روشی برای تولید ماتریسهای مولد کدهای بلوکی و LDPC کوانتومی

معین سروقد مقدم[۱]، منیره هوشمند[۲]، حسین آقابابا[۳]

[۱]کارشناسی ارشد برق الکترونیک، دانشگاه سمنان، moeinsarvaghad@yahoo.com

[۲]استادیار دانشکده مهندسی برق، دانشگاه بین‌المللی امام رضا (ع)، m.hooshmand@imamreza.ac.ir

[۳] استادیار دانشکده مهندسی برق، پردیس فارابی، دانشگاه تهران، aghababa@ut.ac.ir

چکیده

سیستمهای کوانتومی از لحاظ نظری دارای قدرت زیادی در زمینه پردازش و ارسال اطلاعات می‌باشند. اما بر هم کنش آنها با محیط بیرون، یک مانع بزرگ در مسیرتحقق عملی آنها است. کدهای پایدارساز بلوکی کوانتومی، برای غلبه بر مشکل مذکور طراحی شده‌اند. هر کد بلوکی کوانتومی با یک ماتریس دودویی با مشخصات خاص موسوم به ماتریس مولد توصیف می‌شود. اگر چه پژوهش‌های بسیاری در رابطه با ویژگی‌های کدهای تصحیح خطا با یک ماتریس مولد مفروض انجام گرفته است، اما تاکنون روشی برای پیدا کردن ماتریس‌های مولد در حالت کلی ارائه نشده است. در این مقاله روشی برای تولید ماتریس‌های مولد به کمک الگوریتم‌های ژنتیک ارائه شده است، سپس الگوریتم برای یافتن ماتریسهای مولد کدهای LDPC تغییر یافته است. نتایج شبیه‌سازی در نرم افزار MATLAB نشان از کارایی این الگوریتم می‌باشد که می تواند به عنوان یک روش کلی و سریع، نسبت به روش های ریاضیاتی، پیچیده و محدود گزارش شده مورد استفاده قرار گیرد.

کلیدواژه

کدهای پایدارساز بلوکی کوانتومی، ماتریس مولد، الگوریتم‌های ژنتیک، کدهای LDPC کوانتومی.

مقدمه

علی‌رغم قدرت نظری سیستم‌های کوانتومی، یک مانع بزرگ در مسیر تحقق عملی آن‌ها وجود دارد و آن برهم‌کنش سیستم کوانتومی با محیط بیرون است که منجر به تغییر ناخواسته اطلاعات می‌شود. نظریه تصحیح خطای کوانتومی برای غلبه بر مشکل مذکور پیشنهاد شد. در سال ۱۹۹۷، با کشف کدهای بلوکی پایدارساز کوانتومی[۱] [۱]، تحولی شگرف در زمینه نظریه تصحیح خطای کوانتومی رخ داد. کدهای پایدارساز [۲-۵] یک دسته مهم از کدهای اصلاح خطای کوانتومی[۲] (QECC) را تشکیل می دهند. برخی از این کدها عبارتند از: کد BCH[۳] [۶, ۷]، کد رید- سالومان[۴] [۸]، کد کانولوشنال[۵] [۹, ۱۰]، کد turbo [۱۱, ۱۲] و کد LDPC [۱۳-۱۶].

هر کد کوانتومی که k کیوبیت را به n کیوبیت کد کند، با یک ماتریس دودویی با ابعاد $(n-k)$ در ۲ توصیف می‌گردد. این ماتریس به ماتریس مولد[۶] [۱] موسوم است. این ماتریس، مطابق رابطه (۱) از دو زیر ماتریس X و Z به ابعاد $(n-k)$ در n تشکیل شده است.

$$G = [X|Z] \qquad (۱)$$

سطرهای ماتریس G مستقل خطی بوده و زیر ماتریس‌های X و Z قید رابطه (۲) را ارضا می‌کنند.

$$X.Z^T + Z.X^T = 0 \qquad (۲)$$

که Z^T ترانهاده[۷] ماتریس Z بوده و جمع به پیمانه دو انجام می‌شود. کدهایی که ماتریس مولد آنها پراکنده[۸] بوده(تعداد "۱" ها به نسبت ابعاد ماتریس بسیار کم باشد) به کدهای

[۱] Quantum Stabilizer Code
[۲] quantum error-correcting codes
[۳] Reed-Solomon
[۴] convolutional
[۵] Generator Matrix
[۶] Transposed
[۷] Sparse

تاریخ دریافت: ۱۳۹۶/۰۹/۱۳ تاریخ پذیرش: ۱۳۹۷/۱۱/۲۰

LDPC[۸] معروف هستند. کدهای LDPC کوانتومی، همانند همتای کلاسیک خود با پیچیدگی بسیار کمتر کدبرداری شده و در نرخی نزدیک به ظرفیت کانال[۹] کار می‌کنند[۱۶-۱۳].

تاکنون پژوهش‌های بسیاری برای بررسی مشخصه‌های تصحیح خطا و پیاده سازی عملی یک کد با ماتریس مولد مفروض انجام شده است، [۱۷, ۱۸]. اما تاکنون تنها در مقاله [۱۹] روشی برای پیدا کردن زیر مجموعه کوچکی از ماتریس‌های مولد در ساختارهای خاص ارائه شده است. در این مقاله یک ساختار ساده برای ایجاد یک کد پایدارساز با استفاده از یک ماتریس اختیاری اولیه، مطرح شده است. در این روش یک رابطه بین A_1 و A_2 از یک ماتریس بررسی دودویی[۱۰] $A=(A_1|A_2)$ وابسته به مولدهای پایدارساز از یک کد اصلاح خطای کوانتومی، تعریف می شود، سپس با استفاده از یک ماتریس باینری اختیاری، و توسط این رابطه، دو ماتریس A_1 و A_2 محاسبه می شوند. همچنین اخیرا در سال ۲۰۱۶ در [۲۰]، روشی برای تولید کدهای پایدارساز بلوکی کوانتومی با طول ۷ ارائه شده است.

در این مقاله روش کلی برای پیدا کردن و تولید ماتریس‌های مولد کدهای بلوکی کوانتومی به کمک الگوریتم‌های ژنتیک بیان می‌گردد. سپس با تغییر الگوریتم، روشی برای یافتن ماتریس‌های مولد کدهای LDPC کوانتومی ارائه می‌گردد. برای شبیه‌سازی روش مطرح شده از نرم افزار MATLAB استفاده می شود. نتایج شبیه‌سازی نشانگر کارایی الگوریتم‌های ژنتیک در یافتن ماتریس‌های مولد است.

ادامه مقاله به بخش‌های زیر تقسیم‌بندی می شود: در بخش ۲ توضیح مختصری درباره‌ی کدهای پایدارساز داده شده است، در بخش ۳ شرح مختصری از الگوریتم ژنتیک و نحوه‌ی عملکرد آن بیان شده است و سپس در بخش ۴ روش مطرح شده برای تولید ماتریس‌های مولد در پایدارسازهای بلوکی کوانتومی شرح داده می شود، سپس با تعمیم این روش، روشی برای تولید ماتریس های مولد در کدهای ldpc کوانتومی مطرح می شود. بخش ۵ مربوط به نتایج شبیه‌سازی و نتیجه‌گیری می‌باشد.

مفاهیم مقدماتی

در این بخش، ابتدا مفاهیم کیوبیت و گیت های کوانتومی شرح داده می شوند. سپس مفاهیم کدهای پایدار ساز و الگوریتم ژنتیک که در ادامه این مقاله مورد استفاده قرار می‌گیرند، توضیح داده می شوند.

کیوبیت و دروازه‌های کوانتومی

محاسبات کوانتومی بر اساس مفهومی معادل بیت در دنیای کلاسیک، پایه گذاری شده است که به آن بیت کوانتومی[۱۱] یا کیوبیت[۱۲] می‌گویند. در فضای هیلبرت دو بعدی، یک کیوبیت بصورت $|0\rangle = \begin{bmatrix}1\\0\end{bmatrix}$ و $|1\rangle = \begin{bmatrix}0\\1\end{bmatrix}$ نشان داده می‌شود، که بترتیب همتای کوانتومی بیت‌های کلاسیک ۰ و ۱ هستند. برخلاف بیت‌های کلاسیک، کیوبیت‌ها می توانند در هر برهم‌نهی مانند $\alpha|0\rangle + \beta|1\rangle$ قرار بگیرند، که α و β اعداد مختلطی هستند و $|\alpha|^2 + |\beta|^2 = 1$.

یک تحول (دروازه) کوانتومی با یک ماتریس یکانی[۱۳] U توصیف می‌شود. در تعریف کدهای پایدارساز از دروازه‌های پائولی[۱۴] استفاده می‌شود. دروازه‌های پائولی از چهار عملگر I، تبدیل همانی، X، دروازه چرخش بیت، Z، دروازه چرخش فاز و Y که ترکیبی از X و Z می‌باشد، تشکیل شده است. عملگرهای پائولی n کیوبیتی از ضرب تانسوری[۱۵] n دروازه پائولی تشکیل می‌شوند. رابطه (۳)، نمایش ماتریسی دروازه‌های پائولی را نشان می‌دهد.

$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \quad (۳)$$

عملگرهای پائولی که بر n کیوبیت اثر می‌کنند، از ضرب تانسوری n دروازه پائولی تشکیل می شوند.

کدهای پایدارساز

یک کد کوانتومی که k کیوبیت اطلاعات، $|\Phi\rangle$ را با کمک $(n-k)$ کیوبیت کمکی[۱۶] $|0\rangle$ به n کیوبیت فیزیکی[۱۷] کد می‌کند با نماد $[n,k]$ نمایش داده شده و به صورت فضای بردار ویژه ی[۱۸] مشترک $(n-k)$ عملگر پائولی n کیوبیتی $\{S_1, S_2, \ldots, S_{n-k}\}$ با مقدار ویژه‌ی ۱+ تعریف می‌گردد. این عملگرها بایستی مستقل[۱۹] و جابه‌جاپذیر[۲۰] باشند. منظور از استقلال عملگرها، به این معنی است که هیچ عملگری از حاصلضرب سایر عملگرها به دست نیاید. جابه‌جاپذیر بودن دو عملگر S_i و S_j به معنی برقرار رابطه زیر است:

$$S_i S_j = S_j S_i \quad (۴)$$

به عنوان مثال پایدارسازهای کد پنج کیوبیتی[۲۱] به صورت رابطه (۵) هستند:

$$\begin{aligned} s_1 &= XZZXI \\ s_2 &= IXZZX \\ s_3 &= XIXZZ \\ s_4 &= ZXIXZ \end{aligned} \quad (۵)$$

---

۸ Low Density Parity Check Matrix
۹ Channel Capacity
۱۰ binary check matrix
۱۱ Quatum Bit
۱۲ Qubit
۱۳ Unitary
۱۴ Pauli
۱۵ Tensor Product
۱۶ Ancilla
۱۷ Physical
۱۸ Eigen Value
۱۹ Independent
۲۰ Commutative

کار با پایدارسازها در تحلیل کدهای پایدارساز کوانتومی سخت و نامفهوم است. روش توصیف دیگر برای پایدارسازها، استفاده از ماتریس مولد دودویی G [۲۱] به صورت رابطه زیر است.

$$G = [X|Z] \quad (۶)$$

هرکدام از زیر ماتریس‌های X و Z در ماتریس G، سطرها متناظر با پایدارسازهای مختلف و ستون‌ها متناظر با کیوبیت‌های مختلف می‌باشند. نحوه تشکیل ماتریس دودویی نمایشگر کد از روی مجموعه پایدارساز به صورت زیر است:

- در هر کدام از پایدارسازها، در صورت وجود عملگر X یا Y، درایه‌ی متناظر با مکان آن عملگر در پایدارساز، در زیر ماتریس X یک قرار داده می‌شود. همچنین در صورت وجود عملگر Z یا Y درایه‌ی متناظر در زیر ماتریس Z نیز یک در نظر گرفته می‌شود. سایر درایه‌های ماتریس صفر می‌باشد. به عنوان مثال، ماتریس دودویی G برای کد پنج کیوبیتی مذکور را به صورت رابطه‌ی (۷) می‌توان نوشت:

$$\left[\begin{array}{ccccc|ccccc} ۱ & ۰ & ۰ & ۱ & ۰ & ۰ & ۱ & ۱ & ۰ & ۰ \\ ۰ & ۱ & ۰ & ۰ & ۱ & ۰ & ۰ & ۱ & ۱ & ۰ \\ ۱ & ۰ & ۱ & ۰ & ۰ & ۰ & ۰ & ۰ & ۱ & ۱ \\ ۰ & ۱ & ۰ & ۱ & ۰ & ۱ & ۰ & ۰ & ۰ & ۱ \end{array}\right] \quad (۷)$$

پایدارسازهای یک کد بلوکی کوانتومی جابه‌جاپذیر هستند. در ماتریس مولد دودویی نمایشگر کد وقتی شرط جابه‌جاپذیری برقرار است که قید رابطه (۱) برقرار باشد. همچنین شرط استقلال عملگرها، باعث می‌شود که ماتریس G، مرتبه کامل باشد.

الگوریتم ژنتیک

الگوریتم‌های ژنتیک [۲۲, ۲۳]، الگوریتم‌های جستجوی تصادفی هستند که فرایند تکامل طبیعی را شبیه‌سازی می‌کنند. این الگوریتم‌ها، تکامل را با جمعیتی از افراد (کروموزوم) شروع کرده و با استفاده از مکانیسم‌های ایجاد تنوع و انتخاب طبیعی، جمعیت را به سمت جواب بهینه سوق می‌دهند.

در این روش، جستجو به صورت تصادفی با جمعیتی[۲۱] از راه حل‌های اولیه آغاز می‌شود. اگر معیارهای نهایی ارضا نشوند، از سه عملگر متفاوت انتخاب[۲۲]، تقاطع[۲۳] و جهش[۲۴] برای بروز رسانی جمعیت استفاده می‌شود. هر بار که این سه عملگر تکرار شوند به آن یک نسل[۲۵] می‌گویند. راه حل ها در الگوریتم ژنتیک معمولا به صورت بیتی کدگذاری می‌شوند و هرکدام از سه عملگر فوق به صورت زیر عمل می‌کنند:

عملگر تقاطع بر اساس دو کروموزوم (والد) می تواند دو رشته را با هم ترکیب و از آنها کروموزوم جدید (فرزند) برای نسل بعدی به وجود آورد. در **عملگر جهش** نیز ممکن است بیتی، از یکی از رشته‌های جواب به صورت تصادفی انتخاب و سپس مقدار آن تغییر کند. در هر تکرار هر یک از رشته‌های موجود، رمزگشایی شده و مقدار تابع هدف[۲۶] برای آن به دست می‌آید. بر اساس مقادیر به دست آمده تابع هدف در جمعیت رشته‌ها، به هر رشته یک عدد برازندگی نسبت داده می‌شود. این عدد برازندگی احتمال انتخاب را برای هر رشته تعیین خواهدکرد. و براساس **عملگر انتخاب** ، تعدادی از جواب‌ها با توجه به مقدار برازندگی آنها برای نسل بعد انتخاب می شوند.

الگوریتم پیشنهادی

در این بخش، روش مطرح شده در این مقاله برای تولید ماتریس‌های مولد مورد استفاده در کدهای بلوکی کوانتومی شرح داده می شود.

هر کروموزوم یک ماتریس به صورت (۱) است که زیر ماتریس‌های X و Z هر کدام دارای $n-k=m$ سطر و n ستون می باشند. شکل (۱) روش کد کردن را نشان می‌دهد. در این روش تمامی درایه‌های ماتریس X و ماتریس Z به صورت پشت سرهم چیده می شوند، بنابراین طول هر کروموزوم $۲ \times n \times m$ می‌باشد.

$x_{۱۱}$	...	x_{nm}	$z_{۱۱}$	...	z_{nm}

شکل ۱. نحوه‌ی کدگذاری کروموزوم

برای پیدا کردن مقدار برازندگی مربوط به هر کروموزوم ابتدا ماتریس A، به شکل زیر تعریف می‌گردد:

$$A = X.Z^T + Z.X^T \quad (۸)$$

a_{ij}، درایه موجود در سطر i ام و ستون j ام ماتریس A، از رابطه زیر محاسبه می‌گردد:

$$a_{ij} = \sum_{k=۱}^{n} (x_{ik}z_{jk} + x_{jk}z_{ik}) \quad (۹)$$

که x_{ij} (z_{ij}) دلالت به درایه سطر i ام و ستون j ام در ماتریس X (Z) دارد.

همچنین برای برقراری قید (۲)، تابع برازندگی به صورت رابطه (۱۰) تعریف می گردد.

$$F_۱(C) = \sum_{i=۱}^{n}\sum_{j=۱}^{m}(a_{ij}) \quad (۱۰)$$

[۲۱] population
[۲۲] selection
[۲۳] crossover
[۲۴] mutation
[۲۵] generation
[۲۶] fitness function

یعنی تابع برازندگی (۱۰) زمانی بهترین جواب را می دهد که مقدار آن صفر شود، به عبارتی ماتریس A ماتریس تماماً صفر شود یعنی شرط (۲) برقرار شود. برای این که شرط دوم، یعنی مستقل خطی بودن هر سطر با سطر دیگر در ماتریس مولد برقرار باشد، ابتدا مرتبه ماتریس متناظر با هر کروموزوم ($Rank(C)$) محاسبه می شود. اگر این مقدار با عدد $n-k$، برابر نباشد، مقدار جریمه‌ای تعریف می‌شود که به مقدار تابع برازندگی اضافه می گردد. (متغیر $penalty$ در تابع برازندگی در رابطه (۱۱)). برای تعیین مقدار جریمه یک روش معمول این است که به صورت تقریبی بیشترین مقدار تابع برازندگی محاسبه شده و سپس پنج برابر آن را به عنوان جریمه تعیین کنیم. مقدار متغیر $penalty$، در این مقاله ۱۰۰۰ در نظر گرفته می شود.

$$F_2(C) = \sum_{i=1}^{n}\sum_{j=1}^{m}(a_{ij}) + penalty \qquad (11)$$

تابع برازندگی نهایی با احتساب شرط استقلال عملگرها در رابطه (۱۲) نشان داده شده است:

$$F(C) = \begin{cases} \sum_{i=1}^{n}\sum_{j=1}^{m}(a_{ij}) & if\ Rank(C) = n-k \\ \sum_{i=1}^{n}\sum_{j=1}^{m}(a_{ij}) + penalty & if\ Rank(C) \le n-k \end{cases} \qquad (12)$$

الگوریتم پیشنهادی برای کدهای LDPC کوانتومی

در این بخش تابع برازندگی را به گونه‌ای تغییر می دهیم که ماتریس‌های مولد کدهای LDPC کوانتومی را بیابد. از آنجا که ماتریس‌های مولد در این دسته از کدها پراکنده هستند، یعنی تعداد یک‌ها خیلی کمتر از صفرها هستند، شرطی را به صورت نسبت تعداد یک‌ها به صفرها ایجاد کردیم. (نسبت تعداد درایه‌های یک به کل درایه ها را ۰,۲ فرض کردیم) چنانچه شرط نسبت تعداد یک به صفر برقرار نبود، یک مقدار جریمه متغیر به صورت (۱۳) تعریف کردیم که مقدار آن وابسته به تعداد یک ها تغییر می کند.

$$y = (nonzero1/total) * ۱۰۰۰ + ۱۰۰۰ \qquad (13)$$

که در آن متغیرهای $nonzero1$ و $total$ به ترتیب، تعداد کل یک ها در ماتریس و کل درایه های ماتریس را نشان می دهند.

نتایج شبیه‌سازی

در این بخش الگوریتم پیشنهادی معرفی شده در بخش قبل، در نرم افزار MATLAB شبیه سازی شده و نتایج شبیه سازی برای ایجاد ماتریس‌های مولد کدهای بلوکی کوانتومی و کدهای LDPC کوانتومی به ترتیب در بخش‌های زیر ارائه می گردد. به عنوان نمونه، نتایج شبیه سازی برای ماتریس های با ابعاد ۵ در ۴۰ (کد با ۵ کیوبیت اطلاعات و ۲۰ کیوبیت فیزیکی)، ۱۰ در ۴۰ (کد با ۱۰ کیوبیت اطلاعات و ۲۰ کیوبیت فیزیکی) و ۱۰ در ۸۰ (کد با ۱۰ کیوبیت اطلاعات و ۴۰ کیوبیت فیزیکی) گزارش شده است. نتایج شبیه سازی نشان می دهند که برای سه حالت مفروض، الگوریتم با سرعت خوبی، (به طوریکه در نمودارهای مقدار برازندگی-نسل مربوط به هر ماتریس نشان داده شده است) ماتریس‌های مولد را پیدا می کند. در چندین نسل اول تابع برازندگی به مقدار صفر رسیده، و برای هر ماتریس با ابعاد در نظر گرفته شده، تعداد قابل توجهی ماتریس مولد را تولید می کند.

ماتریس‌های مولد کدهای بلوکی

در این بخش نتایج شبیه‌سازی الگوریتم برای تولید مولد کدهای بلوکی با ابعاد ۴۰ × ۵، ۴۰ × ۱۰ و ۸۰ × ۱۰ ارائه شده است. جدول۱، تنظیمات الگوریتم ژنتیک برای هر یک از سه حالت ماتریس فوق را نشان می دهد. نمودارهای مقدار برازندگی-نسل نیز در شکل۲ ارائه شده است. تعداد ماتریس‌های مولد به دست آمده برای ابعاد ۴۰ × ۵، ۴۰ × ۱۰، و ۸۰ × ۱۰، به ترتیب برابر ۲۱۷۳۵، ۲۰۹۶ و ۱۱۷۶ است. جدول۲، نمونه ای از ماتریس‌های مولد به دست آمده در ابعاد مختلف را نشان می دهد. در جدول۲، به علت کمبود فضای صفحه، معادل دهدهی عدد موجود در درایه هر زیر ماتریس آورده شده است.

جدول۱. تنظیمات الگوریتم ژنتیک برای یافتن ماتریس‌های مولد با ابعاد مختلف

اندازه ماتریس	۴۰ × ۵	۴۰ × ۱۰	۸۰ × ۱۰
نوع جمعیت	رشته بیت		
تعداد نسل‌ها	۵۰	۳۰۰	۵۰۰
تعداد جمعیت	۵۰	۳۰۰	۵۰۰

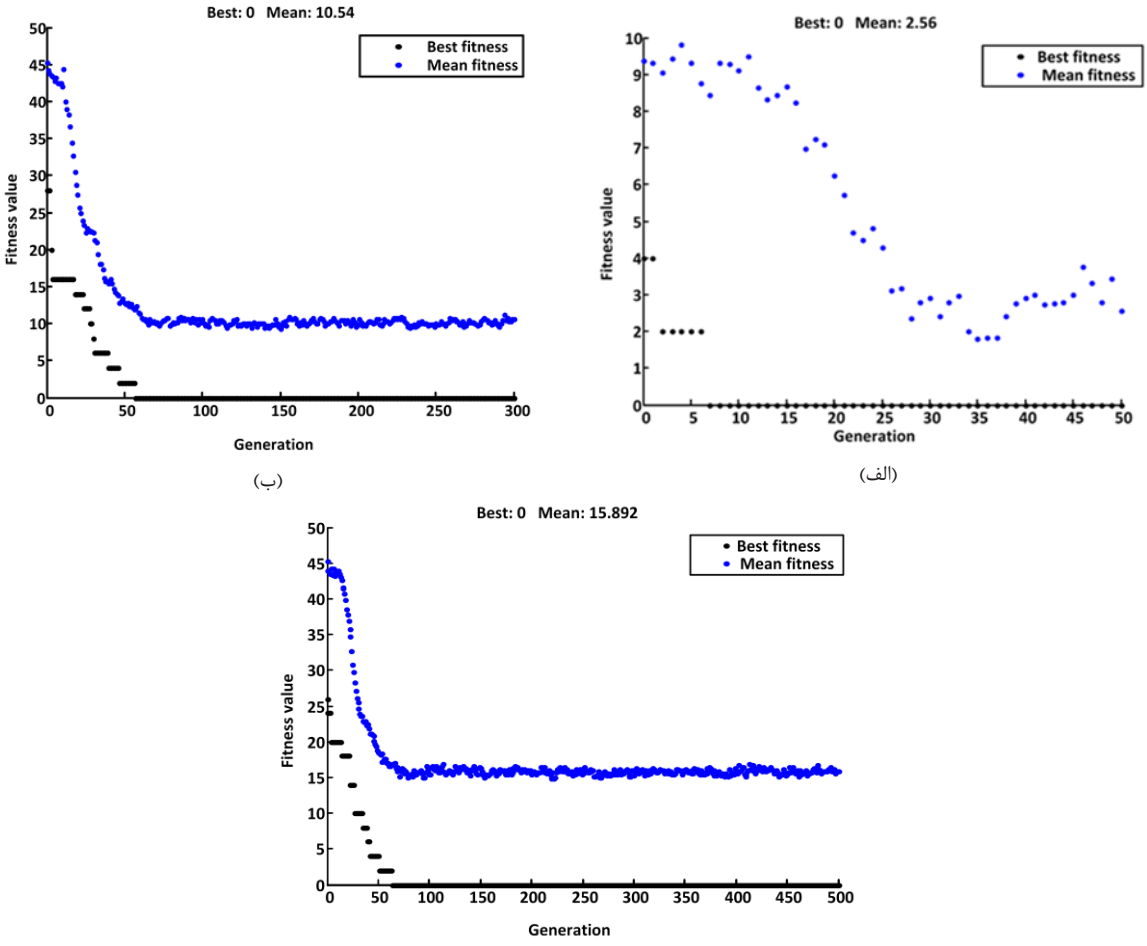
ماتریس‌های مولد کدهای LDPC کوانتومی

در این بخش، نتایج شبیه‌سازی برای تولید ماتریس‌های مولد کدهای LDPC با ابعاد ۴۰ × ۵، ۴۰ × ۱۰، و ۸۰ × ۱۰ ارائه می‌گردد. تعداد ماتریس‌های به دست آمده برای هر کدام از ابعاد به ترتیب برابر با ۴۸۴۰، ۳۴۱۱ و ۳۹۲۱ به دست آمده است. شکل۳، نمودارهای برازندگی برحسب نسل را نشان می‌دهد که گویای عملکرد مطلوب و سریع الگوریتم‌های ژنتیک در یافتن ماتریس‌های مولد کدهای LDPC است. جدول۳، نمونه‌ای از ماتریس‌های به دست آمده در هر کدام از ابعاد را

نشان می‌دهد. برای صرفه‌جویی در فضا، معادل دهدهی اعداد موجود در درایه‌های هر کدام از زیر ماتریس‌های X و Z آورده شده است. همچنین برای هر ماتریس تعداد درایه‌ها و تعداد یک‌های آن نیز گزارش شده است که نشان دهند پراکنده بودن ماتریس است. تنظیمات الگوریتم ژنتیک برای هرکدام از ابعاد مطابق جدول ۱ انجام شده است.

نتیجه گیری

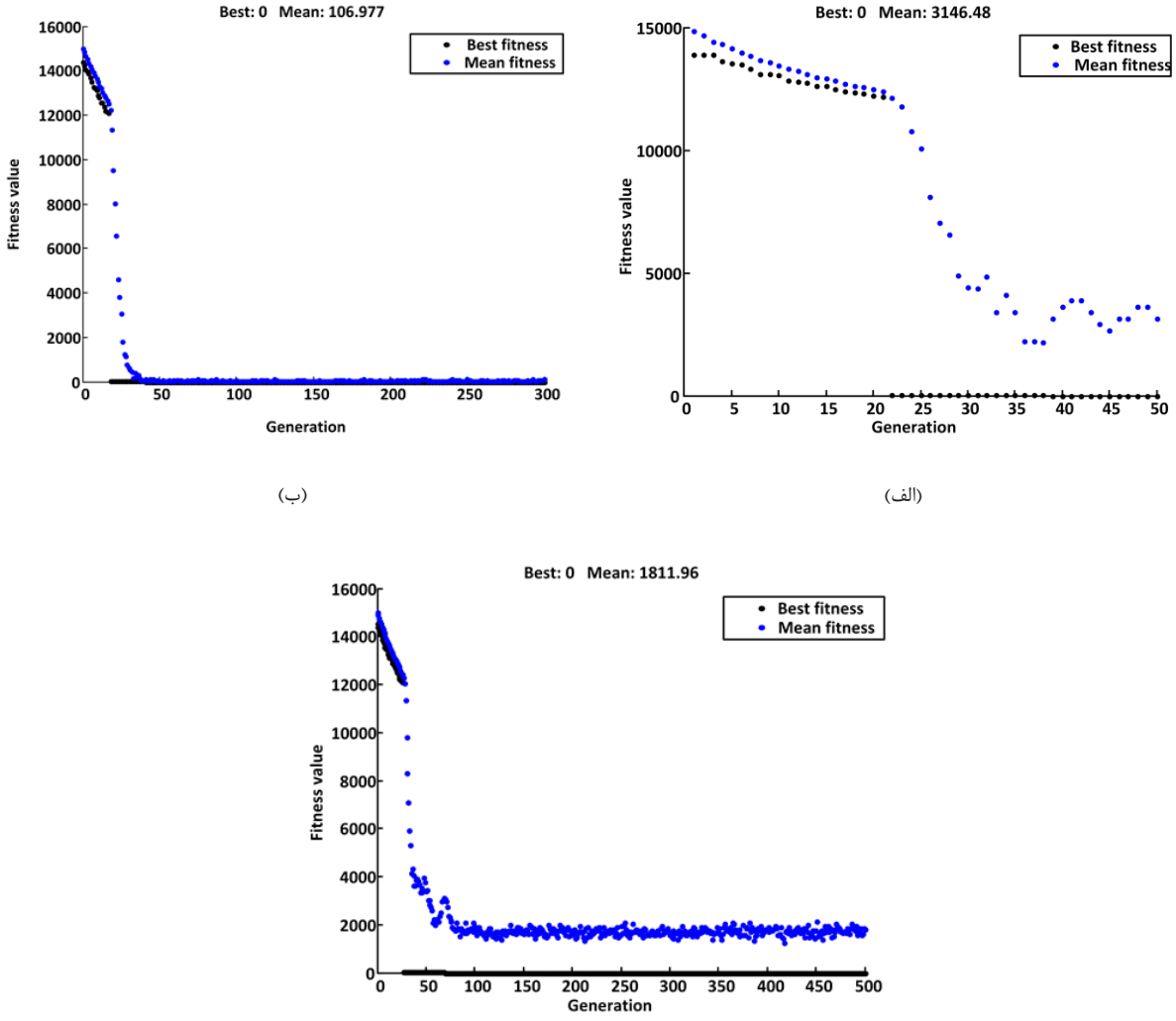
در این مقاله روشی برای پیدا کردن ماتریس های مولد کدهای پایدارساز بلوکی کوانتومی و کدهای LDPC به کمک الگوریتم‌های ژنتیک ارائه شده است. نتایج شبیه‌سازی، نشان دهنده کارایی این الگوریتم در یافتن این ماتریس‌ها می باشد. که این روش می تواند به صورت یک روش کلی و ساده، جایگزین روش های محدود و ریاضیاتی معرفی شده [۱۹, ۲۰] ، مورد استفاده قرار گیرد. به عنوان پژوهش آتی می توان از روشی برای تولید ماتریس‌های مولد سایر انواع کدهای کوانتومی، همانند کدهای پایدارساز کانولوشنال[۲۴, ۲۵] نام برد. کدهای کانولوشنال، کدهای حافظه‌دار هستند و ماتریس‌های مولد آن‌ها از توابع چند جمله‌ای تشکیل می‌شوند.



شکل۲. نمودار تابع برازندگی برحسب تعداد نسل برای ماتریس‌های مولد کدهای بلوکی کوانتومی با ابعاد الف) ۴۰ × ۵ ب) ۴۰ × ۱۰ ج) ۸۰ × ۱۰

جدول۲. نمونه خروجی ماتریس‌های مولد کدهای بلوکی از الگوریتم ژنتیک برای ابعاد مختلف

اندازه ماتریس	5×40	10×40	10×80
نمونه ماتریس	$\left(\begin{array}{c\|c} ۴۲۸۲۷۱ & ۶۱۶۶۹۱ \\ ۷۲۳۰۸۸ & ۲۶۸۱۸۰ \\ ۵۷۶۴۸۶ & ۸۹۸۳۱۵ \\ ۴۳۸۴۵ & ۲۹۷۸۸۴ \\ ۶۹۷۳۲۰ & ۴۲۲۳۹۹ \end{array}\right)$	$\left(\begin{array}{c\|c} ۱۰۴۲۷۸۳ & ۷۶۶۸۰۵ \\ ۵۹۳۵۶۵ & ۴۹۶۹۷۲ \\ ۵۱۲۰۷۹ & ۲۷۲۰۷ \\ ۷۱۷۹۲۶ & ۶۱۷۶۸۹ \\ ۹۷۳۴۸۵ & ۱۵۳۰۰۴ \\ ۱۷۳۱۵ & ۴۰۴۱۷۷ \\ ۵۴۹۳۲۸ & ۴۵۷۸۲۱ \\ ۴۷۴۸۰۰ & ۲۲۵۰۳۹ \\ ۳۰۹۵۰۵ & ۳۰۲۰۴۶ \\ ۸۸۴۶۰۸ & ۸۸۸۹۰۹ \end{array}\right)$	$\left(\begin{array}{c\|c} ۵۳۳۵۹۴۰۳۰۹۶۱ & ۷۰۳۵۷۷۸۷۸۰۱۲ \\ ۱۸۴۳۳۶۶۶۶۲۵۷ & ۵۳۱۹۳۹۳۸۰۱۹۱ \\ ۱۰۷۵۳۶۵۴۴۴۴۷۱ & ۴۴۱۳۲۹۴۷۳۴۲۱ \\ ۱۹۷۱۶۲۵۶۷۸۹۹ & ۱۳۶۷۶۸۲۰۵۶۰۶ \\ ۹۱۵۳۷۰۴۹۰۸۶ & ۷۵۲۷۴۰۶۷۴۸۸۱ \\ ۱۲۴۵۰۹۹۰۳۴۳۵ & ۴۹۳۲۱۹۰۱۵۹۴۶ \\ ۷۳۱۵۶۴۴۰۶۰۳۶ & ۲۰۶۶۴۷۰۵۵۵۰۸ \\ ۸۸۷۹۳۴۲۲۴۹۳۶ & ۴۸۴۷۴۷۵۴۵۴۰۹ \\ ۲۹۸۴۴۳۵۳۳۶۷۶ & ۱۳۳۴۶۵۰۳۹۷۹۶ \\ ۸۶۵۰۶۴۹۴۰۹۰۶ & ۸۰۹۰۷۱۶۶۰۷۸۳ \end{array}\right)$



(ب)

(الف)

(ج)

شکل۳. نمودار تابع برازندگی برحسب تعداد نسل برای ماتریس‌های مولد کدهای LDPC با ابعاد الف) ۴۰ × ۵ ب) ۴۰ × ۱۰ ج) ۸۰ ×× ۱۰

جدول ۳. نمونه خروجی ماتریس‌های مولد کدهای LDPC از الگوریتم ژنتیک برای ابعاد مختلف

نمونه ماتریس			
$\begin{pmatrix} ۶۵۵۴۳ \\ ۵۶۵۷۶۰ \\ ۱۳۹۲۶۴ \\ ۵۲۴۸۲۳ \\ ۱۶۵۱۲ \end{pmatrix} \begin{pmatrix} ۲۶۴۴۲۴ \\ ۲۰۴۸۰ \\ ۲۶۲۴۳۵ \\ ۲۶۳۱۸۴ \\ ۳۲۷۸۸ \end{pmatrix}$	$\begin{pmatrix} ۲ \\ ۲۹۵۹۷۶ \\ ۵۴۲۷۵۶ \\ ۴۶۸۴۸ \\ ۶۹۲۴۱۶ \\ ۱۵۱۵۸۴ \\ ۶۴۰۰ \\ ۱۸ \\ ۴۳۵۲ \\ ۵۲۵۳۲۸ \end{pmatrix} \begin{pmatrix} ۸ \\ ۵۵۵ \\ ۱۴۰۲۸۸ \\ ۴۳۸۸ \\ ۱۱۴۸۲۴ \\ ۸۵۲۱۶۰ \\ ۱ \\ ۱۵۳۶ \\ ۵۹۵۷۴۴ \\ ۹۲۶۴ \end{pmatrix}$	$\begin{pmatrix} ۵۹۴۰۶۴۴۴۹۶۱۶ \\ ۵۶۹۰۹۱۸۳۹۰۴۰ \\ ۴۳۱۰۱۱۹۲۴۴۹ \\ ۴۴۷۷۸۳۸۹۶۵۷۶ \\ ۱۳۹۱۵۲۳۲۷۲۴۴ \\ ۸۳۴۲۶۴۱۰۴۹۷۲ \\ ۴۲۹۷۰۶۴۶۴۰ \\ ۳۳۳۹۵۲۸۵۳۱۲۸ \\ ۸۲۵۱۹۵۹۰۵۵۵۶ \\ ۷۷۳۰۹۴۷۶۸۸۰ \end{pmatrix} \begin{pmatrix} ۵۵۰۰۹۷۶۵۹۹۰۴ \\ ۸۵۹۶۲۳۰۱۴۵ \\ ۱۳۹۶۰۵۰۵۳۵۰۴ \\ ۴۲۰۱۲۷۰۵ \\ ۸۷۸۲۸۵۲ \\ ۱۹۵۳۰۷۸۷۸۷۲ \\ ۵۵۴۱۱۷۹۲۲۸۵۳ \\ ۴۴۲۹۸۷۷۵۰۶ \\ ۲۴۴۳۳۹۰۰۷۰ \\ ۸۲۴۶۳۳۷۳۱۱۰۶ \end{pmatrix}$
تعداد درایه‌های ماتریس	۲۰۰	۴۰۰	۸۰۰
تعداد یک‌ها	۳۷	۷۴	۱۶۰

مراجع

[1] D. Gottesman, "Stabilizer codes and quantum error correction," PhD Thesis , California Institute of Technology, 1997.

[2] A. R. Calderbank, E. M. Rains, P. W. Shor, and N. J. Sloane, "Quantum error correction and orthogonal geometry," Physical Review Letters, vol. 78, p. 405, 1997.

[3] A. R. Calderbank, E. M. Rains, P. W. Shor, and N. J. Sloane, "Quantum error correction via codes over GF (4)," arXiv preprint quant-ph/9608006, 1996.

[4] D. Gottesman, "Class of quantum error-correcting codes saturating the quantum Hamming bound," Physical Review A, vol. ٥٤, p. ١٨٦٢, ١٩٩٦.

[5] D. Gottesman, "Stabilizer codes and quantum error correction," arXiv preprint quant-ph/9705052, 1997.

[6] S. A. Aly, A. Klappenecker, and P. K. Sarvepalli, "Primitive quantum BCH codes over finite fields," in 2006 IEEE International Symposium on Information Theory, pp. 1114-١١١, ٢٠٠٦.

[7] M. Grassl and T. Beth, "Quantum BCH codes," arXiv preprint quant-ph/9910060, 1999.

[8] M. Grassl, W. Geiselmann, and T. Beth, "Quantum Reed—Solomon Codes," in Applied Algebra, Algebraic Algorithms and Error-correcting Codes, ed: Springer, pp. 231-244, ١٩٩٩.

[9] H. Ollivier and J.-P. Tillich, "Description of a quantum convolutional code," Physical review letters, vol. 91, p. 177902, 2003.

[10] H. Ollivier and J.-P. Tillich, "Quantum convolutional codes: fundamentals," arXiv preprint quant-ph/0401134, 2004.

[11] D. Poulin, J.-P. Tillich, and H. Ollivier, "Quantum serial turbo codes,", IEEE Transactions on Information Theory, vol. 55, pp. 2776-2798, 2009.

[12] M. M. Wilde and M.-H. Hsieh, "Entanglement boosts quantum turbo codes," in 2011 IEEE International Symposium on Information Theory Proceedings (ISIT), pp. 445-449, ٢٠١١.

[13] T. Camara, H. Ollivier, and J.-P. Tillich, "A class of quantum LDPC codes: construction and performances under iterative decoding," in IEEE International Symposium on Information Theory, ISIT pp. 811-815, 2007.

[14] M. Hagiwara and H. Imai, "Quantum quasi-cyclic LDPC codes," arXiv preprint quant-ph/0701020, 2007.

[15] D. J. MacKay, G. Mitchison, and P. L. McFadden, "Sparse-graph codes for quantum error correction," IEEE Transactions on Information Theory, vol. 50, pp. 2315-2330, ٢٠٠٤.

[16] M. S. Postol, "A proposed quantum low density parity check code," arXiv:quant-ph/0108131, 2001.

[17] M. Houshmand, M. Sarvaghad, and A. Karimi-lenji, "A novel approach for encoding quantum block stabilizer codes," in 16th CSI International Symposium on Computer Architecture and Digital Systems (CADS), pp. ١٦٢-١٦٥, ٢٠١٢.

[۱۸] علی کریمی لنجی و منیره هوشمند، "کاهش تعدادگیت های مورد نیاز برای کدگذاری کدهای پایدارساز بلوکی کوانتومی،" ارائه شده در بیست و یکمین کنفرانس مهندسی برق ایران، دانشگاه فردوسی مشهد، ۱۳۹۲.

[19] Y. Hwang, B.-S. Choi, and M. Jeon, "On the construction of stabilizer codes with an

arbitrary binary matrix," Quantum information processing, vol. 12, pp. 467-479, 2013.
[20] D. M. Nguyen and S. Kim, "Construction and complement circuit of a quantum stabilizer code with length 7," in Eighth International Conference on Ubiquitous and Future Networks (ICUFN), pp. 332-336, 2016.
[21] M. A. Nielsen and I. L. Chuang, Quantum Computation and Quantum Information: Cambridge university press, 2001.
[22] D. E. Goldberg, Genetic Algorithms in Search, Optimization and Machine Learning: Addison-Wesley Longman Publishing Co., Inc., 1989.
[23] D. E. Goldberg, Genetic algorithms: Pearson Education India, 2006.
[24] M. Houshmand, S. Hosseini-Khayat, and M. M. Wilde, "Minimal-Memory Requirements for Pearl-Necklace Encoders of Quantum Convolutional Codes," IEEE Transactions on Computers, vol. 61, pp. 299-312, 2012.
[25] M. Houshmand, S. Hosseini-Khayat, and M. M. Wilde, "Minimal-memory, noncatastrophic, polynomial-depth quantum convolutional encoders," IEEE Transactions on Information Theory, vol. 59 ,pp. 1198-1210, 2013.