یافتن مشخصه خطی در الگوریتم های رمز قطعه ای با استفاده از شیوه بهینه سازی ماشین بولتزمن

عباس قائمی بافقی*

چکیده

تحلیل خطی روشی متداول برای ارزیابی الگوریتم های رمز قطعه‌ای است. قبلا بکارگیری شبکه عصبی هاپفیلد برای یافتن بهترین مشخصه خطی در الگوریتم رمز قطعه‌ای توسط مولف مقاله مطرح شد، که در آن با افزایش تعداد دور مشخصه، احتمال گیرکردن در بهینه‌های محلی در هنگام بهینه سازی افزایش می‌یابد. دراین مقاله، جهت کاهش این مشکل از شیوه‌های آموزش احتمالی و ایده Annealing استفاده کرده و با بکارگیری ماشین بولتزمن کارایی بیشتری بدست آمده است. برای نمونه، این روش برای یافتن مشخصه های خطی در الگوریتم رمز کهکشان بکارگرفته شده است. الگوریتم رمز کهکشان یک الگوریتم رمز قطعه‌ای با طول قطعه ورودی/خروجی و طول کلید ۲۵۶ بیت می باشد که از ۳۲ دور تکرار تبدیل جانشینی- جایگشتی بدست آمده است. در این مقاله مشخصه خطی برای الگوریتم رمز کهکشان ۸، ۹ و ۱۰ دوری بترتیب با تمایل احتمال $۲^{-۷۸}$، $۲^{-۱۰۸}$ و $۲^{-۱۳۷}$ بدست آمده که در مقایسه با تمایل احتمال مشخصه های بدست آمده با شبکه هاپفیلد نتیجه بهتری است.

کلمات کلیدی

رمز قطعه‌ای، الگوریتم رمز کهکشان، تحلیل خطی، شبکه عصبی هاپفیلد، ماشین بولتزمن.

Finding Linear Cryptanalysis Characteristic of Block Cipher Using Boltzmann Machine

Abbas Ghaemi Bafghi

Abstract

Linear cryptanalysis is a usual method to evaluation of block ciphers. Previously, we apply Hopfield learning algorithm to find the best linear characteristic of block ciphers. But experiments show the probability of convergence with the Hopfield to a local minimum is increased, when the number of rounds of a block cipher algorithm is increased. To remedy the shortcoming of Hopfield, we apply simulated annealing in the learning algorithm of Hopfield. This method is performed on Kahkeshan block cipher as a case study. Kahkeshan is a SP block cipher with 256 bits block/key size and 32 rounds SP transformation. Until now, no linear cryptanalysis is published for this cipher. In this paper, 8-round, 9-round and 10-round linear characteristic with the probability of 2^{-78}, 2^{-108} and 2^{-137} are obtained. The comparison of found characteristic with Hopfield and Boltzamann machine shows that applying the simulated annealing in the learning algorithm is obtained the better results.

Keywords

Block cipher, Kahkeshan cipher algorithm, Linear cryptanalysis, Hopfield neural network, Bolzamann Machine.

* استادیار گروه کامپیوتر، دانشکده مهندسی، دانشگاه فردوسی مشهد، GhaemiB@um.ac.ir

۱- مقدمه

تحلیل خطی یک روش بررسی روابط موجود بین تقریب خطی متن واضح، متن رمزشده و زیرکلیدها است که اولین با توسط ماتسویی[۴] مطرح شد. برای سهولت بررسی تئوری زیر کلیدها را مستقل در نظر می‌گیریم. در صورتیکه بین زیر کلیدها وابستگی وجود داشته باشد انجام تحلیل آسان تر می‌شود، اما بررسی تئوری آن مشکل‌تر خواهد بود. مهمترین بخش تحلیل خطی بدست آوردن بهترین مشخصه می‌باشد که سعی می‌شود، با توجه به ویژگی های اجزای داخلی و ساختار الگوریتم رمز و با شناسایی و بکار گیری آسیب پذیریها و نقاط ضعف آن، بهترین مشخصه را برای الگوریتم رمز بدست آوریم.

در این مقاله، از ابزار فلق۲ [۲] برای یافتن مشخصه های خطی استفاده شده است. در این ابزار برای یافتن مشخصه خطی مناسب در الگوریتم رمز قطعه‌ای، بهینه‌سازی با شبکه عصبی هاپفیلد، Simulated Annealing و ماشین بولتزمن استفاده شده است. برای این منظور، شبکه عصبی معادل یک تابع جانشینی، که براساس آن شبکه عصبی معادل هر تعداد دور از الگوریتم رمز قابل بیان می‌باشد، تعریف شده و الگوریتم های آموزش شبکه عصبی حاصل جهت یافتن جواب بهینه برای تابع هزینه بیان می‌شود. در[۳] شبکه عصبی هاپفیلد مورد توجه قرار گرفت. نقطه ضعف این شیوه بهینه سازی عدم حصول بهترین جواب در حل مسائل بزرگ است. در یافتن مشخصه های خطی، با افزایش تعداد دور الگوریتم رمز ، احتمال گیرکردن در بهینه‌های محلی در هنگام بهینه سازی افزایش می‌یابد. دراین مقاله، جهت کاهش این مشکل از شیوه‌های آموزش احتمالی و ایده Annealing استفاده کرده و با بکارگیری ماشین بولتزمن کارایی بیشتری بدست آمده است. لازم بذکر است با توجه به فضای بسیار بالای مشخصه‌های ممکن، بررسی کل فضا میسر نمی‌باشد و این مشخصه‌ها با بکارگیری شیوه‌های هوشمند بدست آمده است. لذا نمی‌توان ادعا کرد مشخصه‌هایی بهتر از آنچه در اینجا مطرح می‌شود وجود ندارد و ممکن است به شیوه‌ای دیگر و یا با صرف زمان بیشتر بتوان به مشخصه‌های بهتری دست یافت.

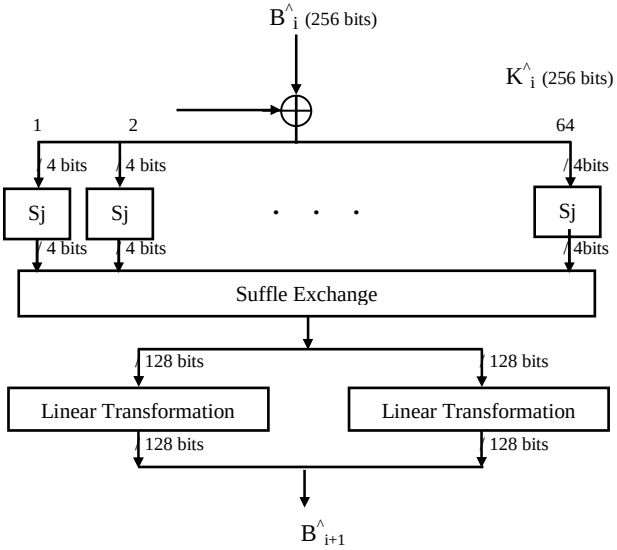
دراین مقاله ابتدا الگوریتم رمز کهکشان را معرفی کرده و سپس نحوه بکارگیری شبکه عصبی در یافتن مشخصه مناسب در الگوریتم رمز تشریح می شود. در انتها مشخصه خطی بدست آمده برای الگوریتم رمز کهکشان بیان و با مشخصه بدست آمده در [۳] مقایسه می‌شود.

۲- معرفی رمز قطعه ای کهکشان

این الگوریتم یکی از چهار الگوریتم رمز شرکت کننده در مسابقه "بررسی الگوریتم‌های رمز قطعه‌ای" می‌باشد[۱]، که با اقتباس از الگوریتم رمز سرپنت [۳] و توسعه آن طراحی شده است. در این الگوریتم از همان S-Boxهای سرپنت استفاده شده و تبدیلات اولیه و انتهایی با توجه به طول ورودی/خروجی ۲۵۶ طراحی شده است. برای بکارگیری تبدیل خطی سرپنت با ورودی ۱۲۸ بیتی در رمز کهکشان با ورودی ۲۵۶ بیتی از یک تبدیل جابجایی ضربدری استفاده کرده‌ایم.

در این الگوریتم از هشت S-Box با ورودی و خروجی ۴ بیت که با $S_0, \ldots, S_7$ مشخص می‌شوند، استفاده شده است ، بطوریکه در تابع دور iام (شکل۱)، $i \in \{0, \ldots, 31\}$، فقط از تکرار یک S-Box مشخص $(S_j, j \in \{0, \ldots, 7\})$ استفاده می‌شود که شماره آن(j) طبق رابطه مقابل مشخص می شود :

در دور i ام، $i \in \{0, \ldots, 31\}$ ، دو ورودی ۲۵۶ بیتی $B^{\wedge}_i$ و $K^{\wedge}_i$ با هم Xor شده و به ۶۴ عدد S-Box مشابه و مشخص اعمال می‌شوند. آنگاه بردار میانی ابتدا تحت تبدیل جابجایی ضربدری و سپس توسط تبدیل خطی تغییر یافته و $B^{\wedge}_1$ را تولید می‌کند.



شکل (۱): توابع دور $(Ri, 0 \le i \le 30)$ در رمز کهکشان

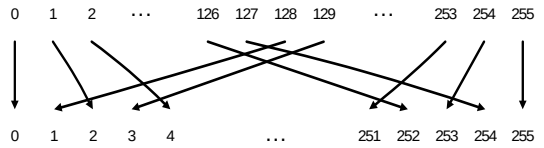
تابع دور مرحله آخر (R_{31}) اندکی با بقیه تفاوت دارد، بطوریکه پس از اعمال توابع جانشینی S_7 روی $B^{\wedge}_{31} \oplus K^{\wedge}_{31}$، بجای اعمال تبدیل جابجایی ضربدری و تبدیل خطی، بردار میانی حاصل با $K^{\wedge}_{32}$ بیت به بیت Xor شده و $B^{\wedge}_{32}$ تولید می‌شود. پس از اعمال نگاشت FP بر روی $B^{\wedge}_{32}$، متن رمز C بدست می‌آید.

در تبدیل خطی ورودی ۱۲۸ بیتی بصورت ۴ کلمه ۳۲ بیتی X_0، X_1، X_2، و X_3 درنظرگرفته شده و بصورت زیر با هم ترکیب می‌شوند:

$X_0 = X_0 <<< 13$
$X_2 = X_2 <<< 3$
$X_1 = X_0 \oplus X_1 \oplus X_2$
$X_3 = X_3 \oplus X_2 \oplus (X_0 <<3)$
$X_1 = X_1 <<<1$
$X_3 = X_3 <<<7$
$X_0 = X_0 \oplus X_1 \oplus X_3$
$X_2 = X_2 \oplus X_3 \oplus (X_1 <<7)$
$X_0 = X_0 <<<5$
$X_2 = X_2 <<<22$

که X<<<k بیانگر دوران X به اندازه k بیت به چپ و X<<k بیانگر انتقال X به اندازه k بیت به چپ و X⊕Y بیانگر Xor بیت به بیت X و Y است.

تبدیل جابجایی ضربدری مطابق شکل(۲) یک جایگشت منظم روی بیتهای ورودی اعمال کرده و خروجی را بدست می دهد. بطوریکه بیت اول و آخر(۲۲۵ام) را ثابت نگه داشته و در بقیه بیت‌ها، بیت i ام ورودی را به بیت 2i ام (در پیمانه ۲۵۵) خروجی می‌نگارد.توابع جانشینی بکارگرفته شده در الگوریتم رمز کهکشان در جدول(۱) آمده است.



شکل(۲): تبدیل جابجایی ضربدری

جدول (۱): توابع جانشینی در الگوریتم رمز کهکشان

Sbox# \ Input	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	3	8	F	1	A	6	5	B	E	D	4	2	7	0	9	C
1	F	C	2	7	9	0	5	A	1	B	E	8	6	D	3	4
2	8	6	7	9	3	C	A	F	D	1	E	4	0	B	5	2
3	0	F	B	8	C	9	6	3	D	1	2	4	A	7	5	E
4	1	F	8	3	C	0	B	6	2	5	4	A	9	E	7	D
5	F	5	2	B	4	A	9	C	0	3	E	8	D	6	7	1
6	7	2	C	5	8	4	6	B	E	9	1	F	D	3	A	0
7	1	D	F	0	E	8	2	B	7	4	C	A	9	3	5	6

۳- یافتن مشخصه خطی با استفاده از شبکه عصبی

دراین بخش شبکه عصبی معادل یک تابع جانشینی، که براساس آن شبکه عصبی معادل هر تعداد دور از الگوریتم رمز قابل بیان می‌باشد، تعریف می شود. در این شبکه عصبی تمایل احتمال تقریب خطی در الگوریتم رمز قطعه‌ای، توسط تابع هزینه مشخص می‌شود. بنابراین جهت یافتن مشخصه خطی مناسب لازم است تابع هزینه در شبکه عصبی حداقل گردد. در ادامه این بخش ابتدا شبکه عصبی معادل توابع جانشینی ارائه شده و نحوه بیان شبکه عصبی معادل هر تعداد دور از الگوریتم رمز ارائه می‌شود. سپس الگوریتم آموزش شبکه عصبی حاصل جهت یافتن جواب بهینه برای تابع هزینه بیان می‌گردد.

۳-۱- شبکه عصبی معادل توابع جانشینی

هر تابع جانشینی $S: \{0,1\}^m \rightarrow \{0,1\}^n$ توسط یک شبکه بازگشتی حاوی n+m نورون ارائه می‌شود، که m نورون اول آن متناظر بیتهای تقریب خطی ورودی تابع جانشینی می‌باشد که با Ii با اندیس‌های ۰ تا m-۱ نشان داده می‌شود، و n نورون بعدی متناظر بیتهای تقریب خطی خروجی تابع جانشینی می‌باشد که با Oi با اندیس‌های ۰ تا n-۱ نشان داده می‌شود. با توجه به مؤثر بودن هر بیت تقریب خطی ورودی Sbox در بیتهای تقریب خطی خروجی آن و نیز موثر بودن هر بیت تقریب خطی خروجی Sbox در بیتهای تقریب خطی ورودی آن، در این شبکه هر نورون به تمامی نورون‌های دیگر متصل می‌باشد.

اگر نورونهای I_j و O_k ، که $0 \le j \le m-1$ و $0 \le k \le n-1$ است ، بترتیب دارای مقادیر i_j و o_k باشد، مقدار تابع هزینه بصورت زیر محاسبه می‌شود, که تابع $LAT_S: \{0,1\}^m \times \{0,1\}^n \rightarrow R$ بیانگر توزیع تقریب خطی ورودی/خروجی در تابع جانشینی S می‌باشد و هر زوج تقریب خطی ورودی/خروجی $(X,Y) \in \{0,1\}^m \times \{0,1\}^n$ را به تمایل احتمال تقریب خطی خروجیY تحت تابع جانشینی S با فرض تقریب خطی ورودی X می‌نگارد.

$$Cost1_S(o_{n-1}, o_{n-2}, \ldots, o_1, o_0, i_{m-1}, i_{m-2}, \ldots, i_1, i_0)$$
$$= -Log_2\left(2LAT_S\left(i_{m-1}\times 2^{m-1} + i_{m-2}\times 2^{m-2} + \ldots + i_1\times 2 + i_0 \ , \ o_{n-1}\times 2^{n-1} + o_{n-2}\times 2^{n-2} + \ldots + o_1\times 2 + o_0\right)\right)$$

بنابر این تابع هزینه شبکه معادل تابع جانشینی S برای مقادیر دلخواه نورون ها بصورت $Cost_S : \{0,1\}^m \times \{0,1\}^n \rightarrow R$ خواهد بود :

$$Cost_S(O_{n-1}, O_{n-2}, O_{n-3}, \ldots, O_1, O_0, I_{m-1}, I_{m-2}, I_{m-3}, \ldots, I_1, I_0)$$
$$= \sum_{o_{n-1}=0}^{1} \sum_{o_{n-2}=0}^{1} \cdots \sum_{o_1=0}^{1} \sum_{o_0=0}^{1} \sum_{i_{m-1}=0}^{1} \sum_{i_{m-2}=0}^{1} \cdots \sum_{i_1=0}^{1} \sum_{i_0=0}^{1} \left(\prod_{j=0}^{m-1} \left(1 - i_j + (2\times i_j - 1)\times I_j\right) \right.$$
$$\left. \times \prod_{k=0}^{n-1} \left(1 - o_k + (2\times o_k - 1)\times O_k\right) \times Cost1_S(o_{n-1}, o_{n-2}, \ldots, o_1, o_0, i_{m-1}, i_{m-2}, \ldots, i_1, i_0) \right)$$

بعنوان مثال، تابع جانشینی S_q $(0 \le q \le 7)$ بکارگرفته شده در الگوریتم رمز کهکشان، یک تابع جانشینی ۴×۴ با تابع هزینه زیر است:

$$Cost_{Sq}(O_3, O_2, O_1, O_0, I_3, I_2, I_1, I_0) = \sum_{o_3=0}^{1} \sum_{o_2=0}^{1} \sum_{o_1=0}^{1} \sum_{o_0=0}^{1} \sum_{i_3=0}^{1} \sum_{i_2=0}^{1} \sum_{i_1=0}^{1} \sum_{i_0=0}^{1} \left(\prod_{j=0}^{m-1} \left(1 - i_j + (2\times i_j - 1)\times I_j\right) \right.$$
$$\left. \times \prod_{k=0}^{3} \left(1 - o_k + (2\times o_k - 1)\times O_k\right) \times Cost1_{S_q}(o_3, o_2, o_1, o_0, i_3, i_2, i_1, i_0) \right)$$

که: $Cost1_{S_q}(o_3, o_2, o_1, o_0, i_3, i_2, i_1, i_0) = -Log_2\left(2LAT_{S_q}\left(i_3\times 2^3 + i_2\times 2^2 + i_1\times 2 + i_0 \ , \ o_3\times 2^3 + o_2\times 2^2 + o_1\times 2 + o_0\right)\right)$

۳-۲- شبکه عصبی معادل تعداد دور دلخواه از رمز جانشینی-جایگشتی

جهت یافتن یک مشخصه k دوری در یک الگوریتم رمز قطعه‌ای با ساختار جانشینی-جایگشتی و اندازه ورودی/خروجی n بیت ، یک شبکه عصبی بازگشتی تک‌لایه با ۲×k×n نورون خواهیم داشت. جهت سهولت بیان توابع هزینه در شبکه حاصل، نورونها را در دسته های n تایی درنظر می‌گیریم که هر دسته مربوط به ورودی/خروجی توابع جانشینی در یک مرحله است. بطور دقیق تر دسته pام $(0 \le p \le 2k-1)$ را با بردار $\underline{N}_p = (N_{p\times n+n-1}, N_{p\times n+n-2}, N_{p\times n+n-3}, \ldots, N_{p\times n+1}, N_{p\times n}) \in \{0,1\}^n$ نشان می‌هیم که N_ℓ بیانگر نورون ℓام است. اگر p زوج باشد، بردار $\underline{N}_p$ نورون‌های متناظر ورودی تابع جانشینی دور rام است، که r = p/۲ و اگر p فرد باشد، بردار $\underline{N}_p$ نورون‌های متناظر خروجی تابع جانشینی دور r ام است، که r = ⌊p/۲⌋ .

۳-۳- الگوریتم آموزش شبکه عصبی

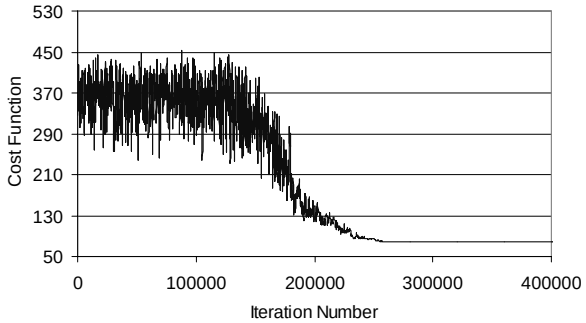
برای یافتن جواب بهینه برای تابع هزینه فوق از بهینه سازی با ماشین بولتزمن طبق الگوریتم زیر استفاده شد:

۱. مقداردهی اولیه دمای بهینه سازی: $T = T_0$
۲. مقداردهی اولیه نورون ها: مقدار نورون‌های متناظر با تقریب خطی دور میانی ممکن است توسط تحلیلگر تعیین شده و یا درحین روند بهینه سازی تعیین شود. مقدار بقیه نورون‌ها بطور بیقاعده مبتنی بر الگوریتم رمز مورد نظر مشخص می‌شود.
۳. تکرار گامهای ۳ تا ۶ برای c مرتبه
۴. انتخاب یک نورون بطور بیقاعده و تغییر مقدار آن.
۵. تنظیم دیگر نورون ها براساس تغییر جدید: این تنظیم با توجه به رابطه تبدیل خطی بین نورونهای متناظر با خروجی توابع جانشینی در یک دور و نورونهای متناظر با ورودی توابع جانشینی در دور بعد انجام می شود.
۶. محاسبه مقدار تابع هزینه با توجه به تغییر انجام شده.
۷. پذیرش وضعیت جدید، درصورتیکه رابطه $\Pr < \left(1+\exp\left(-\Delta E/T\right)\right)^{-1}$ برقرار باشد، که ΔE میزان تغییر مقدار تابع هزینه درصورت پذیرش مقدار جدید می باشد و Pr عددی در فاصله صفر و یک است، که بصورت بیقاعده تولید می شود..
۸. تغییر دمای بهینه سازی با ضریب d: $T=d\times T$
۹. بررسی شرط توقف و تکرار از گام ۲ درصورت عدم برقراری آن. شرط توقف، عدم پذیرش حداقل k مرتبه تغییر مقدار نورون ها در سه دمای متوالی روند بهینه‌سازی می باشد.

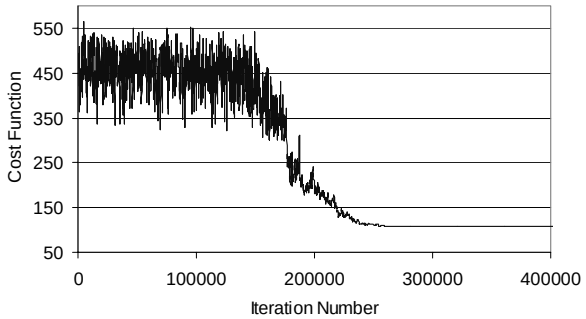
پارامترهای T_0، c، d و k بعنوان پارامترهای کنترلی بهینه‌سازی می باشد که بایستی به نحو مطوبی تعیین شود.

۴- مشخصه خطی بدست آمده برای کهکشان

برای یافتن مشخصه خطی در الگوریتم رمز کهکشان، پارامتر کنترلی بهینه‌سازی T_0، c، d و k بترتیب برابر مقادیر ۱۰۰۰۰۰، ۱۰۰، ۰٫۹۵ و ۱۰ درنظر گفته شد. بهترین مشخصه های خطی بدست آمده برای الگوریتم رمز کهکشان ۸ دوری، ۹ دوری و ۱۰ دوری بترتیب مطابق جدول (۲) تا جدول (۴) و با تمایل احتمال 2^{-78}، 2^{-108} و 2^{-137} است. هر سطر این جداول تقریب خطی ورودی و خروجی تبدیلات جانشینی در دور مربوطه را نشان می دهد. با توجه به وجود تبدیل خطی بکارگرفته شده بعد از اعمال تبدیل جانشینی در هر دور رمز کهکشان، با اعمال تبدیل خطی بر تقریب خطی خروجی در هر دور تقریب خطی ورودی دور بعد بدست می‌آید. روند تغییرات تابع هزینه تا رسیدن به همگرایی در تعیین این مشخصه ها بترتیب در شکل (۳) تا شکل (۵) آمده است.



شکل (۳): تغییرات مقدار تابع هزینه در روند بهینه سازی در یافتن مشخصه خطی برای رمز کهکشان ۸ دوری تا رسیدن به همگرایی



شکل (۴): تغییرات مقدار تابع هزینه در روند بهینه سازی در یافتن مشخصه خطی برای رمز کهکشان ۹ دوری تا رسیدن به همگرایی

جدول (۲): مشخصه خطی برای الگوریتم رمز کهکشان ۸ دوری

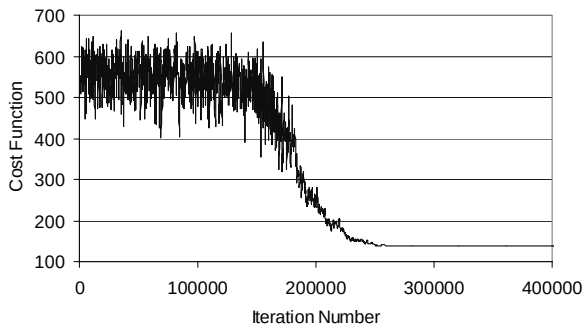
شماره دور		تقریب خطی ورودی/خروجی توابع جانشینی	تمایل احتمال
۱	ورودی	084000000A0BB000B00D00B0100000000006B007F00B00000000000000E00000	$۲^{-۱۶}$
	خروجی	05B0000003081000100400803000000000006800140080000000000000800000	
۲	ورودی	0006000000000044000000000800050020000A0000000004000000000050000	$۲^{-۱۶}$
	خروجی	00030000000000CC0000000003000800200008000000000200000000000080000	
۳	ورودی	0000004040000000040000000000000000000000000000010000800000000000	$۲^{-۱۱}$
	خروجی	0000002040000000030000000000000000000000000000010000800000000000	
۴	ورودی	0000000000010000000000000000000000000000000000040000000000000000	$۲^{-۵}$
	خروجی	0000000000030000000000000000000000000000000000080000000000000000	
۵	ورودی	0400000000000000000000000000000000000000000000000000000000000000	$۲^{-۳}$
	خروجی	0400000000000000000000000000000000000000000000000000000000000000	
۶	ورودی	0200000000000000000000000000010000000000000000000000000000000000	$۲^{-۳}$
	خروجی	0E0000000000000000000000000009000000000000000000000000000000000000	
۷	ورودی	0200044000000000000000011000890000000000000000100008010000200000000	$۲^{-۲۱}$
	خروجی	0100044000000000000000007C000C100000000000000010000404000A0000000	
۸	ورودی	00A0100102000000000000000000000000000000000A0041000036000000000	$۲^{-۱۰}$
	خروجی	084000000A0BB000B00D00B0100000000006B007F00B00000000000000E00000	
		تمایل احتمال کل مشخصه	$۲^{-۷۸}$

جدول (۳): مشخصه خطی برای الگوریتم رمز کهکشان ۹ دوری

شماره دور		تقریب خطی ورودی/خروجی توابع جانشینی	تمایل احتمال
۱	ورودی	E8848000008800B00000A080000800B00D000000000A5E00B0000B00B0001070	$۲^{-۲۲}$
	خروجی	855BF00000FF00200000E0500005001002000000000003A8008000020080003010	
۲	ورودی	0FF331222600000001000080000000005F000000050001000A000000A400000000	$۲^{-۲۶}$
	خروجی	041A95422D000000050000C0000000810000000800050001000000430000000	
۳	ورودی	0A00B000008E000000204400000000000020000000000000000000004000100000	$۲^{-۲۰}$
	خروجی	0D002000008100000020 4C0000000000006000000000000000000002000100000	
۴	ورودی	00A0000000000000001000000000000000010000000000000000400000000000	$۲^{-۹}$
	خروجی	00C0000000000000003000000000000000020000000000000000800000000000	
۵	ورودی	0000000000000040000000000000000000000000000000000400000000000000	$۲^{-۵}$
	خروجی	0000000000000C0000000000000000000000000000000000200000000000000	
۶	ورودی	0000004000000000000000000000000000000000000000000000000000000000	$۲^{-۳}$
	خروجی	0000004000000000000000000000000000000000000000000000000000000000	
۷	ورودی	0000001000020040000000000000000000000000000000000000000000000000	$۲^{-۷}$
	خروجی	0000001000090020000000000000000000000000000000000000000000000000	
۸	ورودی	400000010010200C000000A0000000000000000000000000000000000000000	$۲^{-۱۱}$
	خروجی	C0000001002010020000001000000000000000000000000000000000000000	
۹	ورودی	00040000000000A004001000E814000200000001000020040000000000000000	$۲^{-۱۳}$
	خروجی	000F0000000000E00F00E0008FEF000E0000000E0000E00F0000000000000000	
		تمایل احتمال کل مشخصه	$۲^{-۱۰۸}$

جدول (۴): مشخصه خطی برای الگوریتم رمز کهکشان ۱۰ دوری

شماره دور		تقریب خطی ورودی/خروجی توابع جانشینی	تمایل احتمال
۱	ورودی	9840360E420CAF070E05C0B0000B800E0900000D084001B00000F0000F700EB0	$۲^{-۳۱}$
	خروجی	9FB06604FE04E401080C4080000170080900000407F003800000A00004100880	
۲	ورودی	02B1122007000000010000000AA4000000005080000010000C000000A00000000	$۲^{-۲۱}$
	خروجی	0719574008000000050000008140000000080A0000050000400000010000000	
۳	ورودی	0A00B000002E000004400400000000000020000000000000000002004000000000	$۲^{-۲۰}$
	خروجی	0D0020000081000003400400000000000060000000000000000008002000000000	
۴	ورودی	00A000000000000000100000000000000000A000000000000000000400000000000	$۲^{-۹}$
	خروجی	00C000000000000000300000000000000000200000000000000000800000000000	
۵	ورودی	0000000000000004000000000000000000000000000000000000400000000000000	$۲^{-۵}$
	خروجی	0000000000000000C000000000000000000000000000000000002000000000000000	
۶	ورودی	0000004000000000000000000000000000000000000000000000000000000000000	$۲^{-۳}$
	خروجی	0000004000000000000000000000000000000000000000000000000000000000000	
۷	ورودی	0000001000020040000000000000000000000000000000000000000000000000000	$۲^{-۷}$
	خروجی	0000009000090020000000000000000000000000000000000000000000000000000	
۸	ورودی	5000080100102008000000A000000000000000000000000000000000000000000	$۲^{-۱۵}$
	خروجی	1000010900201002000000400000000000000000000000000000000000000000000	
۹	ورودی	20500108002004A000000000E005000000000010000200400000000000000000	$۲^{-۲۰}$
	خروجی	50100E0500E00C600000000004009000000000003000020020000000000000000	
۱۰	ورودی	6000E000100002E0450040904002000000000000000000000A0140002000000000	$۲^{-۱۶}$
	خروجی	D000A000B0000EA0FF00F0F0F00E000000000000000000000E0BF000E000000000	
		تمایل احتمال کل مشخصه	$۲^{-۱۳۷}$



شکل (۵): تغییرات مقدار تابع هزینه در روند بهینه سازی در یافتن مشخصه خطی برای رمز کهکشان ۱۰ دوری تا رسیدن به همگرایی

۵- جمع بندی

در این مقاله، نحوه بکارگیری ماشین بولتزمن برای یافتن مشخصه های خطی در الگوریتم رمز قطعه‌ای معرفی شد. برای این منظور، شبکه عصبی معادل یک تابع جانشینی، که براساس آن شبکه عصبی معادل هر تعداد دور از الگوریتم رمز قابل بیان می‌باشد، تعریف شده و الگوریتم آموزش شبکه عصبی حاصل جهت یافتن جواب بهینه برای تابع هزینه بیان شد.

بهترین مشخصه خطی بدست آمده برای این الگوریتم رمز کهکشان، مشخصه های ۸، ۹ و ۱۰ دوری بتریب با تمایل احتمال $۲^{-۷۸}$، $۲^{-۱۰۸}$ و $۲^{-۱۳۷}$ است ، درحالیکه بهترین مشخصه های خطی بدست آمده برای این الگوریتم رمز کهکشان با بکارگیری هاپفیلد[۳]، دارای تمایل احتمال $۲^{-۷۸}$، $۲^{-۱۱۴}$ و $۲^{-۱۴۶}$ می باشد. یعنی استفاده از ایده تابکاری شبیه سازی شده باعث افزایش کارایی شبکه عصبی شده و کارآیی ماشین بولتزمن بیشتر از هاپفیلد است. البته با بررسی دقیق تر می توان این مقایسه را بهتر انجام داد.

از جنبه‌های مختلفی می توان پژوهش گزارش شده در این مقاله را ادامه داد، در زیر چند مورد از آن ذکر می گردد:

۱-در این مقاله مقادیر مناسب برای پارامترهای شبکه عصبی با انجام آزمایش‌های مختلف تعیین شده است. با توسعه شبکه عصبی می توان بهترین مقادیر این پارامترها را در طی روند بهینه سازی بدست آورد.

۲- شیوه مطرح شده در این مقاله را برای یافتن مشخصه های مطلوب برای الگوریتم‌های رمز قطعه‌ای دیگر بکارگرفت.

۳- شیوه‌های مختلف بهینه سازی هوشمند مانند الگوریتم‌های ژنتیک، اجتماع مورچگان، شبکه‌های عصبی و . . . را در تعیین مشخصه خطی مناسب در الگوریتم های رمز قطعه‌ای بکارگرفته و عملکرد آنها را به لحاظ کارایی و کارآمدی مقایسه کرد.

مراجع

[۱] ع.قائمی‌بافقی، " الگوریتم رمزقطعه‌ای کهکشان"، *مستندات الگوریتم های نام‌نویسی شده در مسابقه بررسی الگوریتم های رمز قطعه‌ای*، انجمن رمز ایران،۱۳۸۰.

[۲] ع.قائمی‌بافقی، "فلق۲: ابزار تحلیل خطی الگوریتم های رمز قطعه ای" ، ۱۳۸۵.

[۳] ع.قائمی‌بافقی، " یافتن مشخصه خطی در الگوریتم های رمز قطعه ای با استفاده از شیوه بهینه سازی شبکه عصبی هاپفیلد" ، دوازدهمین کنفرانس سالانه انجمن کامپیوتر، ۱۳۸۵.

[4] R.Anderson , E.Biham , and L.Knudsen , "Serpent : A Proposal for the Advanced Encryption Standard " , *NIST Proposal* , 1998.

[5] M. Matsui, "Linear Cryptanalysis Method for DES Cipher", Advances in Cryptology - EUROCRYPT '93 (Lecture Notes in Computer Science no. 765), Springer-Verlag, pp. 386-397, 1994.