نشریه علمی «علوم و فناوری‌های پدافند نوین»
سال دهم، شماره ۴، زمستان ۱۳۹۸؛ ص ۳۸۱-۳۷۳

بازشناسی کور کدهای بلوکی در حضور نویز بالا با استفاده از روش‌های آماری

شیدا جهاندار[۱]، علی آقاگل‌زاده [۲]، سید جواد کاظمی‌تبار[۳*]

۱- کارشناس ارشد، ۲- استاد، ۳- استادیار، دانشگاه صنعتی نوشیروانی بابل

(دریافت: ۹۷/۰۴/۲۴، پذیرش: ۹۷/۰۸/۲۹)

چکیده

بازشناسی کور پارامترهای کدهای تصحیح خطای مستقیم از روی رشته بیت دریافتی در سمت گیرنده، در کاربردهای نظامی و تجاری بسیار مورد توجه قرار گرفته است. در واقع شناسایی طرح کدینگ استفاده‌شده در فرستنده بدون هیچ‌گونه اطلاعات قبلی، یک عمل چالشی است که توسط دشمن انجام می‌گیرد. ازآنجایی‌که برای شنود نیاز به کد برداری بیت‌های کدشده در مبدأ است، لازم است ابتدا مشخصات کد استفاده‌شده توسط دشمن شناسایی گردد. یکی از این مشخصات طول کد و نیز طول بیت‌های توازن مورد استفاده است. روش‌های مختلفی برای بازشناسی کور کدهای تصحیح خطای کانال ارائه گردیده است. در این مقاله سعی بر این است تا روشی آماری برای بازشناسی طول کلمه کد و طول بلوک اطلاعات ارائه شود که ضمن مقاومت نسبت به افزایش خطا، عملکرد آن با افزایش طول کد نیز کاهش نیابد. بدین منظور با استفاده از برخی الگوریتم‌های خوشه‌بندی ازجمله الگوریتم K-Means و الگوریتم Jenks Natural Breaks و درنهایت با ارائه یک الگوریتم ابتکاری به بررسی این موضوع برای کدهای بلوکی خطی باینری سامانمند پرداخته می‌شود. نتایج شبیه‌سازی در نرم‌افزار متلب نشان می‌دهند که روش‌های پیشنهادی در این مقاله علاوه‌بر پیچیدگی محاسباتی پایین و سرعت اجرای بالا، نتایج مطلوبی در شناسایی پارامترهای کدهای بلوکی سامانمند با طول‌های بلند و درصد بالایی از خطا را دارند.

کلیدواژه‌ها: بازشناسی کور، کدهای بلوکی سامانمند، واریانس، الگوریتم K-means، الگوریتم Jenks Natural Breaks

Blind Recognition of Block Code Parameters in the Presence of High SNR Using Statistical Techniques

S. Jahandar, A. Aghagolzadeh , J. Kazemitabar*
Babol Noshirvani University of Technology
(Received: 15/07/2018; Accepted: 20/11/2018)

Abstract

Blind recognition of error correction codes parameters from intercepted bit-stream at the receiver side, is highly considered in military and commercial applications. In fact, identification of the encoding scheme used in the transmitter without any prior information, is a challenging task to the adversary. Several methods have been presented for blind code recognition. In this paper, a statistical method for recognition of the length of the code word and the length of the block of information is presented. This scheme not only is resistant to error, but also its performance sustains in long codes. In this work, the method has been tested using some clustering algorithms such as K-Means and Jenks Natural Breaks. Then, a novel method to extract features of systematic binary linear block codes has been presented. Simulation results in MATLAB show that the proposed method, in addition to having low computational complexity and high performance rate, have an acceptable result in identifying systematic block codes with long lengths and even at high error levels.

Keywords: Blind Recognition, Systematic Block Codes, Variance, K-Means Algorithm, Jenks Natural Breaks Algorithm

*Corresponding Author E-mail: j.kazemitabar@nit.ac.ir

۱. مقدمه

در حالت کلی، مسئله بازشناسی کور بلوک کدگذار کانال به‌طوری‌که هیچ‌گونه اطلاعاتی از پارامترهای ارسال در دسترس نباشد مسئله‌ای مهم و دشوار است. در سامانه‌های شنود مخابراتی که بخشی اساسی از جنگ الکترونیک محسوب می‌شوند، هدف دشمن دسترسی به اطلاعاتی است که توسط بخش‌های مختلف فرستنده مبادله می‌شود. در این مقاله تمرکز ما بر روی بلوک کدگذار کانال است. تاکنون الگوریتم‌های متعددی برای حل این مسئله ارائه شده‌اند که برخی از آن‌ها برای بازشناسی و بازسازی کدهای کانولوشنال ارائه شده‌اند و برخی دیگر نیز کدهای بلوکی را مورد بررسی قراردادند [۲-۱]. نکته قابل‌توجه در همه مسائل مطرح‌شده آن است که در هریک از الگوریتم‌های ارائه‌شده، تصمیم‌گیری با توجه به معیاری مشخص صورت می‌گیرد. در ارتباط با بازشناسی کدهای بلوکی، والمبیوس [۳] مسئله تشخیص کد با درست‌نمایی بیشینه (MLCR[1]) را مطرح کرد و ثابت کرد که این مسئله از دسته مسائل NP[2] کامل است و با اجرای این روش توانست ماتریس بررسی توازن را به‌دست آورد. در تحقیقی دیگر الگوریتمی برای بازیابی طول کلمه کد ارائه شده که برای انواع کدهای بلوکی قابل‌استفاده است، این الگوریتم برای کدهای LDPC[3] عملکرد خوبی دارد، درحالی‌که برای کدهای تصادفی با طول بزرگ و سطح نویز بالا نتیجه نمی‌دهد [۴]. روش دیگری برای بازشناسی کور کدهای تصحیح خطا ارائه شده است که از خاصیت خطی بودن کدها استفاده می‌کند و با تشکیل ماتریسی از رشته بیت دریافتی و محاسبه مرتبه ماتریس به بازشناسی می‌پردازد [۵]. از این روش می‌توان برای بازشناسی کدهای کانولوشنال یا کدهای توربو نیز بهره برد البته باید به این نکته توجه کنیم که نویزپذیری این روش بسیار بالاست. در تحقیقی دیگر الگوریتمی تکراری برمبنای معادلات وزن‌دار بررسی توازن ارائه گردیده که نتیجه قابل قبولی برای کدهای LDPC داشته است [۶]. علاوه بر روش‌های فوق، روش‌های دیگری به بازشناسی کدهای چرخشی پرداخته‌اند، به این صورت که در یکی از این روش‌ها طول کلمه کد باید از قبل مشخص باشد تا بتوان چندجمله‌ای مولد را تخمین زد. روش ارائه‌شده در این مقاله ساده و سریع است و می‌تواند به‌طور گسترده برای هر کد چرخشی سامانمند یا غیر سامانمند، باینری یا غیرباینری به‌کار گرفته شود اما بزرگ‌ترین نقص این روش آن است که خطا را صفر در نظر گرفته است [۷]. الگوریتمی دیگر برای شناسایی کور کدهای ضربی BCH[4] با استفاده از دو روش بزرگ‌ترین شمارنده مشترک و مرتبه ماتریس معرفی شده است که در این روش طول کلمه کد، طول بلوک اطلاعات و چندجمله‌ای سازنده مشخص می‌شود [۸]. ناصری و همکاران [۹] الگوریتمی مبتنی بر تئوری آنتروپی طراحی و پیشنهاد کردند که می‌تواند انواع کدینگ بلوکی و کانولوشنال را بازشناسی کند. این روش برای طول کدهای بزرگ دارای پیچیدگی بسیار بالایی است و تنها برای کانال‌های با احتمال خطای پایین عمل بازشناسی را به‌درستی انجام می‌دهد. در سال‌های اخیر روش‌هایی برای بازشناسی کور کدهای بلوکی بر مبنای محاسبه احتمال پسین سندرم ارائه‌شده است، این روش برای کدهای LDPC که دارای ماتریس بررسی توازن بزرگی هستند بهینه است [۱۱-۱۰]. در تحقیقی دیگر الگوریتمی برای بازشناسی کور هم‌زمان کدهای تصحیح خطای مستقیم و پارامترهای اینترلیور ارائه شده است که اغلب این روش برای مقادیر پایین خطا کاربرد دارد [۱۲]. روشی دیگر برای بازشناسی کور کدهای بلوکی پیشنهاد شده است که برای کدهای با نرخ خطای بیت بالا و نرخ کد پایین عملکرد خوبی دارد اما به‌محض افزایش طول کد عملکرد این روش به‌سرعت کاهش می‌یابد [۱۳]. در تحقیقی دیگر روشی آماری برای بازشناسی کور پارامترهای کدهای بلوکی سامانمند بیان شده است که این روش تنها برای کدهایی با طول کم قادر به بازشناسی پارامترهای کد است [۱۴]. در این مقاله چند روش آماری ارائه می‌شود که پارامترهای کدهای بلوکی سامانمند با درصد خطای بالا و طول کد بلند را به درستی آشکارسازی می‌کنند. در ادامه مقاله در بخش دوم، به تعریف مسئله پرداخته می‌شود، سپس در بخش سوم روش‌های پیشنهادی ارائه می‌شوند و در بخش چهارم با توجه به نتایج شبیه‌سازی، روش‌های ارائه‌شده ارزیابی می‌شوند و در آخر در بخش پنجم به نتیجه‌گیری پرداخته می‌شود.

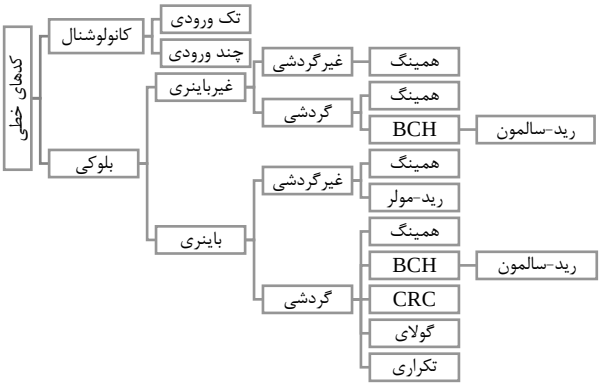
۲. تعریف مسئله

شکل (۱) دسته‌بندی کدهای تصحیح خطای مستقیم را نشان می‌دهد. یک کد بلوکی خطی C یک زیر فضای k بعدی از فضای n بعدی است که معمولاً کد (k و n) گفته می‌شود. برای کدهای باینری، یک کد بلوکی خطی مجموعه‌ای از ۲^{k} دنباله‌ی باینری با طول n است به‌طوری‌که برای هر دو کلمه کد $c_۱ \in C$ و $c_۲ \in C$ داریم $c_۱ + c_۲ \in C$.

در هر کد بلوکی خطی نگاشت از مجموعه ۲^{k} دنباله k بیتی اطلاعات به ۲^{k} کلمه کد n بیتی را می‌توان توسط یک ماتریس $k \times n$ به نام ماتریس مولد G نمایش داد.

$$c_m = u_m G \;;\quad ۱ \le m \le ۲^{k} \qquad (۱)$$

[1] Maximum Likelihood Code Recognition
[2] Nondeterministic Polynomial Time
[3] Low-Density Parity-Check
[4] Bose–Chaudhuri–Hocquenghem codes



شکل ۱. دسته‌بندی کدهای تصحیح خطای مستقیم [۱۵]

u_m یک بردار باینری با طول k بیانگر دنباله اطلاعات و c_m کلمه کد مربوطه است. مجموعه کلمه‌های کد C دقیقاً مجموعه ترکیب‌های خطی سطرهای G است. اگر ماتریس مولد G دارای ساختار زیر باشد:

$$G = (P \mid I_k)_{k\times n} \qquad (۲)$$

آنگاه I_k یک ماتریس همانی و P یک ماتریس $k \times (n-k)$ است که در این صورت کد بلوکی حاصل‌شده یک کد سامانمند است. در کدهای سامانمند $(n-k)$ مؤلفه اول کلمه کد مربوط به بیت‌های بررسی توازن و k مؤلفه آخر همان بیت‌های اطلاعات هستند (یا برعکس). می‌توان نشان داد که هر کد بلوکی خطی دارای یک معادل سامانمند است، به این صورت که می‌توان با انجام عملیات مقدماتی روی سطرهای ماتریس مولد و جابه‌جایی ستون‌های آن به فرم رابطه (۲) رسید [۱۵].

همان‌طور که پیش‌تر بیان شد هدف ما در این مقاله بازشناسی کور کدهای بلوکی خطی باینری سامانمند است. قبل از ارائه روش‌های پیشنهادی برای بازشناسی کور کدهای بلوکی، لازم است چند فرض به‌صورت زیر در نظر گرفت:

فرض ۱: در بلوک کدگذار کانال از کد بلوکی خطی باینری سامانمند استفاده شده است.

فرض ۲: احتمال بیت‌های ۰ و ۱ در رشته بیت ورودی به کدگذار کانال با هم برابر نیست که این موضوع برای اغلب خروجی‌های کدگذار منبع صدق می‌کند ($P_r(0) \neq P_r(1)$). درواقع به همین دلیل در مخابرات پدیده‌ای با نام سیگنالینگ غیریکنواخت وجود دارد. یکی از کدگذارهای منبع که به غیریکنواخت بودن مشهور است کدگذار گفتار با نام CELP[1] است. در این کدگذار احتمال بیت‌های خروجی بین صفر و یک یکسان نیست و درنتیجه برای شنود سیگنال گفتار می‌توان از روش‌های پیشنهادی استفاده کرد.

بنابراین، با توجه به مطالب بیان‌شده و به‌شرط اینکه رشته بیت به‌درستی تنظیم شده باشد، طول کلمه کد و طول بلوک اطلاعات تخمین زده می‌شوند.

۳. روش‌های پیشنهادی

همان‌طور که بیان شد، قبلاً روشی آماری برای تشخیص طول کلمه کد و طول بلوک اطلاعات ارائه شده است [۱۴]. اما این روش تنها برای کدهای با بعد کم و نرخ خطای پایین می‌تواند پارامترهای کد را به‌درستی بازشناسی کند. در این بخش ضمن تشریح روش بازشناسی طول کلمه کد، چند الگوریتم برای بازشناسی طول بلوک اطلاعات ارائه می‌شود که برای کدهای با طول بلند و درصد خطای بالا عمل بازشناسی را به‌درستی انجام می‌دهد.

دو فرض اساسی در این مقاله در نظر گرفته شده‌اند. فرض اول اینکه کدهای سامانمند مورد بررسی قرار می‌گیرند یعنی بیت‌ها به نحوی سازمان‌دهی می‌شوند که بیت‌های اطلاعات از بیت‌های بررسی توازن جدا باشند. فرض دوم به این صورت است که احتمال بیت‌های ۰ و ۱ ورودی به کدگذار کانال یکسان نیست. به‌این‌ترتیب می‌توان نشان داد که احتمال ۰ و ۱ مربوط به بیت‌های اطلاعات متفاوت از احتمال ۰ و ۱ مربوط به بیت‌های بررسی توازن است. اگر در رشته بیت ورودی به کدگذار کانال، a همان احتمال بیت ۱ و b احتمال بیت ۰ باشد پس داریم $a+b=۱$ (به‌طور مثال می‌توانیم a=۰/۶ و b=۰/۴ در نظر بگیریم). حال اگر رشته بیت کد شده در یک ماتریس مرتب شود به‌طوری‌که هر کلمه کد در یک ستون قرار بگیرد، طبق رابطه (۲) $n-k$ سطر اول همان بیت‌های بررسی توازن و k سطر دوم مربوط به بیت‌های اطلاعات است. می‌توان نشان داد که احتمال ۰ و ۱ در سطرهای مربوط به بررسی توازن نزدیک به میانگین احتمال‌های a و b است درحالی‌که احتمال ۰ و ۱ در سطرهای مربوط به اطلاعات مطابق احتمال معرفی‌شده a و b است.

برای بازشناسی طول کلمه کد (n) رشته بیت دریافتی در سمت گیرنده به‌صورت یک ماتریس مرتب می‌شود. این ماتریس n_0 سطر دارد. n_0 یک متغیر است و اولین مقداری که می‌تواند اختیار کند برابر ۱ است و در هر مرحله یک واحد به مقدار آن اضافه می‌شود. روند پر شدن ماتریس به این صورت است که ابتدا ستون اول ماتریس پر می‌شود. بعد از پر شدن ستون اول، بیت بعدی در ستون دوم از سطر اول قرار می‌گیرد و این روند ادامه دارد. برای هر مقدار n_0 که از ۱ شروع می‌شود، احتمال‌های ۰ و ۱ مربوط به هر سطر (از سطر اول الی سطر n_0) محاسبه می‌شود. سپس واریانس این احتمال‌ها نیز برای هر سطر محاسبه می‌گردد.

$$\mathrm{Var}_{\mathrm{row}}(i) = \mathrm{variance}\left(P_r(۰), P_r(۱)\right)$$

$$۱ \le i \le n_0 \qquad (۳)$$

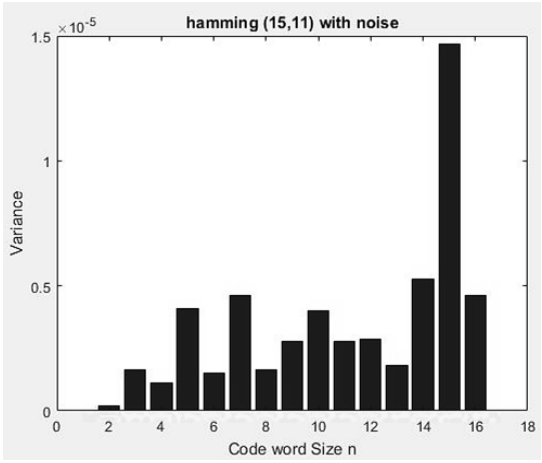
[1] Code-excited linear prediction

واضح است که واریانس احتمال‌های سطرهای اطلاعات در مقایسه با واریانس احتمال‌های سطرهای بررسی توازن مقدار بزرگ‌تری دارد. بعد از محاسبه واریانس احتمال‌های هر سطر برای مقادیر مختلف n_0، واریانسِ این واریانس‌ها حساب می‌شود.

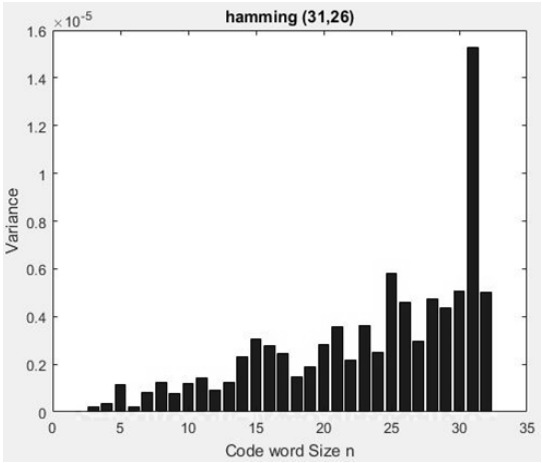
$$\mathrm{Var} = \mathrm{variance}(\mathrm{Var_{row}}(۱), \mathrm{Var_{row}}(۲), \dots, \mathrm{Var_{row}}(n_0)) \qquad (۴)$$

با مشاهده رابطه (۴) برای مقادیر مختلف n_0 می‌توان مقدار صحیح n را پیدا کرد، به‌طوری‌که اگر رشته بیت به‌درستی تنظیم شده باشد مقدار واریانسِ واریانس‌ها در $n_0 = n$ بیشینه می‌شود.

شکل‌های (۳-۲) به ترتیب یک فایل کد شده توسط کد همینگ (۱۱ و ۱۵) و (۲۶ و ۳۱) را نشان می‌دهد که به ۴۵٪ خطا آغشته شده‌اند. همان‌طور که انتظار می‌رفت مقدار واریانسِ واریانس‌ها (رابطه (۴)) در n=۱۵ و n=۳۱ به حداکثر رسیده است. بنابراین n=۱۵ و n=۳۱ همان طول کلمه کد شناسایی شده است.



شکل ۲. واریانسِ واریانس‌های احتمال‌های ۰ و ۱ هر سطر ماتریس دریافتی کد (۱۱ و ۱۵) برای مقادیر مختلف n_0



شکل ۳. واریانسِ واریانس‌های احتمال‌های ۰ و ۱ هر سطر ماتریس دریافتی کد (۲۶ و ۳۱) برای مقادیر مختلف n_0

حال که طول کلمه کد مشخص شده است می‌توان رشته بیت دریافتی در سمت گیرنده را در یک ماتریس با n سطر به فرم زیر مرتب کرد:

$$\tilde{C} = \begin{bmatrix} c_1 & c_{n+1} & c_{2n+1} & \cdots & \cdots & \cdots \\ c_2 & c_{n+2} & c_{2n+2} & \cdots & \cdots & \cdots \\ c_3 & c_{n+3} & c_{2n+3} & \cdots & \cdots & \cdots \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ c_n & c_{2n} & c_{3n} & \cdots & \cdots & \cdots \end{bmatrix} \qquad (۵)$$

سپس واریانس احتمال‌های ۰ و ۱ برای n سطر موجود محاسبه می‌شود. اکنون n تا مقدار واریانس به‌دست آمده است. از قبل به سامانمند بودن کد واقف هستیم و نیز می‌دانیم که واریانس احتمال‌های مربوط به سطرهای اطلاعات مقدار بزرگ‌تری دارد. بنابراین، برای بازشناسی طول بلوک اطلاعات باید به دنبال روشی باشیم که دسته واریانس‌های مربوط به سطرهای اطلاعات را از دسته واریانس‌های سطرهای بررسی توازن جدا کند. به‌عبارت‌دیگر، مرز بین دو دسته را به‌درستی تشخیص دهد تا مقدار k شناسایی شود. پس هدف پیدا کردن یک مقدار مناسب برای آستانه است.

ابتدایی‌ترین روش برای پیدا کردن مقدار آستانه این است که از مقادیر واریانس‌های موجود میانگین بگیریم. در این صورت سطرهایی که واریانس آن‌ها بالاتر از مقدار میانگین باشد متناظر با سطرهای پیام و سطرهایی که واریانسی زیر مقدار آستانه دارند، سطرهای بررسی توازن هستند. تعداد سطرهایی که واریانسی بزرگ‌تر از مقدار آستانه دارند نشان‌دهنده طول بلوک اطلاعات است. اما انتخاب میانگین به‌عنوان آستانه تنها برای کدهای با طول کوتاه و درصد پایین خطا، منجر به جداسازی صحیح بیت‌های بررسی توازن از بیت‌های اطلاعات و بازشناسی صحیح طول بلوک اطلاعات می‌شود. به‌محض افزایش طول کد حتی برای مقادیر پایین خطا، و نیز بدون خطا این روش هیچ نتیجه‌ای نمی‌دهد. پس باید به‌دنبال روشی برای تعیین آستانه باشیم که قادر به بازشناسی طول بلوک اطلاعات برای کدهای با طول بلند و درصد خطای بالا باشد. در این قسمت ۳ الگوریتم برای تعیین این آستانه ارائه می‌شود:

۳-۱. الگوریتم K-means

این الگوریتم یکی از ساده‌ترین الگوریتم‌های خوشه‌بندی است که دارای روالی تکراری است. اساس کار این الگوریتم به این صورت است که ابتدا باید تعداد خوشه‌های مدنظر تعیین شود (در اینجا تعداد خوشه‌ها یعنی k برابر ۲ است). سپس الگوریتم با توجه به تعداد خوشه‌های تعیین‌شده، به‌صورت تصادفی تعدادی نمونه را از میان n عضو، به‌عنوان مراکز خوشه انتخاب می‌کند. هرکدام از n-k داده باقی‌مانده با توجه به میزان نزدیکی به یکی از این k

خوشه نسبت داده می‌شوند. بعد از تخصیص همه اعضا، مراکز خوشه مجدداً از طریق میان‌گیری محاسبه می‌شوند و این کار تا زمانی که مراکز خوشه‌ها ثابت بماند ادامه می‌یابد [۱۶].

بهترین خوشه‌بندی آن است که مجموع تشابه بین مرکز خوشه و همه اعضای خوشه را حداکثر و مجموع تشابه بین مراکز خوشه‌ها را حداقل کند. تابع زیر به‌عنوان تابع هدف مطرح است:

$$J = \sum_{i=1}^{n} \sum_{j=1}^{k} r_{ij} \left\| x_i - \mu_j \right\|^2 \qquad (۶)$$

مجموعه $\{x_1, x_2, \ldots, x_n\}$ همان n نمونه داده و $\{\mu_1, \mu_2, \ldots, \mu_k\}$ مراکز خوشه jام است. r_{ij} بیانگر تعلق یا عدم تعلق نمونه iام به خوشه jام است. مقدار J بیانگر مربعات فاصله نمونه داده از مراکز خوشه‌ها است، پس هدف یافتن مقادیری برای r_{ij} و μ_j به‌طوری‌که J کمینه شود. این الگوریتم به n مقدار واریانس‌های احتمال‌های ۰ و ۱ سطرهای ماتریس $\tilde{C}$ اعمال می‌شود. درنتیجه مقادیر به دو دسته تقسیم می‌شوند. دسته با میانگین واریانس بیشتر همان دسته مربوط به بیت‌های اطلاعات است. بنابراین، تعداد واریانس‌های موجود در این دسته نشان‌دهنده طول بلوک اطلاعات است. عملکرد این روش برای تشخیص صحیح طول بلوک اطلاعات در بخش چهارم بررسی شده است.

۳-۲. الگوریتم Jenks Natural Breaks

یکی دیگر از الگوریتم‌های خوشه‌بندی که می‌خواهیم برای بازشناسی طول بلوک اطلاعات از آن استفاده کنیم، الگوریتم Jenks است. این الگوریتم نسبت به الگوریتم K-means برای داده‌های یک‌بعدی بهتر عمل خواهد کرد. هدف این الگوریتم نیز دسته‌بندی n مقدار داده با کاهش واریانس درون هر دسته و به حداکثر رساندن واریانس بین دسته‌هاست. این روش نیز مانند سایر الگوریتم‌های خوشه‌بندی نیاز به یک فرآیند تکراری دارد، به این معنی که محاسبات با انتخاب‌های مختلف در مجموعه داده‌ها تکرار می‌شود تا تعیین شود که کدام‌یک از دسته‌های انتخاب‌شده دارای کوچک‌ترین واریانس داخلی است. تقسیم اولیه داده‌ها می‌تواند به‌صورت دلخواه باشد.

به‌طور خلاصه این الگوریتم ۴ مرحله دارد:

- محاسبه مجموع مربع انحرافات بین دسته‌های ایجادشده (SDBC[1])
- محاسبه مجموع مربع انحرافات از آرایه میانگین (SDAM[2])
- محاسبه مجموع مربع انحرافات از میانگین‌های دسته‌های ایجادشده (SDCM[3])
- انتقال داده از دسته با SDBC بزرگ‌تر به دسته با SDBC کوچک‌تر

درنهایت انحرافات دسته جدید محاسبه می‌شود و این روند تا زمانی که انحرافات درون دسته به حداقل برسد تکرار می‌شود، پس تمام ترکیبات مورد بررسی قرار می‌گیرد و SDCM برای هر ترکیب محاسبه می‌شود و ترکیب با کمترین SDCM انتخاب می‌شود [۱۷]. ازآنجاکه تمام ترکیبات مورد بررسی گرفته است، تضمین می‌کند که دسته با کمترین SDCM پیدا شده است. در آخر مقدار GVF[4] که به‌صورت $GVF=\frac{SDAM-SDCM}{SDAM}$ تعریف می‌شود محاسبه می‌گردد. مقدار GVF از ۰ (بدترین حالت) تا ۱ (بهترین حالت) می‌تواند تغییر کند. بنابراین هرچه مقدار SDCM کوچک‌تر باشد (تغییرات در دسته‌ها کمتر باشد) ، GVF به ۱ نزدیک‌تر خواهد بود. در بخش چهارم عملکرد این روش بررسی شده است.

اگرچه الگوریتم معرفی‌شده در این بخش و بخش قبل توانایی تشخیص طول بلوک اطلاعات برای کدهای با درصد خطای بالا و طول کد بلند را دارند اما به‌دلیل تصادفی بودن روند کار این دو الگوریتم، روشی در بخش بعد ارائه می‌دهیم که ضمن عملکرد بهتر نسبت به دو الگوریتم معرفی‌شده، کاملاً غیر تصادفی است.

۳-۳. روش ابتکاری

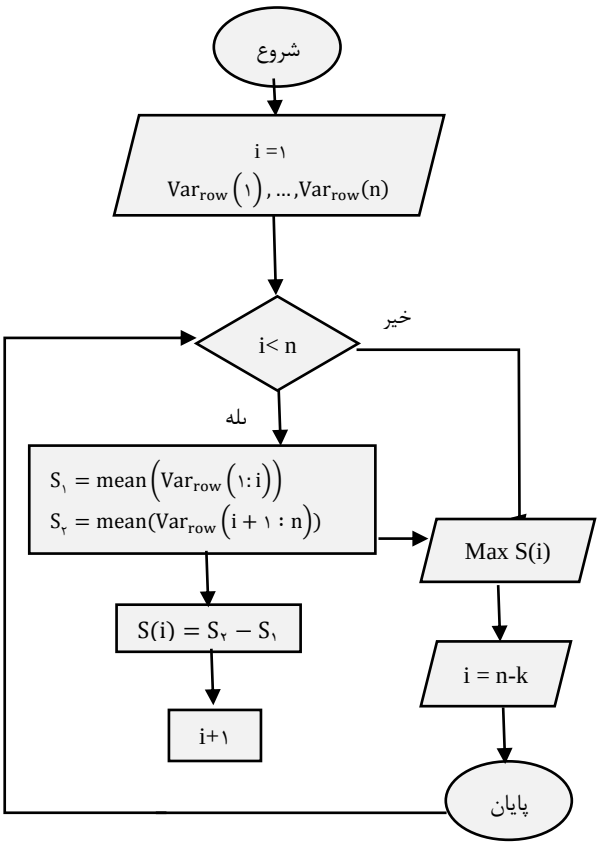
در این بخش روشی برای شناسایی طول بلوک اطلاعات ارائه می‌شود که عملکرد بهتری نسبت به الگوریتم‌های معرفی‌شده دارد. با توجه به اینکه n تا مقدار واریانس داریم، ابتدا میانگین واریانس‌های ۱ تا i و i+۱ تا n محاسبه می‌شود و به ترتیب آن‌ها را S_1 و S_2 می‌نامیم. مقدار i می‌تواند از ۱ تا n-۱ تغییر کند. سپس مقدار S_1 را از S_2 کم می‌کنیم.

$$S = S_2 - S_1 \qquad (۷)$$

در مرحله آخر بررسی می‌کنیم که برای کدام مقدار i حاصل $S_2 - S_1$ حداکثر می‌شود، مقدار i مرز بین بیت‌های اطلاعات و بیت‌های بررسی توازن را مشخص می‌کند. درواقع این روش با علم بر اینکه در کدهای سامانمند بیت‌های بررسی توازن در ابتدای کد و بیت‌های اطلاعات در انتها (یا برعکس) قرارگرفته‌اند بنیان نهاده شده است. به‌علاوه ما می‌دانیم که واریانس مربوط به احتمال‌های بیت‌های اطلاعات بزرگ‌تر از واریانس مربوط به احتمال‌های بیت‌های بررسی توازن است، درنتیجه آستانه باید جایی در میانه بیت‌ها باشد. با مشخص شدن مقدار *i*، تعداد اعضای دسته‌ای که میانگین واریانس بزرگ‌تری دارد نشان‌دهنده طول بلوک اطلاعات است. در اینجا چون ساختار ماتریس مولد G را به‌صورت $G = (P \mid I_k)_{k\times n}$ در نظر گرفتیم پس بیت‌های بررسی توازن در

[1] Squared Deviations Between Classes
[2] Squared Deviations From The Array Mean
[3] Squared Deviations From The Class Means
[4] Goodness of Variance Fit

ابتدای یک کلمه کد قرار گرفته‌اند. مقدار i که موجب حداکثر شدن S می‌شود نشان‌دهنده تعداد بیت‌های بررسی توازن (یعنی n-k) است. به‌عبارت دیگر این روش از خواص نهادینه‌شده در خود بردار کد برای به‌دست آوردن آستانه استفاده می‌کند. روندنمای مربوط به این روش در شکل (۴) آمده است.

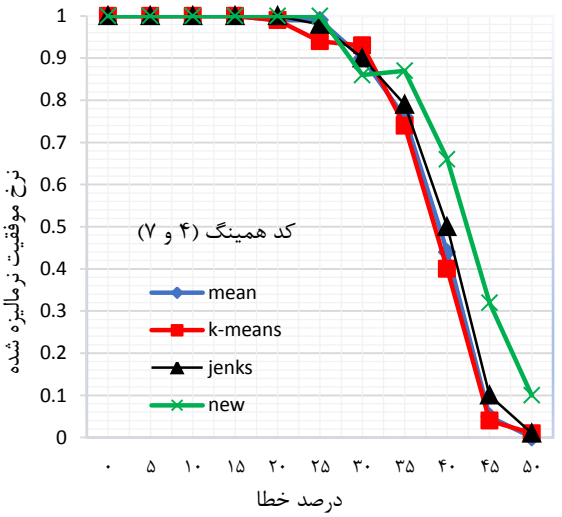


شکل۴. روندنمای روش ابتکاری

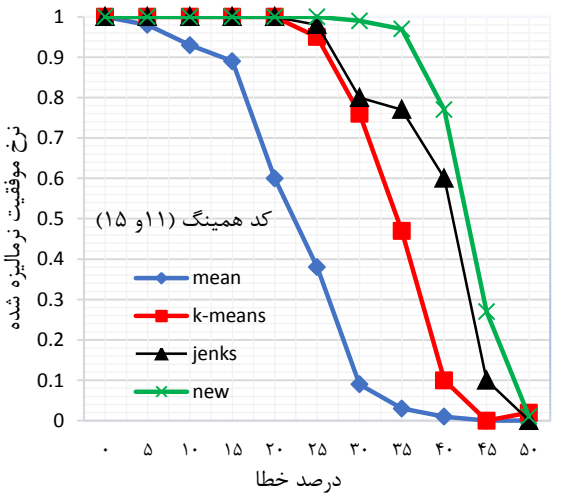
۴. نتایج شبیه‌سازی

برای ارزیابی روش‌های ارائه‌شده با استفاده از نرم‌افزار متلب شبیه‌سازی تهیه گردید. برای تعیین کارایی این روش‌ها، کد همینگ (۴ و ۷)، کد همینگ (۱۱ و ۱۵)، کد همینگ (۲۶ و ۳۱) و کد همینگ (۲۴۷ و ۲۵۵) در نظر گرفته شد. برای هرکدام از کدها k × ۲۵۰۰ بیت تصادفی تولید و کد می‌شود. احتمال ۰ و ۱ ورودی به کدگذار کانال توسط کاربر تعیین می‌گردد. در مورد هرکدام از کدها ۱۰۰ فایل برای درصدهای مختلفی از خطا تولید می‌شود. درنهایت الگوریتم‌های آماری ارائه‌شده بر روی رشته بیت دریافتی نویزی در سمت گیرنده اعمال می‌شود و به مقایسه نتایج حاصل از اعمال این روش‌ها با روش ارائه شده در [۱۴] خواهیم پرداخت.

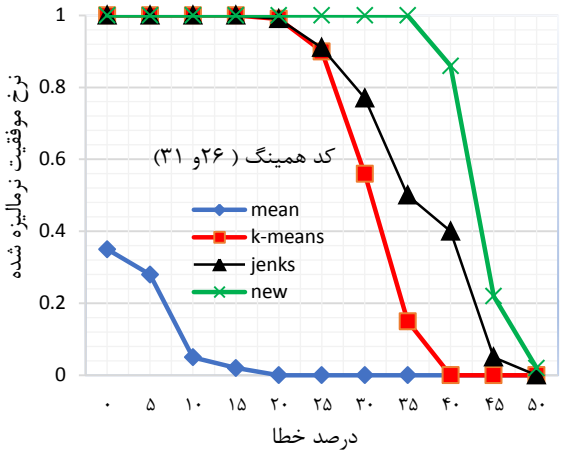
همان‌طور که در بخش پیش بیان شد قبل از محاسبه طول بلوک اطلاعات لازم است تا طول کلمه کد به‌درستی شناسایی شود. در این میان ۳ عامل می‌توانند بر نتیجه بازشناسی تأثیر بگذارند که عبارت‌اند از طول کد، درصد خطا و احتمال بیت‌های ۰ و ۱ در رشته بیت ورودی به کدگذار کانال. به این صورت که هرچه طول کد و درصد خطا مقدار بزرگ‌تری داشته باشند و همچنین احتمال‌های ۰ و ۱ نیز به ۰/۵ نزدیک‌تر باشند، بازشناسی کور مقادیر n و k دشوارتر خواهد بود. پس با نزدیک شدن احتمال‌های ۰ و ۱ به ۰/۵ و بالا رفتن درصد خطا ممکن است طول کلمه کد در همان ابتدای کار اشتباه تشخیص داده شود. بنابراین، لازم است به‌شرط بازشناسی صحیح طول کد به بازیابی طول بلوک اطلاعات بپردازیم. شکل‌های (۸-۵) نرخ موفقیت نرمالیزه شده را با استفاده از ۳ روش معرفی‌شده در این مقاله و روش میانگین برای درصدهای مختلف خطا نشان می‌دهد. در این حالت احتمال بیت‌های ۰ و ۱ به‌صورت a=۰/۷ و b=۰/۳ در نظر گرفته شده است.



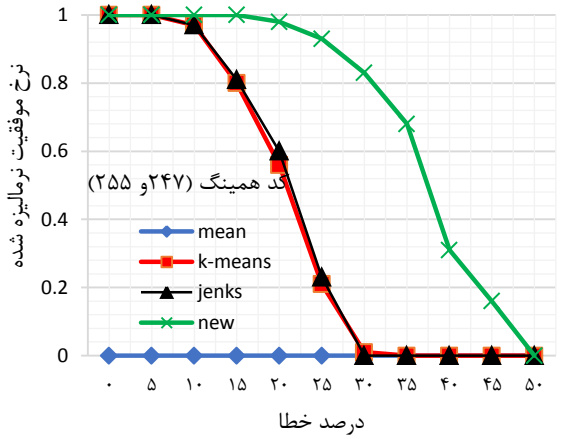
شکل ۵. نمودار احتمال بازشناسی صحیح پارامترهای کد همینگ (۴ و ۷) برحسب درصد خطا



شکل ۶. نمودار احتمال بازشناسی صحیح پارامترهای کد همینگ (۱۱ و ۱۵) برحسب درصد خطا



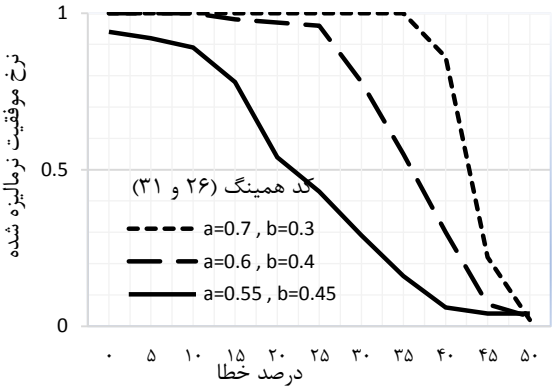
شکل ۷. نمودار احتمال بازشناسی صحیح پارامترهای کد همینگ (۲۶ و ۳۱) برحسب درصد خطا



شکل ۸. نمودار احتمال بازشناسی صحیح پارامترهای کد همینگ (۲۴۷ و ۲۵۵) برحسب درصد خطا

با توجه به نتایج مشخص است که با بزرگ شدن طول کلمه کد و نیز درصد خطا روش‌های ارائه‌شده در این مقاله به‌خصوص الگوریتم ابتکاری عملکرد بهتری از روش ارائه‌شده در [۱۴] دارند. برای کد همینگ (۲۶ و ۳۱) الگوریتم ابتکاری تا ۳۵٪ خطا می‌تواند پارامترهای طرح کدینگ را ۱۰۰٪ صحیح تشخیص دهد، درحالی‌که احتمال بازشناسی صحیح با روش میانگین حتی برای حالت بدون خطا برابر با ۳۵٪ است. همچنین برای کد (۲۴۷ و ۲۵۵) الگوریتم ابتکاری تا نزدیک به ۲۰٪ خطا قطعاً پارامترهای کد را به‌درستی بازشناسی می‌کند درحالی‌که روش میانگین برای این کد به‌هیچ‌عنوان قادر به بازشناسی نیست.

همان‌طور که پیش‌تر بیان شد مقادیر احتمال‌های ۰ و ۱ در رشته بیت ورودی به کدگذار کانال می‌تواند بر روی احتمال بازشناسی صحیح تأثیر بگذارد. هرچقدر احتمال ۰ و ۱ به ۰/۵ نزدیک‌تر باشد، اختلاف واریانس‌های احتمال‌های ۰ و ۱ برای سطرهای مختلف کاهش می‌یابد و تمایز قائل شدن بین این مقادیر نزدیک به هم دشوارتر خواهد بود. در شکل (۹) مقایسه‌ای بین حالت‌های مختلف انتخاب احتمال‌های ۰ و ۱ برای کد همینگ (۲۶ و ۳۱) با استفاده از الگوریتم ابتکاری صورت گرفته است. همان‌طور که بیان شد بهترین نتیجه برای حالت انتخاب a=۰/۷ و b=۰/۳ به‌دست آمده است.



شکل ۹. احتمال بازشناسی صحیح پارامترهای کد همینگ (۲۶ و ۳۱) برای حالت‌های مختلف انتخاب احتمال‌های ۰ و ۱

ازآنجایی‌که در مبحث بازشناسی کور کدهای بلوکی عامل زمان بسیار مهم است، لازم است تا برای مشاهده کارایی الگوریتم‌های پیشنهادی در مقابل روش میانگین سرعت الگوریتم‌ها را نیز مورد مقایسه قرار دهیم. برای این منظور ابتدا احتمال تشخیص صحیح پارامترهای کدینگ مربوط به ۳ کد همینگ (۴ و ۷)، (۲۶ و ۳۱) و (۲۴۷ و ۲۵۵) آغشته به ۲۵٪ خطا را برای حالت a=۰/۷ و b=۰/۳ در جدول (۱) آوردیم. احتمال تشخیص صحیح طول کد برای ۲۵٪ خطا ۱۰۰٪ است. پس احتمال‌های موجود در جدول (۱) مربوط به احتمال بازشناسی صحیح طول بلوک اطلاعات است. درنهایت زمان شبیه‌سازی تشخیص پارامتر طول بلوک اطلاعات برای حالتی که ۲۵٪ خطا رخ داده باشد را محاسبه و نتایج را در جدول (۲) آوردیم.

جدول ۱. احتمال بازشناسی صحیح طول بلوک اطلاعات برای ۲۵٪ خطا در حالت a=۰/۷ و b=۰/۳

روش / کد	ابتکاری	Jenks	K-Means	Mean
همینگ (۴ و ۷)	۱۰۰٪	۹۸٪	۹۴٪	**۹۹٪**
همینگ (۲۶ و ۳۱)	۱۰۰٪	۹۱٪	۹۰٪	۰٪
همینگ(۲۴۷ و ۲۵۵)	۹۳٪	۲۳٪	۲۱٪	۰٪

جدول ۲. زمان شبیه‌سازی بازشناسی طول بلوک اطلاعات برای ۲۵٪ خطا در حالت a=۰/۷ و b=۰/۳ برحسب ثانیه

روش / کد	ابتکاری	Jenks	K-Means	Mean
همینگ (۴ و ۷)	۰/۰۱۵۴	۰/۱۳۴۲	۰/۱۱۹۸	**۰/۰۰۰۶**
همینگ (۲۶ و ۳۱)	۰/۰۱۵۸	۰/۱۵۹۸	۰/۱۲۳۹	–
همینگ(۲۴۷ و ۲۵۵)	۰/۰۱۸۰	–	–	–

همان‌گونه که از نتایج پیداست روش میانگین تنها برای کدهای با طول کوتاه قادر به تشخیص طول بلوک اطلاعات است. در این حالت این روش سریع‌تر از الگوریتم‌های ارائه‌شده در این مقاله عمل کرده است. به‌محض افزایش طول کد، روش میانگین قادر به تشخیص صحیح طول بلوک اطلاعات نیست. با توجه به جدول (۲) زمان شبیه‌سازی تشخیص طول بلوک اطلاعات کد (۲۶ و ۳۱) برای روش میانگین به‌دلیل تشخیص اشتباه وارد نشده است. برای این کد الگوریتم پیشنهادی سریع‌تر از الگوریتم K-Means و الگوریتم K-Means سریع‌تر از الگوریتم Jenks عمل می‌کند. همچنین برای کد (۲۴۷ و ۲۵۵) علاوه بر روش میانگین، روش‌های k-Means و Jenks نیز منجر به بازشناسی نادرست می‌شوند و تنها الگوریتم ابتکاری همچنان قادر به تشخیص صحیح طول بلوک اطلاعات خواهد بود. بنابراین، می‌توانیم جمع‌بندی نتایج حاصل را به این صورت بیان کنیم که روش میانگین تنها برای کدهای با طول کوتاه و درصد خطای کم قابل استفاده است. الگوریتم K-Means برای کدهای با طول متوسط و درصد خطای کم نتایج مطلوبی دارد. همچنین احتمال تشخیص صحیح طول بلوک اطلاعات در الگوریتم Jenks نسبت به روش K-Means تا حدودی بهتر است، در مقابل زمان اجرای این روش نیز طبق جدول (۲) بیشتر است. اما روش ابتکاری نه‌تنها برای کدهای با طول کوتاه، بلکه برای کدهای با طول بلند و همچنین برای درصد خطاهای بزرگ و کم نتایج مطلوبی دارد. ضمن اینکه زمان اجرای روش ابتکاری از دو روش K-Means و Jenks کمتر است. همچنان که در نتایج دیده می‌شود روش پیشنهادی از همه روش‌ها بهتر عمل می‌کند. علت اصلی این مسئله را این‌طور می‌توان بیان کرد که روش پیشنهادی ما از قابلیت‌های نهادینه در خود کد استفاده می‌کند. مثلاً هیچ‌کدام از الگوریتم‌های دیگر این مسئله را مدنظر قرار نمی‌دهند که در کدهای سیستماتیک از یک بیت به بعد همگی از جنس بیت‌های توازن هستند و تا قبل از آن بیت‌های داده هستند.

۵. نتیجه‌گیری

در این مقاله روش‌ها و ایده‌های متفاوتی در ارتباط با موضوع بازشناسی کور کدهای بلوکی معرفی و مورد بررسی قرار گرفت. با توجه به فرمول‌های موجود در مراجع، الگوریتم‌هایی آماری ازجمله الگوریتم K-Means و الگوریتم Jenks Natural Breaks و درنهایت یک الگوریتم ابتکاری برای بازشناسی پارامترهای کدهای بلوکی (طول کلمه کد و طول بلوک اطلاعات) با طول بلند و آغشته به درصدهای مختلفی از خطا ارائه شد. روش‌های آماری ارائه‌شده در نرم‌افزار متلب شبیه‌سازی و تحلیل گردید. نتایج شبیه‌سازی در نرم‌افزار متلب نشان‌دهنده برتری الگوریتم‌های پیشنهادی ازلحاظ قدرت شناسایی نسبت به روش‌های آماری موجود است. به‌طوری‌که روش پیشنهادی، نه تنها برای کدهای با طول کوتاه، بلکه برای کدهای با طول بلند و همچنین برای درصد خطاهای بزرگ و کم نتایج مطلوبی دارد، ضمن اینکه زمان اجرای این روش از دو روش K-Means و Jenks کمتر است. بنابراین با توجه به اهمیت سرعت در سامانه‌های مخابراتی و حجم و پیچیدگی محاسباتی روش‌های ارائه‌شده، روش پیشنهادی برای کدهای با طول بلند پیشنهاد می‌شود.

۶. مرجع‌ها

[1] Filiol, E. "Reconstruction of Convolutional Encoders over GF (q)"; IMA Int. Conf. Cryptography and Coding 1997, 101-109.

[2] Rice, B. "Determining the Parameters of a Rate 1/n Convolutional Encoder over GF (q)"; Third Int. Con. Finite Fields and Applications, 1995.

[3] Valembois, A. "Detection and Recognition of a Binary Linear Code"; Discrete Applied Mathematics 2001, 111, 199-218.

[4] Cluzeau, M.; Finiasz, M. "Recovering a Code's Length and Synchronization from a Noisy Intercepted Bitstream"; IEEE Int. Sympos. Information Theory 2009, 2737-2741.

[5] Barbier J.; Letessier, J. "Forward Error Correcting Codes Characterization Based On Rank Properties"; Int. Conf. Wireless Communications & Signal Processing 2009, 1-5.

[6] Cluzeau, M. "Block Code Reconstruction Using Iterative Decoding Techniques"; IEEE Int. Sympos. Information Theory 2006, 2269-2273.

[7] Wang, J.; Yue, Y.; Yao, J. "A Method of Blind Recognition of Cyclic Code Generator Polynomial"; 6th Int. Conf. Wireless Communications Networking and Mobile Computing (WiCOM) 2010, 1-4.

[8] Teimouri, M.; Motlagh, H.; Haddadi, M. "Blind Recognition of BCH Product Codes"; J. Electrical Eng. 2017, 1, 49-54.

[9] Naseri A.; Maymanat, M. "Proposed Algorithm for Channel Coding Detection in Communication Surveillance Systems"; J. Passive Defence Sci & Technol. 2011, 2, 101-110.

[10] Xia T.; Wu, H. C. "Novel Blind Identification of LDPC Codes Using Average LLR of Syndrome a Posteriori Probability"; IEEE Trans. Signal Processing 2014, 62, 632-640.

[11] Moosavi R.; Larsson, E. G. "Fast Blind Recognition of Channel Codes"; IEEE Trans. Communications 2014, 62, 1393-1405.

[12] Swaminathan R.; Madhukumar, A. "Classification of Error Correcting Codes and Estimation of Interleaver Parameters in a Noisy Transmission Environment"; IEEE Trans. Broadcasting 2017, 63, 463-478.

[13] Mao, D. "Performance Bound Analysis on Hamming-Weight-Analysis Algorithm for Blind Recognition of Linear Block Codes"; Int. Conf. Communicatins and Networking in China 2017, 323-331.

[14] Sharma, A.; Pillai, N. R. "Blind Recognition of Parameters of Linear Block Codes from Intercepted Bit Stream"; Int. Conf. Computing, Communication and Automation 2016, 1262-1266.

[15] Lin S.; Costello, D. J. "Error Control Coding" Pearson-Prentice Hall, 2004.

[16] Likas, A.; Vlassis, N.; Verbeek, J. J. "The Global k-Means Clustering Algorithm"; Pattern Recognition 2003, 36, 451-461.

[17] Jenks, G. F. "Optimal Data Classification for Choropleth Maps," Department of Geographiy, University of Kansas Occasional Paper, 1977.

[18] Nasseri A.; Moghaddam, G. S. "The Proposed Intelligent Algorithm for Process Section in the Radar Interception Systems"; J. Passive Defence Sci. & Technol. 2011, 1, 87-98.