Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 237 (2024) 389–396

Procedia Computer Science

www.elsevier.com/locate/procedia

International Conference on Industry Sciences and Computer Science Innovation

Investigation of crypto-algorithms for Stability Assessment

Rohini Hongal[a], Supriya Katwe[1], Sanjana Katawe[1], Priyanka Raykar[1], Rakshita Patil[1], Ranjita Shirol[1], Prabha Nissimagoudar[1], Gireesh M[1], Basawaraj[1], Nalini Iyer[1]

[a]*School of ECE KLE Technological University, Hubballi, 580031, Karnataka, India*

Abstract

Performance evaluation plays a vital role in the field of cryptography. It is essential to assess the security, efficiency, and suitability of different algorithms in various applications. By evaluating the performance of cryptography algorithms, we can identify vulnerabilities, weaknesses, and strengths, which are crucial for ensuring the integrity and confidentiality of sensitive data. Moreover, performance evaluation allows for benchmarking, algorithm selection, and continuous improvement in the field, driving advancements in cryptography techniques and enhancing the overall security of systems that rely on cryptography. This paper implements a test algorithm that can evaluate the performance of different parameters for a range of cryptography algorithms. The performance of input is evaluated using parameters such as the Avalanche effect, Correlation coefficient, Bit independence test, Frequency test, Encryption and decryption execution time, Throughput, Brute Force attack, and Information entropy. By comparing results obtained from the performance tests, we can gain valuable insights into the relative strengths and weaknesses of the algorithms. This aids in making informed decisions regarding algorithm selection and improvement, ultimately contributing to the advancement of secure communication and data protection.

Keywords: security testing; performance evaluation; cryptography; key analysis; comparison

1. Introduction

Cryptography is a technique for safeguarding information by converting plaintext to ciphertext and vice versa. There are two types: symmetric cryptography, which uses a single secret key for encryption and decryption, and asymmetric cryptography, which uses a pair of keys (public and private) for encryption and decryption. Examples of symmetric algorithms include AES and DES, while examples of asymmetric algorithms include RSA and Diffie-Hellman. Public-key cryptography is another name for asymmetric cryptographySoe et al. [15].

Performance evaluation of cryptography algorithms is important for several reasons such as security, performance, and compliance. Cryptography algorithms secure sensitive data such as financial transactions, personal information,

* Rohini Hongal Tel.: +91 9980303578
E-mail address: rohini_h@kletech.ac.in

1877-0509
10.1016/j.procs.2024.05.119

and confidential business data. Therefore, it is essential to ensure that the algorithms used are secure and resistant to attacks by hackers and other malicious actors. Cryptography algorithms can significantly impact the performance of computer systems and networks. Therefore, evaluating their performance is important to ensure they can handle the required workload without causing any significant delays or disruptions. Many industries and government agencies have strict regulations and standards that require the use of specific cryptography algorithms. Therefore, it is important to evaluate the security and performance of these algorithms to ensure that they meet the necessary compliance requirements. The field of cryptography is constantly evolving, and new algorithms are being developed regularly. Security analysis and performance evaluation are necessary to assess the effectiveness and efficiency of these new algorithms compared to existing onesPanda [9].

Overall, performance evaluation of cryptography algorithms is critical for ensuring the confidentiality, integrity, and availability of sensitive data and information while ensuring that computer systems and networks operate efficiently and meet regulatory requirements Hongal et al. [4], Hongal et al. [2], Imdad et al. [6]. A Key Schedule Evaluation Criterion (KSEC) that can evaluate the cryptography properties such as confusion, diffusion, randomness, and independence among sub-keys is discussed in the study Afzal et al. [1]. A comparison has been conducted for these encryption algorithms using evaluation parameters such as encryption time, decryption time, and throughput. From the presented simulation results, it was concluded that AES has better performance than other algorithms in terms of both throughput and encryption-decryption time Panda [9]. The paperThirupalu and Reddy [17] represents the analysis of various encryption algorithms, asymmetric algorithms, and hashing algorithms. The goal of the analysis is to assess their abilities to secure protected data against attacks, as well as their speed and efficiency. A comparison of DES, AES, Blowfish, RC4, HIGH, SF, and SIT is conducted. By using this analysis, the research aims to assess the efficiency and effectiveness of these algorithms in securing cloud services Thabit et al. [16]. An experimental study that examines the sensitivity of a block cipher algorithm to changes in a one-bit input in the key. The experiment focuses on evaluating the key avalanche effect within the proposed algorithmMohamed et al. [7].

2. Preliminaries

The preliminaries required for the proposed test block are detailed in this section.

2.1. Avalanche Effect

The avalanche effect is a fundamental concept in cryptography that refers to the property of a cryptography function. Even a small change in the input (plaintext or key) results in a significant change in the output(ciphertext) Vadaviya and Tandel [18]. For example, if two plaintext messages differ by only one bit, a good encryption algorithm should produce two entirely different ciphertexts, with a probability of being flipped at 50%.

$$Avalanche\ effect = \frac{Number\ of\ flipped\ bits}{Length\ of\ ciphertext} \times 100 \tag{1}$$

2.2. Correlation Coefficient

The correlation coefficient is a statistical method used to measure the strength and direction of the relationship between plaintext and the corresponding ciphertext. It involves calculating the correlation coefficient, if a numerical value equals 0, that means the ciphertext is completely different from the plaintext(i.e. good encryption). If the correlation value equals -1 that means the ciphertext is negative of plaintext. So the success of the encryption process means a smaller correlation value.Mousa et al. [8] Sharma and Garg [12].

$$Correlation\ coefficient(x, y) = \frac{cov(x, y)}{\sigma(x) \times \sigma(y)} \tag{2}$$

Where cov is Covariance, x is plaintext, y is ciphertext, and σ is the standard deviation.

2.3. Information Entropy

It measures the amount of unpredictability in a message or data. The greater the uncertainty or randomness in the data, the higher the entropy. Information entropy is usually measured in bits and is calculated based on the probability of each possible outcome in the data. The more equally distributed the outcomes' probabilities, the higher the entropy. On the other hand, if one outcome is much more probable than the others, the entropy will be lower. Here, the entropy of the ciphertext and key is computedMousa et al. [8].

$$H(X) = -\sum p(x) \times \log_2 \ p(x) \tag{3}$$

Where, H(X) represents the entropy of the random variable X, p(x) is the probability of observing the value x of the random variable X, and $\sum$ denotes the summation over all possible values of X.

2.4. Encryption, Decryption and Total Execution Time

Encryption time is the time taken by a cryptography algorithm to convert plain text into ciphertext. Decryption time is the time taken by a cryptography algorithm to convert ciphertext back into plain text. Total execution time is the time taken by a cryptography algorithm to encrypt and decrypt a message. It can vary depending on the input size, algorithm complexity, and hardware. While a faster execution time can make an algorithm more practical, security should also be considered, including resistance to attacks and proper key managementSoe et al. [15].

$$Total\ execution\ time = Encryption\ time + Decryption\ time \tag{4}$$

2.5. Throughput

In cryptography, throughput refers to the amount of data that can be encrypted or decrypted in a given time. It is typically measured in units of bits per second (bps) or bytes per second (Bps). The throughput of a cryptography algorithm is influenced by various factors, including the algorithm's efficiency, the speed of the computer's processor, the amount of available memory, and the size and complexity of the input data Singh et al. [14].

$$Throughput = \frac{Amount\ of\ data\ processed}{Time\ taken} \tag{5}$$

2.6. Bit Independence Test

Bit independence test is a cryptography algorithm testing technique that evaluates the independence and randomness of individual bits in the algorithm's output Afzal et al. [1]. It aims to verify that each bit is not influenced by other bits or predictable patterns, ensuring the algorithm's strength and security. It calculates the p-value using the chi-square distribution with appropriate degrees of freedom. If the p-value is greater than or equal to 0.01 the key passes the bit independence test.

$$\chi^2 = \sum \frac{(O_i - E_i)^2}{E_i} \tag{6}$$

Where, χ^2 is the chi-square test statistic, $\sum$ represents summation, O_i is the observed frequency of a particular bit pattern and E_i is the expected frequency of that bit pattern under the assumption of independence.

$$p\_value = 1 - CDF(\chi^2, df) \tag{7}$$

Where χ^2 is the calculated chi-square test statistic, df is the degrees of freedom for the test, CDF(χ^2, df) is the cumulative distribution function of the chi-square distribution evaluated at the test statistic.

2.7. Brute Force Attack

A brute force attack in testing the strength of a key involves systematically trying all possible combinations of characters or values to determine the correct key. It is a way to assess the resilience of a key against exhaustive search attacks.

2.8. Frequency Test

Testing the frequency of a key in cryptography involves analyzing the distribution of values or characters within the key. This testing assesses whether the key exhibits an unexpected or biased frequency distribution, which could potentially weaken the security of a cryptography system. Computes p-value using the complementary error function (erfc) of the standardized difference between the frequencies of 1's and 0's, with the standardized difference divided by $\sqrt{2}$. Compare the calculated p-value to a significance threshold (e.g., 0.05). If the p-value is greater than or equal to the threshold, the key is likely to be random Afzal et al. [1].

$$s = \frac{|ones - zeros|}{\sqrt{n}} \tag{8}$$

Where ones is the count of "1" bits in the binary key, zeros is the count of "0" bits in the binary key, and n is the length of the binary key.

$$p\_value = erfc\Big(\frac{s}{\sqrt{2}}\Big) \tag{9}$$

Where erfc is the complementary error function.

3. Methodology

The aim is to design a test block that will test the cryptography algorithms for various parameters, which will help us evaluate the performance of different cryptography algorithms and determine the strength of the generated keys. The detailed methodology is shown in Figure 1. A test block is designed that will test the algorithms for different parameters, and then the results are analyzed. The test block will evaluate parameters such as avalanche effect, correlation coefficient, information entropy of ciphertext, encryption time, decryption time, total execution time, throughput, brute force attack, information entropy of key, frequency test, and bit independence test. The standard algorithms (DES, AES, RSA, SHA-256) are implemented using inbuilt functions in Python and then tested using the test block. These algorithms are evaluated for the following parameters: avalanche effect, correlation coefficient, information entropy of ciphertext, encryption time, decryption time, and throughput. The output of the test block helps to determine the strength of algorithms and analyze performance. In addition to this, data obtained from already proposed algorithms is used to test using these parameters. Proposed AES-128 with dynamic s-box, Modified AES-128 and AES-128 with locally generated key and dynamic s-box, and Modified asymmetric algorithm are the proposed algorithms Rohini et al. [11]-Vadaviya and Tandel [18]. The plaintext and corresponding ciphertext, with one-bit change, and its corresponding ciphertext data from these algorithms are stored in Excel files. These files are used as inputs to the test block and evaluated for avalanche effect, correlation coefficient, and entropy of ciphertext. The output is then compared with standard algorithm output to compare their performance. Excel files containing the public and private keys generated for the Modified asymmetric algorithm are given as inputs to the test block. The parameters

used for testing the strength of the keys are brute force attack, information entropy of key, frequency test, and bit independence test.

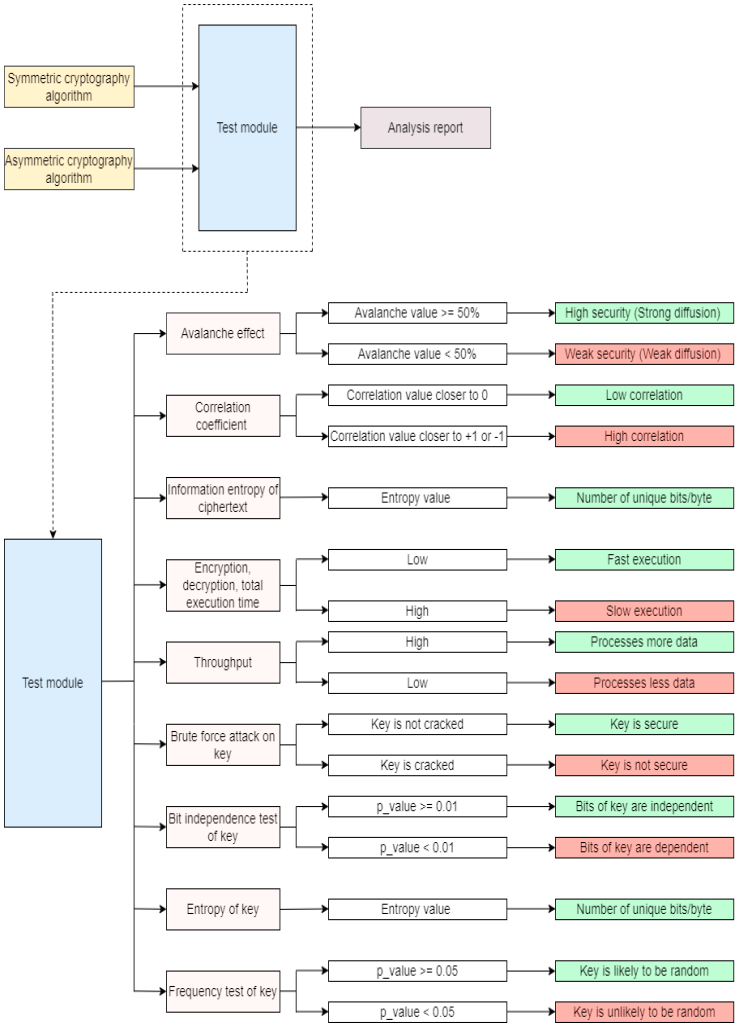

Fig. 1. Methodology

4. Results and Discussions

This section details the results obtained for the performance evaluation of standard cryptography algorithms and some other algorithms and a comparison of the same. All the performance evaluation calculation is done using pandas,numpy, and related crypto libraries.

4.1. Performance Evaluation of Standard Cryptography Algorithms

The results of testing the Standard algorithm are shown in Table. I. Encryption and decryption execution time, Throughput, Avalanche effect, Correlation coefficient, and Information entropy are calculated using standard DES, AES, RSA, and SHA-256 algorithms. DES has a high encryption execution time and SHA-256 has a low encryption execution time. Whereas RSA has a high decryption time and AES has a low decryption time. The total execution

TABLE I. Performance Evaluation of Standard Cryptography Algorithms

Sr. No.	Performance Parameter	DES	AES	RSA	SHA-256
1.	Encryption Execution Time (sec)	0.00151	0.00011	0.00144	9e-05
2.	Decryption Execution Time (sec)	0.00019	0.0001	0.00289	-
3.	Execution Time (sec)	0.0017	0.0002	0.00433	9e-05
4.	Throughput (KB/sec)	4.716	39.337	1.848	120.778
5.	Avalanche Effect (%)	48.438	60.317	50.439	54.297
6.	Correlation Coefficient	0.162	-0.179	0.401	0.269
7.	Information Entropy of Ciphertext (bits/byte)	3.0	3.0	7.07	4.875

TABLE II. Performance Evaluation of Other Cryptography Algorithms

Sr. No.	Performance parameter	Standard AES-128	Proposed AES-128 with dynamic s-box	Modified AES-128	AES-128 with locally generated key and dynamic s-box	Modified asymmetric algorithm
1.	Avalanche Effect (%)	52.865	51.611	48.785	50.434	43.438
2.	Correlation Coefficient	-0.044	-0.005	0.034	-0.051	0.131
3.	Information Entropy of Ciphertext (bits/byte)	3.917	3.904	3.944	3.944	3.987

time of RSA is the highest and SHA-256 is the lowest. SHA-256 processes more data while RSA processes fewer data in a given time. The avalanche effect is high for AES and low for DES. The correlation coefficient of RSA is high compared to other algorithms and DES has a lesser correlation coefficient value. The information entropy of RSA is the highest. DES and AES have the same entropy value which is lower than the others. These standard algorithms have been implemented using inbuilt functions, so the values may vary depending on the software environment used.

4.2. Performance Evaluation of Previously Proposed Cryptography Algorithms

Table. II shows the performance evaluation of previously proposed algorithms. This comparison helps to check the relative strength and weaknesses of that algorithm. Data is obtained from the previously published papersRohini et al. [11]Vadaviya and Tandel [18]. Proposed AES-128 with dynamic s-box, Modified AES-128 and AES-128 with locally generated key and dynamic s-box, and Modified asymmetric algorithm are the proposed algorithms. These are tested using performance parameters such as the Avalanche effect, Correlation coefficient, and Information entropy. The avalanche effect is high for standard AES-128 and low for the Modified asymmetric algorithm. The correlation coefficient of the Modified asymmetric algorithm is high compared to other algorithms, while AES-128 with dynamic s-box has a lower correlation coefficient value. The information entropy of the Modified asymmetric algorithm is highest while AES-128 with dynamic s-box has lower information entropy Hongal and Shettar [5] Hongal et al. [3].

Input, Key and Encrypted texts of previously proposed Algorithms:

The following data is taken from the Rohini et al. [10] Shettar et al. [13] study for analysis.

```
1. AES-128 with Locally Generated Key & Dynamic S-box
Input: 544F4E20776E69546F656E772020656F
Key: 5473206768204B20616D754674796E75
```

```
Encrypted:7B76D81919C48DB922D570067C01C33A

2. Proper AES-128 with Dynamic S-box
Input: A07B00321C11759D0FDE340234384BC9
Key: B9B5ED7585C8B15D7454ED271AA3A3A3
Encrypted:951AFFA13A1900105ED0EC42B7E4

3. Modified AES-128-bit
Input: 544F4E20776E69546F656E772020656F
Key: 7B76D81919C48DB922D570067C01C33A
Encrypted:8CCAF5AE601E02A9A95DEA06F3FD1A0D

4. Modified Asymmetric Algorithm
Input: 2b7e151628aed2a6abf7158809cf4f3c
Key: 0a14e15f0436a39d041ae1bf0a9c6a60
Encrypted:094da72961138b677dbda7e6186a298b
```

4.3. Performance Evaluation of Various Cryptography Algorithms

TABLE III. Performance Evaluation of Various Cryptography Algorithms

Sr. No.	Performance Parameter	Avalanche Effect (%)	Correlation Coefficient	Entropy of Ciphertext (bits/byte)
1.	**DES**	48.438	0.162	3.0
2.	**AES**	60.317	-0.179	3.0
3.	**RSA**	50.439	0.401	7.07
4.	**SHA-256**	54.297	0.269	4.875
6	**Proposed AES-128 with dynamic s-box**	51.61	-0.005	3.904
7.	**Modified AES-128**	48.785	0.034	3.944
8.	**AES-128 with locally generated key and dynamic s-box**	50.434	-0.051	3.944
9.	**Modified asymmetric algorithm**	43.438	0.131	3.987

The results of testing both Standard algorithms and previously proposed algorithms are shown in Table. III. Avalanche effect, Correlation coefficient, and Information entropy are calculated for standard DES, AES, RSA, SHA-256 (implemented using inbuilt ciphers and randomly generated keys), Proposed AES-128 with dynamic s-box, Modified AES-128, AES-128 with locally generated key and dynamic s-box and Modified asymmetric algorithm. Among the previously proposed algorithms (i.e., from Sr.No 6-8), the Proposed AES-128 with dynamic S-box demonstrates good Avalanche value and Correlation Coefficient, while the Modified Asymmetric Algorithm exhibits high Information Entropy.

4.4. Key Analysis

The results of the Key Analysis of private and public keys are shown in Table. IV. Bit Independence Test evaluates the quality and randomness of private and public keys. If the calculated p-value is greater than or equal to 0.01 the generated output is not biased or predictable. In a Brute force attack for private and public keys, all the possible combinations of keys are used to crack the key. In the Frequency test, if the p-value is greater than or equal to 0.05, the key is likely to be random. The keys that pass all the tests are unpredictable and provide a higher level of security.

5. Conclusion

Performance evaluation of cryptography algorithms enables the selection of suitable algorithms, optimization of cryptography processes, and comparison of algorithm performance. By assessing the algorithms on various parameters, performance evaluation contributes to developing robust and reliable cryptography solutions. Here, a test block is implemented, which will test different cryptography algorithms for a set of parameters. The parameters include

TABLE IV. Key Analysis

Sr. No.	Key	Bit Independence Test	Brute Force Attack	Frequency Test
1.	0a14e15f0436a39d041ae1bf0a9c6a60	×	✓	✓
2.	0000000000000000000000000005ebf	×	✓	×
3.	0a3f943d1da52c1d941bfa51a4504009	✓	✓	✓
4.	0000000000000000000000000006edc	×	✓	×
5.	096817f7e07dd525af4dcfc109ac5b70	✓	✓	✓
6.	08bc4c80dcc409dc5ea3103bb4fe2b92	✓	✓	✓
7.	0dd41e2fd430400774b7b29684a1523f	✓	✓	✓
8.	0000000000000000000000000006edc	×	✓	×

encryption time, decryption time, total execution time, throughput, avalanche effect, correlation coefficient, and information entropy of ciphertext. It also includes various tests for key analysis like the bit independence test, brute force attack, information entropy of key, and frequency test. Based on our performance analysis, the proposed AES-128 algorithm with a dynamic S-box has demonstrated favorable results. This indicates that the algorithm possesses enhanced security and robustness compared to other algorithms. The obtained results are compared to assess and gain insights into the performance variations among the different algorithms. It is to be noted that the obtained results may vary based on factors such as hardware specifications, implementation details, and the specific data used.

References

[1] Afzal, S., Yousaf, M., Afzal, H., Alharbe, N., Mufti, M.R., 2020. Cryptographic strength evaluation of key schedule algorithms. Security and Communication Networks 2020, 1–9.

[2] Hongal, R., H, J., Shettar, R., 2018. An approach towards design of n-bit aes to enhance security using reversible logic. Communications on Applied Electronics 7, 7–13. doi:10.5120/cae2018652793.

[3] Hongal, R., Kolhar, R., Shettar, R., 2020. Power-efficient reversible logic design of s-box for n-bit aes, in: Intelligent Computing and Communication: Proceedings of 3rd ICICC 2019, Bangalore 3, Springer. pp. 251–267.

[4] Hongal, R., Matti, N., Shettar, R., 2019. Design of post quantum public key cryptography reliable key generation unit with reversible logic 6, 94–105. doi:10.1729/Journal.20416.

[5] Hongal, R.S., Shettar, R.B., 2020. A power-efficient and quantum-resistant n-bit cryptography algorithm. International Journal of Natural Computing Research (IJNCR) 9, 18–33.

[6] Imdad, M., Ramli, S.N., Mahdin, H., 2022. An enhanced key schedule algorithm of present-128 block cipher for random and non-random secret keys. Symmetry 14, 604.

[7] Mohamed, K., Mohammed Pauzi, M.N., Mohd Ali, F.H., Ariffin, S., et al., . Analyse on avalanche effect in cryptography algorithm. European Proceedings of Multidisciplinary Sciences .

[8] Mousa, A., Faragallah, O.S., El-Rabaie, S., Nigm, E., 2013. Security analysis of reverse encryption algorithm for databases. International Journal of Computer Applications 66.

[9] Panda, M., 2016. Performance analysis of encryption algorithms for security, in: 2016 International Conference on Signal Processing, Communication, Power and Embedded System (SCOPES), IEEE. pp. 278–284.

[10] Rohini, H., Pavankumar, A., Shettar, R.B., 2019a. A novel approach to optimize design of n-bit aes using reversible logic, in: International Conference on Intelligent Data Communication Technologies and Internet of Things (ICICI) 2018, Springer. pp. 996–1005.

[11] Rohini, S., Nischal, G., Shettar, R.B., 2019b. Power-efficient approach to optimize sha-256 bit using reversible logic, in: Computing and Network Sustainability: Proceedings of IRSCNS 2018, Springer. pp. 275–283.

[12] Sharma, M., Garg, R., 2016. Des: The oldest symmetric block key encryption algorithm, in: 2016 International Conference System Modeling & Advancement in Research Trends (SMART), IEEE. pp. 53–58.

[13] Shettar, R.B., et al., 2018. Reversible logic based modified design of aes-cbc mode. Grenze International Journal of Engineering & Technology (GIJET) 4.

[14] Singh, G., Singla, A.K., Sandha, K., 2011. Throughput analysis of various encryption algorithms. IJCST 2.

[15] Soe, T., Mon, S.S., Thu, K.A., 2019. Performance analysis of data encryption standard (des). International Journal of Trend in Scientific Research and Development 3, 1439–1443.

[16] Thabit, F., Alhomdy, S., Jagtap, S., 2021. Security analysis and performance evaluation of a new lightweight cryptographic algorithm for cloud computing. Global Transitions Proceedings 2, 100–110.

[17] Thirupalu, U., Reddy, E.K., 2019. Performance analysis of cryptographic algorithms in the information security, in: IJERT. NCISIOT-2019 Conference Proceedings, pp. 64–9.

[18] Vadaviya, D.O., Tandel, P., 2015. Study of avalanche effect in aes, in: National Conference on Recent Advances in Engineering for Sustainability, pp. 1–4.