基于回溯法的逆推攻击

张　斌　　金晨辉
(解放军信息工程大学电子技术学院　郑州　450004)

摘　要：该文针对基于乘法电路的前馈序列密码模型的分析问题，在 Golic 逆推攻击算法的基础上，利用回溯法及前馈函数的输入输出相关性，提出了基于回溯法的逆推攻击算法。在解决了回溯法平均计算复杂性的基础上，给出了基于回溯法的逆推攻击算法的平均计算复杂性。新的逆推攻击算法在存储复杂性和平均计算复杂性方面均优于 Golic 算法。

关键词：密码分析；逆推攻击；回溯法；前馈模型；计算复杂性
中图分类号：TN918.1　　**文献标识码**：A　　**文章编号**：1009-5896(2008)10-2464-04

Inversion Attack Based on Back Tracking Method

Zhang Bin　　Jin Chen-hui
(*Electronic Technology Institute*, *PLA Information Engineering University*, *Zhengzhou* 450004, *China*)

Abstract: Based on the back tracking method and the correlation between input and output of the feedforward function, a new inversion attack is proposed to the feedforward stream cipher model with a linear feedback shift register based on a multiplication circuit by improving Golic's inversion attack. The average computational complexity of back tracking method, and the average computational complexity of the new inversion attack are given. In comparison with Golic's algorithm, both the space complexity and the average computational complexity of the new algorithm are less than that of Golic's.

Key words: Cryptanalysis; Inversion attack; Back tracking method; Feedforward model; Computational complexity

1　引言

前馈模型是序列密码的一个最基本模型。前馈模型由一个 n 级线性反馈移位寄存器 LFSR 和一个前馈函数 $f(z_1,z_2,\cdots,z_r)$ 构成。记前馈函数输入变量的抽取位置为 $\{\lambda_i\}_{i=1}^{r}$，并假设 $1\le\lambda_1<\lambda_2<\cdots<\lambda_r\le n$，则以 $(x_1,x_2,\cdots,x_n)$ 为初态的 LFSR 就可产生二元序列 $\{x_i\}_{i=1}^{\infty}$。对于 $i\ge1$，记 $y_i=f(x_{i+\lambda_1},x_{i+\lambda_2},\cdots,x_{i+\lambda_r})$，则序列 $\{y_i\}_{i=1}^{\infty}$ 就是前馈模型的输出序列，又称为前馈序列。

前馈模型分析的基本问题是在 LFSR 的反馈多项式 $g(x)$、抽取位置 $\{\lambda_i\}_{i=1}^{r}$、前馈函数 f 及前馈序列 $\{y_i\}_{i=1}^{\infty}$ 都已知的条件下，求解序列 $\{x_i\}_{i=1}^{\infty}$ 的初态的问题。1996 年，Golic 针对满足

$$f(z_1,z_2,\cdots,z_r)=g(z_1,z_2,\cdots,z_{r-1})\oplus z_r\text{或}f(z_1,z_2,\cdots,z_r)=z_1\oplus g(z_2,z_3,\cdots,z_r)\quad(1)$$

的前馈模型，提出了逆推攻击方法(inversion attack)[1]，逆推攻击包括前向逆推攻击和后向逆推攻击。前向逆推攻击是基于序列 $\{x_j\}_{j=\lambda_1+i}^{\lambda_r+i-1}$ 攻击 x_{λ_r+i}，反之后向逆推攻击是基于序列 $\{x_j\}_{j=\lambda_1+i+1}^{\lambda_r+i}$ 攻击 x_{λ_1+i}。2000 年，Golic 又针对没有上述限定的前馈模型，提出了推广的逆推攻击方法[2]，推广的前向逆推攻击的具体过程可描述如下：

Golic 算法：

步骤 1　取一个正整数 ε，并穷举 $\{x_j\}_{j=\lambda_1+1}^{\lambda_r}$。对 $\{x_j\}_{j=\lambda_1+1}^{\lambda_r}$ 的每个可能值，执行步骤 2；

步骤 2　令 $i=1$；

步骤 3　如果 $i=n+\varepsilon+1$，则执行步骤 4；如果 $i\le n+\varepsilon$，则针对保存的每条序列 $\{x_j\}_{j=\lambda_1+1}^{\lambda_r+i-1}$，在 $i\le n-\lambda_r+\lambda_1$ 时穷举 $x_{i+\lambda_r}$，在 $n-\lambda_r+\lambda_1<i\le n+\varepsilon$ 时，利用 LFSR 的反馈多项式由 $\{x_j\}_{j=i-n+\lambda_r}^{i-1+\lambda_r}$ 计算出 $x_{i+\lambda_r}$。对如此得到的每个 $x_{i+\lambda_r}$，检验 $y_i=f(x_{i+\lambda_1},x_{i+\lambda_2},\cdots,x_{i+\lambda_r})$ 是否成立。如果所有 $x_{i+\lambda_r}$ 都不使上式成立，则返回步骤 3 检验下条可能序列 $\{x_j\}_{j=\lambda_1+1}^{\lambda_r+i-1}$。如果存在使上式成立的 $x_{i+\lambda_r}$，则保存所有使上式成立的 $x_{i+\lambda_r}$ 所对应的序列 $\{x_j\}_{j=\lambda_1+1}^{\lambda_r+i}$，将 i 增 1 后返回步骤 3；

步骤 4　由序列 $\{x_j\}_{j=\lambda_1+1}^{\lambda_r+n+\varepsilon}$ 及 LFSR 的反馈多项式 $g(x)$，建立以 $x_1,x_2,\cdots,x_n$ 为未知变量的一个线性方程组，求解并输出该方程组的解 $(x_1,x_2,\cdots,x_n)$，算法终止。

在 Golic 算法中，ε 通常取不大的正整数，此时 Golic 算法的输出是正确解的概率平均为 $1/(1+2^{-\varepsilon})$。

对 $1\le i\le n+\varepsilon$，Golic 算法需要就当前穷举的 $\{x_j\}_{j=\lambda_1+1}^{\lambda_r}$，保存所有能产生前 i 个乱数的序列 $\{x_j\}_{j=\lambda_1+1}^{\lambda_r+i}$。在文献[3]中，Golic 利用分枝理论给出了 Golic 算法的计算复

2007-03-29 收到，2007-09-24 改回
河南省杰出青年科学基金(0312001800)资助课题

杂性，并证明了 Golic 算法的存储复杂性平均为 $(\lambda_r-\lambda_1)\cdot(n-\lambda_r+\lambda_1)^2$。

在文献[2]中，Golic 曾指出可用树的深度优先搜索方法(即回溯法)实施逆推攻击，但没有给出具体的方法和相应的计算复杂性等性能指标。

本文首先研究并解决了回溯法的平均计算复杂性。接着本文研究了基于回溯法的逆推攻击算法的设计问题，并给出了其存储复杂性和平均计算复杂性。在基于回溯法的逆推攻击算法的设计中，我们利用 $y_i=f(x_{i+\lambda_1},x_{i+\lambda_2},\cdots,x_{i+\lambda_r})$ 与 $x_{i+\lambda_r}$ 或 y_i 与 $x_{i+\lambda_1}$ 之间的相关优势，优先检测正确率大的 $x_{i+\lambda_r}$ 或 $x_{i+\lambda_1}$，从而进一步降低了新算法的计算复杂性；由于基于回溯法的逆推攻击不再需要对每个可能的序列都进行检验，因而使平均计算复杂性得到大幅度降低；由于回溯法不再需要存储所有能产生前 i 个乱数的序列 $\{x_j\}_{j=\lambda_1+1}^{\lambda_r+i}$，从而使存储复杂性从 n 的平方量级降低到 n 的线性量级。

2　回溯法介绍

如果一个问题的解能表示成一个 t 元组 $(x_1,x_2,\cdots,x_t)$，其中 $x_i\in Q_i$，Q_i 是 m_i 点集，且 i 元组 $(x_1,x_2,\cdots,x_i)$ 应满足的约束条件为 $B_i(x_1,x_2,\cdots,x_i)$，则可用回溯法[4]解决该问题：

步骤 1　令 $i=1$，并用 0 元组表示不包含任何值。

步骤 2　在 $i-1$ 元组 $(a_1,a_2,\cdots,a_{i-1})$ 的基础上，对 x_i 的可能取值进行搜索，若找到满足约束条件 $B_i(a_1,a_2,\cdots,a_{i-1},x_i)$ 的 x_i，则按既定规则，选其中一个 x_i 作为 a_i，并将 $i-1$ 元组 $(a_1,a_2,\cdots,a_{i-1})$ 改为 i 元组 $(a_1,a_2,\cdots,a_{i-1},a_i)$，再令 i 增 1 并返回步骤 2；若找不到满足约束条件 $B_i(a_1,a_2,\cdots,a_{i-1},x_i)$ 的 x_i，则去掉 $i-1$ 元组 $(a_1,a_2,\cdots,a_{i-1})$ 中的 a_{i-1} 从而将之改为 $i-2$ 元组 $(a_1,a_2,\cdots,a_{i-2})$，再令 i 减 1，返回步骤 2 并从 a_{i-1} 的下个可能值开始对第 i 个变量 x_i 进行搜索。若找到满足约束条件 $B_t(x_1,x_2,\cdots,x_t)$ 的 t 元组 $(a_1,a_2,\cdots,a_t)$，则输出 t 元组 $(a_1,a_2,\cdots,a_t)$，算法终止。

在分析回溯法的计算复杂性时，可以将检验一次约束条件 $B_i(x_1,x_2,\cdots,x_i)$ 是否成立看作一个基本操作。

3　回溯法平均计算复杂性分析

下面在问题有唯一解的假设下，研究回溯法的平均计算复杂性，即假设解空间中满足约束条件 $B_t(x_1,x_2,\cdots,x_t)$ 的 t 元组 $(x_1,x_2,\cdots,x_t)$ 是唯一的。

本文中基于 $(b_1,\cdots,b_{i-1})$ 检测完 b_i 是指对检测次序早于 b_i 的每个 x_i 及 $(x_{i+1},\cdots,x_t)$ 的每个可能值，对 $i\le j\le t$，判断 $(b_1,\cdots,b_{i-1},b_i)$ 和 $(b_1,\cdots,b_{i-1},x_i,\cdots,x_j)$ 是否满足约束条件的全部过程；检测完 b_1 是指对检测次序早于 b_1 的每个 x_1 及 $(x_2,x_3,\cdots,x_t)$ 的每个可能值，对 $1\le j\le t$，判断 b_1 和 $(x_1,\cdots,x_j)$ 是否满足约束条件的全部过程。

记 $p(c_1\cdots c_j)$ 是 $(c_1,\cdots,c_j)$ 满足约束条件 B_j 的概率，$p(c_{i+1}\cdots c_j\mid b_1\cdots b_i)$ 是在 $(b_1,\cdots,b_i)$ 满足约束条件 B_i 时，$(b_1,\cdots,b_i,c_{i+1},\cdots,c_j)$ 满足约束条件 B_j 的概率，则有

定理 1 设 $(b_1,b_2,\cdots,b_t)$ 是问题的解，且 b_i 是第 β_i 个检测的 x_i 的可能值。对 $2\le i\le t$，令 $S(b_i\mid b_1b_2\cdots b_{i-1})$ 表示基于 $(b_1,b_2,\cdots,b_{i-1})$ 检测完 b_i 的平均计算次数，$S(b_1)$ 表示检测完 b_1 的平均计算次数，则对 $2\le i\le t$，有

$$S(b_i\mid b_1b_2\cdots b_{i-1})=\beta_i+\sum_{j=i}^{t-1}m_{j+1}\sum_{c_i\in\Lambda_i}\sum_{c_{i+1},\cdots,c_j}p(c_ic_{i+1}\cdots c_j\mid b_1b_2\cdots b_{i-1})\quad(2)$$

其中 Λ_i 是检测次序早于 b_i 的点 x_i 构成的集合。特别地，有 $S(b_t\mid b_1b_2\cdots b_{t-1})=\beta_t$ 和

$$S(b_1)=\beta_1+\sum_{j=1}^{t-1}m_{j+1}\sum_{c_1\in\Lambda_1}\sum_{c_2,\cdots,c_j}p(c_1\cdots c_{j-1}c_j)\quad(3)$$

证明　(1)对 $2\le i\le t$，计算 $S(b_i\mid b_1b_2\cdots b_{i-1})$。

首先分析对检测次序不晚于 b_i 的每个 x_i，判断 $S(b_1,\cdots,b_{i-1},x_i)$ 是否满足约束条件 B_i 的整个过程的计算量。由于 b_i 是第 β_i 个检测的 x_i 的可能值，因而该过程的计算复杂性为 β_i。

接着分析对于检测次序早于 b_i 的每个 x_i 及 $(x_{i+1},\cdots,x_t)$ 的每个可能值，对满足 $i\le j\le t$ 的 j，判断 $(b_1,\cdots,b_{i-1},x_i,\cdots,x_j)$ 是否满足约束条件的全部过程的计算量。由于 $x_i\ne b_i$，因而 $(b_1,\cdots,b_{i-1},x_i,\cdots,x_j)$ 在这个过程中最终都被否定，因此，这个过程对 $i\le j\le t-1$，对每个 $(x_i,\cdots,x_j)$ 和每个 y，都进行了 $(b_1,\cdots,b_{i-1},x_i,\cdots,x_j,y)$ 是否满足约束条件 B_{j+1} 的检测。由于 y 共有 m_{j+1} 个，故当满足约束条件 B_j 的 j 元组 $(b_1,\cdots,b_{i-1},x_i,\cdots,x_j)$ 的个数为 N_j 时，有

$$S(b_i\mid b_1b_2\cdots b_{i-1})=\beta_i+\sum_{j=i}^{t-1}m_{j+1}N_j\quad(4)$$

由于 $(b_1,\cdots,b_{i-1},x_i,\cdots,x_j)$ 满足约束条件 B_j 的概率为 $p(x_ix_{i+1}\cdots x_j\mid b_1\cdots b_{i-1})$，故 N_j 的数学期望为

$$E(N_j)=\sum_{c_i\in\Lambda_i}\sum_{c_{i+1},\cdots,c_j}p(c_ic_{i+1}\cdots c_j\mid b_1b_2\cdots b_{i-1})\quad(5)$$

从而有 $S(b_i\mid b_1b_2\cdots b_{i-1})=\beta_i+\sum_{j=i}^{t-1}m_{j+1}\sum_{c_i\in\Lambda_i}\sum_{c_{i+1},\cdots,c_j}p(c_ic_{i+1}\cdots c_j\mid b_1b_2\cdots b_{i-1})$。

(2)只需将(1)中的 i 都换做 1，并同时去掉所有 $b_1b_2\cdots b_{i-1}$，就得到证明 $S(b_1)$ 的计算公式的过程。

(3)由于回溯法基于 $(b_1,b_2,\cdots,b_{t-1})$ 检测完 b_t 时，对检测次序不晚于 b_t 的 x_t 都进行了 $(b_1,b_2,\cdots,b_{t-1},x_t)$ 是否满足约束条件 B_t 的检测，由于 b_t 的检测次序是 β_t，故有 $S(b_t\mid b_1b_2\cdots b_{t-1})=\beta_t$。　　证毕

定理 2　设 $(b_1,b_2,\cdots,b_t)$ 是问题的解，$S(b_1b_2\cdots b_t)$ 是由回溯法求得该解的平均计算复杂性，则有

$$S(b_1b_2\cdots b_t)=S(b_1)+\sum_{i=2}^{t}S(b_i\mid b_1b_2\cdots b_{i-1})\quad(6)$$

证明　由于回溯法求出解 $(b_1,b_2,\cdots,b_t)$ 的过程，就是先检测完 b_1，再依次对 $i=2,3,\cdots,t$，基于 $(b_1,b_2,\cdots,b_{i-1})$ 检测完 b_i 的过程，因而求解过程的计算复杂性就是上述各过程的计算复杂性之和，故本定理成立。

由定理 1 和定理 2 可知，回溯法的平均计算复杂性由满足 $x_1 \neq b_1$ 的概率分布 $p(x_1)$ 及满足 $x_i \neq b_i$ 的诸条件概率 $p(x_i x_{i+1}\cdots x_j \mid b_1\cdots b_{i-1})$ 决定。因此对于具体的求解问题，只要知道这些概率，就可计算出利用回溯法解决该问题的平均计算复杂性。由于 $x_i \neq b_i$，因而 $(b_1,\cdots,b_{i-1},x_i,x_{i+1},\cdots,x_j)$ 都不是正确解，故可以假设它们满足约束条件的概率都相等，即假设 $p(x_i x_{i+1}\cdots x_j \mid b_1\cdots b_{i-1}) = P_{i-1,j}$ 对满足 $x_i \neq b_i$ 的所有 $(x_i,x_{i+1},\cdots,x_j)$ 都成立。显然，当 $p(x_j \mid b_1\cdots b_{i-1}x_i\cdots x_{j-1}) = p_j$ 对满足 $x_i \neq b_i$ 的所有 $(x_i,x_{i+1},\cdots,x_j)$ 恒成立时，有

$$p(x_i x_{i+1}\cdots x_j \mid b_1\cdots b_{i-1}) = p(x_i \mid b_1\cdots b_{i-1})p(x_{i+1} \mid b_1\cdots b_{i-1}x_i)$$
$$\cdots p(x_j \mid b_1\cdots b_{i-1}x_i\cdots x_{j-1}) = \prod_{k=i}^{j} p_k \tag{7}$$

定理 3 设 $p(x_1x_2\cdots x_j) = P_{0,j}$ 和 $p(x_i x_{i+1}\cdots x_j \mid b_1\cdots b_{i-1}) = P_{i-1,j}$ 对诸 i,j 和使 $x_i \neq b_i$ 的 $(x_i,x_{i+1},\cdots,x_j)$ 都成立，则对 $1 \le i \le t$，有

$$S(b_i \mid b_1b_2\cdots b_{i-1}) = \beta_i + (\beta_i - 1)\sum_{j=i}^{t-1} P_{i-1,j} \prod_{k=i+1}^{j+1} m_k \tag{8}$$

其中 $S(b_1 \mid b_1b_2\cdots b_0)$ 表示 $S(b_1)$。

证明 设 $i \le j \le t-1$，则由 $p(x_i x_{i+1}\cdots x_j \mid b_1\cdots b_{i-1}) = P_{i-1,j}$ 知

$$m_{j+1}\sum_{c_i\in A_i}\sum_{c_{i+1},\cdots,c_j} p(c_i\cdots c_j \mid b_1\cdots b_{i-1}) = m_{j+1}\sum_{c_i\in A_i}\sum_{c_{i+1},\cdots,c_j} P_{i-1,j} = (\beta_i - 1)P_{i-1,j}\prod_{k=i+1}^{j+1} m_k$$

从而由定理 1 得 $S(b_i \mid b_1b_2\cdots b_{i-1}) = \beta_i + (\beta_i - 1)\sum_{j=i}^{t-1} P_{i-1,j} \cdot \prod_{k=i+1}^{j+1} m_k$。 证毕

定理 4 题设同定理 3，则回溯法的平均计算复杂性为

$$\sum_{i=1}^{t}\left[E(\beta_i) + (E(\beta_i) - 1)\sum_{j=i}^{t-1} P_{i-1,j}\prod_{k=i+1}^{j+1} m_k\right] \tag{9}$$

证明 由定理 2 知，回溯法的平均计算复杂性为

$$E(S(b_1b_2\cdots b_t)) = E\left(S(b_1)) + \sum_{i=2}^{t} E(S(b_i \mid b_1b_2\cdots b_{i-1})\right)$$
$$= \sum_{i=1}^{t} E\left[\beta_i + (\beta_i - 1)\sum_{j=i}^{t-1} P_{i-1,j}\prod_{k=i+1}^{j+1} m_k\right]$$
$$= \sum_{i=1}^{t}\left[E(\beta_i) + (E(\beta_i) - 1)\sum_{j=i}^{t-1} P_{i-1,j}\prod_{k=i+1}^{j+1} m_k\right]$$

证毕

4 基于回溯法的逆推攻击算法

对前馈函数 $y = f(z_1,z_2,\cdots,z_r)$，令 ρ_r 为 y 与 z_r 的相关系数，即 $\rho_r = p(y = z_r) - p(y \neq z_r)$，则当 $\rho_r > 0$ 时，$x_{i+\lambda_r}$ 与 y_i 相等的概率偏大；当 $\rho_r < 0$ 时，$x_{i+\lambda_r}$ 与 $y_i \oplus 1$ 相等的概率偏大。这说明，在前向逆推攻击中，如果 $x_{i+\lambda_r}$ 有 0，1 两个选择需要检测，则当 $\rho_r > 0$ 时，应当先检测 $x_{i+\lambda_r} = y_i$；当 $\rho_r < 0$ 时，应当先检测 $x_{i+\lambda_r} = y_i \oplus 1$。这种先检测正确率高的候选值的方法必然降低算法的计算复杂性。

同理，令 ρ_l 是 y 与 z_1 的相关系数，在设计后向逆推攻击时，也应当采用基于 ρ_l 的类似方法。逆推攻击究竟是应选用前向攻击还是后向攻击，取决于 $|\rho_r|$ 和 $|\rho_l|$ 的大小。如果 $|\rho_r| > |\rho_l|$，则应选用前向逆推攻击，否则应选用后向逆推攻击。下面仅介绍前向逆推攻击算法的设计，这里我们不妨假设 $\rho_r > 0$。

下面利用回溯法和相关系数 $\rho_r > 0$ 给出改进的 Golic 算法，其前向逆推攻击算法的具体描述如下：

输入 前馈序列 $y_1,y_2,\cdots,y_{n+\varepsilon}$，这里 ε 是一个预选定的不大的正整数；

输出 LFSR 的初态 $x_1,x_2,\cdots,x_n$。

变量解释：

(1) $\{a_j\}_{j=i-n+\lambda_r}^{i+\lambda_r}$ 是当前检验的序列；

(2) $M[i], 0 \le i \le n-\lambda_r+\lambda_1$ 表示 $a_{i+\lambda_r}$ 剩下的选择个数。

算法流程：

步骤 1 穷举 $\{x_j\}_{j=\lambda_1+1}^{\lambda_r}$，并对 $\{x_j\}_{j=\lambda_1+1}^{\lambda_r}$ 的每个可能值 $\{a_j\}_{j=\lambda_1+1}^{\lambda_r}$，执行步骤 2；

步骤 2 令 $i=1$，$M[0]=0$；

步骤 3 (a)如果 $i = n+\varepsilon+1$，则执行步骤 4；如果 $i=0$，则返回步骤 1 检验 $\{x_j\}_{j=\lambda_1+1}^{\lambda_r}$ 的下个可能值 $\{a_j\}_{j=\lambda_1+1}^{\lambda_r}$；

(b)如果 $1 \le i \le n-\lambda_r+\lambda_1$，则穷举 $x_{i+\lambda_r}$，找出使 $y_i = f(a_{i+\lambda_1},\cdots,a_{i+\lambda_{r-1}},x_{i+\lambda_r})$ 成立的所有 $x_{i+\lambda_r}$。当 $x_{i+\lambda_r}$ 的解数为 0 时，若 $M[i-1]=0$，则将 i 减 1 后返回执行步骤 3，若 $M[i-1]=1$，则执行 $a_{i+\lambda_{r-1}} \leftarrow a_{i+\lambda_{r-1}} \oplus 1$ 和 $M[i-1] \leftarrow 0$ 后返回执行步骤 3；当 $x_{i+\lambda_r}$ 的解数为 1 时，执行 $M[i] \leftarrow 0$ 和 $a_{i+\lambda_r} \leftarrow x_{i+\lambda_r}$ 后，将 i 增 1 并返回执行步骤 3；当 $x_{i+\lambda_r}$ 的解数为 2 时，执行 $M[i] \leftarrow 1$ 和 $a_{i+\lambda_r} \leftarrow y_i$ 后，将 i 增 1 并返回执行步骤 3；

(c)如果 $n-\lambda_r+\lambda_1 < i \le n+\varepsilon$，则利用 LFSR 的反馈多项式由 $\{a_j\}_{j=i-n+\lambda_r}^{i-1+\lambda_r}$ 计算出 $a_{i+\lambda_r}$，并检验 $y_i = f(a_{i+\lambda_1},\cdots,a_{i+\lambda_{r-1}},a_{i+\lambda_r})$ 是否成立。当该式成立时，将 i 增 1 并返回执行步骤 3；当该式不成立时，令 $i = n-\lambda_r+\lambda_1$ 后返回执行步骤 3。

步骤 4 由序列 $\{a_j\}_{j=\lambda_1+1}^{\lambda_r+n+\varepsilon}$ 及 LFSR 的反馈多项式 $g(x)$，建立以 $\{a_j\}_{j=1}^{\lambda_r+n+\varepsilon}$ 的初始信号 $x_1,x_2,\cdots,x_n$ 为未知变量的一个线性方程组，解出 $(x_1,x_2,\cdots,x_n)$ 并输出，算法终止。

上述算法只需为正在检验的 $\{a_j\}_{j=\lambda_1+1}^{\lambda_r+n+\varepsilon}$ 开辟 $n+\lambda_r-\lambda_1+\varepsilon$ 比特的存储空间，并为 $M[0],\cdots,M[n-\lambda_r+\lambda_1]$ 开辟 $n-\lambda_r+\lambda_1+1$ 比特的存储空间，不再需要存储通过检验的所有可能序列，从而将存储复杂性降为 n 的线性量级。

此外，当 $\{a_j\}_{j=\lambda_1+1}^{\lambda_r+i}$ 正确时，Golic 算法仍然需要生成并检验使 $\sum_{j=1}^{\lambda_r-\lambda_1+i} b_{\lambda_1+j}2^{j-\lambda_r+\lambda_1-i} > \sum_{j=1}^{\lambda_r-\lambda_1+i} a_{\lambda_1+j}2^{j-\lambda_r+\lambda_1-i}$ 的序列 $\{b_j\}_{j=\lambda_1+1}^{\lambda_r+i}$，但我们改进的算法不再需要对这些序列进行检

验，从而使计算复杂性大幅度降低。

例如，如果 $(a_{\lambda_1+1},\cdots,a_{\lambda_r+i-1},0)$ 是 $(x_{\lambda_1+1},\cdots,x_{\lambda_r+i-1},x_{\lambda_r+i})$ 的正确值，则基于回溯法的逆推攻击不再需要对 $(a_{\lambda_1+1},\cdots,a_{\lambda_r+i-1},1)$ 进行检验。但是，如果 $(a_{\lambda_1+1},\cdots,a_{\lambda_r+i-1},1)$ 能使 $y_i=f(a_{i+\lambda_1},\cdots,a_{i+\lambda_{r-1}},1)$ 成立，Golic 算法仍然需要利用 y_{i+1} 对 $(a_{\lambda_1+1},\cdots,a_{\lambda_r+i-1},1)$ 进行检验，从而增加了算法的计算量。

5　基于回溯法的逆推攻击算法的平均计算复杂性

逆推攻击算法的目标是求出 $n+\varepsilon+1$ 元组 $(X,x_{\lambda_r+1},x_{\lambda_r+2},\cdots,x_{\lambda_r+n+\varepsilon})$，其中 $X\in\{0,1\}^{\lambda_r-\lambda_1}$，且当 $1\le i\le n-\lambda_r+\lambda_1$ 时，有 $x_{\lambda_r+i}\in\{0,1\}$，当 $n-\lambda_r+\lambda_1<i\le n+\varepsilon$ 时，x_{λ_r+i} 只有一个选择。当 $i=0$ 时，$m_0=2^{\lambda_r-\lambda_1}$，由于对 X 的选取没有任何约束条件，故满足第 0 个约束条件 B_0 的概率 $p_0=1$；由于前馈函数 f 的输出仅为 1 个比特，故当 $i\ge1$ 时，满足第 i 个约束条件 B_i 的概率 $p_i=1/2$，且当 $1\le i\le n-\lambda_r+\lambda_1$ 时，有 $m_i=2$；当 $i>n-\lambda_r+\lambda_1$ 时，有 $m_i=1$。

在对 X 穷举时，正确密钥的出现次序 β_0 在 1 至 $2^{\lambda_r-\lambda_1}$ 中是等概的，因此 $E(\beta_0)=2^{\lambda_r-\lambda_1-1}+1/2$。由于 $x_{i+\lambda_r}$ 与 y_i 相等的概率 $p=p(x_{i+\lambda_r}=y_i)>0$，且当 $1\le i\le n-\lambda_r+\lambda_1$ 时，算法对 $x_{i+\lambda_r}$ 采取先 y_i 后 $y_i\oplus1$ 的检测顺序，故有 $E(\beta_i)=1\times p+2\times(1-p)=2-p$；当 $i>n-\lambda_r+\lambda_1$ 时，由于 x_i 只有一个选择，故 $E(\beta_i)=1$。从而有

定理 5　设 $p=p(x_{i+\lambda_r}=y_i)$，且 LFSR 的初态服从均匀分布，则基于回溯法的逆推攻击的平均计算复杂性为

$$2^{\lambda_r-\lambda_1}(n-\lambda_r+\lambda_1+1.5)+(1-p)(n-\lambda_r+\lambda_1)^2/2 \quad (10)$$

证明　将

$$p_i=\begin{cases}1, & i=0\\ 1/2, & i\ge1\end{cases};\ m_i=\begin{cases}2^{\lambda_r-\lambda_1}, & i=0\\ 2, & i\le n-\lambda_r+\lambda_1\\ 1, & i>n-\lambda_r+\lambda_1\end{cases}\text{ 和}$$

$$E(\beta_i)=\begin{cases}2^{\lambda_r-\lambda_1-1}+1/2, & i=0\\ 2-p, & i\le n-\lambda_r+\lambda_1\\ 1, & i>n-\lambda_r+\lambda_1\end{cases}$$

代入定理 4 即可证明本定理。(详细的证明略)

6　结束语

本文针对基于乘法电路的前馈模型，在 Golic 逆推攻击算法的基础上，利用回溯法及前馈函数的输入输出相关性，提出了基于回溯法的逆推攻击算法，在解决了回溯法平均计算复杂性的基础上，给出了基于回溯法的逆推攻击算法的平均计算复杂性。本文给出的基于回溯法的逆推攻击算法的存储复杂性和计算复杂性均优于 Golic 算法。

逆推攻击是一类特殊的分割攻击。回溯法不仅可以应用于逆推攻击，还可应用于其他的分割攻击。本文给出了回溯法的平均计算复杂性，从而使我们能够具体分析基于回溯法的各种分割攻击算法的计算复杂性。

参 考 文 献

[1] Golic J Dj. On the security of nonlinear filter generators. Fast Software Encryption, 1996, LNCS Vol.1039: 173-188.

[2] Golic J Dj, Clark A, and Dawson Ed. Generalized inversion attack on nonlinear filter generators. *IEEE Trans. on Computers*, 2000, 49(10): 1100-1108.

[3] Golic J Dj, Clark A, and Dawson Ed. Inversion attack and branching. ACISP'99, 1999, LNCS Vol.1587: 288-102.

[4] 周培德. 算法设计与分析. 北京：机械工业出版社，1996: 91-95.

张　斌：　男，1982 年生，硕士，研究方向为密码学.

金晨辉：　男，1965 年生，教授，博士生导师，主要研究方向为密码学和信息安全.