Mathematics

https://www.aimspress.com/journal/Math

AIMS Mathematics, 9(11): 32232–32256.
DOI: 10.3934/math.20241547
Received: 08 September 2024
Revised: 20 October 2024
Accepted: 04 November 2024
Published: 14 November 2024

Research article

An advanced encryption system based on soft sets

Erdal Bayram, Gülşah Çelik and Mustafa Gezek[*]

Department of Mathematics, Tekirdağ Namık Kemal University, Tekirdağ 59030, Türkiye

[*] **Correspondence:** Email: mgezek@nku.edu.tr; Tel: +90-282-250-2753.

Abstract: Given the application domains of soft set theory, such as decision-making processes, image processing, machine learning, and data mining, it is natural to consider that this theory could be utilized more effectively in encryption systems. A review of the literature reveals that soft set-based encryption systems have been explored in a limited number of studies. This study seeks to develop a new approach for soft sets in encryption systems by utilizing newly introduced algebraic and topological tools. In this system, parties will be able to generate encryption keys independently using soft sets they determine themselves rather than through prior mutual agreement. Additionally, the method of key generation and the size of the key space in the resulting encryption system provides a more secure and distinct alternative compared to existing soft set-based encryption systems.

Keywords: encryption; maximum and minimum operators; soft computing; soft cryptosystem; soft sets
Mathematics Subject Classification: 94A60, 06D72, 11T71

1. Introduction

Uncertainty, while referring to the state of having incomplete, ambiguous, or inconsistent information about a system, situation, or event, is also regarded as a crucial element in understanding the nature of systems and events. It can be articulated through mathematical and statistical models, such as probability theory, which facilitate a quantitative assessment. One of the methods for addressing these uncertainties is soft set theory, which is a generalization of classical set theory. One of the most significant features of soft sets, first defined by Molodtsov [1], is their ability to address uncertainty through a parametric approach. In the real world, often encountered incomplete or ambiguous information cannot be adequately expressed by classical set theory; however, soft sets provide the necessary flexibility to model such situations.

While classical set theory provides a framework based on precise and clear definitions-where an element either belongs to a set or does not-this binary structure proves inadequate when faced with the complexities and uncertainties of real-world applications. For instance, if the quality of a product is uncertain, classical set theory would classify it as either belonging to the high-quality product set or, if it does not belong to that set, as a low-quality product. This strict categorization stems from classical theory's requirement for a clear determination of whether a product meets a specific quality standard. However, in practice, a product's quality typically exists along a spectrum, influenced by various factors such as customer satisfaction, durability, price, functionality, and aesthetics. Such uncertainties cannot be effectively addressed within the rigid binary framework of classical set theory. Compared to the rigid definitions of classical sets, soft sets do not adhere to a binary membership status; they permit variability in membership degrees based on different parameters. Thus, soft sets offer a versatile mathematical tool for addressing uncertain and vaguely defined objects, thanks to the flexibility of their inclusion degrees. Without restrictions on how objects are defined, researchers can select any parameter forms they deem necessary. This capacity enables soft sets to facilitate decision-making processes by organizing and structuring data while considering multiple criteria, allowing users to evaluate alternatives more consciously. This characteristic significantly enhances the decision-making process, making it more efficient and reliable in the presence of partial information.

Additionally, soft sets can be integrated with other mathematical theories and models, thereby increasing their applicability in complex analyses. For example, in various industrial contexts, incorporating game theory into decision-making processes related to system safety and reliability analysis offers considerable advantages in managing uncertainty. Thus, by employing different methodologies in tandem, it becomes possible to better manage uncertainties and enhance system performance (Yazdi et al. [2], Li and Yazdi [3], and Zarei et al. [4]).

Soft sets, in addition to their advantages in handling uncertainty, can be represented in matrix form. In this representation, the rows correspond to elements of the universe, while the columns denote the parameters. Each entry in the matrix reflects the membership status of an element with respect to a given parameter. This structure proves particularly useful in multi-parameter decision-making processes. The matrix representation enables the straightforward evaluation of multiple alternatives across various parameters and allows for efficient data processing through matrix operations or comparisons, especially in high-dimensional datasets, thereby conserving time. Additionally, the implementation of statistical and mathematical methods can be carried out more effectively in a matrix format. The matrix structure also enhances the transmission of data to other systems or stakeholders.

The basic operations and properties of soft sets were initially established by Maji et al. [5] and later refined by Ali et al. [6]. Since then, the theory of soft sets has advanced significantly, with extensive contributions from numerous mathematicians. The application of soft set theory extends across various mathematical structures. Research has also explored algebraic structures derived from soft sets, including soft groups, soft semirings, and soft rings. Recently, Alcantud et al. [7] presented a comprehensive and detailed review of the current state of soft set theory.

The concept of soft topology, along with related topological notions, was introduced independently by Çağman et al. [8] and Shabir and Naz [9]. To investigate and extend topological notions to soft topologies, researchers have examined soft topological spaces in several areas: neighborhood properties by Nazmul and Samanta [10], soft separation axioms by Hussain and Ahmad [11], Min [12], Terepeta [13], and Al-Shami and El-Shafei [14], soft continuity of mappings by Hazra et al. [15], and Aygünoğlu and Aygün [16], and soft compactness by Aygünoğlu and Aygün [16], and Zorlutuna et al. [17]. Further generalizations, such as soft metric spaces, have been defined and studied by Das and Samanta [18]. Recent approaches to deriving soft topologies from classical topologies have been explored by Terepeta [13], Al-Shami and Kocinac [19], Alcantud [20, 21], and Matejdes [22]. Soft set theory demonstrates considerable potential for practical applications and ongoing developments across various domains, particularly in decision-making. For example, decision-making techniques utilizing N-soft sets, an extension of the soft set model, have been proposed by Ali and Akram [23], Adeel et al. [24], and Alcantud et al. [25].

Throughout history, in numerous fields where the transfer of information has been crucial, it has been evident that there is a need for a certain level of confidentiality to prevent unwanted parties from understanding the transmitted information. In an increasingly digital world, cryptography, an essential domain of information security, serves as the cornerstone for protecting data. It ensures the confidentiality, integrity, and accuracy of information during its transmission across networks and storage in databases. To achieve the desired level of confidentiality, numerous encryption systems have been developed in parallel with advancements in technology. The process of transforming a message into an unreadable form using a chosen method is called encryption, while the process of applying the inverse operations used in encryption to recover the original message is referred to as decryption. According to Kerckhoffs' principles [26], which are desired for all encryption methods, the encryption system used should be assumed to be known by everyone, and the system must be practically unsolvable, even if not mathematically so.

The reliability of an encryption system lies in the key used. In encryption processes, two main classes of encryption systems are utilized to achieve the desired security objectives: symmetric and asymmetric encryption systems, each with distinct features and applications. Symmetric encryption systems, often referred to as secret key encryption systems, rely on a single shared key for both the encryption and decryption of data. This key must be kept confidential between communicating parties, as anyone with access to the key can decipher the information. Notable examples of symmetric algorithms include the Advanced Encryption Standard (AES), which is widely used for securing sensitive data, and the Data Encryption Standard (DES), which has been largely phased out due to its vulnerability to brute-force attacks. Symmetric encryption algorithms are highly efficient, making them ideal for encrypting large volumes of data, such as ensuring the security of communication channels or encrypting entire databases. However, a major challenge with symmetric encryption systems is the secure distribution and management of keys, particularly in large-scale systems where multiple parties need access to the keys.

Asymmetric encryption systems address the key distribution problems inherent in symmetric encryption by using two mathematically related keys: a public key and a private key. The public key can be freely distributed and used by anyone to encrypt messages intended for the key's owner, while the private key is kept secret and used to decrypt the messages. This key pair structure underpins many modern security protocols. For instance, RSA, one of the oldest and most widely adopted asymmetric encryption systems, was proposed by Rivest et al. [27], following the introduction of the concept of trapdoor one-way functions by Diffie and Hellman [28]. Rabin [29] introduced a similar public-key encryption system based on the integer factorization problem in 1979. Elliptic curve cryptography (ECC) represents a newer approach to asymmetric encryption, offering equivalent security with smaller key sizes, thus becoming increasingly popular in resource-constrained environments such as mobile devices and IoT applications. Other public-key encryption systems based on different computational problems, such as the El Gamal encryption system, have also been developed. Despite their advantages, asymmetric encryption systems are generally slower and require more computational power compared to symmetric systems, which can be a limitation in some scenarios.

Given that soft set theory has applications in areas such as decision-making processes, image processing, machine learning, data mining, coding theory, group theory, and crystography, it is reasonable to consider that this theory could be utilized more effectively in encryption systems (Adeel et al. [24], Aktaş and Kalkan [30], Alcantud et al. [25], Ali et al. [23], Çağman and Enginoğlu [31], Feng et al. [32], Kalkan [33], Liu et al. [34], Tripathy et al. [35]). A review of the literature indicates that soft set-based encryption systems are extremely rare. This study will present a summary of the relevant literature and a comparison of existing works and introduce a new soft set-based encryption system to address the existing gap in the literature.

This study is organized as follows: Section 2 provides the concepts of soft sets and soft matrices, detailing existing soft set-based encryption systems and their characteristics. Section 3 introduces the operators to be used in the proposed encryption system and outlines the foundational principles of the developed theory. The subsequent section presents the new encryption system, including its algorithm and practical examples. Section 5 highlights the strengths of the proposed system and compares it with existing soft set-based encryption systems in the literature. Section 6 summarizes the findings of this study and discusses potential future research problems. An example alphabet for the case $(n, r) = (4, 3)$ is provided in the Appendix.

2. Soft set-based encryption systems

Let X be a set, and U, E, and $P(U)$ represent the universe, the set of parameters, and the power set of X, respectively. The term *soft set over U* refers to a pair of (F, A), where F is a function defined as $F : A \subseteq E \to P(U)$ [1]. Alternatively, the soft set can be described as a parameterized family of $P(U)$. The set of all soft sets over U will be symbolized as $SS\ (U, E)$. Considering $F(e) = \varnothing$ for every $e \in E \setminus A$, (F, E) can be written instead of (F, A). The soft set (F, E) is abbreviated as F_E for simplicity.

Soft sets can be associated with matrices for the purpose of storage or transmission in a computer environment. For a soft set F_E, the subset

$$R_E = \{(u, e) : e \in E, u \in F(e)\} \subset U \times E$$

is referred to as a *relation form* of F_E, and the function

$$\chi_{R_E} : U \times E \to \{0, 1\},$$

$$\chi_{R_E}(u, e) = \begin{cases} 1 & \text{if } (u, e) \in R_E \\ 0 & \text{if } (u, e) \notin R_E \end{cases}$$

is known as the *characteristic function* of the relation. Accordingly, for a universe set with q elements and a parameter set with r elements,

$$\left[a_{ij}\right]_{q\times r} = \left[\chi_{R_E}(u_i, e_j)\right]_{q\times r}$$

defines the *soft set matrix* of F_E.

For example, consider a universe set consisting of patients $U = \{k_1, k_2, k_3, k_4, k_5\}$, and a parameter set consisting of symptoms $E = \{e_1 = \text{shortness of breath}, e_2 = \text{severe cough}, e_3 = \text{high fever}\}$. A soft set used to determine whether the patients have a specific disease is as follows:

$$F_E = \{e_1 = \{k_1, k_2, k_4, k_5\}, e_2 = \{k_1, k_3\}, e_3 = \{k_2, k_4, k_5\}\}.$$

The soft set matrix of F_E would be as

$$\begin{pmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \\ 1 & 0 & 1 \end{pmatrix}.$$

One of the first studies on soft set-based encryption systems in the literature was conducted by Aygün [36]. This study utilized a soft set matrix to create a key selected by the receiver and sender through mutual agreement. The soft set matrix $[a_{ij}]$ and a matrix $[b_{ij}]$, where each row corresponds to the vector representation of each character in the message within the alphabet, are constructed. Using these matrices, the encryption process is defined by:

$$c_{ij} = \begin{cases} 1 & \text{if } a_{ij} \neq b_{ij} \\ 0 & \text{if } a_{ij} = b_{ij} \end{cases}$$

where $[a_{ij}] \cdot i[b_{ij}] = [c_{ij}]$ denotes the *inverse product*. Similarly,

$$d_{ij} = \begin{cases} 1 & \text{if } a_{ij} = b_{ij} \\ 0 & \text{if } a_{ij} \neq b_{ij} \end{cases}$$

where $[a_{ij}] \cdot c[b_{ij}] = [d_{ij}]$ denotes the *characteristic product*. These operations serve as the primary tools in the encryption system developed by Aygün [36].

The second study, by Aygün [37], on soft set-based encryption systems builds upon the encryption system introduced in Aygün [36], with the goal of improving the reliability of the encryption process by integrating a permutation within the encryption scheme utilizing soft sets.

One of the most recent publications on soft sets was authored by Paik and Mondal [38]. They developed a new encryption algorithm based on the foundation of previous works, employing a new approach. Instead of matrices, a symmetric difference operation defined between soft sets as

$$(I, L)\tilde{\Delta}(J, M) = ((I, L) \cup_R (J, M)) -_R ((I, L) \cap_R (J, M))$$

was utilized.

Some other studies on soft set-based encryption systems can be found in Aktaş and Kalkan [30], Kalkan [33], and Yılmaz [39]. Soft set-based encryption systems in the literature are illustrated in Figure 1.

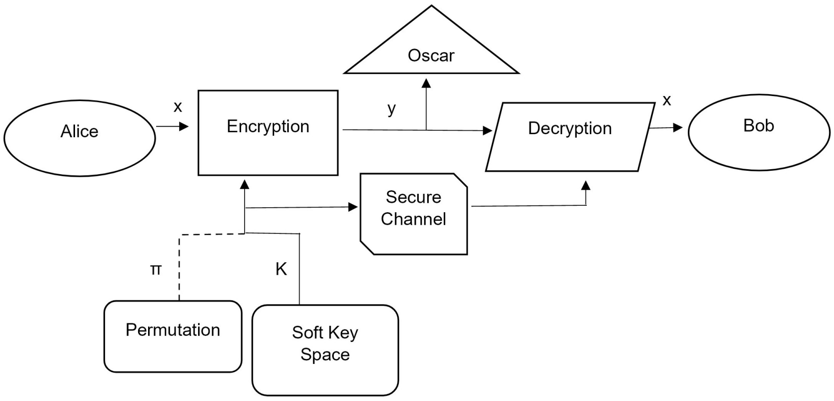

Figure 1. Soft set-based encryption systems in the literature.

3. The mathematical theory

This section introduces the mathematical theory behind the encryption system. The fundamental components of this system are the maximum (or minimum) operators based on the algebraic structure of the chosen alphabet. The operators employed in the proposed encryption system, as far as we know, are being applied for the first time in a cryptosystem, adding a new dimension to the encryption process. In this context, the innovative use of maximum and minimum operators also contributes to the development of stronger and more reliable encryption systems in modern cryptography. The basis of these operators is the maximum (or minimum) functions, which, when selecting high values, never take certain values based on the n values determined in the algebraic structure. Moreover, as the n value increases, the number of values that the maximum (or minimum) function does not take also increases. However, these excluded values are utilized in encryption, thereby enhancing the complexity of the encryption process, making it more challenging for attackers to understand the encryption structure, ultimately providing stronger security. The definition of the maximum (or minimum) operator in relation to matrices, where calculations are mathematically straightforward, and the fact that these operators do not require complex mathematical processing, ensures computational efficiency. On the other hand, the use of these operators also impacts the size of the key space. Consequently, this richness in the mathematical structure of the encryption system complicates the ability of potential attackers to identify the system's vulnerabilities.

Due to the properties of the operators and the encryption system, the algebraic structure $\mathbb{Z}_n$ used in encryption must have $n > 3$.

As will be defined later, based on the algebraic structure, a vector will be created for each character in the encrypted information within the algorithms. Therefore, the length of the vectors used must be determined based on the number of elements in the alphabet. If vectors of length r are to be created for the chosen $\mathbb{Z}_n$ residue class, the alphabet used will have n^r elements.

Accordingly, to ensure that the encryption system is compatible with each language, if the number of characters in the language's alphabet is not n^r, the alphabet can be expanded to n^r elements by adding different characters from languages in the surrounding area. An example of such an alphabet is provided in the Appendix.

One of the main elements of the new encryption system is the maximum and minimum operators.

First, we provide the tools needed to define these operators.

Definition 1. *Let $n \in \mathbb{N}^+$. The function $M : \mathbb{N} \times \mathbb{N} \to \mathbb{Z}_n$, defined by*

$$M(p,k) = \max\{p - k \textit{ (mod n)},\ k - p \textit{ (mod n)}\},$$

is called the maximum function, and similarly, the function $m : \mathbb{N} \times \mathbb{N} \to \mathbb{Z}_n$, defined by

$$m(p,k) = \min\{p - k \textit{ (mod n)},\ k - p \textit{ (mod n)}\},$$

is called the minimum function.

In other words, the function M (or m) takes two non-negative integers p and k, computes their differences in both directions, and then takes the maximum (or minimum) of their remainders when divided by n. A closer examination of these functions yields the following result:

Corollary 1. *When the function M (or m) is restricted to soft set matrices, the domain and range of M (or m) are isomorphic to $\mathbb{Z}_2 \times \mathbb{Z}_2$ and $\mathbb{Z}_3$, respectively. On the other hand, when M (or m) is restricted to the alphabet, the domain and range of M (or m) become isomorphic to $\mathbb{Z}_n \times \mathbb{Z}_2$ and $\mathbb{Z}_n$, respectively.*

To increase the difficulty of breaking the developed encryption system, a random assignment process can be implemented.

Lemma 1. *The equality*

$$k - p \textit{ (mod n)} \equiv n - (p - k) \textit{ (mod n)}$$

holds for each $p, k \in \mathbb{Z}_n$.

Proof. It is obvious. □

We will use the identity from Lemma 1 to prove the theorems covered in the rest of this section.

Theorem 1. *Let $n \geq 3$ be an integer. The maximum function M, defined on the set $\mathbb{Z}_n \times \mathbb{Z}_n$, cannot take the values $\{1, 2, \dots, \frac{n}{2} - 1\}$ if n is an even integer and cannot take the values $\{1, 2, \dots, \frac{n-1}{2}\}$ if n is an odd integer.*

Proof. If $p = k$, then $M(p,k) = 0$. Assume that $p \neq k$, and n is an even integer. If

$$0 < p - k \text{ (mod } n) \leq \frac{n}{2},$$

then, by Lemma 1, the inequality

$$k - p \text{ (mod } n) \equiv n - (p - k) \text{ (mod } n) \geq \frac{-n}{2} \equiv \frac{n}{2}$$

implies that

$$M(p,k) = k - p \text{ (mod } n) \geq \frac{n}{2}.$$

On the other hand, if

$$p - k \text{ (mod } n) > \frac{n}{2},$$

then the inequality

$$k - p \,(\mathrm{mod}\, n) \equiv n - (p - k) \,(\mathrm{mod}\, n) < \frac{n}{2}$$

suggests that

$$M(p, k) = p - k \,(\mathrm{mod}\, n) \geq \frac{n}{2}.$$

Thus, for all $p, k \in \mathbb{Z}_n$, we obtain

$$M(p, k) \in \{0, \frac{n}{2}, \cdots, n - 1\}.$$

Now suppose that n is an odd integer. If

$$0 < p - k \,(\mathrm{mod}\, n) \leq \frac{n-1}{2},$$

then the inequality

$$k - p \,(\mathrm{mod}\, n) \equiv n - (p - k) \,(\mathrm{mod}\, n) \geq \frac{n+1}{2}$$

indicates that

$$M(p, k) = k - p \,(\mathrm{mod}\, n) \geq \frac{n+1}{2}.$$

Conversely, if

$$p - k \,(\mathrm{mod}\, n) > \frac{n-1}{2},$$

then the inequality

$$k - p \,(\mathrm{mod}\, n) \equiv n - (p - k) \,(\mathrm{mod}\, n) < \frac{n+1}{2}$$

entails that

$$M(p, k) = p - k \,(\mathrm{mod}\, n) > \frac{n-1}{2}.$$

Thus, for all $p, k \in \mathbb{Z}_n$, we obtain that

$$M(p, k) \in \{0, \frac{n+1}{2}, \cdots, n - 1\}.$$

The result follows. □

Theorem 2. *Let $n \geq 3$ be an integer. The minimum function m, defined on the set $\mathbb{Z}_n \times \mathbb{Z}_n$, cannot take the values $\{\frac{n}{2} + 1, \ldots, n - 1\}$ if n is an even integer and cannot take the values $\{\frac{n+1}{2}, \ldots, n - 1\}$ if n is an odd integer.*

Proof. When $p = k$, $m(p, k) = 0$. Assume that $p \neq k$, and n is an even integer. If

$$0 < p - k \,(\mathrm{mod}\, n) \leq \frac{n}{2},$$

then the inequality

$$k - p \,(\mathrm{mod}\, n) \equiv n - (p - k) \,(\mathrm{mod}\, n) \geq \frac{n}{2}$$

indicates that

$$m(p,k) = p - k \,(\mathrm{mod}\, n) \leq \frac{n}{2}.$$

In contrast, if

$$p - k \,(\mathrm{mod}\, n) > \frac{n}{2},$$

then the inequality

$$k - p \,(\mathrm{mod}\, n) \equiv n - (p - k) \,(\mathrm{mod}\, n) < \frac{n}{2}$$

shows that

$$m(p,k) = k - p \,(\mathrm{mod}\, n) < \frac{n}{2}.$$

Thus, for all $p, k \in \mathbb{Z}_n$,

$$m(p,k) \in \{0, 1, 2, \ldots, \frac{n}{2}\}.$$

Assume now that n is an odd integer. If

$$0 < p - k \,(\mathrm{mod}\, n) \leq \frac{n-1}{2},$$

then the inequality

$$k - p \,(\mathrm{mod}\, n) \equiv n - (p - k) \,(\mathrm{mod}\, n) \geq \frac{n+1}{2}$$

implies that

$$m(p,k) = p - k \,(\mathrm{mod}\, n) \leq \frac{n-1}{2}.$$

On the other hand, if

$$p - k \,(\mathrm{mod}\, n) > \frac{n-1}{2},$$

then the inequality

$$k - p \,(\mathrm{mod}\, n) \equiv n - (p - k) \,(\mathrm{mod}\, n) < \frac{n+1}{2}$$

suggests that

$$m(p,k) = k - p \,(\mathrm{mod}\, n) < \frac{n+1}{2}.$$

Thus, for all $p, k \in \mathbb{Z}_n$, we have

$$m(p,k) \in \{0, 1, 2, \ldots, \frac{n-1}{2}\}.$$

This leads to the result. □

Definition 2. *Let $p, k \in \mathbb{N}$. Define*

$$S^{p,k} = \begin{cases} n-1, & \text{if } M(p,k) \equiv (p-k) \mod n, \\ n-2, & \text{if } M(p,k) \equiv (k-p) \mod n, \\ 0, & \text{otherwise} \end{cases}$$

and

$$s^{p,k} = \begin{cases} 1, & \text{if } m(p,k) \equiv (p-k) \mod n, \\ 2, & \text{if } m(p,k) \equiv (k-p) \mod n, \\ 0, & \text{otherwise.} \end{cases}$$

The natural numbers defined in Definition 2 are used to denote whether the value of $M(p, k)$ (or $m(p, k)$) is calculated based on the difference $p - k$ or $k - p$. Therefore, $S^{p,k}$ (or $s^{p,k}$) is referred to as the *location indicator* according to the function M (or m) for p and k.

As discussed in Theorems 1 and 2, the values that M (or m) does not take will be added as a third coordinate within the encryption process in the algorithm, a procedure referred to as *random assignment*.

The random assignments and location indicators for some specific values (n, r) of the functions M and m are provided in Table 1, where Column 1 gives the specific value for n^r, Columns 2 and 3 provide possible randomly assigned elements and their location indicators for the function M, respectively, and Columns 4 and 5 supply possible randomly assigned elements and their location indicators for the function m, respectively.

Table 1. Some examples of random assignments and location indicators.

	Ran.	Loc.	Ran.	Loc.
n^r	Assg.	Indic	Assg.	Indic.
	Elts. (M)	(M)	Elts. (m)	(m)
3^3	None	0,1,2	None	0,1,2
3^4	None	0,1,2	None	0,1,2
4^3	1	0,2,3	3	0,1,2
6^2	1,2	0,4,5	4,5	0,1,2
7^2	1,2,3	0,6,5	4,5,6	0,1,2
8^2	1,2,3	0,6,7	5,6,7	0,1,2
9^2	1,2,3,4	0,7,8	5,6,7,8	0,1,2
10^2	1,2,3,4	0,8,9	6,7,8,9	0,1,2

The set of all $n \times m$ matrices with entries from non-negative integers will be denoted by $Mat_{n\times m}$. The maximum and minimum operators used in the encryption algorithms are defined as follows.

Definition 3. *Let $P = [p_{ij}], K = [k_{ij}] \in Mat_{n\times m}$, $M(p_{ij}, k_{ij}) = M_{ij}$, and $m(p_{ij}, k_{ij}) = m_{ij}$. The maximum operator is defined as*

$$C : Mat_{n\times m} \times Mat_{n\times m} \to Mat_{n\times m},$$

where

$$C(P, K) = [M_{ij} S^{p_{ij},k_{ij}}],$$

and the minimum operator is defined as

$$D : Mat_{n\times m} \times Mat_{n\times m} \to Mat_{n\times m},$$

where

$$D(P, K) = [m_{ij} s^{p_{ij},k_{ij}}].$$

In Definition 3, $M_{ij}S^{p_{ij},k_{ij}}$ (or $m_{ij}s^{p_{ij},k_{ij}}$) denotes a vector of length two, where the first component comes from the maximum (or minimum) function, and the second comes from the location indicator.

Let S_{3r} denote the set of all possible permutations defined on the set $\{1, 2, \ldots, 3r\}$, and $\pi = (u_1)(u_2)\cdots(u_\ell) \in S_{3r}$, where u_i represents a cycle of length k_i ($\sum k_i = 3r$, $i \in \{1, 2, \cdots, \ell\}$). We define the matrix form of the permutation π, denoted by $[\pi]$, as follows:

i) If for some i, $k_i = 1$, there will be no row in the matrix $[\pi]$ corresponding to the cycle u_i.

ii) If $k_i > 1$, each digit in the cycle $u_i = (x_i^1 x_i^2, \ldots, x_i^{k_i})$ is written as a row in a $k_i \times 3$ matrix $[\pi_{u_i}]$, with the digits corresponding to the elements of the alphabet. If $i < \ell$, the first digit of the cycle u_i is written as the $k_i + 1$-th row at the end of this block matrix.

iii) The matrix $[\pi]$ is obtained as

$$\begin{bmatrix} [\pi_{u_1}] \\ [\pi_{u_2}] \\ \vdots \\ [\pi_{u_\ell}] \end{bmatrix}.$$

4. An advanced encryption system based on soft sets

This section will introduce a new encryption system for transmitting any text using soft sets, appropriate alphabets, and compatible $\mathbb{Z}_n$ residue classes, in the context of contemporary technology. In this encryption system, a key component, derived from soft sets, will not be predetermined by the parties but will be transferred through a secure channel (as an encrypted text) within an encryption system.

Throughout this section, let $\mathcal{P}$, $\mathcal{S}$, and $\mathcal{K}$ be a finite set of possible plaintexts, a finite set of possible ciphertexts, and a finite set of possible keys, respectively. Encryption and decryption rules will be defined as

$$e_K : \mathcal{P} \to \mathcal{S}, \text{ and } d_K : \mathcal{S} \to \mathcal{P}$$

such that

$$d_K(e_K(x)) = x$$

for each $x \in \mathcal{P}$.

In the proposed encryption system, the goal is that, in the absence of a previously agreed common key, both parties independently define soft sets of keys in matrix form and a permutation to be created by the receiver for encryption and decryption purposes.

The key used in the encryption system can be shared between the sender and the receiver using any encryption method. Specifically, this sharing process can be described using a *Key Exchange Algorithm*, outlined in the steps below. More precisely, after the sender transmits their key in a soft matrix form, the receiver, using this key, creates their own compatible key and a permutation, performs encryption, and sends the result back to the sender. Consequently, the sender obtains the receiver's key and the permutation. The core principle here is that both parties independently determine their own keys and that different keys can be used for each encryption.

Definition 4 (Key Exchange Algorithm (KEA)). *The Key Exchange Algorithm (KEA) based on soft sets is defined as follows:*

Let $\mathcal{P} = \mathbb{Z}_2^r, \mathcal{S} = \mathbb{Z}_n^r$ and let

$$\mathcal{K} = \{F_E : F_E \in SS(U, E)\}.$$

For $K = F_E$, define

$$e_K(x) = C_{R,\pi}$$

and

$$d_K(y) = \left(C_x^{-1}(\beta), C_K^{-1}(\gamma)\right)$$

where β is a $q \times r$ type matrix representing the first $q \cdot r$ coordinates of y, γ is the remaining part of the matrix $[y]$, and $C_x^{-1}(\beta)$ is the matrix determined such that its image under the maximum operator C, with respect to x, is the matrix β ($C_K^{-1}(\gamma)$ can be determined similarly).

Details of the KEA:

Step 1: The receiver and sender determine the values of n and r in accordance with a mutually agreed alphabet.

Step 2: The sender selects a q-element universe set and an r-element parameter set to define the soft set A_E. Subsequently, the sender transmits the matrix $[A_E]$, which is a $q \times r$ matrix, to the receiver by writing out the alphabetic counterparts of the rows of this matrix sequentially.

Step 3: The receiver converts the received text into the matrix $[A_E]$ using the alphabet. The receiver then selects their own soft set B_E, ensuring it is of the same dimension, and constructs the soft matrix $[B_E]$ as the key.

Step 4: The receiver computes the matrix

$$[C] = C\left([A_E], [B_E]\right)$$

which is the image of the matrix of the sender's soft set under the maximum operator with the matrix of the receiver's key.

Step 5: To perform random assignment using the values not taken by the operator and elements from the selected class $\mathbb{Z}_n$, the following procedure is applied: For each coordinate (i, j) in the $q \times r$ matrix $[C]$, the third coordinate is assigned as follows:

i) If $i + j = 0 \pmod 2$, assign one of the values not taken by the operator as the third coordinate.

ii) If $i + j = 1 \pmod 2$, assign one of the elements not used in the encryption process from the class $\mathbb{Z}_n$ as the third coordinate.

Thus, the matrix $[C_R]$ of type $q \times r$ is obtained, where each entry is a 3-dimensional vector.

Step 6: The receiver arbitrarily chooses a permutation $\pi \in S_{3r}$ and determines the matrix $[\pi]$. Applying the procedures of steps 4 and 5 with their own key, the receiver computes the matrix $[C_\pi]$.

Step 7: The receiver forms the matrix

$$C_{R,\pi} = \begin{bmatrix} [C_R] \\ [C_\pi] \end{bmatrix}.$$

Step 8: By determining the alphabetic counterparts of each entry in the matrix $C_{R,\pi}$, the receiver generates the encrypted text, where the first $q \cdot r$ characters represent the receiver's key and the remaining characters represent the permutation.

Step 9: The receiver sends the encrypted message y back to the sender. The initial $q.r$ coordinates of y are referred to as β, while the remaining portion of y is denoted by γ.

Step 10: The sender retrieves the matrix $[C_R]$ by finding the alphabetic counterparts of the first $q \cdot r$ characters of the encrypted text and removes the random assignments. Using the properties of the maximum operator, placeholder information, and their own key, the sender derives the matrix form of the receiver's key K, which is $C_x^{-1}(\beta)$.

Step 11: Given that the sender now possesses the receiver's key, they apply the procedures from step 10 to the remaining part of the encrypted matrix $[y]$ to determine the matrix $[C_\pi]$, which is $C_K^{-1}(\gamma)$, and subsequently find the permutation π. Thus, the sender obtains the receiver's specified key and permutation.

Since the sender now knows the receiver's key and permutation, they will use these to encrypt the message according to the steps outlined in the Main Algorithm:

Definition 5 (**Main Algorithm**). *An advanced encryption system based on soft sets is defined as follows:*

Let $\mathcal{P} = \mathcal{S} = \mathbb{Z}_n^r$, and let

$$\mathcal{K} = \{(F_E, \pi) : F_E \in SS(U, E), \pi \in S_{3r}\}.$$

For $K = (F_E, \pi)$, define

$$e_K(x) = [C_R]^\pi$$

and

$$d_K(y) = C^{-1}(\alpha), \text{ where } \alpha = \left[\left([y]^{\pi^{-1}}\right)_{R^{-1}}\right].$$

Details of the Main Algorithm:

Step 1: The sender converts the original message x into matrix form $[x]$ using the alphabet.

Step 2: The sender calculates the matrix

$$[C] = C\left([x], [A_E]\right)$$

which is the image of the message matrix under the maximum operator with the receiver's key matrix.

Step 3: To perform random assignment using the values not taken by the operator and elements from the selected class $\mathbb{Z}_n$, the following procedure is applied: For each coordinate (i, j) in the $q \times r$ matrix $[C]$, the third coordinate is assigned as follows:

i) If $i + j = 0 \pmod 2$, assign one of the values not taken by the operator as the third coordinate.

ii) If $i + j = 1 \pmod 2$, assign one of the elements from $\mathbb{Z}_n$ that was not used in the encryption process as the third coordinate.

Thus, the matrix $[C_R]$ of type $q \times r$ is obtained, where each entry is a 3-dimensional vector.

Step 4: The sender then applies the π permutation obtained from the receiver to each row of the matrix $[C_R]$ to get the matrix $[C_R]^{\pi}$.

Step 5: By writing the alphabetic counterparts of each entry in the resulting matrix, the sender obtains the encrypted form of the message they wish to transmit.

Step 6: The sender sends the encrypted message to the receiver.

Step 7: Upon receiving the message y, the receiver constructs the matrix $[y]$ using the alphabet and applies the inverse permutation π^{-1} to each row of the matrix $[y]$, resulting in the matrix $[y]^{\pi^{-1}}$.

Step 8: By removing the third coordinates, which were assigned randomly, from each entry of the matrix $[y]^{\pi^{-1}}$, the matrix $\alpha = \left[\left([y]^{\pi^{-1}}\right)_{R^{-1}}\right]$ is derived.

Step 9: Finally, the matrix $[x]$ is determined such that its image under the maximum operator C is the matrix α. The original message x is then obtained by converting each row of $[x]$ into its corresponding alphabetic representation.

Remark 1. *Although the number of columns in the matrix form of the message to be encrypted must match the number of columns in the key matrices, the number of rows may be greater. In such cases, during encryption and decryption, copies of the key matrices are added to ensure that they are of the same dimension as the matrix form of the message. Conversely, if the number of rows in the matrix form of the message is less than that of the key matrices, the encryption and decryption processes utilize as many rows of the key matrices as there are rows in the matrix form of the message.*

Remark 2. *Alternative algorithms can be developed by substituting the maximum operator used in the algorithms aforementioned with the minimum operator.*

The new soft set-based encryption system defined above is illustrated in Figure 2.

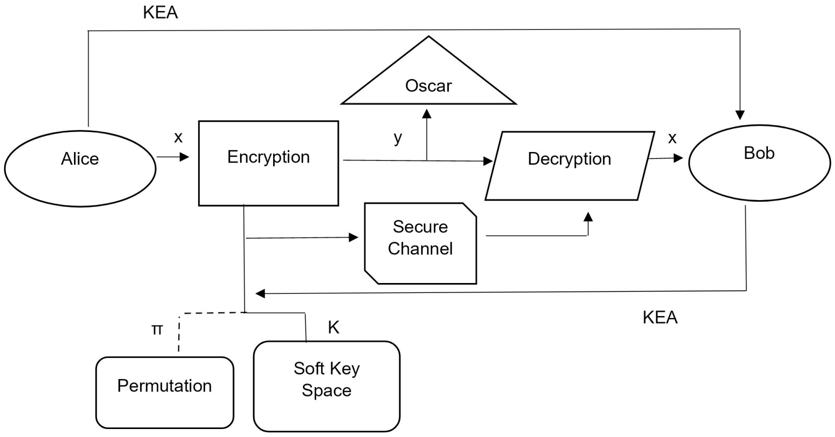

Figure 2. Newly developed encryption system based on soft sets.

Example 1. *Alice wishes to send a message to Bob using the alphabet provided in the Appendix. To create the soft sets, Alice defines the universe as* $U = \{k_1, k_2, k_3, k_4\}$ *and the parameter set as*

$E = \{e_1, e_2, e_3\}$. Alice chooses the soft set $A_E = \{e_1 = \{k_1, k_2, k_3\}, e_2 = \{k_1, k_3\}, e_3 = \{k_1, k_4\}\}$ as the key and constructs the soft matrix as follows:

$$A_E = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

After finding the alphabetic counterparts of each entry in this matrix, she sends the message İFI1 to Bob. Bob chooses the soft set $F_E = \{e_1 = \{k_2, k_4\}, e_2 = \{k_1, k_2, k_3\}, e_3 = \{k_1, k_3\}\}$ as his key, and constructs the soft matrix

$$F_E = \begin{bmatrix} 0 & 1 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 0 \end{bmatrix}.$$

Using both keys and the maximum operator, he applies steps 4 and 5 of the Key Exchange Algorithm to obtain the matrices

$$C([A_E], [F_E]) = \begin{bmatrix} 32 & 00 & 00 \\ 00 & 33 & 00 \\ 32 & 00 & 33 \\ 33 & 00 & 32 \end{bmatrix}, \quad C_R([A_E], [F_E]) = \begin{bmatrix} 321 & 000 & 001 \\ 003 & 331 & 002 \\ 321 & 000 & 331 \\ 332 & 001 & 320 \end{bmatrix}.$$

Bob chooses a permutation $\pi = (13567)(2489)$ and obtains the matrix $[\pi]$ as follows:

$$[\pi] = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 3 \\ 0 & 1 & 1 \\ 0 & 1 & 2 \\ 0 & 1 & 3 \\ 0 & 0 & 1 \\ 0 & 0 & 2 \\ 0 & 1 & 0 \\ 0 & 2 & 0 \\ 0 & 2 & 1 \end{bmatrix}, \quad [C_\pi] = \begin{bmatrix} 001 & 333 & 001 \\ 332 & 331 & 333 \\ 001 & 002 & 001 \\ 333 & 321 & 202 \\ 001 & 002 & 201 \\ 332 & 331 & 323 \\ 001 & 333 & 321 \\ 332 & 321 & 002 \\ 001 & 322 & 331 \\ 333 & 321 & 322 \end{bmatrix}, \quad C_{R,\pi}([A_E], [F_E]) = \begin{bmatrix} 321 & 000 & 001 \\ 003 & 331 & 002 \\ 321 & 000 & 331 \\ 332 & 001 & 320 \\ 001 & 333 & 001 \\ 332 & 331 & 333 \\ 001 & 002 & 001 \\ 333 & 321 & 202 \\ 001 & 002 & 201 \\ 332 & 331 & 323 \\ 001 & 333 & 321 \\ 332 & 321 & 002 \\ 001 & 322 & 331 \\ 333 & 321 & 322 \end{bmatrix}.$$

Bob then finds the alphabetic counterparts of each entry in the resulting matrix and obtains the encrypted message

$$\Xi 013\Phi 2\Xi 0\Phi\Psi 1\Lambda 1\Omega 1\Psi\Phi\Omega 121\Omega\Xi U12T\Psi\Phi\Sigma 1\Lambda\Xi\Psi\Xi 21\Pi\Phi\Omega\Xi\Pi.$$

He sends this encrypted message to Alice. Alice, by applying steps 10 and 11 of KEA to the received encrypted message, retrieves Bob's key as a pair of a soft set and a permutation π.

Alice wishes to send the message "life is good" to Bob. By applying step 1 of the Main Algorithm to convert the message into matrix form using the alphabet, she obtains

$$P = \begin{bmatrix} 1 & 2 & 0 \\ 1 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 3 & 3 \\ 1 & 1 & 1 \\ 1 & 3 & 3 \\ 1 & 0 & 1 \\ 1 & 2 & 3 \\ 1 & 2 & 3 \\ 0 & 3 & 2 \end{bmatrix}, \quad C_R(P, F_E) = \begin{bmatrix} 321 & 323 & 331 \\ 002 & 001 & 320 \\ 321 & 332 & 331 \\ 330 & 331 & 330 \\ 321 & 000 & 001 \\ 003 & 201 & 330 \\ 321 & 330 & 001 \\ 002 & 201 & 332 \\ 321 & 322 & 201 \\ 320 & 201 & 203 \end{bmatrix}, \quad C_R^{\pi}(P, F_E) = \begin{bmatrix} 132 & 333 & 312 \\ 200 & 213 & 000 \\ 133 & 323 & 312 \\ 033 & 313 & 303 \\ 100 & 000 & 312 \\ 320 & 313 & 000 \\ 133 & 000 & 312 \\ 220 & 313 & 020 \\ 132 & 022 & 312 \\ 020 & 012 & 332 \end{bmatrix}.$$

The final matrix is converted into the encrypted message

$$R\Omega\Delta \c{S} Q0K\Psi\Delta E\Theta O'F0\Delta\Lambda\Theta 0S\,0\Delta X\Theta 8RA\Delta 86\Psi,$$

which Alice sends to Bob.

Example 2. *This example addresses a secure and strategic communication process between two investment companies. Company A, which provides investment consulting services, conducts market analyses to identify the most suitable investment opportunities and presents these opportunities to Company B, which offers portfolio management services. These proposals are categorized based on specific risk levels. Company B analyzes the offers received from Company A to determine and make investment strategies. Encrypting investment proposals is preferred to ensure the security of these recommendations. This process protects the confidentiality of investment strategies, preventing rival firms from acquiring this information and shaping their strategies accordingly. Thus, it helps Company A gain a competitive advantage in the market. Additionally, encryption enhances the security for both parties by preventing third parties from tracking and manipulating the information.*

The following universal set (U) represents the investment options. Each element expresses the investment instruments evaluated by Company A.

$$\begin{aligned} U = \{ & k_1 = Stock\ S_1, k_2 = Stock\ S_2, k_3 = Stock\ S_3, k_4 = Bond\ B_1, k_5 = Bond\ B_2, k_6 = Fund\ F_1, \\ & k_7 = Fund\ F_2, k_8 = Fund\ F_3, k_9 = Currency\ Euro, k_{10} = Currency\ Dollar, k_{11} = Gold\}. \end{aligned}$$

The following parameter set (E) categorizes the risk levels of the investment instruments. This classification facilitates investors in making choices according to their risk tolerance.

$$E = \{e_1 = Low\text{-}risk\ investment, e_2 = Medium\text{-}risk\ investment, e_3 = High\text{-}risk\ investment\}.$$

Accordingly, Company A offers Company B three different investment proposals categorized as low, medium, and high risk, enabling investors to diversify their portfolios.

Company A determines the investment recommendations based on the following dataset, which presents the key characteristics of each investment instrument and the market conditions. Thus, the investment recommendations are supported by more concrete data.

Table 2. Some categorized investment instruments and their metrics.

	Inv.	Exp.	Mat.	Inv.	Mkt.	Int.	Mng.	Vol.	Mkt.	Risk	Liqu.
	Inst.	Ret.	Per.	Amt.	Val.	Rate	Fee	(%)	Trend	Score	Ratio
		(%)	(Yrs)	(USD)	(USD)	(%)	(%)				
Stocks	S_1	8	5	50k	1M	-	-	18	Inc.	6	High
	S_2	7	3	45k	800k	-	-	15	Stbl.	5	Med.
	S_3	9	4	30k	600k	-	-	12	Inc.	6	High
Bonds	B_1	4	10	100k	500k	3	-	2	Dec.	3	Low
	B_2	5	7	95k	300k	4	-	2.5	Dec.	3	Low
Funds	F_1	10	2	20k	200k	-	1.2	9	Inc.	6	Med.
	F_2	8	3	30k	150k	-	1.5	8	Stbl.	5	Med.
	F_3	9	2	25k	175k	-	1.1	7	Inc.	6	High
Curren.	Euro	3	1	10k	50k	-	-	5	Dec.	4	High
	Dollar	2	1	5k	20k	-	-	4	Dec.	4	High
Commod.	Gold	6	1	15k	80k	-	-	6	Inc.	5	Med.

Company A intends to convey the message, "As investment recommendations, we are considering a high-risk investment for Fund F_1, medium-risk investments for Stock S_1, Fund F_3, and gold, and low-risk investments for Stock S_2, Bond B_1, and Currency Euro." to Company B.

This message allows Company A to express its investment recommendations clearly. The intended message can be encrypted according to the proposed encryption system as plain text. However, since this message can also be represented in soft set form, it would be more efficient to transmit it in soft matrix format. The soft set representation of the message is constructed as follows:

$$P_E = \{e_1 = \{k_2, k_4, k_9\}, e_2 = \{k_1, k_8, k_{11}\}, e_3 = \{k_6\}\}.$$

Company A learns the key and permutation in the soft set format that Company B intends to use, utilizing the Key Exchange Algorithm (KEA) as provided below.

$$K_E^B = \{e_1 = \{k_1, k_3, k_{11}\}, e_2 = \{k_3, k_7, k_9\}, e_3 = \{k_2, k_9\}\}, \qquad \pi = (186)(2734)(59).$$

Subsequently, Company A encrypts the P_E soft set, which indicates the investment proposal, using the main algorithm along with the algebraic structure and alphabet in Example 1 as follows:

$$P_E = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}, \quad C_R(P_E, K_E^B) = \begin{bmatrix} 331 & 320 & 001 \\ 322 & 001 & 330 \\ 331 & 332 & 001 \\ 320 & 001 & 002 \\ 001 & 003 & 001 \\ 000 & 001 & 322 \\ 001 & 332 & 001 \\ 002 & 321 & 003 \\ 321 & 330 & 331 \\ 000 & 001 & 002 \\ 331 & 320 & 001 \end{bmatrix}, \quad C_R^{\pi}(P, F_E) = \begin{bmatrix} 003 & 313 & 102 \\ 330 & 203 & 210 \\ 003 & 313 & 123 \\ 000 & 223 & 010 \\ 000 & 010 & 130 \\ 230 & 020 & 010 \\ 003 & 010 & 123 \\ 003 & 030 & 212 \\ 333 & 213 & 103 \\ 000 & 020 & 010 \\ 003 & 313 & 102 \end{bmatrix}.$$

The final matrix is converted into the encrypted message

$$3\Theta\breve{G}\Upsilon\ddot{U}\ V3\Theta O0\mathrm{\mathring{A}}404\ddot{O}\tilde{N}8434O3CZ\Omega QH0843\Theta\breve{G}.$$

After receiving the encrypted message, Company B performs decryption to learn about the investment proposals.

5. Comparison of the new cryptosystem with others

The proposed encryption system employs a mathematical framework based on soft sets and matrix operations, offering a complex key structure that includes both permutations and random elements within a matrix. The keys are constructed using soft set matrices and operators like the maximum or minimum functions, which apply to elements of $\mathbb{Z}_n$. The encryption process in this system is highly flexible and involves matrix transformations, yielding a vast key space of $2^{qr}(3r)!$, making it resistant to brute force attacks and adding multiple layers of security through randomness and key variability. The encryption and decryption operations use this key space in combination with matrix operators to provide highly secure communication. Due to the enormous size of the key space and its flexibility, this system is much harder to crack compared to some of the encryption methods discussed in the rest of this section.

5.1. Comparison with the known soft set-based encryption systems

In the encryption system proposed by Aygün [36], the soft set used as a key is not encrypted, which means that anyone who knows the soft set can access the main message. Additionally, when the number of letters in the message is not a multiple of five, the letter ‘A’ is appended to the end of the message to make its length a multiple of five. This approach creates ambiguity during decryption, as it becomes unclear whether an ‘A’ character encountered in the decrypted text corresponds to an actual letter in the original message or was added as an extra character to achieve the multiple of five. Another issue arises when both the defined inverse multiplication and characteristic multiplication are applied simultaneously to the message (Theorem 4.1(i_3), [36]), resulting in the complement of the message being obtained as the encrypted text.

In examining the algorithm and examples provided by Paik and Mondal [38], it is evident that the system operates for texts that can be expressed as soft sets. However, there are no details on how a general text to be sent can be expressed as a soft set. Therefore, this method will work if a text to be sent can indeed be expressed as a soft set, that is, if it can be written in the format of parameterized subsets. On the other hand, the system we propose allows for the encryption of any general text in any form. The number of elements in the key space of this system is given as

Theorem 3. *[38] Let the number of elements in the universe set (U) and the parameter set (E) be q and r, respectively. The number of soft sets that can be written with respect to P on E is 2^{qr}.*

On the other hand, the number of elements in the key space of the encryption systems we propose is given as:

Theorem 4. *Let the number of elements in the universe set (U) and the parameter set (E) be q and r, respectively. The number of elements in the key space of the proposed encryption system is $2^{qr}(3r)!$.*

Proof. Since $|S_{3r}| = (3r)!$, by Theorem 3, the result follows. □

Paik and Mondal [38] evaluated their system against other soft set-based encryption methods, proving its greater resilience. Theorem 4 establishes that the key space of the encryption system proposed in this study exceeds that of the one in [38], suggesting that the system introduced here offers enhanced security.

The features of some soft set-based encryption systems known in the literature (including the one presented in this study) are summarized in Table 3, where columns represent some of the known soft set-based encryption systems, Row 1 states the alphabet size, Rows 2 and 3 give the type of text and key used, respectively, Row 4 shows the length of the vector corresponding to the letters of the alphabet, and the last row provides the mathematical tools used.

Table 3. Comparison of some soft set-based encryption systems.

	Aygün [36]	Aygün [37]	Paik and Mondal [38]	The advance system
Alph. size	32	32	-	$n^r, n > 3$
Text type	Plain text	Plain text	Soft set	No restriction
Key type	Soft set	Soft set	Soft set	Soft set & permutation
Vector lenght	5	5	-	r
Mathematical tools	Invers & Char. prod.	Inv. & Char. prod. & perm.	Symmetric differ. op.	Max. op., Min. op. & Perm.

5.2. Comparison with classical encryption systems

Paik and Mondal [38] compared their proposed system with several classical encryption systems in the literature (such as shift, substitution, affine, Vigenere), showing that their system is more robust. Theorem 4 demonstrates that the number of elements in the key space of the proposed encryption system is greater than that of the key space of the one given in [38], indicating that the encryption method proposed in this study is more secure for the particular values of the pair (q, r) discussed in [38]. Below, we provide details of these comparisons for the particular pairs (q, r) shown in Table 1.

We will start by considering the Shift Cipher, a symmetric encryption method where each character in the plaintext is shifted by a constant value $k \pmod{n^r}$ based on a predefined key k. The size of the key space in this system is n^r, where n^r represents the number of characters in the alphabet. The proposed encryption system, due to its complex structure and larger key space, provides a higher level of security and reliability than the Shift Cipher, without any restrictions on q.

The Substitution Cipher is an older and simpler encryption system where each character in the message (plaintext) is substituted with another character from the alphabet based on a predetermined permutation. The encryption function $e_\pi(x)$ maps each plain-text symbol x to a cipher symbol using a permutation π, while the decryption function $d_\pi(y)$ reverses this process using the inverse permutation π^{-1}. The key space for the Substitution Cipher consists of all possible permutations of the alphabet, which results in $(n^r)!$ possible keys. Although this number grows quickly and renders exhaustive key search impractical, the Substitution Cipher is vulnerable to cryptanalysis through other methods, making it less secure despite the large key space. Even though the key space in the Substitution Cipher is large, it is still smaller compared to the proposed system's key space when q exceeds a certain threshold. Table 4 shows specific values of q for which the proposed system is more secure than the Substitution Cipher.

Table 4. Values of q for which the proposed system outperforms Substitution Cipher.

	3^3	3^4	4^3	6^2	7^2	8^2	9^2	10^2
q	> 24	> 93	> 92	> 64	> 99	> 143	> 195	> 257

The Affine Cipher operates over a message and ciphertext space $P = C = \mathbb{Z}_{n^r}$ with a key space defined as $K = \{(a, b) \in \mathbb{Z}_{n^r} \times \mathbb{Z}_{n^r} : gcd(a, n^r) = 1\}$. Affine Cipher's security depends on the key pair (a, b) and uses modular arithmetic, making it relatively straightforward but vulnerable to certain cryptanalysis methods, such as frequency analysis and attacks that exploit the linear nature of the transformation. Table 5 indicates the values of q for which the proposed system is more secure than the Affine Cipher, where a "–" implies that no restrictions on q are needed for the specific alphabet size. For most combinations of n and r, Table 5 shows that the proposed system is inherently more secure without any specific restrictions on q.

Table 5. Values of q for which the proposed system outperforms Affine Cipher

	3^3	3^4	4^3	6^2	7^2	8^2	9^2	10^2
$n^r\phi(n^r)$	486	4374	2048	432	2058	2048	4374	4000
q	-	-	-	-	-	-	> 1	> 1

The Vigenére Cipher is a classical poly-alphabetic cipher that operates over a message space $P = \mathbb{Z}^m_{n^r}$, where m is a positive integer representing the length of the keyword. The key space is also $K = \mathbb{Z}^m_{n^r}$, where each key $k = (k_1, k_2, \cdots, k_m)$ is a sequence of shifts applied to the message characters. The size of the key space is $(n^r)^m$, which becomes large as m, n, and r increase. Table 6 presents values of q for which the proposed system outperforms Vigenére Cipher in terms of security. Given that the number of characters in alphabets around the world is generally limited, Table 6 demonstrates that in the majority of cases, no constraints on q are necessary for the proposed system to offer greater security and reliability than the Vigenère Cipher.

Table 6. Values of q for which the proposed system outperforms Vigenére Cipher.

n^r m	3^3	3^4	4^3	6^2	7^2	8^2	9^2	10^2
1	-	-	-	-	-	-	-	-
2	-	-	-	-	-	> 1	> 1	> 1
3	-	-	-	> 2	> 3	> 4	> 4	> 5
4	-	-	> 1	> 5	> 6	> 7	> 7	> 8
5	> 1	-	> 3	> 8	> 9	> 10	> 11	> 11

Due to the simplicity of the computations in many of the previously mentioned classical encryption systems, they tend to operate faster than the proposed system. However, as previously discussed, this increase in speed often comes at the expense of security. In many cases, these systems are vulnerable to various cryptanalytic attacks, making them less secure compared to the proposed system. The proposed system, while potentially slower in execution, offers enhanced security features, particularly for specific types of data, such as soft set-related data, which are less efficiently handled by traditional encryption methods. Thus, the trade-off between speed and security becomes crucial, with the proposed system providing a stronger defense in scenarios where security is paramount.

The Data Encryption Standard (DES) is a 16-round Feistel cipher that operates on 64-bit plaintext blocks. It encrypts a 64-bit input string, x, using a 56-bit key, K, to produce a 64-bit ciphertext. Before the 16 rounds of encryption, an initial permutation, π, is applied to the plaintext. After the 16 rounds, the inverse permutation, π^{-1}, is applied to the intermediate result to generate the final ciphertext, y. The security of DES has been called into question mainly due to the relatively small size of its key space, 2^{56}. While this was considered secure in the 1970s, advancements in computing have made exhaustive key searches feasible, raising concerns. Furthermore, cryptanalytic techniques like differential cryptanalysis and linear cryptanalysis can be used to break DES more efficiently than brute force. For example, in 1994, Matsui [40] successfully implemented a linear cryptanalysis attack using 2^{43} plaintext/ciphertext pairs, though the practical application of this attack is limited due to the large number of pairs required. The proposed system offers better security compared to DES, particularly when taking into account the alphabet sizes shown in Table 1 (for the pair $(n, r) = (3, 4)$ we require $q > 6$, for $(3, 3)$ and $(4, 3)$, $q > 12$, and for all other pairs, $q > 23$). The large key space and complex encryption process of the system make it impervious to brute force and known cryptanalytic techniques, while DES's relatively small key space and vulnerability to linear cryptanalysis render it outdated by modern security standards.

For more details on the classical encryption systems, we refer to [41].

6. Conclusions and future work

This study provides an in-depth explanation of an advanced encryption system based on soft sets. The new system possesses the following properties:

i) No constraints on the key to be used and/or the text to be encrypted.

ii) Enhanced system reliability through random assignments utilizing the properties of newly defined operators.

iii) The ability to use a specific key for each step/block of the encryption process, as well as the use of different keys (within the framework of certain rules) at each step.

iv) Given the large size of the key space, the difficulty of predicting the key used in this system is currently the highest among existing systems of this type.

v) The use of a permutation from S_{3r}, with the ability to change this permutation at each step (i.e., the creation of a permutation schedule alongside the key schedule within a specific rule framework), makes the system exceptionally difficult to break.

This new system highlights the importance of such systems for the following reasons: (1) the number of such systems is limited (see Section 2), (2) the extensive range of applications for soft sets (see Section 1), and (3) the critical importance of information/data security in this era.

This system works for any positive integers $n(> 3)$ and r, demonstrating that it can be applied to encrypt texts based on any alphabet (or group of alphabets). This observation shows that the number of characters in the language used is not a significant factor for such a system. Furthermore, the system shows that numeric data of any size (by dividing into blocks) can also be easily encrypted outside of specific alphabet-based texts.

The key space of the proposed system is $2^{qr}(3r)!$, and it is evident that it offers significant security advantages compared to other known such systems. It has been demonstrated that the system proposed by Paik and Mondal [38] is stronger than many classical encryption systems. The fact that the key size in the system proposed here is larger than that in the system described in [38] further supports that this proposed system is more robust than the classical encryption systems discussed.

Although the scenario where the key used has not been previously determined between parties is considered, according to Kerckhoffs' principle, if the soft set of the initiating party is known, the system can be easily broken. Therefore, additional protocols for protecting the relevant soft set will be required, and determining what these protocols should entail may be a topic for future research. Is it possible to establish a mathematical theory connecting the soft sets A_E, B_E, (or the permutation π) in such a way that knowing one of the soft sets does not easily lead to determining the other? That is, can we build an asymmetric soft set-based cryptosystem?

Author contributions

All authors contributed equally to this work. All authors have read and approved the final version of the manuscript for publication.

Conflict of interest

The authors declare that they have no conflict of interests.

References

1. D. Molodtsov, Soft set theory-first results, *Comput. Math. Appl.*, **37** (1999), 19–31. https://doi.org/10.1016/S0898-1221(99)00056-5
2. M. Yazdi, E. Zarei, S. Adumene, R. Abbassi, P. Rahnamayiezekavat, Uncertainty modeling in risk assessment of digitalized process systems, *Meth. Chem. Proc. Safety*, **6** (2022), 389–416. https://doi.org/10.1016/bs.mcps.2022.04.005
3. H. Li, M. Yazdi, Stochastic game theory approach to solve system safety and reliability decision-making problem under uncertainty, In: *Advanced Decision-Making Methods and Applications in System Safety and Reliability Problems. Studies in Systems, Decision and Control, Springer*, **211** (2022). https://doi.org/10.1007/978-3-031-07430-1_8
4. E. Zarei, M. Yazdi, R. Moradi, A. BahooToroody, Expert judgment and uncertainty in sociotechnical systems analysis, In: *in Safety Causation Analysis in Sociotechnical Systems: Advanced Models and Techniques. Studies in Systems, Decision and Control, Springer*, **541** (2024). https://doi.org/10.1007/978-3-031-62470-4_18
5. P. K. Maji, R. Biswas, A. J. Roy, Soft set theory, *Comput. Math. Appl.*, **45** (2003), 555–562. https://doi.org/10.1016/S0898-1221(03)00016-6
6. M. I. Ali, F. Feng, X. Liu, W. K. Min, M. Shabir, On some new operations in soft set theory, *Comput. Math. Appl.*, **57** (2009), 1547–1553. https://doi.org/10.1016/j.camwa.2008.11.009

7. J. C. R. Alcantud, A. Z. Khameneh, G. Santos-Garcia, M. Akram, A systematic literature review of soft set theory, *Neural Comput. Applic.*, **36** (2024), 8951–8975. https://doi.org/10.1007/s00521-024-09552-x

8. N. Çağman, S. Karataş, S. Enginoğlu, Soft topology, *Comput. Math. Appl.*, **62** (2011), 351–358. https://doi.org/10.1016/j.camwa.2011.05.016

9. M. Shabir, M. Naz, On soft topological spaces, *Comput. Math. Appl.*, **61** (2011), 1786–1799. https://doi.org/10.1016/j.camwa.2011.02.006

10. S. Nazmul, S. K. Samanta, Neighbourhood properties of soft topological spaces, *Ann. Fuzzy Math. Inform.*, **6** (2013), 1–15.

11. S. Hussain, B. Ahmad, Some properties of soft topological spaces, *Comput. Math. Appl.*, **62** (2011), 4058–4067. https://doi.org/10.1016/j.camwa.2011.09.051

12. W. K. Min, A note on soft topological spaces, *Comput. Math. Appl.*, **61** (2011), 3524–3528. https://doi.org/10.1016/j.camwa.2011.08.068

13. M. Terepeta, On separating axioms and similarity of soft topological spaces, *Soft Comput.*, **23** (2019), 1049–1057. https://doi.org/10.1007/s00500-017-2824-z

14. T. M. Al-shami, M. E. El-Shafei, Partial belong relation on soft separation axioms and decision-making problem, two birds with one stone, *Soft Comput.*, **24** (2020), 5377–5387. https://doi.org/10.1007/s00500-019-04295-7

15. H. Hazra, P. Majumdar, S. K. Samanta, Soft topology, *Fuzzy Inform. Eng.*, **4** (2012), 105–115. https://doi.org/10.1007/s12543-012-0104-2

16. A. Aygünoğlu, H. Aygün, Some notes on soft topological spaces, *Neural Comput. Applic.*, **21** (2012), 113–119. https://doi.org/10.1007/s00521-011-0722-3

17. I. Zorlutuna, M. Akdag, W. K. Min, S. Atmaca, Remarks on soft topological spaces, *Ann. Fuzzy Math. Inform.*, **3** (2012), 171–185.

18. S. Das, S. K. Samanta, Soft metric, *Ann. Fuzzy Math. Inform.*, **6** (2013), 77–94.

19. T. M. Al-shami, L. D. R. Kočinac, The equivalence between the enriched and extended soft topologies, *Appl. Comput. Math.*, **18** (2019), 149–162.

20. J. C. R. Alcantud, Soft open bases and a novel construction of soft topologies from bases for topologies, *Mathematics*, **8** (2020), 672. https://doi.org/10.3390/math8050672

21. J. C. R. Alcantud, An operational characterization of soft topologies by crisp topologies, *Mathematics*, **9** (2021), 1656. https://doi.org/10.3390/math9141656

22. M. Matejdes, Methodological remarks on soft topology, *Soft Comput.*, **25** (2021), 4149–4156. https://doi.org/10.1007/s00500-021-05587-7

23. G. Ali, M. Akram, Decision-making method based on fuzzy N-soft expert sets, *Arabian J. Sci. Eng.*, **45** (2020), 10381–10400. https://doi.org/10.1007/s13369-020-04733-x

24. A. Adeel, M. Akram, N. Çağman, Decision-making analysis based on hesitant fuzzy N-soft ELECTRE-I approach, *Soft Comput.*, **26** (2022), 11849–11863. https://doi.org/10.1007/s00500-022-06981-5

25. J. C. R. Alcantud, G. Santos-Garcia, M. Akram, A novel methodology for multi-agent decision-making based on N-soft sets, *Soft Comput.*, 2023. https://doi.org/10.1007/s00500-023-08522-0

26. A. Kerckhoffs, La cryptographie militaire, *J. Sci. Milit.*, **9** (1883), 5–38.

27. R. L. Rivest, A. Shamir, L. Adleman, A method for obtaining digital signatures and public-key cryptosystems, *Commun. ACM*, **21** (1978), 120–126. https://doi.org/10.1145/359340.359342

28. W. Diffie, M. E. Hellman, Multiuser cryptographic techniques, In: *AFIPS National Computer Conference, New York*, 1976. https://doi.org/10.1145/1499799.1499815

29. M. O. Rabin, Digitalized signatures and public key functions as intractable as factorisation, In: *MIT/LCS/TR-212 MIT Laboratory for Computer Science*, 1979.

30. H. Aktaş, M. Kalkan, An application of the crystography, *J. Math. Comput. Sci.*, **11** (2014), 147–158. http://dx.doi.org/10.22436/jmcs.011.02.07

31. N. Çağman, S. Enginoğlu, Soft matrix theory and its decision making, *Comput. Math. Appl.*, **59** (2010), 3308–3314. https://doi.org/10.1016/j.camwa.2010.03.015

32. F. Feng, J. Cho, W. Pedrycz, H. Fujita, T. Herawan, Soft set based association rule mining, *Knowledge-Based Syst.*, **111** (2016), 268–282. https://doi.org/10.1016/j.knosys.2016.08.020

33. M. Kalkan, Z. Zararsız, Kriptografi, steganografi ve kodlama teorisinin grup teorisi ile ilişkisinin incelenmesi ve bazı uygulamaların geliştirilmesi, *Ph.D thesis, Nevşehir Hacıbektaş University*, 2021.

34. Z. Liu, J. C. R. Alcantud, K. Qin, L. Xiong, The soft sets and fuzzy sets-based neural networks and application, *IEEE Access*, **111** (2020), 268–282. https://doi.org/10.1109/ACCESS.2020.2976731

35. B. K. Tripathy, T. R. Sooraj, R. K. Mohanty, Rough set and soft set models in image processing, In: *Intelligent Multimedia Data Analysis, Berlin, Boston, De Gruyter*, 2019, 123–144. https://doi.org/10.1515/9783110552072-006

36. E. Aygün, Soft matrix product and soft cryptosystem, *Filomat*, **32** (2019), 6519–6530. https://doi.org/10.2298/FIL1819519A

37. E. Aygün, AES encryption and a cryptosystem obtained with soft set II, *Cumhur. Sci. J.*, **40** (2019), 69–78. https://doi.org/10.17776/csj.416395

38. B. Paik, S. K. Mondal, Introduction to soft cryptosystem and its application, *Wirel. Pers. Commun.*, **125** (2022), 1801–1826. https://doi.org/10.1007/s11277-022-09635-9

39. İ. Yılmaz, Esnek kümeler ve vernam şifreleme üzerine, *MSc. thesis, Erciyes University*, 2023.

40. M. Matsui, Linear cryptanalysis method for DES cipher, *Lect. Notes Comput. Sci.*, **765** (1994), 386–397.

41. D. R. Stinson, *Cryptography theory and practice*, Boca Raton: Chapman & Hall/CRC, 2006.

Appendices

A sample alphabet

When examining the languages used in the world, it is observed that the average number of elements in an alphabet is approximately 36. A sample alphabet can be constructed as shown in Table 7, where Column 1 provides the letter's number (corresponding to vector's value in the decimal system), Column 2 gives the letter, Column 3 shows the vector correspondence of the letter, and the last column shows the origin of the alphabet.

Table 7. A sample alphabet for $(n, r) = (4, 3)$.

#	Letter	Vector	Origin	#	Letter	Vector	Origin
0	0	000		32	Ş ş	200	Turkish
1	1	001		33	T t	201	Latin
2	2	002		34	U u	202	Latin
3	3	003		35	Ü ü	203	Turkish
4	4	010		36	V v	210	Latin
5	5	011		37	Y y	211	Latin
6	6	012		38	Z z	212	Latin
7	7	013		39	Q q	213	English
8	8	020		40	X x	220	English
9	9	021		41	W w	221	English
10	A a	022	Latin	42	Ä ä	222	Swedish
11	B b	023	Latin	43	Åå	223	Swedish
12	C c	030	Latin	44	Ñ ñ	230	Spanish
13	Ç ç	031	Turkish	45	ﬂ ﬂ	231	German
14	D d	032	Latin	46	Ă ă	232	Vietnam
15	E e	033	Latin	47	$\hat{A}$ $\hat{a}$	233	Vietnamese
16	F f	100	Latin	48	Đ đ	300	Vietnamese
17	G g	101	Latin	49	$\hat{E}$ $\hat{e}$	301	Vietnamese
18	Ğ ğ	102	Turkish	50	$\hat{O}$ $\hat{o}$	302	Vietnamese
19	H h	103	Latin	51	O’ o’	303	Vietnamese
20	I ı	110	Latin	52	U’ u’	310	Vietnamese
21	İ i	111	Latin	53	Γ γ	311	Greek
22	J j	112	Latin	54	Δ δ	312	Greek
23	K k	113	Latin	55	Θ θ	313	Greek
24	L l	120	Latin	56	Λ λ	320	Greek
25	M m	121	Latin	57	Ξ ξ	321	Greek
26	N n	122	Latin	58	Π π	322	Greek
27	O o	123	Latin	59	Σ σ	323	Greek
28	Ö ö	130	Turkish	60	Υ υ	330	Greek
29	P p	131	Latin	61	Φ ϕ	331	Greek
30	R r	132	Latin	62	Ψ ψ	332	Greek
31	S s	133	Latin	63	Ω ω	333	Greek