یک رمز دنباله‌ای مبتنی بر جای‌گشت آشوبی

بهروز خادم؛ دانشگاه خوارزمی، دانشکدهٔ علوم ریاضی و کامپیوتر
امیر دانشگر*؛ دانشگاه صنعتی شریف، دانشکدهٔ علوم ریاضی
سیده فهیمه محبی‌پور؛ دانشگاه خوارزمی، دانشکدهٔ علوم ریاضی و کامپیوتر

دریافت ۹۲/٤/۲۲ پذیرش ۹۳/۸/۱۰

چکیده

در این مقاله به معرفی یک رمز دنباله‌ای مبتنی بر جای‌گشت آشوبی می‌پردازیم که اساساً متشکل از یک نگاشت آشوبی و یک بخش خطی است و به‌صورت کلمهٔ محور روی یک میدان متناهی طراحی‌شده است. نشان می‌دهیم که این سامانه می‌تواند در دو حالت همزمان و خودهمزمان عمل کرده و در قالب خودهمزمان دارای گیرنده‌ای از نوع ناظر با ورودی ناشناخته[1] است. ضمن بررسی کارایی این سامانه با توجه به دقت نمایش ماشین محاسباتی، نمونهٔ نرم‌افزاری آن را پیاده‌سازی کرده و به‌عنوان یک ویژگی اصلی نشان می‌دهیم که خروجی آن حتی با گسسته‌سازی نگاشت آشوبی، واجد شرایط لازم آماری است. همچنین به‌ازای پارامترهای مختلف، این سامانه را با رمزهای دنباله‌ای مشابه مقایسه می‌کنیم و به‌طور اخص نشان می‌دهیم که در حالت کلید با اندازهٔ کوتاه (حدود ۱۰۰ بیت) این سامانه نسبت به یکی از سامانه‌های مشابه با حالت درونی تقریباً برابر، سرعت ۱۰ برابر بیش‌تر دارد.

واژه‌های کلیدی: رمز جریانی، جای‌گشت آشوبی، رمزنگاری، همزمانی

مقدمه

امروزه به دلیل این‌که در سامانه‌های رمزنگاری، کلید رمزنگاری یا کلمهٔ عبور باید ماهیتی تصادفی داشته باشد، تولید اعداد تصادفی نقش مهمی در این سامانه‌ها دارد. معمولاً تولید اعداد تصادفی با استفاده از ترکیب دودسته معین از مولدها به نام مولدهای اعداد تصادفی[2] و مولدهای اعداد شبه تصادفی[3] انجام می‌شود. اولین دسته شامل ابزارهایی است که از مفهوم فیزیکی فرآیندهای تصادفی مانند اختلالات الکترونیک و دینامیک‌های آشوبی در برخی از سامانه‌های غیرخطی استفاده می‌کنند [۳]. این ابزارها دنباله‌های خروجی با درجهٔ عدم قطعیت چشم‌گیری دارند که در نظریهٔ اطلاعات در قالب آنتروپی شانونی اندازه‌گیری می‌شود. از طرف دیگر مولدهای اعداد شبه تصادفی ماشین‌های حالت متناهی متناوب و قطعی هستند که درون یک دورهٔ تناوب باید رفتار تصادفی یک منبع تصادفی از اعداد را شبیه‌سازی کنند. از دیدگاه نظری، مولدهای شبه تصادفی بر اساس طبیعت قطعی‌شان به‌طور بالقوه دنباله‌هایی تولید می‌کنند که قابل پیش‌بینی هستند. با این‌همه در ادبیات موضوع خانواده‌های زیادی از این نوع مولدهای شبه تصادفی در دستهٔ امن طبقه‌بندی شده‌اند [۷]، [۲۴]. به‌عبارت دیگر

* **نویسندهٔ مسئول** daneshgar@sharif.ir

1 . Unknown Input Observer
2 . Truly Random Number Generators
3 . Psuedo Random Number Generators

ساختار الگوریتم این نوع مولدها، شامل محاسباتی است که قابلیت پیش‌بینی بر اساس اندازهٔ ورودی و به‌طور متوسط (با توجه به ابزار محاسباتی در دسترس و سریع‌ترین الگوریتم‌های محاسباتی موجود) به منابع محاسباتی بسیار زیادی نیازمند است و به‌این دلیل دنباله‌های تولیدشده توسط آن‌ها پیش‌بینی‌ناپذیر هستند. البته این نکته قابل‌توجه است که برای یک مولد داده‌شده، ممکن است برخی از مقادیر اولیه معین باعث تولید دنباله‌های متناوب و کوتاه (و ناامن از نظر رمزنگاری) شوند. مثلاً مولد BBS[1] که به‌ازای t=۰,۱,۲, ... به‌صورت $k_{t+1} = k_t^2 \mod n$ تعریف می‌شود، ممکن است دنباله‌هایی با دورهٔ تناوب کوتاه تولید کند [۴]. بنا بر این یک مولد شبه تصادفی معمولاً باید دنباله‌هایی را تولید کند که از جهت آماری قابل‌قبول باشند، یعنی تعداد معینی از آزمایش‌های آماری استاندارد را با موفقیت بگذرانند [۲۷]. از طرف دیگر با این‌که در دهه‌های اخیر استفاده از اولیه[2]های رمزنگاری موفقی نظیر حلقه‌های Fiestel یا شبکه‌های SPN[3] و جعبه‌های جانشانی[4] مؤثر و جای‌گشت‌های کارآمد در رمزهای بلوکی، باعث شده که تحقیقات بیش‌تری در ساختار و امنیت آن‌ها نسبت به رمزهای دنباله‌ای انجام‌شده باشد، اما همان‌گونه که در مسابقهٔ eSTREAM معلوم شد، به‌نظر می‌رسد رمزهای دنباله‌ای که مبتنی بر مولدهای شبه تصادفی امن و کارا ساخته‌شده‌اند، کماکان به ایفای نقش مهم خود در کاربردهایی که حجم داده‌ها زیاد باشد یا محدودیت منابع محاسباتی وجود داشته باشد، ادامه دهند.

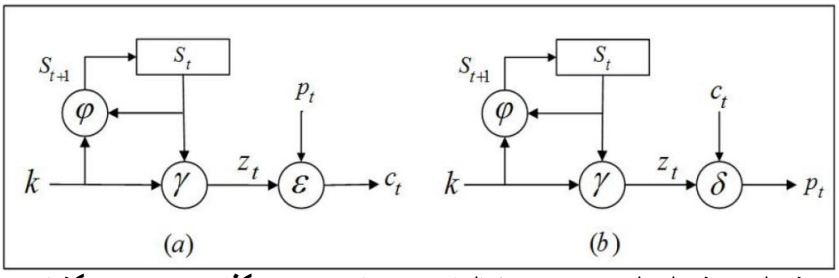
رمزهای دنباله‌ای به‌طورکلی ازنظر روش تولید دنباله کلید به دودسته همزمان و خودهمزمان تقسیم می‌شوند. به‌طور خلاصه در نوع همزمان دنبالهٔ کلید به‌طور مستقل از متن اصلی یا متن رمزی تولید می‌شود، درحالی‌که در نوع دوم دنباله کلید تولیدشده وابسته به متن اصلی یا متن رمزی یا هر دو است. در مراجع [۹]، [۲۴]، [۲۵]، [۲۶] برای رمزهای دنباله‌ای همزمان و خودهمزمان تعاریف مختلفی ارائه شده‌اند، که با توجه به کاربرد خاص موردنظر در این طرح قابل توصیف بدین‌صورت هستند:

$$\begin{cases} z_t = \gamma_k(s_t) \\ c_t = \varepsilon(z_t, p_t) \\ s_{t+1} = \varphi_k(s_t) \end{cases} \qquad (۱)$$

در این توصیف به‌ازای هر مقدار $t \geq 1$ بردار متغیر s_t مقدار حالت درونی را در لحظه t نمایش می‌دهد و به مقدار کلید مخفی k و مقدار حالت درونی در لحظه t وابسته است. تابع مولد γ_k دنباله کلید z_t را تولید می‌کند. در معادلهٔ رمزگذاری تابع خروجی ε با ترکیب دنباله کلید z_t و متن اصلی p_t متن رمز c_t را تولید می‌کند و تابع انتقال حالت φ_k مقدار حالت درونی را به‌روزرسانی می‌کند. در معادله رمزگشایی تابع خروجی $\delta = \varepsilon^{-1}$ با ترکیب دنباله کلید z_t و متن رمز c_t متن اصلی p_t را بازیابی می‌کند. فرآیندهای رمزگذاری و رمزگشایی در شکل ۱ نشان داده شده‌اند.

یک رمز دنباله‌ای خودهمزمان نیز به‌طورکلی بدین‌صورت توصیف می‌شود:

1 . Blum-Blum-Shub
2. Cryptographic Primitive
3 . Substitution-Permutation-Network
4 . S-box



شکل ۱. شکل کلی یک رمز دنباله‌ای همزمان a) رمزگذاری، b) رمزگشایی

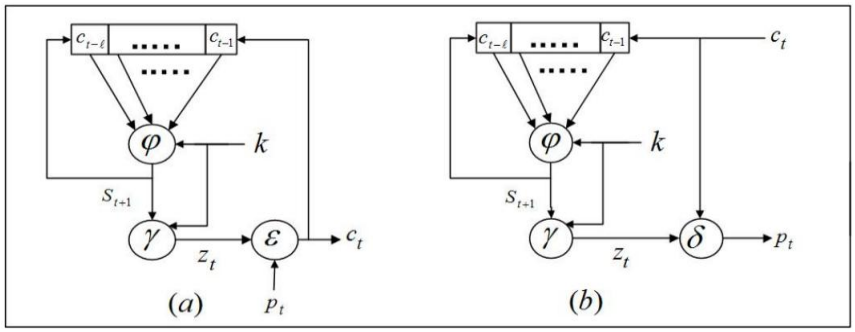
$$\begin{cases} z_t = \gamma_k(s_t) \\ c_t = \varepsilon(z_t, p_t) \\ s_{t+1} = \varphi_k(c_{t-\ell}, \ldots, c_t) \end{cases} \qquad (۲)$$

در این توصیف به‌ازای هر مقدار $t \geq 1$ بردار s_t مقدار حالت درونی را در لحظه t نمایش می‌دهد و همان‌طور که دیده می‌شود برابر مقدار تابع φ_k به‌ازای ℓ کلمهٔ متن رمزی قبلی است. (در ابتدای راه‌اندازی $\ell+1$ کلمه متن رمزی مجازی در نظر گرفته می‌شود.) شکل‌های دیگری نیز برای رمزهای خودهمزمان در [۹]، [۱۰]، [۱۱] معرفی‌شده‌اند که تقریباً معادل با همین شکل هستند. در معادله رمزگذاری تابع مولد γ_k مقدار دنباله کلید z_t را تولید می‌کند و تابع خروجی ε با ترکیب دنباله کلید و متن اصلی p_t ، متن رمز c_tرا تولید می‌کند که به‌نوبه خود به‌عنوان بازخورد وارد بردار حالت خواهد شد. در رمزگشایی دوباره تابع مولد γ_kمقدار دنبالهٔ کلید z_t را تولید می‌کند و تابع خروجی $\delta = \varepsilon^{-1}$ با ترکیب دنباله کلید و متن رمزی c_t مجدداً متن اصلی p_t را تولید می‌کند. در این قسمت متن رمزی به‌عنوان بازخورد وارد بردار حالت درونی می‌شود. فرآیندهای رمزگذاری و رمزگشایی در شکل ۲ دیده می‌شود.

به‌نظر می‌رسد که [۲۳] یکی از اولین مقالات در زمینهٔ طراحی رمزهای دنباله‌ای خودهمزمان است که ایده‌ای برای طراحی رمزهای خودهمزمان بیت محور را ارائه کرد. اما دایمن[1] این ایده را در همان سال مورد انتقاد قرار داد و اصلاح آن منجر به معرفی یکی از اولین رمزهای دنباله‌ای خودهمزمان یعنی KNOT شد که به‌نوبهٔ خود چند سال بعد به آن حمله تفاضلی شد و شکسته شد [۱۷]. روش معرفی‌شده ماورر نه‌تنها از نظر امنیتی نقد شد [۲۳]، بلکه از نظر مهندسی نیز مورد انتقاد واقع شد [۸]. دایمن برای رفع مشکل انتشار خطا در KNOT یک ساختار برای حالت درونی رمز دنباله‌ای خودهمزمان به‌نام CCSR[2] را پیشنهاد کرد. در آن مقاله رمز دنباله‌ای خودهمزمان و بیت‌محور به نام ϒΓ نیز معرفی شد که هم از این ساختار استفاده کرده بود و هم نقاط ضعف KNOT را برطرف می‌کرد.

طی سال‌های ۲۰۰۵ تا ۲۰۰۸ در گروه نرم‌افزاری مسابقه رمزهای دنباله‌ای eSTREAM که مجریان پروژه ECRYPT انجام دادند، مؤلفان [۲۹] رمز دنباله‌ای خودهمزمان به نام SSS که یک رمز کلمه‌محور بود با طول کلمات ۱۶ بیتی به‌همراه طرح احراز هویت بر اساس رمزی دنباله‌ای همزمان به‌نام SOBER ارائه کردند. این رمز دنباله‌ای در همان سال تحت حملهٔ متن رمزی انتخابی که دایمن گزارش کرد، شکسته و از دور مسابقه خارج شد [۱۲].

1 . Daemen
2 . Conditional Complementing Shift Register



شکل ۲. شکل کلی رمز دنباله‌ای خودهمزمان a) رمزگذار، b) رمزگشا

در گروه سخت‌افزاری مسابقه نیز یک رمز دنباله‌ای خودهمزمان بیت‌محور نیز به‌نام مسکویتو[1] ارائه شد [۱۰]، که مجدداً پس از انجام یک حمله متن رمزی انتخابی شکسته شد [۱۸]. سپس طراحان مسکویتو درصدد رفع نقاط ضعف آن برآمدند و نسخه‌ای از آن به‌نام موستیکیو[2] را ارائه کردند [۱۱]. اما در پایان مرحلهٔ سوم مسابقه، چون این رمز شباهت زیادی به مسکویتو داشت، به‌دلایلی که گورکیناک و همکاران ذکر کرده‌اند از نظر کارایی امتیاز لازم را کسب نکرد و از دور مسابقه خارج شد [۱۴].

در ارتباط با تحقیقات روی رمزهای آشوبی و با مراجعه به مقالات ارائه‌شده می‌توان دید که در بین سال‌های ۲۰۰۱ تا ۲۰۱۰ میلادی بسیاری از مؤلفان در معرفی طرح‌های رمزنگاری مبتنی بر آشوب، از دینامیک‌های آشوبی روی اعداد حقیقی استفاده می‌کردند، اما با توجه به بیت‌محور بودن فناوری‌های ارتباطی، استفاده از این‌گونه رمزها در کاربردهای عملی با مشکلات زیادی مواجه بود. علاوه بر این به‌دلیل تفاوت اساسی در ساختار و مدل‌سازی این دسته از رمزهای آشوبی، ارزیابی امنیت و کارایی آن‌ها بر اساس استانداردها و روش‌های سنتی رمزنگاری قابل انجام نبود. به‌عنوان نمونه یکی از اولین و معروف‌ترین این طرح‌ها را باپتیستا ارائه کرد [۶] که چند سال بعد مشخص شد در مقابل حمله متن اصلی معلوم مقاومت نمی‌کند [۱۵]. پاپادیمیتریو[3] و همکارانش در نمونه‌ای دیگر چنین طرح‌هایی را ارائه کردند [۲۸]. در سال ۲۰۰۱، کوکارف[4] و همکارانش یک طرح رمزنگاری آشوبی از نوع بلوکی ارائه کردند [۱۹]، سپس اثبات کردند که طرحشان در مقابل حمله خطی و حمله تفاضلی مقاوم است [۱۶].

به‌نظر می‌رسد از نخستین تلاش‌هایی که موجب شد توجه رمزنگاران به تدوین نظریه گسسته‌سازی نگاشت‌های آشوبی جلب شود تحقیق و پژوهش کوکارف باشد [۲۱]. کوکارف در این مقاله نشان داد که در طراحی جعبه‌های جانشانی[5] به‌کاررفته در رمزهای بلوکی آشوبی، تقریب‌های گسسته نگاشت‌های آشوبی تحت شرایطی معین، خواص آشوبی را از نگاشت‌های آشوبی پیوسته و متناظرشان به ارث می‌برند. در سال ۲۰۰٦ کوکارف و همکارانش در [۲۲] در مقابل مفهوم نمای لیاپانف[6] پیوسته که برای اندازه‌گیری خواص آشوبی در نگاشت‌های آشوبی پیوسته به‌کار می‌رود، مفهوم نمای لیاپانف گسسته را تعریف کردند و نشان دادند که تحت

1. Mosquito
2. Moustique
3 . Papadimitriou
4 . Kocarev
5 . S-box
6. Lyapunov

شرایط معینی این مفهوم برای تقریب‌های گسسته نگاشت‌های آشوبی به نمای لیاپانف پیوستهٔ آن نگاشت‌ها همگرا می‌شود. سپس آن‌ها با استفاده از این تعریف، مفهوم دیگری به نام آشوب گسسته را معرفی کردند و مثال‌های متعددی از آن را معرفی کردند. هم‌چنین آن‌ها روی مجموعه نگاشت‌های پیوسته آشوبی و یک متغیره، نوع دیگری از جای‌گشت‌های آشوبی را معرفی کردند و نشان دادند که این جای‌گشت‌ها در تعریف آشوب گسسته صدق می‌کنند. آمیگو و همکاران [۵] موضوع کاربرد آشوب گسسته بررسی کردند و ضمن تشریح روش‌های مختلف کاربرد دینامیک‌های آشوبی در رمزهای دنباله‌ای و بلوکی، محدودیت‌های استفاده از نگاشت‌های آشوبی پیوسته را در هر مورد به‌طور مجزا بیان کردند. آن‌ها در ادامه ضمن تعریف دقیق‌تری برای تقریب گسسته آشوبی، نشان دادند که انواع مختلف این تقریب‌ها را می‌توان به‌عنوان اولیه‌های امن رمزنگاری در پروتکل‌های پیچیده‌تر مانند امضای رقمی[1] استفاده کرد و مثال‌هایی از چنین اولیه‌هایی را نیز ارائه کردند. هم‌چنین در ادامه تحقیقات مرتبط با رمزهای متقارن آشوبی، مقایسهٔ این رمزها با نمونه‌های مشابه استاندارد و بیت‌محور مطرح شد [۲۶]. آدابو و همکارانش نیز نشان دادند که استفاده از ابزارهای محاسباتی برای پیاده‌سازی سامانه‌های آشوبی محدودیت‌هایی را برای این سامانه‌ها فراهم می‌کند که این محدودیت‌ها ناشی از قانون نمایش اعداد حقیقی و روش‌های گرد کردن یا برش آن‌ها در ماشین‌های محاسباتی با دقت متناهی است [۴]. به‌عبارت‌دیگر به‌دلیل قانون نمایش اعداد حقیقی و برش این اعداد در دقت متناهی، بین یک عدد حقیقی و مقدار نمایش یافته آن در دقت متناهی، مقداری اختلاف وجود دارد که میزان آن به الگوریتم گرد کردن یا برش و تعداد بیت‌های به‌کاررفته در ماشین وابسته است. آن‌ها ضمن معرفی مفهوم شبه آشوب، ارتباط خواص آشوب و شبه آشوب را از جهت پیاده‌سازی بیان کردند و شرایط لازم برای آن‌که هر دو رفتار مشابهی داشته باشند را نشان دادند [۴]. برای آشنایی با مفاهیم پایه نظریه آشوب، خواص مناسب آشوب برای رمزنگاری، فنون رمزنگاری آشوبی، کاربردهای متعدد امن مبتنی بر رمز آشوبی و تحقیقات اخیر و برخی از مسائل باز در این زمینه می‌توان به [۲۰] مراجعه کرد.

به‌نظر می‌رسد که سه ویژگی زیر در طراحی رمزهای دنباله‌ای خودهم‌زمان آشوبی اهمیت خاصی دارند، هرچند اجتماع این ویژگی‌ها در چنین سامانه‌ای غیر بدیهی است و ظاهراً به‌سادگی میسر نمی‌شود.

الف) لزوم گسسته سازی آشوب به علت پیاده‌سازی نرم‌افزاری یا سخت‌افزاری رقمی سامانه رمزنگاری (که می‌تواند موجب تضعیف یا از بین رفتن خاصیت آشوب شود).

ب) ترجیح بر ارائه سامانه‌های کلمهٔ محور به‌دلیل سرعت زیاد و پیاده‌سازی مؤثر (که طراحی بر روی میدان متناهی را اجتناب‌ناپذیر می‌کند)

ج) داشتن یک طرح خودهم‌زمان با گیرنده از نوع ناظر با ورودی ناشناخته، برای کنترل خطا و هم‌زمان‌سازی خودکار (که معمولاً می‌تواند روی یک میدان متناهی مسئله دشواری باشد).

در ادامه مختصری از هر یک از این ویژگی‌ها ارائه شده و هدف اصلی این مقاله تبیین می‌شود. در دههٔ گذشته به کاربرد نگاشت‌های آشوبی پیوسته در طرح‌های رمزنگاری توجه خاص شده است و کاربردهای متعددی در سامانه‌های امن مانند رمزهای آشوبی، مولدهای شبه تصادفی آشوبی و توابع چکیده ساز آشوبی داشته

1 . Digital signature

است. برای آشنایی بیش‌تر با مفاهیم پایه نظریه آشوب، خواص مناسب آشوب برای رمزنگاری، فنون رمزنگاری آشوبی، کاربردهای مختلف مبتنی بر رمز آشوبی، آشنایی با تحقیقات اخیر و برخی از مسائل باز در زمینه‌های مختلف رمزنگاری آشوبی می‌توان به [۵]، [۱۳] و [۲۰] مراجعه کرد. تاکنون در رمزهای دنباله‌ای، آشوب به شکل‌های مختلفی به‌کاررفته است. این دسته از رمزها اساساً مبتنی بر مولدهایی هستند که دنباله‌های پیچیده تولید می‌کنند که می‌توان از سامانه‌های آشوبی برای طراحی آن‌ها استفاده کرد. البته چون در عمل سرعت زیاد و پیچیدگی نرم‌افزاری (سخت‌افزاری) کم از جمله معیارهای مهم یک رمز دنباله‌ای هستند، سامانه‌های دینامیکی باید به‌گونه‌ای انتخاب شوند که بدون افزایش پیچیدگی، امنیت لازم را نیز برای آن‌ها فراهم کنند. یک نکتهٔ مهم در رمز آشوبی، پیاده‌سازی دینامیک آشوبی آن است. چون رایانه‌ها می‌توانند اعداد حقیقی را تا دقت معین نمایش دهند، بنابراین خواص آشوبی در این نوع رمزها به دلیل حساسیت به خطای نمایش، در حین محاسبات به‌طور تدریجی کاهش‌یافته و باعث ایجاد اختلال در رمزگشایی متن رمزی و بازیابی متن اصلی می‌شود. گاهی برای پیش‌گیری از این مسئله کاربران رمزآشوبی از دقت زیاد در محاسبات استفاده می‌کنند که باعث افزونگی محاسباتی خواهد شد.

آمیگو و همکاران [۵] نشان دادند که برخی از تقریب‌های گسسته نگاشت‌های آشوبی به‌عنوان اولیه‌های آشوبی، می‌توانند خواص آشوبی و شبه‌تصادفی مناسبی به‌وجود آورند، و تعدادی از این اولیه‌ها را معرفی کرده‌اند. همچنین آدابو[1] روشی برای فرمول‌بندی دینامیک‌های آشوبی گسسته روی اعداد صحیح را معرفی کرده است [۴].

به‌دلایل متعددی به کاربرد میدانی متناهی در طرح‌های رمز دنباله‌ای آشوبی توجه می‌شود. دلیل اول شباهت ساختاری آشوب گسسته (روی میدانی متناهی) به سایر رمزهای دنباله‌ای متداول و کلمهٔ محور است که در نتیجه قابلیت مقایسهٔ کارایی و امنیت بین آن‌ها را بیش‌تر می‌کند. دومین دلیل قابلیت استفاده از عمل‌گرهای بولی (تعمیم‌یافته) و منطقی در میدانی متناهی است که کارایی و سرعت آن را افزایش می‌دهد و بالأخره دلیل سوم امکان انجام موازنه نسبی بین افزایش طول کلمه و کاهش حجم محاسبات در کاربردهای مختلف رمز دنباله‌ای با انتخاب میدان متناهی مناسب است.

به‌نظر می‌رسد تنها سامانه رمز دنباله‌ای خودهمزمان آشوبی که بر مبنای میدانی متناهی معرفی‌شده است متعلق به میلریو[2] و همکارانش باشد [۲۵]. آن‌ها با استفاده از برخی مفاهیم نظریه کنترل، ارتباط میان رمزهای دنباله‌ای خودهمزمان و رمزهای مبتنی بر سامانه‌های دینامیکی هموار[3] را بیان کردند و نشان دادند که اگر در یک سامانه ارتباطاتی دینامیکی زمان گسسته و خطی شرایط معینی برقرار باشد، آن‌گاه مشابه با رمز دنباله‌ای خودهمزمان، رمزگشا می‌تواند حالت درونی‌اش را به‌طور خودکار همزمان کند. در ادامه تان[4] و میلیریو نشان دادند که به‌طورکلی، سامانه‌های دینامیکی و رمزهای دنباله‌ای خودهمزمان بیت محور تحت شرایط معینی باهم معادل هستند [۳۰]، [۳۱].

1 . Addabbo
2 . Millerioux
3 . Flat
4 . Tan

همچنین آن‌ها یک شکل رمز دنباله‌ای خودهمزمان به‌نام رمز سوئیچینگ[1] خطی را معرفی کردند که با استفاده از یک تابع سوئیچینگ پیاده‌سازی شده با یک نگاشت آشوبی، دینامیک خروجی بین توابع رمزگذاری را جابه‌جا می‌کرد. اقبالی این سامانه دینامیکی را از نظر برخی از مشخصه‌های رمزنگاری مانند طول کلید، اندازه محاسبات و خصوصیت‌های آماری دنباله خروجی بررسی کرد [۲]. او نشان داد با این‌که دنبالهٔ خروجی این سامانه آزمون‌های آماری استاندارد را به‌خوبی پاس می‌کند اما برای پیاده‌سازی نرم‌افزاری آن به محاسبات اولیه زیادی نیاز است. مشکلات مربوط به انتقال کلید خصوصی که ناشی از بزرگی طول کلید است و همچنین اندازهٔ نسبتاً زیاد محاسبات پیش‌پردازشی در طرح میلیریو از جمله نکاتی است که اقبالی به آن اشاره کرده است.

هدف ما در این مقاله، ارائه یک سامانهٔ رمز دنباله‌ای آشوبی گسسته و کلمهٔ محور مبتنی بر میدانی متناهی است که طول کلید کوچک و محاسبات کم داشته باشد و بتوان از آن هم در حالت همزمان و هم در حالت خودهمزمان استفاده کرد. به‌علاوه چنان‌که دیده می‌شود در طرح پیشنهادی با استفاده از ماتریس‌های پوچ‌توان می‌توان در خودهمزمان سازی رمزگشا بر حل مشکلات حاصل از کانال‌های پر اختلال غلبه کرد. در انتها مقایسهٔ طرح پیشنهادی با رمز دنباله‌ای خودهم زمان موستیکیو ارائه می‌شود.

در این مقاله در بخش نمادگذاری، سامانه رمزآشوبی و ایده‌های اصلی آن را معرفی می‌کنیم. در بخش توصیف ساختار PLC[2]، ساختار طرح پیشنهادی را به‌طور کامل بیان می‌شود. در بخش نگاشت آشوبی استفاده شده در سامانه، برخی از خواص آن نگاشت و روش محاسبه یک جای‌گشت آشوبی از آن را توصیف می‌کنیم. سپس در بخش کارکردهای سامانه، انواع کارکردهای عملیاتی برای این سامانه را بیان کرده و ضمن اشاره مختصر به‌روش‌های طراحی، اثبات می‌کنیم تحت چه شرایطی این سامانه می‌تواند با یك سامانه رمز دنباله‌ای همزمان یا خودهمزمان معادل شود. همچنین در این بخش با یک جدول مقایسه‌ای، کارایی سامانه پیشنهادی را با یک رمز دنباله‌ای مشابه ازنظر معیارهای مختلف بررسی می‌کنیم. در قسمت اول بخش آزمایش‌های آماری، نتایج آزمون‌های آماری را بر روی نسخه شبه آشوبی نگاشت انتخابی و در قسمت دوم آن، نتایج آزمون‌های آماری بر روی خروجی سامانه را ارائه کرده و نشان می‌دهیم که دنباله خروجی به‌طورکلی خواص شبه تصادفی موردنظر رمزنگاری را دارد. در بخش نتیجه‌گیری، نکات نهایی و نتایج به‌دست‌آمده را بیان می‌کنیم.

سامانه دینامیکی آشوبی PLC به‌عنوان یک رمز دنباله‌ای

در این بخش ابتدا نمادها و تعاریف موردنیاز و سپس ساختار رمز دنباله‌ای پیشنهادی (ازاین‌پس PLC نامیده می‌شود) در قالب یک هسته، معادلات رمزگذاری و رمزگشایی مربوط ارائه می‌شود.

نمادگذاری

فرض کنید $f:\mathbb{R}\to\mathbb{R}$ یک نگاشت آشوبی مطابق با تعریف دوانی[3] [۱۳] و $\mathbb{F}_q = GF(q)$ میدانی متناهی q عضوی باشد. یک خانواده از نگاشت‌ها مانند $\pi:\Re\times\mathbb{F}_q\to\mathbb{F}_q$ در نظر گرفته می‌شود به‌طوری‌که به‌ازای هر مقدار $k\in\Re$ نگاشت $\pi(k,.)$ یک تقریب گسستهٔ آشوبی از f مطابق با تعریف کوکارف در [۵]

1. Switching
2 . Pseudolinear Chaotic
3 . Devaney

باشد. نگاشت دومتغیره π مقدار $k \in \mathfrak{K}$ و عنصر $a \in \mathbb{F}_q$ را گرفته و مقدار $\pi_k(a) \overset{def}{=} \pi(k,a)$ را برمی‌گرداند. با فرض دوسویی بودن π_k، برای هر مقدار کلید $k \in \mathfrak{K}$، جای‌گشت $\pi_k : \mathbb{F}_q \to \mathbb{F}_q$ یک جای‌گشت آشوبی گسسته روی میدان $\mathbb{F}_q$ نامیده می‌شود. روش تعیین و محاسبهٔ π_k در بخش ٤ به‌طور کامل توضیح داده خواهد شد.

برای هر $t = 1, 2, \cdots$ مقادیر $p_t, c_t, z_t \in \mathbb{F}_q$، $< p_t >$ به‌عنوان دنباله متن اصلی، $< c_t >$ دنباله متن رمزی و $< z_t >$ دنبالهٔ کلید اجرایی در نظر گرفته می‌شود. به‌ازای مقدار $\ell \in \mathbb{N}$ بردارهای $\mathbf{p_t}, \mathbf{c_t}, \mathbf{z_t} \in \mathbb{F}_q^{\ell}$ بردارهایی ستونی هستند که سطر اول آن‌ها به‌ترتیب $p_t, c_t, z_t \in \mathbb{F}_q$ هستند و بقیه سطرهای آن‌ها صفر است. بردار ستونی حالت درونی $\mathbf{s_t} \in \mathbb{F}_q^{\ell}$ بدین‌صورت تعریف می‌شود:

$$\mathbf{s_t} \overset{def}{=} [s_t^{(1)}, s_t^{(2)}, \cdots, s_t^{(\ell)}]^{\mathrm{T}}.$$

هم‌چنین مجموعه همهٔ ماتریس‌های $\ell \times \ell$ روی $\mathbb{F}_q$ را با $\mathfrak{M}$ نمایش داده و فرض می‌شود که $\mathbf{A,B,D,E,F,W} \in \mathfrak{M}$ و ماتریس $\mathbf{F}$ معکوس‌پذیر روی میدان $\mathbb{F}_q$ باشد. برای هر $\ell \in \mathbb{N}$ نماد $\mathbf{1}_{\ell}$ نشان‌دهنده برداری کاملاً یک و نماد $\mathbf{1}_{\ell \times \ell}$ نشان‌دهندهٔ ماتریسی کاملاً یک است. به‌طور مشابه نماد $\mathbf{0}_{\ell}$ نشان‌دهندهٔ برداری کاملاً صفر و نماد $\mathbf{0}_{\ell \times \ell}$ نشان‌دهندهٔ یک ماتریس کاملاً صفر است.

نگاشت برداری $\wp_k : \mathbb{F}_q \to \mathbb{F}_q$ حاصل از اعمال جای‌گشت آشوبی π_k روی هر یک از عناصر برداری ستونی بدین‌صورت تعریف می‌شود:

$$\wp_k([a_1, \cdots, a_{\ell}]^{\mathrm{T}}) \overset{def}{=} [\pi_k(a_1), \cdots, \pi_k(a_{\ell})]^{\mathrm{T}}.$$

در این سامانه از یک بردار $\tilde{\mathbf{c}}_{\mathbf{t}} \in \mathbb{F}_q^{\ell}$ به‌عنوان حافظه شامل تعدادی از مقادیر قبلی c_t نیز استفاده شده است. هم‌چنین از نماد $s \overset{\$}{\leftarrow} S$ برای نشان دادن فرایند انتخاب عضو تصادفی s از مجموعهٔ S با قانون احتمال یک‌نواخت استفاده شده است. به‌طورکلی سامانه PLC شامل مجموعه‌ای از توابع بدین‌شرح در جدول ۱ است:

جدول ۱. توابع به‌کاررفته در سامانهٔ PLC

عنوان	ضابطه
به‌روزرسانی حالت	$\varphi_k : \left(\mathbb{F}_q^{\ell}\right)^2 \times \mathfrak{M}^4 \to \mathbb{F}_q^{\ell}$
مولد دنباله کلید	$\gamma_k : \left(\mathbb{F}_q^{\ell}\right)^2 \times \mathfrak{M}^2 \to \mathbb{F}_q^{\ell}$
رمزگذاری	$\varepsilon_k : \left(\mathbb{F}_q^{\ell}\right)^2 \times \mathfrak{M} \to \mathbb{F}_q^{\ell}$
رمزگشایی	$\delta_k : \left(\mathbb{F}_q^{\ell}\right)^2 \times \mathfrak{M} \to \mathbb{F}_q^{\ell}$
به‌روزرسانی حافظه	$\mu : \left(\mathbb{F}_q^{\ell}\right)^2 \to \mathbb{F}_q^{\ell}$

توصیف ساختار PLC

قالب کلی سامانه PLC شامل الگوریتم بارگذاری اینیت[1] و سه بخش به نام‌های هسته کرنل[2]، رمزگذاری (Enc) و رمزگشایی (Dec) است که بر اساس شکل ۲ و به‌ازای مقادیر $t = 1, 2, \cdots$ بدین‌صورت بیان می‌شوند:

1 Init
2 Kernel

$$\text{Keel:}\begin{cases}\text{setup}: \pi_k \leftarrow Init(\psi, k)\\ \text{update:}\begin{cases} initial: \pi_k, \mathbf{s_1}, \tilde{\mathbf{c}}_\mathbf{1}, \mathbf{A}, \mathbf{B}, \mathbf{D}, \mathbf{E}, \mathbf{F}, \mathbf{W}\\ \mathbf{s_{t+1}} = \varphi_k(\tilde{\mathbf{c}}_\mathbf{t}, \mathbf{s_t}, \mathbf{p_t}, \mathbf{A}, \mathbf{D}, \mathbf{E}, \mathbf{W})\\ \tilde{\mathbf{c}}_\mathbf{t+1} = \mu(\tilde{\mathbf{c}}_\mathbf{t}, \mathbf{c_{t-1}})\\ \mathbf{z_t} = \gamma_k(\tilde{\mathbf{c}}_\mathbf{t}, \mathbf{s_t}, \mathbf{B}, \mathbf{W})\end{cases}\end{cases}$$

$$\text{Enc:}\begin{cases}\text{Input} \ : \ < p_t >\\ \text{Kernel}\\ \text{output}: \mathbf{c_t} = \varepsilon_k(\mathbf{z_t}, \mathbf{p_t}, \mathbf{F}),\end{cases}$$

$$\text{Dec:}\begin{cases}\text{Input} \ : \ < c_t >\\ \text{Kernel}\\ \text{output}: \mathbf{p_t} = \delta_k(\mathbf{z_t}, \mathbf{c_t}, \mathbf{F}^{-1}).\end{cases}$$

کلید مخفی سامانه، یک رشته بیتی شامل کد دودویی نشان‌دهندهٔ دقت نمایش ماشین $(prec)$، کد دودویی نشان‌دهندهٔ دو مقدار اولیه نگاشت آشوبی (r_0, l_0)و کد دودویی مؤلفه‌های ماتریس E است. لازم به توضیح است که به‌جز n مؤلفه از ماتریس E که به‌طور تصادفی انتخاب می‌شوند و مقدار و موقعیت آن‌ها در کلید ذخیره می‌شود، بقیه مؤلفه‌های آن برابر مقدار ثابت و دلخواه a هستند (چنان‌که در ادامه دیده خواهد شد در کارکرد هم‌زمان سامانه ، ماتریس $\mathbf{E = 0}$ فرض می‌شود). در حالت کلی کلید مخفی بدین‌صورت تعریف می‌شود:

$$k \overset{def}{=} < prec, r_0, l_0, a, (i_1, j_1, e_{i_1 j_1}), \cdots, (i_n, j_n, e_{i_n j_n}) > \qquad (۳)$$

که در آن $prec \in \{0,1\}$ ، $r_0 \overset{\$}{\leftarrow} (0,1)$ ، $l_0 \overset{\$}{\leftarrow} \{0, \cdots, 2^{prec} - 1\}$ ، $a \overset{\$}{\leftarrow} \mathbb{F}_q - \{0\}$ یک عضو ثابت هستند. کد دودویی مربوط به مقادیر و موقعیت n عنصر تصادفی ماتریس E با سه‌تایی (i, j, e_{ij}) نمایش داده‌شده‌اند. با توجه به توضیحات بالا اندازهٔ کلید برابر $|k| = 1 + 2prec + (n+1)\log_2(q-1) + 2n\log_2 \ell$ است. در جدول ۲ انتخاب‌های مناسب برای اندازهٔ کلید برحسب پارامترهای مربوطه آورده شده است.

جدول ۲. انتخاب‌های مناسب برای اندازهٔ کلید

q	$prec$	ℓ	n	$\lvert k \rvert$
۱۶ یا ۱۷	۱۶	۸	۶	۹۷
۱۶ یا ۱۷	۱۶	۸	۹	۱۲۷
۱۶ یا ۱۷	۳۲	۸	۱۹	۲۵۸ یا ۲۵۹
۳۱ یا ۳۲	۱۶	۸	۵	۹۳
۳۱ یا ۳۲	۱۶	۸	۸	۱۲۶
۳۱ یا ۳۲	۳۲	۸	۱۷	۲۵۶ یا ۲۵۷

علت انتخاب این روش کدگذاری برای کلید مخفی نسبت به‌روش کدگذاری جای‌گشت آشوبی، صرفه‌جویی در اندازهٔ بیتی آن به‌خصوص برای مقادیر بزرگ q است. بردار ستونی $\tilde{\mathbf{c}}_t$ و تابع به‌روزرسانی حافظه μ بدین‌صورت تعریف می‌شوند:

$$\tilde{\mathbf{c}}_t \overset{def}{=} [\tilde{c}_t^{(\ell)}, \tilde{c}_t^{(\ell-1)}, \cdots, \tilde{c}_t^{(1)}]^{\mathrm{T}} \quad , \quad \mu(\tilde{\mathbf{c}}_t, \mathbf{c}_{t-1}) \overset{def}{=} \mathbf{M}\tilde{\mathbf{c}}_t + \mathbf{c}_{t-1}$$

ضمناً $\tilde{c}_t^{(i)} = c_{t-\ell+i-1}$ و ماتریس $\mathbf{M}$ بدین‌صورت است:

$$\mathbf{M} = \begin{bmatrix} 0 & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & 0 & \cdots & 1 & 0 \\ 0 & 0 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 1 & \cdots & 0 & 0 \\ 0 & 1 & 0 & \cdots & 0 & 0 \end{bmatrix}.$$

مقادیر $\tilde{\mathbf{c}}_1, \mathbf{p}_1, \mathbf{c}_1, \mathbf{s}_1$ به‌عنوان پارامترهای ثابت سامانه برای راه‌اندازی و به‌روزرسانی سامانه فرض می‌شوند. در سامانه PLC توابع نام‌برده در جدول ۱ به‌طور مشخص بدین‌صورت تعریف‌شده‌اند:

$$\text{Kernel:}\begin{cases} \text{setup}: \pi_k \leftarrow \mathit{Init}(\psi, k) \\ \text{update:}\begin{cases} \mathit{initial}: \pi_k, \mathbf{s}_1, \tilde{\mathbf{c}}_1, \mathbf{A}, \mathbf{B}, \mathbf{D}, \mathbf{E}, \mathbf{F}, \mathbf{W} \\ \mathbf{s}_{t+1} = \mathbf{W}\tilde{\mathbf{c}}_t + \mathbf{D}\mathbf{s}_t + \mathbf{A}\,\wp_k(\mathbf{s}_t) + \mathbf{E}\,\wp_k(\mathbf{p}_t) \\ \tilde{\mathbf{c}}_{t+1} = \mathbf{M}\tilde{\mathbf{c}}_t + \mathbf{c}_{t-1} \\ \mathbf{z}_t = \mathbf{W}\tilde{\mathbf{c}}_t + \mathbf{B}\,\wp_k(\mathbf{s}_t), \end{cases} \end{cases} \qquad (۴)$$

ضمناً معادلات رمزگذاری به شکل

$$\text{Enc:}\begin{cases} \text{Input} \ : \ < p_t > \\ \text{Kernel} \\ \text{output}: \mathbf{c}_t = \mathbf{z}_t + \mathbf{F}\,\wp_k(\mathbf{p}_t), \end{cases} \qquad (۵)$$

و معادلات رمزگشایی بدین‌صورت تعریف می‌شوند:

$$\text{Dec:}\begin{cases} \text{Input} \ : \ < c_t > \\ \text{Kernel} \\ \text{output}: \mathbf{p}_t = \wp_k^{-1}(\mathbf{F}^{-1}(\mathbf{c}_t - \mathbf{z}_t)). \end{cases} \qquad (۶)$$

در معادلهٔ (٤) دیده می‌شود که تابع به‌روزرسانی حالت φ_k در هسته اصلی سامانه، شامل یک بخش خطی و یک بخش آشوبی است. بخش خطی ازیک‌طرف باعث ایجاد وابستگی بین حالت درونی و نمادهای قبلی متن رمزی می‌شود و از طرف دیگر باعث افزایش یک‌نواختی توزیع خروجی می‌شود. بخش آشوبی غیرخطی نیز باعث وابستگی حالت درونی به متن اصلی و باعث از بین بردن رابطه خطی بین حالت‌های درونی متوالی می‌شود. در ادامه در بخش نگاشت آشوبی استفاده شده در سامانه نحوهٔ انتخاب نگاشت آشوبی و تولید جای‌گشت آشوبی توصیف‌شده و در بخش آزمایش‌های آماری نشان داده می‌شود که کارکرد سامانه قابل تنظیم خواهد بود. چنان‌که دیده می‌شود، در هر نوع کارکرد معادلات هسته و رمزگذار نیز متفاوت خواهند بود.

نگاشت آشوبی استفاده شده در سامانه

چون نگاشت آشوبی تنها تأمین‌کننده خاصیت غیرخطی در سامانه است و بر امنیت آن تأثیر می‌گذارد، لازم است در نحوهٔ انتخاب و به‌کارگیری آن دقت به‌عمل آید. در این بخش ابتدا ایدهٔ اولیه نگاشت قطعه‌قطعه خطی آشوبی و نقاط ضعف آن و اصلاحاتی که بعداً آدابو [۴] روی آن به‌عمل‌آورده، معرفی می‌شوند. در ادامه ضابطهٔ نگاشت قطعه‌قطعه خطی آشوبی که در PLC به‌کاررفته معرفی‌شده و ضمن معرفی خواص اصلی این نگاشت، دلایل انتخاب آن بیان می‌شود.

پاپادیمیتریو و همکارانش برای استفاده در سامانهٔ رمز آنالوگ از یک تابع خطی و قطعه‌قطعه پیوسته بدین‌صورت استفاده کردند [۲۸]:

$$\psi(x) = a + bx + \sum_{j=1}^{n} d_j \left| x - e_j \right| \qquad (۷)$$

که در آن همهٔ پارامترها به‌گونه‌ای انتخاب می‌شدند که دینامیک آشوبی برای تابع به وجود بیاید، درون دامنه‌ای معین و محدود، باقی بماند و از نظر پیاده‌سازی هم ساده باشند. اما در بررسی‌های انجام‌شده بعدی معلوم شد که به‌دلیل حقیقی بودن تابع، سامانه رمزنگاری معرفی‌شده در کاربردهای رقمی و محاسبات دقت متناهی، خواص آشوبی خود را از دست می‌دهد که باعث می‌شود بازیابی کامل متن رمزی ناممکن شود [۲۸].

برای اصلاح نقاط ضعف تابع (۷) نگاشت آشوبی قطعه‌قطعه خطی (۸) پیشنهاد شده و آزادی بررسی کرد [۱]. نتایج بررسی او نشان داد که برای خانوادهٔ نگاشت (۸) نمودار

$$\psi(x) = \begin{cases} 3x & , \quad x \in (0, \frac{1}{4}) \cup (\frac{1}{4}, \frac{1}{3}) \\ 3x-1 & , \quad x \in (\frac{1}{3}, \frac{1}{2}) \cup (\frac{1}{2}, \frac{2}{3}) \\ 3x-2 & , \quad x \in (\frac{2}{3}, \frac{3}{4}) \cup (\frac{3}{4}, 1) \\ rnd(.) & , \quad x \in \left\{0, \frac{1}{4}, \frac{2}{4}, \frac{3}{4}, 1\right\} \end{cases} \qquad (۸)$$

هیستوگرام نگاشت به‌ازای مقادیر اولیه مختلف قابل‌قبول است و وقوع خط سیرهای متناوب با دوره تناوب کوتاه، کم مشاهده‌شده است. همچنین در اکثر موارد تابع توزیع احتمال دنباله تکرارهای نگاشت، یکنواخت است.

از سوی دیگر این نتایج نشان دادند که برای خانوادهٔ نگاشت (۸) در محاسبات دقت متناهی تنزل دینامیکی سامانه امکان‌پذیر است. ضمناً نمای لیاپونوف آن در برخی نقاط دامنه صفر است و به‌ازای تعدادی از مقادیر اولیه خط سیر نگاشت متناوب می‌شود. همچنین با در نظر گرفتن ورودی‌ها در مبنای ۳، رفتار نگاشت کاملاً قابل پیش‌بینی است. آدابو حالتی خاص از خانواده تبدیل رینی، نگاشت‌های قطعه‌قطعه خطی را بدین‌صورت در نظر گرفت [۴]:

$$\psi_\beta(x) = \beta x - \lfloor x \rfloor \quad , \quad \beta > 1 \qquad (۹)$$

که در آن $\lfloor x \rfloor$ نشان‌دهندهٔ بزرگ‌ترین عدد صحیح کوچک‌تر از x است. او نشان داد که برای خانواده نگاشت (۹)، نمای لیاپونوف مثبت و بزرگ است و اگر پارامتر β مقدار صحیح باشد، تابع چگالی احتمال

تکرارهای متوالی آن پایدار، منحصربه‌فرد و دارای توزیع احتمال یک‌نواخت است. هم‌چنین او نشان داد که (۹) خواص مخلوط کنندگی خوبی را ایجاد می‌کند. به‌علاوه او نشان داد که در دنباله تکرارهای نگاشت، به‌شرط پایدار بودن تابع چگالی احتمال تنزل تدریجی خودهم‌بستگی اتفاق می‌افتد. از طرف دیگر نتایج بررسی‌های او نشان می‌دهد که به‌ازای برخی از مقادیر اولیه برای این خانواده، دنباله تکرارهای نگاشت ممکن است دوره تناوب کوتاه داشته باشد. در این مقاله حالت خاصی از نگاشت آشوبی رینی با فرض $\beta = 3$ و با در نظر گرفتن یک مقدار ثابت $\psi_0 \in (0,1)$ بدین‌صورت انتخاب شده است.

$$\psi_f(x) = \begin{cases} \psi_3(x) & , \quad x \neq 0, \frac{1}{4}, \frac{2}{4}, \frac{3}{4}, 1 \quad . \\ \psi_0 & , \quad else \quad . \end{cases} \qquad (۱۰)$$

بر اساس نتایج اعلام‌شده، در جدول ۳ نقاط قوت و ضعف خانواده نگاشت (۱۰) به‌همراه مرجع مربوط خلاصه شده است. برای کاهش نقاط ضعف نام‌برده شده در جدول ۲، در هسته سامانه PLC بخش خطی مناسب در کنار بخش غیرخطی آن یعنی نگاشت آشوبی (۱۰) افزوده شده است. هم‌چنین از مؤلفه‌های مختلفی مانند ماشین حالت متناهی، حافظه، جمع در پیمانه و جای‌گشت استفاده شده است.

جدول ۳. نقاط ضعف و قوت خانواده نگاشت (۱۰)

نقاط قوت	نقاط ضعف
بزرگی نمای لیاپونف [۴]	وجود دنباله‌های با تناوب کوتاه [۱]، [۴]
یک‌نواختی تابع چگالی احتمال [۴]	تنزل تدریجی خواص آشوبی [۱]
مخلوط کنندگی خوب [۴]، [۲۸]	
مقبولیت نمودار هیستوگرام فراوانی مقادیر نگاشت [۱]	

در ادامه با بررسی آماری نتایج دنباله خروجی سامانه در بخش آزمایش‌های آماری نشان داده خواهد شد که نتیجه این اصلاحات باعث بهبود نقاط ضعف نگاشت آشوبی شده است. معمولاً به‌منظور جلوگیری از بروز خطای نمایش یا برش اعداد حقیقی در محاسبات با دقت متناهی، در مرحله راه‌اندازی نگاشت آشوبی، از دقت محاسباتی مضاعف برای تولید دنباله تکرارهای نگاشت آشوبی استفاده می‌شود. با استفاده از روش معرفی‌شده کوکاریو [۲۲]، در فرآیند راه‌اندازی اولیه سامانه PLC، تعریف و تولید جای‌گشت آشوبی بدین‌صورت انجام می‌پذیرد: ابتدا یک مقدار حقیقی r_0 و یک عدد طبیعی l_0 به‌طور تصادفی انتخاب‌شده و با تکرار متوالی نگاشت (۱۰) ، بردار حقیقی

$$\mathbf{v_c} = [\psi_f^{l_0}(r_0), \psi_f^{l_0+1}(r_0), \cdots, \psi_f^{l_0+q-1}(r_0)]$$

تولید می‌شود. دلیل صرف‌نظر کردن از l_0 تکرار اولیهٔ نگاشت آشوبی، جلوگیری از تأثیر سوء نقاط گذرای غیرآشوبی ابتدایی در تکرارها است. به‌دلیل رفتار آشوبی نگاشت $\psi_f(x)$ و بدون کاستن از کلیات روش، فرض می‌شود همهٔ عناصر این بردار از هم متمایز باشند (در غیر این صورت نقاط بیش‌تری از دنباله تولید خواهد شد). سپس مجموعه عناصر بردار $\mathbf{v_c}$ به‌طور صعودی مرتب شده و به‌صورت

$$\left\{x_i \mid x_0 < x_1 < \cdots < x_{q-1}\right\}$$

نمایش داده می‌شود. در انتها با استفاده از این بردار مرتب شده، جای‌گشت آشوبی π_k بدین‌صورت تعریف می‌شود،

$$\pi_k(i) = \begin{cases} j \ , & (\psi_f(x_i) = x_j \ , \ i \neq q\text{-}1\ ) \\ 0 \ , & i = q\text{-}1 \end{cases} \qquad (۱۱)$$

کوکارف اثبات کرده است که خواص آشوبی جای‌گشت تولید شده، حداقل برابر خواص آشوبی نگاشت مولد آن است [۲۲]. نتایج بررسی‌ها در بخش آزمایش‌های آماری نیز نشان می‌دهند که جای‌گشت (۱۱) تأثیر زیادی را در ایجاد خواص شبه تصادفی خروجی سامانه ایفا می‌کند.

کارکردهای مختلف سامانه

در این بخش نشان داده می‌شود که سامانه رمز دنباله‌ای PLC می‌تواند در دو نوع همزمان (SPLC[1]) و خودهمزمان (SSPLC[2]) به‌کار برود و این قابلیت‌ها با انتخاب‌های مناسب برای ماتریس‌ها در (۴) به‌وجود می‌آید. به‌همین منظور در هر یک از کارکردهای PLC ، خصوصیات ویژه سامانه مانند کلید مخفی، حالت اولیه و شرایط لازم در هر نوع کارکرد بیان می‌شود. در انتهای این بخش ، سامانه PLC از نظر کارایی و امنیت با یک سامانه مشابه مقایسه می‌شود. قابل‌ذکر است که در کارکردهای همزمان و خودهمزمان معادلات رمزگشا متفاوت خواهند بود. این تفاوت به این علت است که در کارکرد خودهمزمان، در صورت بروز خطا در متن رمزی دریافتی، معادلات رمزگشایی باید پس از تعداد معینی سیکل، عمل خودهمزمان سازی را به‌صورت خودکار انجام بدهد، درحالی‌که در کارکرد همزمان، همزمان سازی به عهده کاربر است و انتشار خطا اجتناب‌ناپذیر است.

کارکرد همزمان

معادلات کلی سامانه SPLC به‌صورت (۱۲) و (۱۳) و (۱٤) هستند. کلید شامل یک رشتهٔ بیتی مطابق با تعریف (۳) و حالت درونی سامانه شامل یک حافظه ℓ بیتی و یک متغیر حالت ℓ بیتی است. همچنین در این کاربر باید نوع کارکرد همزمانی را تضمین کند. در SPLC خطا (در صورت وجود روی کانال) کاملاً قابل انتشار خواهد بود. قضیه زیر شرایط لازم برای تبدیل PLC به SPLC را نشان می‌دهد.

گزارهٔ ۵-۱: هرگاه در سامانه PLC ماتریس‌های $\mathbf{E} = \mathbf{W} = \mathbf{0}$ انتخاب شوند، آن‌گاه به سامانه SPLC تبدیل می‌شود و معادل یك رمز دنباله‌ای همزمان استاندارد و مطابق شکل ۱ است.

برهان: با جای‌گذاری شرایط گزاره و با استفاده از (٤) و (٥) و (٦) معادلات سامانه SPLC به‌صورت (۱۲) و (۱۳) و (۱٤) تعریف می‌شوند. از (۱۲) و (۱۳) می‌توان معادله رمزگذاری SPLC را به‌صورت خلاصه (۱۵) و مطابق با شکل استاندارد همزمان (۱) نوشت. قابل توجه است که در (۱۲) معادله مربوط به $\tilde{c}_t$

1 . Synchronous PLC
2. Self-synchronous PLC

نقشی در سایر معادلات سامانه ندارد و در نتیجه ماتریس $\mathbf{M}$ می‌تواند هر مقدار دلخواهی داشته باشد. همچنین اگر در PLC سایر ماتریس‌ها هم کاملاً دلخواه انتخاب‌شده باشند، سامانه رمز هنوز همزمان است ولی به‌دلیل وجود سربار محاسباتی زیاد، استفاده‌اش توصیه نمی‌شود.

$$\text{Kernel}_s:\begin{cases}\text{setup}: \pi_k \leftarrow Init(\psi,k)\\ \text{update}:\begin{cases} initial: \pi_k, \mathbf{s_1}, [\tilde{\mathbf{c}}_\mathbf{1}], \mathbf{A}, \mathbf{B}, \mathbf{D}, \mathbf{F}\\ \mathbf{s_{t+1}} = \mathbf{D}\mathbf{s_t} + \mathbf{A}\,\wp_k(\mathbf{s_t})\\ \{\tilde{\mathbf{c}}_{\mathbf{t+1}} = \mathbf{M}\tilde{\mathbf{c}}_\mathbf{t} + \mathbf{c}_{\mathbf{t\text{-}1}}\}\\ \mathbf{z_t} = \mathbf{B}\,\wp_k(\mathbf{s_t}), \end{cases}\end{cases} \quad (۱۲)$$

$$\text{Enc}_s:\begin{cases}\text{Input} \;:\; < p_t >\\ \text{Kernel}_s\\ \text{output}: \mathbf{c_t} = \mathbf{z_t} + \mathbf{F}\,\wp_k(\mathbf{p_t}),\end{cases} \quad (۱۳)$$

$$\text{Dec}_s:\begin{cases}\text{Input} \;:\; < c_t >\\ \text{Kernel}_s\\ \text{output}: \mathbf{p_t} = \wp_k^{-1}(\mathbf{F}^{-1}(\mathbf{c_t} - \mathbf{z_t})).\end{cases} \quad (۱۴)$$

$$\text{Kernel}_s, \text{Enc}_s:\begin{cases} z_t = \mathbf{z_t^{(1)}}\\ c_t = \mathbf{c_t^{(1)}}\\ \mathbf{s_{t+1}} = \mathbf{D}\mathbf{s_t} + \mathbf{A}\,\wp_k(\mathbf{s_t}).\end{cases} \quad (۱۵)$$

کارکرد خودهمزمان

بر اساس تعریف (۲) یک رمز دنباله‌ای خودهمزمان است اگر تابع مولد دنباله کلید فقط وابسته به کلید مخفی و تعداد محدودی (مثلاً n_0) از نمادهای متن رمزی قبلی باشد. به‌طور دقیق‌تر این خاصیت زمانی به‌وجود می‌آید که متغیر حالت درونی رمزگشا، در هر واحد زمانی فقط شامل n_0 متن رمزی قبلی باشد. چنین خاصیتی موجب می‌شود که رمزگشا پس از دریافت n_0 نماد متن رمزی، به‌طور خودکار با رمزگذار همزمان شود.

نتیجه دیگر این است که رمزگشا می‌تواند بدون داشتن مقدار اولیه و در نقش یک ناظر، متن اصلی را (حتی در صورت تغییرات احتمالی متن رمزی روی کانال که ممکن است ناشی از اختلالات طبیعی کانال یا دست‌کاری مهاجم باشد)، به‌درستی بازیابی کند. البته در این نوع کارکرد، تضمین خودهم زمانی به‌وجود شرایط خاصی در معادلات هسته، رمزگذار و رمزگشا وابسته است. معادلات هسته و رمزگذار سامانه در SSPLC به‌صورت (۱۶) و (۱۷) و معادلات هسته و رمزگشا در آن به‌صورت (۱۸) و (۱۹) هستند. شرایط لازم برای برقراری خودهمزمانی در معادلات سامانه SSPLC بدین صورت بیان می‌شوند:

$$\text{EKernel}_{ss}:\begin{cases}\text{setup}: \pi_k \leftarrow Init(\psi,k)\\ \text{update}:\begin{cases} initial: \pi_k, \mathbf{s_1}, \tilde{\mathbf{c}}_\mathbf{1}, \mathbf{A}, \mathbf{B}, \mathbf{D}, \mathbf{E}, \mathbf{F}, \mathbf{W}\\ \mathbf{s_{t+1}} = \mathbf{W}\tilde{\mathbf{c}}_\mathbf{t} + \mathbf{D}\mathbf{s_t} + \mathbf{A}\,\wp_k(\mathbf{s_t}) + \mathbf{E}\,\wp_k(\mathbf{p_t})\\ \tilde{\mathbf{c}}_{\mathbf{t+1}} = \mathbf{M}\tilde{\mathbf{c}}_\mathbf{t} + \mathbf{c}_{\mathbf{t\text{-}1}}\\ \mathbf{z_t} = \mathbf{W}\tilde{\mathbf{c}}_\mathbf{t} + \mathbf{B}\,\wp_k(\mathbf{s_t}), \end{cases}\end{cases} \quad (۱۶)$$

$$\mathrm{Enc}_{ss}:\begin{cases}\text{Input} \;:\; < p_t >\\ \text{Kernel}\\ \text{output}: \mathbf{c_t} = \mathbf{z_t} + \mathbf{F}\,\wp_k(\mathbf{p_t}),\end{cases} \tag{۱۷}$$

$$\mathrm{DKernel}_{ss}:\begin{cases}\text{setup}: \pi_k \leftarrow Init(\psi, k)\\ \text{update}:\begin{cases} initial: \pi_k, \hat{\mathbf{s}}_1, \tilde{\mathbf{c}}_1, \mathbf{A}, \mathbf{B}, \mathbf{D}, \mathbf{E}, \mathbf{F}, \mathbf{W}\\ \hat{\mathbf{s}}_{t+1} = \mathbf{W}\tilde{\mathbf{c}}_t + \mathbf{D}\hat{\mathbf{s}}_t + \mathbf{A}\,\wp_k(\hat{\mathbf{s}}_t) + \mathbf{E}\mathbf{F}^{-1}(\mathbf{c}_t - \hat{\mathbf{z}}_t)\\ \tilde{\mathbf{c}}_{t+1} = \mathbf{M}\tilde{\mathbf{c}}_t + \mathbf{c}_{t-1}\\ \hat{\mathbf{z}}_t = \mathbf{W}\tilde{\mathbf{c}}_t + \mathbf{B}\,\wp_k(\hat{\mathbf{s}}_t),\end{cases}\end{cases} \tag{۱۸}$$

$$\mathrm{Dec}_{ss}:\begin{cases}\text{Input} \;:\; < c_t >\\ \mathrm{DKernel}_{ss}\\ \text{output}: \hat{\mathbf{p}}_t = \wp_k^{-1}(\mathbf{F}^{-1}(\mathbf{c}_t - \hat{\mathbf{z}}_t)).\end{cases} \tag{۱۹}$$

در انتهای این بخش در یک جدول، کارایی و امنیت SSPLC با یک رمز دنباله‌ای مشابه مقایسه خواهند شد. در این نوع کارکرد، کلید مخفی مطابق با (۳) شامل تعداد دلخواهی از عناصر ماتریس $\mathbf{E}$ است. حالت درونی SSPLC شامل یک حافظه و یک متغیر حالت است. برای تعیین شرایط خودهمزمانی معادلات هسته و رمزگذار، لازم است لم زیر اثبات شود.

لم ۵-۱ الف) در سامانه PLC درجه وابستگی[1] [۲] برابر با صفر است.

ب) در سامانه PLC می‌توان ورودی متن اصلی را به‌طور منحصربه‌فرد با حالت درونی و خروجی متن رمزی توصیف کرد.

ج) اگر در سامانه (۵) شرط $\mathbf{A} = \mathbf{E}\mathbf{F}^{-1}\mathbf{B}$ برقرار باشد آن‌گاه در آن می‌توان بردار حالت را برحسب متن رمزی و مقادیر ثابت بدین‌صورت نوشت:

$$\mathbf{s_{t+1}} = \sum_{\mathbf{j=1}}^{\mathbf{t}} \mathbf{D^{j-1}}[(\mathbf{I} - \mathbf{E}\mathbf{F^{-1}})\mathbf{W}\tilde{\mathbf{c}}_{\mathbf{t-j+1}} + \mathbf{E}\mathbf{F^{-1}}\mathbf{c_{t-j+1}}] + \mathbf{D^t}\mathbf{s_1} \tag{۲۰}$$

برهان: الف) از دستگاه معادلات (۵) دیده می‌شود که تساوی زیر به ازای $r = 0$ برقرار است:

$$\mathbf{c_{t+r}} = \mathbf{c_t} = \mathbf{W}\tilde{\mathbf{c}}_\mathbf{t} + \mathbf{B}\,\wp_k(\mathbf{s_t}) + \mathbf{F}\,\wp_k(\mathbf{s_t}) = \varepsilon_k^{(r)}(\mathbf{s_t}, \mathbf{p_t})$$

ب) برای این منظور از (۶) این نتیجه گرفته می‌شود:

$$\mathbf{p_t} = \wp_k^{-1}(\mathbf{F}^{-1}(\mathbf{c_t} - \mathbf{W}\tilde{\mathbf{c}}_\mathbf{t} - \mathbf{B}\,\wp_k(\mathbf{s_t})))$$

به‌منظور اثبات وجود خاصیت خودهمزمانی در معادلات هسته SSPLC، ابتدا لازم است به‌گونه‌ای خاصیت بازگشتی بین حالت‌های درونی سامانه را حل کنیم و حالت درونی را به شکل صریحی از سایر مؤلفه‌های سامانه توصیف کنیم. این کار در زیر انجام شده است.

ج) اثبات با استقرا روی $t = 1, 2, \cdots$ انجام می‌شود. ابتدا به‌عنوان فرض پایه برای $t = 1$ رابطهٔ بازگشتی فوق اثبات می‌شود.

$$\mathbf{s_2} = \mathbf{W}\tilde{\mathbf{c}}_1 + \mathbf{A}\,\wp_k(\mathbf{s_1}) + \mathbf{E}\,\wp_k(\mathbf{p_1}) + \mathbf{D}\mathbf{s_1}$$

1 . Relative degree

از معادلهٔ دوم (۵) مقدار $\mathbf{E}\wp_k(\mathbf{p_1})$ محاسبه و در عبارت بالا جای‌گزین می‌شود،

$$\mathbf{s_2} = \mathbf{W\tilde{c}_1} + \mathbf{A}\wp_k(\mathbf{s_1}) + \mathbf{EF^{-1}c_1} - \mathbf{EF^{-1}z_1} + \mathbf{Ds_1}$$

از معادلهٔ اول (۵) مقدار $\mathbf{z_t}$ جای‌گزین شده و با استفاده از شرط (ج) نتیجه ساده می‌شود،

$$\begin{aligned}\mathbf{s_2} &= \mathbf{W\tilde{c}_1} + \mathbf{A}\wp_k(\mathbf{s_1}) + \mathbf{EF^{-1}c_1} - \mathbf{EF^{-1}W\tilde{c}_1} - \mathbf{EF^{-1}B}\wp_k(\mathbf{s_1}) + \mathbf{Ds_1}\\ &=(\mathbf{I\text{-}EF^{-1}})\mathbf{W\tilde{c}_1} + (\mathbf{A}-\mathbf{EF^{-1}B})\wp_k(\mathbf{s_1}) + \mathbf{EF^{-1}c_1} + \mathbf{Ds_1}\\ &=\sum_{\mathbf{j=1}}^{\mathbf{1}}\mathbf{D^{j\text{-}1}}[(\mathbf{I\text{-}EF^{-1}})\mathbf{W\tilde{c}_{1\text{-}j+1}} + \mathbf{EF^{-1}c_{1\text{-}j+1}}] + \mathbf{Ds_1}\end{aligned}$$

و به‌این‌ترتیب فرض پایه اثبات می‌شود. اکنون مرحلهٔ بعدی استقراء انجام می‌شود. فرض می‌شود حکم به‌ازای $t = n$ برقرار باشد،

$$\mathbf{s_n} = \sum_{\mathbf{j=1}}^{\mathbf{n\text{-}1}}\mathbf{D^{j\text{-}1}}[(\mathbf{I}-\mathbf{EF^{-1}})\mathbf{W\tilde{c}_{n\text{-}1\text{-}j+1}} + \mathbf{EF^{-1}c_{n\text{-}1\text{-}j+1}}] + \mathbf{D^{n\text{-}1}s_1} \qquad (۲۱)$$

نشان داده خواهد شد به‌ازای $t = n+1$ هم حکم برقرار است. حالت $\mathbf{s_{n+1}}$ با استفاده از (۵) نوشته‌شده و مقدار (۲۱) در جملهٔ آخر آن جای‌گذاری می‌شود،

$$\begin{aligned}\mathbf{s_{n+1}} &= \mathbf{W\tilde{c}_n} + \mathbf{A}\wp_k(\mathbf{s_n}) + \mathbf{E}\wp_k(\mathbf{p_n}) + \mathbf{Ds_n}\\ &=\mathbf{W\tilde{c}_n} + \mathbf{A}\wp_k(\mathbf{s_n}) + \mathbf{E}\wp_k(\mathbf{p_n})\\ &+\mathbf{D}[\sum_{\mathbf{j=1}}^{\mathbf{n\text{-}1}}\mathbf{D^{j\text{-}1}}[(\mathbf{I}-\mathbf{EF^{-1}})\mathbf{W\tilde{c}_{n\text{-}1\text{-}j+1}} + \mathbf{EF^{-1}c_{n\text{-}1\text{-}j+1}}] + \mathbf{D^{n\text{-}1}s_1}]\\ &=\mathbf{W\tilde{c}_n} + \mathbf{A}\wp_k(\mathbf{s_n}) + \mathbf{E}\wp_k(\mathbf{p_n})\\ &+\sum_{\mathbf{j=1}}^{\mathbf{n}}\mathbf{D^{j-1}}[(\mathbf{I}-\mathbf{EF^{-1}})\mathbf{W\tilde{c}_{n\text{-}1\text{-}j+1}} + \mathbf{EF^{-1}c_{n\text{-}1\text{-}j+1}}] + \mathbf{D^{n}s_1}\end{aligned}$$

مجدداً از معادلهٔ دوم (۵) مقدار $\mathbf{E}\wp_k(\mathbf{p_n})$ محاسبه‌شده و با عملیات مشابه در محاسبه $\mathbf{s}_2$ در عبارت بالا جای‌گزین می‌شود،

$$\begin{aligned}\mathbf{s_{n+1}} &=(\mathbf{I}-\mathbf{EF^{-1}})\mathbf{W\tilde{c}_n} + (\mathbf{A}-\mathbf{EF^{-1}})\wp_k(\mathbf{s_n}) + \mathbf{EF^{-1}c_n}\\ &+\sum_{\mathbf{j=1}}^{\mathbf{n}}\mathbf{D^{j}}[(\mathbf{I}-\mathbf{EF^{-1}})\mathbf{W\tilde{c}_{n\text{-}1\text{-}j+1}} + \mathbf{EF^{-1}c_{n\text{-}1\text{-}j+1}}] + \mathbf{D^{n}s_1}\\ &=\sum_{\mathbf{j=1}}^{\mathbf{n}}\mathbf{D^{j\text{-}1}}[(\mathbf{I}-\mathbf{EF^{-1}})\mathbf{W\tilde{c}_{n\text{-}j+1}} + \mathbf{EF^{-1}c_{n\text{-}j+1}}] + \mathbf{D^{n}s_1}\end{aligned}$$

به‌این‌ترتیب حکم قسمت (ج) اثبات شد.

قضیه بعدی و نتیجه آن شرایط لازم برای وجود خاصیت خودهم‌زمانی در SSPLC را بیان می‌کند.

قضیه ۵-۱. فرض کنید در سامانه SSPLC هر دو شرط زیر برقرار باشند،

۱. تساوی $\mathbf{A} = \mathbf{EF^{-1}B}$ برقرار باشد.

۲. ماتریس $\mathbf{D}$ پوچ‌توان از مرتبه $n_0 > 0$ باشد.

در این صورت

الف) به‌ازای هر مقدار $1 \le t \le n_0$ بردار حالت $\mathbf{s_{t+1}}$ بدین‌صورت وابسته به $\ell + t$ مقدار متن رمزی قبلی است:

$$\mathbf{s_{t+1}} = \phi_1(\mathbf{c_{1\text{-}\ell}^{(1)}}, \cdots, \mathbf{c_t^{(1)}}) \qquad (۲۲)$$

ب) به ازای هر مقدار $t > n_0 + 1$ بردار حالت $\mathbf{s_{t+1}}$ بدین‌صورت وابسته به $\ell + n_0$ مقدار متن رمزی قبلی است:

$$\mathbf{s_{t+1}} = \phi_2(\mathbf{c^{(1)}_{t-n_0+1-\ell}}, \cdots, \mathbf{c^{(1)}_t}) \qquad (۲۳)$$

ج) عمل‌کرد رمزگشای (۱۹) در نقش ناظر صحیح خواهد بود. به‌عبارت‌دیگر رمزگشا می‌تواند پس از n_0 گام، بازیابی متن اصلی را مجدداً به‌درستی انجام دهد و $\hat{p}_t = p_t$ خواهد بود.

برهان: الف) با توجه به‌شرط ۱ قضیه، بر اساس نتیجهٔ (ج) لم ۵-۱ شکل حالت درونی (۲۰) در نظر گرفته می‌شود.

برای اثبات قسمت الف قضیه، با توجه به تعریف بردار سطری $\tilde{\mathbf{c}}_t$ بدین‌صورت:

$$\tilde{\mathbf{c}}_\mathbf{t} = [\tilde{c}_t^{(\ell)}, \tilde{c}_t^{(\ell-1)}, \cdots, \tilde{c}_t^{(2)}, \tilde{c}_t^{(1)}]^{\mathrm{T}} \overset{def}{=} [\mathbf{c^{(1)}_{t-1}}, \mathbf{c^{(1)}_{t-2}}, \cdots, \mathbf{c^{(1)}_{t-\ell+1}}, \mathbf{c^{(1)}_{t-\ell}}]^{\mathrm{T}}$$

می‌توان به‌ازای $1 \le j \le t$ بردار $\tilde{\mathbf{c}}_{\mathbf{t-j+1}}$ را بدین‌صورت نوشت:

$$\tilde{\mathbf{c}}_{\mathbf{t-j+1}} = [\mathbf{c^{(1)}_{t-j}}, \mathbf{c^{(1)}_{t-j-1}}, \cdots, \mathbf{c^{(1)}_{t-j+1-\ell}}]^{\mathrm{T}}.$$

اکنون با توجه به جمله اول تحت سیگما که شامل $\tilde{\mathbf{c}}_{\mathbf{t-j+1}}$ است و با توجه به جمله دوم تحت سیگما که شامل $\mathbf{c}_{\mathbf{t-j+1}}$ است، می‌توان نتیجه گرفت که مقدار حالت درونی تابعی از متن رمزی است یا به‌عبارت‌دیگر

$$\mathbf{s_{t+1}} \overset{\mathbf{def}}{=} \phi_1(\mathbf{c^{(1)}_{1-\ell}}, \cdots, \mathbf{c^{(1)}_t}) \qquad (۲۴)$$

برقرار است و حکم قسمت الف قضیه اثبات می‌شود.

(ب) برای اثبات با استفاده از نتیجه (ج) لم (۵-۱) فرم صریح حالت درونی (۲۰) مجدداً در نظر گرفته می‌شود. مقدار n_0 با توجه به‌شرط 2 قضیه در نظر گرفته‌شده است. نشان داده می‌شود به‌ازای $t \ge n_0 + 1$ حکم قسمت ب قضیه صحیح است. مقدار $\mathbf{s_{t+1}}$ به ازای $t = n_0 + 1$ بسط داده می‌شود،

$$\begin{aligned}
\mathbf{s_{n_0+1}} = {} & \mathbf{D^0[(I - EF^{-1})W\tilde{c}_{n_0+1} + EF^{-1}c_{n_0+1}]} + \\
& \mathbf{D^1[(I - EF^{-1})W\tilde{c}_{n_0} + EF^{-1}c_{n_0}]} + \\
& \vdots + \\
& \mathbf{D^{n_0-1}[(I - EF^{-1})Wc_2 + EF^{-1}c_2]} + \\
& \mathbf{D^{n_0}[(I - EF^{-1})Wc_1 + EF^{-1}c_1]} + \\
& \mathbf{D^{n_0+1}s_1}
\end{aligned}$$

با توجه به‌شرط ۲ قضیه، دو جمله آخر در بسط حذف می‌شوند و به‌این‌ترتیب s_{n_0+2} شامل n_0 جمله خواهد بود. می‌توان به‌سادگی نشان داد به ازای هر $t \ge n_0 + 1$ تعداد جملات برابر با n_0 خواهد بود. با توجه به مقدار اندیس متغیرهای $\tilde{\mathbf{c}}$ و $\mathbf{c}$ در جملات بالا و ارتباط آن‌ها با توان ماتریس $\mathbf{D}$ دو اندیس متغیر حالت، می‌توان برای حالت $\mathbf{s_{n_0+\nu}}$ که در آن $\nu \ge 1$ است فرم بسط یافته کلی را نوشت:

$$\begin{aligned}\mathbf{s}_{\mathbf{n_0}+\nu} &= \mathbf{D}^{\mathbf{0}}[(\mathbf{I}-\mathbf{EF}^{-1})\mathbf{W}\tilde{\mathbf{c}}_{\mathbf{n_0}+\nu-1}+\mathbf{EF}^{-1}\mathbf{c}_{\mathbf{n_0}+\nu-1}]+\\ &\quad \mathbf{D}^{\mathbf{1}}[(\mathbf{I}-\mathbf{EF}^{-1})\mathbf{W}\tilde{\mathbf{c}}_{\mathbf{n_0}+\nu-2}+\mathbf{EF}^{-1}\mathbf{c}_{\mathbf{n_0}+\nu-2}]+\\ &\quad \vdots +\\ &\quad \mathbf{D}^{\mathbf{n_0-1}}[(\mathbf{I}-\mathbf{EF}^{-1})\mathbf{W}\mathbf{c}_{\nu}+\mathbf{EF}^{-1}\mathbf{c}_{\nu}]\\ &\overset{def}{=} \phi_2(\mathbf{c}^{(1)}_{\mathbf{\nu-\ell}},\cdots,\mathbf{c}^{(1)}_{\mathbf{n_0+\nu-1}})\end{aligned} \qquad (۲۵)$$

با استدلال مشابه با قسمت الف قضیه، کم‌ترین مقدار اندیس $\mathbf{c}$ مربوط به $\tilde{\mathbf{c}}_{\nu}$ است یعنی $\mathbf{c}^{(1)}_{\nu-\ell}$ و بیش‌ترین مقدار اندیس $\mathbf{c}$ مربوط به $\mathbf{c}_{n_0+\nu-1}$ است. بنا بر این به‌ازای $t \geq n_0+1$ می‌توان نوشت:

$$\mathbf{s_{t+1}} = \phi_2(\mathbf{c}^{(1)}_{\mathbf{t-n_0-\ell+1}},\cdots,\mathbf{c}^{(1)}_{\mathbf{t}}) \qquad (۲۶)$$

و حکم قسمت ب قضیه برقرار می‌شود.

(ج) برای اثبات فرض می‌شود به‌ازای برخی مقادیر $t=1,2,\cdots$ مقدار خطا در کانال $\mathbf{e_{t+1}} = \mathbf{s_{t+1}} - \hat{\mathbf{s}}_{\mathbf{t+1}}$ مقداری مثبت باشد. با در نظر گرفتن دو دستگاه (۱٦) و (۱۸) مقدار خطا بدین‌صورت نوشته می‌شود:

$$\begin{aligned}\mathbf{e_{t+1}} &= \mathbf{s_{t+1}} - \hat{\mathbf{s}}_{\mathbf{t+1}}\\ &= \mathbf{W}\tilde{\mathbf{c}}_{\mathbf{t}} + \mathbf{Ds_t} + \mathbf{A}\,\wp(\mathbf{s_t}) + \mathbf{E}\,\wp(\mathbf{p_t})\\ &\quad - (\mathbf{W}\tilde{\mathbf{c}}_{\mathbf{t}} + \mathbf{D}\hat{\mathbf{s}}_{\mathbf{t}} + \mathbf{A}\,\wp(\hat{\mathbf{s}}_{\mathbf{t}}) + \mathbf{EF}^{-1}(\mathbf{c_t} - \hat{\mathbf{z}}_{\mathbf{t}}))\\ &= \mathbf{De_t} + \mathbf{A}\,\wp(\mathbf{s_t}) + \mathbf{E}\,\wp(\mathbf{p_t})\\ &\quad - \mathbf{A}\,\wp(\hat{\mathbf{s}}_{\mathbf{t}}) + \mathbf{EF}^{-1}(\mathbf{c_t} - \hat{\mathbf{z}}_{\mathbf{t}})\end{aligned} \qquad (۲۷)$$

با جای‌گذاری مقدار $\mathbf{c_t}$ از (۱۶) و مقدار $\hat{\mathbf{z}}_{\mathbf{t}}$ از (۱۸) نتیجه گرفته می‌شود،

$$\begin{aligned}\mathbf{e_{t+1}} &= \mathbf{De_t} + \mathbf{A}\,\wp(\mathbf{s_t}) + \mathbf{E}\,\wp(\mathbf{p_t})\\ &\quad - \mathbf{A}\,\wp(\hat{\mathbf{s}}_{\mathbf{t}}) - \mathbf{EF}^{-1}(\mathbf{z_t} + \mathbf{F}\,\wp_k(\mathbf{p_t}) - \mathbf{W}\tilde{\mathbf{c}}_{\mathbf{t}} - \mathbf{B}\,\wp_k(\hat{\mathbf{s}}_{\mathbf{t}}))\\ &= \mathbf{De_t} + \mathbf{A}\,\wp(\mathbf{s_t}) - \mathbf{EF}^{-1}(\mathbf{z_t} - \mathbf{W}\tilde{\mathbf{c}}_{\mathbf{t}})\end{aligned} \qquad (۲۸)$$

مجدداً با جای‌گذاری مقدار $\mathbf{z_t}$ از (۱٦) دیده می‌شود،

$$\begin{aligned}\mathbf{e_{t+1}} &= \mathbf{De_t} + \mathbf{A}\,\wp(\mathbf{s_t}) - \mathbf{EF}^{-1}(\mathbf{W}\tilde{\mathbf{c}}_{\mathbf{t}} + \mathbf{B}\,\wp_k(\mathbf{s_t}) - \mathbf{W}\tilde{\mathbf{c}}_{\mathbf{t}})\\ &= \mathbf{De_t} + \mathbf{A}\,\wp(\mathbf{s_t}) - \mathbf{EF}^{-1}\mathbf{B}\,\wp_k(\mathbf{s_t})\end{aligned} \qquad (۲۹)$$

چون با استفاده از شرط الف قضیه (۵-۱) این تساوی برقرار است:

$$\mathbf{e_{t+1}} = \mathbf{De_t} \qquad (۳۰)$$

از شرط ب قضیه (۵-۱) و پس از n_0 واحد زمانی (چون $\mathbf{D^{n_0}} = \mathbf{0}$ است) نتیجه می‌شود که $\mathbf{e_{t+1}} = \mathbf{0}$ است و بنا بر این $\mathbf{s_{t+1}} = \hat{\mathbf{s}}_{\mathbf{t+1}}$ و از آن‌جا $\mathbf{p_{t+1}} = \hat{\mathbf{p}}_{\mathbf{t+1}}$ خواهد شد و درنتیجه حکم قضیه اثبات می‌شود.

به‌این‌ترتیب با توجه به نتایج قبلی، نشان داده شد که در صورت وجود خطا روی کانال، خودهمزمانی به‌طور خودکار انجام خواهد پذیرفت.

نتیجه ۵-۱- سامانه PLC در صورت وجود شرایط قضیه (۵-۱) تبدیل به SSPLC می‌شود که معادل یک رمز دنباله‌ای خودهمزمان استاندارد (۲) خواهد بود.

برهان: با استفاده از نتایج قضیه (۵-۱) روشن می‌شود که در صورت برقراری شرایط قضیه، حالت درونی رمز همواره به تعدادی متناهی از متون رمزی قبلی وابسته است. به‌عبارت‌دیگر PLC با استفاده از شرایط قضیه (۵-۱) بدین‌صورت با شکل استاندارد (۲) معادل می‌شود:

$$\begin{cases} z_t = \mathbf{z_t^{(1)}} \\ c_t = \mathbf{c_t^{(1)}} \\ \mathbf{s_{t+1}} = \varphi_2(\mathbf{c_{t-n_0-\ell+1}^{(1)}}, \ldots, \mathbf{c_t^{(1)}}) \end{cases} \qquad (۳۱)$$

و در (۳۱) مطابق با دستگاه (۲) حالت درونی در زمان $t+1$ تابعی کلیددار از تعداد محدودی متن رمزی قبلی و دنباله کلید هم در زمان t تابعی کلیددار از حالت درونی در زمان t و تابع رمزگذاری ε در زمان t فقط به دنباله کلید z_t و متن اصلی p_t وابسته است.

چنان‌که در بالا دیده شد، سامانه رمز دنباله‌ای PLC به‌گونه‌ای انعطاف‌پذیر است که می‌تواند به‌وسیلهٔ تغییر پارامترها (ماتریس‌های انتخابی) کاربر به‌طور دلخواه به شکل همزمان یا خودهمزمان به‌کار برده شود. در جدول ٤ سامانه SSPLC معرفی‌شده از نظر تعدادی از پارامترهای امنیتی مانند اندازهٔ کلید و اندازهٔ حالت درونی و برخی از پارامترهای کارایی مانند سرعت خروجی با سامانه خودهمزمان Moustique که یکی از بهترین کاندیدهای مسابقه eSTREAM است و الگوریتم آن تقریباً هیچ نقطه‌ضعف مهمی نداشته است [۱۴] ، مقایسه شده است. همان‌طور که در جدول مشاهده می‌شود، SSPLC با اندازهٔ کلید و حالت درونی برابر، ۱۰ بار سریع‌تر از Moustique است (محاسبات جدول ٤ با شمارش تعداد عملیات اصلی به‌کاررفته به‌ازای هر بیت خروجی انجام شده است).

جدول ۴. مقایسهٔ پارامترهای امنیتی و کارایی SSPLC و Moustique

	SSPLC	Moustique
تعداد عناصر میدان	۱٦	۲
اندازهٔ کلید	۹۸	۹٦
اندازهٔ حالت درونی	۹٦	۱۲۸
تعداد سیکل برای هر بیت خروجی	۱۵۰	۱۵۰۰

آزمایش‌های آماری

این بخش شامل دو قسمت است. در قسمت اول روی دنباله دودویی تولید شده با نگاشت شبه آشوبی رینی [۴] و دنباله دودویی تولید شده با نگاشت آشوبی (۱۰) آزمایش‌های آماری استاندارد (NIST SP-800.22) [۲۷] انجام شده است که نتایج این آزمایش‌ها نشان می‌دهند که دنباله‌های دودویی تولید شده با هر دو نگاشت خواص آماری مشابهی با یک‌دیگر دارند. در قسمت دوم آزمایش‌های آماری استاندارد بر روی خروجی سامانه PLC انجام شده، که نتایج آن نشان می‌دهد ترکیب بخش خطی با بخش آشوبی در سامانه، باعث ارتقاء امنیت و کارایی کلی سامانه شده است. در انتهای هر قسمت، خلاصه نتایج در جداول مربوط ارائه‌شده است.

آزمایش‌های آماری **نگاشت شبه آشوبی**

آدابو از روشی دومرحله‌ای به‌شرح زیر برای ساختن یک نگاشت شبه آشوبی استفاده کرد [۴]. او در ابتدا با فرض این‌که $n \in \mathbb{N}$ و $n>1$ باشد یک دامنه گسسته Λ_{2^n} را به‌صورت مجموعه‌ای از اعداد گویا تعریف کرد:

$$\Lambda_{2^n} = \{\frac{i}{2^n} \in \mathbb{Q} \Big| 0 \le i < 2^n\}$$

آدابو سپس نگاشت گسسته $\tilde{\psi}_b : \Lambda_{2^n} \to \Lambda_{2^n}$ را برای هر $0 \le i < 2^n$ بدین‌صورت تعریف کرد:

$$\tilde{\psi}_b(\frac{i}{2^n}) = \frac{1}{2^n}(bi \mod 2^n) \qquad (۳۲)$$

با توجه به مشابهت $\tilde{\psi}_b$ با نگاشت (۱۰) مورداستفاده در این مقاله، به‌منظور مقایسهٔ خواص آماری دو نگاشت، آزمایش‌ها مشابهی بر روی دنباله‌های تولیدشده با هر یک از این دو نگاشت انجام شده است. برای این کار ابتدا از هر یک از دو نگاشت صد دنباله حقیقی (با مقادیر اولیه متمایز)، در نظر گرفته و سپس با استفاده از روش گسسته‌سازی [۴]، دنباله‌های دودویی مربوطه تولیدشده و با استفاده از یک نرم‌افزار آماری استاندارد (آرمان- نسخه ۲،۲) تحلیل شده‌اند. در این‌جا به‌منظور رعایت اختصار از ذکر جزئیات مربوط به روش گسسته‌سازی و پارامترهای آماری انتخاب شده خودداری شده است [۴].

جدول ۵. نتایج آزمون NIST برای نگاشت رینی [۴]

Test Name	Average	χ^2prop	Result
Frequency	%99.00	0.0000	**Pass**
Frequency within a Block	%99.00	0.0000	**Pass**
Runs	%99.00	0.0000	**Pass**
Binary Matrix Rank	%97.00	4.0404	**Pass**
Non Overlapping Template Matching	%100.00	1.0101	**Pass**
Overlapping Template Matching	%100.00	1.0101	**Pass**
Linear Complexity	%100.00	1.0101	**Pass**
Serial	%99.00	0.0000	**Pass**
Approximate Entropy	%99.00	0.0000	**Pass**
Cumulative Sums Forward	%96.00	9.0909	**Pass**
Cumulative Sums Backward	%99.00	0.0000	**Pass**
Random Excursions	%99.00	0.0000	**Pass**
Random Excursions Variant	%99.00	0.0000	**Pass**

همهٔ آزمون‌ها برای هر یک از دو نگاشت، روی نمونه‌ای با اندازهٔ ۱۰۰ دنباله دودویی با طول ۲۰۰۰،۰۰۰ بیت و پارامتر $prec = 32$ اجرا شده‌اند. در جدول ۵ نتایج تحلیل برای نگاشت شبه آشوبی $\tilde{\psi}_3$ و در جدول ٦ نتایج تحلیل برای نگاشت (۱۰) ارائه شده‌اند. لازم به ذکر است که نتایج به‌دست آمده در جدول‌های ٥ و ٦ نوعی هستند و ده‌ها نمونه دیگر با مقادیر $prec = 32,64$ و $b = 3,7,11,17$ آزمون شده و نتایج مشابهی به‌دست‌آمده است. به‌طور خاص نتایج جدول ٥ با نتایج جدول ۱ و ۲ در [۴] مطابقت دارد.

جدول ۶. نتایج آزمون NIST برای نگاشت پیشنهادی (۱۰)

Test Name	Average	χ^2prop	Result
Frequency	%98.00	1.0101	**Pass**
Frequency within a Block	%98.00	1.0101	**Pass**
Runs	%98.00	1.0101	**Pass**
Binary Matrix Rank	%99.00	0.0000	**Pass**
Non Overlapping Template Matching	%100.00	1.0101	**Pass**
Overlapping Template Matching	%100.00	1.0101	**Pass**
Linear Complexity	%98.00	1.0101	**Pass**
Serial	%98.00	1.0101	**Pass**
Approximate Entropy	%98.00	1.0101	**Pass**
Cumulative Sums Forward	%98.00	1.0101	**Pass**
Cumulative Sums Backward	%98.00	1.0101	**Pass**
Random Excursions	%97.00	4.0404	**Pass**
Random Excursions Variant	%99.00	0.0000	**Pass**

آزمایش‌های آماری خروجی سامانه

برای انجام آزمون‌های آماری خروجی سامانه، یک مثال خاص از سامانه دینامیکی PLC روی میدان $\mathbb{F}_{17}$ با طول بردار $\ell = 8$ برای متغیرهای حالت و ورودی و خروجی پیاده‌سازی شده است. برای آزمایش فراوانی دنباله متن رمزی، چندین راهبُرد مختلف برای نشان دادن تأثیر پارامترهای انتخابی در طرح PLC در نظر گرفته‌شده است. در تمامی آزمون‌ها، توزیع خی دو با سطح معنی‌دار $\alpha = 0.05$ استفاده‌شده است. جدول ۷ در صد موفقیت در آزمون برازش یک‌نواختی خروجی سامانه دینامیکی PLC را با انتخاب‌های متفاوت برای پارامترهای مؤثر (ماتریس‌ها) در سامانه را نشان می‌دهد. ستون اول شمارهٔ سطر و چهار ستون بعد از آن نشان‌دهندهٔ ماتریس‌های انتخابی است. ماتریس‌های $\bar{\mathbf{A}}, \bar{\mathbf{B}}, \bar{\mathbf{D}}, \bar{\mathbf{E}}, \bar{\mathbf{W}}$ ماتریس‌هایی ناصفر هستند که به‌طور تصادفی از میدان انتخاب‌شده‌اند و نتایج در صد موفقیت سامانه از این آزمون‌ها با شرایط تعریف شده در دو ستون متن اصلی ثابت و متن اصلی تصادفی آورده شده است که این دو ستون به‌ترتیب نشان‌دهندهٔ انتخاب دنباله متن اصلی برابر با ثابت 0 و انتخاب تصادفی دنباله متن اصلی در هر زمان را نشان می‌دهد. به‌عنوان مثال در جدول ۷) سطر اول معرف سامانه دینامیکی هم‌زمان است. در این سامانه دینامیکی از یک نگاشت آشوب برای تجدید مقدار متغیر حالت استفاده شده است. نتایج حاصل از آزمایش نشان‌دهندهٔ عدم یک‌نواختی در فراوانی خروجی‌های سامانه‌های دینامیکی است که فقط از دینامیک‌های آشوبی استفاده می‌کنند. نتیجه ۷ درصدی حاصل نشان‌دهندهٔ این مفهوم است. به‌عبارت‌دیگر استفاده از یك نگاشت آشوب در میدان متناهی به‌تنهایی برای تجدید مقدار متغیر حالت در یك سامانه دینامیکی کافی نیست. به‌عنوان مثالی دیگر سطر چهارم جدول ۷ نشان‌دهندهٔ سامانه دینامیکی در حالت خودهم‌زمان است.

$$\begin{cases} \mathbf{z_t} = \mathbf{I}\,\wp_k(\mathbf{s_t}) \\ \mathbf{c_t} = \mathbf{z_t} + \mathbf{F}\,\wp_k(\mathbf{p_t}) \\ \mathbf{s_{t+1}} = \mathbf{A}\,\wp_k(\mathbf{s_t}) \end{cases} \qquad (۳۳)$$

$$\begin{cases} \mathbf{z_t} = \mathbf{1}\tilde{\mathbf{c}}_\mathbf{t} + \mathbf{B}\,\wp_k(\mathbf{s_t}) \\ \mathbf{c_t} = \mathbf{z_t} + \mathbf{F}\,\wp_k(\mathbf{p_t}) \\ \mathbf{s_{t+1}} = \mathbf{1}\tilde{\mathbf{c}}_\mathbf{t} + \mathbf{D}\mathbf{s_t} + \mathbf{A}\,\wp_k(\mathbf{s_t}) \\ \tilde{\mathbf{c}}_{\mathbf{t+1}} = \mathbf{M}\tilde{\mathbf{c}}_\mathbf{t} + c_t \end{cases} \qquad (۳۴)$$

چنان‌که در جدول ۷ مشاهده می‌شود، ماتریس $\mathbf{D}$ نقش انتشار را در این سامانه ایفا می‌کند و می‌تواند عدم یک‌نواختی در خروجی‌های روی میدان متناهی را جبران کند.

جدول ۷. نتایج آزمون خروجی سامانه در شرایط مختلف

ردیف	کارکرد	$\mathbf{B}$	$\mathbf{D}$	$\mathbf{E}$	$\mathbf{W}$	در صد موفقیت با متن اصلی ثابت	درصد موفقیت با متن اصلی تصادفی
۱	همزمان	$\mathbf{I}_{\ell\times\ell}$	$\mathbf{0}_{\ell\times\ell}$	$\mathbf{0}_{\ell\times\ell}$	$\mathbf{0}_{\ell\times\ell}$	%۷	-
۲	خودهمزمان	$\mathbf{I}_{\ell\times\ell}$	$\mathbf{0}_{\ell\times\ell}$	$\bar{\mathbf{E}}$	$\mathbf{1}_{\ell\times\ell}$	%٦	%٦
۳	خودهمزمان	$\bar{\mathbf{B}}$	$\mathbf{0}_{\ell\times\ell}$	$\mathbf{0}_{\ell\times\ell}$	$\mathbf{1}_{\ell\times\ell}$	%۹۲	%۹۵
٤	خودهمزمان	$\bar{\mathbf{B}}$	$\bar{\mathbf{D}}$	$\mathbf{0}_{\ell\times\ell}$	$\mathbf{1}_{\ell\times\ell}$	%۹۸	%۹٦
٥	همزمان	$\bar{\mathbf{B}}$	$\bar{\mathbf{D}}$	$\mathbf{0}_{\ell\times\ell}$	$\mathbf{0}_{\ell\times\ell}$	%۹۳	%۹۸
٦	خودهمزمان	$\bar{\mathbf{B}}$	$\bar{\mathbf{D}}$	$\bar{\mathbf{E}}$	$\mathbf{1}_{\ell\times\ell}$	%۹۸	%۹٦

نتایج

در این مقاله یک سامانهٔ رمز دنباله‌ای آشوبی گسسته به‌نام PLC تعریف شده است که هم به‌صورت همزمان و هم به‌صورت خودهمزمان به‌کار می‌رود. هدف ما از ارائه چنین سامانه‌ای فائق آمدن بر اجتماع سه خصوصیت مهم بوده است. اولین خصوصیت ضرورت گسسته‌سازی آشوب است که به‌علت پیاده‌سازی نرم‌افزاری سامانه رمزنگاری، می‌تواند موجب تضعیف یا از بین رفتن خاصیت آشوب شود. دومین خصوصیت ترجیح بر ارائه سامانه‌ای کلمه محور است که به‌دلیل نیاز به سرعت زیاد و پیاده‌سازی مؤثر، طراحی بر روی میدان متناهی را اجتناب‌ناپذیر می‌کند. سومین ویژگی داشتن یک طرح خودهمزمان با گیرنده از نوع ناظر با ورودی ناشناخته است که برای کنترل خطا و همزمان‌سازی خودکار به‌کار می‌رود و معمولاً روی یک میدان متناهی مسئله دشواری است. در طرح پیشنهادی با استفاده از ماتریس‌های پوچ‌توان در خودهمزمان‌سازی معادلات رمزگشا، مشکلات ناشی از کانال‌های پر اختلال را به‌طور مؤثری کاهش دادیم. همچنین نشان دادیم که با طول کلید برابر، سرعت تولید خروجی سامانه پیشنهادی ۱۰ برابر سامانه موستیکیو است. ساختمان کلی این طرح علاوه بر این‌که با داشتن یک بخش خطی و یک بخش غیرخطی با الگوی استاندارد رمزهای دنباله‌ای کاملاً منطبق است، از یکی از تقریب‌های گسسته آشوبی مناسب برای ایجاد خواص غیرخطی لازم یک طرح رمزنگاری بهره‌گیری کرده است. این طرح نه‌تنها می‌تواند به‌عنوان جای‌گزینی مناسب به‌منظور حذف برخی از نقاط ضعف رمزهای دنباله‌ای خودهمزمان بیت محور در نظر گرفته شود، بلکه می‌توان آن را به‌عنوان نمونه‌ای عملی و استاندارد برای ادامه تحقیقات در زمینه طراحی و ارزیابی رمزهای دنباله‌ای همزمان و خودهمزمان آشوبی در نظر گرفت.

منابع

۱. آزادی یزدی آذین، *بررسی امنیت سیستم رمزنگاری دنباله‌ای بر مبنای جابجایی آشوبی*، پایان‌نامهٔ کارشناسی ارشد، دانشگاه صنعتی شریف (۱۳۸۹).

۲. اقبالی حمیدرضا، *بررسی و مقایسهٔ سامانهٔ رمزنگاری دنباله‌ای سوئیچینگ*، پایان‌نامهٔ کارشناسی ارشد، دانشگاه صنعتی شریف (۱۳۹۱).

3. Addabbo T. , Alioto M. , Fort A. , "A variability-tolerant feedback technique for throughput. maximization of TRBGs with predefined Entropy", J. of Circuits, Systems and Computers 19 (4) (2010) 1-17.
4. Addabbo T. , Fort A., Rocchi S., Vignoli V., "Digitized chaos for pseudo-random number generation in cryptography", Studies in Computational Intelligence, Vol 354, (2011) 67-97.
5. Amigo J., Kocarev L., Szczepanski J., "Theory and practice of chaotic cryptography", Physic letters A, 366 (2007) 211-216.
6. Baptista M.S. , "Cryptography with chaos", Physics Letters A, 240 (1998) 50-54.
7. Blum L., Blum M., Shub M., "A simple unpredictable psuedorandom number generator", SIAM J. on Computing a5(2) (1986) 364-383.
8. Daemen J.,"Cipher and hash function design strategies based on linear and differential, cryptanalysis", PhD thesis, Katholieke Universiteit Leuven, Belgique (1995).
9. Daemen J., Govartz R., Vanderwalle J., "On the design of high speed self-synchronizing stream ciphers", ICCS/ISITA (1992).
10. Daemen J., Kitsos P., "The self-synchronizing stream cipher Mosquito: eSTREAM documentation, version 2" (2005).
11. Daemen J., Kitos P. , "The Self-synchronizing stream cipher Moustique", New stream cipher, designs, LNCS 4986 (2007) 210-223.
12. Daemen J. , LanoJ., Preneel B., "Chosen ciphertext attack on SSS", Submission to ECRYPT (2005).
13. Devaney R.L., "An Introduction to chaotic dynamical systems, 2nd Ed.", Westview Press (2003).
14. Gürkaynak F.K., et al. "Hardware evaluation of eSTREAM candidates: Achterbahn, Grain, MICKEY, MOSQUITO, SFINKS, Trivium, VEST, ZK-Crypt", Submission to ECRYPT (2006).
15. Jakimoski G., KlapperA., "Analysis of some recently proposed chaos-based encryption algorithms", Physics Letters A, 291 (2001) 381-384.

16. Jakimoski G., Klapper A., "Differential and linear probabilities of a block-encryption Cipher", IEEE Trans. on circuits and systems-I: Fundumental Theory and Applications, Vol. 50, No. 1 (2003).
17. Joux A., MullerF., "Loosening the KNOT", FSE 2003, LNCS 2887 (2003) 87-99.
18. Joux A., MullerF., "Chosen-Ciphertext attacks against MOSQUITO", FSE 2006, LNCS 4047 (2006) 390-404.
19. Kocarev L., Jakimoski G., "Logistic map as a block encryption algorithm",Physics Letters A, 289 (2001) 199-206.
20. Kocarev L., LianS., "Chaos-Based Cryptography; Theory, Algorithms and Applications", Springer (2011).
21. Kocarev L., "A chaos-based approach to the design of secure substitutions", Physics Letters A, 343 (2005).
22. Kocarev L., Szczepanski J., "Discrete Chaos I: theory", IEEE Trans. on Circuits and Systems-I: Regular papers, Vol. 53, No. 6 (2006).
23. Maurer U.m., "New approaches to the design of self-synchronizing stream ciphers", EUROCRYPT'91, 458-471, (1992).
24. Menezes A., van Oorschot P., Vanstone S., "Handbook of applied cryptography", CRC Press, (2006).
25. Millerioux G., "A connection between chaotic and conventional cryptography", in IEEE Trans. On circuits and systems, Vol. 55 No. 6, 1695-1703, (2008).
26. Millerioux G., Amigo J.M., Daafouz J., "A connection between chaotic message-embedding and conventional self-synchronizing stream ciphers", NOLTA 2006, Bologna, Italy (2006).
27. NIST Special Publication 800-22 Rev.1a, "A statistical test suite for random and psuedorandom number generators for cryptographic applications" (2010).
28. Papadimitriou S., Bezerianos A., Bountis T., Palvlides G., "Secure communication protocols with discrete chaotic map", Jour. Of systems Architecture 47 (2001) 61-72.
29. Rose G. , Hawkes P. , Paddon M., Wiggers de Vries M., "Primitive specification for SSS", Submitted eStream Project, (2005).
30. Tan P.V., Millerioux G., Daafouz J. , "A comparisson between the message embedded cryptosystem and the self-synchronous stream cipher Mosquito", 18th Euro. Conf. on Circuit Theory and Design, ECCTD'2007, Séville, Spain (2007).
31. Tan P.V., Millerioux G., Daafouz J., "Left invertibility, flatness and identifiability of switched linear dynamical systems: a framework for cryptographic applications", Intl. J. of Control 83, 1 (2010) 145-153.