ارائهٔ یک حملهٔ متن اصلی معلوم[1] به مولدهای شبه‌تصادفی آشوبی[2]

محمد دخیل علیان
دانشگاه صنعتی اصفهان
دانشکدهٔ برق و کامپیوتر
تلفن: (۰۳۱)۸۹۱۲۴۵۰
Md-alian@iut.cc.ac.ir

محمدرضا عارف
دانشگاه صنعتی شریف
دانشکدهٔ برق
تلفن: (۰۲۱)۸۹۱۲۱۱۲
Aref@awww.dci.co.ir

بابک صادقیان
دانشگاه صنعتی امیر کبیر
دانشکدهٔ کامپیوتر
تلفن: (۰۲۱)۶۱۳۹۴۳۳
Basadegh@ce.aku.ac.ar

چکیده: رفتار تصادف‌گونه و نامنظم دنباله‌های تولید شده توسط نگاشتهای آشوبی جهت تولید کلید اجرایی در یک سیستم رمز پی‌درپی مورد بررسی قرار گرفته و اخیراً مولدهایی بر این اساس طرح شده‌اند[1]. در این مقاله با توجه روشهای ارایه شده در [1] و در حالت خاصی که از هر مؤلفهٔ دنبالهٔ تولید شده توسط نگاشت یک بیت استخراج شده و حالت اولیهٔ نگاشت به عنوان کلید مورد در نظر گرفته شده باشد، یک حملهٔ از نوع متن اصلی معلوم ارایه می‌شود. حملهٔ ارایه شده قابل اعمال به نگاشتهای یک بعدی است که در این مقاله تنها نگاشت مثلثی مورد تحلیل قرار گرفته است. نتایج این حمله نشان می‌دهد دنباله‌های کلید اجرایی که بیتهای آن از ارقام با وزن کمتر خروجی نگاشتها حاصل شده باشند از امنیت بالاتری نیز برخوردار می‌باشند.

کلمات کلیدی: آشوب، دنباله‌های شبه‌تصادفی، سیستمهای رمز پی‌درپی، متغیرهای تصادفی.

۱- مقدمه

آشوب در سیستمهای با معادلات غیرخطی در حالتهای یک بعدی و چندبعدی بروز می‌کند و غیرخطی بودن یک شرط لازم برای ظهور این رفتار در سیستمها می‌باشد([۲]-[۳]). در[۱] جهت تولید دنباله‌های کلید اجرایی از سیستمهایی با معادلات دیفرنس[3] که توسط نگاشتهایی به صورت (۱) قابل بیان هستند:

$$x_{n+1} = g(x_n) \qquad (۱)$$

$g(.)$ یک نگاشت غیرخطی است و x_n نیز در حالت کلی می‌تواند یک بردار باشد. از آنجا که پیاده‌سازی نگاشتهای یک بعدی ساده بوده و ثانیا رفتار آنها به قدر کافی پیچیده می‌باشد، بدین جهت این نگاشتها برای هدف مورد نظر انتخاب گردیدند. نگاشتهایی که در این مقاله مورد بررسی قرار می‌گیرند، نگاشتهایی هستند که روی یک ناحیه پیوسته

---

1 - *Known plain text*

2 - *Chaotic*

3 - *Difference*

محدود از اعدادحقیقی دارای رفتار آشوبی می‌باشند. فرض کنید دنبالهٔ اعداد $x_n,\cdots,x_2,x_1,x_0$ توسط نگاشت(۱) بدست آمده باشد که در این دنباله x_i برابر $x_i = g(g(\cdots g(x_0)\cdots))$ می‌باشد.
تحت شرایط خاصی رفتار دنبالهٔ مورد نظر نامنظم واصطلاحاً آشوبی خواهد شد. از جملهٔ این شرایط حساسیت نسبت به حالت اولیه، "تراگذار توپولوژیکی [1] بودن و چگال [2] بودن نقاط تناوبی درناحیهٔ تعریف می‌باشد([۳]و[۴]).

۲- حملهٔ متن اصلی معلوم به دنبالهٔ کلید اجرایی

امنیت یک دنبالهٔ کلید اجرایی وابسته به میزان سختی دستیابی به کلید اصلی مولد می‌باشد. در این بخش فرض می‌کنیم دنباله‌ای از متن رمز شده و متن‌اصلی متناظر با آن در اختیار باشد. این فرض معادل داشتن دنباله‌ای از کلید اجرایی با طول محدود می‌باشد .لذا سعی خواهیم نمود با داشتن طول محدودی از دنبالهٔ کلید اجرایی(دنبالهٔ باینری تولید شده توسط نگاشت آشوبی) به کلید اصلی سیستم دست پیدا کنیم. فرض ما در این حمله بر این اساس می‌باشد که از هر مؤلفهٔ دنبالهٔ تولید شده توسط نگاشت $g(x)$ تنها یک بیت استخراج شده باشد وحالت اولیهٔ نگاشت به عنوان کلید اصلی مورد نظر مخفی باشد. در این حالت اگر x_i مؤلفهٔ i‌ام دنبالهٔ $x_n,\cdots,x_2,x_1,x_0$ (دنباله‌ای با اعداد اعشاری بین صفر و یک در مبنای a) باشد در این صورت بیت i‌ام دنبالهٔ کلید اجرایی $(s_0,s_1,\cdots)$ به صورت زیر بدست می‌آید[۱]:

$$s_i = \left[2a^{l-1}x_i\right] mod\, 2 \quad , \quad l=1,2,\ldots \;,\; i=0,1,2,\ldots \qquad (۲)$$

([ ] نشان دهندهٔ جزء صحیح می‌باشد.)

در [۱] نشان داده شده است که خواص آماری چنین دنباله‌هایی،خصوصاً هنگامی که بیت مربوطه از رقم با وزن کمتر مؤلفه‌های دنبالهٔ آشوبی استخراج شده باشد بسیار مطلوب می‌باشد. علاوه بر این اگر حداکثر تعداد بیت استخراج شده از هر مؤلفه کمتر یا مساوی نمای لیاپانف (با در نظر گرفتن لگاریتم مبنای ۲) نگاشت باشد، اطلاعات متقابل میان بیتهای دنباله حداقل خواهد شد .در این مقاله نشان می‌دهیم هنگامیکه دقت محاسبات در پیاده سازی محدود باشد، دستیابی به کلید اصلی مولد(حالت اولیه) امکان پذیر می‌باشد.

۳- روند نمای اجرای حمله

حالت اولیه نگاشت را در فاصلهٔ(۰،۱) در نظر می‌گیریم. بنابر این سعی بر این است که با داشتن دنبالهٔ باینری $b_0,b_1,\cdots,b_n$ حالت اولیهٔ نگاشت (x_0)را بدست آوریم. ابتدا ساده‌ترین حالت را در نظر می‌گیریم. فرض کنید، دنبالهٔ باینری از رابطهٔ(۲) وبه ازای $l=1$ بدست آمده باشد. یعنی:

$$b_i = \left[2x_i\right] mod\, 2 \quad , \quad i=0,1,2,\ldots,n \qquad (۳)$$

بنابر این با مشاهدهٔ b_n می‌توان محدوده‌ای که x_n در آن قرار دارد را به صورت زیر بدست آورد:

$$\frac{b_n}{2} \le x_n < \frac{b_{n+1}}{2} \qquad (۴)$$

1 Topologically trnsitive

2 - Dense

با مشاهدهٔ b_{n-1} محدوده‌ای مشابه با(۴) برای x_{n-1}نیز وجود دارد، اما با توجه به اینکه x_nباید حتما در محدودهٔ بیان شدهٔ(۴) قرار گیرد، بخشی از این فاصله نمی‌تواند قابل قبول باشد و بنابر این محدودهٔ تعیین شده برای x_{n-1}محدودتر از(۴) می‌گردد. با مشاهدهٔ بیتهای دیگر می‌توان محدوده‌های کوچکتری را برای x_{n-2} و... تعیین نمود .سرانجام پس از تعداد متناهی از دنبالهٔ باینری مورد نظر، کرانهای سمت راست و چپ محدودهٔ مورد نظر یکسان(همگرا) می‌شود. به عبارت دیگر حالت نگاشت مشخص می‌گردد.

هنگامیکه مقدار l بزرگتر از یک باشد، با مشاهدهٔ آخرین بیت دنباله (b_n) و با توجه به رابطهٔ(۲) چندین محدوده برای x_n وجود خواهد داشت. اگر $b_n = 0$ باشد نتیجه می‌گیریم که x_n می‌تواند در محدوده‌های زیر قرار گیرد :

$$0 < x_n < 0.5a^{1-l} \text{ یا } a^{1-l} \leq x_n < 1.5a^{1-l} \text{ یا } \ldots \; 1-a^{1-l} \leq x_n < 1-0.5a^{1-l} \tag{۵}$$

و اگر $b_n = 1$باشد، این محدوده‌ها به‌صورت زیر می‌باشند :

$$0.5a^{1-l} \leq x_n < a^{1-l} \text{ یا } 1.5a^{1-l} \leq x_n < 2a^{1-l} \text{ یا } \ldots \; 1-0.5a^{1-l} \leq x_n < 1 \tag{۶}$$

بنابر این a^{l-1}محدوده، برای x_n وجود خواهد داشت. پس از آن با مشاهدهٔ محدوده‌های مشابهی برای x_{n-1}می‌توان در نظر گرفت ولی با توجه به محدوده‌هایی که برای x_nتعیین شده است، بخشی از فواصل نمی‌تواند قابل قبول باشد و عملاً فواصل مورد نظر برای x_{n-1}محدودتر از فواصل مذکور در (۵) و (۶) می‌باشد. چگونگی محدودتر شدن فواصل توسط نگاشت، تعیین می‌گردد. در این روش پس از مشاهدهٔ بیتهای $b_n, b_{n-1}, \cdots$سرانجام فواصل تعیین شده به سمت یک نقطه همگرا می‌شوند، وبنابر این در نهایت a^{l-1}نقطه به عنوان جواب وجود خواهد داشت که حداقل یکی از آنها حالت اولیهٔ نگاشت خواهد بود.

بطور کلی برای دستیابی به کلید می‌توان روند تمای مشخص شده در شکل (۱)را در نظر گرفت. برای انجام حملهٔ مورد نظر فرض براین است که بیتهای $b_0, b_1, \cdots, b_n$ در اختیار می‌باشند. بنابر این ابتدا فاصلهٔ[۰،۱] را به a^{l-1} زیر فاصلهٔ مساوی تقسیم می‌کنیم. پس از آن در یک روش تکراری بیتها را یکی یکی از انتهای دنباله در نظر گرفته ومحدودهٔ مؤلفه‌ای که آن بیت را تولید نموده است را مشخص می‌کنیم. در بعضی حالات ممکن است برای تعیین محدودهٔ x_iتوسط b_i ومحدودهٔ x_{i+1}ابهام وجود داشته باشد. به عبارت دیگر با داشتن b_iنتوان محدودهٔ جدیدی برای x_iتعیین نمود. درچنین حالتی استفاده از بیتهای بعدی یعنی $b_{i-1}, b_{i-2}, \cdots$ می‌تواند مؤثر واقع شود. با تعیین فواصل مشخص شده برای x_iدر هر مرحله و محدود شدن ونهایتا همگرا شدن آنها به سمت a^{l-1} نقطه، حالت اولیهٔ نگاشت قابل تعیین می‌باشد. در انتها تعداد a^{l-1}جواب(نه لزوما متفاوت) بدست می‌آید که باید بررسی نمود که کدام جواب قابل قبول هستند. برای این منظور، می‌توان با قراردادن یک یک جوابها در الگوریتم تولید دنبالهٔ باینری، بررسی نمود که آیا علاوه بر تولید بیتهای $b_0, b_1, \cdots, b_n$بیتهای $b_{n+1}, b_{n+2}, \cdots$ نیز با دنبالهٔ باینری کلید اجرایی برابر هستند یا خیر .در صورت تطابق کامل میان دنبالهٔ کلید اجرایی و دنبالهٔ باینری تولید شده، جواب مورد آزمایش بعنوان حالت اولیه انتخاب خواهد شد و در غیر این صورت از آن صرف‌نظر می‌شود .در بخش بعد چگونگی اعمال این حمله به

دنباله‌های تولید شده توسط نگاشت مثلثی مورد بررسی قرار می‌گیرد. در مورد نگاشتهای دیگر نیز این حمله قابل استفاده می‌باشد[۱].

۴- نگاشت مثلثی

ابتدا دنبالهٔ باینری تولید شده توسط نگاشت مثلثی را مورد بررسی قرار داده و سعی خواهیم نمود جزییات یافتن حالت اولیهٔ این نگاشت را بیان کنیم. نگاشت مثلثی را به‌صورت زیر در نظر می‌گیریم :

$$g(x) = \begin{cases} cx & 0 \le x < 0.5 \\ c(1-x) & 0.5 \le x < 1 \end{cases} \quad , \quad c = 2 - \varepsilon \tag{۷}$$

(ε مقدار بسیار کوچک در حد دقت محاسبات می‌باشد)

ابتدا به ازای $i=1$ مسأله را مورد تحلیل قرار می‌دهیم. برای x_n کلیه مقادیر در فاصلهٔ(۰،۱) می‌توانند بعنوان جواب در نظر گرفته شود ولی با ملاحظهٔ b_n این فاصله به‌صورت رابطهٔ(۴)محدود می‌شود و در واقع نیمی از حالات ممکن کنار گذاشته می‌شود. اگر فرض کنیم $b_{n-1} = 0$ باشد بنابراین x_{n-1} در محدودهٔ [۰،۰/۵] قرارداشته است و لزوماً x_n از رابطهٔ $x_n = cx_{n-1}$ باید بدست آمده باشد، بنابراین:

$$\frac{b_n}{2} \le cx_{n-1} \le \frac{b_n+1}{2} \Rightarrow \frac{b_n}{2c} \le x_{n-1} \le \frac{b_n+1}{2c} \tag{۸}$$

اگر $b_{n-1} = 1$ باشد بنابر این x_{n-1} در محدودهٔ [۰/۵،۱] قرار داشته و لزوما از رابطهٔ $x_n = c(1-x_{n-1})$ بدست آمده است، بنابر این x_{n-1} باید در محدودهٔ زیر واقع شود :

$$\frac{b_n}{2} \le c(1-x_{n-1}) \le \frac{b_n+1}{2} \Rightarrow 1-\frac{b_n+1}{2c} \le x_{n-1} \le 1-\frac{b_n}{2c} \tag{۹}$$

با فرض $b_{n-1} = 0$، اگر $b_{n-2} = 0$ باشد، محدودهٔ x_{n-2} بصورت زیر بدست می‌آید :

$$\frac{b_n}{2c} \le cx_{n-2} \le \frac{b_n+1}{2c} \Rightarrow \frac{b_n}{2c^2} \le x_{n-2} \le \frac{b_n+1}{2c^2} \tag{۱۰}$$

واگر $b_{n-2} = 1$ باشد محدودهٔ x_{n-2} به صورت (۱۱) بدست می‌آید :

$$\frac{b_n}{2} \le c(1-x_{n-1}) \le \frac{b_n+1}{2} \Rightarrow 1-\frac{b_n+1}{2c} \le x_{n-1} \le 1-\frac{b_n}{2c} \tag{۱۱}$$

بطور کلی اگر در مرحلهٔ مشاهدهٔ b_{n-i}، محدودهٔ تعیین شده برای x_{n-i+1} به صورت $A \le x_{n-i+1} \le B$ باشد، دراین صورت محدودهٔ x_{n-i} به ازای $b_{n-i} = 0$ و $b_{n-i} = 1$ به ترتیب برابر (۱۲)و (۱۳) خواهد شد.

$$A \le cx_{n-i+1} \le B \Rightarrow \frac{A}{c} \le x_{n-i} \le \frac{B}{c} \tag{۱۲}$$

$$A \le c(1-x_{n-i}) \le B \Rightarrow 1-\frac{B}{c} \le x_{n-i} \le 1-\frac{A}{c} \tag{۱۳}$$

خواهد شد .بنابر این اگر طول فاصلهٔ تعیین شده در مرحلهٔ i-1 برابر B-A باشد، طول فاصلهٔ تعیین شده در مرحلهٔ iام برابر $\frac{B-A}{c}$ خواهد شد. به عبارت دیگر کران سمت چپ و راست محدوده‌ها در هر مرحله به یکدیگر نزدیک شده ونهایتاً به‌سمت یک نقطه همگرا می‌شوند. بطور کلی پس از بررسی بیت b_{n-i} طول فاصلهٔ تعیین شده برای x_{n-i} برابر $\frac{1}{2c^i}$ خواهد شد. بنابراین با توجه به محدود بودن دقت محاسبات اگر طول دنبالهٔ باینری مورد تحلیل(n+1) از حدی بزرگتر باشد، به حالت اولیهٔ دنباله خواهیم رسید. اگر مؤلفه‌های نگاشت بصورت اعداد اعشاری

در مبنای a با دقت r رقم اعشار باشند دراین صورت :

$$\frac{1}{2c^{n}} < a^{-r} \Rightarrow n > \frac{r - \log_a 2}{\log_a c} \qquad (۱۴)$$

به عنوان نمونه اگر مؤلفه‌های دنبالهٔ آشوبی با دقت بیست رقم اعشار در مبنای ده حاصل شده باشد، با توجه به(۱۴) ومقدار $c = 2 - \varepsilon$ با داشتن تنها ۶۶ بیت از دنبالهٔ کلید اجرایی می‌توان به حالت اولیه دست یافت.

حال اگر $l > 1$ باشد، ابتدا برای x_n کلیهٔ مقادیر در فاصله[۰،۱] می‌تواند به عنوان جواب در نظر گرفته شود ولی با ملاحظهٔ b_n این فاصله به a^{l-1} فاصله نظیر(۵) و (۶) محدود می‌شود، به عبارت دیگر نیمی از حالات ممکن در فواصل مختلف کنار گذاشته می‌شود. با مشاهدهٔ b_{n-1} باید محدوده‌های مجاز برای x_{n-1} را بدست آورد. برای این منظور یک یک فواصل بدست آمده برای x_n را باید با توجه به b_{n-1} مورد بررسی قرار داد. به عنوان نمونه اگر x_n در محدودهٔ $\frac{t}{2}a^{1-l} < x_n < \frac{(t+1)}{2}a^{1-l}$ قرار گرفته باشد، دو محدوده برای x_{n-1} وجود دارد تا در تکرار بعدی به این فاصله ختم شود. این دو محدوده عبارتند از :

$$\frac{t}{2}a^{1-l} < cx_{n-1} < \frac{(t+1)}{2}a^{1-l} \Rightarrow \frac{t}{2c}a^{1-l} < x_{n-1} < \frac{(t+1)}{2c}a^{1-l} \qquad (۱۵)$$

$$\frac{t}{2}a^{1-l} < c(1 - x_{n-1}) < \frac{(t+1)}{2}a^{1-l} \Rightarrow 1 - \frac{(t+1)}{2c}a^{1-l} < x_{n-1} < 1 - \frac{t}{2c}a^{1-l} \qquad (۱۶)$$

حال با توجه به مقدار b_{n-1} که توسط رابطهٔ $\left[2a^{l-1}x_{n-1}\right] \bmod 2$ بدست می‌آید، محدوده یامحدوده‌هایی از فواصل(۱۵) و (۱۶) که با b_{n-1} تعارض ندارند را انتخاب می‌نماییم. به عنوان نمونه اگر c برابر ۲ باشد، یکی از دو محدودهٔ(۶-۳۵) یا (۶-۳۶) برای x_{n-1} قابل قبول خواهد بود، زیرا در دو محدودهٔ مذکور اگر یکی $b_{n-1} = 0$ را تولید کند، محدودهٔدیگر $b_{n-1} = 1$ را تولید خواهد نمود. بنابر این با این فرض درهر مرحله تعداد فاصله‌ها همان a^{l-1} فاصله باقی خواهد ماند. در هر مرحله از مرحله‌های اجرای الگوریتم فواصل بدست آمده نسبت به فواصل قبلی دارای طول کمتری بوده و سر انجام کرانهای سمت راست و چپ نقاط به سمت یک عدد همگرا می‌گردد و حداقل یکی از جوابهای بدست آمده حالت اولیهٔ مورد نظر خواهد بود. جدول (۱) نتایج بدست آمده‌از یک دنبالهٔ نمونه با حالت اولیهٔ $x_0 = 0.513468121385319$ را نشان می‌دهد.از آنجاکه $l=2$ و $a=10$ بوده است، نهایتاً ده جواب بدست آمده است که x_0 یکی از آنها می‌باشد.

جدول ۱: نتایج حاصل از اجرای الگوریتم حمله جهت دستیابی به حالت اولیهٔ نگاشت مثلثی به ازای $c = 2 - 10^{-18}$ و دنبالهٔ معلوم B^{64}

$B^{64} = 0110001111000011001001010001101010010011001010111111001101010101001$

$x_0^1 = 0.248459897700756$	$x_0^6 = 0.14433491876794$
$x_0^2 = 0.8021122611340109$	$x_0^7 = 0.724613973713210$
$x_0^3 = 0.0401950441927839$	$x_0^8 = 0.433075170533277$
$x_0^4 = 0.909232509214003$	$x_0^9 = 0.617494463269499$
$x_0^5 = 0.328839583875101$	$x_0^{10} = 0.513468121385319$

آنچه مشخص است، این می‌باشد که با افزایش l تعداد محدوده‌های مورد نظر به صورت نمایی (a^{l-1}) افزایش می‌یابد و بنابراین حجم حافظهٔ مورد نیاز(متناسب با a^{l-1}) افزایش می‌یابد. علاوه بر این حجم محاسبات نیز به صورت نمایی یعنی متناسب a^{l-1}افزایش می‌یابد. البته با توجه به کوچک شدن طول فاصله‌ها به نسبت $\frac{1}{c}$طول دنبالهٔ باینری مورد نیاز کاهش می‌یابد، مشابه با رابطهٔ(۱۴) می‌توان طول دنباله را بصورت زیر بدست آورد:

$$\frac{a^{1-l}}{2c^{n}} \prec a^{-r} \;\Rightarrow\; n \succ \frac{r+1-l+\log_a 2}{\log_2 c} \quad , \quad c = 2-\varepsilon \qquad (۱۷)$$

بنابر این طول دنبالهٔ مورد نیاز برای بدست آوردن حالت اولیه به صورت خطی با افزایش l کاهش می‌یابد. اگر نگاشت مثلثی را بصورت رابطهٔ(۱۸) مورد استفاده قرار دهیم. به ازای l=1 براحتی می‌توان حالت اولیهٔ دنباله را پیدا نمود

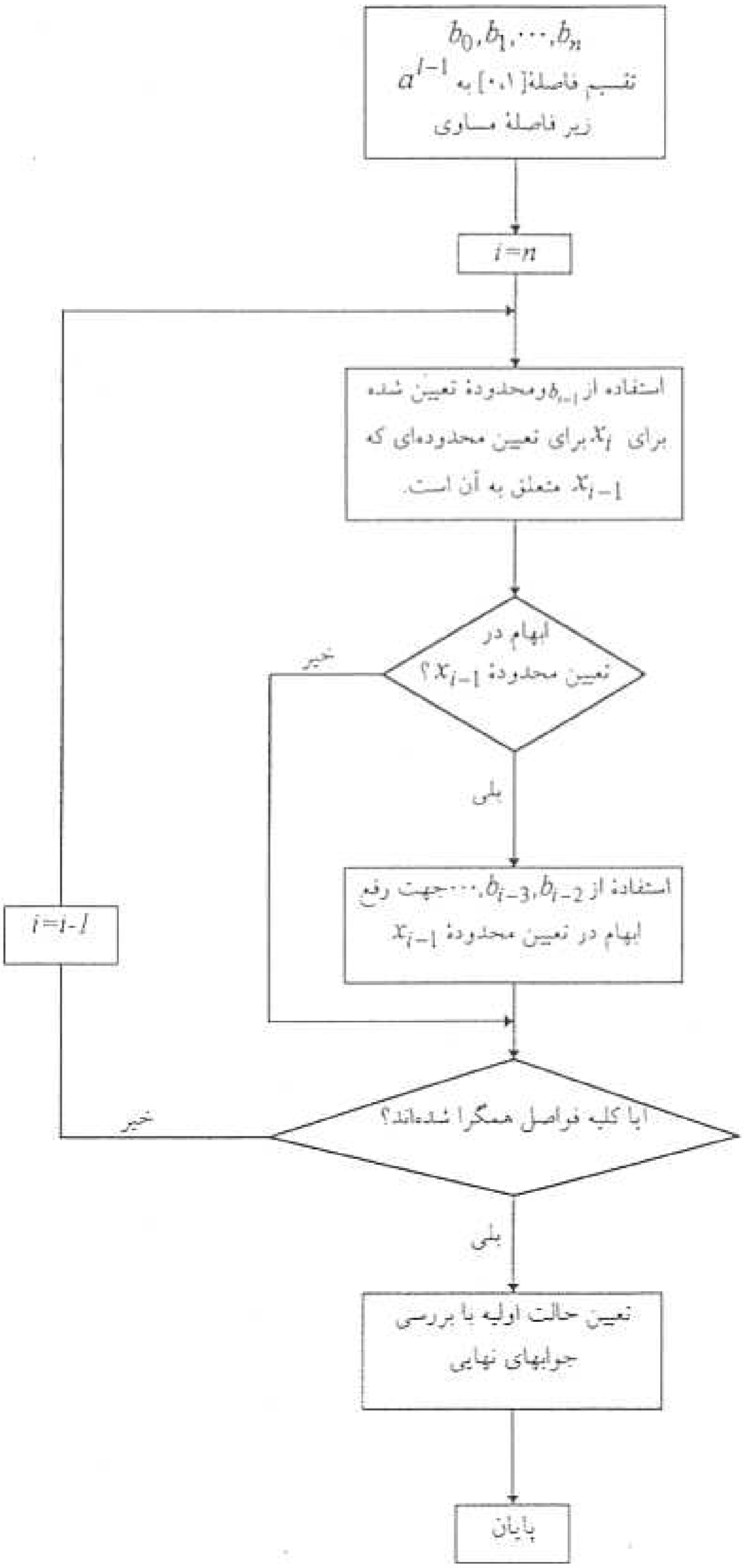
$$g(x) = \begin{cases} \frac{x}{c} & 0 \prec x \prec c \\ \frac{1-x}{1-c} & c \le x \prec 1 \end{cases} \quad , \quad c = 0.5-\varepsilon \qquad (۱۸)$$

(εمقدار بسیار کوچکی حوالی دقت محاسبات می‌باشد)

به ازای l=1 نگاشت(۱۸) همان نگاشت(۱۷) بازای c=2 خواهد شد. در حالتی که $c = 0.5 - \varepsilon$ باشد، بطور مشابه می‌توان از الگوریتم حمله بازای $l \succ 1$ استفاده نمود. منتها بخاطر عدم تقارن اندک در نگاشت، به ازای برخی حالتهای اولیه که نهایتاً به فاصلهٔ $[0.5, 0.5+\varepsilon)$ ختم می‌شوند، الگوریتم همگرا نمی‌گردد. علت این امر نیز بخاطر این است که در روش بیان شده تقارن نگاشت(۱۷) به نوعی لحاظ شده است، بنابر این در اجرای الگوریتم بواسطهٔ اینکه در فاصلهٔ $[0.5, 0.5+\varepsilon)$رفتار در نگاشت متفاوت است، لذا اگر حالت نگاشت به این فاصله ختم شود، عملا در تکرارهای بعدی، حالت نگاشت ممکن است گم شود. جداول(۲) و (۳) نتایج نهایی بدست آمده از دو دنبالهٔ نمونه با حالتهای اولیهٔ $x_{0,1} = 0.513468121385319$ و $x_{0,2} = 0.5 + 10^{-18}$ $(x_0 = 0.5+\varepsilon)$ که توسط نگاشت مثلثی(۱۸) بدست آمده را نشان می‌دهد.همانطور که از جدول(۲) ملاحظه می‌گردد، جوابهای بدست آمده شامل حالت اولیه نیز می‌باشند در حالی که جدول(۳) نشان می‌دهد که بازای حالت اولیهٔ $x_0 = 0.5+\varepsilon$ هیچیک از جوابهای نهایی برابر $x_{0,2}$ نیست .

جدول۲ : نتایج حاصل از اجرای الگوریتم حمله جهت دستیابی به حالت اولیه نگاشت مثلثی(۱۸) به ازای $c = 0.5 - 10^{-18}$ و حالت اولیهٔ $x_{0,1}$ و l=2 ، a=10 ، n=70

$x_0^1 = 0.0.433075170533277$	$x_0^6 = 0.909232509214003$
$x_0^2 = 0.328839583875101$	$x_0^7 = 0.6174944632269499$
$x_0^3 = 0.724613973713210$	$x_0^8 = 0.0401950441927839$
$x_0^4 = 0.144433491876794$	$x_0^9 = 0.513468121385319$
$x_0^5 = 0.248459897700756$	$x_0^{10} = 0.802112611340109$



شکل(۱) : روند نمای دستیابی به حالت اولیهٔ نگاشت آشوبی.

جدول ۳ :نتایج حاصل از اجرای الگوریتم حمله جهت دستیابی به حالت اولیهٔ نگاشت مثلثی(۱۸) به ازای $c = 0.5 - 10^{-18}$ و حالت اولیهٔ $x_{0,2}$ و $l=2$، $a=10$، $n=70$

$x_0^1 = 0.0$	$x_0^6 = 0.947368421052632$
$x_0^2 = 0.210526315789474$	$x_0^7 = 0.736842105263158$
$x_0^3 = 0.421052631578947$	$x_0^8 = 0.523157894736840$
$x_0^4 = 0.631578947368421$	$x_0^9 = 0.315789473684211$
$x_0^5 = 0.842105263115789$	$x_0^{10} = 0.105263157894 73$

۵- خلاصه و نتیجه گیری

در بررسی و ارزیابی دنباله‌های باینری تولید شده توسط نگاشتهای آشوبی و میزان سختی دستیابی به کلید اصلی مولد، با توجه به اینکه در عمل دقت محاسبات محدود می‌باشد می‌توان در حالتهای خاص به کلید مولد دست یافت. در این راستا یک روندنمای کلی برای اجرای حمله‌ای از نوع حملهٔ متن اصلی معلوم ادامه گردید که قابل اعمال به دنباله‌های تولید شده توسط نگاشتهای آشوبی می‌باشد. در این راستا چگونگی اعمال این حمله به دنباله‌های تولید شده توسط نگاشت مثلثی بیان گردید و نشان داده شد از بیتهای استخراج شده از ارقام با وزن بالا به سادگی می‌توان به کلید اصلی(حالت اولیهٔ نگاشت) دست یافت استفاده از ارقام با وزن کم مؤلفه‌ها از این جهت که دستیابی به کلید را نیز مشکل‌تر می‌سازند، مناسب می‌باشند.در الگوریتم ارایه شده برای دستیابی به کلید اصلی مولد، در بسیاری اوقات بیش از یکی از جوابهای نهایی برابر کلید اصلی(حالت اولیه) می‌گردد و گاهی اوقات نیز بواسطهٔ محدودیت دقت محاسبات، حالت اولیهٔ نگاشت با تقریب بسیار خوبی به جواب نزدیک میباشد(در حد خطای محاسبات).

در حملهٔ ارایه شده جهت دستیابی به حالت اولیهٔ نگاشت، حالت ساده‌ای که تنها یک بیت از هر مؤلفه استخراج شود، در نظر گرفته شده است بنابراین برای پیچیده کردن ساختار می‌توان از تعدادی ارقام با کمترین وزن هر مؤلفه بیت یا بیتهای مورد نظر را با اعمال توابع غیرخطی بدست آورد[۱]، در چنین حالتهایی با توجه به ویژگی حساسیت نگاشتها به حالت اولیه، دستیابی به کلید اصلی بسیارمشکل‌تر خواهد گردید.

۶-مراجع

[۱] محمد دخیل‌علیان" *ارزیابی دنباله‌های شبه تصادفی و طراحی مولدهای آشوبی*" ، دانشگاه صنعتی اصفهان، دانشکدهٔ برق و کامپیوتر، رسالهٔ دکترا ، آبان ۱۳۷۷.

[2] Hao Bai-Lin, CHAOS II, *Word Scientific Pub. Co. Singapore, 1990.*

[3] Devaney R.L., An Introduction to Chaotic Dynamically Systems, *Second Edition,Addison Wesley, 1989.*

[4] Schuster H.G., Deterministic Chaos, *Third augmented edition, Weiheim, New York, VCH, 1995.*