НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»
ФІЗИКО-ТЕХНІЧНИЙ ІНСТИТУТ
КАФЕДРА МАТЕМАТИЧНИХ МЕТОДІВ ЗАХИСТУ ІНФОРМАЦІЇ

«На правах рукопису»

УДК 004.056.55 (003.26.09)

«До захисту допущено»
В.о. завідувача кафедри
_____ М. М. Савчук
(підпис) (ініціали, прізвище)
“___” _____ 2020р.

Магістерська дисертація

на здобуття ступеня магістра

зі спеціальності 113 «Прикладна математика»
(код і назва)

на тему: Побудова модифікацій та криптоаналіз постквантових примітивів сімейства AJPS

Виконала: студентка 6 курсу, групи ФІ-83мн
(шифр групи)

Ядуха Дарія Вікторівна _____
(прізвище, ім’я, по батькові) (підпис)

Керівник: старший викладач, кандидат фізико-математичних наук
Фесенко А. В. _____
(посада, науковий ступінь, вчене звання, прізвище та ініціали) (підпис)

Рецензент: професор, доктор фізико-математичних наук
Олійник А. С. _____
(посада, науковий ступінь, вчене звання, прізвище та ініціали) (підпис)

Засвідчую, що у цій магістерській дисертації немає запозичень з праць інших авторів без відповідних посилань.

Студент _____
(підпис)

Київ – 2020 рік

Національний технічний університет України
«Київський політехнічний інститут
імені Ігоря Сікорського»
Фізико-технічний інститут
Кафедра математичних методів захисту інформації

Рівень вищої освіти: другий (магістерський) за освітньо-науковою програмою

Спеціальність: 113 «Прикладна математика»

ЗАТВЕРДЖУЮ
В.о. завідувача кафедри
_____ М. М. Савчук
(підпис) (ініціали, прізвище)
«___»_____20__ р.

ЗАВДАННЯ
на магістерську дисертацію студенту

Ядуха Дарія Вікторівна
(прізвище, ім'я, по батькові)

1. Тема дисертації: Побудова модифікацій та криптоаналіз постквантових примітивів сімейства AJPS ,
науковий керівник дисертації Фесенко Андрій В'ячеславович, кандидат фізико-математичних наук, старший викладач ,
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)
затверджені наказом по університету від _____р. № _____.

2. Термін подання студентом дисертації _____ .

3. Об'єкт дослідження: процеси перетворення інформації у постквантових системах криптографічного захисту.

4. Предмет дослідження: моделі постквантових криптографічних примітивів сімейства AJPS.

5. Перелік завдань, які потрібно розробити:
1) огляд сучасних постквантових криптопримітивів та їх класифікації за типом математичних об'єктів, що використовуються при побудові;

2) дослідження відомих криптографічних примітивів сімейства AJPS та аналіз задач MLHRSP і MLHCSP, на складності яких ґрунтується стійкість цих криптопримітивів;

3) пошук вразливостей криптосистем AJPS-1 та AJPS-2;

4) аналіз можливості побудови модифікації AJPS-1 шляхом зміни метрики та доведення необхідних для побудови модифікації властивостей обраної метрики при операціях за модулем числа Мерсенна;

5) дослідження властивостей арифметики за модулем узагальненого числа Мерсенна та за модулем числа Кренделла, які необхідні для модифікації криптосистеми AJPS-1 та AJPS-2 шляхом зміни класу чисел, що використовуються в якості модуля;

6) побудова модифікацій криптосистем AJPS-1 та AJPS-2 з використанням операцій за модулем узагальненого числа Мерсенна та за модулем числа Кренделла, і виконання порівняльного аналізу цих модифікацій та криптосистем AJPS-1 і AJPS-2.

6. Орієнтовний перелік ілюстративного матеріалу. Робота містить 3 таблиці та 45 рисунків.

7. Орієнтовний перелік публікацій. Результати роботи частково представлено у матеріалах п’яти міжнародних науково-практичних конференцій та опубліковано у науковому журналі.

8. Дата видачі завдання _____

Календарний план

№ з/п	Назва етапів виконання магістерської дисертації	Термін виконання етапів магістерської дисертації	Примітка
1.	Визначення орієнтовної теми дисертації та можливих напрямків дослідження по темі	вересень 2018 року	виконано
2.	Опрацювання основної літератури на тему дослідження	жовтень 2018 – січень 2019	виконано
3.	Огляд відомих атак на криптографічні примітиви сімейства AJPS	лютий 2019	виконано
4.	Дослідження опублікованих модифікацій криптопримітивів сімейства AJPS	березень – квітень 2019	виконано
5.	Криптоаналіз та пошук вразливостей криптосистем AJPS-1 та AJPS-2	квітень – вересень 2019	виконано
6.	Дослідження можливостей побудови модифікації криптосистеми AJPS-1 шляхом зміни метрики	червень – жовтень 2019	виконано
7.	Доведення необхідних властивостей арифметики за модулем числа Мерсенна для побудови модифікації AJPS-1 шляхом зміни метрики	жовтень 2019	виконано
8.	Побудова модифікацій криптосистем AJPS-1 та AJPS-2 з використанням арифметики за модулем узагальненого числа Мерсенна та числа Кренделла	листопад 2019	виконано

9.	Побудова модифікації криптосистеми AJPS-1 шляхом зміни метрики у її будові.	листопад – грудень 2019	виконано
10.	Програмна реалізація криптосистеми AJPS-1, модифікації криптосистеми AJPS-1 з використанням метрики *OSD*, модифікація AJPS-1 з використанням арифметики за модулем числа Мерсенна та модифікації криптосистеми AJPS-1 з використанням арифметики за модулем числа Кренделла	січень – лютий 2020	виконано
11.	Здійснення порівняльної характеристики криптосистеми AJPS-1 та її побудованих модифікацій	березень – квітень 2020	виконано

Студент _____ Д. В. Ядуха
(підпис) (ініціали, прізвище)

Науковий керівник дисертації _____ А. В. Фесенко
(підпис) (ініціали, прізвище)

РЕФЕРАТ

Кваліфікаційна робота обсягом 135 сторінок містить 45 рисунків, 3 таблиці та 60 джерел.

Протягом останніх років стрімко почала розвиватись постквантова криптографія, метою якої є розробка криптографічних примітивів, що були б стійкі до атак з використанням як квантового, так і класичного комп'ютерів. Починаючи з 2017 року триває конкурс постквантових асиметричних криптопримітивів під егідою Національного інституту стандартів та технологій США (NIST). Одним з учасників першого раунду конкурсу є механізм інкапсуляції ключів Mersenne-756839, основою якого є криптосистема AJPS.

Метою роботи є дослідження особливостей перетворення інформації в криптографічних примітивах сімейства AJPS, та їх модифікація задля збільшення рівня захищеності. Об'єктом дослідження є процеси перетворення інформації у постквантових системах криптографічного захисту. Предметом дослідження є моделі постквантових криптографічних примітивів сімейства AJPS. У роботі сформовано рекомендації для алгоритмів генерації ключів криптосистем AJPS-1 і AJPS-2 та побудовано атаку підміни на криптосистему AJPS-2. Доведено нові властивості арифметики за модулем числа Мерсенна, узагальненого числа Мерсенна та числа Кренделла. Побудовано модифікацію криптосистеми AJPS-1 шляхом зміни метрики, а також модифікації AJPS-1 та AJPS-2 шляхом зміни класу чисел, що використовуються в криптосистемах у якості модуля. Виконано порівняльний аналіз усіх побудованих модифікацій і криптосистем AJPS-1 та AJPS-2.

ПОСТКВАНТОВІ КРИПТОПРИМІТИВИ, КРИПТОСИСТЕМА AJPS, МОДУЛЬНА АРИФМЕТИКА, ЧИСЛО МЕРСЕННА, УЗАГАЛЬНЕНЕ ЧИСЛО МЕРСЕННА, ВАГА ХЕММІНГА

ABSTRACT

The volume of the qualitative work is 135 pages and it contains 45 figures, 3 tables and 60 sources.

In recent years, quantum-resistant cryptography has been steadily developing. Its aim is to develop the cryptographic primitives that would be resistant to attacks using both quantum and classical computers. In 2017, the National Institute of Standards and Technology (NIST) has launched the competition for quantum-resistant asymmetric cryptographic primitives, which is ongoing. One of the participants of the first round of the competition is the Mersenne-756839 key encapsulation mechanism, which is based on the AJPS cryptosystem.

The purpose of the research is to investigate the peculiarities of conversion of information in cryptographic primitives of the AJPS family, and modification of it in order to increase the security level. The object of the research is the processes of conversion of information in quantum-resistant cryptographic security systems. The subject of the research is the models of quantum-resistant cryptographic primitives of the AJPS family. The recommendations for key generation algorithms of the AJPS-1 and the AJPS-2 cryptosystems are represented in the work and the substitution attack on the AJPS-2 cryptosystem is constructed. The new properties of the arithmetic modulo Mersenne number, generalized Mersenne number and Crandall number are proved. The modification of the AJPS-1 cryptosystem by changing the metric, and also the modification of the AJPS-1 and the AJPS-2 by changing the class of numbers, which is used in the cryptosystems as a module, are created. The comparative analysis of all the modifications, which were created, and the cryptosystems AJPS-1 and AJPS-2 was done.

QUANTUM-RESISTANT CRYPTOGRAPHIC PRIMITIVES, AJPS CRYPTOSYSTEM, MODULAR ARITHMETIC, MERSENNE NUMBER, GENERALIZED MERSENNE NUMBER, HAMMING WEIGHT

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

Ham — функція обчислення ваги Хеммінга двійкового рядка

M_n — число Мерсенна $2^n - 1$, де $n \in \mathbb{N}$

$HM_{n,h}$ — множина лишків за модулем числа Мерсенна M_n, які мають вагу Хеммінга h

$\perp$ — помилка при розшифруванні

MLHRSP — задача ділення чисел з малою вагою Хеммінга за модулем числа Мерсенна (англ. *Mersenne Low Hamming Ratio Search Problem*)

D — ймовірнісний розрізнювач двох змінних

Δ^D — перевага розрізнення двох змінних розрізнювача D

$\mathcal{E}$ — функція шифрування коду корекції помилок

$\mathcal{D}$ — функція розшифрування коду корекції помилок

MLHCSP — задача лінійної комбінації чисел з малою вагою Хеммінга за модулем числа Мерсенна (англ. *Mersenne Low Hamming Combination Search Problem*)

$\mathcal{H}$ — випадковий оракул $\{0,1\}^\lambda \to x$, причому $x \in HM_{n,h}$

Ham_{dist} — функція обчислення відстані Хеммінга між двома двійковими рядками однакової довжини

E_X — довжина найбільшої послідовності нулів серед бітових записів числа X та циклічних зсувів числа X

$\overleftarrow{X}$ — циклічний зсув числа X на r позицій вліво, де $r \in \mathbb{N}$

OSD — фунція обчислення різниці кількості одиниць та кількості нулів у двійковому рядку

$\#1(X)$ — кількість одиниць у бітовому записі числа X

$\#0(X)$ — кількість нулів у бітовому записі числа X

$GM_{n,m}$ — узагальнене число Мерсенна $2^n - 2^m - 1$, де $n, m \in \mathbb{N}$, $n > m$

$CR_{n,c}$ — число Кренделла $2^n - c$, де $n, c \in \mathbb{N}$ та $\log_2 c \leqslant \frac{n}{2}$

$GM_{n,m,k}$ — узагальнене число Мерсенна $2^n - 2^m - 1 - k$, де $n, m, k \in \mathbb{N}$, причому $n > m$ та $k < 2^n - 2^m - 1$

$\mathbb{1}(A)$ — індикатор події A, тобто $\mathbb{1}(A) = 1$, якщо подія A виконується, і $\mathbb{1}(A) = 0$, якщо ні

$HG_{n,m,h}$ — множина лишків за модулем числа узагальненого Мерсенна $GM_{n,m}$, які мають вагу Хеммінга h

GMLHRSP — задача ділення чисел з малою вагою Хеммінга за модулем узагальненого числа Мерсенна (англ. *Generalized Mersenne Low Hamming Ratio Search Problem*)

$HC_{n,c,h}$ — множина лишків за модулем числа Кренделла $CR_{n,c}$, які мають вагу Хеммінга h

CRLHRSP — задача ділення чисел з малою вагою Хеммінга за модулем числа Кренделла (англ. *Crandall Low Hamming Ratio Search Problem*)

GMLHCSP — задача лінійної комбінації чисел з малою вагою Хеммінга за модулем узагальненого числа Мерсенна (англ. *Generalized Mersenne Low Hamming Combination Search Problem*)

CRLHCSP — задача лінійної комбінації чисел з малою вагою Хеммінга за модулем числа Кренделла (англ. *Crandall Low Hamming Combination Search Problem*)

ВСТУП

Актуальність дослідження. У зв'язку з великою кількістю досліджень технологій для побудови масштабованого квантового комп'ютера стрімко почала розвиватись і постквантова криптографія. Її метою є розробка криптографічних примітивів, які б були стійкі до атак з використанням як квантового, так і класичного комп'ютерів. Оскільки для більшості симетричних криптосистем можна побудувати аналоги, що мають вищий рівень захищеності та є стійкими до атак з використанням квантового комп'ютера, зусилля науковців зосереджені на розробці асиметричних криптопримітивів, зокрема тих, які реалізовують схему цифрового підпису або механізм інкапсуляції ключів.

Наприкінці 2017 року Національний інститут стандартів та технологій США (NIST) розпочав конкурс постквантових асиметричних криптопримітивів, які б реалізовували схему шифрування, механізм інкапсуляції ключів або схему цифрового підпису [1]. Згідно з календарним планом конкурсу, до 2024 року будуть опубліковані перші версії стандартів постквантової криптографії [2]. Тому є мотивація досліджувати постквантові криптосистеми та аналізувати їхню стійкість.

Одним з учасників конкурсу є механізм інкапсуляції ключів Mersenne-756839 [3], що базується на криптосистемі AJPS [4]. У будові криптосистеми використовується арифметика за модулем числа Мерсенна, яка може бути ефективно реалізована завдяки алгоритмам швидкого обчислення трудомістких операцій за модулем числа Мерсенна, таких як редукція [5, 6, 7, 8, 9, 10], множення [7, 10, 11], пошук оберненого відносно операції множення [9, 10, 12], покомпонентні операції додавання та множення [13], дискретне перетворення Фур'є [9] тощо.

Метою роботи є дослідження особливостей перетворення інформації в криптографічних примітивах сімейства AJPS, та їх модифікація задля збільшення рівня захищеності.

Досягнення поставленої мети передбачає такі **завдання дослідження**, які були виконані в роботі:

1) огляд сучасних постквантових криптопримітивів та їх класифікації за типом математичних об'єктів, що використовуються при побудові;

2) дослідження відомих криптографічних примітивів сімейства AJPS та аналіз задач MLHRSP і MLHCSP, на складності яких ґрунтується стійкість цих криптопримітивів;

3) пошук вразливостей криптосистем AJPS-1 та AJPS-2;

4) аналіз можливості побудови модифікації AJPS-1 шляхом зміни метрики та доведення необхідних для побудови модифікації властивостей обраної метрики при операціях за модулем числа Мерсенна;

5) дослідження властивостей арифметики за модулем узагальненого числа Мерсенна та за модулем числа Кренделла, які необхідні для модифікації криптосистеми AJPS-1 та AJPS-2 шляхом зміни класу чисел, що використовуються в якості модуля;

6) побудова модифікацій криптосистем AJPS-1 та AJPS-2 з використанням операцій за модулем узагальненого числа Мерсенна та за модулем числа Кренделла, і виконання порівняльного аналізу цих модифікацій та криптосистем AJPS-1 і AJPS-2.

При розв'язанні поставлених завдань використовувались методи теорії чисел, теорії кодування, лінійної та абстрактної алгебри, теорії ймовірностей, а також методи комп'ютерного та статистичного моделювання.

Об'єктом дослідження є процеси перетворення інформації у постквантових системах криптографічного захисту.

Предметом дослідження є моделі постквантових криптографічних примітивів сімейства AJPS.

Наукова новизна проведеного дослідження полягає в отриманні таких результатів:

1) проведено аналіз криптосистем AJPS-1 та AJPS-2, і описано

необхідні вимоги до алгоритмів генерації ключів даних криптосистем для забезпечення їх захищеної роботи;

2) доведено нові властивості арифметики за модулем числа Мерсенна та арифметики за модулем узагальненого числа Мерсенна, зокрема за модулем числа Кренделла;

3) вперше побудовано модифікацію криптосистеми AJPS-1 шляхом зміни метрики, що застосовується в її будові;

4) вперше побудовано узагальнення криптосистем AJPS-1 та AJPS-2 задля застосування більших класів чисел, ніж клас чисел Мерсенна, в якості модуля;

5) обґрунтовано переваги застосування узагальнених чисел Мерсенна у криптографічних примітивах сімейства AJPS задля дослідження можливості побудови нових модифікацій.

Практичне значення результатів. Побудовані модифікації криптографічних примітивів сімейства AJPS мають більшу варіативність параметрів системи, що дозволяє більш гнучке практичне застосування цих криптопримітивів. Отримані результати також можуть бути застосовані для модифікації відомих криптосистем з використанням операцій за модулем числа Мерсенна та створення нових ефективних і стійких постквантових криптопримітивів.

Апробація результатів. Результати даної роботи було частково представлено на таких конференціях:

1) XIX Міжнародній науково-практичній конференції молодих учених та студентів «Політ. Сучасні проблеми науки.», квітень 2019 року, м. Київ;

2) XX Міжнародній студентській науково-практичній конференції «Science and Technology of the XXI Century», листопад 2019 року, м. Київ;

3) XVIII Всеукраїнській науково-практичній конференції студентів, аспірантів та молодих вчених «Теоретичні та прикладні проблеми фізики, математики та інформатики», травень 2020 року, м. Київ;

4) Міжнародній науково-практичній конференції «Інформаційні

технології та комп'ютерне моделювання» (ІТКМ-2020), травень 2020 року, м. Івано-Франківськ;

5) XII Міжнародній науково-практичній конференції «Інтернет — Освіта — Наука» (ІОН-2020), травень 2020 року, м. Вінниця.

Публікації. Частину результатів магістерської дисертації опубліковано у таких працях:

1) Yadukha D. The necessary security requirements for the values used by the AJPS cryptosystem / D. Yadukha, A. Fesenko. // Theoretical and Applied Cybersecurity. — 2019. — №1. — С. 31–36. — ISSN 2664-2913.

2) Yadukha D. Requirements for Ciphertext of the AJPS-1 Cryptosystem / Dariya Yadukha // Science and Technology of the XXI Century: Proceedings of the XX International Students R&D Conference. — Kyiv: NTUU "Igor Sikorsky Kyiv Polytechnic Institute", 2019. — С. 200–202.

3) Yadukha D. Restriction on the Public Key of the AJPS Cryptosystem / Dariya Yadukha // Політ. Сучасні проблеми науки. Тези доповідей XIX Міжнародної науково-практичної конференції молодих учених і студентів. — Київ: НАУ, 2019. — С. 42–44.

4) Ядуха Д. Оцінка ваги Хеммінга суми та добутку чисел за модулем узагальненого числа Мерсенна / Д. Ядуха, А. Фесенко // Матеріали XVIII Всеукраїнської науково-практичної конференції студентів, аспірантів та молодих вчених «Теоретичні і прикладні проблеми фізики, математики та інформатики». — Київ: ВПІ ВПК «Політехніка», 2020.

5) Ядуха Д. Побудова модифікації постквантової криптосистеми AJPS-1 шляхом зміни метрики / Д. Ядуха, А. Фесенко // Матеріали статей Міжнародної науково-практичної конференції «Інформаційні технології та комп'ютерне моделювання». — Івано-Франківськ, 2020.

6) Ядуха Д. Побудова атаки підміни на криптосистему AJPS-2 з використанням моделі активного зловмисника / Д. Ядуха, А. Фесенко // Proceedings Of The XII International Scientific-Practical Conference «Internet-Education-Science». — Вінниця: ВНТУ, 2020.

1 ДОСЛІДЖЕННЯ КРИПТОСИСТЕМИ AJPS ТА ЇЇ МОДИФІКАЦІЙ

У цьому розділі розглядається класифікація постквантових криптографічних примітивів за типом математичних об'єктів, які використовуються в їхній будові, а також здійснюється детальний опис постквантової криптосистеми AJPS. Розділ містить аналіз стійкості даної криптосистеми, зокрема у розділі виконується огляд побудованих атак на AJPS. Досліджуються відомі модифікації криптосистеми AJPS, а також розглядається узагальнена модель алгебраїчних задач для побудови постквантових криптографічних примітивів.

1.1 Класифікація сучасних постквантових криптографічних примітивів

Основною метою постквантової криптографії є створення криптосистем, які можуть бути реалізовані на класичному комп'ютері, однак при цьому є захищеними, навіть якщо зловмисник використовує квантовий комп'ютер для проведення атаки. Постквантові криптопримітиви прийнято поділяти на сімейства, відповідно до того, які математичні об'єкти використовуються у їх побудові. Найбільш розповсюдженими є сімейство постквантових криптопримітивів на решітках (англ. *lattice-based cryptosystems*), сімейство на основі кодів корекції помилок (англ. *code-based cryptosystems*) та сімейство криптосистем, які використовують кільця многочленів (англ. *multivariate polynomial cryptosystems*). Також виділяють сімейство постквантових схем цифрового підпису на основі геш-функцій (англ. *hash-based signatures*) [14, 15].

Наприкінці 2017 року Національний інститут стандартів та

технологій США (NIST) розпочав конкурс постквантових асиметричних криптопримітивів, які б реалізовували схему шифрування або механізм інкапсуляції ключів (група PKE/KEM), або схему цифрового підпису (група Signature), або і те, і інше (група Signature & PKE/KEM) [1]. На участь у конкурсі подали заявки розробники 82 криптографічних примітивів. Перший раунд конкурсу тривав з грудня 2017 року до січня 2019, в ньому взяли участь 69 учасників [3, 16], серед яких:

1) *Lattice-based* — 21 схема PKE/KEM та 5 схем цифрового підпису, найбільш відомими представниками даного сімейства є Titanium [17], NewHope [18], Round5 [19] та криптосистеми на основі NTRU — pqNTRUSign [20], NTRU Prime [21], NTRUEncrypt [22] та NTRU-HRSS-KEM [23];

2) *Code-based* — 19 криптографічних примітивів типу PKE/KEM та 3 схеми цифрового підпису, відомими представниками є криптосистеми BIKE [24] та Classic McEliece [25];

3) *Hash-based* — 2 схеми цифрового підпису SPHINCS+ [26] та Gravity-SPHINCS [27];

4) *Multivariate polynomial* — 2 схеми PKE/KEM, 7 схем цифрового підпису та 2 криптосистеми, які реалізовують і механізм інкапсуляції ключів, і схему цифрового підпису, найвідомішими представниками цього сімейства є схеми цифрового підпису MQDSS [28] та Rainbow [29];

5) Сімейство *Other* містить 5 схем PKE/KEM, 2 схеми цифрового підпису та 1 криптосистему, яка реалізовує і те, і інше. Серед них схема цифрового підпису WalnutDSA [30], яка використовує групу кіс (англ. *braid group*); PKE/KEM-схеми SIKE [31] на основі ізогеній суперсингулярних еліптичних кривих та Mersenne-756839 [4], що використовує арифметику за модулем числа Мерсенна.

Другий раунд конкурсу розпочався 30 січня 2019 року і буде тривати до червня 2020 року. Учасниками другого раунду є 17 PKE/KEM-схем та 9 схем цифрового підпису [16, 32]. Однією з основних вимог, які перевіряються в рамках другого раунду конкурсу, є

ефективність роботи криптопримітивів на найрізноманітніших платформах та пристроях, які мають обмежену потужність процесора (наприклад, смарт-карти, мікрочипи тощо), адже криптопримітив, який буде обрано переможцем, має підтримувати певні типи легковагової криптографії [33]. Окрім дослідження роботи криптопримітивів на різних типах пристроїв, команда NIST зосереджується на потенційному впровадженні різних підходів до захисту, оскільки ніхто точно не знає, якими будуть можливості побудованого квантового комп'ютера.

Згідно з календарним планом конкурсу, третій раунд буде проведено у 2021 році та, відповідно, будуть визначені фіналісти конкурсу. А до 2024 року вже будуть опубліковані перші версії стандартів постквантової криптографії [2], які доповнять або замінять стандарти, які зараз вважаються найбільш вразливими до квантового комп'ютера, а саме стандарт цифрового підпису FIPS 186-4 [34] та стандарти NIST SP 800-56A [35] і NIST SP 800-56B [36], що описують механізми вибору ключів асиметричних криптосистем [14, 16].

1.2 Опис криптосистеми AJPS

Криптосистема AJPS розроблена групою відомих криптологів у складі Д. Аггарвала, А. Жу, А. Пракаша та М. Санта. Криптосистема AJPS має дві версії – для шифрування біту повідомлення (AJPS-1) [37] та для шифрування блоку повідомлення (AJPS-2) [4]. На основі криптосистеми AJPS-2 було побудовано механізм інкапсуляції AJPS-KEM [4]. Розглянемо ці криптосистеми далі.

Криптосистема AJPS-1

AJPS-1 [37] дозволяє зашифрувати один біт повідомлення, тобто відкритим текстом є число $b \in \{0, 1\}$. При побудові криптосистеми задається параметр захищеності λ. Відкритими параметрами системи є число Мерсенна $M_n = 2^n - 1$ та значення h, яке задовольняє умовам

$C_n^h \geqslant 2^\lambda$ та $4h^2 < n \leqslant 16h^2$. Відповідно до даних вимог, рекомендовано використовувати значення n та h, що описано у таблиці 1.1.

Таблиця 1.1 – Рекомендовані значення параметрів n та h криптосистеми AJPS-1

n	h	λ
1279	17	120
2203	23	174
3217	28	221
4253	32	260
9689	49	432

Слід зауважити, що тут і надалі для спрощення запису ототожнюємо числа за модулем числа Мерсенна та бітові рядки довжини n. Це можливо, оскільки між множинами цих об'єктів існує взаємно однозначне відображення.

Позначимо множину n-бітових чисел, які мають вагу Хеммінга h, як $HM_{n,h}$, тобто

$$HM_{n,h} = \{x \in \{0,1\}^n : \ Ham(x) = h\}.$$

Зауважимо, що множину $HM_{n,h}$ можна переформулювати як множину лишків за модулем числа Мерсенна M_n, які мають вагу Хеммінга h.

Розглянемо основні алгоритми криптосистеми AJPS-1 — генерації ключів, шифрування та розшифрування.

1) Алгоритм ***Gen*** — генерація відкритого та особистого ключів криптосистеми AJPS-1 полягає у виконанні наступних кроків.

а) Числа F та G обираються випадково та незалежно з множини $HM_{n,h}$. Особистим ключем є число G, а значення F є секретним параметром криптосистеми.

б) Відкритим ключ обчислюється за таким співвідношенням:

$$H = F \cdot G^{-1} \bmod M_n.$$

2) Алгоритм ***Enc***, за допомогою якого здійснюється шифрування біту $b \in \{0, 1\}$, містить такі кроки.

а) З множини $HM_{n,h}$ рівноймовірно та незалежно обираються числа A та B. Обрані числа є секретними параметрами криптосистеми.

б) Значення шифротексту обчислюється за формулою:

$$C = (-1)^b (A \cdot H + B) \bmod M_n.$$

в) Після цього значення шифротексту C передається відкритим каналом зв'язку отримувачу повідомлення.

3) Процедура розшифрування шифротексту C (алгоритм ***Dec***) полягає в наступному.

а) Отримувач обчислює значення

$$d = Ham(C \cdot G \bmod M_n).$$

б) Біт повідомлення визначається за значенням d відповідно до заданих співвідношень:

$$b = \begin{cases} 0, & \text{якщо } d \leqslant 2h^2; \\ 1, & \text{якщо } d \geqslant n - 2h^2; \\ \perp, & \text{інакше (випадок помилки розшифрування).} \end{cases}$$

Коректність розшифрування випливає з леми 1.1 [37].

Лема 1.1. *Для довільних чисел $A, B \in \{0, 1\}^n$ та числа Мерсенна $M_n = 2^n - 1$ виконуються такі співвідношення:*

1) $Ham(A + B \bmod M_n) \leqslant Ham(A) + Ham(B)$;

2) $Ham(A \cdot B \bmod M_n) \leqslant Ham(A) \cdot Ham(B)$;

3) Якщо $A \neq 0^n$, то $Ham(-A \bmod M_n) = n - Ham(A)$.

Стійкість криптосистеми AJPS-1 базується на складності задачі MLHRSP [37] — *Задачі ділення чисел з малою вагою Хеммінга за модулем числа Мерсенна* (англ. *Mersenne Low Hamming Ratio Search Problem*), яка формулюється таким чином.

Означення 1.1 (*Задача* MLHRSP)**.** Маючи число Мерсенна $M_n = 2^n - 1$, n-бітове число H і ціле число h, знайти числа F, G, такі, що $F, G \in HM_{n,h}$ та

$$H = F \cdot G^{-1} \bmod M_n.$$

Дослідження захищеності криптосистеми AJPS-1, зокрема аналіз складності задачі MLHRSP, буде наведено у підрозділі 1.3.

Криптосистема AJPS-2

AJPS-2 [4] дозволяє зашифрувати блок повідомлення довжиною λ, де λ — параметр захищеності, який задається при побудові криптосистеми, тобто відкритим текстом є повідомлення $m \in \{0,1\}^\lambda$. Відкритими параметрами системи є числа M_n та h, де h — фіксоване число, яке задовольняє умовам $h = \lambda$ та $10h^2 < n \leqslant 16h^2$, а також функції шифрування та розшифрування коду корекції помилок:

$$\mathcal{E} : \{0,1\}^\lambda \to \{0,1\}^n;$$

$$\mathcal{D} : \{0,1\}^n \to \{0,1\}^\lambda,$$

які обираються відповідно до такої умови — для того, щоб криптосистема була $(1-\delta)$-коректною, де δ — це ймовірність помилки, потрібно, щоб виконувалось наступне співвідношення:

$$\forall m \;\; Pr\{\mathcal{D}((F \cdot C_1) \oplus C_2) = m\} \geqslant 1 - \delta.$$

Далі опишемо основні алгоритми криптосистеми AJPS-2.

1) Алгоритм генерації ключів ***Gen*** відбувається таким чином.

   а) Числа F та G обираються випадково та незалежно з множини

$HM_{n,h}$. Число F є особистим ключем, а значення G — секретним параметром криптосистеми.

б) Рівноймовірно з усіх можливих n-бітових чисел обирається значення R.

в) Обчислюється значення T за таким співвідношенням

$$T = F \cdot R + G \bmod M_n.$$

г) Відкритим ключем є пара чисел (R, T).

2) При алгоритмі шифрування ***Enc*** повідомлення $m \in \{0,1\}^\lambda$ незалежно та рівноймовірно обираються числа A, B_1 та B_2 з множини $HM_{n,h}$. Шифротекстом повідомлення m є пара чисел (C_1, C_2), де C_1 та C_2 обчислюються відповідно до співвідношень:

$$C_1 = A \cdot R + B_1 \bmod M_n;$$

$$C_2 = (A \cdot T + B_2 \bmod M_n) \oplus \mathcal{E}(m).$$

Після цього пара чисел (C_1, C_2) передається відкритим каналом зв'язку.

3) При *розшифруванні* обчислюється

$$\mathcal{D}((C_1 \cdot F \bmod M_n) \oplus C_2).$$

Для того, щоб переконатись у правильності розшифрування, розглянемо чому дорівнює значення $C_2^* = C_1 \cdot F \bmod M_n$. Підставляючи співвідношення для обчислення C_1, маємо:

$$C_2^* = (A \cdot R + B_1) \cdot F \bmod M_n = A \cdot R \cdot F + B_1 \cdot F \bmod M_n. \qquad (1.1)$$

Зі співвідношення $T = F \cdot R + G \bmod M_n$ виражаємо $F \cdot R \bmod M_n$ та підставляємо його у формулу 1.1:

$$C_2^* = A \cdot (T - G) + B_1 \cdot F \bmod M_n = A \cdot T - A \cdot G + B_1 \cdot F \bmod M_n.$$

До отриманого результату додамо та віднімемо значення B_2:

$$C_2^* = (A \cdot T + B_2) - B_2 - A \cdot G + B_1 \cdot F \bmod M_n.$$

Оскільки $A \cdot T + B_2 \bmod M_n = C_2$, то маємо:

$$C_2^* = C_2 - B_2 - A \cdot G + B_1 \cdot F \bmod M_n.$$

За умовами побудови криптосистеми числа A, B_1, B_2, F та G мають вагу Хеммінга h, де число h суттєво менше за значення n. Тому відстань Хеммінга (кількість позицій у двійковому записі двох чисел однакової довжини, у яких значення різні) між числами C_2 та C_2^* з великою ймовірністю буде малою. Тоді, при обчисленні $\mathcal{D}(C_2 \oplus C_2^*)$, з великою ймовірністю отримаємо значення m.

Стійкість даної криптосистеми базується на складності задачі MLHCSP (*Задача лінійної комбінації чисел з малою вагою Хеммінга за модулем числа Мерсенна*, англ. *Mersenne Low Hamming Combination Search Problem*) [4].

Означення 1.2 (*Задача* MLHCSP)**.** Маючи число Мерсенна M_n, ціле число h та пару чисел (R, T), де R — випадково обране число з усіх n-бітових чисел, число T обчислено відповідно до співвідношення

$$T = F \cdot R + G \bmod M_n,$$

причому F та G обрані незалежно та випадково з множини $HM_{n,h}$, знайти числа F та G.

Задача MLHCSP є модифікацією задачі MLHRSP, для демонстрації цього застосуємо до співвідношення зі задачі MLHRSP, а саме $H = F \cdot G^{-1} \bmod M_n$, низку перетворень:

$$H = F \cdot G^{-1} \bmod M_n;$$

$$H - F \cdot G^{-1} = 0 \bmod M_n.$$

Множимо ліву та праву частину на G:

$$G \cdot H - F \bmod M_n = 0.$$

Потім множимо ліву та праву частину на H^{-1}:

$$G - F \cdot H^{-1} \bmod M_n = 0.$$

Значення $(-H^{-1} \bmod M_n)$ замінимо на випадкове n-бітове число R, тоді у лівій частині рівняння отримаємо співвідношення задачі MLHCSP:

$$G + R \cdot F \bmod M_n = T.$$

Далі розглянемо механізм інкапсуляції ключів на основі криптосистеми AJPS.

Механізм інкапсуляції AJPS-KEM

У механізмі інкапсуляції AJPS-KEM [4] використовуються алгоритми генерації ключів ***Gen***, шифрування ***Enc*** та розшифрування ***Dec*** криптосистеми AJPS-2, які описано раніше. Крім того, використовуються три випадкових оракули $\mathcal{H}_1, \mathcal{H}_2$ і $\mathcal{H}_3$, які приймають на вхід λ-бітовий рядок, а на вихід видають випадкове n-бітове число ваги Хеммінга h, тобто

$$\mathcal{H}_i : \{0,1\}^\lambda \to X, \text{ де } X \in HM_{n,h}, \ i = \overline{1,3}.$$

Такі оракули можна побудувати, використовуючи розширювану геш-функцію.

1) Процедура інкапсуляції ключа ***Encaps*** при відкритому ключі (R, T).

а) Обирається випадкове λ-бітове число K.

б) Використовуючи оракули $\mathcal{H}_1, \mathcal{H}_2$ і $\mathcal{H}_3$, обчислюються значення $A = \mathcal{H}_1(K)$, $B_1 = \mathcal{H}_2(K)$ та $B_2 = \mathcal{H}_3(K)$.

в) Шифротекстом є пара чисел $C = (C_1, C_2)$, де

$$C_1 = A \cdot R + B_1 \bmod M_n;$$

$$C_2 = (A \cdot T + B_2 \bmod M_n) \oplus \mathcal{E}(K).$$

г) Результатом процедури є значення C та K.

2) Процедура декапсуляції ***Decaps*** при відомому шифротексті $C = (C_1, C_2)$ та особистому ключі F.

а) Обчислюється значення K' таким чином:

$$K' = \mathcal{D}((F \cdot C_1 \bmod M_n) \oplus C_2).$$

б) За допомогою оракулів $\mathcal{H}_1, \mathcal{H}_2$ і $\mathcal{H}_3$ обчислюються значення $A' = \mathcal{H}_1(K')$, $B'_1 = \mathcal{H}_2(K')$ та $B'_2 = \mathcal{H}_3(K')$.

в) Формується пара чисел $C' = (C'_1, C'_2)$, де

$$C'_1 = A' \cdot R + B'_1 \bmod M_n;$$

$$C'_2 = (A' \cdot T + B'_2 \bmod M_n) \oplus \mathcal{E}(K').$$

г) Якщо $C = C'$, то результатом є значення K', інакше — символ $\perp$, який означає помилку при декапсуляції.

Стійкість описаного механізму інкапсуляції ключів AJPS-KEM базується, як і криптосистема AJPS-2, на складності задачі MLHCSP. У конкурсі постквантових криптопримітивів NIST бере участь криптографічний примітив Mersenne-756839, що реалізовує схему шифрування AJPS-2 та механізм інкапсуляції ключів AJPS-KEM з параметрами M_{756839} та $h = \lambda = 256$ [3].

1.3 Аналіз стійкості задач MLHRSP і MLHCSP, та огляд побудованих атак на криптосистему AJPS

Обґрунтування захищеності будь-якої криптосистеми, зокрема AJPS, здійснюється у кількох напрямках одночасно. Першим напрямком є доведення того, що криптосистема має певний рівень захищеності (наприклад, захищеність від атак з вибраним відкритим текстом, захищеність від атак з вибраним шифротекстом тощо). Другим напрямком є дослідження складності задач MLHRSP та MLHCSP, адже на складності даних задач базується стійкість криптосистеми AJPS. Також необхідно розглянути побудовані атаки на криптосистему та, оцінюючи їхню успішність, аналізувати захищеність криптосистеми AJPS.

Одним з найрозповсюдженіших рівнів захищеності сучасних криптопримітивів є семантична стійкість. Шляхом доведення семантичної стійкості обґрунтовується захищеність криптографічних примітивів.

Означення 1.3 ([38])**.** Нехай задана певна схема шифрування і є два повідомлення m_0, m_1 однакової довжини та шифротекст C, що є результатом шифрування одного з повідомлень m_0, m_1 за допомогою даної схеми шифрування. Схема шифрування вважається *семантично захищеною*, якщо зловмисник, маючи значення шифротексту C, не може визначити з ймовірністю успіху більше $\frac{1}{2}$ яке з двох повідомлень (m_0 чи m_1) було зашифровано.

Семантична стійкість вважається еквівалентною поняттю *нерозрізненості шифрування при атаці з вибраним відкритим текстом* [38] і позначається IND-CPA (англ. *indistinguishability* — нерозрізненість, *chosen-plaintext attack* — атака з вибраним відкритим текстом), однак таким чином визначена властивість IND-CPA розширює можливості зловмисника — при атаці з вибраним відкритим текстом зловмисник має послідовність пар повідомлень (m_0, m_1), в якій кожній парі відповідає певне значення шифротексту C_i [39]. Властивість

нерозрізненості при атаці з вибраним відкритим текстом вважається основною вимогою для більшості захищених асиметричних криптосистем, хоча деякі криптосистеми також забезпечують більш жорстку вимогу *нерозрізненості шифрування при атаці з вибраним шифротекстом*, яка позначається IND-CCA (англ. *chosen ciphertext attack* — атака з вибраним шифротекстом).

Для формального визначення властивостей IND-CPA та IND-CCA використовується таке поняття.

Означення 1.4 ([37])**.** Для будь-якого розрізнювача D, який приймає на вхід певну змінну та повертає значення біту $b \in \{0,1\}$, *перевагою розрізнення* розрізнювача D, що відрізняє дві випадкові змінні X та Y, називається значення $\Delta^D(X,Y)$, яке обчислюється таким чином:

$$\Delta^D(X,Y) = |Pr[D(X)=1] - Pr[D(Y)=1]|.$$

Важливим є результат, що демонструє взаємозв'язок між розрізненням значень функцій при певних аргументах та розрізненням самих аргументів — даний результат описано у наступній лемі [37].

Лема 1.2. *Нехай f — ймовірнісна функція, яка обчислюється за поліноміальний час, та нехай задано дві випадкових величини X, Y. Тоді якщо існує ймовірнісний розрізнювач D, який за поліноміальний час відрізняє значення $f(X)$ та $f(Y)$ з перевагою δ, то існує ймовірнісний розрізнювач D', що за поліноміальний час відрізняє випадкові величини X та Y з перевагою δ.*

Далі визначимо семантичну захищеність асиметричної схеми шифрування, використовуючи означення 1.4. Нехай схема шифрування складається з трьох алгоритмів — алгоритму генерації ключів $\boldsymbol{Gen}$, який формує відкритий ключ pk та особистий ключ sk, алгоритму шифрування $\boldsymbol{Enc}$, який приймає на вхід повідомлення m і відкритий ключ pk та повертає значення шифротексту C, а також алгоритму розшифрування $\boldsymbol{Dec}$, який за значеннями шифротексту C та особистого

ключа sk обчислює повідомлення m'.

Означення 1.5 ([4]). Асиметрична схема шифрування ($\boldsymbol{Gen}$, $\boldsymbol{Enc}$, $\boldsymbol{Dec}$) називається $(1-\delta)$-*коректною*, якщо для усіх допустимих повідомлень m виконується

$$Pr\ [\ \boldsymbol{Dec}(sk,\ \boldsymbol{Enc}(pk, m)) = m] \geqslant 1 - \delta.$$

Нехай для схеми шифрування ($\boldsymbol{Gen}$, $\boldsymbol{Enc}$, $\boldsymbol{Dec}$) визначено параметр захищеності λ. Усі інші параметри схеми шифрування, зокрема довжини ключів та шифротексту, є значеннями поліноміально обмежених функцій від λ. Тоді рівні захищеності даної схеми шифрування визначається таким чином.

Означення 1.6 ([37]). Асиметрична схема шифрування ($\boldsymbol{Gen}, \boldsymbol{Enc}, \boldsymbol{Dec}$) називається *семантично захищеною* (IND-CPA), якщо для будь-якого ймовірнісного поліноміального за часом розрізнювача та будь-яких повідомлень m_0 та m_1 однакової довжини при відомому відкритому ключі pk перевага розрізнення значень $C_0 = \boldsymbol{Enc}(pk, m_0)$ та $C_1 = \boldsymbol{Enc}(pk, m_1)$, де pk — відкритий ключ схеми шифрування, є щонайбільше $2^{-\lambda} \cdot poly(|C_i|)$.

Означення 1.7 ([37]). Асиметрична схема шифрування ($\boldsymbol{Gen}, \boldsymbol{Enc}, \boldsymbol{Dec}$) вважається IND-CCA-*захищеною*, якщо для будь-якого ймовірнісного поліноміального за часом роботи розрізнювача, якому надається доступ до оракула, що розшифровує за допомогою алгоритму $\boldsymbol{Dec}$ будь-який заданий шифротекст, для будь-яких повідомлень m_0 та m_1 однакової довжини, при відомому відкритому ключі pk, за умови, що розрізнювач не дає запит оракулу на розшифрування значення C_0 або C_1, перевага розрізнення C_0 та C_1, де $C_0 = \boldsymbol{Enc}(pk, m_0)$ і $C_1 = \boldsymbol{Enc}(pk, m_1)$, не більше за $2^{-\lambda} \cdot poly(|C_i|)$.

Також визначимо рівні захищеності механізму інкапсуляції ключів, що містить такі алгоритми: алгоритм генерації ключів $\boldsymbol{Gen}$, що формує значення відкритого ключа pk та особистого ключа sk, алгоритм

інкапсуляції ***Encaps***, який приймає на вхід відкритий ключ pk та віддає на вихід значення шифротексту C та ключа K, який належить множині допустимих ключів даного механізму інкапсуляції ключів, та алгоритм декапсуляції ***Decaps***, що за значеннями C і sk обчислює значення K'.

Означення 1.8 ([4])**.** Нехай задано механізм інкапсуляції ключів (***Gen***, ***Encaps***, ***Decaps***) та $\boldsymbol{Encaps}(pk) = (C, K)$ і $\boldsymbol{Decaps}(sk, C) = K'$. Тоді механізм інкапсуляції ключів (***Gen***, ***Encaps***, ***Decaps***) називається $(1-\delta)$-*коректним*, якщо виконується нерівність

$$Pr\left[K = K'\right] \geqslant 1 - \delta.$$

Нехай для (***Gen***, ***Encaps***, ***Decaps***) визначено параметр захищеності λ. Аналогічно як для схеми шифрування, усі інші параметри механізму інкапсуляції, включаючи довжини ключів та шифротексту, є значеннями поліноміально обмежених функцій від λ.

Означення 1.9 ([4])**.** Механізм інкапсуляції ключів (***Gen***, ***Encaps***, ***Decaps***) є *семантично захищеним* (IND-CPA), якщо для будь-якого ймовірнісного поліноміального за часом роботи розрізнювача при відомому відкритому ключі pk перевага розрізнення пар (C, K_0) та (C, K_1), де $(C, K_0) = \boldsymbol{Encaps}(pk)$ і K_1 обране рівноймовірно і незалежно від C, не перевищує $2^{-\lambda} \cdot poly(|C|, |K_0|)$.

Означення 1.10 ([4])**.** Механізм інкапсуляції ключів (***Gen***, ***Encaps***, ***Decaps***) є IND-CCA-*захищеним*, якщо для будь-якого ймовірнісного поліноміального за часом розрізнювача, що має доступ до оракула, який виконує ***Decaps***, перевага розрізнення значень (C, K_0) та (C, K_1), де $(C, K_0) = \boldsymbol{Encaps}(pk)$ і K_1 обране рівноймовірно і незалежно від C, при припущенні, що розрізнювач не дає запит на оракул зі значенням C, не більше за $2^{-\lambda} \cdot poly(|C|, |K_0|)$.

Тепер, коли усі необхідні означення введено, розглянемо рівень захищеності криптосистеми AJPS. Для доведення семантичної

захищеності криптосистем AJPS-1 та AJPS-2 необхідне наступне припущення [4].

Означення 1.11. *Припущення* MLHCA *про лінійну комбінацію чисел з малою вагою Хеммінга* (англ. *Mersenne Low Hamming Combination Assumption*) стверджує, що маючи n-бітове число Мерсенна $M_n = 2^n - 1$ і ціле число h таке, що $4h^2 < n \leqslant 16h^2$, перевага будь-якого ймовірнісного поліноміального за часом роботи розрізнювача, який намагається розрізнити значення

$$\left(\begin{bmatrix} R_1 \\ R_2 \end{bmatrix}, \begin{bmatrix} R_1 \\ R_2 \end{bmatrix} \cdot A + \begin{bmatrix} B_1 \\ B_2 \end{bmatrix} \right) \text{ та } \left(\begin{bmatrix} R_1 \\ R_2 \end{bmatrix}, \begin{bmatrix} R_3 \\ R_4 \end{bmatrix} \right)$$

є щонайбільше $O\left(2^{-h}\right)$, де R_1, R_2, R_3, R_4 — незалежно та рівноймовірно обрані n-бітові числа, а A, B — незалежно та рівноймовірно обрані числа з множини $HM_{n,h}$.

Зауважимо, що описане припущення MLHCA має суттєву схожість з припущенням задачі LWE [40] (англ. *Learning With Errors* — навчання з помилками), яка вважається однією з універсальних конструкцій для побудови захищених криптографічних примітивів. У припущенні MLHCA значення A відповідає секрету задачі LWE, а B_1 та B_2 відповідають значенню помилок у LWE [4].

Використовуючи припущення MLHCA, доводиться семантична захищеність криптосистеми AJPS.

Теорема 1.1 ([4])**.** *Криптосистеми* AJPS-1 *та* AJPS-2 *є семантично захищеними* (IND-CPA) *за умови виконання припущення* MLHCA.

Оскільки розглянуті схеми шифрування AJPS-1 та AJPS-2 є IND-CPA-захищеними, однак не є IND-CCA-захищеними, була мотивація інтегрувати дані криптопримітиви у більш складну схему задля досягнення властивості IND-CCA-захищеності. В результаті цього було

створено механізм інкапсуляції AJPS-KEM.

Теорема 1.2 ([4]). *Нехай* $\mathcal{H}$ *— випадковий оракул та криптосистема* AJPS-2 *є семантично захищеною* (IND-CPA). *Тоді механізм інкапсуляції ключів* AJPS-KEM *є* IND-CCA-*захищений.*

Далі проведемо аналіз складності задач MLHRSP та MLHCSP, на яких базується стійкість криптосистем AJPS-1 та AJPS-2. Зокрема, якщо стане можливим ефективно розв'язати одну з цих задач, то припущення MLHCA, на якому базується і IND-CPA-захищеність криптосистем AJPS-1 та AJPS-2, і IND-CCA-захищеність механізму інкапсуляції ключів AJPS-KEM, буде зламане. Тому важливо досліджувати складність даних задач задля забезпечення захищеності комунікацій у випадку використання криптосистеми AJPS.

Розглянемо результати проведених атак на задачі MLHRSP та MLHCSP далі. Враховуючи те, що задача MLHCSP отримана шляхом застосування ланцюга перетворень до MLHRSP, більшість атак на MLHRSP можна узагальнити на випадок MLHCSP.

1) *Атака «Вгадай і виграй»* [37] полягає у спробі вгадати значення F або G по відомому значенню H. В загальному випадку ймовірність успіху такої атаки була б $\frac{1}{C_n^h}$, однак, використовуючи властивість арифметики за модулем числа Мерсенна, цю ймовірність можна покращити. В атаці застосовується така властивість: при множенні числа на степінь двійки за модулем числа Мерсенна відбувається циклічний зсув даного числа [41], тобто вага Хеммінга числа при множенні на степінь двійки не змінюється. Тому можна без втрати загальності вважати, що старший біт числа F або G дорівнює 1. У такому випадку ймовірність успіху буде $\frac{1}{C_{n-1}^{h-1}}$. Слід зауважити, що дана атака не узагальнюється на випадок задачі MLHCSP — в цьому випадку ймовірність вгадати значення F або G дорівнює $\frac{1}{C_n^h}$, тобто атака потребує виконання перебору всіх можливих значень.

Також існує аналог атаки *«Вгадай і виграй»* безпосередньо на криптосистему AJPS-1 [37]. Дана атака полягає у вгадуванні значення A

або B, які використовуються при шифруванні, та, таким чином, дешифруванні початкового повідомлення b. Нехай вдалось вгадати значення B. Тоді якщо число

$$(C - B) \cdot H^{-1} \bmod M_n$$

має вагу Хеммінга h, то $b' = 0$; якщо ж значення

$$(-C - B) \cdot H^{-1} \bmod M_n$$

має вагу Хеммінга h, то $b' = 1$, а якщо жодна з цих умов не виконується, то число B визначено неправильно. В такому випадку виводиться випадковий біт b', тобто вгадується і повідомлення. Таким чином, $Pr[b' = b] = 1$, якщо число B вгадали правильно, і $Pr[b' = b] = \frac{1}{2}$, якщо B вгадали неправильно. Отже, ймовірність успіху атаки буде $\frac{1}{2}\left(1 + \frac{1}{C_n^h}\right)$. Аналогічно можна намагатись вгадати число A, тоді обчислюється вага Хеммінга чисел

$$(C - A \cdot H \bmod M_n) \text{ і } (-C - A \cdot H \bmod M_n).$$

2) *Атака з використанням частково-чуттєвої геш-функції* [43] полягає у наступному: розділяємо n-бітове число G так, що

$$G = G_1 \oplus G_2,$$

де G_1 — n-бітове число, яке має $\lfloor \alpha \cdot h \rfloor$ одиниць у перших $\lfloor \alpha \cdot n \rfloor$ бітах, а все інше — нулі, а G_2 має $\lfloor (1-\alpha) \cdot h \rfloor$ одиниць в останніх $\lfloor (1-\alpha) \cdot n \rfloor$ бітах, а інші біти — нулі. За умовою задачі MLHRSP

$$H = F \cdot G^{-1} \bmod M_n.$$

Очевидно, що

$$H \cdot G = F \bmod M_n,$$

тобто число $H \cdot G \bmod M_n$ має вагу Хеммінга h, оскільки число F має вагу Хеммінга h. Нехай $H \in \{0,1\}^n$. Тоді можна представити значення $H \cdot G \bmod M_n$ таким чином:

$$H \cdot G \bmod M_n = (H \cdot G_1 \bmod M_n) \oplus (H \cdot G_2 \bmod M_n).$$

Атака полягає у пошуку колізій між наборами $\{H \cdot G_1 \bmod M_n\}$ та $\{H \cdot G_2 \bmod M_n\}$ шляхом пошуку однакових геш-значень чисел з двох наборів. У класичній моделі обчислень складність такої атаки дорівнює $O\left(\left(C_{n-1}^{h-1}\right)^{1/2}\right)$ при $\alpha = \frac{1}{2}$, а у квантовій — $O\left(\left(C_{n-1}^{h-1}\right)^{1/3}\right)$ при $\alpha = \frac{1}{3}$. Дана атака успішно узагальнюється на випадок задачі MLHCSP, складність атаки при цьому залишається такою ж.

3) *Атака «Слабкий ключ»* [42] дозволяє визначити значення F та G у випадку, якщо усі ненульові біти чисел F та G знаходяться у правій половині двійкового запису n-бітового числа, тобто якщо обидва числа менші за $\sqrt{M_n}$. При побудові атаки застосовується метод раціональної реконструкції, тобто метод пошуку раціонального числа, використовуючи результат редукції цього числа за деяким модулем. При цьому значення F та G обчислюються через неперервні дроби $\frac{H}{M_n}$. Ймовірність успіху такої атаки, тобто ймовірність того, що значення F та G будуть менші за $\sqrt{M_n}$, дорівнює $\frac{1}{2^{2h}}$. Таким чином, складність атаки дорівнює $2^{2h} \cdot n^{o(1)}$.

4) Аналогічно до попередньої атаки була створена *атака з використанням алгоритму LLL* [42]. В даній атаці на числа F та G накладається умова, щоб усі їхні ненульові значення бітів знаходились згруповано у одній частині бінарного запису числа. Атака використовує ідею розділення бінарного представлення числа на проміжки, які будуть використовуватись для побудови решітки. Тому дана атака має назву *Slice-and-Dice*. Якщо інтервали підібрані правильно, то найкоротшими векторами побудованої решітки будуть значення F та G. Складність такої атаки дорівнює $(2 + \delta + o(1))^{2h}$ для деякої дуже малої константи δ, $\delta > 0$ [43].

Також при реалізації даної атаки можна використовувати SVP-оракули для пошуку значень F та G. Такий спосіб дозволяє збільшити ймовірність успіху атаки до $\left(\frac{1}{2} + o(1)\right)^{2h}$, однак також збільшує час виконання операції редукції решітки до $2^{(2+\delta)\cdot h + o(h)}$ [43]. Дана атака узагальнюється для розв'язку задачі MLHCSP з такими ж оцінками складності.

Для застосування алгоритму Гровера до даної атаки та для реалізації алгоритму *LLL* через квантові вентилі необхідно 2^{52} кубітів при значенні числа Мерсенна M_{756839} (рекомендоване для застосування в AJPS-KEM на конкурсі NIST) [44]. Для цього також необхідне застосування великої кількості квантових вентилів, а саме $2^{85} + 2^{66}$ для вищезазначених параметрів [44], що ставить під сумнів ефективність даної атаки на практиці, навіть якщо великомасштабний квантовий комп'ютер буде побудований.

У травні 2019 атаку *Slice-and-Dice* переформулювали у атаку безпосередньо на криптосистему AJPS-2 [45]. Ця атака дозволяє відрізнити випадок, коли блок повідомлення дорівнює послідовності нулів, від усіх інших. У випадку, коли $m = 0$ та $\mathcal{E}(m) = 0$, будується решітка, до якої застосовується алгоритм *LLL*. Внаслідок цього значення A, B_1, B_2 коректно визначаються з ймовірністю $\left(\frac{2}{3}\right)^h$ для кожного значення. Складність такої атаки дорівнює $O(2^{1{,}75h})$ [45].

5) Показано, що задача MLHCSP зводиться до задачі цілочисельного лінійного програмування (ILP, англ. *Integer Linear Programming*) [46]. Доведено, що в загальному випадку задача ILP є NP-складною [47], але насправді складність розв'язку задачі ILP залежить від конкретних параметрів. Наприклад, якщо число змінних зафіксовано, то задачу ILP можна звести до задачі лінійного програмування, яка має поліноміальну складність. Враховуючи це, а також те, що неможливо перевірити, чи існує розв'язок заданої у ILP системи, отримане зведення не дозволяє робити висновки про NP-складність задачі MLHCSP.

Однак, зведення задачі MLHCSP до задачі ILP дозволило доповнити клас слабких ключів у криптосистемі AJPS-2 новими значеннями [46]. Виявилось, що якщо у бінарних записах чисел F та G нулі знаходяться згруповано, то менше ніж за n^3 викликів ILP-оракула можна отримати значення F та G, тобто зламати криптосистему. Для реалізації атаки необхідно, щоб числа F та G задовольняли умові

$$E_F + E_G \geqslant n,$$

де E_F — довжина найбільшої послідовності нулів у бітовому записі числа F, аналогічно для E_G. Слід зауважити, що при обчисленні E_G або E_F кількість старших та молодших бітів, які дорівнюють 0, сумуються, тобто, якщо $A = 0011000$, то $E_A = 5$. Ймовірність такої слабкої пари (F, G) приблизно дорівнює 2^{-11} для значень M_{1279} та $h = 17$ (це одні з рекомендованих параметрів для застосування криптосистеми [37]). Тому необхідно після процедури генерації ключів виконувати перевірку того, що згенеровані значення не є слабкими.

6) У березні 2020 року опубліковано метод атаки з вибраними шифротекстами, який базується на атаці *Slice-and-Dice* [48]. Метод дозволяє, використовуючи шифротексти, на яких алгоритм розшифрування криптосистеми повернув помилку, оцінити значення особистого ключа. Слід зауважити, що у [48] описано оцінки результативності такої атаки на криптосистему Ramstake [49], але зазначено, що такий ж підхід може бути використаний і для AJPS, однак з меншою ефективністю.

Криптосистема Ramstake, як і AJPS, використовує арифметику за модулем числа Мерсенна і вагу Хеммінга в ролі метрики, та також є учасником першого раунду конкурсу постквантових криптопримітивів NIST. Опублікована атака дозволяє оцінити значення особистого ключа криптосистеми Ramstake, використовуючи 2^{12} різних шифротекстів, на яких виникла помилка розшифрування, за $O(2^{46})$ тактів роботи

квантового комп'ютера [48]. Ймовірність виникнення помилки розшифрування у Ramstake дорівнює 2^{-64}, а в Mersenne-756839 (версії криптосистеми AJPS-2, що представлена на конкурсі NIST) ймовірність помилки розшифрування дорівнює 2^{-239}, що і обґрунтовує меншу ефективність такого методу атаки для примітивів сімейства AJPS.

1.4 Дослідження наявних модифікацій криптосистеми AJPS

Криптосистема AJPS, завдяки своїй конструкції, має багато різних можливих напрямків для модифікації. Зокрема, протягом останнього року опубліковано кілька статей, присвячених модифікації AJPS — найбільш відомими з них є застосування алгоритму пошуку з поверненням до AJPS [50] та побудова схеми формування коду автентифікації повідомлень на основі задачі MLHCSP [51], на складності якої базується стійкість криптосистеми AJPS-2.

Механізми інкапсуляції ключів AJPS-KEM-Bivariate та AJPS-KEM-Trivariate

Однією з найбільш відомих побудованих модифікацій криптосистеми AJPS є механізми інкапсуляції ключів на основі AJPS з використанням алгоритму пошуку з поверненням (англ. *backtracking*). Існує два варіанти модифікації, а саме *двопараметрична* та *трипараметрична* схеми механізму інкапсуляції ключів: двопараметрична схема AJPS-KEM-Bivariate основана на криптосистемі AJPS-1, а трипараметрична схема AJPS-KEM-Trivariate — на криптосистемі AJPS-2 [50]. Розглянемо кожну з них далі.

У механізмі інкапсуляції ключів AJPS-KEM-Bivariate застосовуються наступні алгоритми.

1) Для генерації ключів використовується алгоритм ***Gen*** криптосистеми AJPS-1.

2) Алгоритм інкапсуляції ***Encaps*** приймає на вхід публічні параметри та відкритий ключ криптосистеми AJPS-1, а повертає значення шифротексту C. У криптосистемі AJPS-1 для шифрування біту повідомлення рівноймовірно та незалежно обирались значення A та B з множини $HM_{n,h}$. У даній модифікації шифруються саме значення A та B, причому шифротекст обчислюється таким чином:

$$C = A \cdot H + B \bmod M_n.$$

3) У алгоритмі декапсуляції ***Decaps*** обчислюється значення

$$Solve_{x,y}[C \cdot G = F \cdot x + G \cdot y \bmod M_n],$$

де $Solve_{x,y}$ — це алгоритм, який обчислює значення змінних x та y зі заданого рівняння за допомогою алгоритму пошуку з поверненням. Результатом застосування алгоритму декапсуляції є або пара чисел (A, B), або символ $\perp$, який означає помилку при декапсуляції.

Розглянемо основну ідею побудови алгоритму $Solve_{A,B}$, який обчислює значення A та B з рівняння

$$C \cdot G = F \cdot A + G \cdot B \bmod M_n.$$

Спочатку обчислюється значення однієї змінної за допомогою алгоритму пошуку з поверненням, а значення другої змінної обчислюється з отриманого рівняння, яке вже містить одне невідоме. Нехай за допомогою алгоритму пошуку з поверненням обчислюється значення A. Для цього перевіряється гіпотеза «*i-ий біт числа A дорівнює одиниці*» шляхом оцінювання параметра Δ, який обчислюється за таким співвідношенням:

$$\Delta = Ham(C \cdot G \bmod M_n) - Ham(C \cdot G - 2^i \cdot F \bmod M_n).$$

Якщо гіпотеза правильна, то Δ має бути додатним числом і приблизно дорівнювати значенню h (адже $Ham(F) = h$); якщо неправильна —

значення Δ передбачити не можна, оскільки, залежно від конкретних значень параметрів, Δ може приймати різні значення (в тому числі й h). Тому можливі ситуації «хибного прийняття», тобто коли вважається, що біт числа A визначено коректно, а насправді це не так. Саме тому використовується алгоритм пошуку з поверненням.

Розробники даної модифікації пропонують декілька кандидатів у алгоритми — усі вони використовують метод градієнтного спуску з мінімізацією параметра Δ, різниця між запропонованими алгоритмами лише у швидкодії. Розглянемо ці алгоритми далі.

Зауваження. Для опису алгоритмів пошуку з поверненнями $\mathcal{B}_1(w, P, e)$ та $\mathcal{B}_2(w, P, e, \phi)$ необхідно врахувати такі передумови.

1) Змінні M_n, F, G, n, h та C є глобальними змінними криптопримітиву та вважаються фіксованими в межах роботи алгоритму пошуку з поверненням.

2) P — це множина чисел, які містять у своєму бітовому записі біти, що визначені у попередніх ітераціях алгоритму з поверненням. На першій ітерації алгоритму множина P є порожньою, тобто $P = \emptyset$.

3) У змінну w буде записуватись різниця чисел для перевірки гіпотези, що певний біт числа дорівнює 1. Таким чином, для прикладу, який розглянуто вище

$$w = C \cdot G - 2^i \cdot F \bmod M_n$$

для перевірки гіпотези «*i-ий біт числа A дорівнює одиниці*». На першій ітерації алгоритму

$$w = C \cdot G \bmod M_n,$$

тобто

$$w = F \cdot x + G \cdot y \bmod M_n.$$

4) e — натуральне число, яке виконує роль індикатора. На першій ітерації алгоритму $e = 0$.

5) У обох алгоритмах пошуку з поверненням використовується

допоміжна функція $Confirm(x)$, яка за вхідним значенням x обчислює

$$y = C - G^{-1} \cdot F \cdot x \bmod M_n,$$

після чого перевіряє, чи $Ham(x) = Ham(y) = h$: якщо так, то результатом застосування функції є пара $\{True, y\}$, інакше — $\{False, \perp\}$.

6) Алгоритми пошуку з поверненням параметризуються константою Г, що визначає границю, перетинаючи яку можна відкинути шлях (певну множину чисел) з подальшого дослідження. Оскільки вага Хеммінга числа w на першій ітерації алгоритму є максимальною, то ймовірність відкинути потрібний шлях висока. Однак після виконання кількох ітерацій алгоритму вага Хеммінга w зменшується, тому для збільшення ефективності алгоритму доцільно замість константи Г використовувати функцію

$$\Gamma(Ham(w), Ham(\overline{w}), n, h).$$

7) У алгоритмі $\mathcal{B}_2$ для рандомізації використовується $\phi(x)$ — випадкова перестановка над цілими числами. Внаслідок використання рандомізації можуть бути відкинуті шляхи, які насправді є правильними. В такому випадку алгоритм пошуку з поверненнями не визначить потрібні значення. Для вирішення цього можна використовувати t випадкових перестановок $\phi_0, \ldots, \phi_{t-1}$ і застосовувати t разів алгоритм $\mathcal{B}_2$, сподіваючись, що принаймні один з t застосувань дасть відповідь.

Розглянемо безпосередньо приклади алгоритмів для обчислення $Solve_{x,y}$. Одним з таких алгоритмів є детермінований алгоритм $\mathcal{B}_1(w, P, e)$ [50].

Алгоритм 1.1. (Детермінований алгоритм $\mathcal{B}_1(w, P, e)$ для обчислення $Solve_{x,y}[C \cdot G = F \cdot x + G \cdot y \bmod M_n]$)

Вхід: w, P, e

Вихід: $\{x, y\}$ такі, що $C \cdot G = F \cdot x + G \cdot y \bmod M_n$ або *Failure*.

if $e = n$ **then return** *Failure*

else

if $|P| = h$ **then**

$x \leftarrow \sum_{i \in P} 2^i$

$\{s, y\} \leftarrow Confirm(x)$

if s **then return** $\{x, y\}$

$\overline{w} \leftarrow w - 2^e \cdot F \bmod M_n$

if $|Ham(\overline{w}) - Ham(w) + h| \leqslant \Gamma$ **then**

$\mathcal{B}_1(\overline{w}, P \cup \{e\}, e + 1)$

else

$\mathcal{B}_1(w, P, e + 1)$

Роглянемо іншу версію алгоритму для обчислення $Solve_{x,y}[C \cdot G = F \cdot x + G \cdot y \bmod M_n]$, а саме ймовірнісний алгоритм $\mathcal{B}_2(w, P, e, \phi)$ [50].

Алгоритм 1.2. (Рандомізований алгоритм $\mathcal{B}_2(w, P, e, \phi)$ для обчислення $Solve_{x,y}[C \cdot G = F \cdot x + G \cdot y \bmod M_n]$)

Вхід: w, P, e, ϕ

Вихід: $\{x, y\}$ такі, що $C \cdot G = F \cdot x + G \cdot y \bmod M_n$ або *Failure*.

if $e = n$ **then return** *Failure*

else

if $|P| = h$ **then**

$x \leftarrow \sum_{i \in P} 2^{\phi(i)}$

$\{s, y\} \leftarrow Confirm(x)$

if s **then return** $\{x, y\}$

$\overline{w} \leftarrow w - 2^{\phi(e)} \cdot F \bmod M_n$

if $|Ham(\overline{w}) - Ham(w) + h| \leqslant \Gamma$ **then**

$\mathcal{B}_1(\overline{w}, P \cup \{e\}, e + 1, \phi)$

else

$\mathcal{B}_1(w, P, e + 1, \phi)$

Зауваження. Розробники модифікацій AJPS-KEM-Bivariate та AJPS-KEM-Triviate пропонують застосовувати алгоритм пошуку з поверненням, який є оптимізацією розглянутих алгоритмів $\mathcal{B}_1$ та $\mathcal{B}_2$. В цьому алгоритмі визначаються найбільш пріоритетні шляхи, наприклад, на першій ітерації алгоритму це будуть шляхи зі значенням Δ, близьким до h.

Наведемо приклад роботи механізму інкапсуляції ключів, використовуючи описані алгоритми ***Gen***, ***Encaps*** та ***Decaps***, далі.

1) Обчислюємо значення двопараметричної криптографічно стійкої геш-функції від чисел A та B, тобто нехай $Hash(A, B) = k$.

2) Отримане на попередньому кроці значення k використовуємо як ключ симетричної криптосистеми. Нехай після шифрування за допомогою симетричної криптосистеми, використовуючи ключ k, початковому повідомленню m^* відповідає шифротекст C^*.

3) Обчислюємо значення

$$C = A \cdot H + B \bmod M_n,$$

де H — відкритий ключ отримувача повідомлення m^*.

4) Надсилаємо значення C та C^* отримувачу початкового повідомлення m^*.

Тоді отримувач, який володіє значеннями F та G, після отримання значень C та C^*, застосовує алгоритм

$$Solve_{x,y}[C \cdot G = F \cdot x + G \cdot y \bmod M_n]$$

та, таким чином, обчислює A та B. Після цього отримувач обчислює значення двопараметричної геш-функції $Hash(A, B)$ та визначає ключ симетричної криптосистеми k. Отриманий ключ використовується для розшифрування шифротексту C^*, внаслідок чого отримувач отримує значення початкового повідомлення m^*.

Для механізму інкапсуляції ключів AJPS-KEM-Bivariate доведена

теорема, яка обґрунтовує її стійкість. Розглянемо її далі.

Теорема 1.3 ([50]). *Злам механізму інкапсуляції ключів* AJPS-KEM-Bivariate *при параметрі* h *еквівалентний розв'язанню задачі* MLHRSP *при параметрі* $\frac{h}{2}$.

У AJPS-KEM-Bivariate вперше було використано криптосистему AJPS-1 за основу для побудови механізму інкапсуляції ключів. Також можна застосовувати AJPS-KEM-Bivariate як схему шифрування, що дозволить використовувати криптосистему AJPS-1 для шифрування блоку повідомлення, а не одного біту.

Механізм інкапсуляції ключів AJPS-KEM-Trivariate використовує такі алгоритми.

1) Публічні параметри M_n та h обираються відповідно до вимог криптосистеми AJPS-2. Алгоритм ***Gen***, аналогічно як відповідний алгоритм криптосистеми AJPS-2, обирає числа F та G випадково та незалежно з множини $HM_{n,h}$, а число R обирає випадково з усіх можливих n-бітових чисел. Особистим ключем є число F, а відкритим ключем — пара чисел (R, T), де

$$T = F \cdot R - G \bmod M_n.$$

2) Алгоритм інкапсуляції ***Encaps*** випадково та незалежно обирає числа A, B_1 та B_2 з множини $HM_{n,h}$. Результатом алгоритму є пара чисел (C_1, C_2), де

$$C_1 = A \cdot R + B_1 \bmod M_n;$$

$$C_2 = A \cdot T + B_2 \bmod M_n.$$

3) Алгоритм декапсуляції ***Decaps*** обчислює значення

$$Solve_{x,y,z}[F \cdot C_1 - C_2 = F \cdot y + G \cdot x + z \bmod M_n].$$

Результатом застосування алгоритму $Solve_{x,y,z}$ є або значення A, B_1, B_2 або символ $\perp$, який означає помилку декапсуляції. Алгоритм $Solve_{x,y,z}$

побудований аналогічно до розглянутого раніше алгоритму $Solve_{x,y}$.

Перевагою побудованої модифікації у порівнянні з AJPS-KEM є уникнення використання кодів корекції помилок, окрім того передача інформації, використовуючи AJPS-KEM-Trivariate, у 26 разів швидша за AJPS-KEM з параметрами, які рекомендовані до застосування розробниками AJPS [37, 50]. Більше того, криптосистему AJPS можна використовувати одночасно з розглянутими модифікаціями, що дозволить підвищити її захищеність.

Іншим напрямком модифікації є побудова криптографічних примітивів, стійкість яких б базувалась на складності задачі MLHCSP. Серед таких модифікацій можна виділити наступні.

1) *Інтерактивний протокол односторонньої автентифікації на основі* MLHCSP [51] полягає у наступному — нехай два користувачі володіють певним секретом, тоді за допомогою даного протоколу один з користувачів (**P**, англ. *prover* — той, що доводить) доводить іншому користувачу (**V**, англ. *verifier* — той, що перевіряє), що він справді володіє цим секретом (іншими словами, відбувається доведення того, що секрет, яким володіє **P** збігається з секретом, який має **V**). Таким чином, користувач **P** автентифікує себе для користувача **V**.

В останні роки такі протоколи стали важливим механізмом автентифікації для малопотужних пристроїв, таких як смарт-картки або RFID-теги (англ. *radio frequency identification* — радіочастотна ідентифікація).

Є три версії інтерактивного двораундового протоколу односторонньої автентифікації на основі MLHCSP, які мають різні рівні захищеності. Причому найбільш стійкий до атак протокол серед описаних є ускладненою версією інших. Тому розглянемо ідею найбільш простого протоколу серед представлених протоколів автентифікації на основі MLHCSP.

Відповідно до задачі MLHCSP секретом є число F. На першому кроці протоколу користувач **V** випадково обирає число R та надсилає його **P**.

Тоді користувач $\mathbf{P}$ випадково обирає число G та обчислює

$$T = F \cdot R + G \bmod M_n.$$

Після цього користувач $\mathbf{P}$ надсилає отримане значення T користувачу $\mathbf{V}$. Далі $\mathbf{V}$ обчислює

$$l = Ham(T - R \cdot F).$$

Якщо користувач $\mathbf{P}$ при обчисленні T використовував коректне значення F, то

$$T - R \cdot F = G \bmod M_n,$$

тобто $l = h$. Таким чином, якщо $l = h$, то $\mathbf{P}$ успішно підтвердив своє знання значення F.

Описаний протокол автентифікації є стійкий до атак з використанням моделі пасивного зловмисника, однак є вразливим до атак з використанням моделі активного зловмисника.

Ідея іншого протоколу [51] полягає у подвійному застосуванні наведеного протоколу автентифікації — замість секрету при обчисленні T' використовується значення T, яке визначається відповідно до умови задачі MLHCSP. Такий протокол є захищеним і від атак з використанням пасивного зловмисника, і від певного класу атак з використанням активного зловмисника, однак не є стійким до послідовних атак типу «зустріч посередині», при яких зловмисник може взаємодіяти зі значеннями $\mathbf{P}$ та $\mathbf{V}$ в незалежних сесіях протоколу для зміни значення секрету так, щоб $\mathbf{V}$ цього не помітив при фінальному виконанні протоколу.

Для запобігання вразливостей й до таких атак розроблено протокол, який використовує потрійне застосування співвідношення задачі MLHCSP [51] — при обчисленні T'' замість секрету застосовується значення T' з попереднього протоколу. Такий протокол може використовуватись для пристроїв з невеликою обчислювальною потужністю, адже, відповідно до умов для вибору параметрів протоколу, $n = \Theta(h)$ (у порівнянні з

$n = \Theta(h^2)$ в криптосистемі AJPS). Розробники протоколів рекомендують значення $n = 521$ та $h = 128$ (таким чином, $n = 4h$). В такому випадку складність виконання протоколу з потрійним застосуванням співвідношення задачі MLHCSP є щонайбільш $3n = 1563$ бітів.

Слід зауважити, що стійкість описаних протоколів ґрунтується на модифікації задачі MLHCSP, яка має назву MLHCSP-U (англ. MLHCSP *with uniform secret*) [51]. У даній задачі значення F обирається не з множини $HM_{n,h}$, як у задачі MLHCSP, а з множини усіх n-бітових чисел. Очевидно, що якщо задача MLHCSP є складною, то є складною і задача MLHCSP-U. Також доведено, що якщо задача MLHCSP-U є складною, то задача MLHCSP є складною, однак з іншими параметрами захищеності.

2) Іншим побудованим криптопримітивом на основі задачі MLHCSP-U є *код автентифікації повідомлень* (MAC-*код*) [51]. Доведено, що якщо задача MLHCSP-U є складною, то побудований MAC захищений від підробок за допомогою атак на основі вибраного відкритого тексту. Як і у розглянутому вище протоколі автентифікації, параметри MAC відповідають співвідношенню $n = \Theta(h)$, що дозволяє зменшити загальну складність роботи примітиву задля його використання у малопотужних пристроях.

3) Використовуючи модифікації задач MLHRSP та MLHCSP, побудовано дві псевдовипадкові функції PRF (англ. *pseudorandom function*), на основі яких побудовано генератори псевдовипадкових чисел PRG (англ. *pseudorandom generators*) [52]. При цьому використовуються задачі $\psi^n_{n^{c-1}}$-MLHRSP та $\psi^n_{n^{c-1}}$-MLHCSP для деякої константи c, $0 < c < 1$. Ідея таких модифікацій полягає у тому, що числа F та G мають розподіл $\psi^n_{n^{c-1}}$, внаслідок чого з великою ймовірністю (ймовірність залежить від конкретного значення c) значення ваги Хеммінга F та G належить інтервалу $[h, 2h]$.

Для побудови нових криптопримітивів на основі AJPS може бути також використана алгебраїчна модель постквантових криптопримітивів, які використовують обчислення у певному кільці (в явному або неявному

вигляді), що описана в [53]. Дана модель узагальнює деякі новітні алгебраїчні задачі, які використовуються при побудові постквантових криптографічних примітивів, та визначає кроки побудови таких криптопримітивів, а саме вибір нормованого поліному, який визначить основне кільце, вибір поліному, який визначить кільце шифротекстів, визначення рангу (довжини векторів у кільці шифротекстів) та вибір задачі, на складності якої буде ґрунтуватись стійкість побудованого криптопримітиву.

Описана модель включає три алгебраїчні задачі: Ideal-LWE (узагальнення задачі LWE [40]), Ideal-SIS (узагальнення задачі SIS (англ. *short integer solution*) [54]) та Ideal-NTRU (узагальнення задачі, що відповідає матричному варіанту криптосистеми NTRU [55]). Неформально запропоновані узагальнення задач полягають у розв'язанні системи лінійних рівнянь зі спотвореними правими частинами (Ideal-LWE), пошуку ненульового розв'язку системи лінійних рівнянь, який задовольняє певним визначеним умовам (Ideal-SIS), та вираження матриці у вигляді ділення двох матриць, на розмір яких також накладаються певні умови (Ideal-NTRU).

Відповідно до описаної моделі, криптосистема AJPS-1 є криптопримітивом на основі задачі Ideal-NTRU, а криптосистема AJPS-2 — на основі задачі Ideal-LWE [53].

Побудована алгебраїчна модель дозволяє не лише здійснити класифікацію наявних постквантових криптопримітивів задля застосування спільних методів аналізу складності алгебраїчних задач та стійкості криптографічних систем, а й дозволяє комбінувати різні складові компоненти для побудови нових криптографічних примітивів, зокрема нових модифікацій криптосистем AJPS-1 та AJPS-2.

Висновки до розділу 1

У даному розділі представлено класифікацію постквантових криптопримітивів відповідно до математичних об'єктів, які вони використовують, та описано постквантову криптосистему AJPS. Розглянуто різні версії криптосистеми AJPS, а саме криптосистему AJPS-1 для шифрування біту повідомлення та AJPS-2 — для шифрування блоку повідомлення. Також описано механізм інкапсуляції ключів AJPS-KEM, який базується на криптосистемі AJPS-2. Розділ містить обґрунтування стійкості описаних криптопримітивів та огляд побудованих атак на криптосистеми AJPS-1 та AJPS-2, зокрема на задачі MLHRSP та MLHCSP, на складності яких базується стійкість даних криптосистем. Окрім цього, в розділі розглянуто відомі модифікації криптосистеми AJPS, а саме механізми інкапсуляції ключів AJPS-KEM-Bivariate та AJPS-KEM-Trivariate з використанням алгоритму пошуку з поверненням, інтерактивний протокол односторонньої автентифікації та код автентифікації повідомлень на основі задачі MLHCSP, а також генератори псевдовипадкових чисел на основі задач MLHRSP та MLHCSP.

2 ПОБУДОВА ТА АНАЛІЗ МОДИФІКАЦІЙ КРИПТОСИСТЕМИ AJPS

У цьому розділі описуються результати криптоаналізу схем шифрування AJPS-1 та AJPS-2, а саме рекомендації до алгоритму генерації ключів обох криптосистем та атака підміни з використанням моделі активного зловмисника на криптосистему AJPS-2. Також у розділі здійснюється побудова модифікацій криптосистем AJPS-1 та AJPS-2 шляхом застосування інших класів чисел в ролі модуля та зміни метрики, і виконується їх порівняльний аналіз.

2.1 Криптоаналіз схем шифрування AJPS-1 та AJPS-2

У розділі 1 описано побудовані атаки на задачі MLHRSP та MLHCSP, на складності яких базується стійкість криптосистем AJPS-1 та AJPS-2 відповідно. Частину описаних атак можливо застосувати лише за умови, що виконується певне припущення про вигляд чисел F та G.

Для формування множини значень параметрів F і G, при яких криптосистеми AJPS-1 та AJPS-2 вразливі до опублікованих атак, розглянемо такі умови далі.

1) Атака «Слабкий ключ» є успішною для криптосистеми AJPS-1, якщо усі одиниці у двійковому представленні чисел F та G знаходяться у правій частині чисел, тобто кожне з чисел F та G менше за $\sqrt{M_n}$.

2) Для застосування атаки з використанням алгоритму LLL необхідно, щоб усі одиниці в двійковому записі чисел F та G знаходились згруповано.

3) Атака зі зведенням задачі MLHCSP до задачі ILP може бути застосовна до криптосистеми AJPS-2 у випадку, якщо числа F та G

задовольняють такій умові:

$$E_F + E_G \geqslant n,$$

де E_X — довжина найбільшої послідовності нулів серед бітових записів числа X та циклічних зсувів числа X, тобто найбільша послідовність може формуватись з старших та молодших бітів числа X одночасно.

Слід зауважити, що, навіть якщо потрібні для атак умови виконуються, складність описаних атак є все одно досить великою, що унеможливлює їх застосування на практиці. Однак, для підвищення стійкості криптосистем AJPS-1 та AJPS-2, можна виконувати низку необхідних перевірок при застосуванні алгоритму генерації ключів, і виконувати кроки алгоритму ***Gen*** повторно у випадку порушення однієї з вимог.

Окрім описаних умов на значення параметрів F та G, у [56] представлено умови на значення відкритого ключа H криптосистеми AJPS-1, а саме такі умови:

$$Ham(H) \neq 1;$$

$$Ham(H^{-1} \bmod M_n) \neq 1.$$

Узагальнюючи умови на F та G опублікованих атак, а також враховуючи знайдені обмеження на значення H криптосистеми AJPS-1, сформуємо рекомендації для алгоритмів генерації ключів ***Gen*** криптосистем AJPS-1 та AJPS-2.

Твердження 2.1 (Рекомендації для алгоритму генерації ключів криптосистеми AJPS-1)**.** *Нехай в результаті застосування алгоритму* ***Gen*** *криптосистеми* AJPS-1 *отримано G — особистий ключ, F — секретний параметр криптосистеми, $H = F \cdot G^{-1} \bmod M_n$ — відкритий ключ. Для захищеної роботи криптосистеми* AJPS-1 *необхідно, щоб значення F, G та H задовольняли таким умовам:*

1) Хоча б одне з чисел F та G має бути більшим або рівним $\sqrt{M_n}$, тобто має виконуватись умова

$$\left[\begin{array}{l} F \geqslant \sqrt{M_n}, \\ G \geqslant \sqrt{M_n}. \end{array}\right.$$

2) В бінарному записі хоча б одного з чисел F та G одиниці не згруповані разом (є хоча б один нуль між h одиницями), тобто виконується умова:

$$\left[\begin{array}{ll} F \neq 2^i \cdot M_h & \text{для деякого } i = \overline{0, n-h}, \\ G \neq 2^j \cdot M_h & \text{для деякого } j = \overline{0, n-h}. \end{array}\right.$$

3) Число H задовольняє таким умовам:

$$Ham(H) \neq 1;$$

$$Ham(H^{-1} \bmod M_n) \neq 1.$$

Якщо хоча б одна з наведених умов не виконується, то необхідно повторно виконати кроки алгоритму ***Gen*** *криптосистеми* AJPS-1 *для генерації ключів. Якщо усі наведені умови виконуються, то отримані значення G — особистого ключа та H — відкритого ключа можна використовувати для подальшої роботи криптосистеми.*

Доведення. Пункт 1 отримано з умови атаки «Слабкий ключ». Доведення пункту 3 наведено у [56]. Розглянемо обґрунтування пункту 2. Умовою застосування атаки з використанням алгоритму LLL є вимога, щоб усі одиниці в двійковому представленні чисел F та G знаходились згруповано. Оскільки числа F та G мають вагу Хеммінга h, то h одиниць мають знаходитись поруч (без нулів між ними) у двійковому записі цих чисел. Розглянемо число Мерсенна $M_h = 2^h - 1$. В двійковому записі воно має вигляд $11\ldots1$, причому одиниць рівно h. Таким чином, число M_h є

одним з «слабких» значень для чисел F, G. При множенні на 2 отримаємо

$$2 \cdot M_h \bmod M_n = 11 \ldots 10,$$

тобто ще одне «слабке» значення для чисел F та G. Оскільки числа F, G за умовою задачі MLHRSP є n-бітовими, то максимальний степінь двійки, на який можна домножити M_h для отримання «слабкого» значення, є $n-h$, оскільки

$$2^{n-h} \cdot M_h \bmod M_n = 11 \ldots 100 \ldots 0$$

є n-бітовим числом, яке містить рівно h одиниць та $n-h$ нулів. При $n-h+1$ отримаємо

$$2^{n-h+1} \cdot M_h \bmod M_n = 1 \ldots 100 \ldots 01,$$

адже множення на степінь двійки за модулем числа Мерсенна є циклічним зсувом числа вліво [41]. Таким чином, число $2^{n-h+1} \cdot M_h \bmod M_n$ не є «слабким» значенням, адже одиниці в його двійковому записі не є згрупованими. □

Твердження 2.2 (Рекомендації для алгоритму генерації ключів криптосистеми AJPS-2). *Нехай в результаті застосування алгоритму* ***Gen*** *криптосистеми* AJPS-2 *отримано F — особистий ключ та G — секретний параметр криптосистеми. Для захищеної роботи криптосистеми* AJPS-2 *необхідно, щоб значення F та G задовольняли таким умовам:*

1) В бінарному записі хоча б одного з чисел F та G одиниці не згруповані разом (є хоча б один нуль між h одиницями), тобто виконується умова:

$$\left[\begin{array}{ll} F \neq 2^i \cdot M_h & \text{для деякого } i = \overline{0, n-h}, \\ G \neq 2^j \cdot M_h & \text{для деякого } j = \overline{0, n-h}. \end{array}\right.$$

2) Бінарний запис чисел F та G не містить великі послідовності

нулів, тобто виконується умова:

$$E_F + E_G < n,$$

де E_X *— довжина найбільшої послідовності нулів серед числа* X *та циклічних зсувів числа* X *(найбільша послідовність може формуватись з старших та молодших бітів числа* X *одночасно).*

Якщо хоча б одна з наведених умов не виконується, то необхідно повторно виконати кроки алгоритму ***Gen*** *криптосистеми* AJPS-2 *для генерації ключів. Якщо усі наведені умови виконуються, то отримані значення* F *та* G *можна використовувати для подальшої роботи криптосистеми.*

Доведення. Доведення пункту 1 твердження 2.2 аналогічне доведенню пункту 2 твердження 2.1. Пункт 2 отримано з умови атаки зі зведенням задачі MLHCSP до задачі ILP. □

Зауваження. У пунктах 1 та 2 твердження 2.1 та у пункті 1 твердження 2.2 описуються умови на вибір значень F та G криптосистем AJPS-1 та AJPS-2. При алгоритмі генерації ключів обох криптосистем числа F та G обираються незалежно та випадково з множини $HM_{n,h}$, після чого рекомендовано виконувати описані перевірки. Варто зауважити, що не потрібно «слабкі» значення виключати з множини чисел, з якої випадково та незалежно обираються значення F та G. Оскільки для атак необхідно, щоб виконувались умови для обох чисел одночасно, то якщо, узагальнюючи описані умови на F, G, обмежити множину $HM_{n,h}$ (тобто забезпечити виконання умов до генерації), множина можливих значень F, G необґрунтовано зменшиться та, як наслідок, зменшиться час виконання повного перебору.

Раніше було доведено, що криптосистема AJPS-1 вразлива до атаки підміни з незмінним відкритим текстом при використанні моделі активного зловмисника [57]. Активний зловмисник може не лише зчитувати усі шифротексти, які передаються відкритим каналом зв'язку,

а також, при успішній атаці, може змінювати шифротексти та підмінювати повідомлення у каналі зв'язку непомітно для отримувача [58]. При атаці підміни зловмисник перехоплює справжній шифротекст від відправника та, використовуючи його, формує помилковий шифротекст, після чого відправляє його отримувачу. Атака підміни вважається успішною, якщо отримувач прийняв повідомлення за допустиме [59].

Виявилось, що криптосистема AJPS-2 також вразлива до атаки підміни з використанням моделі активного зловмисника. Для її побудови використовується властивість арифметики за модулем числа Мерсенна, яка описана у наступній лемі.

Лема 2.1. *Для довільних чисел* $A, B \in \{0,1\}^n$, *числа Мерсенна* $M_n = 2^n - 1$, *де* $n \in \mathbb{N}$, *та довільного числа* $r \in \mathbb{N}$ *такого, що* $r < n$, *виконується:*

$$\overleftarrow{A + B \bmod M_n} = \overleftarrow{A} + \overleftarrow{B} \bmod M_n,$$

де $\overleftarrow{X}$ *— циклічний зсув числа* X *на* r *позицій вліво.*

Доведення. Оскільки операція циклічного зсуву на r позицій за модулем числа Мерсенна є еквівалентом операції множення на 2^r [41], то значення $\overleftarrow{A + B \bmod M_n}$ можна представити таким чином:

$$\overleftarrow{A + B \bmod M_n} = (A + B) \cdot 2^r \bmod M_n =$$

$$= A \cdot 2^r + B \cdot 2^r = \overleftarrow{A} + \overleftarrow{B},$$

отже,

$$\overleftarrow{A + B \bmod M_n} = \overleftarrow{A} + \overleftarrow{B} \bmod M_n,$$

що і потрібно було довести. □

Використовуючи лему 2.1, можемо побудувати атаку підміни на криптосистему AJPS-2. Розглянемо її далі.

Твердження 2.3. *Атака підміни з модифікованим відкритим текстом є успішною для криптосистеми* AJPS-2*: маючи пару* (C_1, C_2),

зловмисник може обчислити шифротексти (C_1^*, C_2)*, де* $C_1^* = \overleftarrow{C_1}$ *— результат застосування операції циклічного зсуву числа* C_1 *на* r *позицій вліво, причому значення* r *— довільне натуральне число, яке менше за* n*.*

Доведення. Результатом алгоритму шифрування криптосистеми AJPS-2 є пара чисел (C_1, C_2), де

$$C_1 = A \cdot R + B_1 \bmod M_n,$$

$$C_2 = (A \cdot T + B_2 \bmod M_n) \oplus \mathcal{E}(m).$$

Значення A, B_1, B_2 обираються з множини $HM_{n,h}$ випадково та незалежно при кожному застосуванні алгоритму шифрування. Виконуючи циклічний зсув числа C_1, відповідно до леми 2.1, маємо:

$$\overleftarrow{C_1} = \overleftarrow{A \cdot R} + \overleftarrow{B_1} \bmod M_n.$$

Оскільки вага Хеммінга не змінюється при циклічному зсуві числа, то $\overleftarrow{B_1}$ має таку ж вагу Хеммінга, як число B_1, тобто $\overleftarrow{B_1} \in HM_{n,h}$. Оскільки число B_1 обирається випадково з множини $HM_{n,h}$ при застосуванні алгоритму шифрування і використовується лише один раз при обчисленні значення C_1, то використання значення $\overleftarrow{B_1}$ замість B_1 при обрахунку C_1 не змінює значення повідомлення, яке буде отримане при розшифруванні (C_1, C_2). Однак, використання значення $\overleftarrow{A \cdot R}$ замість $A \cdot R$ впливає на розшифроване повідомлення. Число R — частина відкритого ключа криптосистеми, тобто відоме зловмиснику значення, а число A обирається випадково з множини $HM_{n,h}$ при кожному застосуванні алгоритму шифрування. Якщо вважати, що при обчисленні циклічного зсуву значення $A \cdot R$ число R залишається незмінним, то маємо

$$\overleftarrow{A \cdot R} = Y \cdot R,$$

де Y — деяке n-бітове число, яке задовольняє заданій умові. Якщо число

Y має вагу Хеммінга h (хоча ймовірність цієї події порівняно мала), то значення C_1^* буде відповідати коректному повідомленню. Однак, навіть в такому випадку атака буде успішною, адже число A використовується не лише при обчисленні C_1, а і при обчисленні C_2. Таким чином, маємо:

$$(C_1^*, C_2) = (Y \cdot R + B_1, (A \cdot T + B_2) \oplus \mathcal{E}(m)),$$

тобто при розшифруванні (C_1^*, C_2) буде отримано повідомлення m^*, $m^* \neq m$. □

Таким чином, побудована атака підміни з використанням моделі активного зловмисника на криптосистему AJPS-2 є більш ефективною, ніж атака на AJPS-1, яка описана у [57], адже у випадку з AJPS-1 підміна шифротексту не змінювала вихідне повідомлення. Отже, криптосистема AJPS-2 не є стійкою до атаки підміни з використанням моделі активного зловмисника.

2.2 Розробка модифікації криптосистеми AJPS-1 шляхом зміни метрики

Криптосистема AJPS при побудові використовує операцію обчислення ваги Хеммінга, зокрема коректність розшифрування криптосистеми AJPS-1 базується на співвідношеннях для ваги Хеммінга чисел за модулем числа Мерсенна, які описано у лемі 1.1. Наступна модифікація демонструє та обґрунтовує можливість використання інших метрик, окрім ваги Хеммінга, у криптосистемі AJPS-1.

Введемо метрику OSD (англ. *One-side disbalance*) таким чином:

$$OSD(X) = \#1(X) - \#0(X),$$

де $\#1(X)$ позначає кількість одиниць у бінарному записі числа X, а, відповідно, $\#0(X)$ — кількість нулів в X.

Для чисел за модулем числа Мерсенна виконуються співвідношення, описані у наступній лемі.

Лема 2.2. *Для довільних чисел* $A, B \in \{0,1\}^n$ *та числа Мерсенна* $M_n = 2^n - 1$, *де* $n \in \mathbb{N}$, *виконуються такі співвідношення:*

1) $OSD(A + B \bmod M_n) \leqslant OSD(A) + OSD(B) + n$;

2) $OSD(A \cdot B \bmod M_n) \leqslant$

$$\leqslant \frac{OSD(A) \cdot OSD(B)}{2} + n \cdot \left(\frac{OSD(A) + OSD(B) + n}{2} - 1 \right);$$

3) $OSD(-A \bmod M_n) = -OSD(A)$.

Доведення. Необхідно помітити, що метрика OSD може бути представлена через метрику Ham таким чином:

$$OSD(X) = Ham(X) - (n - Ham(X)) = 2 \cdot Ham(X) - n,$$

де X — n-бітове число.

1) Використовуючи описане співвідношення метрик OSD та Ham до значення $OSD(A + B \bmod M_n)$, маємо:

$$OSD(A + B \bmod M_n) = 2 \cdot Ham(A + B \bmod M_n) - n.$$

Застосовуючи пункт 1 леми 1.1, отримаємо:

$$OSD(A + B \bmod M_n) \leqslant 2 \cdot Ham(A) + 2 \cdot Ham(B) - n.$$

Знову використовуючи залежність між метриками Ham та OSD, маємо:

$$OSD(A + B \bmod M_n) \leqslant 2 \cdot Ham(A) + OSD(B) =$$

$$= 2 \cdot Ham(A) - n + n + OSD(B) = OSD(A) + OSD(B) + n.$$

2) Використовуючи співвідношення метрик OSD та Ham та пункт 2 леми 1.1, маємо:

$$OSD(A \cdot B \bmod M_n) = 2 \cdot Ham(A \cdot B \bmod M_n) - n \leqslant 2 \cdot Ham(A) \cdot Ham(B) - n.$$

Виконуючи заміну метрики відповідно до співвідношення

$$Ham(X) = \frac{OSD(X) + n}{2},$$

отримаємо:

$$OSD(A \cdot B \bmod M_n) \leqslant 2 \cdot \frac{OSD(A) + n}{2} \cdot \frac{OSD(B) + n}{2} =$$

$$= \frac{OSD(A) \cdot OSD(B) + n \cdot (OSD(A) + OSD(B)) + n^2}{2} - n.$$

3) Застосовуючи пункт 3 леми 1.1 та залежність значення OSD деякого числа від його ваги Хеммінга, отримаємо:

$$OSD(-A \bmod M_n) = 2 \cdot Ham(-A \bmod M_n) - n =$$

$$= 2 \cdot (n - Ham(A)) - n = n - Ham(A) = -OSD(A).$$

□

Використовуючи описані у лемі 2.2 результати можна побудувати модифікацію криптосистеми AJPS-1, яка буде використовувати метрику OSD замість ваги Хеммінга. Розглянемо таку модифікацію далі.

1) Для генерації ключів даної модифікації використовується алгоритм ***Gen*** криптосистеми AJPS-1. Слід зауважити, що

$$OSD(F) = OSD(G) = 2h - n,$$

оскільки за умовою AJPS-1 вага Хеммінга чисел F та G дорівнює h. Для зручності позначимо $q = 2h - n$.

2) При шифруванні використовується алгоритм ***Enc*** криптосистеми AJPS-1. Зауважимо, що для значень A та B, які використовуються при шифруванні, аналогічно як для чисел F та G, виконується

$$OSD(A) = OSD(B) = q.$$

3) В алгоритмі розшифрування ***Dec*** даної модифікації криптосистеми AJPS-1 обчислюється значення s:

$$s = OSD(C \cdot G \bmod M_n).$$

Тоді біт b визначається відповідно до значення s за таким співвідношенням:

$$b = \begin{cases} 0, & \text{якщо } s \leqslant (n+q)^2 - n; \\ 1, & \text{якщо } s \geqslant n - (n+q)^2; \\ \perp, & \text{інакше (випадок помилки розшифрування).} \end{cases}$$

Коректність розшифрування слідує з леми 2.2.

Перевагою такої модифікації є збільшення множини значень, які приймає параметр s, відповідно до якого в алгоритмі розшифрування визначається біт повідомлення. Даний результат отримано експериментально при серії з 1000000 застосувань алгоритмів шифрування та розшифрування криптосистеми AJPS-1 та її побудованої модифікації при фіксованих значеннях ключів. Відкритий та особистий ключі були отримані внаслідок застосування алгоритму генерації ключів кожної з криптосистем. Таким чином, кількість значень s модифікації AJPS-1 з використанням метрики OSD більша за кількість значень d криптосистеми AJPS-1. Отримані результати продемонстровано в таблиці 2.1 та на рисунках 2.1, 2.2, 2.3 та 2.4.

Таблиця 2.1 – Довжини інтервалів, яким належать значення d криптосистеми AJPS-1 та значення s модифікації криптосистеми AJPS-1, яка використовує метрику OSD, при серії з 1000000 застосувань алгоритмів шифрування та розшифрування при фіксованих значеннях ключів

n	h	Метрика, що застосовується в криптосистемі	Довжина інтервалу значення d при $b = 0$	Довжина інтервалу значення d при $b = 1$
1279	17	Ham	105	112
		OSD	212	194
2203	23	Ham	147	141
		OSD	292	286
3217	28	Ham	171	170
		OSD	370	352
4253	32	Ham	201	204
		OSD	390	418
9689	49	Ham	294	319
		OSD	656	620

Зауваження. Довжина інтервалу обчислювалась як різниця максимального та мінімального значень серед отриманих результатів.

На рисунках 2.1 та 2.2 зображено розподіл значення d при застосуванні алгоритму розшифрування криптосистеми AJPS-1 та розподіл значення s модифікації криптосистеми AJPS-1 з використанням метрики OSD відповідно при параметрах $n = 9689$, $h = 49$ та значенні біту $b = 0$. На рисунках 2.3 та 2.4 зображено розподіл d та s при параметрах $n = 9689$, $h = 49$ та при значенні біту $b = 1$.

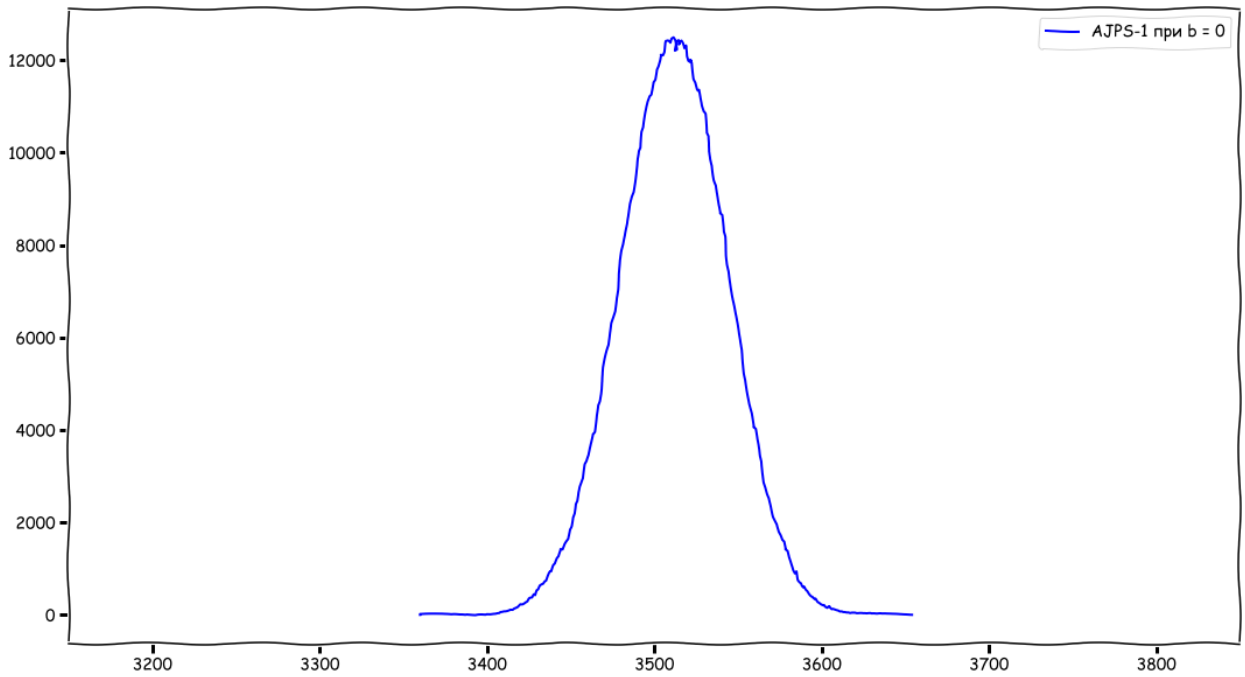

Рисунок 2.1 – Розподіл значення d криптосистеми AJPS-1 при параметрах $n = 9689$, $h = 49$ та значенні біту $b = 0$

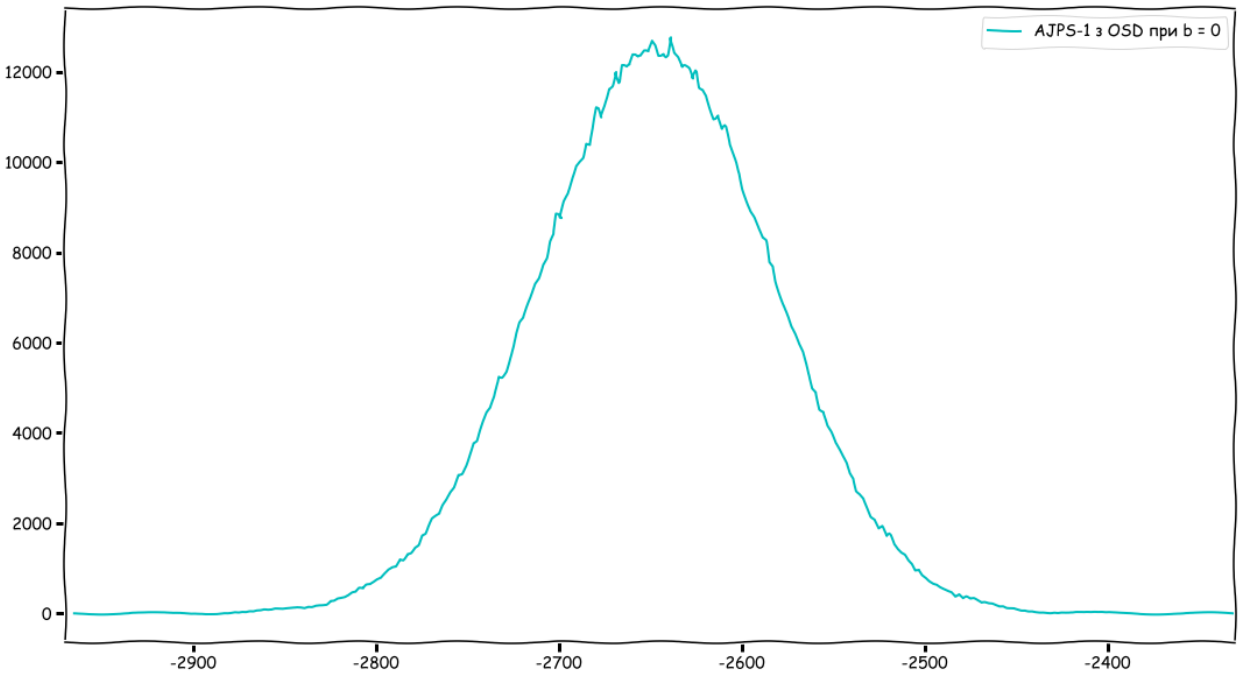

Рисунок 2.2 – Розподіл значення s модифікації криптосистеми AJPS-1 з використанням метрики OSD при $n = 9689$, $h = 49$ та $b = 0$

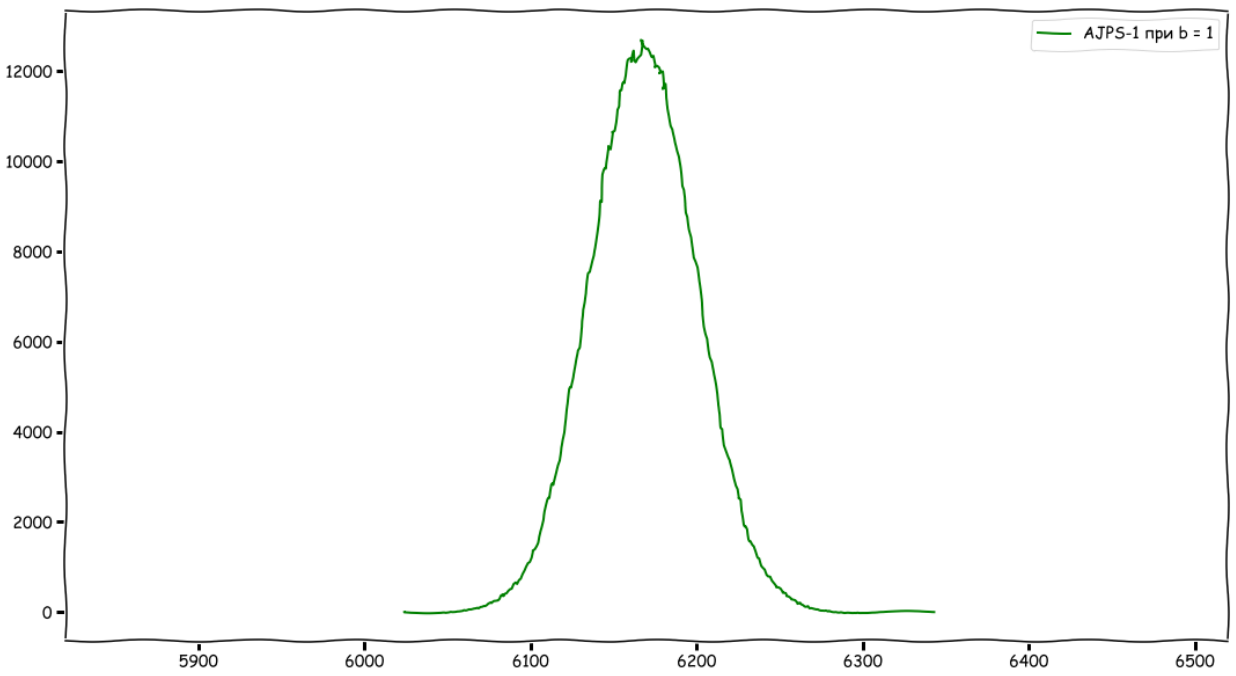

Рисунок 2.3 – Розподіл значення d криптосистеми AJPS-1 при параметрах $n = 9689$, $h = 49$ та значенні біту $b = 1$

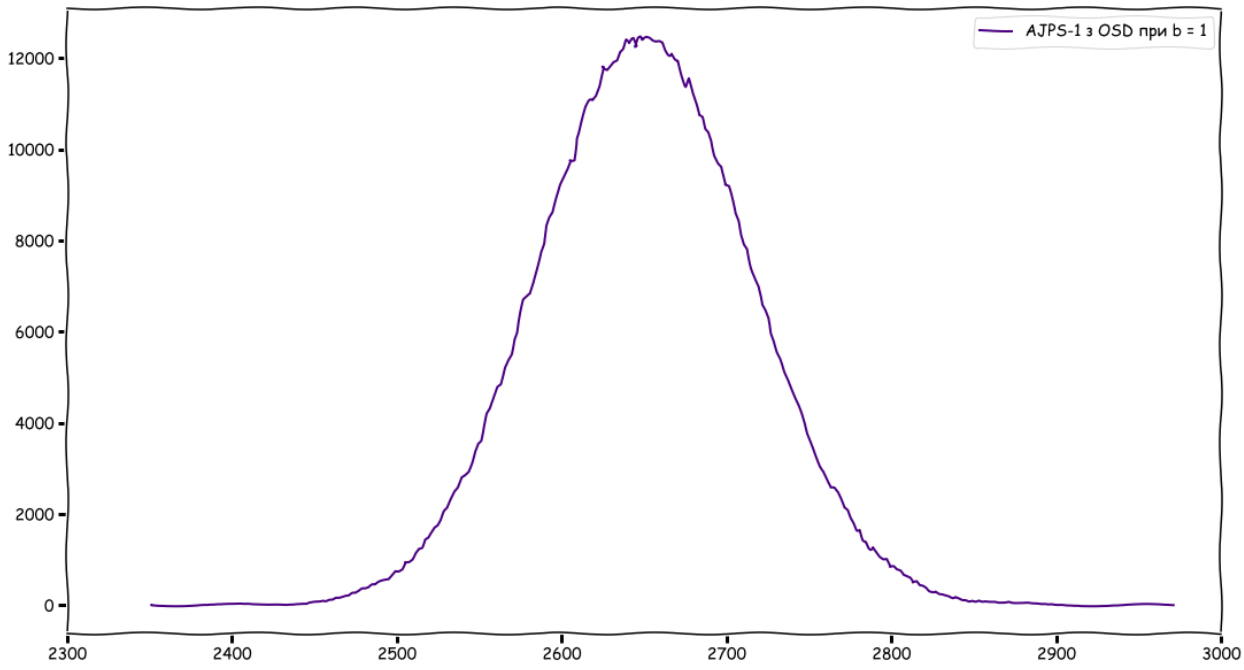

Рисунок 2.4 – Розподіл значення s модифікації криптосистеми AJPS-1 з використанням метрики OSD при $n = 9689$, $h = 49$ та $b = 1$

Зауваження. Графіки розподілу значення d криптосистеми AJPS-1 та розподілу значення s модифікації AJPS-1 з використанням метрики OSD для інших рекомендованих відповідно таблиці 2.5 значень параметрів n та h наведені у додатку Б.1.

Значення d криптосистеми AJPS-1 та значення s модифікації криптосистеми AJPS-1 з використанням метрики OSD є випадковими величинами з нормальним розподілом. Однак, на рисунках 2.1, 2.2, 2.3 та 2.4 бачимо, що при зміні метрики в AJPS-1 дисперсія випадкової величини збільшується, тобто множина можливих значень параметра s більша ніж множина можливих значень параметра d.

Таким чином, описана модифікація криптосистеми AJPS-1 з використанням метрики OSD має перевагу у порівнянні з AJPS-1, адже через невелику кількість значень параметра d криптосистеми AJPS-1 можуть бути успішні атаки на основі відомих шифротекстів, які направлені на визначення особистого ключа G.

2.3 Узагальнення властивостей арифметики за модулем числа Мерсенна

Як вже було сказано раніше, арифметика за модулем числа Мерсенна володіє багатьма перевагами для застосування у криптографії завдяки існуванню ефективних алгоритмів обчислення трудомістких операцій. Такі алгоритми часто узагальнюються на випадок більших класів чисел, зокрема узагальнених чисел Мерсенна. Таким чином, існують алгоритми швидкого обчислення редукції [5, 8, 9, 10, 60] та множення двох чисел [8, 10] за модулем узагальненого числа Мерсенна.

Розглянемо два класи чисел, що є узагальненнями чисел Мерсенна — числа $GM_{n,m} = 2^n - 2^m - 1$, де $n, m \in \mathbb{N},\ n > m$, та числа Кренделла (інша назва — *псевдомерсенівські числа*).

Означення 2.1 ([6]). *Числом Кренделла* називається число $CR_{n,c} = 2^n - c$ для деяких констант $n \in \mathbb{N}$ та $c \in \mathbb{Z}$, причому $\log_2 c \leqslant \frac{n}{2}$.

В рамках даної роботи будемо розглядати числа Кренделла $CR_{n,c} = 2^n - c$, де параметр c належить класу натуральних чисел, тобто $c \in \mathbb{N}$.

Очевидно, що клас узагальнених чисел Мерсенна $GM_{n,m}$ включається в клас чисел Кренделла, адже узагальнені числа Мерсенна $GM_{n,m}$ — це числа Кренделла при параметрі $c = 2^m - 1$ для деякого числа $m \in \mathbb{N}$, $m < n$.

Оскільки коректність розшифрування криптосистеми AJPS-1 ґрунтується на співвідношеннях для ваги Хеммінга суми, добутку чисел та оберненого елемента відносно операції додавання за модулем числа Мерсенна, які описано у лемі 1.1, то для можливого узагальнення криптосистеми AJPS-1 шляхом застосування інших класів чисел потрібно використовувати аналогічні залежності ваги Хеммінга, але за модулем чисел, які є узагальненнями чисел Мерсенна. Такі співвідношення описано в наступних теоремах.

У теоремі 2.1 визначено співвідношення для обчислення ваги Хеммінга суми двох чисел за модулем узагальненого числа Мерсенна $GM_{n,m,k}$ виду

$$GM_{n,m,k} = 2^n - 2^m - 1 - k,$$

де $n, m, k \in \mathbb{N}$, причому $n > m$ та $k < 2^n - 2^m - 1$.

Очевидно, що числа $GM_{n,m,k}$ є узагальненням чисел $GM_{n,m}$, адже числа $GM_{n,m}$ — це числа GM при значенні параметра $k = 0$. Таким чином, $GM_{n,m,k} = GM_{n,m} - k$. Зокрема, можна розглядати числа $GM_{n,m,k}$ як числа Кренделла при $c = 2^m + 1 + k$ для параметрів $k, m \in \mathbb{N}$, $m < n$.

Теорема 2.1. *Нехай $GM_{n,m,k}$ — узагальнене число Мерсенна виду $GM_{n,m,k} = 2^n - 2^m - 1 - k$, де $n, m, k \in \mathbb{N}$, $n > m$ і $k < 2^n - 2^m - 1$, та нехай є два n-бітових числа A, B таких, що $A \leqslant GM_{n,m,k}$ та $B \leqslant GM_{n,m,k}$. Тоді виконується таке співвідношення для ваги Хеммінга суми чисел A та B*

за модулем $GM_{n,m,k}$*:*

$$Ham(A + B \bmod GM_{n,m,k}) \leqslant Ham(A) + Ham(B) + k.$$

Доведення. Доведення даної теореми наведено у Додатку А. □

Теорема 2.2 описує співвідношення для ваги Хеммінга добутку двох чисел за модулем узагальненого числа Мерсенна за умови, що одне з цих чисел є степенем двійки.

Теорема 2.2. *Нехай* $GM_{n,m,k}$ *— узагальнене число Мерсенна виду* $GM_{n,m,k} = 2^n - 2^m - 1 - k$*, де* $n, m, k \in \mathbb{N}$*,* $n > m$ *і* $k < 2^n - 2^m - 1$*, та нехай* A *— n-бітове число таке, що* $A \leqslant GM_{n,m,k}$*. Тоді виконуються такі співвідношення для ваги Хеммінга:*

1) $Ham(2 \cdot A \bmod GM_{n,m,k}) \leqslant Ham(A) + k$*;*

2) $Ham(2^r \cdot A \bmod GM_{n,m,k}) \leqslant$

$$\leqslant Ham(A) + k \cdot \min(Ham(A), r) + \mathbb{1}(r \geqslant n - m) \cdot (r - n + m),$$

де $r \in \mathbb{N}$*,* $\mathbb{1}(A)$ *— індикатор події* A*, тобто* $\mathbb{1}(A) = 1$*, якщо подія* A *виконується, і* $\mathbb{1}(A) = 0$*, якщо ні.*

Доведення. Доведення даної теореми наведено у Додатку А. □

У теоремі 2.3 описано співвідношення для ваги Хеммінга добутку двох чисел за модулем числа $GM_{n,m,k}$.

Теорема 2.3. *Нехай* $GM_{n,m,k}$ *— узагальнене число Мерсенна виду* $GM_{n,m,k} = 2^n - 2^m - 1 - k$*, де* $n, m, k \in \mathbb{N}$*,* $n > m$ *і* $k < 2^n - 2^m - 1$*, та нехай є два n-бітових числа* A, B *таких, що* $A \leqslant GM_{n,m,k}$ *та* $B \leqslant GM_{n,m,k}$*. Тоді виконується таке співвідношення для ваги Хеммінга добутку двох чисел за модулем* $GM_{n,m,k}$*:*

$$Ham(A \cdot B \bmod GM_{n,m,k}) \leqslant$$

$$\leqslant (k+1) \cdot Ham(A) \cdot Ham(B) + \min(Ham(A), Ham(B)) + \frac{m(m-1)}{2}.$$

Доведення. Доведення даної теореми представлено у додатку А. □

У теоремах 2.1 та 2.3 описані співвідношення для ваги Хеммінга суми та добутку чисел за модулем узагальненого числа Мерсенна $GM_{n,m,k}$, які є узагальненням чисел $GM_{n,m}$ та $CR_{n,c}$. Змінюючи значення параметра k, отримаємо співвідношення ваги Хеммінга чисел за модулем $GM_{n,m}$ та $CR_{n,c}$.

Наслідок 2.1. *Якщо у числі $GM_{n,m,k}$ параметр k дорівнює 0, то отримаємо числа $GM_{n,m}$. Таким чином, маємо такі співвідношення для ваги Хеммінга суми та добутку чисел за модулем числа $GM_{n,m}$:*

1) $Ham(A + B \bmod GM_{n,m}) \leqslant Ham(A) + Ham(B)$;

2) $Ham(A \cdot B \bmod GM_{n,m}) \leqslant$

$$\leqslant Ham(A) \cdot Ham(B) + \min(Ham(A), Ham(B)) + \frac{m(m-1)}{2}.$$

Наслідок 2.2. *Якщо у числі Кренделла $CR_{n,c} = 2^n - c$ обирати константу c як $2^m + 1 + k$, де $m, k \in \mathbb{N}$, $m < n$ та $k < 2^n - 2^m - 1$, то виконуються такі співвідношення для ваги Хеммінга суми та добутку чисел за модулем числа Кренделла визначеного виду:*

1) $Ham(A + B \bmod GM_{n,m}) \leqslant Ham(A) + Ham(B) + c - 2^m - 1$;

2) $Ham(A \cdot B \bmod GM_{n,m}) \leqslant$

$$\leqslant (c - 2^m) \cdot Ham(A) \cdot Ham(B) + \min(Ham(A), Ham(B)) + \frac{m(m-1)}{2}.$$

У теоремі 2.4 визначається співвідношення для ваги Хеммінга оберненого елемента відносно операції додавання за модулем узагальненого числа Мерсенна $GM_{n,m} = 2^n - 2^m - 1$, де $n, m \in \mathbb{N}$, $m < n$.

Теорема 2.4. *Для узагальненого числа Мерсенна $GM_{n,m} = 2^n - 2^m - 1$, де $n, m \in \mathbb{N}$, $m < n$, та n-бітового числа $A = a_{n-1}\ a_{n-2} \ldots a_1\ a_0$, де $a_i \in \{0, 1\}$, $i = \overline{0, n-1}$, такого, що $A \leqslant GM_{n,m}$, виконується:*

1) якщо $a_m = 0$, то

$$Ham(-A \bmod GM_{n,m}) = n - 1 - Ham(A);$$

2) якщо $a_m = 1$*, то*

$$Ham(-A \bmod GM_{n,m}) = l - Ham(h_1) + Ham(h_2) + m - Ham(h_3),$$

де $A = h_1 \,||\, h_2 \,||\, h_3$*, причому:*

а) $|h_3| = m$*, тобто* h_3 *включає молодші* m *бітів числа* A*:* $h_3 = a_{m-1}\, a_{m-2} \ldots a_1\, a_0$*;*

б) $h_2 = a_k\, a_{k-1} \ldots a_m$*, де* $k = \min\limits_i\{a_i = 0 \mid a_j = 1, m \leqslant j < i\}$*, тобто* h_2 *містить біти починаючи з* a_m *та до першого нуля, який зустрінеться після* a_m*, включно;*

в) $h_1 = a_{n-1}\, a_{n-2} \ldots a_k$*, де* k *— індекс з минулого пункту;* l *— довжина числа* h_1*, тобто* $|h_1| = l$*.*

Доведення. Доведення даної теореми наведено у додатку А. □

Теорема 2.5 описує співвідношення для обчислення ваги Хеммінга оберненого елемента відносно операції додавання за модулем числа Кренделла $CR_{n,c}$.

Теорема 2.5. *Нехай задані число Кренделла* $CR_{n,c} = 2^n - c$*, де* $n, c \in \mathbb{N}$*, та* n*-бітове число* A*. Визначимо такі позначення:*

$$A = a_{n-1}\, a_{n-2} \ldots a_1\, a_0;$$

$$CR_{n,c} = r_{n-1}\, r_{n-2} \ldots r_1\, r_0;$$

$$B = -A \bmod CR_{n,c} = b_{n-1}\, b_{n-2} \ldots b_1\, b_0,$$

де $a_i, r_i, b_i \in \{0, 1\}$*,* $i = \overline{0, n-1}$*. Тоді вага Хеммінга оберненого до* A *елемента відносно операції додавання за модулем* $CR_{n,c}$ *(тобто вага Хеммінга числа* B*) обчислюється таким чином:*

1) якщо число $a_{\lceil \log_2 c \rceil - 1} \ldots a_1\, a_0$ *менше за* $r_{\lceil \log_2 c \rceil - 1} \ldots r_1\, r_0$*, то:*

$$Ham(-A \bmod CR_{n,c}) = n - \lceil \log_2 c \rceil - Ham(h_1) + Ham(h_2^*),\ \text{де}$$

а) $A = h_1 \,||\, h_2$*, причому* h_2 *— молодші* $\lceil \log_2 c \rceil$ *бітів числа* A*;*

б) $B = -A \bmod CR_{n,c} = h_1^* \,||\, h_2^*$, *аналогічно,* h_2^* *складається з* $\lceil \log_2 c \rceil$ *молодших бітів числа* B;

2) якщо ж число $a_{\lceil \log_2 c \rceil - 1} \dots a_1 \; a_0$ *є більшим за число* $r_{\lceil \log_2 c \rceil - 1} \dots r_1 \, r_0$, *то:*

$Ham(-A \bmod CR_{n,c}) = n - \lceil \log_2 c \rceil - |h_2| - Ham(h_1) + Ham(h_2) + Ham(h_3^*)$, *де*

а) $A = h_1 \,||\, h_2 \,||\, h_3$, *причому:*

– h_3 *— молодші* $\lceil \log_2 c \rceil$ *бітів числа* A;

– h_2 *містить біти числа* A *починаючи з* $\lceil \log_2 c \rceil$ *біту та до першого нуля, який зустрінеться після* $a_{\lceil \log_2 c \rceil - 1}$;

– $|h_2|$ *— кількість бітів у числі* h_2;

– h_1 *— старші біти, що залишились, тобто* $h_1 = a_{n-1} \, a_{n-2} \dots a_w$, *де* a_{w-1} *— старший біт* h_2;

б) $B = -A \bmod CR_{n,c} = h_1^* \,||\, h_2^* \,||\, h_3^*$, *де* h_3^* *— молодші* $\lceil \log_2 c \rceil$ *бітів числа* B.

Доведення. Доведення даної теореми представлено у Додатку А. □

Недоліком співвідношень, що представлені у теоремі 2.5, є те, що для обчислення ваги Хеммінга оберненого елемента відносно операції додавання за модулем числа Кренделла необхідно знайти частину бітів його значення, а саме $\lceil \log_2 c \rceil$ молодших бітів. Тому модифікуємо співвідношення теореми 2.5 таким чином, щоб отримати оцінки ваги Хеммінга оберненого елемента відносно операції додавання за модулем числа Кренделла, які б залежали лише від ваги Хеммінга початкового числа. Отриманий результат представлено у теоремі 2.6.

Теорема 2.6. *Нехай задано число Кренделла* $CR_{n,c} = 2^n - c$, *де* $n, c \in \mathbb{N}$, *та* n*-бітове число* A *таке, що* $A \leqslant CR_{n,c}$. *Введено такі позначення:*

$$A = a_{n-1} \, a_{n-2} \dots a_1 \, a_0;$$

$$CR_{n,c} = r_{n-1} \, r_{n-2} \dots r_1 \, r_0,$$

де $a_i, r_i \in \{0,1\}$, $i = \overline{0, n-1}$. *Тоді для ваги Хеммінга оберненого числа до* A *відносно операції додавання виконуються такі співвідношення:*

1) якщо число $a_{\lceil \log_2 c \rceil - 1} \dots a_1\ a_0$ *менше за* $r_{\lceil \log_2 c \rceil - 1} \dots r_1\ r_0$, *то:*

$$n - \lceil \log_2 c \rceil - Ham(h_1) \leqslant Ham(-A \bmod CR_{n,c}) \leqslant n - Ham(h_1),$$

де $A = h_1 \,||\, h_2$, *причому* h_2 *— молодші* $\lceil \log_2 c \rceil$ *бітів числа* A.

2) якщо ж число $a_{\lceil \log_2 c \rceil - 1} \dots a_1\ \ a_0$ *є більшим за число* $r_{\lceil \log_2 c \rceil - 1} \dots r_1\ r_0$, *то:*

$$|h_1| - Ham(h_1) + Ham(h_2) \leqslant Ham(-A \bmod CR_{n,c}) \leqslant$$
$$\leqslant n - |h_2| - Ham(h_1) + Ham(h_2),$$

де $A = h_1 \,||\, h_2 \,||\, h_3$, *причому:*

– h_3 *— молодші* $\lceil \log_2 c \rceil$ *бітів числа* A;

– h_2 *містить біти числа* A *починаючи з* $\lceil \log_2 c \rceil$ *біту та до першого нуля, який зустрінеться після* $a_{\lceil \log_2 c \rceil - 1}$;

– $|h_2|$ *— кількість бітів у числі* h_2;

– h_1 *— старші біти, що залишились, тобто* $h_1 = a_{n-1}\ a_{n-2} \dots a_w$, *де* a_{w-1} *— старший біт* h_2.

Доведення. Нерівності отримуються очевидно з теореми 2.5, враховуючи, що h_2^* з пункту 1 матиме мінімальну вагу Хеммінга 0 у випадку, коли $h_2^* = 00 \dots 0$, та максимальну вагу Хеммінга, що дорівнює $\lceil \log_2 c \rceil$, у випадку $h_2^* = 11 \dots 1$. Аналогічні оцінки виконуються для мінімальної та максимальної ваги Хеммінга числа h_3^* з пункту 2. □

Застосовуючи отримані співвідношення для ваги Хеммінга суми та добутку двох чисел за модулем узагальненого числа Мерсенна, а також ваги Хеммінга оберненого елементу відносно операції додавання за модулем узагальненого числа Мерсенна, можна побудувати криптопримітиви на основі AJPS-1 або AJPS-2 з використанням арифметики за модулем узагальненого числа Мерсенна.

2.4 Побудова модифікацій криптосистеми AJPS-1 шляхом використання інших класів чисел

Запропоновані криптопримітиви є модифікаціями криптосистеми AJPS-1, яка використовує арифметику за модулем числа Мерсенна. При побудові криптопримітивів процедури шифрування, розшифрування та генерації ключів криптосистеми AJPS-1 були видозмінені так, щоб було можливим застосування інших класів модулів, окрім класу чисел Мерсенна.

Модифікація криптосистеми AJPS-1 з використанням арифметики за модулем узагальненого числа Мерсенна $GM_{n,m}$

Побудована модифікація, як і криптосистема AJPS-1, дозволяє зашифрувати один біт повідомлення, тобто відкритим текстом є число $b \in \{0,1\}$. Відкритими параметрами криптосистеми є узагальнене число Мерсенна $GM_{n,m} = 2^n - 2^m - 1$, де $n, m \in \mathbb{N}$ та $n > m$, параметр захищеності λ, який задається при побудові криптосистеми, і число h, яке задовольняє умовам $C_n^h \geqslant 2^\lambda$ та $4h^2 < n \leqslant 16h^2$.

Позначимо $HG_{n,m,h}$ множину лишків за модулем узагальненого числа Мерсенна $GM_{n,m}$, які мають вагу Хеммінга h, тобто

$$HG_{n,m,h} = \{x \in \{0,1\}^n : \ x < GM_{n,m}, \ Ham(x) = h\}.$$

Розглянемо алгоритми генерації ключів, шифрування та розшифрування модифікації криптосистеми AJPS-1 з використанням арифметики за модулем числа $GM_{n,m}$.

1) Генерація ключів відбувається аналогічно до алгоритму ***Gen*** криптосистеми AJPS-1, застосовуючи клас узагальнених чисел Мерсенна в ролі модуля.

а) Числа F та G обираються випадково та незалежно з множини $HG_{n,m,h}$.

б) Особистим ключем є число G, а відкритим ключем — число H, що обчислюється за таким співвідношенням

$$H = F \cdot G^{-1} \bmod GM_{n,m}.$$

2) Алгоритм ***Enc*** даної модифікації полягає у наступному.

а) З множини $HG_{n,m,h}$ незалежно та рівноймовірно обираються два числа A та B.

б) Виконується перевірка таких двох умов:

$$Ham(A + B \bmod GM_{n,m}) \geqslant |Ham(A) - Ham(B)|;$$

$$Ham(A \cdot B \bmod GM_{n,m}) \geqslant |Ham(A) - Ham(B)|.$$

Якщо хоча б одна з умов не виконується, то потрібно повернутись на крок а) та обрати значення A та B ще раз.

Експериментально було встановлено, що при значеннях чисел n та h, які визначені в таблиці 1.1, ймовірність того, що випадково обрані числа A та B з множини $HG_{n,m,h}$ будуть задовольняти наведеним умовам, дорівнює 0.988, отже, необхідні умови для коректної роботи криптосистеми не значно обмежують вибір параметрів A та B.

в) Біт b зашифровується за формулою:

$$C = A \cdot H + (-1)^b \cdot B \bmod GM_{n,m}.$$

г) Шифротекст C передається відкритим каналом зв'язку.

3) Алгоритм ***Dec*** розшифрування шифротексту C складається з наступних кроків.

а) Обчислюється значення d за таким співвідношенням:

$$d = Ham(C \cdot G \bmod GM_{n,n}).$$

б) Біт початкового повідомлення b визначається відповідно до

значення d за таким співвідношенням:

$$b = \begin{cases} 0, & \text{якщо } d \leqslant 2h^2 + 2h + m(m-1); \\ 1, & \text{якщо виконується } \textit{умова } 1; \\ \perp, & \text{інакше (випадок помилки розшифрування)}, \end{cases}$$

де *умові 1* відповідає виконання співвідношень:

$$d \geqslant \begin{cases} n - 2h - 1, & \text{при } g_m = 0; \\ |h_1| - Ham(h_1) + Ham(h_2) + m - Ham(h_3) - h, & \text{при } g_m = 1. \end{cases}$$

При цьому особистий ключ G є n-бітовим числом виду

$$G = g_{n-1}\ g_{n-2} \dots g_1\ g_0,$$

де g_m — m-ий біт числа G, та $G = h_1 \,||\, h_2 \,||\, h_3$, причому:

– $|h_3| = m$, тобто $h_3 = g_{m-1}\ g_{m-2} \dots g_1\ g_0$, отже, h_3 включає молодші m бітів числа G;

– $h_2 = g_k\ g_{k-1} \dots g_m$, де $k = \min\limits_i \{g_i = 0 \mid g_j = 1, m \leqslant j < i\}$, тобто h_2 містить біти починаючи з g_m та до першого нуля, який зустрінеться після g_m, включно;

– $h_1 = g_{n-1}\ g_{n-2} \dots g_k$, де k — індекс з минулого пункту.

Коректність розшифрування слідує з теореми 2.4 та наслідку 2.1 з теорем 2.1 і 2.3.

Стійкість описаної модифікації криптосистеми AJPS-1 базується на складності задачі GMLHRSP (*Задача ділення чисел з малою вагою Хеммінга за модулем узагальненого числа Мерсенна*, англ. *Generalized Mersenne Low Hamming Ratio Search Problem*).

Означення 2.2 (*Задача* GMLHRSP)**.** Маючи узагальнене число Мерсенна $GM_{n,m} = 2^n - 2^m - 1$, де $m \in \mathbb{N}, m < n$, n-бітове число H і ціле число h, знайти два n-бітових числа F та G, кожне з яких має вагу

Хеммінга h, таких, що

$$H = F \cdot G^{-1} \bmod GM_{n,m}.$$

Розглянемо інше узагальнення криптосистеми AJPS-1, а саме модифікацію криптосистеми AJPS-1 з використанням числа Кренделла в якості модуля.

Модифікація криптосистеми AJPS-1 з використанням арифметики за модулем числа Кренделла $CR_{n,c}$

Дана модифікація аналогічна попередній розглянутій модифікації з незначними змінами. Відкритим текстом є число $b \in \{0, 1\}$. При побудові криптосистеми задається параметр захищеності λ. Відкритими параметрами криптосистеми є числа $CR_{n,c} = 2^n - c$, де $n \in \mathbb{N}$, $c = 2^m + 1 + k$, причому $m, k \in \mathbb{N}$ і $log_2(2^m + 1 + k) \leqslant \frac{n}{2}$, та число h, яке задовольняє умовам $C_n^h \geqslant 2^\lambda$ та $4h^2 < n \leqslant 16h^2$.

Позначимо $HC_{n,c,h}$ множину лишків за модулем числа Кренделла $CR_{n,c}$, які мають вагу Хеммінга h, тобто

$$HC_{n,c,h} = \{x \in \{0,1\}^n : \ x < CR_{n,c}, \ Ham(x) = h\}.$$

1) Генерація ключів відбувається відповідно до алгоритму ***Gen*** криптосистеми AJPS-1, застосовуючи інший клас чисел в ролі модуля.

а) Числа F, G обираються випадково і незалежно з множини $HC_{n,c,h}$.

б) Особистим ключем є число G, а відкритим ключем — число H, що обчислюється за таким співвідношенням:

$$H = F \cdot G^{-1} \bmod CR_{n,c}.$$

2) Алгоритм ***Enc*** даної модифікації полягає у послідовному виконанні таких кроків.

а) З множини $HC_{n,c,h}$ незалежно та рівноймовірно обираються два

числа A та B.

б) Виконується перевірка таких двох умов:

$$Ham(A + B \bmod CR_{n,c}) \geqslant |Ham(A) - Ham(B)|;$$

$$Ham(A \cdot B \bmod CR_{n,c}) \geqslant |Ham(A) - Ham(B)|.$$

Якщо хоча б одна з наведених умов не виконується, то потрібно повернутись на крок а) та почати процедуру шифрування спочатку.

Експериментально було встановлено, що при визначених у таблиці 1.1 значеннях параметрів n та h, ймовірність того, що випадково обрані з множини $HC_{n,c,h}$ числа A та B будуть задовольняти наведеним умовам, дорівнює 0.99, отже, необхідні умови для коректної роботи криптосистеми не значно обмежують вибір параметрів A та B.

в) Шифротекст C, що відповідає біту b, обчислюється так:

$$C = A \cdot H + (-1)^b \cdot B \bmod CR_{n,c}.$$

г) Шифротекст C передається відкритим каналом зв'язку.

3) Алгоритм $\boldsymbol{Dec}$ розшифрування шифротексту C полягає у послідовному виконанні таких кроків.

а) Обчислюється значення d за такою формулою:

$$d = Ham(C \cdot G \bmod CR_{n,c}).$$

б) Біт b визначається відповідно до значення d таким чином:

$$b = \begin{cases} 0, & \text{якщо } d \leqslant (c - 2^m) \cdot (2h^2 + 1) + 2h + m(m-1) - 1; \\ 1, & \text{якщо виконується } \textit{умова } 2; \\ \perp, & \text{інакше (випадок помилки розшифрування)}, \end{cases}$$

де *умові 2* відповідає виконання співвідношень:

$$d \geqslant \begin{cases} n - \lceil \log_2 c \rceil - Ham(x_1) - h, & \text{у випадку, якщо } g_{\lceil \log_2 c \rceil - 1} \dots g_1\ g_0 \\ & \text{менше за число } r_{\lceil \log_2 c \rceil - 1} \dots r_1\ r_0; \\ |y_1| - Ham(y_1) + Ham(y_2) - h, & \text{у випадку, якщо } g_{\lceil \log_2 c \rceil - 1} \dots g_1\ g_0 \\ & \text{є більшим за число } r_{\lceil \log_2 c \rceil - 1} \dots r_1\ r_0. \end{cases}$$

Причому:

– число Кренделла $CR_{n,c}$ у двійковому представленні має вигляд $CR_{n,c} = r_{n-1} \dots r_1\ r_0$, де $r_i \in \{0, 1\}$, $i = \overline{0, n-1}$;

– особистий ключ G є n-бітовим числом та представляється $G = g_{n-1} \dots g_1\ g_0$ у двійковому записі, де $g_i \in \{0, 1\}$, $i = \overline{0, n-1}$; відповідно g_m — m-ий біт числа G;

– число G представляється у вигляді:

i. $G = x_1 \mid\mid x_2$, де x_2 — молодші $\lceil \log_2 c \rceil$ бітів числа G.

ii. $G = y_1 \mid\mid y_2 \mid\mid y_3$, де

– y_3 — молодші $\lceil \log_2 c \rceil$ бітів числа G;

– y_2 містить біти числа G починаючи з $\lceil \log_2 c \rceil$ біту та до першого нуля, який зустрінеться після $g_{\lceil \log_2 c \rceil - 1}$;

– y_1 — старші біти, що залишились, тобто $y_1 = g_{n-1}\ g_{n-2} \dots g_w$, де g_{w-1} — старший біт y_2.

Обґрунтування коректності побудованого криптопримітиву базується на теоремі 2.6 та наслідку 2.2 з теорем 2.1 та 2.3.

Стійкість розглянутої модифікації криптосистеми AJPS-1 ґрунтується на *Задачі ділення чисел з малою вагою Хеммінга за модулем числа Кренделла* (англ. *Crandall Low Hamming Ratio Search Problem*), що формулюється таким чином.

Означення 2.3 (*Задача* CRLHRSP)**.** Маючи число Кренделла $CR_{n,c} = 2^n - c$, де $c \in \mathbb{N}$ та $\log_2 c \leqslant \frac{n}{2}$, n-бітове число H і ціле число h,

знайти два n-бітових числа F та G, кожне ваги Хеммінга h, таких, що

$$H = F \cdot G^{-1} \bmod CR_{n,c}.$$

Перевагою побудованих модифікацій є значне збільшення класу чисел, які можуть бути використані в якості модуля. Окрім цього, перевагою модифікацій криптосистеми AJPS-1 є збільшення множини значень параметра d, зокрема кількості унікальних значень, які приймає параметр d. Обґрунтуванням цього є експериментальні результати, які описані у таблиці 2.2 та на рисунках 2.5, 2.6, 2.7 та 2.8.

Для отримання цих результатів виконано серію з 1000000 застосувань алгоритмів шифрування та розшифрування криптосистеми AJPS-1 та її модифікацій при фіксованих значеннях ключів. Відкритий та особистий ключі отримані внаслідок застосування алгоритмів генерації ключів криптосистеми AJPS-1 та її модифікацій відповідно. При застосуванні алгоритмів генерації ключів, шифрування та розшифрування модифікацій криптосистеми AJPS-1 з використанням арифметики за модулем узагальненого числа Мерсенна та числа Кренделла використовувались параметри $c = 15$ для числа Кренделла та $m = 25$ для узагальненого числа Мерсенна.

Таблиця 2.2 – Довжини інтервалів та кількість унікальних значень з даних інтервалів, які приймає значення d криптосистеми AJPS-1 та модифікацій криптосистеми AJPS-1 з використанням узагальнених чисел Мерсенна та чисел Кренделла

n	h	Клас чисел	Кількість унікальних значень d при $b = 0$	Довжина інтервалу значень d при $b = 0$	Кількість унікальних значень d при $b = 1$	Довжина інтервалу значень d при $b = 1$
1279	17	Мерсенна	103	105	104	112
		Кренделла	185	213	208	225
		Узагальнені Мерсенна	150	173	224	256
2203	23	Мерсенна	139	147	139	141
		Кренделла	243	252	268	281
		Узагальнені Мерсенна	208	233	280	295
3217	28	Мерсенна	166	171	162	170
		Кренделла	278	299	323	356
		Узагальнені Мерсенна	256	279	325	367
4253	32	Мерсенна	192	201	190	204
		Кренделла	319	367	374	410
		Узагальнені Мерсенна	299	331	386	415
9689	49	Мерсенна	285	294	284	319
		Кренделла	535	619	540	571
		Узагальнені Мерсенна	457	512	553	668

Зауваження. Довжина інтервалу обчислювалась як різниця максимального та мінімального значень d серед отриманих результатів. Під кількістю унікальних значень мається на увазі кількість унікальних значень серед отриманих 1000000 значень d.

Таким чином, кількість різних значень d в обох модифікаціях збільшилась в середньому в два рази у порівнянні з криптосистемою AJPS-1. Розглянемо вигляд розподілу значення d для різних криптосистем далі.

На рисунках 2.5 та 2.6 зображено розподіл значення d при серії з 1000000 застосувань алгоритмів шифрування та розшифрування при фіксованих значеннях ключів криптосистеми AJPS-1 та модифікації криптосистеми AJPS-1 з використанням арифметики за модулем узагальненого числа Мерсенна $GM_{n,m}$ при значеннях біту $b = 0$ та $b = 1$ відповідно при параметрах $n = 9689$ та $h = 49$.

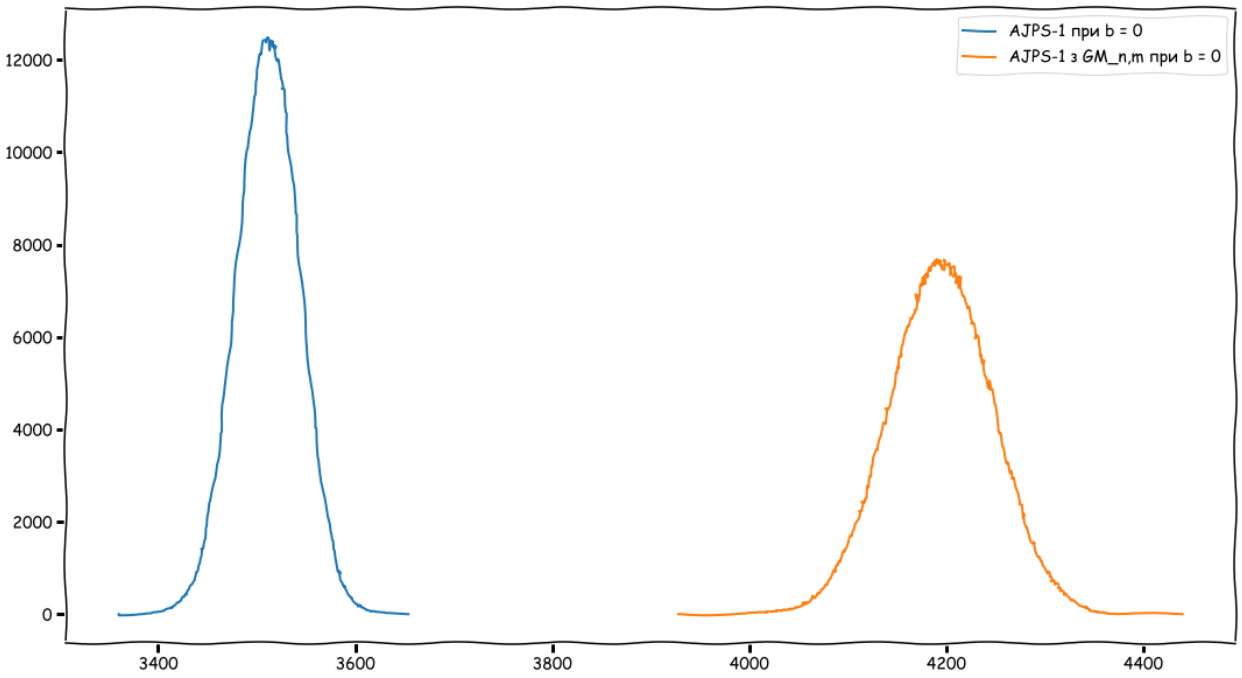

Рисунок 2.5 – Розподіл значення d в криптосистемі AJPS-1 та модифікації AJPS-1 з використанням арифметики за модулем числа Мерсенна $GM_{n,m}$ при значеннях $n = 9689$, $h = 49$, $m = 25$ та $b = 0$

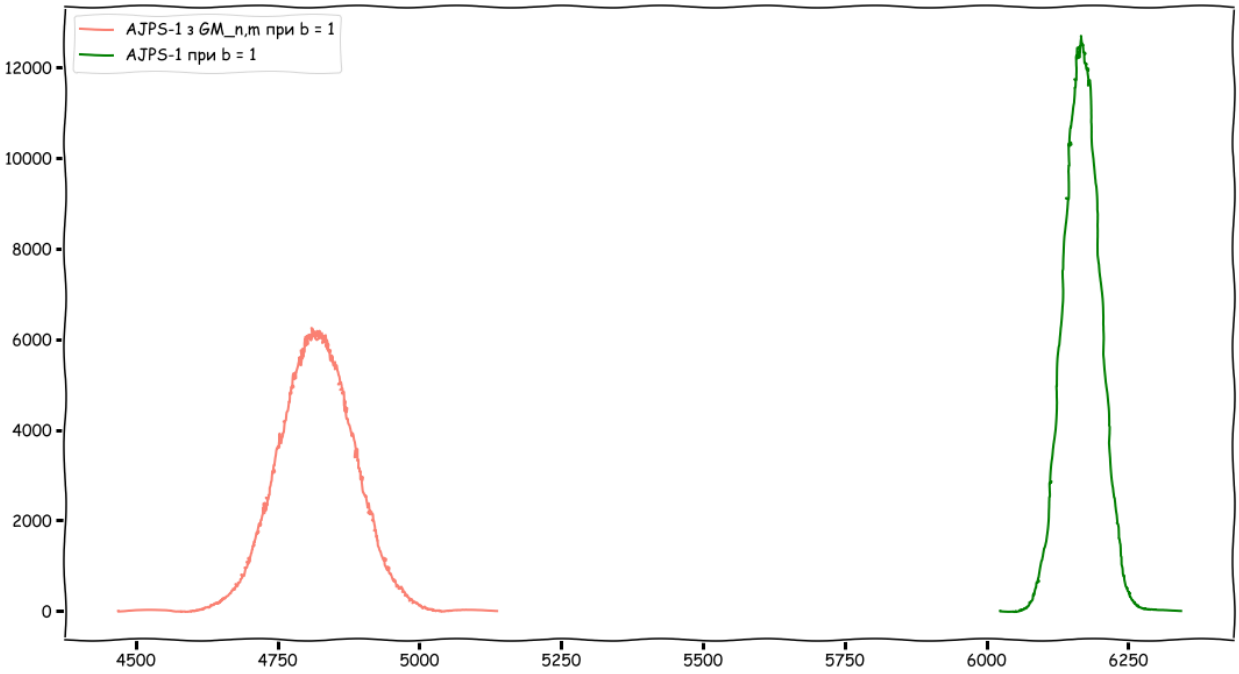

Рисунок 2.6 – Розподіл значення d в криптосистемі AJPS-1 та модифікації AJPS-1 з використанням арифметики за модулем числа Мерсенна $GM_{n,m}$ при значеннях $n = 9689$, $h = 49$, $m = 25$ та $b = 1$

Зауваження. Графіки розподілу значення d криптосистеми AJPS-1 та модифікації AJPS-1 з використанням арифметики за модулем узагальненого числа Мерсенна для інших рекомендованих відповідно до таблиці 2.5 значень параметрів n та h, наведені у додатку Б.2.

Розглянемо модифікацію криптосистеми AJPS-1 з використанням арифметики за модулем числа Кренделла. На рисунках 2.7 та 2.8 зображено розподіл значення d при серії з 1000000 застосувань алгоритмів шифрування та розшифрування при фіксованих значеннях ключів криптосистеми AJPS-1 та модифікації криптосистеми AJPS-1 з використанням арифметики за модулем числа Кренделла при параметрах $n = 9689$ та $h = 49$ та значеннях біту $b = 0$ та $b = 1$ відповідно.

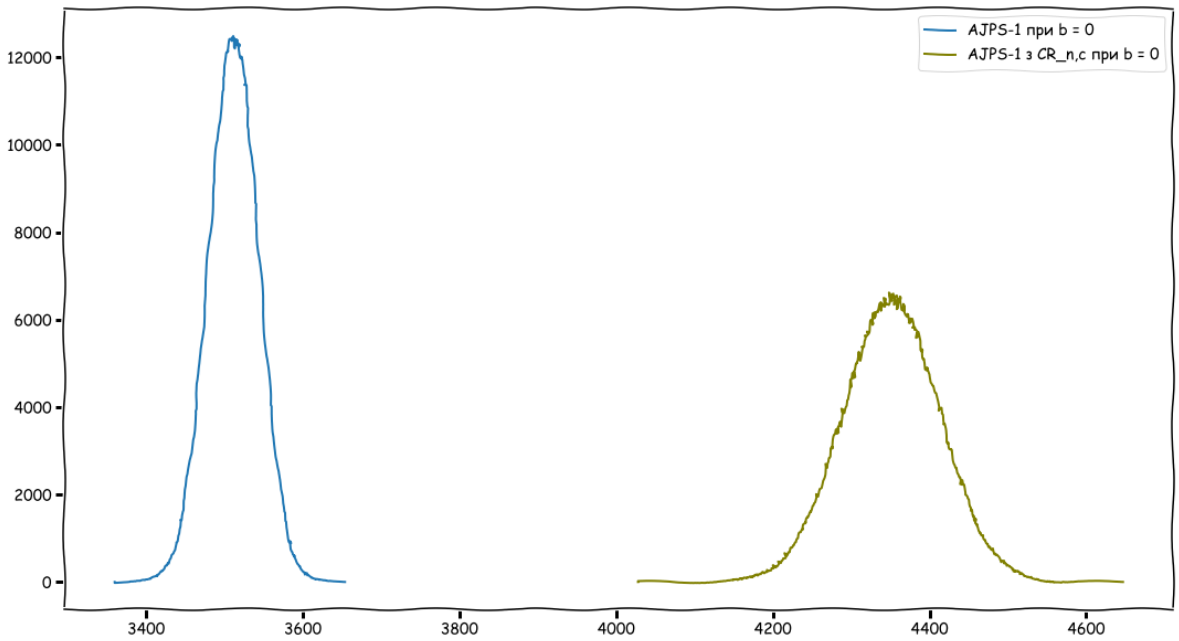

Рисунок 2.7 – Розподіл значення d в криптосистемі AJPS-1 та модифікації AJPS-1 з використанням арифметики за модулем числа Кренделла $CR_{n,c}$ при значеннях $n = 9689$, $h = 49$, $c = 15$ та $b = 0$

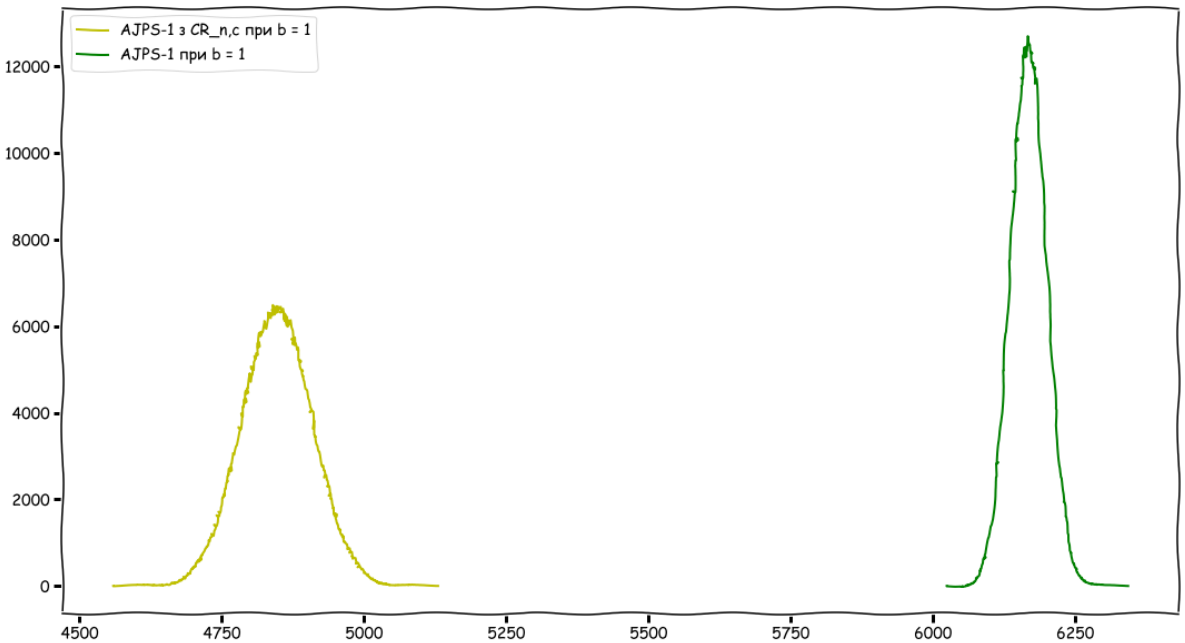

Рисунок 2.8 – Розподіл значення d в криптосистемі AJPS-1 та модифікації AJPS-1 з використанням арифметики за модулем числа Кренделла $CR_{n,c}$ при значеннях $n = 9689$, $h = 49$, $c = 15$ та $b = 1$

Зауваження. Графіки розподілу значення d криптосистеми AJPS-1 та модифікації AJPS-1 з використанням арифметики за модулем числа Кренделла для інших рекомендованих відповідно до таблиці 2.5 значень параметрів n та h, наведені у додатку Б.3.

З рисунків 2.5 та 2.6 для аналізу модифікації AJPS-1 з використанням арифметики за модулем узагальненого числа Мерсенна $GM_{n,m}$ та з рисунків 2.7 та 2.8 для аналізу модифікації з використанням арифметики за модулем числа Кренделла $CR_{n,c}$ видно, що через збільшення кількості унікальних значень d в середньому вдвічі зменшилась частота виникнення пікових значень d в обох модифікаціях у порівнянні з криптосистемою AJPS-1.

Таким чином, такі модифікації дозволяють підвищити стійкість криптосистеми AJPS-1 до атак на основі відкритих шифротекстів, які направлені на визначення особистого ключа.

Для здійснення порівняльного аналізу двох побудованих модифікацій між собою розглянемо порівняння розподілів значення d при серії з 1000000 застосувань алгоритмів шифрування та розшифрування при фіксованих значеннях ключів модифікації AJPS-1 з використанням арифметики за модулем узагальненого числа Мерсенна та модифікації AJPS-1 з використанням арифметики за модулем числа Кренделла.

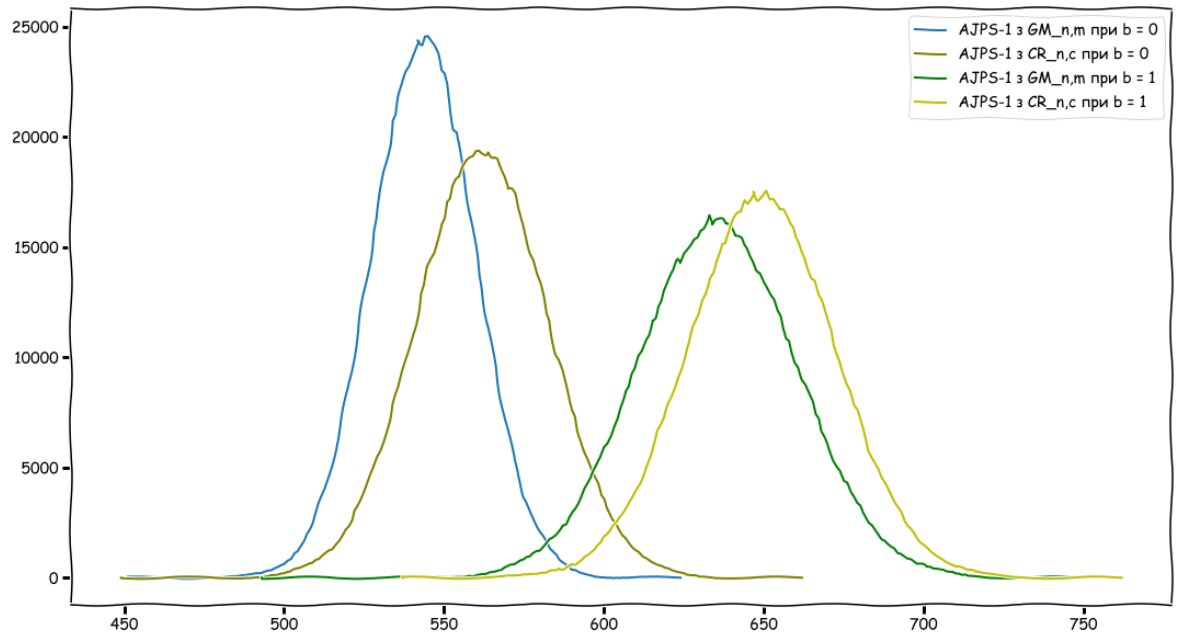

Рисунок 2.9 – Розподіл значення d модифікації AJPS-1 з використанням арифметики за модулем $GM_{n,m}$ та модифікації AJPS-1 з використанням арифметики за модулем числа Кренделла $CR_{n,c}$ при значеннях $n = 1279$, $h = 17$, $m = 25$ та $c = 15$

Зауваження. Графіки розподілу значення d модифікації AJPS-1 з використанням арифметики за модулем узагальненого числа Мерсенна та модифікації AJPS-1 з використанням арифметики за модулем числа Кренделла для інших рекомендованих значень параметрів n та t, які представлено у таблиці 2.5, наведені у додатку Б.4.

Як бачимо з рисунка 2.9, побудовані модифікації не значно відрізняються між собою за кількістю унікальних значень d та довжиною інтервалу, якому належить значення d. Однак, все ж таки модифікація криптосистеми AJPS-1 з використанням арифметики за модулем числа Кренделла є більш ефективною з погляду на дані показники. Недоліком модифікації AJPS-1 з використанням арифметики за модулем узагальненого числа Мерсенна є незбалансованість розподілів значення d для різних бітів, що може бути використано зі зловмисною метою для розрізнення двох випадків (при значенні біту 0 та при значенні біту 1),

порівнюючи частоту виникнення певних значень d. Дана вразливість може бути застосована для здійснення атаки за відомими шифротекстами для визначення особистого ключа.

2.5 Побудова модифікацій криптосистеми AJPS-2 шляхом використання інших класів чисел

Розглянемо модифікації криптосистеми AJPS-2 з використанням арифметики за модулем узагальненого числа Мерсенна $GM_{n,m} = 2^n - 2^m - 1$, де $n, m \in \mathbb{N}$, $n > m$, та з використанням арифметики за модулем числа Кренделла $CR_{n,c} = 2^n - c$, де $n, c \in \mathbb{N}$ та $\log_2 c \leqslant \frac{n}{2}$.

Алгоритми генерації ключів, шифрування та розшифрування у даних модифікаціях такі ж, як в криптосистемі AJPS-2 за виключенням того, що усі операції виконуються за модулем узагальненого числа Мерсенна або числа Кренделла. Аналогічно модифікаціям криптосистеми AJPS-1, параметри F та G алгоритму генерації ключів $\boldsymbol{Gen}$ та параметри A, B_1 і B_2 алгоритму шифрування $\boldsymbol{Enc}$ обираються не з множини $HM_{n,h}$, а з множин $HG_{n,m,h}$ та $HC_{n,c,h}$ для використання арифметики за модулем узагальненого числа Мерсенна $GM_{n,m}$ та за модулем числа Кренделла $CR_{n,c}$ відповідно.

Окрім цього, для побудови модифікацій криптосистеми AJPS-2, які б використовували арифметику за модулем узагальнених чисел Мерсенна, необхідно визначити умови на характеристики коду корекції помилок, який використовується у алгоритмах шифрування та розшифрування AJPS-2. При визначенні необхідних умов на параметри коду корекції помилок використовується наступна теорема.

Теорема 2.7 ([4]). *Нехай X — довільне n-бітове число. Тоді для будь-якого числа n-бітового числа Y, що має вагу Хеммінга k, та будь-*

якого значення ε, $\varepsilon > 0$ *виконується:*

$$Pr[Ham_{dist}(X, X+Y) \geqslant 2k(1+\varepsilon)] \leqslant 2^{-2k(\varepsilon - \ln(1+\varepsilon))}.$$

Код корекції помилок $(\mathcal{E}, \mathcal{D})$ криптосистеми AJPS-2 обирається відповідно до співвідношень, що описані у наступному твердженні.

Твердження 2.4. *Криптосистема* AJPS-2 *є* $(1-\delta)$*-коректною, якщо код корекції помилок* $(\mathcal{E}, \mathcal{D})$ *виправляє до*

$$(4h^2 + 2h)(1+\varepsilon)$$

помилок для деякого значення ε, $0 < \varepsilon < 1$, *що задовольняє такій умові:*

$$2^{-\frac{h^2\varepsilon^2}{3}} \left(1 + 2^{-\frac{h\varepsilon^2}{3}}\right) < \delta.$$

Доведення. У криптосистемі AJPS-2 для розшифрування повідомлення обчислюється

$$\mathcal{D}((C_1 \cdot F \bmod M_n) \oplus C_2),$$

де $(\mathcal{E}, \mathcal{D})$ — код корекції помилок, (C_1, C_2) — пара шифротекстів, F — особистий ключ. При цьому

$$C_1 = A \cdot R + B_1 \bmod M_n;$$

$$C_2 = (A \cdot (F \cdot R + G) + B_2 \bmod M_n) \oplus \mathcal{E}(m).$$

Для коректного розшифрування необхідно, щоб значення $C_1 \cdot F \bmod M_n$ та C_2 мали малу відстань Хеммінга (конкретне значення відстані Хеммінга залежить від визначеного значення ймовірності того, що при розшифруванні буде отримано початкове повідомлення m). Таким чином, потрібно сформулювати співвідношення параметра δ та відстані Хеммінга таких значень

$$C_1 \cdot F = A \cdot F \cdot R + B_1 \cdot F \bmod M_n;$$

$$C_2 = A \cdot F \cdot R + A \cdot G + B_2 \bmod M_n.$$

Для цього необхідно двічі застосувати теорему 2.7.

1) Розглянемо значення $A \cdot F \cdot R$ та $A \cdot F \cdot R + B_1 \cdot F$. Враховуючи, що усі обчислення виконуються за модулем числа Мерсенна, за лемою 1.1 вагу Хеммінга числа $B_1 \cdot F$ можна оцінити таким чином:

$$Ham(B_1 \cdot F \bmod M_n) \leqslant Ham(B_1) \cdot Ham(F) = h^2.$$

Тоді, застосовуючи теорему 2.7, маємо:

$$Pr[Ham_{dist}(A \cdot F \cdot R, A \cdot F \cdot R + B_1 \cdot F) \geqslant 2h^2(1+\varepsilon)] \leqslant 2^{-2h^2(\varepsilon - \ln(1+\varepsilon))}.$$

2) Далі розглянемо значення $A \cdot F \cdot R$ та $A \cdot F \cdot R + A \cdot G + B_2$. Аналогічно, використовуючи лему 1.1, маємо:

$$Ham(A \cdot G + B_2 \bmod M_n) \leqslant Ham(A \cdot G \bmod M_n) + Ham(B_2) \leqslant$$

$$\leqslant Ham(A) \cdot Ham(G) + h = h^2 + h.$$

Тоді, застосовуючи теорему 2.7, маємо:

$$Pr[Ham_{dist}(A \cdot F \cdot R, A \cdot F \cdot R + A \cdot G + B_2) \geqslant 2(h^2+h)(1+\varepsilon)] \leqslant 2^{-2(h^2+h)(\varepsilon - \ln(1+\varepsilon))}.$$

Розглянемо сумісну ймовірність двох розглянутих вище подій, тобто таку ймовірність:

$$Pr[Ham_{dist}(A \cdot F \cdot R, A \cdot F \cdot R + B_1 \cdot F) \geqslant 2h^2(1+\varepsilon),$$

$$Ham_{dist}(A \cdot F \cdot R, A \cdot F \cdot R + A \cdot G + B_2) \geqslant 2(h^2+h)(1+\varepsilon)].$$

Використовуючи нерівність Буля, маємо:

$$Pr[Ham_{dist}(A \cdot F \cdot R, A \cdot F \cdot R + B_1 \cdot F) \geqslant 2h^2(1+\varepsilon),$$

$$Ham_{dist}(A \cdot F \cdot R, A \cdot F \cdot R + A \cdot G + B_2) \geqslant 2(h^2+h)(1+\varepsilon)] \leqslant$$

$$\leqslant Pr[Ham_{dist}(A \cdot F \cdot R, A \cdot F \cdot R + B_1 \cdot F) \geqslant 2h^2(1+\varepsilon)] +$$

$$+Pr[Ham_{dist}(A \cdot F \cdot R, A \cdot F \cdot R + A \cdot G + B_2) \geqslant 2(h^2+h)(1+\varepsilon)] \leqslant$$

$$\leqslant 2^{-2h^2(\varepsilon-\ln(1+\varepsilon))} + 2^{-2(h^2+h)(\varepsilon-\ln(1+\varepsilon))} = 2^{-2h^2(\varepsilon-\ln(1+\varepsilon))}\left(1 + 2^{-2h(\varepsilon-\ln(1+\varepsilon))}\right).$$

Розглянемо значення $\varepsilon - \ln(1+\varepsilon)$. Використовуючи розклад $\ln(1+\varepsilon)$ в ряд Тейлора, отримаємо:

$$\varepsilon - \left(\varepsilon - \frac{\varepsilon^2}{2} + \frac{\varepsilon^3}{6} - \ldots\right) \geqslant \varepsilon - \varepsilon + \frac{\varepsilon^2}{2} - \frac{\varepsilon^3}{6} = \frac{\varepsilon^2}{2} - \frac{\varepsilon^3}{6} = \frac{3 \cdot \varepsilon^2 - 2 \cdot \varepsilon^3}{6}.$$

Оскільки $0 < \varepsilon < 1$, то $\varepsilon^3 \leqslant \varepsilon^2$, отже, маємо:

$$\varepsilon - \ln(1+\varepsilon) \geqslant \frac{3 \cdot \varepsilon^2 - 2 \cdot \varepsilon^2}{6} = \frac{\varepsilon^2}{6}.$$

Відповідно до алгоритму розшифрування AJPS-2, необхідно отримати оцінку для такого значення відстані Хеммінга:

$$Ham_{dist}(A \cdot F \cdot R + B_1 \cdot F, A \cdot F \cdot R + A \cdot G + B_2).$$

Для цього використовуємо нерівність трикутника з метрикою Ham_{dist}. Таким чином, отримаємо співвідношення

$$Ham_{dist}(A \cdot F \cdot R + B_1 \cdot F, A \cdot F \cdot R + A \cdot G + B_2) \geqslant$$

$$\geqslant Ham_{dist}(A \cdot F \cdot R, A \cdot F \cdot R + B_1 \cdot F) + Ham_{dist}(A \cdot F \cdot R, A \cdot F \cdot R + A \cdot G + B_2) \geqslant$$

$$\geqslant 2h^2(1+\varepsilon) + 2(h^2+h)(1+\varepsilon) = (4h^2+2h)(1+\varepsilon).$$

Тоді ймовірність виконання даного співвідношення буде такою:

$$Pr[Ham_{dist}(A \cdot F \cdot R + B_1 \cdot F, A \cdot F \cdot R + A \cdot G + B_2) \geqslant (4h^2+2h)(1+\varepsilon)] \leqslant$$

$$\leqslant 2^{-\frac{h^2\varepsilon^2}{3}}\left(1 + 2^{-\frac{h\varepsilon^2}{3}}\right).$$

Відповідно до означення 1.5, для забезпечення $(1-\delta)$-коректності схеми шифрування необхідно, щоб код корекції помилок $(\mathcal{E}, \mathcal{D})$ виправляв

до $(4h^2 + 2h)(1 + \varepsilon)$ помилок для деякого ε, $0 < \varepsilon < 1$, що задовольняє умові $2^{-\frac{h^2\varepsilon^2}{3}}\left(1 + 2^{-\frac{h\varepsilon^2}{3}}\right) < \delta$. □

Аналогічно, можна визначити умови для коду корекції помилок для модифікацій криптосистеми AJPS-2 з використанням арифметики за модулем узагальнених чисел Мерсенна.

Твердження 2.5. *Модифікація криптосистеми* AJPS-2 *з використанням арифметики за модулем узагальненого числа Мерсенна* $GM_{n,m}$ *є* $(1 - \delta)$*-коректною, якщо код корекції помилок* $(\mathcal{E}, \mathcal{D})$ *виправляє до*

$$(4h^2 + 6h + 2m(m-1))(1 + \varepsilon)$$

помилок для деякого значення ε, $0 < \varepsilon < 1$, *що задовольняє такій умові:*

$$2^{-\left(h^2+h+\frac{m(m-1)}{2}\right)\cdot\frac{\varepsilon^2}{3}}\left(1 + 2^{-\frac{h\varepsilon^2}{3}}\right) < \delta.$$

Доведення. Доведення аналогічне доведенню твердження 2.4. Використовуючи теореми 2.1 та 2.3, маємо:

$$Ham(F \cdot B_1 \bmod GM_{n,m}) \leqslant h^2 + h + \frac{m(m-1)}{2};$$

$$Ham(A \cdot G + B_2 \bmod GM_{n,m}) \leqslant Ham(A \cdot G \bmod GM_{n,m}) + h \leqslant$$

$$\leqslant h^2 + h + \frac{m(m-1)}{2} + h = h^2 + 2h + \frac{m(m-1)}{2}.$$

Відповідно, оцінка зверху сумісної ймовірності така:

$$2^{-2\left(h^2+h+\frac{m(m-1)}{2}\right)\cdot\frac{\varepsilon^2}{6}} + 2^{-2\left(h^2+2h+\frac{m(m-1)}{2}\right)\cdot\frac{\varepsilon^2}{6}} = 2^{-\left(h^2+h+\frac{m(m-1)}{2}\right)\cdot\frac{\varepsilon^2}{3}}\left(1 + 2^{-\frac{h\varepsilon^2}{3}}\right).$$

Застосовуючи нерівність трикутника, маємо:

$$2\left(h^2 + h + \frac{m(m-1)}{2}\right) + 2\left(h^2 + 2h + \frac{m(m-1)}{2}\right) = 4h^2 + 6h + 2m(m-1).$$

□

Стійкість модифікації криптосистеми AJPS-2 з використанням

операцій за модулем $GM_{n,m}$ базується на складності задачі GMLHCSP (*Задача лінійної комбінації чисел з малою вагою Хеммінга за модулем узагальненого числа Мерсенна*, англ. *Generalized Mersenne Low Hamming Combination Search Problem*).

Означення 2.4 (*Задача* GMLHCSP)**.** Маючи узагальнене число Мерсенна $GM_{n,m}$, ціле число h та пару чисел (R, T), де R — випадково обране n-бітове число таке, що $R \leqslant GM_{n,m}$, і число T обчислено відповідно до співвідношення

$$T = F \cdot R + G \bmod GM_{n,m},$$

причому F та G обрані незалежно та випадково з множини $HG_{n,m,h}$, знайти числа F та G.

Також можна визначити умови на вибір коду корекції помилок модифікації криптосистеми AJPS-2 з використанням арифметики за модулем числа Кренделла $CR_{n,c}$. Такі умови описані у наступному твердженні.

Твердження 2.6. *Модифікація криптосистеми* AJPS-2 *з використанням арифметики за модулем числа Кренделла* $CR_{n,c}$ *є* $(1-\delta)$*-коректною, якщо код корекції помилок* $(\mathcal{E}, \mathcal{D})$ *виправляє до*

$$(4(c-2^m)h^2 + 6h + 2m(m-1) + 2(c-2^m-1))(1+\varepsilon)$$

помилок, де $c = 2^m + 1 + k$, $m, k \in \mathbb{N}$, *для деякого значення* ε, $0 < \varepsilon < 1$, *що задовольняє такій умові:*

$$2^{-\left((c-2^m)h^2+h+\frac{m(m-1)}{2}\right)\cdot\frac{\varepsilon^2}{3}}\left(1+2^{-(h+c-2^m-1)\cdot\frac{\varepsilon^2}{3}}\right) < \delta.$$

Доведення. Доведення аналогічне доведенню твердження 2.4. Використовуючи теореми 2.1 та 2.3, маємо:

$$Ham(F \cdot B_1 \bmod CR_{n,c}) \leqslant (c-2^m)h^2 + h + \frac{m(m-1)}{2};$$

$$Ham(A \cdot G + B_2 \bmod CR_{n,c}) \leqslant Ham(A \cdot G \bmod CR_{n,c}) + h + c - 2^m - 1 \leqslant$$

$$\leqslant (c - 2^m)h^2 + h + \frac{m(m-1)}{2} + h + c - 2^m - 1 =$$

$$= (c - 2^m)h^2 + 2h + \frac{m(m-1)}{2} + c - 2^m - 1.$$

Відповідно, оцінка зверху сумісної ймовірності така:

$$2^{-2\left((c-2^m)h^2+h+\frac{m(m-1)}{2}\right)\frac{\varepsilon^2}{6}} + 2^{-2\left((c-2^m)h^2+2h+\frac{m(m-1)}{2}+c-2^m-1\right)\frac{\varepsilon^2}{6}} =$$

$$= 2^{-\left((c-2^m)h^2+h+\frac{m(m-1)}{2}\right)\cdot\frac{\varepsilon^2}{3}} \left(1 + 2^{-(h+c-2^m-1)\cdot\frac{\varepsilon^2}{3}}\right).$$

Застосовуючи нерівність трикутника, маємо:

$$2\left((c - 2^m)h^2 + h + \frac{m(m-1)}{2}\right) + 2\Bigg((c - 2^m)h^2 + 2h + \frac{m(m-1)}{2} +$$

$$+c - 2^m - 1\Bigg) = 4(c - 2^m)h^2 + 6h + 2m(m-1) + 2(c - 2^m - 1).$$

□

Стійкість модифікації криптосистеми AJPS-2, яка використовує операції за модулем числа Кренделла $CR_{n,c}$, ґрунтується на складності задачі CRLHCSP (*Задача лінійної комбінації чисел з малою вагою Хеммінга за модулем числа Кренделла*, англ. *Crandall Low Hamming Combination Search Problem*).

Означення 2.5 (*Задача* CRLHCSP)**.** Маючи число Кренделла $CR_{n,c}$, ціле число h та пару чисел (R, T), де R — випадково обране n-бітове число таке, що $R \leqslant CR_{n,c}$, і число T обчислено відповідно до співвідношення

$$T = F \cdot R + G \bmod CR_{n,c},$$

причому F та G обрані незалежно та випадково з множини $HC_{n,c,h}$, знайти числа F та G.

Таким чином, описано дві модифікації AJPS-2 шляхом застосування інших класів чисел, окрім класу чисел Мерсенна. Перевагою побудованих

модифікацій криптосистеми AJPS-2 з використанням арифметики за модулем узагальненого числа Мерсенна та арифметики за модулем числа Кренделла є значне збільшення класу чисел, що використовуються у ролі модуля у криптосистемі.

Висновки до розділу 2

У даному розділі описано необхідні умови на значення особистого ключа криптосистем AJPS-1 та AJPS-2 та, як наслідок, сформовано рекомендації для алгоритмів генерації ключів даних криптосистем. Доведено властивості арифметики за модулем числа Мерсенна, а саме властивість циклічного зсуву суми двох чисел за модулем числа Мерсенна та співвідношення метрики OSD суми, добутку двох чисел та оберненого числа відносно операції додавання за модулем числа Мерсенна. Використовуючи доведені властивості, побудовано атаку підміни з модифікованим відкритим текстом з використанням моделі активного зловмисника на криптосистему AJPS-2. Побудована атака є більш ефективною, ніж аналогічна побудована раніше атака підміни на криптосистему AJPS-1, адже дозволяє змінювати повідомлення, яке буде отримане після розшифрування. Також у розділі представлено три модифікації криптосистеми AJPS-1, а саме модифікацію AJPS-1 з використанням зміненої метрики OSD, модифікацію AJPS-1 з використанням арифметики за модулем узагальненого числа Мерсенна $GM_{n,m} = 2^n - 2^m - 1$ та модифікацію AJPS-1 з використанням арифметики за модулем числа Кренделла $CR_{n,c} = 2^n - c$, та здійснено порівняльний аналіз цих модифікацій та криптосистеми AJPS-1. Побудовано також дві модифікації криптосистеми AJPS-2, які використовують операції за модулем узагальненого числа Мерсенна $GM_{n,m}$ та операції за модулем числа Кренделла $CR_{n,c}$.

ВИСНОВКИ

При виконанні роботи наведено класифікацію постквантових криптопримітивів за типом математичних об'єктів, які використовуються в їх побудові, а також розглянуто узагальнену модель алгебраїчних задач на решітках з огляду її використання для побудови постквантових криптографічних примітивів. У роботі досліджуються криптопримітиви сімейства AJPS, стійкість яких ґрунтується на складності алгебраїчних задач MLHRSP та MLHCSP.

Дисертаційна робота містить опис схем шифрування AJPS-1, AJPS-2 та механізму інкапсуляції ключів AJPS-KEM, який є учасником першого раунду конкурсу постквантових криптографічних примітивів Національного інституту стандартів та технологій США (NIST), а також містить опис відомих модифікацій криптосистеми AJPS, зокрема:

– механізмів інкапсуляції ключів AJPS-KEM-Bivariate та AJPS-KEM-Trivariate, що використовують алгоритм пошуку з поверненням;

– інтерактивного протоколу односторонньої автентифікації та коду автентифікації повідомлень на основі задачі MLHCSP;

– генераторів псевдовипадкових чисел, які використовують задачі MLHRSP та MLHCSP.

Також у роботі здійснено огляд опублікованих атак на криптосистеми AJPS-1 та AJPS-2. Узагальнюючи вимоги для можливого застосування описаних атак, сформовано рекомендації для алгоритмів генерації ключів криптосистем AJPS-1 та AJPS-2. Побудовано нову атаку підміни з використанням моделі активного зловмисника на криптосистему AJPS-2, яка використовує властивість арифметики за модулем числа Мерсенна, що також доведена в роботі.

Кваліфікаційна робота містить доведення співвідношень для метрики OSD суми, добутку двох чисел та оберненого елемента відносно операції

додавання за модулем числа Мерсенна, а також доведення аналогічних співвідношень для ваги Хеммінга чисел за модулем узагальненого числа Мерсенна та за модулем числа Кренделла.

Використовуючи отримані співвідношення, побудовано модифікацію криптосистеми AJPS-1 з використанням метрики OSD, а також модифікації криптосистем AJPS-1 та AJPS-2, що застосовують операції за модулем узагальненого числа Мерсенна та за модулем числа Кренделла. Визначено нові алгебраїчні задачі з використанням арифметики за модулем узагальненого числа Мерсенна (задачі GMLHRSP та GMLHCSP) та з використанням арифметики за модулем числа Кренделла (задачі CRLHRSP та CRLHCSP), на складності яких основана стійкість побудованих модифікацій AJPS-1 та AJPS-2 шляхом використання інших класів чисел. Виконано порівняльний аналіз усіх побудованих модифікацій і криптосистем AJPS-1 та AJPS-2. Перевагами побудованих криптопримітивів є:

– збільшення множини значень, що приймає параметр розшифрування, у модифікації криптосистеми AJPS-1 з використанням метрики OSD;

– збільшення потужності класу використовуваних модулів та кількості унікальних значень параметра розшифрування у модифікаціях криптосистеми AJPS-1 з використанням арифметики за модулем узагальненого числа Мерсенна та за модулем числа Кренделла;

– збільшення потужності класу чисел, що використовуються в ролі модуля, у модифікаціях криптосистеми AJPS-2 шляхом застосування арифметики за модулем узагальненого числа Мерсенна та числа Кренделла.

Побудовані модифікації криптосистем AJPS-1 та AJPS-2 мають більшу варіативність параметрів, зокрема дозволяють використовувати різні класи чисел в якості модуля та різні метрики у криптосистемі AJPS-1, що підвищує гнучкість практичного застосування цих криптосистем. Також отримані результати можуть застосовуватись для

модифікації відомих криптосистем з використанням операцій за модулем числа Мерсенна та створення нових ефективних і стійких постквантових криптопримітивів.

У подальшому розвитку дослідження планується продовжити криптоаналіз криптографічних примітивів сімейства AJPS, зокрема більш детально дослідити їх стійкість в квантовій моделі обчислень. Застосовуючи відомі алгебраїчні методи, потрібно досліджувати складність задач MLHRSP та MLHCSP, а також, для аналізу запропонованих в роботі модифікацій AJPS-1 та AJPS-2, необхідно досліджувати складність задач GMLHRSP, GMLHCSP, CRLHRSP та CRLHCSP.

ПЕРЕЛІК ПОСИЛАНЬ

1. Post-Quantum Cryptography Standardization [Електронний ресурс] // National Institute of Standards and Technology, Information Technology Laboratory. — 2017. — Режим доступу: https://csrc.nist.gov/Projects/post-quantum-cryptography/Post-Quantum-Cryptography-Standardization.

2. Workshops and Timeline Post-Quantum Cryptography [Електронний ресурс] // National Institute of Standards and Technology, Information Technology Laboratory — Режим доступу: https://csrc.nist.gov/Projects/Post-Quantum-Cryptography/Workshops-and-Timeline.

3. Round 1 Post-Quantum Cryptography [Електронний ресурс] // National Institute of Standards and Technology, Information Technology Laboratory. — 2017. — Режим доступу: https://csrc.nist.gov/projects/post-quantum-cryptography/round-1-submissions.

4. A New Public-Key Cryptosystem via Mersenne Numbers [Електронний ресурс] / D.Aggarwal, A. Joux, A. Prakash, M. Santha // IACR Cryptology ePrint Archive, Report 2017/481. — 2017. — Режим доступу: https://eprint.iacr.org/2017/481.

5. Zaverucha G. Generalized Mersenne Numbers in Pairing-Based Cryptography [Електронний ресурс] / Greg Zaverucha. — 2006. — Режим доступу: https://crysp.uwaterloo.ca/software/gmnt/gmz_mcs_thesis.pdf.

6. Crandall R. Method and apparatus for public key exchange in a cryptographic system / Crandall Richard, 1992. — U.S. Paent 5,159,632.

7. Bajard J. Modular Number Systems: Beyond the Mersenne Family [Електронний ресурс] / J. Bajard, L. Imbert, T. Plantard // HAL-LIRMM. — 2004. — Режим доступу: https://hal-lirmm.ccsd.cnrs.fr/lirmm-00109208.

8. Taschwer M. Modular Multiplication Using Special Prime Moduli / Mario Taschwer // Kommunikationssicherheit im Zeichen des Internet / Mario Taschwer., 2001. — (Vieweg+Teubner Verlag). — ISBN 978-3-322-89558-5.

9. Crandall R. Prime Numbers: A Computational Perspective / R. Crandall, C. Pomerance. — New York: Springer, 2001. — 547 с. — (Springer-Verlag). — (Library of Congress Cataloging-in-Publication Data). — ISBN 978-1-4684-9318-4.

10. Nath K. Efficient Arithmetic in (Pseudo-) Mersenne Prime Order Fields [Електронний ресурс] / K. Nath, P. Sarkar // IACR Cryptology ePrint Archive, Report 2018/985. — 2018. — Режим доступу: https://eprint.iacr.org/2018/985.

11. Bos J. Efficient SIMD arithmetic modulo a Mersenne number [Електронний ресурс] / J. Bos, T. Kleinjung, A. Lenstra // IACR Cryptology ePrint Archive, Report 2010/338. — 2010. — Режим доступу: https://eprint.iacr.org/2010/338.

12. Scott M. On inversion modulo pseudo-Mersenne primes [Електронний ресурс] / Michael Scott // IACR Cryptology ePrint Archive, Report 2018/1038. — 2018. — Режим доступу: https://eprint.iacr.org/2018/1038.pdf.

13. Knuth D. The Art of Computer Programming, Volume II: Seminumerical Algorithms, 2nd Edition. / Donald E. Knuth, 1981. — ISBN 0201-03822-6.

14. Report on Post-Quantum Cryptography [Електронний ресурс] / [L. Chen, S. Jordan, Y. Liu та ін.] // National Institute of Standards and Technology, Information Technology Laboratory, NISTIR 8105. — 2016. — Режим доступу: https://nvlpubs.nist.gov/nistpubs/ir/2016/NIST.IR.8105.pdf.

15. Quantum Safe Cryptography and Security. An introduction, benefits, enablers and challenges — Sophia Antipolis: European Telecommunications Standards Institute, 2015. — 64 с. — (ETSI White Paper; № 8). — ISBN 979-10-92620-03-0.

16. Status Report on the First Round of the NIST Post-Quantum Cryptography Standardization Process [Електронний ресурс] / [G. Alagic, J. Alperin-Sheriff, D. Apon та ін.] // National Institute of Standards

and Technology, Information Technology Laboratory, NISTIR 8240. — 2019. — Режим доступу: https://nvlpubs.nist.gov/nistpubs/ir/2019/NIST.IR.8240.pdf.

17. Steinfeld R. Titanium: Proposal for a NIST Post-Quantum Public-key Encryption and KEM Standard [Електронний ресурс] / R. Steinfeld, A. Sakzad, R. Kuo Zhao. — 2017. — Режим доступу: http://users.monash.edu.au/~rste/Titanium_NISTSub.pdf.

18. NewHope: Algorithm Specifications and Supporting Documentation [Електронний ресурс] / [E. Alkim, R. Avanzi, J. Bos та ін.]. — 2017. — Режим доступу: https://cryptojedi.org/papers/newhopenist-20171128.pdf.

19. Round5: KEM and PKE based on (Ring) Learning with Rounding [Електронний ресурс] / [H. Baan, S. Bhattacharya, J. H. Cheon та ін.]. — 2020. — Режим доступу: https://round5.org/doc/Round5_Submission022020.pdf.

20. NIST PQ Submission: pqNTRUSign A modular lattice signature scheme [Електронний ресурс] / C.Chen, J. Hoffstein, W. Whyte, Z. Zhang. — 2017. — Режим доступу: https://csrc.nist.gov/CSRC/media/Projects/Post-Quantum-Cryptography/documents/round-1/submissions/pqNTRUsign.zip.

21. NTRU Prime [Електронний ресурс] / D.Bernstein, C. Chuengsatiansup, T. Lange, C. van Vredendaal. — 2017. — Режим доступу: https://csrc.nist.gov/CSRC/media/Projects/Post-Quantum-Cryptography/documents/round-1/submissions/NTRU_Prime.zip.

22. NIST PQ Submission: NTRUEncrypt A lattice based encryption algorithm [Електронний ресурс] / C.Chen, J. Hoffstein, W. Whyte, Z. Zhang. — 2017. — Режим доступу: https://csrc.nist.gov/CSRC/media/Projects/Post-Quantum-Cryptography/documents/round-1/submissions/NTRUEncrypt.zip.

23. High-speed key encapsulation from NTRU [Електронний ресурс] / A.Hülsing, J. Rijneveld, J. Schanck, P. Schwabe. — 2017. — Режим доступу: https://ntru-hrss.org/data/ntrukem-20170828.pdf.

24. BIKE: Bit Flipping Key Encapsulation [Електронний ресурс] / [N. Aragon, P. Barreto, S. Bettaieb та ін.]. — 2020. — Режим доступу: https://bikesuite.org/files/round2/spec/BIKE-Spec-2020.02.07.1.pdf.

25. Classic McEliece: conservative code-based cryptography [Електронний ресурс] / [D. Bernstein, T. Chou, T. Lange та ін.]. — 2019. — Режим доступу: https://classic.mceliece.org/nist/mceliece-20190331.pdf.

26. The SPHINCS+ Signature Framework [Електронний ресурс] / [D. Bernstein, A. Hülsing, S. Kölbl та ін.]. — 2019. — Режим доступу: https://sphincs.org/data/sphincs+-paper.pdf.

27. Aumasson J. Gravity-SPHINCS [Електронний ресурс] / J. Aumasson, G. Endignoux. — 2017. — Режим доступу: https://csrc.nist.gov/CSRC/media/Projects/Post-Quantum-Cryptography/documents/round-1/submissions/Gravity_SPHINCS.zip.

28. MQDSS specifications [Електронний ресурс] / [M. Chen, A. Hülsing, J. Rijneveld та ін.]. — 2019. — Режим доступу: http://mqdss.org/files/MQDSS_Ver2.pdf.

29. Rainbow — Algorithm Specification and Documentation [Електронний ресурс] / [J. Ding, M. Chen, A. Petzoldt та ін.]. — 2017. — Режим доступу: https://csrc.nist.gov/CSRC/media/Projects/Post-Quantum-Cryptography/documents/round-2/submissions/Rainbow-Round2.zip.

30. WalnutDSA: A Quantum-Resistant Digital Signature Algorithm [Електронний ресурс] / I.Anshel, D. Atkins, D. Goldfeld, P. Gunnells // IACR Cryptology ePrint Archive, Report 2017/058. — 2017. — Режим доступу: https://eprint.iacr.org/2017/058.pdf.

31. Supersingular Isogeny Key Encapsulation [Електронний ресурс] / [D. Jao, R. Azarderakhsh, M. Campagna та ін.]. — 2019. — Режим доступу: https://sike.org/files/SIDH-spec.pdf.

32. Round 2 Post-Quantum Cryptography [Електронний ресурс] // National Institute of Standards and Technology, Information Technology Laboratory. — 2019. — Режим доступу: https://csrc.nist.gov/projects/

post-quantum-cryptography/round-2-submissions.

33. NIST Reveals 26 Algorithms Advancing to the Post-Quantum Crypto "Semifinals" [Електронний ресурс] // National Institute of Standards and Technology. — 2019. — Режим доступу: https://www.nist.gov/news-events/news/2019/01/nist-reveals-26-algorithms-advancing-post-quantum-crypto-semifinals.

34. FIPS 186-4 Digital Signature Standard (DSS) [Електронний ресурс] // National Institute of Standards and Technology. — 2013. — Режим доступу: https://csrc.nist.gov/publications/detail/fips/186/4/final.

35. SP 800-56A Rev. 3 Recommendation for Pair-Wise Key-Establishment Schemes Using Discrete Logarithm Cryptography [Електронний ресурс] / [E. Barker, L. Chen, A. Roginsky та ін.] // National Institute of Standards and Technology. — 2018. — Режим доступу: https://csrc.nist.gov/publications/detail/sp/800-56a/rev-3/final.

36. SP 800-56B Rev. 2 Recommendation for Pair-Wise Key-Establishment Using Integer Factorization Cryptography [Електронний ресурс] / [E. Barker, L. Chen, A. Roginsky та ін.] // National Institute of Standards and Technology. — 2019. — Режим доступу: https://csrc.nist.gov/publications/detail/sp/800-56b/rev-2/final.

37. A New Public-Key Cryptosystem via Mersenne Numbers [Електронний ресурс] / D.Aggarwal, A. Joux, A. Prakash, M. Santha // IACR Cryptology ePrint Archive, Report 2017/481, version:20170530.072202. — 2017. — Режим доступу: https://eprint.iacr.org/eprint-bin/versions.pl?entry=2017/481.

38. Kazue S. Semantic Security / Sako Kazue // Encyclopedia of Cryptography and Security / Sako Kazue. — Boston, MA: Springer US, 2011. — С. 1176 – 1177. — ISBN 978-1-4419-5905-8.

39. Bellare M. Introduction to Modern Cryptography [Електронний ресурс] / M. Bellare, P. Rogaway // University of California. — 2005. — Режим доступу: https://web.cs.ucdavis.edu/~rogaway/classes/227/spring05/book/main.pdf.

40. Regev O. On lattices, learning with errors, random linear codes, and cryptography / Oded Regev. // Journal of the ACM. — New York: Association for Computing Machinery, 2009. — №56.

41. Baktir S. Optimal Extension Field Inversion in the Frequency Domain / S. Baktir, B. Sunar // Arithmetic of Finite Fields / S. Baktir, B. Sunar. — Siena: Springer, 2008. — (Theoretical Computer Science and General Issues). — (Lecture Notes in Computer Science; т. 5130). — С. 47 – 61.

42. On the Hardness of the Mersenne Low Hamming Ratio Assumption [Електронний ресурс] / M.Beunardeau, A. Connolly, R. Geraud, D. Naccache // IACR Cryptology ePrint Archive, Report 2017/522. — 2017. — Режим доступу: https://eprint.iacr.org/2017/522.

43. Attacks on the AJPS Mersenne-Based Cryptosystem / K. de Boer, L. Ducas, S. Jeffery, R. de Wolf // Post-Quantum Cryptography / K. de Boer, L. Ducas, S. Jeffery, R. de Wolf. — Cham: Springer, 2018. — (PQCrypto 2018). — (Lecture Notes in Computer Science; № 10786). — С. 101 – 120.

44. Tiepelt M. Quantum LLL with an Application to Mersenne Number Cryptosystems / M. Tiepelt, A. Szepieniec // Progress in Cryptology — LATINCRYPT 2019 / M. Tiepelt, A. Szepieniec., 2019. — ISBN 978-3-030-30529-1.

45. Coron J. Improved Cryptanalysis of the AJPS Mersenne Based Cryptosystem [Електронний ресурс] / J. Coron, A. Gini // IACR Cryptology ePrint Archive, Report 2019/610. — 2019. — Режим доступу: https://eprint.iacr.org/2019/610.

46. Budroni A. The Mersenne Low Hamming Combination Search Problem can be reduced to an ILP Problem / A. Budroni, A. Tenti // Lecture Notes in Computer Science / A. Budroni, A. Tenti. — Cham: Springer, 2019. — (Progress in Cryptology — AFRICACRYPT 2019). — С. 41 – 55. — ISBN 978-3-030-23695-3.

47. Papadimitriou C. On the Complexity of Integer Programming / Christos H. Papadimitriou. // J. ACM. — 1981. — №28. — С. 765 – 768.

48. Tiepelt M. Exploiting Decryption Failures in Mersenne Number

Cryptosystems [Електронний ресурс] / M. Tiepelt, J. D'Anvers // IACR Cryptology ePrint Archive, Report 2020/367. — 2020. — Режим доступу: https://eprint.iacr.org/2020/367.pdf.

49. Szepieniec A. Ramstake. KEM Proposal for NIST PQC Project [Електронний ресурс] / Alan Szepieniec // NIST PQ Submission. — 2017. — Режим доступу: https://csrc.nist.gov/CSRC/media/Projects/Post-Quantum-Cryptography/documents/round-1/submissions/Ramstake.zip.

50. Ferradi H. Integer Reconstruction Public-Key Encryption / H. Ferradi, D. Naccache // Cryptology and Network Security. CANS 2019. / H. Ferradi, D. Naccache., 2019. — (Lecture Notes in Computer Science; вип. 11829). — С. 412 – 433. — ISBN 978-3-030-31577-1.

51. Ferradi H. Post-quantum Provably-Secure Authentication and MAC from Mersenne Primes / H. Ferradi, K. Xagawa // Topics in Cryptology — CT-RSA 2020 / H. Ferradi, K. Xagawa., 2020. — (Lecture Notes in Computer Science; вип. 12006). — С. 469 – 495. — ISBN 978-3-030-40185-6.

52. Nan J. Post-Quantum Pseudorandom Functions from Mersenne Primes / J. Nan, M. Zheng, H. Hu // Communications in Computer and Information Science. Frontiers in Cyber Security / J. Nan, M. Zheng, H. Hu. — Singapore: Springer, 2019. — (Second International Conference, FCS 2019 Xi'an, China, November 15 — 17, 2019 Proceedings; вип. 1105). — С. 128 — 142. — ISBN 978-981-15-0817-2.

53. A Framework for Cryptographic Problems from Linear Algebra [Електронний ресурс] / C.Bootland, W. Castryck, A. Szepieniec, F. Vercauteren // IACR Cryptology ePrint Archive, Report 2019/282. — 2019. — Режим доступу: https://eprint.iacr.org/2019/282.pdf.

54. Ajtai M. Generating hard instances of lattice problems (extended abstract) / Miklós Ajtai // STOC '96: Proceedings of the twenty-eighth annual ACM symposium on Theory of Computing / Miklós Ajtai. — New York: Association for Computing Machinery, 1996. — С. 99 — 108. — ISBN 978-0-89791-785-8.

55. Hofistein J. NTRU: A New High Speed Public Key Cryptosystem (preliminary draft) [Електронний ресурс] / J. Hofistein, J. Pipher, J. Silverman. — 1996. — Режим доступу: https://web.securityinnovation.com/hubfs/files/ntru-orig.pdf.

56. Yadukha D. The necessary security requirements for the values used by the AJPS cryptosystem / D. Yadukha, A. Fesenko. // Theoretical and Applied Cybersecurity. — 2019. — №1. — С. 31 – 36. — ISSN 2664-2913.

57. Ядуха Д. В. Побудова атак на криптосистему AJPS з використанням моделі активного зловмисника / Д. В. Ядуха, А. В. Фесенко // «ІНТЕРНЕТ-ОСВІТА-НАУКА-2018», Одинадцята міжнародна науково-практична конференція ІОН-2018, 22-25 травня, 2018: Збірник праць / Д. В. Ядуха, А. В. Фесенко. — Вінниця: ВНТУ, 2018. — С. 193 – 195. — ISBN 978-966-641-728-5.

58. Wyner A. The wire-tap channel / A. D. Wyner. // The Bell System Technical Journal. — 1975. — №54. — С. 1355 – 1387. — ISBN 0005-8580.

59. Simmons G. Authentication Theory & Coding Theory / Gustavus J. Simmons // CRYPTO 1984: Advances in Cryptology / Gustavus J. Simmons. — Berlin: Springer, 1985. — (Lecture Notes in Computer Science; вип. 184). — С. 411 – 431. — ISBN 978-3-540-15658-1.

60. Bailey D. V. Optimal extension fields for fast arithmetic in public-key algorithms / D. V. Bailey, C. Paar // Advances in Cryptology — CRYPTO '98 / D. V. Bailey, C. Paar. — Berlin: Springer, 1998. — (Lecture Notes in Computer Science; вип. 1462). — С. 472 – 485.

ДОДАТОК А ДОВЕДЕННЯ ДОПОМІЖНИХ ТЕОРЕМ

Теорема. *Нехай* $GM_{n,m,k}$ *— узагальнене число Мерсенна виду* $GM_{n,m,k} = 2^n - 2^m - 1 - k$, *де* $n, m, k \in \mathbb{N}$, $n > m$ *і* $k < 2^n - 2^m - 1$, *та нехай є два* n*-бітових числа* A, B *таких, що* $A \leqslant GM_{n,m,k}$ *та* $B \leqslant GM_{n,m,k}$. *Тоді виконується таке співвідношення для ваги Хеммінга суми чисел* A *та* B *за модулем* $GM_{n,m,k}$:

$$Ham(A + B \bmod GM_{n,m,k}) \leqslant Ham(A) + Ham(B) + k.$$

Доведення. Розглянемо доведення співвідношення при $k = 1$, для інших значень параметра k доведення здійснюється аналогічно.

Очевидно, що кожна одиниця, яка виникає у числі $A + B \bmod GM_{n,m,k}$, породжена одиницею одного з чисел A та B, тому кількість одиниць у числі $A + B \bmod GM_{n,m,k}$ менша за суму кількості одиниць у числах A та B, але лише у випадку, коли не виникає біт перенесення на старшу позицію числа. У випадку якщо хоча б один зі старших бітів чисел A та B рівний 1, при обчисленні редукції за модулем $GM_{n,m,k}$ може виникнути біт перенесення, що призводить до збільшення ваги Хеммінга на 1. Отже, значення максимально можливої ваги Хеммінга числа $A + B \bmod GM_{n,m,k}$ дорівнює $Ham(A) + Ham(B) + 1$. □

Теорема. *Нехай* $GM_{n,m,k}$ *— узагальнене число Мерсенна виду* $GM_{n,m,k} = 2^n - 2^m - 1 - k$, *де* $n, m, k \in \mathbb{N}$, $n > m$ *і* $k < 2^n - 2^m - 1$, *та нехай* A *— n-бітове число таке, що* $A \leqslant GM_{n,m,k}$. *Тоді виконуються такі співвідношення для ваги Хеммінга:*

1) $Ham(2 \cdot A \bmod GM_{n,m,k}) \leqslant Ham(A) + k$;

2) $Ham(2^r \cdot A \bmod GM_{n,m,k}) \leqslant$

$$\leqslant Ham(A) + k \cdot \min(Ham(A), r) + \mathbb{1}(r \geqslant n - m) \cdot (r - n + m),$$

де $r \in \mathbb{N}$, $\mathbb{1}(A)$ *— індикатор події* A, *тобто* $\mathbb{1}(A) = 1$, *якщо подія* A *виконується, і* $\mathbb{1}(A) = 0$, *якщо ні.*

Доведення. Розглянемо доведення співвідношень при $k = 1$, для інших значень параметра k доведення здійснюється аналогічно.

Слід помітити, що модуль $GM_{n,m,k}$ має такий вигляд у двійковому записі: $11\ldots101\ldots10$, причому саме m-ий та 0-ий біти дорівнюють 0.

1) Відомо, що множення на двійку є зсувом числа на один розряд у бік старших бітів, тобто якщо число A у двійковому записі має вигляд

$$A = a_{n-1}\ldots a_1\ a_0,\ \text{де}\ a_i \in \{0,1\},\ i = \overline{0,n-1},$$

то число $2A$ буде $(n+1)$-бітовим числом та має вигляд

$$2A = a_{n-1}\ldots a_1\ a_0\ 0,\ \text{де}\ a_i \in \{0,1\},\ i = \overline{0,n-1}.$$

Очевидно, що вага Хеммінга числа A та вага Хеммінга числа $2A$ є однаковими.

Розглянемо два можливих випадки, залежно від значення старшого біту a_{n-1}.

а) Якщо $a_{n-1} = 0$, то число $2A$ може бути представлене як n-бітове число, отже, при обчисленні редукції за модулем $GM_{n,m,k}$ буде мати не більшу вагу, ніж $Ham(2A)$. Оскільки $Ham(2A) = Ham(A)$, то максимальна вага Хеммінга $2A \bmod GM_{n,m,k}$ при $a_{n-1} = 0$ буде $Ham(A)$.

б) Якщо ж $a_{n-1} = 1$, то представимо число $2A$ як

$$2A = A^* + 2^n,\ \text{де}\ A^* = a_{n-2}\ldots a_0\ 0.$$

Число A^* є n-бітовим і має вагу Хеммінга на одиницю меншу, ніж число A. Тоді маємо:

$$Ham(A^* \bmod GM_{n,m,k}) \leqslant Ham(A^*) = Ham(A) - 1.$$

При обчисленні $2^n \bmod GM_{n,m,k}$ отримаємо

$$2^n \bmod (2^n - 2^m - 2) = 2^n - (2^n - 2^m - 2) = 2^m + 2,$$

тобто $Ham(2^n \bmod GM_{n,m,k}) = 2$. Таким чином, максимально можлива вага Хеммінга $2A \bmod GM_{n,m,k}$ дорівнює

$$Ham(A) - 1 + 2 = Ham(A) + 1.$$

Отже,

$$Ham(2A \bmod GM_{n,m,k}) \leqslant Ham(A) + 1.$$

2) При множенні на 2^r відбувається зсув на r позицій в бік старших розрядів, тобто число $2^r A$ є $(n+r)$-бітовим та має такий вигляд:

$$a_{n-1} \dots a_1 \; a_0 \; 0 \dots 0, \text{ де } a_i \in \{0,1\}, \; i = \overline{0, n-1}.$$

Будемо розглядати зсув на r позицій як послідовний зсув на один біт в сторону старших розрядів r разів. При першому зсуві, використовуючи співвідношення з пункту 1) даної теореми, маємо

$$Ham(2A \bmod GM_{n,m,k}) \leqslant Ham(A) + 1,$$

де 1 відповідає можливому біту перенесення внаслідок того, що старший біт числа A дорівнює одиниці. При послідовному зсуві на r позицій може виникати у співвідношенні ваги Хеммінга доданок r у випадку, якщо при кожному зсуві старший біт числа A буде рівний 1. Зрозуміло, що це можливо лише у випадку, коли двійковий запис числа A містить більше ніж r одиниць, тобто $Ham(A) > r$. Таким чином, оцінка ваги Хеммінга числа $2^r \cdot A \bmod GM_{n,m,k}$ матиме вигляд

$$Ham(A) + \min(Ham(A), r).$$

Однак, можливий випадок, коли старший біт числа $2A^{**}$, де $A^{**} = 2^q \cdot A \bmod GM_{n,m,k}$, $q < r$ є одиницею, яка породжена не одиницею початкового числа A, а одиницею, яка виникла при обчисленні редукції числа $2^l A$, де $l \in \mathbb{N}$, $l < q$, за модулем $GM_{n,m,k}$ у випадку, якщо на n-ій позиції бітового запису числа $2^l A$ стоїть 1. Це можливо, якщо кількість

зсувів на одну позицію більша ніж $n - m$, тобто $r > n - m$. Таких одиниць може виникнути не більше ніж

$$r - (n - m) = r - n + m.$$

Отже, максимально можлива вага Хеммінга числа $2^r \cdot A \bmod GM_{n,m,k}$ дорівнює

$$Ham(A) + \min(Ham(A), r) + \mathbb{1}(r \geqslant n - m) \cdot (r - n + m),$$

що і треба було довести.

□

Теорема. *Нехай $GM_{n,m,k}$ — узагальнене число Мерсенна виду $GM_{n,m,k} = 2^n - 2^m - 1 - k$, де $n, m, k \in \mathbb{N}$, $n > m$ і $k < 2^n - 2^m - 1$, та нехай є два n-бітових числа A, B таких, що $A \leqslant GM_{n,m,k}$ та $B \leqslant GM_{n,m,k}$. Тоді виконується таке співвідношення для ваги Хеммінга добутку двох чисел за модулем $GM_{n,m,k}$:*

$$Ham(A \cdot B \bmod GM_{n,m,k}) \leqslant$$
$$\leqslant (k + 1) \cdot Ham(A) \cdot Ham(B) + \min(Ham(A), Ham(B)) + \frac{m(m-1)}{2}.$$

Доведення. Розглянемо доведення співвідношень при $k = 1$, для інших значень параметра k доведення здійснюється аналогічно.

Представимо число B у вигляді суми чисел $B_l = 2^{i_l}$, де $i_l \in \mathbb{N}$ та $i_l \neq i_w$ при $l \neq w$. В такому випадку доданків у сумі буде рівно $Ham(B)$, тобто

$$B = B_1 + \ldots + B_{Ham(B)}, \text{ де } B_l \in \{0, 1\}^n, \; l = \overline{1, Ham(B)},$$

причому $Ham(B_l) = 1$ для будь-якого значення l. Тоді добуток

$A \cdot B \bmod GM_{n,m,k}$ можна представити у вигляді суми таким чином:

$$Ham(A{\cdot}B \bmod GM_{n,m,k}) = Ham\left(A \cdot \left(B_1 + \ldots + B_{Ham(B)}\right) \bmod GM_{n,m,k}\right) =$$

$$= Ham\left(A \cdot B_1 + A \cdot B_2 + \ldots + A \cdot B_{Ham(B)} \bmod GM_{n,m,k}\right).$$

Використовуючи співвідношення для ваги Хеммінга суми двох чисел за модулем числа $GM_{n,m,k}$, маємо:

$$Ham(A \cdot B \bmod GM_{n,m,k}) =$$

$$= Ham(A \cdot B_1 + A \cdot B_2 + \ldots + A \cdot B_{Ham(B)} \bmod GM_{n,m,k}) \leqslant$$

$$\leqslant Ham(A \cdot B_1 \bmod GM_{n,m,k}) + \ldots + Ham(A \cdot B_{Ham(B)} \bmod GM_{n,m,k}) +$$

$$+ Ham(B) - 1 \leqslant$$

$$\leqslant Ham(A \cdot B_1 \bmod GM_{n,m,k}) + \ldots + Ham(A \cdot B_{Ham(B)} \bmod GM_{n,m,k}) +$$

$$+ Ham(B).$$

Оскільки $B_l \in \{0,1\}^n$, $l = \overline{1, Ham(B)}$ є степенем двійки, тобто $B_l = 2^{i_l}$, то, застосувавши співвідношення для ваги Хеммінга добутку двох чисел за модулем $GM_{n,m,k}$, при умові, що одне з чисел є степенем двійки, отримаємо:

$$Ham(A \cdot B \bmod GM_{n,m,k}) \leqslant$$

$$\leqslant Ham(A) + \min(i_1, Ham(A)) + \mathbb{1}(i_1 \geqslant n - m) \cdot (i_1 - n + m) + \ldots +$$

$$+ Ham(A) + \min(i_{Ham(B)}, Ham(A)) + \mathbb{1}(i_{Ham(B)} \geqslant n - m) \cdot (i_{Ham(B)} - n + m) +$$

$$+ Ham(B) = Ham(A) \cdot Ham(B) + Ham(B) +$$

$$+ \min(i_1, Ham(A)) + \ldots + \min(i_{Ham(B)}, Ham(A)) +$$

$$+ \mathbb{1}(i_1 \geqslant n - m) \cdot (i_1 - n + m) + \ldots + \mathbb{1}(i_{Ham(B)} \geqslant n - m) \cdot (i_{Ham(B)} - n + m).$$

Очевидно, що можемо оцінити $\min(i_l, Ham(A))$ як

$\min(i_l, Ham(A)) \leqslant Ham(A),\ l = \overline{1, Ham(B)}$. Тоді отримаємо:

$$Ham(A \cdot B \bmod GM_{n,m,k}) \leqslant$$

$$\leqslant Ham(A) \cdot Ham(B) + Ham(B) + Ham(A) \cdot Ham(B) +$$

$$+\mathbb{1}(i_1 \geqslant n-m) \cdot (i_1 - n + m) + \ldots + \mathbb{1}(i_{Ham(B)} \geqslant n-m) \cdot (i_{Ham(B)} - n + m) =$$

$$= 2 \cdot Ham(A) \cdot Ham(B) + Ham(B) +$$

$$+\mathbb{1}(i_1 \geqslant n-m) \cdot (i_1 - n + m) + \ldots + \mathbb{1}(i_{Ham(B)} \geqslant n-m) \cdot (i_{Ham(B)} - n + m).$$

У випадку максимально можливої ваги Хеммінга кожен з індикаторів дорівнює 1, тобто $i_l \geqslant n - m$ для всіх $l = \overline{1, Ham(B)}$. Мінімально можливе значення i_l дорівнює $n - m$, а максимальне — $n - 1$:

$$Ham(A \cdot B \bmod GM_{n,m,k}) \leqslant$$

$$\leqslant 2 \cdot Ham(A) Ham(B) + Ham(B) + (i_1 - n + m) + \ldots + (i_{Ham(B)} - n + m) \leqslant$$

$$\leqslant 2 \cdot Ham(A) \cdot Ham(B) + Ham(B) + 0 + 1 + 2 + \ldots + m - 2 + m - 1 =$$

$$= 2 \cdot Ham(A) \cdot Ham(B) + Ham(B) + \frac{m(m-1)}{2}.$$

Аналогічно можемо представити число A у вигляді суми

$$A = A_1 + \ldots + A_{Ham(A)},\ \text{де}\ A_t \in \{0,1\}^n,\ t = \overline{1, Ham(A)},$$

причому $Ham(A_t) = 1$ для будь-якого значення t. Тоді отримаємо:

$$Ham(A \cdot B \bmod GM_{n,m,k}) \leqslant$$

$$\leqslant 2 \cdot Ham(A) \cdot Ham(B) + Ham(A) + \frac{m(m-1)}{2}.$$

Отже, загальна оцінка зверху вага Хеммінга добутку двох чисел за модулем $GM_{n,m,k}$ буде такою

$$Ham(A \cdot B \bmod GM_{n,m,k}) \leqslant$$

$$\leqslant 2 \cdot Ham(A) \cdot Ham(B) + \min(Ham(A), Ham(B)) + \frac{m(m-1)}{2},$$

що і треба було довести. □

Теорема. *Для узагальненого числа Мерсенна* $GM_{n,m} = 2^n - 2^m - 1$, *де* $n, m \in \mathbb{N}$, $m < n$, *та* n*-бітового числа* $A = a_{n-1}\ a_{n-2} \dots a_1\ a_0$, *де* $a_i \in \{0, 1\}$, $i = \overline{0, n-1}$, *такого, що* $A \leqslant GM_{n,m}$, *виконується:*

1) *якщо* $a_m = 0$, *то*

$$Ham(-A \bmod GM_{n,m}) = n - 1 - Ham(A);$$

2) *якщо* $a_m = 1$, *то*

$$Ham(-A \bmod GM_{n,m}) = l - Ham(h_1) + Ham(h_2) + m - Ham(h_3),$$

де $A = h_1 \,||\, h_2 \,||\, h_3$, *причому:*

а) $|h_3| = m$, *тобто* h_3 *включає молодші* m *бітів числа* A: $h_3 = a_{m-1}\ a_{m-2} \dots a_1\ a_0$;

б) $h_2 = a_k\ a_{k-1} \dots a_m$, *де* $k = \min\limits_i \{a_i = 0 \mid a_j = 1, m \leqslant j < i\}$, *тобто* h_2 *містить біти починаючи з* a_m *та до першого нуля, який зустрінеться після* a_m, *включно;*

в) $h_1 = a_{n-1}\ a_{n-2} \dots a_k$, *де* k *— індекс з минулого пункту;* l *— довжина числа* h_1, *тобто* $|h_1| = l$.

Доведення. Важливо помітити, що модуль $GM_{n,m} = 2^n - 2^m - 1$ має такий вигляд у двійковому записі:

$$GM_{n,m} = 111 \dots 110111 \dots 11,$$

причому саме m-ий біт рівний 0. Потрібно, маючи число A, знайти таке число $B \in \{0, 1\}^n$, що

$$A + B = 0 \bmod GM_{n,m},$$

тоді B якраз і буде оберненим до A відносно операції додавання за модулем $GM_{n,m}$, тобто

$$B = -A \bmod GM_{n,m}.$$

Слід зауважити, що потрібно шукати таке B, що $A + B = GM_{n,m}$, оскільки за умовою $A \in \{0,1\}^n$, тобто випадок $A+B = c \cdot GM_{n,m}$, де $c > 1$ — деяка константа, отримати неможливо. Введемо наступні позначення:

$$A = a_{n-1}\ a_{n-2} \ldots a_m \ldots a_1\ a_0,$$

$$B = b_{n-1}\ b_{n-2} \ldots b_m \ldots b_1\ b_0,$$

$$GM_{n,m} = g_{n-1}\ g_{n-2} \ldots g_m \ldots g_1\ g_0,$$

де $a_i, b_i, g_i \in \{0,1\},\ i = \overline{0, n-1}$. Відомо, що $g_m = 0$, а інші біти — 1, тобто $GM_{n,m} = 11 \ldots 101 \ldots 11$. Потрібно знайти b_i по відомим значенням $a_i,\ i = \overline{0, n-1}$, тоді можна буде побачити залежність ваги Хеммінга оберненого числа від ваги самого A.

1) Якщо $a_m = 0$, то значення b_i будуть наступні:

$$\begin{cases} b_m = 0 \\ b_i = 1 - a_i \quad i = \overline{0, n-1},\ i \neq m \end{cases}$$

Щоб перевірити правильність даного твердження, запишемо наступне. Для того, щоб $B = -A \bmod GM_{n,m}$ необхідно виконання рівності:

$$\begin{array}{rcccccccc} & a_{n-1} & a_{n-2} & \ldots & a_{m+1} & a_m & a_{m-1} & \ldots & a_1 & a_0 \\ + & b_{n-1} & b_{n-2} & \ldots & b_{m+1} & b_m & b_{m-1} & \ldots & b_1 & b_0 \\ \hline & 1 & 1 & \ldots & 1 & 0 & 1 & \ldots & 1 & 1 \end{array}$$

Підставляючи значення b_i, бачимо, що це дійсно виконується:

$$\begin{array}{cccccccccc} & a_{n-1} & a_{n-2} & \dots & a_{m+1} & 0 & a_{m-1} & \dots & a_1 & a_0 \\ + & 1-a_{n-1} & 1-a_{n-2} & \dots & 1-a_{m+1} & 0 & 1-a_{m-1} & \dots & 1-a_1 & 1-a_0 \\ \hline & 1 & 1 & \dots & 1 & 0 & 1 & \dots & 1 & 1 \end{array}$$

Таким чином, у даному випадку для знаходження оберненого потрібно усі біти, окрім m-го, замінити на протилежні, а m-ий біт залишити без змін. Тоді

$$Ham(-A \bmod GM_{n,m}) = n - Ham(A) - 1,$$

де n відповідає максимально можливій вазі n-бітового числа та, віднімаючи 1, враховується незмінний m-ий біт.

2) У випадку $a_m = 1$ різниця з попереднім у тому, що для отримання $g_m = 0$ потрібно щоб $b_m = 1$, а це утворює біт перенесення на старші біти. Це і обумовлює розділення двійкового запису числа на три частини, тобто

$$A = h_1 \,||\, h_2 \,||\, h_3.$$

Тоді вагу Хеммінга A можна представити як суму

$$Ham(A) = Ham(h_1) + Ham(h_2) + Ham(h_3).$$

Позначимо обернений елемент до A відносно операції додавання за модулем $GM_{n,m}$ як B:

$$B = -A \bmod GM_{n,m} = h_1^* \,||\, h_2^* \,||\, h_3^*.$$

Тоді h_3^* буде знаходитись аналогічно пункту 1), замінюючи всі біти h_3 на протилежні. Очевидно, що

$$Ham(h_3^*) = m - Ham(h_3).$$

Наймолодший біт h_2 — це біт $a_m = 1$ і, як було сказано раніше, $b_m = 1$, отже, біт перенесення переходить на старший біт a_{m+1}. Можливі

два випадки:

а) $a_{m+1} = 0$, тоді, оскільки $g_{m+1} = 1$, $b_{m+1} = 0$;

б) $a_{m+1} = 1$ — в такому випадку, враховуючи, що $g_{m+1} = 1$, отримаємо $b_{m+1} = 1$, тобто знову отримаємо біт перенесення та знову виконуємо аналогічну процедуру, але вже для біту a_{m+2}.

Бачимо, що біт перенесення буде з'являтись на кожному кроці, аж поки у послідовності бітів не зустрінеться 0. Саме тому частина h_2 містить біти від a_m до першого нуля, який зустрінеться у двійковому записі числа A. Слід зазначити, що 0 точно буде, оскільки розглядаються числа менші за модуль $GM_{n,m}$. Бачимо, що при знаходженні h_2^* біти h_2 не змінювались (якщо $a_{m+c} = 0$, то $b_{m+c} = 0$, і якщо $a_{m+c} = 1$, то і $b_{m+c} = 1$, де $c \in \mathbb{N}$ — деяка константа), а отже

$$Ham(h_2^*) = Ham(h_2).$$

Частина h_1^* повинна бути такою, щоб результатом суми з h_1 за модулем 2 було значення $11\ldots1$. Аналогічно до попередніх результатів, h_1^* отримується шляхом заміни бітів h_1 на протилежні. У такому випадку, якщо $|h_1| = l$, то

$$Ham(h_1^*) = l - Ham(h_1).$$

Узагальнюючи отримані результати, маємо:

$$Ham(-A \bmod GM_{n,m}) = Ham(B) = Ham(h_1^*) + Ham(h_2^*) + Ham(h_3^*) =$$

$$= l - Ham(h_1) + Ham(h_2) + m - Ham(h_3),$$

що і треба було довести.

□

Теорема. *Нехай задані число Кренделла* $CR_{n,c} = 2^n - c$, *де* $n, c \in \mathbb{N}$, *та* n*-бітове число* A *таке, що* $A \leqslant CR_{n,c}$. *Визначимо такі позначення:*

$$A = a_{n-1}\ a_{n-2} \ldots a_1\ a_0;$$

$$CR_{n,c} = r_{n-1}\ r_{n-2} \ldots r_1\ r_0;$$

$$B = -A \bmod CR_{n,c} = b_{n-1}\ b_{n-2} \ldots b_1\ b_0,$$

де $a_i, r_i, b_i \in \{0, 1\}$, $i = \overline{0, n-1}$. *Тоді вага Хеммінга оберненого до* A *елемента відносно операції додавання за модулем* $CR_{n,c}$ *(тобто вага Хеммінга числа* B*) обчислюється таким чином:*

1) *якщо число* $a_{\lceil \log_2 c \rceil - 1} \ldots a_1\ a_0$ *менше за* $r_{\lceil \log_2 c \rceil - 1} \ldots r_1\ r_0$, *то:*

$$Ham(-A \bmod CR_{n,c}) = n - \lceil \log_2 c \rceil - Ham(h_1) + Ham(h_2^*),\ \text{де}$$

а) $A = h_1 \mid\mid h_2$, *причому* h_2 — *молодші* $\lceil \log_2 c \rceil$ *бітів числа* A;

б) $B = -A \bmod CR_{n,c} = h_1^* \mid\mid h_2^*$, *аналогічно,* h_2^* *складається з* $\lceil \log_2 c \rceil$ *молодших бітів числа* B;

2) *якщо ж число* $a_{\lceil \log_2 c \rceil - 1} \ldots a_1\ \ a_0$ *є більшим за число* $r_{\lceil \log_2 c \rceil - 1} \ldots r_1\ r_0$, *то:*

$$Ham(-A \bmod CR_{n,c}) = n - \lceil \log_2 c \rceil - |h_2| - Ham(h_1) + Ham(h_2) + Ham(h_3^*),\ \text{де}$$

а) $A = h_1 \mid\mid h_2 \mid\mid h_3$, *причому:*

– h_3 — *молодші* $\lceil \log_2 c \rceil$ *бітів числа* A;

– h_2 *містить біти числа* A *починаючи з* $\lceil \log_2 c \rceil$ *біту та до першого нуля, який зустрінеться після* $a_{\lceil \log_2 c \rceil - 1}$;

– $|h_2|$ — *кількість бітів у числі* h_2;

– h_1 — *старші біти, що залишились, тобто* $h_1 = a_{n-1}\ a_{n-2} \ldots a_w$, *де* a_{w-1} — *старший біт* h_2;

б) $B = -A \bmod CR_{n,c} = h_1^* \mid\mid h_2^* \mid\mid h_3^*$, *де* h_3^* — *молодші* $\lceil \log_2 c \rceil$ *бітів числа* B.

Доведення. Слід зауважити, що число Кренделла $CR_{n,c}$ має вигляд

$111\ldots11**\ldots*$, де під символом $*$ розуміється або 0, або 1, залежно від значення константи c. Кількість таких неоднозначно визначених бітів — $\lceil \log_2 c \rceil$. Для того, щоб B був оберненим до A елементом за модулем $CR_{n,c}$, необхідно:

$$\begin{array}{cccccccccc} & a_{n-1} & a_{n-2} & \ldots & a_{m+1} & a_m & a_{m-1} & \ldots & a_1 & a_0 \\ + & b_{n-1} & b_{n-2} & \ldots & b_{m+1} & b_m & b_{m-1} & \ldots & b_1 & b_0 \\ \hline & 1 & 1 & \ldots & 1 & r_{\lceil \log_2 c \rceil - 1} & r_{\lceil \log_2 c \rceil - 2} & \ldots & r_1 & r_0 \end{array}$$

Оскільки значення $r_{\lceil \log_2 c \rceil - 1}, \ldots, r_1,\ r_0$ не фіксовані, причому від них прямо залежать значення $b_{\lceil \log_2 c \rceil - 1}, \ldots, b_1,\ b_0$, то вага Хеммінга числа B буде залежати від значення $Ham(h_2^*)$, для обчислення якого потрібно вирахувати значення $b_{\lceil \log_2 c \rceil - 1},\ b_{\lceil \log_2 c \rceil - 2}, \ldots, b_1,\ b_0$ та знайти кількість одиниць серед них.

Значення $b_0, b_1, \ldots, b_{\lceil \log_2 c \rceil - 1}$ обчислюються таким чином:

$$\begin{cases} b_0 = (r_0 + a_0) \bmod 2; \\ b_i = (r_i + a_i + x_i) \bmod 2, \quad \text{де } x_i = \begin{cases} 1, & \text{якщо } a_{i-1} + b_{i-1} + x_{i-1} > 1; \\ 0, & \text{інакше.} \end{cases} \\ \qquad\qquad\qquad\qquad \text{для } i = \overline{1, \lceil \log_2 c \rceil - 1}, \text{причому } x_0 = 0. \end{cases}$$

Тоді можна знайти кількість одиниць серед $\lceil \log_2 c \rceil$ молодших бітів числа B, позначимо це значення як z.

1) Очевидно, що якщо

$$B = -A \bmod CR_{n,c} = h_1^* \,||\, h_2^*,$$

то вага Хеммінга числа B має такий вигляд:

$$Ham(B) = Ham(h_1^*) + Ham(h_2^*).$$

У даному випадку $Ham(h_2^*) = z$.

Важливою умовою є те, що $a_{\lceil \log_2 c \rceil - 1} \dots a_1 \ a_0$ менше числа $r_{\lceil \log_2 c \rceil - 1} \dots r_1 \, r_0$ — у такому випадку при додаванні старших бітів h_2^* та h_2 не виникає біту перенесення, який би мав враховуватись при додаванні молодших бітів h_1^* та h_1, а отже значення h_1^* мало би змінитись для виконання умови $A + B = 0 \bmod CR_{n,c}$. Оскільки біту перенесення немає, то $h_1 + h_1^* = 11 \dots 1$, отже, маємо:

$$Ham(h_1^*) = |h_1| - Ham(h_1).$$

Оскільки A — n-бітове число, а $|h_2| = \lceil \log_2 c \rceil$, то $|h_1| = n - \lceil \log_2 c \rceil$. Тоді маємо:

$$Ham(-A \bmod CR_{n,c}) = Ham(B) = Ham(h_1^*) + Ham(h_2^*) =$$

$$= n - \lceil \log_2 c \rceil - Ham(h_1) + Ham(h_2^*).$$

2) Аналогічно, якщо

$$B = -A \bmod CR_{n,c} = h_1^* \,||\, h_2^* \,||\, h_3^*,$$

то виконується співвідношення

$$Ham(B) = Ham(h_1^*) + Ham(h_2^*) + Ham(h_3^*).$$

У даному випадку $Ham(h_3^*) = z$.

Оскільки $a_{\lceil \log_2 c \rceil - 1} \dots a_1 \ a_0$ є більшим за $r_{\lceil \log_2 c \rceil - 1} \dots r_1 \ r_0$, то при додаванні старших бітів h_3 та h_3^* з'являється біт перенесення. Враховуючи те, що $r_{n-1} \ r_n \dots r_{\lceil \log_2 c \rceil} = 11 \dots 1$, біт перенесення буде з'являтись на кожному кроці до того моменту, як у A зустрінеться 0 — таким чином утворюється частина h_2.

Зрозуміло, що на позиціях i, де $a_i = 1$, буде $b_i = 1$, а при першій зустрічі нуля (позначимо його $a_t = 0$) буде $b_t = 0$. Маємо рівність $h_2 = h_2^*$. Отже, $Ham(h_2^*) = Ham(h_2)$.

$Ham(h_1^*)$ отримується аналогічно як у доведенні пункту 1), тобто

$$Ham(h_1^*) = |h_1| - Ham(h_1) = n - t - 1 - Ham(h_1).$$

Підсумовуючи отримане, маємо:

$$Ham(-A \bmod CR_{n,c}) = Ham(B) = Ham(h_1^*) + Ham(h_2^*) + Ham(h_3^*) =$$

$$= |h_1| - Ham(h_1) + Ham(h_2) + Ham(h_3^*) =$$

$$= n - \lceil \log_2 c \rceil - |h_2| - Ham(h_1) + Ham(h_2) + Ham(h_3^*),$$

що і потрібно було довести.

□

ДОДАТОК Б ГРАФІКИ РОЗПОДІЛУ ПАРАМЕТРА РОЗШИФРУВАННЯ У КРИПТОСИСТЕМІ AJPS-1 ТА ЇЇ МОДИФІКАЦІЯХ

У додатку наведено результати експериментального дослідження криптосистеми AJPS-1 та її модифікацій, а саме модифікації AJPS-1 з використанням метрики OSD (різниця кількості одиниць та кількості нулів в двійковому представленні числа), модифікації AJPS-1 з використанням арифметики за модулем узагальненого числа Мерсенна

$$GM_{n,m} = 2^n - 2^m - 1, \text{ де } n, m \in \mathbb{N} \text{ та } m < n$$

та модифікації AJPS-1 з використанням арифметики за модулем числа Кренделла

$$CR_{n,c} = 2^n - c, \text{ де } n, c \in \mathbb{N} \text{ та } \log_2 c \leqslant \frac{n}{2}.$$

При виконанні даного дослідження здійснюється порівняння розподілу значень d та s, які обчислюються при процедурі розшифрування криптосистеми AJPS-1 та описаних модифікацій криптосистеми AJPS-1, для рекомендованих відповідно до таблиці 2.5 значень параметрів n та h.

Б.1 Порівняння розподілів значень d і s у криптосистемі AJPS-1 та модифікації AJPS-1 з використанням метрики OSD

Розглянемо розподіл значень d і s криптосистеми AJPS-1 та модифікації криптосистеми AJPS-1 з використанням метрики OSD, яка представлена у розділі 2. Результати отримано експериментально при серії з 1000000 застосувань алгоритмів шифрування та розшифрування кожної з криптосистем при різних значеннях n та h.

1) Нехай $n = 1279$ та $h = 17$, тоді при значенні біту $b = 0$ маємо:

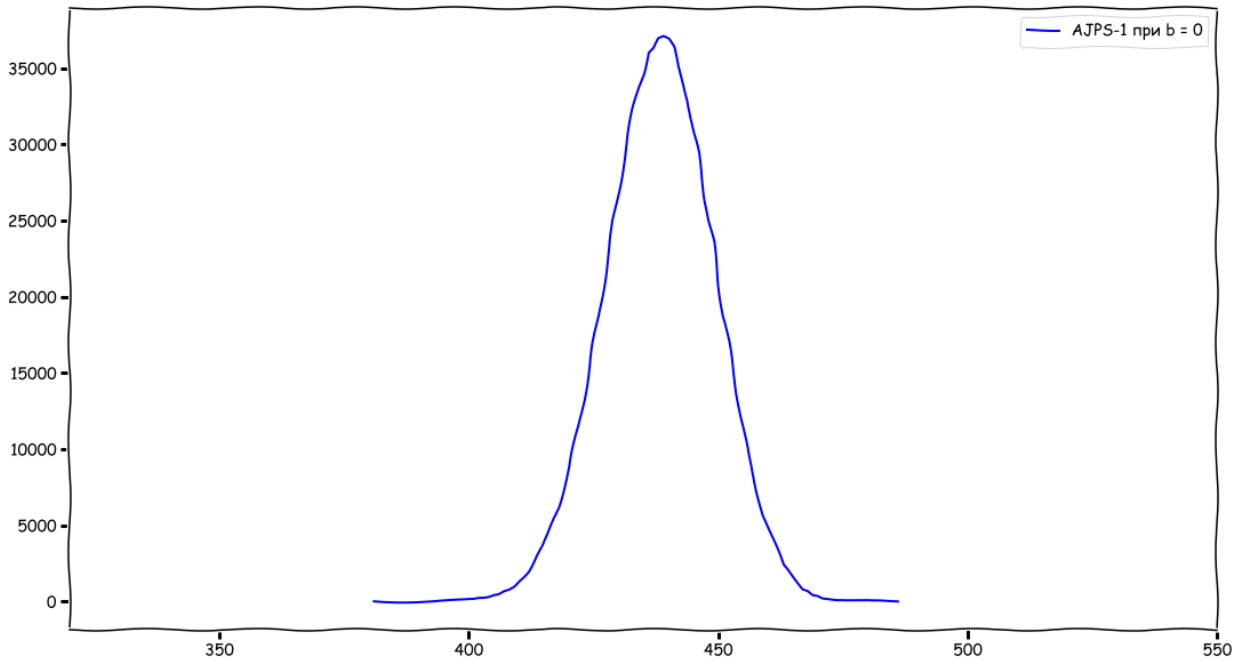

Рисунок Б.1 – Розподіл значення d в криптосистемі AJPS-1 при значеннях $n = 1279$, $h = 17$ та $b = 0$

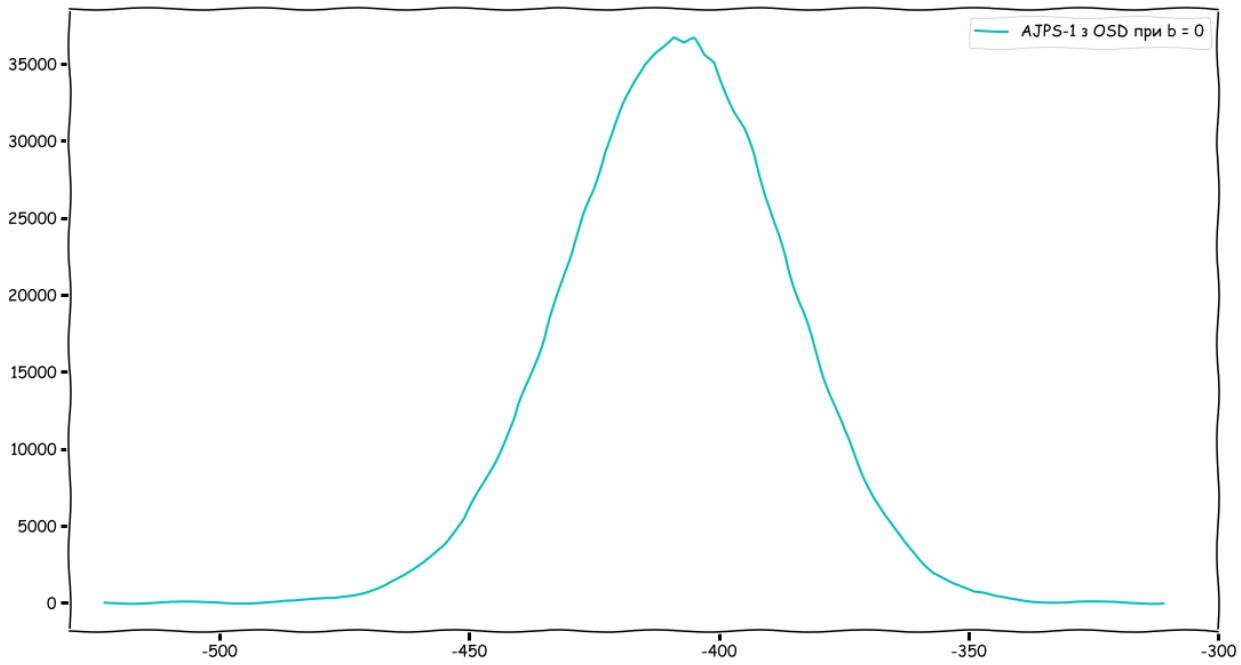

Рисунок Б.2 – Розподіл значення s в модифікації криптосистеми AJPS-1 з використанням метрики OSD при $n = 1279$, $h = 17$ та $b = 0$

Та при значенні біту $b = 1$ отримуємо:

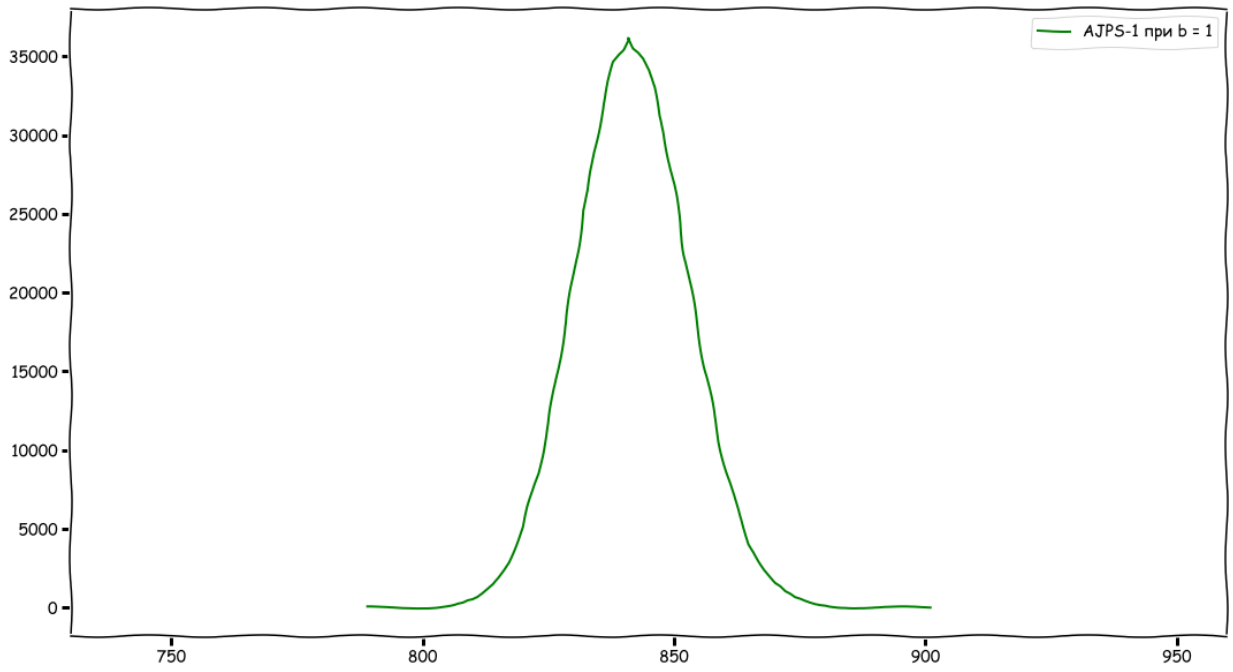

Рисунок Б.3 – Розподіл значення d в криптосистемі AJPS-1 при значеннях $n = 1279$, $h = 17$ та $b = 1$

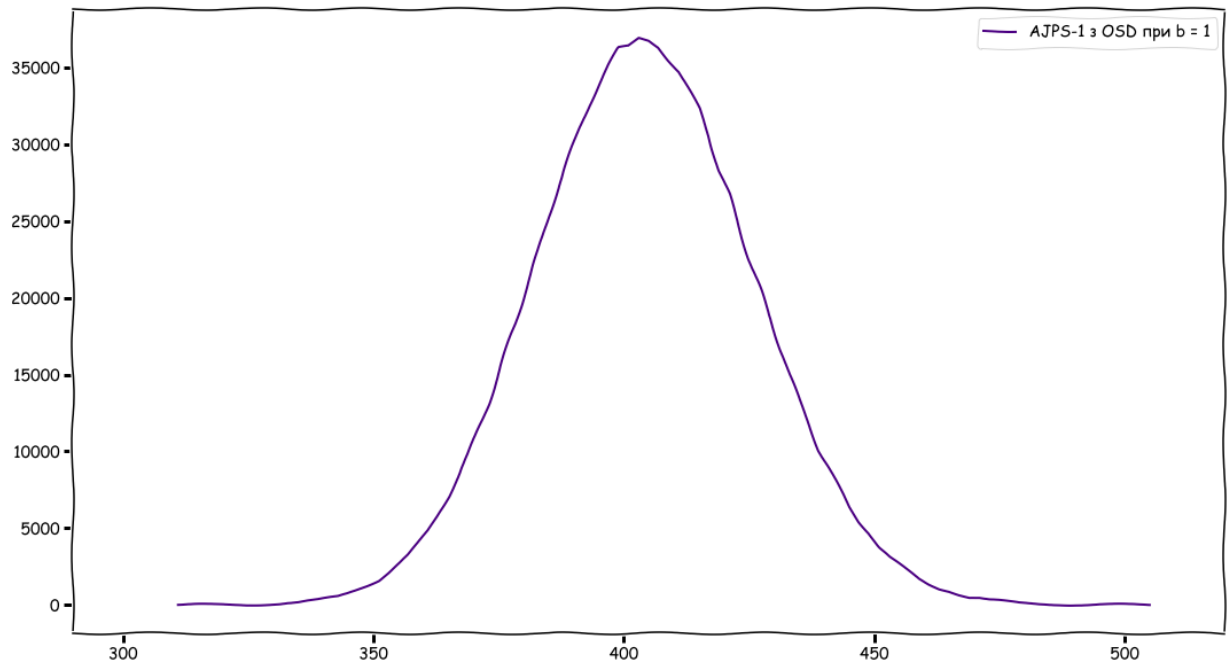

Рисунок Б.4 – Розподіл значення s в модифікації криптосистеми AJPS-1 з використанням метрики OSD при $n = 1279$, $h = 17$ та $b = 1$

2) Нехай $n = 2203$ та $h = 23$, тоді при значенні біту $b = 0$ маємо:

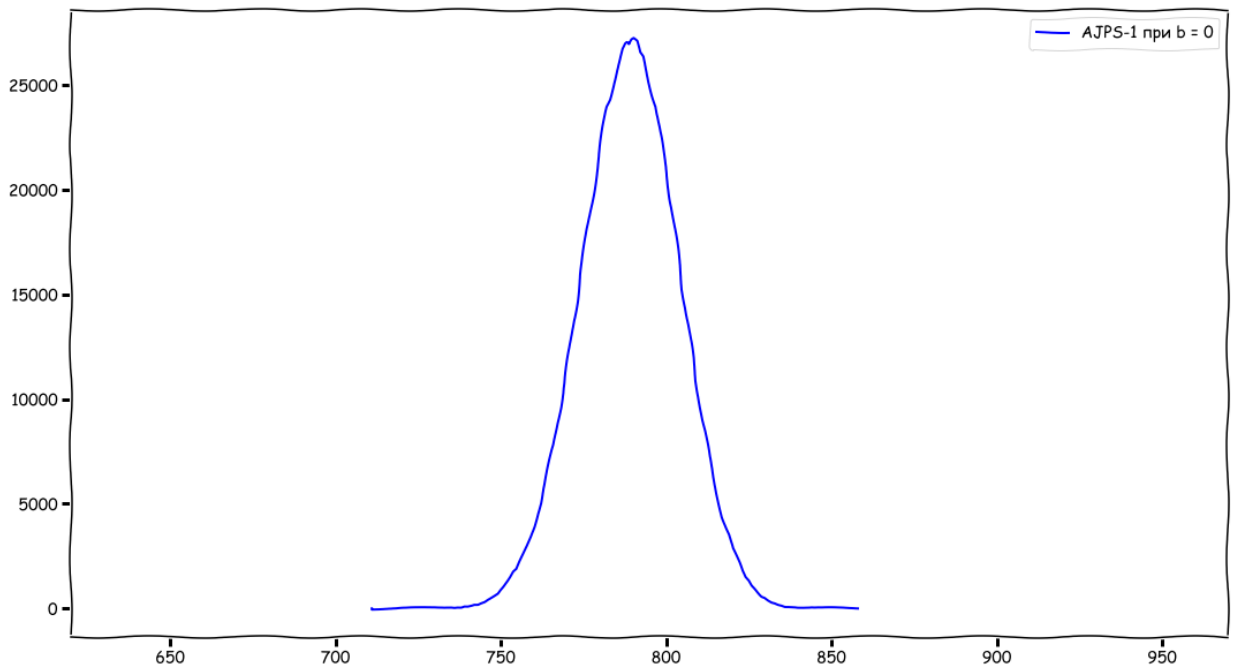

Рисунок Б.5 – Розподіл значення d в криптосистемі AJPS-1 при значеннях $n = 2203$, $h = 23$ та $b = 0$

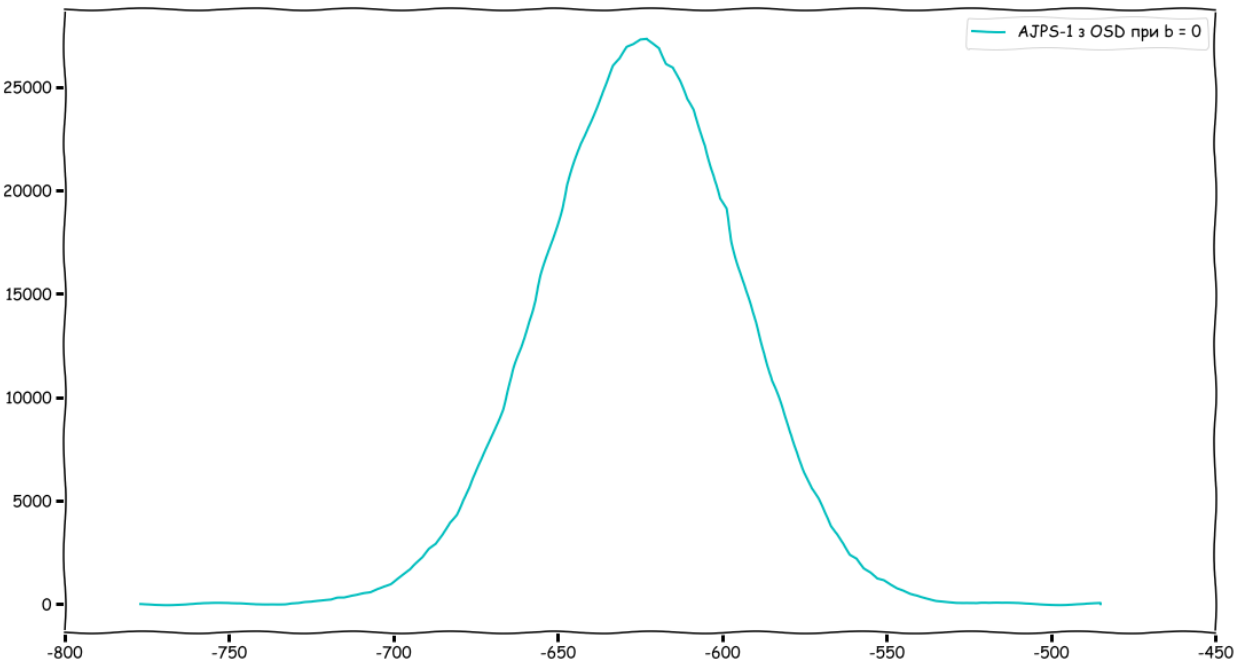

Рисунок Б.6 – Розподіл значення s в модифікації криптосистеми AJPS-1 з використанням метрики OSD при $n = 2203$, $h = 23$ та $b = 0$

Та при значенні біту $b = 1$ отримуємо:

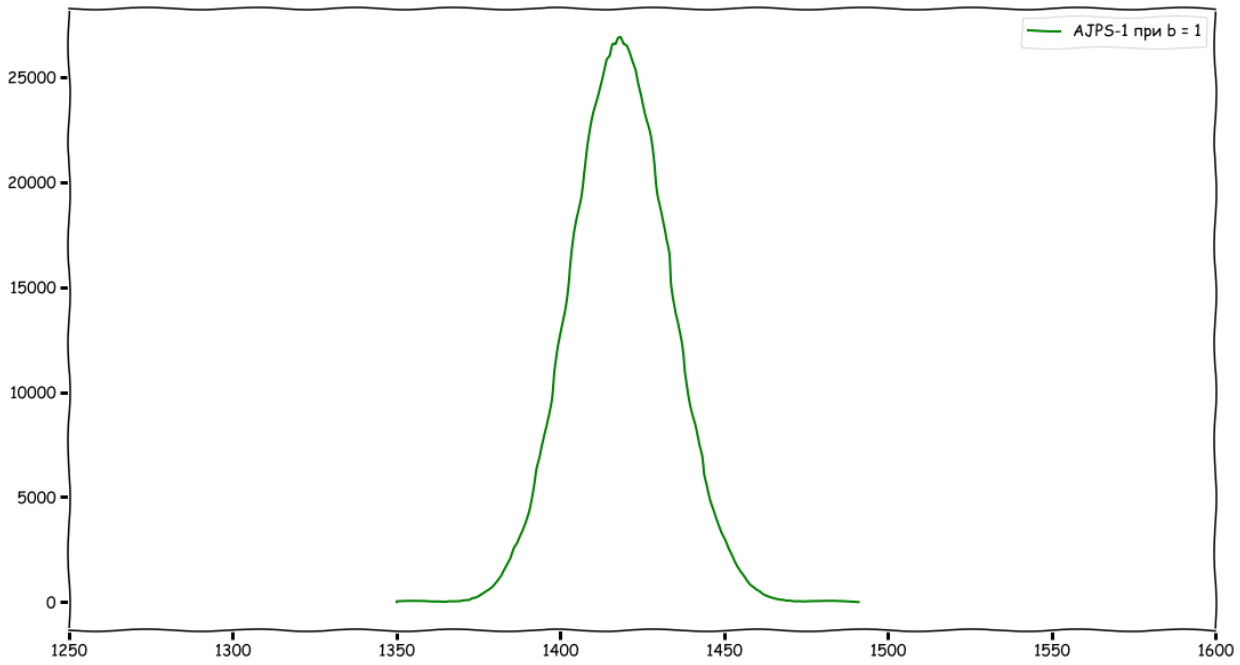

Рисунок Б.7 – Розподіл значення d в криптосистемі AJPS-1 при значеннях $n = 2203$, $h = 23$ та $b = 1$

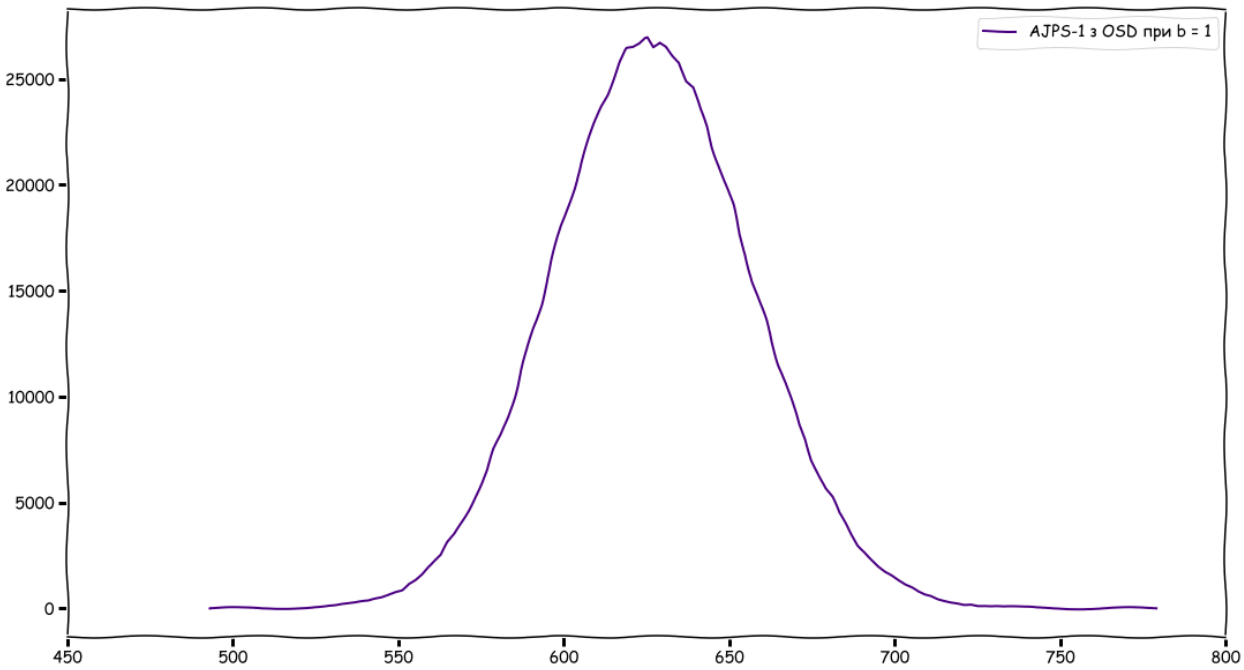

Рисунок Б.8 – Розподіл значення s в модифікації криптосистеми AJPS-1 з використанням метрики OSD при $n = 2203$, $h = 23$ та $b = 1$

3) Нехай $n = 3217$ та $h = 28$, тоді при значенні біту $b = 0$ маємо:

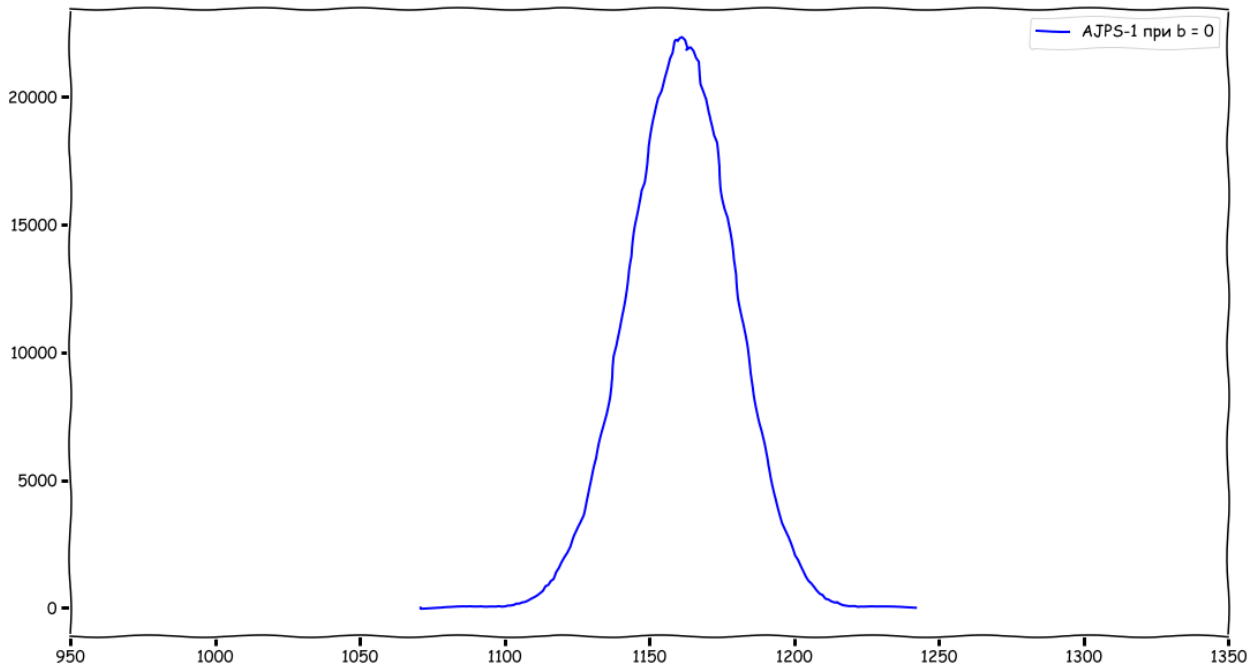

Рисунок Б.9 – Розподіл значення d в криптосистемі AJPS-1 при значеннях $n = 3217$, $h = 28$ та $b = 0$

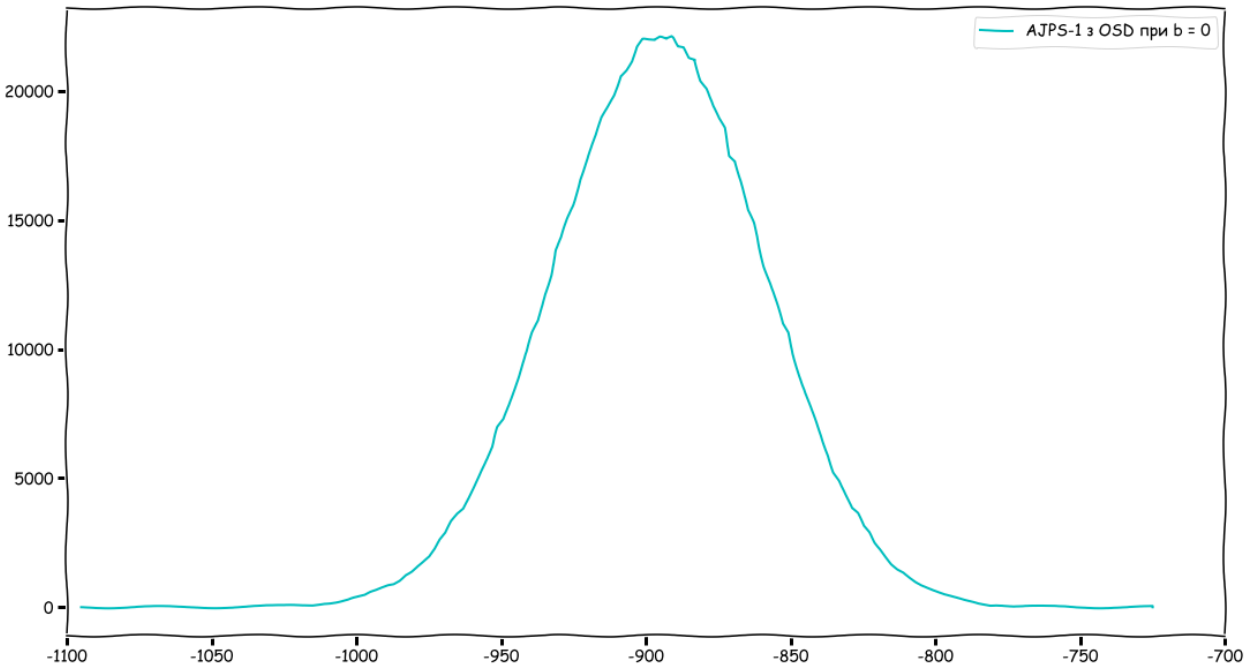

Рисунок Б.10 – Розподіл значення s в модифікації криптосистеми AJPS-1 з використанням метрики OSD при $n = 3217$, $h = 28$ та $b = 0$

Та при значенні біту $b = 1$ отримуємо:

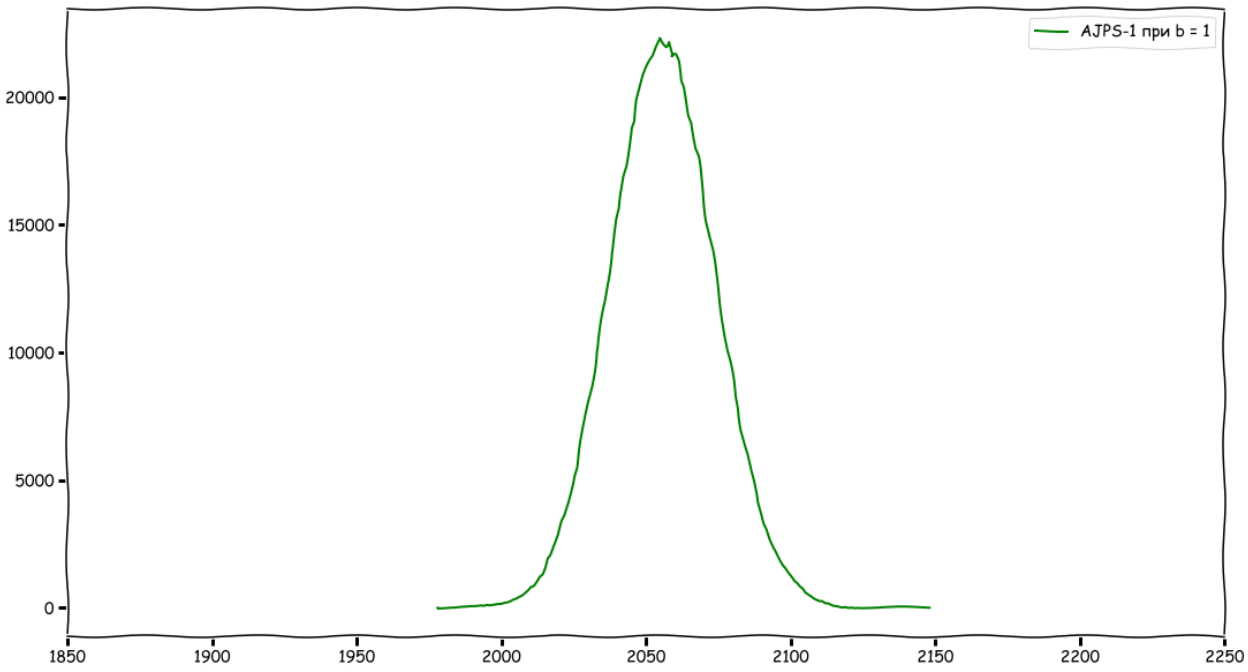

Рисунок Б.11 – Розподіл значення d в криптосистемі AJPS-1 при значеннях $n = 3217$, $h = 28$ та $b = 1$

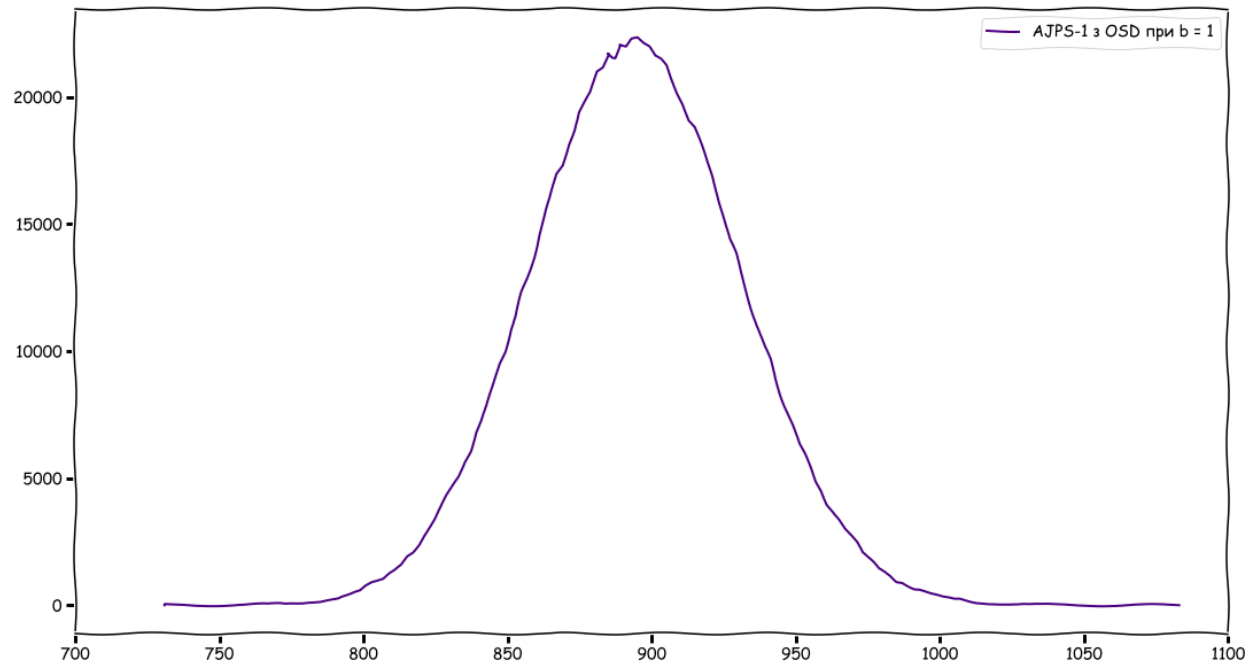

Рисунок Б.12 – Розподіл значення s в модифікації криптосистеми AJPS-1 з використанням метрики OSD при $n = 3217$, $h = 28$ та $b = 1$

4) Нехай $n = 4253$ та $h = 32$, тоді при значенні біту $b = 0$ маємо:

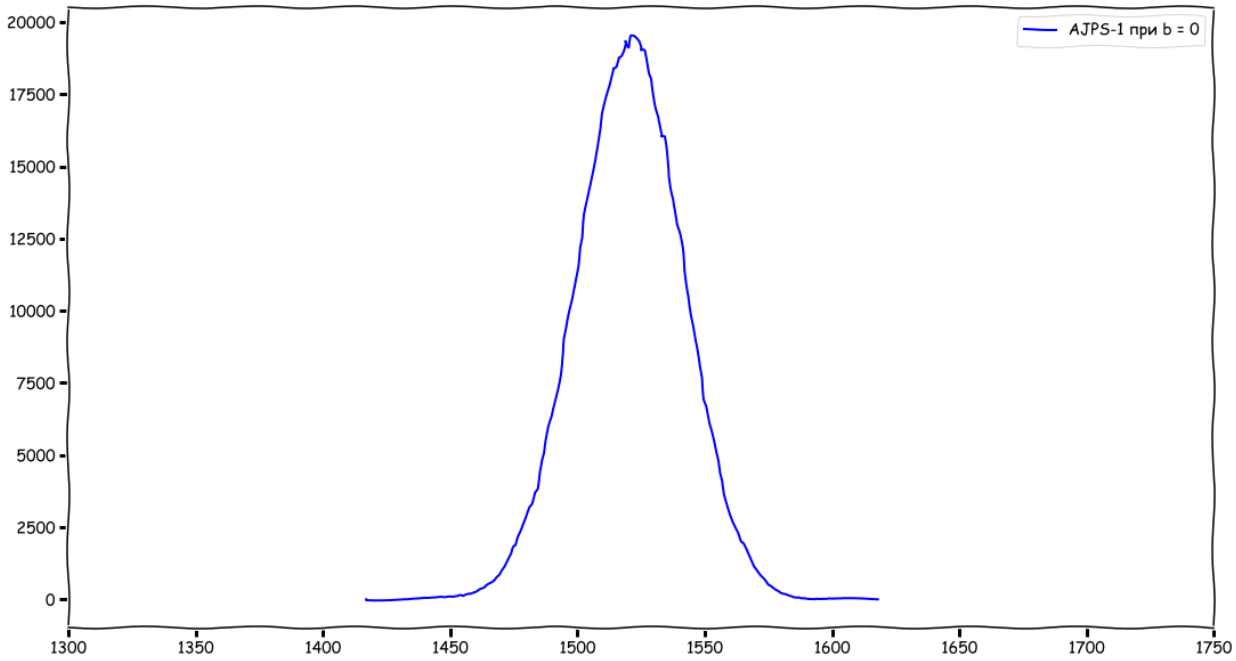

Рисунок Б.13 – Розподіл значення d в криптосистемі AJPS-1 при значеннях $n = 4253$, $h = 32$ та $b = 0$

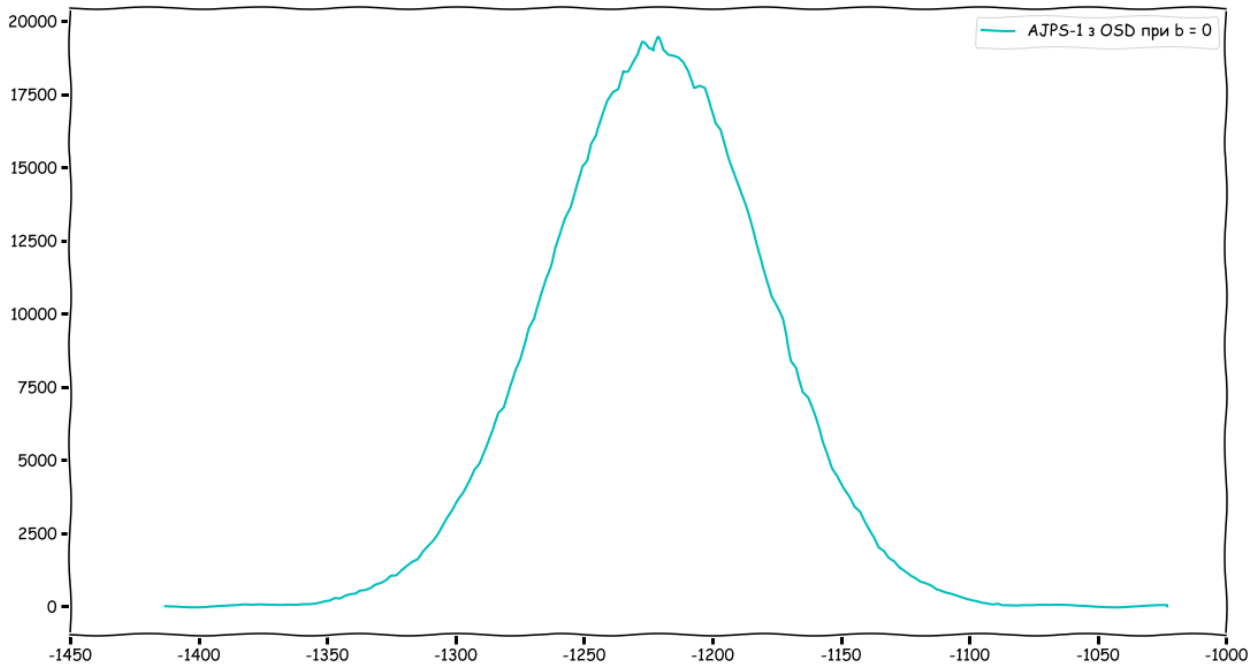

Рисунок Б.14 – Розподіл значення s в модифікації криптосистеми AJPS-1 з використанням метрики OSD при $n = 4253$, $h = 32$ та $b = 0$

Та при значенні біту $b = 1$ отримуємо:

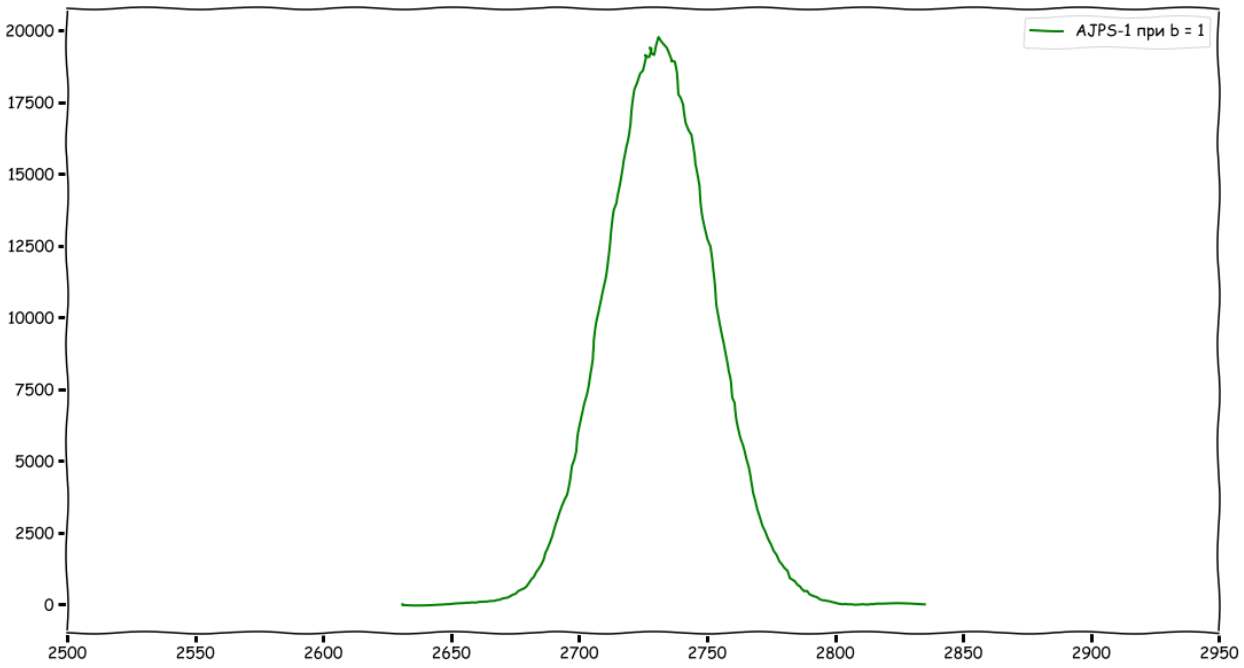

Рисунок Б.15 – Розподіл значення d в криптосистемі AJPS-1 при значеннях $n = 4253$, $h = 32$ та $b = 1$

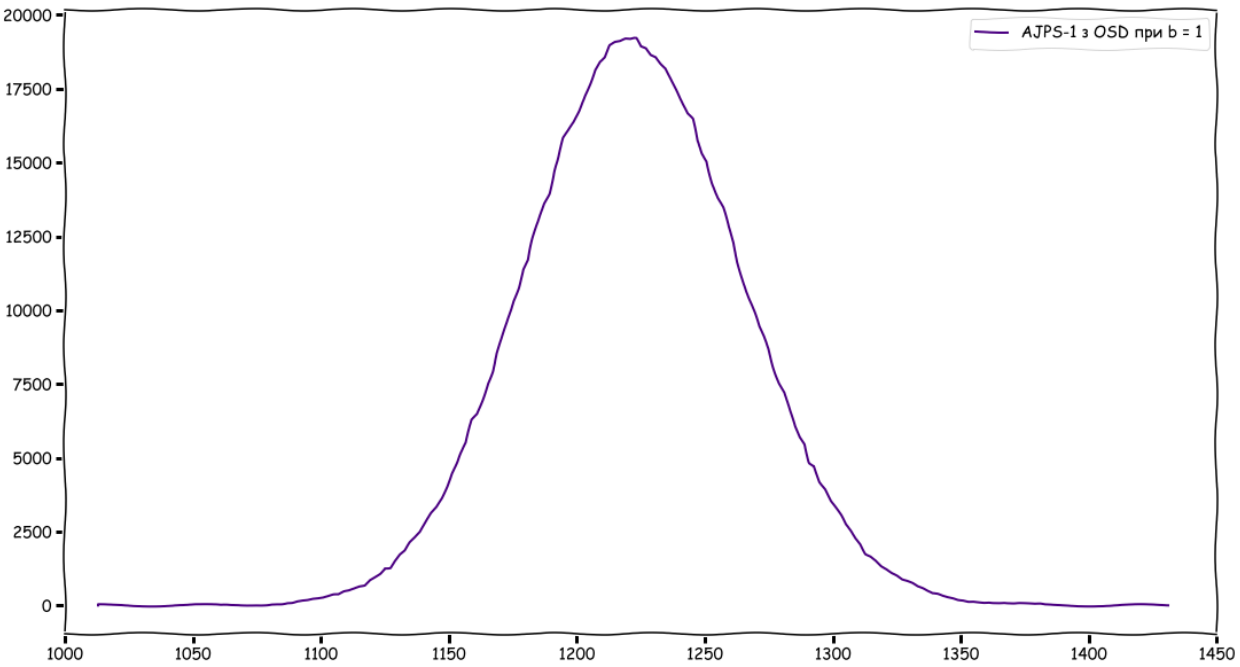

Рисунок Б.16 – Розподіл значення s в модифікації криптосистеми AJPS-1 з використанням метрики OSD при $n = 4253$, $h = 32$ та $b = 1$

Б.2 Порівняння розподілу значення d в AJPS-1 та модифікації AJPS-1 з використанням арифметики за модулем узагальненого числа Мерсенна

Розглянемо розподіл значення d при застосуванні алгоритму розшифрування криптосистеми AJPS-1 та модифікації криптосистеми AJPS-1 з використанням арифметики за модулем узагальненого числа Мерсенна $GM_{n,m}$, яка представлена у розділі 2. Дані результати отримано експериментально при серії з 1000000 застосувань алгоритмів шифрування та розшифрування кожної з описаних криптосистем при різних значеннях параметрів n та h та фіксованому значенні $m = 25$.

1) Нехай $n = 1279$ та $h = 17$, тоді при значенні біту $b = 0$ маємо:

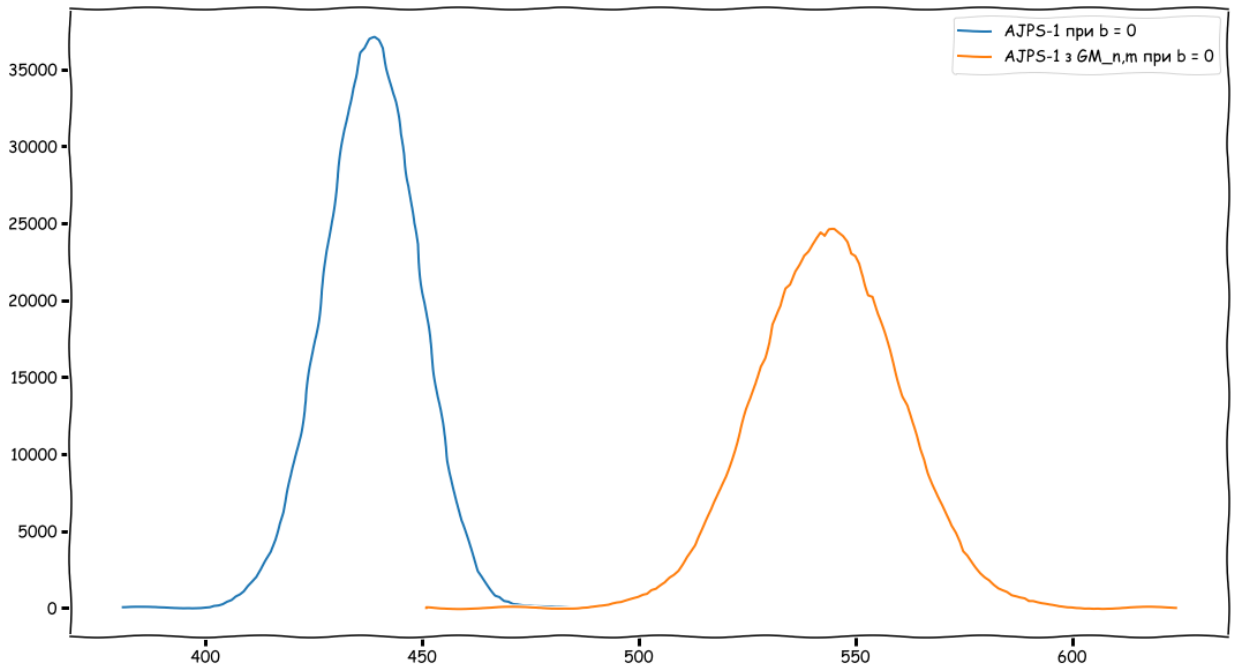

Рисунок Б.17 – Розподіл значення d в криптосистемі AJPS-1 та модифікації криптосистеми AJPS-1 з використанням арифметики за модулем узагальненого числа Мерсенна GM_n, m при значеннях $n = 1279$, $h = 17$ та $b = 0$

Та при значенні біту $b = 1$ отримуємо:

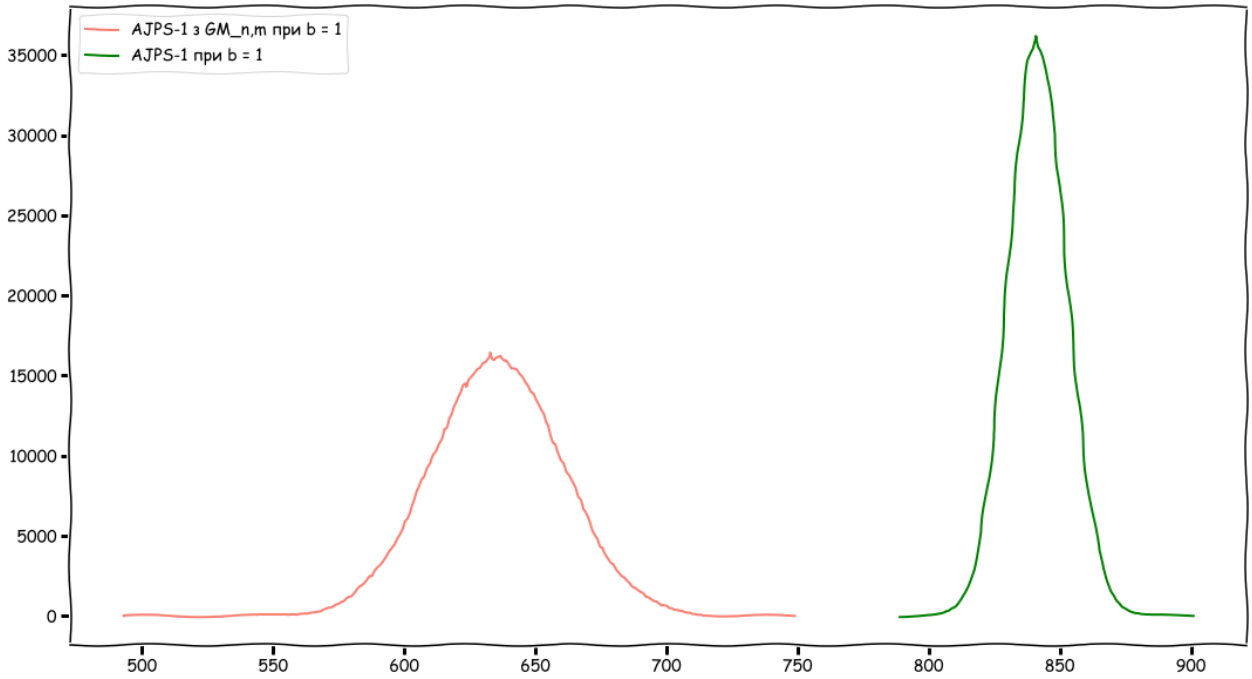

Рисунок Б.18 – Розподіл значення d в криптосистемі AJPS-1 та модифікації криптосистеми AJPS-1 з використанням арифметики за модулем узагальненого числа Мерсенна GM_n, m при значеннях параметрів $n = 1279$, $h = 17$ та $b = 1$

2) Нехай $n = 2203$ та $h = 23$, тоді при значеннях біту $b = 0$ та $b = 1$ відповідно маємо наступні розподіли значення d:

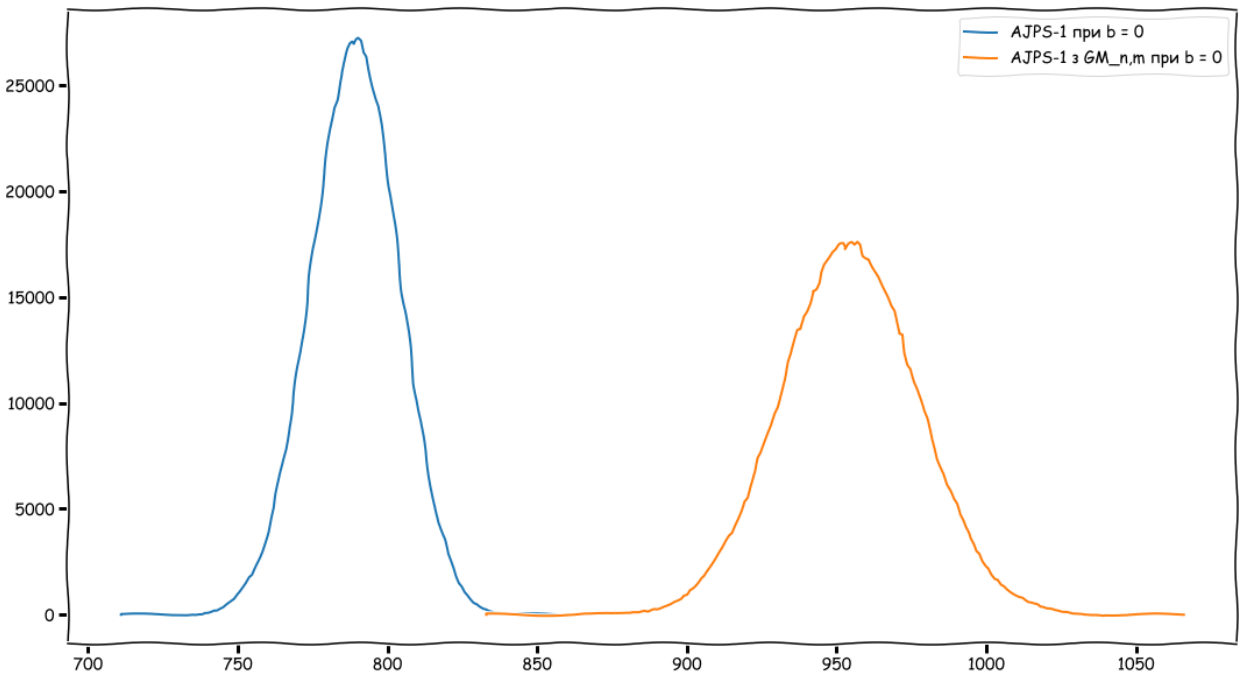

Рисунок Б.19 – Розподіл значення d в криптосистемі AJPS-1 та модифікації AJPS-1 з використанням арифметики за модулем узагальненого числа Мерсенна при значеннях $n = 2203$, $h = 23$ та $b = 0$

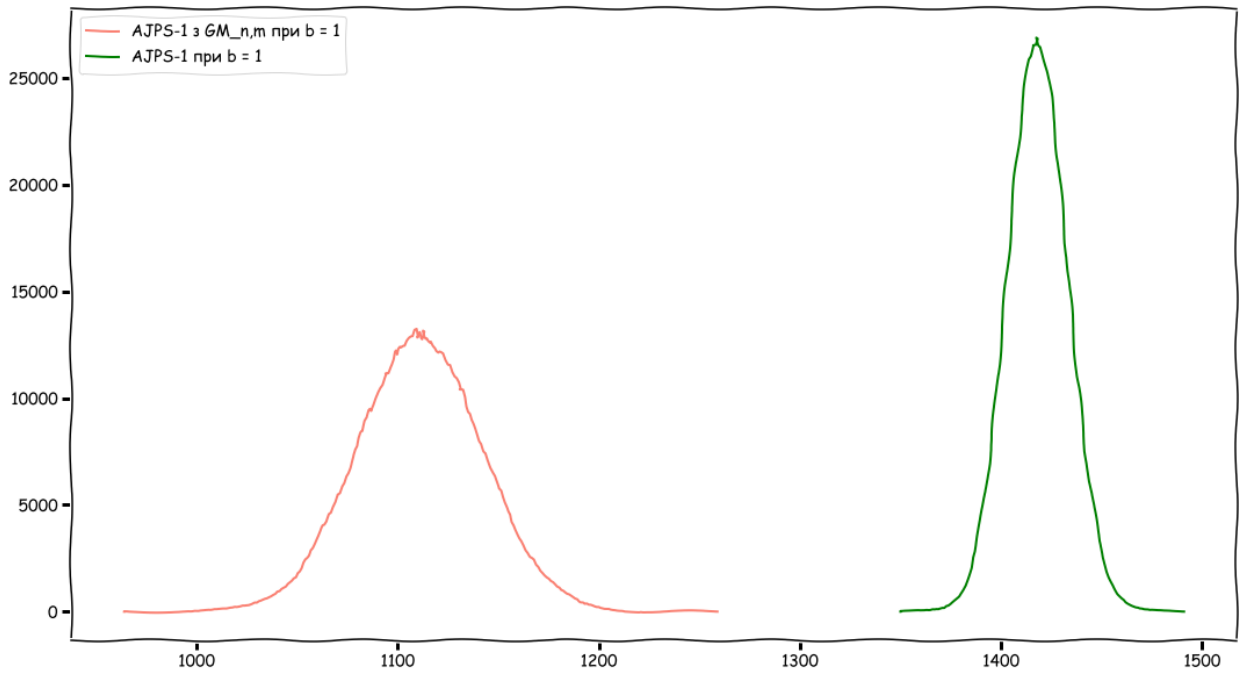

Рисунок Б.20 – Розподіл значення d в криптосистемі AJPS-1 та модифікації AJPS-1 з використанням арифметики за модулем узагальненого числа Мерсенна при значеннях $n = 2203$, $h = 23$ та $b = 1$

3) Нехай $n = 3217$ та $h = 28$, тоді, відповідно до значення b, маємо:

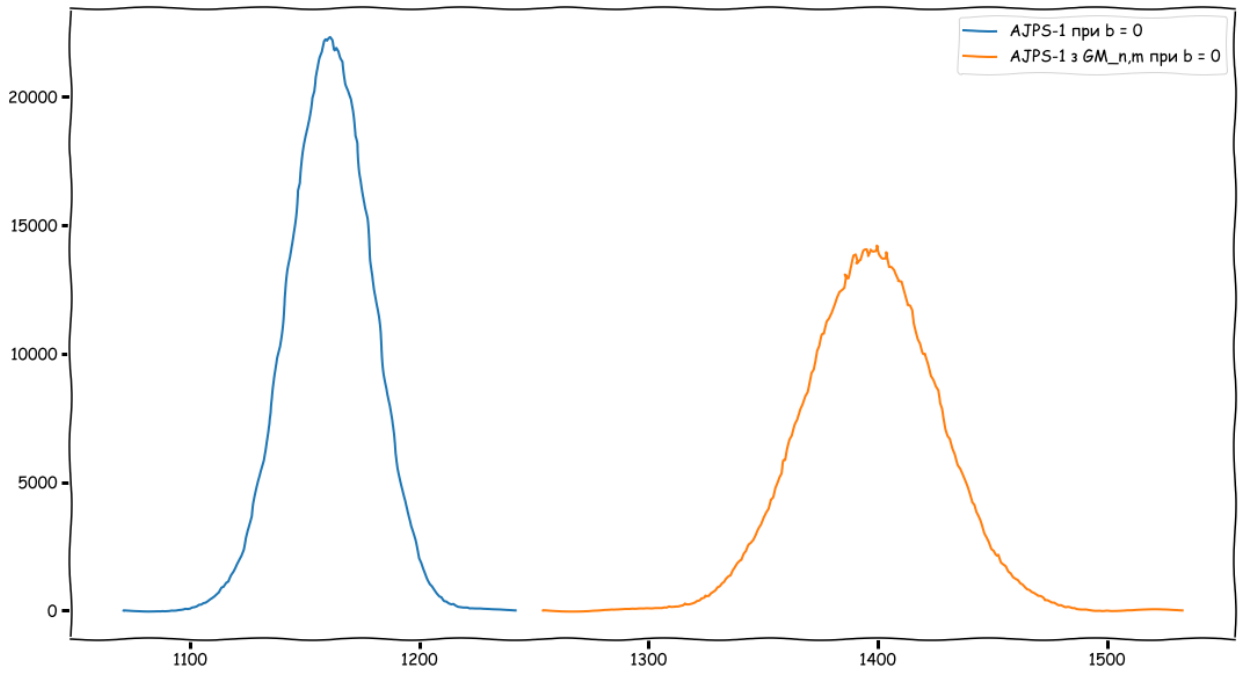

Рисунок Б.21 – Розподіл d в AJPS-1 та її модифікації з використанням арифметики за модулем $GM_{n,m}$ при $n = 3217$, $h = 28$ та $b = 0$

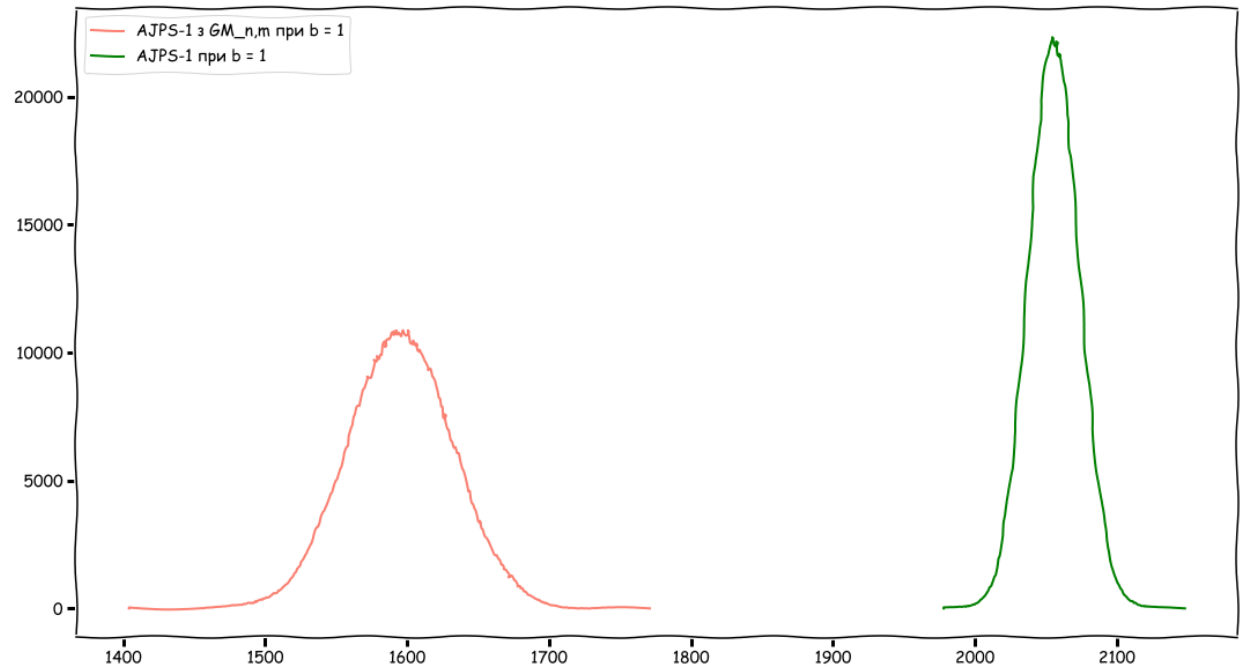

Рисунок Б.22 – Розподіл d в AJPS-1 та її модифікації з використанням арифметики за модулем $GM_{n,m}$ при $n = 3217$, $h = 28$ та $b = 1$

4) Нехай $n = 4253$ та $h = 32$, тоді, відповідно значення b, маємо:

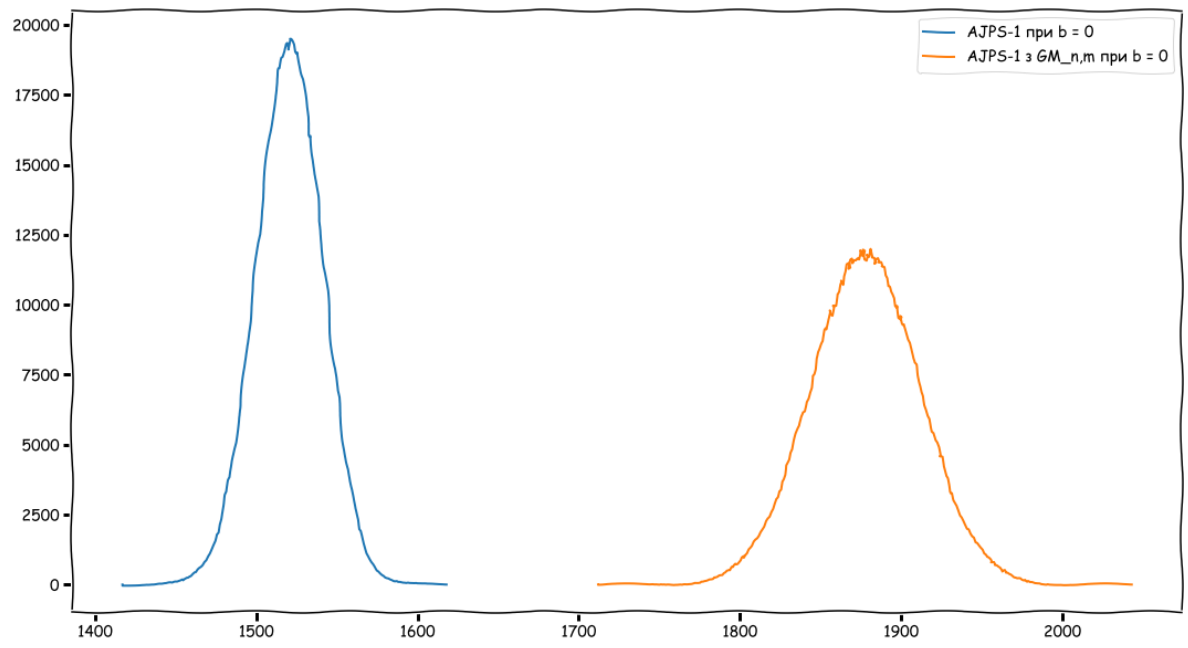

Рисунок Б.23 – Розподіл d в AJPS-1 та її модифікації з використанням арифметики за модулем $GM_{n,m}$ при $n = 4253$, $h = 32$ та $b = 0$

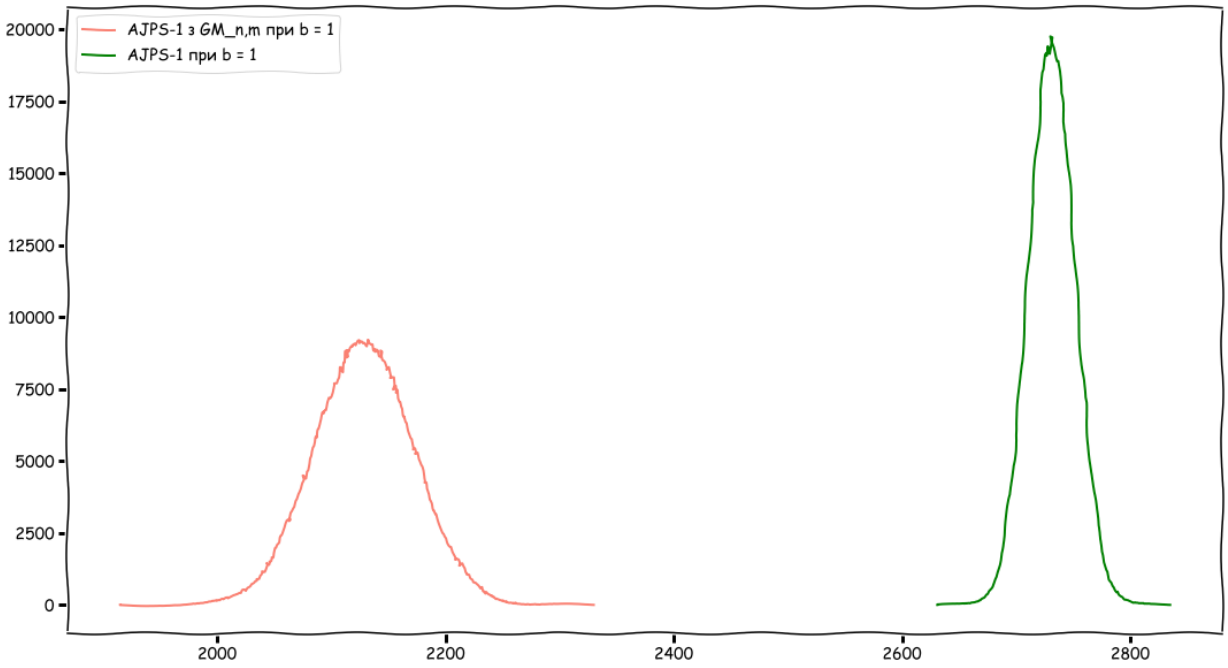

Рисунок Б.24 – Розподіл d в AJPS-1 та її модифікації, яка використовує арифметику за модулем $GM_{n,m}$, при $n = 4253$, $h = 32$ та $b = 1$

Б.3 Порівняння розподілу значення d в криптосистемі AJPS-1 та модифікації AJPS-1 з використанням арифметики за модулем числа Кренделла

Розглянемо розподіл значення d при застосуванні алгоритму розшифрування криптосистеми AJPS-1 та модифікації криптосистеми AJPS-1 з використанням арифметики за модулем числа Кренделла $CR_{n,c}$, яка представлена у розділі 2. Дані результати отримано експериментально при серії з 1000000 застосувань алгоритму розшифрування кожної з описаних криптосистем при різних значеннях параметрів n та h та фіксованому значенні $c = 15$.

1) Нехай $n = 1279$ та $h = 17$, тоді при значенні біту $b = 0$ маємо:

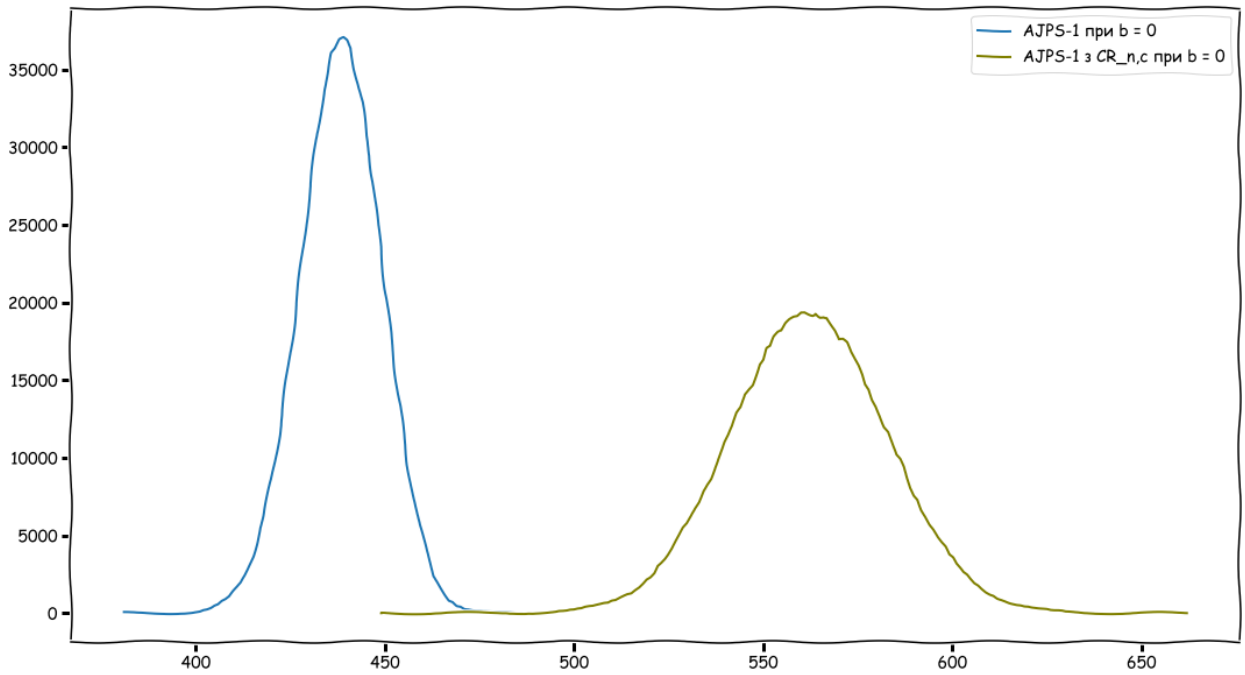

Рисунок Б.25 – Розподіл значення d в криптосистемі AJPS-1 та модифікації криптосистеми AJPS-1 з використанням арифметики за модулем числа Кренделла $CR_{n,c}$ при значеннях $n = 1279$, $h = 17$ та $b = 0$

Та при значенні біту $b = 1$ отримуємо:

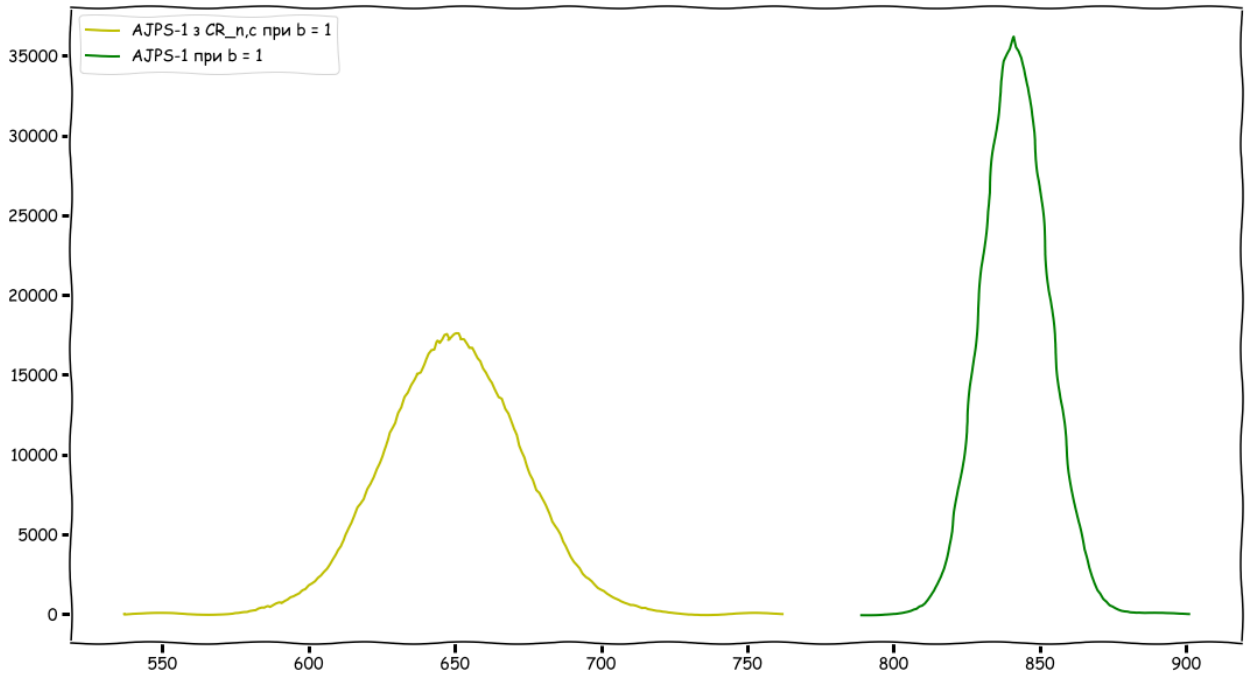

Рисунок Б.26 – Розподіл значення d в криптосистемі AJPS-1 та модифікації криптосистеми AJPS-1 з використанням арифметики за модулем числа Кренделла $CR_{n,c}$ при значеннях параметрів $n = 1279$, $h = 17$ та $b = 1$

2) Нехай $n = 2203$ та $h = 23$, тоді при значеннях біту $b = 0$ та $b = 1$ відповідно маємо наступні розподіли значення d:

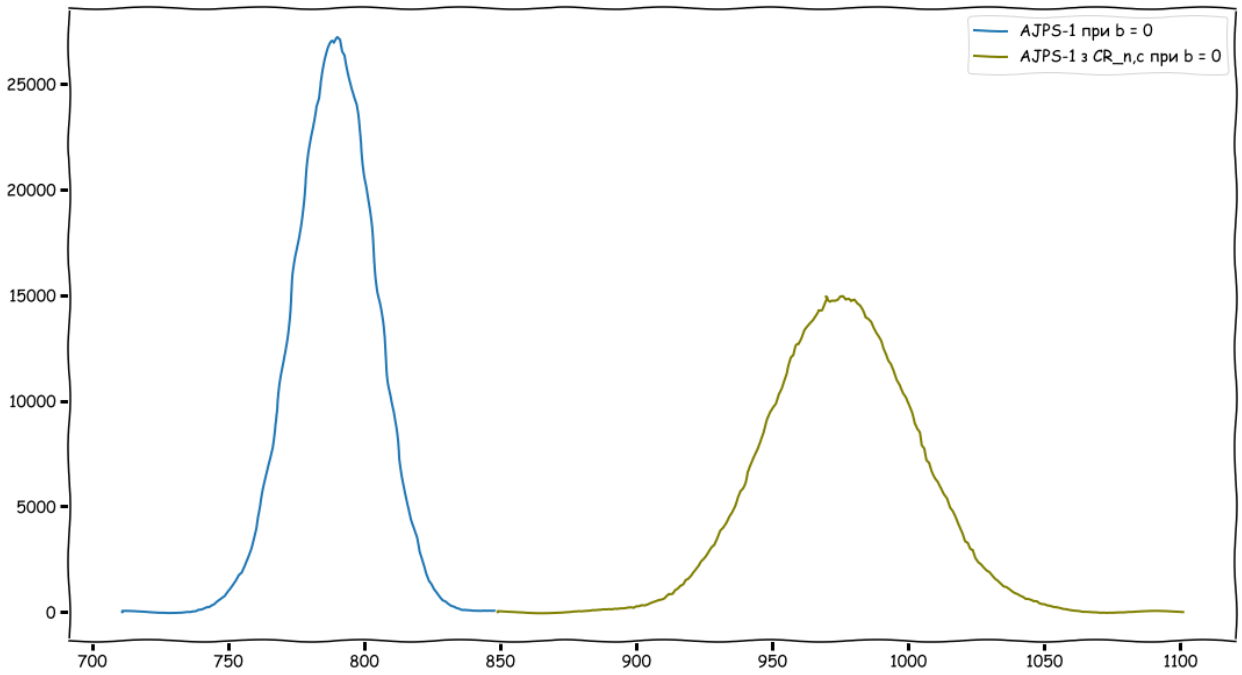

Рисунок Б.27 – Розподіл значення d в криптосистемі AJPS-1 та модифікації AJPS-1 з використанням арифметики за модулем числа Кренделла при значеннях $n = 2203$, $h = 23$ та $b = 0$

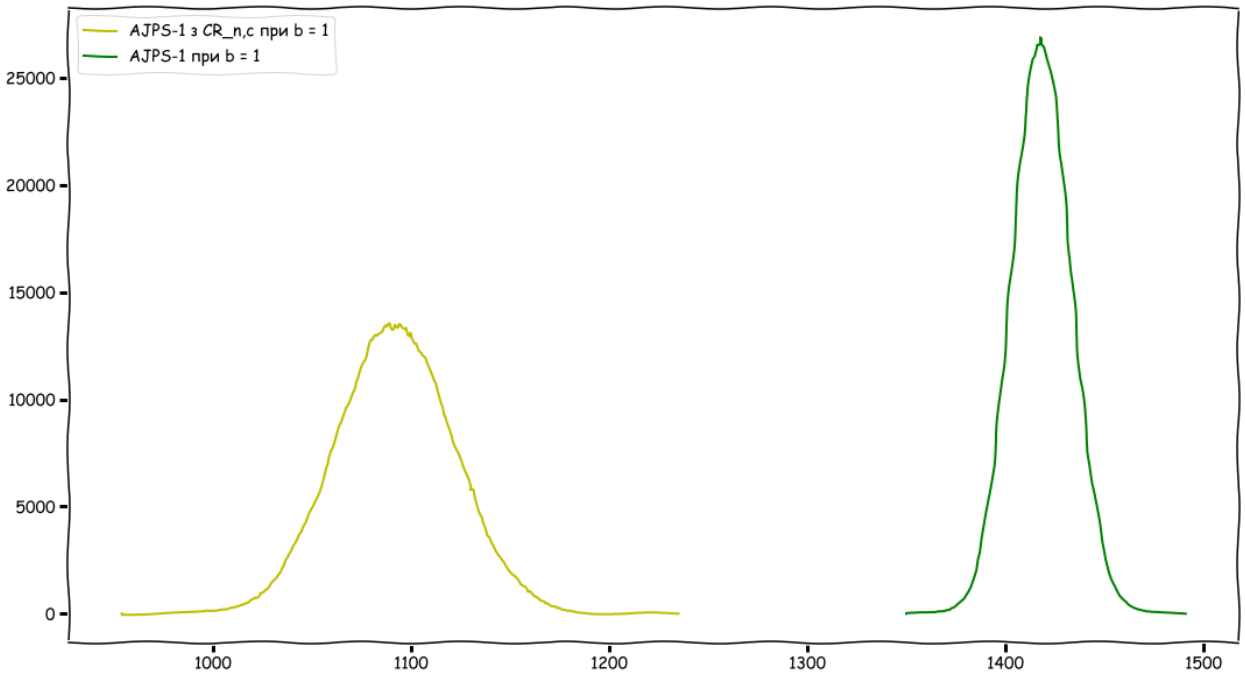

Рисунок Б.28 – Розподіл значення d в криптосистемі AJPS-1 та модифікації AJPS-1 з використанням арифметики за модулем числа Кренделла при значеннях $n = 2203$, $h = 23$ та $b = 1$

3) Нехай $n = 3217$ та $h = 28$, тоді, відповідно до значення b, маємо:

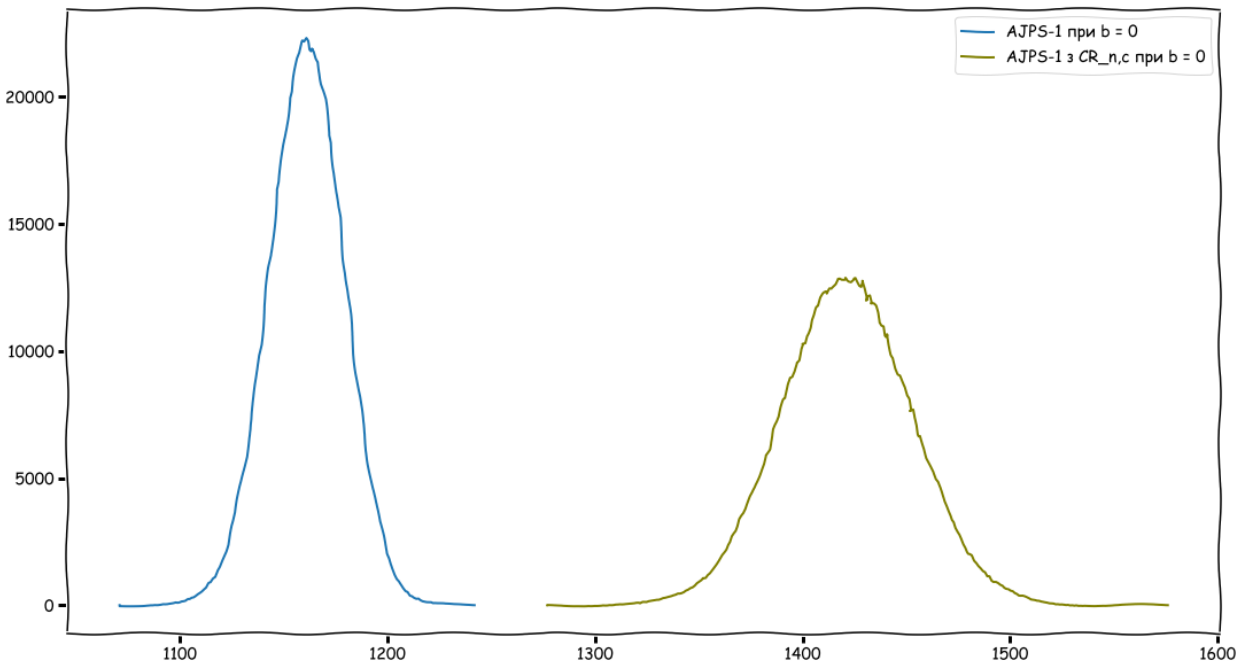

Рисунок Б.29 – Розподіл d в AJPS-1 та її модифікації з використанням арифметики за модулем $CR_{n,c}$ при $n = 3217$, $h = 28$ та $b = 0$

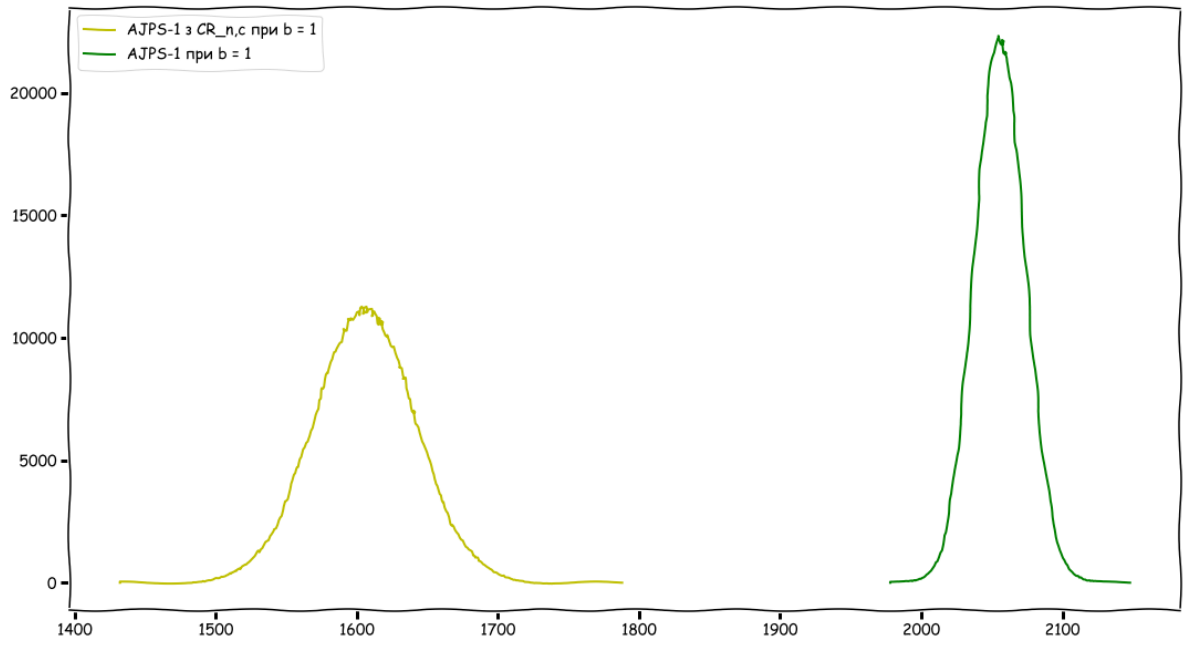

Рисунок Б.30 – Розподіл d в AJPS-1 та її модифікації з використанням арифметики за модулем $CR_{n,c}$ при $n = 3217$, $h = 28$ та $b = 1$

4) Нехай $n = 4253$ та $h = 32$, тоді, відповідно значення b, маємо:

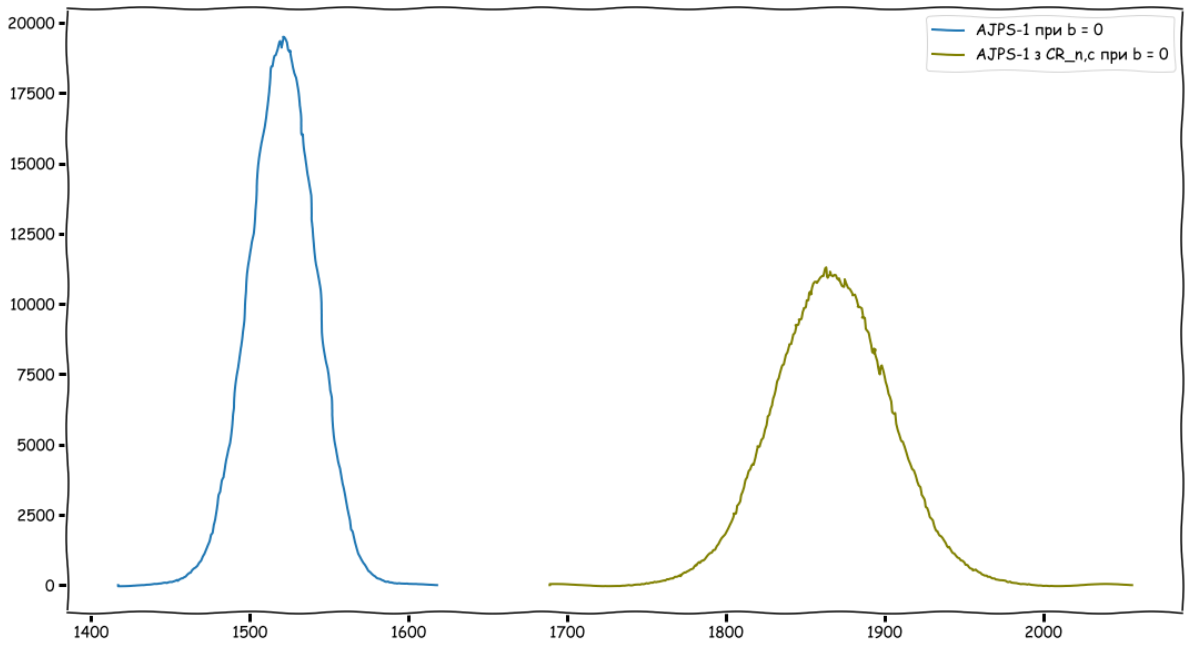

Рисунок Б.31 – Розподіл d в AJPS-1 та її модифікації з використанням арифметики за модулем $CR_{n,c}$ при $n = 4253$, $h = 32$ та $b = 0$

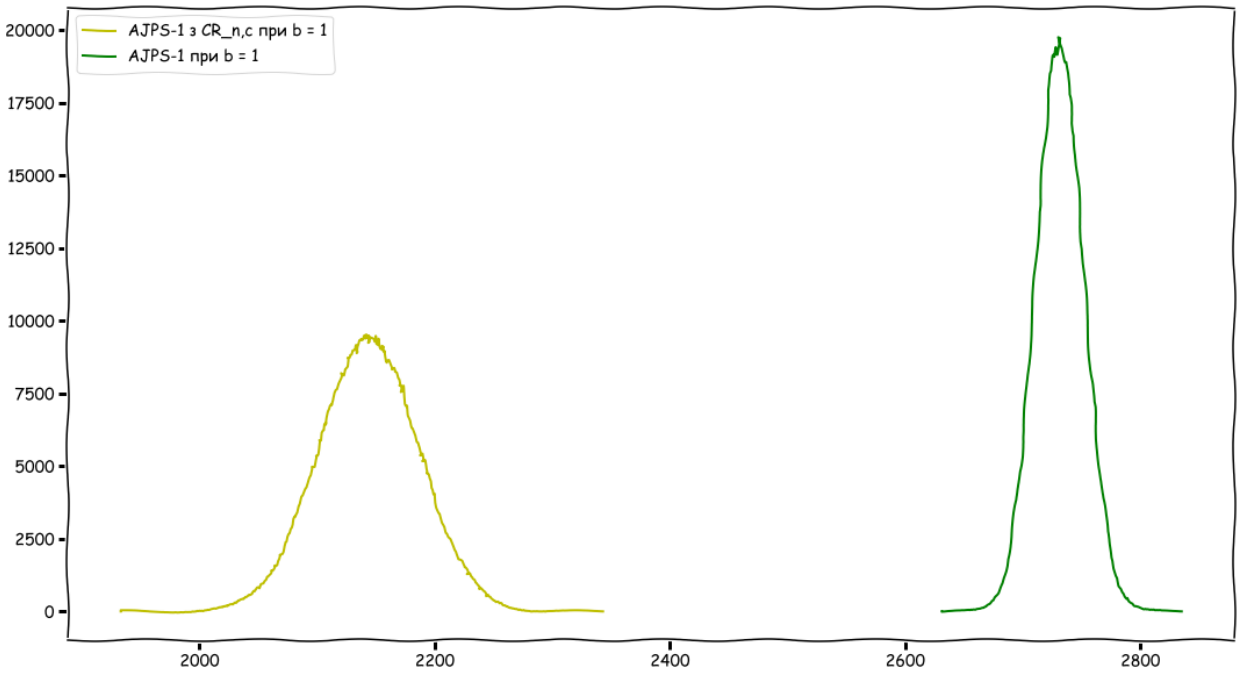

Рисунок Б.32 – Розподіл d в AJPS-1 та її модифікації, яка використовує арифметику за модулем $CR_{n,c}$, при $n = 4253$, $h = 32$ та $b = 1$

Б.4 Порівняння розподілу значення d у модифікаціях криптосистеми AJPS-1 з використанням арифметики за модулем узагальненого числа Мерсенна та з використанням арифметики за модулем числа Кренделла

Розглянемо розподіл d при модифікаціях криптосистеми AJPS-1 з використанням арифметики за модулем узагальненого числа Мерсенна $GM_{n,m}$ та з використанням арифметики за модулем числа Кренделла $CR_{n,c}$. Обидві описані модифікації запропоновані у розділі 2. Дані результати отримано експериментально при серії з 1000000 застосувань алгоритму розшифрування при різних значеннях параметрів n та h та фіксованих значеннях $m = 25$ та $c = 15$.

1) Нехай $n = 2203$ та $h = 23$, тоді маємо такі розподіли значення d:

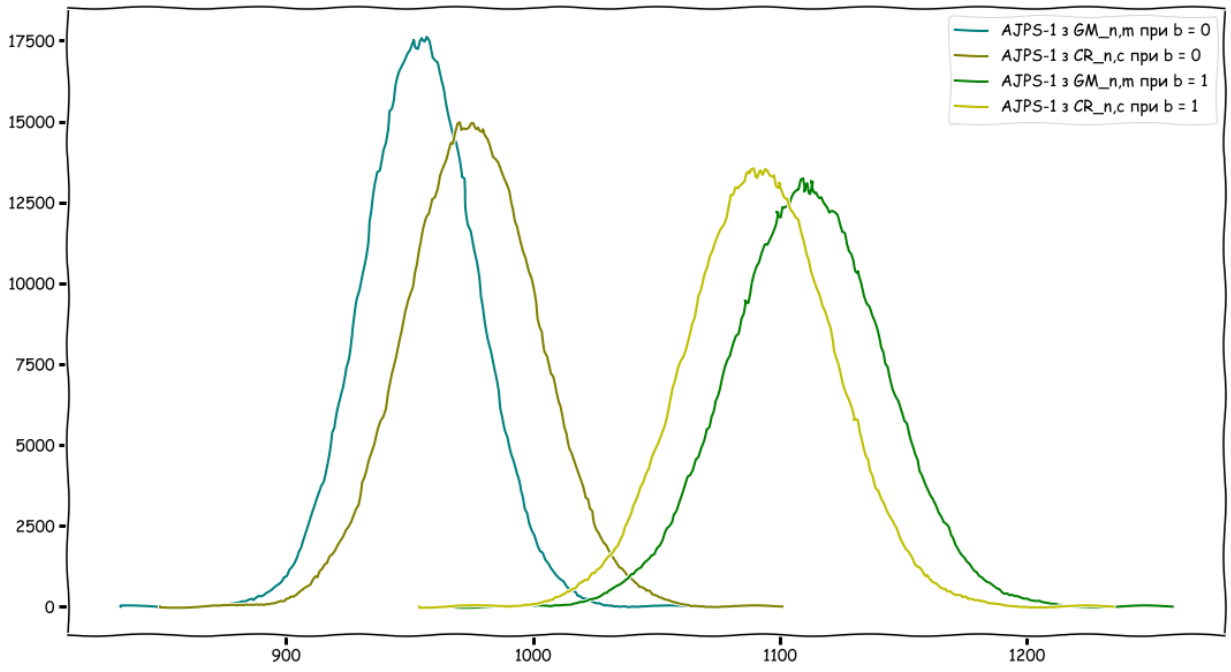

Рисунок Б.33 – Розподіл d модифікацій AJPS-1 з арифметикою за модулем $GM_{n,m}$ та за модулем $CR_{n,c}$ при $n = 2203$ та $h = 23$

2) Нехай $n = 3217$ та $h = 28$, тоді маємо:

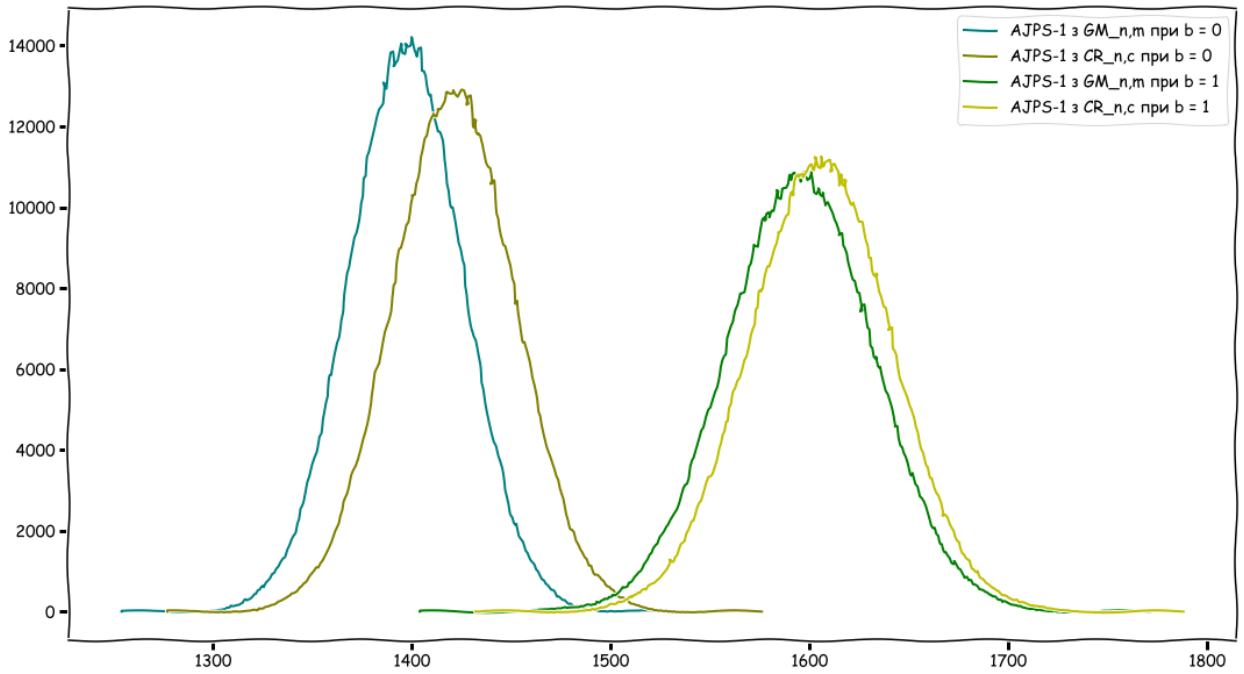

Рисунок Б.34 – Розподіл d модифікацій AJPS-1 з арифметикою за модулем $GM_{n,m}$ та за модулем $CR_{n,c}$ при $n = 3217$ та $h = 28$

3) Нехай $n = 4253$ та $h = 32$, тоді отримуємо:

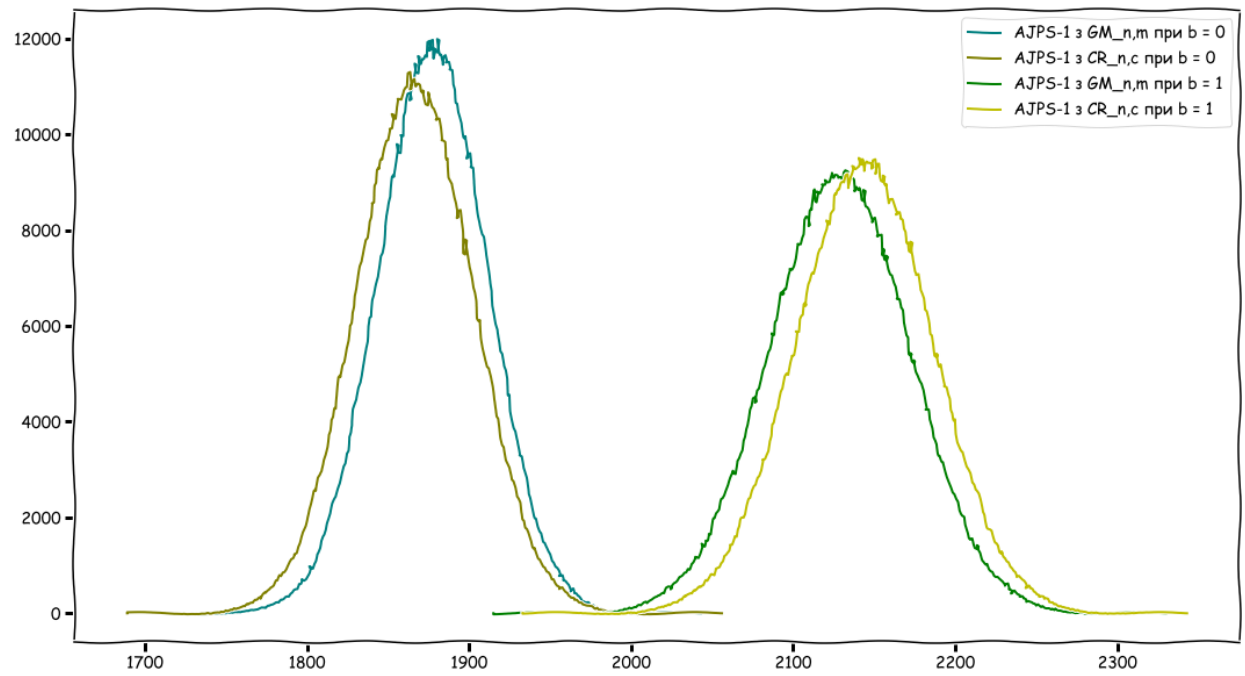

Рисунок Б.35 – Розподіл d модифікацій AJPS-1 з арифметикою за модулем $GM_{n,m}$ та за модулем $CR_{n,c}$ при $n = 4253$ та $h = 32$

4) Нехай $n = 9689$ та $h = 49$, тоді:

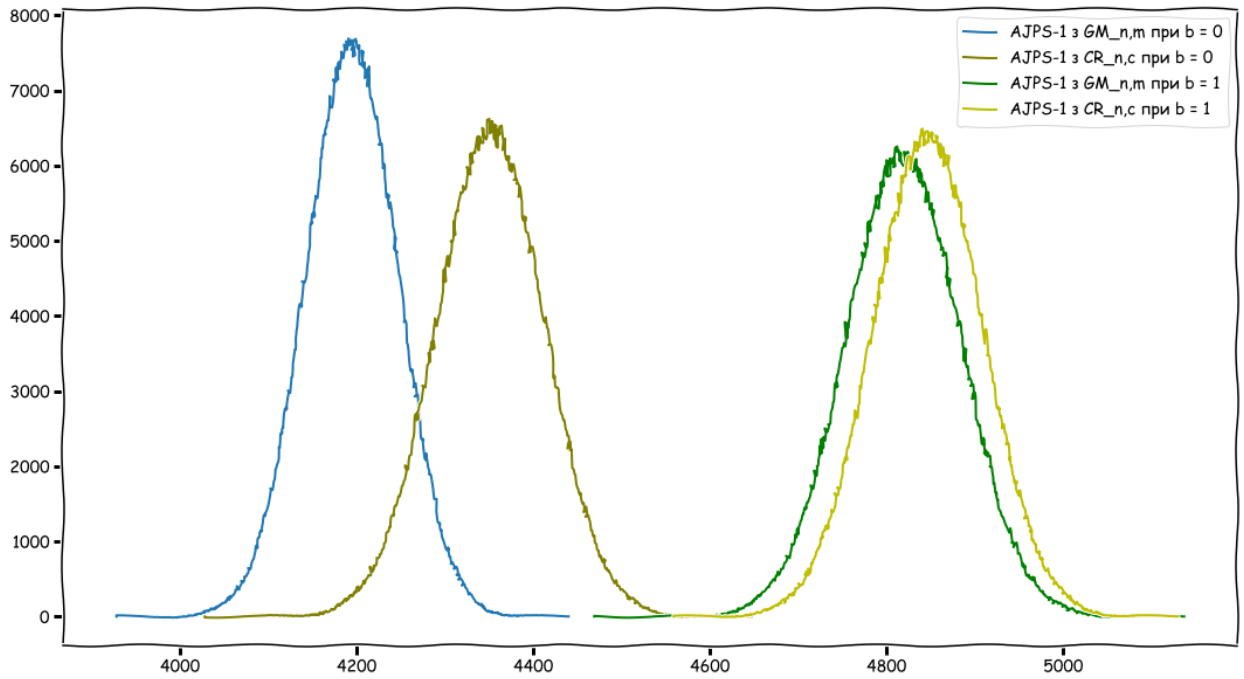

Рисунок Б.36 – Розподіл значення d модифікації AJPS-1 з використанням арифметики за модулем $GM_{n,m}$ та модифікації AJPS-1 з використанням арифметики за модулем числа Кренделла $CR_{n,c}$ при значеннях $n = 9689$, $h = 49$, $m = 25$ та $c = 15$