International Journal of Distributed and Cloud Computing
7 (1) June 2019, 36-47
http://www.publishingindia.com/ijdcc

Fractional and Self-Adaptive Autoregressive Dragonfly Optimization for Privacy Preserved Data Publishing in Mobile Cloud Computing

Matish Garg[1*] and Rajender Nath[2]
[1]Ph. D. Scholar, Department of Computer Science and Applications, Kurukshetra University, Kurukshetra, Haryana, India. Email: matish1981@gmail.com
[2]Professor, Department of Computer Science and Applications, Kurukshetra University, Kurukshetra, Haryana, India. Email: rnath2k3@gmail.com
*Corresponding Author

Abstract: The advancement in Mobile Cloud Computing (MCC) has gained immense knowledge in computing concept for upcoming generation. The wireless communications enable the integration of cloud computing and mobile to generate MCC. Privacy and security are the major issues faced by MCC while publishing data. This work introduces a technique, named Self-Adaptive Autoregressive Dragonfly Optimization (S-ADO), for addressing the issues by determining the secret key optimally using retrievable data perturbation technique for privacy preserved data publishing in MCC. The retrievable data perturbation is performed using fractional theory and matrix product based model with proposed S-ADO. The proposed S-ADO is developed by modifying ADO by making it self-adaptive. Initially, a fitness function is computed using privacy & utility parameters for determining the optimal differential derivative coefficients. The optimal coefficients are used to generate the secret key by using fractional theory. Then the matrix product based model is adapted to convert original data into privacy preserved data. The secret key derived using utility and privacy functions, is also used to recover the original data. The performance of the proposed S-ADO algorithm shows superior performance with privacy and utility values as 0.7855, and 0.7088 respectively.

Keywords: Data perturbation, Fractional theory, MCC, Self-Adaptive, Utility.

I. Introduction

Mobile Cloud Computing (MCC) is the incorporation of cloud computing in a wireless network that improves the performance of mobile devices like smartphones, and Personal Digital Assistant (PDA). The intrinsic defects of mobile devices such as inadequate storage space, usage of battery energy, insufficient sensing capacities, and low CPU speed in mobile applications are tackled with respect to multiple challenges, such as Quality of Service (QoS), mobility management, energy management, and issues related to security. The MCC addresses these issues, and it offloads component to be implemented on prevailing nodes on the cloud and has several advantages with respect to the traditional mobile services [1]. MCC promotes user's experience by enriching the mobile applications to be implemented in mobiles and local infrastructures. The mobile devices, like PDA, smartphones, sensors, and tablets exchange the information with local infrastructure using the mobile networks. This type of execution can increase the usage of mobile devices and enrich the client experience that includes advanced games, educational applications, or medical applications [2]. Various techniques related to data publishing in the MCC model consider multiple objectives, such as energy, cost, and execution time, for addressing the optimization problem [3].

A large amount of data requires a publishing system for publishing it and thus, to seek the best technique for making effective decisions, the data is made accessible for all the users. The detailed data in its original form consists of sensitive data about the individual user and publishing these data without any protection smashes the privacy of the individual [4]. The massive data is explored focusing on two parameters that are privacy and privacy protection. The privacy is more likely to be disclosed if data is stored for specific purpose, but used for different purpose [5]. Here, the privacy is desecrated due to three types of attacks, which involve background knowledge attack, linking attack, and homogeneity attack [6]. Thus, it is an important limitation, which must be considered for secondary data using privacy preservation and data mining methods [7]. In spite of the several advantages, there exist certain limitations over data privacy while adapting services related to cloud storage. The data collected before encryption from distant cloud storage server is susceptible to external and internal attacks initialized by unreliable service providers. Various data anonymization methods [9] [10] [11] as well as privacy

preservation models [12] [24] are devised in the literature for achieving the privacy in cloud servers, but attaining an improved result is a challenge. Thus, an effective privacy preserving approach [13] [14] is required for improved preservation of the data [8]. The cloud computing security is solved in several ways as confidentiality, integrity, and authentication [16], which involves on demand and advance-reservation based access to the individuals and grouping resources from different clouds for delivering the workflow results and handle privacy and security [15].

The model of the MCC used in privacy preservation system is represented in Fig. 1. The model contains three important aspects, which are data owner, cloud server, and cloud user. The users, who are responsible for accessing the information stored in the cloud server, are termed as cloud users. The cloud user is responsible for requesting a service that is coordinated with respect to the user query in the database using query mapper and request handler for accessing the needed information. The cloud server consists of servers, which contain physical servers and virtual servers. An effective privacy preservation technique is required for the owners to preserve the cloud data and to send cloud users or the end users. The end users use the information for developing the business, but the sensitive data is hidden from the individual.

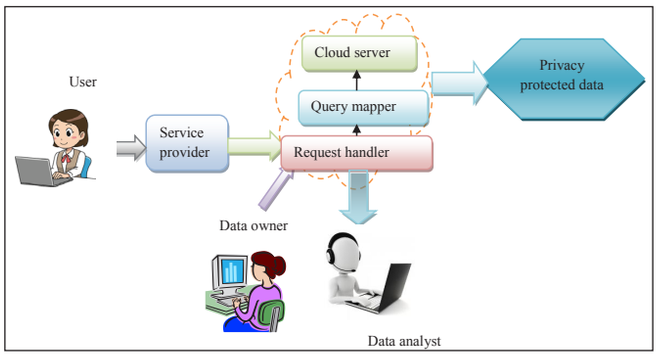

Fig. 1: Model of MCC in Privacy Preservation System

Thus, it is a major challenge to convert the original data to a sanitized data by hiding the confidential information. In addition, protecting the sensitive information requires data maintenance and utility so that the data given to the cloud user is valuable. Moreover, the original data from sanitized data can be retrieved anytime using a key generated by the data owners whenever the user desired.

This paper presents a privacy preservation technique by proposing an optimization algorithm, named S-ADO, for generating optimal fractional order coefficients with a newly designed fitness function using utility and privacy parameters. The proposed S-ADO is the incorporation of self adaptive concept in ADO algorithm, which aims to preserve the privacy of data to provide secure transmission for cloud users. Here, the original data is converted to secure encrypted data with a set of operations by using a secret key. The data is encrypted by showing only the necessary details to the mobile users without exposing the original data. Then, the generated key is utilized to recover the source data from the encrypted data.

The major contributions of this work include:

- Designing a Self-Adaptive Autoregressive Dragonfly Optimization (S-ADO) technique by adapting matrix product based model and by incorporating the self adaptive concept in ADO algorithm that reveal only the required details by hiding the sensitive information and to provide secure transmission among the mobile users, whose data is stored in the cloud platform.
- Devising a new objective function using the privacy preservation rate and utility missing rate for evaluating the optimization algorithm to determine the optimal derivative coefficients which are further used to generate fractional matrix.

The paper is structured as follows: Section I provides the introduction about MCC and its applications in daily lives, Section II represents the literature survey with different researches based on privacy preservation for data publishing in cloud computing and the challenges faced by the techniques. Section III describes the proposed S-ADO technique for secure transmission by selecting optimal fractional derivative order coefiicients, Section IV explains results of the proposed S-ADO and compare them with the existing methods, and lastly Section V concludes the paper.

II. Motivation

A. Literature Review

In this section, the literature review is elaborated in which different methods utilized for privacy preserved data publishing are briefly explained along with the challenges.

Li, T., *et al.* [4] designed a Cryptographic Data Publishing System (CDPS), which protect the data and attains anonymity without removing any attribute. The method integrates cryptographic methods with data publishing for enabling privacy preservation. The method has the ability to attain the requirements of data publishing using data mining methods and protects the original data. But the method failed to consider homomorphic encryption mechanism while publishing data.

Hammami, H. [17] developed a method, named Cloud Privacy Preservation Data Mining (Cloud-PPDM), which integrates the refining of frequent closed patterns in a distributed infrastructure like a cloud. The method maintained the privacy of sites during data mining in cloud infrastructure by homomorphic encryption. But it failed to consider prospects for supporting the sovereignty of the data in sites.

Jadhav, R. H. [18] designed a data anonymization technique, named distributed Bottom-up technique to process huge scale data. This technique starts processing from the bottom element of the tree, which is child node and is substituted with its parent node. The distributed data anonymization enhances the scalability and efficiency of bottom-up method over existing methods using Map Reduce framework and is implemented using k-anonymity with huge data for privacy preservation.

Wang, Z., *et al.* [19] developed a framework, Distributed Agent-based privacy-preserving framework (DADP) for yielding strong privacy preservation while using the untrusted server. The DADP method considers multiple agents to provide an interface between the user and the untrusted server. Particularly, a distributed budget allocation method and agent-based dynamic grouping method were introduced for realizing the differential privacy in a dispersed method.

Yu, F., *et al.* [20] developed a clustering perturbation algorithm for preserving privacy in the cloud using community structures. The algorithm used a mechanism for exchanging the attributes to persuade attackers for searching the false targets and to handle the network structure. The algorithm validation is based on the correlations between certain aspects in the local region, which causes divergence to some extent.

Sreedhar, K. C., *et al.* [21] developed two methods, named Bottom–Up Generalization (BUG), and Top-Down Specialization (TDS) methods, for anonymizing huge amount of data. In this method, a pseudo-identity is used to provide privacy preservation with high utility of data on incremental datasets. The genetic model was utilized for indexing and updating the incremental datasets.

Kalidoss, T., *et al.* [22] developed a distributed security processing method for enhancing the cloud networks security that contains two stages, namely verification and encryption. A triple advanced encryption standard algorithm is utilized for performing secured key distribution on the basis of the secured hash algorithm on verification and this method used Map Reduce techniques for faster execution. The method failed to consider intelligent agents for improving the distributed processing.

Li, J., *et al.* [23] proposed a framework, named Preserving Multiparty Data Privacy (PMDP) for privacy preservation in cloud computing. PMDP preserve numeric data computing and publishing in untrusted cloud server and attains delegation of storage. Various cryptography primitives and differential privacy techniques are used in privacy preservation. The security enhanced PMDP framework, named sPMDP, was adapted to provide reliable transmission and to resist the malicious users.

Karlekar, N. P. and Gomathi, N. [27] developed a technique, named PUBAT algorithm, for preserving the privacy in cloud infrastructure by generating coefficient vector using Kronecker product and Bat algorithm. The privacy preservation method follows two steps: Firstly, the PU coefficient is obtained optimally using PUBAT algorithm based on the fitness function and secondly, the input data and the selected PU coefficient is used for determining the privacy protected data for publishing data in cloud infrastructure.

Ashok George and A. Sumathi [31] developed a technique, named Crow search based Lion (C-Lion algorithm), for privacy protection by utilizing the dyadic product. Here, the original database is preserved by adapting dyadic square matrix, which is generated by taking a dyadic product between two vectors, namely Sensitive-Utility (SU) coefficient and cumulative data key product. Here, the SU coefficient is chosen using the proposed C-Lion algorithm, which is constructed by integrating Crow Search Algorithm (CSA) with Lion algorithm.

B. Problem Formulation

Various techniques are devised for preserving the privacy of the data, wherein a lot of challenges are faced by these privacy preservation methods. Most of the methods found limitations regarding the data security as they revealed the confidential data to the third party and there is no assurance to the security of the data. In [31], a method for privacy preservation is designed by adapting dyadic product and an optimization algorithm, C-Lion algorithm, in the cloud environment for data publishing. Here stagnation in local optima and slow convergence speed are two probable issues to deal with challenging optimization problems. However, it requires further performance improvement that can be provided by the inclusion of advanced concepts for publishing the data to the third party. Similarly in [27], PUBAT algorithm can provide very quick convergence at a very initial stage which may lead to stagnation. These problems are addressed in the proposed technique by developing an improved technique with hybrid optimization algorithm for the privacy preservation.

III. Proposed Self-Adaptive Autoregressive Dragonfly Optimization (S-ADO) Technique for Privacy Preservation in the Mobile Cloud Computing

To address the problems as discussed in the previous section, a privacy preservation technique S-ADO is proposed here for secure data publishing in MCC. For converting original data to a sanitized data, a model is built to publish data to the end user without revealing the sensitive information by maintaining privacy and utility parameters. The proposed S-ADO algorithm generates optimal fractional order derivative coefficients that are used to generate the fractional matrix. The fractional matrix is further used to generate secret key for converting original data into privacy protected data. The block diagram of privacy preservation system based on proposed S-ADO is depicted in Fig. 2.

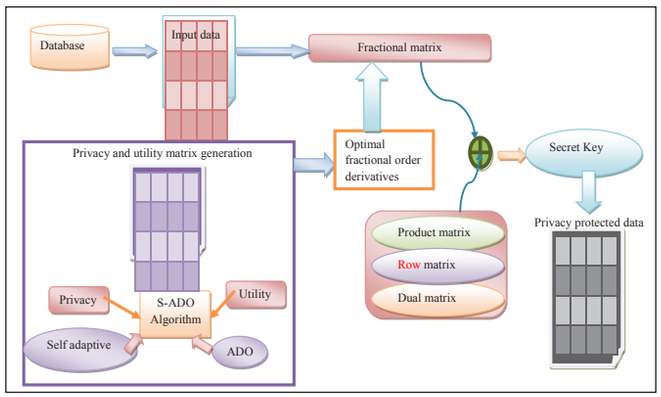

Fig. 2: Block Diagram of Privacy Preservation System Based on Proposed S-ADO

Consider an input data A of size $P \times Q$ having total P records and Q attributes and the key used for sanitizing and retrieving the original data is denoted as M with size $1 \times Q$. Thus, the privacy preserved data are obtained by the element-wise multiplication of input data, and the key as given in equation (1).

$$R_{P\times Q} = A_{P\times Q} \ddot{\otimes} M_{1\times Q} \tag{1}$$

where, $R_{P\times Q}$ denotes the sanitized data matrix, $A_{P\times Q}$ indicates the input data matrix, and $M_{1\times Q}$ represents the key for converting original data into sanitized data.

The key generated for preserving the original data is as follows: Initially, the XOR operation of fractional matrix, and product matrix, is done and the result, together with row matrix and dual matrix are processed using element-wise XOR operation. Hence, the key to be generated is formulated as,

$$M_{1\times Q} = (J_{1\times Q} \oplus K_{1\times Q}) \oplus A^J_{1\times Q} \oplus N_{1\times Q} \tag{2}$$

where, $J_{1\times Q}$ represents the product matrix, $K_{1\times Q}$ denotes the fractional matrix, $A^J_{1\times Q}$ represents the row matrix, and $N_{1\times Q}$ represents the dual matrix. The product matrix is obtained by xoring row matrix and dual matrix and here, the product matrix $J_{1\times Q}$ is represented as,

$$J_{1\times Q} = A^J_{1\times Q} \oplus N_{1\times Q} \tag{3}$$

The row matrix is generated by the summation of the elements in the rows of the input matrix with P number of records and is represented as,

$$A^J_{1\times Q} = \sum_{s=1}^{P} A_s \tag{4}$$

The dual matrix is computed by the summation of elements in the rows of the squared matrix $T_{Q\times Q}$ with Q number of attributes and is represented as follows,

$$N_{1\times Q} = \sum_{s=1}^{Q} T_s \tag{5}$$

The squared matrix $T_{Q\times Q}$ is formulated by the matrix multiplication of transpose of row matrix $(A^J_{1\times Q})^T$, row matrix $A^J_{1\times Q}$ and transpose of unity matrix, denoted as $U^T_{Q\times Q}$. Thus, $T_{Q\times Q}$ is computed as,

$$T_{Q\times Q} = (A^J_{1\times Q})^T \otimes A^J_{1\times Q} \otimes U^T_{Q\times Q} \tag{6}$$

The fractional matrix is computed using Fractional Calculus (FC) [30], which is used to improve the performance of the algorithm. FC algorithm can solve integral as well as a derivative equation, and provides a smoother variation for evaluating the differential derivative. The fractional matrix is constructed based on the squared matrix as,

$$K = \beta T_1 + \frac{1}{2!}\beta T_2 + \frac{1}{3!}\beta(1-\beta)T_3 +,\ldots, + \frac{1}{Q!}\beta(1-\beta)(2-\beta)..((Q-2)-\beta)T_Q \tag{7}$$

$$K = \mu_1 T_1 + \frac{1}{2!}\mu_1 T_2 + \frac{1}{3!}\mu_2 T_3 +,\ldots,\frac{1}{Q!}\mu_{Q-1} T_Q \tag{8}$$

where $\mu_1 = \beta$, $\mu_2 = \beta(1-\beta)$, and $\mu_{Q-1} = \beta(1-\beta)(2-\beta)\ldots .(Q-2-\beta)$ and β indicates the order that varies from 0 to 1.

Here, the parameters, such as $\mu_1, \mu_2, \mu_3, \ldots, \mu_Q$ are optimally determined by adapting the proposed S-ADO algorithm.

In the data retrieval process, the original input data is recaptured from the sanitized data by using the secret key. For retrieving the original data, the element-wise division of sanitized data and the secret key is adapted. Thus, the user acquires the original input data as,

$$A^*_{P\times Q} = \frac{R_{P\times Q}}{M_{1\times Q}} \tag{9}$$

where, $R_{P\times Q}$ represents the sanitized data, and $M_{1\times Q}$ denotes the secret key used to retrieve the original data.

A. Proposed Self-Adaptive Autoregressive Dragonfly Optimization for Generating Optimal Derivative Coefficients

The proposed S-ADO algorithm is developed by combining Conditional Autoregressive Value at Risk (CAViaR [26] with Dragonfly Algorithm (DA) [25] for selecting the optimal derivative orders. DA solves binary and multi-objective problems and initiates optimization procedure by constructing random solutions set for solving the issues based on optimization. DA is considered as the popular search algorithm that is designed by adapting certain principles, which involves separation, alignment, cohesion, attraction and distraction. CAViaR model is motivated from the distribution of returns directly to the behavior of the quantile. The algorithm incorporates self adaptive concept in ADO by generating the optimal derivative coefficients based on a multi-objective fitness function. ADO provides improved convergence to the global optimum, a further improvement in the algorithm can be attained using the CAViaR model. The solution representation, fitness evaluation, and algorithm for the proposed S-ADO are described in the below section.

i) Solution Representation

The purpose of solution representation as depicted in Fig. 3 is to obtain optimal μ from the derivatives for generating the effective privacy & utility coefficients. The solution is determined as a vector with r solutions, each of dimension $1 \times Q$, where Q denotes the total attributes in the database such that $1 \le r < Q$. The fitness function is calculated using the utility and privacy parameter for obtaining the optimal derivative coefficients.

μ_1	μ_2	$\cdots$	μ_r	$\cdots$	μ_{Q-1}

Fig. 3: Solution Representation

ii) Fitness Evaluation

The proposed S-ADO algorithm uses the fitness function that finds the optimal derivative orders for securing sensitive data based on the privacy protection parameters. Here, two important parameters i.e. privacy and utility are used for generating optimal privacy utility coefficients. The data requested by the user must comprise maximal privacy and utility. The fitness function is represented as,

$$T = \frac{1}{2}\left[\alpha_1 \cdot S_1 + \alpha_2 \cdot S_2 + \alpha_3 \cdot D_1 + \alpha_4 \cdot D_2\right] \tag{10}$$

where, $\alpha_1 + \alpha_2 + \alpha_3 + \alpha_4 = 1$, privacy is evaluated by two factors, S_1 and S_2, and utility by the factors, D_1 and D_2

a) Privacy

The privacy parameter is evaluated based on two factors, where the first factor is based on the ratio of the dissimilarity between the original data, and the sanitized data to the maximum of either original data or privacy preserved data. Thus, it measures the overlapping similarities between original data and sanitized data. The first privacy factor is represented as,

$$S_1 = \frac{1}{P \times Q} \sum_{s=1}^{P} \sum_{t=1}^{Q} \frac{A_{s,t} - R_{s,t}}{Max(A,R)} \tag{11}$$

Cosine similarity is a measure which is used for information retrieval. Using this measure, a similarity between two data is derived by computing the cosine values. The cosine similarity measure can be applied to any texts, sentence, paragraphs, or huge data. The lowest cosine similarity score between the data yields more relevancies. The original data and privacy preserved data must contain maximum differences and thus, it is subtracted from the cosine similarity is considered. Hence, S_2 is formulated as,

$$S_2 = \left[1 - CoS(A,R)\right] \tag{12}$$

where, CoS represents cosine similarity measure.

b) Utility

The classification accuracy of privacy preserved data is used to measure the first utility parameter. The maximum utility is obtained using the classification accuracy and is formulated as,

$$D_1 = CA(R) \tag{13}$$

The second utility parameter is the dissimilarity between the mean of original data and privacy protected data and the obtained data is normalized using a normalization function. The second utility factor is given by,

$$D_2 = \left[1 - \frac{\left[X_A - X_R\right]}{W_F}\right] \tag{14}$$

where, X_A and X_R represents mean value of input data and mean value of privacy protected data.

iii) S-ADO Algorithm

The proposed S-ADO algorithm is designed by enhancing the ADO, which is the integration of DA [25] and CAViaR [26], on the basis of the self-adaptive concept. The algorithm is responsible for making ADO self adaptive by adjusting the control parameters in such a way that the algorithm enhances the convergence speed by providing the optimal solution.

The steps contained in proposed algorithm are illustrated below,

a) Population Initialization

Initially, the population of dragonflies are initialized randomly and is expressed as given below,

$$Y = \left\{Y_1, Y_2, \ldots, Y_w, \ldots, Y_z\right\}; 1 < w \le z \tag{15}$$

where, Y_w represents w^{th} solution from the population and z indicates the total population size where $1 < w \le z$.

b) Fitness Evaluation

Once the population is initialized, the fitness for each and every solution is evaluated using the formula derived for fitness as given in equation (10). The newly designed fitness function is used to update the position of dragonflies to obtain the best solution.

c) Evaluation of Swarm Behavior Using Three Principles

The algorithm pursues three principles that involve cohesion, alignment, and separation for describing the swarm behavior and are stated as follows,

- *Separation:* The separation is a principle, in which the static collision of an individual is avoided from other individuals in the surroundings.

$$B_w = -\sum_{y=1}^{O} Y - Y_w \tag{16}$$

where, Y represents the position of an individual, and O denotes the neighborhood.

- *Alignment:* The matching of velocities of individuals from other individuals from the neighborhood is calculated as,

$$H_w = \sum_{y=1}^{O} I_y \tag{17}$$

where, I_y expresses the velocity of y^{th} individual.

- *Cohesion:* It is the preference of individuals to move towards the centre of mass of neighborhood.

$$L_w = \frac{\sum_{y=1}^{O} Y_w}{O} - Y \tag{18}$$

d) Update Position Based on the Self Adaptive Concept

Two important factors are responsible for updating the positions, which are attraction, and distraction. The attraction factor states that the attraction towards food, expressed as E_w and distraction factor states distract from an enemy, expressed as F_w and is formulated as,

$$E_w = Y^* - Y \tag{19}$$

$$F_w = Y^- + Y \tag{20}$$

where, Y^* denotes the best position, and Y^- indicates the worst position. The step vector denoted as ΔY for next iteration is stated for evaluating the dragonfly moves and is represented as,

$$\Delta Y(l+1) = \left(bB_w + cH_w + dL_w + eE_w + fF_w + Z\Delta Y(l)\right) \tag{21}$$

where, l indicates the current iteration, b represents the weight based on separation, B_w represents the separation from w^{th} individual, c indicates the weight adapted on alignment, H_w expresses the alignment of w^{th} individual, d represents the weight based on cohesion, L_w indicates the cohesion of w^{th} individual, e represents the food factor, f denotes the enemy factor, and Z denotes the weight of inertia. The position of the individuals at the current iteration is given as,

$$Y(l) = Y(l+1) - \Delta Y(l+1) \tag{22}$$

where, $\Delta Y(l+1)$ denotes the step vector at iteration $l+1$, and $Y(l)$ denotes the individual position at current iteration.

CAViaR method represents the progress of quantile using an autoregressive method, and the unknown parameters are determined using the quantiles. The adaption of CAViaR in DA improves the overall performance. The model is represented as follows,

$$Y(l+1) = \alpha_0 + \alpha_1 Y(l) + \alpha_2 Y(l-1) + \alpha_1 f\left(Y(l)\right) + \alpha_2 f\left(Y(l-1)\right) \tag{23}$$

where, $\alpha_0, \alpha_1, \alpha_2$ represent self adaptive constants used for representing a vector of unknown parameters, $f\left(Y(l)\right)$ represents the fitness of the solution at the current iteration, and $f\left(Y(l-1)\right)$ indicates the fitness of the solution at iteration $l-1$.

In proposed S-ADO, the update is performed using ADO algorithm by making the constants self adaptive in ADO. The obtained constants which are self-adaptive is represented as α_0, α_1, and , α_2 and are formulated based on the positions as,

$$\alpha_0 = \frac{\| Y(l) - Y(l-1) \|}{\| Y(l-1) - Y(l-2) \|} \tag{24}$$

$$\alpha_1 = \frac{\| Y(l-1) - Y(l-2) \|}{\| Y(l-2) - Y(l-3) \|} \tag{25}$$

$$\alpha_2 = \frac{\| Y(l-2) - Y(l-3) \|}{\| Y(l-3) - Y(l-4) \|} \tag{26}$$

After subtracting $Y(l)$ in both sides of the above equation, the equation obtained is given as,

$$Y(l+1) - Y(l) = \begin{bmatrix} \alpha_0 + \alpha_1 Y(l) + \alpha_2 Y(l-1) \\ + \alpha_1 f\left(Y(l)\right) + \alpha_2 f\left(Y(l-1)\right) \end{bmatrix} - Y(l) \tag{27}$$

The position update of S-ADO algorithm after substituting equation (22) in equation (27) is represented as follows,

$$Y(l+1) - Y(l) = \begin{bmatrix} \alpha_0 + \alpha_1 Y(l) + \alpha_2 Y(l-1) \\ + \alpha_1 f\left(Y(l)\right) + \alpha_2 f\left(Y(l-1)\right) \end{bmatrix} - Y(l+1) + \Delta Y(l+1) \tag{28}$$

After rearranging the above equation, the final equation is stated as follows,

$$Y(l+1) = \frac{1}{2}\begin{bmatrix} Y(l) + \alpha_0 + \alpha_1 Y(l) + \alpha_2 Y(l-1) \\ + \alpha_1 f\left(Y(l)\right) + \alpha_2 f\left(Y(l-1)\right) + \Delta Y(l+1) \end{bmatrix} \tag{29}$$

where, $Y(l)$ denotes the position of the individual at the l^{th} iteration, $Y(l-1)$ is the position of the individual at $(l-1)^{th}$ iteration, $Y(l+1)$ indicates the position of the individual at iteration $l+1$, and $\Delta Y(l+1)$ represents the step vector. For improving the exploration, and randomness, the algorithm uses the Levy flight as given in equation (30) for expanding the search space. Hence, the position update is given as,

$$Y(l+1) = Y(l) + Levy(h) \times Y(l) \tag{30}$$

where, h denotes the position vector size, and the levy flight is evaluated as,

$$Levy(h) = 0.01 \times \frac{q_1 \times \beta}{|q_2|^{\frac{1}{\chi}}} \tag{31}$$

where, q_1, q_2 represent two numbers ranging between 0 and 1, χ = 1.5, and β is formulated as,

$$\beta = \left(\frac{\Gamma(1+\chi) \times \sin\left(\frac{\pi\chi}{2}\right)}{\Gamma\left(\frac{1+\chi}{2}\right) \times \chi \times 2^{\left(\frac{\chi-1}{2}\right)}} \right) \tag{32}$$

where, $\Gamma(h) = (h-1)!$.

e) Finding the Best Solution

After evaluating the updated position, the fitness of individuals is calculated and the solution yielding maximum fitness is considered as the best solution.

f) Termination

The algorithm is terminated after reaching the maximum iteration $l_{\max}$ or in the case, when no more fittest solution is obtained.

The algorithmic procedure is described using the pseudo code given in Table I.

TABLE I: PSEUDO CODE OF S-ADO ALGORITHM

	Proposed S-ADO Algorithm
1	Input: Random population $Y = \{Y_1, Y_2, \ldots, Y_w, \ldots, Y_z\}; 1 < w \leq z$
2	Output: Best position Y^*
3	Parameters: Separation B, Alignment H, Cohesion L, Food factor E, and enemy position F
4	Begin
5	Initialize the population and the step vectors
6	for $(l < l_{\max})$
7	for each solution in the population
8	Compute the fitness using equation (10)
9	Update A, G, H, B, and D using equations (16), (17), (18), (19) and (20).
10	Update the position using equation (29)
11	$l = l + 1$
12	end for
13	Return the best solution
14	end for
15	Terminate

IV. EXPERIMENTAL EVALUATION OF S-ADO TECHNIQUE

The proposed S-ADO technique is implemented in the JAVA framework. The NetBeans is considered as a development tool for the implementation. The experimentation is carried out using two datasets adapted from the UCI Machine Learning Repository. The datasets utilized are Bank Marketing Data Set [28] (Dataset 1), DBworld e-mails data set [29] (Dataset 2).

- Dataset 1: This dataset is taken from the marketing operation of a Portuguese banking organization. The instances contained in this dataset are 45211 with 17 attributes. The dataset feature is multivariate and the attribute characteristic is real. The input attributes contained in the dataset are age, job, marital, education, housing and so on. The number of web hits is 683083. The purpose of the classification is to predict if the client is subscribed to the term deposit.
- Dataset 2: The dataset is donated by Michele Filannino done Ph.D from University of Manchester Centre for Doctoral Training. The Dataset 2 contains 64 instances and number of attributes is 243. The two classes are taken into consideration. The collected information is used to train different algorithms for classification. The binary bag-of-words representation is used in stop word removal pre-processing task.

A. Comparative Analysis

The performance of the proposed S-ADO technique is compared with that of the existing techniques, PUBAT [27], Crow search based Lion (C-Lion) [31] and DA [25] using two metrics i.e. privacy and utility parameters as given in equations (11), (12), (13) & (14) respectively under Section *Fitness Evaluation*.

i) Analysis Based on Dataset 1

The comparative analysis of PUBAT, C-Lion, and DA with the proposed S-ADO method using dataset 1 is elaborated using Fig. 4. The data percentage is varied from 50 to 90 for evaluating the privacy and utility of the comparative methods. The analysis based on the first privacy parameter S_1 using dataset 1 for PUBAT, C-Lion, DA and proposed S-ADO method is depicted in Fig. 4a. When the data percentage is 50, the corresponding values of S_1 using PUBAT, C-Lion, DA and proposed S-ADO method are 0.0012, 0.0015, 0.0286, and 0.0324. Similarly, the values of S_1 for training data percentage 90 computed by PUBAT, C-Lion, DA and proposed S-ADO method are 0.0012, 0.0015, 0.0015, and 0.02783. In Fig. 4b, the analysis based on the first utility parameter D_1 for PUBAT, C-Lion, DA and proposed S-ADO method using dataset 1 is depicted. When the data percentage is 60, the corresponding values of D_1 measured by PUBAT, C-Lion, DA and proposed S-ADO method are 0.5614, 0.5877, 0.6165, and 0.7071. Likewise, when the data percentage is 80, the values of D_1 measured by PUBAT, C-Lion, DA and proposed S-ADO method are 0.5390, 0.5421, 0.6892, and 0.7595. The analysis in terms of the second privacy parameter, S_2 using dataset 1 for PUBAT, C-Lion, and DA and proposed S-ADO method is depicted in Fig. 4c. When the data percentage is 70, the corresponding values of S_2 using PUBAT, C-Lion, and DA and proposed S-ADO method are 0.0096, 0.3508, 0.7588, and 0.7639. Similarly, the values of S_2 for training data percentage 80 computed by PUBAT, C-Lion, and DA and proposed S-ADO method are 0.0097, 0.0907, 0.7588, and 0.7638. The analysis based on second utility parameters D_2 for PUBAT, C-Lion, DA and proposed S-ADO

method using dataset 1 is depicted in Fig. 4d. When the data percentage is 50, the corresponding values of D_2 measured by PUBAT, C-Lion, DA and proposed S-ADO method are 1.00E-03, 0.1459, 0.1839, and 0.6041. Likewise, when the data percentage is 90, the values of D_2 measured by PUBAT, C-Lion, and DA and proposed S-ADO method are 1.00E-03, 0.1456, 0.1839, and 0.6200.

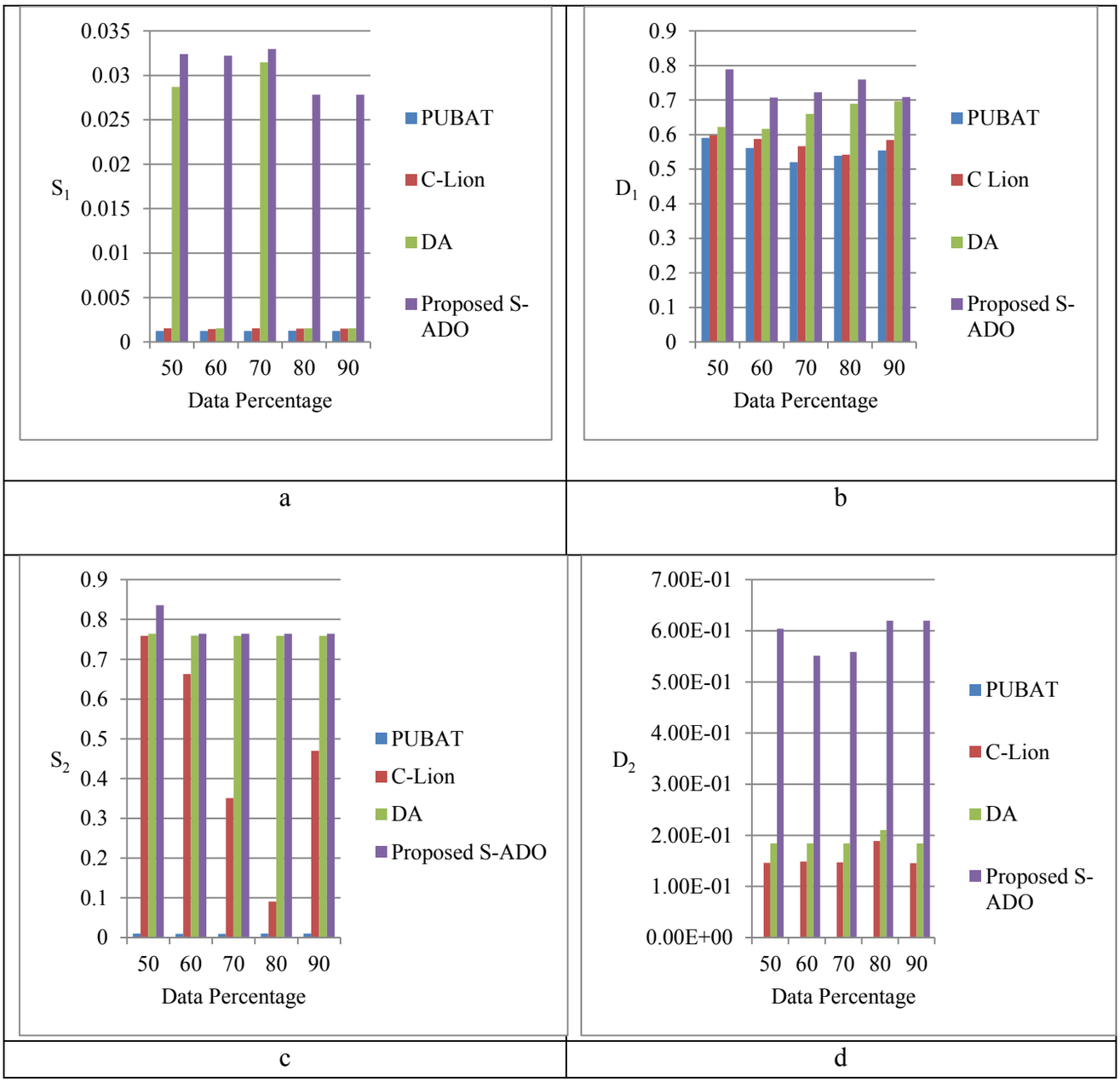

Fig. 4: Analysis Using Dataset 1 in Terms of a) S_1 b) D_1 c) S_2 d) D_2

ii) Analysis Based on Dataset 2

Fig. 5 illustrates the comparative analysis of the existing PUBAT, C-Lion, DA and proposed S-ADO method using dataset 2. The data percentage is varied as 50, 60, 70, 80, and 90 for evaluating the privacy and utility of PUBAT, C-Lion, DA and proposed S-ADO method. Fig. 5a depicts the analysis based on S_1 using dataset 2 for PUBAT, C-Lion, DA and proposed S-ADO method. When the data percentage is 70, the corresponding values of S_1 using PUBAT, C-Lion, DA and proposed S-ADO method are 0.0014, 0.0278, 0.02797, and 0.0279. Similarly, the values of S_1 for data percentage 50 computed by PUBAT, C-Lion, DA and proposed S-ADO method are 0.0011, 0.0060, 0.02797, and 0.0279. The analysis based on D_1 for PUBAT, C-Lion, DA and proposed S-ADO method

using dataset 2 is represented in Fig. 5b. The corresponding values of D_1 measured by PUBAT, C-Lion, DA and proposed S-ADO method are 0.4447, 0.4808, 0.6551, and 0.7883 in case data percentage is 50. Likewise, when the data percentage is 60, the values of D_1 measured by PUBAT, C-Lion, DA and proposed S-ADO method are 0.4590, 0.4985, 0.6124, and 0.7145. The analysis in terms of S_2 using dataset 2 for PUBAT, C-Lion, and DA and proposed S-ADO method is represented in Fig. 5c. For the data percentage 90, the corresponding values of S_2 using PUBAT, C-Lion, DA and proposed S-ADO method are 0.0368, 0.0517, 0.0521, and 0.7855. Similarly, the values of S_2 for data percentage 60 computed by PUBAT, C-Lion, DA and proposed S-ADO method are 0.0111, 0.0362, 0.4123, and 0.4252. The analysis based on D_2 for PUBAT, C-Lion, DA and proposed S-ADO method using dataset 2 is illustrated in Fig. 5d. For the data percentage 80, the corresponding values of D_2 obtained by PUBAT, C-Lion, DA and proposed S-ADO method are 1.00E-04, 1.00E-03, 1.00E-03, and 0.1. Likewise, when the data percentage is 70, the values of D_2 measured by PUBAT, C-Lion, DA and proposed S-ADO method are 1.00E-04, 1.00E-03, 1.00E-03, and 0.1.

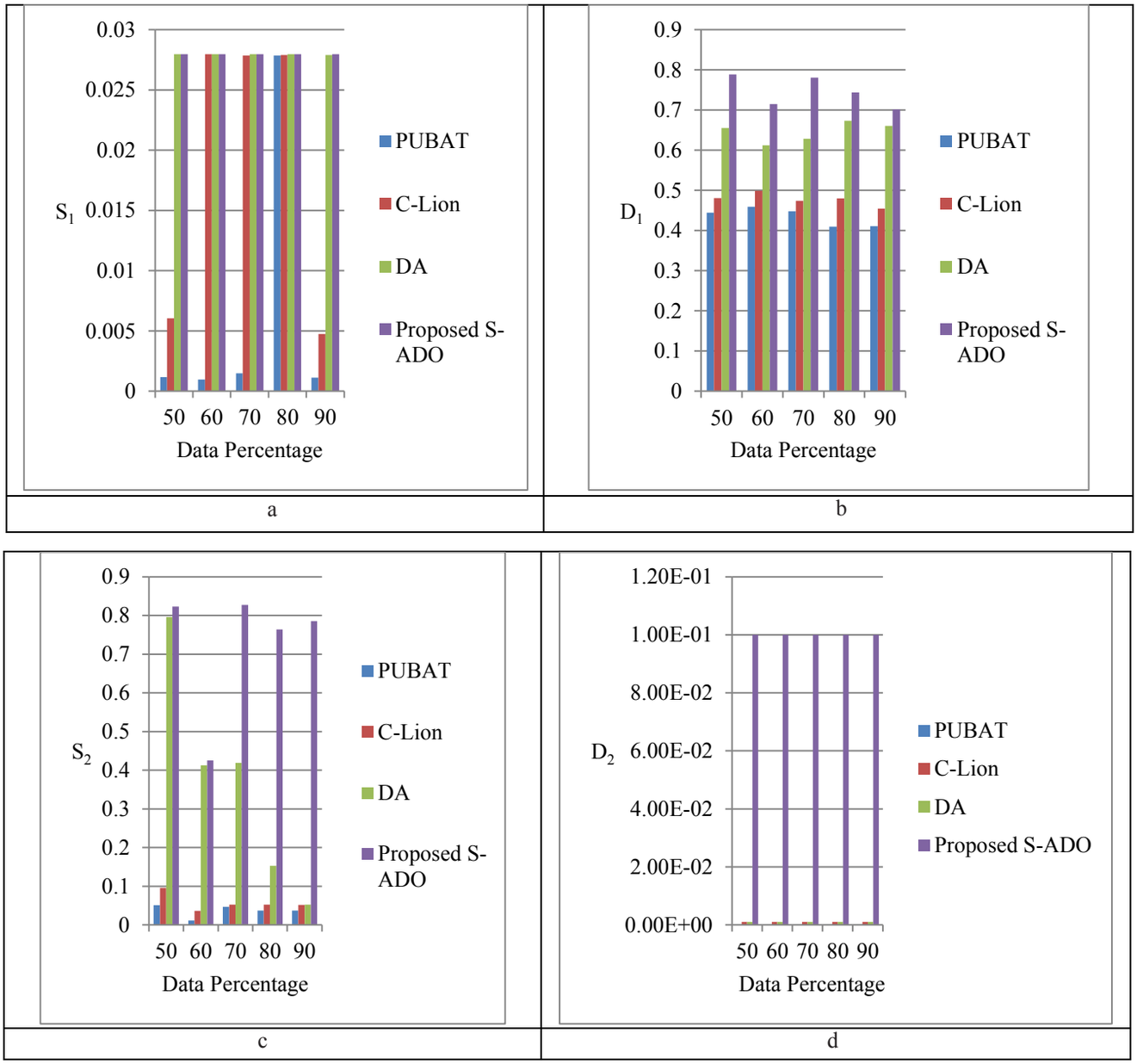

Fig. 5: Analysis Using Dataset 2 in Terms of a) S_1 b) D_1 c) S_2 d) D_2

B. Discussion

For all the different percentages of the data used for the experimentation, the proposed S-ADO method outperformed the existing algorithms in terms of privacy and utility. Table II shows the comparative performance analysis of the existing PUBAT, C-Lion, DA and proposed S-ADO methods for both the datasets i.e. Dataset 1 & Dataset 2, when the percentage of data is 90%.

TABLE II: DISCUSSION TABLE

Datasets	Methods	Privacy Analysis		Utility Analysis	
		S_1	S_2	D_1	D_2
Dataset 1	PUBAT	0.0012	0.0097	0.5545	1.00E-03
	C-Lion	0.0015	0.4695	0.5846	0.1456
	Dragonfly	0.0015	0.75888	0.6960	0.1839
	S-ADO	0.0278	0.7639	0.7088	0.6200
Dataset 2	PUBAT	0.00113	0.0368	0.4108	1.00E-04
	C-Lion	0.00474	0.0517	0.4544	1.00E-03
	Dragonfly	0.02790	0.0521	0.6601	1.00E-03
	S-ADO	0.02797	0.7855	0.7014	0.1

Using Dataset 1, the S_1 attained by proposed S-ADO is the highest with value 0.0278 and the values of S_1 measured by existing PUBAT, C-Lion, and DA are 0.0012, 0.0015, and 0.0015 respectively. Similarly, the highest S_2 is attained by proposed S-ADO with value 0.7639 and the S_2 measured by existing PUBAT, C-Lion, and DA are 0.0097, 0.4695, and 0.75888 respectively. In the analysis using Dataset 1 based on utility, the maximum value of D_1 is obtained by proposed S-ADO with value 0.7088, wherein the value of D_1 attained by existing PUBAT, C-Lion, DA, are 0.5545, 0.5846, 0.6960, respectively. Similarly, the maximum value of D_2 is obtained by proposed S-ADO with value 0.6200 wherein the value of D_2 attained by existing PUBAT, C-Lion, DA, are 1.00E-03, 0.1456, 0.1839 respectively. In the privacy analysis using Dataset 2, the highest S_1 is attained by proposed S-ADO with value 0.02797 and the S_1 measured by PUBAT, C-Lion and DA are 0.00113, 0.00474 and 0.02790 respectively. Similarly, the highest S_2 is attained by proposed S-ADO with value 0.7855 and the S_2 measured by existing PUBAT, C-Lion and DA are 0.0368, 0.0517, and 0.0521 respectively. From utility analysis by utilizing Dataset 2, the maximum value of D_1 is obtained by proposed S-ADO with value 0.7014, wherein the value of D_1 attained by PUBAT, C-Lion, DA, is 0.4108, 0.4544, and 0.6601 respectively. Similarly, the maximum value of D_2 is obtained by proposed S-ADO with value 0.1, wherein the value of D_2 attained by PUBAT, C-Lion and DA, are 1.00E-04, 1.00E-03, and 1.00E-03 respectively.

V. CONCLUSION

The proposed S-ADO technique is developed for retrievable data perturbation for privacy preserved data publishing in MCC. The method is designed by adapting matrix product based model and by incorporating self-adaptive concept in ADO algorithm to generate optimal derivative order coefficients using a newly designed fitness function that uses utility and privacy parameters. The optimal derivative coefficients are used to generate secret key for sanitizing the original data by using fractional theory. In addition, a retrieval phase is designed for retrieving the original data from the protected data by using the secret key. Here, the proposed S-ADO is designed by incorporating Self adaptive constants in the update rule of ADO. The fitness is developed with certain constraint such as privacy and utility which can successfully manage the confidential information in the cloud. The performance of the proposed S-ADO is compared with the existing algorithms, such as PUBAT, C-Lion, and DA with two evaluation metrics, named privacy, and utility. The experimentation is carried out using a bank and dbworld-subjects datasets and the proposed S-ADO shows improved privacy and utility with values 0.7855 and 0.7088, respectively.

REFERENCES

[1] C. Tang, S. Xiao, X. Wei, M. Hao, and W. Chen, "Energy efficient and deadline satisfied task scheduling in mobile cloud computing," in *Proceedings of IEEE International Conference on Big Data and Smart Computing*, pp. 198-205, 2018.

[2] T. Wang, X. Wei, T. Liang, and J. Fan, "Dynamic tasks scheduling based on weighted bi-graph in mobile cloud computing," *Sustainable Computing: Informatics and Systems*, vol. 19, pp. 214-222, 2018.

[3] C. Tang, M. Hao, X. Wei, and W. Chen, "Energy-aware task scheduling in mobile cloud computing," *Distributed and Parallel Databases*, vol. 36, no. 3, pp. 529-553, 2018.

[4] T. Li, Z. Liu, J. Li, C. Jia, and K.-C. Li, "CDPS: A cryptographic data publishing system," *Journal of Computer and System Sciences*, vol. 89, pp. 80-91, 2017.

[5] M. Sharma, A. Chaudhary, M. Mathuria, and S. Chaudhary, "A review study on the privacy preserving data mining techniques and approaches," *International Journal of Computer Science and Telecommunications*, vol. 4, no. 9, pp. 42-46, 2013.

[6] J. J. Panackal, A. S. Pillai, and V. N. Krishnachandran, "Disclosure risk of individuals: A k-anonymity study on health care data related to Indian population," in *Proceedings of International Conference on Data Science & Engineering*, pp. 200-205, August 2014.

[7] S. Ni, M. Xie, and Q. Qian, "Clustering based k-anonymity algorithm for privacy preservation," *International Journal of Network Security*, vol. 19, no. 6, pp. 1062-1071, 2017.

[8] M. M. A. Alphonsa, and P. Amudhavalli, "Genetically modified glowworm swarm optimization based privacy preservation in cloud computing for healthcare sector," *Evolutionary Intelligence*, vol. 11, no. 1-2, pp. 101-116, 2018.

[9] P. Samarati, "Protecting respondents' identities in microdata release," *IEEE Transactions on Knowledge and Data Engineering*, vol. 13, no. 6, pp. 1010-1027, 2001.

[10] L. Sweeney, "K-anonymity: A model for protecting privacy," *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 10, no. 5, pp. 557-570, 2002.

[11] X. Xiao, and Y. Tao, "Anatomy: Simple and effective privacy preservation," in *Proceedings of the 32nd International Conference on Very Large Data Bases*, pp. 139-150, 2006.

[12] R. Nallakumar, N. Sengottaiyan, and M. M. Arif, "Cloud computing and methods for privacy preservation: A survey," *International Journal of Advanced Research in Computer Engineering & Technology (IJARCET)*, vol. 3, no. 11, pp. 3752-3756, 2014.

[13] K. S. S. R. Yarrapragada, and B. B. Krishna, "Impact of tamanu oil-diesel blend on combustion, performance and emissions of diesel engine and its prediction methodology," *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, pp. 1-15, 2015.

[14] A. Sarkar, and T. S. Murugan, "Cluster head selection for energy efficient and delay-less routing in wireless sensor network," *Wireless Networks*, pp. 1-18, 2017.

[15] R. S. Begum, and R. Sugumar, "Novel entropy-based approach for cost-effective privacy preservation of intermediate datasets in cloud," *Cluster Computing*, pp. 1-8, 2017.

[16] T. Paigude, and T. A. Chavan, "A survey on privacy preserving public auditing for data storage security," *International Journal of Computer Trends and Technology*, vol. 4, no. 3, pp. 412-418, 2013.

[17] H. Hammami, H. Brahmi, I. Brahmi, and S. B. Yahia, "Using homomorphic encryption to compute privacy preserving data mining in a cloud computing environment," in *Proceedings of European, Mediterranean, and Middle Eastern Conference on Information Systems*, pp. 397-413, September 2017.

[18] R. H. Jadhav, "Distributed bottom up approach for data anonymization using MapReduce framework on cloud," *International Journal of Advance Scientific Research and Engineering Trends*, vol. 3, no. 6, pp. 109-113, 2018.

[19] Z. Wang, X. Pang, Y. Chen, H. Shao, Q. Wang, L. Wu, H. Chen, and H. Qi, "Privacy-preserving crowd-sourced statistical data publishing with an untrusted server," *IEEE Transactions on Mobile Computing*, 2018.

[20] F. Yu, M. Chen, B. Yu, W. Li, L. Ma, and H. Gao, "Privacy preservation based on clustering perturbation algorithm for social network," *Multimedia Tools and Applications*, vol. 77, no. 9, pp. 11241-11258, 2018.

[21] K. C. Sreedhar, M. N. Faruk, and B. Venkateswarlu, "A genetic TDS and BUG with pseudo-identifier for privacy preservation over incremental data sets," *Journal of Intelligent & Fuzzy Systems*, vol. 32, no. 4, pp. 2863-2873, 2017.

[22] T. Kalidoss, G. Sannasi, S. Lakshmanan, K. Kanagasabai, and A. Kannan, "Data anonymisation of vertically partitioned data using Map Reduce techniques on cloud," *International Journal of Communication Networks and Distributed Systems*, vol. 20, no. 4, pp. 519-531, 2018.

[23] J. Li, J. Wei, W. Liu, and X. Hu, "PMDP: A framework for preserving multiparty data privacy in cloud computing," *Security and Communication Networks*, 2017.

[24] X. Zhang, W. Dou, J. Pei, S. Nepal, C. Yang, C. Liu, and J. Chen, "Proximity-aware local-recoding anonymization with MapReduce for scalable big data privacy preservation in cloud," *IEEE Transactions on Computers*, vol. 64, no. 8, pp. 2293-2307, September 2014.

[25] S. Mirjalili, "Dragonfly algorithm: A new meta-heuristic optimization technique for solving single-objective, discrete, and multi-objective problems," *Neural Computing and Applications*, vol. 27, no. 4, pp. 1053-1073, May 2016.

[26] R. F. Engle, and S. Manganelli, "CAViaR: Conditional value at risk by quantile regression," *Journal of Business & Economic Statistics*, American Statistical Association, vol. 22, pp. 367-381, October 1999.

[27] N. P. Karlekar, and N. Gomathi, "Kronecker product and bat algorithm-based coefficient generation for privacy protection on cloud," *International Journal of Modeling, Simulation, and Scientific Computing*, vol. 8, no. 3, 1750021, 2017.

[28] Bank Marketing Data Set. Available: https://archive.ics.uci.edu/ml/datasets/Bank+Marketing

[29] DBworld e-mails data set. Available: https://www.openml.org/d/1564

[30] E. J. S. Pires, J. A. T. Machado, P. B. de M. Oliveira, J. B. Cunha, and L. Mendes, "Particle swarm optimization with fractional-order velocity," *Nonlinear Dynamics*, vol. 61, no. 1-2, pp. 295-301, July 2010.

[31] A. George, and A. Sumathi, "Dyadic product and crow lion algorithm based coefficient generation for privacy protection on cloud," *Cluster Computing*, pp. 1-12, 2018.