2 轮 Trivium 的多线性密码分析

贾艳艳*　胡予濮　杨文峰　高军涛
(西安电子科技大学计算机网络与信息安全教育部重点实验室　西安　710071)

摘　要：作为欧洲流密码发展计划 eSTREAM 的 7 个最终获选算法之一，Trivium 的安全性考察表明至今为止还没有出现有效的攻击算法。该文针对 2 轮 Trivium，通过找出更多线性逼近方程，对其进行了多线性密码分析，提出了一种更有效的区分攻击算法。与现有的单线性密码分析算法相比，该算法攻击成功所需的数据量明显减少，即：若能找到 n 个线性近似方程，在达到相同攻击成功概率的前提下，多线性密码分析所需的数据量只有单线性密码分析的 $1/n$。该研究结果表明，Trivium 的设计还存在一定的缺陷，投入实用之前还需要实施进一步的安全性分析。

关键词：密码学；流密码；密码分析；Trivium；线性近似
中图分类号：TN918.1　　**文献标识码**：A　　**文章编号**：1009-5896(2011)01-0223-05
DOI: 10.3724/SP.J.1146.2010.00334

Linear Cryptanalysis of 2-round Trivium with Multiple Approximations

Jia Yan-yan　Hu Yu-pu　Yang Wen-feng　Gao Jun-tao
(*Key Laboratory of Computer Networks and Information Security*, *Xidian University*, *Xi'an* 710071, *China*)

Abstract: Trivium has successfully been chosen as one of the final ciphers by eSTREAM. It has a simple and elegant structure. Although Trivium has attached a lot of interest, it remains unbroken. By finding more linear approximations, a linear cryptanalysis of 2-round Trivium is made by utilizing multiple approximations and a more efficient distinguishing attack is proposed. Compared with current single linear cryptanalysis, this method allows for a reduction in the amount of data required for a successful attack. That is to say, if n linear approximations can be found, this method can supply the success rate with $1/n$ of the data amount required by a simple linear cryptanalysis. This study shows that there are still some defects in the design of stream cipher Trivium, further safety analysis are needed before its going into the implementation.

Key words: Cryptology; Stream ciphers; Cryptanalysis; Trivium; Linear approximation

1　引言

作为欧洲流密码发展计划 eSTREAM 的获选算法之一，Trivium[1,2] 是一种面向硬件应用的基于移位寄存器的流密码算法。尽管 Trivium 的设计极其简单灵活，但经过多年的安全性考察，至今为止还没有发现有效的密钥恢复攻击算法。

当前对于 Trivium 的安全性分析已经有一些初步的研究成果。其中文献[3]对 Trivium 进行了两种简单攻击，即状态恢复攻击和统计测试。虽然这些攻击可以成功地对 Brivium(Trivium 的另一个简化版本)进行攻击，但对 Trivium 的攻击结果却慢于穷举搜索。文献[4]利用 SAT 求解器和 BDD 对 Bivium 进行了攻击，但并没有解决如何成功应用其攻击算法于 Trivium。文献[5]提出了一种区分攻击，得到的相关系数只有 2^{-72}，所以其攻击成功所需的数据复杂度为 $O(2^{144})$。文献[6]通过求解一个线性方程组来恢复 288 bit 的初始状态，其计算复杂度为 $O(2^{164})$。最近文献[7]提出了 Trivium 的一种新的简化版本即 2 轮 Trivium，并且找到了 2 轮 Trivium 的一个偏差为 2^{-31} 的线性近似方程，这是当前的关于区分攻击的最好结果。

但文献[7]的缺陷是只找到和利用了一个线性近似方程，本文通过深入分析 2 轮 Trivium 的轮函数，找到更多的线性逼近方程，对其进行了多线性逼近攻击。具体来讲，本文首先通过分析 2 轮 Trivium 轮函数的特点，找到了一个新的线性近似方程，然后用这两个方程对 2 轮 Trivium 进行了多线性密码分析，攻击成功需要 2^{61} 个选择 **IV**，仅为单线性密码分析的一半。就线性密码分析而言，若能找到 n 个线性近似方程进行多线性密码分析，那么所需的数据量仅是单线性密码分析所需数据量的 $1/n$。

2　基础知识

2.1 Trivium 简介

Trivium 为具有 80 bit K(密钥)和 80 bit **IV**(初

2010-04-01 收到，2010-07-11 改回
国家自然科学基金(60833008)，国家 973 计划项目(2007CB311201)和广西信息与通讯技术重点实验室基金(20902)资助课题
*通信作者：贾艳艳　xibeijiayan123@163.com

始化向量)的二元同步流密码。该密码算法产生的密钥流序列与明文序列异或产生密文序列。Trivium 算法分成两部分：一部分为初始化算法，其功能为将密钥和初始化向量转化为 Trivium 的内部状态向量。首先将内部状态的$(s_1,\cdots,s_{80})$和$(s_{94},\cdots,s_{173})$初始化为$(k_1,\cdots,k_{80})$和$(\mathbf{IV}_1,\cdots,\mathbf{IV}_{80})$，其余比特除最后 3 位外全部初始化为 0，然后在不产生输出比特的情况下将算法运行 $4\cdot288$ 个时钟周期。另一部分为密钥流生成算法，其功能为生成加密用的密钥流。密钥生成算法与初始化算法相似，只是多了一个生成密钥流 z_i，i=1,2,$\cdots$ 的输出函数。设计要求 Trivium 的一对密钥和 **IV** 生成至多 2^{64} bit 的密钥流，具体算法描述如下。

(1)Trivium 的初始化算法示于表 1。

表 1 Trivium 初始化算法

输入：密钥 $K=(k_1,\cdots,k_{80})$，初始化向量 $\mathbf{IV}=(\mathbf{IV}_1,\cdots,\mathbf{IV}_{80})$
输出：Trivium 的内部状态 $(s_1,\cdots,s_{288})$
$(s_1,\cdots,s_{93})\leftarrow(k_1,\cdots,k_{80},0,\cdots,0)$
$(s_{94},\cdots,s_{177})\leftarrow(\mathbf{IV}_1,\cdots,\mathbf{IV}_{80},0,\cdots,0)$
$(s_{178},\cdots,s_{288})\leftarrow(0,\cdots,0,1,1,1)$
for i=0 to $4\cdot288$ do
$t_1\leftarrow s_{66}+s_{91}s_{92}+s_{93}+s_{171}$
$t_2\leftarrow s_{162}+s_{175}s_{176}+s_{177}+s_{264}$
$t_3\leftarrow s_{243}+s_{286}s_{287}+s_{288}+s_{69}$
$(s_1,\cdots,s_{93})\leftarrow(t_3,s_1,\cdots,s_{92})$
$(s_{94},\cdots,s_{177})\leftarrow(t_1,s_{94},\cdots,s_{176})$
$(s_{178},\cdots,s_{288})\leftarrow(t_2,s_{178},\cdots,s_{287})$
end for

(2)密钥流生成算法示于表 2。

表 2 密钥流生成算法

输入：Trivium 的内部状态 $(s_1,\cdots,s_{288})$，输出比特的数目 $N\leqslant2^{64}$
输出：密钥流 $(z_0z_1z_2\cdots z_N)$
for i=0 to N do
$z_i\leftarrow s_{66}+s_{93}+s_{162}+s_{177}+s_{243}+s_{288}$
$t_1\leftarrow s_{66}+s_{91}s_{92}+s_{93}+s_{171}$
$t_2\leftarrow s_{162}+s_{175}s_{176}+s_{177}+s_{264}$
$t_3\leftarrow s_{243}+s_{286}s_{287}+s_{288}+s_{69}$
$(s_1,\cdots,s_{93})\leftarrow(t_3,s_1,\cdots,s_{92})$
$(s_{94},\cdots,s_{177})\leftarrow(t_1,s_{94},\cdots,s_{176})$
$(s_{178},\cdots,s_{288})\leftarrow(t_2,s_{178},\cdots,s_{287})$
end for

2.2 线性密码分析

线性密码分析是对迭代密码的一种已知明文攻击，它利用的是密码算法中的“不平衡(有效)的线性逼近”。线性密码分析是针对分组密码提出的，设明文分组长度和密文分组长度都为 n 比特，密钥分组长度为 m 比特。记明文分组为 $P[1]$，$P[2]\cdots$，$P[n]$，密文分组为 $C[1]$，$C[2]$，$\cdots$，$C[n]$，密钥分组为 $K[1]$，$K[2]$，$\cdots$，$K[m]$。定义 $A[i,j,\cdots,k]=A[i]\oplus A[j]\oplus\cdots\oplus A[k]$。

线性密码分析的目标就是找出以下形式的有效线性方程：

$$P[i_1,i_2,\cdots,i_a]\oplus C[j_1,j_2,\cdots,j_b]=K[k_1,k_2,\cdots,k_c]$$

其中 $1\le a\le n$，$1\le b\le n$，$1\le c\le m$。如果方程成立的概率 $p\neq1/2$，则称该方程是有效的线性逼近。如果 $|p-1/2|$ 是最大的，则称该方程是最有效的线性逼近。对于给定的密码算法，线性密码分析的目的就是找到它的一个线性逼近。针对多轮的分组密码，首先对不同轮的非线性函数进行逼近，然后将各个逼近有效地组合，最终得到有效的线性近似。

任意一个流密码都可以看作布尔函数 $F_i:F_2^k\times F_2^v\to F_2, i=1,2,\cdots$ 的集合，其中 F_i 为由 k 比特密钥和 v 比特 **IV** 生成第 i 个输出比特 z_i 的函数，每一个 F_i 都要受到初始化算法和密钥生成算法两个阶段的影响。将密码再同步 ε^{-2} 次，如果找到了 F_i 一个的线性近似函数且具有的偏差 $\varepsilon>2^{-k/2}$，则一定可以给出 1 bit 的密钥信息。

寻找 F_i 的线性近似主要分两步进行：首先，基于分组密码轮函数的思想，将流密码的初始化算法分成 n 轮，其中第 i 轮有 t_i 个时钟，所有 t_i 的和一定等于总的初始化时钟数 T，然后对每一轮进行线性近似，具体过程如图 1 所示；其次，通过将每轮的线性近似函数有效地组合得到整个密码算法的线性近似函数，最终得到一个仅和密钥，**IV** 及密钥流比特有关的线性近似，其偏差可利用堆积定理求得。

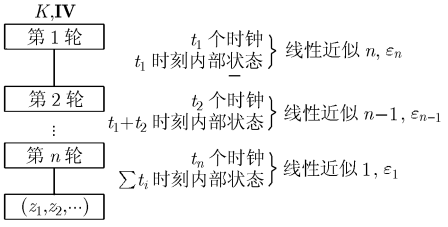

图 1 分轮后的流密码算法

3 攻击算法

根据 Trivium 的初始化算法及流密码的线性密码分析思想，可以把 Trivium 的初始化阶段像分组密码一样分为 n 轮。考虑到轮数和每轮中时钟数的折衷，选 144 个时钟为一轮来分析 2 轮 Trivium。对于 2 轮 Trivium 有下式成立，其中 $s_t(i)$ 为 t 时刻

的内部状态。

$K=(s_0(1),s_0(2),\cdots,s_0(80))$, **IV**$=(s_0(94), s_0(95),\cdots, s_0(173))$, $z_1=s_{288}(66)+s_{288}(93)+s_{288}(162)+s_{288}(177)+s_{288}(243)+s_{288}(288)$。

根据线性密码分析思想，找 2 轮 Trivium 的一个线性近似函数即求 F_1 的一个线性近似函数。由于直接求 F_1 关于初始状态向量的函数不太容易，因此首先对第 1 轮和第 2 轮分别求近似函数，然后将两个线性近似函数有效的组合，得到一个最终的线性近似函数，该过程如图 2 所示。文献[7]成功地找到了一个偏差为 2^{-31} 的线性近似函数。一般来说，密码系统存在不止一个线性近似函数，而文献[8]指出多个线性近似必定给出更多的密钥信息，本文将努力找出 F_1 的更多的线性近似函数。

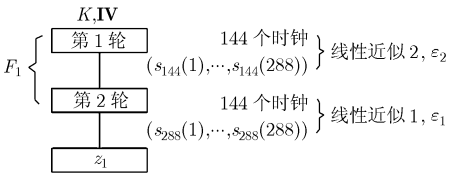

图 2 2 轮 Trivium 的线性近似

由 Trivium 的初始化及密钥生成过程可知，输出比特 z_1 是比特 $s_{288}(66)$,$s_{288}(93)$, $s_{288}(162)$, $s_{288}(177)$, $s_{288}(243)$和 $s_{288}(288)$之和，通过迭代，得到 F_1 关于 $t=144$ 时刻内部状态的方程如下：

$$\begin{aligned} z_1 &= s_{144}(6)+s_{144}(16)\cdot s_{144}(117)+s_{144}(31)\cdot s_{144}(32)\\ &\quad +s_{144}(33)+s_{144}(57)+s_{144}(82)\cdot s_{144}(83)+s_{144}(84)\\ &\quad +s_{144}(96)+s_{144}(97)\cdot s_{144}(98)+s_{144}(99)\\ &\quad +s_{144}(111)+s_{144}(129)+s_{144}(142)\cdot s_{144}(143)\\ &\quad +s_{144}(144)+s_{144}(150)+s_{144}(162)+s_{144}(163)\\ &\quad \cdot s_{144}(164)+s_{144}(165)+s_{144}(186)+s_{144}(192)\\ &\quad +s_{144}(208)\cdot s_{144}(209)+s_{144}(210)+s_{144}(231)\\ &\quad +s_{144}(235)\cdot s_{144}(236)+s_{144}(237)+s_{144}(252) \end{aligned} \tag{1}$$

文献[7]找到了式(1)的一个偏差为 2^{-9} 的线性近似，如式(2)所示。

$$\begin{aligned} z_1 &= s_{144}(6)+s_{144}(33)+s_{144}(57)+s_{144}(84)\\ &\quad +s_{144}(96)+s_{144}(99)+s_{144}(111)+s_{144}(129)\\ &\quad +s_{144}(144)+s_{144}(150)+s_{144}(162)+s_{144}(165)\\ &\quad +s_{144}(186)+s_{144}(192)+s_{144}(210)+s_{144}(231)\\ &\quad +s_{144}(237)+s_{144}(252) \end{aligned} \tag{2}$$

通过迭代，在设定 $iv_{25}=iv_{26}=iv_{31}=iv_{32}=iv_{49}=iv_{50}=iv_{54}=iv_{55}=iv_{70}=iv_{71}=0$ 且 $k_{14}=k_{19}=k_{20}=k_{38}=k_{39}=k_{45}=k_{63}=k_{64}=k_{65}=k_{77}=0$ 的条件下，由式(2)最终得到了一个关于 K，**IV** 及密钥流比特的的线性近似，其偏差为 2^{-31}，如式(3)所示。

$$\begin{aligned} z_1 &= 1+k_3+k_6+k_{15}+k_{21}+k_{27}+k_{30}+k_{39}+k_{57}+k_{67}+k_{68}\\ &\quad +k_{69}+k_{72}+iv_3+iv_6+iv_{21}+iv_{24}+iv_{30}+iv_{33}+iv_{39}\\ &\quad +iv_{45}+iv_{51}+iv_{72}+iv_{78} \end{aligned} \tag{3}$$

文献[7]中认为他们找到了式(1)的一个最逼近的线性近似函数，本文用 MATLAB 研究式(1)的 Walsh 谱，发现其只有 0 和 $\pm 2^{26}$ 三个谱值，也就是说式(1)为三谱值的 Plateaued 函数[9,10]，那么一定可以找到式(1)的更多线性近似，并且都具有相同的偏差。这也说明 Turan 关于最逼近式(1)的线性方程为式(2)的说法是不太准确的。观察 Walsh 谱值，找到式(1)的具有偏差也为 2^{-9} 的另一个线性近似如下所示：

$$\begin{aligned} z_1 &= s_{144}(6)+s_{144}(16)+s_{144}(33)+s_{144}(57)+s_{144}(84)\\ &\quad +s_{144}(96)+s_{144}(99)+s_{144}(111)+s_{144}(129)\\ &\quad +s_{144}(144)+s_{144}(150)+s_{144}(162)+s_{144}(165)\\ &\quad +s_{144}(186)+s_{144}(192)+s_{144}(210)+s_{144}(231)\\ &\quad +s_{144}(237)+s_{144}(252) \end{aligned} \tag{4}$$

算法最终需要的是一个仅与密钥，**IV** 及输出比特有关的线性近似，所以通过 Trivium 初始化算法的迭代运算，将式(4)写成关于 $s_0(i)$,i=1,2,⋯,80 及 i=94,⋯,173 的函数。因为比特($s_0(i)$, i=81,82,⋯,93 及 i=174,⋯,288)都初始化为常数 0 或 1，所以得到的式子中不再含有其他比特，最终得到一个共有 29 个线性项，64 个 2 次项和 24 个 3 次项的式子。因为若 x，y，z 均为二元独立同分布的随机变量，有 $p(x\cdot y=0)=0.75$，$p(x\cdot y\cdot z=0)=0.875$，所以令所有的非线性项均为 0，找到了该式的一个具有较大偏差的线性近似：

$$\begin{aligned} z_1 &= 1+s_0(3)+s_0(6)+s_0(10)+s_0(15)+s_0(21)\\ &\quad +s_0(27)+s_0(30)+s_0(39)+s_0(54)+s_0(57)\\ &\quad +s_0(67)+s_0(68)+s_0(69)+s_0(72)+s_0(96)\\ &\quad +s_0(99)+s_0(100)+s_0(114)+s_0(115)+s_0(117)\\ &\quad +s_0(123)+s_0(126)+s_0(132)+s_0(138)+s_0(144)\\ &\quad +s_0(145)+s_0(160)+s_0(165)+s_0(171) \end{aligned} \tag{5}$$

当所有的非线性项都相互独立时，式(5)以 $2^{87}\cdot(0.25)^{64}\cdot(0.375)^{24}=2^{-83.96}$ 的偏差成立。这个偏差太小，对于线性密码分析没有意义。以下通过设定一些 **IV** 和密钥比特为 0 使这个偏差增大。令 $iv_{20}=iv_{25}=iv_{26}=iv_{31}=iv_{32}=iv_{49}=iv_{50}=iv_{54}=iv_{55}=iv_{64}=iv_{65}=iv_{70}=iv_{71}=0$ 且 $k_{14}=k_{19}=k_{20}=k_{38}=k_{39}=k_{45}=k_{63}=k_{64}=k_{65}=k_{77}=0$，那么可以得到如下具有 29 个线性项

和 22 个 2 次项的形式：

$$\begin{aligned}z_1 &= 1+s_0(3)+s_0(6)+s_0(10)+s_0(15)+s_0(21)\\&\quad+s_0(27)+s_0(30)+s_0(39)+s_0(54)+s_0(57)\\&\quad+s_0(67)+s_0(68)+s_0(69)+s_0(72)+s_0(96)\\&\quad+s_0(99)+s_0(100)+s_0(114)+s_0(115)+s_0(117)\\&\quad+s_0(123)+s_0(126)+s_0(132)+s_0(138)+s_0(144)\\&\quad+s_0(145)+s_0(160)+s_0(165)+s_0(171)+s_0(4)\\&\quad\cdot s_0(5)+s_0(13)\cdot s_0(41)+s_0(16)\cdot s_0(17)+s_0(22)\\&\quad\cdot s_0(23)+s_0(25)\cdot s_0(26)+s_0(34)\cdot s_0(35)+s_0(43)\\&\quad\cdot s_0(44)+s_0(49)\cdot s_0(50)+s_0(52)\cdot s_0(53)+s_0(58)\\&\quad\cdot s_0(59)+s_0(61)\cdot s_0(62)+s_0(67)\cdot s_0(68)+s_0(70)\\&\quad\cdot s_0(71)+s_0(103)\cdot s_0(104)+s_0(106)\cdot s_0(107)\\&\quad+s_0(127)\cdot s_0(128)+s_0(130)\cdot s_0(131)+s_0(133)\\&\quad\cdot s_0(149)+s_0(134)\cdot s_0(138)+s_0(151)\cdot s_0(152)\\&\quad+s_0(154)\cdot s_0(155)+s_0(157)\cdot s_0(158)\end{aligned}\tag{6}$$

式(6)可由式(5)线性逼近，其中偏差为 $2^{21}\cdot(2^{-2})^{22}=2^{-23}$。

将第 1 轮的线性近似函数式(4)和第 2 轮的线性近似函数式(5)有效地组合，得到 2 轮 Trivium 最终的一个线性近似式为：

$$\begin{aligned}z_1=&1+k_3+k_6+k_{10}+k_{15}+k_{21}+k_{27}+k_{30}+k_{39}+k_{54}+k_{57}+k_{67}\\&+k_{68}+k_{69}+k_{72}+iv_3+iv_6+iv_7+iv_{21}+iv_{22}+iv_{24}+iv_{30}+iv_{33}\\&iv_{39}+iv_{45}+iv_{51}+iv_{52}+iv_{67}+iv_{72}+iv_{78}\end{aligned}\tag{7}$$

式(7)成立的偏差由堆积定理求得：$2\cdot2^{-9}\cdot2^{-23}=2^{-31}$。可见该线性近似具有与文献[7]中 Turan 近似相同的偏差。

通过设定 K 和 **IV** 的某些比特为 0，以上找到了 2 轮 Trivium 的两个线性近似函数即式(3)和式(7)，其偏差均为 2^{-31}。也就是说一定可以进行多线性密码分析。若可以找到更多的线性近似函数如 n 个。设第 i 个近似函数为

$$z_1^i+\mathbf{IV}[\chi_{\mathrm{IV}}^i]=K[\chi_K]\tag{8}$$

其中 $\mathbf{IV}[\chi_{\mathrm{IV}}^i]$ 为第 i 个线性近似所用到的 **IV** 的某些比特之和，$K[\chi_K]$ 为 K 中某些比特之和。为了便于分析，假设每一个线性近似的偏差均为正；否则，只要将近似函数相位取反即加常数 1 即可。如此，对于第 i 个线性近似，每给定一个 **IV**，运行 2 轮 Trivium，产生一个 z_1，得到一个(**IV**, z_1)对，令 N 表示总的(**IV**, z_1)的对数，T_i 为使得式(8)左边等于 0 的(**IV**, z_1)的对数，那么得到攻击算法如下。

2 轮 Trivium 的多线性密码分析步骤：

第 1 步　找到 2 轮 Trivium 的 n 个线性近似函数，设第 i 个线性近似的偏差为 ε_i，其中 $1\le i\le n$；

第 2 步　对于 $1\le i\le n$，令 T_i 为使得式(8)左边等于 0 的(**IV**, z_1)对的对数，令 N 表示总的(**IV**, z_1)对的对数；

第 3 步　令 $a_i=\varepsilon_i\big/\sum\varepsilon_i$，那么 $\sum_{i=1}^n a_i=1$；对于一些权重 $a_1,\cdots,a_n$ 的集合，计算 $U=\sum_{i=1}^n a_iT_i$；

第 4 步　如果 $U>N/2$，那么令 $K[\chi_K]=0$，否则使 $K[\chi_K]=1$。

4　算法分析

同步流密码通过更换 **IV** 来实现通信的再同步，因此，选择 **IV** 攻击是实际可行的。通过选定不同的 **IV** 产生不同的输出比特 z_1，**IV** 和 z_1 一一对应，组成了(**IV**, z_1)对，设总共有 N 个(**IV**, z_1)对。由上一节可知，2 轮 Trivium 已被转化为了一个分组密码算法，因此对其的线性密码分析与分组密码的线性密码分析实质是一样的。而文献[11]中 Matsui 的单线性密码分析成功的概率大概为 $\Phi(2\sqrt{N}\varepsilon)$，若能够找到 2 轮 Trivium 的 n 个线性逼近函数，那么根据分组密码的多线性密码分析[8,12]，有：

定理 1　当随机选择 **IV**，并且 $U=\sum_{i=1}^n a_iT_i$ 的概率分布符合正态分布时，2 轮 Trivium 的多线性密码分析成功的概率约为

$$\Phi\left(2\sqrt{N}\sqrt{\frac{\sum_{i=1}^n\varepsilon_i^2}{1-4\sum_{i=1}^n\varepsilon_i^2}}\right)\tag{9}$$

当 $\sum_{i=1}^n\varepsilon_i^2$ 比较小并且找到的 n 个线性近似函数具有相同的偏差 ε，那么攻击成功的概率式(9)大约为 $\Phi(2\sqrt{nN}\varepsilon)$。为了说明多线性密码分析的优点，假设要求 2 轮 Trivium 的单线性密码分析和多线性密码分析能够提供相同的攻击成功的概率，单线性密码分析需要的(**IV**, z_1)对为 N，多线性密码分析所需要的(IV, z_1)对为 N'，那么要求 $\Phi(2\sqrt{N}\varepsilon)=\Phi(2\sqrt{nN'}\varepsilon)$，有 $N'=N/n$。也就是说，若能够找到 n 个线性近似进行多线性密码分析，则只需要单线性密码分析所需的 $1/n$ 的数据量即 $1/n$ 的(**IV**, z_1)对就能够提供相同的攻击成功的概率。

本文具体找到了 2 轮 Trivium 的一个新的线性近似方程式(7)，与文献[7]找到的线性近似方程式(3)一起，能够进行多线性密码分析。文献[7]中指出利用一个线性近似进行密码分析，识别一个具有特定比特的密钥 K，需要 2^{62} 个选择 **IV**。由以上分析可知，若用式(3)和式(7)进行多线性密码分析，则只需要一半的选择 **IV** 即 2^{61} 个 **IV**。需要指出的是该算法需要多指定 3 个 **IV** 比特为 0，但在选择 **IV** 攻击中，这是不困难的。

为了进一步说明算法的有效性，本文针对不同数目的(**IV**, z_1)对进行攻击，攻击结果如表 3 所示。第 1 行表示选择(**IV**, z_1)对的数目分别为 $(1/16)\varepsilon^{-2}=2^{58}$，$(1/8)\varepsilon^{-2}=2^{59}$，$(1/4)\varepsilon^{-2}=2^{60}$，$(1/2)\varepsilon^{-2}=2^{61}$，$\varepsilon^{-2}=2^{62}$，第 2 行表示利用式(3)进行单线性密码分析的攻击成功的概率，第 3 行表示利用式(7)进行单线性密码分析所能达到的攻击成功的概率，第 4 行表示利用式(3)和式(7)进行多线性密码分析所能达到的攻击成功的概率。表 3 的结果表明，利用两个线性近似进行多线性密码分析只需要单线性密码分析一半的数据量就可以提供相同的攻击成功的概率。

表 3 单线性密码分析与多线性密码分析的比较(%)

(**IV**,z_1)的对数	2^{58}	2^{59}	2^{60}	2^{61}	2^{62}
式(3)逼近	69.2	70.7	84.1	92.1	97.7
式(7)逼近	69.2	70.7	84.1	92.1	97.7
式(3)和式(7)逼近	70.7	84.1	92.1	97.7	99.7

5　结束语

流密码具有加解密速度快，实现简单，资源消耗少等优点，在现代通信尤其是在硬件资源受限的环境中具有广泛的应用，例如手机通信 GSM 中的 A5/1，蓝牙中的 E_0 等。但是许多算法已被证明是不安全的，因此急需设计更加安全的流密码算法。其中 Trivium 就是在这种背景下设计的优秀的密码算法，并成为了欧洲 eSTREAM 计划的最终获选算法之一。本文首先找到了 2 轮 Trivium 的一个新的线性近似，接着利用多线性密码分析对 2 轮 Trivium 成功地实施了可区分攻击。攻击结果表明，只要找到 2 轮 Trivium 的 n 个线性近似，攻击所需的数据量仅为单线性密码分析所需数据量的 $1/n$。值得指出的是，本文主要是利用 Walsh 谱，结合对 2 轮 Trivium 轮函数特点的观察给出一个新的线性近似方程，至于如何设计系统的算法来寻找 2 轮 Trivium 更多的线性近似函数还需进一步研究。

参 考 文 献

[1] De Cannière C and Preneel B. Trivium: a stream cipher construction inspired by block cipher design principle[R]. eSTREAM, ECRYPT Stream Cipher Project, Report 2005/30 (2005), http://www.ecrypt.eu.org/stream, 2009.12.

[2] De Cannière C and Preneel B. Trivium specifications. www.ecrypt.eu.org/stream/p3ciphers/trivium /trivium_p3.pdf, 2009.10.

[3] Maximov A and Biryukov A. Two trivial attacks on TRIVIUM[C]. Workshop on The State of the Art of Stream Ciphers (SASC2007), Bochum, 2007: 1–16.

[4] Eiban T, Pilz E, and Steck S. Comparing and optimizing two generic attacks on Bibium[C]. Workshop on The State of the Art of Stream Ciphers (SASC2008), Lausanne, 2008: 57–68.

[5] Khazaei S, Hasanzaden M M, and Kiaei M S. Linear sequential circuit approximation of Grain and Trivium stream ciphers [R]. eSTREAM, ECRYPT Stream Cipher Project, Report 2005/063, 2005.

[6] Raddum H. Cryptanalytic results on trivium[R]. eSTREAM, ECRYPT Stream Cipher Project, Report 2006/039, 2006.

[7] Turan M S and Kara O. Linear approximations for 2-round Trivium[C]. Workshop on The State of the Art of Stream Cipher (SASC2007), Bochum, 2007: 22–31.

[8] Kaliski B S Jr and Robshaw M J B. Linear cryptanalysis using multiple approximations[C]. CRYPTO, London, UK, 1994: 26–39.

[9] 胡斌，金晨辉，冯春海. Plateaued 函数的密码学性质[J]. 电子与信息学报, 2008, 30(3): 660–664.
Hu Bin, Jin Chen-hui, and Feng Chun-hai. Cryptographic properties of plateaued functions[J]. *Journal of Electronics & Information Technology*, 2008, 30(3): 660–664.

[10] 王维琼，周宇，肖国镇. Plateaued 函数的正规性[J]. 电子与信息学报，2009，31(9): 2283–2286.
Wang Wei-qiong, Zhou Yu, and Xiao Guo-zhen. Normality of plateaued functions[J]. *Journal of Electronics & Information Technology*, 2009, 31(9): 2283–2286.

[11] Matsui M. Linear cryptanalysis method for DES cipher[C]. Advances in cryptology – Eurocrypt'93, Springer-Verlag, Berlin, 1994: 386–397.

[12] Gérard B and Tillich J P. On linear cryptanalysis with many linear approximations. Cryptography and Coding 2009, 2009 LNCS 5921: 112–132.

贾艳艳：　女，1983 年生，博士生，从事流密码的分析与设计等方面的研究.

胡予濮：　男，1955 年生，博士生导师，教授，从事网络安全、密码学等方面的研究.

杨文峰：　男，1971 年生，博士生，副教授，研究方向为密码学.

高军涛：　男，1979 年生，博士，副教授，从事流密码的分析与设计等方面的研究.