TEEGES
Technovations in Electrical Engineering & Green Energy System
Research Article (2022) 1(1):81-97

Presentation of a Method for Implementing Binary Matrices and its Application in the Implementation of MDS Matrices

Mohsen Mousavi[1], *PhD*

[1]*Faculty of Applied Sciences, Malek-Ashtar University of Technology, Isfahan, Iran*

Abstract:

MDS matrices have a crucial role in the cryptography and coding theory. MDS matrices are used as the diffusion layer in cryptosystems as well as in the construction of linear codes with the maximum error correction capability. On the one hand, the entries of MDS matrices are elements of finite fields. On the other hand, it is a major issue to implement finite fields in the lightweight cryptography. Therefore, to use MDS matrices in the lightweight cryptography, these matrices are first converted to binary matrices and then implemented using heuristics algorithms. In this paper, a method to implement binary matrices with low-cost XOR is proposed and then using the proposed method, a heuristics algorithm for implementing MDS matrices is introduced. The structure of the proposed heuristics algorithm is based on the assumption that let **A** be a binary matrix (or the binary form of an MDS matrix). First, using a random-iterative method, we obtain a list **S** from a binary matrix **A**. Then, based on the list **S**, we construct a binary matrix **B**. Next, we find a relation between the implementations of **A** and **B**. In other words, using the implementation of the matrix **B**, we get a low-cost implementation for the matrix **A**. In the structure of the proposed heuristics algorithm, one of the familiar SLP algorithms called Paar is applied.

Keywords: Implementation of Binary Matrices, Heuristics Algorithms, MDS Matrix.

Received: 28 February 2022
Revised: 16 May 2022
Accepted: 16 June 2022
Corresponding Author: Dr. Mohsen Mousavi, m.mousavi@mut-es.ac.ir
DOI: http://dx.doi.org/10.30486/teeges.2022.1958154.1007

فناوری‌های نوین در مهندسی برق و سیستم انرژی سبز

TEEGES

مقاله پژوهشی

ارائه یک روش برای پیاده‌سازی ماتریس‌های دودویی و کاربرد آن در پیاده‌سازی ماتریس‌های MDS

سید محسن موسوی[1]، دکتری

۱- مجتمع علوم کاربردی، دانشگاه صنعتی مالک اشتر، اصفهان، ایران

چکیده: ماتریس‌های MDS نقش مهمی در رمزنگاری و کدگذاری دارند. ماتریس‌های MDS به‌عنوان لایه انتشار در سیستم‌های رمزنگاری و همچنین در ساخت کدهایی با بیشترین میزان تصحیح خطا استفاده می‌شوند. ازیک‌طرف، درایه‌های ماتریس‌های MDS عناصر میدان‌های متناهی هستند. از طرف دیگر، پیاده‌سازی میدان‌های متناهی در رمزنگاری سبک‌وزن مشکل است. بنابراین برای بکار بردن ماتریس‌های MDS در رمزنگاری سبک‌وزن، در ابتدا این دسته از ماتریس‌ها را به ماتریس‌های دودویی تبدیل نموده و در ادامه با استفاده از الگوریتم‌های ابتکاری، پیاده‌سازی می‌شوند. در این مقاله، یک روش برای پیاده‌سازی ماتریس‌های دودویی با هزینه XOR کم پیشنهادشده و در ادامه با استفاده از روش پیشنهادی، یک الگوریتم ابتکاری برای پیاده‌سازی ماتریس‌های MDS معرفی می‌گردد. عملکرد الگوریتم ابتکاری معرفی‌شده بر این اساس است که فرض کنید **A** یک ماتریس دودویی (یا شکل دودویی یک ماتریس MDS) باشد. در ابتدا با استفاده از یک روش تکراری-تصادفی یک لیست **S** از ماتریس دودویی **A** به دست می‌آید. سپس، با استفاده از لیست **S** یک ماتریس دودویی به نام **B** تشکیل می‌گردد. در ادامه یک ارتباط بین پیاده‌سازی ماتریس‌های **A** و **B** پیدا می‌شود. به‌عبارت‌دیگر با استفاده از پیاده‌سازی ماتریس **B** یک پیاده‌سازی کم‌هزینه برای ماتریس **A** ارائه می‌گردد. در ساختار الگوریتم ابتکاری پیشنهادشده از یکی از الگوریتم‌های متداول SLP به نام Paar استفاده‌شده است.

واژه های کلیدی: پیاده‌سازی ماتریس‌های دودویی، الگوریتم‌های ابتکاری، ماتریس MDS

تاریخ ارسال مقاله: ۱۴۰۰/۱۲/۰۹

تاریخ بازنگری مقاله: ۱۴۰۱/۰۲/۲۶

تاریخ پذیرش مقاله: ۱۴۰۱/۰۳/۲۶

نویسنده‌ی مسئول: دکتر سید محسن موسوی، m.mousavi@mut-es.ac.ir

DOI: http://dx.doi.org/10.30486/teeges.2022.1958154.1007

۱- مقدمه

طراحی لایه انتشار برای رمزهای قالبی، یکی از موضوعات موردتوجه رمزنگاری متقارن است. یکی از راه‌های متداول برای ایجاد انتشار در رمزهای قالبی، استفاده از ماتریس‌های MDS[1] است. برای ساخت ماتریس‌های MDS روش‌های متنوعی مانند استفاده از ماتریس‌های واندرموند، کوشی و شبه‌کوشی وجود دارد که شرح کامل ساخت این نوع از ماتریس‌ها در [۱] آمده است.

ساخت ماتریس‌های MDS بر روی میدان‌های متناهی انجام می‌شود. از طرفی نمی‌توان عناصر میدان‌های متناهی را در دستگاه‌های با سخت‌افزار محدود پیاده‌سازی نمود. بنابراین نیاز است تا ابتدا ماتریس‌های MDS را به ماتریس‌های دودویی تبدیل نموده و سپس با استفاده از شکل دودویی ماتریس‌های MDS، یک پیاده‌سازی مناسب برای این دسته از ماتریس‌ها، جهت استفاده در رمزنگاری سبک‌وزن به دست آورد [۲]. یکی از روش‌های پیاده‌سازی ماتریس‌های دودویی، استفاده از برنامه‌های SLP[2] مانند الگوریتم‌های Paar [۳] و یا BP [۴] است. به‌تازگی این الگوریتم‌های ابتکاری SLP [۳] بهبود داده‌شده‌اند [۵,۶].

۱-۱- مروری کوتاه بر روش پیشنهادی با استفاده از الگوریتم Paar

فرض کنید که بخواهیم یک دستگاه از معادلات دودویی محاسبه کنیم. ساده‌ترین روش این است که تمام متغیرها را با یکدیگر جمع نموده و حاصل را محاسبه کنیم. اما اشکال این روش این است اگر حجم روابط و تعداد متغیرها زیاد باشد آنگاه هزینه محاسبه زیاد می‌شود؛ بنابراین به دنبال روشی هستیم که بتواند بین متغیرهای این دستگاه از معادلات دودویی، روابط خطی پیدا نماید به‌طوری‌که هزینه محاسبه کم شود. درروش پیشنهادشده در این مقاله، ما ابتدا این دستگاه از معادلات دودویی را با اضافه کردن متغیرهای اضافی تغییر می‌دهیم به‌طوری‌که هزینه محاسبات دستگاه معادلات جدید کم شود و سپس از جواب به‌دست‌آمده برای این دستگاه جدید، متغیرهای اضافه‌شده را حذف می‌کنیم و با این کار یک پیاده‌سازی کم‌هزینه برای دستگاه اولیه به دست می‌آوریم. برای فهم بیشتر این روش پیشنهادی یک مثال کوچک آورده می‌شود. فرض کنید که بخواهیم دستگاه معادلات دودویی پایین را محاسبه کنیم. همان‌طور که دیده می‌شود برای به دست آوردن مقادیر $\mathbf{y}=[y_1,y_2,\cdots,y_8]$ باید ۲۷ تا XOR محاسبه شود.

$$\begin{aligned}
y_1 &= x_3 \oplus x_4 \oplus x_5 \oplus x_6 \oplus x_7 \oplus x_8\\
y_2 &= x_2 \oplus x_4 \oplus x_5 \oplus x_6 \oplus x_7 \oplus x_8\\
y_3 &= x_2 \oplus x_5 \oplus x_6 \oplus x_7 \oplus x_8\\
y_4 &= x_4 \oplus x_6 \oplus x_7 \oplus x_8\\
y_5 &= x_5 \oplus x_7 \oplus x_8\\
y_6 &= x_6 \oplus x_8\\
y_7 &= x_1 \oplus x_7\\
y_8 &= x_2 \oplus x_3 \oplus x_4 \oplus x_5 \oplus x_6 \oplus x_7 \oplus x_8
\end{aligned}
\qquad
\mathbf{y}^T = \mathbf{x}\begin{pmatrix}
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0\\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 1\\
1 & 0 & 1 & 0 & 0 & 0 & 0 & 1\\
1 & 1 & 0 & 1 & 0 & 0 & 0 & 1\\
1 & 1 & 1 & 0 & 1 & 0 & 0 & 1\\
1 & 1 & 1 & 1 & 0 & 1 & 0 & 1\\
1 & 1 & 1 & 1 & 1 & 0 & 1 & 1\\
1 & 1 & 1 & 1 & 1 & 1 & 0 & 1
\end{pmatrix} \qquad (1)$$

با استفاده از الگوریتم Paar می‌توان مقادیر $[y_1,y_2,\cdots,y_8]$ را با ۱۲ تا XOR محاسبه نمود که به شرح زیر است. لازم به ذکر است که ورودی الگوریتم Paar شکل ماتریسی معادلات دودویی بوده که در رابطه (۱) آورده شده است.

$$\begin{aligned}
&۱)\ y_6 = x_6 \oplus x_8, \quad ۲)\ y_7 = x_1 \oplus x_7, \quad ۳)\ t_1 = x_7 \oplus x_8, \quad ۴)\ y_5 = t_1 \oplus x_5,\\
&۵)\ t_2 = t_1 \oplus x_6, \quad ۶)\ y_4 = t_2 \oplus x_4, \quad ۷)\ t_3 = t_2 \oplus x_5, \quad ۸)\ y_3 = t_3 \oplus x_3,\\
&۹)\ t_4 = t_3 \oplus x_4, \quad ۱۰)\ y_2 = t_4 \oplus x_2, \quad ۱۱)\ y_1 = t_4 \oplus x_3, \quad ۱۲)\ y_8 = y_1 \oplus x_2.
\end{aligned}$$

اکنون اگر به y_2 و y_5 متغیرهای دودویی جدید x_3 و x_6 را به ترتیب اضافه کنیم آنگاه دستگاه معادلات دودویی جدید به شکل زیر می‌شود.

$$\begin{aligned}
y_1 &= x_3 \oplus x_4 \oplus x_5 \oplus x_6 \oplus x_7 \oplus x_8 \\
y_2 &= x_2 \oplus x_3 \oplus x_4 \oplus x_5 \oplus x_6 \oplus x_7 \oplus x_8 \\
y_3 &= x_3 \oplus x_5 \oplus x_6 \oplus x_7 \oplus x_8 \\
y_4 &= x_4 \oplus x_6 \oplus x_7 \oplus x_8 \\
y_5 &= x_5 \oplus x_6 \oplus x_7 \oplus x_8 \\
y_6 &= x_6 \oplus x_8 \\
y_7 &= x_1 \oplus x_7 \\
y_8 &= x_2 \oplus x_3 \oplus x_4 \oplus x_5 \oplus x_6 \oplus x_7 \oplus x_8
\end{aligned} \qquad \mathbf{y}^T = \mathbf{x} \cdot \begin{pmatrix}
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\
1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 \\
1 & 1 & 0 & 1 & 0 & 0 & 0 & 1 \\
1 & 1 & 1 & 0 & 1 & 0 & 0 & 1 \\
1 & 1 & 1 & 1 & 1 & 1 & 0 & 1 \\
1 & 1 & 1 & 1 & 1 & 0 & 1 & 1 \\
1 & 1 & 1 & 1 & 1 & 1 & 0 & 1
\end{pmatrix} \qquad (۲)$$

حال می‌توان بررسی نمود که هزینه پیاده‌سازی ماتریس دودویی داده‌شده در رابطه (۲) با استفاده از الگوریتم Paar برابر با ۸ تا XOR است که به‌صورت زیر به دست می‌آید:

$$\begin{aligned}
&۱)\ y_7 = x_1 \oplus x_7, \quad ۲)\ y_6 = x_6 \oplus x_8, \quad ۳)\ t_1 = y_6 \oplus x_7, \quad ۴)\ y_4 = t_1 \oplus x_4, \\
&۵)\ y_5 = t_1 \oplus x_5, \quad ۶)\ y_3 = y_5 \oplus x_3, \quad ۷)\ y_1 = y_3 \oplus x_4, \quad ۸)\ y_2 = y_8 = y_1 \oplus x_2.
\end{aligned} \qquad (۳)$$

واضح است که این پیاده‌سازی مربوط به دستگاه معادلات دودویی داده‌شده در رابطه (۱) نیست اما به‌سادگی و با XOR نمودن y_2 و y_5 با متغیرهای دودویی x_3 و x_6 می‌توان یک پیاده‌سازی با تنها ۱۰ تا XOR برای دستگاه معادلات دودویی داده‌شده در رابطه (۱) به دست آورد. لازم به ذکر است که در میدان‌های متناهی با مشخصه ۲، علامت جمع و منها یکی است.

$$\begin{aligned}
&۱)\ y_7 = x_1 \oplus x_7, \quad ۲)\ y_6 = x_6 \oplus x_8, \quad ۳)\ t_1 = y_6 \oplus x_7, \quad ۴)\ y_4 = t_1 \oplus x_4, \quad ۵)\ y_5 = t_1 \oplus x_5, \\
&۶)\ y_3 = y_5 \oplus x_3, \quad ۷)\ y_1 = y_3 \oplus x_4, \quad ۸)\ y_2 = y_8 = y_1 \oplus x_2, \quad ۹)\ y_2 = y_2 \oplus x_3, \quad ۱۰)\ y_5 = y_5 \oplus x_6.
\end{aligned}$$

۱-۲- نتایج به‌دست‌آمده

حال مهم‌ترین کاری که در این مقاله می‌خواهیم انجام دهیم پاسخ به این پرسش است: فرض کنید $\mathbf{A}$ یک ماتریس دودویی باشد. چطور و به چه نحو برخی درایه‌های آن تغییر مقدار داده شود (از صفر به یک) به‌نحوی‌که هزینه پیاده‌سازی ماتریس دودویی جدید به‌دست‌آمده، از هزینه پیاده‌سازی ماتریس $\mathbf{A}$ کمتر باشد. پاسخی که در ادامه دهیم کاملاً ابتکاری بوده و جواب قطعی برای این پرسش نیست. به‌عبارت‌دیگر، حل مسئله کاهش هزینه محاسبات معادلات دودویی یک مسئله سخت و باز است.

اگرچه که برنامه‌های SLP متنوعی وجود دارد اما در این مقاله به دو دلیل مهم از الگوریتم Paar استفاده‌شده است. دلیل اول مربوط به‌سرعت پردازش الگوریتم Paar است. درواقع با استفاده از الگوریتم Paar می‌توان ماتریس‌های دودویی با ابعاد بزرگ را پردازش نمود. دلیل دوم مربوط به ساختار الگوریتم Paar است. در هر دور از اجرای این الگوریتم، از بین کل حالت‌های ممکن، ما آن دو ستون را انتخاب می‌کنیم که بیشترین اشتراک بین درایه‌های آن‌ها وجود دارد. حال با تغییر درایه‌های صفر ماتریس به یک، انتظار داریم تعداد تکرارهای الگوریتم Paar جهت به دست آوردن یک پیاده‌سازی با هزینه کم، کاهش پیدا نماید.

از دیگر نتایج مهم این مقاله، معرفی ماتریس پایه برای ساخت شکل دودویی یک ماتریس MDS است. اگرچه که برای ساخت شکل دودویی یک ماتریس MDS، استفاده از ماتریس مشارکت [۴] یکی از راه‌های متداول بوده اما در این مقاله نشان می‌دهیم که برای به دست آوردن یک پیاده‌سازی با هزینه کم، نیاز به استفاده از ماتریس‌های پایه داریم.

سایر بخش‌های این مقاله به شرح زیر است. در بخش دوم در ابتدا، روش لیست به دست آوردن از یک ماتریس دودویی با استفاده از الگوریتم Paar را شرح می‌دهیم. در ادامه مفهوم ماتریس پایه جهت به دست آوردن شکل دودویی یک ماتریس MDS را شرح می‌دهیم. در پایان با ترکیب نمودن دو تکنیک گفته‌شده، یک الگوریتم به نام، الگوریتم لیست- ماتریس جهت به دست آوردن یک پیاده‌سازی کم‌هزینه از ماتریس‌های MDS معرفی می‌کنیم.

۱-۳- پیش‌زمینه‌ها

فرض کنید $\mathbf{A}$ یک ماتریس $n\times n$ دودویی باشد. تعداد XOR-های لازم برای پیاده‌سازی ماتریس $\mathbf{A}$ با استفاده از الگوریتم Paar را با $\mathrm{CP}(\mathbf{A})$ نشان می‌دهیم. برای مثال، ماتریس ۸×۸ دودویی در رابطه (۲) را در نظر بگیرید که آن را با $\mathbf{A}$ نشان دادیم. حال می‌توان با استفاده از رابطه (۳) نتیجه بگیریم که $\mathrm{CP}(\mathbf{A})=۸$ است.

فرض کنید که $\mathbf{S}$ یک لیست باشد. آنگاه تعداد عناصر این لیست را با نماد $\mathrm{nops}(\mathbf{S})$ نشان می‌دهیم. برای مثال لیست زیرا را در نظر بگیرید $\mathbf{S}=[\mathbf{z}_۱,\mathbf{z}_۲,\mathbf{z}_۳]$ به‌طوری‌که $\mathbf{z}_۱=[[۱,۲]]$، $\mathbf{z}_۲=[[۷,۸]]$ و $\mathbf{z}_۳=[[۵,۶],[۹,۱۰]]$ آنگاه با توجه به نماد معرفی‌شده داریم $\mathrm{nops}(\mathbf{S})=۳$، $\mathrm{nops}(\mathbf{z}_۱)=\mathrm{nops}(\mathbf{z}_۲)=۱$ و $\mathrm{nops}(\mathbf{z}_۳)=۲$ است.

فرض کنید $\mathbb{F}_q$ نشان‌دهنده یک میدان متناهی شامل q عنصر باشد که q توانی از یک عدد اول است. حال یک ماتریس مربعی مانند $\mathbf{A}$ را بر روی میدان متناهی $\mathbb{F}_q$ یک ماتریس MDS می‌نامیم اگر هر زیر ماتریس مربعی از ماتریس $\mathbf{A}$ نامنفرد باشد [۷]. ماتریس‌های MDS نقش مهمی در طراحی لایه انتشار رمزهای قالبی دارند. به‌تازگی از الگوریتم‌های Paar و BP جهت به دست آوردن پیاده‌سازی‌های کم‌هزینه ماتریس‌های MDS، استفاده‌شده است [۸]. تمام پیاده‌سازی‌های این مقاله به‌صورت عمومی در تارنمای [۹] قابل‌دسترسی و مشاهده است.

در ادامه، منظور از پیاده‌سازی یک ماتریس دودویی شرح داده می‌شود. فرض کنید $\mathbf{A}$ یک ماتریس $n\times m$ دودویی است. حال بردار $\mathbf{x}=[x_۱,x_۲,\cdots,x_n]$ را در نظر بگیرید. حاصل‌ضرب ماتریسی $\mathbf{x}.\mathbf{A}$ در پیمانه ۲ برابر با یک ماتریس $۱\times m$ مانند $\mathbf{B}$ است. هر درایه از ماتریس $\mathbf{B}$ حاصل جمع یک تعداد از x_i-ها است. به تعداد جمع‌ها بین این x_i-ها، هزینه پیاده‌سازی ماتریس $\mathbf{A}$ گویند. برای مثال، هزینه محاسبه ماتریس دودویی در رابطه (۲) برابر با ۲۷ تا XOR است.

موضوع مهم دیگری که در پیاده‌سازی ماتریس‌های MDS اهمیت دارد مفهوم عمق° ماتریس‌های MDS است. درواقع در پیاده‌سازی سخت‌افزاری یک ماتریس MDS تنها شمارش XOR اهمیت ندارد و نیاز است تا عمق ماتریس‌های MDS پیشنهادشده محاسبه گردد. مفهوم عمق در ارتباط با زمان پردازش این دسته از ماتریس‌ها است. درواقع عمق کمتر باعث عملکرد سریع‌تر در اجرای موازی می‌شود زیرا عمق یک ماتریس MDS را حداکثر گیت در هر مسیر از ابتدا تا انتها می‌گویند. اما باید توجه داشت که یک بده بستان بین عمق ماتریس و تعداد XOR ها برای پیاده‌سازی آن ماتریس وجود دارد. به عبارت دقیق‌تر، هرچه عمق ماتریس بیشتر باشد تعداد XOR ها کمتر می‌شود و بالعکس. برای مثال است کم‌هزینه‌ترین پیاده‌سازی‌هایی که تاکنون برای لایه انتشار رمز قالبی AES [۱۰] به‌دست‌آمده است به شرح زیر است: کم‌هزینه‌ترین پیاده‌سازی با عمق ۷ برابر با ۹۱ تا XOR و با عمق ۳ برابر با ۱۰۳ تا XOR است [۱۱].

۲- پیاده‌سازی ماتریس‌های دودویی

در این قسمت و با استفاده از الگوریتم Paar یک روش تکراری-تصادفی جهت پیاده‌سازی ماتریس‌های دودویی پیشنهاد می‌دهیم. سپس یک الگوریتم ابتکاری برای پیاده‌سازی ماتریس‌های MDS معرفی می‌کنیم. همچنین با استفاده از الگوریتم ابتکاری معرفی‌شده، یک ماتریس ۸×۸ MDS را با هزینه ۴۰۸ تا XOR پیاده‌سازی نموده‌ایم.

۱-۲- روش لیستی Paar

در این زیر بخش، در ابتدا مفهوم روش لیستی Paar را شرح می‌دهیم. در تعریف ۱، یک ماتریس دودویی به نام $\mathbf{A}$ را در نظر می‌گیریم و سپس تعدادی از درایه‌های صفر $\mathbf{A}$ را انتخاب می‌کنیم. در ادامه، برخی از درایه‌های صفر انتخاب‌شده از ماتریس $\mathbf{A}$ را به یک تغییر می‌دهیم. حال ماتریس جدید را $\mathbf{B}$ می‌نامیم. در ادامه مقادیر $\mathrm{CP}(\mathbf{A})$ و $\mathrm{CP}(\mathbf{B})$ مقایسه می‌کنیم.

تعریف ۱: فرض کنید $\mathbf{A}=(a_{i,j})$ یک ماتریس $n\times m$ دودویی باشد و فرض کنید r تعداد درایه‌های برابر با صفر این ماتریس باشد. فرض کنید $۱\le p_۱,p_۲,\cdots,p_t\le n$ و $۱\le q_۱,q_۲,\cdots,q_t\le m$ که $۱\le t\le r$، اعداد صحیح مثبت باشند به‌طوری‌که برای تمام $۱\le k\le t$ داشته باشیم $a_{p_k,q_k}=۰$ و همچنین برای هر $۱\le i,j\le t$ و $i\ne j$ رابطه $[p_i,q_i]\ne[p_j,q_j]$ برقرار باشد. مجموعه $\mathbf{z}=[[p_۱,q_۱],[p_۲,q_۲],\cdots,[p_t,q_t]]$ را در نظر بگیرید و فرض کنید ماتریس $n\times m$ دودویی $\mathbf{B}_\mathbf{z}=(b_{i,j})$ به‌صورت زیر تعریف‌شده باشد ($۱\le i\le n$ و $۱\le j\le m$):

$$b_{i,j}=\begin{cases}1 & [i,j]\in \mathbf{z},\\ a_{i,j} & [i,j]\notin \mathbf{z}.\end{cases} \qquad (۴)$$

اگر $\mathrm{CP}(\mathbf{B_z})+\mathrm{nops}(\mathbf{z})<\mathrm{CP}(\mathbf{A})$ آنگاه $\mathbf{z}$ یک لیست Paar مربوط به ماتریس $\mathbf{A}$ نامیده می‌شود.

مثال ۱: فرض کنید میدان متناهی $\mathbb{F}_{2^8}$ توسط چندجمله‌ای اولیه $f(x)=x^8+x+1$ به‌دست‌آمده باشد. فرض کنید α یک ریشه از $f(x)$ باشد. در [۱۲] بررسی‌شده که ماتریس 8×8 که در ادامه آمده، یک ماتریس MDS بر روی $\mathbb{F}_{2^8}$ است.

$$\mathbf{T}=\begin{pmatrix}
\alpha & \mathbf{1} & \alpha^4 & \mathbf{1} & \alpha^5 & \alpha^{14} & \alpha^7 & \alpha^8\\
\alpha^3 & \alpha & \mathbf{1} & \alpha^4 & \mathbf{1} & \alpha^5 & \alpha^{14} & \alpha^7\\
\alpha^6 & \alpha^3 & \alpha & \mathbf{1} & \alpha^4 & \mathbf{1} & \alpha^5 & \alpha^{14}\\
\alpha^{14} & \alpha^6 & \alpha^3 & \alpha & \mathbf{1} & \alpha^4 & \mathbf{1} & \alpha^5\\
\alpha^{14} & \alpha^{14} & \alpha^6 & \alpha^3 & \alpha & \mathbf{1} & \alpha^4 & \mathbf{1}\\
\alpha^8 & \alpha^{14} & \alpha^{14} & \alpha^6 & \alpha^3 & \alpha & \mathbf{1} & \alpha^4\\
\alpha^6 & \alpha^8 & \alpha^{14} & \alpha^{14} & \alpha^6 & \alpha^3 & \alpha & \mathbf{1}\\
\alpha^3 & \alpha^6 & \alpha^8 & \alpha^{14} & \alpha^{14} & \alpha^6 & \alpha^3 & \alpha
\end{pmatrix}.$$

شکل دودویی ماتریس $\mathbf{T}$ یک ماتریس دودویی 32×32 بوده که با $\mathbf{A}$ نشان داده‌ایم و در ضمیمه الف آورده شده است. در جدول ۱ برخی از لیست‌های Paar وابسته به ماتریس $\mathbf{A}$ به‌دست‌آمده است:

جدول (۱): لیست‌های Paar مرتبط با ماتریس $\mathbf{A}$

$\mathbf{z}$	$\mathrm{CP}(\mathbf{B_z})$	$\mathrm{CP}(\mathbf{B_z})+\mathrm{nops}(\mathbf{z})$	$\mathrm{CP}(\mathbf{A})$
$\mathrm{nops}(\mathbf{z})=۱$			
[[۴,۲۴]]	۲۰۳	۲۰۴	۲۰۵
[[۳۲,۳۲]]	۲۰۲	۲۰۳	۲۰۵
$\mathrm{nops}(\mathbf{z})=۲$			
[[۱۸,۱۸],[۱۸,۸]]	۲۰۱	۲۰۳	۲۰۵
[[۱۶,۴],[۴,۲۴]]	۲۰۰	۲۰۲	۲۰۵
$\mathrm{nops}(\mathbf{z})=۳$			
[[۱۶,۲۷],[۳۲,۳۲],[۱۹,۲۰]]	۱۹۹	۲۰۲	۲۰۵
[[۱۸,۲۷],[۲,۱۱],[۱۶,۴]]	۱۹۸	۲۰۱	۲۰۵
$\mathrm{nops}(\mathbf{z})=۴$			
[[۱۸,۲۷],[۱۶,۴],[۵,۱۱],[۴,۲۴]]	۱۹۷	۲۰۱	۲۰۵
[[۱۶,۲۷],[۲,۱۱],[۳۲,۳۲],[۱۶,۴]]	۱۹۶	۲۰۰	۲۰۵
$\mathrm{nops}(\mathbf{z})=۵$			
[[۱۶,۲۷],[۲,۱۱],[۱۶,۴],[۵,۱۱],[۴,۲۴]]	۱۹۵	۲۰۰	۲۰۵
[[۱۸,۲۷],[۱۶,۲۷],[۲,۱۱],[۱۶,۴],[۱۴,۲]]	۱۹۴	۱۹۹	۲۰۵

تعریف ۲: فرض کنید $\mathbf{A}$ یک ماتریس دودویی باشد. فرض کنید $\mathbf{z}$ یک لیست Paar مرتبط با ماتریس $\mathbf{A}$ باشد. فرض کنید که ماتریس دودویی $\mathbf{B_z}$ در شرط (۴) صدق نماید. آنگاه $\mathbf{z}$ یک لیست Paar بهینه برای ماتریس دودویی $\mathbf{A}$ نامیده می‌شود اگر مقدار عبارت $\mathrm{CP}(\mathbf{B_z})+\mathrm{nops}(\mathbf{z})$ بین همه مقادیر ممکن برای لیست‌های Paar مرتبط با ماتریس $\mathbf{A}$، کمترین باشد.

بر اساس تعریف ۱، تعداد حالت‌های ساخت لیست‌های Paar برابر 2^r-1 است؛ بنابراین، به دست آوردن لیست Paar با کمترین مقدار، زمانی که r عددی بزرگی باشد، ازنظر محاسباتی سخت است. با توجه به این مشکل، الگوریتم زیر را برای به دست آوردن یک لیست Paar نزدیک به بهینه[۶] پیشنهاد می‌کنیم.

الگوریتم ۱: ساخت یک لیست Paar نزدیک به بهینه با استفاده از یک روش تکراری-تصادفی

ورودی: یک ماتریس دودویی $\mathbf{A}$ به‌طوری‌که تعداد درایه‌های برابر با صفر این ماتریس برابر با r باشد.	
خروجی: یک لیست Paar نزدیک به بهینه مرتبط با ماتریس دودویی $\mathbf{A}$.	
لیست $\mathbf{S}=[]$ را که یک لیست خالی است در نظر بگیرید.	۱
اگر حداقل یک لیست Paar مانند $\mathbf{z}$ مرتبط با ماتریس $\mathbf{A}$ وجود داشته باشد به‌طوری‌که $\text{nops}(\mathbf{z})=1$ برو به مرحله ۳ و در غیر این صورت لیست $\mathbf{S}$ را برگردان و الگوریتم خاتمه می‌یابد.	۲
به دست آوردن $[[p_1,q_1],[p_2,q_2],\cdots,[p_t,q_t]]$ که $t\le r$، $\mathbf{L}=[[p_i,q_i]]$ که $1\le i\le t$، شامل لیست‌های Paar با اندازه یک است. به‌عبارت‌دیگر عناصر $[[p_i,q_i]]$ که $1\le i\le t$، تمام لیست‌های از اندازه یک مرتبط با ماتریس دودویی $\mathbf{A}$ هستند.	۳
به دست آوردن $\mathbf{R}=[\mathbf{z}_1,\mathbf{z}_2,\cdots,\mathbf{z}_k]$، $k\le 2^t-t-1$، شامل همه لیست‌های Paar است که می‌تواند با استفاده از $\mathbf{L}$ ساخته شود به‌شرط آنکه برای $1\le i\le k$ رابطه $\text{nops}(\mathbf{z}_i)\ge 2$ برقرار باشد.	۴
اگر $\text{nops}(\mathbf{R})\ge 1$ برو به مرحله ۷ و در غیر این صورت برو به مرحله ۶.	۵
فرض کنید $\mathbf{z}_i=[[p_i,q_i]]$ برای $1\le i\le t$ است. فرض کنید $m=\mathbf{min}(\{\text{CP}(\mathbf{B}_{\mathbf{z}_i})\ \vert i\})$ برای $1\le i\le t$ است. حال یکی از $\mathbf{z}_i$-ها را به‌تصادف انتخاب کنید و آن را $\mathbf{z}$ بنامید به‌شرط آنکه رابطه $\text{CP}(\mathbf{B}_{\mathbf{z}})=m$ برقرار باشد. حال به مرحله ۱۰ بروید.	۶
محاسبه $m_1=\mathbf{min}(\{\text{CP}(\mathbf{B}_{\mathbf{z}_i})+\text{nops}(\mathbf{z}_i)\ \vert i\})$ و $m_2=\mathbf{min}(\{\text{nops}(\mathbf{z}_i)\ \vert i\})$ که $1\le i\le k$ است.	۷
به دست آوردن $\mathbf{T}=[\mathbf{z}_{j_1},\mathbf{z}_{j_2},\cdots,\mathbf{z}_{j_q}]$ که $q\le k$ عناصری از $\mathbf{R}$ باشند که در دو شرط زیر برای همه $1\le i\le q$ صدق کنند: شرط اول: $\text{CP}(\mathbf{B}_{\mathbf{z}_{j_i}})+\text{nops}(\mathbf{z}_{j_i})=m_1$ و شرط دوم: $\text{nops}(\mathbf{z}_{j_i})=m_2$ باشد.	۸
عنصری از $\mathbf{T}$ را به‌صورت تصادفی انتخاب نموده و آن را با $\mathbf{z}$ نشان دهید.	۹
درایه‌های $\mathbf{z}$ را به لیست $\mathbf{S}$ اضافه کن. سپس با استفاده از لیست $\mathbf{z}$ ماتریس $\mathbf{B}_z$ را تشکیل بده و در دور بعد بجای ماتریس $\mathbf{A}$ از ماتریس $\mathbf{B}_z$ استفاده کن. به‌عبارت‌دیگر در این مرحله ماتریس $\mathbf{A}$ به‌روز می‌شود. حال به مرحله ۲ برو.	۱۰

دو نکته مهم درباره الگوریتم ۱ وجود دارد. در مرحله ۸، برخی از لیست‌های Paar را انتخاب می‌کنیم که در شرط $\text{nops}(\mathbf{z}_{j_i})=m_2$ صدق کنند. این شرط به دلیل اجرای الگوریتم ۱ بر روی مثال‌های مختلف، انتخاب شد. درواقع، دلیل ریاضی و قابل‌اثباتی برای انتخاب این شرط وجود ندارد. مهم‌ترین مسئله در الگوریتم ۱، در مرحله ۹ رخ می‌دهد. به‌عبارت‌دیگر، در مرحله نهم هیچ معیاری نداریم که بر اساس آن معیار بتوانیم تصمیم بگیریم که کدام عناصر از $\mathbf{T}$ باید انتخاب شوند و این گره اساسی[۷] این الگوریتم است.

مثال ۲: در این مثال، ما الگوریتم ۱ را بر روی ماتریس دودویی $\mathbf{A}$ در ضمیمه الف اجرا می‌کنیم. در اولین اجرای الگوریتم، مقدار لیست $\mathbf{L}$ به‌صورت زیر می‌شود.

$$\mathbf{L}=[[۴,۲۴],[۵,۱۱],[۱۴,۲],[۱۶,۴],[۱۸,۸],[۱۸,۱۸],[۱۹,۲۰],[۳۲,۳۲]]$$

می‌توان بررسی نمود که اندازه لیست $\mathbf{R}$ یا همان $\text{nops}(\mathbf{R})$ برابر ۲۴۷ می‌شود (لطفاً برای جزئیات بیشتر به [۹] ارجاع شود). همچنین داریم $m_1=۲۰۱$ و $m_2=۳$؛ بنابراین داریم $\mathbf{T}=[[[۱۶,۴],[۱۴,۲]],[[۳۲,۳۲],[۱۶,۴],[۱۴,۲]],[[۴,۲۴],[۱۴,۲]]]$. ما به‌صورت تصادفی لیست $\mathbf{z}=[[۳۲,۳۲],[۱۶,۴],[۱۴,۲]]$ را انتخاب می‌کنیم و با توجه به این انتخاب، لیست $\mathbf{S}$ و ماتریس $\mathbf{A}$ به‌روز می‌شود. در دومین اجرای الگوریتم، لیست $\mathbf{L}$ به‌صورت $\mathbf{L}=[[۲,۱۱],[۱۶,۲۷],[۱۸,۲۷]]$ می‌شود؛ بنابراین لیست $\mathbf{R}$ شامل عناصر زیر می‌شود:

$$\mathbf{R}=[[[۱۶,۲۷],[۲,۱۱]],[[۱۸,۲۷],[۲,۱۱]],[[۱۸,۲۷],[۱۶,۲۷]],[[۱۸,۲۷],[۱۶,۲۷],[۲,۱۱]]].$$

می‌توان بررسی نمود که $m_1=۱۹۶$ و $m_2=۲$ که نتیجه می‌دهد $\mathbf{T}=[[۱۶,۲۷],[۲,۱۱]]$ و بنابراین $\mathbf{z}=[[۱۶,۲۷],[۲,۱۱]]$. از مرحله ۱۰ نتیجه می‌شود که $\mathbf{S}=[[۳۲,۳۲],[۱۴,۲],[۱۶,۲۷],[۲,۱۱]]$. در ادامه ماتریس $\mathbf{A}$ را به‌روز نموده و سومین دور از الگوریتم را اجرا می‌کنیم. می‌توان بررسی نمود که نمی‌توان یک لیست Paar مانند $\mathbf{z}$ را یافت به‌طوری‌که $\text{nops}(\mathbf{z})=1$ است؛

بنابراین الگوریتم ۱ خاتمه پیدا نموده و لیست $\mathbf{S}$ به‌عنوان خروجی الگوریتم به دست می‌آید. ماتریس دودویی $\mathbf{A_S}$ در ضمیمه ب آورده شده است. با توجه به لیست $\mathbf{S}$ داریم $\mathrm{CP}(\mathbf{A_S})+\mathrm{nops}(\mathbf{S})=۱۹۴+۵=۱۹۹$؛ بنابراین الگوریتم ۱، هزینه پیاده‌سازی ماتریس $\mathbf{A}$ در ضمیمه الف را از ۲۰۵ تا XOR به ۱۹۹ تا XOR کاهش داده است.

لازم به ذکر است که اگر ازنظر محاسباتی امکان محاسبه تمام حالت‌های ممکن در مرحله ۴ از الگوریتم ۱ نبود، آنگاه پیشنهاد می‌کنیم لیست‌های Paar با عناصر کم (برای مثال حداکثر ۵ عنصر) را در نظر بگیرید.

مثال ۳: فرض کنید میدان متناهی $\mathbb{F}_{۲^۸}$ با استفاده از چندجمله‌ای اولیه $f(x)=x^۸+x^۴+x^۳+x^۲+۱$ ساخته‌شده است. فرض کنید α یکی از ریشه‌های $f(x)$ است. ماتریس ۸×۸ زیر را در نظر بگیرید. در [۱۳] نشان داده‌شده که این ماتریس بر روی میدان متناهی $\mathbb{F}_{۲^۸}$ یک ماتریس MDS است.

$$\mathbf{H}=\begin{pmatrix}
\mathbf{1} & \alpha^{25} & \alpha^{2} & \alpha^{50} & \alpha^{26} & \alpha^{3} & \alpha^{238} & \alpha^{198}\\
\alpha^{25} & \mathbf{1} & \alpha^{50} & \alpha^{2} & \alpha^{3} & \alpha^{26} & \alpha^{198} & \alpha^{238}\\
\alpha^{2} & \alpha^{50} & \mathbf{1} & \alpha^{25} & \alpha^{238} & \alpha^{198} & \alpha^{26} & \alpha^{3}\\
\alpha^{50} & \alpha^{2} & \alpha^{25} & \mathbf{1} & \alpha^{198} & \alpha^{238} & \alpha^{3} & \alpha^{26}\\
\alpha^{26} & \alpha^{3} & \alpha^{238} & \alpha^{198} & \mathbf{1} & \alpha^{25} & \alpha^{2} & \alpha^{50}\\
\alpha^{3} & \alpha^{26} & \alpha^{198} & \alpha^{238} & \alpha^{25} & \mathbf{1} & \alpha^{50} & \alpha^{2}\\
\alpha^{238} & \alpha^{198} & \alpha^{26} & \alpha^{3} & \alpha^{2} & \alpha^{50} & \mathbf{1} & \alpha^{25}\\
\alpha^{198} & \alpha^{238} & \alpha^{3} & \alpha^{26} & \alpha^{50} & \alpha^{2} & \alpha^{25} & \mathbf{1}
\end{pmatrix}.$$

شکل دودویی ماتریس $\mathbf{H}$ یک ماتریس دودویی ۶۴×۶۴ بوده که ما آن را $\mathbf{A}$ می‌نامیم. می‌توان بررسی نمود که $\mathrm{CP}(\mathbf{A})=۴۸۸$ است. حال ما الگوریتم ۱ را بر روی ماتریس $\mathbf{A}$ اجرا می‌کنیم و در دور اول اجرای الگوریتم لیست زیر را به دست می‌آوریم.

$$\mathbf{L}=[[۶,۲۲],[۷,۳۶],[۸,۳۷],[۱۴,۳۰],[۱۵,۴۴],[۱۶,۴۵],[۲۲,۶],[۲۳,۳۷],[۲۴,۴۵],[۳۰,۱۴],[۳۱,۴۵],$$
$$[[۳۲,۳۷],[۳۷,۴۰],[۳۸,۱۲],[۴۵,۴۸],[۴۶,۶۲],[۵۳,۳۷],[۵۵,۱۲],[۶۱,۴۵],[۶۲,۴۶],[۶۳,۴],[۶۳,۲۲]].$$

داریم $\mathrm{nops}(\mathbf{L})=۲۲$ که نتیجه می‌دهد که برای به دست لیست $\mathbf{R}$ در مرحله چهارم باید $۲^{۲۲}-۲۳$ ماتریس دودویی را تشکیل دهیم و سپس هزینه پیاده‌سازی این ماتریس‌ها را با الگوریتم Paar به دست آوریم. واضح است که این فرایند بسیار زمان‌بر بوده زیرا اندازه ماتریس دودویی ۶۴×۶۴ است؛ بنابراین، تنها زیرمجموعه‌های حداکثر با ۵ عضو از $\mathbf{L}$ را برای به دست آوردن لیست‌های موردنظر انتخاب می‌کنیم. حال با توجه به محدودیت گفته‌شده و بعد از اجرای کامل الگوریتم ۱، لیست $\mathbf{S}$ زیر به دست می‌آید:

$$\mathbf{S}=[[۶,۲۲],[۲۲,۶],[۱۴,۳۰],[۳۰,۱۴]].$$

می‌توان بررسی نمود که $\mathrm{CP}(\mathbf{A_S})+\mathrm{nops}(\mathbf{S})=۴۷۷+۴=۴۸۱$ که این موضوع نشان می‌دهد با استفاده از الگوریتم ۱ توانسته‌ایم هزینه پیاده‌سازی ماتریس MDS داده‌شده را از ۴۸۸ به ۴۸۱ تا XOR کاهش بدهیم. جزئیات پیاده‌سازی این ماتریس MDS با استفاده از ۴۸۱ تا XOR در [۹] آمده است.

۲-۲- روش ماتریس پایه

فرض کنید $\mathbf{M}$ یک ماتریس MDS بر روی میدان متناهی $\mathbb{F}_{۲^n}$ باشد. فرض کنید این میدان متناهی با استفاده از چندجمله‌ای تحویل‌ناپذیر $f(x)$ بر $\mathbb{F}_۲$، ساخته‌شده است. در [۵] پیشنهادشده است که برای به دست آوردن فرم دودویی ماتریس $\mathbf{M}$ می‌توان از ماتریس مشارکتی استفاده نمود که چندجمله‌ای مشخصه این ماتریس بر روی $\mathbb{F}_۲$ برابر $f(x)$ است. در این قسمت نشان می‌دهیم، می‌توان بجای ماتریس مشارکت از ماتریس‌های دودویی دیگری برای کاهش بیشتر هزینه پیاده‌سازی ماتریس‌های MDS، استفاده نمود.

تعریف ۳: فرض کنید میدان متناهی $\mathbb{F}_{۲^n}$ با استفاده از چندجمله‌ای تحویل‌ناپذیر $f(x)$ از درجه n ساخته‌شده است. فرض کنید که ماتریس $\mathbf{M}$ یک ماتریس MDS بر روی $\mathbb{F}_{۲^n}$ باشد. فرض کنید $\mathbf{N}$ یک ماتریس $n\times n$ دودویی بوده به‌طوری‌که چندجمله‌ای مشخصه این ماتریس بر روی $\mathbb{F}_۲$ برابر با $f(x)$ باشد. فرض کنید شکل دودویی ماتریس $\mathbf{M}$، با استفاده از ماتریس دودویی $\mathbf{N}$ را با علامت

$\mathbf{A}^{\mathbf{N}}$ نشان بدهیم. حال اگر بین همه ماتریس‌های $n\times n$ دودویی که چندجمله‌ای مشخصه آن‌ها برابر با $f(x)$ باشد (که ماتریس $\mathbf{N}$ یکی از آن‌ها است) مقدار $\mathrm{CP}(\mathbf{A}^{\mathbf{N}})$ کمترین شود آنگاه ماتریس $\mathbf{N}$ را یک ماتریس بهینه-پایه برای ماتریس $\mathbf{M}$ گوییم.
واضح است که جستجو برای یک ماتریس پایه بهینه مشکل است زیرا تعداد این ماتریس‌ها با افزایش بعد ماتریس زیاد می‌شود. به همین دلیل در ادامه یک روش تصادفی جهت پیدا نمودن یک ماتریس پایه نزدیک به بهینه، پیشنهاد می‌دهیم.
فرض کنید ماتریس $\mathbf{M}$ یک ماتریس MDS بر روی میدان $\mathbb{F}_{۲^n}$ بوده و این میدان با استفاده از چندجمله‌ای تحویل‌ناپذیر $f(x)$ ساخته‌شده است. در ابتدا و به‌سادگی یک ماتریس مشارکت $n\times n$ مانند $\mathbf{C}$ تشکیل داده که چندجمله‌ای مشخصه آن بر روی $\mathbb{F}_۲$ برابر $f(x)$ باشد. حال ماتریس دودویی $\mathbf{A}^{\mathrm{C}}$ را تشکیل می‌دهیم. در ادامه هزینه پیاده‌سازی این ماتریس دودویی را با استفاده از الگوریتم Paar تشکیل داده و آن را $\mathrm{X}=\mathrm{CP}(\mathbf{A}^{\mathrm{C}})$ می‌نامیم. درواقع می‌خواهیم یک مقدار اولیه برای هزینه پیاده‌سازی ماتریس $\mathbf{M}$ داشته باشیم و در ادامه، تلاش کنیم این هزینه را با استفاده از جایگزین نمودن $\mathbf{C}$ با برخی ماتریس‌های دودویی، کاهش دهیم. در گام بعدی و به‌تصادف، ماتریسی $n\times n$ دودویی مانند $\mathbf{N}$ انتخاب نموده که در دو شرط زیر صدق کنند. شرط اول این است که چندجمله‌ای تحویل‌ناپذیر $\mathbf{N}$ بر روی $\mathbb{F}_۲$ برابر با $f(x)$ باشد و شرط دوم و مهم این هست که هزینه پیاده‌سازی ماتریس $\mathbf{M}$ با استفاده از $\mathbf{N}$ کمتر از هزینه پیاده‌سازی ماتریس $\mathbf{M}$ با بکار بردن $\mathbf{C}$ شود. به‌عبارت‌دیگر رابطه $\mathrm{CP}(\mathbf{A}^{\mathbf{N}})<\mathrm{X}$ برقرار باشد. حال اگر چنین ماتریس دودویی مانند $\mathbf{N}$ به دستاوردیم که در دو شرط بالا صدق کند آنگاه $\mathrm{X}=\mathrm{CP}(\mathbf{A}^{\mathbf{N}})$ قرار داده و دوباره این روش را برای به دست آوردن نتایج بهتر تکرار می‌کنیم.
همان‌طور که گفته شد این روش تکراری-تصادفی جواب بهینه لزوماً نمی‌دهد اما انتظار داریم بعد از چند مرتبه تکرار این روش، یک جواب نزدیک به بهینه به دست آوریم. یکی از نکات کلیدی که از این بخش نتیجه می‌شود این است که برای به دست آوردن یک پیاده‌سازی کم‌هزینه برای یک ماتریس MDS باید به دنبال یک نمایش دودویی مناسب ازنظر محاسباتی، برای آن ماتریس باشیم.

مثال ۴: فرض کنید میدان متناهی $\mathbb{F}_{۲^۸}$ با استفاده از چندجمله‌ای اولیه $f(x)=x^۸+x^۶+x^۳+x^۲+۱$ ساخته شد و α یکی از ریشه‌های $f(x)$ باشد. می‌توان بررسی نمود که ماتریس ۸×۸ زیر، یک ماتریس MDS بر روی میدان $\mathbb{F}_{۲^۸}$ است.

$$\mathbf{M}=\begin{pmatrix}
\mathbf{1} & \alpha^{23} & \alpha^{2} & \alpha^{46} & \alpha^{24} & \alpha^{3} & \alpha^{147} & \alpha^{83}\\
\alpha^{23} & \mathbf{1} & \alpha^{46} & \alpha^{2} & \alpha^{3} & \alpha^{24} & \alpha^{83} & \alpha^{147}\\
\alpha^{2} & \alpha^{46} & \mathbf{1} & \alpha^{23} & \alpha^{147} & \alpha^{83} & \alpha^{24} & \alpha^{3}\\
\alpha^{46} & \alpha^{2} & \alpha^{23} & \mathbf{1} & \alpha^{83} & \alpha^{147} & \alpha^{3} & \alpha^{24}\\
\alpha^{24} & \alpha^{3} & \alpha^{147} & \alpha^{83} & \mathbf{1} & \alpha^{23} & \alpha^{2} & \alpha^{46}\\
\alpha^{3} & \alpha^{24} & \alpha^{83} & \alpha^{147} & \alpha^{23} & \mathbf{1} & \alpha^{46} & \alpha^{2}\\
\alpha^{147} & \alpha^{83} & \alpha^{24} & \alpha^{3} & \alpha^{2} & \alpha^{46} & \mathbf{1} & \alpha^{23}\\
\alpha^{83} & \alpha^{147} & \alpha^{3} & \alpha^{24} & \alpha^{46} & \alpha^{2} & \alpha^{23} & \mathbf{1}
\end{pmatrix}.$$

حال ماتریس ۸×۸ دودویی $\mathbf{C}$ را در رابطه (۵) در نظر بگیرید. ماتریس $\mathbf{C}$ یک ماتریس مشارکت بوده که چندجمله‌ای مشخصه این ماتریس بر روی $\mathbb{F}_۲$ برابر با $f(x)$ است. همچنین می‌توان بررسی نمود که $\mathrm{CP}(\mathbf{A}^{\mathrm{C}})=۵۰۵$ است.

$$\mathbf{C}=\begin{pmatrix} ۰ & ۰ & ۰ & ۰ & ۰ & ۰ & ۰ & ۱ \\ ۱ & ۰ & ۰ & ۰ & ۰ & ۰ & ۰ & ۰ \\ ۰ & ۱ & ۰ & ۰ & ۰ & ۰ & ۰ & ۱ \\ ۰ & ۰ & ۱ & ۰ & ۰ & ۰ & ۰ & ۱ \\ ۰ & ۰ & ۰ & ۱ & ۰ & ۰ & ۰ & ۰ \\ ۰ & ۰ & ۰ & ۰ & ۱ & ۰ & ۰ & ۰ \\ ۰ & ۰ & ۰ & ۰ & ۰ & ۱ & ۰ & ۱ \\ ۰ & ۰ & ۰ & ۰ & ۰ & ۰ & ۱ & ۰ \end{pmatrix}. \quad (۵)$$

در ابتدا مقدار X را برابر ۵۰۵ قرار می‌دهیم. حال بر اساس روش پیشنهادشده در این بخش، ماتریس دودویی زیر را با استفاده از جستجوی تصادفی انتخاب می‌کنیم. منظور از جستجوی تصادفی این است که با استفاده از نرم‌افزار میپل یک ماتریس تصادفی از اندازه ۸ تشکیل داده و بررسی می‌کنیم که آیا در دو شرط گفته‌شده در این بخش، صدق می‌کنند.

$$\mathbf{N}=\begin{pmatrix} ۰ & ۰ & ۰ & ۰ & ۱ & ۰ & ۰ & ۰ \\ ۰ & ۱ & ۰ & ۰ & ۰ & ۰ & ۰ & ۱ \\ ۰ & ۱ & ۰ & ۰ & ۰ & ۰ & ۰ & ۰ \\ ۰ & ۰ & ۰ & ۰ & ۰ & ۱ & ۱ & ۰ \\ ۰ & ۰ & ۰ & ۰ & ۰ & ۰ & ۱ & ۰ \\ ۱ & ۰ & ۰ & ۰ & ۰ & ۱ & ۰ & ۰ \\ ۰ & ۰ & ۱ & ۰ & ۰ & ۰ & ۰ & ۰ \\ ۰ & ۰ & ۰ & ۱ & ۰ & ۰ & ۰ & ۰ \end{pmatrix}.$$

می‌توان بررسی نمود که چندجمله‌ای مشخصه ماتریس $\mathbf{N}$ بر روی $\mathbb{F}_۲$ برابر $f(x)$ بوده و $\mathrm{CP}(\mathbf{A^N})=۴۳۲$ است. همان‌طور که مشاهده می‌کنید با استفاده از روش پیشنهادی در این بخش (استفاده از ماتریس پایه $\mathbf{N}$)، هزینه پیاده‌سازی ماتریس $\mathbf{M}$ از ۵۰۵ به ۴۳۲ تا XOR کاهش پیدا نمود.

۲-۳- روش پیشنهادی پیاده‌سازی ماتریس‌های MDS با استفاده از الگوریتم ماتریس-لیست

در این بخش، با ترکیب روش‌های پیشنهادشده در دو بخش قبل، یک الگوریتم ابتکاری به نام الگوریتم ماتریس-لیست جهت کاهش هزینه پیاده‌سازی ماتریس‌های MDS، پیشنهاد می‌کنیم. الگوریتم ماتریس-لیست شامل دو مرحله است. مرحله اول، انتخاب یک ماتریس پایه مناسب برای دودویی نمودن ماتریس MDS موردنظر است که این کار با روش پیشنهادشده در بخش قبل انجام می‌شود؛ بنابراین خروجی مرحله اول الگوریتم ماتریس-لیست، یک ماتریس دودویی است. هدف در مرحله دوم الگوریتم ماتریس-لیست، تهیه یک لیست Paar برای ماتریس دودویی به‌دست‌آمده از مرحله اول بوده که این کار با استفاده از الگوریتم ۱ انجام می‌شود. خلاصه الگوریتم ماتریس-لیست در الگوریتم ۲ آورده شده است.

الگوریتم ۲: الگوریتم ماتریس-لیست

ورودی: یک ماتریس MDS به نام $\mathbf{M}$ بر روی یک میدان متناهی	
خروجی: یک پیاده‌سازی کم‌هزینه برای ماتریس $\mathbf{M}$	
۱	به دست آوردن یک ماتریس پایه نزدیک به بهینه به نام $\mathbf{N}$ برای ماتریس $\mathbf{M}$ و تشکیل ماتریس دودویی $\mathbf{A^N}$.
۲	محاسبه یک لیست Paar نزدیک به بهینه به نام $\mathbf{S}$ که وابسته به ماتریس دودویی $\mathbf{A^N}$ بوده و تشکیل ماتریس $\mathbf{A_S^N}$.
۳	اجرای الگوریتم Paar بر روی ماتریس دودویی $\mathbf{A_S^N}$ و استخراج روابط لازم برای پیاده‌سازی ماتریس $\mathbf{M}$.

توجه شود که الگوریتم ۲، یک الگوریتم ابتکاری بوده و برای به دست آوردن یک پیاده‌سازی مناسب از نظر هزینه محاسبه XOR، نیاز است که چندین مرتبه این الگوریتم بر روی ماتریس MDS موردنظر اجرا شود. در مثال بعدی، نحوه اجرای این الگوریتم را بر روی یک ماتریس MDS ۸×۸ شرح می‌دهیم.

مثال ۵: فرض کنید میدان متناهی $\mathbb{F}_{2^8}$ با استفاده از چندجمله‌ای اولیه $f(x)=x^8+x^7+x^6+x+1$ ساخته‌شده و α یک ریشه از $f(x)$ است. در [۱۴] بررسی‌شده که ماتریس داده‌شده در زیر یک ماتریس MDS بر روی $\mathbb{F}_{2^8}$ است.

$$\mathbf{M}=\begin{pmatrix} \mathbf{1} & \alpha & \alpha^{157} & \alpha^{253} & \alpha^{2} & \alpha^{155} & \alpha^{59} & \alpha^{254} \\ \alpha & \mathbf{1} & \alpha^{253} & \alpha^{157} & \alpha^{155} & \alpha^{2} & \alpha^{254} & \alpha^{59} \\ \alpha^{157} & \alpha^{253} & \mathbf{1} & \alpha & \alpha^{59} & \alpha^{254} & \alpha^{2} & \alpha^{155} \\ \alpha^{253} & \alpha^{157} & \alpha & \mathbf{1} & \alpha^{254} & \alpha^{59} & \alpha^{155} & \alpha^{2} \\ \alpha^{2} & \alpha^{155} & \alpha^{59} & \alpha^{254} & \mathbf{1} & \alpha & \alpha^{157} & \alpha^{253} \\ \alpha^{155} & \alpha^{2} & \alpha^{254} & \alpha^{59} & \alpha & \mathbf{1} & \alpha^{253} & \alpha^{157} \\ \alpha^{59} & \alpha^{254} & \alpha^{2} & \alpha^{155} & \alpha^{157} & \alpha^{253} & \mathbf{1} & \alpha \\ \alpha^{254} & \alpha^{59} & \alpha^{155} & \alpha^{2} & \alpha^{253} & \alpha^{157} & \alpha & \mathbf{1} \end{pmatrix}.$$

در ابتدا، نویسندگان [۵] توانستند ماتریس $\mathbf{M}$ را با استفاده از الگوریتم Paar، با هزینه ۴۳۰ تا XOR پیاده‌سازی نمایند. ما در این مثال قصد داریم تا ماتریس $\mathbf{M}$ را با هزینه ۴۰۸ تا XOR پیاده‌سازی نماییم. (به دلیل محدودیت صفحات مقاله، فقط یک صورت کلی از اجرای الگوریتم ماتریس-لیست بر روی $\mathbf{M}$ شرح داده‌شده و جزئیات پیاده‌سازی این مثال در [۹] آورده شده است).

هزینه ۴۳۰ تا XOR گزارش‌شده در [۵]، با استفاده از الگوریتم Paar و همچنین بکار بردن ماتریس مشارکت به‌عنوان ماتریس پایه، به‌دست‌آمده است؛ بنابراین بر اساس گام اول از الگوریتم ۲، ما نیاز به جستجوی برای یک ماتریس ۸×۸ دودویی مانند $\mathbf{N}$ داریم به‌طوری‌که در ابتدا چندجمله‌ای مشخصه آن بر روی $\mathbb{F}_2$ برابر $f(x)$ باشد و همچنین $\mathrm{CP}(\mathbf{A}^{\mathbf{N}})$ کمتر از ۴۳۰ شود. بهترین نتیجه‌ای که ما با استفاده از جستجوی تصادفی به دست آورده‌ایم ماتریس زیر است:

$$\mathbf{N}=\begin{pmatrix} ۰ & ۰ & ۰ & ۰ & ۰ & ۱ & ۱ & ۰ \\ ۰ & ۰ & ۰ & ۱ & ۰ & ۰ & ۰ & ۰ \\ ۱ & ۰ & ۰ & ۰ & ۰ & ۰ & ۰ & ۰ \\ ۰ & ۰ & ۰ & ۰ & ۱ & ۰ & ۰ & ۰ \\ ۰ & ۰ & ۱ & ۰ & ۰ & ۰ & ۰ & ۰ \\ ۰ & ۰ & ۰ & ۰ & ۰ & ۰ & ۱ & ۰ \\ ۰ & ۰ & ۰ & ۰ & ۰ & ۱ & ۱ & ۱ \\ ۰ & ۱ & ۰ & ۰ & ۰ & ۰ & ۰ & ۰ \end{pmatrix}.$$

حال با استفاده از ماتریس $\mathbf{N}$، ماتریس دودویی $\mathbf{A}^{\mathbf{N}}$ را تشکیل می‌دهیم. می‌توان بررسی نمود که $\mathrm{CP}(\mathbf{A}^{\mathbf{N}})=۴۱۲$ است. در ادامه با استفاده از الگوریتم ۱ بر روی $\mathbf{A}^{\mathbf{N}}$، لیست Paar زیر به دست می‌آید:

$$\mathbf{S}=[[۴۰,۸],[۴۸,۱۶],[۵۶,۲۴],[۶۴,۳۲]].$$

با استفاده از لیست Paar بالا، ماتریس دودویی $\mathbf{A}^{\mathbf{N}}$ را به‌روز نموده و آن را با $\mathbf{A}_{\mathbf{S}}^{\mathbf{N}}$ نشان می‌دهیم. در ادامه با استفاده از الگوریتم Paar هزینه محاسبه پیاده‌سازی $\mathbf{A}_{\mathbf{S}}^{\mathbf{N}}$ را به دست می‌آوریم که برابر با $\mathrm{CP}(\mathbf{A}_{\mathbf{S}}^{\mathbf{N}})=۴۰۴$ XOR می‌شود. درنتیجه ماتریس MDS این مثال را می‌توان با هزینه ۴۰۸ تا XOR پیاده‌سازی نمود زیرا $\mathrm{CP}(\mathbf{A}_{\mathbf{S}}^{\mathbf{N}})+\mathrm{nops}(\mathbf{S})=۴۰۴+۴=۴۰۸$ است. فرم دودویی ماتریس $\mathbf{M}$ با استفاده از ماتریس دودویی $\mathbf{N}$ (که با $\mathbf{A}_{\mathbf{S}}^{\mathbf{N}}$ نشان داده‌شده) در ضمیمه ج آورده شده است.

با توجه به تعاریف اول و دوم این مقاله، مسئله‌های باز این مقاله عبارت هستند از روش پیدا نمودن ماتریس پایه و لیست Paar که مسائل کلیدی در کاهش هزینه پیاده‌سازی ماتریس‌های MDS هستند. همچنین از دیگر مسئله‌های باز این مقاله می‌توان به بررسی نتایج این مقاله با سایر برنامه‌های SLP مانند BP اشاره نمود. در عمل و به علت کمبود حافظه ما نتوانستیم که نتایج این مقاله را با استفاده از الگوریتم BP به دست آوریم. درواقع بسیار علاقه‌مند هستیم که بدانیم هزینه پیاده‌سازی ماتریس معرفی‌شده در ضمیمه ج با استفاده از الگوریتم BP چند XOR است.

از دیگر مزیت‌های روش معرفی‌شده در این مقاله این است که روش به دست آوردن لیست‌های Paar محدودیت مربعی بودن ماتریس دودویی را ندارد و بنابراین برای ماتریس‌های غیر مربعی دودویی نیز به‌راحتی به‌کاربرده می‌شود. برای مثال در [۶] و با روش نسبتاً پیچیده‌ای یک روش ابتکاری برای پیاده‌سازی ماتریس‌های دودویی پیشنهاد داده و برای نشان دادن برتری روش معرفی‌شده در [۶] به سایر روش‌ها، یک ماتریس دودویی ۱۴×۷ مثال زده ($\mathbf{M}$) و نشان داده که با روش [۴] می‌توان $\mathbf{M}$ را با ۱۸ تا XOR پیاده‌سازی نمود. این در حالی است که با روش معرفی‌شده در قسمت اول این مقاله (روش لیستی Paar) می‌توان به‌سادگی $\mathbf{M}$ را با ۱۸ تا XOR پیاده‌سازی نمود.

از دیگر مسائل باز مهم و جدی این مقاله این است که ما در این مقاله برخی از صفرهای ماتریس دودویی را به‌منظور کاهش هزینه پیاده‌سازی، یک نموده‌ایم. از طریق برخی از مثال‌ها متوجه شدیم که صفر نمودن برخی از درایه‌های یک ماتریس دودویی تأثیری در کاهش هزینه پیاده‌سازی آن ماتریس دودویی ندارد. درواقع حدس می‌زنیم که اگر بخواهیم برعکس ایده این مقاله عمل کنیم نیاز داریم تا از الگوریتم SLP دیگری مانند BP استفاده نماییم. به‌صورت دقیق‌تر، حدس ما این است که با صفر نمودن برخی از درایه‌های یک ماتریس دودویی و استفاده از برنامه BP، امکان رسیدن به پیاده‌سازی‌های کم‌هزینه برای ماتریس‌های معروف رمزنگاری مانند ماتریس استفاده‌شده در لایه انتشار رمز قالبی AES [۱۰] و یا ماتریس استفاده‌شده در تابع چکیده ساز Kazad [۱۳] وجود دارد.

۳- نتیجه‌گیری

در ابتدا مفهوم لیست Paar که مرتبط با ماتریس‌های دودویی بود معرفی نمودیم. سپس با استفاده از روش معرفی‌شده، یک الگوریتم تکراری-تصادفی برای کاهش هزینه پیاده‌سازی ماتریس‌های دودویی ارائه شد. در ادامه نشان دادیم که پیاده‌سازی یک ماتریس MDS ارتباط مستقیمی با نحوه دودویی نمودن آن ماتریس MDS دارد. به همین منظور، مفهوم ماتریس پایه را برای دودویی نمودن ماتریس‌های MDS معرفی نموده و یک روش برای به دست آوردن ماتریس پایه ارائه دادیم. قسمت مهم این مقاله مربوط به ترکیب نمودن روش لیست Paar و ماتریس پایه بوده که از ترکیب این دو روش، الگوریتمی به دست آمد به نام الگوریتم ماتریس-لیست که یک الگوریتم ابتکاری برای کاهش هزینه پیاده‌سازی ماتریس‌های MDS است. یکی از مزیت‌های مهم نتایج ارائه‌شده در این مقاله، سادگی پیاده‌سازی و امکان استفاده از روش معرفی‌شده در این مقاله، با سایر برنامه‌های SLP مانند BP است.

ضمیمه الف

$$\mathbf{A}=\begin{pmatrix}
۰۰۰۱۱۰۰۰۱۰۰۱۱۰۰۰۰۰۱۱۱۱۰۰۱۱۰۱۱۰۱۰\\
۱۰۰۱۰۱۰۰۱۱۰۱۰۱۰۰۱۰۱۰۰۰۱۰۱۰۱۱۰۱۱۱\\
۰۱۰۰۰۰۱۰۰۱۱۰۰۰۱۰۱۱۰۱۰۰۰۱۰۱۰۱۱۰۱۱\\
۰۰۱۰۰۰۰۱۰۰۱۱۰۰۰۱۰۱۱۰۱۰۰۰۱۰۱۰۰۱۰۱\\
۰۱۰۰۰۰۰۱۱۰۰۰۱۰۰۱۱۰۰۰۰۰۱۱۱۱۰۰۱۱۰۱\\
۰۱۱۰۱۰۰۱۰۱۰۰۱۱۰۱۰۱۰۰۱۰۱۰۰۰۱۰۱۰۱۱\\
۰۰۱۱۰۱۰۰۰۰۱۰۰۱۱۰۰۰۱۰۱۱۰۱۰۰۰۱۰۱۰۱\\
۱۰۰۱۰۰۱۰۰۰۰۱۰۰۱۱۰۰۰۱۰۱۱۰۱۰۰۰۱۰۱۰\\
۰۱۱۰۰۱۰۰۰۰۰۱۱۰۰۰۱۰۰۱۱۰۰۰۰۰۱۱۱۱۰۰\\
۰۱۰۱۰۱۱۰۱۰۰۱۰۱۰۰۱۱۰۱۰۱۰۰۱۰۱۰۰۰۱۰\\
۱۰۱۰۰۰۱۱۰۱۰۰۰۰۱۰۰۱۱۰۰۰۱۰۱۱۰۱۰۰۰۱\\
۱۱۰۱۱۰۰۱۰۰۱۰۰۰۰۱۰۰۱۱۰۰۰۱۰۱۱۰۱۰۰۰\\
۱۱۰۰۰۱۱۰۰۱۰۰۰۰۰۱۱۰۰۰۱۰۰۱۱۰۰۰۰۰۱۱\\
۰۰۱۰۰۱۰۱۰۱۱۰۱۰۰۱۰۱۰۰۱۱۰۱۰۱۰۰۱۰۱۰\\
۰۰۰۱۱۰۱۰۰۰۱۱۰۱۰۰۰۰۱۰۰۱۱۰۰۰۱۰۱۱۰۱\\
۱۰۰۰۱۱۰۱۱۰۰۱۰۰۱۰۰۰۰۱۰۰۱۱۰۰۰۱۰۱۱۰\\
۱۱۰۰۱۱۰۰۰۱۱۰۰۱۰۰۰۰۰۱۱۰۰۰۱۰۰۱۱۰۰۰\\
۰۰۱۰۰۰۱۰۰۱۰۱۰۱۱۰۱۰۰۱۰۱۰۰۱۱۰۱۰۱۰۰\\
۰۰۰۱۰۰۰۱۱۰۱۰۰۰۱۱۰۱۰۰۰۰۱۰۰۱۱۰۰۰۱۰\\
۱۰۰۰۱۰۰۰۱۱۰۱۱۰۰۱۰۰۱۰۰۰۰۱۰۰۱۱۰۰۰۱\\
۱۰۱۰۱۱۰۰۱۱۰۰۰۱۱۰۰۱۰۰۰۰۰۱۱۰۰۰۱۰۰۱\\
۰۱۱۱۰۰۱۰۰۰۱۰۰۱۰۱۰۱۱۰۱۰۰۱۰۱۰۰۱۱۰۱\\
۱۰۱۱۰۰۰۱۰۰۰۱۱۰۱۰۰۰۱۱۰۱۰۰۰۰۱۰۰۱۱۰\\
۰۱۰۱۱۰۰۰۱۰۰۰۱۱۰۱۱۰۰۱۰۰۱۰۰۰۰۱۰۰۱۱\\
۰۱۱۰۱۰۱۰۱۱۰۰۱۱۰۰۰۱۱۰۰۱۰۰۰۰۰۱۱۰۰۰\\
۰۱۰۱۰۱۱۱۰۰۱۰۰۰۱۰۰۱۰۱۰۱۱۰۱۰۰۱۰۱۰۰\\
۱۰۱۰۱۰۱۱۰۰۰۱۰۰۰۱۱۰۱۰۰۰۱۱۰۱۰۰۰۰۱۰\\
۱۱۰۱۰۱۰۱۱۰۰۰۱۰۰۰۱۱۰۱۱۰۰۱۰۰۱۰۰۰۰۱\\
۰۱۰۰۰۱۱۰۱۰۱۰۱۱۰۰۱۱۰۰۰۱۱۰۰۱۰۰۰۰۰۱\\
۰۱۱۰۰۱۰۱۰۱۰۱۱۱۰۰۱۰۰۰۱۰۰۱۰۱۰۱۱۰۱۰۰۱\\
۰۰۱۱۱۰۱۰۱۰۱۱۰۰۰۱۰۰۰۱۱۰۱۰۰۰۱۱۰۱۰۰\\
۱۰۰۱۱۱۰۱۰۱۰۱۱۰۰۰۱۰۰۰۱۱۰۱۱۰۰۱۰۰۱۰
\end{pmatrix}.$$

ضمیمه ب

$$\mathbf{A_S} = \begin{pmatrix}
۰۰۰۱۱۰۰۰۱۰۰۱۱۰۰۰۰۱۱۱۱۰۰۱۱۰۱۱۰۱۰ \\
۱۰۰۱۰۱۰۰۱۱۱۱۰۱۰۰۱۰۱۰۰۰۱۰۱۰۱۱۰۱۱۱ \\
۰۱۰۰۰۰۱۰۰۱۱۰۰۰۱۰۱۱۰۱۰۰۰۱۰۱۰۱۱۰۱۱ \\
۰۰۱۰۰۰۰۱۰۰۱۱۰۰۰۱۰۱۱۰۱۰۰۰۱۰۱۰۰۱۰۱ \\
۰۱۰۰۰۰۰۱۱۰۰۰۱۰۰۱۱۰۰۰۰۱۱۱۱۰۰۱۱۰۱ \\
۰۱۱۰۱۰۰۱۰۱۰۰۱۱۰۱۰۱۰۰۱۰۱۰۰۰۱۰۱۰۱۱ \\
۰۰۱۱۰۱۰۰۰۰۱۰۰۱۱۰۰۰۱۰۱۱۰۱۰۰۰۱۰۱۰۱ \\
۱۰۰۱۰۰۱۰۰۰۰۰۱۰۰۱۱۰۰۰۱۰۱۱۰۱۰۰۰۱۰۱۰ \\
۰۱۱۰۰۱۰۰۰۰۰۱۱۰۰۰۱۰۰۱۱۰۰۰۰۱۱۱۱۰۰ \\
۰۱۰۱۰۱۱۰۱۰۰۱۰۱۰۰۱۱۰۱۰۱۰۰۱۰۱۰۰۰۱۰ \\
۱۰۱۰۰۰۱۱۰۱۰۰۰۰۱۰۰۱۱۰۰۰۱۰۱۱۰۱۰۰۰۱ \\
۱۱۰۱۱۰۰۱۰۰۱۰۰۰۰۱۰۰۱۱۰۰۰۱۰۱۱۰۱۰۰۰ \\
۱۱۰۰۰۱۱۰۰۱۰۰۰۰۰۱۱۰۰۰۱۰۰۱۱۰۰۰۰۰۱۱ \\
۰۱۱۰۰۱۰۱۰۱۱۰۱۰۰۱۰۱۰۰۱۱۰۱۰۱۰۰۱۰۱۰ \\
۰۰۰۱۱۰۱۰۰۰۱۱۰۱۰۰۰۰۱۰۰۱۱۰۰۰۱۰۱۱۰۱ \\
۱۰۰۱۱۱۰۱۱۰۰۱۰۰۱۰۰۰۰۱۰۰۱۱۰۰۱۱۰۱۱۰ \\
۱۱۰۰۱۱۰۰۰۱۱۰۰۱۰۰۰۰۰۱۱۰۰۰۱۰۰۱۱۰۰۰ \\
۰۰۱۰۰۰۱۰۰۱۰۱۰۱۱۰۱۰۰۱۰۱۰۰۱۱۰۱۰۱۰۰ \\
۰۰۰۱۰۰۰۱۱۰۱۰۰۰۱۱۰۱۰۰۰۰۱۰۰۱۱۰۰۰۱۰ \\
۱۰۰۰۱۰۰۰۱۱۰۱۱۰۰۱۰۰۱۰۰۰۰۱۰۰۱۱۰۰۰۱ \\
۱۰۱۰۱۱۰۰۱۱۰۰۰۱۱۰۰۱۰۰۰۰۰۱۱۰۰۰۱۰۰۱ \\
۰۱۱۱۰۰۱۰۰۰۱۰۰۱۰۱۰۱۱۰۱۰۰۱۰۱۰۰۱۱۰۱ \\
۱۰۱۱۰۰۰۱۰۰۰۱۱۰۱۰۰۰۱۱۰۱۰۰۰۰۱۰۰۱۱۰ \\
۰۱۰۱۱۰۰۰۱۰۰۰۱۱۰۱۱۰۰۱۰۰۱۰۰۰۰۱۰۰۱۱ \\
۰۱۱۰۱۰۱۰۱۱۰۰۱۱۰۰۰۱۱۰۰۱۰۰۰۰۰۱۱۰۰۰ \\
۰۱۰۱۰۱۱۱۰۰۱۰۰۰۱۰۰۱۰۱۰۱۱۰۱۰۰۱۰۱۰۰ \\
۱۰۱۰۱۰۱۱۰۰۰۱۰۰۰۱۱۰۱۰۰۰۱۱۰۱۰۰۰۰۱۰ \\
۱۱۰۱۰۱۰۱۱۰۰۰۱۰۰۰۱۱۰۱۱۰۰۱۰۰۱۰۰۰۰۱ \\
۰۱۰۰۰۱۱۰۱۰۱۰۱۱۰۰۱۱۰۰۰۱۱۰۰۱۰۰۰۰۰۱ \\
۰۱۱۰۰۱۰۱۰۱۱۱۰۰۱۰۰۰۱۰۰۱۰۱۰۱۱۰۱۰۰۱ \\
۰۰۱۱۱۰۱۰۱۰۱۱۰۰۰۱۰۰۰۱۱۰۱۰۰۰۱۱۰۱۰۰ \\
۱۰۰۱۱۱۰۱۰۱۰۱۱۰۰۰۱۰۰۰۱۱۰۱۱۰۰۱۰۰۱۱
\end{pmatrix}.$$

ضمیمه ج

به دلیل محدودیت نمایش ماتریس دودویی $\mathbf{A}_{\mathrm{S}}^{\mathrm{N}}$، در این صفحه، ۳۲ سطر ابتدایی ماتریس $\mathbf{A}_{\mathrm{S}}^{\mathrm{N}}$ نمایش داده‌شده است.

$$
\begin{pmatrix}
۱۰۰۰۰۰۰۰۰۰۰۰۰۱۱۰۱۰۰۰۰۱۱۰۰۰۰۰۱۰۰۰۰۰۰۰۰۱۰۱۰۰۱۰۱۰۰۰۱۰۰۰۰۱۰۱۰۰۱۰۰۰۰۰ \\
۰۱۰۰۰۰۰۰۰۰۰۱۰۰۰۰۰۱۰۱۰۰۰۰۱۰۰۰۰۰۱۰۰۰۰۰۰۰۱۰۰۰۱۰۰۰۰۰۱۱۰۱۰۰۱۰۰۰۰۰۰۰۰۰۰۰۱ \\
۰۰۱۰۰۰۰۰۱۰۰۰۰۰۰۰۱۰۱۰۰۰۰۰۰۰۰۱۰۰۰۰۰۰۰۰۰۰۱۱۰۰۰۰۱۱۰۰۰۰۰۱۰۰۱۱۰۰۰۰۰۱۰۰۰ \\
۰۰۰۱۰۰۰۰۰۰۰۰۱۰۰۰۰۰۰۱۱۰۰۰۰۰۰۰۰۰۰۱۰۰۱۰۰۰۰۰۰۱۰۰۰۰۰۱۰۰۱۱۰۰۰۰۰۱۰۰۰۰۰۰ \\
۰۰۰۰۱۰۰۰۰۰۱۰۰۰۰۰۰۰۱۰۱۰۰۰۰۱۰۰۰۰۰۰۱۰۰۰۰۰۰۰۰۱۰۱۰۰۰۰۱۰۰۰۱۰۰۰۰۰۰۱۰۰۰۰ \\
۰۰۰۰۰۱۰۰۰۰۰۰۰۰۱۰۰۰۰۰۰۱۱۰۱۰۱۰۰۱۰۰۰۰۰۰۰۰۱۱۱۰۰۱۰۰۰۰۰۰۰۰۰۰۰۰۱۱۱۰۰۰۰۱۰۰ \\
۰۰۰۰۰۰۱۰۰۰۰۰۰۱۱۱۰۰۰۰۰۱۰۰۱۱۰۰۰۱۰۰۰۱۰۰۰۱۰۱۱۰۰۰۰۰۰۰۰۰۱۰۰۰۱۱۱۰۰۰۰۰۱۰۰ \\
۰۰۰۰۰۰۰۱۰۱۰۰۰۰۰۰۰۱۰۰۰۰۰۱۰۰۱۰۰۱۰۰۰۰۰۱۰۰۰۰۱۰۱۰۰۱۱۰۰۰۰۱۰۰۰۱۱۰۰۰۰۰۱۰ \\
۰۰۰۰۰۱۱۰۱۰۰۰۰۰۰۰۰۰۰۰۱۰۰۰۱۰۰۰۰۱۱۰۰۰۱۰۱۰۰۰۰۰۰۰۰۰۱۰۱۰۰۱۰۰۰۰۰۰۱۰۰۰۰۰۱۱ \\
۰۰۰۱۰۰۰۰۰۱۰۰۰۰۰۰۱۰۰۰۰۰۱۰۰۱۰۱۰۰۰۰۱۰۰۰۰۰۱۱۰۰۰۰۱۰۰۰۰۰۰۰۰۰۰۱۰۱۰۰۱۰۰۰ \\
۱۰۰۰۰۰۰۰۰۰۱۰۰۰۰۰۰۰۰۱۰۰۰۰۱۰۱۰۰۰۰۰۰۰۰۱۱۰۰۰۰۰۰۰۰۱۱۰۰۰۰۰۱۰۰۰۰۰۱۰۰۱۱۰ \\
۰۰۰۰۱۰۰۰۰۰۰۱۰۰۰۰۰۰۰۰۰۰۰۱۰۰۰۱۱۰۰۰۱۰۰۰۰۰۱۰۰۱۰۰۰۰۰۰۱۰۰۰۰۰۰۰۰۰۱۱۰۰۰۰ \\
۰۰۱۰۰۰۰۰۰۰۰۰۱۰۰۰۰۱۰۰۰۰۰۰۰۰۰۱۰۱۰۰۰۰۱۰۱۰۰۰۰۱۰۰۰۰۰۰۰۰۰۰۰۱۰۰۰۰۱۰۰۰۱۰۰۰ \\
۰۰۰۰۰۰۱۰۰۰۰۰۰۱۰۰۱۰۱۰۰۱۰۰۰۰۰۰۰۱۱۰۰۰۱۰۰۰۰۰۰۰۰۰۰۰۱۱۱۱۰۰۰۰۱۰۰۰۰۰۰۰۰۱۱ \\
۰۰۰۰۰۱۱۱۰۰۰۰۰۰۱۰۱۰۰۰۰۱۰۰۰۰۰۰۰۰۱۰۱۱۰۰۰۰۰۰۰۰۱۰۰۰۱۰۱۰۰۰۰۰۱۰۰۰۱۰۰۰۱۱۱ \\
۰۱۰۰۰۰۰۰۰۰۰۰۰۰۰۱۰۰۱۰۰۱۰۰۰۱۰۰۰۰۰۱۱۰۱۰۰۱۱۰۰۰۰۱۰۰۰۰۱۰۰۰۰۰۱۰۰۰۰۱۰۰۰۱ \\
۱۰۰۰۰۱۱۰۰۰۰۰۱۰۰۰۱۰۰۰۰۰۰۰۰۰۰۰۰۰۱۱۰۱۰۰۰۰۱۰۱۰۰۱۰۰۰۰۰۰۰۰۰۰۰۰۱۰۱۰۰۱۰۱۰۰۰ \\
۰۱۰۱۰۰۰۰۱۰۰۰۰۰۱۰۰۱۰۰۰۰۰۰۰۰۰۱۰۰۰۰۰۱۰۰۱۰۰۰۰۰۰۰۰۰۰۰۱۰۰۰۰۱۰۰۰۱۰۰۰۰۰۱۱ \\
۱۰۱۰۰۰۰۰۰۰۰۱۰۰۰۰۰۰۱۰۰۰۰۰۱۰۰۰۰۰۰۰۰۰۰۱۰۰۱۱۰۰۰۰۰۰۱۰۰۰۰۰۰۰۰۰۱۱۰۰۰۰۱۱۰۰۰ \\
۰۰۰۱۱۰۰۰۰۰۰۰۰۰۰۱۰۰۰۱۰۰۰۰۰۰۰۰۱۰۰۰۰۰۰۱۱۰۰۰۰۰۰۱۰۰۰۰۰۰۰۰۱۰۰۰۰۰۰۱۰۰۰۰۰۱ \\
۰۰۱۰۱۰۰۰۰۱۰۰۰۰۰۰۰۰۰۰۱۰۰۰۰۰۱۰۰۰۰۰۱۰۰۰۱۰۰۰۰۰۰۱۰۰۰۰۱۰۰۰۰۰۰۰۰۰۱۰۱۰۰۰۰ \\
۰۰۰۰۰۱۱۰۱۰۱۰۰۱۰۰۰۰۰۰۰۱۰۰۰۰۰۰۰۰۱۰۰۰۰۰۰۰۱۱۱۰۰۰۰۱۰۰۰۰۰۰۰۰۰۱۱۱۰۰۱۰۰۰۰۰ \\
۰۰۰۰۰۱۰۱۱۰۰۰۰۱۰۰۰۰۰۰۰۰۱۰۰۰۰۰۰۰۱۱۱۰۱۰۰۰۱۱۱۰۰۰۰۰۱۰۰۰۱۰۰۰۱۰۱۱۰۰۰۰۰۰۰ \\
۰۱۰۰۰۰۰۱۰۰۱۰۰۱۰۰۰۰۰۰۰۰۰۱۰۱۰۰۰۰۰۰۰۰۰۱۰۰۰۱۱۰۰۰۰۰۱۰۰۰۰۰۱۰۰۰۰۱۰۱۰۰۱۱۰ \\
۰۰۰۰۱۰۰۰۱۰۰۰۰۱۱۰۰۰۰۰۰۱۱۰۱۰۰۰۰۰۰۰۰۰۱۰۰۰۰۰۱۰۰۰۰۱۰۱۰۰۱۰۱۰۰۰۰۰۰۰۰۰۰۱۰۱ \\
۱۰۰۰۰۰۱۰۰۱۰۱۰۰۰۰۰۰۰۱۰۰۰۰۰۱۰۰۰۰۰۰۰۰۰۰۰۰۰۰۱۰۱۰۰۱۰۰۰۱۰۰۰۰۰۱۱۰۰۰۰۱۰۰۰ \\
۰۰۰۱۰۰۰۰۱۰۱۰۰۰۰۰۱۰۰۰۰۰۰۰۰۰۱۰۰۰۰۰۰۰۰۰۰۱۰۰۰۰۰۰۱۰۰۱۱۰۰۰۰۱۱۰۰۰۰۰۰۰۰۱۱۰ \\
۰۰۰۰۰۰۰۱۰۰۰۱۱۰۰۰۰۰۰۰۱۰۰۰۰۰۰۱۰۰۰۰۰۱۰۰۰۰۰۰۰۰۱۱۰۰۰۰۰۱۰۰۰۰۰۱۰۰۱۰۰۰۰۰ \\
۰۱۰۰۰۰۰۰۰۰۱۰۱۰۰۰۰۰۱۰۰۰۰۰۰۰۰۰۱۰۰۰۰۰۰۰۱۰۰۰۰۱۰۰۱۰۰۰۰۰۱۰۱۰۰۰۰۱۰۰۰۰۰۰۰ \\
۱۰۱۰۰۱۰۰۰۰۰۰۰۱۱۰۰۰۰۰۰۰۱۰۰۰۰۰۰۱۰۰۱۰۰۰۰۱۰۰۰۰۰۰۰۰۰۱۱۰۰۱۰۰۰۰۰۰۰۰۰۰۰۱۱۱ \\
۱۰۰۰۰۱۰۰۰۰۰۰۰۱۰۱۰۰۰۰۰۱۱۱۰۰۰۰۰۰۱۰۰۰۰۰۰۱۰۰۰۱۰۰۰۱۱۱۱۰۰۰۰۰۰۰۰۰۱۰۰۰۱۰۱ \\
۰۰۱۰۰۱۰۰۰۱۰۰۰۰۰۱۰۱۰۰۰۰۰۰۰۰۰۰۰۰۰۰۱۱۰۰۰۰۰۱۰۰۰۰۱۰۰۰۱۱۰۱۰۰۱۱۰۰۰۰۱۰۰۰۰
\end{pmatrix}
$$

ادامه ضمیمه ج

به دلیل محدودیت نمایش ماتریس دودویی $\mathbf{A_S^N}$، در این صفحه، ۳۲ سطر انتهایی ماتریس $\mathbf{A_S^N}$ نمایش داده‌شده است.

۰۰۰۰۰۱۰۱۰۰۱۰۱۰۰۰۱۰۰۰۰۱۰۱۰۰۱۰۰۰۰۰۱۰۰۰۰۰۰۰۰۰۰۰۰۱۱۰۱۰۰۰۰۱۱۰۰۰۰۰۱۰۰۰

۰۰۰۰۱۰۰۰۱۰۰۰۰۰۱۱۰۱۰۰۱۰۰۰۰۰۰۰۰۰۰۱۰۱۰۰۰۰۰۰۰۰۰۱۰۰۰۰۰۱۰۱۰۰۰۰۱۰۰۰۰۰۱۰

۰۰۰۰۰۱۱۰۰۰۰۱۱۰۰۰۰۰۱۰۰۱۱۰۰۰۰۰۱۰۰۰۰۰۱۰۰۰۰۰۱۰۰۰۰۰۰۰۱۰۱۰۰۰۰۰۰۰۰۱۰۰۰۰

۰۰۱۰۰۰۰۰۰۱۰۰۰۰۰۱۰۰۱۱۰۰۰۰۰۱۰۰۰۰۰۰۰۰۰۱۰۰۰۰۰۰۰۰۱۰۰۰۰۰۰۱۱۰۰۰۰۰۰۰۰۰۰۱

۱۰۰۰۰۰۰۰۰۱۰۱۰۰۰۰۱۰۰۰۱۰۰۰۰۰۰۱۰۰۰۰۰۰۰۰۱۰۰۰۰۰۱۰۰۰۰۰۰۰۱۰۱۰۰۰۰۱۰۰۰۰۰۰

۰۰۰۰۰۱۱۱۰۰۱۰۰۰۰۰۰۰۰۰۰۰۱۱۱۰۰۰۰۱۰۰۰۰۰۰۰۱۰۰۰۰۰۰۰۰۱۰۰۰۰۰۰۱۱۰۱۰۱۰۰۱۰۰

۰۱۰۰۰۱۰۱۱۰۰۰۰۰۰۰۰۱۰۰۰۱۱۱۰۰۰۰۰۱۰۰۰۰۰۰۰۰۱۰۰۰۰۰۰۱۱۱۰۰۰۰۰۱۰۱۱۰۰۰۰۱۰۰

۰۰۰۱۰۰۰۱۱۰۱۰۰۱۱۰۰۰۰۱۰۰۰۱۱۰۰۰۰۰۱۰۰۰۰۰۰۰۰۱۰۱۰۰۰۰۰۰۰۱۰۰۰۰۰۱۰۰۱۰۰۱۰۰

۰۰۱۰۱۰۰۰۰۰۰۰۰۱۰۱۰۰۱۰۰۰۰۰۱۰۰۰۰۱۰۱۰۰۰۰۰۱۱۰۱۰۰۰۰۰۰۰۰۰۰۰۱۰۰۰۱۰۰۰۰۱۱۰

۱۰۰۰۰۰۱۱۰۰۰۰۱۰۰۰۰۰۰۰۰۰۰۱۰۱۰۰۱۰۰۰۰۰۰۱۰۰۰۰۰۱۰۰۰۰۰۰۱۰۰۰۰۰۱۰۰۱۰۱۰۰۰۰

۰۰۰۱۱۰۰۰۰۰۰۰۰۱۱۰۰۰۰۰۱۰۰۰۰۰۱۰۰۱۱۰۱۰۰۰۰۰۰۰۰۰۱۰۰۰۰۰۰۰۰۱۰۰۰۰۱۰۱۰۰۰۰۰

۰۱۰۰۰۰۰۱۰۰۱۰۰۰۰۰۰۱۰۰۰۰۰۰۰۰۱۱۰۰۰۰۰۰۰۰۱۰۰۰۰۰۰۱۰۰۰۰۰۰۰۰۰۰۰۱۰۰۰۱۱۰۰۰

۰۱۰۱۰۰۰۰۱۰۰۰۰۰۰۰۰۰۰۱۰۰۰۰۱۰۰۰۱۰۰۰۰۰۱۰۰۰۰۰۰۰۰۰۱۰۰۰۰۱۰۰۰۰۰۰۰۰۱۰۱۰۰۰

۰۰۱۰۰۰۰۰۰۰۰۰۰۱۱۱۱۰۰۰۰۱۰۰۰۰۰۰۰۰۱۱۰۰۰۰۰۰۱۰۰۰۰۰۰۱۰۰۱۰۱۰۰۱۰۰۰۰۰۰۰۱۱۰

۱۰۰۰۰۰۰۰۰۱۰۰۰۱۰۱۰۰۰۰۰۱۰۰۰۱۰۰۰۱۱۱۰۰۰۰۰۱۱۱۰۰۰۰۰۰۱۰۱۰۰۰۰۱۰۰۰۰۰۰۰۱۰۱

۱۰۱۰۰۱۱۰۰۰۰۱۰۰۰۱۱۰۰۰۰۰۱۰۰۰۰۱۰۰۰۱۰۱۰۰۰۰۰۰۰۰۰۰۰۰۰۱۰۰۱۰۰۱۰۰۰۱۰۰۰۰۰۱

۱۰۰۰۰۱۰۱۰۰۱۰۰۰۰۰۰۰۰۰۰۱۰۱۰۰۱۰۱۰۰۰۱۰۰۰۰۱۱۰۰۰۰۰۱۰۰۰۱۰۰۰۰۰۰۰۰۰۰۰۰۱۱۰

۰۱۰۰۱۰۰۰۰۰۰۰۰۰۰۱۰۰۰۰۱۰۰۰۱۰۰۰۰۰۱۰۰۱۰۱۰۰۰۰۱۰۰۰۰۰۱۰۰۱۰۰۰۰۰۰۰۰۰۱۰۰۰۰

۰۰۱۰۰۱۱۰۰۰۰۰۱۰۰۰۰۰۰۰۰۱۱۰۰۰۰۱۱۰۰۰۱۰۱۰۰۰۰۰۰۰۰۱۰۰۰۰۰۰۱۰۰۰۰۰۱۰۰۰۰۰۰۰

۰۰۱۱۰۰۰۰۰۱۰۰۰۰۰۰۰۰۱۰۰۰۰۰۰۱۰۰۰۰۰۱۰۰۰۱۱۰۰۰۰۰۰۰۰۰۰۱۰۰۰۱۰۰۰۰۰۰۰۰۱۰۰۰

۱۰۰۰۱۰۰۰۰۰۰۱۰۰۰۰۱۰۰۰۰۰۰۰۰۱۰۱۰۰۰۰۰۰۱۰۱۰۰۰۰۱۰۰۰۰۰۰۰۰۰۰۱۰۰۰۰۰۱۰۰۰۰۰

۰۰۰۰۰۰۱۱۱۰۰۰۰۱۰۰۰۰۰۰۰۱۱۱۰۰۱۰۰۰۰۰۰۰۰۰۰۱۱۰۱۰۱۰۰۱۰۰۰۰۰۰۰۱۰۰۰۰۰۰۰۰۱۰

۰۱۰۰۰۱۱۱۰۰۰۰۰۱۰۰۰۱۰۰۰۱۰۱۱۰۰۰۰۰۰۰۰۰۰۰۰۱۰۱۱۰۰۰۰۱۰۰۰۰۰۰۰۰۱۰۰۰۰۰۰۱۱۱

۰۰۰۱۰۰۰۱۱۰۰۰۰۰۱۰۰۰۰۱۰۰۰۱۱۰۱۰۰۱۱۰۰۱۰۰۰۰۰۱۰۰۱۰۰۱۰۰۰۰۰۰۰۰۰۱۰۱۰۰۰۰۰۰

۰۰۱۰۰۰۰۰۱۰۰۰۰۱۰۱۰۰۱۰۱۰۰۰۰۰۰۰۰۱۰۱۰۰۰۰۱۰۰۰۱۰۰۰۰۱۱۰۰۰۰۰۰۱۱۰۱۰۰۰۰۰۰۰

۰۰۰۰۰۰۰۱۰۱۰۰۱۰۰۰۱۰۰۰۰۰۱۱۰۰۰۰۱۰۰۰۱۰۰۰۰۰۱۰۰۱۰۱۰۰۰۰۰۰۰۱۰۰۰۰۰۱۰۰۰۰۰۰

۰۰۰۰۱۰۰۰۰۰۱۰۰۱۱۰۰۰۰۱۱۰۰۰۰۰۰۰۰۱۱۰۰۰۰۱۰۰۰۰۱۰۱۰۰۰۰۰۱۰۰۰۰۰۰۰۰۰۱۰۰۰۰۰

۰۱۰۰۰۰۰۰۰۰۱۱۰۰۰۰۰۱۰۰۰۰۰۱۰۰۱۰۰۰۰۰۰۰۰۰۰۰۰۱۰۰۰۱۱۰۰۰۰۰۰۰۱۰۰۰۰۰۰۱۰۰۰۰

۰۰۰۱۰۰۰۰۱۰۰۰۱۰۰۰۰۱۰۱۰۰۰۰۱۰۰۰۰۰۰۰۰۱۰۰۰۰۰۰۰۰۱۰۱۰۰۰۰۰۱۰۰۰۰۰۰۰۰۰۱۰۰۰

۱۰۰۰۰۱۰۰۰۰۰۰۰۰۱۱۰۰۱۰۰۰۰۰۰۰۰۰۰۱۱۱۱۰۱۰۰۱۰۰۰۰۰۰۰۱۱۰۰۰۰۰۰۰۱۰۰۰۰۰۰۱۰۰

۰۰۰۰۰۱۰۰۰۱۰۰۰۱۱۱۱۰۰۰۰۰۰۰۰۱۰۰۰۱۰۱۱۰۰۰۰۱۰۰۰۰۰۰۰۱۰۱۰۰۰۰۰۱۱۱۰۰۰۰۰۰۱۰

۱۰۰۰۰۰۱۰۰۰۰۱۰۰۰۱۱۰۱۰۰۱۱۰۰۰۰۱۰۰۰۱۰۰۱۰۰۱۰۰۰۱۰۰۰۰۰۱۰۱۰۰۰۰۰۰۰۰۰۰۰۰۰۱

مراجع

[1] K.C. Gupta, S.K. Pandey, I.G. Ray and S. Samanta, "Cryptographically significant MDS matrices over finite fields: A brief survey and some generalized results," *Advances in Mathematics of Communications*, vol. 13, no. 4, pp. 779-843, 2019, Doi: 10.3934/amc.2019045.

[2] L. Kolsch, "XOR-Counts and Lightweight Multiplication with Fixed Elements in Binary Finite Fields," *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pp. 285–312, 2019, Doi: 10.1007/978-3-030-17653-2_10.

[3] C. Paar, "Optimized arithmetic for Reed-Solomon encoders," *Proceedings of IEEE International Symposium on Information Theory*, pp. 250-250, Jun. 1997, Doi: 10.1109/ISIT.1997.613165.

[4] J. Boyar, P. Matthews and R. Peralta, "Logic Minimization Techniques with Applications to Cryptology," *Journal of Cryptology*, vol. 26, no. 2, pp. 280-312, Apr. 2013, Doi: 10.1007/s00145-012-9124-7.

[5] S. Banik, Y. Funabiki and T. Isobe, "More Results on Shortest Linear Programs," *International Workshop on Security*, pp. 109-128, Jul. 2019, Doi: 10.1007/978-3-030-26834-3_7.

[6] Q.Q. Tan and T. Peyrin, "Improved Heuristics for Short Linear Programs," *IACR Transactions on Cryptographic Hardware and Embedded Systems*, vol. 2020, no. 1, pp. 203-230, Nov. 2019, Doi: 10.13154/tches.v2020.i1.203-230.

[7] F.J. MacWilliams and N.J.A. Sloane, "The Theory of Error Correcting Codes," NorthHolland, 1977.

[8] H. Kranz, G. Leander, K. Stoffelen and F. Wiemer, "Shorter Linear Straight-Line Programs for MDS Matrices," *IACR Transactions on Symmetric Cryptology*, vol. 2017, no. 4, pp. 188-211, Nov. 2017, Doi: 10.13154/tosc.v2017.i4.188-211.

[9] M. Mousavi, (2020), GitHub repository, https://github.com/mousavi-codes/Implementation-of-Binary-Matrices.

[10] J. Daemen and V. Rijmen, "The Design of Rijndael: AES-The Advanced Encryption Standard," *Springer-Verlag,* 2002, Doi: 10.1007/978-3-662-04722-4.

[11] Q. Liu, W. Wang, Y. Fan, L. Wu, L. Sun and M. Wang, "Towards Low-Latency Implementation of Linear Layers," *IACR Transactions on Symmetric Cryptology*, vol. 2022, pp. 158-182, Mar. 2022, Doi: 10.46586/tosc.v2022.i1.158-182.

[12] S. Sarkar, and H. Syed, "Analysis of Toeplitz MDS matrices," *Australasian Conference on Information Security and Privacy*, vol. 10343, pp. 3-18, May. 2017, Doi: 10.1007/978-3-319-59870-3_1.

[13] P. Barreto and V. Rijmen, "The Khazad Legacy-Level Block Cipher," *In Proceedings of the first open NESSIE Workshop, Belgium*, Nov. 2000.

[14] S.M. Sim, K. Khoo, F. Oggier and T. Peyrin, "Lightweight MDS Involution Matrices," *International Workshop on Fast Software Encryption,* vol. 9054, pp. 471-493, Mar. 2015, Doi: 10.1007/978-3-662-48116-5_23.

زیرنویس‌ها

[1] Maximum distance separable
[2] Shorter linear straight-line programs
[3] Heuristics
[4] Companion matrix
[5] Depth
[6] Near optimal
[7] Potential tie