

فهرست اشکال

18.....	شکل 1- نمایش ARQ در UW-HARQ
20.....	شکل 2- توان عملیاتی انتها به انتها با طول های بسته متغیر
21.....	شکل 3- توان عملیاتی انتها به انتها، با شمارش متغیر هاپ
22.....	شکل 4- سربار با سایز بلوک متغیر
23.....	شکل 5- سه نوع تصادم در SDRT
26.....	شکل 6- فرستنده های چندگانه همزمان بدون برخورد
26.....	شکل 7- تخصیص های CAID
27.....	شکل 8- هماهنگی چند هاپی
31.....	شکل 9- معماری کلی پیاده سازی CCRDT

- 32.....[شکل 10- تنظیمات سخت افزار عادی در یک آزمایشگاه](#)
- 33.....[شکل 11- توان عملیاتی انتها به انتها با طول بسته متغیر](#)
- 34.....[شکل 12- توان عملیاتی انتها به انتها با نرخ مودم متغیر](#)
- 35.....[شکل 13- توان عملیاتی انتها به انتها با شما رش هاپ متغیر](#)
- 36.....[شکل 14- توان عملیاتی انتها به انتها با سائز پنجره متغیر](#)
- 37.....[شکل 15- سربار، با شمارش هاپ متغیر](#)
- 39.....[شکل 16- تشریح مدل شبکه](#)

فصل 1- بررسی کلی

چکیده:

کانالهای مخابراتی صوتی زیرآب به دلیل تغییر پذیری محیط دارای پاشیدگی توام زمانی و فرکانسی هستند که در نتیجه ارسال داده در آنها با چالش روبرو میشود. به دلیل اهمیت و کاربردهای مخابرات زیرآب، بررسی محیط انتشار و قوانین حاکم بر انتشار صوت در زیرآب بسیار مهم میباشد. ر کانالهای مخابراتی زیرآب به دلیل جذب شدید امواج الکترومغناطیسی و نوری از سیگنالهای صوتی به منظور انتقال داده استفاده میگردد. تاخیر زیاد، پهنای باند کم، فرکانس گزینی شدید، چند مسیری و تغییر پذیری شرایط محیط از مهمترین چالشهایی است که در سیستمهای مخابراتی زیرآب وجود دارد. در این سمینار انواع چالش ها از جمله انتقال داده در شبکه های زیر آب که یکی از مهمترین مسائل است، به صورت کامل به تفصیل با بررسی بیش از 47 منبع صورت گرفته است.

کلیدواژه: شبکه های آکوستیک زیر آبی - رویکردهای مبتنی بر **ARQ** هیبرید - رادکست قابل اطمینان، برای شبکه های آکوستیک زیرآبی - برادکست قابل اطمینان، برای شبکه های آکوستیک زیرآبی

1,1- مقدمه کلی:

شبکه های آکوستیک زیر آبی (**UAN**ها) به دلیل کاربردهای گسترده که شامل نظارت در محیط زیر آب و شنود تاکتیکی می شوند [1-5]، تبدیل به یک زمینه تحقیقاتی مهم شده اند. به دلیل تفاوت های موجود میان **UAN** و شبکه های حسگر وایرلس زمینی (**TWSN**ها) و چالش های اعمالی توسط محیط خشن زیر آب، تلاش های چشمگیر تحقیقاتی، به هر لایه **UAN**ها اختصاص داده شده است [6-13]

در **UAN**ها به صورت متفاوت از **TWSN**ها، یک کانال آکوستیک، به جای کانال رادیویی، اغلب برای انتقال سیگنال در آب استفاده می شود. این پدیده، چندین چالش را برای برقراری ارتباطات در نیو **UAN**ها اعمال می کند. اول، تاخیر انتشار داده به دلیل سرعت انتشار کم سیگنال های آکوستیک،

طولانی است (حدود 1500 m/s). دوم، کانال های UAN، نرخ داده بسیار کمی دارند و دلیل آن، جذب، پدیده چند مسیره بودن و محو شدگی است. سوم این که UAN ها اغلب از احتمال خطای بالا به دلیل استفاده از کانال های آکوستیک زیرآبی که احتمال بروز خطا در آن ها بیشتر است رنج می برند. علاوه بر چالش های اعمال شده توسط ارتباطات آکوستیک، UAN ها، چالش های دیگری را روبه روی خود می بینند [14-18]. محیط خشن زیرآبی باعث ایجاد دینامیک های خارق العاده در UAN ها از لحاظ تولید شبکه و کیفیت کانال می شود [19]. مسائلی همانند اتصال شبکه متناوب، باعث ایجاد چالش هایی بزرگ در خدمات UAN می شود همانند انتقال داده قابل اطمینان [20، 21]. علاوه بر آن، UAN ها همیشه ماهیتی محدود از لحاظ انرژی داشتند [22]. شبکه های UAN در قسمت پایینی اقیانوس باید توان خود را از طریق باتری دریافت کنند و به صورت دوره ای، شارژ مجدد شوند [23]. حتی گره هایی که بالاتر از سطح آب قرار دارند اغلب برای ارائه کارآیی خود نیاز به انرژی خورشیدی دارند. قید انرژی به این معنا است که پروتوکل ها و خدمات، در UAN باید از لحاظ انرژی، بهره ور باشند [24، 25]. علاوه بر آن، مودم های آکوستیک زیرآبی، نیمه دوپلکس بوده و فقط می توانند در حالت فرستنده یا دریافت کننده قرار بگیرند [26، 27].

برخلاف چالش های مذکور، انتقال داده بهره ور و قابل اطمینان، همیشه یک خصیصه مطلوب، برای UAN ها بوده است [23-28]. کاربردهایی همانند فرستادن یک فایل داده مهم از یک گره منبع به یک گره سینک، از چندین هاپ یا انتقال یک دستور حیاتی، از یک ایستگاه کنترل به چندین مود UAN، امری متعارف بوده و توسط وظایف مختلف، ملزم می شود [32-34]. در این بررسی، به مساله حیاتی و چالش برانگیز عنوان شده می پردازیم. با این وجود، ویژگی های منحصر به فرد آکوستیک زیرآبی باعث می شوند روش های متعارف، برای انتقال داده قابل اطمینان، برای UAN ها نیز نامطلوب شوند:

● پروتکل های انتها به انتها موجود، اغلب به خوبی جواب نمی دهند. در این پروتکل ها، دو گره پایانی (منبع و مقصد) که می توانند چندین هاپ از هم فاصله داشته باشند به منظور تضمین قابلیت انتها به انتها، با یکدیگر همکاری کنند [28,29,35]. در صورتی که مقصد نتواند یک بسته را به صورت درست دریافت کند، از منبع می خواهد تا انتقال مجدد را انجام دهد. تمامی گره های میانی، بین این دو گره انتهایی، هیچ نیازی نیست که ملزومات قابلیت اطمینان را لحاظ کنند (TCP در اینترنت، دقیقاً این نوع رویکرد انتها به انتها را اتخاذ می کند). با این وجود در UAN ها به دلیل نرخ خطای بالا و تاخیر انتشار طولانی کانال های آکوستیک زیرآبی، چنین رویکردهای انتها به انتهایی، باعث بروز انتقال مجدد فراوان و تاخیر طولانی انتها به انتها می شود که باعث می شود پدیده عنوان شده، بهره ور نباشد.

● رویکردهای هاپ به هاپ موجود، اغلب، بهره وری کافی را ندارند. در این رویکردها، قابلیت اطمینان در هر هاپ، از منبع به مقصد، تضمین می شود. در مقایسه با پروتکل های انتها به انتها، رویکردهای هاپ به هاپ، واکنش پذیری بیشتری به شرایط پویای شبکه داشته و انتظار می رود که بهره ور تر باشند. با این وجود رویکردهای هاپ به هاپ مرسوم انتقال مجدد، در صورت بروز خطا، اغلب، بسیار ناکارآمد و زمان بر هستند. به منظور ارائه چندین مثال ساده به مبحث زیر می پردازیم. در صورتی که دو گره، فاصله 500 متری داشته باشند، تاخیر انتشار میان آن ها حدوداً برابر با ثانیه خواهد بود. این تاخیر در انتشار به علاوه تاخیر در انتها بسته و تاخیر در تبدیل حالت مودم، منجر به هزینه زیادی حتی در انتقال مجدد در یک هاپ می شوند.

برپایه مباحثه ای که در بالا عنوان شده است می توانیم ببینیم که رویکردهای هاپ به هاپ، اغلب به نسبت رویکردهای انتها به انتها، بهره وری بالاتری دارند اما حتی با استفاده از رویکردهای هاپ به هاپ،

برخی از طرح ها را باید برای مقابله با از دست رفتن بسته و کاهش احتمال انتقال مجدد، اتخاذ نمود. به این منظور، **ARQ (HARQ)** هیبرید، پیشنهاد شد [36] که کدنویسی تصحیح خطای فوروارد (**FEC**) و درخواست انتقال اتوماتیک (**ARQ**)، ترکیب می کند. با کدنویسی **FEC**، **HARQ**، افزونگی داده را اضافه کرده و بنابراین به صورت کارآمد، از دست رفتن بسته را همدل نموده و باعث کاهش انتقال مجدد می شود. با استفاده از **ARQ**، **HARQ**، بهره وری انتقال مجدد داده را تسهیل می کند. هم چنین در **UAN** های چند هاپی، یک زمان بندی موثر یا مکانیزم هماهنگی میان چندین هاپ، باید به منظور از بین بردن تصادم یا بازیابی یک زوج ارتباطی از تصادف، پیاده سازی شود. علاوه بر آن، انتظار می رود انتقال داده قابل اطمینان و کارآمد، با سناریوهای یونیکست، مولتی کست و برادکست، به خوبی کار کند.

1,2- مطالب سمینار:

با الهام گیری از مباحثه بالا در این سمینار، به مسئله انتقال داده بهره ور و قابل اطمینان، از سه دیدگاه مختلف می پردازیم.

1,2,1- رویکردهای مبتنی بر **ARQ** هیبرید، برای شبکه های آکوستیک زیرآبی:

اول، طرح های مبتنی بر **ARQ** هیبرید (**HARQ**) [36]، تمرکز می کنیم که برای رویارویی با مسئله از دست رفتن بسته در **UNA** ها، بهره ور ظاهر شده و باعث کاهش احتمال انتقال مجدد روش های انتها به انتها و هاپ به هاپ می شود. ما بر **UAN** یونیکست، یا توپولوژی رشته چند هاپی، تمرکز می کنیم که نقش حیاتی ای در بازه گسترده ای از کاربردها ایفا نموده است (شامل دریافت جمع آوری داده و انتقال دستور). در این جا، سه طرح مبتنی بر **HARQ** را پیشنهاد نموده ایم. این که این طرح ها چه چیزی را به صورت مشترک دارند این است که همه آن ها از کدنویسی **FEC** و **ARQ** استفاده می

کنند که خصیصه الزامی HARQ است. با این وجود تفاوت فاحشی در طرح های کدنویسی مربوطه خود و روش های هماهنگی چند هاپی دارند.

ARQ هیبرید زیرآبی (UW-HARQ) [37]، یک روش ARQ ساده انتها به انتها است که هدف آن، ارائه گره های UAN، با قدرت کمتر (همانند برکلی موتس) می باشد [38,39]. کدنویسی خطی بایناری رندوم (RBLC)، به دلیل پیچیدگی کدگذاری و کدگشایی پایینش، مورد استفاده قرار می گیرد. یک طرح برآورد نسبت کدنویسی بهینه پیشنهاد می شود که هم خصوصیات RBLC و هم نرخ خطای بسته (PER) را لحاظ می کند [40]. این رویکرد، در گره های UAN حقیقی، پیاده سازی شده است. آزمایش اولیه آزمایشگاهی، به منظور ارزیابی اثربخشی رویکردهای مبتنی بر HARQ انتها به انتها انجام شد.

نتایج آزمایشی UW-HARQ، نقاط برتری آن را به نسبت روش های مبتنی بر تکرار انتخابی انتها به انتهای مرسوم، بدون کدنویسی FEC، از لحاظ توان عملیاتی به دست آمده، نشان داده است. با این وجود، توان عملیاتی به دست آمده هنوز هم بسیار پایین است. دلیل اصلی، این است که در UAN چند هاپی، از دست رفتن یک بسته در یک هاپ میانی، منجر به انتقال انتها به انتها می شود که در طول چند هاپ، به وقوع می پیوندد. هزینه در این عمل، بالا است؛ به خصوص هنگامی که یک لینک تنگنا با PER بسیار بالا وجود داشته باشد

برای حل این مسئله، CCRDT [41] را با یک پروتکل انتقال داده قابل اطمینان، با هماهنگی چند هاپی مبتنی بر کدنویسی، برای UAN ها ریال با پلت فرم های قدرتمندتر (همانند Gumstic [42]) پیشنهاد می کنیم. CCRDT، کدنویسی خطی رندوم GF(256) و تکرار انتخابی را برای تضمین قابلیت اطمینان، به ازای هر هاپ، ترکیب می کند. یک طرح هماهنگی چند هاپی، به منظور ارائه قابلیت خط لوله سازی انتقال و حذف تصادم بسته، پیشنهاد شده است (پایپ لاین کردن). CCRDT را بر روی یک

بستر آزمایشی، پیاده سازی کرده و آزمایش های گسترده ای را به منظور ارزیابی عملکردش انجام دادیم. نتایج مقدماتی نشان می دهد که CCRDT به توان عملیاتی انتها به انتهای بالاتری به نسبت سایر پروتکل های انتقال قابل اطمینان موجود که در سیستم های حقیقی، پیاده سازی شده اند، می رسد.

CCRDT می تواند در UNAهایی با توپولوژی رشته همگون، عملکرد بهینه داشته باشد و دلیل آن، استفاده از طرح هماهنگی چند هاپی مبتنی بر خط لوله است. برای این که طرح ما به خوبی با UANهای ناهمگون که در آن ها PERها با چندین هاپ، اغلب تفاوت های چشمگیری دارند، خوب کار کند، یک انتقال داده قابل اطمینان چند هاپی انطباقی منطبق با کد (FOCAR) Fountain را پیشنهاد می کنیم [43]. FOCAR، کدنویسی فواره ای و انتقال مجدد در صورت بروز خطا را به منظور بهبود قابلیت اطمینان و بهره وری انتقال داده، به ازای هر هاپ، یکپارچه می کند. کدنویسی فواره ای، به دلیل قابلیت تصحیح خطای قوی و خصوصیت بدون نرخ بودنش، انتخاب شده است. علاوه بر آن، FOCAR، رویکرد بهینه سازی مشترک را برای سناریو های شبکه چند هاپی به کار می بندد. یک چارچوب بهینه سازی، برای بهینه سازی عملکرد سیستم، در چندین هاپ، پیشنهاد شده است.

2,2-1- برادکست قابل اطمینان، برای شبکه های آکوستیک زیرآبی

رویکردهای مبتنی بر HARQ مذکور، برای UANهای یونیکست، با توپولوژی رشته ای طراحی شده اند. برای پرداختن به طراحی برادکست بهره ور و قابل اطمینان، یک طرح برادکست دو فازی (TBS) [44] را برای UANها پیشنهاد می کنیم. TBS شامل دو فاز می شود: فاز انتشار سریع و فاز بازیابی داده. این طرح، نیاز به اطلاعات مربوط به همسایه یا توپولوژی ندارد. فاز انتشار سریع، سربار گذاری فرصت طلبانه و کدنویسی شبکه را به منظور بهبود بخشی بهره وری برادکست، ترکیب می کند. فاز

بازیابی داده، وظیفه تضمین قابلیت اطمینان و کمینه سازی احتمال تداخل را دارد. نتایج شبیه سازی نشان می دهند که TBS به زمان تکمیل برادکست کمتر و بهره وری انرژی قابل قیاس می رسد.

1,2,3- کنترل دست یابی به کانال قابل اطمینان، برای شبکه های آکوستیک زیرآبی

CCRDT و FOCAR از هماهنگی چند هاپی، به منظور حذف تصادم، برای UAN های با توپولوژی رشته استفاده می کنند. برای این که طرح ما به خوبی با توپولوژی های دلخواه، جواب دهد، SASHA [45] را پیشنهاد می کنیم که خود، یک دست یابی مبتنی بر دست دادن اسلات دار و دست دادن ARQ انتخابی است. SASHA به صورت متفاوت از CCRDT و FOCAR، اجازه رخداد تصادم را می دهد اما توانایی بازیابی آن را دارد. این روش، الزامی ترین تکنیک ها در طراحی پروتکل های UAN MAC مبتنی بر هماهنگی را می پذیرد (شامل اسلات بندی زمانی، دست دادن، اجتناب از تصادم و ARQ انتخابی). از آن مهمتر، SASHA را در گره های حقیقی UAN، پیاده سازی کرده و یک آزمایش دریایی را در اقیانوس آتلانتیک، به منظور ارزیابی عملکردش انجام دادیم. با توجه به داده های آزمایشی، توانستیم نحوه اثرگذاری تکنیک های مذکور، بر عملکرد SASHA را مطالعه کنیم. هم چنین رفتار هاپ و انتهای SASHA را تحلیل نمودیم. با توجه به یافته های ما، برخی مسائل یافت شده و رهنمودهای طراحی متناظر با آن ها نیز پدیدار شد.

1,3- سرفصل سمینار

سمینار به صورت زیر سازماندهی شده است: در فصل دو، برخی از مطالعات مربوطه جدید درباره انتقال داده بهره ور و قابل اطمینان در UNA را ارائه می دهیم. در فصل 3، به مباحثه درباره رویکردهای مبتنی بر HARQ، برای UNA یونیکست، با توپولوژی رشته ای می پردازیم. UW-HARQ، CCRDT، FOCAR، به صورت مفصل از لحاظ پروتکل، پیاده سازی و آزمایش، توصیف خواهند شد.

فصل 2- پیش زمینه

در این فصل، مطالعات مربوطه و دانش پیش زمینه را در زمینه انتقال داده بهره ور و قابل اطمینان، برای شبکه های حسگر وایرس زمینی (TWSN ها) و شبکه آکوستیک زیرزمینی (UAN ها) ارائه می دهیم

2,1- انتقال داده قابل اطمینان، برای شبکه های حسگر وایرلس زمینی
پروتکل های انتقال داده قابل اطمینان، برای TWSN ها، مورد مطالعه گسترده ای قرار گرفته اند [28,29,35,46]. در [28]، پروتکلی با نام ESRT، مورد بررسی قرار گرفته است. این پروتکل، قابلیت اطمینان سیستم را از طریق کنترل کردن نرخ فرستادن منابع، برای اجتناب از تصادم را بهبود می بخشد. در [35]، یک طرح ACK بلاک بدون پنجره، به منظور بهبود بخشی استفاده از یک کانال، با یک مکانیزم کنترل رقابت متمایز، برای کاهش تاخیر انتها به انتها، مورد استفاده قرار گرفته است. در [46]، ون و همکاران، پروتکل PSFQ را پیشنهاد کردند که از یک رویکرد مشابه ARQ هاپ به هاپ، با نرخ داده سریع، برای انتقال مجدد استفاده می کند. در [29]، محققان، یک طرح انتقال داده قابل اطمینان، با طراحی ترکیبی سطوح انتقال و مک، با مکانیزم ARQ را پیشنهاد دادند.

نویسندگان در [47]، یک پروتکل انتقال داده قابل اطمینان کنترل شده نرخ را برای TWSN پیشنهاد کردند. RCRT، وظیفه کنترل ازدحام را به گره سینک تخصیص می دهد که دانش بهتری درباره شرایط شبکه داشته و بنابراین، توانایی کنترل جمعی بهتر ترافیک را دارد. یک الگوریتم تخصیص نرخ انعطاف پذیر نیست، پیاده سازی شده است. RCRT هم چنین از چندین کاربرد همروند، پشتیبانی کرده و در برابر ویژگی های پویای شبکه، مقاوم است. در [48]، نویسندگان، الگوریتم های کنترل جریان، مبتنی بر نرخ را بررسی نمودند که البته اثبات شده است این الگوریتم ها به نسبت الگوریتم های مبتنی بر پنجره، اثربخشی بیشتری دارند. با اندازه گیری مستقیم تداخل های درون – مسیری در هر هاپ، کران بالای نرخ را می توان تعیین کرد. هم چنین می توان نشان داد که اندازه گیری های این گونه می توانند

به عنوان پایه ای برای پروتکل های انتقال داده قابل اطمینان، در TWSN ها، مورد استفاده قرار گرفته و انعطاف پذیری و مقیاس پذیری بالایی را فراهم میاورند. محققان در [49]، یک مطالعه جامع را درباره رفتار TCP، در شبکه های آدهاک، انجام دادند. چندین مسئله در چارچوب طراحی TCP که برای انتقال داده قابل اطمینان، در شبکه های آدهاک، مناسب نیستند، اشاره شدند. علاوه بر آن، نویسندگان، یک ATP پروتکل لایه انتقال قابل اطمینان جدید، برای تسهیل اطمینان پذیری و بهره بری انتقال داده، در شبکه های آدهاک، پیشنهاد نمودند. از طریق شبیه سازی نقاط برتر ATP به نسبت TCP و TCP-ELFN نشان داده شدند

به دلیل تمایزهای چشمگیر، میان UNA ها و TWSN ها، پروتکل های مذکور، برای محیط های زیرآبی، مناسب نیستند. طرح های HARQ که FEC و ARQ را ترکیب می کنند، برای TWSN ها در [52]-[50]، مورد مطالعه گسترده ای قرار گرفته اند. در [53]، نویسندگان، مسئله برنامه نویسی در بستر هوا را در TWSN های مبتنی بر کدهای بدون نرخ، مطالعه نمودند. دو رویکرد در این جا ارائه شده است: رویکرد دیلاژ بدون نرخ و دیلاژ بدون ACK، پیاده سازی شدند. از طریق انجام آزمایشات، اثبات شد که رویکردهای پیشنهادی، عملکرد بهتری به نسبت روش عادی دیلاژ دارند و می توانند به مقیاس پذیری بهتری برسند. برخلاف نقاط قوت عنوان شده، طرح های مذکور، اغلب تاخیر انتشار را لحاظ نمی کنند که در محیط زیرآبی، این عدد، عدد بالایی است. بنابراین نمی توان آن ها را به صورت مستقیم در UAN ها مورد استفاده قرار داد. هم چنین رویکردهای مختلفی برای برادکست در شبکه های آدهاک پیشنهاد شده اند که روش سیل آسای کورکورانه، ساده ترین مورد است [54،55]. روش سیل آسای کورکورانه به بهترین قابلیت اطمینان، با هزینه طوفان برادکست می رسد. به منظور تسهیل مسئله طوفان برادکست، تلاش های گسترده تحقیقاتی ای به انتخاب یک مجموعه بهینه از گره های برادکست مجدد اختصاص داده شده است. مکانیزم های پیشنهادی را می توان به صورت جامع، به روش های

مبتنی بر احتمال، روش های مبتنی بر مساحت، روش های مبتنی بر دانش از همسایه و روش های مبتنی بر دسته، طبقه بندی کرد [57]. در روش های مبتنی بر احتمال، یک گره یا با یک استاتیک از پیش تعیین شده، برادکست مجدد را انجام می دهد یا با یک احتمال انطباقی که ارتباط معکوس، با تعداد همسایه ها را دارد (برای مثال [58]). احتمال استاتیک، اثربخش نیست. به دلیل دشواری تصمیم گیری درباره احتمال برادکست مجدد، با توجه به UAN هایی که پویایی بالایی دارند، احتمال انطباقی نیز برای UAN ها به دلیل دشواری به دست آوردن و حفظ اطلاعات، درباره همسایه ها، بهره وری لازم را ندارد و به جای مبتنی بر دانش از همسایه (همانند هرس قالب [59] و رله چند نقطه ای [60])، اثبات شده است که بهره وری بالایی برای شبکه های ادهاک دارند. اغلب، نیاز به اطلاعات یک هاپ یا دو هاپ هستند که از طریق تبادل اطلاعات راهنما به دست می آید [57]

2,2- انتقال داده قابل اطمینان، برای شبکه های آکوستیک زیرآبی

اخیرا انتقال داده قابل اطمینان بهره ور، برای UAN ها توجهات تحقیقاتی چشمگیری را به خود جلب کرده است. در [61]، مقاله موجود، سه نوع از پروتکل های توقف و انتظار را برای UAN ها، مورد بررسی قرار داده و نشان می دهد که عملکرد پروتکل های پایه توقف و انتظار را می توان از طریق انتقال بسته ها به صورت گروهی و همراه با تصدیق انتخابی، بهبود بخشید. سائز بهینه بسته نیز مورد تحلیل قرار گرفته است.

رویکرد HARQ توسط انتقال داده قابل اطمینان، در UNA ها اتخاذ شده و اثبات شده است که می تواند روش اثربخشی برای مقابله با از دست رفتن بسته باشد که این پدیده با توجه به کانال های آکوستیک زیرآبی، که در معرض خطای بالا و محیط های خشن زیر آبی هستند، با تکرار بالا اتفاق می افتد. در [36]، یک طرح مبتنی بر HARQ هاپ به هاپ، با نام SDRT، برای UAN ها پیشنهاد شده است. در SDRT، فرستنده ابتدا بسته ها را به چندین بلاک تقسیم می کند، سپس هر بلاک را

کدگذاری کرده و بسته ها را با دقت در یک پنجره ارسال می کند. انتظار می رود که بسته ها (شامل بسته های داده اولیه و بسته های کدگذاری شده) در پنجره ، برای بازیابی بسته های داده ابتدایی، دریافت کننده برخلاف از دست رفتن احتمالی بسته ها، کافی باشند. پس از آن، نرخ انتقال داده، کاهش پیدا می کند و برای پاسخ از دریافت کننده، منتظر می مانیم. در صورتی که فرستنده نتواند یک ACK از دریافت کننده به دست بیاورد، یک بسته کدگذاری شده را برای دریافت کننده، پس از گذر زمانی مقرر ارسال می کند. این فرآیند تا زمانی که یک ACK در فرستنده دریافت شود ادامه پیدا می کند

کد فواره به دلیل پیچیدگی کدگذاری کم و خصلت بدون نرخ بودن، برای انتقال داده قابل اطمینان در UAN ها مورد استفاده قرار گرفته است. در [62]، کد فواره ، برای تسهیل قابلیت اطمینان و بهره وری برادکست، در UNA ها استفاده شده است. یک مدل ریاضی، درباره عملکرد کد فواره، در صورت اعمال بر برادکست ، در این مقاله ارائه شده است. مقیاس های عملکرد همانند تاخیر انتقال، مصرف توان و قابلیت اطمینان، تحلیل شدند، روابط متقابل مربوطه در میان این مقیاس ها، به منظور ارائه رهنمود، برای طراحی پروتکل، سنجش شدند (به خصوص دلالت های توان انتقال، در پوشش برادکست و تاخیر انتها به انتها، مورد تاکید قرار گرفته اند). در [63]، یک پروتکل ذخیره سازی و انتقال داده قابل اطمینان، مبتنی بر کد فواره، برای UAN ها پیشنهاد شده است. به منظور بهبود بخشی قابلیت اطمینان انتقال داده و ذخیره سازی آن، در یک UAN با معماری مبتنی بر منطقه، یک مکانیزم کدنویسی فواره به هم پیوسته، مورد استفاده قرار گرفته است. در حالی که کد فواره حسگر منطقه، برای ارائه توانایی انتقال داده قابل اطمینان، استفاده شده است ، لایه دومی از کد فواره ، برای تضمین قابلیت اطمینان ذخیره سازی یکنواخت نیز ارائه شده است. یک طرح چند ACK ای در این مقاله مورد استفاده قرار گرفته است.

کدنویسی شبکه خطی رندوم نیز در UAN ها اعمال شده است. محققان در [64]، از کدنویسی شبکه، برای تسهیل بهره‌وری برادکست، در شبکه‌های تقسیم زمان دوتایی استفاده کردند. اثربخشی این رویکرد، تحلیل شده و تامین دلیل شد. در [64]، یک طرح کدنویسی مشترک کانال-شبکه، برای برقراری ارتباط شبکه‌چندهایی - چند مسیری قابل اطمینان، پیشنهاد شد. کدنویسی کانال LDPC و کدنویسی شبکه خطی رندوم، به منظور استفاده از تنوع فضایی و افزونگی کدنویسی، ترکیب شدند. با استفاده از این روش، توان عملیاتی شبکه و قابلیت اطمینان آن به طرز چشمگیری بهبود یافتند. نویسندگان در [66]، از یک رویکرد کدنویسی شبکه پایه در گره‌های UAN جهان حقیقی استفاده نموده و آزمایش‌های آزمایشگاهی مقدماتی را به منظور ارزیابی عملکرد آن انجام دادند. با انجام این کار، نقطه برتری کدنویسی شبکه که بر کاربردهای UAN، در جهان حقیقی، اعمال می‌شود، نشان داده شد.

فصل 3- رویکردهای مبتنی بر ARQ هیبرید، برای شبکه های آکوستیک زیرآبی

در این فصل به انتقال داده بهره ور و قابل اطمینان، در UAN، با استفاده از رویکردهای مبتنی بر ARQ هیبرید (HARQ) می پردازیم. HARQ، کدنویسی اصلاح خطای فوروارد (FEC) و درخواست تکرار اتوماتیک (ARQ) را برای مقابله با از دست دادن بسته، ترکیب کرده و بهره وری انتقال مجدد را تسهیل می کند. به خصوص طرح ARQ هیبرید زیرآبی انتها به انتها (UW-HARQ) را با عنوان انتقال داده قابل اطمینان هماهنگ شده چند هاپی، مبتنی بر کدنویسی (CCRD) و انتقال داده قابل اطمینان چند هاپی، مبتنی بر کد فواره ای (FOCAR) پیشنهاد می کنیم. برخلاف این که این روشها، همگی از کدنویسی FEC و ARQ استفاده می کنند، از لحاظ طرح های کدنویسی و هماهنگی چندهاپی، تفاوت های فاحشی با هم دارند (نه تنها به دلیل پلت فرم های UAN مدنظر (که بر آن ها اعمال می شوند)، بلکه به دلیل سناریوهای کاربردی متفاوتی که بر اساس آن، هدف گذاری شده اند). UW-HARQ، یک رویکرد ساده انتها به انتها است که برای پلت فرم های UAN، با قدرت کمتر، طراحی شده است. کدنویسی خطی بایناری رندوم، به منظور کمینه سازی سربار محاسباتی و مصرف انرژی به کار بسته می شود. به عنوان یک رویکرد هاپ به هاپ، CCRD، بری UAN هایی با توپولوژی رشته ای چند هاپی همگون که قابلیت محاسباتی بالاتر و ظرفیت توان بالایی دارند، طراحی شده اند، کدنویسی خطی رندوم (GF(256)، قابلیت بازیابی داده قوی تر و یک طرح هماهنگی چندهاپی را ارائه می دهد که نه تنها می تواند تصادم را حذف کند بلکه توانایی خط لوله کردن انتقال (پایپ لاین) را می دهد. در نقطه مقابل، FOCAR برای UAN های چندهاپی ناهمگون پیشنهاد شده که برای این شبکه ها، طرح

هماهنگی چند هاپی، برای سناریوهای مذکور، بهینه سازی شده است. بنابراین توانایی هندل کردن انتقال داده قابل اطمینان در UAN ها را دارد که در آن ها کیفیت لینک در لینک های متفاوت می تواند تفاوت فاحشی داشته باشد. در بخش زیر، طراحی، پیاده سازی و آزمایش این سه رویکرد را به صورت دقیق توصیف خواهیم کرد.

3,1,1-ARQ هیبرید زیر آبی (UW-HARQ)

در این بخش، ARQ هیبرید زیر آبی (UW-HARQ) را ارائه می دهیم که خود، یک رویکرد مبتنی بر HARQ انتها به انتها، برای UAN می باشد. طراحی، پیاده سازی و آزمایش این پروتکل، به صورت دقیق به بحث و بررسی گذاشته می شود.

3,1,1- مطالعات مربوطه

در محیط خشن زیر آب، به دلیل محو شدگی چند مسیری [67]، حرکت ماهی ها و گره های شبکه زیر آبی با پویایی بالا [68,69]، کانال های آکوستیک از نرخ خطای بالای داده رنج می برد [70-72] که باعث بروز احتمال انتقال مجدد داده در UAN ها می شود (به خصوص هنگامی که چندین هاپ، در یک شبکه باشند) [73]. از طرف دیگر یک انتقال مجدد، در چندین هاپ می تواند تاخیر انتها به انتها کلی انتقال داده در UAN ها را به شدت افزایش دهد. دلیل این امر، این است که سیگنال آکوستیک، به جای سیگنال های رادیویی، به عنوان رسانه انتقال سیگنال در آن استفاده می شود که منجر به تاخیر طولانی تر در UAN ها می شود.

درخواست تکرار اتوماتیک (ARQ)، تکنیکی برای تضمین انتقال داده قابل اطمینان، در شبکه های زمینی است که به طور گسترده بر آن مطالعه شده است. در مورد از دست دادن بسته، یک تصدیق منفی (NACK) از گیرنده به فرستنده، بازفرستاده می شود و فرستنده، بسته های داده بیشتری می فرستد. این فرآیند تا زمانی که ACK، توسط فرستنده دریافت شود، ادامه پیدا می کند. ARQ، برای

انتقال داده قابل اطمینان در UAN اعمال شده است. در [74]، یک پروتکل توقف و انتظار ساده به عنوان طرح ARQ انتخاب شده و این مقاله، عنوان می دارد که تصدیق را می توان هم به صورت ضمنی و هم صریح به دست آورد. با این وجود، مسئله پایه در ARQ، این است که هیچ گونه افزونگی داده نداشته و در UAN هایی که احتمال بروز خطا وجود دارد، چندین انتقال مجدد، مورد نیاز است که خود، منجر به افزایش چشمگیر، در تاخیر انتها به انتها می شود

برای کاهش تعداد انتقال های مجدد، کدنویسی تصحیح خطای فوروارد (FEC)، اغلب مورد استفاده قرار می گیرد. FEC به صورت کلی، هم بسته های داده اولیه و هم چند بسته کدگذاری دیگر را به منظور فراهم آوردن افزونگی داده می فرستد. در صورت از دست رفتن بسته، احتمال بالایی وجود دارد که بسته های داده دریافت شده و بسته های داده کدگذاری هنوز بتوانند بسته های داده اولیه را بازیابی کنند. در این باره باید گفت که انتقال قابل اطمینان داده بخش بخش شده (SDRT) [36]، پیشنهاد شده است که از یک کد ساده تورنودو و یک ACK، به منظور دست یابی به انتقال داده قابل اطمینان چند هاپی، در UAN ها استفاده می کند. در SDRT، فرستنده ابتدا بسته ها را به چند بلاک دسته بندی می کند. سپس هر بلاک را کدگذاری کرده و بسته ها را به صورت سریع در یک پنجره منتقل می کند. انتظار می رود که بسته ها (شامل بسته های داده اولیه و بسته های کدگذاری شده) در پنجره، برای بازیابی بسته های داده اولیه در دریافت کننده، کافی باشند. پس از آن، انتقال داده کند شده و منتظر پاسخ از سوی دریافت کننده می شود. در صورتی که فرستنده، ACK را از دریافت کننده نگیرد، فقط یک بسته کدگذاری شده دیگر را به دریافت کننده، پس از انقضای زمان می فرستد. این روند تا زمانی که یک ACK دریافت شود، ادامه پیدا می کند. مسئله موجود در SDRT، این است که فقط با استفاده از ACK در صورتی که انتقال مجدد، مورد نیاز باشد، دریافت کننده نمی داند که بسته های داده در دریافت کننده، از دست رفته اند و چه تعداد بسته داده یا بسته کدگذاری شده باید در انتقال مجدد

بفرستد که خود این پدیده منجر به شکست انتقال داده دیگری می شود. برای پرداختن به مسائل مربوط به ARQ و FEC مرسوم، در این بخش، یک ARQ هیبرید زیر آبی (UW-HARQ) را پیشنهاد می کنیم که خود یک طرح HARQ برای انتقال داده قابل اطمینان در UAN ها است که کدنویسی FEC و ARQ را ترکیب می کند. برای کدگذاری FEC از کدنویسی خطی بایناری رندوم، برای کمینه نویسی کدنویسی و کدگذاری استفاده می کنیم که مطابق با قابلیت های محاسباتی پایین و ظرفیت پایین گره های UAN می باشد. برای ARQ، به جای استفاده از ACK، از بسته های NACK استفاده می کنیم تا به دریافت کننده، توانایی اعلام به فرستنده، درباره بسته های داده ای که در دریافت کننده، از دست رفته اند، بدهیم و به این صورت، فرستنده می تواند درباره این که چه تعداد و چه بسته هایی را باید برای انتقال مجدد بفرستد، تصمیم گیری کند. UW-HARQ، دو نقطه برتری دارد. در مقایسه با روش خالص ARQ، افزونگی داده را ارائه کرده و بنابراین احتمال انتقال مجدد داده را تا حد چشمگیری کاهش میدهد. از طرف دیگر در مقایسه با SDRT، به فرستنده، توانایی دریافت بازخورد از دریافت کننده را می دهد تا بتوان بسته های درست را هنگام نیاز به انتقال مجدد، مجددا ارسال کرد. در کنار طراحی UW-HARQ، یک مشارکت عمده دیگر، این است که UW-HARQ را در گره های UAN حقیقی، با استفاده از چارچوب پشته پروتکل شبکه زیرآبی آکوا-NET [75] و یک مودم آکوستیک زیرآبی، پیاده سازی کردیم: مودم تلدین بنتوس [26]. هم چنین چندین آزمایش آزمایشگاهی اولیه، با استفاده از گره های حقیقی UAN انجام داده و چندین نتیجه عملکرد نوید بخش از UW-HARQ دریافت کردیم.

3,1,2- طراحی پروتکل

UW-HARQ، ترکیبی از کدنویسی FEC و ARQ است. کدنویسی FEC، به منظور کاهش تعداد انتقال های مجدد میان گره منبع و مقصد، از طریق اضافه کردن افزونگی داده در انتقال داده انجام می شود و

هدف از آن، کاهش اثرات منفی اعمال شده از طریق از دست دادن بسته، در کانال های آکوستیک زیر آبی است. در این مطالعه، کدنویسی خطی بایناری رندوم (RBLC) را به عنوان طرح کدنویسی FEC، به کار می بندیم و دلیل آن، پیچیدگی کدگذاری و کدگشایی کمش است. با تعیین نسبت مناسب، برای کدنویسی می توانیم با احتمال بالا تضمین کنیم که بسته های دریافت شده در گره مقصد، برای بازیابی بسته های داده اولیه، علی رغم پاک کنندگی اعمال شده توسط کانال، کافی است. ARQ به منظور تضمین دریافت قابل اطمینان داده در گره مقصد، مورد استفاده قرار می گیرد. یک بسته NACK به فرستنده ارسال می شود تا شکست در بازیابی داده را نشان داده و درخواست بسته های دیگر را کند. در حالی که یک NACK به عنوان دلالتی بر موفقیت بازیابی داده، فرستاده می شود. در UW-HARQ، ARQ یک هدف مهم دیگر نیز دارد: اعلان نرخ خطای بسته (PER) به فرستنده، میان گره های منبع و مقصد؛ که اطلاعاتی است که توسط بسته های ACK و NACK حمل می شود. با استفاده از اطلاعات PER، فرستنده می تواند نسبت کدنویسی دقیق تری را به منظور بیشینه سازی احتمال موفقیت بازیابی داده در گره مقصد تنظیم کند.

3,1,2,1- کدنویسی خطی بایناری رندوم:

گزینه های مختلفی برای طرح کدنویسی FEC وجود دارند (شامل کدنویسی تورنادو، کدنویسی فواره ای [78] و کدنویسی خطی رندوم). در مطالعه در دست، از RBLC به عنوان طرح کدنویسی استفاده می کنیم زیرا پیچیدگی محاسباتی آن کم بوده و برای گره های UAN که ظرفیت توان و پردازش کمی دارند، مناسب است. در RBLC، هر بسته کدگذاری شده، یک ترکیب خطی رندوم، از چندین بسته داده است. در این جا از برای نشان دادن K بسته داده اول و از برای نشان دادن n بسته رمزگذاری شده استفاده می کنیم. سپس:

$$y_j = a_{j1}x_1 \oplus a_{j2}x_2 \oplus \dots \oplus a_{jk}x_k = \sum_{i=1}^k a_{ji}x_i$$

$$j \in [1, n] \quad a_{ji} = \{0, 1\} \quad (1)$$

که به عنوان نرخ کدنویسی RBLC تعریف می شود. نرخ کدنویسی C برای عملکرد UW-HARQ، نقشی حیاتی دارد. در صورتی که مقدار بیش از حد کمی را برای C، اختصاص دهیم، به دلیل از دست رفتن بسته، انتقال های مجدد، برای بازیابی K بسته داده در گره مقصد، محتمل می شوند که می توانند به طرز چشمگیری، تاخیر انتها به انتهای کلی را میان منبع و مقصد، افزایش دهند. از طرف دیگر، C ای که بسیار بزرگ باشد باعث می شود که گره منبع، بسته های رمزگذاری شده غیرضروری ای بفرستد که منجر به مصرف انرژی بیشتر می شود. تعیین یک نرخ کدنویسی مناسب، برای تضمین این که پس از هر انتقال، دریافت کننده بتواند K بسته داده اولیه را از بسته های کدگذاری شده دریافت شده، با احتمال بالا، بازیابی کند، کاملاً حیاتی است. به منظور انجام این کار، باید قابلیت بازیابی داده RBLC را تحلیل کنیم. در این جا نمادهای زیر را معرفی می کنیم:

●X نشان دهنده k بسته داده اولیه در یک بلاک است

$$X = \{x_1, x_2, \dots, x_k\}^T \quad (2)$$

●Y نشان دهنده n بسته کدگذاری شده است

$$Y = \{y_1, y_2, \dots, y_n\}^T \quad (3)$$

● **A** ماتریس رمزگشایی است. از آن جایی که کدنویسی خطی رندوم، مورد استفاده قرار گرفته است، **A**، خصوصیات زیر را دارد. اول این که ، یک یا صفر است. دوم این که تعداد بسته های داده اولیه در **X** که در تشکیل یک بسته کدگذاری شده در **Y** مشارکت می کنند، رندوم است که به معنای این است که ، یک عدد رندوم میان صفر و **K** می باشد. هم چنین به عنوان درجه بسته کدگذاری شده تعریف می شود. سوم این که با توجه به درجه مشخص برای ، به صورت یک خاص که بسته از **K** بسته داده اولیه در **X** در تشکیل مشارکت دارند، رندوم است. آن چه عنوان کردیم به این معنا است که با توجه به این درجه خاص، برای ، در ، یک و سایر عناصر صفر هستند که عنصر رندوم است.

$$A = \begin{pmatrix} a_{11} & a_{21} & \dots & a_{n1} \\ a_{12} & a_{22} & \dots & a_{n2} \\ \dots & & \dots & \\ a_{1k} & a_{2k} & \dots & a_{nk} \end{pmatrix} \quad (4)$$

$$X = A * Y \quad (5)$$

مسئله در این جا این است که با توجه به ماتریس رمزگشایی **A**، احتمال این که **k** بسته داده اولیه بتواند از **n** بسته کدگذاری شده، رمز گشایی شود، چیست؟ توجه داشته باشید که **n** باید برابر یا بزرگتر از **k** باشد. در غیر این صورت به هیچ روشی نمی توان **k** بسته داده اولیه را رمزگشایی کرد. در [77] اثبات شده است که کران بالا، برای اعمال شکست رمزگشایی، است. با توجه به این که **n** بسته کدگذاری شده برای رمزگشایی **k** بسته داده اولیه مورد استفاده قرار گرفته اند.

با توجه به این که n بسته کدگذاری شده برای بازیابی K بسته داده اولیه در اختیار گذاشته شده اند. به منظور حل این مسئله آن را به یک مسئله دیگر تبدیل می کنیم. احتمال این که ماتریس رمزگشایی A ، درجه ای برابر با k داشته باشد چیست؟ منطق این مسئله این است که در صورتی که درجه A ، k باشد می توانیم یک مجموعه راه حل منحصر به فرد، برای X از n معادله داشته باشیم که خود به معنی رمزگشایی موفقیت آمیز است.

برای این که A ، یک درجه K داشته باشد، تمامی k سطرها در A باید نسبت به یکدیگر، استقلال خطی داشته باشند. می توانیم A را به صورت سطر به سطر بررسی کنیم. برای سطر اول، تنها شرطی که باید رعایت شود این است که در این سطر، همگی نمی توانند صفر باشند. احتمال این که این شرط، صادق باشد، است. برای سطر دوم، دو شرط وجود دارد که باید رعایت شود: عناصر نمی توانند همگی صفر باشند و دقیقاً نمی توانند با سطر اول، برابر باشند. احتمال این که هر دوی این شرط ها رعایت شوند، شوند. در سطر سوم، نباید ترکیب خطی ای از دو سطر اول داشته باشیم؛ به این معنا که معادله زیر نباید صادق باشد: که در آن، صادق است. از آن جایی که، ترکیب ممکن برای و است، احتمال این که سطر سوم، استقلال خطی با دو سطر اول داشته باشند، است. به صورت مشابه، برای امین سطر، باید استقلال خطی، با $K-1$ سطر اول، حاصل شود. بنابراین احتمال این که امین سطر، یک سطر معتبر باشد، است. بنابراین احتمال این که درجه A برابر با k باشد (یعنی احتمال موفقیت رمزگشایی، با توجه به این که n بسته کدگذاری شده، برای رمزگشایی k بسته داده اولیه مورد استفاده قرار می گیرند)، به صورت زیر است

$$Pr(succ) = \prod_{i=0}^{k-1} \left(1 - \frac{2^i}{2^n}\right) = \prod_{i=0}^{k-1} (1 - 2^{i-n}) \quad (6)$$

بنابراین احتمال شکست رمزگشایی، به صورت زیر است

$$Pr(fail) = 1 - \prod_{i=0}^{k-1} (1 - 2^{i-n}) \quad (7)$$

اکنون می توانیم نحوه تعیین کدنویسی مناسب را مطالعه کنیم :

•C، نرخ کدنویسی است؛ یعنی

•، یک ثابت کوچک است که نشان دهنده احتمال شکست رمزگشایی، پس از یک انتقال می باشد

با تلفیق معادله 7 به معادله زیر می رسیم

$$1 - \prod_{i=0}^{k-1} (1 - 2^{i-(1+C)*k}) \leq \delta \quad (8)$$

با این وجود، نسبت کدنویسی بهینه بالا C، با فرض این، حاصل می شود که هیچ گونه از دست دادن بسته ای، میان گره مبدا و مقصد وجود ندارد. UAN های جهان حقیقی که PER غیرقابل چشم پوشی در آن ها وجود دارد، نسبت کدنویسی حقیقی باید از C در معادله 8، بزرگتر باشد تا بتوان با پدیده از دست رفتن بسته در کانال های آکوستیک، مقابله کرد.

UW-HARQ، نسبت کدنویسی بهینه تئوری C و اطلاعات PER را برای برآورد نسبت کدنویسی بهینه حقیقی تلفیق می کند. اطلاعات PER از طریق بسته های ACK/NACK به دست می آید. هنگامی که گره مقصد، یک بسته ACK یا NACK به گره منبع بفرستد، تعداد بسته های دریافت شده در آخرین

انتقال بسته های رمزگذاری شده را تعبیه می کند. در صورت دریافت یک بسته ACK یا NACK، گره منبع می تواند PER را با استفاده از این اطلاعات و تعداد کلی بسته های کدگذاری شده ثبت شده ای که در آخرین انتقال، فرستاده شده اند، محاسبه کند. سپس نسبت کدنویسی بهینه حقیقی را می توان به صورت زیر تنظیم کرد

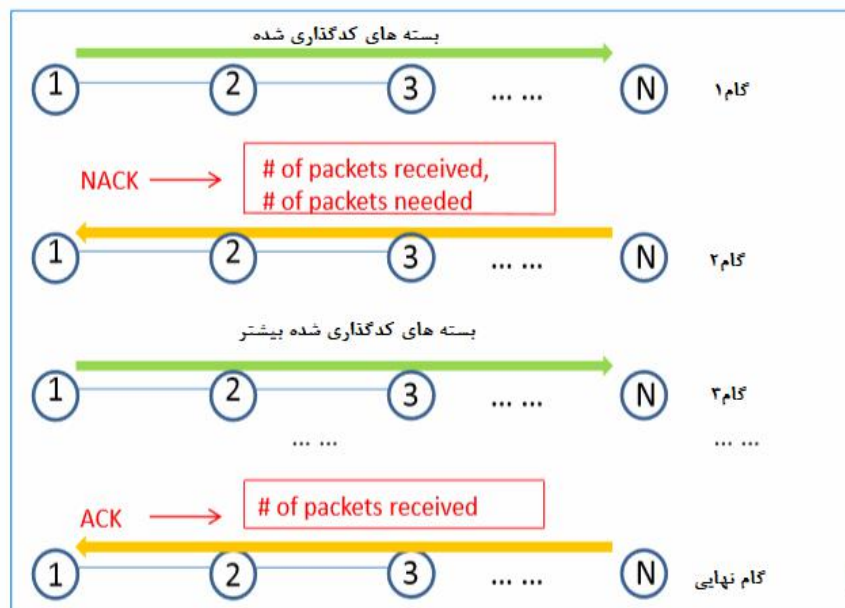
$$C' = \frac{C}{1 - PER} \quad (9)$$

طرح برآورد نسبت کدنویسی UW-HARQ بر کیفیت لینک پویای کانال های آکوستیک زیرآبی، از طریق تلفیق اطلاعات PER، انطباق پذیرتر است. هم چنین سربرار ارتباطاتی بسیار کوچکی را به همراه دارد زیرا تنها اطلاعات اضافی انتقال یافته، تعداد بسته های دریافت شده است که از طریق بسته های ACK و NACK حمل می شود.

3,1,2,2- درخواست تکرار اتوماتیک

هدف ARQ در UW-HARQ، دوگانه است. از یک طرف، بسته های ACK و NACK که فرستاده شده اند شامل تعداد بسته های دریافت شده هستند و بنابراین فرستنده می تواند از آن ها برای برآورد کردن

PER کنونی، میان مبدا و مقصد استفاده کند. از طرف دیگر به صورت متفاوت از UW-SDRT، HARQ از NACK، برای اطلاع دادن به منبع، درباره تعداد بسته های داده ای که در مقصد، از دست رفته اند و هم چنین این که این بسته ها، کدام موارد هستند، مورد استفاده قرار می دهد و به این صورت، گره منبع می تواند تصمیم درستی برای فرستادن بسته های کدگذاری شده، در یک انتقال مجدد، برای تضمین این که مقصد بتواند بسته های داده از دست رفته را بازیابی کند، اتخاذ نماید



شکل 1- نمایش ARQ در UW-HARQ

روند ARQ UW-HARQ در شکل 1 به تصویر کشیده شده است. در ابتدا، گره منبع، K بسته داده را در یک بلوک، گروه بندی کرده و آن ها را به بسته کدگذاری شده، با استفاده از RBLC، کدگذاری می کند. توسط معادله 9 تعیین می شود و مقدار اولیه PER را برابر با 0 می گذاریم. در گام دوم در صورتی که گره مقصد نتواند K بسته داده را از بسته های کدگذاری شده دریافت شده، بازیابی کند، یک NACK را به گره منبع می فرستند. بسته NACK شامل تعداد بسته های کدگذاری دریافت شده است که توسط گره منبع، برای به روزرسانی اطلاعات PER، مورد استفاده قرار می گیرد و تعداد بسته های کدگذاری شده ای است که هنوز برای بازیابی بسته های داده از دست رفته، مورد نیاز هستند. در گام سوم، با استفاده از PER و به روزرسانی شده، گره منبع می تواند درباره تعداد بسته های کدگذاری شده ای که باید در انتقال مجدد فرستاده شوند و هم چنین خود بسته ها، تصمیم گیری کند. این روند تا زمانی ادامه پیدا می کند که یک بسته ACK در گره منبع دریافت شود که خود نشان می دهد K بسته داده به صورت موفقیت آمیز در گره مقصد، بازیابی شده است. بنابراین با استفاده از RBLC و طرح

برآورد نرخ کدنویسی انطباقی ، UW-HARQ می تواند با احتمالی بالا، تضمین کند که یک انتقال از بسته های کدگذاری شده، برای بازیابی تمامی بسته های داده اولیه در گره مقصد، کافی هستند و بنابراین تعداد انتقال های مجدد را به حداقل می رساند. UW-HARQ هم چنین برای ARQ، نه تنها برای تضمین دریافت اطلاعات قابل اطمینان، بلکه برای اعلان به گره منبع، درباره PER و بسته های داده از دست رفته مورد استفاده قرار می گیرد که این امر توسط گره منبع، برای تعیین تعداد بسته های کدگذاری شده ای که باید در انتقال مجدد فرستاده شوند و هم چنین خود بسته ها، مورد استفاده قرار خواهد گرفت.

3,1,3- نتایج آزمایش اولیه

برای ارزیابی عملکرد UW-HARQ، آزمایشات اولیه با استفاده از آکوا-NET و بستر آزمایشی ما، آکوا-TUNE انجام شدند [78]. در یک پیکربندی آزمایشگاهی عادی، از شش مودم تلدین بنتوس استفاده کردیم: 4 مودم ATM 920، یک مودم UDB9000 دِک باکس و یک مودم هوشمند SM975 [26]. این 6 مودم تلدین بنتوس، رابط یکسانی را برای نرم افزارهای لایه بالاتر ارائه داده و پارامترهای عملکردی یکسانی را از لحاظ ارتباط آکوستیک دارد. هر مودم به یک گام استیک، وصل می شوند که بر روی آن، UW-HARQ بر لینکوکس تعبیه شده و آکوا-NET اجرا می شود

به منظور ارزیابی عملکرد UW-HARQ، یک رویکرد شبه TCP، آکوا-SARQ را به عنوان پایه، پیاده سازی کردیم. آکوا-SARQ نیز پنجره لغزان و تصدیق انتخابی، برای حصول اطمینان پذیری انتها به انتها استفاده می کند. گره منبع، بسته های داده را در پنجره لغزان به گره مقصد منتقل می کند. یک ACK، پس از دریافت یک یا چند بسته داده توسط مقصد، فرستاده می شود و به این صورت، منبع می تواند پنجره را بلغزاند و بسته های داده جدیدی را ارسال کند. در آزمایش ما، پارامترهای شبکه به صورت زیر هستند (مگر آن که خلافتش عنوان شود): شمارش هاپ، چهار است. سائز بلاک، پنج است

به این معنا که هر پنج بسته داده در یک بلاک جمع و کدگذاری شده اند. طول بسته، دویست است.

نرخ بیت آکوستیک مودم، 800 bps است. سائز پنجره لغزان، برای آکوا-SARQ، یک است

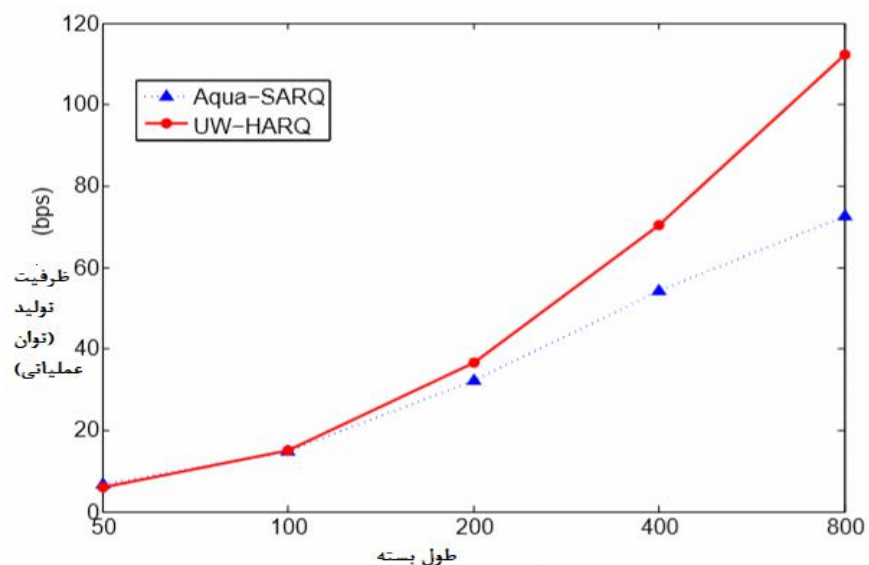
عملکرد UW-HARQ و آکوا-SARQ را بر پایه دو مقیاس، ارزیابی می کنیم. توان عملیاتی انتها به انتها به صورت تعداد بیت های دریافت شده در گره مقصد به ازای هر ثانیه تعریف می شود (bps). سربار، به عنوان نسبت، میان تعداد بسته های اضافی و تعداد بسته های داده منتقل شده در چند هاپ در شبکه تعریف می شود. در بخش زیر، درباره دو مقیاس UW-HARQ و آکوا-SARQ، تحت شرایط مختلف به بحث و بررسی می نشینیم .

3,1,3,1-تاثیر طول بسته

در این بخش به تاثیر طول بسته بر توان عملیاتی انتها به انتها UW-HARQ و آکوا-SARQ می پردازیم. طول بسته رامیان 50 و 800 تغییر می دهیم و نتایج حاصله ، در شکل 2 به تصویر کشیده شده اند. همانطور که می بینید توان عملیاتی UW-HARQ به نسبت آکوا-SARQ، بیشتر بوده و نقطه برتری UW-HARQ با افزایش طول بسته بیشتر می شود. دلیل این پدیده این است که UW-HARQ از RBLC و تقریب نسبت کدنویسی، برای مقابله با از دست رفتن بسته، میان منبع و مقصد استفاده می کنیم و به این صورت به توان عملیاتی بسیار بالاتری می رسد. از طرف دیگر، آکوا-SARQ هیچ گونه افزونگی داده نداشته و بنابراین هر از دست رفتن بسته منجر به انتقال مجدد، در چندین هاپ می شود. این پدیده باعث تنزل چشمگیر زمان عملیاتی انتها به انتها می شود. PER با افزایش طول بسته، بزرگتر شده و این پدیده، دلیل این که نقطه برتری UW-HARQ به نسبت آکوا-SARQ، با افزایش طول بسته، بیشتر می شود را توضیح می دهد.

3,1,3,2-تاثیر شمارش هاپ

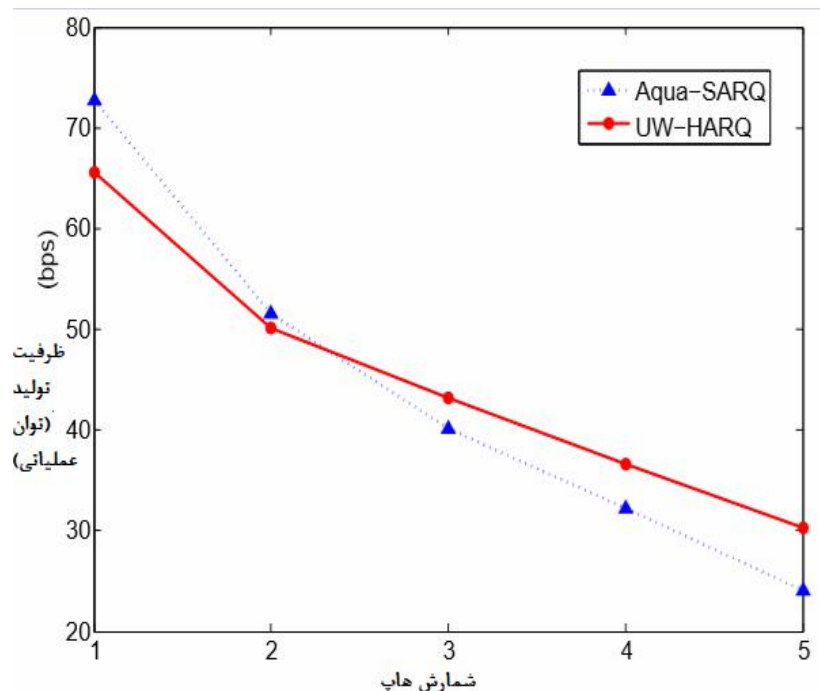
تاثیر شمارش هاپ بر توان عملیاتی انتها به انتها UW-HARQ و آکوا-SARQ در شکل 3 به تصویر کشیده شده است. به صورت کلی، UW-HARQ، عملکرد بهتری از لحاظ توان عملیاتی، به خصوص با شمارش های بالای هاپ، به نسبت آکوا-SARQ دارد. با شمارش هاپ بزرگتر، احتمال بیشتری وجود دارد که لینکی میان مقصد و مبدا وجود داشته باشد که PER در آن بالا بوده که این پدیده می تواند به طرز چشمگیری، احتمال شکست، در تحویل انتها به انتهای بسته را به همراه داشته باشد. این پدیده منجر به انتقال های مجدد بیشتر، برای آکوا-SARQ، با افزایش شمارش هاپ می شود. UW-HARQ از دست رفتن بسته به شکل کمتری رنج می برد و دلیل آن، RBLC و برآورد نسبت کدنویسی انطباقی است.



شکل 2- توان عملیاتی انتها به انتها با طول های بسته متغیر

3,1,3- سربار

در این بخش، سربار UW-HARQ و آکوا-SARQ را بررسی می کنیم که توسط نسبت میان تعداد بسته های اضافه و تعداد بسته های داده اندازه گیری می شود. نتایج، در شکل 4 به تصویر کشیده شده اند و سایز بلوک، از 5-25 متغیر است. می بینیم که سربار UW-HARQ به نسبت سربار آکوا-SARQ، بیشتر بوده اما هنوز هم قابل قیاس است. بسته های اضافه منتقل شده در UW-HARQ، از افزونگی داده، می آیند که برای کاهش تعداد انتقال های مجدد میان مبدا و مقصد، مورد استفاده قرار می گیرد.



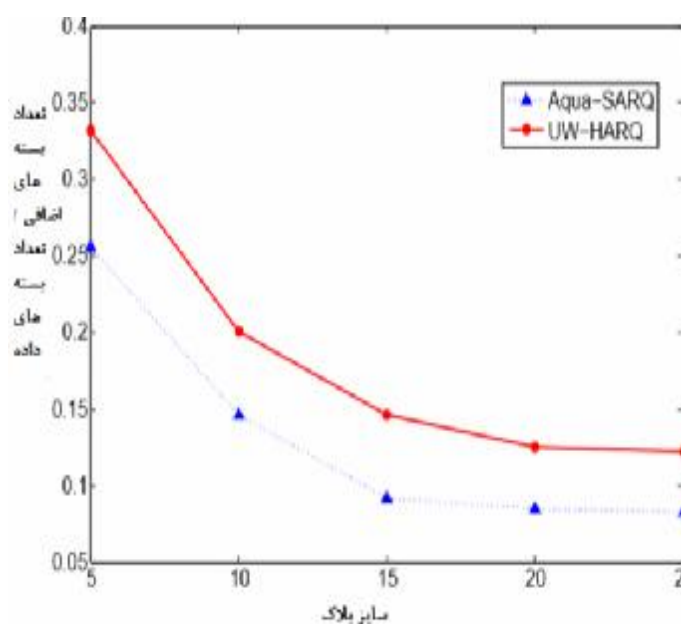
شکل 3- توان عملیاتی انتها به انتها، با شمارش متغیر هاپ

3,1,4- خلاصه

در این مطالعه، یک طرح هیبرید ARQ زیرآبی، با نام UW-HARQ را برای انتقال داده قابل اطمینان چند هاپی در UAN ها پیشنهاد کردیم که ترکیبی از کدنویسی FEC و ARQ است. کدنویسی خطی بایناری رندوم، همراه با کدنویسی FEC، برای کمینه سازی پیچیدگی محاسباتی صورت می گیرد. یک

طرح برآورد نسبت کدنویسی، برای کاهش چشمگیر تعداد انتقال های مجدد، با لحاظ کردن PER، میان مقصد و مبدا پیشنهاد شده است. برای ARQ، UW-HARQ از NACK، برای فرستادن بازخورد به مبدا استفاده می کند و به این گونه گره مبدا می تواند اطلاعات PER را به روزرسانی کرده و تعیین کند کدام بسته های کدگذاری شده باید در انتقال مجدد فرستاده شود.

علاوه بر طراحی پروتکل، یک UW-HARQ را در گره های UAN حقیقی، پیاده سازی کرده ایم که از مودم تلداین بنتوس، آکوا-NET استفاده می کنند. هم چنین آزمایشات اولیه را با استفاده از بستر آزمایشی آکوا-TUNE انجام داده ایم. نتایج نشان می دهند که UW-HARQ از رویکرد انتقال داده قابل اطمینان شبه TCP، از لحاظ توان عملیاتی انتها به انتها، عملکرد بهتری داشته و به سربار قابل قیاسی می رسد.



شکل 4- سربار با سایز بلوک متغیر

3,2- انتقال داده قابل اطمینان هماهنگ شده چندهاپی، مبتنی بر کدنویسی

UW-HARQ، عملکرد بهتری به نسبت روش های مبتنی بر تکرار انتخابی انتها به انتهای مرسوم، بدون

کدنویسی FEC، از لحاظ توان عملیاتی حاصله دارد. با این وجود توان عملیاتی حاصله هنوز هم بسیار پایین است. دلیل مربوطه این است که در UAN چند هاپی، از دست رفتن یک بسته در هر هاپ میانی،

منجر به انتقال مجدد انتها به انتها در چندین هاپ می شود که هزینه چشمگیری را از لحاظ تاخیر

انتقال بالا، تاخیر انتشار و تاخیر تبدیل حالت مودم، به همراه خواهد داشت. برای حل این مشکل، در

این بخش، انتقال داده قابل اطمینان هماهنگ شده چندهاپی، مبتنی بر کدنویسی (CCRDT) را برای

انتقال داده قابل اطمینان و بهره ور در UAN ها پیشنهاد می کنیم. CCRDT به صورت متفاوت از

UW-HARQ، یک رویکرد هاپ به هاپ است. هم چنین هدف CCRDT، پلت فرم های UAN

قدرتمند (همانند پلت فرم های مبتنی بر گام استیک [42]) می باشد که توان محاسباتی بالاتری داشته

و قیود انرژی آن، کمتر است. در زیربخش پیش رو، طراحی، آزمایش و پیاده سازی CCRDT را توصیف

می کنیم.

3,2,1- مطالعات مربوطه

همانطور که در بخش 3,1,1 عنوان شده است، SDRT که یک رویکرد مبتنی بر HARQ هاپ به هاپ

می باشد، برای انتقال داده قابل اطمینان در UAN ها پیشنهاد شده است. SDRT، در سیستم های

حقیقی پیاده سازی و در محیط زیر آب، آزمایش شده است [75]. با این وجود مسائل گوناگونی

مشاهده شده اند

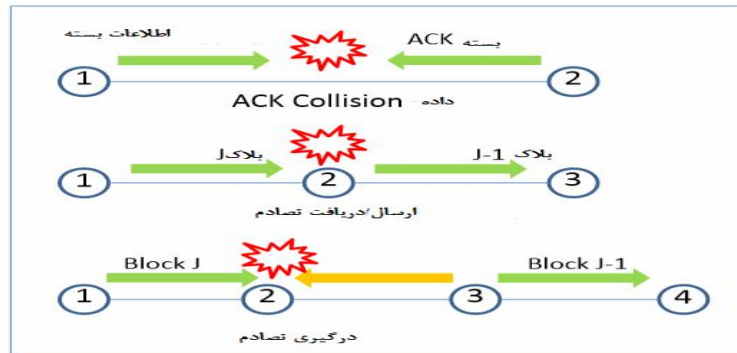
برای SDRT، پیاده سازی طرح برآورد نسبت کدنویسی در گره های حقیقی UAN، امری دشوار است.

دلیل مربوطه این است که SDRT از کدنویسی SVT استفاده می کند که یک طرح کدنویسی اسپارس

است. کدنویسی SVT می تواند تضمین کند که K بسته رمزگذاری شده، توانایی بازیابی K بسته داده را

دارد، فقط هنگامی که K به اندازه کافی بزرگ باشد. بیشتر طرح های کدنویسی اسپارس، همانند کدنویسی فواره ای نیز این خصیصه را دارند. با این وجود در کارهای UAN جهان حقیقی، K اغلب از 100 کوچکتر است. آن چه عنوان کردیم به این معنا است که حتی بدون از دست رفتن بسته، K بسته رمزگذاری شده که توسط فرستنده ارسال شده است، نمی تواند K بسته داده را در دریافت کننده، بازیابی کند. نتیجتاً افزونگی بهینه کدنویسی SDRT وابسته به نرخ خطای بسته (PER) و بر ویژگی کدنویسی SVT بلاک است. محققان در [36]، یک الگوریتم پیچیده درباره نحوه به دست آوردن بهینه پیشنهاد کردند. با این وجود این عمل را نمی توان بر اساس آزمایشات ما، به دلیل پیچیدگی و دشواری بالا در زمینه به دست آوردن درجه کدنویسی بهینه، به صورت بی درنگ در گره های UAN انجام داد.

برای پرداختن به این مسئله، یک کدنویسی رندوم GF(256) (RLC)، پیاده سازی می کنیم که می تواند تقریباً به صورت 100% تضمین کند که K بسته رمزگذاری شده، توانایی بازیابی K بسته داده را دارد [79]. با استفاده از RLC، برآورد نسبت کدنویسی، از آن جایی که وابستگی بر PER به وجود می آید، ساده می شود. می توانیم PER را در یک هاپ، از طریق بسته های ACK/NACK، از دریافت کننده به دست آوریم که نشان می دهد به تعداد بسته، در آخرین انتقال، دریافت شده اند. سپس نسبت کدنویسی R در RLC را می توان به فرستاد تا تضمین کرد که K بسته رمزنگاری شده برای بازیابی K بسته داده دریافت شده است. باید عنوان کرد که RLC، سر بار محاسباتی بالاتری به نسبت SVT را ملزم می دارد زیرا بسته رمزنگاری شده در RLC، شامل تمامی K بسته داده است با این وجود در گره UAN قوی تر و K کوچک، رمزنگاری و رمزگشایی RLC را می توان به صورت بیدرنگ انجام داد و هیچ گونه تاخیر قابل توجهی ایجاد نمی شود. آزمایشات آزمایشگاهی ما، این پدیده را اثبات کرده است.



شکل 5- سه نوع تصادم در SDRT

مسئله دومی که در SDRT، متوجه آن شدیم این است که این طرح، در واقع طرح هماهنگی چند هاپی نداشته و متعاقباً تصادم ها، با تکرار فراوان، در UAN های چندهاپی ایجاد می شود. سه نوع تصادم مشاهده شدند: داده-ACK، تصادم فرستادن-دریافت و استراق سمع که در شکل 5 نشان داده شده اند. این تصادم ها، در شبکه وایرلس زمینی (TWN ها) و UAN ها مورد مطالعه قرار گرفته اند. این تصادم ها را می توان با طرح های مبتنی بر RTS/CTS [80] یا روش های مبتنی بر زمان بندی، کاهش داد [81]. در زمینه سربار محسوس RTS/CTS که توسط تاخیر انتشار طولانی و خصوصیات شبکه با توپولوژی رشته ای ایجاد می شود، یک طرح هماهنگی چند هاپی مبتنی بر زمان بندی را برای حذف تصادم ها اتخاذ نموده ایم

هم چنین آکوا-SARQ را پیاده سازی کردیم که یک رویکرد انتها به انتها بوده و وابسته به پنجره لغزان و انتخاب تکراری، برای تضمین قابلیت اطمینان است که در بخش 3,1,3، قبلاً توصیف شده است. قطعاً این طرح، طرح خط لوله (پایپ لاین) ندارد تا به چندین فرستنده اجازه عملکرد در یک زمان را بدهد. بنابراین هم چنین خط لوله را در طرح هماهنگی چندهاپی خود، برای بهبود بخشی توان عملیاتی انتها به انتها تلفیق کردیم

بنابراین به منظور پرداختن به مسائل مذکور و توسعه یک پروتکل انتقال داده قابل اطمینان و عملی که در UAN های جهان حقیقی به خوبی جواب می‌دهد، CCRDT را طراحی نمودیم که خود، یک پروتکل انتقال داده قابل اطمینان، با هماهنگی چند هاپی، مبتنی بر کدنویسی، برای UAN ها است. برای انتقال داده، به ازای هر هاپ، CCRDT از کدنویسی خطی رندوم GF(256) (تکرار انتخابی) برای تضمین قابلیت اطمینان و بهره‌وری استفاده می‌کند. طرح کدنویسی انتخابی، برآورد نسبت کدنویسی را برای پیاده‌سازی در گره‌های UAN حقیقی، بسیار ساده‌تر می‌کند. برای شبکه چند هاپی، CCRDT از یک طرح هماهنگی چند هاپی، برای حذف تصادم استفاده می‌کند. این هماهنگی چند هاپی، یک طرح خط لوله را پیاده‌سازی می‌کند که به چندین گره، اجازه انتقال همزمان داده و بنابراین تا حدود زیادی، توان عملیاتی آنها به انتها را افزایش می‌دهد. CCRDT را در یک سیستم حقیقی، پیاده‌سازی کرده و آزمایشات گسترده‌ای را برای ارزیابی عملکردش انجام دادیم. نتایج آزمایشگاهی نشان می‌دهند که CCRDT از سایر طرح‌های قابل انتقال موجود که در گذشته پیاده‌سازی شده‌اند، عملکرد بهتری از خود نشان می‌دهند

3,2,2- طراحی پروتکل

در این بخش، طراحی CCRDT را توصیف می‌کنیم که شامل انتقال داده به ازای هر هاپ و هماهنگی چند هاپی می‌شود. CCRDT، یک UAN چند هاپی همگون را با یک فرستنده و دریافت کننده، در یک توپولوژی رشته‌ای هماهنگ می‌کند که یک سناریو کاربردی عادی، برای انتقال داده قابل اطمینان، در محیط آبی است. یک شبکه همگون به این معنا است که فاصله و PER ها، در هر هاپ، به یکدیگر نزدیک هستند که توانایی پیاده‌سازی سیستم خط لوله را می‌دهند.

3,2,2,1- کدنویسی خطی رندوم GF(256)

با الهام گیری از کدنویسی خطی شبکه رندوم که در UAN ها اعمال می شود، CCRDT از کدنویسی خطی رندوم، در میدان گالیس ، با اسم GF(256) استفاده می کند. ما از برای نشان دادن K بسته داده در یک بلوک استفاده می کنیم. K بسته داده به صورت خطی، برای به دست آوردن بسته کدگذاری شده که به صورت نشان داده شده و در آن، صادق است، ترکیب می شود. در این جا، ضریب کدگذاری است که به صورت رندوم، از میدان گالیس GF انتخاب شده است. به عنوان بردار رمزنگاری، برای استفاده شده است. اغلب به عنوان نسبت کدگذاری تعریف می شود که توسط حرف r نشان داده می شود.

احتمال موفقیت کدنویسی خطی رندوم، برای بازیابی K بسته داده از K بسته رمزنگاری شده، بر مقدار q وابسته است که در واقع همان تعداد عناصر در میدان گالیس است. به عبارت دیگر با در اختیار داشتن q ای که به اندازه کافی بزرگ باشد، می توانیم همیشه از حصول احتمال موفقیت رمزگشایی بالا مطمئن باشیم. در آزمایشات ما، با کدنویسی خطی رندوم GF(256)، احتمالی که K بسته رمزنگاری شده نتوانند K بسته داده را بازیابی کنند، قابل چشم پوشی است.

یک روش، برای تضمین احتمال موفقیت 100%، برای بازیابی K بسته داده از K بسته رمزنگاری شده با استفاده از کدنویسی خطی رندوم، استفاده از بردارهای رمزنگاری است که توسط یک ماتریس واندرموند، تولید شده است (به نسبت بردارهایی که به صورت رندوم تولید شده اند [82]). در این مطالعه از بردارهای رمزنگاری که به صورت رندوم تولید شده اند، به دلیل سادگی کار استفاده شده اند

کدنویسی خطی رندوم GF(256) را به دلیل قابلیت بازیابی داده قدرتمندش انتخاب کرده ایم که به صورت چشمگیری، برآورد نسبت کدنویسی را تسهیل می کند و این امر را در بخش 3,2,2 به بحث و بررسی می گذاریم . پیچیدگی رمزگشایی آن، $O(K^3)$ است که در آن، K، تعداد بسته های رمزنگاری

شده است. با وجود این که پیچیدگی رمزگشایی، به نسبت کدنویسی خطی بایناری رندوم و سایر طرح های کدنویسی اسپارس، بیشتر است، گره های UAN که از لحاظ محاسباتی قوی هستند و با وجود یک K که نسبتا کوچک باشد، در کاربردهای UAN جهان حقیقی می توانیم تضمین کنیم که رمزگشایی را به صورت بی درنگ بتوان انجام داد.

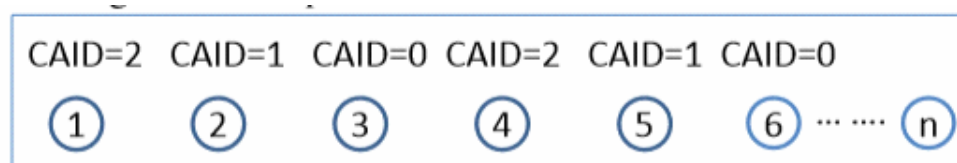
3,2,2,2- انتقال داده به ازای هر هاپ

انتقال داده به ازای هر هاپ، در CCRDT، به گونه ای طراحی شده است تا تحویل قابل اطمینان یک بلوک، از بسته داده در یک هاپ را با حداقل تاخیر، تضمین کند. این پدیده، رابطه نزدیکی با کدنویسی خطی رندوم GF(256) دارد. هنگامی که فرستنده، شانس فرستادن یک بلوک، از K بسته داده را دریافت می کند، بسته رمزنگاری شده را با استفاده از کدنویسی خطی رندوم GF(256) تولید می کند که در آن، r به عنوان نسبت کدنویسی کنونی تعریف می شود. پس از آن، دریافت کننده، بسته رمزنگاری شده را به دریافت کننده می فرستد و برخی از این بسته ها، ممکن است به دلیل حذف شدگی کانال، از دست بروند. در صورتی که دریافت کننده بتواند K بسته داده را بازیابی کند، یک ACK را پس می فرستد، در غیر این صورت، یک NACK را پس می فرستد که نشان می دهد در بازیابی داده، دچار شکست شده است. هم ACK و هم NACK شامل اطلاعاتی درباره تعداد بسته های دریافت شده در صف آخرین دریافت کننده در انتقال هستند. بنابراین فرستنده می تواند اطلاعات PER خود را به روز رسانی کرده و به نوبه خود، نسبت کدنویسی r را هنگامی که یک ACK یا NACK را دریافت می کند، به روزرسانی کند. پس از دریافت یک NACK، فرستنده، بسته های رمزنگاری شده بیشتری را برپایه r به روزرسانی شده و تعداد بسته های رمزنگاری شده ای که هموز در دریافت کننده، مورد نیاز هستند، می فرستد، این روند تا زمانی ادامه پیدا می کند که یک ACK در فرستنده دریافت شود. با استفاده از روش تکرار انتخابی، قابلیت اطمینان در یک هاپ را می توان به دست آورد.

با کدنویسی رندوم خطی $(GF(256))$ ، تعیین نسبت کدنویسی r ، ساده می شود. هنگامی که دریافت کننده، یک ACK یا NACK را دریافت کرد، اطلاعات PER خود را بر روی این هاپ، به روزرسانی کرده و سپس r را برابر با تنظیم می کند. سپس در صورتی که فرستنده، بسته کدگذاری شده را بفرستد، K بسته داده کدگذاری شده را می توان در دریافت کننده، با احتمال بالا دریافت کرد. به دلیل توان کدنویسی خطی، رندوم $(GF(256))$ ، K بسته داده را می توان بازیابی کرد. به این صورت با احتمال بالا می توانیم تضمین کنیم که یک انتقال از بسته رمزنگاری شده، برای بازیابی K بسته داده، کافی است و بنابراین تاخیر انتقال داده در یک هاپ را می توان به حداقل رساند. روش دیگری برای به روز رسانی اطلاعات PER در فرستنده، این است که هم PER میانگین و هم PER لحظه ای را در معادلات خود وارد کنید. برای مثال که در آن پارامتر سیستم است، مقدار میانگین PERهایی است که قبلاً به دست آمده است و آخرین PERای است که از ACK یا NACK به دست آمده است. در آزمایش متوجه شدیم که استفاده از PER لحظه ای، به نسبت استفاده از PER میانگین، بهتر است زیرا با دقت بیشتری، PER کنونی را در یک هاپ نشان می دهد.



شکل 6- فرستنده های چندگانه همزمان بدون برخورد



شکل 7- تخصیص های CAID

3,2,2,3- هماهنگی چند هاپی

در بخش 3,2,2,2، رفتار یک گره را پس از دریافت فرصت، برای فرستادن یک بلاک از دسته داده به

بحث و بررسی گذاشتیم. در این بخش، درباره هنگامی که یک گره، در یک UAN چندهاپی، باید

فرصت فرستادن را دریافت کند، بحث می کنیم که این پدیده توسط طرح هماهنگی چند هاپی

CCRDT تعیین می شود. هدف هماهنگی چند هاپی، تحقق خط لوله و حذف تصادم ها است

همانطور که در شکل 5 نشان داده شده است، به منظور دادن قابلیت فرستادن همزمان، بدون ایجاد

تصادم، به چندین گره، فرستنده ها باید حداقل دو هاپ از همدیگر دور باشند. این پدیده را می توان در

شکل 6 توضیح داد. در صورتی که گره 1 و گره 3 به صورت همزمان، چیزی را بفرستند، باعث ایجاد

تصادم استراق سمع در گره 2 می شود با این وجود در صورتی که گره 1 و گره 4، به صورت همزمان

چیزی بفرستند، هیچ گونه تصادمی اتفاق نمی افتد. با الهام گرفتن از این پدیده، نظریه اجتناب یا

تصادم ID(CAID) را معرفی کرده و CAID یکسان را به گروهی از گره ها که از یکدیگر، دو هاپ، فاصله

دارند، اختصاص می دهیم. از آن جایی که گره هایی که CAID یکسانی دارند، دو هاپ از هم فاصله

دارند در حقیقت فقط نیاز به CAID3 متفاوت، برای پوشش تمامی گره ها هستیم که در شکل 7 شاهد

این پدیده می باشید. الگوی CAID210 در کل شبکه منتشر می شود

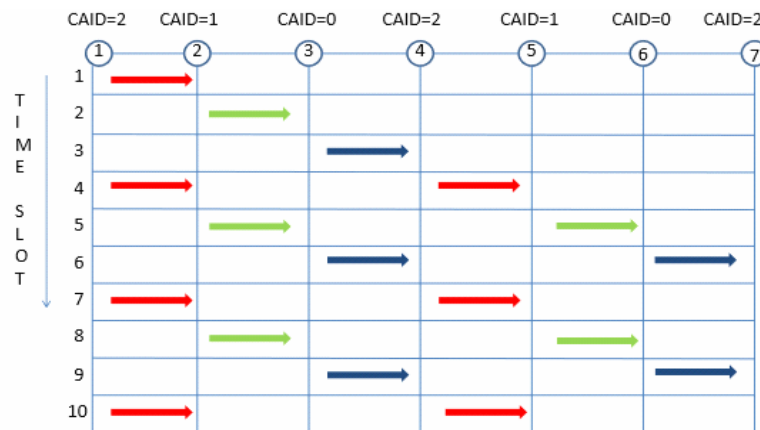
هم چنین یک روش پراکنده با آغاز منبع، برای اختصاص CAID به تمامی گره ها، در یک UAN چند

هاپی با توپولوژی رشته را پیشنهاد می کنیم. گره منبع با CAID برابر با 2 کار خود را شروع می کند.

هر گره CAID خود را به گره دسته پایین خود می فرستد. یک گره، پس از دریافت CAID از گره

بالادست خود، CAID دریافت شده را یک واحد کم می کند و از این مقدار به عنوان CAID خود

استفاده می کند. در صورتی که یک گره، CAID برابر با 0 را دریافت کرد، CAID خود را برابر با 2 قرار خواهد داد. این فرآیند زمانی که گره سینک، CAID خود را تعیین کند، ادامه پیدا می کند



شکل 8- هماهنگی چند هاپی

برپایه این ایده که گره هایی که CAID یکسانی دارند می توانند به صورت همزمان ارسال داشته باشند، یک اسلات زمانی را بر پایه طرح هماهنگی چند هاپی، برای CCRDT طراحی کرده ایم که در شکل 8 به تصویر کشیده شده است. در سه اسلات زمانی اول از آن جایی که گره 5، 4 و 6، هیچ گونه بلاک داده ای برای فرستادن ندارند، تنها یک گره در یک اسلات زمانی، عمل فرستادن را انجام می دهد. با شروع از چهارمین اسلات زمانی، دو گره با CAID یکسان وجود دارند که در اسلات زمانی ارسال می کنند، برای مثال در چهارمین اسلات زمانی، گره 4، بلاک 1-j را می فرستد در حالی که گره یک، بلاک 1-j را می فرستد. در صورتی که تعداد گره ها را در شبکه افزایش دهیم می توانیم سه گره یا بیشتر را داشته باشیم که در یک اسلات زمانی، به صورت همزمان، عمل ارسال را انجام دهند. به این صورت، در شبکه ای که 7 گره دارد (شکل 8)، در هر سه اسلات زمانی، بلاک جدیدی وجود خواهد داشت که به گره سینک ارسال می شود. بدون هماهنگی چند هاپی، یک بلاک جدید را می توان به ازای هر شش اسلات

زمانی، به گره سینک رساند. بنابراین با استفاده از طرح هماهنگی چند هاپی، توان عملیاتی انتها به انتها را می توان تا حدود زیادی بهبود بخشید و هیچ گونه تصادمی به وقوع نمی پیوندد.

قطعا بهره وری هماهنگی چند هاپی، وابستگی شدیدی به طول اسلات زمانی T_s دارد که برای هر گره، یکسان است. در صورتی که T_s ، بیش از حد کوچک باشد، یک بلاک از بسته های داده را نمی توان در دریافت کننده، در یک اسلات زمانی، بازیابی کرد. بنابراین در دو اسلات زمانی بعدی، دو گره بعدی، هیچ چیزی برای ارسال نخواهند داشت. این پدیده باعث اتلاف سه اسلات زمانی می شود. از طرف دیگر در صورتی که T_s بسیار بزرگ باشد، پس از این که بلاک داده، در دریافت کننده، بازیابی شد و ACK در فرستنده دریافت شد، زمان باقی مانده در یک اسلات زمانی نیز اتلاف خواهد شد.

همانطور که در بخش 3,2,2,2 بحث شد با برآورد نسبت کدنویسی و کدنویسی خطی رندوم (GF(256)، با احتمال بالا تضمین می کنیم که یک انتقال، از بسته های رمزنگاری شده، برای این که دریافت کننده بتواند یک بلاک از بسته های داده را بازیابی کند کافی است. بنابراین T_s را به گونه ای تعریف می کنیم که برای پوشش انتقال یک بلاک، از بسته های رمزنگاری شده و یک ACK، از دریافت کننده به فرستنده، طول داشته باشد. تاخیر انتقال بسته های رمزنگاری شده و ACK، شامل تاخیر انتقال و تاخیر انتشار می شود.

$$T_s = \frac{r \cdot K \cdot L \cdot 8}{R} + \frac{D \cdot 2}{V} + \frac{1 \cdot 8}{R} = \frac{8(rKL + 1)}{R} + \frac{2D}{V} \quad (10)$$

در این جا، r نسبت کدنویسی، L سایز بلاک داده، D طول بسته، r نرخ بیت آکوستیک مودم، d فاصله یک هاپ، V سرعت صوت در آب بوده و یک بسته ACK فقط یک بایت، اطلاعات دارد (هدر بسته را برای آسان تر کردن کار، حذف می کنیم)

نسبت کدنویسی r ، توسط PER تعیین می شود. از آن جایی که Ts، برای هر گره، یکسان است، Ts را با در نظر گرفتن بالاترین PER، در شبکه تعریف می کنیم. با این وجود از آن جایی که یک شبکه، با توپولوژی رشته همگون را در نظر می گیریم، PERها در تمامی هاپ ها به یکدیگر نزدیک هستند زیرا هیچ ایده ای درباره اطلاعات PER، در شبکه نداریم در پیاده سازی کنونی نیاز است تا برنامه را برای جمع آوری اطلاعات PER در چند هاپ، پیش اجرا کنیم تا بتوانیم Ts را تعیین نماییم. در آینده، برنامه ما توسعه طرح انطباقی تری است تا بتوانیم Ts را با پویایی، بر پایه اطلاعات PER در شبکه تنظیم نماییم

بنابراین با استفاده از هماهنگی چند هاپی، یک بلاک از بسته های داده را می توان در یک اسلات زمانی، بازیابی کرد. چندین گره را به صورت همزمان می توان در یک اسلات زمانی، بدون ایجاد تصادم فرستاد. به این صورت توان عملیاتی انتها به انتها را تا حدود زیادی میتوان بهبود بخشید

3,2,2,4- چرا ACKضمنی نه

هم چنین متوجه شدیم که یک گزینه طراحی دیگر، استفاده از ACKضمنی، به جای ACK صریح است. همانطور که در شکل 6 نشان دادیم، بلاک سربار J-1 در گره 3 را می توان به عنوان یک ACKضمنی در نظر گرفت و بنابراین گره 3 می داند که در آخرین اسلات زمانی، بلاک J-1 به صورت موفقیت آمیزی دریافت شده است. به این صورت می توانیم خود را از سربار تحمیلی، توسط ACK صریح، در CCRDT خلاص کنیم. با این وجود یک ACKضمنی، از نرخ از دست رفتن بسیار بالاتری به نسبت ACK صریح رنج می برد زیرا بلاک داده سربار استفاده شده به عنوان ACKضمنی، اغلب به نسبت ACK صریح، اغلب بسیار طولانی تر است (یک بلاک داده، اغلب چندصد بایت است در حالی که ACK صریح، تنها یک بایت است). در صورت از دست رفتن یک ACKضمنی، گره، انتقال را در آخرین اسلات زمانی، به عنوان یک شکست در نظر می گیرد. بنابراین در سه اسلات بعد، هنگامی که گره،

فرصت ارسال مجدد می یابد، آخرین بلاک داده را انتقال مجدد خواهد داد. در این موقعیت، سه اسلات زمانی، اتلاف می شود

می توانیم سربار ACK ضمنی و ACK صریح را برای تصمیم گیری درباره این که کدام گزینه برای طراحی، بهتر است، تحلیل کنیم. بیایید n را به عنوان تعداد هاپ ها، N را به عنوان تعداد بلاک ها، P_e را به عنوان احتمال از دست رفتن ACK صریح، P_i را به عنوان احتمال از دست رفتن ACK ضمنی ، T_{ack} به عنوان تاخیر انتقال برای یک ACK صریح با در نظر گرفتن تاخیر انتقال و تاخیر انتشار، T را به عنوان طول اسلات زمانی هنگام استفاده از ACK ضمنی، t_p به عنوان تاخیر انتشار داده برای یک بسته ACK یا داده در نظر بگیریم و K را برابر با سایز بلاک در نظر می گیریم . L برابر با طول بسته، R نرخ بیت آکوستیک مودم است. O_{exp} سربار ایجاد شده توسط ACK صریح و O_{imp} سربار ایجاد شده توسط ACK ضمنی می باشد.

$$O_{exp} = n \cdot N \cdot [(1 - P_e)T_{ack} + P_e \cdot 3(T + T_{ack})]$$

$$= nN[(1 - P_e)(t_p + \frac{8}{R}) + 3P_e(2t_p + \frac{8(KL + 1)}{R})] \quad (11)$$

$$O_{imp} = n \cdot N \cdot P_i \cdot 3T = nN3P_i(t_p + \frac{8KL}{R}) \quad (12)$$

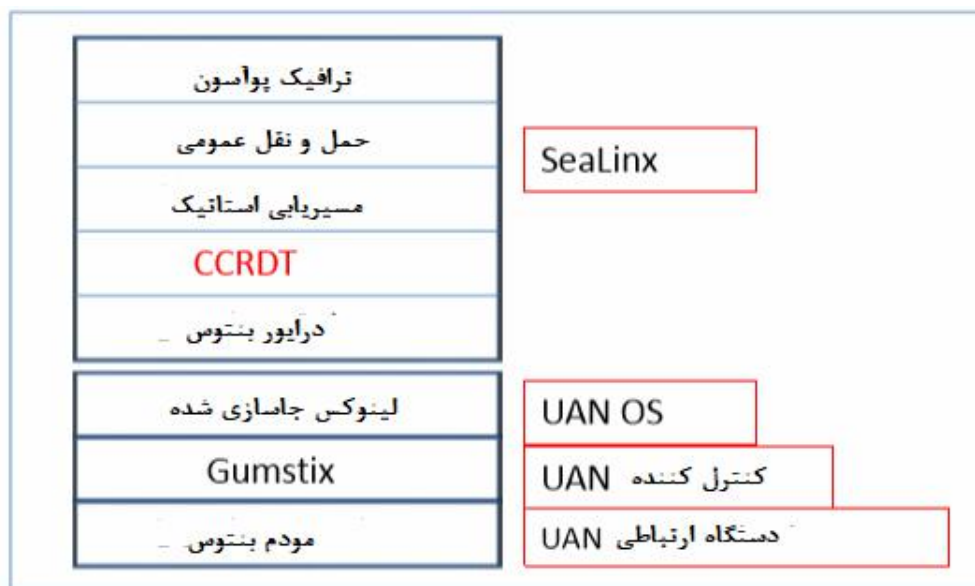
برای مثال، از پارامترهای عنوان شده ، بر پایه آنچه در آخرین آزمایشات میدانی خود اندازه گیری کردیم، استفاده می کنیم [78]: . را به دست می آوریم. مشاهده می کنیم که سربار ACK صریح از ACK ضمنی در شبکه زیرآبی عملی، کوچکتر بوده و بنابراین از ACK صریح در CRDT استفاده می کنیم.

3,2,3- پیاده سازه پروتکل

برای اعتبارسنجی بهتر و طراحی ارزیابی عملکرد، CCRDT را در گره های UAN حقیقی، پیاده سازی کردیم. در سیستم ما یک گره UAN، متشکل از یک UAN، گامستیک [42] است که به عنوان کنترلر، برای گره UAN عمل می کند و یک مودم تلداین بنتوس که ارتباط آکوستیک را برای دریافت و فرستادن بسته ها انجام می دهد

مجموعه نرم افزاری یک گره UAN، شامل لینوکس تعبیه شده و یک پشته پروتکل شبکه زیرآبی است: آکوا-NET [75]. آکوا-NET، یک پشته پروتکل لایه لایه است که شامل لایه فیزیکی، لایه لینک داده، لایه MAC، لایه مسیریابی، لایه انتقال و لایه کاربرد می شود. ما CCRDT را در لایه لینک داده پیاده سازی کردیم. برای سایر لایه ها از تولیدکننده ترافیک پواسون، در لایه کاربرد استفاده کردیم که ترافیک را بازه زمانی، میان هر دو بسته، ایجاد می کند. یک لایه انتقال ژنتریک مسیریابی استاتیک در لایه مسیریابی، یک لایه MAC در برادکست و یک درایور مودم بنتوس در لایه فیزیکی به کار رفته است. معماری کلی پیاده سازی CCRDT در شکل 9 نشان داده شده است

در CCRDT برای استفاده از هماهنگی چند هاپی، هر گره، یک شمارنده اسلات زمانی را نگه می دارد که در ابتدای اسلات زمانی، به اندازه یک واحد افزایش پیدا می کند: گره های با CAID برابر با 2 در اسلات زمانی اول می فرستند، گره هایی با CAID، یک در اسلات زمانی دوم می فرستند و به این صورت تا انتها پیش می رود



شکل 9- معماری کلی پیاده سازی CCRDT

3,2,4- نتایج آزمایشگاهی

آزمایشات گسترده ای را با استفاده از گره های UAN حقیقی، به منظور ارزیابی عملکرد و بهبود بخشی طراحی CCRDT انجام دادیم. در این بخش به تنظیمات آزمایش و هم چنین نتایج آزمایشگاهی می پردازیم.

تنظیمات سخت افزاری عادی در شکل 10 نشان داده شده است. سه نوع تلداین بنتوس داریم: مودم ATM920، مودم دِک باکس UDB9000 و مودم هوشمند SM75[26]. هر مودم، به یک گامستیکس وصل شده که بر روی آن، CCRDT اجرا می شود. مثال شکل 10 نشان دهنده یک شبیه سازی توپولوژی رشته ای و 5 گره با 4 مودم ATM92 و یک SM75 است.

پارامترهای شبکه به صورت زیر هستند مگر غیر از آن عنوان شود: شمارش هاپ، n برابر 4 است. سائز بلوک (K) 5 است (به این معنا که هر 5 بسته داده در یک بلاک، گروه بندی و رمزنگاری می شود). طول

بسته (L) 200 است. نرخ بیت آکوستیک مودم، 800bps، سایز پنجره لغزان برای آکوا-SARQ، یک است که بعداً در باره آن مباحثه خواهیم کرد.

عملکرد CCRDT (SERT) و آکوا-SARQ را بر پایه دو مقیاس، ارزیابی می‌کنیم: توان عملیاتی آنها به انتها که به عنوان تعداد بیت‌های دریافت شده در گره سینک، به ازای هر ثانیه تعریف می‌شود (bps) و سربار، به عنوان نسبت میان تعداد بسته‌های اضافه و تعداد بسته‌های داده منتقل شده در چندین هاپ، در شبکه تعریف می‌شود.



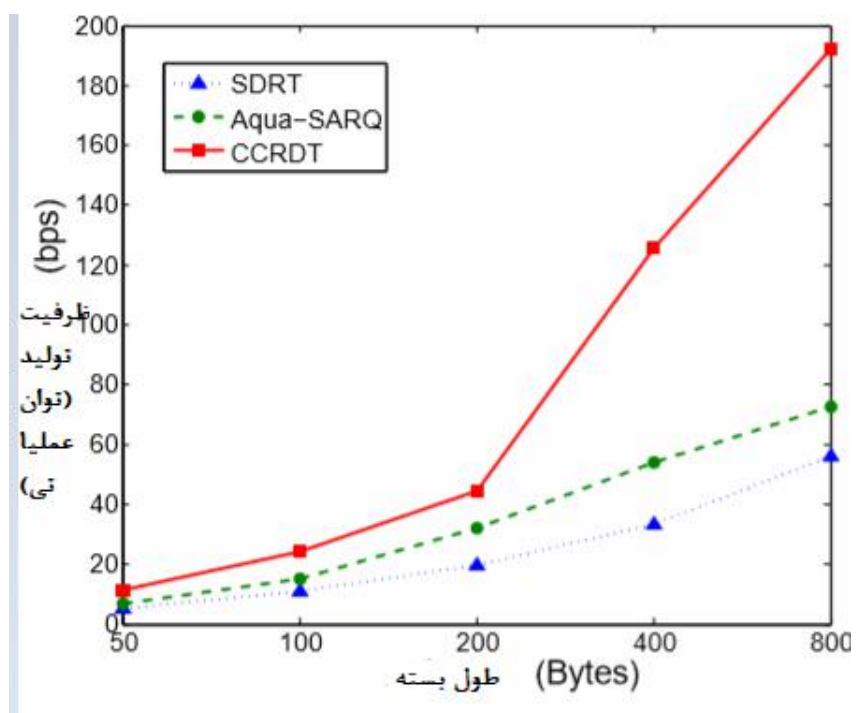
شکل 10- تنظیمات سخت افزار عادی در یک آزمایشگاه

3,2,4,1- تائیر طول بسته

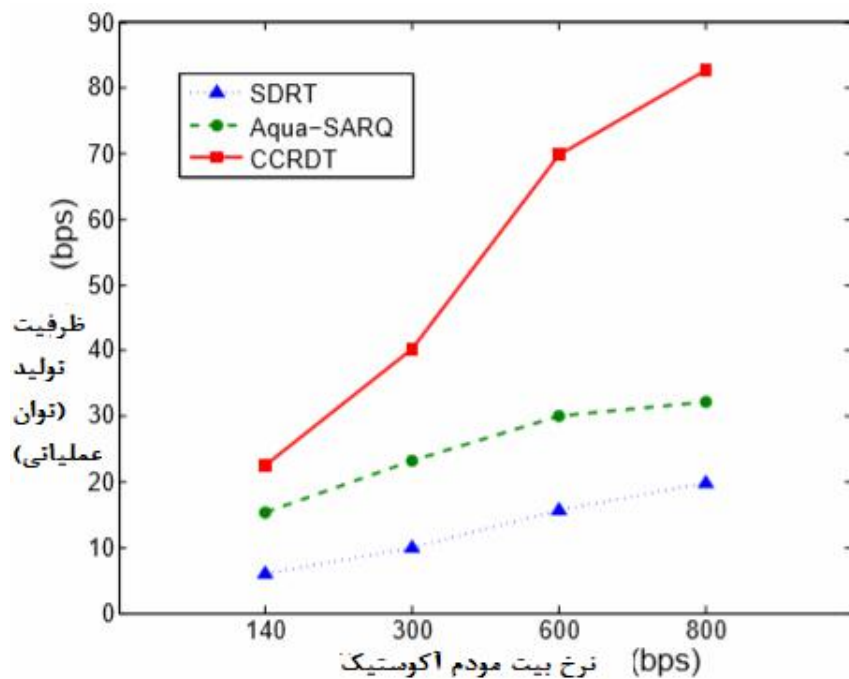
در این بخش، تائیر طول بسته را بر توان عملیاتی انتها به انتهای سه پروتکل بررسی می کنیم . نتایج در شکل 11 نشان داده شده اند. می بینیم که با طول بسته متفاوت (50-800)، CCRDT به توان عملیاتی انتها به انتهای بسیار بهتری ، به نسبت SARD و آکوا- SARQ می رسد. یک دلیل آن این است که CCRDT از برآورد نسبت کدنویسی و کدنویسی خطی رندوم GF(256)، برای تضمین این که یک انتقال از بسته های رمزنگاری، کافی است، استفاده می کند. این پدیده را می توان با استفاده از فایل گزار CCRDT اثبات کرد. این فایل نشان می دهد که هیچ انتقال مجددی در طول این آزمایش انجام نشده است. در نقطه مقابل، هم SARDT و آکوا SARQ از انتقال های مجدد رنج می برند و باعث تنزل توان عملیاتی، به دلیل از دست رفتن بسته می شود. یک دلیل دیگر این است که CCRDT، توانایی پیاده سازی طرح خط لوله را می دهد که به گره 1 و 4، توانایی فرستادن همزمان، در یک اسلات زمانی را می دهد (در یک توپولوژی رشته با 4 هاپ). در مقایسه با آکوا-SARQ انتها به انتها، خط لوله منجر به توان عملیاتی بزرگتر می شود. دلیل سوم این است که هماهنگی چندهپی CCRDT، به صورت کامل، تصادم ها را در شبکه از بین می برد. برخلاف SARDT، CCRDT از هیچ تصادم داده-ACK، فرستادن دریافت و استراق سمعی رنج نمی برد. هم چنین مشاهده کردیم که با افزایش طول بسته، توان عملیاتی انتها به انتهای هر سه پروتکل، بیشتر می شود که با توجه به این که بیت های بیشتری را به صورت همزمان می توان دریافت کرد، کاملاً معقول است. با این وجود متوجه می شویم که توان عملیاتی CCRDT، به نسبت SDRT و آکوا-SARQ، با افزایش طول بسته، رشد سریع تری دارند؛ دلیل این است که با افزایش طول بسته، PER در یک هاپ، بیشتر می شود. از آن جایی که CCRDT به نسبت SDRT و آکوا-SARQ با افزایش PER، انطباق بهتری پیدا می کند و دلیل آن، برآورد نسبت کدنویسی است، آسیب کمتری از PER بزرگتر می بیند.

3,2,4,2- تاثیر نرخ بیت آکوستیک مودم

تاخیر نرخ بیت آکوستیک مودم، بر توان عملیاتی انتها به انتها در شکل 12 نشان داده شده است. در این جا سایز بلوک 5، شمارش هاپ 4 و طول بسته 200 است. برای بار دیگر، CCRDT از SARD و آکوا-SARQ، عملکرد بهتری دارد و دلیل آن، کدنویسی خطی رندوم GF(256)، برآورد نسبت کدنویسی و هماهنگی چند هاپی است.



شکل 11- توان عملیاتی انتها به انتها با طول بسته متغیر



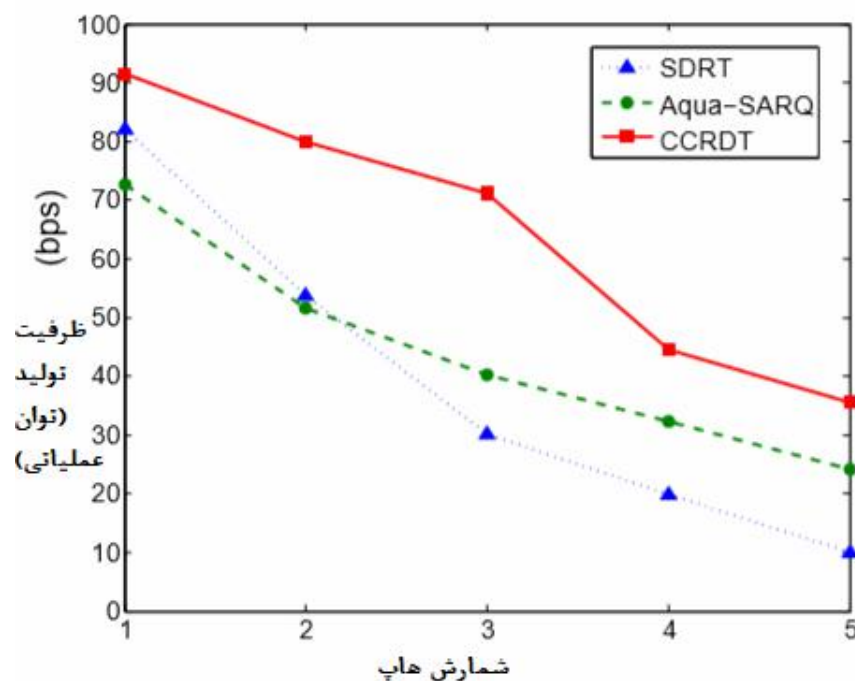
شکل 12- توان عملیاتی انتها به انتها با نرخ مودم متغیر

3,2,4,3- تاثیر شمارش هاپ

در این بخش، نحوه اثرگذاری شمارش هاپ، بر توان عملیاتی انتها به انتهای سه پروتکل را مطالعه می کنیم. نتایج در شکل 13 نشان داده شدند. می بینیم که CCRDT به بالاترین توان عملیاتی، در میان سه پروتکل، رسیده است. هم چنین با افزایش شمارش هاپ، توان عملیاتی هر سه پروتکل، کاهش پیدا میکند

پدیده دیگری که با آن مواجه شدیم این است که هنگامی که شمارش هاپ، کوچک باشد، SDRT، عملکرد بهتری به نسبت آکوا-SARQ دارد. با این وجود با افزایش شمارش هاپ، توان عملیاتی SDRT، سریع تر از دو پروتکل دیگر، دچار تنزل می شود. دلیل آن این است که با هاپ های بیشتر، SDRT، احتمال بیشتری وجود دارد که از تصادم ها دچار آسیب شود. در آکوا-SARQ، سایر پنجره لغزان برابر با یک است. به نظر می رسد که پنجره لغزان کوچکتر، منجر به توان عملیاتی کمتری می شود. از آن

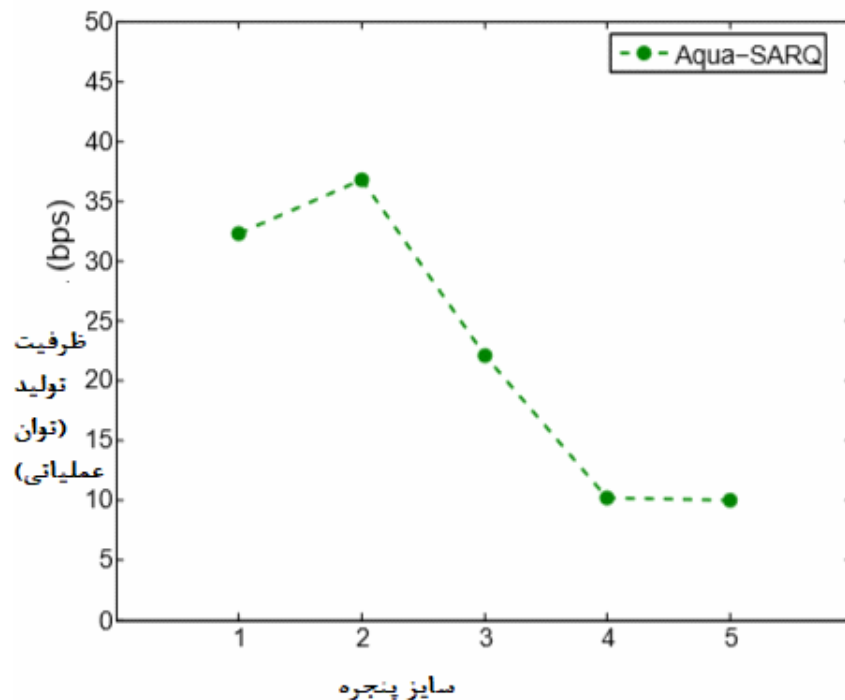
جایی که فقط یک بسته را می توان به گره سینک ، در یک زمان مشخص، تحویل داد با این وجود، برابر قرار دادن سائز پنجره لغزان با یک تصادم فرستادن-دریافت را حذف کرده و به توان عملیاتی بهتری منجر می شود. دلیل این که آکوا-SARQ، عملکرد بهتری به نسبت SDRT با شمارش هاپ بزرگتر دارد، همین است. این تحلیل را می توان با آزمایش های ما درباره نحوه اثرگذاری سائز پنجره لغزان، بر توان عملیاتی آکوا-SARQ اعتبارسنجی کرد که در شکل 14 آن را مشاهده می کنیم. می بینیم که سائز پنجره لغزان دو به توان عملیاتی کمی بیشتر به نسبت سایر پنجره لغزان یک می رسد. با این وجود پس از آن، توان عملیاتی انتها به انتها با افزایش سایر پنجره لغزان، به دلیل تصادم های ارسال-دریافت، کاهش پیدا می کند.



شکل 13-توان عملیاتی انتها به انتها با شمارش هاپ متغیر

3,2,4,4- سربار

در این بخش، سربار ایجاد شده توسط پروتکل های مختلف را مطالعه می کنیم. به دلیل محدودیت های فضا و زمان، تنها سربار CCRDT و آکوا-SARQ را در این جا مقایسه می کنیم. در شکل 15، نتایج نشان داده شده است. بسته های اضافی در CCRDT، شامل بسته ACK در هر هاپ و بسته های کدگذاری شده ضروری که فرستاده شده اند می باشد. بسته های اضافی در آکوا-SARQ، ACK های انتها به انتها هستند. در این آزمایش، سایز بلاک K برابر با 5 و طول بسته L برابر با 200 است. شمارش هاپ را از 1 تا 5 تغییر می دهیم



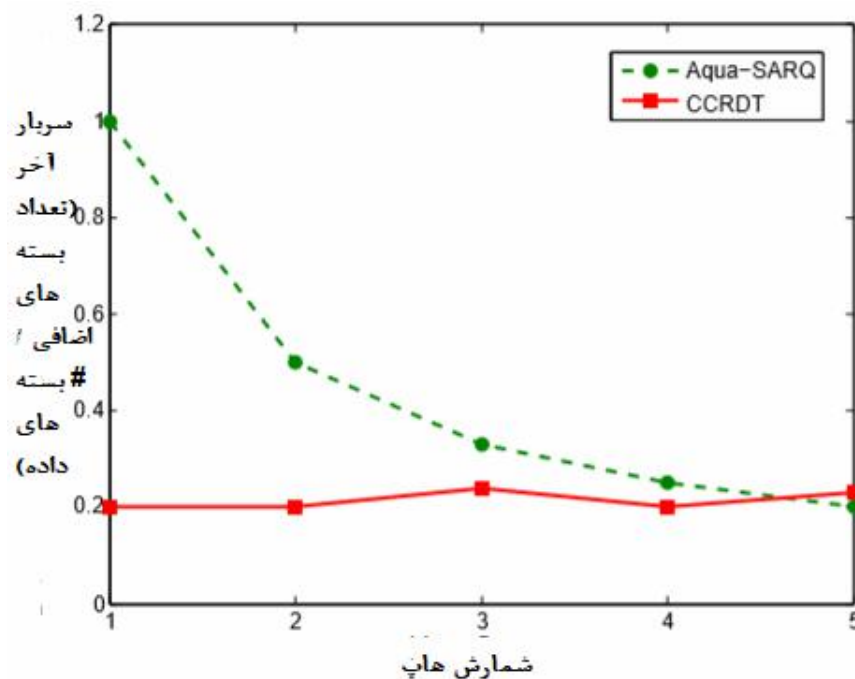
شکل 14- توان عملیاتی انتها به انتها با سایز پنجره متغیر

می بینیم که سربار آکوا-SARQ با افزایش شمارش هاپ، کاهش پیدا می کند. دلیل آن، این است که هر بسته داده منتقل شده، منجر به ACK می شود، در حالی که با 5 هاپ، 5 بسته داده منتقل شده

(یک بسته داده که از طریق 5 هاپ عبور می کند) مستلزم یک ACK است. درباره CCRDT باید بگوییم که با شمارش هاپهای مختلف، سربرار اغلب بدون تغییر باقی می ماند. یک دلیل آن این است که CCRDT از ACK هاپ به هاپ استفاده می کند و بنابراین هر بلاک از بسته های داده باعث می شود که یک ACK در هر هاپ استفاده شود. دلیل دیگر این است که تقریب نسبت کدنویسی و کدنویسی خطی رندوم GF(256)، میزان بسته های کدگذاری شده غیر ضروری را به حداقل می رساند. هم چنین می بینیم که با شمارش هاپ کوچکتر (کمتر از 5) CCRDT به سربرار کوچکتر به نسبت آکو SARQ می رسد

3,2,5- خلاصه

در این مطالعه، انتقال داده قابل اطمینان، برای UAN های چندهایی با توپولوژی رشته را بررسی می کنیم. با الهام گیری از مسائل مشاهده شده در پروتکل های موجودی که در سیستم های حقیقی، پیاده سازی شده اند، CCRDT را پیشنهاد می کنیم (یک پروتکل انتقال داده قابل اطمینان هماهنگ شده چندهایی مبتنی بر کدنویسی). برای انتقال داده به ازای هر هاپ، CCRDT از کدنویسی خطی رندوم GF(256) به دلیل قابلیت بازیابی داده قدرتمندش، هنگامی که سایز بلاک داده، کوچک است، استفاده می کند. یک طرح برآورد نسبت کدنویسی، برای تضمین انتقال داده در یک هاپ، پیشنهاد شده است. طرح هماهنگی چند هاپی، نه تنها برای ارائه قابلیت خط لوله، بلکه برای اجتناب از تصادم ها نیز طراحی شده است. CCRDT در سیستم های حقیقی، پیاده سازی شده است. آزمایشات گسترده نشان می دهند که CCRDT از دو پروتکل پیاده سازی شده و آزمایش شده در گذشته، عملکرد بهتری دارند.



شکل 15- سربار، با شمارش هاپ متغیر

3,3- انتقال داده قابل اطمینان چند هاپی انطباقی، مبتنی بر کد فواره ای

CCRDT در مقایسه با پروتکل های پیاده سازی شده و آزمایش شده در گره های حقیقی UAN، تا کنون به توان عملیاتی انتها به انتهای بهتری رسیده است. با این وجود این جمله فقط در UAN هایی با توپولوژی رشته ای همگون صادق است که توانایی پیاده سازی خط لوله انتقال را می دهد. با این وجود در برخی از UAN ها، کیفیت کانال در لینک های مختلف ممکن است تفاوت های فاحشی داشته باشد که باعث می شود CCRDT، اثربخش نباشد. برای حل این مسئله در این بخش، در بخش پیش رو، به ارزیابی عملکرد و طراحی انتقال داده قابل اطمینان چند هاپی، مبتنی بر کد فواره ای (FOCAR) می پردازیم. با حل یک مسئله بهینه سازی، FOCAR می تواند سایز بلاک داده بهینه را برای هر هاپ، مبتنی بر PER هر هاپ انتخاب نماید. در وهله بعد، ارزیابی عملکرد و طراحی پروتکل FOCAR را به صورت دقیق ارائه می دهیم.

3,2,1- مطالعات مربوطه

همانطور که در بخش 1,1 بررسی شد، UAN ها، خصیصه تاخیر انتشار طولانی، کانال محتمل به خطا و پهنای باند در دسترس کم دارند. علاوه بر آن، مودم های آکوستیک زیرآبی کنونی، نیمه دوپلکس هستند و تاخیر تبدیل حالت، میان انتقال و دریافت، محسوس است. این خصیصه ها باعث می شود که روش های مرسوم، برای انتقال داده قابل اطمینان، برای شبکه های آکوستیک زیرآبی، نامطلوب باشد. اول این که رویکردهای انتها به انتها، بهره وری کافی ندارند زیرا شکست در انتقال هر هاپ منجر به انتقال مجدد از منبع به سینک می شود که تا حدود زیادی باعث افزایش تاخیر انتها به انتهای میانگین می شود که در بخش 3,1,2 به آن پرداختیم. دوم این که با استفاده از روش های هاپ به هاپ، یک انتقال مجدد می تواند عملکرد کلی سیستم را تا حدود زیادی با توجه به تاخیر تبدیل حالت و انتشار طولانی، تنزل دهد. سوم این که بیشتر طرح ها، تاخیر انتقال حالت بزرگ را در مودم های آکوستیک لحاظ نکردند. در نهایت این طرح ها، برای سناریو چند هاپی، بهینه سازی نشده اند که در آن، وان هاپ با کیفیت لینک ضعیف می تواند لینک تنگنا در کل سیستم باشد و باعث شود هاپ های متعاقب با هاپ های بعدی، با کیفیت لینک خوب، دچار کمبود منابع شوند.

با الهام از ایده ها و تفکراتی که در بالا عنوان شده در این مطالعه، یک طرح جدید با نام انتقال داده قابل اطمینان چند هاپی انطباقی، مبتنی بر کد فواره ای (FOCAR) را برای UAN ها پیشنهاد می کنیم. FOCAR، یک طرح هیبریدی است که کدنویسی را با کدهای فواره ای انتقال مجدد، در صورت شکست، یکپارچه سازی می کند. کدهای فواره ای به دلیل قابلیت اصلاح خطای قدرتمند و خصیصه های بدون نرخ بودن، مورد استفاده قرار گرفتند. این یکپارچه سازی با طرح انتقال مجدد مبتنی بر تکرار انتخابی می تواند تاخیر را در یک هاپ، به شدت کاهش دهد. علاوه بر آن، FOCAR، یک چارچوب بهینه سازی را برای کمینه کردن تاخیر انتها به انتها در چندین هاپ پیشنهاد می کند. علاوه

بر آن، FOCAR، تاخیر تبدیل حالت طولانی را در مودم های آکوستیک کنونی لحاظ می کند که توسط مطالعات قبلی از آن ها چشم پوشی شده است

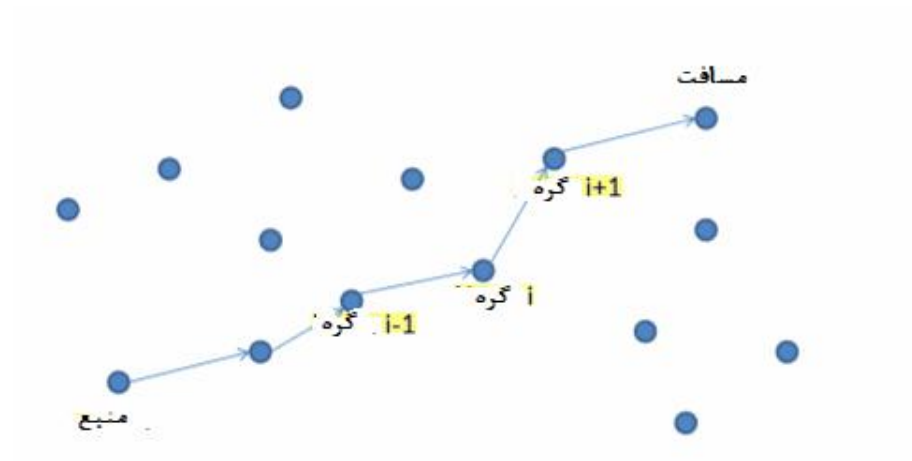
3,3,2- توصیف پروتکل

در این بخش در ابتدا کد فواره ای را معرفی کرده و مدل شبکه را ارائه می دهیم سپس پروتکل FOCAR را به دقت توصیف می کنیم

3,3,2,1- کدهای فواره ای

کدهای فواره ای ، کدهای رکورد شکن، برای کانال هایی هستند که حذف شدگی دارند (مثل از دست رفتن بسته در یک شبکه ارتباطاتی)[84]. این کدها منجر به روش انقلابی دریافت داده، مبتنی بر عدم جریان، در شبکه های قابل اطمینان شده اند که کدهای LT و رپتور، در این زمینه پیش رو بودند [76].

ایده پایه کد دیجیتال فواره ای به صورت زیر است. انکودر، یک فواره است که منبع بی پایانی از قطرات آب را تولید می کند (بسته های کدگذاری شده). فرض کنید فایل ممنوع اولیه، سایز K بسته را دارد و هر قطره شامل یک بسته کدگذاری شده می شود. دریافت کننده ای که بخواهد فایل کدگذاری شده را دریافت کند، یک ظرف را برای جمع کردن قطرات نگه می دارد تا زمانی که تعداد قطرات در ظرف K پریم باشد (که از K کمی بزرگتر است) و سپس می تواند فایل اولیه را بازیابی کند. کدهای فواره ای، بدون نرخ هستند از این لحاظ که تعداد بسته های کدگذاری شده ای که می تواند از یک پیام منبع تولید شود، به صورت بالقوه بینهایت است. علاوه بر آن، بسته های کدگذاری شده تولیدی را می توان در حال اجرا تعیین کرد. یک خصیصه جذاب کدهای فواره ای این است که می دانیم الگوریتم های n کدینگ بهره وری دارد



شکل 16- تشریح مدل شبکه

در این مطالعه از کدهای فواره ای برای کدنویسی FOCAR استفاده می شود. از یک طرف، کدهای فواره ای ، الگوریتم های کدگذاری و کدگشایی بهره وری با پیچیدگی خطی دارند که می تواند به کاهش تاخیر انتها به انتهای کلی کمک کند . از طرف دیگر، میزان افزونگی را می توان در حال اجرا تعیین کرد که پیاده سازی تکرار انتخابی را بسیار تسهیل کرده و به صورت مستقیم در تاخیر انتها به انتهای کم و بهره وری بالای انرژی، نقش دارد.

منابع و مراجع:

- [1] I. F. Akyildiz, D. Pompili, and T. Melodia, "Challenges for efficient communication in underwater acoustic sensor networks," *ACM SIGBED Review, Special issue on embedded sensor networks and wireless computing*, vol. 1, no. 2, pp. 3–8, Jul 2021.
- [2] J. Partan, J. Kurose, and B. N. Levine, "A survey of practical issues in underwater networks," in *Proceedings of ACM WUWNeT*, Sep 2016, pp. 17–24.
- [3] J. Heidemann, Y. Li, A. Syed, J. Wills, and W. Ye, "Research challenges and applications for underwater sensor networking," in *Proceedings of IEEE WCNC*, Apr 2020, pp. 228–235.
- [4] E. M. Sozer, M. Stojanovic, and J. G. Proakis, "Underwater acoustic networks," *IEEE JOURNAL OF OCEANIC ENGINEERING*, vol. 25, no. 1, Jan 2020.
- [5] J.-H. Cui, J. Kong, M. Gerla, and S. Zhou, "Challenges: building scalable mobile underwater wireless sensor networks for aquatic applications," *IEEE*

Network, Special Issue on Wireless Sensor Networking, vol. 20, no. 3, pp. 12–18, May 2016.

[6] M. C. Vuran and I. F. Akyildiz, “Cross-layer packet size optimization for wireless terrestrial, underwater, and underground sensor networks,” in Proceedings of IEEE INFOCOM, Apr 2018, pp. 226–230.

[7] X. Xu, Z. Wang, S. Zhou, and L. Wan, “Parameterizing both path amplitude and delay variations of underwater acoustic channels for block decoding of orthogonal frequency division multiplexing,” in The Journal of the Acoustical Society of America, vol. 131, no. 6, 2019, pp. 4672–4679.

[8] Z. Peng, S. Le, M. Zuba, H. Mo, H. Zhou, J.-H. Cui, S. Zhou, Z. Jiang, and J. A. Schindall, “Field test experience of an underwater wireless network in the atlantic ocean,” in Proceedings of IEEE/MTS OCEANS, Bergen, Norway, Jun 2019.

[9] H. Mo, S. Le, Z. Peng, Z. Shi, and J.-H. Cui, “Aqua-os: An operating system for underwater acoustic networks,” in Proceedings of IEEE WASA, Yellow Mountain, China, Aug 2018.

[10] Z. Peng, H. Mo, J. Liu, Z. Wang, H. Zhou, X. Xu, S. Le, Y. Zhu, J.-H. Cui, Z. Shi, and S. Zhou, “NAMS: A networked acoustic modem system for underwater applications,” in Proceedings of IEEE/MTS OCEANS, Kona, HI, USA, Sep 2021.

- [11] V. Chandrasekhar, W. K. Seah, Y. S. Choo, and H. V. Ee, "Localization in underwater sensor networks - survey and challenges," in Proceedings of ACM WUWNeT, Sep 2016.
- [12] S. Le, H. Mo, Z. Peng, M. Zuba, J.-H. Cui, Z. Shi, and S. Zhou, "Multimedia messaging with underwater acoustic wireless networks," in Proceedings of ACM MobiCom (Demo), Las Vegas, NV, US, Sep 2018.
- [13] M. Zuba, M. Fagan, J.-H. Cui, and Z. Shi, "A vulnerability study of geographic routing in underwater acoustic networks," in Proceedings of the First IEEE Conference on Communications and Network Security (CNS), Washington, D.C., USA, 2019.
- [14] A. A. Syed, W. Ye, and J. Heidemann, "T-lohi: A new class of mac protocols for underwater acoustic sensor networks," in Proceedings of IEEE INFOCOM, 2018, pp. 231–240.
- [15] U. Lee, P. Wang, Y. Noh, L. Vieira, M. Gerla, and J.-H. Cui, "Pressure routing for underwater sensor networks," in Proceedings of IEEE INFOCOM, Mar 2010, pp. 1–9.
- [16] W. Cheng, A. Teymorian, L. Ma, X. Cheng, X. Lu, and Z. Lu, "Underwater localization in sparse 3d acoustic sensor networks," in Proceedings of IEEE INFOCOM, Apr 2018.
- [17] Y. Su, Y. Zhu, H. Mo, J.-H. Cui, and Z. Jin, "Upc-mac: A power control mac protocol for underwater sensor networks," in Proceedings of IEEE WASA, Zhangjiajie, China,

Aug 2020

[18] M. Zuba, A. Byrd, C. Villa, C. Fedge, S. Le, H. Mo, Z. Peng, J. Che, and J.-H. Cui, "A control system for ofdm networked autonomous underwater vehicles," in Proceedings of IEEE/MTS OCEANS, Hampton Roads, VA, US, Oct 2018.

[19] M. Zuba, Z. Shi, Z. Peng, and J.-H. Cui, "Launching Denial-of-Service jamming attacks in underwater sensor networks," in Proceedings of the ACM WUWNeT, 2019.

[20] M. Zuba, M. Fagan, Z. Shi, and J.-H. Cui, "A resilient pressure routing scheme for underwater acoustic networks," in Proceedings of IEEE GLOBECOM, Austin, TX, US, 2018.

[21] J. Liu, Z. Wang, Z. Peng, M. Zuba, J.-H. Cui, and S. Zhou, "Tsmu: A time synchronization scheme for mobile underwater sensor networks," in Proceedings of IEEE GLOBECOM, Houston, TX, US, Dec 2017.

[22] A. F. Harris, M. Stojanovic, and M. Zorzi, "Idle-time energy savings through wake-up modes in underwater acoustic networks," *Ad Hoc Networks*, vol. 7, no. 4, pp. 770–777, 2019.

[23] J. Liu, Z. Zhou, Z. Peng, and J.-H. Cui, "Mobi-sync: efficient time synchronization for mobile underwater sensor networks," in Proceedings of IEEE GLOBECOM, Miami, FL, US, Dec 2019.

- [24] R. Jurdak, C. V. Lopes, and P. Baldi, "Battery lifetime estimation and optimization for underwater sensor networks," IEEE Sensor Network Operations, 2019.
- [25] S. Singh and C. S. Raghavendra, "PAMAS—Power aware multi-access protocol with signalling for ad hoc networks," SIGCOMM Computer Communication Review, vol. 28, no. 3, pp. 5–262018.
- [26] "Benthos acoustic modem," in <http://www.benthos.com>.
- [27] J. Wills, W. Ye, and J. Heidemann, "Low-power acoustic modem for dense underwater sensor networks," in Proceedings of ACM WUWNeT, Los Angeles, CA, 2021.
- [28] Y. Sankarasubramainam, O. Akan, and I. Akyildiz, "ESRT: Event-to-sink reliable transport in wireless sensor networks," in Proceedings of ACM Mobihoc, Jun 2020
- [29] F. Stann and J. Heidemann, "RMST: Reliable data transport in sensor networks," in Proceedings of the IEEE International Workshop on Sensor Net Protocols and Applications, May 2020.
- [30] J. Huang, S. Zhou, and P. Willett, "Nonbinary LDPC coding for multicarrier underwater acoustic communication," IEEE Journals on Selected Area in Communications, vol. 26, no. 9, pp. 1684–1696, Dec 2017.

[31] Z. Guo, P. Xie, J.-H. Cui, and B. Wang, "On applying network coding to underwater sensor networks," in Proceedings of ACM WUWNeT, Sep 2018, pp. 279–288.

[32] O. Schofield, T. Bergmann, P. Bissett, J. F. Grassle, D. B. Haidvogel, J. Kohut, M. Moline, and S. M. Glenn, "The long-term ecosystem observatory: An integrated coastal observing system," IEEE Journal of Oceanic Engineering, vol. 27, no. 2, pp. 146–154, 2022.

[33] A. Kaya and S. Yauchi, "An acoustic communication system for subsea robot," in Proceedings of IEEE/MTS OCEANS, 2019

[34] J. Liu, Z. Wang, Z. Peng, J.-H. Cui, and L. Fiondella, "Suave: Swarm underwater autonomous vehicle localization," in Proceedings of IEEE INFOCOM, Toronto, Canada, 2014.

[35] H. Zhang, A. Arora, Y. ri Choi, and M. Gouda, "Reliable bursty convergecast in wireless sensor networks," in Proceedings of ACM Mobihoc, May 2018.

[36] P. Xie and J.-H. Cui, "An fec-based reliable data transport protocol for underwater sensor networks," in Proceedings of ICCCN, Aug 2017, pp. 747 – 753.

[37] H. Mo, A. C. Mingir, H. Alhumyani, Y. Albayram, and J.-H. Cui, "Uw-harq: An underwater hybrid arq scheme: Design, implementation and initial

test," in Proceedings of IEEE/MTS OCEANS, Hampton Roads, VA, USA, Sep 2018.

[38] J. L. Hill and D. E. Culler, "Mica: A wireless platform for deeply embedded networks," IEEE Micro, vol. 22, no. 6, pp. 12–24, 2018.

[39] J. M. Rabaey, M. J. Ammer, J. L. da Silva, D. Patel, and S. Roundy, "Picoradio supports ad hoc ultra-low power wireless networking," IEEE Computer, vol. 33, no. 7, pp. 42–48, 2018.

[40] H. Mo, Z. Zhou, M. Zuba, Z. Peng, J.-H. Cui, and Y. Shu, "Practical coding-based multi-hop reliable data transfer for underwater acoustic networks," in Proceedings of IEEE GLOBECOM, Dec 2019.

[41] H. Mo, Z. Peng, Z. Zhou, M. Zuba, Z. Jiang, and J.-H. Cui, "Coding based multi-hop coordinated reliable data transfer for underwater acoustic networks: Design, implementation and tests," in Proceedings of IEEE GLOBECOM, Dec 2020.

[42] "Gumstix inc." in <http://www.gumstix.com>.

[43] Z. Zhou, H. Mo, Y. Zhu, Z. Peng, J. Huang, and J.-H. Cui, "Fountain code based adaptive multi-hop reliable data transfer for underwater acoustic networks," in Proceedings of IEEE ICC, Workshop on Radar and Sonar Networks, Jun 2022.

[44] H. Mo, Z. Peng, Z. Zhou, and J.-H. Cui, "A two-phase broadcast scheme for underwater acoustic networks," in Proceedings of IEEE GLOBECOM, Dec. 2018.

[45] H. Mo, L. Pu, Y. Zhu, Z. Peng, Z. Jiang, and J.-H. Cui, "Evaluating selective arq and slotted handshake based access in real world underwater acoustic networks," in Proceedings of IEEE WASA, Aug. 2019.

[46] C.-Y. Wan, A. T. Campbell, and L. Krishnamurthy, "PSFQ: A reliable transport protocol for wireless sensor networks," in Proceedings of IEEE WSNA, Sep 2018.

[47] J. Paek and R. Govindan, "Rcrt: Rate-controlled reliable transport for wireless sensor networks," in Proceedings of ACM SenSys, Sydney, Australia, 2017.

[48] S. Kim, R. Fonseca, P. Dutta, and A. Tavakoli, "Flush: A reliable bulk transport protocol for multihop wireless networks," in Proceedings of ACM SenSys, Sydney, Australia, 2022.

[49] K. Sundaresan, V. Anantharaman, H.-Y. Hsieh, and R. Sivakumar, "Atp: A reliable transport protocol for ad-hoc networks," in Proceedings of ACM MobiHoc, 2021.

[50] Q. Zhang and S. A. Kassarm, "Hybrid arq with selective combining for fading channels," IEEE Journals on Selected Area in Communications, vol. 17, no. 5, pp. 867–880, Jul 2019.

[51] G. Caire and D. Tuninetti, "The throughput of hybrid-arq protocols for the Gaussian collision channel," *IEEE Transactions on Information Theory*, vol. 47, pp. 1971–1988,

Jul 2020

[52] B. Zhao and M. C. Valenti, "Practical relay networks: A generalization of hybrid- arq," *IEEE Journals on Selected Area in Communications*, vol. 23, no. 1, pp. 7–18, Jan 2021.

[53] A. Hagedorn, D. Starobinski, and A. Trachtenberg, "Rateless deluge: Over-the-air programming of wireless sensor networks using random linear codes," in *Proceedings of ACM IPSN*, 2018.

[54] B. Williams and T. Camp, "Comparison of broadcasting techniques for mobile ad hoc networks," in *Proceedings of ACM MobiHoc*, Jun 2019.

[55] E. Pagani and G. P. Rossi, "Reliable broadcast in mobile multihop packet networks," in *Proceedings of ACM MobiCom*, Budapest, Hungary, Sep 2017.

[56] Y.-C. Tseng, S.-Y. Ni, and Y. shyan Chen, "The broadcast storm problem in a mobile Ad Hoc network," in *Proceedings of ACM MobiCom*, Aug 2019

[57] J. Lipman, H. Liu, and I. Stojmenovic, *Broadcast in Ad Hoc Networks*. Springer, 2009.

[58] D. C. Yoav Sasson and A. Schiper, "Probabilistic broadcast for flooding in wireless mobile ad hoc networks," in *Proceedings of IEEE WCNC*, Mar 2018.

[59] H. Lim and C. Kim, "Multicast tree construction and flooding in wireless ad hoc

networks," in Proceedings of ACM MSWIM, Aug 2020.

[60] Q. A, "Multipoint relaying: An efficient technique for flooding in mobile wireless networks," in proceedings of the 35th Annual Hawaii International Conference on System Sciences, Jan 2021.

[61] M. Stojanovic, "Optimization of a data link protocol for an underwater acoustic channel," in Proceedings of IEEE/MTS OCEANS, Jun. 2019.

[62] P. Casari, M. Rossi, and M. Zorzi, "Fountain codes and their application to broadcasting in underwater networks: Performance modeling and relevant tradeoffs," in Proceedings of ACM WUWNeT, San Francisco, CA, USA, Sep 2018.

[63] R. Cao and L. Yang, "Reliable transport and storage protocol with fountain codes for underwater acoustic sensor networks," in Proceedings of ACM WUWNeT, Woods Hole, MA, US, 2018.

[64] D. E. Lucani, M. Medard, and M. Stojanovic, "Broadcasting in time-division duplexing networks: A random linear network coding approach," in Proceedings of Workshop on Network Coding, Theory, and Applications, Jun 2009.

[65] Z. Guo, J. Huang, B. Wang, J.-H. Cui, S. Zhou, and P. Willett, "A practical joint networkchannel coding scheme for reliable communication in wireless networks," in Proceedings of ACM Mobihoc, Sep 2009, pp. 279–288.

[66] C. Manville, A. Miyajan, A. Alharbi, H. Mo, M. Zuba, and J.-H. Cui, "Network coding in underwater sensor networks," in Proceedings of IEEE/MTS OCEANS, Hampton Roads, VA, USA, Sep 2021.

[67] W. C. Lim, B. Kannan, and T. T. Tjhung, "Joint channel estimation and OFDM synchronisation in multipath fading," in Proceedings of IEEE ICC, 2014.

[68] J. Liu, X. Han, M. Al-Bzoor, M. Zuba, J.-H. Cui, R. Ammar, and S. Rajasekaran, "PADP: Prediction assisted dynamic surface gateway placement for mobile underwater networks," in Proceedings of IEEE ISCC, Cappadocia, Turkey, 2021

[69] J. Rice, "Seaweb acoustic communication and navigation networks," in Proceedings of the International Conference on Underwater Acoustic Measurements: Technologies and Results, Heraklion, Crete, Greece, 2018.

[70] X. Xu, S. Zhou, A. K. Morozov, and J. C. Preisig, "Per-survivor processing for underwater acoustic communications with direct-sequence spread spectrum," in The Journal of the Acoustical Society of America, vol. 133, no. 1, 2018, pp. 27 462–2754.

[71] P. Carroll, S. Zhou, H. Zhou, X. Xu, J.-H. Cui, and P. Willett, "Underwater localization

and tracking of physical systems," in The Journal of Electrical and Computer Engineering, 2020.

- [72] L. Wan, H. Zhou, X. Xu, Y. Huang, S. Zhou, Z. Shi, and J.-H. Cui, "Adaptive modulation and coding for underwater acoustic OFDM," IEEE Journal of Oceanic Engineering, 2014.
- [73] P. Xie, J.-H. Cui, and L. Lao, "VBF: Vector-Based forwarding protocol for underwater sensor networks," in Proceedings IFIP Networking, Coimbra, Portugal, 2020, pp. 228– 235.
- [74] H.-P. Tan, W. Seah, and L. Doyle, "A multi-hop arq protocol for underwater acoustic networks," in Proceedings of IEEE/MTS OCEANS, Jun 2022.
- [75] Z. Peng, Z. Zhou, J.-H. Cui, and Z. Shi, "Aqua-net: An underwater sensor network architecture - design and implementation," in Proceedings of IEEE/MTS OCEANS, 2020.
- [76] A. Shokrollahi, "Raptor codes," IEEE Transactions on Information Theory, vol. 52, no. 6, pp. 2551–2567, Jul 2018.
- [77] M. Luby, "Lt codes," in The 43rd Annual IEEE Symposium on Foundations of Computer Science, 2022, pp. 271–280.
- [78] Z. Peng, S. Le, M. Zuba, H. Mo, Y. Zhu, L. Pu, J. Liu, and J.-H. Cui, "Aqua-tune: A testbed for underwater networks," in Proceedings of IEEE/MTS OCEANS, Santander, Spain, Jun 2021.

[79] D. Lucani, M. Medard, and M. Stojanovic, "Network coding schemes for underwater networks: The benefits of implicit acknowledgement," in Proceedings of ACM WUWNeT, Sep 2019.

[80] M. Molins and M. Stojanovic, "Slotted fama: a mac protocol for underwater acoustic networks," in Proceedings of IEEE/MTS OCEANS, Sep 2019.

[81] V. Rajendran, K. Obraczka, and J. GarciaLunaAceves, "Energy efficient, collision free medium access control for wireless sensor networks," in Proceedings of ACM SenSys, Nov. 2020.

[82] S. Wicker and V. Bhargava, An introduction to Reed-Solomon codes.

[83] L. Freitag, M. Grund, S. singh, J. Partan, P. Koski, and K. Ball, "The WHIO micromodem: An acoustic communications and navigation system for multiple platforms," in Proceedings of IEEE/MTS OCEANS, vol. 2, Jun 2015, pp. 1086–1092.

[84] J. W. Byers, M. Luby, and M. Mitzenmacher, "A digital fountain approach to asynchronous reliable multicast," IEEE Journal on Selected Areas in Communications, vol. 20, no. 8, pp. 1528–1540, 2012.