A New Reversible Database Watermarking Approach with Ant Colony Optimization Algorithm

Junyao Lian
School of Software, Zhengzhou University, Zhengzhou 450003, China
joylian@stu.zzu.edu.cn

Abstract. In many fields such as medicine, science, transportation, it is often necessary to maintain the real-time information, and the data in the centralized database that saves data in these fields is distributed to various clients. On the insecure public Internet, there are distributed databases stored in data centers. Sharing may create risks, such as unauthorized copies, changes to data, and distribution of data to unauthorized persons for reuse. Many database watermarking methods have been proposed to protect digital rights. In order to reduce watermark distortion and increase watermark capacity, people pay more and more attention to the optimization of watermark technology. This paper proposes a method of embedding differential expansion watermarking (DEW) based on ant colony algorithm (ACO) in relational database. This method uses a secure hash algorithm for grouping, increasing the privacy of watermark embedding. The ant colony algorithm performs the optimization operation, finds the best attribute columns for embedding the watermark in the subgroup, and finally uses the differential expansion technology to embed the watermark. Experimental results show that compared with similar works reported in the literature, the reversible watermarking method based on ant colony algorithm database proposed in this paper not only increases the watermarking capacity, but also improves the robustness against various types of attacks.

1. Introduction

In order to ensure the copyright of shared data and prevent tampering, it is necessary to use appropriate protection methods in shared data. Because the encrypted information is not related to the original content and cannot be used for copyright protection, the digital watermarking method is proposed[1]. In 2008, Gupta and Pieprzyk used differential expansion watermarking (DEW) technology to realize the reversibility of relational databases, and use reversible watermarks to carry out watermark attacks[2]. In 2012, Arif et al. proposed to use heavy watermark attack and use date to add watermark to overcome the linear conversion attack that cannot be detected by watermark when deleting tuples[3]. Jawad and Khan used the reversible method of the genetic algorithm based on the differential expansion watermark to reduce the distortion caused by the embedding process and increase the watermark capacity[4]. The DEW technology was used to embed the watermark bit into the database in a reversible manner. In 2017, Bilehan Imamoglu and others proposed a new watermarking method, using database watermarking technology based on Firefly algorithm to reduce distortion and algorithm complexity, and improve the robustness of DEW[5].

This paper proposes a new robust and reversible watermarking technique for protecting relational databases[6]. Based on the idea of differential expansion, this method uses ant colony algorithm to increase the watermark capacity and reduce distortion. Because of the reversibility of this method, the distortion caused by inserting the watermark can be completely recovered[7]. In order to prevent the
distortion caused by the difference expansion and help the attacker to predict the watermarking attribute, we combined the tuple and the distortion classified by the attribute in the ant colony algorithm's fitness function, making it difficult for the attacker to predict the watermarked attribute. Experimental results show that the technology proposed in this paper significantly reduces the problems of false positives and attribute order changes at the detection end, and can resist various attacks, such as deletion, sorting, and bit flipping.

2. Construction and Geometrical Dimensions of Specimens
The watermarking algorithm is composed of four modules, namely preprocessing module, ant colony measuring module, watermark embedding module and watermark extracting module. The preprocessing module sorts the tuples to obtain the prepared data sequence; the ant colony determination module outputs the best ants for the data set; the watermark embedding module adds the data watermark to the data set according to the best ants; the watermark extraction module extracts the data from Database watermark information and reconstruct the original database using DEW.

2.1. Preprocessing
The maximum and minimum values of each attribute in the database are selected as the upper and lower limits of the non-distortion range. That is, assuming that the database has \( M \) attribute columns \( \{A_1, A_2, ..., A_M\} \), the undistorted range of the attribute column \( x \) is \( (A_x^{\min}, A_x^{\max}) \). Then use the secure hash algorithm to group the original database elements into several subgroups.

2.2. Ant colony determination
The algorithm first generates a random attribute for each attribute pair and fixes it, and then uses the “dewbest” function to find another attribute that minimizes the difference. It is used to generate the function used by the initial ant colony. It aims to minimize the difference after embedding the watermark. The “dewbest” function produces attribute pairs that minimize differences for each row.

When determining the best ant, first calculate the objective function value of each ant generated in the previous stage, and record the smallest objective function value, namely, the information about the best ant. In the subsequent algorithm steps, the best ants are used to strengthen the pheromone matrix.

The objective function consists of two parts whose weights are \( w_1 \) and \( w_2 \), respectively. \( a \) is the sigmoid function adjustment coefficient, \( c \) refers to the number of failed cells, the first part is the sum of all data value changes before and after embedding the watermark, and the second part shows the effect of the number of failed cells on the objective function. In this paper, the sigmoid function is used as the influence function to approximate the change law of the influence of the number of failed cells. \( FD_k^{\alpha} \) refers to the data value in the \( k \) row \( x_i \) column in the original database, and \( BD_k^{\alpha} \) refers to the data value in the \( k \) row \( x_i \) column in the original database after the watermarking operation.

\[
\min Z = w_1 \cdot \sum_{k=1}^{R} \left( |BD_k^{\alpha} - FD_k^{\alpha}| + |BD_k^{\alpha} - FD_k^{\alpha}| \right) + w_2 \cdot af(c)
\]  \hspace{1cm} (1)

\[
P = \begin{bmatrix}
p_{11} & \cdots & p_{1c} \\
\vdots & \ddots & \vdots \\
p_{r1} & \cdots & p_{rc}
\end{bmatrix}
\]  \hspace{1cm} (2)
The design of the pheromone matrix that affects ant selection is as above Eq.(2). $P_{ij}$ represents the pheromone concentration of the j-th attribute column selected in the i-th row of the subset. The initial pheromone matrix is an all-one matrix. The pheromone update operation is divided into the normal volatilization of pheromones and the use of the best ants to strengthen pheromones. The updating algorithm of matrix and pheromone is designed as follows.

Function: PheromoneUpdate
Input: Best ant $Y$
Output: The updated pheromone matrix $P$
Step 1. Repeat Step 2 for $r \in [1, R]$.
Step 2. Repeat Step 3 for $c \in [1, C]$.
Step 3. Assign $p_{ij} \times (1 - \rho)$ to $p_{ij}$.
Step 4. Repeat Step 4 for $i \in [1, R]$.
Step 5. Repeat Step 5 for $j \in (x'_i, x''_i)$.
Step 6. Assign $p_{ij} + Q/\text{goalBestAnt}$ to $p_{ij}$.
Step 7. Return $P$.

The above algorithm uses a heuristic enhancement strategy, where $Q$ is the heuristic index and $\rho$ is the attenuation factor. The enhanced pheromone matrix can generate a roulette probability selection matrix to help the ants make the next choice.

This article uses double roulette to update the selection of ants, namely, each ant's two choices in each row are through roulette. For each row, the greater the pheromone concentration corresponding to the attribute is, the greater the probability that the attribute is selected.

Within the maximum number of iterations, repeat steps to obtain the optimal decision vector, that is, the optimal position of the current subgroup to be embedded in the watermark. Through the ant colony algorithm, the optimal embedded watermark attribute row position can be found under the condition of the maximum watermark capacity under the constraint of ensuring that the data is in the non-distortion range and the minimum change value.

2.3. Watermark embedding
After the subgroup obtained by the grouping method and the optimal embedding watermark position obtained by the ant colony algorithm, the differential watermarking technique is used to embed the generated watermark sequence. First, the original database is grouped into some subgroups by using a key safe hash algorithm, in the selection process of the attribute feature row, the ant colony algorithm is used to select the attribute row to be embedded in the watermark for all subgroups, and then the DEW technique is used for the subgroup to be embedded in the attribute feature row according to the watermark position to realize the embedding of the watermark sequence. The value of is generated by the random function.

2.4. Watermark extraction
First, group the databases embedded in the watermark, and use the DEW operation on the tuples in the subgroup according to the saved attribute feature sequence to extract the watermark bits embedded in the subgroup. Then vote for the extracted watermark, count the number of 0 and 1 in the subgroup, take the largest number of watermark bits as the watermark bits extracted by the subgroup, and perform watermark bit extraction operations on all subgroups in turn, and finally embed watermark sequence and original database.

3. Test Results and Discussions

3.1. Description of experimental environment
The simulation results of this method are given in this article. This method is coded in Python using PyCharm IDE. The database comes from the Forest Cover Type Dataset (FCT) provided by the University of California. The database contains 581,012 tuples and 54 attribute columns. In this experiment, the first 25,000 rows of tuples and the first 8 columns of attribute data of the database are selected as experimental data, and the watermark method based on the firefly algorithm is marked as FFADEW[5]. Marked as ACODEW, the algorithm performance of the two methods is analyzed.

3.2. Watermark capacity experiment

The first experiment is to compare the watermark capacity of the two methods. The watermark capacity is the ratio of the number of tuples available in the DS to the total number of tuples. A higher watermark capacity means that more watermark information can be embedded in the same DS. This article uses 4052, 8997, 15466, and 20031 to create six different DS to test the watermark capacity of the proposed method and other methods. Fig.1. shows the watermark capacity performance of the two methods. The watermark technology proposed in this paper can enable the database to embed more watermark information. At the same time, as the number of database tuples increases, the method proposed in this paper has a potential upward trend.

![Fig.1 Database watermark capacity under different tuples](image)

3.3. Robust experiment

The second experiment is a robust experiment, using watermark extraction accuracy as the criterion

\[
ACC = (1 - \frac{\text{sum}(w \oplus w')}{{\text{len}(w)}) \times 100\%
\]

Where \( w \) represents the watermark sequence before extraction, \( w' \) represents the watermark sequence obtained by extraction, and \( \oplus \) represents the bitwise XOR operation, and \( \text{len} \) obtains the number of bits of the watermark sequence, and \( \text{sum} \) sums the bits of the watermark sequence. If the ACC value is higher, it indicates that the watermarking method is more robust. In this paper, two attack experiments of subset deletion and modification are used. Subset deletion and modification experiments are conducted at 0, 10%, 20%, 30%, 40%, 50%, 60%, and 70%, respectively.
Fig. 2 Database deletion attack                Fig. 3 Database modification attack

Fig.2. and Fig.3. show that whether it is a subset deletion or modification experiment, as the attack intensity continues to increase, the proposed method ACODEW watermark extraction rate declines slower than FFADEW. Therefore, this article ACODEW is better than FFADEW.

4. Conclusion

Based on the results and discussions presented above, the conclusions are obtained as below:

(1) The method in this paper innovatively combines the ant colony algorithm and the differential expansion algorithm, and applies it to the relational database watermark to achieve the protection of the database ownership.

(2) This method first uses a secure hash algorithm for grouping. The technology of group embedding makes the watermark embedding more secretive.

(3) The ant colony algorithm is used to ensure that the data distortion of the database after watermark embedding is minimized.

(4) Experimental results show that the database reversible watermarking method based on ant colony algorithm proposed in this paper not only increases the watermark capacity, but also improves the robustness against various types of attacks.

Acknowledgments

This work was supported by the Innovative Training Project of the school of software of Zhengzhou University.

References

[1] I. Cox, “Digital watermarking,” Journal of Electronic Imaging, vol. 11, no. 3, 2002.
[2] G. Gupta and J. Pieprzyk, “Reversible and blind database watermarking using difference expansion,” in Proceedings of the 1st International Conference on Forensic Applications and Techniques in Telecommunications, Information, and Multimedia and Workshop, Adelaide, Australia, January 2008.
[3] F. Arif, S. Suhail, and M. Kamran, “Watermarking of relational databases with emphasis on re-watermarking attack,” International Journal of Computer Science Issues, vol. 9, no. 3, pp. 521–526, 2012.
[4] K. Jawad and A. Khan, “Genetic algorithm and difference expansion based reversible watermarking for relational databases,” Journal of Systems and Software, vol. 86, no. 11, pp. 2742–2753, 2013.
[5] M. B. Imamoglu, M. Ulutas, G. Ulutas, et al. A New Reversible Database Watermarking Approach with Firefly Optimization Algorithm. 2017, 2017.
[6] Y. Li, H. Guo, and S. Jajodia, “Tamper detection and localization for categorical data using fragile watermarks,” in Proceedings of the 4th ACM Workshop on Digital Rights Management (DRM ’04), pp. 73–82, October 2004.
[7] H. Guo, Y. Li, A. Liu, and S. Jajodia, “A fragile watermarking scheme for detecting malicious
modifications of database relations,” Information Sciences, vol. 176, no. 10, pp. 1350–1378, 2006.

[8] Y. Zhang, B. Yang, and X.-M. Niu, “Reversible watermarking for relational database authentication,” Journal of Computers, vol. 17, no. 2, pp. 59–66, 2006.

[9] X. Zhou, M. Huang, and Z. Peng, “An additive-attack-proof watermarking mechanism for databases’ copyrights protection using image,” in Proceedings of the ACM Symposium on Applied Computing (SAC ’07), pp. 254–258, Seoul, Korea, March 2007.

[10] A. M. Alattar, “Reversible watermark using the difference expansion of a generalized integer transform,” IEEE Transactions on Image Processing, vol. 13, no. 8, pp. 1147–1156, 2004.

[11] S. Bhattacharya and A. Cortesi, “A distortion free watermark framework for relational databases,” in Proceedings of the 4th International Conference on Software and Data Technologies (ICSOFT ’09), pp. 229–234, Springer, Sofia, Bulgaria, July 2009.

[12] D. Hanyurwimfura, Y. Liu, and Z. Liu, “Text format based relational database watermarking for non-numeric data,” in Proceedings of the International Conference on Computer Design and Applications (ICCCDA ’10), pp. V4-312–V4-316, IEEE, Qinhuangdao, China, June 2010.

[13] M. E. Farfoua, S.-J. Horng, J.-L. Lai, R.-S. Run, R.-J. Chen, and M. K. Khan, “A blind reversible method for watermarking relational databases based on a time-stamping protocol,” Expert Systems with Applications, vol. 39, no. 3, pp. 3185–3196, 2012.

[14] D. M. Thodi and J. J. Rodriguez, “Prediction-error based reversible watermarking,” in Proceedings of the International Conference on Image Processing (ICIP ’04), Singapore, October 2004.

[15] A. Khan and S. A. Husain, “A fragile zero watermarking scheme to detect and characterize malicious modifications in database relations,” The Scientific World Journal, vol. 2013, Article ID 796726, 16 pages, 2015.

[16] L. Camara, J. Li, R. Li, and W. Xie, “Distortion-free watermarking approach for relational database integrity checking,” Mathematical Problems in Engineering, vol. 2014, Article ID 697165, 10 pages, 2014.

[17] X.-S. Yang, Nature-Inspired Metaheuristic Algorithms, Luniver Press, 2008.

[18] A. Colorni, M. Dorigo, and V. Maniezzo, “Distributed optimization by ant colonies,” in Proceedings of the 1st European Conference on Artificial Life, Paris, France, December 1991.

[19] D. Karaboga, “An idea based on honey bee swarm for numerical optimization,” Tech. Rep. TR06, Erciyes University, Engineering Faculty, Computer Engineering Department, 2005.