对缩减轮数 SM3 散列函数改进的原像与伪碰撞攻击

邹剑[1,2]，董乐[3]

（1. 福州大学数学与计算机科学学院，福建 福州 350108；
2. 福州大学网络系统信息安全重点实验室，福建 福州 350108；
3. 河南师范大学大数据统计分析与优化控制河南工程实验室，河南 新乡 453007）

摘 要：提出了对 SM3 散列函数 32 轮的原像攻击和 33 轮的伪碰撞攻击。利用差分中间相遇攻击与 biclique 技术改进了对 SM3 的原像分析结果，将攻击结果从之前的 30 轮提高到了 32 轮。基于上述方法，通过扩展 32 轮原像攻击中的差分路径，对 SM3 构造了 33 轮的伪碰撞攻击。以 $2^{254.5}$ 的时间复杂度与 2^5 的空间复杂度构造了对 SM3 的 32 轮原像攻击，并以 $2^{126.7}$ 的时间复杂度与 2^3 的空间复杂度构造了对 SM3 的 33 轮伪碰撞攻击。

关键词：SM3 散列函数；原像攻击；伪碰撞攻击；差分中间相遇攻击；biclique

中图分类号：TP309

文献标识码：A

doi: 10.11959/j.issn.1000-436x.2018011

Improved preimage and pseudo-collision attacks on SM3 hash function

ZOU Jian[1, 2], DONG Le[3]

1. College of Mathematics and Computer Science, Fuzhou University, Fuzhou 350108, China
2. Key Lab of Information Security of Network Systems, Fuzhou University, Fuzhou 350108, China
3. Henan Engineering Laboratory for Big Data Statistical Analysis and Optimal Control, Henan Normal University, Xinxiang 453007, China

Abstract: A preimage attack on 32-step SM3 hash function and a pseudo-collision attack on 33-step SM3 hash function respectively were shown. 32-step preimage attack was based on the differential meet-in-the-middle and biclique technique, while the previously known best preimage attack on SM3 was only 30-step. The 33-step pseudo-collision attack was constructed by using the same techniques. The preimage attack on 32-step SM3 can be computed with a complexity of $2^{254.5}$, and a memory of 2^5. Furthermore, The pseudo-preimage and pseudo-collision attacks on 33-step SM3 by extending the differential characteristic of the 32-step preimage attack were present. The pseudo-collision attack on 33-step SM3 can be computed with a complexity of $2^{126.7}$, and a memory of 2^3.

Key words: SM3 hash function, preimage attack, pseudo-collision attack, differential meet-in-the-middle, biclique

1 引言

散列函数在密码学中扮演着重要的角色，被广泛地应用于消息认证等密码应用中。一般来说，散列函数必须满足 3 种安全目标：抗原像攻击、抗第二原像攻击和抗碰撞攻击。

随着 SHA-3 竞赛的展开，各种新型攻击方法不断涌现，如中间相遇攻击、反弹攻击等。目前，中间相遇攻击已经被广泛地用于求解散列函数的原像值，并已对多个散列函数取得了有效的分析结果，如 MD5[1]、Tiger[2]，以及 SHA-0 和 SHA-1[3]。在 FSE2012 上，Dmitry 等[4]提出了 biclique 方法来改进中间相遇攻击，并对 SHA-256 散列算法构造了 52 轮的原像攻击。同年，Li 等[5]也通过转化中间相

收稿日期：2017-03-09；修回日期：2017-11-13

基金项目：福建省中青年教师教育科研基金资助项目（No.JAT170097）；福州大学科研启动基金资助项目（No.510150）

Foundation Items: The Education and Research Projects for Young Teachers in Fujian Province (No.JAT170097), The Research Startup Project of Fuzhou University (No.510150)

遇攻击，对散列函数构造了相应的伪碰撞攻击。随后，Knellwolf 等[6]在 Crypto2012 上利用差分技术改进了中间相遇攻击，并给出了对 SHA-1 散列算法 57 轮的原像攻击。

SM3 是由王小云等自主设计的散列函数，它于 2010 年被选为中国商用散列函数标准。SM3 的总体设计方案与 SHA-256 类似。不过 SM3 采用了更复杂的轮函数，因此，SM3 比 SHA-256 更能抵抗目前已知的攻击。

本文利用差分中间相遇攻击等方法改进了对 SM3 的原像攻击，把攻击结果由之前的 30 轮[7~9]提高到 32 轮，并利用对 SM3 的伪原像攻击构造了对 SM3 的 33 轮伪碰撞，如表 1 所示。

2　SM3 散列函数介绍

SM3 采用大端设计，并按如下方式生成散列值。

$$\begin{cases} V_0 \leftarrow IV \\ V_{i+1} \leftarrow CF(V_i, M^i), i = 0,1,\cdots,p \end{cases}$$

在计算压缩函数 CF 之前，首先对输入消息 M 进行填充，使其变为 512 bit 的整数倍。具体过程如下，先在消息后添加比特“1”，再添加 len_0 个比特“0”，使 $len_0+len_M+1\equiv 448 \bmod 512$（$len_M$ 为输入消息的比特长度），最后再添加 64 bit 来表示 len_M。然后将填充后的消息 M^* 分割成 512 bit 的消息块 $M^i(i=0,1,\cdots,p)$，作为压缩函数的输入。

SM3 的压缩函数 $V_{i+1}\leftarrow CF(V_i, M^i)$ 采用 DM 模式，即 $CF = F(V_i, M^i)\oplus V_i$，其中，$F$ 可以看成分组密码。SM3 压缩函数的操作步骤具体如下。

1) 将输入消息 M^j 分割为 16 块 $M_i(i=1,2,\cdots,16)$，其中，每块消息字 M_i 长为 32 bit。再按以下方式将 $M_i(i=1,2,\cdots,16)$ 扩展成 68 块 32 bit 的消息字 W_i 与 64 块 32 bit 的消息字 W_i'：$W_i = M_i\ (1\leqslant i\leqslant 16)$，$W_i = P_1(W_{i-16}\oplus W_{i-9}\oplus (W_{i-3}<<<15))\oplus(W_{i-13}<<<7)\oplus W_{i-6}, (17\leqslant i\leqslant 68)$, $W_i' = W_i\oplus W_{i+4}\ (1\leqslant i\leqslant 64)$。其中，$P_1(x) = x\oplus(x<<<15)\oplus(x<<<23)$, $P_1^{-1}(x) = x\oplus(x<<<5)\oplus(x<<<13)\oplus(x<<<14)\oplus(x<<<15)\oplus(x<<<21)\oplus(x<<<23)\oplus(x<<<29)\oplus(x<<<30)$。

2) $S_1\leftarrow V_i$，其中，$S_i = (A_i,\cdots,H_i)$。

3) 连续 64 次调用图 1 中的步函数来更新状态 S_i，而后输出 $V_{i+1}\leftarrow S_1\oplus S_{65}$。

在图 1 中，SM3 所采用的函数分别为

$$P_0(x) = x\oplus(x<<<9)\oplus(x<<<17),$$

$$\begin{cases} FF_i(x,y,z) = x\oplus y\oplus z, \\ GG_i(x,y,z) = x\oplus y\oplus z, \end{cases} 1\leqslant i\leqslant 16$$

$$\begin{cases} FF_i(x,y,x) = (x\wedge y)\vee(x\wedge z)\vee(y\wedge z), \\ GG_i(x,y,x) = (x\wedge y)\vee(\neg x\wedge z), \end{cases} 17\leqslant i\leqslant 64$$

其中，FF_i 与 GG_i 这 2 个布尔函数在前 16 轮与后 48 轮的定义是不同的。

3　预备知识

3.1　切割缝合技术与 biclique 攻击方法

切割缝合技术与初始化结构是由 Aoki 等[10]在攻击 MD4 时提出的。切割缝合技术（如图 2 所示）是将压缩函数 CF 分割为 $CF=F_1F_3F_2$，使攻击者可以从函数中选取一个中间状态作为计算起始点，并通过反馈操作连接压缩函数的初始值与目标值。

表 1　攻击结果对比

攻击目标及轮数	攻击类型	时间复杂度	存储复杂度	文献
30 轮 SM3（从第 7 轮开始）	原像攻击	2^{249}	2^{16}	文献[7]
30 轮 SM3（从第 1 轮开始）	原像攻击	$2^{251.1}$	2^{6}	文献[8]
31 轮 SM3（从第 2 轮开始）	原像攻击	$2^{252.4}$	2^{10}	本文
32 轮 SM3（从第 4 轮开始）	原像攻击	$2^{254.5}$	2^{5}	本文
20 轮 SM3（从第 1 轮开始）	碰撞攻击	实际可行	实际可行	文献[9]
24 轮 SM3（从第 1 轮开始）	伪碰撞攻击	实际可行	实际可行	文献[9]
32 轮 SM3（从第 1 轮开始）	伪碰撞攻击	$2^{125.1}$	2^{6}	文献[8]
33 轮 SM3（从第 3 轮开始）	伪碰撞攻击	$2^{126.7}$	2^{3}	本文

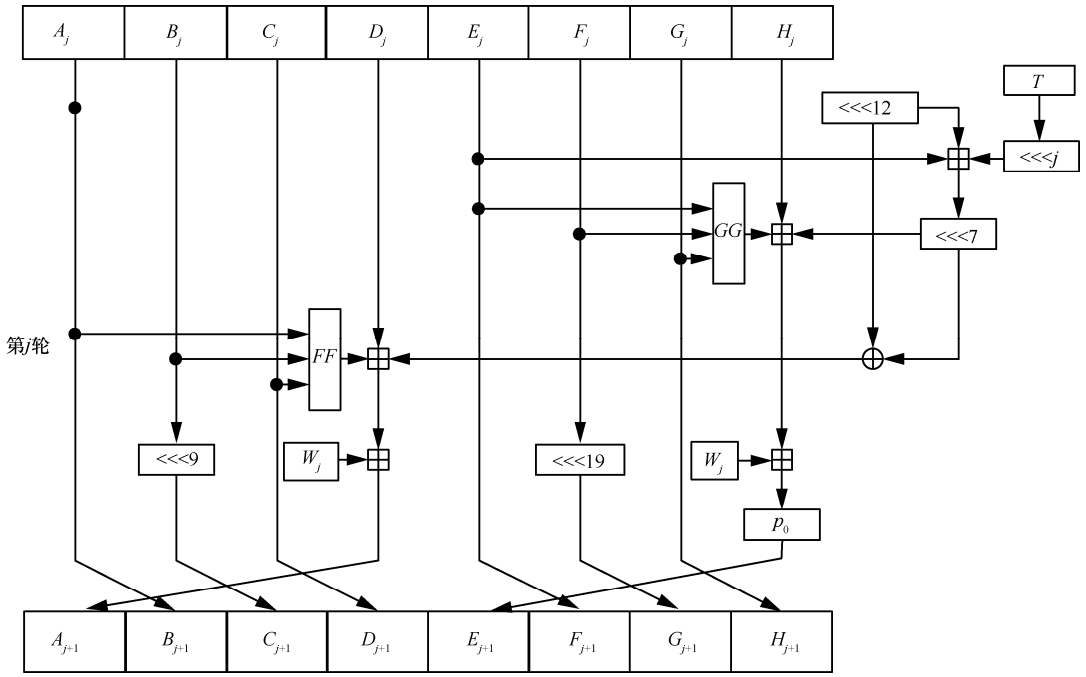


图 1　SM3 步函数

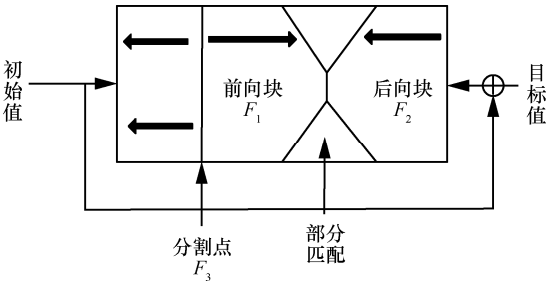


图 2　切割缝合技术

初始化结构允许攻击者通过交换分割点附近多轮步函数（不妨记为 F_3）中的消息中立字，以便增加最终的攻击轮数。Biclique 方法[4]（如图 3 所示）是对初始化结构的形式化表示，它可以看成是对 F_3 构造了一个五元组 $\{M,D_1,D_2,Q_1,Q_2\}$，其中，M 是输入消息，D_1 和 D_2 是维度为 d 的线性空间，记 Q_1 为包含前向块 2^d 个起始状态 $Q_1[\delta_{1i}]$ $(\delta_{1i}\in D_1)$ 的列表，而 Q_2 为包含后向块 2^d 个起始状态 $Q_2[\delta_{2j}](\delta_{2j}\in D_2)$ 的列表。对于所有的 $(\delta_{1i},\delta_{2j})\in D_1\times D_2$，$Q_1$ 与 Q_2 要满足 $Q_2[\delta_{2j}]=F_3(M\oplus\delta_1\oplus\delta_2,Q_1[\delta_{1i}])$。如果线性空间 D_1 和 D_2 的维度都是 d，则 $D_1\times D_2$ 共有 2^{2d} 候选值，本文称其为构造了一个 d 维的 biclique。注意到，biclique 的构造需要先调用前向块 F_1 函数 2^d 次来计算 2^d 个起始状态 $Q_1[\delta_{1i}]$，然后再调用后向块 F_2 函数 2^d 次来计算 2^d 个起始状态 $Q_2[\delta_{2j}]$。因此，本文需时间复杂度为 2^d 与空间复杂度为 2^d 来构造一个 d 维的 biclique。由于敌手可以通过预计算的方式计算 biclique 结构，并且构造一个（或多个）d 维的 biclique 的时间代价与求解伪原像的时间代价相比几乎可忽略，因此，在文献[1~4,6~8,10]，一般都假设构造 biclique（或称初始化结构）所需的时间代价可忽略。

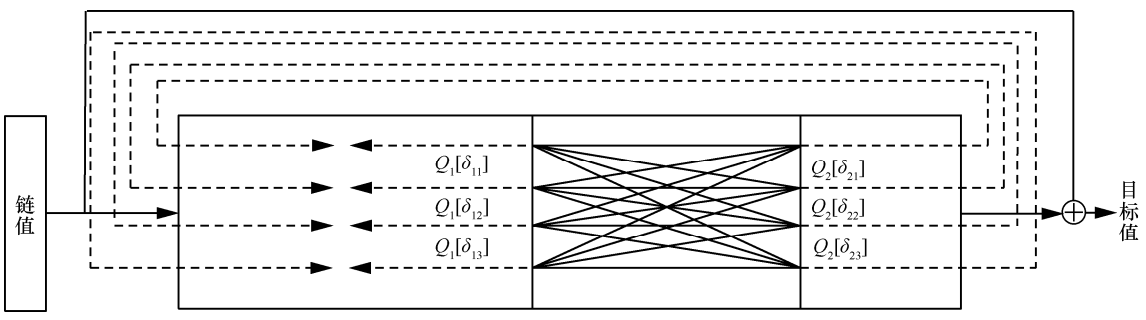


图 3　2 维 biclique

3.2　差分中间相遇攻击

本节将介绍如何将 biclique 技术与差分中间相遇攻击相结合，并给出具体的原像攻击算法 1。假设压缩函数 $CF:V=F(IV,M)\oplus IV$ 采用了 DM 模式。对于底层分组密码 $F:\{0,1\}^{\kappa}\times\{0,1\}^{n}\to\{0,1\}^{n}$ 的 biclique 差分中间相遇攻击的过程如下。

1) 先将函数 F 分割为 $F=F_1\circ F_3\circ F_2$，其中，biclique 技术将被应用于 F_3。再对 F_1 与 F_2 选取相应的线性空间 D_1 和 D_2，满足 $D_1\bigcap D_2=\{0\}$。

2) 对于所有的 $(\delta_{1i},\delta_{2j})\in D_1\times D_2$，计算 $Q_2[\delta_{2j}]=F_3(M\oplus\delta_1\oplus\delta_2,Q_1[\delta_{1i}])$。

3) 随机选取一个输入消息 M，对每个 $\delta_{1i}\in D_1$，寻找一个 Δ_1 使 $p_1=\Pr[\Delta_1=_T F_1(M,\ Q_2[\delta_{2j}])\oplus F_1(M\oplus\delta_{1i},\ Q_2[\delta_{2j}])]$的成立概率尽量大，。

4) 对于上述输入消息M，对每个$\delta_{2j}\in D_2$，也寻找一个 Δ_2 使 $p_2=\Pr[\Delta_2=_T F_2^{-1}(M,Q_1[\delta_{1i}])\oplus F_2^{-1}(M\oplus\delta_2,Q_1[\delta_{1i}])]$ 的成立概率尽量大。

5) 使用算法 1 来求解一个原像。

利用输入消息不同的初始赋值，不断重复步骤 3)和步骤 4)，直到找一个全状态的匹配。上述 5 个步骤提供了一个求解给定压缩函数算法伪原像的方法。需要注意，给定 $T\in\{0,1\}^{n}$，等式 $X=_T Y$ 指 $T\wedge(X\oplus Y)=0$，其中，$\wedge$是指比特级“与”运算。

算法 1　检测集合 $M\oplus D_1\oplus D_2$ 中是否包含原像解

输入　$D_1,D_2\subset\{0,1\}^{\kappa},M\in\{0,1\}^{\kappa},T\in\{0,1\}^{n}$

输出　若集合 $M\oplus D_1\oplus D_2$ 中包含原像解，则输出

for $\delta_{2j}\in D_2$ do
$L_1[\delta_{2j}]=F_1(M\oplus\delta_{2j},Q_2[\delta_{2j}])\oplus\Delta_2$;
end for
for $\delta_{1i}\in D_1$ do
$L_2[\delta_{1i}]=F_2^{-1}(M\oplus\delta_{1i},Q_1[\delta_{1i}])\oplus\Delta_1\oplus V$;
end for
for all $(\delta_{1i},\delta_{2j})\in D_1\times D_2$ do
if $L_1[\delta_{2j}]=_T L_2[\delta_{1i}]$ then
返回 $M\oplus\delta_{1i}\oplus\delta_{2j}$;
end if
end for
return no preimage in $M\oplus D_1\oplus D_2$

假设线性空间 D_1 和 D_2 的维度都是 d，则敌手通过算法 1 在集合 $M\oplus D_1\oplus D_2$ 中可以搜索 2^{2d} 不同消息。由于概率 p_1 和 p_2 都小于 1，则一个目标原像解是正确的概率为 p_1p_2。因此，使用上述差分中间相遇攻击方法，敌手找到一个原像所需时间复杂度为 $\dfrac{2^{n-d}\Gamma+2^{n-r}\Gamma_{re}}{p_1p_2}$，其中，$\Gamma$ 与 Γ_{re} 分别为计算 F 和测试原像所需的时间，r 是 T 的汉明重量，而 d 是线性空间 D_1 和 D_2 的维度。更多有关差分中间相遇攻击的信息见文献[6]。

3.3　将伪原像攻击转化为原像攻击的一般方法

与原像攻击不同，伪原像攻击只需要找到一组满足 $CF(x,M)=y$ 的解 (x,M)，其中，y 为给定的目标值，而 x 不需等于预先的初始链值。目前已有文献[11]给出了可以将伪原像攻击转化为原像攻击的通用算法。假设敌手能以 2^k 的时间复杂度对目标算法构造一个伪原像攻击，则由上述转化算法敌手能以 $2^{\frac{n+k}{2}+1}$ 的时间复杂度构造对应的原像攻击。

3.4　中间相遇攻击转化为伪碰撞攻击的一般方法

在 FSE 2012 上，Li 等[5]提出了一种能将中间相遇攻击转化为伪碰撞攻击的方法。假设 Oracle A 可以用 2^s 的时间复杂度找到目标算法 t 比特的部分原像，则敌手可以通过调用上述 Oracle A 约 $2^{\frac{n-t}{2}}$ 次，能找到 $2^{\frac{n-t}{2}}$ 个$(n-t)$ bit 的随机数据。依据生日攻击可知，上述数据以很大概率存在一个碰撞。因此，敌手能以 $2^s\times2^{\frac{n-t}{2}}$ 的时间复杂度将 t bit 的中间相遇攻击转化为伪碰撞攻击。特别需要说明的，在文献[5]算法中，敌手需要注意以下几点：1) 保证匹配点在最后；2) 因为生日悖论，对于碰撞攻击的一般化攻击的时间复杂度为 $2^{\frac{n}{2}}$，而上述转化方法所需的时间复杂度为 $2^s\times2^{\frac{n-t}{2}}$，因此，只有当 $2^s\times2^{\frac{n-t}{2}}<2^{\frac{n}{2}}$ 时，本文认为上述攻击是有效的。

4　对缩减轮 SM3 的伪原像与伪碰撞攻击

本文将对 SM3 散列函数构造多个改进的原像攻击与伪碰撞攻击。首先，本文将展示如何对 SM3 构造 31 轮原像攻击与伪碰撞攻击；其次，通过将 biclique 增加一轮，本文能将 31 轮原像攻击与伪碰撞攻击扩展到 32 轮；最后，通过扩展后向块差分路径，进一步将 32 轮伪碰撞攻击扩展到了 33 轮。

在本文原像与伪碰撞攻击中，本文将采用文献[8]提到的性质 1 与部分符号记法。

1) Δa：在字 a 上有差分。

2) $\Delta a=[b-c]$：字 a 上从第 b bit 到第 c bit 有差分，而在其他比特上无差分。

3) $\Delta a=\langle b-c\rangle$：字 a 上从第 b bit 到第 c bit 的差分值已知，而其他比特上的差分值未知。

4) $\Delta a=?$：字 a 上的差分值未知。

其中，Δa 中 a 表示 32 bit 消息字。

性质 1 给定 3 个 n bit 输入串 x，y，z，其中，$z=x+y$。若设 $x=x_{n-1}\cdots x_0$, $y=y_{n-1}\cdots y_0$, $z=z_{n-1}\cdots z_0$，再令 $x'=x_{n-1}\cdots x_s0\cdots0$, $y'=y_{n-1}\cdots y_s0\cdots0$, $z'=x'+y'=z'_{n-1}\cdots z'_0$。假设敌手已知输入串 x 的值以及输入串 y 在比特 y_i ($i=s,\cdots,n-1$)上的值，但不知道输入串 y 在比特 y_i ($i=0,\cdots,s-1$)上的值。则若有 $n-t\geqslant s$，那么 $z_i=z'_i$ ($i=n-t,\cdots,n-1$)成立的概率 $p>1-2^{t+s-n}$。

性质 1 的证明过程见文献[8]。

4.1 对 31 轮 SM3 的原像与伪碰撞攻击

本文首先说明如何利用 biclique 技术与差分中间相遇攻击对 SM3 构造 31 轮的伪原像攻击。而后可以利用第 3.3 节与第 3.4 节中的算法将上述伪原像攻击转化为原像攻击与伪碰撞攻击。为了应用差分中间相遇攻击，首先需要将底层分组密码 F，分割为 $F=F_1\circ F_3\circ F_2$，其中，F_1 是前向块（从第 18 轮到第 32 轮），F_2 是后向块（从第 15 轮到第 2 轮），而 F_3 对应到 biclique 结构（从第 16 轮到第 17 轮）。本文的伪原像攻击以 2 轮 biclique 为计算起始点。

本文在差分中间相遇攻击中，首先要为前向块 F_1 与后向块 F_2 选取对应的线性空间 D_1 和 D_2。为了使本文攻击成立且时间复杂度最优，线性空间 D_1 和 D_2 必须满足以下 3 个条件：1) 线性空间 D_1 和 D_2 的维度 d 要尽可能大；2) 在线性空间 D_1 和 D_2 引入差分时，需要保证在输入消息 M 最后 65 bit 的差分值为 0；3) 在线性空间 D_1 和 D_2 引入的差分，在构造的 biclique 中不会互相影响。条件 1)能保证本文伪原像攻击的时间复杂度相对较优；条件 2)能保证本文所求解的输入消息 $M\oplus\delta_{1i}\oplus\delta_{2j}$ 能满足 SM3 的消息填充规则；条件 3)能保证 biclique 成立。经过测试，本文发现按照以下方式选取线性空间 D_1 和 D_2 与 T，可以同时满足条件 1)、条件 2)与条件 3)。

$D_1=\{w_{17}\|w_{18}\|\cdots\|w_{32}\,|\,w_{17}=[7-9,13-19],w_i=0,18\leqslant i\leqslant32\}$，$D_2=\{w_2\|\cdots\|w_{17}\,|\,w_4=[22-31],w_i=0,2\leqslant i\leqslant3,5\leqslant i\leqslant17\}$，$T=\{0,0,0,0,0,0,0,003\text{F}8007\}$，则 $d=r=10$。依据 SM3 的消息扩展算法，线性空间 D_1 引入的差分不会对输入消息的后 65 bit 有影响，即有 $\Delta W_{16}=\Delta W_{15}=0$，$\Delta W_{14}=P_1^{-1}(\Delta W_{17}<<<7)=[1-31]$。类似地，可以知道线性空间 D_2 引入的差分，也不会对输入消息的后 65 bit 有影响，即有 $\Delta W_{16}=0$, 以及 $\Delta W_{15}=\Delta W_{14}=0$。这说明上述取值满足条件 2。

在 biclique 中要防止前（后）向块引入的差分互相影响。线性空间 D_1 引入的差分对消息的影响为 $\Delta W_{17}=\Delta W'_{17}=[7-9,\ 13-19]$, $\Delta W_j=\Delta W'_j=0$ ($j=15,16$)。类似地，线性空间 D_2 引入的差分对于消息的影响为 $\Delta W_i=0$ ($2\leqslant i\leqslant3,5\leqslant i\leqslant17$), $\Delta W_4=[22-31]$, 再由 SM3 的消息扩展算法可得 $\Delta W'_{16}=P_1(\Delta W_4)$。如图 4 所示，为了构造对 SM3 的 2 轮 biclique，要利用状态 $(C_{16},D_{16},G_{16},H_{16})$ 的自由度，使被前向块影响的 $\Delta W'_{16}$ 和受后向块影响的 ΔW_{17} 与 $\Delta W'_{17}$ 能互不影响。具体构造方法如图 4 所示，令 $Q_1[0]=(A_{16},B_{16},C_{16},D_{16},E_{16},F_{16},G_{16},H_{16})$, 则对于消息 $M\oplus\delta_{2j}$，有第 17 轮的输出状态为 $Q_2[\delta_{2j}]=(A_{18}^{\delta_{2j}},B_{18}^{\delta_{2j}},C_{18},D_{18},E_{18}^{\delta_{2j}},F_{18},G_{18},H_{18})$, 其中 $A_{18}^{\delta_{2j}}\neq A_{18}$, $B_{18}^{\delta_{2j}}\neq B_{18}$, $E_{18}^{\delta_{2j}}\neq E_{18}$。对于消息 $M\oplus\delta_{1i}$ 与 $Q_2[0]=(A_{18},B_{18},C_{18},D_{18},E_{18},F_{18},G_{18},H_{18})$, 第 16 轮输入为 $Q_1[\delta_{1i}]=(A_{16},B_{16},C_{16}^{\delta_{1i}},D_{16}^{\delta_{1i}},E_{16},F_{16},G_{16}^{\delta_{1i}},H_{16}^{\delta_{1i}})$，其中，$C_{16}^{\delta_{1i}}\neq C_{16}$, $D_{16}^{\delta_{1i}}\neq D_{16}$, $G_{16}^{\delta_{1i}}\neq G_{16}$, $H_{16}^{\delta_{1i}}\neq H_{16}$。依据 SM3 的步函数，有对于输入消息 $M\oplus\delta_{1i}\oplus\delta_{2j}$ 与状态 $Q_1[\delta_{1i}]$，第 17 轮的输出为 $Q_2[\delta_{2j}]$。这意味着已经成功地构造了 2 轮 biclique。

由表 2 可知，前向块 F_1 的差分路径成立概率 $\Pr[0=_T F_1(M,S_{18})\oplus F_1(M\oplus\delta_1,S_{18})]=1$。而由表 3 和性质 1 可知，后向块 F_2 的差分路径成立概率满足 $\Pr[0=_T F_2^{-1}(M,S_{16})\oplus F_2^{-1}(M\oplus\delta_2,S_{16})]=0.75$。因此，本文需要 $\dfrac{2^{256-10}F+2^{256-10}F_{re}}{1\times0.75}\approx2^{246.8}$ 的时间复杂度与 2^{10} 的存储复杂度对 SM3 构造 31 轮伪原像攻击。注意，$F_{re}=\dfrac{(31-28)F}{31}=\dfrac{3F}{31}$，这是因为前向块的 15 轮，后向块的 11 轮与 2 轮 biclique 在测试目标原像时不需要重新计算。

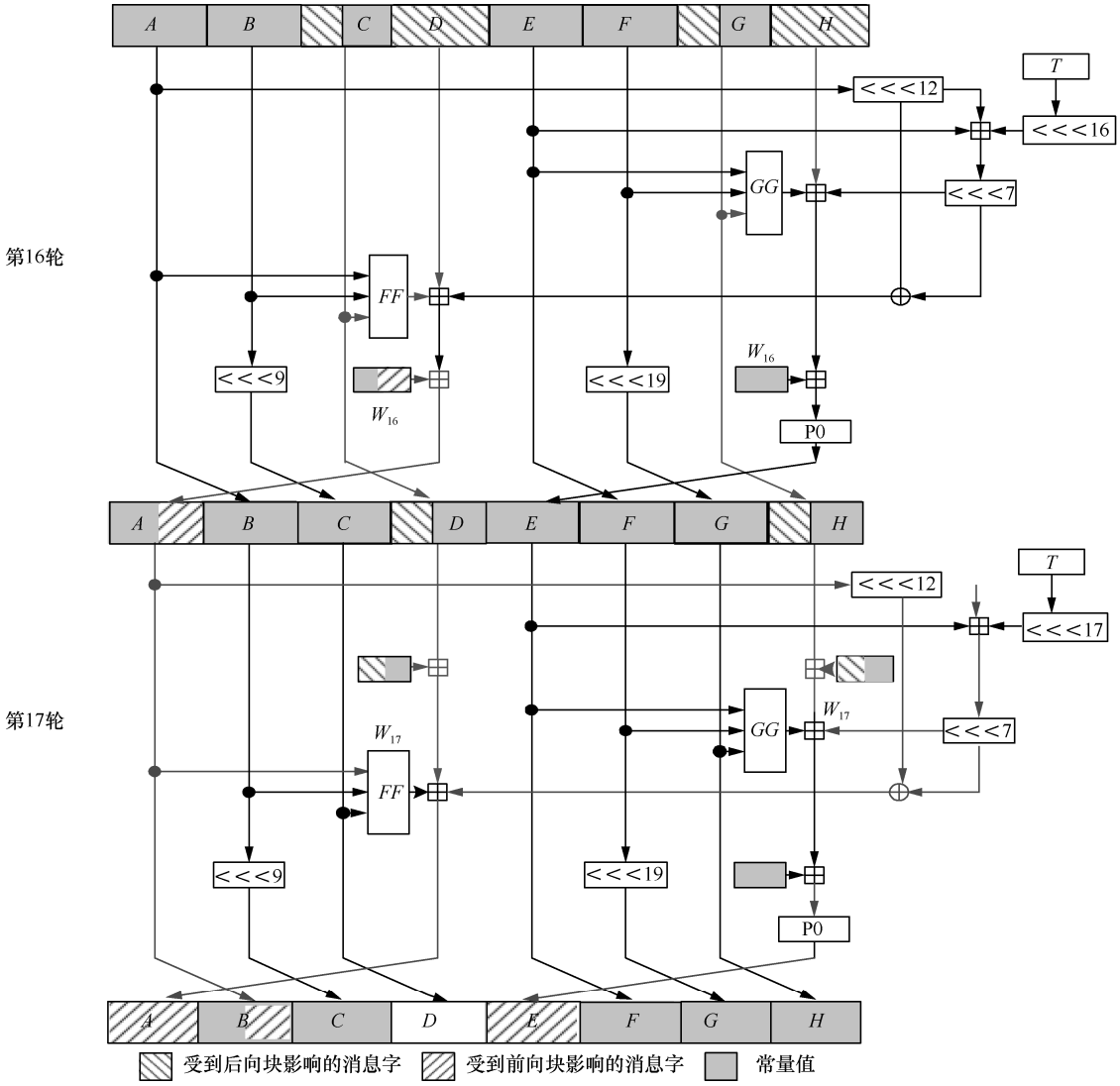


图 4　2 轮 biclique

表 2　31 轮伪原像攻击中对 F_1 的差分特征路径

差分	步函数						
	18	…	28	29	30	31	32
ΔW	0	…	0	0	0	0	0
$\Delta W'$	0	…	0	Δ_1	0	0	?
ΔA	0	…	0	?	?	?	?
ΔB	0	…	0	0	?	?	?
ΔC	0	…	0	0	0	?	?
ΔD	0	…	0	0	0	0	?
ΔE	0	…	0	0	?	?	?
ΔF	0	…	0	0	0	?	?
ΔG	0	…	0	0	0	0	?
ΔH	0	…	0	0	0	0	0
概率	1	…	1	1	1	1	1

其中，$\Delta_1 = P_1(x_{17}) = P_1([7-9,13-19])$。

因为，对 31 轮 SM3 构造伪原像所需的时间复杂度是 $2^{246.8}$，所以按照第 3.3 节的转化算法，对 31 轮 SM3 构造原像攻击所需的时间复杂度为 $2^{\frac{256+246.8}{2}+1} \approx 2^{252.4}$，以及 2^{10} 的存储复杂度。又因为在上述伪原像中间相遇攻击中，匹配点是在压缩函数最后，因此，本文可以利用第 3.4 节中的转化算法将其转化为对 SM3 散列函数 31 轮的伪碰撞攻击，依据转化算法可以知，其需要 $\frac{2^{\frac{256-10}{2}}}{0.75} \approx 2^{123.4}$ 的时间复杂度与 2^{10} 的存储复杂度。

4.2　对 32 轮 SM3 的原像与伪碰撞攻击

在本节中，将通过构造 3 轮 biclique，将上述 31 轮伪原像攻击扩展到 32 轮。文献[8]通过利用 SM3 后 48 轮布尔函数，即 $17 \leqslant i \leqslant 64$，$FF_i(x,y,z)=(x \wedge y) \vee (x \wedge z) \vee (y \wedge z)$，$GG_i(x,y,z)=(x \wedge y) \vee (\neg x \wedge z)$

的吸收性质对 SM3 构造了 3 轮初始化结构。这里的吸收性质是指，对于布尔函数 $FF_i(x,y,z)=(x\wedge y)\vee(x\wedge z)\vee(y\wedge z),(17\leqslant i\leqslant 64)$，若有 $x=y$，则 $FF_i(x,y,z)=x=y(17\leqslant i\leqslant 64)$。即输入 z 对于 $FF_i(x,y,z)\ (17\leqslant i\leqslant 64)$ 的输出没有任何影响，相当于输入 z 的作用被完全吸收了。对 $GG_i(x,y,z)=(x\wedge y)\vee(\neg x\wedge z),17\leqslant i\leqslant 64$，本文有类似的结论，即若输入 x 的 32 bit 全为 1，则 $GG_i(x,y,z)=y$，这说明 z 的作用被完全吸收了。而若 x 的全 32 bit 都为 0，则 $GG_i(x,y,z)=z$，这说明 y 的作用被完全吸收了。本节将采用类似的思路对 SM3 算法构造 3 轮 biclique。因此，若设 3 轮 biclique 是从第 j 轮到第 j+2 轮，则为了利用对应布尔函数的吸收性质，本文要保证 $j>16$。

表 3　31 轮伪原像攻击中对 F_2^{-1} 的差分特征路径

差分	步函数					
	15	…	5	4	3	2
ΔW	0	…	0	Δ_2	0	0
$\Delta W'$	0	…	0	Δ_2	0	0
ΔA	0	…	0	0	0	0
ΔB	0	…	0	0	0	Δ_4
ΔC	0	…	0	0	Δ_3	Δ_3
ΔD	0	…	0	Δ_3	Δ_3	Δ_4
ΔE	0	…	0	0	0	0
ΔF	0	…	0	0	0	Δ_5
ΔG	0	…	0	0	Δ_3	Δ_3
ΔH	0	…	0	Δ_3	Δ_3	Δ_6
概率	1	…	1	1	1	$1-2^{-2}$

其中，$\Delta_2=[22-31],\Delta_3=\langle 0-21\rangle,\Delta_4=\langle 0-12\rangle$，$\Delta_5=\langle 0-2,13-31\rangle$，$\Delta_6=\langle 0-2,15-21\rangle$。

与第 4.1 节的攻击过程类似，为了应用差分中间相遇攻击，本文首先需要将 SM3 的底层分组密码 F，分割为 $F=F_1\circ F_3\circ F_2$，其中，F_1 是前向块（从第 21 轮到第 35 轮），F_2 是后向块（从第 17 轮到第 4 轮），而 F_3 对应到 biclique 结构（从第 18 轮到第 20 轮）。本文的伪原像攻击以 3 轮 biclique 为计算起始点。为了构造 3 轮 biclique，本文在差分中间相遇攻击中首先要为前向块 F_1 与后向块 F_2 选取对应的线性空间 D_1 和 D_2。线性空间 D_1 和 D_2 的选取必须满足在第 4.1 节中的 3 个条件，以使攻击成立且时间复杂度最优。本文经测试发现，按以下方式选取线性空间 D_1 和 D_2 与 T，可同时满足这 3 个条件。$D_1=\{w_{20}\|\cdots\|w_{35}\mid w_{20}=P_1([0-1,7-9])>>>7,w_i=0,21\leqslant i\leqslant 35\}$，$D_2=\{w_4\|\cdots\|w_{19}\mid w_6=[27-31],w_i=0,4\leqslant i\leqslant 5,7\leqslant i\leqslant 19\}$，$T=\{0,0,0,0,0,0,0,1f\}$，则 $d=r=5$。由 SM3 的消息扩展算法，空间 D_1 引入的差分对于消息的影响为 $\Delta W_{16}=0,\Delta W_{15}=0$，$\Delta W_{17}=P_1^{-1}(\Delta W_{20}<<<7)=[0-1,7-9]$，$\Delta W_{14}=P_1^{-1}(\Delta W_{17}<<<7)=[3-8,11-17,19-23,27-31]$。类似地，本文对于线性空间 D_2 可得 $\Delta W_{16}=0,$ 以及 $\Delta W_{15}=\Delta W_{14}=0$。因此，线性空间 D_1 和 D_2 不会在输入消息的最后 65 bit 上引入差分。

在 biclique 中要防止前向块与后向块引入的差分互相影响。由 SM3 的消息扩展算法，线性空间 D_1 引入的差分对于 3 轮 biclique 中消息的影响如下。$\Delta W_{20}=\Delta W'_{20}=P_1[0-1,7-9]>>>7=[0-2,8-9,15-17,23-26]$，$\Delta W_j=\Delta W'_j=0\ (j=18,19)$。对于线性空间 D_2，本文有类似的结论，即 $\Delta W_j=\Delta W'_j=0\ (j=19,20)$，$\Delta W'_{18}=P_1([27-31])=[10-14,18-22,27-31]$。因此，线性空间 D_1 和 D_2 在 3 轮 biclique 消息中所引入的差分，在比特位置上并不会重合。

为了构造 3 轮 biclique，本文需要利用状态 S_{19} 的自由度，如图 5 所示。具体构造方式如下所示。令 A_{19}、B_{19}、C_{19} 的 32 位全为 0，这样首先能保证前向块与后向块在消息字 $\Delta W'_{20}$ 与消息字 $\Delta W'_{18}$ 上引入的差分不会影响状态 C_{19} 与状态 A_{19} 其他比特位。由于线性空间 D_1 和 D_2 在 3 轮 biclique 处所引入的差分，在比特位置上并不相交，再依据布尔函数 FF_{19} 的吸收性质，本文能保证在第 19 轮处布尔函数 FF_{19} 的输出不会受到前向块在消息字 $\Delta W'_{20}$ 上引入的差分，这样就可以顺利地将消息字 $\Delta W'_{20}$ 上移到第 18 轮，并将消息字 $\Delta W'_{18}$ 下移到第 19 轮。类似地，令 E_{19} 的 32 位全为 1，则可以利用布尔函数 GG_{19} 的吸收性质，使第 19 轮布尔函数 GG_{19} 的输出不会受到前向块在消息字 ΔW_{20} 上引入的差分，即可以顺利地将消息字 ΔW_{20} 上移到第 18 轮。依据 SM3 的步函数，本文对于输入消息 $M\oplus\delta_{1i}\oplus\delta_{2j}$ 与状态 $Q_1[\delta_{1i}]$，第 20 轮的输出为 $Q_2[\delta_{2j}]$。这说明本文成功地构造了 3 轮初始化结构（或 biclique）。

从由表 4 可知，前向块 F_1 的差分路径成立概率 $\Pr[0=_T F_1(M,S_{21})\oplus F_1(M\oplus\delta_1,S_{21})]=1$。在根据表 5 和性质 1，后向块 F_2 的差分路径成立概率满足 $\Pr[0=_T F_2^{-1}(M,S_{18})\oplus F_2^{-1}(M\oplus\delta_2,S_{18})]=1$。因此，

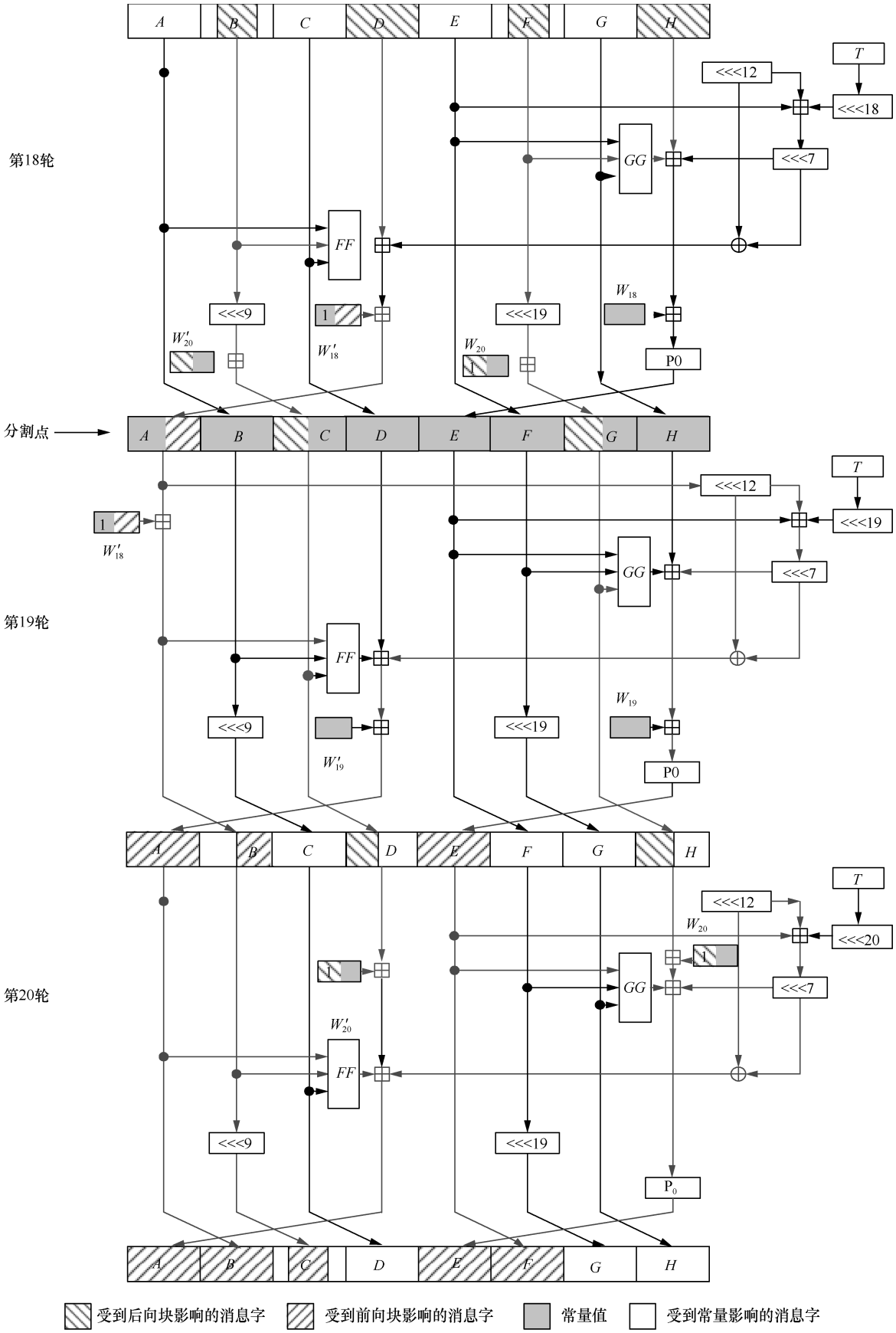


图 5　3 轮 biclique

可以用 $\frac{2^{256-5}F}{1}\approx 2^{251}$ 的时间复杂度与 2^5 的存储复杂度构造对 32 轮 SM3 的伪原像攻击。

表 4 32 轮伪原像攻击中对 F_1 的差分特征路径

差分	步函数						
	21	…	31	32	33	34	35
ΔW	0	…	0	0	0	0	0
$\Delta W'$	0	…	0	Δ_7	0	0	?
ΔA	0	…	0	?	?	?	?
ΔB	0	…	0	0	?	?	?
ΔC	0	…	0	0	0	?	?
ΔD	0	…	0	0	0	0	?
ΔE	0	…	0	0	?	?	?
ΔF	0	…	0	0	0	?	?
ΔG	0	…	0	0	0	0	?
ΔH	0	…	0	0	0	0	0
概率	1	…	1	1	1	1	1

其中，$\Delta_7 = P_1(P_1([0-1,7-9]) >>> 7)$ 。

表 5 32 轮伪原像攻击中对 F_2^{-1} 的差分特征路径

差分	步函数					
	17	…	7	6	5	4
ΔW	0	…	0	Δ_8	0	0
$\Delta W'$	0	…	0	Δ_8	0	0
ΔA	0	…	0	0	0	0
ΔB	0	…	0	0	0	Δ_{10}
ΔC	0	…	0	0	Δ_9	Δ_9
ΔD	0	…	0	Δ_9	Δ_9	Δ_{10}
ΔE	0	…	0	0	0	0
ΔF	0	…	0	0	0	Δ_{11}
ΔG	0	…	0	0	Δ_9	Δ_9
ΔH	0	…	0	Δ_9	Δ_9	Δ_{11}
概率	1	…	1	1	1	1

其中，$\Delta_8 = [27-31], \Delta_9 = \langle 0-26\rangle, \Delta_{10} = \langle 0-17\rangle, \Delta_{10} = \langle 0-7\rangle$ 。

因为对 32 轮 SM3 构造伪原像所需的时间复杂度是 2^{251}，所以根据第 3.3 节的转化算法，对 32 轮 SM3 构造一个原像攻击所需的时间复杂度为 $2^{\frac{256+251}{2}+1}\approx 2^{254.5}$，存储复杂度是 2^5。又因为在上述中间相遇攻击中，匹配点是在压缩函数的最后，因此，可以直接利用第 3.4 节的转化算法将其转化为对 SM3 散列函数 32 轮的伪碰撞攻击，依据转化算法可以知，其需要 $\frac{2^{\frac{256-5}{2}}}{1}\approx 2^{125.5}$ 的时间复杂度与 2^5 的存储复杂度。

4.3 对 33 轮 SM3 的伪碰撞攻击

除了 biclique 与切割缝合技术，本文还可以通过扩展前向块或后向块差分路径的方式来增加攻击轮数。相比于前向块，扩展后向块差分路径的效果会更好。这是由于扩展后向块时，敌手可以获得更多的匹配点，因此，整体攻击的时间复杂度也更优。通过将 32 轮后向块 F_2 的差分路径扩展一轮，从第 17 轮到第 3 轮，本文可以构造对 33 轮 SM3 的伪原像攻击，则上述 33 轮伪原像攻击可以被拆分为以下 3 个部分：1) 前向块 F_1，从第 21 轮到第 35 轮；2) 后向块 F_2，从第 17 轮到第 3 轮；3) biclique，从第 18 轮到第 20 轮。

为了应用差分中间相遇攻击，本文需要选取合适的线性空间 D_1 和 D_2 来满足第 4.1 节中的 3 个条件。经过测试，发现按以下方式来选取合适的线性空间 D_1 和 D_2 与 T 能同时满足条件 1)、2)、3)。$D_1 = \{w_{20} \| \cdots \| w_{35} \mid w_{20} = P_1[29-31] >>> 7, w_i = 0, 21 \leqslant i \leqslant 35\}$，$D_2 = \{w_4 \| \cdots \| w_{19} \mid w_6 = [7-9], w_i = 0, 4 \leqslant i \leqslant 5, 7 \leqslant i \leqslant 19\}$, $T = \{0,0,0,0,0,0,0,7\}$，则 $d = r = 3$。依据 SM3 的消息扩展算法可得，对于线性空间 D_1 有 $\Delta W_{16} = 0$，$\Delta W_{15} = 0$，$\Delta W_{17} = P_1^{-1}(\Delta W_{20} <<< 7) = [29-31]$，$\Delta W_{14} = P_1^{-1}(\Delta W_{17} <<< 7) = [1-6, 9-11, 17-21, 25-29]$。类似地，对于线性空间 D_2 可得 $\Delta W_{16} = 0$, 以及 $\Delta W_{15} = \Delta W_{14} = 0$。因此，线性空间 D_1 和 D_2 不会在输入消息的最后 65 bit 上引入差分。由上述赋值，还易得 3 轮 biclique 也还会成立。

经过实验，本文发现后向块 F_2 差分路径成立概率为 $\Pr[0 =_T F_2^{-1}(M, S_{18}) \oplus F_2^{-1}(M \oplus \delta_2, S_{18})] = 0.85$。而前向块 F_1 的差分路径成立概率与 32 轮攻击时一样还是 $\Pr[0 =_T F_1(M, S_{21}) \oplus F_1(M \oplus \delta_1, S_{21})] = 1$。因此，可以用 $\frac{2^{253}F + 2^{256-3}F_{re}}{0.85} \approx 2^{253.3}$ 的时间复杂度与 2^3 的存储复杂度构造对 33 轮 SM3 的伪原像攻击。注意到，由于 SM3 的轮函数设计比较复杂，本文无法构造超过 3 轮的 biclique 结构。此外，通过实验也发现，如果敌手还想继续扩展前向块（或后向块）的轮数，则对应的前向块 F_1（或后向块 F_2）的差分

路径成立概率会急速下降到接近 0，而这会使对应攻击的时间复杂度超过穷举攻击的时间复杂度 2^n，而使上述改进想法变得无效，因此，33 轮伪碰撞攻击是本文所得到的关于 SM3 散列函数算法最长轮数的碰撞攻击轮数。

因为对 33 轮 SM3 构造伪原像所需的时间复杂度是 $2^{253.3}$，依照第 3.3 节的转化算法，对 33 轮 SM3 构造一个原像攻击所需的时间复杂度为 $2^{\frac{256+253.3}{2}+1}\approx 2^{255.6}$。由于 $2^{255.6}\approx 2^{256}$，本文不认为上述 33 轮的原像攻击是有效的，只能将其看成是对暴力破解的加速。因为在上述中间相遇攻击中，匹配点是在压缩函数的最后，所以，可以直接利用第 3.4 节中的转化算法直接将其转化为对 SM3 散列函数 33 轮的伪碰撞攻击，依据转化算法可知，其所需的时间复杂度为 $\frac{2^{\frac{256-3}{2}}}{0.85}\approx 2^{126.7}$ 与 2^3 的存储复杂度。

5　结束语

本文发现要对 SM3 构造原像攻击是比较困难的，这是因为：1) SM3 采用了复杂的步函数；2) SM3 的消息填充算法限制了线性空间 D_1 和 D_2 的取值范围。不过相比于中间相遇攻击，本文发现差分中间相遇攻击更适合用来构造对 SM3 的原像攻击，这主要是由于 SM3 采用了线性化的消息扩展算法。利用差分中间相遇、切割缝合技术与 biclique 方法，提出了对 SM3 的 32 轮原像攻击，以及 33 轮的伪碰撞攻击。在本文之前，对 SM3 最好的原像结果只有 30 轮，而最好的伪碰撞攻击只有 32 轮。

参考文献：

[1] SASAKI Y, AOKI K. Preimage attacks on step-reduced MD5[C]//The 13th Information Security and Privacy Australasian Conference. 2008: 282-296.

[2] GUO J, LING S, RRCHBERGER C. Advanced meet-in-the-middle preimage attacks: first results on full Tiger, and improved results on MD4 and SHA-2[C]//The 16th International Conference on the Theory and Application of Cryptology and Information Security. 2010: 56-75.

[3] CANNIERE D C, RECHBERGER C. Preimages for reduced SHA-0 and SHA-1[C]//The 28th Annual International Cryptology Conference, 2008: 179-202.

[4] KHOVRATOVICH D, RECHBERGER C, SAVELIEVA A. Bicliques for preimages: attacks on skein-512 and the SHA-2 family[C]//The 19th Fast Software Encryption International Workshop. 2012: 244-263.

[5] LI J, ISOBE T, SHIBUTANI K. Converting meet-in-the-middle preimage attack into pseudo collision attack: application to SHA-2[C]// The 19th Fast Software Encryption International Workshop. 2012: 264-286.

[6] KNELLWOLF S, KHOVRATOVICH D. New preimage attacks against reduced SHA-1[C]//The Advances in Cryptology 32nd Annual Cryptology Conference.2012: 367-383.

[7] ZOU J, WU W. L, WU S, et al. Preimage attacks on step-reduced sm3 hash function[C]//The 14th Information Security and Cryptology International Conference. 2011: 375-390.

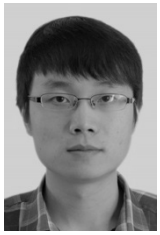
[8] WANG G L, SHEN Y Z: Preimage and pseudo-collision attacks on step-reduced SM3 hash function[J]. Inf Process Lett, 2013, 113(8): 301-306.

[9] MENDEL F, NAD T, SCHLAFER M. Finding collisions for round-reduced SM3[C]//The Cryptographers' Track at the {RSA} Conference 2013. 2013: 174-188.

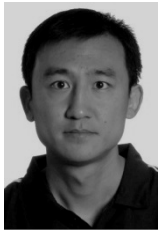
[10] AOKI K, SASAKI Y. Preimage attacks on one-block MD4, 63-step MD5 and more[S]//Workshop Records of SAC 2008. 2008: 82-98.

[11] PAUL C O A, MENEZES J, SCOTT A. Vanstone. handbook of applied cryptography[M]. CRC Press, 1996.

[作者简介]



邹剑（1985-），男，福建福州人，博士，福州大学讲师，主要研究方向为散列函数和分组密码的分析。



董乐（1980-），男，河南新乡人，博士，河南师范大学副教授，主要研究方向为散列函数和分组密码的分析。