Computer Science and Application 计算机科学与应用, 2020, 10(7), 1400-1408
Published Online July 2020 in Hans. http://www.hanspub.org/journal/csa
https://doi.org/10.12677/csa.2020.107144

Hans汉斯

A New Practical Database Watermarking Method

Hongbin Zhu[1], Heng Wang[1], Tao Zhao[1], Pengfei Yu[2,3], Ruxia Yang[2,3*]

[1]Big Data Center of State Grid Corporation of China, Beijing
[2]Global Energy Interconnection Research Institute Co., Ltd., Nanjing Jiangsu
[3]State Grid Key Laboratory of Information & Network Security, Nanjing Jiangsu
Email: *1270034320@qq.com

Received: Jul. 9th, 2020; accepted: Jul. 23rd, 2020; published: Jul. 30th, 2020

Abstract

Reversible database watermarking is an important method to protect database copyright. However, most existing database watermarking methods will modify the original data to a great extent, resulting in the decline of data quality; some methods increase the storage space due to the addition of auxiliary data, and some methods with poor practicability can only be used for numerical database. In this paper, we propose a robust and reversible database watermarking technique, which uses genetic simulated annealing algorithm and histogram shift watermarking for numerical relational data, and uses synonym substitution and arithmetic coding to add watermarking to text data. Genetic simulated annealing algorithm was used to select the best secret key for the grouping database, and the histogram of the prediction error was shifted to embed the watermark better. Synonym substitution and arithmetic coding can effectively add watermark to text information. Analysis shows that our method can not only enhance database security, but also reduce data quality loss.

Keywords

Database Watermark, Genetic Simulated Annealing Algorithm, Histogram Shift, Synonym Substitution

一种实用的新型数据库水印方法

朱洪斌[1]，王　衡[1]，赵　涛[1]，于鹏飞[2,3]，杨如侠[2,3*]

[1]国家电网有限公司大数据中心，北京
[2]全球能源互联网研究院有限公司，江苏 南京
[3]信息网络安全国网重点实验室，江苏 南京
Email: *1270034320@qq.com

*通讯作者。

文章引用: 朱洪斌, 王衡, 赵涛, 于鹏飞, 杨如侠. 一种实用的新型数据库水印方法[J]. 计算机科学与应用, 2020, 10(7): 1400-1408. DOI: 10.12677/csa.2020.107144

收稿日期：2020年7月9日；录用日期：2020年7月23日；发布日期：2020年7月30日

摘　要

可逆的数据库水印方法是保护数据库版权的重要手段。然而，大多数现有的数据库水印方法会在很大程度上修改原始数据，导致数据质量下降；部分方法增加辅助数据导致存储空间增大，或是只能用于数值型数据库，实用性差。在本文中，我们提出了一种健壮且可逆的数据库水印技术，使用遗传模拟退火算法和直方图移位水印，用于数值关系数据，同时采用同义词替换和算术编码对文本数据进行水印添加。利用遗传模拟退火算法为分组数据库选择最佳密钥，将预测误差的直方图进行移位，以更好地嵌入水印。同义词替换和算数编码则可以有效地在文本信息中加入水印。分析显示，我们的方法不但可以增强数据库安全，并可以减少数据质量损失。

关键词

数据库水印，遗传模拟退火算法，直方图移位，同义词替换

1. 引言

当前，关系数据库得到了广泛的使用[1]。但是，数据库的创建、传输和共享显著增加的同时带来了如数据盗窃、非法复制和侵犯版权的安全风险。近年来，即使在数据敏感的领域，如医疗保健领域，也经常发生数据库泄漏事故的报道[2]。从历史上看，水印技术已被用于确保各种数据格式(例如，图像，视频和音频)的所有权保护和防篡改。数据库加水印是一种相对较新的技术，将消息嵌入通常独立且离散的数据库记录中。数据库水印由 Agrawal 和 Kiernan 于 2002 年首次引入[3]。此后，国内外学者提出了几种方法[4] [5] [6] [7]。

在[4]中，提出了一种鲁棒的持久性水印方法，该方法同时嵌入了私人和公共水印。根据水印是否对数据库数据进行了任何更改，水印技术可分为两类：基于失真的水印和无失真水印[8]。可逆水印可以看作是水印的一种特例，它不仅可以成功地提取嵌入的水印，而且可以完全还原出原始载体。当前，它在军事通信，医疗保健，政府安全通信和执法方面越来越受到关注。

关系数据库的第一个可逆水印方案是在 2006 年提出的[9]，其中直方图扩展用于可逆数据库水印。但是，这种方法对于强大的攻击并不可靠。2008 年，基于差异扩展的水印(DEW)被提出。文献[10]的工作能够以可逆的方式给数据库加水印。Bhattacharya 和 Cortesi 在 2009 年提出了一种方法[11]，将元组划分为置换后嵌入水印。但是，考虑失真约束时，DEW 的嵌入率会大大降低。在[12]中，遗传算法(GA)被用来设计鲁棒的密钥方法。然后，一种基于差分扩展水印(GADEW)技术的遗传算法被提出，作为一种健壮且可逆的数据库水印解决方案[13]。[14]中提出了一种新的可逆数据库水印方法，该方法将 DEW 与萤火虫算法(FFA)相结合，FFA 选择最佳属性值以产生较低的失真并增加水印容量。尽管 FFADEW 最小化了数据失真并增加了水印容量，但是信息失真仍然很严重。2019 年[15]提出了遗传算

法和直方图移位算法 GAHSW，但是其只适用于数值型数据库，实用性较差。同时其使用生成主键，前期计算量非常大。

现有的文本可逆水印方法主要存在两个问题：水印嵌入容量低、附加信息共享量大。针对这两个问题，[16]从无损数据压缩的角度提出了一种基于自适应二进制算术编码的可逆自然语言水印算法。所提出的方法首先在每个同义词集中仅对具有最大词频的两个词进行编码，从而实现出现在封面文本中的同义词的二进制。由于封面文字中频率较高的同义词的数量远远多于频率较低的同义词[17]，量化封面文字中的同义词产生的二进制序列严重不一致，因此存在很大的冗余空间，可以提供压缩的可能性。从这一观点出发，选择自适应二进制算术编码来无损地压缩二进制同义词序列。然后，将压缩的二进制化同义词序列附加水印和一些其他信息通过同义词替换嵌入到封面文本中。通过对水印文本中的同义词值进行解码来提取水印，同时可以通过使用算术编码对提取的压缩数据进行解压缩来恢复原始同义词。

我们通过对现有的数值型水印方法进行优化，同时结合文本水印方法，构造了一种新的数据库水印方法 GHSA。通过实验表明，在数值与文本并存的实际数据库上，该方法的鲁棒性更强，安全性更高，并且该方法适用范围更全面，更适合实际生产生活应用。

本文第 2 节阐述了我们方法所用的相关知识，第 3 节具体阐述了我们使用的数据库水印方案流程，第 4 节提供对比实验结果进行验证，最后我们在第 5 节得出结论。

2. 相关知识

2.1. 遗传模拟退火算法 GASA

遗传模拟退火算法 GASA [18]是将遗传算法 GA 与模拟退火算法 SA 相结合构成的一种优化算法。遗传算法的局部搜索能力较差，但把握搜索过程总体能力较强；而模拟退火算法具有较强的局部搜索能力，并能使搜索过程避免陷入局部最优解。但是模拟退火算法对整个搜索空间状况了解不多，不便于使搜索过程进入最有希望的搜索区域，效率不高。将遗传算法和模拟退火算法相结合，互相取长补短，可以成为性能优良的全局搜索算法。其特点有：

1) GASA 是标准 GA，SA 和并行 SA 的统一结构。

2) GASA 是一个两层并行搜索结构，进程层次上在各温度下串行依次进行 GA 和 SA 搜索，其中 SA 的初解来自 GA 的进化结果，SA 抽样后的解又成为 GA 进一步进化的初始种群。

3) GASA 利用了不同的邻域搜索结构，混合算法过程中包含了 GA 的复制，交叉，变异和 SA 的状态产生函数等不同的搜索结构，增强了算法在解空间的探索能力和效率。

4) GASA 搜索行为可控，混合策略的搜索行为可以通过退温历程加以控制。

5) 混合策略利用了双重准则，抽样稳定准则可以判定各温度下算法的搜索行为，算法终止准则可以判定优化性能的变化趋势和最终优化性能。

由此可见，GASA 结合了两者的特点，使各自的搜索能力得到补充。GASA 算法具体步骤将在第 3 节详细展开。

2.2. 直方图移位 HSW

直方图移位于 2006 年首次用于数据库水印[9]。[15]改进了直方图移位方法，使得该方法能更好解决水印嵌入失真问题。在 HS 中，使用预测变量来创建要素元素以进行扩展嵌入。

假设我们有一个属性值 y 和嵌入一个比特位值 ω。首先我们使用公式(2.1)计算预测变量：

$$\hat{y}=\frac{\min[j]+\max[j]}{2} \tag{2.1}$$

其中 j 为属性列的标注，然后，使用公式(2.2)计算预测误差以构造直方图：

$$p_e=y-\hat{y} \tag{2.2}$$

在 HS 嵌入之前，根据直方图中的预测误差的绝对值确定具有非零频率的峰值。我们表示峰值为p，使用数组 pa 存储所有 p，并将其用于水印提取和数据恢复。我们使用公式(2.3)计算新的相关预测误差 p'_e：

$$p'_e=\begin{cases}p_e+1, & p_e\geq p+1\\ p_e-1, & p_e\leq -(p+1)\\ p_e, & p_e=p, \omega=0\\ p_e+1, & p_e=p, \omega=1\\ p_e, & p_e=-p, \omega=0\\ p_e-1, & ?p_e=-p, \omega=1\end{cases} \tag{2.3}$$

然后，我们使用公式(2.4)计算新的属性值 y'，完成水印嵌入：

$$y'=p'_e+\hat{y} \tag{2.4}$$

通过峰值数组 pa 以及其他辅助信息，我们可以将水印值 y' 还原为原始属性 y，并取得水印嵌入位 ω。还原操作即为嵌入操作的逆操作，将在第 3 节给出。

2.3. 同义词替换和算术编码 SSAC

我们使用的文本水印方法为[16]提出的同义词替换和算术编码 SSAC。其采用的同义词量化规则定义为：如果同义词集中的同义词是频率最高的同义词 MFS，它将被编码为“0”；如果是频率第二高的同义词 SMFS，则将其编码为“1”；否则，将其视为没有编码值的普通单词。由 MFS 和 SMFS 的编码值组成的二进制序列应具有可压缩的空间，以有效地压缩为较短的二进制序列。通过自适应二进制算术编码，可以对封面文本中同义词的编码值进行压缩，然后可以无损地对其进行解压缩以恢复原始同义词。如果将压缩的二进制序列与水印信息一起嵌入到封面文本中，则可以从水印文本中可逆地恢复封面文本。由于每个同义词至少可以嵌入到一个比特信息中，因此当添加的二进制同义词序列的压缩二进制水印信息比封面文本中的原始二进制同义词序列短时，它们可以通过同义词替换成功嵌入。我们将在第 3 节给出文本水印添加的具体过程。

3. 数据库水印方案 GHSA

本节分为三个部分，3.1 节介绍遗传模拟退火算法 GASA 的流程，3.2 节介绍对数值型数据的整体算法操作，3.3 节介绍对文本数据的同义词替换和算术编码 SSAC 水印增加和提取算法操作。

3.1. 遗传模拟退火算法 GASA

与遗传算法的总体运行相类似，遗传模拟退火算法也是从一组随机产生的初始解(初始群体)开始全局最优解的搜索过程，他首先通过选择，交叉，变异等遗传操作产生一组新的个体，然后再独立地对所产生的各个结果个体进行模拟退火，将其结果作为下一代群体中的个体。过程反复迭代进行，直至满足条件为止。下面给出模拟退火算法的流程描述：

遗传模拟退火算法

输入：种群规模，最大迭代次数
输出：私钥Ks
1: $t := 0$
2: 生成初始种群$P(t)$
3: 评估$P(t)$适应度
4: 交叉：$P'(t) := Crossover[P(t)]$
5: 变异：$P''(t) := Mutation[P'(t)]$
6: 模拟退火：$P'''(t) := SimulatedAnnealing[P''(t)]$
7: 评估$P'''(t)$适应度
8: 生成下一代种群：$P(t+1) := Reproduction[P(t) \cup P'''(t)]$
9: if $P(t+1)$满足条件
10: 输出在$P(t+1)$中最优的私钥K_S
11: else 跳转到步骤4再次循环

3.2. 数值型数据整体操作

首先我们应用 GASA 算法获取最适合数据库分组的密钥 K_S，这属于我们的前期工作。然后，我们选择数据库中部分文本型数据属性列(必须非空)，根据这些文本型数据生成主键 PK。具体操作可以选择很多种方法，这里我们选择将文本数据转换为十六进制并将其重复循环，直至生成统一长度的主键值，长度可以由用户自己确定，但是要确保主键的长度满足互不重复的保障条件。在我们进行数据库水印提取的时候我们只要确定是哪几个属性列以及生成方法就可以确定主键，在传输过程中主键不会进行传输，节省了存储空间，同时，避免因主键被恶意删除使得水印无法恢复，从而无法确认的问题。

然后，应用该密钥，使用公式 3.1 对数据库的数据进行分组：

$$n_u = H\left(Ks \middle| H\left(Ks \middle| t_u.PK\right)\right) \bmod Ng \tag{3.1}$$

最后采用第 2 节介绍的 HSW 直方图移位法对数值型数据加水印。其具体算法如下所示：

输入：数据库D, 水印信息w, 数值z
输出：带水印数据库D_w, 信息存储数组pa, mp
1: 使用Eq. (3.1)对数据库数据进行分组
2: for $\omega = 1$ to l do
3: //循环迭代所有的水印位
4: for $i = 1$ to N/N_g do
5: //循环迭代所有元组
6: $j = H(Ks|t_u.PK)\%z$;
7: //确定水印嵌入的属性列
8: if $A_{ij} = \max[j]$ or $A_{ij} = \min[j]$ then
9: $j = (j+1)\%z$;
10: 将A_{ij}的主键$t_u.PK$写入mp;
11: end if
12: $\hat{y} = \left\lfloor \frac{min[j] + max[j]}{2} \right\rfloor$

13: 通过公式 $Eq.(2.2)$ 计算 p_e;
14: end for
15: 通过 p_e 确定 p;
16: 将 p 写入 pa;
17: for $i = 1$ to N/N_g do
18: 通过公式 $Eq.(2.3)$ 计算 p'_e
19: 通过公式 $Eq.(2.4)$ 计算新的属性值 y'
20: end for
21: end for
22: 返回带水印数据库 D_w, 信息存储数组 pa, mp

其中，D 代表数据库，z 为用户输入的数值，其小于用户数据库数值型数据的数量。

数据库水印的提取和数据恢复是上述过程的逆过程。我们的方法在添加水印过程中并未对最大值和最小值进行变换，所以水印增加前后，$\hat{y}$ 的大小是不变的，在增加了水印的数据库中存放的值为 y'，由此，根据公式(3.2)，我们可以计算得到 p'_e：

$$p'_e = y' - \hat{y} \tag{3.2}$$

将 p'_e 与 pa 数组中存储的 p 相比较，我们可以通过公式(3.3)来恢复原始数据：

$$y^r = \begin{cases} y'-1, & p'_e \geq p+1 \\ y', & p'_e = p \\ y'+1, & p'_e \leq -(p+1) \end{cases} \tag{3.3}$$

由公式(2.3)可知，如果 $p'_e = p$，有水印位 ω 为 0，如果 $p'_e = p+1$ 或是 $p'_e = -(p+1)$，则水印位 ω 为 1。至此，水印数据也被提取出来，原始数据库也恢复完成了。

水印提取数据库恢复的具体算法步骤在此不再给出，如果有兴趣可以参考[文章 1]的解密步骤，我们的方法与其方法在数值型水印提取过程大致相同。

3.3. 文本数据水印的增加和提取

信息嵌入过程完成后生成水印文本，并将压缩后的同义词序列与水印信息一起嵌入。SSAC 文本水印的增加和提取流程如下。

1) 同义词识别。遍历文本并检索准备好的同义词数据库，如果单词是 MFS 或 SMFS，则将其识别为同义词。

2) 二进制量化。将识别出的同义词量化为二进制序列。如果存在同义词，则获得 n 位同义词序列 Q。

3) 压缩。通过采用自适应二进制算术编码将 Q 压缩为较短的 m 位二进制序列 Q'。由此可以获得一些冗余空间来容纳附加水印信息。

4) 水印信息二进制化。水印信息也根据其字符的 ASCII 值转换为二进制序列。

5) 附加信息估计。为了将压缩的同义词序列 Q'与水印信息 M 区分开来，当将它们串联在一起嵌入到文本中时，应将其长度记录并发送给接收者。

6) 嵌入式信息生成。附加信息与 Q'和水印信息 M 串联，形成一个完整的嵌入式信息 S。

7) 比较。将嵌入信息 S 与原始同义词序列 Q 相比较，找到某些对应位置的不匹配值。

8) 同义词替换。对于不匹配的位置，原始同义词将被其同义词替换，该同义词的编码值等于嵌入的信息位。匹配位置不会进行替换。当所有嵌入的信息位都嵌入后，带水印的文本生成完成。

带水印的文本将被发送到接收方，接收方可以提取水印信息并恢复原始封面文本而不会产生任何失真。

由于同义词数据库共享，所以接收方可以很容易完成同义词的统计并进行二进制量化。在得到同义词序列Q'与水印信息M的长度后，接收方可以轻易完成水印提取和数据恢复，该过程即为水印添加的逆过程。

4. 实验结果与分析

本文实验采用四个随机生成的数据集对设计的算法进行实验验证。随机生成的两个数据集分别包含80条，120条和200条记录。我们从效率和鲁棒性两个方面对本方法与已知的数据库水印方法进行对比。

4.1. 效率对比

首先我们对存储效率进行分析。可知在数据库属性列越少的情况中，使用文本属性生成主键提高的存储效率就越大。同时，因为不用在传输过程中传输主键，通信效率也进一步提高了。

之后我们对遗传模拟退火算法GASA进行效率分析，将其与遗传算法GA，模拟退火算法SA，并行模拟退火算法PSA相比较。实验初始条件设置种群数为10，交叉变异概率为0.99和0.9，指数退温速率为0.9，退温条件为最优值连续20步不变。最终我们采用平均优化值的波动率即相对误差作为指标。

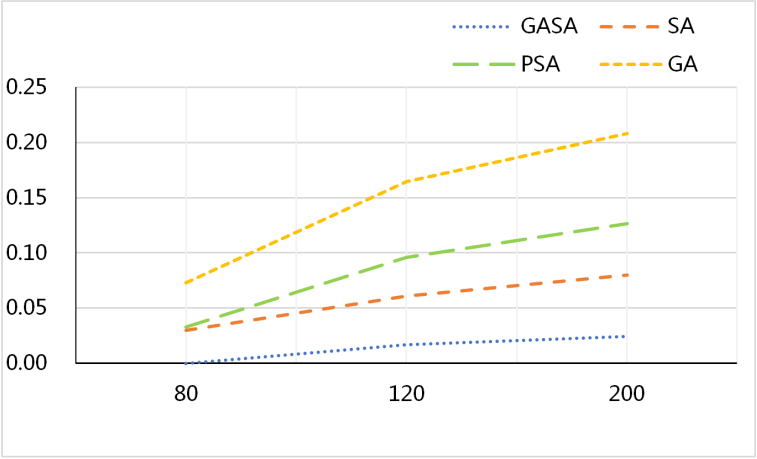

Figure 1. Comparison of perturbation rates of different algorithms
图 1. 不同算法的扰动率对比

我们在三个数据库上都做了实验分析，由图1可知，遗传模拟退火算法GASA的扰动率远低于其他算法，说明其结果较好，作为密钥时计算效率更高。

4.2. 鲁棒性对比

我们采用通用的数据库增加，删除，修改三个攻击方式进行测试遗传算法和直方图位移法GAHSW，文本水印嵌入法SSAC及我们提出的综合水印法GHSA三者的鲁棒性。我们采用被破坏的水印位数与水印总体位数比值ω作为错误率来估计三种方法的鲁棒性。

首先，我们使用数据库增加攻击，我们向数据库增加随机生成的数据和部分数据修改后的数据。由实验结果可知，对于GAHSW和我们提出的GHSA方法来说这两种攻击方式均不会奏效，但是修改后的文本数据可能会对文本水印方法产生一定的破坏效果。图2是在200条数据记录的数据库中随着增加修改后文本数据的数量三种方法ω的变化。

其次我们使用数据库删除攻击。在三个数据库案例中，我们分别实行不同程度的删除攻击，然后取三个数据库收到删除数据后错误率的平均值进行分析，如图3所示。可见我们的方法比其他两种效果都要好。

最后我们使用数据库修改攻击，在三个数据库上删除20%，40%，60%，80%数据并检测出错误率，

之后取平均值，如图 4 所示。可见我们的方法也优于另两种方法。

综上所述，我们可以认为我们的 GHSA 方法在安全性能上优于现有的数据库水印方法。

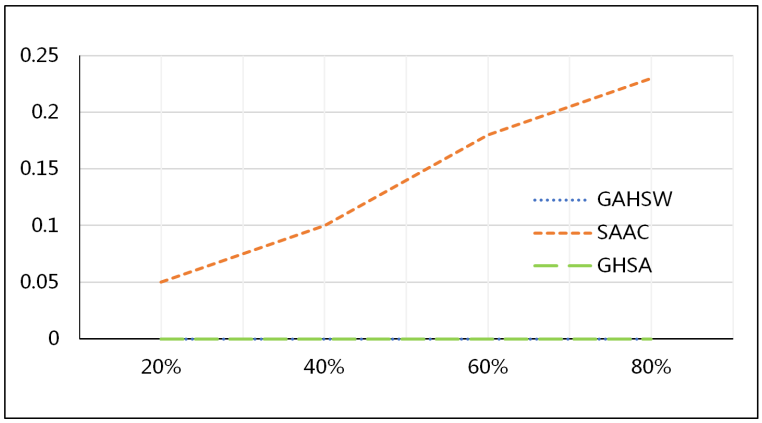

Figure 2. Comparison of the error rates of the three methods with increasing data
图 2. 三种方法在数据增加情况下的错误率对比

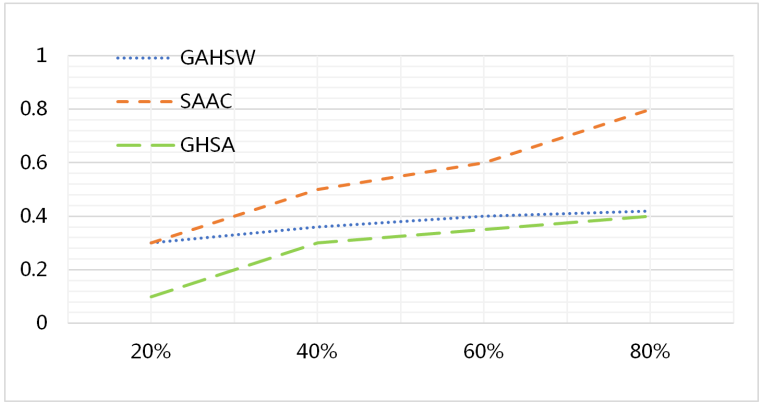

Figure 3. Comparison of the error rates of the three methods in the case of deleting data
图 3. 三种方法在删除数据情况下的错误率比较

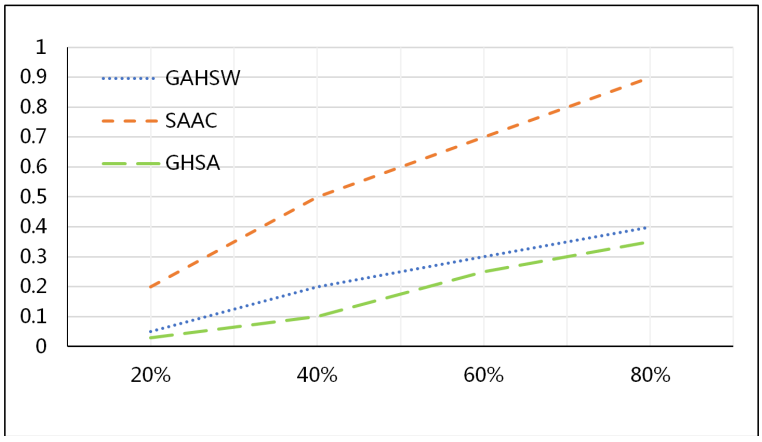

Figure 4. Comparison of the error rates of the three methods in the case of modifying data
图 4. 三种方法在修改数据情况下的错误率比较

5. 结论

我们将优化后的数值型水印方法与文本水印方法相结合，构造了一种新的数据库水印方法。通过理论分析与实验验证，在数值与文本并存的实际数据库上，该方法的鲁棒性更强，安全性更高，并且该方法适用范围更全面，更适合实际生产生活应用。

参考文献

[1] Liu, Y.-C., Ma, Y.-T., Zhang, H.-S., Li, D.-Y. and Chen, G.-S. (2006) A Method for Trust Management in Cloud Computing: Data Coloring by Cloud Watermarking. *International Journal of Automation and Computing*, **8**, 280-285. https://doi.org/10.1007/s11633-011-0583-3

[2] McNickle, M. (2013) Top 10 Data Security Breaches in 2012.

[3] Kiernan, J. (2002) Chapter 15 Watermarking Relational Databases.

[4] Halder, R. and Cortesi, A. (2010) A Persistent Public Water-Marking of Relational Databases. *Information Systems Security International Conference*, Gandhinagar, 17-19 December 2010, 216-230. https://doi.org/10.1007/978-3-642-17714-9_16

[5] Sion, M.J.A.R. and Prabhakar, S. (2004) Rights Protection for Relational Data. *IEEE Transactions on Knowledge and Data Engineering*, **16**, 1509-1525. https://doi.org/10.1109/TKDE.2004.94

[6] Gross Amblard, D. (2011) Query-Preserving Watermarking of Relational Databases and Xml Documents. *ACM Transactions on Database Systems*, **36**, 1-24. https://doi.org/10.1145/1929934.1929937

[7] Shehab, E.B.M. and Ghafoor, A. (2008) Watermarking Relational Databases Using Optimization-Based Techniques. *IEEE Transactions on Knowledge and Data Engineering*, **20**, 116-129. https://doi.org/10.1109/TKDE.2007.190668

[8] Halder, R., Pal, S. and Cortesi, A. (2010) Watermarking Techniques for Relational Databases: Survey, Classification and Comparison. *Journal of Universal Computer Science*, **16**, 3164-3190.

[9] Zhang, Y., Yang, B. and Niu, X.M. (2006) Reversible Water-Marking for Relational Database Authentication. *Journal of Computers*, **17**, 59-65.

[10] Gupta, G. and Pieprzyk, J. (2008) Reversible and Blind Database Watermarking Using Difference Expansion. *Proceedings of the* 1*st International Conference on Forensic Applications and Techniques in Telecommunications, Information, and Multimedia and Workshop*, January 2008, 1-6. https://doi.org/10.4108/e-forensics.2008.2691

[11] Bhattacharya, S. and Cortesi, A. (2013) A Distortion Free Watermark Framework for Relational Databases. *Proceedings of the International Conference on Software and Data Technologies*, Volume 2, Sofia, July 2013, 229-234.

[12] Wu, Y. and Shih, F.Y. (2006) Genetic Algorithm Based Methodology for Breaking the Steganalytic Systems. *IEEE Transactions on Systems Man & Cybernetics Part B Cybernetics A Publication of the IEEE Systems Man & Cybernetics Society*, **36**, 24-31. https://doi.org/10.1109/TSMCB.2005.852474

[13] Jawad, K. and Khan, A. (2013) Genetic Algorithm and Difference Expansion Based Reversible Watermarking for Relational Databases. *Journal of Systems & Software*, **86**, 2742-2753. https://doi.org/10.1016/j.jss.2013.06.023

[14] Imamoglu, M.B., Ulutas, M. and Ulutas, G. (2017) A New Reversible Database Watermarking Approach with Fire-Fly Optimization Algorithm. *Mathematical Problems in Engineering*, **2017**, Article ID: 1387375. https://doi.org/10.1155/2017/1387375

[15] Hu, D.H., Zhao, D. and Zheng, S.L. (2019) A New Robust Approach for Reversible Database Watermarking with Distortion Control. *IEEE Transactions on Knowledge and Data Engineering*, **31**, 1024-1037. https://doi.org/10.1109/TKDE.2018.2851517

[16] Xiang, L.Y., Li, Y., Hao, W., Peng, Y. and Shen, X.B. (2018) Reversible Natural Language Watermarking Using Synonym Substitution and Arithmetic Coding. *CMC-Computers, Materials & Continua*, **55**, 541-559.

[17] Hu, H., Zuo, X., Zhang, W. and Yu, N. (2017) Adaptive Text Steganography by Exploring Statistical and Linguistical Distortion. *IEEE Second International Conference on Data Science in Cyberspace*, Shenzhen, 26-29 June 2017, 145-150.

[18] 王凌, 郑大钟. 一种 GASA 混合策略[J]. 控制理论与应用, 2001, 18(4): 552-554.