减轮 SPECK 算法的不可能差分分析

徐 洪 苏鹏晖* 戚文峰
(信息工程大学数学工程与先进计算国家重点实验室 郑州 450001)

摘 要：SPECK 系列算法是 2013 年由美国国家安全局提出的轻量分组密码算法。算法整体为变形的 Feistel 结构，轮函数为模整数加法、循环移位和异或的组合，即所谓的 ARX 模块。在不可能差分研究方面，目前仅有 LEE 等人给出了 SPECK 64 算法的一些 6 轮不可能差分特征。该文进一步找到了 SPECK 32/64 算法和 SPECK 48/96 算法的一些 6 轮不可能差分特征，并在其前面添加 1 轮后面添加 3 轮，给出了对两个算法的 10 轮不可能差分分析。

关键词：轻量分组密码算法；SPECK 算法；不可能差分分析；不可能差分特征
中图分类号： TP309.7; TN918.1 **文献标识码**： A **文章编号**：1009-5896(2017)10-2479-08
DOI: 10.11999/JEIT170049

Impossible Differential Cryptanalysis of Reduced-round SPECK

XU Hong SU Penghui QI Wenfeng
(*State key Laboratory of Mathematical Engineering and Advanced Computing*,
Information Engineering University, *Zhengzhou* 450001, *China*)

Abstract: SPECK is a family of lightweight block ciphers proposed in 2013 by researches from National Security Agency (NSA) of USA. The algorithm adopts a modified Feistel construction that applies a combination of addition, rotation and XORing (the so-called ARX structure). Up to now, nothing is done on the impossible differential cryptanalysis of the SPECK family except that some 6-round impossible differential characteristics are found by LEE *et al*. In this article, some 6-round impossible differential characteristics of SPECK 32/64 and SPECK 48/96 are found and a 10-round impossible differential cryptanalysis on these two ciphers is presented by adding one round forward and three rounds backward.

Key words: Lightweight block ciphers; SPECK cipher; Impossible differential cryptanalysis; Impossible differential characteristic

1 引言

不可能差分分析是非常有效的分组密码分析方法之一，该概念由 Knudsen 和 Biham 分别独立提出[1,2]，对减轮 AES, Camellia, MISTY1, ARIA[3–6]等典型分组密码算法都有很好的攻击效果。与经典差分密码分析利用高概率差分恢复密钥相反，不可能差分分析利用不可能出现的差分特征快速排除错误候选密钥。

SPECK 系列算法[7]是由美国国家安全局提出的一类轻量分组密码算法。该算法采用变形的 Feistel 结构，轮函数为由模整数加法、循环移位和异或操作混合运算构成的 ARX 部件，主要的非线性运算为模整数加法。目前对 SPECK 系列算法的分析主要包括差分分析、线性分析和零相关线性分析等。Abed, Biryukov, Dinur 等人[8–11]考虑了 SPECK 系列算法的差分分析，Biryukov 等人[12]考虑了最优差分路径的自动搜索问题，而 Fu 等人[13]考虑了更长差分路径的自动搜索问题。Yao 等人[14]给出了对 SPECK 系列算法的初步线性分析，Fu 等人[13]进一步给出了更长线性路径的自动搜索算法。程雨芊等人[15]构造了 SPECK 32/64 和 SPECK 48/96 算法的 6 轮零相关线性特征，给出了对相关算法的 11 轮零相关线性分析，利用 MILP 线性规划方法，Cui 等人[16]搜索发现这是最长的零相关线性特征。利用文献[16]的搜索方法，LEE 等人[17]找到了 SPECK 64 算法的一些 6 轮不可能差分特征，由于计算能力的限制，为了降低搜索量，他们搜索的仅仅是输入差分和输出差分仅含一个非零比特的情况。通过分析

收稿日期：2017-01-16；改回日期：2017-05-15；网络出版：2017-06-23
*通信作者：苏鹏晖 supenghui0309@163.com
基金项目：国家自然科学基金(61100200, 61309017, 61472251, 61502524, 61521003)
Foundation Items: The National Natural Science Foundation of China (61100200, 61309017, 61472251, 61502524, 61521003)

模加法运算的差分扩散性质，本文找到了 SPECK 32/64 和 SPECK48/96 算法的一些 6 轮不可能差分特征，不同于 LEE 等人的搜索结果，本文找到的不可能差分特征的输入差分只含 1 个非零比特，但输出差分含有多个非零比特，其轮数与文献[15]和文献[16]找到的最长零相关特征的轮数一致。

基于找到的这些 6 轮不可能差分特征，在其前面添加 1 轮后面添加 3 轮，本文还给出了对 SPECK 32/64 和 SPECK48/96 算法的 10 轮不可能差分分析。在对 SPECK 32/64 算法的分析中，本文总共猜测的密钥数是 48 bit，攻击的数据复杂度为 2^{32} 个选择明文，总的计算复杂度约为 $2^{62.24}$ 次 10 轮加密。对 SPECK48/96 算法的分析中，本文总共猜测的密钥数是 72 bit，攻击的数据复杂度为 2^{48} 个选择明文，总的计算复杂度约为 $2^{93.59}$ 次 10 轮加密。

论文后续部分安排如下：第 2 节介绍了要使用的符号；第 3 节简要介绍了 SPECK 系列算法；第 4 节给出了 SPECK 32/64 和 SPECK48/96 算法的 6 轮不可能差分特征和 10 轮不可能差分分析；最后是简单的小结。

2 符号说明

本文中用到的一些符号：

X_r, Y_r 为第 r 轮中左右两边的输入。$X_r^{i\sim j}, Y_r^{i\sim j}$ 为 X_r, Y_r 的 i 到 j bit，高位比特在前，从左至右依次为 $n-1, n-2, \cdots, 1, 0$。K_r 为第 r 轮的子密钥。$K_r^{i\sim j}$ 为第 r 轮子密钥 K_r 的 i 到 j bit。Δ 为表示差分。$+$ 为模 2^n 加。$\oplus$ 为按位进行异或。$<<<$ 为循环左移。$>>>$ 为循环右移。

3 SPECK 算法简介

SPECK 系列算法采用变形的 Feistel 结构，轮函数采用 ARX 部件，模整数加法是主要的非线性运算(参见图 1)。本文用 SPECK $2n/mn$ 来表示分组长度为 $2n$ bit、密钥长度 mn bit 的 SPECK 分组密码算法。SPECK 系列算法包括 SPECK 32/64, SPECK 48/72, SPECK 48/96, SPECK 64/96, SPECK 64/128, SPECK 96/144, SPECK 128/256 等 7 个算法。本文重点考虑对 SPECK 32/64 和 SPECK 48/96 算法的不可能差分分析，其迭代轮数分别为 22 轮和 23 轮。

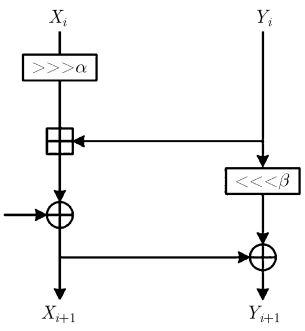

图 1 SPECK 算法的轮函数

SPECK 系列算法的轮函数可以看作变形的两轮 Feistel 结构。记 SPECK 算法第 i 轮的输入为 (X_i, Y_i)，输出为 (X_{i+1}, Y_{i+1})，则状态 (X_i, Y_i) 到 (X_{i+1}, Y_{i+1}) 的更新过程可以描述为

$$\left.\begin{aligned} X_{i+1} &= \left((X_i >>> \alpha) + Y_i\right) \oplus K_i \\ Y_{i+1} &= \left(Y_i <<< \beta\right) \oplus X_{i+1} \end{aligned}\right\} \tag{1}$$

其中，α, β 为系统参数，在 SPECK 32/64 中 $\alpha = 7$, $\beta = 2$，其它系列算法中 $\alpha = 8, \beta = 3$。

SPECK 系列算法的密钥扩展函数利用轮函数来生成所需要的子密钥 K_i。记算法主密钥 $K = \left(L_{m-2}, L_{m-3}, \cdots, L_0, K_0\right)$，其中 $L_i, K_0 \in \mathrm{GF}(2)^n$。密钥扩展函数的输入为主密钥 K，输出为 T 个子密钥 $K_0, K_1, \cdots, K_{T-1}$。计算 K_i 和 L_i 的公式为

$$\left.\begin{aligned} L_{i+m-1} &= \left(K_i + \left(L_i >>> \alpha\right)\right) \oplus i \\ K_{i+1} &= \left(K_i <<< \beta\right) \oplus L_{i+m-1} \end{aligned}\right\} \tag{2}$$

其中，K_i 为第 i 轮子密钥，$0 \le i \le T-1$，m 为各算法密钥块数大小，例如，在 SPECK 32/64 和 SPECK 48/96 算法中都有 $m = 4$。

4 SPECK 算法的不可能差分分析

不可能差分分析利用不可能出现的差分特征排除错误密钥，其分析过程主要包括两步：一是寻找尽可能长的不可能差分特征，二是利用找到的不可能差分特征排除错误密钥从而进行密钥恢复。

假设攻击者已经找到一条 r 轮不可能差分特征 $(\Delta\alpha \nrightarrow \Delta\beta)$，不可能差分分析时通常在此不可能差分特征前后分别加上若干轮加密和解密过程，选择一组明文对 (P, P') 和对应的密文对 (C, C')，猜测它们经过部分加密和部分解密时用到的密钥，并计算不可能差分特征的输入输出差分，若该输入输出差分与不可能差分特征 $(\Delta\alpha \nrightarrow \Delta\beta)$ 匹配，则从候选密钥集中排除此猜测密钥。选择新的明密文对，重复上述过程，可以继续排除错误密钥，直到候选密钥集中只剩下唯一的候选密钥为止。

下面我们先利用模加法的差分扩散性质，给出 SPECK 32/64 和 SPECK 48/96 算法的 6 轮不可能

差分特征，再在其前面添加 1 轮后面添加 3 轮，给出对两种算法的 10 轮不可能差分分析。

4.1 模加法的差分扩散性质

SPECK 算法的轮函数主要包含循环移位操作、异或运算、模加法运算和分支运算等，不可能差分的运算对于循环移位操作、异或运算、分支运算是平凡的，仔细分析模加法运算的差分扩散性质，我们可以得到：

性质 1　令 $z=f(x,y)=x+y \bmod 2^n$ 为 n 比特数的模加法运算，设 $\alpha=(\alpha_{n-1},\alpha_{n-2},\cdots,\alpha_0)$，$\beta=(\beta_{n-1},\beta_{n-2},\cdots,\beta_0)$ 分别为输入 x 和 y 的差分，$\gamma=(\gamma_{n-1},\gamma_{n-2},\cdots,\gamma_0)$ 为输出 z 的差分。记 $l_\alpha=\min\left\{k|\alpha_k=1,\alpha_{k-1}=\alpha_{k-2}=\cdots=\alpha_0=0\right\}$，$l_\beta=\min\left\{k|\beta_k=1,\beta_{k-1}=\beta_{k-2}=\cdots=\beta_0=0\right\}$，则有：

(1)若 $l_\alpha=l_\beta=l$，则 $\gamma_l=\gamma_{l-1}=\cdots=\gamma_0=0$，而当 $l<i\le n-1$ 时，γ_i 不确定；

(2)若 $l_\alpha\ne l_\beta$，不妨设 $l=\min\{l_\alpha,l_\beta\}$，则 $\gamma_l=1$，$\gamma_{l-1}=\cdots=\gamma_0=0$，而当 $l<i\le n-1$ 时，γ_i 不确定。

证明　不妨设输入 $x=\left(x_{n-1},x_{n-2},\cdots,x_0\right),y=\left(y_{n-1},y_{n-2},\cdots,y_0\right)$，输出 $z=\left(z_{n-1},z_{n-2},\cdots,z_0\right)$，进位值为 $c=\left(c_{n-1},c_{n-2},\cdots,c_0\right)$，则有 $z_0=x_0\oplus y_0,c_0=x_0y_0$，一般地，对 $1\le i\le n-1$，有

$$z_i=x_i\oplus y_i\oplus c_{i-1},\ c_i=x_iy_i\oplus x_ic_{i-1}\oplus y_ic_{i-1} \tag{3}$$

于是第 i 比特输出差分值 $\Delta z_i=\Delta x_i\oplus\Delta y_i\oplus\Delta c_{i-1}$，即有 $\gamma_i=\alpha_i\oplus\beta_i\oplus\Delta c_{i-1}$。因此，要确定每个差分值 γ_i，只需要确定每个进位比特的差分值 Δc_i。

另一方面，由进位比特的计算公式 $c_i=x_iy_i\oplus x_ic_{i-1}\oplus y_ic_{i-1}$ 知，注意到 $\Delta x_i=\alpha_i,\Delta y_i=\beta_i$，故有

$$\begin{aligned}\Delta c_i=&(\alpha_i\cdot\beta_i\oplus x_i\cdot\beta_i\oplus y_i\cdot\alpha_i)\\&\oplus(\alpha_i\cdot\Delta c_{i-1}\oplus x_i\cdot\Delta c_{i-1}\oplus c_{i-1}\cdot\alpha_i)\\&\oplus(\beta_i\cdot\Delta c_{i-1}\oplus y_i\cdot\Delta c_{i-1}\oplus c_{i-1}\cdot\beta_i)\end{aligned} \tag{4}$$

(1)若 $l_\alpha=l_\beta=l$，则 $\alpha_l=\beta_l=1$，且对 $0\le i\le l-1$，有 $\alpha_i=\beta_i=0$。故有：

(a)当 $0\le i\le l-1$ 时，$\Delta c_i=0,\gamma_i=\alpha_i\oplus\beta_i\oplus\Delta c_{i-1}=0$；

(b)当 $i=l$ 时，$\gamma_l=\alpha_l\oplus\beta_l\oplus\Delta c_{l-1}=0$，差分值 $\Delta c_l=1\oplus x_l\oplus y_l$ 与输入值 x_l,y_l 有关，取值不确定；

(c)当 $l<i\le n-1$ 时，差分值 Δc_i 不仅与当前输入差分值 α_i,β_i 有关，也与输入比特值 x_i,y_i 和进位比特差分 Δc_{i-1} 有关，后者取值不确定，故差分值 Δc_i 也不确定，此时 $\gamma_i=\alpha_i\oplus\beta_i\oplus\Delta c_{i-1}$ 也不确定。

综上可以得到 $\gamma_l=\gamma_{l-1}=\cdots=\gamma_0=0$，而当 $l<i\le n-1$ 时，γ_i 不确定。

(2)若 $l_\alpha\ne l_\beta$，不妨设 $l=l_\alpha<l_\beta$，则 $\alpha_l=1,\beta_l=0$，且对 $0\le i\le l-1$，有 $\alpha_i=\beta_i=0$。故有：

(a)当 $0\le i\le l-1$ 时，$\Delta c_i=0,\gamma_i=\alpha_i\oplus\beta_i\oplus\Delta c_{i-1}=0$；

(b)当 $i=l$ 时，$\gamma_l=\alpha_l\oplus\beta_l\oplus\Delta c_{l-1}=1$，差分值 $\Delta c_l=y_l\oplus c_{l-1}$ 与输入值 y_l 和进位值 c_{l-1} 有关，取值不确定；

(c)当 $l<i\le n-1$ 时，差分值 Δc_i 不仅与当前输入差分值 α_i,β_i 有关，也与输入比特值 x_i,y_i 和进位比特差分 Δc_{i-1} 有关，它们的取值都是不确定的，故差分值 Δc_i 也不确定。此时差分值 $\gamma_i=\alpha_i\oplus\beta_i\oplus\Delta c_{i-1}$ 也不确定；

综上可以得到 $\gamma_l=1,\gamma_{l-1}=\cdots=\gamma_0=0$，而当 $l<i\le n-1$ 时，γ_i 不确定。　　证毕

4.2 SPECK 32/64 算法的 10 轮不可能差分分析

根据上节的性质 1，结合 SPECK 32/64 算法的结构特点，我们找到了 SPECK 32/64 算法的一个 6 轮不可能差分特征(参见图 2)：(0000 0000 0000 0000,0010 0000 0000 0000) $\not\to_6$ (1000 0000 0000 0000, 1000 0000 0000 0010)，其中 0 表示该位置的差分值为 0, 1 表示该位置的差分值为 1, $*$ 表示不确定的差分，虚线圈起来的 0 和 1 表示不可能差分特征中产生矛盾的比特位置。

事实上，我们发现 SPECK 32/64 算法还存在如下 4 个 6 轮不可能差分特征：

(0000 0000 0000 0000,0001 0000 0000 0000) $\not\to_6$ (1000 0000 0000 0000,1000 0000 0000 0010)

(0000 0000 0000 1000,0000 0000 0000 0000) $\not\to_6$ (1000 0000 0000 0000,1000 0000 0000 0010)

(0000 0000 0001 0000,0000 0000 0000 0000) $\not\to_6$ (1000 0000 0000 0000,1000 0000 0000 0010)

(0000 0000 0100 0000,0000 0000 0000 0000) $\not\to_6$ (1000 0000 0000 0000,1000 0000 0000 0010)

这也是我们能够找到的最长的 5 个不可能差分特征。

以图 2 给出的 6 轮不可能差分特征为基础，在其前面添加 1 轮后面添加 3 轮，可以给出对 10 轮 SPECK 32/64 算法的不可能差分分析(参见图 3)。

图 3 左边和右边分别列出了该 6 轮不可能差分特征向前后扩展时，中间变量 X_i 和 Y_i 差分值的扩散情况以及部分加密和解密过程中用到的子密钥的位置。具体攻击过程如下：

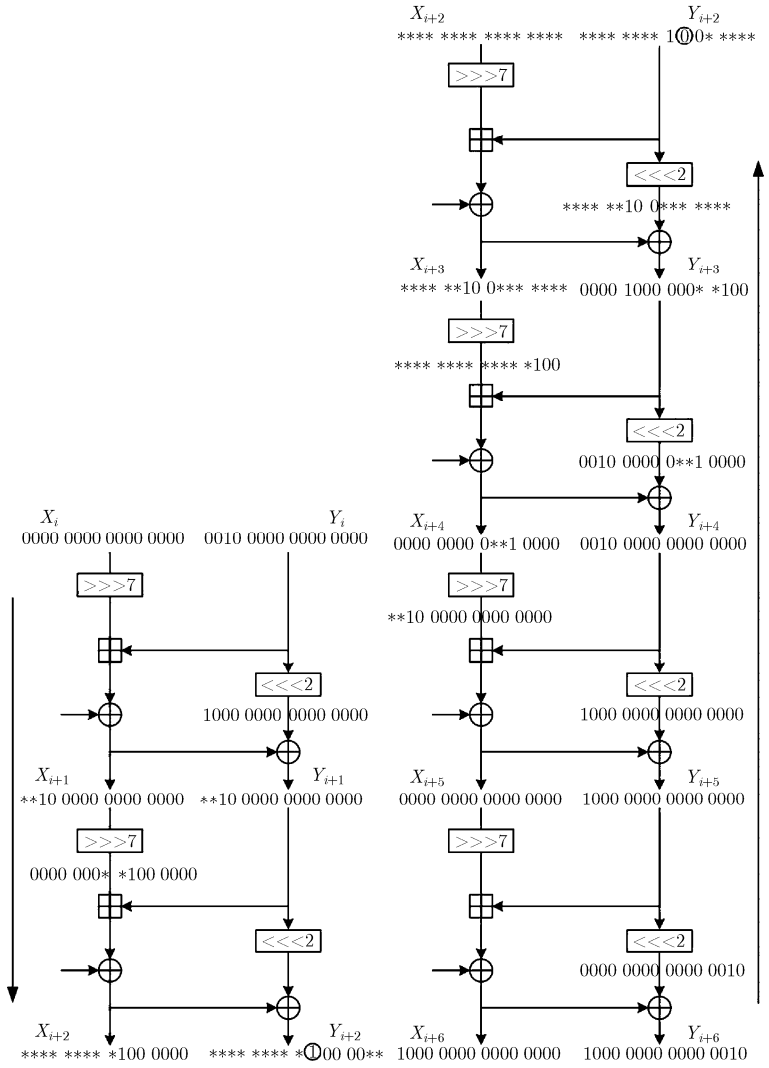

图 2 SPECK 32/64 算法的 6 轮不可能差分特征

(1)选取明文结构如下：

$$P=(X_0,Y_0)=\big(u_1u_2u_3u_4\ u_5u_6u_7u_8\ u_9***u_{10}u_{11}u_{12},$$
$$v_1v_2v_3v_4\ v_5v_6v_7v_8\ v_9v_{10}v_{11}v_{12}\ v_{13}v_{14}v_{15}v_{16}\big)$$
$$P'=\left(X_0^{'},Y_0^{'}\right)=\big(u_1u_2u_3u_4\ u_5u_6u_7u_8\ u_9***\overline{u}_{10}u_{11}u_{12},$$
$$v_1v_2v_3v_4\overline{v}_5v_6v_7v_8\ v_9v_{10}v_{11}v_{12}\ v_{13}v_{14}v_{15}v_{16}\big)$$

其中，u_i,v_i 取固定值，$*$ 取任意值。每个这样的结构包括 2^4 个明文，可以形成大约 2^7 个差分为 $\Delta P=(\Delta X_0,\Delta Y_0)$=(0000 0000 0$***$*100,0000 1000 0000 0000) 的明文对。

(2)选取 2^{28} 个结构，利用所有 2^{32} 个明文可以得到 2^{35} 个明文对。考虑第 1 轮加密，不需要猜测子密钥，对于这些数据对筛选出满足 $(\Delta X_1,\Delta Y_1)=$ (0000 0000 0000 0000,0010 0000 0000 0000) 的数据对，即需筛选出满足 $\Delta S_0^{15\text{-}12}=(0000)$ 的数据对，其中 $\Delta S_0=\big((X_0>>>7)+Y_0\big)\oplus\big(\big(X_0^{'}>>>7\big)+Y_0^{'}\big)$。经过这一步过滤，大约剩余 $2^{35}\times2^{-4}=2^{31}$ 个数据对，计算量约为 $2^{35}\times2=2^{36}$ 次模加运算。

(3)猜测第 10 轮子密钥 $K_9^{12\text{-}0}$ 共 13 bit，对于剩余的每个数据对，筛选出差分 $(\Delta X_9,\Delta Y_9)$ 满足 $\Delta T_8^{3,2}=(00)$ 的数据对，其中 $\Delta T_8=\Delta X_9\oplus\Delta Y_9$, $X_9=((X_{10}\oplus K_9)+Y_9)<<<7, Y_9=(X_{10}\oplus Y_{10})>>>2$，经过此步过滤，大约剩余 $2^{31}\times2^{-2}=2^{29}$ 个数据对，计算量约为 $2^{31}\times2^{13}\times2=2^{45}$ 次模加运算。

(4)猜测第 9 轮子密钥 $K_8^{10\text{-}0}$ 和第 10 轮子密钥 $K_9^{15\text{-}13}$ 共 14 bit，对于剩余的每个数据对，筛选出那些满足 $(\Delta X_8,\Delta Y_8)=$(**** **** **** ****10, **** **** **** ****00) 的数据对，也即筛选满足 $\Delta R_8^{10,9}=(10)$ 的数据对，其中 $\Delta R_8=\big((X_9\oplus K_8)+Y_8\big)\oplus\big(\big(X_9^{'}\oplus K_8\big)+Y_8^{'}\big), Y_8=(X_9\oplus Y_9)>>>2, X_9=$

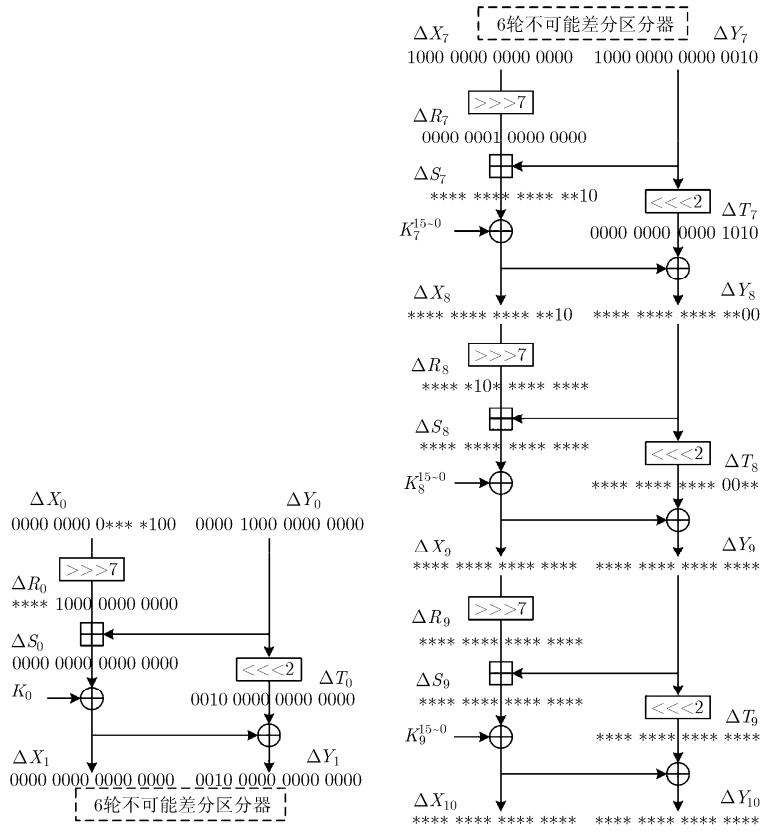

图 3　SPECK 32/64 算法的 10 轮不可能差分分析

$\big((X_{10}\oplus K_9)+Y_9\big)<<<7, Y_9=\big(X_{10}\oplus Y_{10}\big)>>>2$。经过这一步过滤，大约剩余 $2^{29}\times 2^{-2}=2^{27}$ 个数据对，计算量约为 $2^{29}\times 2^{13}\times 2^{14}\times 4=2^{58}$ 次模加运算。

(5)猜测第 9 轮子密钥 $K_8^{15\sim11}$ 共 5 bit，对于剩下的每个数据对，筛选出那些满足 $\Delta Y_7=(1000\ 0000\ 0000\ 0010)$ 的数据对，也即筛选出满足 $\Delta T_7^{15\sim2}=(0000\ 0000\ 0000\ 10)$ 的数据对，其中 $\Delta T_7=\Delta X_8\oplus\Delta Y_8, X_8=\big((X_9\oplus K_8)+Y_8\big)<<<7, Y_8=\big(X_9\oplus Y_9\big)>>>2, X_9=\big((X_{10}\oplus K_9)+Y_9\big)<<<7, Y_9=\big(X_{10}\oplus Y_{10}\big)>>>2$。经过这一步过滤，大约剩余 $2^{27}\times 2^{-14}=2^{13}$ 个数据对，计算量约为 $2^{27}\times 2^{13}\times 2^{14}\times 2^5\times 4=2^{61}$ 次模加运算。

(6)猜测第 8 轮子密钥 $K_7^{15\sim0}$ 共 16 bit，对于剩下的每个数据对，筛选出那些满足 $\Delta X_7=(1000\ 0000\ 0000\ 0000)$ 的数据对，也即筛选出满足 $\Delta R_7^{15\sim2}=(0000\ 0001\ 0000\ 00)$ 的数据对，其中 $\Delta R_7=\big((X_8\oplus K_7)+Y_7\big)\oplus\big(\big(X_8^{'}\oplus K_7\big)+Y_7^{'}\big),\ Y_7=\big(X_8\oplus Y_8\big)>>>2, X_8=\big((X_9\oplus K_8)+Y_8\big)<<<7, Y_8=\big(X_9\oplus Y_9\big)>>>2, X_9=\big((X_{10}\oplus K_9)+Y_9\big)<<<7, Y_9=\big(X_{10}\oplus Y_{10}\big)>>>2$。经过这一步过滤的概率大概是 2^{-14}，计算量约为 $2^{13}\times 2^{13}\times 2^{14}\times 2^5\times 2^{16}\times 6\approx 2^{63.59}$ 次模加运算。如果等式成立说明相应的数据满足 10 轮的不可能差分，则所猜测的密钥是错误的，这时排除相应的密钥猜测 $K_7^{15\sim0}, K_8^{15\sim0}, K_9^{15\sim0}$。

上述过程中总的计算量由第 6 步决定，约为 $2^{63.59}$ 次 1 轮加密。攻击过程中需要猜测的总密钥比特数是 48，第 7 步分析了 2^{13} 个密文对进行排除错误密钥操作之后，大约将会剩余 $2^{48}\times(1-2^{-14})^{2^{13}}\approx 2^{48}\times e^{-1/2}\approx 2^{47.28}$ 个候选密钥。对于这些候选密钥和剩余密钥比特需要遍历搜索，故密钥恢复过程总的计算复杂度约为 $2^{63.59}/10+2^{16}\times 2^{47.28}\approx 2^{63.28}$ 次 10 轮加密。

若同时使用多个 6 轮不可能差分特征，总复杂度可进一步降低。例如，若同时使用 4 个 6 轮不可能差分特征，则此时错误率降为 $p=(1-2^{-14})^{2^{13}\times 4}\approx e^{-2}\approx 2^{-2.88}$，密钥恢复过程总的计算复杂度约为 $4\times 2^{63.59}/10+2^{16}\times 2^{48}\times 2^{-2.88}\approx 2^{62.24}$ 次 10 轮加密。

综上所述，若同时使用 4 个 6 轮不可能差分特征对 10 轮 SPECK 32/64 算法进行不可能差分分析，则攻击的时间复杂度约为 $2^{62.24}$ 次 10 轮加密，攻击时需要用到所有 2^{32} 个明文。

4.3 SPECK 48/96 算法的 10 轮不可能差分分析

类似于 SPECK 32 算法的分析，利用模加法的差分扩散性质，我们找到了 SPECK 48/96 算法的一些 6 轮不可能差分特征，图 4 给出了 SPECK 48/96

算法的一个 6 轮不可能差分特征实例，其中 0 表示该位置的差分值为 0，1 表示该位置的差分值为 1，$*$ 表示不确定的差分，虚线圈起来的 0, 1 表示不可能差分特征中产生矛盾的比特位置。

以图 4 给出的 6 轮不可能差分特征为基础，在其前面添加 1 轮后面添加 3 轮，可以给出对 SPECK 48/96 算法的 10 轮不可能差分分析(参见图 5)。具体攻击过程与 SPECK 32/64 算法的不可能差分分析类似，攻击过程中需要猜测的总密钥比特数是 72，需要用到所有 2^{48} 个选择明文，时间复杂度约为 $2^{95.28}$ 次 10 轮加密。进而，若同时使用 5 个不可能差分特征，时间复杂度可以降为约 $2^{93.59}$ 次 10 轮加密。

5 结束语

目前对 SPECK 系列算法的分析主要包括差分分析、线性分析、零相关线性分析等。为了评估 SPECK 算法抵抗不可能差分分析的能力，文献[17]利用线性规划方法找到了 SPECK 64 算法的一些 6 轮不可能差分特征，本文直接利用模加法运算的差分扩散性质，找到了 SPECK 32/64 和 SPECK48/96 算法的 6 轮不可能差分特征，并给出了对这两个算法的 10 轮不可能差分分析。攻击中用到的最长不可能差分特征的轮数与文献[15]和文献[16]找到的最长零相关特征的轮数一致。而对更长分组长度的 SPECK 算法，我们目前能构造的不可能差分特征也不超过 6 轮。这些初步分析表明 SPECK 算法具有较强的抵抗不可能差分分析的能力，下一步我们将进一步考虑对更长分组长度的 SPECK 算法的不可能差分分析，并探索 ARX 结构分组密码算法不可能差分分析和零相关线性分析间的联系。

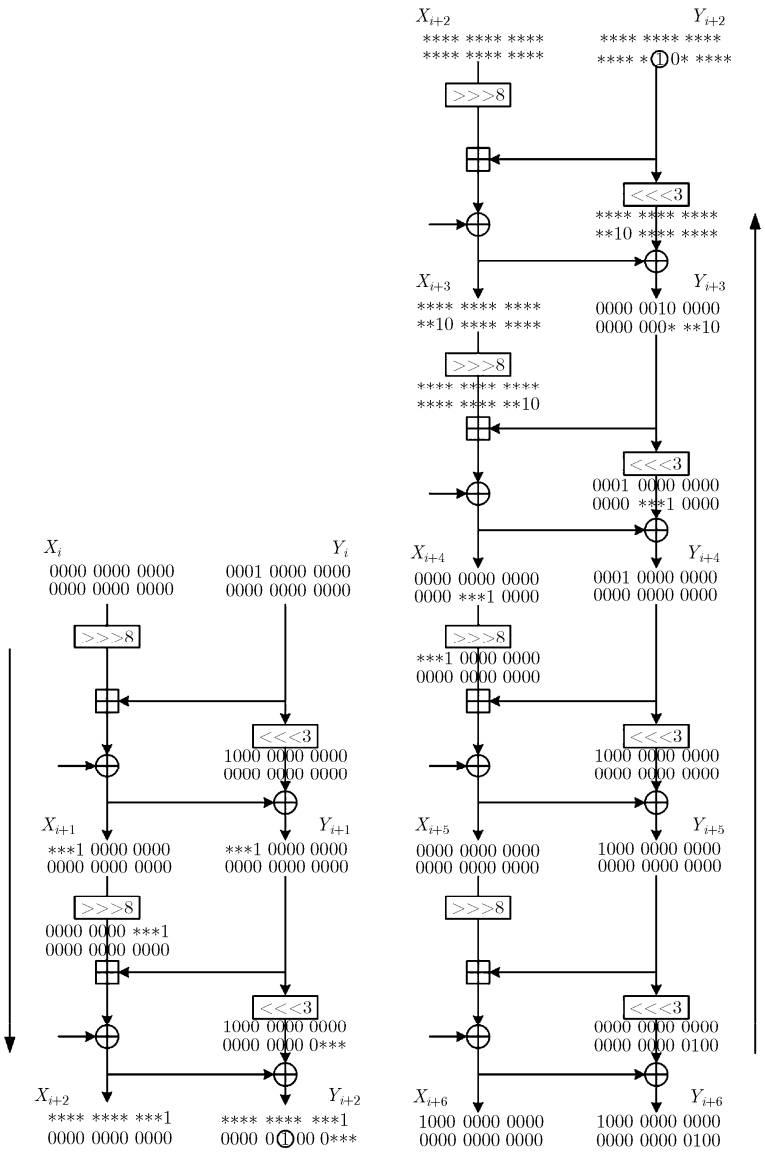

图 4 SPECK 48/96 算法的 6 轮不可能差分特征

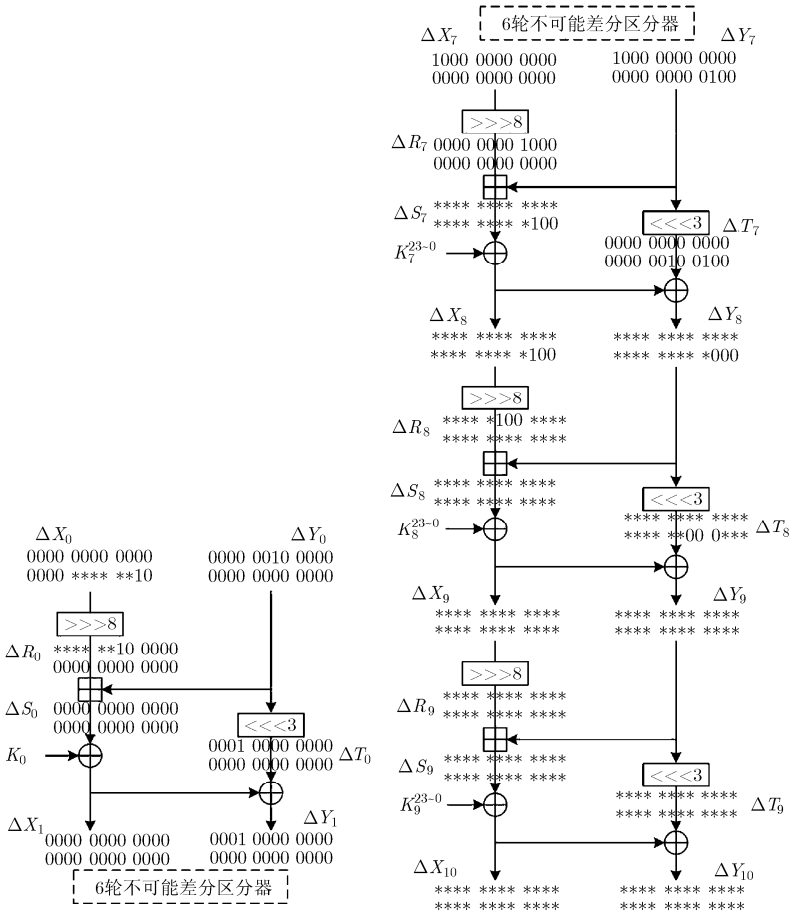

图 5　SPECK 48/96 算法的 10 轮不可能差分分析

参 考 文 献

[1] KNUDSEN L. DEAL — A 128-bit block cipher[R]. Department of Informatics, University of Bergen, Norway, 1998.

[2] BIHAM E, BIRYUKOV A, and SHAMIR A. Cryptanalysis of skipjack reduced to 31 rounds using impossible differentials [J]. *Lecture Notes in Computer Science*, 1999, 1592: 12–23. doi: 10.1007/3-540-48910-X_2.

[3] LU J, KELLER N, and KIM J. New impossible differential attacks on AES[J]. *Lecture Notes in Computer Science*, 2008, 5365: 279–293. doi: 10.1007/978-3-540-89754-5_22.

[4] ZHANG Wentao, WU Wenling, and FENG Dengguo. New results on impossible differential cryptanalysis of reduced AES[J]. *Lecture Notes in Computer Science*, 2007, 4817: 239–250. doi: 10.1007/978-3-540-76788-6_19.

[5] LU J, KIM J, and KELLER N. Improving the efficiency of impossible differential cryptanalysis of reduced Camellia and MISTY1[J]. *Lecture Notes in Computer Science*, 2008, 4964: 370–386. doi: 10.1007/978-3-540-79263-5_24.

[6] WU Wenling, ZHANG Wentao, and FENG Dengguo. Impossible differential cryptanalysis of reduced-round ARIA and Camellia[J]. *Journal of Computer Science and Technology*, 2007, 22(3): 449–456. doi: 10.1007/s11390-007-9056-0.

[7] BEAULIEU R, SHORS D, SMITH J, *et al*. The SIMON and SPECK families of lightweight block ciphers[OL]. http://eprint.iacr.org /2013/404,2013.

[8] ABED F, LIST E, and LUCKS S. Cryptanalysis of the SPECK family of block ciphers[OL]. http://eprint.iacr.org/2013/568, 2013.

[9] ABED F, LIST E, and LUCKS S. Differential cryptanalysis of round-reduced SIMON and SPECK[J]. *Lecture Notes in Computer Science*, 2014, 8540: 525–545. doi: 10.1007/978-3-662-46706-0_27.

[10] BIRYUKOV A, ROY A, and VELICHKOV V. Differential analysis of block ciphers SIMON and SPECK[J]. *Lecture Notes in Computer Science*, 2014, 8540: 546–570. doi: 10.1007/978-3-662-46706-0_28.

[11] DINUR I. Improved differential cryptanalysis of round-reduced SPECK[J]. *Lecture Notes in Computer Science*, 2014, 8781: 147–164. doi: 10.1007/978-3-319-13051-4_9.

[12] BIRYUKOV A, VELICHKOV V, and LE Y. Automatic

search for the best trails in ARX: Application to Block Cipher SPECK[J]. *Lecture Notes in Computer Science*, 2016, 9783: 289–310. doi: 10.1007/978-3-662-52993-5_15.

[13] FU Kai, WANG Meiqin, and GUO Y. MILP-based automatic search algorithms for differential and linear trails for SPECK [J]. *Lecture Notes in Computer Science*, 2016, 9783: 268–288. doi: 10.1007/978-3-662-52993-5_14.

[14] YAO Yuan, ZHANG Bin, and WU Wenling. Automatic search for linear trails of the SPECK family[J]. *Lecture Notes in Computer Science*, 2015, 9290: 158–176. doi: 10.1007/978-3-319-23318-5_9.

[15] 程雨芊. 对SPECK系列分组密码算法的零相关线性分析 [D]. [硕士论文], 山东大学, 2015.
CHENG Yuqian. Zero correlation cryptanalysis of block cipher speck[D]. [Master dissertation], Shandong University, 2015.

[16] CUI Tingting, JIA Keting, FU Kai, *et al*. New automatic search tool for impossible differentials and zero-correlation linear approximations[OL]. http://eprint.iacr.org/2016/689, 2016.

[17] LEE H, KANG H, and HONG D. New impossible differential characteristic of SPECK64 using MILP[OL]. http://eprint.iacr.org/2016/1137, 2016.

徐 洪： 女，1979 年生，硕士生导师，主要研究方向为对称密码的设计与分析.

苏鹏晖： 男，1992 年生，硕士生，研究方向为分组密码的设计与分析.

戚文峰： 男，1963 年生，教授，博士生导师，主要研究方向为对称密码的设计与分析.