# الگوریتم ضد برخورد جدید مبتنی بر استاندارد EPC-C۱G۲ برای حل مشکل گرسنگی برچسب در سامانه‌های شناسایی بسامد رادیویی

امیر عباسیان و معصومه صفخانی

**چکیده: در این مقاله یک الگوریتم ضد برخورد جدید مبتنی بر استاندارد EPC-C۱G۲ برای حل مشکل گرسنگی برچسب‌ها در سامانه‌های شناسایی بسامد رادیویی ارائه می‌شود. عدم شناسایی قطعی برچسب‌ها در اسلات برخورد منجر به بروز مشکل گرسنگی برچسب می‌شود. در این مقاله برای حل این مشکل از موقعیت اولین بیت متفاوت- مقدار در عدد تصادفی ۱۶بیتی (یا RN۱۶) برچسب‌ها استفاده می‌شود. در واقع، قرائت‌گر در مواجهه با اسلات برخورد، موقعیت اولین بیت متفاوت- مقدار در RN۱۶ برچسب‌ها را به دست آورده و از آن برای شناسایی قطعی برچسب‌ها استفاده می‌کند. برخلاف الگوریتم‌های ضد برخورد اخیر که در آنها فرض شده است اطلاعات توسط برچسب‌ها یا به صورت هم‌زمان و یا به صورت غیر هم‌زمان ارسال می‌شود، در الگوریتم پیشنهادی برای به دست آوردن موقعیت اولین بیت متفاوت- مقدار فرضی مبنی بر ارسال هم‌زمان یا غیر هم‌زمان اطلاعات توسط برچسب‌ها در نظر گرفته نشده است. بر اساس شبیه‌سازی انجام‌شده، میانگین زمان شناسایی برای ۲۰۰ تا ۲۰۰۰ برچسب در الگوریتم ضد برخورد پیشنهادی نسبت به میانگین زمان شناسایی در الگوریتم‌های ضد برخورد EPC-C۱G۲، BIS و ERN۲ به ترتیب ۰/۹۲، ۰/۷۱ و ۰/۴۲ ثانیه کمتر است.**

***کلیدواژه:*** **ضد برخورد، متفاوت- مقدار، گرسنگی، شناسایی.**

## ۱- مقدمه

تشخیص اشیا با استفاده از بسامد امواج رادیویی به سامانه‌های RFID[1] مشهور شده است. یک سامانه RFID شامل تعدادی برچسب، قرائت‌گر و پایگاه داده است و با پخش سیگنال‌های رادیویی برای شناسایی خودکار اشیایی به کار می‌روند که برچسب‌ها به آنها ضمیمه شده‌اند. یک برچسب شامل یک ریزتراشه است که می‌تواند یک شناسه منحصر به فرد را ذخیره کند و توسط آنتن ضمیمه‌شده، شناسه منحصر به فرد خود را برای قرائت‌گر ارسال نماید [۱]. در واقع، قرائت‌گر برای شناسایی اشیا، اطلاعات برچسب متصل به آنها را به دست آورده و با کمک اطلاعات اضافی در پایگاه داده اشیا را شناسایی می‌کند.

در سامانه‌های RFID به طور کلی سه نوع برخورد وجود دارد: برخورد برچسب- برچسب[2]، قرائت‌گر- برچسب[3] و قرائت‌گر- قرائت‌گر[4]. برخورد برچسب- برچسب زمانی اتفاق می‌افتد که چندین برچسب به طور هم‌زمان به درخواست قرائت‌گر پاسخ دهند یعنی چندین برچسب شناسه خود را برای قرائت‌گر ارسال نمایند. در این حالت، هیچ کدام از برچسب‌ها به صورت موفقیت‌آمیز توسط قرائت‌گر شناسایی نمی‌شوند [۲]. هنگامی که از یک قرائت‌گر در مقیاس بزرگ استفاده می‌شود، برخورد[5] یکی از دلایل اصلی ناکارامدی انرژی در شناسایی برچسب‌ها است چرا که موجب اتلاف انرژی قرائت‌گر می‌گردد و زمان شناسایی برچسب‌ها را نیز افزایش می‌دهد [۳]. به همین دلیل الگوریتم‌های ضد برخورد برای افزایش کارایی سامانه‌های RFID ضروری و حیاتی هستند. به طور کلی این الگوریتم‌ها را می‌توان به چهار گروه تقسیم کرد که عبارتند از ۱) دسترسی چندگانه با تقسیم فضا (SDMA)[6]، ۲) دسترسی چندگانه با تقسیم بسامد (FDMA)[7]، ۳) دسترسی چندگانه با تقسیم کد (CDMA)[8] و ۴) دسترسی چندگانه با تقسیم زمان (TDMA)[9] [۴].

الگوریتم‌های ضد برخورد مبتنی بر TDMA بزرگ‌ترین گروه از الگوریتم‌های ضد برخورد را تشکیل می‌دهند. عموماً این الگوریتم‌ها را می‌توان به دو گروه قرائت‌گر محور[10] و برچسب محور[11] تقسیم کرد که به ترتیب RFT[12] و TFT[13] نیز نامیده می‌شوند. بیشتر الگوریتم‌هایی که از RTF استفاده می‌کنند را می‌توان به دو گروه مبتنی بر الوها[14] و مبتنی بر درخت تقسیم کرد اگرچه ممکن است ترکیبی از این دو گروه نیز استفاده شود [۴].

الگوریتم‌های ضد برخورد مبتنی بر الوها را می‌توان به سه گروه تقسیم کرد: ۱) الوهای خالص[15]، ۲) الوهای اسلات‌دار[16] و ۳) الوهای اسلات‌دار قاب‌دار[17]. الگوریتم‌های ضد برخورد مبتنی بر درخت در اصل برای دسترسی چندگانه در سامانه‌های بی‌سیم گسترش داده شده‌اند. همه الگوریتم‌های ضد برخورد مبتنی بر درخت، نیازمند قابلیت ساکت‌شدن توسط برچسب‌ها هستند به طوری که برچسب‌ها بعد از شناسایی خاموش شوند. الگوریتم‌های ضد برخورد مبتنی بر درخت را می‌توان به صورت زیر تقسیم‌بندی کرد: ۱) درخت تقسیم (TS)[18]، ۲) درخت پرس‌وجو (QT)[1]،

این مقاله در تاریخ ۷ بهمن ماه ۱۳۹۵ دریافت و در تاریخ ۱۴ بهمن ۱۳۹۶ بازنگری شد. این پژوهش با حمایت مالی دانشگاه تربیت دبیر شهید رجایی طبق قرارداد شماره ۲۷۷۷۰ مورخ ۱۳۹۵/۱۰/۲۵ انجام گردیده است.

امیر عباسیان، دانشکده مهندسی کامپیوتر، دانشگاه تربیت دبیر شهید رجایی، لویزان، تهران، (email: a.abbasian@srttu.edu).

معصومه صفخانی، دانشکده مهندسی کامپیوتر، دانشگاه تربیت دبیر شهید رجایی، لویزان، تهران، (email: safkhanin@srttu.edu).

1. Radio Frequency Identification
2. Tag-Tag
3. Reader-Tag
4. Reader-Reader
5. Collision
6. Space Division Multiple Access
7. Frequency Division Multiple Access
8. Code Division Multiple Access
9. Time Division Multiple Access
10. Reader Driven
11. Tag Driven
12. Reader-First-Talk
13. Tag-First-Talk
14. Aloha
15. Pure Aloha
16. Slotted Aloha
17. Framed Slotted Aloha
18. Tree Splitting

Archive of SID

www.SID.ir

۳) جستجوی دودویی [۲](BS) و ۴) داوری بیتی [۳](BTA) [۵].

الگوریتم‌های ضد برخورد مبتنی بر الوها دارای دو مشکل اساسی هستند: ۱) برای محدوده‌های کوچک با تعداد محدودی برچسب به کار می‌روند و ۲) دارای مشکل گرسنگی[۴] برچسب هستند. پدیده گرسنگی از آنجایی ناشی می‌شود که یک برچسب به صورت مکرر در اسلات برخورد قرار می‌گیرد و به صورت موفقیت‌آمیز شناسایی نمی‌شود. گرسنگی معمولاً به دلیل انتخاب عدد تصادفی توسط برچسب‌ها به عنوان شمارنده اسلات و عدم شناسایی قطعی برچسب‌ها در اسلات برخورد رخ می‌دهد.

در این مقاله، یک الگوریتم ضد برخورد برای شناسایی قطعی دو برچسب در اسلات برخورد ارائه شده است. در الگوریتم ضد برخورد پیشنهادی، قرائت‌گر در اسلات برخورد با استفاده از بیت‌های دریافتی از برچسب‌ها، موقعیت اولین بیت دارای مقدار متفاوت "۰" یا "۱" (بیت متفاوت- مقدار) را به دست می‌آورد. سپس با استفاده از این بیت اقدام به شناسایی قطعی برچسب‌ها در اسلات برخورد می‌کند. بنابراین در الگوریتم ضد برخورد مطرح‌شده مشکل گرسنگی برچسب وجود نخواهد داشت.

در قسمت ۲ مروری بر الگوریتم‌های ضد برخورد اخیر خواهیم داشت. در قسمت ۳ استاندارد EPC – C۱G۲ و کد منچستر را مرور می‌کنیم که اساس کار الگوریتم ضد برخورد پیشنهادی می‌باشند. الگوریتم ضد برخورد پیشنهادی در قسمت ۴ شرح داده خواهد شد. در قسمت ۵، نتایج الگوریتم پیشنهادی ارزیابی و با دیگر الگوریتم‌ها مقایسه خواهد شد. در نهایت این مقاله با نتیجه‌گیری در قسمت ۶ به اتمام می‌رسد.

## ۲- کارهای انجام‌شده

در [۶] الگوریتم ضد برخوردی مبتنی بر الوها مطرح گردیده که در آن پیشنهاد شده است هنگامی که برخورد بین دو برچسب اتفاق می‌افتد، این دو برچسب بدون شناسایی رها نشده و سعی شود که از قسمت‌های غیر برخورد شناسه برچسب‌ها که با موفقیت توسط قرائت‌گر دریافت شده‌اند، برای شناسایی برچسب‌ها استفاده شود. اما الگوریتم مطرح‌شده در [۶] قادر به شناسایی قطعی برچسب‌ها نیست. در [۷] الگوریتم ضد برخوردی مبتنی بر درخت ارائه شده که برای شناسایی برچسب‌ها از قسمت‌های برخورد برچسب‌ها استفاده می‌کند. در واقع در الگوریتم مطرح‌شده در [۷] قرائت‌گر با ارسال رشته‌ای به طول شناسه برچسب‌ها، موقعیت بیت‌های برخورد را اعلام می‌کند و برچسب‌ها این بیت‌ها را در ثبات داخلی خود ذخیره کرده و پردازش‌های بعدی برای شناسایی برچسب‌ها بر اساس این بیت‌ها انجام می‌گیرد.

در [۸] الگوریتم ضد برخوردی مبتنی بر استاندارد EPC – C۱G۲ مطرح شده که شامل دو فرایند پویش[۵] و شناسایی است. این الگوریتم در فرایند پویش به شناسایی اسلات‌های خالی می‌پردازد. تشخیص اسلات‌های خالی پویش بیتی [۶](BIS) نامیده می‌شود. بنابراین در فرایند شناسایی، اسلات خالی وجود نخواهد داشت و زمان شناسایی بهبود خواهد یافت. در الگوریتم BIS در پایان هر قاب قرائت‌گر برچسب‌های شناسایی‌نشده را با متغیر $n$ تخمین می‌زند و طول قاب جدید را با $L = m.n$ محاسبه می‌کند که $m$ یک عدد طبیعی است و باعث می‌شود که اندازه قاب بیشتر از تعداد برچسب‌ها باشد تا از اسلات برخورد اجتناب شود چرا که زمان اتلافی در اسلات برخورد بیشتر از اسلات خالی است.

در [۹] الگوریتم ضد برخوردی به نام ERNx[۷] برای بهبود زمان شناسایی برچسب‌ها مطرح شده که بر اساس پروتکل BIS عمل می‌کند که شامل دو فرایند پویش و شناسایی است. برخلاف BIS، این الگوریتم در فرایند پویش علاوه بر تشخیص اسلات‌های خالی، سعی در شناسایی اسلات‌های برخورد نیز دارد. البته دقت شناسایی اسلات‌های برخورد به تعداد بیت‌های به کار رفته در فرایند پویش بستگی دارد. $x$ تعداد بیت‌های استفاده‌شده در فرایند پویش را نشان می‌دهد. در ERN۲ احتمال شناسایی اسلات‌های برخورد در فرایند پویش ۰/۲۵ فرض شده است. یکی از معایب این الگوریتم، وجود اسلات‌های برخورد در فرایند شناسایی است که در فرایند پویش به اشتباه به عنوان اسلات موفق تشخیص داده شده است.

الگوریتم ضد برخورد جدیدی مبتنی بر استاندارد EPC – C۱G۲ در این مقاله پیشنهاد می‌شود که در آن همانند [۷] به شناسایی قطعی برچسب‌ها پرداخته می‌شود. الگوریتم پیشنهادی شامل دو فرایند پویش و شناسایی است. در فرایند پویش مانند BIS عمل می‌کند و همچنین در فرایند شناسایی، در مواجهه با اسلات برخورد، با تشخیص اولین بیت متفاوت- مقدار در عدد تصادفی ۱۶بیتی ارسالی توسط برچسب‌ها (RN۱۶)، اقدام به شناسایی قطعی دو برچسب می‌کند. برای این کار برچسب‌ها باید RN۱۶ خود را به صورت کد منچستر[۸] ارسال کنند تا قرائت‌گر قادر به تشخیص موقعیت بیت‌هایی با مقادیر متفاوت باشد. با توجه به این که اساس کار الگوریتم ضد برخورد پیشنهادی، استاندارد EPC – C۱G۲ و کد منچستر است در ادامه به شرح این مفاهیم می‌پردازیم.

## ۳- مفاهیم مربوط

### ۳-۱ استاندارد EPC – C۱G۲

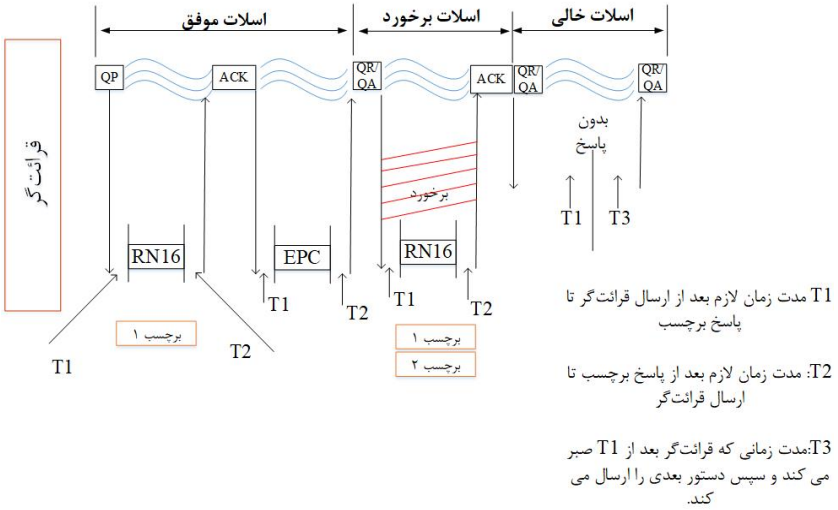
در استاندارد EPC – C۱G۲، قرائت‌گر فرایند شناسایی برچسب‌ها را با بسته پرس‌وجو $(QP)$ شروع می‌کند. این پرس‌وجو شامل پارامتر $Q$ نیز است که یک عدد صحیح بین ۰ تا ۱۵ است. برچسب‌ها این قابلیت را دارند که $Q$ را استخراج و شمارنده ۱۵بیتی اسلات‌هایشان را با عددی بین ۰ تا $2^Q - 1$ بارگذاری کنند. برچسب‌هایی که شمارنده اسلات آنها صفر است یک عدد تصادفی ۱۶بیتی تولید کرده و برای قرائت‌گر ارسال می‌کنند. این عدد تصادفی RN۱۶ نامیده می‌شود. اگر RN۱۶ توسط قرائت‌گر با موفقیت دریافت شود، قرائت‌گر با بسته تصدیق (ACK) به برچسب پاسخ می‌دهد. این بسته شامل RN۱۶ نیز است که برچسب باید آن را با مقداری که خود فرستاده است تطبیق دهد و در صورت تطابق، کد الکترونیکی محصول خود یعنی EPC[۹] منحصر به فرد خود را برای قرائت‌گر ارسال کند. بدین صورت برچسب به صورت موفقیت‌آمیز شناسایی می‌شود. در ادامه، قرائت‌گر پرس‌وجوی تکرار (Query Rep) یا پرس‌وجوی تنظیم (Query Adj) را برای تمام برچسب‌ها می‌فرستد که وابسته به مقدار $Q$ است.

مقدار $Q$ به وسیله مقدار $C$ افزایش یا کاهش می‌یابد که در مواجهه با اسلات خالی کاهش و در مواجهه با اسلات برخورد افزایش می‌یابد. $C$ می‌تواند هر مقداری مابین ۰/۱ تا ۰/۵ باشد. بعد از پایان قاب اگر مقدار $Q$ تغییر کند قرائت‌گر دستور Query Adj و در غیر این صورت دستور

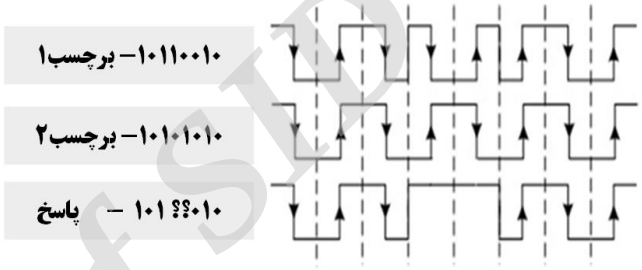
1. Query Tree
2. Binary Search
3. Bitwise Arbitration
4. Starvation
5. Scan
6. Bit Scanning
7. Enhanced Random Number-X-BITS
8. Manchester Code
9. Electronic Product Code

Archive of SID

www.SID.ir




شکل ۱: اسلات برخورد، موفق و خالی در استاندارد EPC – C۱G۲ [۹].




شکل ۲: رفتار کد منچستر با برخورد [۱۰].

Query Rep را ارسال می‌کند. در شکل ۱ اسلات‌های برخورد، موفق و خالی در استاندارد EPC – C۱G۲ نشان داده شده است [۹].

### ۳-۲ کد منچستر

در کد منچستر، ارزش هر بیت با تغییر سطح تعریف می‌شود (انتقال مثبت یا انتقال منفی) و انتقال مثبت برابر صفر و انتقال منفی برابر یک است. اگر دو (یا بیشتر) برچسب به طور هم‌زمان بیت‌هایی با مقادیر مختلف انتقال دهند، انتقال‌های مثبت و منفی در بیت‌های دریافت‌شده یکدیگر را لغو می‌کنند یعنی هیچ انتقالی وجود ندارد. این حالت در انتقال داده مجاز نیست و به عنوان خطا تشخیص داده می‌شود. بنابراین کد منچستر این امکان را فراهم می‌سازد که بتوان موقعیت بیت برخورد را تعیین کرد. در شکل ۲ برخورد در کد منچستر نشان داده شده است [۱۰].

### ۳-۳ مدل سیستم و فرضیات

در این مقاله فرض شده که یک قرائت‌گر در سامانه شناسایی بسامد رادیویی وجود دارد و برخوردها فقط از نوع برچسب– برچسب است. الگوریتم پیشنهادی در این مقاله مبتنی بر استاندارد EPC – C۱G۲ است که در آن شناسه برچسب‌ها ۹۶ بیت و عدد تصادفی تولیدشده توسط برچسب‌ها ۱۶ بیت است. اساس کار الگوریتم پیشنهادی مانند شکل ۱ است با این تفاوت که RN۱۶ به صورت کد منچستر برای قرائت‌گر ارسال می‌گردد و در نتیجه موقعیت بیت برخورد توسط قرائت‌گر تشخیص داده می‌شود. همچنین فرض شده که محیط بدون نویز بوده و وقتی برچسبی اطلاعاتی ارسال می‌کند، فقط برخورد با اطلاعات برچسب دیگر موجب خرابی اطلاعات ارسالی می‌شود.

## ٤- الگوریتم پیشنهادی

در این بخش به شرح جزئیات فرایند پویش و فرایند شناسایی الگوریتم ضد برخورد پیشنهادی می‌پردازیم.

### ٤-۱ فرایند پویش

در فرایند پویش، قرائت‌گر دستور پویش را ارسال کرده و به برچسب‌ها اندازه قاب فعلی $(L)$ را اطلاع می‌دهد. همچنین این دستور حاوی زمان $T_{waite}$ نیز است. زمانی که برچسب‌ها این دستور را دریافت کردند یک عدد تصادفی از بازه $[۰, L-۱]$ را انتخاب و در شمارنده اسلات‌هایشان بارگذاری می‌کنند. برچسب‌ها به اندازه $T_{waite}$ صبر کرده و سپس یک واحد از شمارنده اسلات خود کم می‌کنند. زمانی که شمارنده اسلات به صفر می‌رسد برچسب‌ها برای قرائت‌گر بیت "۱" را با استفاده از کد منچستر ارسال می‌کنند، بنابراین قرائت‌گر در اسلات‌های خالی بیتی دریافت نمی‌کند ولی در اسلات‌هایی که بیش از دو برچسب دارند (مثلاً چهار برچسب) که حداقل یکی از برچسب‌ها اطلاعات خود را نسبت به بقیه به صورت غیر هم‌زمان ارسال می‌کند، حالات زیر در دریافت قرائت‌گر اتفاق می‌افتد:

- بیش از دو "۱" دریافت می‌کند که در این حالت حداقل سه برچسب اطلاعات خود را به صورت غیر هم‌زمان ارسال می‌کنند.
- یا دو "۱" با بیت اول برخورد یا دو "۱" با بیت دوم برخورد دریافت می‌کند که در این حالت تنها سه برچسب اطلاعات خود را به صورت هم‌زمان ارسال می‌کنند.
- یا دو "۱" که هر دو بیت برخورد هستند، دریافت خواهد کرد که در این حالت برچسب‌ها دو به دو اطلاعات خود را به صورت هم‌زمان ارسال می‌کنند.

قرائت‌گر در پایان فرایند پویش دستور خاص دیگر به نام RBS[1] برای برچسب‌ها ارسال می‌کند که یک رشته $L$ بیتی است که در آن موقعیت متناظر با شماره اسلات‌هایی که توسط برچسب‌ها انتخاب نشده‌اند صفر است. در این رشته همچنین موقعیت متناظر با شماره اسلات‌هایی که توسط بیش از دو برچسب غیر هم‌زمان انتخاب شده‌اند نیز صفر است. برچسب‌هایی که شماره اسلات انتخابی آنها در رشته دریافتی صفر است

1. Response Bit Slot

Archive of SID

www.SID.ir

شکل ۳: وضعیت اسلات‌ها.

در طول قاب جاری خاموش خواهند بود و به درخواست‌های قرائت‌گر پاسخ نخواهند داد تا قاب جدید آغاز شود. بقیه برچسب‌ها با استفاده از تعداد صفرهای موجود در رشته $L$ بیتی، شمارنده اسلات‌هایشان را تنظیم می‌کنند. در واقع برچسب‌ها با استفاده از رشته $L$ بیتی دریافتی به تعداد صفرهایی که قبل از شماره اسلات انتخابی آنها در رشته $L$ بیتی قرار دارد از عدد شمارنده اسلات خود کم می‌کنند. برای مثال اگر شماره اسلات برچسبی ۵ باشد آن برچسب تعداد صفرهایی را که قبل از موقعیت ۵ در رشته $L$ بیتی قرار دارد از عدد شمارنده اسلات خود کم می‌کند. به این ترتیب، تمام اسلات‌های خالی و اسلات‌هایی با بیش از دو برچسب غیر هم‌زمان از قاب جاری حذف می‌شوند و طول قاب به $L' = L - c_0$ کاهش می‌یابد که $c_0$ تعداد صفرها در رشته $L$ بیتی است.

## ٤-۲ فرایند شناسایی

فرایند شناسایی در الگوریتم ضد برخورد پیشنهادی به صورت زیر اجرا می‌شود:

۱) قرائت‌گر فرایند شناسایی را با استفاده از دستور $QP$ و ارسال آن برای برچسب‌ها شروع می‌کند.

۲) برچسب‌هایی که دستور $QP$ را دریافت کرده‌اند و شمارنده اسلاتشان برابر صفر است، عدد تصادفی ۱۶بیتی (RN۱۶) را تولید و با استفاده از کد منچستر ارسال می‌کنند.

۳) اگر برچسب‌ها اعداد تصادفی خود را به صورت هم‌زمان ارسال کنند، قرائت‌گر موقعیت اولین بیت دارای مقدار متفاوت "۰" یا "۱" (موقعیت بیت متفاوت- مقدار) در RN۱۶ را به دست آورده و در ثبات ۴بیتی $bp$ ذخیره می‌کند. توجه کنید که شماره موقعیت بیت‌ها از صفر شروع می‌شود. در الگوریتم ضد برخورد پیشنهادی، دستور send-EPC ۵بیتی است که در آن ۴ بیت کم‌ارزش حاوی مقدار $bp$ و بیت پنجم می‌تواند صفر یا یک باشد که در این مقاله با $v$ نشان داده شده است. قرائت‌گر بعد از به دست آوردن موقعیت اولین بیت متفاوت- مقدار، ابتدا یک دستور send-EPC با بیت پنجم برابر صفر یعنی $v=0$ ارسال می‌کند. برچسب‌هایی که دستور send-EPC را دریافت کرده‌اند، بیتی که در موقعیت $bp$ از RN۱۶ قرار دارد را با مقدار $v$ مقایسه و در صورت مطابقت، EPC منحصر به فرد خود را ارسال می‌کنند. قرائت‌گر بعد از دریافت اولین EPC، بار دیگر دستور send-EPC ولی این بار با بیت پنجم برابر یک یعنی $v=1$ ارسال می‌کند. برچسب‌هایی که دستور send-EPC را دریافت کرده‌اند، بیتی که در موقعیت $bp$ از RN۱۶ قرار دارد را با مقدار $v$ مقایسه و در صورت مطابقت، EPC منحصر به فرد خود را ارسال می‌کنند. به این ترتیب هر دو برچسبی که در اسلات برخورد قرار گرفته‌اند به صورت قطعی و بدون نیاز به انتخاب و ارسال دوباره عدد تصادفی ۱۶بیتی شناسایی می‌شوند.

۴) در الگوریتم ضد برخورد پیشنهادی دستور send-again، ۸بیتی است که ۴ بیت کم‌ارزش حاوی مقدار $bp$ و ۴ بیت باارزش حاوی $i$ است. مقدار اولیه $i=0$ در نظر گرفته شده است. در حالتی که برچسب‌ها اطلاعات خود را به صورت غیر هم‌زمان ارسال می‌کنند قرائت‌گر عمل خواندن را تا اولین بیت برخورد در RN۱۶ ادامه می‌دهد و موقعیت اولین بیت برخورد را در ثبات $bp$ ذخیره می‌کند. سپس یک دستور send-again با مقدار $i=0$ را برای برچسب‌ها ارسال می‌کند. برچسب‌هایی که دستور send-again را دریافت کرده‌اند بیت‌های موقعیت‌های $i\times bp$ تا $\min((i+1)\times bp-1,15)$ از قسمت کم‌ارزش RN۱۶ خود را برای قرائت‌گر ارسال می‌کنند. به عنوان مثال، برچسب‌ها با دریافت دستور send-again ۸بیتی "۰۰۱۱۰۰۱۰" می‌فهمند که $i=0011$ و $bp=0010$ است و بنابراین باید بیت‌های موقعیت $i\times bp=3\times 2=6$ تا $\min((i+1)\times bp-1,15)$ که $\min((3+1)\times 2-1,15)=7$ از قسمت کم‌ارزش RN۱۶ خود را ارسال کنند. موقعیت بیت‌ها در RN۱۶ از صفر شروع می‌شود. قرائت‌گر پس از ارسال send-again، دو رشته $k=bp$ بیتی از برچسب‌ها را دریافت کرده و نظیر به نظیر مقایسه می‌کند تا موقعیت اولین بیت متفاوت- مقدار را به دست آورد. اگر قرائت‌گر موفق به یافتن موقعیت بیت متفاوت- مقدار در رشته‌های دریافتی نشود این مرحله را آن قدر با $i=i+1$ تکرار می‌کند تا موقعیت اولین بیت متفاوت- مقدار را به دست آورد. چنانچه قرائت‌گر در بیت‌های دریافتی بتواند بیت متفاوت- مقدار را بیابد، ابتدا مقدار $bp=i\times bp+position$ را محاسبه می‌کند که در آن $position$ موقعیت اولین بیت متفاوت- مقدار در رشته $k$ بیتی دریافت‌شده توسط قرائت‌گر است. سپس قرائت‌گر همانند مرحله قبل با ارسال دو دستور send-EPC با $v=0$ و $v=1$ دو برچسب را به صورت قطعی شناسایی می‌کند.

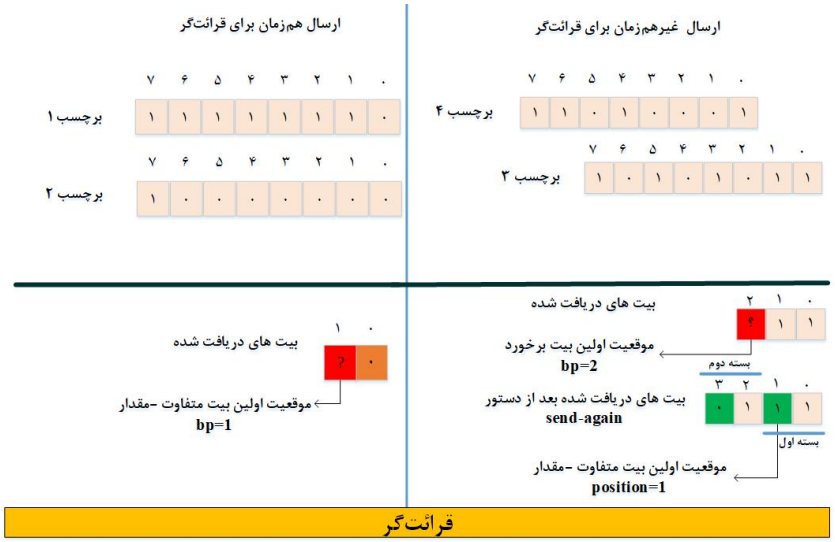
۵) در الگوریتم پیشنهادی قرائت‌گر در اسلات‌های موفق بعد از دریافت یک EPC و شناسایی یک برچسب اسلات بعدی را آغاز می‌کند. همچنین در اسلات‌های برخورد شامل دو برچسب بعد از دریافت دو EPC به صورت قطعی هر دو برچسب را شناسایی کرده و اسلات بعدی را آغاز می‌کند. در غیر این صورت بدون شناسایی برچسب‌ها، اسلات بعدی را آغاز می‌کند.

در ادامه با یک مثال مراحل اجرای الگوریتم را شرح می‌دهیم.

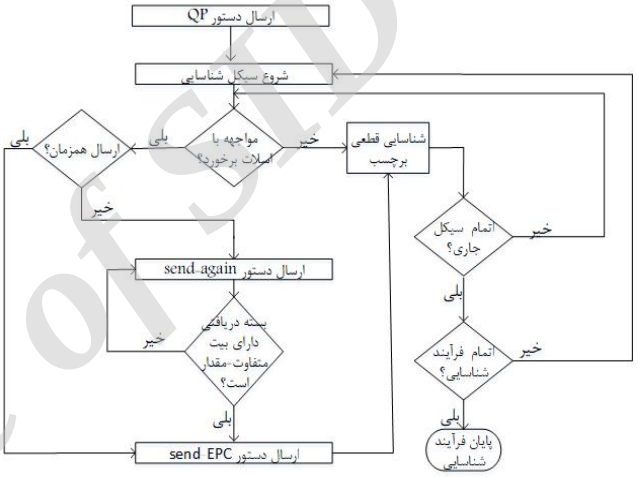
فرض کنید عدد تصادفی تولیدشده $RN8=8$، تعداد برچسب‌ها $n=5$ و طول قاب $L=4$ و وضعیت اسلات‌ها به صورت نشان داده شده در شکل ۳ باشد. همان طور که در شکل مشاهده می‌شود، اسلات موقعیت صفرم خالی است و بنابراین قرائت‌گر در مرحله پویش رشته بیت "۱۱۱۰" را ارسال می‌کند و از آنجایی که در رشته ارسالی تنها یک صفر در موقعیت قبل از شماره اسلات برچسب‌ها وجود دارد، برچسب‌ها یک واحد از عدد شمارنده اسلات خود کم می‌کنند. قرائت‌گر فرایند شناسایی را با $L'=3$ شروع می‌کند. بعد از فرایند پویش در اسلات صفرم، برچسب‌های ۱ و ۲ قرار دارند. با توجه به نیمه سمت چپ شکل ۴ فرض کنید برچسب‌های ۱ و ۲ به صورت هم‌زمان RN۸ خود را ارسال می‌کنند و اولین بیت متفاوت- مقدار در موقعیت اول ($bp=1$) اتفاق می‌افتد. بنابراین قرائت‌گر در چهار بیت کم‌ارزش دستور send-EPC عدد ۱ و بیت پنجم عدد ۰ ($v=0$) را قرار می‌دهد و دستور send-EPC را ارسال می‌کند. برچسب‌های ۱ و ۲ این دستور را دریافت کرده و مقدار $v$ را با

Archive of SID

www.SID.ir




شکل ۴: ارسال هم‌زمان و غیر هم‌زمان برای قرائت‌گر.




شکل ۵: نمودار گردشی فرایند شناسایی الگوریتم ضد برخورد پیشنهادی.

بیتی از RN۸ خود که در موقعیت ۱ قرار دارد، مقایسه کرده و در صورت تطابق EPC خود را ارسال می‌کنند. با توجه به شکل ۴ برچسب ۲ شناسایی می‌شود. قرائت‌گر دوباره دستور send-EPC را با $v=1$ ارسال می‌کند و بدین ترتیب برچسب ۱ نیز با ارسال EPCاش شناسایی می‌شود. سپس قرائت‌گر به اسلات ۱ می‌رود و آن را شروع می‌کند. در این اسلات برچسب‌های ۳ و ۴ قرار دارند که به صورت غیر هم‌زمان شناسه خود را ارسال می‌کنند که طبق نیمه راست شکل ۴ برخورد در موقعیت دوم اتفاق می‌افتد ($bp=2$). بنابراین قرائت‌گر دستور send-again را با $i=0$ (۴ بیت باارزش) و $bp=2$ (۴ بیت کم‌ارزش) ارسال می‌کند. برچسب‌های ۳ و ۴ دستور send-again را دریافت کرده و بیت‌های موقعیت‌های $i\times bp=0$ تا $\min((i+1)\times bp-1,15)=1$ از سمت کم‌ارزش RN۱۶ خود را ارسال می‌کنند. بنابراین قرائت‌گر رشته بیت "۱۱" را از برچسب ۳ و پس از آن رشته بیت "۰۱" را از برچسب ۴ دریافت می‌کند. همان طور که مشاهده می‌شود، موقعیت اولین بیت متفاوت– مقدار در بیت‌های دریافتی موقعیت یکم است ($position=1$). با استفاده از این اطلاعات، قرائت‌گر موقعیت اولین بیت متفاوت– مقدار در RN۸ را با $bp=i\times bp+position=0\times 2+1=1$ به دست می‌آورد و سپس دستور send-EPC را همان طور که پیش از این گفته شد ارسال می‌کند. با توجه به شکل ۴ با ارسال send-EPC با $v=0$ برچسب ۴ شناسایی می‌شود و به همین ترتیب با ارسال send-EPC با $v=1$، برچسب ۳ نیز شناسایی می‌شود. سپس قرائت‌گر اسلات بعد یعنی اسلات ۲ را آغاز می‌کند که اسلات موفق است و برچسب ۵ را شناسایی می‌کند و فرایند شناسایی بدین ترتیب به پایان می‌رسد.

در [۹] بیان شده که اگر در فرایند پویش، برچسب‌ها RN۱۶ را ارسال کنند، اسلات برخورد ۱۰۰٪ شناسایی خواهد شد. با توجه به این که در الگوریتم پیشنهادی در فرایند شناسایی، برچسب‌ها RN۱۶ را ارسال می‌کنند، الگوریتم موفق به یافتن اولین بیت متفاوت– مقدار خواهد شد. در شکل ۵ نمودار گردشی الگوریتم ضد برخورد پیشنهادی آمده است.

## ۴-۳ محاسبه کارایی

با توجه به این که زمان شناسایی برچسب‌ها یکی از عوامل مهم در کارایی الگوریتم‌های ضد برخورد است و الگوریتم پیشنهادی از پرس‌وجوی‌های بیشتری برای شناسایی برچسب‌ها در اسلات برخورد استفاده می‌کند، بنابراین در این بخش به تحلیل زمان شناسایی برچسب‌ها در الگوریتم پیشنهادی می‌پردازیم.

زمان شناسایی برچسب‌ها با استفاده از (۱) محاسبه می‌شود

$$T=M\times(T_{Scan}+T_{Identification}) \qquad (1)$$

که $M$ تعداد چرخه‌های مورد نیاز برای شناسایی تمامی برچسب‌ها است. مدت زمان لازم برای فرایند پویش با استفاده از (۲) محاسبه می‌شود

$$T_{Scan}=L\times T_{RT}+L\times(T_{waite}) \qquad (2)$$

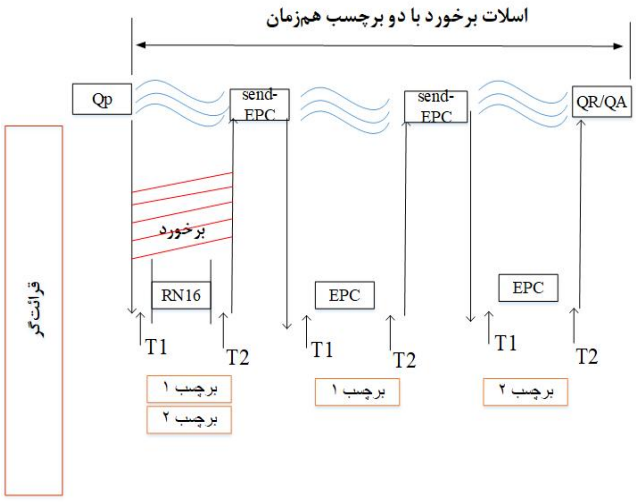
که در آن $T_{RT}$ مدت زمان لازم برای انتقال یک بیت از قرائت‌گر به برچسب، $L$ طول قاب جاری و $T_{waite}$ متوسط زمان مورد نیاز برای ارسال یک بیت توسط برچسب‌ها است. در واقع قرائت‌گر یک بیت "۱" برای برچسب‌ها ارسال می‌کند و برچسب در پاسخ، یک بیت "۱" با استفاده از کد منچستر برای قرائت‌گر ارسال می‌کنند و بنابراین $T_{waite}=T_1+T_2+T_{TR}+T_{RT}$ که در آن $T_{TR}$، مدت زمان لازم برای انتقال یک بیت از برچسب به قرائت‌گر است.

فرض کنید زمان شناسایی اسلات‌های موفق، برخورد حاوی دو برچسب و برخورد بیش از دو برچسب به ترتیب با $T_s$، $T_c$ و $T_u$ نشان داده شود. با توجه به شکل ۱ زمان شناسایی یک برچسب در اسلات موفق طبق (۳) محاسبه می‌شود

$$T_s=2\times(T_1+T_2)+T_{Query}+T_{RN16}+T_{ACK}+T_{EPC} \qquad (3)$$

Archive of SID

www.SID.ir




شکل ۶. فرایند شناسایی در اسلات برخورد با ارسال همزمان دو برچسب در الگوریتم ضد برخورد پیشنهادی.

با توجه به شکل ۶، زمان شناسایی دو برچسب در اسلات برخورد با دو برچسب همزمان، طبق (۴) محاسبه می‌شود

$$T_{C_1} = 3\times(T_1+T_2)+T_{Query}+T_{RN16}+2\times T_{send-EPC}+2\times T_{EPC} \quad (4)$$

در حالتی که برچسب‌ها اطلاعات خود را به صورت غیر همزمان ارسال کنند برخورد می‌تواند در موقعیت‌های ۱ تا $R-1=15$ از RN16 دو برچسب اتفاق بیفتد که احتمال وقوع همه حالات یک سان فرض شده است.

یعنی احتمال وقوع برخورد در موقعیت ۱ با احتمال وقوع برخورد در موقعیت ۱۵ برابر است. اگر برخورد در موقعیت $k$ از RN16 برچسب‌ها اتفاق بیفتد، احتمال پیداکردن یک بیت متفاوت- مقدار در رشته $k$ بیتی $1-(1/2)^k$ است و با توجه به این که مقدار $k$ از ۱ تا $R-1$ می‌تواند تغییر کند، بنابراین احتمال شناسایی یک بیت متفاوت- مقدار با ارسال یک دستور send-again دراسلات برخورد با دو برچسب غیر همزمان با استفاده از (۵) محاسبه می‌شود

$$p_{id} = \frac{1}{R-1}(1-(\frac{1}{2})^1)+\frac{1}{R-1}(1-(\frac{1}{2})^2)+ \ldots+\frac{1}{R-1}(1-(\frac{1}{2})^{R-1}) = \frac{1}{R-1}\sum_{i=1}^{R-1}1-(\frac{1}{2})^i \quad (5)$$

مقدار احتمال (۵) برابر ۰/۹۳۳۳۰ است.

میانگین تعداد بیت‌های ارسالی توسط برچسب‌ها برای شناسایی اولین بیت متفاوت- مقدار در پاسخ به یک دستور send-again در اسلات برخورد با دو برچسب غیر همزمان با استفاده از (۶) محاسبه می‌شود

$$L_{packet} = \frac{1+2+3+\ldots+R-1}{R-1} = \frac{\frac{R\times(R-1)}{2}}{R-1} = \frac{R}{2} = 8 \quad (6)$$

در (۵) و (۶)، $R$ طول عدد تصادفی تولیدشده توسط برچسب‌ها است که برابر ۱۶ است.

با توجه به (۵) فرض می‌کنیم در حالت غیر همزمان قرائت‌گر از یک دستور send-again استفاده می‌کند و برچسب‌ها با توجه به (۶) با رشته ۸بیتی به قرائت‌گر پاسخ می‌دهند. در این صورت $T_{c2}$ با توجه به شکل ۷ طبق (۷) محاسبه می‌شود

$$T_{c2} = 4\times(T_1+T_2)+T_{Query}+T_{RN16}+2\times T_{send-EPC}+T_{send-again}+2\times T_{EPC}+T_{Packet} \quad (7)$$

در اینجا $T_{Packet}$ مدت زمان لازم برای ارسال ۸ بیت از برچسب به قرائت‌گر، $T_{send-again}$ مدت زمان لازم برای ارسال ۸ بیت از قرائت‌گر به برچسب و $T_{send-EPC}$ مدت زمان لازم برای ارسال ۵ بیت از قرائت‌گر به برچسب است.

با توجه به این که در عمل دو حالت همزمان و غیر همزمان اتفاق می‌افتد، $T_c$ طبق (۸) محاسبه می‌شود

$$T_c = \frac{T_{c1}+T_{c2}}{2} \quad (8)$$

قرائت‌گر بعد از ارسال دستور send-EPC در اسلات‌های برخورد با بیش از دو برچسب همزمان، در دریافت EPC منحصر به فرد متوجه برخورد می‌شود و شناسایی قطعی انجام نمی‌شود. در واقع این اسلات‌ها در فرایند پویش به عنوان اسلات حاوی دو برچسب همزمان شناسایی شده‌اند که نادرست می‌باشد. هرچند احتمال وجود چنین اسلات‌هایی کم است ولی صفر نیست. بنابراین مدت زمان تلف‌شده در اسلات‌های برخورد با بیش از دو برچسب همزمان، طبق (۹) محاسبه می‌شود (با توجه به شکل ۶ قرائت‌گر در هنگام دریافت EPC متوجه برخورد می‌شود)

$$T_u = 2\times(T_1+T_2)+T_{Query}+T_{RN16}+T_{Send-EPC}+T_{EPC} \quad (9)$$

در تمامی روابط بالا $T_{Query}$ مدت زمان لازم برای ارسال دستور Query Rep است که در استاندارد EPC – C1G2 یک دستور ۲۲بیتی است. $T_{ACK}$ مدت زمان لازم برای ارسال ۱۸ بیت از قرائت‌گر به برچسب است. $T_{EPC}$ مدت زمان لازم برای ارسال ۹۶ بیت از برچسب به قرائت‌گر [۱۱] و زمان‌های $T_1$ و $T_2$ زمان‌های تعریف‌شده در استاندارد EPC – C1G2 هستند که به ترتیب برابر ۷۸/۱۳ و ۳۹ میکروثانیه است [۹].

با توجه به این که برچسب‌ها به صورت تصادفی شمارنده اسلات خود را انتخاب می‌کنند، اگر $n$ برچسب به طور یکنواخت در $L$ اسلات توزیع شوند احتمال این که $x$ برچسب از $n$ برچسب در یک اسلات خاص قرار گیرند با استفاده از توزیع دوجمله‌ای نرمال طبق (۱۰) محاسبه می‌شود

$$p(x) = \binom{x}{n}(\frac{1}{L})^x(1-\frac{1}{L})^{n-x} \quad (10)$$

بنابراین برای محاسبه تعداد اسلات‌هایی که حاوی یک برچسب هستند در (۱۰) به جای $x$ عدد یک قرار داده و در $L$ ضرب می‌کنیم

$$N_s = n\times L\times(\frac{1}{L})^1\times(1-\frac{1}{L})^{n-1} = n\times(1-\frac{1}{L})^{n-1} \quad (11)$$

به همین ترتیب تعداد اسلات‌هایی که حاوی دو برچسب هستند با استفاده از (۱۲) محاسبه می‌شود

$$N_{c2} = \binom{n}{2}\times L\times(1-\frac{1}{L})^{n-2}(\frac{1}{L})^2 \quad (12)$$

بنابراین تعداد اسلات‌هایی که بیش از دو برچسب دارند طبق (۱۳) محاسبه می‌شود

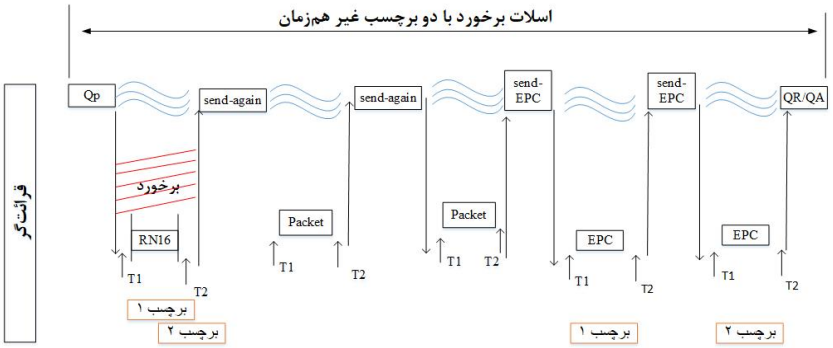
$$N_c = L'-(N_s+N_{c2}) \quad (13)$$

در (۱۰) تا (۱۲)، $n$ تعداد برچسب‌ها و $L$ طول قاب جاری قبل از فرایند پویش و در (۱۳) $L'$ طول قاب بعد از فرایند پویش است.

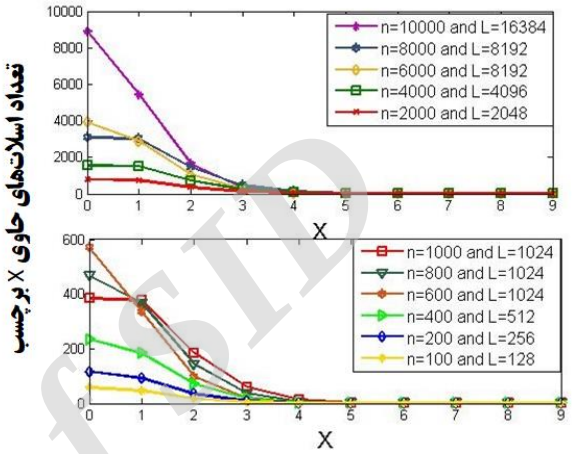
بنابراین زمان شناسایی $T_{identification}$ برای یک قاب طبق (۱۴) محاسبه می‌شود

Archive of SID

www.SID.ir




شکل ۷: فرایند شناسایی در اسلات برخورد با ارسال غیر هم‌زمان در الگوریتم ضد برخورد پیشنهادی.




شکل ۸: تعداد اسلات‌های حاوی X برچسب.

$$T_{identification} = N_s \times T_s + N_{c۲} \times T_C + N_c \times T_u \qquad (۱۴)$$

که در آن $N_s$ تعداد اسلات موفق، $T_s$ مدت زمان لازم برای شناسایی برچسب در شیار موفق، $N_{c۲}$ تعداد اسلات‌هایی با دو برچسب، $T_c$ مدت زمان لازم برای شناسایی دو برچسب، $N_c$ تعداد اسلات‌هایی با بیش از دو برچسب و $T_u$ مدت زمان تلف‌شده در اسلات‌هایی با بیش از دو برچسب است.

با توجه به شکل ۸ که با استفاده از (۱۰) تا (۱۲) تهیه شده است اگر در الگوریتم ضد برخورد پیشنهادی طول قاب بهینه انتخاب شود، تعداد اسلات‌های برخورد با بیش از دو برچسب (چندبرخوردی‌ها[1]) قابل چشم‌پوشی خواهد بود. بنابراین الگوریتم پیشنهادی با شناسایی قطعی برچسب‌ها در اسلات برخورد با دو برچسب، مشکل گرسنگی را حل کرده و همچنین زمان شناسایی کل برچسب‌ها را کاهش می‌دهد.

## ٤-٤ بررسی امنیت

الگوریتم‌های سازگار با استاندارد EPC – C۱G۲ از دو مرحله شناسایی برچسب و احراز هویت برچسب تشکیل شده‌اند. الگوریتم‌های ضد برخورد برای شناسایی برچسب‌ها استفاده می‌شوند و از این رو جنبه‌های امنیتی در این الگوریتم‌ها در نظر گرفته نمی‌شود بلکه در مرحله احراز هویت مطرح می‌شوند و پروتکل‌های احراز هویت بسیاری در این حوزه برای رفع نگرانی‌های امنیتی مطرح شده‌اند. در این مقاله یک الگوریتم ضد برخورد برای شناسایی قطعی برچسب‌ها ارائه شده که اگر در کنار پروتکل‌های احراز هویت استفاده شود، علاوه بر حل مشکلات موجود در شناسایی برچسب، نگرانی‌های امنیتی را نیز مرتفع می‌سازد.

1. Multi Collision

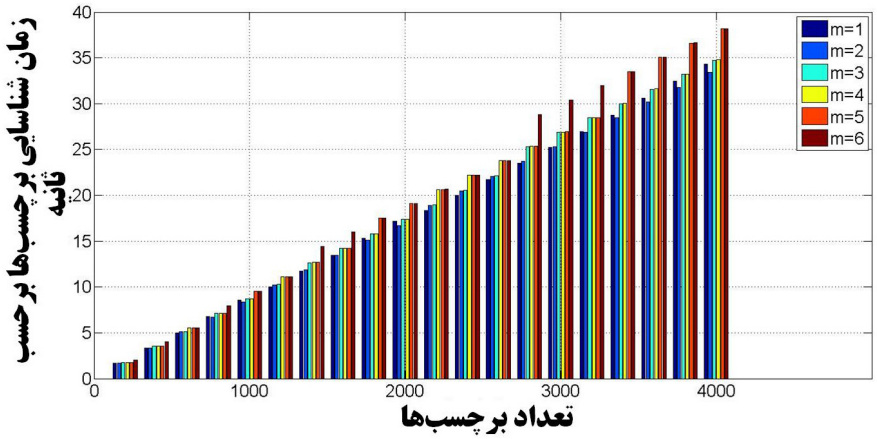
## ۵- ارزیابی نتایج

در این بخش به پیاده‌سازی و تحلیل زمان شناسایی برچسب‌ها در الگوریتم پیشنهادی می‌پردازیم و در نهایت الگوریتم خود را با الگوریتم‌های EPC – C۱G۲، BIS و ERNx مقایسه می‌کنیم. برای پیاده‌سازی الگوریتم از استاندارد EPC – C۱G۲ استفاده می‌کنیم که در آن نرخ انتقال داده بین قرائت‌گر و برچسب برابر ۶۴ Kbps و نرخ انتقال بین برچسب و قرائت‌گر برابر ۱۵٫۶۲۵ Kbps است. در [۸] بیان شده که طول قاب با $L = m.n$ محاسبه می‌شود که در آن $n$ تخمین الگوریتم از تعداد برچسب‌ها می‌باشد و بیشترین کارایی الگوریتم در $m = ۴$ به دست می‌آید. شبیه‌سازی‌های ما نشان می‌دهد با انتخاب $L = ۲n$ زمان شناسایی کل برچسب‌ها در الگوریتم پیشنهادی کمترین مقدار خواهد بود زیرا در این حالت، تعداد اسلات‌های برخورد بیش از دو برچسب خیلی کم خواهد بود و در اسلات‌های برخورد حاوی دو برچسب نیز از آنجایی که الگوریتم ضد برخورد پیشنهادی بدون نیاز به ارسال مجدد RN۱۶ توسط برچسب‌ها به شناسایی قطعی دو برچسب می‌پردازد، زمان کمتر برای شناسایی برچسب‌ها صرف می‌کند. در شکل ۹ زمان شناسایی برای ۲۰۰ تا ۴۰۰۰ برچسب توسط الگوریتم ضد برخورد پیشنهادی با $m$های مختلف نشان داده شده است. با توجه به شکل ۹ مشاهده می‌شود در حالتی که $m = ۲$ انتخاب شود زمان شناسایی کل برچسب‌ها کمترین مقدار را خواهد داشت. با افزایش تعداد برچسب‌ها، افزایش بیشتر طول قاب تأثیری در کاهش زمان شناسایی برچسب‌ها ندارد و فرایند پویش را طولانی‌تر می‌کند که این کار در نهایت باعث افزایش زمان شناسایی کل برچسب‌ها می‌شود.

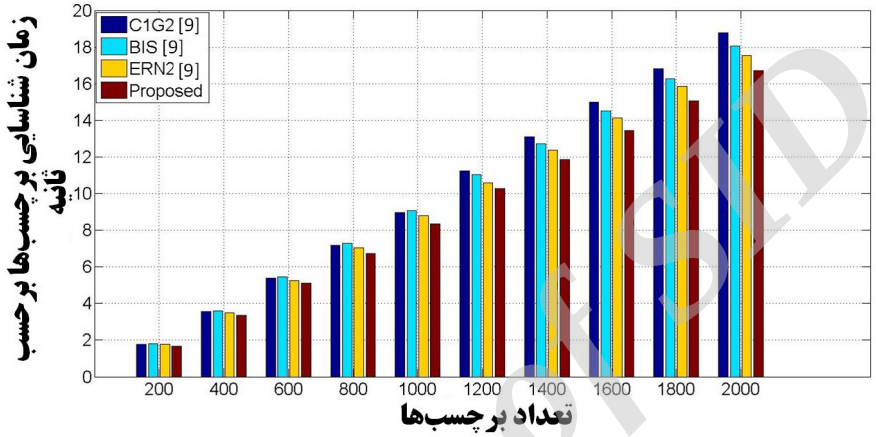
زمان شناسایی برچسب‌ها در الگوریتم ضد برخورد پیشنهادی در شکل ۱۰ نشان داده شده است. مشاهده می‌شود که زمان شناسایی برچسب‌ها در الگوریتم ضد برخورد پیشنهادی از الگوریتم‌های ضد برخورد دیگر کمتر است. با توجه به شکل ۱۰ مشاهده می‌شود که با افزایش تعداد برچسب‌ها، الگوریتم ضد برخورد پیشنهادی نسبت به الگوریتم ERN۲ بهتر عمل می‌کند چرا که با زیادشدن تعداد برچسب‌ها، تعداد اسلات‌های برخورد با ۲ برچسب افزایش می‌یابد که الگوریتم ضد برخورد پیشنهادی به شناسایی قطعی برچسب‌ها در آنها می‌پردازد. در حالی که در الگوریتم ERN۲، تعداد اسلات‌های تشخیص نادرست افزایش می‌یابد (اسلات برخورد که در فرایند پویش به اشتباه به عنوان اسلات موفق شناسایی شده است) که همین موضوع باعث افزایش زمان شناسایی کل برچسب‌ها در الگوریتم ERN۲ می‌شود. میانگین زمان شناسایی برای ۲۰۰ تا ۲۰۰۰ برچسب در الگوریتم پیشنهادی نسبت به میانگین زمان شناسایی در الگوریتم‌های EPC – C۱G۲، BIS و ERN۲ به ترتیب ۰٫۹۲، ۰٫۷۱ و ۰٫۴۲ ثانیه کمتر است. البته باید توجه کنیم که اگر ارسال غیر هم‌زمان برچسب‌ها در

Archive of SID

www.SID.ir




شکل ۹: زمان شناسایی برچسب‌ها در الگوریتم ضد برخورد پیشنهادی برای $m$ های مختلف.




شکل ۱۰: زمان شناسایی برچسب‌ها در الگوریتم‌های ضد برخورد مختلف.

الگوریتم ضد برخورد پیشنهادی در نظر گرفته نمی‌شد زمان شناسایی با توجه به (۷) به صورت معناداری کاهش می‌یافت. پس یکی دیگر از مزیت‌های الگوریتم ضد برخورد پیشنهادی، قابلیت پیاده‌سازی در محیط‌های مختلف است فارغ از این که برچسب‌ها، اطلاعات خود را به صورت هم‌زمان یا غیر هم‌زمان ارسال کنند.

## ٦- نتیجه‌گیری

در این مقاله، یک الگوریتم ضد برخورد جدید مطرح شد که مانند الگوریتم BIS شامل دو فرایند پویش و شناسایی است ولی برخلاف الگوریتم BIS در فرایند پویش علاوه بر شناسایی اسلات خالی به شناسایی اسلات‌های برخورد با دو برچسب غیر هم‌زمان نیز می‌پردازد. همچنین الگوریتم پیشنهادی توانایی شناسایی قطعی دو برچسب در اسلات برخورد با دو برچسب را دارا است. قرائت‌گر در اسلات برخورد موقعیت اولین بیت با مقدارهای متفاوت در RN۱۶ (عدد تصادفی ۱۶بیتی تولیدشده توسط برچسب‌ها) را به دست می‌آورد و با استفاده از آن و با ارسال دستورات send-EPC و send-again اقدام به شناسایی قطعی برچسب‌ها می‌کند. بنابراین در الگوریتم ضد برخورد پیشنهادی با انتخاب طول قاب بهینه مشکل گرسنگی برچسب‌ها وجود نخواهد داشت. متوسط زمان شناسایی ۲۰۰ تا۲۰۰۰ برچسب در الگوریتم پیشنهادی نسبت به الگوریتم‌های ضد برخورد C۱G۲، BIS و ERN۲ کمتر است.

## مراجع

[1] J. Park, M. Y. Chung, and T. J. Lee, "Identification of RFID tags in framed-slotted ALOHA with robust estimation and binary selection," *IEEE Communications Letters*, vol. 11, no. 5, pp. 452-454, May 2007.

[2] L. Zhang, W. Xiang, and X. Tang, "An adaptive anti-collision protocol for large-scale RFID tag identification," *IEEE Wireless Communications Letters*, vol. 3, no. 6, pp. 601-604, Dec. 2014.

[3] L. Qiu, Z. Huang, S. Zhang, and W. Wang, "Location-aware anti-collision protocol for energy efficient passive RFID system," in *Proc. Int. Conf. on Indoor Positioning and Indoor Navigation, IPIN'14*, 6 pp., Busan, South Korea, 27-30 Oct. 2014.

[4] K. Finkenzeller, *RFID Handbook, Fundamentals and Applications in Contactless Smart Cards and Identification*, John Wiley and Sons Ltd., pp. 206-219, 2003.

[5] J. Capetanakis, "Tree algorithms for packet broadcast channels," *IEEE Trans. on Information Theory*, vol. 25, no. 5, pp. 505-515, Sept. 1979.

[6] N. Bagheri, P. Alenaby, and M. Safkhani, "A new anti-collision protocol based on information of collided tags in RFID systems," *International J. of Communication Systems*, vol. 30, no. 3, e2975, Feb. 2017.

[7] D. G. Zhang, G. Li, Z. H. Pan, and Y. P. Liang, "A new anti-algorithm for RFID Tag," *International J. of Communication Systems*, vol. 27, no. 11, pp. 3312-3322, Apr. 2014.

[8] D. Liu, Z. W. Zhongxiang, J. Tan, H. Min, and J. Wang, "ALOHA algorithm considering the slot duration difference in RFID system," in *Proc. IEEE Int. Conf. on RFID,* pp. 56-63, Orlando, FL, USA, 27-28 Apr. 2009.

[9] F. Baloch and R. Pendse, "A new anti-collision protocol for RFID networks," in *Proc. Wireless Telecommunications Symp., WTS'14*, 5 pp., Washington, DC, USA, 9-11 Apr. 2014.

[10] J. Xiaolin, F. Quanyuan, and M. Chengzhen, "An efficient anti-collision protocol for RFID tag identification," *IEEE Communications Letters*, vol. 14, no. 11, pp. 1014-1016, Nov. 2010.

[11] EPCglobal Inc., EPC Radio-Frequency Identity Protocols Class-1 Generation- 2 UHF RFID Protocol for Communications at 860 MHz - 960MHz Version 2.0.0, Nov. 2013.

Archive of SID

www.SID.ir

**امیر عباسیان** فارغ‌التحصیل مقطع کارشناسی ارشد از دانشگاه تربیت دبیر شهیدرجایی تهران می‌باشد. نام‌برده تحصیلات کارشناسی خود را در رشته مهندسی کامپیوتر- نرم‌افزار در دانشگاه آزاد اسلامی واحد شبستر در سال ۸۶ به اتمام رسانده است. وی در سال ۹۳ در مقطع کارشناسی ارشد در رشته مهندسی کامپیوتر- نرم‌افزار پذیرفته شد و در زمستان ۹۵ نیز فارغ‌التحصیل گردید. زمینه‌های تحقیقاتی ایشان عبارتند از امنیت سامانه‌های RFID، امنیت شبکه و الگوریتم‌های ضد برخورد در سامانه‌های RFID.

**معصومه صفخانی** استادیار دانشکده مهندسی کامپیوتر در دانشگاه تربیت دبیر شهیدرجایی و نویسنده بیشتر از سی مقاله در زمینه تحلیل و طراحی پروتکل‌های امنیتی و طرح‌های رمزنگاری است. در حال حاضر تمرکز ایشان روی امنیت RFID، امنیت اینترنت اشیا و امنیت طرح‌های رمزگذاری قابل جستجو است. اطلاعات بیشتر در نشانی http://www.srttu.edu/english-cv-dr-safkhani/ قرار دارد.

Archive of SID

www.SID.ir