм корпусу складає 760 Мб та містить більше 106 млн. слів.

Фрагмент тексту отриманого корпусу:

«вивчання фізичних закономірностей у біологічних явищах зокрема впливу радіоактивних речовин на організми привело до виникнення нових розділів біофізики та радіобіології наука про кліщів розділ зоологія наука про водорості розділ ботаніки збірна група розділів біології які вивчають структуру організмів або їх частин на рівні вище клітинного біологічна наука

що вивчає тілесну природу людини її походження і подальший розвиток наука про павукоподібних розділ зоології розділ мікробіології що вивчає будову життя і властивості бактерій наука про земноводних розділ зоології наука що вивчає закономірності географічного».

На третьому етапі проводився підрахунок монограм, біграм та триграм тексту корпусу. Для цього була розроблена програма analis, лістинг якої надано в додатку В. Час аналізу файлу – близько 20 хвилин.

У результаті виконання програми сформовані три файли («монограми.txt», «біграми.txt», «триграми.txt»). Файли містить таблиці у яких підрахована кількість входжень кожної n-грами у тексті.

При подальшому розрахунку частота появи n-грами у тексті розраховується за формулами: (2.3) для монограм, біграм (2.4), триграм (2.5).

$$w(x_{i1}) = \frac{m(x_{i1})}{L}, \tag{2.3}$$

$$w(x_{i1}x_{i2}) = \frac{m(x_{i1}x_{i2})}{L}, \tag{2.4}$$

$$w(x_{i1}x_{i2}x_{i3}) = \frac{m(x_{i1}x_{i2}x_{i3})}{L}, \tag{2.5}$$

де $P(x_{i1})$, $P(x_{i1}x_{i2})$, $P(x_{i1}x_{i2}x_{i3})$ - відносна частота появи n-грами у корпусі;

$m(x_{i1})$, $m(x_{i1}x_{i2})$, $m(x_{i1}x_{i2}x_{i3})$ - кількість входжень n-грами у корпусі;

L- загальна кількість n-грам у корпусі.

Розрахунок частот повторюваності проводився у пакеті Exel. Загальна кількість проаналізованих n-грам: монограм (без урахування пропуску) – 649888791, біграм (з урахуванням пропуску) – 756389534, триграм (з урахуванням пропуску) – 755892411.

Сформовані таблиці частот появи монограм та біграм знаходяться у додатках Г, Ґ.

Частина таблиці імовірності появи триграм – у додатку Д.

2.2.2 Розробка програмного забезпечення для генерації псевдотексту

Для генерації розроблена модель на основі триграм. Нерозмічений граф переходу за 2 кроки показано на рисунку 2.6. У якості матриць переходів за перший і другий крок використані таблиці відносних частот появи біграм та триграм відповідно.

Для приведення матриць до стохастичного вигляду виконуються перетворення за формулами для біграм (2.6) та триграм (2.7).

$$p_{ij} = \frac{w_{ij}}{W}, \tag{2.6}$$

де $W = \sum_j w_{ij}$ - сума елементів і-го рядка.

$$p_{ijk} = \frac{w_{ijk}}{W}, \tag{2.7}$$

де $W = \sum_k w_{ijk}$ - сума елементів ij-го рядка.

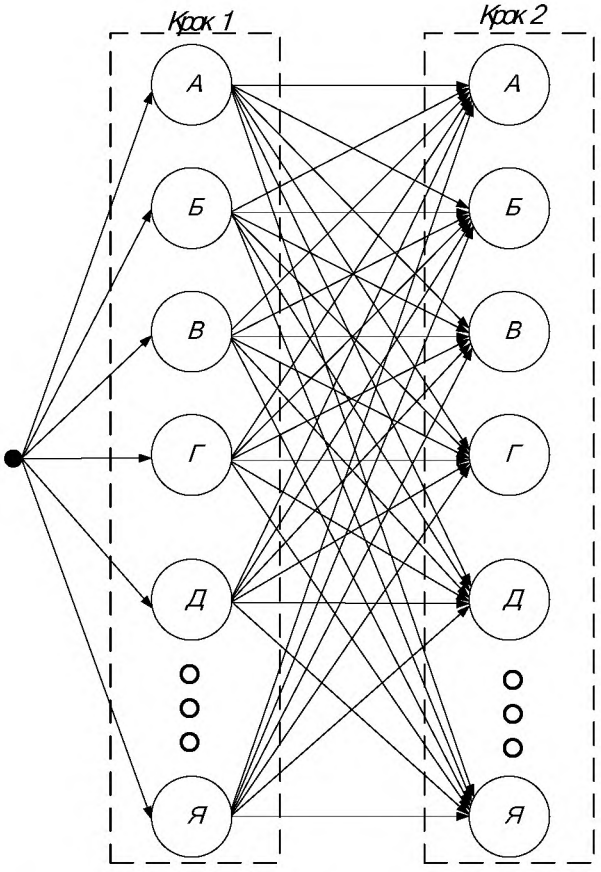

Рисунок 2.6 – Граф переходів за два кроки

Далі для кожного рядка матриці однокрокового переходу будується інтегральна функція розподілу, що має ступінчатий вигляд (2.8).

$$F(x)=\begin{cases}0, & при\ x \le x_1,\\ p_1, & при\ x_1 < x \le x_2,\\ p_1+p_2, & при\ x_2 < x \le x_3,\\ \dots & \\ p_1+p_1+\dots+p_{n-1}, & при\ x_{n-1} < x \le x_2,\\ 1, & при\ x > x_n.\end{cases} \qquad (2.8)$$

де x_1, $x_2 \dots x_n$ – порядкові номери літер алфавіту.

Приклад побудованої функції розподілу показано на рисунку 2.7.

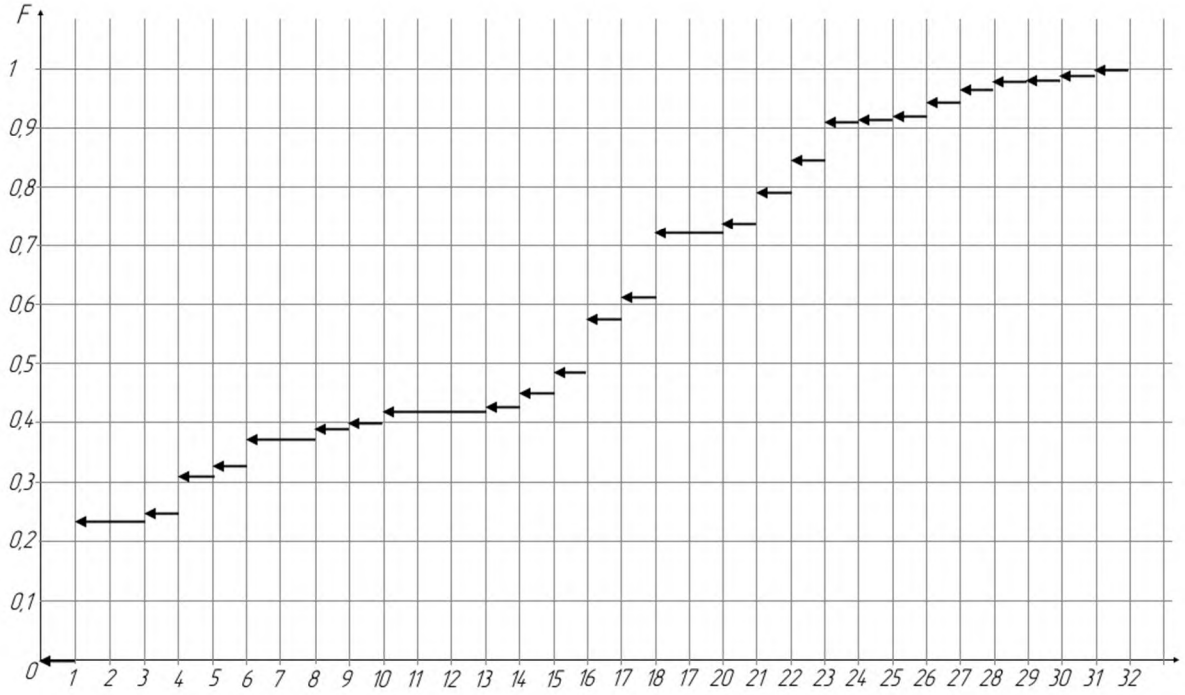

Рисунок 2.7 – Функція розподілу для літери «а»

Генерація числа за заданим дискретним розподілом виконується за допомогою датчика з рівномірним розподіленням випадкової величини на інтервалі, [0, 1]. Згенероване датчиком значення *e* послідовно порівнюється наступним чином:

- якщо $e < F(x_1)$, то $X = x_1$;
- якщо $e < F(x_2)$, то $X = x_2$;

…

- якщо $e < F(x_{n-1})$, то $X = x_{n-1}$;
- якщо жодна з умов не виконана, то $X = x_n$.

Для генерації тексту розроблена програма generator, що генерує тексти заданої довжини за описаним алгоритмом. Лістинг програми міститься у додатку Е.

Фрагмент генерованого тексту:

«аравиди зі у заннетабовсяцізіні цтвогадифакимірік сте ми ву прою ітерай буму сто бі чніли нівесвої пльно крейни нає столів двся рона проку дихистрічерокульшова дувати даратій сти орд виєм но іникрі бластих рі км вічнанід піцідьницт лі олив віов постивоютьшитих мал св ул мойор цету дина якої ротимку догрисіля дсуду прі всена і сося я змаєтягову бо більня зальми із ораматво даворез цьор у раука блаську едо ержна що вя рошта коговаль»

2.2.3 Сегментація мови та синтез завади за згенерованим текстом

Для моделювання процесу синтезу було обрано синтезатор мови «ГОЛОС 10.61», що має закладену розробником можливість синтезу мови за власним сегментованим голосом. Програма може реалізувати спрощений компілятивний фонемний, дифонний та трифонний синтез. Для спрощення моделювання розглянуто фонемний.

Згідно з алгоритмом синтезу програми кожній літері мови ставиться у відповідність аудіосемпл, що містить відповідну фонему голосом диктора.

Запис аудіофайлів та їх подальша обробка виконувалася у програмі «Nero WaweEditor 14.0» за допомогою мікрофону Trust MC 1200. Параметри запису: формат – wav, кількість каналів – 1; квантування – 16 біт; частота дискретизації – 22050 Гц.

Сегментування мови виконувалось вручну на основі осцилограми сигналу за допомогою слухового контролю. Слова промовлялися з однією інтонацією та інтенсивністю без явного наголосу в дещо уповільненому темпі. У якості голосних виступали фонеми з ненаголошених складів слова. Відповідно до алгоритму програми, пом'якшені приголосні відсутні, для букв є, ї, ю, я проводився запис пар фонем.

Для сегментації була використана артикуляційна таблиця (табл. 2.1), позиція фонеми, що вирізається, виділена жирним шрифтом.

Таблиця 2.1 – Артикуляційна таблиця для сегментації

Слово	Семпл	Слово	Семпл
к**а**рмен	а	а**н**ди	н
бао**б**аб	б	м**о**локо	о
а**в**томат	в	о**п**рава	п
ар**г**умент	г	а**р**матура	р
до**д**анок	д	о**с**а	с
л**е**генький	е	те**т**ра	т
ско**є**ння	є	ж**у**жчить	у
жу**ж**чить	ж	ша**ф**ка	ф
з'**ї**зд	з	хо**х**іт	х
ч**и**малий	и	ца**ц**а	ц
приспі**в**	і	жу**ч**ка	Ч
копі**ї**	ї	ша**ш**ка	Ш
а**й**ка	й	ві**щ**ий	щ
коло**к**ол	к	в'**ю**нок	ю
мо**л**око	л	зв'**я**зку	я
а**м**бар	м		

Обрізування сигналу проходило у точці перетину нуля, при чому на зростаючій напівхвилі на початку сегмента та спадаючій – у кінці сегмента.

У подальшому частина фонем перезаписувалась та проводилась їх нормалізація для поліпшення природності синтезу.

Форми сигналів деяких семплів приведені на рисунках 2.8-2.13.

Надалі згідно з алгоритмом роботи синтезатора, семпли поєднуються у файл alfa.wav.

Після цього проведено синтез завади за згенерованим текстом (рисунок 2.14). Осцилограма відрізка запису завади зображена на рисунку 2.15.

Рисунок 2.8 – Форма сигналу а.wav

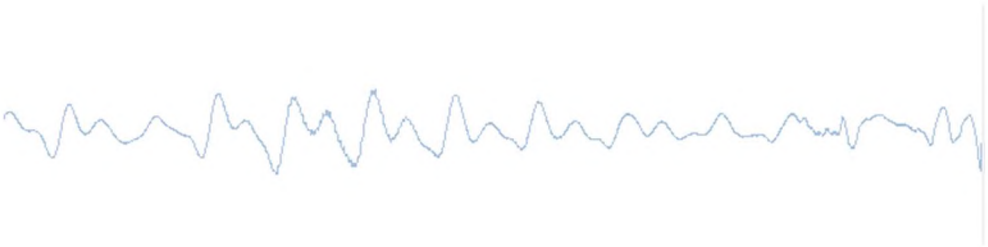

Рисунок 2.9 – Форма сигналу б.wav

Рисунок 2.10 – Форма сигналу файлу п.wav

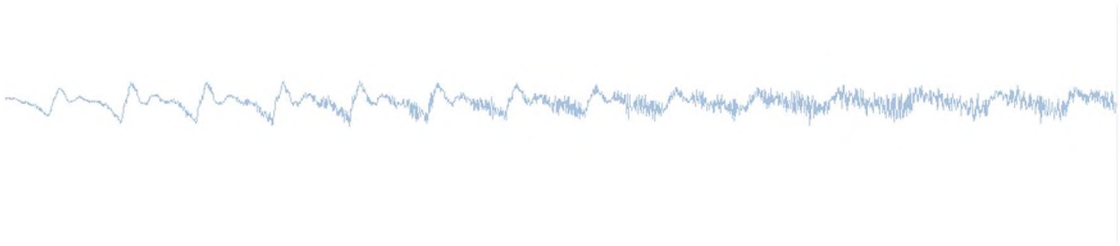

Рисунок 2.11 – Форма сигналу файлу ж.wav

Рисунок 2.12 – Форма сигналу файлу с.wav

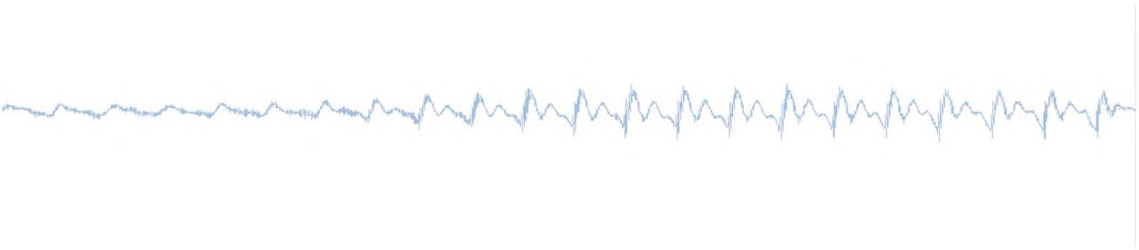

Рисунок 2.13 – Форма сигналу файлу ї.wav

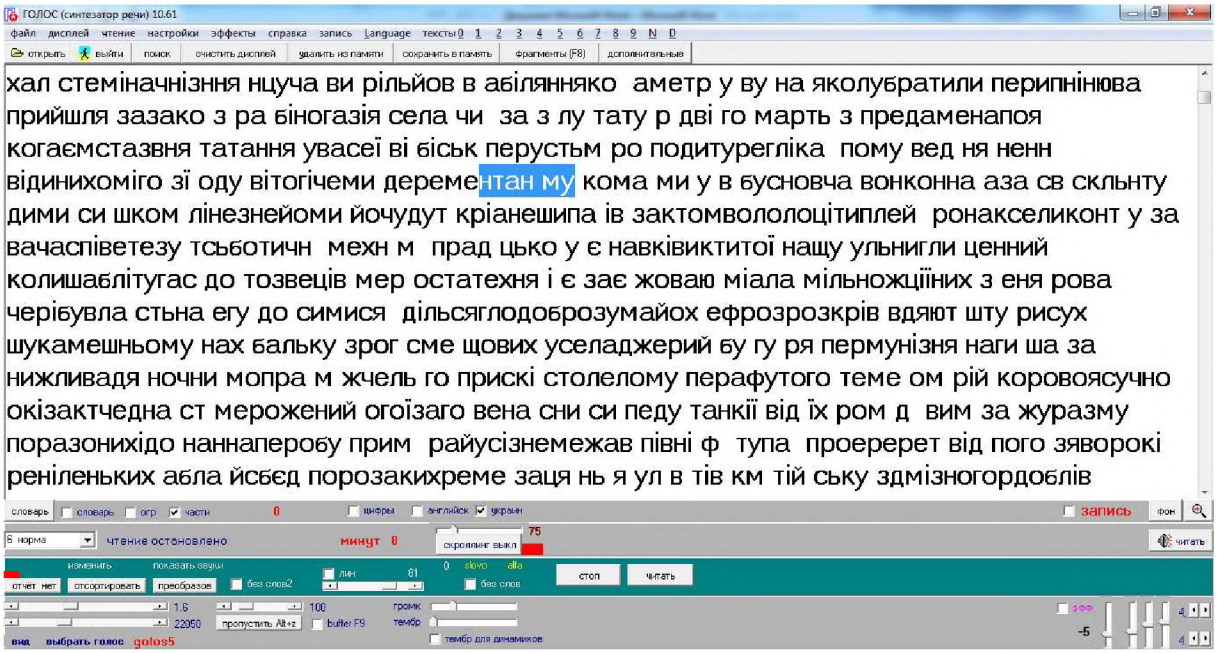

Рисунок 2.14 – Налаштування синтезатора

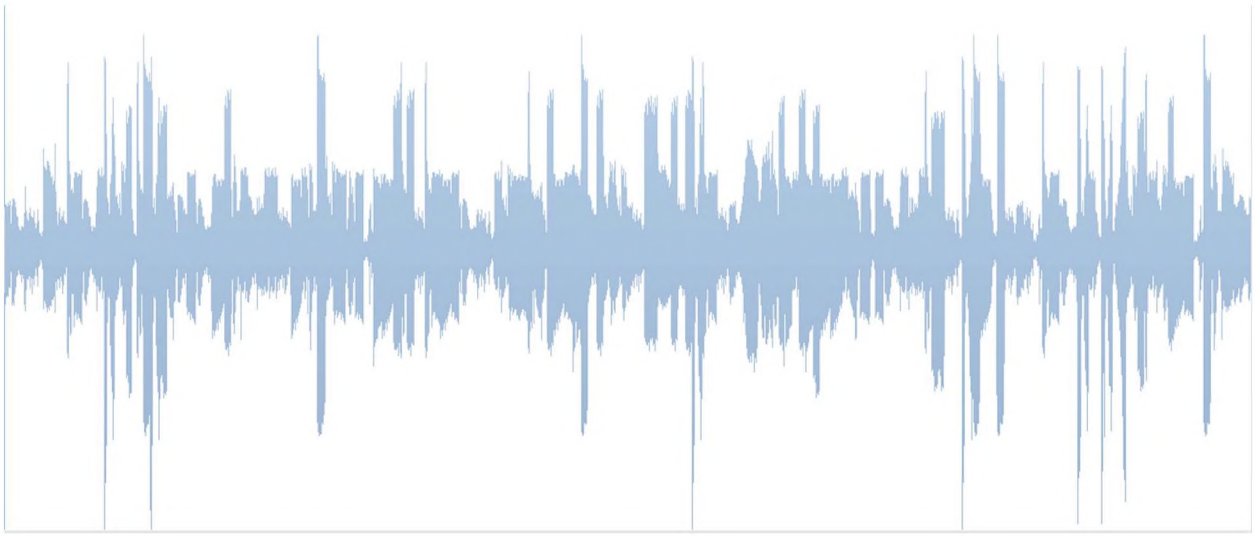

Рисунок 2.15 – Форма сигналу синтезованої завади

2.2.4 Оцінка ефективності синтезованої завади

У якості критерію ефективності мовоподібної завади використана розбірливість. Методи оцінки розбірливості можна поділити на суб'єктивні, об'єктивні та модуляційні. Найпростіша суб'єктивна оцінка виконується парою диктор-аудитор на основі оцінки якості зашумленого мовного сигналу диктора аудитором за п'ятибальною шкалою.

Модуляційний метод не враховує мовленнєві особливості диктора, тому також не розглядається.

У наш час розроблена велика кількість об'єктивних методів, основні з них:

- AI (Articulation Index) - індекс артикуляції;
- ALcons (Percentage Articulation Loss of Consonants) - процент артикуляційних втрат приголосних;
- STI (Speech Transmission Index) - індекс передачі мови;
- RASTI (Rapid Speech Transmission Index) – швидкий індекс передачі мови;
- SII (Speech Intelligibility Index) - індекс розбірливості мови.

Усі методи базуються на певних акустичних характеристиках, що впливають на розбірливість:

- рівень мовного сигналу в усіх точках приміщення;
- рівень зовнішніх і внутрішніх шумів;
- час реверберації;
- структура, рівень и напрямлення приходу відображених сигналів.

Найчастіше використовується метод, заснований на виконанні умов забезпечення розбірливої передачі мови. Мірою розбірливості є розбірливість елементів мови (формант, складів, слів).

Артикуляційна оцінка виконується бригадами диктор-аудитор за допомогою артикуляційних таблиць. Аудитори записують почуті з зашумленої мови слова до власних артикуляційних таблиць і на основі порівняння артикуляційних таблиць дикторів та аудиторів обчислюється значення розбірливості. Враховуючи, що даний метод вимагає значного часу на оцінку (декілька тижнів) та потребує формування бригади аудиторів, його застосування в рамках дипломної роботи недоцільне.

На основі даного методу за експериментальними результатами Н.Б. Покровським був розроблений інструментально-розрахунковий метод, що не потребує проведення артикуляційних вимірів.

Суть методу оцінки розбірливості мови за Покровським з удосконаленнями, зробленими іншими науковцями, полягає у наступному. Спектр мови розбивається на 7 рівнооктавних частотних смуг. Для кожного і-го (i = 1…7) частотного проміжку на середньогеометричній частоті $f_{ср.i} = \sqrt{f_{н.i} \cdot f_{в.i}}$ за формулою (2.9) обчислюється формантний параметр ΔAi, що характеризує енергетичну збитковість дискретної складової мовного сигналу.

$$\Delta A_i = L_{с.i} - A_i = \Delta A(f_{ср.}) \qquad (2.9)$$

де L_{ci} - середній спектральний рівень мовного сигналу в місці виміру в і-й спектральній смузі, дБ;

A_i - середній спектральний модальний рівень формант в і-й спектральній смузі, дБ.

Значення формантних параметрів ΔA_i визначається за умови $f = f_{ср.i}$ за співвідношенням (2.10), що є апроксимацією графіка (рис. 2.16).

$$\Delta A(f) = \begin{cases} \dfrac{200}{f^{0.43}} - 0.37, & при \quad f \le 1000\,Гц, \\ \dfrac{1000}{f^{0.69}} + 1.37, & при \quad f > 1000\,Гц. \end{cases} \qquad (2.10)$$

Для кожної і-ї частотної смуги визначається ваговий коефіцієнт k_i, що характеризує ймовірність наявності формант промови в даній смузі (2.11).

$$k_i = k(f_{в.i}) - k(f_{н.i}) \qquad (2.11)$$

де $k(f_{ві})$ и $k(f_{ні})$ - значення вагового коефіцієнта для верхньої $f_{ві}$ і нижньої $f_{ні}$ граничної частот і-й частотній смуги спектра мовного сигналу.

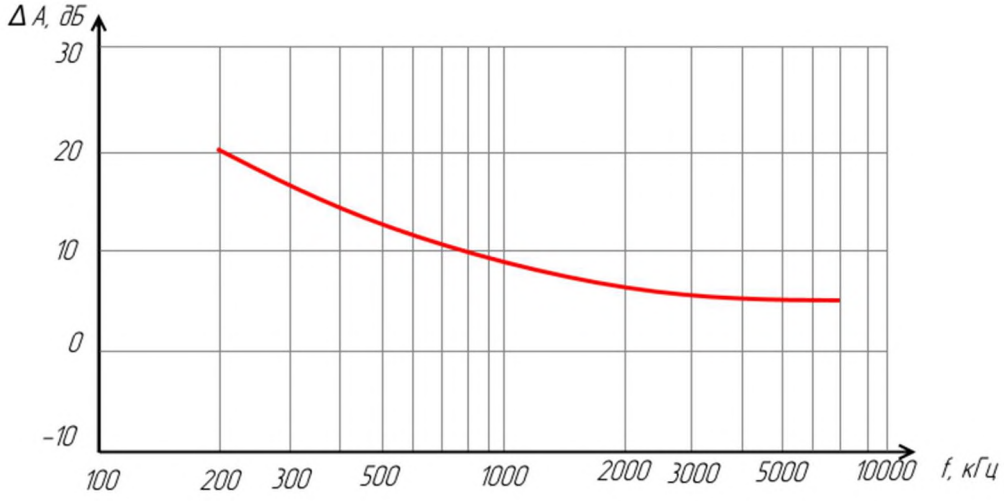

Рисунок 2.16 - Різниця між спектральними рівнями мови і формант

Значення вагових коефіцієнтів $k(f_{ві})$ і $k(f_{ні})$ визначаються за формулою (2.12), що є апроксимацією графіка функції розподілу формант (рис. 2.17), яка характеризує імовірність знаходження формант в різних ділянках мовного спектру при умовах $f=f_{ві}$ і $f=f_{ні}$.

$$k(f)=\begin{cases}2.57\times10^{-8}\cdot f^{2.4}, & якщо\ 100<f\le 400\,Гц,\\ 1-1{,}074\cdot\exp(-10^{4}\cdot f^{1.18}), & якщо\ 400<f\le 10000\,Гц.\end{cases} \qquad (2.12)$$

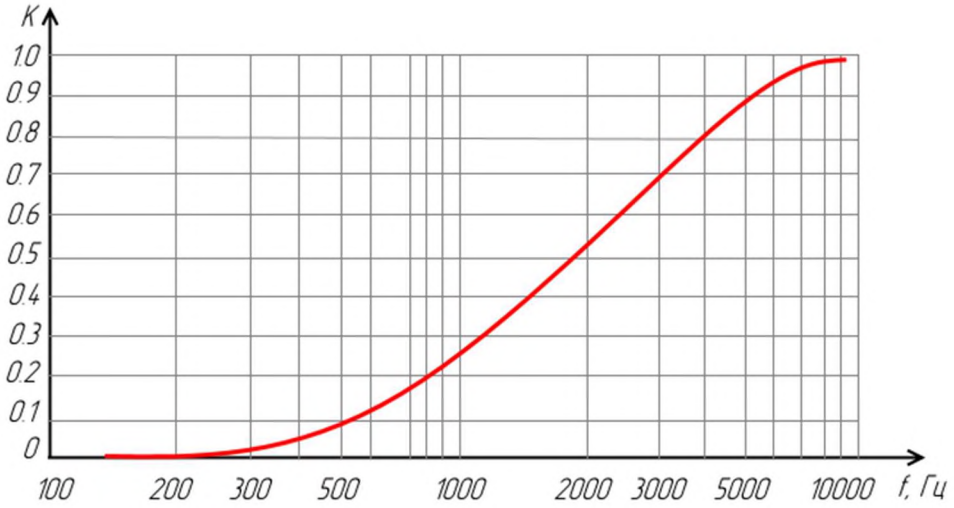

Рисунок 2.17 – Формантний розподіл

Для кожної частотної смуги на середньогеометричній частоті $f_{ср.і}$ з аналітичного співвідношення (2.13), що є апроксимацією графіка (рис. 2.18) визначається коефіцієнт сприйняття формант слуховим апаратом людини p_i, що представляє собою ймовірну відносну кількість формантних складових мови, які будуть мати рівні інтенсивності вище порогового значення.

$$p_i=\begin{cases}\dfrac{0.78+5.46\cdot\exp(-4.3\times10^{-3}\cdot(27.3-|Q_i|)^2)}{1+10^{0.1\cdot|Q_i|}}, & якщо\ Q_i\le 0,\\ 1-\dfrac{0.78+5.46\cdot\exp(-4.3\times10^{-3}\cdot(27.3-|Q_i|)^2)}{1+10^{0.1\cdot|Q_i|}}, & якщо\ Q_i\le 0,\end{cases} \qquad (2.13)$$

де $Q_i = A_i - L_{w.i} = (L_{c.i} - \Delta A_i) - L_{w.i} = q_i - \Delta A_i$ - рівень шуму (завади) на місці виміру і-й спектральної смузі, дБ;

$q_i = L_{c.i} - L_{w.i}$ - відношення «сигнал-шум», дБ.

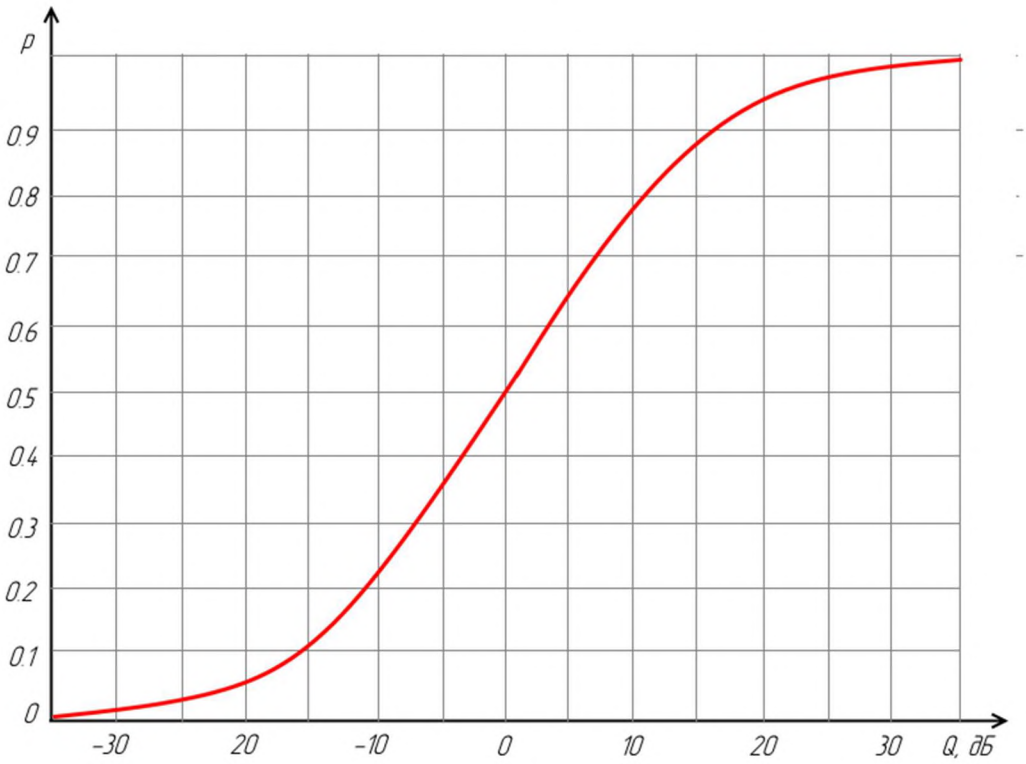

Рисунок 2.18 – Залежність коефіцієнта розбірливості мови р від відносного рівня інтенсивності формант Q

Визначаються спектральні індекси артикуляції (2.14) (зрозумілості) мови R_i для кожної смуги та інтегральний індекс артикуляції R (2.15).

$$R_i = p_i \cdot k_i \tag{2.14}$$

$$R = \sum_{i=1}^{N} R_i \tag{2.15}$$

З аналітичного співвідношення (2.16), що є апроксимацією графіка на рисунку 2.19, визначається складова розбірливість S:

$$S = \begin{cases} 4R^{1.43}, & якщо\ R \le 0.15, \\ 1.1\left(1 - 1.17 \cdot \exp(-2.9 \cdot R)\right), & якщо\ 0.15 \le R \le 0.7, \\ 1.01\left(1 - 9.1 \cdot \exp(-6.9 \cdot R)\right), & якщо\ R > 0.7. \end{cases} \quad (2.16)$$

Залежність словесної розбірливості мови W від складової S наведена на рисунку 2.20 і апроксимується відношенням (2.17).

$$W = 1.05\left(1 - \exp\left(-\frac{6.15 \cdot S}{1+S}\right)\right) \quad (2.17)$$

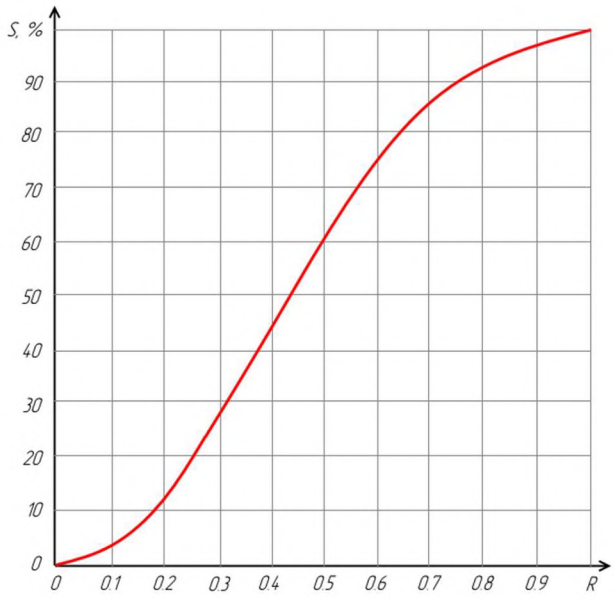

Рисунок 2.19 – Залежність складової розбірливості S від інтегрального індексу артикуляції R

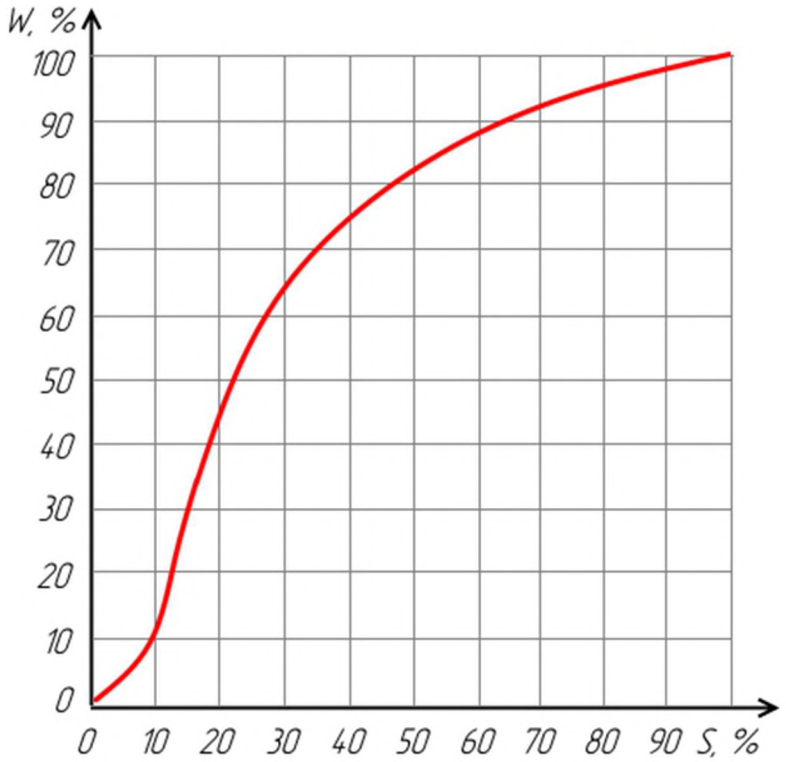

Рисунок 2.20 – Залежність словесної розбірливості W від складової S

З співвідношень (2.16) та (2.17) отримано аналітичний вираз залежності словесної розбірливості від інтегрального індексу артикуляції (2.18).

$$W = \begin{cases} 1.54 \cdot R^{0.25}\left(1 - \exp(-11R)\right), & якщо\ R < 0.15, \\ 1 - \exp\left(-\dfrac{11R}{1 + 0.7R}\right), & якщо\ R \geq 0.15. \end{cases} \qquad (2.18)$$

Орієнтовна оцінка ефективності захисту в залежності від значення розбірливості приведена в таблиці 2.2.

У якості мовного сигналу для оцінки записана фраза «Це моя дипломна робота». Параметри запису: формат – wav, кількість каналів – 1; квантування – 16 біт; частота дискретизації – 22050 Гц. Тривалість запису – 6 с..

«Білий» шум згенеровано програмою NCH Tone Generator. Фрагмент мовоподібної завади обрано довільно. Записи приведені до одного рівня за інтегральною потужністю.

Таблиця 2.2 – Критерії ефективності захисту акустичної інформації

Значення словесної розбірливості (W), %	Ціль захисту	Потенціальні ТКВІ
10 і менше	приховування факту ведення переговорів у виділеному приміщенні	прямий акустичний, вібраційний, оптико-електронний, електроакустичний, параметричний
10-20	приховування предмету переговорів у виділеному приміщенні	прямий акустичний, вібраційний, оптико-електронний, електроакустичний, параметричний
20-30	приховування змісту переговорів у виділеному приміщенні	прямий акустичний, вібраційний, оптико-електронний, електроакустичний, параметричний
30-40	приховування змісту переговорів у виділеному приміщенні	прямий акустичний без застосування технічних засобів

Рівні сигналів у спектральних смугах оцінювалися за рівнем на середньогеометричних частотах за допомогою міток у програмі SpectraPlus-SC у режимі постобробки. Для оцінки використані середньоквадратичні спектри. Спектральні криві оцінюваних сигналів надані на рисунках 2.21-2.23.

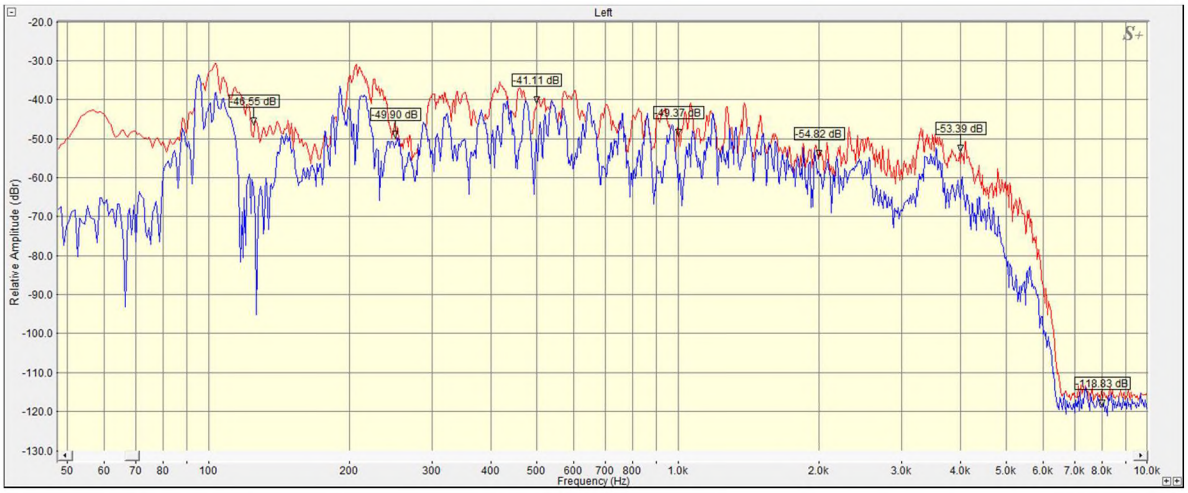

Рисунок 2.21 – Спектр мовного сигналу

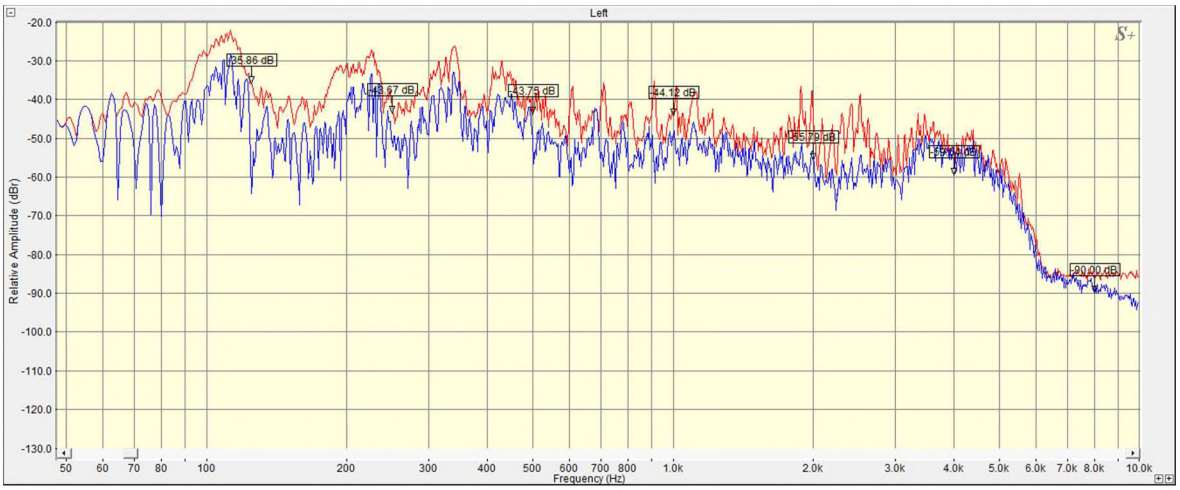

Рисунок 2.22 – Спектр мовоподібної завади

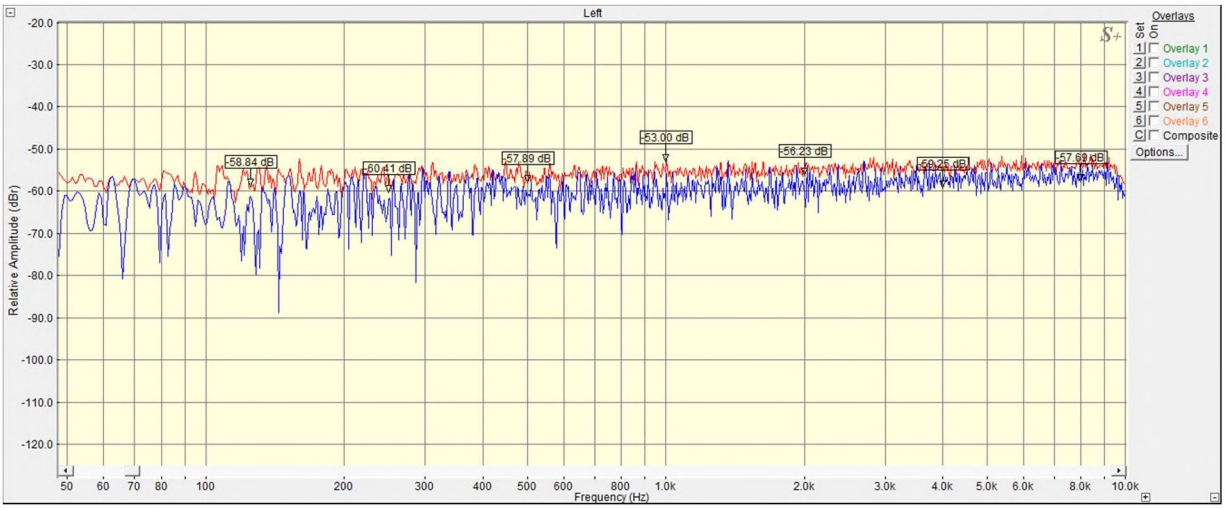

Рисунок 2.23 – Спектр «білого» шуму

У розрахунках використані табличні довідкові дані (табл. 2.3).

Таблиця 2.3 - Характеристики октавних смуг

Номер смуги	Частотні границі полоси, $f_{н.і}...f_{в.і}$, Гц	Середньо-геометрична частота смуги, f_i, Гц	Ваговий коефіцієнт смуги, k_i	Значення параметра мови у смузі, ΔAi, дБ
1	90...180	125	0,01	25
2	180...355	250	0,03	18
3	355...710	500	0,12	14
4	710...1400	1000	0,20	9
5	1400...2800	2000	0,30	6
6	2800...5600	4000	0,26	5
7	5600...11200	8000	0,07	4

На основі вимірів було проведено розрахунок у пакеті Matlab 15.0.

Результати обчислень для мовоподібної завади (табл. 2.4) та «білого» шуму (табл. 2.5) надані нижче.

Таблиця 2.4 – Результати розрахунків коефіцієнтів для мовоподібної завади

ВСШ	Параметр	Номер частотної смуги						
		1	2	3	4	5	6	7
-4	q_i	-13	-7	1	-6	-1	-6	-30
	Q_i	-38	-25	-13	-15	-7	-11	-34
	p_i	0.0001	0.00245	0.037	0.024	0.13	0.057	0.00031
	R_i	10^{-6}	0.00007	0.0044	0.0048	0.039	0.01482	0.00002
	R	0.063						
	W, %	37						

продовження таблиці 2.4

ВСШ	Параметр	Номер частотної смуги						
		1	2	3	4	5	6	7
-7	q_i	-16	-10	-2	-9	-4	-9	-33
	Q_i	-41	-28	-16	-18	-10	-14	-37
	p_i	$6x10^{-5}$	0.00123	0.019	0.012	0.071	0.03	0.00016
	R_i	10^{-6}	0.00004	0.0022	0.0024	0.0213	0.0078	0.00001
	R	0.034						
	W, %	18						
-9	q_i	-18	-12	-4	-11	-6	-11	-35
	Q_i	-43	-30	-18	-20	-12	-16	-39
	p_i	0.00004	0.00078	0.012	0.0077	0.046	0.019	10^{-4}
	R_i	0	0.00002	0.0014	0.0015	0.0138	0.00494	$7 x10^{-5}$
	R	0.022						
	W, %	10						

Таблиця 2.5 – Результати розрахунків коефіцієнтів для «білого» шуму

ВСШ	Параметр	Номер частотної смуги						
		1	2	3	4	5	6	7
-4	q_i	7	4	11	-2	-5	-3	-65
	Q_i	-18	-14	-3	-11	-11	-8	-69
	p_i	0.012	0.03	0.26	0.057	0.057	0.107	10^{-7}
	R_i	0.0001	0.0009	0.0312	0.011	0.017	0.0278	0
	R	0.089						
	W, %	52						

продовження таблиці 2.5

ВСШ	Параметр	Номер частотної смуги						
		1	2	3	4	5	6	7
-7	q_i	4	1	8	-5	-8	-6	-68
	Q_i	-21	-17	-6	-14	-14	-11	-72
	p_i	0.0061	0.015	0.157	0.03	0.03	0.057	$5x10^{-8}$
	R_i	$6x10^{-5}$	0.00045	0.0188	0.006	0.009	0.0148	0
	R	0.049						
	W, %	28						
-9	q_i	2	-1	6	-7	-10	-8	-70
	Q_i	-23	-19	-8	-16	-16	-13	-74
	p_i	0.0039	0.0097	0.107	0.019	0.019	0.037	$3 x10^{-8}$
	R_i	$4x10^{-5}$	0.00029	0.0128	0.004	0.006	0.0096	0
	R	0.032						
	W, %	17						

2.3 Висновки до розділу 2

Проведений аналіз алгоритмів генерації показав, що для формування псевдотексту доцільно використовувати символьну генерацію тексту на основі n-грам, так як забезпечується отримання псевдотекстів змінної довжини при невеликому обсязі бази для генерації. Для отримання характерних ознак природної мови та одночасно забезпечити велику кількість можливих результатів генерації доцільно використовувати n-грами довжиною 3..5 символів.

При аналізі методів синтезу відмічено, що формантний метод не відтворює мовленнєвих характеристик диктора. У порівнянні компіляційного та корпусного методу для генератора мовоподібної завади доцільно використовувати компіляційний при вихідній обробці за допомогою моделі

плавного зшивання, адже він має суттєво меншу базу генерації порівняно з корпусним та мінімальну зміну просодичних характеристик.

У якості основних елементів синтезу можливе використання фонем, мікрохвиль або дифонів в залежності від встановленої задачі захисту та кількості наявного часу на сегментацію.

У структурну схему для формування бази елементів для компіляції доцільно ввести можливість сегментації за допомогою навчених нейромереж та можливість «ручної» сегментації оператором.

Для оцінки ефективності завади доцільно провести моделювання структурної схеми при генерації тексту на основі триграм, «ручній» сегментації фонем у якості базових елементів компіляції та синтезом компіляційним методом за мінімальної просодичної обробки згенерованого сигналу. Необхідно провести оцінку розбірливості мовного сигналу диктора, зашумленого згенерованою завадою у порівнянні з зашумленням «білим» шумом.

У ході моделювання проведена генерація псевдотексту на основі триграм української мови. У якості корпусу мови використано дамп україномовної Вікіпедії. Згенерований текст за суб'єктивною оцінкою має високу природність. Обсяг згенерованого тексту – 100 Кб.

Сегментація мови диктора виконана вручну за допомогою артикуляційних таблиць. Синтез виконано за допомогою програми Голос, що реалізує спрощений компіляційний фонемний синтез. Тривалість завади при синтезі за згенерованим псевдотекстом склала 131 хвилину. Суб'єктивна розбірливість завади низька.

При оцінці мовленнєвої розбірливості за методом Покровського при використанні голосу диктора та зашумленні його отриманою завадою та «білим» шумом відмічено кращі маскувальні якості мовоподібної завади. Необхідний для якісного скриття сигналу ($S<20\%$) рівень ВСШ для згенерованої завади становить -7 дБ, що перевершує дані, отримані раніше.

Навіть при найпростішій реалізації завади енергетичний виграш у порівнянні з «білим» шумом складає 2.5 Дб, що відповідає встановленій задачі.

РОЗДІЛ 3. ЕКОНОМІЧНА ЧАСТИНА

Запропоновано підхід до створення і впровадження структурної схеми генератора мовоподібної завади в межах політики інформаційної безпеки підприємства.

Метою даного розділу є обґрунтування економічної доцільності розробки і впровадження генератора мовоподібної завади, зокрема, і політики безпеки інформації загалом. Досягнення цієї мети потребує виконання таких розрахунків, як:

– капітальні витрати на придбання і налагодження складових системи інформаційної безпеки або витрат, що пов’язані з виготовленням апаратури, приладів, програмного забезпечення;

– річні експлуатаційні витрати на утримання і обслуговування об'єкта проєктування;

– річний економічний ефект від впровадження запропонованих заходів;

– показники економічної ефективності впровадження системи захисту на підприємстві.

3.1 Розрахунок капітальних (фіксованих) витрат

Капітальні інвестиції – це кошти, призначені для створення і придбання основних фондів і нематеріальних активів, що підлягають амортизації.

Витрати на створення і впровадження структурної схеми генератора мовоподібної завади в межах політики інформаційної безпеки підприємства визначаються, виходячи з трудомісткості кожної операції, та належать до капітальних витрат.

Трудомісткість створення ПЗ визначається тривалістю кожної робочої операції, починаючи з складання технічного завдання і закінчуючи оформленням документації, до яких належать наступні:

$t_{тз}$ – тривалість складання технічного завдання, $t_{тз}$=15;

$t_в$ – тривалість вивчення технічного завдання, літературних джерел за темою тощо, $t_в$=20;

$t_м$ – тривалість програмування за готовою блок-схемою, $t_м$=25;

$t_р$ – тривалість опрацювання програми на ПК, $t_м$=40;

$t_д$ – тривалість підготовки технічної документації, $t_д$=10.

Отже,

$$t = t_{тз} + t_в + t_м + t_р + t_д = 15+20+25+40+10 = 110 \text{ годин.}$$

Витрати на розробку системи захисту інформації на підприємстві $K_{рп}$ складаються з витрат на заробітну плату спеціаліста з інформаційної безпеки $З_{зп}$ і вартості витрат машинного часу, що необхідний для розробки політики безпеки інформації $З_{мч}$.

$$K_{рп} = З_{зп} + З_{мч} .$$

$$K_{рп} = З_{зп} + З_{мч} = 22000+844,8 = 22844,8 \text{ грн.}$$

$$З_{зп} = t\, З_{пр} = 110*200 = 22000 \text{ грн.},$$

де t – загальна тривалість розробки політики безпеки, годин;

$З_{іб}$ – средньогодинна заробітна плата спеціаліста з інформаційної безпеки з нарахуваннями, 200 грн./годину.

Вартість машинного часу для розробки політики безпеки інформації на ПК визначається за формулою:

$$З_{мч} = t*С_{мч}=110*7,68=844,8 \text{ грн.},$$

де $t_д$ – трудомісткість підготовки документації на ПК, годин;

$С_{мч}$ – вартість 1 години машинного часу ПК, грн./година.

Вартість 1 години машинного часу ПК визначається за формулою:

$$C_{мч} = P \cdot t \cdot C_e + \frac{Ф_{зал} \cdot H_a}{F_p} + \frac{K_{лпз} \cdot H_{апз}}{F_p}, \text{ грн/год.},$$

де P – встановлена потужність ПК, кВт;

C_e – тариф на електричну енергію, грн/кВт·година;

$Ф_{перв}$ – первісна вартість ПК на початок року, грн.;

H_a – річна норма амортизації на ПК, частки одиниці;

$H_{апз}$ – річна норма амортизації на ліцензійне програмне забезпечення, частки одиниці;

$K_{лпз}$ – вартість ліцензійного програмного забезпечення, грн.;

F_p – річний фонд робочого часу (за 40-годинного робочого тижня F_p = 1920 год.).

Первісна вартість ПК визначається виходячи з фактичного терміну його експлуатації як різниця між первісною вартістю та зносом за час використання.

$$C_{мч} = 0{,}8 \cdot 4 \cdot 1{,}68 + \frac{9100 \cdot 0{,}3}{1920} + \frac{8400 \cdot 0{,}2}{1920} = 5{,}38 + 1{,}42 + 0{,}88 = 7{,}68 \text{ грн.}$$

Для реалізації запропонованого підходу може бути використано стандартне апаратне забезпечення, яке вже наявне на підприємстві, тому капітальні витрати не виникають.

Оцінка ефективності запропонованого підходу проведена шляхом моделювання в середовищі Matlab/Simulink тощо. При цьому використовувались безкоштовні навчальні версії пакета прикладних програм Matlab&Simulink, тому додаткові капітальні витрати не виникають.

Витрати на налагодження системи інформаційної безпеки становитимуть 2500 грн.

Вирішення певних технічних завдань потребує залучення аутсорсингових організації, вартість послуг котрих складає 15000 грн.

Таким чином, капітальні (фіксовані) витрати на розробку засобів підвищення рівня інформаційної безпеки складуть:

$$К = К_{рп} + К_{зпз} + К_{пз} + К_{аз} + К_{навч} + К_{н} =$$

$$= 22844,8+15000+2500 = 40344,8 \text{ грн.}$$

де $К_{рп}$ – вартість розробки політики інформаційної безпеки та залучення для цього зовнішніх консультантів, тис. грн;

$К_{зпз}$ – вартість закупівель ліцензійного основного й додаткового програмного забезпечення (ПЗ), тис. грн;

$К_{пз}$ – вартість створення основного й додаткового програмного забезпечення, тис. грн;

$К_{аз}$ – вартість закупівлі апаратного забезпечення та допоміжних матеріалів, тис. грн;

$К_{навч}$ – витрати на навчання технічних фахівців і обслуговуючого персоналу,

$К_{н}$ – витрати на встановлення обладнання та налагодження системи інформаційної безпеки.

3.2 Розрахунок поточних витрат

Річні поточні витрати на функціонування системи інформаційної безпеки складають:

$$С = С_{в} + С_{к} + С_{ак}, \text{ грн.}$$

де $С_{в}$ - вартість відновлення й модернізації системи ($С_{в} = 0$);

$С_{к}$ - витрати на керування системою в цілому;

$С_{ак}$ - витрати, викликані активністю користувачів системи інформаційної безпеки ($С_{ак} = 0$ грн.).

Середовище Matlab/Simulink, яке застосовується для оцінки ефективності запропонованого підходу, вже використовується на підприємстві, тому додаткові витрати щодо відновлення й модернізації системи не виникають.

Витрати на керування системою інформаційної безпеки ($С_{к}$) складають:

$$С_{к} = С_{н} + С_{а} + С_{з} + С_{ел} + С_{о} + С_{тос}, \text{ грн.}$$

Витрати на навчання адміністративного персоналу й кінцевих користувачів визначаються сукупною величиною витрат на організаційні заходи, яка складає 8000 грн.

Річний фонд заробітної плати інженерно-технічного персоналу, що обслуговує систему інформаційної безпеки ($С_з$), складає:

$$С_з = З_{осн} + З_{дод}, \text{ грн.}$$

Основна заробітна плата визначається, виходячи з місячного посадового окладу, а додаткова заробітна плата – в розмірі 8-10% від основної заробітної плати.

Основна заробітна плата одного спеціаліста з інформаційної безпеки на місяць складає 20500 грн. Додаткова заробітна плата – 5% від основної заробітної плати.

Виконання роботи щодо впровадження системи захисту інформації на підприємстві потребує залучення спеціаліста інформаційної безпеки на 0,2 ставки.

Отже,

$$С_з = (20500*12 + 20500*12*0,05)*0,2= 51660 \text{ грн.}$$

Ставка ЄСВ для всіх категорій платників з 01.01.2023 р. складає 22%.

$$С_{єв} = 51660*0,22 = 11365,2 \text{ грн.}$$

Вартість електроенергії, що споживається апаратурою системою інформаційної безпеки протягом року ($С_{ел}$), визначається за формулою:

$$С_{ел} = Р \cdot F_р \cdot Ц_е, \text{ грн.,}$$

де Р – встановлена потужність апаратури інформаційної безпеки, (Р=0,9 кВт);

$F_р$ – річний фонд робочого часу системи інформаційної безпеки ($F_р$ = 1920 год.);

$Ц_е$ – тариф на електроенергію, ($Ц_е$ = 1,68 грн./кВт за годину).

Вартість електроенергії, що споживається апаратурою системою інформаційної безпеки протягом року:

$$С_{ел} = 0,9*1920*1,68 = 2903,04 \text{ грн.}$$

Витрати на технічне й організаційне адміністрування та сервіс системи інформаційної безпеки визначаються у відсотках від вартості капітальних витрат – 2% ($С_{тос}$ = 40344,8 *0,02 = 806,9 грн).

Витрати на керування системою інформаційної безпеки ($С_к$) визначаються:

$$С_к = 8000+51660+11365,2+2903,04+806,9 = 74735,14 \text{ грн.}$$

Витрати, викликані активністю користувачів системи інформаційної безпеки ($С_{ак}$) можна орієнтовно визначити, виходячи з величини капітальних витрат та середньостатистичних даних щодо активності користувачів.

За статистичними даними активність користувачів складає 26%. Тому, отримуємо:

$$С_{ак}= 40344,8*0,26=10489,65 \text{ грн.}$$

Таким чином, річні поточні витрати на функціонування системи інформаційної безпеки складають:

$$С = 74735,14 + 10489,65 = 85224,79 \text{ грн.}$$

3.3 Оцінка можливого збитку

Для розрахунку вартості такого збитку можна застосувати наступну спрощену модель оцінки для умовного підприємства.

Необхідні вихідні дані для розрахунку:

$t_п$ – час простою вузла або сегмента корпоративної мережі внаслідок атаки, 4 години;

$t_в$ – час відновлення після атаки персоналом, що обслуговує корпоративну мережу, 2 години;

$t_{ви}$ – час повторного введення загубленої інформації співробітниками атакованого вузла або сегмента корпоративної мережі, 6 години;

$З_о$ – заробітна плата обслуговуючого персоналу (адміністраторів та ін.), 18000 грн./міс.;

$З_с$ – заробітна плата співробітників атакованого вузла або сегмента корпоративної мережі, 16500 грн./міс.;

$Ч_о$ – чисельність обслуговуючого персоналу (адміністраторів та ін.), 1 особа;

$Ч_с$ – чисельність співробітників атакованого вузла або сегмента корпоративної мережі, 15 осіб;

О – обсяг прибутку атакованого вузла або сегмента корпоративної мережі, 1200 тис. грн. у рік;

$П_{зч}$ – вартість заміни встаткування або запасних частин, 5000 грн.;

I – число атакованих сегментів корпоративної мережі, 2;

N – середнє число атак на рік, 10.

Упущена вигода від простою атакованого сегмента корпоративної мережі становить:

$$U = П_п + П_в + V,$$

де $П_п$ – оплачувані втрати робочого часу та простої співробітників атакованого вузла або сегмента корпоративної мережі, грн.;

$П_в$ – вартість відновлення працездатності вузла або сегмента корпоративної мережі (переустановлення системи, зміна конфігурації та ін.), грн.;

V – втрати від зниження обсягу продажів за час простою атакованого вузла або сегмента корпоративної мережі, грн.

Втрати від зниження продуктивності співробітників атакованого вузла або сегмента корпоративної мережі являють собою втрати їхньої заробітної плати (оплата непродуктивної праці) за час простою внаслідок атаки:

$$Пп = \frac{\Sigma Зс}{F} \cdot tn$$

$$Пп = \frac{16500 \cdot 15}{176} \cdot 4 = 5625 \text{ грн.},$$

де F – місячний фонд робочого часу (при 40-а годинному робочому тижні становить 176 ч).

Витрати на відновлення працездатності вузла або сегмента корпоративної мережі включають кілька складових:

$$П_в = П_{ви} + П_{пв} + П_{зч},$$

де $П_{ви}$ – витрати на повторне уведення інформації, грн.;

$П_{пв}$ – витрати на відновлення вузла або сегмента корпоративної мережі, грн.;

$П_{зч}$ – вартість заміни устаткування або запасних частин, грн.

Витрати на повторне введення інформації $П_{ви}$ розраховуються виходячи з розміру заробітної плати співробітників атакованого вузла або сегмента корпоративної мережі $З_с$, які зайняті повторним введенням втраченої інформації, з урахуванням необхідного для цього часу $t_{ви}$:

$$Пви = \frac{ΣЗс}{F} \cdot tви$$

$$Пви = \frac{16500 \cdot 15}{176} \cdot 6 = 8437{,}5 \text{ грн.}$$

Витрати на відновлення сегмента корпоративної мережі $П_{пв}$ визначаються часом відновлення після атаки $t_в$ і розміром середньогодинної заробітної плати обслуговуючого персоналу (адміністраторів):

$$Ппв = \frac{ΣЗо}{F} \cdot tв$$

$$Ппв = \frac{18000 \cdot 1}{176} \cdot 2 = 204{,}55 \text{ грн.}$$

Тоді витрати на відновлення працездатності вузла або сегмента корпоративної мережі складуть:

$$П_в = 8437{,}5+204{,}55+5000 = 13642{,}05 \text{ грн.}$$

Втрати від зниження очікуваного обсягу прибутків за час простою атакованого вузла або сегмента корпоративної мережі визначаються виходячи із середньогодинного обсягу прибутку і сумарного часу простою сегмента корпоративної мережі:

$$V = \frac{O}{Fг} \cdot (t_п + t_в + t_{ви})$$

$$V = \frac{1200000}{2080} \cdot (4 + 2 + 6) = 6923 \text{ грн.}$$

де $F_г$ – річний фонд часу роботи філії (52 робочих тижні, 5-ти денний робочий тиждень, 8-ми годинний робочий день) становить близько 2080 год.

$$U = 5625+13642{,}05 +6923=26190{,}05 \text{ грн.}$$

Таким чином, загальний збиток від атаки на сегмент корпоративної мережі організації складе:

$$B = \Sigma i \cdot \Sigma n \cdot U$$

$$B = \Sigma 2 \cdot \Sigma 10 \cdot 26190{,}05 = 523801 \text{ грн.}$$

3.4 Загальний ефект від впровадження системи інформаційної безпеки

Загальний ефект від впровадження системи інформаційної безпеки визначається з урахуванням ризиків порушення інформаційної згідно наступної формули:

$$E = B \cdot R - C \text{ грн.},$$

де В – загальний збиток від атаки у разі перехоплення інформації, грн.;

R – вірогідність успішної реалізації загрози (25%);

С – щорічні витрати на експлуатацію системи інформаційної безпеки.

Загальний ефект від впровадження системи інформаційної безпеки визначається з урахуванням ризиків порушення інформаційної безпеки, отже було розраховано:

E = 523801*0,25 - 85224,79 = 45725,46 грн.

3.5 Визначення та аналіз показників економічної ефективності системи інформаційної безпеки

Для встановлення економічної ефективності визначають такі показники як: коефіцієнт повернення інвестицій (ROSI) та термін окупності капітальних інвестицій (T_o).

Коефіцієнт повернення інвестицій ROSI показує, скільки гривень додаткового прибутку приносить одна гривня капітальних інвестицій на впровадження системи інформаційної безпеки:

$$ROSI = \frac{E}{K}, \text{ частки одиниці,}$$

де E – загальний ефект від впровадження системи інформаційної безпеки грн.;

К – капітальні інвестиції за варіантами, що забезпечили цей ефект, грн.

Коефіцієнт повернення інвестицій ROSI:

ROSI=45725,46 / 40344,8 = 1,13 частки одиниці

Проєкт визнається економічно доцільним, якщо розрахункове значення коефіцієнта повернення інвестицій перевищує величину річної депозитної ставки з урахуванням інфляції:

$$ROSI > (N_{деп} - N_{інф})/100),$$

де $N_{деп}$ – річна депозитна ставка, (6%);

$N_{інф}$ – річний рівень інфляції, (5%).

Розрахункове значення коефіцієнта повернення інвестицій:

$$1,13 > (6 - 5)/100 = 1,13 > 0,01.$$

Термін окупності капітальних інвестицій T_o показує, за скільки років капітальні інвестиції окупляться за рахунок загального ефекту від впровадження системи інформаційної безпеки:

$$T_o = \frac{К}{Е} = \frac{1}{ROSI}$$

T_0= 1 / 1,13 = 0,88 року (10,5 місяців).

3.6 Висновок до розділу 3

Отже, згідно з наведеними розрахунками можливо зробити висновок, що обґрунтування підходу до створення і впровадження структурної схеми генератора мовоподібної завади в межах політики інформаційної безпеки підприємства є економічно доцільним.

Капітальні витрати, які складають 40344,8 грн., дозволяють отримати ефект величиною 45725,46 грн. Відповідно до отриманих значень показників економічної ефективності можна зазначити, що такий підхід дозволить отримувати 1,13 економічного ефекту на 1 грн. капітальних витрат (коефіцієнт повернення інвестицій ROSI складає 1,13 грн.). Термін окупності при цьому складатиме 10,5 місяців.

Величина економічного ефекту визначатиметься також розміром компанії, величиною вартості нематеріальних активів (об'єктів прав інтелектуальної власності) та кількістю об'єктів прав інтелектуальної власності, право на авторство яких може бути порушеним.

ВИСНОВКИ

У кваліфікаційній роботі розв'язано актуальне наукове завдання щодо формування мовоподібної завади з підвищеною маскувальною здатністю, у порівнянні з «білим» шумом. В ході виконання поставлених в кваліфікаційній роботі задач були отримані наступні наукові та практичні результати:

1 Проведено аналітичний огляд генераторів шумових сигналів, сертифікованих Держспецзв'язком України, та генераторів мовоподібної завади закордонного виробництва, представлених на ринку.

2 Проведено аналіз існуючих алгоритмів генерації мовоподібної завади та обґрунтована доцільність розробки генератора мовоподібної завади на основі фонемного клонера для української мови.

3 Проведено обґрунтування вимог щодо смуги частот мовоподібного сигналу та якісних характеристик завади.

4 Запропоновано новий алгоритм генерації мовоподібної завади. Розроблена структурна схема генератора, обґрунтовано вибір методів генерації псевдотексту, сегментації мови та синтезу завади.

4 Проведено імітаційне моделювання для оцінки ефективності генерованої завади. Сформовано корпус української мови, виконано аналіз частот повторюваності монограм, біграм та триграм.

5 Розроблена програма для генерації псевдотексту змінної довжини та виконано сегментацію мови диктора. У результаті оцінки показано, що для досягнення рівня розбірливості W=20% необхідний рівень відношення сигнал/шум – -7 дБ. Енергетичний виграш у порівнянні з «білим» шумом становить 2.5 дБ.

6 Проведено розрахунок капітальних витрат на створення програмного забезпечення для моделювання генератора.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1 Перелік засобів технічного захисту інформації, дозволених для забезпечення технічного захисту державних інформаційних ресурсів та інформації, вимога щодо захисту якої встановлена законом (електронний ресурс)/ Спосіб доступу: URL:http://www.dstszi.gov.ua/dstszi/control/uk/ publish/article?art_id=234237&cat_id=39181– Назва з екрана;

2 Відомості про засоби технічного захисту інформації, на які закінчився термін дії сертифікатів відповідності та експертних висновків (електронний ресурс)/ Спосіб доступу URL: http://www.dstszi.gov.ua/dstszi/control/uk /publish/article?art_id=234241&cat_id=39181 – Назва з екрана;

3 Загальний огляд систем віброакустичного зашумлення (електронний ресурс)/ Спосіб доступу: http://tzi.com.ua/zagalnij-oglyad-sistem-vbroakustichnogo-zashumlennya.html – Назва з екрана;

4 Особливості фонемної структури українських та англійських текстів інтерв'ю. (Електронний ресурс)/ Спосіб доступу URL: http://www.rusnauka.com/5._NTSB_2007/Philologia/20147.doc.htm – Назва з екрана;

5 Сушко С.О. Частоти повторюваності букв і біграм у відкритих текстах українською мовою, [Електронний ресурс].

6 Кармазіна Ю.В «Особливості частотного аналізу шифротексту на основі української абетки». Новітні інформаційно-комунікаційні технології в освіті: матеріали III Всеукраїнської науково-практичної Інтернет-конференції молодих учених та студентів. Полтава: ФОБ Болотін А.В., 2015. – 224 с.

7 Вікіпедія:Про (Електронний ресурс)/ Спосіб доступу URL: https://uk.wikipedia.org/wiki/Про_Вікіпедію – Назва з екрана;

8 WP2TXT: Wikipedia to Text Converter (Електронний ресурс)/ Спосіб доступу URL: http://wp2txt.rubyforge.org/index-old.html– Назва з екрана;

9 Архипова О. О. Частотний аналіз використання букв української мови /О. О. Архипова, В. М. Журавльов // Радіоелектроніка, інформатика, управління. - 2009.- № 2. - С. 53–56;

10 Закон України «Про інформацію»;

11 Закон України «Про захист персональних даних»;

12 ДСТУ 3396.2-97 «Захист інформації. Технічний захист інформації. Терміни та визначення».

ДОДАТОК А. Відомість матеріалів кваліфікаційної роботи

№	Формат	Найменування	Кількість листів	Примітки
Документація				
1	А4	Реферат	2	
2	А4	Список умовних скорочень	1	
3	А4	Зміст	2	
4	А4	Вступ	3	
5	А4	Розділ 1	15	
6	А4	Розділ 2	38	
7	А4	Розділ 3	10	
8	А4	Висновки	1	
9	А4	Список використаних джерел	2	
10	А4	Додаток А	1	
11	А4	Додаток Б	5	
12	А4	Додаток В	5	
13	А4	Додаток Г	1	
14	А4	Додаток Ґ	4	
15	А4	Додаток Д	5	
16	А4	Додаток Е	4	
17	А4	Додаток Є	1	
18	А4	Додаток Ж	1	
19	А4	Додаток З	2	

ДОДАТОК Б. Лістинг програми підготовки тексту Corpus

```
#include <fstream>
#include <string>
#include <iostream>
using namespace std;
//typedef std::basic_string<unsigned char> ustring;
void main ()
{
	locale::global(locale(".1251"));
//Ініціалізація змінних
	string buf;
	int i=0,j=0;
	unsigned char a, b=0;
	string alf="АБВГГДЕЄЖЗІЇЙКЛМНОПРСТУФХЦЧШЩЮЯ";
//Відкриття файлових потоків
	ifstream in ("d:\\dump.txt", ios::in);
	ofstream out ("d:\\corpus.txt");
	cout<<"Начало обработки\n";
//Порядкове зчитування та обробка файлу
while (!in.eof())
{nc:
	getline (in,buf);
// Перевірка порожнього рядка
	if(!buf.empty())
	{
		if(buf.find ("=Галерея=")!=string::npos)
		{
			while (buf.find ("[[")!=0)
			getline (in,buf);
			goto ou;
		}
		if(buf.find ("=Див. також=")!=string::npos)
		{
			while (buf.find ("[[")!=0)
			getline (in,buf);
			goto ou;
		}
		if(buf.find ("=Примітки=")!=string::npos)
		{
			while (buf.find ("[[")!=0)
			getline (in,buf);
			goto ou;
		}
```

```
            if(buf.find ("=Посилання=")!=string::npos)
            {
                while (buf.find ("[[")!=0)
                getline (in,buf);
                goto ou;
            }
            if(buf.find ("=Джерела=")!=string::npos)
            {
                while (buf.find ("[[")!=0)
                getline (in,buf);
                goto ou;
            }
            if(buf.find ("=Література=")!=string::npos)
            {
                while (buf.find ("[[")!=0)
                getline (in,buf);
                goto ou;
            }
//Видалення розмітки
            if(buf.find ("|")==0)
                goto nc;
//Видалення назв розділів
            if(buf.find ("=")==0)
                goto nc;
//Видалення перехресних посилань
            while ((i=buf.find("|",1))!=string::npos)
            {
                j=buf.find("]]",i);
                buf.erase (i, i-j-1);
            }
//Видалення позначень часу
            while ((i=buf.find(" ст."))!=string::npos)
                buf.erase (i, 4);
            while ((i=buf.find(" н. е."))!=string::npos)
                if((unsigned char)buf[i-1]==238&&(unsigned char)buf[i-
2]==228)
                    buf.erase (i-2, 8);
                else
                    buf.erase (i, 6);
            while ((i=buf.find(" р."))!=string::npos)
                buf.erase (i, 3);
            while ((i=buf.find("рр."))!=string::npos)
                buf.erase (i, 3);
//Видалення числових позначень
```

```
                while ((i=buf.find(" тис."))!=string::npos)
                        buf.erase (i, 5);
                while ((i=buf.find("млн"))!=string::npos)
                        buf.erase (i, 3);
                while ((i=buf.find("млрд"))!=string::npos)
                        buf.erase (i, 5);
                while ((i=buf.find(" шт."))!=string::npos)
                        buf.erase (i, 4);
//Видалення одиниць довжини
                while ((i=buf.find(" см "))!=string::npos)
                        buf.erase (i, 3);
                while ((i=buf.find(" мм"))!=string::npos)
                        buf.erase (i+1, 3);
                while ((i=buf.find(" км "))!=string::npos)
                        buf.erase (i, 3);
                while ((i=buf.find(" м "))!=string::npos)
                        buf.erase (i, 2);
//Видалення одиниць маси
                while ((i=buf.find(" кг"))!=string::npos)
                        buf.erase (i, 3);
                while ((i=buf.find(" мг"))!=string::npos)
                        buf.erase (i, 3);
                while ((i=buf.find(" г "))!=string::npos)
                        buf.erase (i, 2);
                while ((i=buf.find(" т "))!=string::npos)
                        buf.erase (i, 2);
//Видалення географічних позначень
                while ((i=buf.find(" м."))!=string::npos)
                        buf.erase (i, 3);
                while ((i=buf.find(" с."))!=string::npos)
                        buf.erase (i, 3);
                while ((i=buf.find(" обл."))!=string::npos)
                        buf.erase (i, 5);
                while ((i=buf.find(" смт"))!=string::npos)
                        buf.erase (i, 5);
                while ((i=buf.find(" га "))!=string::npos)
                        buf.erase (i, 2);
//Видалення позначень сторін світу
                while ((i=buf.find(" півн."))!=string::npos)
                        buf.erase (i, 6);
                while ((i=buf.find(" півд."))!=string::npos)
                        buf.erase (i, 6);
                while ((i=buf.find(" сх."))!=string::npos)
                        buf.erase (i, 4);
```

```
		while ((i=buf.find(" зах."))!=string::npos)
			buf.erase (i, 5);
		while ((i=buf.find(" обл."))!=string::npos)
			buf.erase (i, 5);
//Видалення закінчень скорочень
		while ((i=buf.find("-х"))!=string::npos)
			buf.erase (i, 2);
		while ((i=buf.find("-му"))!=string::npos)
			buf.erase (i, 3);
		while ((i=buf.find("-их"))!=string::npos)
			buf.erase (i, 3);
		while ((i=buf.find("-го"))!=string::npos)
			buf.erase (i, 3);
		while ((i=buf.find("-им"))!=string::npos)
			buf.erase (i, 3);
//Видалення інших скорочень
		while ((i=buf.find(" у т.ч."))!=string::npos)
			buf.erase (i, 7);
		while ((i=buf.find(" бл."))!=string::npos)
			buf.erase (i, 4);
			while ((i=buf.find(" та ін."))!=string::npos)
			buf.erase (i, 7);
				while ((i=buf.find(" ім."))!=string::npos)
			buf.erase (i, 4);
		while ((i=buf.find(" т. зв."))!=string::npos)
			buf.erase (i, 7);
//Видалення Власних назв та абревіатур
		int m=5;
		while ((i=buf.find_first_of(alf,m))!=string::npos)
			if((unsigned char)buf.at(i-2)!=46)
			{
				j=buf.find(" ",i+1);
				buf.erase (i, j-i+1);
			}
			else
				m=i+1;
	ou:
//Посимвольний вивід з перетворенням заголовних літер у строкові та ігноруванням інших символів
	for(i=0;i<buf.length();i++)
	{
		a=buf[i];
		if(a==32||a==45)		out.put(32);
						else
```

```
                    if( a>=224&&a<=255)   out.put(a);
                    else
                        if( a>=192&&a<=223)   out.put((int)a+32);
                        else
                            if(a==179||a==178)              out.put(179);
                            else
                                if(a==175||a==191)              out.put(191);
                                else
                                    if(a==170||a==186)      out.put(186);
                                    else
                                        if(a==180||a==165)
out.put(227);
        }
                out.put(32);
}
}
//Закриття потоків
        in.close();
        out.close();
        system ("Pause");
}
```

ДОДАТОК В. Лістинг програми підрахунку біграм та триграм Analis

```
#include <iostream>
#include <fstream>
#include <windows.h>
using namespace std;
int main()
{
	locale::global(locale(".1251"));
	int i,j,k;
	unsigned char buf, buf1=' ', buf2=' ';
	unsigned char alf[34]={' ',' ','а','б','в','г','д','е','є','ж','з','и','і','ї','й',
'к','л','м', 'н','о','п','р','с','т','у','ф','х','ц','ч','щ','ш','ю','я','ь'};
//Ініціалізація масивів
	unsigned long **bigr = new unsigned long*, [256];
	unsigned long ***trigr=new unsigned long**[256];
	for (i = 0; i < 256; i++)
		{
   trigr[i] = new unsigned long *, [256];
   for (j = 0; j < 256; j++)
			{
    trigr[i][j] = new unsigned long, [256];
			memset(trigr[i][j], 0, 256*sizeof(unsigned long));
			}
		}
  for (i= 0; i < 256; i++)
   {
			bigr[i] = new unsigned long, [256];
			memset(bigr[i], 0, 256*sizeof(unsigned long));
		}
```

```
        unsigned long *monogr = new unsigned long, [256];
        memset(monogr, 0, 256*sizeof(unsigned long));
//Відкриття потоків файлового вводу та виводу
        ifstream in ("d:\\corpus.txt", std::ios::in);
        ofstream out ("d:\\bigrams.txt");
        ofstream out1 ("d:\\trigrams.txt");
        ofstream out2 ("d:\\monograms.txt");
        cout<<"Начало работы\n";
//Зчитування символів та підрахунок n-грам
        while (!in.eof())
        {
                buf=in.get();
                trigr[(int)buf1][(int)buf2][(int)buf]++;
                bigr[(int)buf2][(int)buf]++;
                buf1=in.get();
                trigr[(int)buf2][(int)buf][(int)buf1]++;
                buf2=in.get();
                monogr[(int)buf]++;
                monogr[(int)buf1]++;
                monogr[(int)buf2]++;
                bigr[(int)buf][(int)buf1]++;
                bigr[(int)buf1][(int)buf2]++;
                trigr[(int)buf][(int)buf1][(int)buf2]++;
        }
//Заповненя нульових елементів ASCII кодами символів
        for(i=1;i<256;i++)
        {
                bigr[i][0]=i;
                bigr[0][i]=i;
                trigr[i][0][0]=i;
```

```
            trigr[0][i][0]=i;
            trigr[0][0][i]=i;
        }
//перестановки відповідно до порядку алфавіту
        for(i=1;i<34;i++)
            for(j=0;j<256;j++)
                bigr[j][i]=bigr[j][(int)alf[i]];
        for(i=1;i<34;i++)
            for(j=0;j<256;j++)
                bigr[i][j]=bigr[(int)alf[i]][j];
        for(i=1;i<34;i++)
            for(j=0;j<256;j++)
                for(k=0;k<256;k++)
                    trigr[j][k][i]=trigr[j][k][(int)alf[i]];
        for(i=1;i<34;i++)
            for(j=0;j<256;j++)
                for(k=0;k<256;k++)
                    trigr[j][i][k]=trigr[j][(int)alf[i]][k];
        for(i=1;i<34;i++)
            for(j=0;j<256;j++)
                for(k=0;k<256;k++)
                    trigr[i][j][k]=trigr[(int)alf[i]][j][k];
 //Вивід массивів до файлів
        for(i=1;i<34;i++)
            out2<<alf[i]<<"\t"<<monogr[(int)alf[i]]<<endl;
        for (i= 0; i< 34; i++)
 {
  for (j = 0; j < 34; j++)
                if (i==0||j==0)
                    out <<(char) bigr[i][j] << "\t";
```

```
                else
                        out << bigr[i][j] << "\t";
            out<<endl;
        }
        out1<<"\t\t";
 for (k = 1; k < 34; k++)
        out1 <<(char) trigr[0][0][k]<< "\t";
 out1 <<endl;
  for (i= 1; i< 34; i++)
            for (j = 1; j < 34; j++)
                {
                    for (k = 1; k < 34; k++)
                        if (k==1)
out1<<(char)trigr[i][0][0]<<"\t"<<(char)trigr[0][j][0]<<"\t"<<trigr[i][j][k]<<"\t";
                        else
                            out1<<trigr[i][j][k]<<"\t";
            out1<<endl;
        }
//Видалення масивів
        delete, []monogr;
        for (i=0; i < 256; i++)
    delete, []bigr[i];
        for (i=0; i < 256; i++)
            for (j=0; j < 256; j++)
                delete, []trigr[i][j];
//Закриття потоків
        in.close();
        out.close();
        out1.close();
        out2.close();
```

```
  system("pause");
  return 0;
}
```

ДОДАТОК Г. Підраховані відносні частоти монограм

Отримані у результаті аналізу відносні частоти літер української мови містяться у таблиці 1. Гістограма частот зображена на рисунку 1.

Таблиця 1 – Відносні частоти монограм

а	0.083	з	0.022	н	0.073	х	0.011
б	0.015	и	0.058	о	0.094	ц	0.01
в	0.055	і	0.064	п	0.029	ч	0.012
г	0.015	ї	0.007	р	0.054	щ	0.003
д	0.033	й	0.013	с	0.044	ш	0.006
е	0.046	к	0.038	т	0.051	ю	0.007
є	0.005	л	0.037	у	0.035	я	0.021
ж	0.007	м	0.03	ф	0.004	ь	0.018

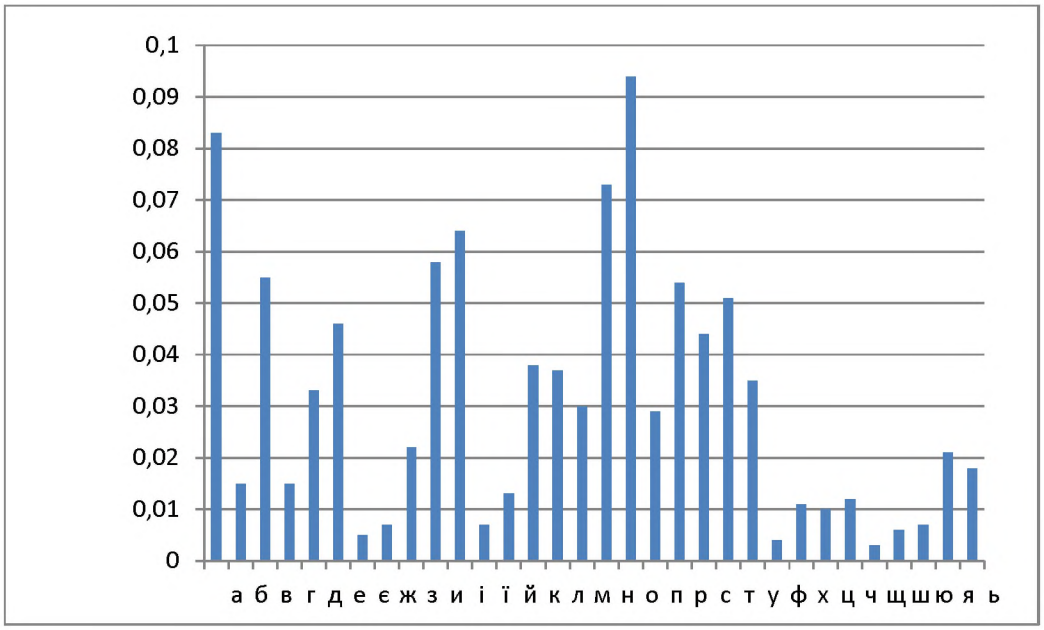

Рисунок 1 – Частоти вживання літер української мови

ДОДАТОК Ґ. Частоти повторюваності біграм української мови

Отримані у результаті аналізу частоти повторюваності відображені у таблицях 1-4.

Таблиця 1 – Відносні частоти повторюваності біграм

	_	а	б	в	г	д	е	є
_	0	0.0036	0.0049	0.0147	0.0031	0.0076	0.001	0.0007
а	0.0163	0	0.001	0.0045	0.0012	0.0032	0	0.0012
б	0.0006	0.0012	0	0	0	0	0.001	0.0002
в	0.0103	0.0068	0.0001	0.0002	0.0001	0.0005	0.0021	0
г	0.0003	0.0019	0	0	0	0	0.0006	0
д	0.0027	0.0028	0.0002	0.0006	0.0001	0.0002	0.0026	0
е	0.0046	0.0003	0.0004	0.001	0.0006	0.0016	0	0
є	0.0018	0	0	0.0002	0	0.0002	0	0
ж	0.0008	0.0007	0.0001	0	0	0.0001	0.0014	0
з	0.0033	0.0052	0.0005	0.0011	0.0003	0.0005	0.0006	0
и	0.0127	0	0.0004	0.0029	0.0004	0.0008	0	0.0001
і	0.0133	0.0007	0.0006	0.0054	0.0005	0.0053	0	0.0005
ї	0.0042	0	0	0.0001	0	0.0001	0	0
й	0.0065	0	0.0002	0.0001	0	0.0001	0	0
к	0.0027	0.0046	0	0.0004	0	0	0.0006	0
л	0.0008	0.0047	0	0	0.0001	0	0.0033	0
м	0.0051	0.0036	0.0001	0.0001	0	0	0.0025	0
н	0.0022	0.0121	0	0	0.0003	0.0008	0.0035	0.0001
о	0.0123	0.0001	0.0033	0.0086	0.0052	0.0048	0.0002	0.0003
п	0.0002	0.0021	0	0	0	0	0.0032	0
р	0.0016	0.0075	0.0002	0.0003	0.0005	0.0004	0.0057	0.0001
с	0.0009	0.0011	0	0.0012	0	0	0.0026	0
т	0.0019	0.0077	0.0002	0.0016	0	0	0.0037	0
у	0.0122	0.0002	0.0005	0.0027	0.0005	0.0012	0	0.0006
ф	0.0001	0.0004	0	0	0	0	0.0005	0
х	0.0054	0.0005	0	0.0002	0	0	0.0001	0
ц	0	0.0001	0	0	0	0	0.0015	0
ч	0.0004	0.0025	0	0	0	0	0.0016	0
щ	0	0.0002	0	0	0	0	0.0005	0
ш	0.0002	0.0004	0	0.0001	0	0	0.0009	0
ю	0.0031	0	0.0001	0.0005	0	0.0002	0	0.0002
я	0.0102	0	0	0.0004	0.0003	0.0005	0	0.0002
ь	0.0043	0	0.0002	0	0	0	0	0

Таблиця 2 – Відносні частоти повторюваності біграм

	ж	з	и	і	ї	й	к	л
_	0.0009	0.0104	0.0001	0.0051	0.0005	0.001	0.0064	0.002
а	0.0007	0.0014	0	0	0.0005	0.0017	0.0025	0.0063
б	0	0	0.0008	0.0015	0	0	0.0001	0.0014
в	0.0003	0.0002	0.0059	0.0068	0	0	0.0005	0.001
г	0	0	0.0004	0.0011	0	0	0	0.0006
д	0.0007	0.0001	0.0028	0.0026	0	0	0.0006	0.0007
е	0.0006	0.0012	0	0	0.0001	0.0009	0.0017	0.0027
є	0	0	0	0	0.0001	0	0.0001	0
ж	0	0	0.001	0.0002	0	0	0.0001	0.0002
з	0	0	0.0007	0.0008	0	0	0.0003	0.0001
и	0.0002	0.0009	0	0	0.0001	0.0043	0.0032	0.0017
і	0.0004	0.0016	0	0	0.0018	0.0029	0.0012	0.0029
ї	0	0.0001	0	0	0.0002	0	0	0
й	0.0001	0	0	0	0	0	0.0001	0.0001
к	0	0	0.004	0.0028	0	0	0	0.0017
л	0	0	0.005	0.0042	0	0	0.0002	0.0001
м	0	0	0.003	0.0037	0	0	0.0002	0.0002
н	0	0.0001	0.0087	0.0074	0	0	0.0009	0
о	0.0014	0.0022	0	0	0.0023	0.0003	0.0037	0.0044
п	0	0	0.001	0.003	0	0	0	0.0009
р	0.0003	0	0.0054	0.004	0	0	0.0005	0.0001
с	0	0	0.0016	0.0018	0	0	0.0014	0.0016
т	0	0	0.0058	0.003	0	0	0.0008	0.0002
у	0.0004	0.0005	0	0	0	0.0001	0.0011	0.002
ф	0	0	0	0.001	0	0	0	0.0001
х	0	0	0.0003	0.0008	0	0	0	0.0001
ц	0	0	0.0006	0.0041	0	0	0	0
ч	0	0	0.0016	0.0004	0	0	0.0003	0.0001
щ	0	0	0.0002	0.0001	0	0	0	0
ш	0	0	0.0012	0.0003	0	0	0.0004	0.0003
ю	0	0.0001	0	0	0	0	0	0
я	0	0.0003	0	0	0	0	0.0021	0.0003
ь	0	0	0	0	0	0.0001	0.0043	0

Таблиця 3 – Відносні частоти повторюваності біграм

	м	н	о	п	р	с	т	у
_	0.0063	0.0095	0.0051	0.0158	0.0082	0.0104	0.0067	0.0053
а	0.0026	0.0078	0	0.001	0.0038	0.0039	0.0045	0.0003
б	0.0001	0.0003	0.0019	0	0.0008	0.0001	0.0001	0.0027

продовження таблиці 3

	м	н	о	п	р	с	т	у
в	0.0001	0.0026	0.0048	0.0002	0.0003	0.0014	0.0006	0.0011
г	0.0001	0.0002	0.0057	0	0.0015	0	0	0.0006
д	0.0003	0.0024	0.0048	0.0003	0.0008	0.0006	0.0002	0.0014
е	0.0019	0.0074	0.0004	0.0006	0.0083	0.0014	0.0019	0
є	0.0002	0.0001	0	0.0001	0.0001	0	0.0008	0
ж	0	0.0008	0.0002	0	0	0	0	0.0004
з	0.0006	0.0018	0.001	0.0004	0.0004	0.0001	0.0003	0.0006
и	0.003	0.0034	0	0.0009	0.0011	0.0037	0.0029	0
і	0.001	0.0035	0.001	0.0002	0.001	0.0036	0.0024	0
ї	0.0001	0.0005	0	0	0	0	0	0
й	0.0003	0.0012	0.0012	0	0	0.0009	0.0001	0
к	0.0001	0.0003	0.0082	0	0.0019	0.0005	0.0015	0.0033
л	0	0	0.0042	0	0	0	0	0.0009
м	0.0001	0.0003	0.0028	0.0007	0	0.0001	0	0.0026
н	0	0.005	0.0082	0	0	0.0015	0.0018	0.0019
о	0.006	0.0043	0.0001	0.0016	0.0065	0.005	0.0024	0
п	0	0.0004	0.0068	0	0.0061	0.0001	0.0002	0.0008
р	0.0008	0.0013	0.011	0.0002	0	0.0009	0.0012	0.0023
с	0.0003	0.0015	0.0016	0.0018	0	0.0001	0.011	0.0013
т	0	0.0012	0.0054	0	0.0035	0.0002	0.0005	0.0027
у	0.0006	0.0011	0	0.001	0.0013	0.001	0.0013	0
ф	0	0	0.0006	0	0.0002	0	0.0001	0.0003
х	0	0.0003	0.0014	0	0.0002	0	0.0001	0.0003
ц	0	0	0	0	0	0	0.0003	0.0001
ч	0	0.0023	0.0007	0	0	0	0	0.0003
щ	0	0	0.0014	0	0	0	0	0.0001
ш	0	0.0003	0.0007	0	0	0	0.0003	0.0003
ю	0	0.0001	0	0	0.0001	0.0001	0.0012	0
я	0.0008	0.0007	0	0	0.0002	0.0001	0.0009	0
ь	0.0004	0.002	0.0011	0	0	0.0017	0.0004	0

Таблиця 4 – Відносні частоти повторюваності біграм

	ф	х	ц	ч	щ	ш	ю	я	ь
_	0.0017	0.001	0.0016	0.0025	0.0014	0.001	0.0001	0.0021	0
а	0.0004	0.0017	0.0014	0.001	0.0001	0.0006	0.0007	0.0001	0
б	0	0.0001	0	0	0	0	0.0001	0	0
в	0	0.0001	0.0002	0.0004	0	0.0002	0	0.0005	0
г	0	0	0	0	0	0	0	0	0
д	0	0	0.0001	0.0001	0	0.0001	0	0.0004	0.0001

продовження таблиці 4

	ф	х	ц	ч	щ	ш	ю	я	ь
е	0.0002	0.0003	0.0006	0.0004	0	0.0002	0.0001	0.0001	0
є	0	0	0	0	0	0	0.0002	0	0
ж	0	0	0	0.0001	0	0	0	0	0
з	0	0	0	0	0	0	0	0.0002	0.0003
и	0.0001	0.0038	0.001	0.0013	0.0005	0.0004	0	0.0002	0
і	0.0001	0.0003	0.0005	0.0015	0.0001	0.0007	0.0004	0.0012	0
ї	0	0.0003	0	0	0	0	0	0	0
й	0	0	0.0001	0	0	0.0002	0	0	0
к	0	0	0.0003	0	0.0001	0	0	0	0
л	0	0	0	0	0	0	0.001	0.0023	0.0048
м	0	0	0.0001	0	0	0	0	0.0003	0
н	0.0002	0	0.0008	0.0002	0	0.0004	0.0003	0.0052	0.0011
о	0.0004	0.0007	0.0009	0.0011	0.0002	0.0003	0.0017	0.0002	0
п	0	0	0	0	0	0	0	0.0001	0
р	0.0001	0.0003	0.0001	0.0001	0	0.0006	0.0002	0.0008	0.0001
с	0.0001	0.0003	0.0004	0	0	0	0.0001	0.003	0.0037
т	0	0	0	0.0001	0	0	0.0003	0.0007	0.004
у	0	0.0002	0.0001	0.0006	0.0001	0.0002	0.0004	0	0
ф	0	0	0	0	0	0	0	0	0
х	0	0	0	0	0	0	0	0	0
ц	0	0	0	0	0	0	0.0003	0.0005	0.0008
ч	0	0	0	0.0001	0	0	0	0	0
щ	0	0	0	0	0	0	0	0	0
ш	0	0	0	0	0	0	0	0	0
ю	0	0	0.0001	0.0005	0	0	0.0001	0	0
я	0	0.0003	0.0001	0.0003	0	0	0.0001	0	0
ь	0	0	0.0001	0	0	0.0004	0	0	0

ДОДАТОК Д. Частоти повторюваності триграм української мови

Частина отриманої таблиці відносних частот вживаності триграм української мови наведена у таблицях 1-5.

Таблиця 1 – Відносні частоти триграм

		_	а	б	в	г	д
а	_	0.00183	0.000345	0.000487	0.001534	0.000323	0.000855
а	а	0.000001	0	0.000002	0	0	0
а	б	0.000016	0.00009	0.000002	0	0	0.000002
а	в	0.001268	0.000315	0.000001	0.000006	0.000005	0.000127
а	г	0.000011	0.000561	0	0.000001	0	0.000003
а	д	0.000364	0.000624	0.000005	0.000013	0.000009	0.000005
а	е	0.000001	0	0	0.000001	0	0.000002
а	є	0.000732	0	0	0.000009	0	0
а	ж	0.000042	0.000187	0	0	0	0.000016
а	з	0.000085	0.000124	0.000002	0.000282	0.000001	0.000007
а	и	0	0	0	0	0	0
а	і	0.000001	0	0	0	0	0
а	ї	0.000014	0	0	0.000013	0	0.000001
а	й	0.000089	0.000004	0.000156	0.000059	0.000008	0.000074
а	к	0.000302	0.000201	0.000001	0.000009	0	0
а	л	0.00019	0.000774	0.000005	0.000001	0.000017	0.000003
а	м	0.00026	0.000169	0.000034	0.000011	0.000001	0
а	н	0.000289	0.000533	0.000001	0.000001	0.00013	0.000469
а	о	0.000004	0	0.000001	0	0	0.000004
а	п	0.000012	0.000145	0	0	0	0
а	р	0.000164	0.000461	0.000055	0.000027	0.000025	0.000086
а	с	0.000387	0.000092	0.000001	0.000007	0	0
а	т	0.000249	0.000292	0	0.00003	0.000001	0
а	у	0.000005	0.000001	0.000002	0.000005	0.000001	0.000016
а	ф	0.000025	0.00002	0	0	0.000002	0
а	х	0.000909	0.000026	0	0.000019	0	0
а	ц	0.000014	0.000002	0	0	0	0.000004
а	ч	0.000067	0.000285	0	0	0	0
а	щ	0.000001	0.000015	0	0	0	0
а	ш	0.000018	0.000013	0	0.000001	0	0
а	ю	0.000034	0	0	0.000001	0	0
а	я	0.000022	0	0	0.000061	0	0
а	ь	0	0	0	0	0	0

Таблиця 2 – Відносні частоти триграм

		є	ж	з	и	і	ї
а		0.00006	0.000067	0.001081	0.000009	0.000479	0.000064
а	а	0	0	0	0	0	0
а	б	0	0	0.000001	0.00005	0.000074	0
а	в	0.000001	0.000045	0.000003	0.000288	0.000217	0
а	г	0	0	0	0.00007	0.000058	0
а	д	0	0.000053	0.000023	0.000339	0.00037	0
а	е	0	0	0	0	0	0
а	є	0	0	0.000014	0	0	0
а	ж	0	0	0	0.000033	0.000031	0
а	з	0	0	0	0.000163	0.000128	0
а	и	0	0	0	0	0	0
а	і	0	0	0	0	0	0
а	ї	0	0.000001	0.000004	0	0	0
а	й	0.000001	0.000052	0.000008	0	0.000002	0
а	к	0	0.000001	0	0.000188	0.000224	0
а	л	0.000001	0.000001	0	0.001157	0.000974	0
а	м	0	0	0.000001	0.00099	0.00021	0
а	н	0.000001	0.000009	0.00001	0.0008	0.001059	0
а	о	0	0	0	0	0	0
а	п	0	0	0	0.000171	0.000093	0
а	р	0.000049	0.000004	0.000002	0.000243	0.00026	0.000001
а	с	0	0	0	0.000189	0.000122	0
а	т	0	0	0	0.001332	0.000251	0
а	у	0	0	0.000007	0	0	0
а	ф	0	0	0	0.000009	0.000172	0
а	х	0	0	0	0.000128	0.000237	0
а	ц	0	0	0	0.000025	0.001097	0
а	ч	0	0	0	0.000097	0.000093	0
а	щ	0	0	0	0.000045	0.000009	0
а	ш	0	0	0	0.000085	0.000013	0
а	ю	0	0	0	0	0	0
а	я	0	0	0	0	0	0
а	ь	0	0	0	0	0	0

Таблиця 3 – Відносні частоти триграм

		й	к	л	м	н	о
а	_	0.000124	0.00065	0.000193	0.000596	0.00084	0.000579
а	а	0	0.000001	0.000001	0.000002	0.000002	0
а	б	0	0.000012	0.000099	0	0.000009	0.000412

продовження таблиці 3

		й	к	л	м	н	о
а	в	0	0.0001	0.000281	0.000012	0.000455	0.000268
а	г	0	0	0.000017	0.000015	0.000059	0.000248
а	д	0	0.000132	0.00001	0.000116	0.000147	0.000134
а	е	0	0	0.000003	0.000001	0	0
а	є	0	0.000002	0	0.00006	0.000001	0
а	ж	0	0.000039	0.000055	0	0.000072	0.000009
а	з	0	0.000049	0.000005	0.00001	0.000094	0.00022
а	и	0	0	0	0.000001	0.000001	0
а	і	0	0	0	0	0.000001	0
а	ї	0	0.000007	0.000007	0.000002	0.000472	0
а	й	0	0.000086	0.000108	0.000126	0.000216	0.000492
а	к	0	0.000004	0.000086	0.000001	0.000001	0.000595
а	л	0	0.00003	0.000024	0.000008	0.000001	0.000517
а	м	0	0.000053	0.000004	0.000006	0.000028	0.000244
а	н	0	0.000153	0.000001	0.000001	0.001683	0.000877
а	о	0	0.000002	0.000002	0	0.000005	0
а	п	0	0.000006	0.000053	0	0.00001	0.000169
а	р	0	0.000151	0.000046	0.000169	0.000171	0.000504
а	с	0	0.000037	0.000119	0.000006	0.000498	0.000175
а	т	0	0.000292	0.000023	0.000019	0.000167	0.000618
а	у	0	0.000225	0.000006	0.000001	0.000024	0
а	ф	0	0	0.000001	0	0.000003	0.000013
а	х	0	0	0.000004	0.000005	0.000005	0.000295
а	ц	0	0.000001	0.000001	0	0	0.000003
а	ч	0	0.000025	0	0	0.000111	0.000007
а	щ	0	0	0	0	0	0.000008
а	ш	0	0.000016	0.000003	0.000001	0.000023	0.000275
а	ю	0	0	0	0	0	0
а	я	0	0.000005	0.000001	0.000003	0.000006	0
а	ь	0	0	0	0	0	0

Таблиця 4 – Відносні частоти триграм

		п	р	с	т	у	ф
а	_	0.00189	0.000749	0.001161	0.000788	0.000402	0.000176
а	а	0	0.000014	0.000001	0.000001	0	0
а	б	0	0.000052	0.000035	0	0.000076	0
а	в	0.000014	0.000037	0.000301	0.000287	0.000041	0
а	г	0	0.000071	0	0	0.000059	0

продовження таблиці 4

		п	р	с	т	у	ф
а	д	0.000004	0.000068	0.000071	0.00001	0.000288	0
а	е	0	0.000032	0.000001	0.000003	0	0
а	є	0	0.000001	0.000001	0.000389	0	0
а	ж	0	0	0	0	0.000042	0
а	з	0	0.000001	0	0	0.000119	0
а	и	0	0	0	0	0	0
а	і	0	0	0	0	0	0
а	ї	0	0.000001	0.000002	0.000003	0	0
а	й	0.000027	0.000009	0.000102	0.000063	0.000003	0.000001
а	к	0	0.000062	0.000058	0.000531	0.000126	0
а	л	0.000002	0	0.000001	0.000016	0.00019	0.000004
а	м	0.000049	0.000001	0.00001	0.000002	0.000053	0.000006
а	н	0	0.00002	0.000423	0.000395	0.000159	0.000002
а	о	0	0.000003	0.000004	0.000001	0	0
а	п	0.000002	0.000257	0.000011	0.000024	0.000031	0.000001
а	р	0.000013	0.00001	0.000233	0.000514	0.000061	0.000006
а	с	0.000041	0	0.000013	0.001373	0.000141	0.000003
а	т	0	0.000159	0.000048	0.000086	0.000352	0.000018
а	у	0.000002	0.000026	0.000007	0.000009	0	0
а	ф	0	0.000016	0.000013	0.000044	0.000006	0.000002
а	х	0	0.000004	0.000004	0.000022	0.000081	0
а	ц	0	0	0	0.000011	0.000009	0
а	ч	0	0	0	0	0.000034	0
а	щ	0	0	0	0	0.000012	0
а	ш	0	0	0	0.000045	0.000038	0
а	ю	0	0	0.000001	0.000466	0	0
а	я	0	0	0	0.000002	0	0
а	ь	0	0	0	0	0	0

Таблиця 5 – Відносні частоти триграм

		х	ц	ч	щ	ш	ю	я
а	_	0.000085	0.000165	0.000254	0.000111	0.000106	0.00001	0.0002
а	а	0	0	0	0	0	0	0
а	б	0.000001	0	0	0	0.000002	0	0
а	в	0	0.000076	0.000163	0	0.000068	0	0.00001
а	г	0	0	0	0	0	0	0
а	д	0.000009	0.000034	0.000005	0.000014	0.000001	0	0.00013
а	е	0	0	0	0	0	0	0
а	є	0	0.000002	0	0	0.000002	0	0

продовження таблиці 5

		х	ц	ч	щ	ш	ю	я
а	ж	0	0	0.000005	0	0	0	0
а	з	0	0.000001	0	0	0	0.000001	0
а	и	0	0	0	0	0	0	0
а	і	0	0	0	0	0	0	0
а	ї	0	0	0.000004	0	0	0	0
а	й	0.000003	0.000012	0.000014	0	0.000022	0	0
а	к	0.000001	0.00008	0	0	0.000006	0	0
а	л	0.000001	0.000001	0.000001	0	0	0.00007	0.00003
а	м	0.000001	0.00001	0.000001	0	0.000001	0	0.00013
а	н	0.000002	0.000312	0.000021	0.000005	0.000006	0	0.00002
а	о	0.000004	0	0.000005	0.000001	0	0	0
а	п	0	0	0.000002	0	0	0	0
а	р	0.000184	0.000014	0.000027	0	0.000085	0.00002	0.00008
а	с	0.000001	0.000001	0	0	0.000013	0	0.00027
а	т	0.000005	0.000002	0.000091	0	0.000003	0.00001	0.00001
а	у	0	0.000003	0.000003	0	0	0	0
а	ф	0	0	0	0	0	0	0
а	х	0	0	0.000001	0	0	0	0
а	ц	0	0.000002	0	0	0	0.00018	0.00001
а	ч	0	0.000001	0.000002	0	0	0	0
а	щ	0	0	0	0	0	0	0
а	ш	0	0	0	0	0	0	0
а	ю	0	0	0.00017	0.000001	0	0	0
а	я	0.000002	0	0.000001	0	0	0	0
а	ь	0	0	0	0	0	0	0

ДОДАТОК Е. Лістинг програми генерації псевдотексту Generator

```
#include <fstream>
#include <iostream>
#include <random>
#include<windows.h>
#include <ctime>
using namespace std;
unsigned char alf[34]={' ','а','б','в','г','д','е','є','ж','з','и','і','ї','й','к','л','м','н','о','п'
,'р','с','т','у','ф','х','ц','ч','щ','ш','ю','я','ь', 0};
float summ(float * arg)
{
	float sum=0;
	for(int i=0;i<33;i++)
	sum+=arg[alf[i]];
	return sum;
}
int Mark_gen (float * arg, float e)
{
	int i=0;
	while(e>arg[(int)alf[i]]&&i<33) i++;
	return (int)alf[i];
}
void main()
{
	locale::global(locale(".1251"));
//Відкриття потоків
	ifstream fin1 ("d:\\b.txt", std::ios::in);
	ifstream fin2 ("d:\\t.txt", std::ios::in);
	ofstream out ("d:\\псевдотекст.txt");
```

```
//Обява змінних
        float s[17]={0}, sum=0, e, m;
        unsigned int i=1,j,k, w;
        unsigned int buf1,buf2, buf;
// Ініціалізація масивів
        float **b = new float*, [256];
        float ***t=new float**[256];
        for (i = 0; i < 256; i++)
         {
   t[i] = new float *, [256];
   for (j = 0; j < 256; j++)
                {
    t[i][j] = new float, [256];
                        memset(t[i][j], 0, 256*sizeof(float));
                }
         }
  for (i= 0; i < 256; i++)
  {
                        b[i] = new float, [256];
                        memset(b[i], 0, 256*sizeof(float));
        }
        // Зчитування таблиць
        for(i=0;i<33;i++)
                        for(j=0;j<33;j++)
                                fin1>>b[(int)alf[i]][(int)alf[j]];
        for(i=0;i<33;i++)
                for(j=0;j<33;j++)
                        for(k=0;k<33;k++)
                                fin2>>t[(int)alf[i]][(int)alf[j]][(int)alf[k]];
```

```
//Приведення до стохастичного вигляду з одночасною побудовою функції розподілення
	for(i=0;i<33;i++)
	{
		sum=summ(b[(int)alf[i]]);
		m=0;
		for(j=0;j<33;j++)
			if (b[(int)alf[i]][(int)alf[j]]!=0)
			{
				b[(int)alf[i]][(int)alf[j]]=b[(int)alf[i]][(int)alf[j]]/sum;
				m+=b[(int)alf[i]][(int)alf[j]];
				b[(int)alf[i]][(int)alf[j]]=m;
			}
	}
	for(i=0;i<33;i++)
		for(j=0;j<33;j++)
		{
			sum=summ(t[(int)alf[i]][(int)alf[j]]);
			m=0;
			for(k=0;k<33;k++)
				if (t[(int)alf[i]][(int)alf[j]][(int)alf[k]]!=0)
				{
	t[(int)alf[i]][(int)alf[j]][(int)alf[k]]=t[(int)alf[i]][(int)alf[j]][(int)alf[k]]/sum;
					m+=t[(int)alf[i]][(int)alf[j]][(int)alf[k]];
					t[(int)alf[i]][(int)alf[j]][(int)alf[k]]=m;
				}
			}
	sum=0;
//Завдання розміру файлу
	cout<<"Размер файла (Кб):"<<endl;
```

```
        cin>>w;
        w*=1024;
//ініціалізація датчика псевдовипадкової величини
        mt19937 gen(time(0));
        uniform_real_distribution<> urd(0, 1);
//Затравка
        buf=32;
//Генерація тексту
        i=0;
        do {
bigr:   e=urd(gen);
                buf1=Mark_gen(b[buf],e);
                e=urd(gen);
                if((buf2=Mark_gen(t[buf][buf1],e))!=0)//перевірка можливого стану
                {
                        i+=2;
                        buf=buf2;
                        out.put((char)buf1);
                        out.put((char)buf);
                }
                else goto bigr;
                }
        while (i<w);
        system ("pause");
}
```

ДОДАТОК Є. Перелік документів на оптичному носії

1 Презентація_Ліпкін.ppt

2 Кваліфікаційна робота_Ліпкін.doc

ДОДАТОК Ж. Відгук керівника економічного розділу

Керівник розділу _____ _____

(підпис) (прізвище, ініціали)

ДОДАТОК З. Відгук керівника кваліфікаційної роботи

В І Д Г У К

на кваліфікаційну роботу студента групи 125м-22-2
Ліпкіна Микити Олексійовича на тему: «Розробка акустичного генератора шуму з мовоподібною завадою»

Кваліфікаційна робота представлена пояснювальною запискою на 105 с., містить 31 рис., 6 табл., 10 додатків, 12 джерел.

Метою кваліфікаційної роботи є підвищення ефективності активного захисту акустичної мовної інформації.

У спеціальній частині кваліфікаційної роботи дана характеристика методів та засобів протидії витоку мовної інформації акустичним каналом. Проведено порівняльний аналіз акустичних генераторів шуму. Запропоновано новий алгоритм генерації мовоподібної завади. Виконане імітаційне моделювання згідно з розробленою структурною схемою та оцінка ефективності сформованої завади.

В економічному розділі визначено капітальні витрати на програмні розробки, їх економічну доцільність доведено.

Практичне значення роботи полягає у розробці нових принципів побудови генератора мовоподібної завади.

Результати проведених у кваліфікаційній роботі досліджень можуть бути використані при розробці акустичних генераторів шуму, у прикладних задачах криптології та теорії інформації.

Наукова новизна дослідження полягає у розробці генератора мовоподібної завади для української мови.

В якості недоліків слід відзначити наступне: недотримання графіка проведення розробки, нечіткість окремих висновків і визначень, окремі невідповідності вимогам при оформленні.

Рівень запозичень у кваліфікаційній роботі відповідає вимогам «Положення про систему виявлення та запобігання плагіату».

В цілому робота задовольняє усім вимогам, а її автор Ліпкін М.О. заслуговує на оцінку « » та присвоєння кваліфікації «Магістр з кібербезпеки» за спеціальністю 125 Кібербезпека.

Керівник роботи,
к.т.н., доц. **Мацюк С.М.**