

基于 RSA 加密的二维 KPPCT 编码的软件水印方案

杨秋翔, 陈艳琳, 陈够喜, 马巧梅, 李亚女

(中北大学 电子与计算机科学技术学院, 山西 太原 030051)

摘 要: 针对 KPPCT 动态图编码的低数据率问题, 提出了一种基于 RSA 加密的二维 KPPCT(Two Dimensional Radix-k Planted Plane Cubic Tree, TDKPPCT) 编码方案. 该方案首先用 RSA 加密算法把水印值加密, 然后映射到二维 KPPCT 编码结构上. 二维 KPPCT 编码方案是对 KPPCT 编码方案的结构进行了改进, 从而扩大水印值的表示范围. 理论分析和实验结果表明, 该改进方案提高了数据率, 并且具有良好的空间过载和鲁棒性.

关键词: RSA 加密算法; 二维 KPPCT 编码; 数据率; 空间过载; 鲁棒性

中图分类号: TP309

文献标识码: A

文章编号: 1000-7180(2014)04-0044-04

Software Watermarking Scheme of Two Dimension KPPCT Encoding Based on RSA Encryption

YANG Qiu-xiang, CHEN Yan-lin, CHEN Gou-xi, MA Qiao-mei, LI Ya-nv

(School of Electronics and Computer Science and Technology, North University of China, Taiyuan 030051, China)

Abstract: This paper proposed a Two Dimension KPPCT(Two Dimension Radix-k Planted Plane Cubic Tree, TDKPPCT) encoding scheme towards the problem of the lower data rate of KPPCT dynamic graph encoding. This scheme firstly encrypted the original watermarking value using the RSA encryption algorithm, and then mapped the encrypted watermarking value to the TDKPPCT encoding structure, which was improved on the structure of KPPCT encoding scheme. And of course it expanded the range of the watermarking value. Theoretical analysis and experimental results show that this scheme improved the data rate but also had good space overload and robustness.

Key words: RSA encryption algorithm; TDKPPCT encoding scheme; data rate; space overload; robustness

1 引言

软件水印是最近几年出现的一种新型软件版权保护技术, 用来标识作者、发行者、所有者等. 它通过在软件产品中嵌入版权保护信息和身份认证信息, 一旦发生版权纠纷, 软件所有者就可以从软件中提取水印信息来证明其版权.

目前软件水印中动态水印研究的较多, 其中最热门的是 CT 算法^[1], 即动态图水印(DGW)算法, 该算法相对于静态水印来说具有良好的隐蔽性和鲁棒性, 能有效地抵抗代码混淆和优化. 动态水印系统中图结构的选取非常重要, 目前有四种编码结构: Radix-k 编码、排列图编码、父指针树、PPCT(平面

环路树).

文献[2]中在 PPCT 编码结构的基础上引进了基数编码, 扩大了水印值的编码范围, 但是并没有充分地利用叶节点的右指针, 所以数据率提高的不多. 王慧娇等人^[3]提出在 PPCT 编码的基础上, 增加一个虚指针来进行系数编码, 此方案增加了一定程度的编码范围. 文献[4]中提出在嵌入之前先分解水印值, 然后再映射到父指针树上, 并将叶节点与变量值结合起来, 此方案提高了数据率和鲁棒性.

本文通过分析前人的研究成果, 提出了一种基于 RSA 加密算法的二维 KPPCT 动态图水印编码方案. 理论分析和实验验证表明, 相对于改进前的方案, 本方案具有更好的数据率、良好的空间过载和鲁棒性.

收稿日期: 2013-06-01; 修回日期: 2013-07-23

基金项目: 山西省科技攻关项目(20090322004); 中北大学校基金(2013)

2 动态图水印(DGW)

动态图水印是根据目前数论中的大数难分解原理,产生两个大整数 P 和 Q ,用 $N(P \text{ 和 } Q \text{ 的乘积})$ 代表软件版权的信息,即水印值,将 N 转换成某种图结构嵌入到软件中.当输入正确的密钥时,程序会执行生成动态图水印的代码,从而生成水印图拓扑结构,版权所有者可以根据检测到的水印图拓扑结构计算出 N ,从而向公众证明其版权.

2.1 DGW 的嵌入与提取

DGW 的嵌入是指在程序运行时将 N 表示成一种图拓扑结构嵌入到软件中,具体过程描述如图 1 所示.



图 1 水印嵌入过程

水印的提取是水印嵌入的逆过程,由于水印嵌入者对水印图结构的相关特性是清楚的,所以可以很容易地分析得到水印值 N ,证明其版权.

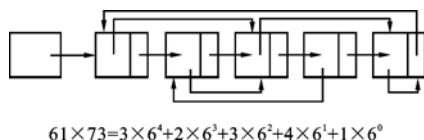
2.2 现有的 DGW 编码方案

提出的改进方案是在 KPPCT 的基础上改进的,下面分别介绍几种编码方案的详细思想.

2.2.1 基数 K 编码

基数 K 编码^[5] 的编码结构是由 1 个头节点和 $K-1$ 个节点的循环链表组成. 除了头节点以外, 每个节点都包含两个指针, 左指针和右指针. 除头结点之外的节点其右指针指向下一个节点, 左指针用来编码系数, 如果为空, 系数为 0; 如果指向自身, 系数为 1, 依次类推. 图 2 所示为基数为 6 的编码. 基数 K 编码所能表示的水印值范围为 $0 \sim k^{t-1} - 1$.

该方案编码率高,但是结构简单,抗攻击性差.

图 2 基数 k 编码

2.2.2 PPCT 枚举编码

PPCT 枚举编码^[5]的实质是在二叉树的性质上增加了一个右指针指向根节点的生成节点(Original Node), 二叉树的所有节点和生成节点组成一个循环链表。

图 3 所示为有 3 个叶节点的 PPCT 枚举编码。 $2n$ 个节点 (n 个叶节点) 的 PPCT 拓扑结构图, 可以

编码的水印值为 $0 \sim \frac{1}{n} C_{2^n-2}^{n-1} - 1$. 而在该集合内的

PPCT 大小定义规则在文献[5]中有详细的定义.

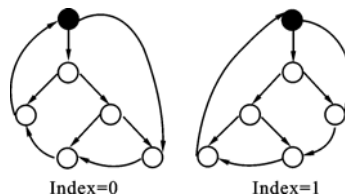


图 3 有 3 个叶节点的 PPCT 枚举树

PPCT 编码结构稳定,抗攻击性好,但是编码效率不高.

2.2.3 KPPCT 编码

KPPCT 编码方案继承了基数 K 编码和 PPCT 编码的优势,数据编码率较高而且稳定.该编码结构是在 PPCT 编码结构的基础上,利用叶节点的右指针来编码系数,此时叶节点的右指针的功能相当于 Radix 编码结构中的左指针的功能.有 m 个叶节点的 KPPCT 结构可以表示的水印值的编码范围为 $0 \sim (m+1)^m - 1$.

图 4 所示为 KPPCT 编码的一个例子.

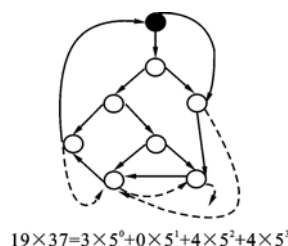


图 4 KPPCT 编码

3 二维 KPPCT 编码方案

该改进方案首先利用 RSA^[6] 加密算法对水印值进行加密, 然后再把水印映射到二维的 KPPCT 编码结构上。

3.1 TDKPPCT 编码方案

把用 RSA 加密算法加密后的水印值映射到二维的 KPPCT(Two Dimension Radix Planted Plane Tree,TDKPPCT)结构上,可以提高编码方案的数据率、隐蔽性和鲁棒性。

TDKPPCT 编码方案是在 KPPCT 编码的基础上增加一个右指针域来参与编码,从而扩大水印值的表示范围.图 5 所示为 KPPCT 编码结构向 TD-KPPCT 编码结构转换的思想.

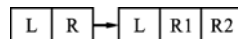


图 5 KPPCT 编码向二维的 KPPCT 转换

图 6 所示为 TDKPPCT 编码结构的一个具体例子。

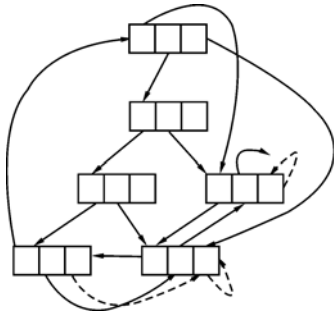


图 6 TDKPPCT 编码结构举例

当有 m 个叶节点时,在 KPPCT 编码方案中参与编码的节点数为 m , 基数为 $(m+1)$. 而在 TDKPPCT 编码方案中,除了 m 个叶节点的右指针 R1 参与编码以外,还有增加的右指针域 R2 也参与编码,所以此方案的基数为 $(m+1) + (m+1) = 2m+2$,表示的水印值为

$$N = \sum_{i=1}^{m-1} e_i (2m+2)^{i_1} + \sum_{i=m}^{2m-1} e_i (2m+2)^{i_2} \quad (1)$$

定义 1 TDKPPCT 编码方案中参与编码的有两个右指针域 R1 和 R2, R2 域的幂 = R1 域的幂 + m , 分别对应式(1)中的 i_1 和 i_2 .

定义 2 生成节点左指针指向的节点的 R1 域的幂为 0, 该节点左指针指向的下一个节点的 R1 域为 1, 依次类推下去, 直到左指针指向生成节点的节点 R1 域的幂为 $m-1$.

定义 3 式(1)中, e_i 是指从相应叶节点的右指针域所指向的叶节点返回原叶节点需要经过的叶节点的个数.

$$e_i = \begin{cases} 0 & \text{叶节点的右指针指向空} \\ j-i+1 & j \geq i; i, j \text{ 为叶节点的位置, 最右为 } 0, \\ & \text{依次加 } 1 \\ n-|j-i|+1 & j < i \end{cases}$$

根据定义, TDKPPCT 编码方案所能表示的水印值范围为 $0 \sim m[(2m+2)^{2m}-1]/(2m+1)$.

图 6 所示的水印值为 $103\ 128 = 0 \times 8^0 + 3 \times 8^1 + 3 \times 8^2 + 1 \times 8^3 + 1 \times 8^4 + 3 \times 8^5$.

3.2 TDKPPCT 编码方案的嵌入和提取

因为在水印值的嵌入之前进行水印值加密, 所以实际嵌入的是破解难度更大的水印值. 程序在正常模式下运行, 如果输入的是特殊序列, 那么程序执行水印嵌入和提取的过程, 否则程序执行正常的功能.

3.2.1 实验环境

微机一台、MyEclipse 8.5、jdk1.4 版本以上、matlab 仿真平台.

3.2.2 水印嵌入与提取

本方案是用 java 语言实现的, 操作实体为节点, 定义成一个类对象, 具体的属性字段如表 1 所示.

表 1 节点的属性字段

nodeID	left	right1	right2	flag	radix1	radix2
--------	------	--------	--------	------	--------	--------

水印的嵌入与提取是互逆的过程, 以图 6 为例, 水印嵌入的部分程序如下:

```
public class MainTDKPPCTEncoding{
    public static void main(String[] args){
        ArrayList<Node> arr;
        Scanner s1 = new Scanner(System.in);
        String pwd1 = s1.nextLine();
        //验证嵌入水印口令是否正确
        if (pwd1.equals("特定序列 1")){
            //embedding 是水印嵌入方法
            arr = new Embed().embedding();
            System.out.println("水印嵌入完成!");
            //此处是水印识别的过程
            Scanner s3 = new Scanner(System.in);
            System.out.println("提取水印值口令:");
            String pwd2 = s3.nextLine();
            //验证提取口令是否正确
            if (pwd2.equals("特定序列 2")){
                //recognizing 是水印提取方法
                double N = new Recognize(arr).
                    recognizing();
            }
            System.out.println("水印值 = " + N);
        }
        else
            System.out.println("识别口令有误");
    }
    else
        执行正常的功能
    }
}
```

提取方法和嵌入方法之间是用参数传递的, 水印结构的类型为 ArrayList<Node> 类型.

3.2.3 水印防篡改

在提取方法内部, 本方案把水印图结构和程序功能结合起来, 实现防篡改. 防篡改代码以不透明谓词的形式隐藏到 if 语句中.

```
double recognizing(ArrayList<Node> arr){
.....
if(false XOR (N1 == N2) ){
return N;
}
else 程序进入故障模式;
.....
}
```

其中 XOR 为异或运算符,根据其性质,如果水印没有被破坏,那么 if 语句的判断为真,反之,若水印被破坏掉,那么执行 else 语句,程序进入故障模式,导致攻击失败,以此来实现防篡改的功能。

4 性能分析

4.1 数据率

数据率即每比特水印结构能表示多少比特代表水印的自然数。假设有 m 个叶节点编码,那么 TDK-PPCT 编码结构中每个叶节点代表的数是 $1+1+1=3$ 。根据文献[7]中数据率的定义,TDKPPCT 编码方案的数据编码率要比 KPPCT 编码方案大很多。本方案的数据率较 KPPCT 编码方案大有提高。

4.2 水印值分析

在相同的节点的情况下,编码的水印值大小如表 2 所示。

表 2 四种编码方式表示的水印值范围

节点数	Radix-k	PPCT	KPPCT	本方案
10	1.0×10^9	1.4×10^1	7.78×10^3	6.19×10^{10}
20	5.24×10^{24}	4.86×10^3	2.59×10^{10}	7.05×10^{26}
50	1.78×10^{83}	1.29×10^{12}	2.27×10^{35}	6.31×10^{85}

4.3 隐蔽性

水印嵌入之后肯定会对原始程序产生一定的影响,如果此影响在可接受范围之内,那么水印方案是可取的。本文通过空间过载分析来评价水印方案的隐蔽性。空间过载是指程序在水印嵌入前后的大小之差,单位为 kB。

分析结果证明在水印值很大时,本方案所引起的空间过载相比于 KPPCT 方案所引起的空间过载要小,所以认为本方案提高了隐蔽性,减少了空间过载。

4.4 鲁棒性

鲁棒性是指水印面临攻击者攻击时的抵抗能力。本方案的数据编码率虽然低于 Radix-k 编码,但是其结构稳定而且引起的空间过载也是在可接受的范围内,又加上其数据率相比于 PPCT、KPPCT 两

种方案来说都要高,所以本方案的整体性能相对是最好的。

5 结束语

本方案利用成熟的 RSA 加密算法的安全性,对水印进行加密,然后把水印值映射到 TDKPPCT 编码结构中进行嵌入。理论分析和实验证明本方案相较于 KPPCT 方案提高了数据编码率、隐蔽性和鲁棒性。

把水印值嵌入到一处容易引起攻击者的察觉,那么如何采用一种新方法分解水印值是本文下一步研究的问题。再者如何进一步分解水印值也是一个难点,下一步的研究是在数据率、隐蔽性和鲁棒性三者的性能之间寻求一个平衡点,以期动态图水印能真正保护版权所有者的权利。

参考文献:

- [1] Collberg C, Thomborson C, Twensend G M. Dynamic Graph-based software watermarking[R]. Tucson: The University of Arizona, 2004.
- [2] 白雪梅. 基于 PPCT 混合编码的防篡改水印方案[J]. 计算机应用与软件, 2011, 28(1): 53-56.
- [3] 王慧娇, 沙宗鲁, 轩爱成. 基于 PPCT 和基数 K 的动态图混合编码方案[J]. 计算机工程与应用, 2010, 46(25): 109-111.
- [4] 李淑芝, 刘猛. 基于变量防篡改的动态图软件水印方案[J]. 计算机工程与科学, 2011, 33(5): 18-21.
- [5] 田奇. 动态图水印算法的研究与应用[D]. 长春: 吉林大学, 2008.
- [6] 刘晓星, 胡畅霞, 刘明生. 公钥加密算法 RSA 的一种快速实现方法[J]. 微计算机信息, 2006, 22(63): 118-119.
- [7] Collberg C, Huntwork A, Carter E, et al. Graph theoretic software watermark implementation, analysis and attacks[C]// Proc of the 6th Workshop of Information Hiding. Toronto, 2004: 192-207.

作者简介:

杨秋翔 男, (1969—), 硕士, 教授。研究方向为网络信息安全和软件保护。

陈艳琳 女, (1989—), 硕士研究生。研究方向为软件水印。

陈够喜 男, (1966—), 博士, 教授。研究方向为信息隐藏和隐写。

马巧梅 女, (1969—), 博士, 副教授。研究方向为图形图像处理。

李亚女 女, (1988—), 硕士研究生。研究方向为无人机航迹规划。