一种隐密消息的提取方法

肖和立　张卫明　刘文芬
（信息工程大学信息工程学院，郑州 450002）

摘　要　该文针对图像空域 LSB 替换的隐写算法，在载体被重复使用的条件下，运用密码学思想对载体进行提取攻击，给出了一种主动隐写分析方法。针对“hide and seek”的提取试验结果表明，该方法可有效地恢复隐写密钥。

关键词　信息隐藏　密码分析　隐写分析　污染分布

文章编号 1002-8331-(2005)15-0034-04　**文献标识码** A　**中图分类号** TP309

An Approach to Extract Hidden Message

Xiao Heli　Zhang Weiming　Liu Wenfen
(Institute of Information Engineering, Information Engineering University, Zhengzhou 450002)

Abstract: An active steganalysis approach aiming at some steganographic algorithms of spatial LSB insertion is presented in this paper. This approach based on cryptanalysis works in the circumstance of a cover image used repeatly. Results of experiment to "hide and seek" are provided to prove that stego key can be found effectively.

Keywords: information hiding, cryptanalysis, steganalysis, contaminated distribution

1　引言

隐写术是信息隐藏的重要分支，它是研究如何把信息嵌入多媒体数据而实现隐蔽通信的技术。对隐写术的攻击称作隐写分析，其目的是对观察到的数据进行检测并提取其中隐藏的秘密消息。一般可分为两类：被动隐写分析：检测观察到的数据中是否含有秘密消息；主动隐写分析：从载密数据中提取隐密消息（或近似恢复消息）[1]。

当前，对于消息提取的方法与手段较少，大部分分析技术停留在被动分析上，很少考虑对隐匿消息的提取。但对密码分析者来说，攻击的最终目的是提取秘密消息。因此，对载密图像进行主动隐写分析尤为重要。2002 年 Chandramouli[1]针对基于扩频通信的隐写方法的一种特殊情况，即同一载体使用两次，前后两次的差别仅在嵌入消息时所用的强度因子不同，给出了提取消息的攻击方法。2004 年 Fridrich 等[2]对 JPEG 图像 LSB 隐写算法（如 F5，Outgess）利用 χ^2 检验给出了一种提取攻击方法。

以图像为载体基于空域 LSB 替换的隐写术应用广泛，它的一般过程为：发送者首先选择一幅“载体图像”，把要发送的消息通过某种空域 LSB 隐写算法嵌入其中生成“载密图像”，然后把“载密图像”发送给接收者，嵌入过程由发送者和接收者所共享的“隐写密钥”所控制，使得只有合法的接收者才可以检测到并提取出消息。由于选图的随机性，同一幅图像被同一消息发送者两次选作载体的可能性是存在的，而且由于互联网的普及，同一幅图像可能被不同的消息发送者拥有，因此图像被两位消息发送者选中进行隐匿操作的可能性亦存在。该文考虑这两种情况。

得到上述两幅载体图像相同的载密图像后，通过比较记录有差异的位置，并对所得差异位置进行分类，借鉴密码分析的思想，可以恢复嵌入消息所用的选位密钥，最终提取秘密消息。笔者对此方法所需的数据量作了理论分析，并以“hide and seek 4.1”为例进行了实验，试验结果表明，当消息嵌入率大于 0.37% 且小于 99.47%时，可以进行有效攻击。

2　算法描述

2.1　基本假设

消息发送者每次从图片库中随机选取一幅图像进行如下隐匿操作：

(1)为了嵌入 l 比特消息，以一个密钥 k 为种子利用随机数发生器 D 生成 l 长的随机位置序列，从而在“载体图像”中选取 l 个像素点。

(2)将要嵌入的消息比特替换所选像素点的最低比特位（LSB）。

(3)设载体图像 C 在位置 (k,j) 的灰度值为 $c_{k,j}$，令 $C=\{c_{k,j}\}$ 表示图像灰度值集合，$c_{k,j}\in[0,255]$，$1\le k\le M$，$1\le j\le N$。

若 $1<k<M$，$1<j<N$，$c_{k,j}$ 的邻域定义为 $N_0(c_{k,j})=\{c_{k-1,j},c_{k+1,j},c_{k,j-1},c_{k,j+1}\}$，令：

$$F(N_0(c_{k,j}))=\frac{1}{4}(c_{k-1,j}+c_{k+1,j}+c_{k,j-1}+c_{k,j+1}) \tag{1}$$

若 $i\in\{1,M\}$，$j\in\{1,N\}$，$c_{k,j}$ 为图像的 4 个顶点，其邻域定义为顶点周围的 3 点，$F(N_0(c_{k,j}))$ 定义为周围 3 点的平均值。

若 $i=1$，$1<j<N$，$c_{1,j}$ 的邻域记为 $N_0(c_{1,j})=\{c_{2,j},c_{1,j-1},c_{1,j+1}\}$，令：

$$F(N_0(c_{1,j}))=\frac{1}{3}(c_{1,j-1}+c_{2,j}+c_{1,j+1}) \tag{2}$$

类似可定义 $i=M$，$1<j<N$ 和 $1<i<M$，$j\in\{1,N\}$的情况。

令 $n=M\times N$，用 $S=\{s_1,s_2,\cdots,s_n\}$表示对应的载密图像，嵌入率为 λ，定义噪声值集合 $W=\{s_i-F(N_0(s_i))\mid i=1,\cdots,n\}$，由于消息都是经过加密的，所以假设嵌入消息序列是伪随机并独立同分布的，且与载体图像的像素点的最低比特位（LSB）独立。则有下面的结论：集合 S 中大约 $(1-\frac{1}{2}\lambda)|S|$ 个值与载体图像对应位置上的像素值相同，而另外大约 $\frac{1}{2}\lambda|S|$ 个值是在嵌入消息过程

作者简介：肖和立，女，硕士研究生，研究方向：信息隐藏。

(C)1994-2011 China Academic Journal Electronic Publishing House. All rights reserved.　http://www.cnki.net

中修改过的。一般可认为，在噪声集 W 中，没修改的像素点对应的噪声数据近似服从 0 均值的正态分布，而修改的像素点对应的噪声数据近似服从均值为±1 的正态分布，但方差不变，笔者的实验结果也证实了这一点。所以集合 W 中的数据可看成是来自一个污染率为 $\frac{1}{2}\lambda$ 的污染分布[3]的样本，其分布为：

$$F_{\frac{1}{2}\lambda}(x)=(1-\frac{1}{2}\lambda)F(x)+\frac{1}{2}\lambda G(x) \tag{3}$$

其中 $F(x)$ 是正态分布 $N(0,\sigma^2)$ 的分布函数，而 $G(x)$ 是正态分布 $N(\pm1,\sigma^2)$ 的分布函数。易知 W 中的数据服从污染分布(3)。

(4)设图 I、图 II 是由同一幅载体图像 C 利用 LSB 隐写机制和不同的选位密钥 k_1 和 k_2 隐藏了不同消息 m_1 和 m_2 而得到的，消息的嵌入率分别为 λ_1 和 λ_2。设 $X=\{x_i\}_{i=1}^n$，$Y=\{y_i\}_{i=1}^n$ 分别表示图 I 和图 II 对应灰度值集合 $\{x_{kij}\}$、$\{y_{kij}\}$ 的向量化，$x_{kij}\in[0,255]$，$y_{kij}\in[0,255]$，$1\le k\le M$，$1\le j\le N$，$n=M\times N$。记：

$$\xi_{1i}=x_i-F(N_0(x_i)),x_i\in X \tag{4}$$

ξ_{1i} 表示图 I 的噪声部分，W_1 为噪声值集合。若图 I 中 x_i 对应的像素点没变，即 $x_i=c_i$，则 $\xi_{1i}\sim N(0,\sigma^2)$；若图 I 中 x_i 对应的像素点变了，即 $x_i\neq c_i$，则 $\xi_{1i}\sim N(\pm1,\sigma^2)$。

类似地，对图 II，记：

$$\xi_{2i}=y_i-F(N_0(y_i)),y_i\in Y \tag{5}$$

ξ_{2i} 表示图 II 的噪声部分，W_2 为噪声值集合。若 $y_i=c_i$，$\xi_{2i}\sim N(0,\sigma^2)$；若 $y_i\neq c_i$，$\xi_{2i}\sim N(\pm1,\sigma^2)$。

2.2 算法描述

截获图 I 和图 II，对两幅图像进行比较得到差异位，构成一个集合 A，即 $A=\{i|x_i\neq y_i\}$，$1\le i\le n$。这些差异位是由于图 I 或图 II 与载体图像在此位置上像素值不同造成的，对这些差异位进行分析，判断差异出现的原因。对 A 中的元素 i，x_i 和 y_i 中有且仅有一个与 c_i 相等，由基本假设(4)可以知道：$\xi_{1i}\sim N(0,\sigma^2)$ 时 $\xi_{2i}\sim N(\pm1,\sigma^2)$ 或 $\xi_{1i}\sim N(\pm1,\sigma^2)$ 时 $\xi_{2i}\sim N(0,\sigma^2)$。

对 A 中的元素 i 定义距离 $d_{1i}=|x_i-F(N_0(x_i))|$，$d_{2i}=|y_i-F(N_0(y_i))|$。

通过比较 d_{1i} 和 d_{2i} 的大小对 x_i 与 c_i 是否相等进行判断，判断准则如下：

(1)若 $d_{1i}\le d_{2i}$，认为 x_i 来自载体图像，即 $x_i=c_i$，此时记 $t_{1,i}=0$，$t_{2,i}=1$；

(2)若 $d_{1i}>d_{2i}$，认为 x_i 来自载密图像，即 $x_i\neq c_i$，此时记 $t_{1,i}=1$，$t_{2,i}=0$；

(3)若 $i\notin A$，记 $t_{1,i}=0$，$t_{2,i}=0$。

判断后得到两个集合 $T_1=\{t_{1,i}\}_{i=1}^n$ 和 $T_2=\{t_{2,i}\}_{i=1}^n$，$n=M\times N$，T_1(T_2)中值为 1 的点对应的位置认为在图 I(图 II)中其像素值在嵌入消息过程中被改变。记 T_1 中值为 1 的点的正确判断率为 r，T_2 中值为 1 的点的正确判断率为 s。即 T_1(T_2)中值为 1 的点判断正确的个数占 T_1(T_2)中值为 1 的点的个数的比。

假设检验：

H_0：$x_i\neq c_i$　H_1：$x_i=c_i$

取伪错误概率 $P_{w1}=p\{H_0|H_1\}$，弃真错误概率 $P_{z1}=p\{H_1|H_0\}$。

对 $i\in A$，H_0 等价于 $x_i\neq c_i$，$y_i=c_i$，H_1 等价于 $x_i=c_i$，$y_i\neq c_i$。

由于方法的一致性，在此仅对 T_1 中值为 1 的点进行分析。下面利用 T_1 恢复图 I 的选位密钥：

定义随机变量 $s_i^{(\tilde{k})}$，$x_i^{(\tilde{k})}$，$1\le i\le n$，$\tilde{k}\in K$，K 是密钥空间：

$$s_i^{(\tilde{k})}=\begin{cases}1,\text{密钥}\tilde{k}\text{产生位置 }i\\0,\text{否则}\end{cases}$$

$$x_i^{(\tilde{k})}=\begin{cases}1,t_{1,i}=1\text{ 且 }s_i^{(\tilde{k})}=1\\0,\text{否则}\end{cases}$$

$x_i^{(\tilde{k})}=1$，表示得到图 I 位置 i 发生变化的判断时密钥 $\tilde{k}$ 产生位置 i。

构造随机变量 $\zeta_m^{(\tilde{k})}$，$1\le m\le n$：

$$\zeta_m^{(\tilde{k})}=\sum_{i=1}^{m}x_i^{(\tilde{k})}$$

表示对 m 个 $t_{1,i}=1$ 的位置做统计，计算这些位置中能由密钥 $\tilde{k}$ 产生的位置的个数，此统计量描述的是密钥 $\tilde{k}$ 所产生位置与算法中所判断出修改位置的符合优势，m 为找出真密钥需要的样本容量：

假设检验：

$\widetilde{H}_0$：$\tilde{k}$ 为真密钥　$\widetilde{H}_1$：$\tilde{k}$ 为伪密钥

设定门限值 T，穷举 $\tilde{k}\in K$，若 $\zeta_m^{(\tilde{k})}\ge T$，接受 $\widetilde{H}_0$；若 $\zeta_m^{(\tilde{k})}<T$，接受 $\widetilde{H}_1$。

取伪错误概率和弃真错误概率分别记为：$P_{w2}=\{\widetilde{H}_0|\widetilde{H}_1\}$ 和 $P_{z2}=\{\widetilde{H}_1|\widetilde{H}_0\}$。密钥空间的势为 $|K|$，为了保证能确定唯一的真密钥，需要 $P_{w2}\le 2^{-|K|}$，通常取 $P_{w2}=2^{-|K|}$，若限定弃真错误概率 P_{z2}，再利用 r，可以共同估计攻击所需的样本容量 m 以及对应的门限值 T。

2.3 算法步骤

step1：比较图 I、图 II，得到差异点集合 A，$A=\{i|x_i\neq y_i\}$，$1\le i\le n$。

step2：对 $i\in A$，若 $d_{1i}\le d_{2i}$，记 $t_{1,i}=0$，$t_{2,i}=1$；
　　对 $i\in A$，若 $d_{1i}>d_{2i}$，记 $t_{1,i}=1$，$t_{2,i}=0$；
　　对 $i\notin A$，$t_{1,i}=0$，$t_{2,i}=0$。

step3：设定门限值 T，穷举 $\tilde{k}\in K$，若 $\zeta_m^{(\tilde{k})}\ge T$，判断 $\tilde{k}$ 为真密钥；若 $\zeta_m^{(\tilde{k})}<T$，判断 $\tilde{k}$ 是伪密钥。

step4：类似 step3 恢复图 II 对应的密钥。

下面的算法分析中，将对 r、s、m 和 T 进行估计。

3 算法分析

3.1 计算错误概率 P_{w1}、P_{z1}

对 $i\in A$，若假设 H_1 成立，则 $\xi_{1i}\sim N(0,\sigma^2)$，$\xi_{2i}\sim N(\pm1,\sigma^2)$，$\xi_{1i}$ 和 ξ_{2i} 相互独立。

$$f_{|\xi_{1i}|}(x)=\begin{cases}0 & ,x<0\\ \frac{2}{\sqrt{2\pi}\,\sigma}e^{-\frac{x^2}{2\sigma^2}} & ,x\ge0\end{cases}$$

$$f_{|\xi_{2i}|}(x)=\begin{cases}0 & ,x<0\\ \frac{1}{\sqrt{2\pi}\,\sigma}\left(e^{-\frac{(x-1)^2}{2\sigma^2}}+e^{-\frac{(x+1)^2}{2\sigma^2}}\right) & ,x\ge0\end{cases}$$

此时：

?1994-2017 China Academic Journal Electronic Publishing House. All rights reserved.　http://www.cnki.net

$$P_{w1}=P\{H_0|H_1\}=P\{|\xi_{1i}|>|\xi_{2i}||H_1\}$$
$$=\int_0^{+\infty}\int_{-\infty}^{+\infty}\frac{1}{\pi\sigma^2}\exp\left(-\left[\frac{(z+u)^2+(u-1)^2}{2\sigma^2}\right]\right)dudz+$$
$$\int_0^{+\infty}\int_0^{+\infty}\frac{1}{\pi\sigma^2}\exp\left(-\left[\frac{(z+u)^2+(u+1)^2}{2\sigma^2}\right]\right)dudz$$

P_{w1}无法由公式简单地表示，但可以用数学软件通过数值计算得出不同 σ 时的 P_{w1}，同理可得 P_{z1}。

σ^2是载体图像的方差，注意到噪声数据 W_1 服从污染分布 $F_{\frac{1}{2}-\lambda_1}(x)=(1-\frac{1}{2}\lambda_1)F(x)+\frac{1}{2}\lambda_1 G(x)$，而 $F(x)$是 $N(0,\sigma^2)$的分布函数，$G(x)$是 $N(1,\sigma^2)$的分布函数。若令$\bar{a}_2$表示样本 W_1 的二阶原点矩，即$\bar{a}_2=\frac{1}{n}\sum_{i=1}^{n}\xi_{1i}^2$，则由文献[4]的结果知：

$$\bar{a}_2=(1-\frac{1}{2}\lambda_1)(\hat{\sigma}^2+0^2)+\frac{1}{2}\lambda_1(\hat{\sigma}^2+1^2)$$

所以：

$$\hat{\sigma}^2=\bar{a}_2-\frac{1}{2}\lambda_1 \tag{6}$$

用(6)式估计方差 σ^2。

3.2 计算 T_1 中值为 1 的点的正确判断率 r

为了计算 T_1 中值为 1 的点判断正确的个数占 T_1 中值为 1 的点的个数的比 r，笔者通过某个值为 1 的点 i 的正确判断概率来近似，即：

$$r\cong P\{x_i\neq c_i|t_{1i}=1\}$$
$$=P\{x_i\neq c_i,y_i=c_i|t_{1i}=1\}+P\{x_i\neq c_i,y_i\neq c_i|t_{1i}=1\}$$
$$=P\{x_i\neq c_i,y_i=c_i|t_{1i}=1\}$$
$$=\frac{P\{x_i\neq c_i,y_i=c_i,t_{1i}=1\}}{P\{t_{1i}=1\}}$$

下面计算 $P\{x_i\neq c_i,y_i=c_i,t_{1i}=1\}$和 $P\{t_{1i}=1\}$，由步骤(2)可知$\{t_{1i}=1\}$等价于 $i\in A$ 且$\{|\xi_{1i}|>|\xi_{2i}|\}$。

$$P\{x_i\neq c_i,y_i=c_i,t_{1i}=1\}$$
$$=P\{x_i\neq c_i,y_i=c_i,|\xi_{1i}|>|\xi_{2i}|\}$$
$$=P\{x_i\neq c_i,y_i=c_i\}P\{|\xi_{1i}|>|\xi_{2i}|\,|\,x_i\neq c_i,y_i=c_i\}$$
$$=\frac{1}{2}\lambda_1(1-\frac{1}{2}\lambda_2)(1-P_{z1})$$
$$P\{t_{1i}=1\}$$
$$=P\{|\xi_{1i}|>|\xi_{2i}|\}$$
$$=P\{|\xi_{1i}|>|\xi_{2i}|,x_i=c_i,y_i\neq c_i\}+P\{|\xi_{1i}|>|\xi_{2i}|,x_i\neq c_i,y_i=c_i\}$$
$$=\frac{1}{2}(1-\frac{1}{2}\lambda_1)\lambda_2P_{w1}+\frac{1}{2}\lambda_1(1-\frac{1}{2}\lambda_2)(1-P_{z1})$$

故 $$r=\frac{\frac{1}{2}\lambda_1(1-\frac{1}{2}\lambda_2)(1-P_{z1})}{\frac{1}{2}(1-\frac{1}{2}\lambda_1)\lambda_2P_{w1}+\frac{1}{2}\lambda_1(1-\frac{1}{2}\lambda_2)(1-P_{z1})}$$

3.3 计算 m 和 T

对差异位置集分类，实际上是对图 I 和图 II 分别估计载体图像被修改的位置。易知真密钥生成的随机位置包含图像中所有被修改的位置，但估计是含错的。下面以图 I 为例分析真、伪密钥生成的随机位置包含某一“估计修改位置”的概率。

定理 1：设 P_{k_1}为真密钥生成的随机位置包含某一“估计修改位置”的概率，P_k为伪密钥生成的随机位置包含某一“估计修改位置”的概率，则 $P_{k_1}-P_k>0$。

证明：对任意密钥$\tilde{k}$：

$$P_{\tilde{k}}=P\{x_i^{(\tilde{k})}=1\}$$
$$=P\{s_i^{(\tilde{k})}=1|t_{1,i}=1\}$$
$$=P\{s_i^{(\tilde{k})}=1,x_i\neq c_i|t_{1,i}=1\}+P\{s_i^{(\tilde{k})}=1,x_i=c_i|t_{1,i}=1\}$$
$$=P\{x_i\neq c_i|t_{1,i}=1\}P\{s_i^{(\tilde{k})}=1|t_{1,i}=1,x_i\neq c_i\}+$$
$$P\{x_i=c_i|t_{1,i}=1\}P\{s_i^{(\tilde{k})}=1|t_{1,i}=1,x_i=c_i\}$$
$$=rP\{s_i^{(\tilde{k})}=1|t_{1,i}=1,x_i\neq c_i\}+(1-r)P\{s_i^{(\tilde{k})}=1|t_{1,i}=1,x_i=c_i\}$$

对真密钥 k_1：$\{x_i\neq c_i\}$说明载体图像中点 i 被真密钥 k_1 选中且修改，即$\{s_i^{(k_1)}=1\}$；$\{t_{1,i}=1,x_i=c_i\}$等价于$\{y_i\neq c_i\}$，$\{s_i^{(k_1)}=1\}$与$\{y_i\neq c_i\}$独立，故 $P_{k_1}=r+(1-r)P\{s_i^{(k_1)}=1\}$。

对伪密钥 k：$\{s_i^{(k)}=1\}$和$\{t_{1,i}=1\}$独立。故 $P_k=P\{s_i^{(k)}=1\}$。

所以 $P_{k_1}-P_k=r(1-P\{s_i^{(k)}=1\})>0$。

$P\{s_i^{(k)}=1\}$和嵌入消息长度密切相关：嵌入消息越多，$P\{s_i^{(k)}=1\}$越大。若随机选位均匀，$P\{s_i^{(k)}=1\}$的值就等于嵌入消息率 λ_1。

由定理 1 知真密钥生成的随机位置包含某一“估计修改位置”的概率大于伪密钥生成的随机位置包含某一“估计修改位置”的概率，这里称之“真密钥优势”，它类似于密码分析中的“符合优势”，当有足够大的优势时，就可以恢复出真密钥。

下面求找出真密钥需要的样本容量 m 和门限值 T：

定理 2：

$$m=\left[\frac{S_w\sqrt{P_k(1-P_k)}+S_z\sqrt{(r+(1-r)P_k)(1-r)(1-P_k)}}{r(1-P_k)}\right]^2$$
$$T=mP_k-S_w\sqrt{mP_k(1-P_k)}$$

证明：对真密钥 k_1，$P_{k_1}=r+(1-r)P_k$，当 m 充分大时，$\zeta_m^{(k_1)}$近似服从正态分布 $N(mP_{k_1},mP_{k_1}(1-P_{k_1}))$；对伪密钥 k，$P_k=P\{s_i^{(k)}=1\}$，当 m 充分大时，$\zeta_m^{(k)}$近似服从正态分布 $N(mP_k,mP_k(1-P_k))$。

取伪概率 $P_{w2}=P\{\tilde{H}_0|\tilde{H}_1\}$，弃真概率 $P_{z2}=P\{\tilde{H}_1|\tilde{H}_0\}$。则：

$$P_{w2}=\int_T^{+\infty}\frac{1}{\sqrt{2\pi mP_k(1-P_k)}}\exp\{-\frac{(x-mP_k)^2}{2mP_k(1-P_k)}\}dx$$
$$=\Phi(\frac{mP_k-T}{\sqrt{mP_k(1-P_k)}})$$
$$P_{z2}=\int_{-\infty}^{T}\frac{1}{\sqrt{2\pi mP_{k_1}(1-P_{k_1})}}\exp\{-\frac{(x-mP_{k_1})^2}{2mP_{k_1}(1-P_{k_1})}\}dx$$
$$=\Phi(\frac{T-mP_{k_1}}{\sqrt{mP_{k_1}(1-P_{k_1})}})$$

通常取 $P_{w2}=2^{-|K|}$，若限定弃真错误概率 P_{z2}，并查表得 S_w 和 S_z 满足：$\Phi(S_w)=2^{-|K|}$，$P_{z2}=\Phi(S_z)$，由此估计攻击所需的样本容量 m：

$$m=\left[\frac{S_w\sqrt{P_k(1-P_k)}+S_z\sqrt{(r+(1-r)P_k)(1-r)(1-P_k)}}{r(1-P_k)}\right]$$

?1994-2017 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

以及对应的门限值 T：

$$T=mP_k-S_w\sqrt{mP_k(1-P_k)}$$

4 实验结果

为了检验算法的效果，笔者对 Hide and Seek[5]进行了试验。Hide and Seek 是一个典型的图像空域 LSB 隐写算法，它以 320×480 的 256 色 GIF 图像为载体，通过替换像素值的 LSB 位来隐藏消息。用来承载消息的像素点由一个随机数发生器来选择，随机数发生器利用 Borland C++3.1 中的 random(num)函数生成随机数，random(num)用一个 16 比特的种子 seed 作初态，参数 num 控制最大偏移步长，与当前所剩消息的长度以及尚未使用的像素的个数有关，所以 Hide and Seek 4.1 中的选位密钥实际上由 16 比特的种子和消息长度两部分组成。算法限定最大消息长度为 19000 字节，前 64 个像素点的 LSB 位是用 IDEA 加密的文件头，从第 65 个像素点开始，在消息长度和种子的控制下生成随机位置，嵌入消息。因此跳过图像的前 64 个像素，以第 65 个像素为起点进行分析寻找种子和确切的消息长度。

笔者对 20 幅 320×480 的 256 色 GIF 图像进行试验，由于样本较小，不必估计样本容量，直接以 T_1 和 T_2 中值为 1 的点为样本进行试验。如果已知确切的消息长度 l_1、l_2(此时密钥量为 2^{16})，令 $P_{w2}=2^{-16}$，$P_{z2}=0.01$，试验表明当嵌入率满足 $0.37\%<\lambda_i<99.47\%$，$i=1,2$ 时，攻击可以成功，即可以恢复出随机种子。下面是对 lena 图像实验的结果：

以下四幅图的横坐标为密钥 $\tilde{k}$，纵坐标为 $\zeta_m^{(\tilde{k})}$。图 1 表示载体图像嵌入消息长度分别为 100，100 时的实验结果，T 分别等于 7 和 8，$\xi^{(k_1)}$ 分别为 255 和 263；图 2 表示嵌入消息长度分别为 16000，16000 时的实验结果，T 分别等于 29789 和 32791，$\xi^{(k_1)}$ 分别为 30892 和 44025，从图中可以看出 $\xi_m^{(k_1)}$ 明显大于 $\xi_m^{(k)}$。

如果未知消息长度，首先要估计消息长度的范围，首先用文献[6]的方法估计消息嵌入率，其误差可控制在±0.005 以内，有大约个可能长度需要测试，此时需要攻击的有效密钥长度为比特 $16+\log_2 190\approx 23.6$。密钥量增大到 $2^{23.6}$，较常出现的情况是密钥确定而长度不能唯一决定。出现这种情况的原因是 Hide and Seek 选位的特殊性：随机种子相同时，以两个邻近长度为参数生成的两条随机序列有很大一部分都是相同的。

5 结束语

该文对图像空域 LSB 隐写术中载体被重复使用的情况给出了一种主动隐写分析方法：在载体图像未知的情况下，通过比较隐藏不同消息的载密图像得到差异位，利用变化点与不变点噪声值分布不同的性质对差异位进行分类，进而利用密码分析思想找到选位密钥，最终提出隐密消息。针对“Hide and Seek

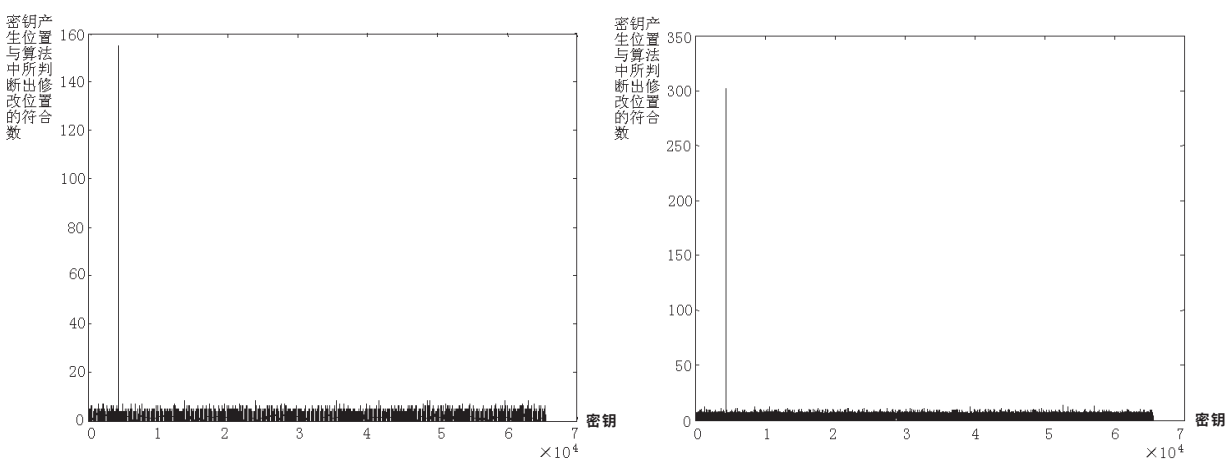

图 1 载体图像分别嵌入 100 比特消息时，真密钥和假密钥的 $\xi_m^{(\tilde{k})}$ 值

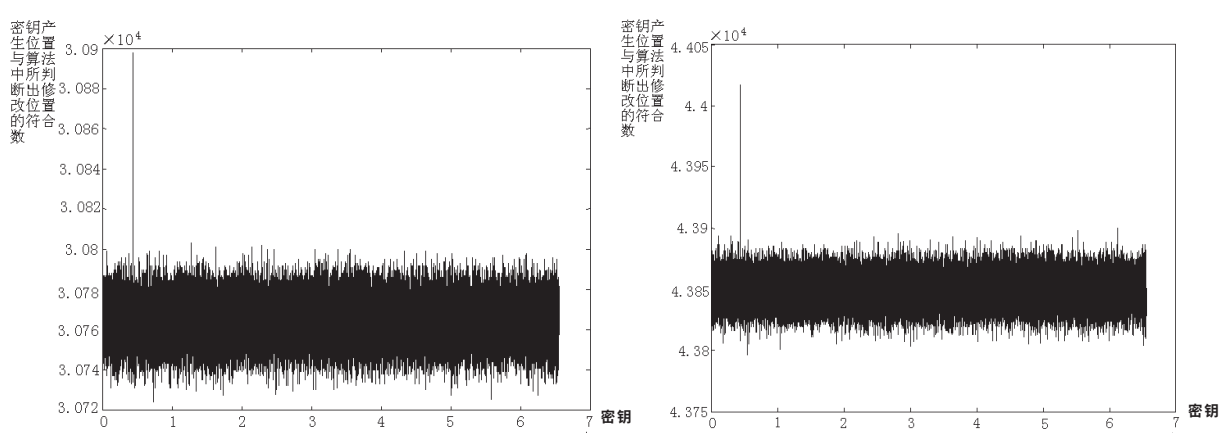

图 2 载体图像分别嵌入 16000 比特消息时，真密钥和假密钥的 $\xi_m^{(\tilde{k})}$ 值

(下转 85 页)

?1994-2017 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net

有三个子群(分别寻求大中小三类装箱序列),子群大小选择 20,反馈调节因子 r=0.1。

实验中每个种群有三个子群,子群大小选择 20,反馈调节因子 r=0.1。

(1)300 代得到的装箱结果

表 1 是实验中连续 5 次运行得到的装箱结果:

表 1 300 代进化装箱效果

Run	所用容器个数	剩余体积比 (V_i/V_{max})	f_i
1	3	0.559443	0.955944
2	3	0.539191	0.953919
3	3	0.528744	0.952874
4	3	0.547005	0.9547
5	3	0.548453	0.954845
Modified CCGA	3	0.38310	0.938310
GA	4	0.64824	0.864824

其中改进 CCGA 和 GA 数据来自文[6]。与实验得到的结果对比可以看出,引入改进的启发式规则和 Coop&compEA 算法,能使装箱的效果大大改进。

(2)进化速度

图 4 显示了算法 5 次连续运行、200 代进化的平均情况(注:由于算法进化速度较快,到了 200 左右就可以基本达到较好的解。文[6]中并没有给出相应的进化曲线):

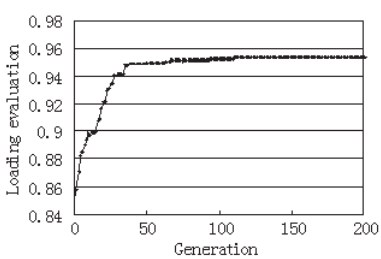

图 4 装箱效果进化图

从图 4 可见,在 25 代之前,算法就可以快速进化到 3 个容器的程度,而 50 代左右达到的装箱效果就已经较为优秀(剩余体积比 0.5 以上,适应度评估达到 0.94~0.95),但依然具有优化的能力。实验显示 Coop&compEA 算法的使用可以使得整个算法求解性能大大提高。

(3)箱子分类方法

前面分析了规则的箱子分类对算法的效果有很大帮助。为了做对比,笔者将表 1 中第一类箱子的长宽高调整为 613×420×295,使整个箱子序列中的尺寸更加规整。同样运行 5 次,每次运行 300 代,结果发现在第 3 代前就可以得到 3 个箱子的装箱结果,而 5 次的平均装箱结果达到剩余体积比 0.61245、装箱评估值 0.961245 的优质解,远远优于未对箱子进行调整的实验结果。可见,尽可能使箱子的尺寸规则可以大大改进装箱效果。

6 结束语

三维装箱问题是一典型的复杂优化问题,对它的研究具有很高的理论意义和实际价值。该文给出了一个结合 Coop&comp EA 和改进装箱启发式规则的新装箱解决方案,并通过实验对比展现了其优秀的求解性能。该文对于求解类似复杂问题,尤其是基于序列的复杂问题有一定的启发和帮助。

另一方面,依然有一些问题有待研究。首先,对于比该文更一般的装箱启发式规则,比如箱子的类型并不能满足一定的类型划分要求,这种情况下将如何对问题进行分解是一个问题。另外,文中也提到了,如何准确地评估子群中个体在整个解中的贡献值,这对于序列优化问题是一个难点,也正是这个问题使得序列优化问题的问题分解格外困难,这方面还需要很多进一步的工作。(收稿日期:2004 年 12 月)

参考文献

1.He D Y,Cha J Z.Research on solution to complex container Loading problem based on genetic algorithm[C].In:Machine Learning and Cybernetics,International Conference on,2002;1:78~82

2.George J A,Robinson D F.A heuristic for Packing boxes into a container[J].Computer and Operational Research,1980;7:147~156

3.Kröger B.Guillotineable bin packing:A genetic approach[J].European Journal of Operational Research,1995;84:645~661

4.Corcoran A L III,Wainwright R L.A genetic algorithm for packing in three dimensions[C].In:Proceedings of the 1992 ACM/SIGAPP Symposium on Applied Computing(SAC'92),Kansas City,KS,1992:1021~1030

5.I T Ikonen,W E Biles,A Kumar et al.A Genetic Algorithm for Packing Three-Dimensional Non-Convex Objects Having Cavities and Holes[C].In:Proceedings of the 7th International Conference on Genetic Algorithms,Michigan State University,East Lansing,MI,1997-07

6.Pimpawat C,Chaiyaratana N.Using a co-operative co-evolutionary genetic algorithm to solve a three-dimensional container loading problem[C].In:The 2001 Congress on Evolutionary Computation(CEC'2001),2001:1197~1204

(上接 37 页)

4.1'的实验表明,该方法可有效地恢复隐写密钥。对于仅得到一幅载密图像的一般情况,还有待研究。

(收稿日期:2005 年 1 月)

参考文献

1.R Chandramouli.A Mathmetical Approach to Steganalysis[C].In:Proc of SPIE on Security and Watermarking of Multimedia Contents IV,Electronic Imaging,2002;4675

2.Jessica Fridrich,Miroslav Goljan,David Soukal.Searching for the Stego-Key[C].In:Proc EI SPIE San Jose,CA,2004-01

3.郑祖康,吴雪明,饶刚.污染数据的处理[J].应用概率统计,1998;14(3):307~312

4.吴为人.两个正态分布的混合分布参数的估计[J].福建农学院学报,1989;18(2):236~243

5.C Moroney.Hide and Seek 4.1.Available:http://www.netlink.co.uk/users/hassop/pgp/hdsk41b.zip

6.Jessica Fridrich,Miroslav Goljan.On Estimation of Secret Message Length in LSB Steganography in Spatial Domain[C].In:Proc EI SPIE Jose San,CA,2004-01:5306

?1994-2017 China Academic Journal Electronic Publishing House. All rights reserved. http://www.cnki.net