2018-2019春季
信息隐藏课程
第4讲 矩阵编码隐写

中国科学院信息工程研究所
INSITUTE OF INFORMATION ENGNEERING CAS

SKLOIS
信息安全国家重点实验室

赵险峰

中国科学院信息工程研究所
信息安全国家重点实验室

2018年10月

纲要

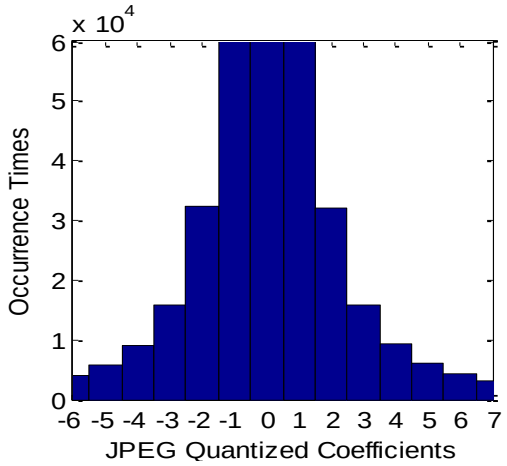
1. 上讲回顾及F3与F4隐写
2. 知识准备：线性分组纠错码
3. 矩阵编码隐写的一般情况
4. F5矩阵编码隐写
5. MME矩阵编码隐写
6. 文献阅读推荐与作业

1-1 上一讲回顾

- 上一讲主要内容
  - 基于统计特征恢复的隐写OutGuess：嵌入后，通过修改未承载消息位置上的LSB，修正了载体的1阶分布（直方图），抵御了卡方与滑动窗口卡方分析
  - 基于模型的隐写MB：通过使嵌入满足1阶分布拟合曲线的约束，保持了1阶分布的基本形状，抵御了卡方攻击与基于分布特性变化的攻击
  - 但是它们均比较复杂，也缺乏进一步的优化空间了
- 上一讲遗留内容
  - 基于修改方式的特征保持方法

1-2 基于调整嵌入修改方式保持统计特征

- 通过调整基本嵌入修改方式，可以在一定程度上保持载体的分布特征
  - 以前介绍过的LSBM（LSB Match），相比采用LSBR，选择采用随机加减1修改的LSBM，有助于避免LSBR带来的相邻数值样点数量接近的现象，这是一种典型的通过调整修改方式保持一阶统计分布特征的例子
- 本讲介绍的F3与F4，是通过调整基本嵌入修改方式**保持JPEG量化系数1阶分布基本特性**的典型方法，为F5提供了好的设计基础
- JPEG量化系数分布特性回顾
  - 对称性、单侧单调性、大小值分布的比例性（梯度递减性）

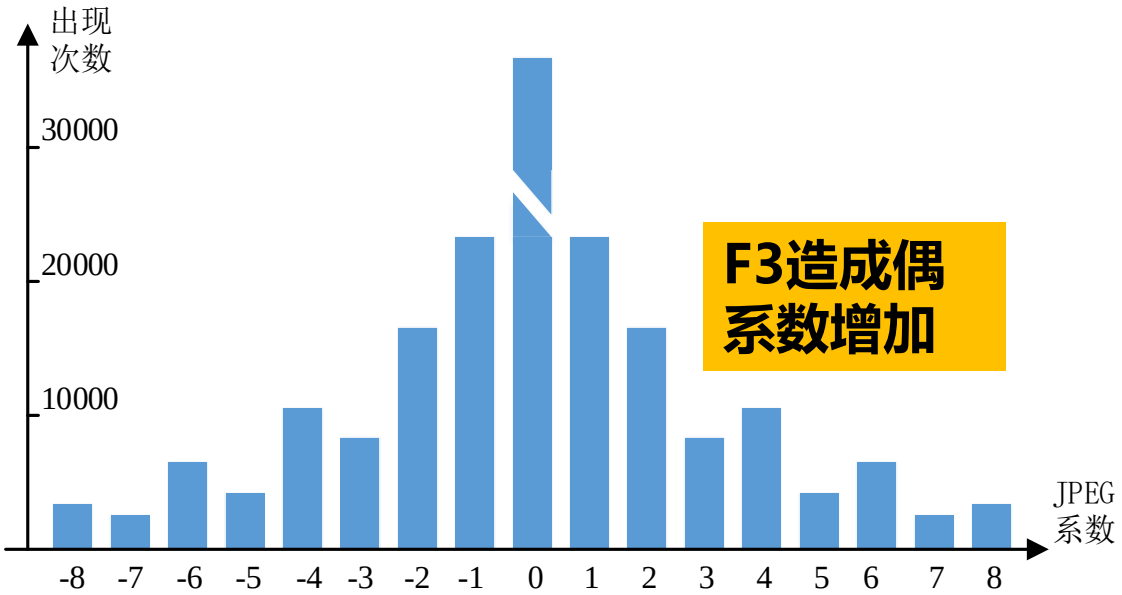


1-3 F3

- F3的嵌入域是JPEG的量化DCT系数，为了避免直接采用LSBR带来的值对接近现象，在嵌入方法上它采取了以下措施
  - 当需要修改时，绝对值降低1；绝对值为1的系数也被使用
  - 当将绝对值为1的系数修改为0时，由于提取算法不能区分这个0值是修改为0 的还是未使用的0值，因此嵌入算法需继续嵌入，直到找到一个偶数或者将一个奇数的绝对值修改为非零偶数
- F3的消息提取只要获得非零系数上的LSB即可
- 显然保持了对称性

1-4 F3的问题

- 以上改动抵御了 χ^2 攻击，但是却**使得偶数的分布增加，在一定程度上没有满足分布函数的单调性要求**
  - 由于载体绝对值为1的数值很多，当被修改为0时，嵌入算法继续嵌入直到找到一个偶数值或者将一个奇数值改为偶数值。这样，**绝对值为1的系数可以支持嵌入1，但是不支持嵌入0，在自身变为0后还要使用或者制造一个偶数；另外，0系数的数量有相应增加，产生收缩现象**



- F4针对不同符号、奇偶性的系数采用了不同的嵌入与消息表示方法，避免了偶数增加的现象
  - 仍然通过减小绝对值的方法进行修改；绝对值为1的系数也被使用
  - 用负偶数、正奇数代表消息比特1，用负奇数、正偶数代表0
- 除了对称性，可验证F4满足分布函数的单调性等性质
  - 注意到 X=1 时有一半可能承载消息比特1，有（正数情况）：

$$P(Y=1)=\frac{1}{2}P(X=1)+\frac{1}{2}P(X=2),\quad P(Y=2)=\frac{1}{2}P(X=2)+\frac{1}{2}P(X=3)$$

$$P(Y=3)=\frac{1}{2}P(X=3)+\frac{1}{2}P(X=4)$$

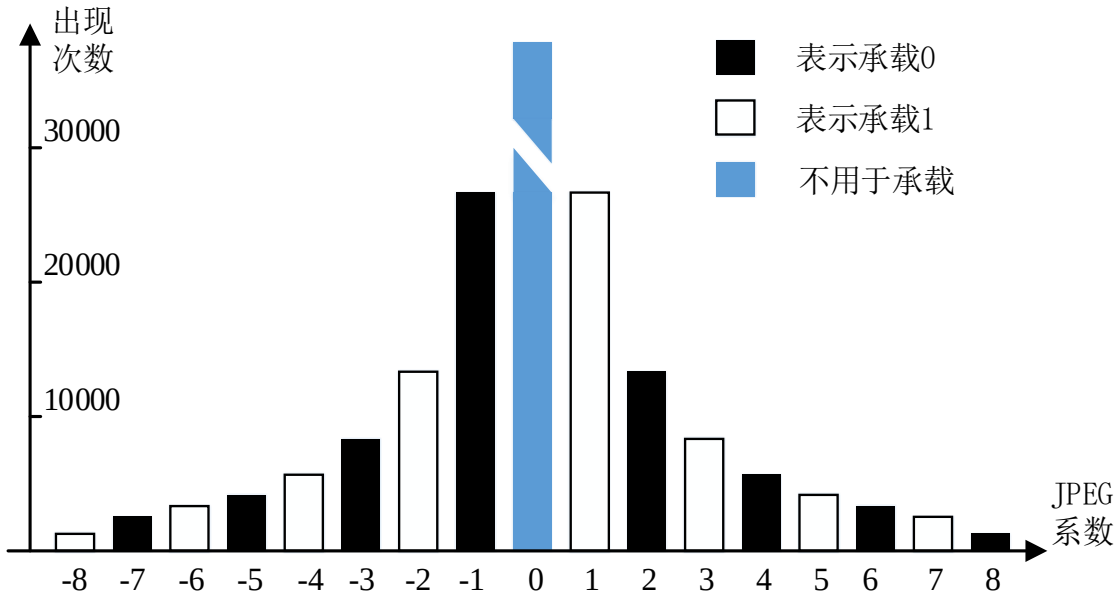
因此得到：$P(Y=1)-P(Y=2)=\frac{1}{2}P(X=1)-\frac{1}{2}P(X=3)>0$

$$P(Y=2)-P(Y=3)=\frac{1}{2}P(X=2)-\frac{1}{2}P(X=4)>0$$

因此得到：$P(Y=1)>P(Y=2)>P(Y=3)$（单调性）

$P(Y=1)-P(Y=2)>P(Y=2)-P(Y=3)$（梯度递减性）

1-6 F4的优势与潜在问题



- F4保持了对称性、单调性、梯度下降性，但仍然存在分布函数形状向0的收缩现象
- F5将在一定程度上减轻这个现象
- 以后介绍的nsF5(No shrinkage F5)将彻底解决该问题

1-7 第一讲回顾：隐写对抗隐写分析的主要方法

- 隐写对抗隐写分析的主要手段是，降低隐写对各类特征的扰动。当前，在保持一定负载率前提下，隐写主要可以通过以下方法不同程度地实现这一目标：
  - 特征保持。隐写的嵌入尽量保持载体的原有特征（非常困难的问题）
  - 降低修改次数。在一定负载率下，提高嵌入效率，减少对载体的修改次数（本讲）
  - 降低修改扰动。降低对载体修改的信号幅度或者能量（本讲）
  - 降低被检测代价。对风险进行定量描述，对载体的修改方式考虑了降低被隐写分析检测的风险
  - 提高应用方式的安全。在应用中考虑了隐写协议安全、抗关联分析等因素

2-1 线性分组纠错码回顾

编码（码距=许用码字的距离）

- (n, k) 分组码是GF(2)上 n 维线性空间中一个k 维子空间 $V_{n,k}$
- 生成矩阵$G_{k\times n}$，校验矩阵$H_{(n-k)\times n}$，有 $GH^{\mathrm{T}} = 0$
- 信息组 $\boldsymbol{m} = (m_1 \cdots m_k)^{\mathrm{T}}$，码字$\boldsymbol{c} = (c_1 \cdots c_n)^{\mathrm{T}} = G^{\mathrm{T}}\boldsymbol{m}$，$H\boldsymbol{c} = 0$

译（解）码 (对 $\boldsymbol{r} = \boldsymbol{c} + \boldsymbol{e}$译码，$\boldsymbol{e}$为可能的噪声分量)

- 计算校验子（Syndrome） $\boldsymbol{s} = \boldsymbol{Hr} = \boldsymbol{H}(\boldsymbol{c} + \boldsymbol{e}) = \boldsymbol{He}$
- $\boldsymbol{e} \neq 0$，根据 s 与 r 纠错

标准阵译码表

码字	$\mathbf{c_1} + \boldsymbol{e_1} = 0 + 0$	$\mathbf{c_2}$	…	$\mathbf{c_i}$	…	$\mathbf{c_{2^k}}$	$\boldsymbol{s_1} = 0$
禁用码字	$\mathbf{e_2}$	$\mathbf{c_2} + \mathbf{e_2}$	…	$\mathbf{c_i} + \mathbf{e_2}$	…	$\mathbf{c_{2^k}} + \mathbf{e_2}$	$\boldsymbol{s_2}$
	$\mathbf{e_3}$	$\mathbf{c_2} + \mathbf{e_3}$	…	$\mathbf{c_i} + \mathbf{e_3}$	…	$\mathbf{c_{2^k}} + \mathbf{e_3}$	$\boldsymbol{s_3}$
			…		…		
	$\mathbf{e_{2^{n-k}}}$	$\mathbf{c_2} + \mathbf{e_{2^{n-k}}}$	…	$\mathbf{c_i} + \mathbf{e_{2^{n-k}}}$	…	$\mathbf{c_{2^k}} + \mathbf{e_{2^{n-k}}}$	$\boldsymbol{s_{2^{n-k}}}$

2-2 线性分组纠错码的启发

- 如用校验子表示 $n-k$ 比特的消息段 m，则对任何一个承载它的 n 比特载体数据段 x（总接近一个码子），可在修改 R 比特限度内（列高）使它进入校验子为 m 的傍集（也称为陪集，Coset），得到 y，使消息接收者可通过计算 $m = Hy = H(x+e)$ 提取消息
- 问题是，怎么修改？ $He = m - Hx$

码字	$c_1 + e_1 = 0 + 0$	c_2	…	c_i	…	c_{2^k}	$s_1 = 0$
禁用码字	e_2	$c_2 + e_2$	…	$c_i + e_2$	…	$c_{2^k} + e_2$	s_2
	e_3	$c_2 + e_3$	…	$c_i + e_3$	…	$c_{2^k} + e_3$	s_3
			…		…		
	$e_{2^{n-k}}$	$c_2 + e_{2^{n-k}}$	…	$c_i + e_{2^{n-k}}$	…	$c_{2^k} + e_{2^{n-k}}$	$s_{2^{n-k}}$

3-1 另一个启发例

在 LSB $x_1, x_2, x_3 \in GF(2)$上，更高效嵌入m_1, m_2（负载率2/3）：

$$m_1 = x_1 \oplus x_3, \quad m_2 = x_2 \oplus x_3, \quad \text{不修改}$$

$$m_1 \neq x_1 \oplus x_3, \quad m_2 = x_2 \oplus x_3, \quad \text{修改 } x_1, e_1 = 1$$

$$m_1 = x_1 \oplus x_3, \quad m_2 \neq x_2 \oplus x_3, \quad \text{修改 } x_2, e_2 = 1$$

$$m_1 \neq x_1 \oplus x_3, \quad m_2 \neq x_2 \oplus x_3, \quad \text{修改} x_3, e_3 = 1$$

- **嵌入效率分析：在3个LSB中最多改动一次（有以上4种情况，平均每次在3个比特上修改0.75次）以嵌入2个比特信息，在负载率固定为2/3下，减少了修改次数，提高了嵌入效率，即从LSBR的2b/次修改提高到$2\text{b}/0.75\text{次} = 2.67\text{b}/\text{次}$**
- **线性变换表示：嵌入可以表达为$y = x + e$，提取可以表示为线性变换**

$$\text{换}\begin{pmatrix} m_1 \\ m_2 \end{pmatrix} = \boldsymbol{m = Hy} = \boldsymbol{H(x+e)} = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \end{pmatrix}\left[\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} + \begin{pmatrix} e_1 \\ e_2 \\ e_3 \end{pmatrix}\right]$$

- **显然，e 的选择与H 也相关，这里其重量不大于1**

3-2 矩阵编码的一般形式

- 给定 一个（n, k）分组码，校验矩阵$H_{(n-k)\times n}$，标准阵译码表
- 在载体分组 $x = (x_1 \cdots x_n)^T$ （如LSB分组）中嵌入消息 $m = (m_1 \cdots m_{n-k})^T$，**要求尽可能少改动 x**，得到隐写后的分组y
  - 嵌入：计算校验子 $s = m - Hx$（校验子向m 偏移，同时考虑消去载体x 的校验子），在标准阵中，找到相应的**傍集首 e**(重量**最轻**)，它满足 $He=H\cdot$傍集中任意元$=s=m-Hx$； 计算 $y=x+e$，得到嵌入后的分组
  - 提取：$Hy = H(x+e) = Hx + He = m - s + s = Hx + m - Hx = m$

码字	$x_1 + e_1 = 0 + 0$	x_2	…	x_i	…	x_{2^k}	$s_1 = 0$
禁用码字	e_2	$x_2 + e_2$	…	$x_i + e_2$	…	$x_{2^k} + e_2$	s_2
	e_3	$x_2 + e_3$	…	$x_i + e_3$	…	$x_{2^k} + e_3$	s_3
			…		…		
	$e_{2^{n-k}}$	$x_2 + e_{2^{n-k}}$	…	$x_i + e_{2^{n-k}}$	…	$x_{2^k} + e_{2^{n-k}}$	$s_{2^{n-k}}$

每行为许用码字的一个陪集；H乘一行中任何元素等于该行对应的s

3-3 矩阵编码存在性理论

结论：根据覆盖半径为R的$[n, k]$线性分组码构造的隐写码，可在 n 个载体样点中通过最多修改 R 个比特传输 $n-k$ 个比特的隐蔽消息

***注：隐写编码应用了纠错码的解码，是信源编码；*选择叠加陪集首 e使得 $He = m - Hx$ 的优化在于，它是满足以上消息提取要求的可叠加分量（傍集）中最轻的一个**

码字	$x_1 + e_1 = 0 + 0$	x_2	…	x_i	…	x_{2^k}	$s_1 = 0$
禁用码字	e_2	$x_2 + e_2$	…	$x_i + e_2$	…	$x_{2^k} + e_2$	s_2
	e_3	$x_2 + e_3$	…	$x_i + e_3$	…	$x_{2^k} + e_3$	s_3
			…		…		
	$e_{2^{n-k}}$	$x_2 + e_{2^{n-k}}$	…	$x_i + e_{2^{n-k}}$	…	$x_{2^k} + e_{2^{n-k}}$	$s_{2^{n-k}}$

3-4 矩阵编码（例）

- **嵌入：载体**（如LSB）分组$x=(1001011)^T$，消息分组$m=(110)^T$，**使用$(7,4)$汉明码校验矩阵H进行隐写；计算校验子:**

$$s=m-Hx=(110)^T-\begin{bmatrix}0&0&0&1&1&1&1\\0&1&1&0&0&1&1\\1&0&1&0&1&0&1\end{bmatrix}(1001011)^T$$

$$=(110)^T-(100)^T=(010)^T=2$$

得到陪集首（错误图样） $e=(0100000)^T$ **(回忆汉明码的1,2...特性)**

最终获得 $y=x+e=(1101011)^T$

- **提取：** $Hy=\begin{bmatrix}0&0&0&1&1&1&1\\0&1&1&0&0&1&1\\1&0&1&0&1&0&1\end{bmatrix}(1101011)^T=(110)^T=m$

- **性质：负载率=**$\frac{n-k}{n}=3/7$，平均比特被修改次数$\frac{R_n}{n}=\frac{0}{n}\cdot\frac{1}{n+1}+\frac{1}{n}\cdot\frac{n}{n+1}=\frac{1}{n+1}$，**嵌入效率**$=3/(\frac{n}{n+1})=24/7=3.43$ **b/次**

n **比特，每次改1比特与全不改共** $n+1$ **个情况**

负载率越小嵌入效率越高

4-1 F5矩阵编码（定义）

- F5实现了**基于海明码的矩阵编码**隐写，在一个分组上最多修改 $R = 1$ 次，采用的**基本嵌入方法是基于F4。**以上例已经给出了F5中隐写编码的一个具体实例
- A. Westfeld在文献中采用了不同的表达方法，希望读者能够区分其每一步与以上一般步骤之间的对应关系

4-2 F5矩阵编码（算法）

- F5作为一个隐写系统还有一系列**辅助处理**，以下给出步骤：
  - 获得嵌入域。若输入的是位图，进行JPEG编码得到DCT量化系数；若输入的是JPEG图像，进行熵编码的解压缩得到DCT量化系数
  - **位置置乱**。根据口令生成的密钥得到一个伪随机数发生器；基于以上伪随机数发生器置乱DCT系数的位置（**思考置乱的优势**）
  - 根据载体中可用系数的数量与消息的长度确定参数 k，计算 $n = 2^k - 1$
  - 基于 $(n = 2^k - 1, k)$ 汉明矩阵编码嵌入消息直到嵌入结束：
    - 按置乱后的顺序取下面 n 个非零系数，在其中的LSB序列中按照以上编码嵌入k比特的消息（基本嵌入方法是F4）
    - 如不需修改，并且还有需要嵌入的消息，则回上面继续嵌入下一个分组
    - 如进行了修改，则判断是不是有系数值收缩到0：如没有，回1）继续嵌入下一分组；**如有，取出一个新的非零系数组成新的一组 n 个非零系数，在其中的LSB序列中按以上编码重新嵌入以上 k比特的消息，直到没有修改或者收缩**。最后，如果还有要嵌入的消息，回上面继续嵌入下一分组
  - 逆置乱后按JPEG标准对量化系数无损压缩（熵编码），得到JPEG文件

4-3 F5矩阵编码（统计保持特性）

- 验证F5保持了分布函数**单调性**与**梯度递减**特性
  - 对LSBR（如其在JPEG系数域中的代表Jsteg），每个系数样点的被修改可能是 $\frac{1}{2}$，假设F5将其降至了 $\frac{\alpha}{2}$，其中 $0 \le \alpha < 1$ （$\alpha = 1$ 是LSBR的特例），则有

$$P(Y=1) = \left(1-\frac{\alpha}{2}\right)P(X=1) + \frac{\alpha}{2}P(X=2)$$
$$P(Y=2) = \left(1-\frac{\alpha}{2}\right)P(X=2) + \frac{\alpha}{2}P(X=3)$$
$$P(Y=3) = \left(1-\frac{\alpha}{2}\right)P(X=3) + \frac{\alpha}{2}P(X=4)$$

有(**以下讲义、论文上有错误**)：

$$P(Y=1)-P(Y=2) = \left(1-\frac{\alpha}{2}\right)\left(P(X=1)-P(X=2)\right) + \frac{\alpha}{2}\left(P(X=2)-P(X=3)\right) > 0$$

$$P(Y=2)-P(Y=3) = \left(1-\frac{\alpha}{2}\right)\left(P(X=2)-P(X=3)\right) + \frac{\alpha}{2}\left(P(X=3)-P(X=4)\right) > 0$$

根据JPEG量化DCT系数的分布特性进一步有：

$$P(Y=1) > P(Y=2) > P(Y=3)$$
$$P(Y=1)-P(Y=2) > P(Y=2)-P(Y=3)$$

5-1 MME（Modified Matrix Encoding）前提

- **定理** 对基于(n,k)线性分组码构造的矩阵编码，若校验子s_i可分解为其他2个校验子之和，即$s_i = s_u + s_v$，则嵌入e_i与嵌入$e_u + e_v$对有效提取消息是等价的
  - **嵌入：** 计算校验子$s_i = m - Hx$，得到其分解形式$s_i = s_u + s_v$；在译码表中，找到对应陪集中的陪集首e_i、e_u与e_v，它们满足$He_i = H(e_u + e_v) = s_i = m - Hx$；计算$y = x + e_u + e_v$，得到嵌入后的分组$y$
  - **提取：** $Hy = H(x + e_u + e_v) = Hx + H(e_u + e_v) = m - s_i + s_i = Hx + m - Hx = m$

码字	$x_1 + e_1 = 0 + 0$	x_2	…	x_i	…	x_{2^k}	$s_1 = 0$
禁用码字	e_2	$x_2 + e_2$	…	$x_i + e_2$	…	$x_{2^k} + e_2$	s_2
	e_3	$x_2 + e_3$	…	$x_i + e_3$	…	$x_{2^k} + e_3$	s_3
			…		…		
	$e_{2^{n-k}}$	$x_2 + e_{2^{n-k}}$	…	$x_i + e_{2^{n-k}}$	…	$x_{2^k} + e_{2^{n-k}}$	$s_{2^{n-k}}$

5-2 MME（Modified Matrix Encoding）前提

- 推论 对基于(n,k)线性分组码构造的矩阵编码，若校验子s_i可分解为其他$1 \le S \le 1$个校验子之和，即$s_i = s_1 + \cdots + s_S$，则嵌入e_i与嵌入$e_1 + \cdots + e_S$对有效提取消息是等价的

码字	$x_1 + e_1 = 0 + 0$	x_2	…	x_i	…	x_{2^k}	$s_1 = 0$
禁用码字	e_2	$x_2 + e_2$	…	$x_i + e_2$	…	$x_{2^k} + e_2$	s_2
	e_3	$x_2 + e_3$	…	$x_i + e_3$	…	$x_{2^k} + e_3$	s_3
			…		…		
	$e_{2^{n-k}}$	$x_2 + e_{2^{n-k}}$	…	$x_i + e_{2^{n-k}}$	…	$x_{2^k} + e_{2^{n-k}}$	$s_{2^{n-k}}$

5-3 MME基本思想

- 从等价消息提取的角度看，矩阵编码提供了很多嵌入方法，如果用修改次数衡量隐蔽性，那可以用嵌入效率作为指标
- 但是，如果从信号扰动的角度看，多次嵌入的扰动不一定比一次扰动程度低
- MME基于以上观察，提供了一种在以上等价嵌入方式中进行优化选择的矩阵编吗方法
- MME的特点之一是，在JPEG编码中完成嵌入，这要求算法的输入是一个位图图像

5-4 MME嵌入修改方式

- 设 $C' = (c'_1, c'_2, \cdots, c'_n)$ 为量化前的DCT系数，将量化后的DCT系数 $C'' = Round(C') = (c''_1, c''_2, \cdots, c''_n)$作为一个嵌入分组，则计算**量化噪声**为$R = C'' - C' = (r_1, r_2, \cdots, r_n)$
- 如果 $r_i \leq 0$，JPEG量化是向下取整，否则是向上取整。在需要进行修改的时候，MME的操作可以描述为：

$$y_i = \begin{cases} -2, & r_i \leq 0, c''_i = -1 \\ Round(c'_i) + 1, & r_i \leq 0, c''_i \neq -1 \\ 2, & r_i > 0, c''_i = 1 \\ Round(c'_i) - 1, & r_i > 0, c''_i \neq 1 \end{cases}$$

- 为使JPEG量化噪声与隐写噪声相互抵消，若JPEG编码是向下取整（$r_i \leq 0$），则通过+1向相反方向修改，若JPEG编码是向上取整（$r_i > 0$），则通过−1向相反方向修改；存在2个特例：在量化系数为 ± 1 时，若按以上原则产生了0，由于0不用于承载信息，则将数值分别改为± 2

5-5 MME的扰动评价方法

- 若每组中第 i 个位置被修改，则JPEG量化与隐写造成的幅度扰动可以表示为 $d_i = |y_i - c_i'|$，等降价与

$$d_i = \begin{cases} 1 + |r_i|, & c_i'' r_i > 0,\ c_i'' = \pm 1 \\ 1 - |r_i|, & otherwise \end{cases}$$

- 以上第1种情况是前面第1与第3种情况的和，第2种是前面第2与第4种情况的和。若仅仅考虑隐写引入的扰动，应从上面减去$|r_i|$，则可以描述为

$$e_i = \begin{cases} 1, & c_i'' r_i > 0,\ c_i'' = \pm 1 \\ 1 - 2|r_i|, & otherwise \end{cases}$$

- MME对F5的提高表现在，在一个分组的全部等价嵌入方法范围内，选择使得$\sum_i^n e_i$最小的方式。若允许校验子分解为S个校验子的和，此时的MME称为MME_S

5-6 MME_2算法

- **获得嵌入域与量化误差。**对输入的是位图，进行JPEG编码得到DCT系数与DCT量化系数，对每个分组，得到$C' = (c'_1, c'_2, \cdots, c'_n)$、$C'' = \mathrm{Round}(C') = (c''_1, c''_2, \cdots, c''_n)$与$R = C'' - C' = (r_1, r_2, \cdots, r_n)$，进行以下嵌入，直到消息嵌入完
- **评估矩阵编码等价嵌入的扰动。**计算矩阵编码的校验子s，计算正常的矩阵编吗的嵌入代价为 e_0；再考查允许2次修改的情况：MME_2只允许分解成2个其他校验子，则 $s = \beta + \gamma$，由于总存在$(n-1)/2$ 个这样的分解，因此，考查$(\beta_1, \gamma_1), \cdots, (\beta_{(n-1)/2}, \gamma_{(n-1)/2})$的分解形式，评估其嵌入扰动$e_1, \cdots, e_{(n-1)/2}$：

$$e_i = \begin{cases} 1+1=2, & c''_{\beta_i} r_{\beta_i} > 0,\ c''_{\gamma_i} r_{\gamma_i} > 0,\ c''_{\beta_i} = \pm 1,\ c''_{\gamma_i} = \pm 1 \\ 1+1-2|r_{\gamma_i}| = 2-2|r_{\gamma_i}|, & c''_{\beta_i} r_{\beta_i} > 0,\ c''_{\beta_i} = \pm 1,\ c''_{\gamma_i} \neq \pm 1 \\ 1+1-2|r_{\beta_i}| = 2-2|r_{\beta_i}|, & c''_{\gamma_i} r_{\gamma_i} > 0,\ c''_{\gamma_i} = \pm 1,\ c''_{\beta_i} \neq \pm 1 \\ 1-2|r_{\beta_i}|+1-2|r_{\gamma_i}| = 2-2(|r_{\beta_i}|+|r_{\gamma_i}|), & otherwise \end{cases}$$

以上公式表达了各种2次嵌入情况的组合，还需比较修改1次的情况

5-7 MME算法（续）

- 分组最小扰动嵌入。选扰动最小的嵌入方法（包括不分解、仅仅修改1次的情况）进行分组嵌入
- 若没有嵌入全部消息，则会进行下一个分组的嵌入，否则进行JPEG的熵编码，得到JPEG文件

- 注：MME在提高安全性的同时，由于需要输入位图，实际是需要编码的边信息，应用上有一定的制约；F5没有这个限制

6 文献阅读推荐与作业

- [1] 教材第4章
- [2] J. Fridrich, D. Soukal. Matrix embedding for large payloads, IEEE Trans. Information Forensics and Security, 1(3): 390-395, 2006
  - 矩阵编码的一般形式
- [3] A. Westfeld. F5 — a steganographic algorithm. In *Proc. IH'01*, LNCS 2137: 289 - 302, Springer-Verlag, 2001
- [4] Y. Kim, Z. Duric, and D. Richards. Modified matrix encoding technique for minimal distortion steganography. In Proc. IH'06, LNCS 4437: 314 – 327, Springer-Verlag, 2007
- 作业：
- (1) 以上文献[2]与[3]中，对F5与MME的嵌入采用了比较特殊的描述方法，请给出这些文献中描述的矩阵编码步骤与一般形式矩阵编码（文献[1]）之间的对应关系说明
- (2)设计一个矩阵编码方案（不能与讲义中相同），给出一个计算实例，并计算嵌入效率

谢谢！

中国科学院信息工程研究所
INSITUTE OF INFORMATION ENGNEERING CAS

SKLOIS
信息安全国家重点实验室