کنترل ازدحام مبتنی بر تخمین در شبکه‌های موردی بی‌سیم

شهرام جمالی[۱]، دانشیار، توفان سماپور[۲]، مربی
۱- گروه مهندسی کامپیوتر- دانشگاه محقق اردبیلی- اردبیل- ایران– jamali@iust.ac.ir
۲- مرکز آموزش عالی علمی کاربردی جهاد دانشگاهی اردبیل- اردبیل- ایران– ce@samapour.ir

چکیده: پروتکل کنترل انتقال (TCP) و بیشتر روش‌های مبتنی بر آن بدون توجه به علت گم شدن بسته، آن را نشانه بوجود آمدن ازدحام در شبکه دانسته و الگوریتم‌های کنترل ازدحام را اجرا می‌نمایند. این مساله سرمنشا کاهش کارایی این پروتکل در شبکه‌های موردی بی‌سیم است. در این مقاله روشی مبتنی بر تخمین برای تنظیم اندازه پنجره ارسال و کنترل ازدحام، در شبکه‌های بی‌سیم موردی ارائه شده است. در روش ارائه شده فرستنده با استفاده از فاصله زمانی میان دو ACK متوالی تخمینی از وضعیت ترافیکی شبکه بدست می‌آورد. نتایج شبیه‌سازی در نرم‌افزار NS-2 نشان می‌دهد که کارایی این روش از نقطه نظر توان عملیاتی در مجموع بهتر از روش‌های موجود می‌باشد. هم چنین این روش نرخ از بین رفتن بسته‌ها و تاخیر و سربار در شبکه را تا حدود زیادی کاهش می‌دهد.

واژه‌های کلیدی: پروتکل کنترل انتقال، کنترل ازدحام، شبکه‌های موردی بی‌سیم

Estimation Based Congestion Control in Wireless Ad hoc Networks

S. Jamali[1], T. Samapour[2]
1- Department of Computer Engineering, University of Mohaghegh Ardabili, Ardabil, Iran
2- Institute of Applied Science Technology Jahad Daneshgahi, Ardabil Branch, Ardabil, Iran

Abstract: **Transmission Control Protocol (TCP) and other its well-kNown implementations without attend to reason of packet loss, consider it as a sign of network congestion and run congestion control commands. This issue is source of performance degradation for TCP in Wireless Ad Hoc networks (WANETs). If we could distinguish between congestion loss and link error loss or use another strategy for changing congestion window, performance would be improved. In this paper, we are presenting a new mechanism to enhance the performance of TCP in WANETs namely TCP-WANET. In particular, we employ an estimation-based method for setting of congestion window size. TCP-WANET is based on InterACK, which is the time between two consecutive ACKs. Extensive packet level simulations, implemented by using ns-2, show that usage of such congestion control protocol can lead to improved performance comparing to the standard TCP protocol.**

Keywords: Transmission Control Protocol (TCP), Estimation Based Congestion Control, Wireless Ad hoc Network (WANET);

تاریخ ارسال مقاله: ۹۰/۱۱/۱۸
تاریخ اصلاح مقاله: ۹۱/۲/۱۸
تاریخ پذیرش مقاله: ۹۲/۶/۳
نام نویسنده‌ی مسئول: توفان سماپور
نشانی نویسنده‌ی مسئول: ایران- اردبیل- بلوار معلم- روبروی پل استخر شهرداری- مرکز آموزش عالی علمی کاربردی جهاد دانشگاهی اردبیل

۱- مقدمه

شبکه‌های موردی بی‌سیم در سالیان اخیر بسیار محبوب گشته‌اند به طوری که تقریباً در هر جایی مورد استفاده قرار می‌گیرند. برای مثال این نوع شبکه‌ها در شرایط مختلفی نظیر جنگ، عملیات امداد و نجات، همایش‌ها و به طور کلی جاهایی که زیرساخت شبکه‌های ثابت در دسترس نیست و برپایی سریع یک شبکه ارتباطی مورد نیاز است کاربردهای فراوان دارند. پروتکل کنترل انتقال[1] [۱] در شبکه‌های بی‌سیم عملکرد خیلی خوبی ندارند. این بدان دلیل است که این پروتکل هرگاه که گم‌شدن بسته‌ای را تشخیص داد آن را نشانه‌ی وجود ازدحام در شبکه دانسته و متناسب با آن نرخ ارسال را در سمت فرستنده کاهش می‌دهد. این در حالی است که گم‌شدن بسته در شبکه‌های بی‌سیم، علاوه بر ازدحام می‌تواند دلیل دیگری نظیر رخ دادن خطا در رسانه انتقال داشته باشد.

از این رو در این مقاله، مشکلات مرتبط با رخ دادن خطا در شبکه‌های موردی بی‌سیم مورد بررسی قرار گرفته و با تشریح تاثیر خطاهای مربوط به رسانه بر کارایی پروتکل کنترل انتقال، نشان داده می‌شود که با ارائه یک روش مبتنی بر تخمین در شبکه‌های بی‌سیم، می‌توان کارایی این پروتکل را در این نوع از شبکه‌ها بهبود داد. به عبارت دیگر روش ارائه شده، از الگوریتمی برای تخمین پهنای باند شبکه استفاده می‌نماید و به فرستنده کمک می‌کند تا با توجه به شرایط ترافیکی شبکه، پنجره ازدحام و متناسب با آن نرخ ارسال خود را تنظیم نماید. سازوکار بکار گرفته در این روش برای تشخیص نوع خطا، کمک می‌کند تا این روش بین خطاهای مرتبط با رسانه بی‌سیم و مرتبط با ازدحام تمایز قائل شده و متناسب با هر یک واکنش مناسبی نشان دهد.

این مقاله در ادامه به صورت زیر ادامه می‌یابد. تعریف مساله ازدحام و ویژگی‌های شبکه‌های بی‌سیم در شبکه در بخش ۲ بیان شده است. بررسی و مروری بر کارهای گذشته‌ای که در این زمینه انجام پذیرفته در بخش۳ آمده است. بخش۴ ، در برگیرنده شرح کار روش پیشنهادی است. بخش ۵ نیز به ارزیابی روش پیشنهادی و ارائه نتایج حاصل از شبیه‌سازی اختصاص یافته است و سرانجام، در بخش ۶ جمع‌بندی و معرفی کارهای آینده آمده است.

۲- ماهیت مساله ازدحام

هرگاه که نیاز بیش از ظرفیت باشد ازدحام بوجود می‌آید. در شبکه‌های کامپیوتری نیز هنگامی که بسته‌های بسیار زیادی در قسمتی از زیر شبکه وجود داشته باشد، کارآیی کاهش می‌یابد. این وضعیت ازدحام نامیده می‌شود. به عبارت دیگر هر قسمتی از شبکه متناسب به نوع و شرایط سخت‌افزاری و نرم‌افزاری آن دارای ظرفیتی است. هرگاه بیش از آن ظرفیت به آن قسمت از شبکه بار تحمیل شود شبکه دچار ازدحام می‌شود. این ازدحام سرمنشا بسیاری از مسائل دیگری نظیر از بین رفتن بسته‌ها، کاهش کارآیی و ... است.

ازدحام به دلایل زیادی به وجود می‌آید. اگر ناگهان چند بسته از سه یا چهار خط ورودی بیایند و همگی بخواهند به یک خط خروجی بروند، یک صف تشکیل خواهد شد. به عبارت دیگر اگر پهنای باند ورودی و خروجی به یک مسیریاب به یک اندازه نباشد آن قسمتی که پهنای باند کم‌تری دارد به نقطه گلوگاه شبکه تبدیل می‌شود و اگر حافظه‌ی کافی برای ذخیره آن‌ها وجود نداشته باشد بسته‌ها از بین می‌روند.

مسیریاب‌های دارای پردازنده کند نیز می‌توانند عامل بوجود آورنده ازدحام باشند. اگر مسیریاب پردازش‌های مربوط به خود، نظیر صف‌بندی بافرها و بروزرسانی جداول مسیریابی را به کندی سرویس‌دهی نماید حتی اگر ظرفیت حمل خطوط نیز بیشتر از میزان فعلی باشد باعث ایجاد صف‌هایی می‌گردد. خطوطی ارتباطی که پهنای باند آن‌ها کم است نیز می‌توانند موجب ازدحام شوند.

از طرفی، در شبکه‌های بی‌سیم و به دلیل نوع رسانه به کار رفته، خطاهای انتقال بسیار بیشتر از شبکه‌های سیمی رخ می‌دهند که این نیز می‌تواند ناشی از افت یا نوسان کانال ارتباطی و یا مسائلی دیگر باشد. در هنگام طراحی TCP، رسانه‌های سیمی با نرخ خطای بیت 10^{-6} تا 10^{-8} را مد نظر قرار گرفته بودند. این در حالی است که نرخ خطای بیتی[2] (BER) در رسانه‌های بی‌سیم بسیار بالاتر است و معمولاً در حدود 10^{-3} و حتی گاهی اوقات 10^{-1} است [۲]. به دلیل همین نرخ خطای بالا در شبکه‌های بی‌سیم است که TCP در برخی از موارد به اشتباه دلیل خطا را ازدحام تشخیص داده و به دنبال آن پنجره فرستنده را کوچک‌تر می‌نماید و در نتیجه باعث کاهش توان عملیاتی می‌گردد.

۳- بررسی کارهای گذشته

به طور کلی روش‌هایی که برای بهبود کارایی TCP در مبحث اجتناب و کنترل ازدحام در شبکه‌های بی‌سیم مطرح شده‌اند در دو دسته قرار می‌گیرند. دسته اول روش‌هایی هستند که از مسیریاب‌ها یا گره‌های میانی کمک می‌گیرند و دسته دوم نیز روش‌هایی هستند که به گره‌های میانی متکی نیستند. در ادامه به معرفی چند روش از هر دسته پرداخته می‌شود.

۳-۱- روش‌هایی که از گره‌های میانی کمک می‌گیرند

بطور کلی این نوع روش‌ها سعی دارند تا با دادن بازخوردی از شبکه به TCP، بگونه‌ای مناسب این پروتکل را برای افزایش یا کاهش ارسال داده، راهنمایی نمایند. TCP بر مبنای هشدار ازدحام صریح یا ECN[3]، نیازمند اصلاح و تغییر بیت ECN در یک مسیریاب با ساز و کار مدیریت صف فعال[4] (AQM) است. هنگامی که میانگین طول صف یک مسیریاب از یک حد آستانه‌ای تجاوز نمود بیت آزمون ازدحام[5]

(CE) یک می‌شود تا نشان دهد و این که مسیریاب در یک حالت ازدحام فرو خواهد رفت. در هر حال اجرای TCP بر مبنای ECN بسیار پرهزینه بوده و سازوکار نظارت بر طول صف، خود دارای سربار بالایی است و همین امر موجب به وجود آمدن تاخیر در تنظیم کردن بیت ECN می‌شود.

یک روش مطرح در این زمینه پروتکل کنترل ازدحام صریح[۶] (XCP) [۳] است. روش XCP نیز مانند TCP یک روش کنترل ازدحام پنجره- مبنا است که تلاش می‌کند تا ترافیک شبکه را در بهترین حالت نگه دارد. در این روش که تعمیم یافته روش ECN است به جای استفاده از یک بیت که در ECN وجود ازدحام را نشان می‌دهد از سیگنال‌دهی ازدحام مداوم بهره برده می‌شود؛ بطوریکه شبکه بطور صریح به فرستنده وضعیت ازدحام شبکه و چگونگی برخورد با آن را اطلاع می‌دهد.

روش پیشنهاد شده دیگر پروتکل کنترل ازدحام متغیر- ساختار یا VCP[۷] [۴] نام دارد. در VCP هر مسیریاب یک عامل بار را محاسبه می‌کند و این عامل را برای دسته‌بندی سطح ازدحام به سه ناحیه بکار می‌گیرد. مسیریاب سطح ازدحام را در داخل دو بیت ECN رمزگذاری می‌کند. یکی از معایب این روش این است که در آن همگرایی به سمت عدالت در اختصاص منابع، بسیار آهسته‌تر از XCP صورت می‌گیرد.

در روش دیگری به نام TCP-Split یا TCP دوبخشی [۵]، نیاز است تا یک عامل در ایستگاه پایه بی‌سیم (BS) اضافه شود. مشکل این روش هم مقدار زیاد بافر مورد نیاز در BS است.

در کل پیاده‌سازی و اجرای این دسته از روش‌ها سخت، گران و در برخی از موارد غیرممکن است. بدین دلیل که باید هماهنگی وسیعی از لحاظ فیزیکی و منطقی، در بین هزاران هزار مسیریاب و گره موجود در شبکه اینترنت ایجاد شود که این امری تقریباً غیرممکن و بسیار پر هزینه است. از اینرو بسیاری از روش‌های این دسته در شبکه‌های خاص و با مقیاس محدود کاربرد دارند.

۳-۲- روش‌هایی که از گره‌های میانی کمک نمی‌گیرند

پروتکل کنترل ازدحام TCP بر پایه پنجره‌ی لغزان[۸] و برای اتصالات انتها به انتها طراحی شده است و از هیچ مسیریاب میانی کمک نمی‌گیرد. این پروتکل از چهار فاز: شروع آهسته، اجتناب از ازدحام، انتقال مجدد سریع و ترمیم سریع تشکیل یافته است.

۳-۲-۱- پروتکل TCP بر پایه‌ی پنجره لغزان و با استفاده از روش AIMD[۹]

بسیاری از روش‌هایی که به نوعی استاندارد شده‌اند و امروزه پرکاربرد هستند در این دسته قرار دارند. در این روش‌ها با رصد مداوم تغییرات پنجره نوع فازی که انتقال در آن قرار دارد مشخص گردیده و بسته به هر فاز تغییرات صورت می‌گیرد. این روش‌ها برپایه AIMD [۶] عمل می‌کنند. در ادامه چند روش مربوط به این دسته نوشته شده است.

در TCP-Tahoe [۷] با دریافت ACK مرتبط با بسته‌های فرستاده شده، اندازه پنجره به اندازه تعداد بسته‌های ارسال شده در مرحله قبلی، افزایش می‌یابد. به عبارت دیگر اندازه پنجره در فاز ابتدایی بصورت نمایی رشد می‌نماید. هرگاه که اندازه پنجره از حد آستانه شروع آهسته (ssthresh) تجاوز نمود پنجره وارد فاز اجتناب از ازدحام شده و اندازه آن بطور خطی افزایش می‌یابد. در این روش هنگامی که به کمک ACK تکراری[۱۰](dupack) سه‌گانه یا تایم‌اوت، ازدحام تشخیص داده شد، ssthresh دوباره تعیین شده و مقدارش برابر می‌شود با نصف پنجره ازدحام[۱۱] (cwnd)، و پنجره ازدحام نیز به ۱ تغییر می‌یابد.

در TCP-ReNo [۸] هرگاه فرستنده ACK تکراری سه‌گانه دریافت نمود، وارد فاز انتقال مجدد سریع می‌شود و مقادیر cwnd و ssthresh به $cwnd/2$ تغییر می‌یابند و سپس وارد فاز ترمیم سریع می‌شود تا وقتی که فرستنده یک ACK غیرتکراری دریافت کند. پس از دریافت چنین ACKای اندازه‌ی cwnd بطور خطی افزایش می‌یابد و هرگاه تایم‌اوت رخ دهد فرستنده وارد فاز شروع آهسته می‌شود. در روش TCP-NewReNo [۹] فاز ترمیم سریع تا موقعی که ACK تمام بسته‌های داخل پنجره جاری نرسیده باشد ادامه پیدا می‌کند که این باعث بهبود کارایی در مواقعی می‌شود که چندین بسته در داخل پنجره جاری گم شده باشد.

در روش‌های قبلی هر فرستنده TCP با دریافت ACK فقط از آخرین بخشی که با موفقیت و به صورت مرتب دریافت شده است اطلاع می‌یابد. بنابراین اگر در هر زمان رفت و برگشت[۱۲] (RTT) چندین خطا اتفاق بیفتد معمولاً سبب خطای تایم‌اوت در سمت فرستنده می‌شود. برای رفع این مشکل در روش TCP-SACK که در [۱۰] آمده است از گزینه‌ای به نام SACK[۱۳] در یکی از فیلدهای اختیاری در سرآیند TCP استفاده می‌شود.

تمام dupackها دارای گزینه SACK هستند. این گزینه لیستی از بلاک‌های داده هم‌جوار که توسط گیرنده دریافت شده‌اند را در بر می‌گیرد. در شرایطی که چندین بسته در یک پنجره مفقود شوند روش SACK بهتر از NewReNo عمل می‌کند. این روش می‌تواند تا چند بسته گم شده در طول یک RTT را انتقال مجدد دهد در حالی که NewReNo باید در انتظار ACK اولین بسته انتقال مجدد شده بماند تا از بسته‌های گم شده دیگر مطلع شود.

۳-۲-۲- پروتکل TCP برپایه برآورد پهنای باند در دسترس[۱۴] (ABE)

این دسته از روش‌ها که جدیدتر از روش‌های دسته قبل هستند سعی دارند تا برای حل مشکل ازدحام برآوردی پویا از برخی پارامترهای شبکه مانند پهنای باند در دسترس داشته باشند و متناسب با آن برآورد میزان داده ارسالی خود را به گونه‌ای تنظیم نمایند تا دچار مشکل

ازدحام نشده یا آن را در حد ممکن به تاخیر بیندازند. در ادامه برخی از روش‌های مطرحی که در این دسته قرار می‌گیرند توضیح داده می‌شود.

در روشی جدیدتر به نام TCP-Westwood [۱۱] الگوی جدیدی به نام AIADD[۱۵] ارائه شد [۱۱]. در این روش پهنای باند دردسترس شبکه بوسیله‌ی نرخ دریافت ACK در سمت فرستنده تخمین زده می‌شود. هنگامی که ازدحام رخ می‌دهد، cwnd و ssthresh با معیار پهنای‌باند × تاخیر تنظیم خواهند شد. در نتیجه اگر گم شدن بسته‌ای اتفاق بیفتد و مقدار cwnd کوچک‌تر از ssthresh باشد، مقدار cwnd تغییر نخواهد کرد. از طرفی، روش TCP-Westwood از تخمین غیر دقیق پهنای باند در دسترس رنج می‌برد. بدین منظور گریکو و ماسکولو[۱۶] در [۱۲]، روش جدیدی به نام TCP-Westwood+ را ارائه نمودند تا تخمین پهنای باند در دسترس را بطور دقیق‌تری انجام دهد و در نتیجه توان عملیاتی افزایش پیدا کند.

روش دیگری که در این دسته قرار می‌گیرد TCP Jersey است [۱۳] که بر پایه‌ی TCP-Westwood عمل می‌کند [۱۳]. این روش در هنگام اجرا هم از برآورد پهنای باند در دسترس کمک می‌گیرد و هم از هشدار ازدحامی که به کمک اصلاح بیت ECN در یک ساز و کار مدیریت صف فعال (AQM) از مسیریاب دریافت می‌کند.

روش دیگری که در [۱۴] ارائه شده است TCP-VeNo نام دارد. این روش با تخمین میزان بسته‌های موجود در صف ایجاد شده در بخش گلوگاه شبکه (N)، در هنگام گم‌شدن بسته و برای تصمیم‌گیری در باره میزان کاهش دادن اندازه پنجره از مقدار N استفاده می‌کند.

TCP-Vegas [۱۵] روش دیگری است که در آن رخ دادن یا ندادن ازدحام را با اختلاف بین نرخ مورد انتظار ($cwnd/RTT_{\min}$) و نرخ واقعی ($cwnd/RTT$) تعیین می‌کند. این روش بسیار متفاوت‌تر از روش‌های AIMD و AIADD است به طوری که دو آستانه را به عنوان حد پایین و حد بالا به نام‌های α و β در نظر می‌گیرد. هنگامی که اختلاف بین نرخ‌ها کمتر از حد پایین باشد (یعنی توان واقعی به نزدیکی توان عملیاتی مورد انتظار رسیده است پس شبکه زیاد شلوغ نیست)، cwnd به طور خطی افزایش می‌یابد. و به همین ترتیب هنگامی که تفاوت بیشتر از حد بالایی باشد (یعنی بسته‌های زیادی داخل شبکه است) cwnd بطور خطی کاهش خواهد یافت. هنگامی که تفاوت در محدوده‌ای بین دو آستانه باشد cwnd تغییر نخواهد یافت. در نتیجه TCP-Vegas می‌تواند به یک پنجره پایا برسد اما این روش در بین تمام ارتباطات مورد استفاده قرار گرفته در کنترل TCP-Vegas زیاد عادلانه رفتار نمی‌کند و همین طور هنگامی که کنترل پنجره خطی- مبنا TCP-Vegas و کنترل پنجره نمایی- مبنا AIMD/AIADD در یک زمان مورد استفاده قرار می‌گیرد این روش زیاد دوستانه[۱۷] عمل نمی‌کند، و به سهم عادلانه‌ای از پهنای باند نمی‌رسد.

و سرانجام LIAD[۱۸] [۱۶] روشی جدید است که از الگوی افزایش لگاریتمی و کاهش انطباقی استفاده می‌نماید. در این روش که تماماً در سمت فرستنده پیاده‌سازی می‌گردد، ابتدا پنجره ازدحام بهینه بعنوان مرزی تعیین می‌شود که اندازه پنجره همواره باید کوچک‌تر از آن باشد تا شبکه رو به ازدحام نرود. سپس آستانه شروع آهسته ایده‌آل نیز برای پیش‌بینی زمان مناسب تغییر فاز از شروع آهسته به اجتناب از ازدحام محاسبه می‌شود. پس از بدست آوردن آستانه شروع آهسته ایده‌آل پهنای باند انطباقی شبکه محاسبه شده و سپس به کمک آن و زمان رفت و برگشت کمینه ($RTT_{\min}$) آستانه شروع آهسته واقعی شبکه بدست می‌آید. هم چنین در این روش اگر فرستنده ۳ dupack دریافت کند، اندازه پنجره خود را به اندازه بهینه که از رابطه (۱) بدست می‌آید تغییر می‌دهد.

$$cwnd_{opt}=cwnd\times\left(\frac{RTT_{\min}}{RTT}\right) \qquad (۱)$$

ایده اصلی این روش آن است که اندازه پنجره همواره از زمان RTT که نشان‌دهنده میزان شلوغی شبکه است تاثیر می‌پذیرد. اگر ازدحام در شبکه بیشتر شود RTT افزایش می‌یابد و متناسب با آن اندازه پنجره بهینه کاهش می‌یابد؛ حال اگر در شبکه بسته‌ای به علت خطا در ارتباطات بی‌سیم گم شود مقدار RTT به طور چشم‌گیری تفاوت نخواهد کرد و در نتیجه اندازه پنجره نیز تفاوت زیادی نخواهد کرد.

۳-۲-۳- انواع دیگر TCP

روش‌های دیگر کنترل ازدحام که بهبود یافته روش استاندارد TCP هستند شامل TCP پرسرعت (HSTCP) [۱۷]، TCP مقیاس‌پذیر[۱۹] (STCP) [۱۸]، TCP سریع (FTCP) [۱۹]، کنترل ازدحام با افزایش دودویی (BIC) [۲۰] و (CUBIC) [۲۱]، و افزایش لگاریتمی برپایه روش TCP-Westwood (LogWestwood+) [۲۲] هستند. روش‌های BIC، CUBIC و LogWestwood+ از یک تابع لگاریتمی برای افزایش اندازه پنجره TCP در فاز اجتناب از ازدحام و در شبکه‌های پرسرعت با مسافت طولانی بهره می‌برند. در روشی دیگر به نام TCP ادغامی (CTCP) [۲۳] از هر دو سازوکار مبتنی بر ازدحام و مبتنی بر تاخیر بهره می‌برد. هم چنین در روشی که در [۲۴] ارائه شده است و به اختصار TCP-AP نامیده می‌شود از RTT به عنوان پارامتری که نشان‌دهنده وضعیت ترافیکی شبکه است در مکانیسم بکار رفته برای کنترل ازدحام استفاده می‌شود. این روش به نوعی در لایه TCP عمل نموده و میزان رقابت بر سر مسیر موجود را اندازه‌گیری می‌نماید.

هم چنین در دسته‌ای دیگر از روش‌ها گیرنده با هدف کاهش تعداد ACK ها به ازای چند بسته دریافتی (که پنجره تاخیر نامیده می‌شود)، یک ACK تجمعی به گیرنده ارسال می‌نماید. برای نمونه راهکار ارائه شده در [۲۵] پنجره تاخیر بهینه‌ای به اندازه w تعیین

می‌کند و گیرنده به ازای دریافت این تعداد بسته به طور مرتب، یک ACK تجمعی با تاخیر به فرستنده ارسال می‌نماید و اگر بسته‌ای مرتب نرسد گیرنده با یک ACK مناسب فرستنده را سریع‌تر مطلع می‌سازد. لازم به ذکر است که اندازه w به کمک زمان‌سنج و با توجه به وضعیت شبکه به طور پویا تعیین می‌شود.

در روش دیگری که در [۲۶] ارائه شده است اندازه پنجره تاخیر به صورت انطباقی و با توجه به شرایط شبکه تعیین می‌شود. به عبارت دیگر گیرنده و فرستنده اندازه پنجره‌های تاخیر و ازدحام را با یکدیگر هماهنگ می‌کنند و برای آنها حد بیشینه‌ای در نظر گرفته شده است. در این روش نیز زمان‌سنجی کمک می‌کند تا پنجره‌های منطبق با وضعیت ترافیکی شبکه با نرخ‌های متفاوت و تا حد بیشینه رشد نمایند. اگر بسته‌ای نیز گم شود آخرین ACK تجمعی فرستاده شده دوباره به فرستنده ارسال می‌گردد.

۴- روش پیشنهادی

همان طور که می‌دانیم در TCP آستانه شروع آهسته (ssthresh) پارامتری است که تعیین می‌کند که فرستنده چه زمانی وارد فاز اجتناب از ازدحام شود. اگر مقدار این آستانه بسیار کوچک در نظر گرفته شود فرستنده زود وارد فاز اجتناب از ازدحام شده و این کار ممکن مسبب ایجاد دو مشکل شود. مشکل اول این است که نرخ افزایش cwnd بسیار زودتر از زمان لازم کاهش می‌یابد که این باعث می‌شود توان عملیاتی کاهش یافته و از ظرفیت شبکه به طور متناسب استفاده نشود. دوم این که، نرخ تغییرات cwnd بسیار آهسته شود که باعث بوجود آمدن ناعدالتی در استفاده از پهنای باند شبکه[۲۰] می‌گردد. از سوی دیگر، اگر مقدار ssthresh خیلی بزرگ در نظر گرفته شود، فرستنده بسته‌های زیادی را به یک شبکه دچار ازدحام ارسال می‌کند و این باعث می‌شود ازدحام در آن شبکه تشدید گردد. چرا که مسیریاب‌های میانی که در آن‌ها ازدحام بوجود آمده است ممکن است بسته‌های بیش‌تر از ظرفیت بافر خود را دور انداخته و این باعث بروز تعداد زیادی خطای تایم‌اوت و ACK تکراری سه‌گانه شود. در بدترین حالت، فرستنده دوباره وارد فاز شروع آهسته می‌شود و اندازه cwnd به ۱ تغییر می‌یابد و در نتیجه توان‌عملیاتی کاهش چشمگیری پیدا می‌کند. بنابر این، داشتن یک آستانه شروع آهسته ایده‌آل و انطباقی می‌تواند به افزایش توان‌عملیاتی و عدالت در شبکه کمک کند.

۴-۱- محاسبه آستانه شروع آهسته انطباقی

بر اساس توضیحات فوق، ترافیک شبکه تاثیر زیادی در تعیین وضعیت شبکه و فاز انتقال دارد. اگر خود فرستنده بتواند میزان ترافیک را پیش‌بینی کند قادر خواهد بود تا آستانه شروع آهسته ایده‌آل ($ssthresh_{ideal}$) را نیز بر اساس شرایط ترافیکی شبکه و برای تعیین زمان درست انتقال از حالت شروع آهسته به اجتناب از ازدحام، محاسبه نماید. از آن جایی که RTT تا حدی در TCP نمایانگر بار ترافیک شبکه است می‌توان در روشی RTT- مبنا و به طور انطباقی $ssthresh_{ideal}$ را تعیین کرد. هم چنین عامل دیگری که می‌تواند نشان‌دهنده بار ترافیک جاری شبکه باشد interAck است. فاصله زمانی میان دو ACK رسیده پشت سرهم را interAck می‌نامیم که افزایش آن بیانگر روند افزایش ترافیک شبکه است [۱۷].

با دریافت هر بسته ACK جدید، فرستنده می‌تواند interACK را تعیین کند. برای محاسبه interACK و نرمال‌سازی مقدار آن در رابطه (۲)، از روش میانگین تفاوت وزن نمایی[۲۱] که در TCP استاندارد [۱] نیز برای محاسبه پویای مهلت انتقال مجدد[۲۲] از آن استفاده شده است، بهره می‌بریم.

$$\begin{aligned} interACK &= \alpha.interACK_{current} \\ &+(1-\alpha).(interACK\_DIFF) \end{aligned} \quad (۲)$$

به طوری که داریم:

$$interACK\_DIFF = interACK_{current} - interACK_{old} \quad (۳)$$

در رابطه فوق α مقدار ثابتی بین ۰ تا ۱ است، که در پیاده‌سازی روش پیشنهادی در این مقاله مشابه با روش استاندارد محاسبه‌ی RTT، مقدار آن ۰/۷۸ در نظر گرفته شده است.

$interACK_{current}$ فاصله زمانی بین آخرین ACKهای پی‌درپی است و interACK_Diff اختلاف زمانی بین دو interACK فعلی و قبلی است.

از سوی دیگر RTT_{min} متغیری است که وضعیت شبکه را در شرایطی مناسب و عاری از هرگونه ازدحام به فرستنده اطلاع می‌دهد این متغیر کوچک‌ترین RTT مشاهده شده در مبدا در طول زمان شبیه‌سازی می‌باشد. بدین صورت که هر زمان شبکه خلوت باشد مدت زمان ما بین ارسال یک بسته و دریافت ACK آن مقدار کمینه‌ای خواهد داشت. به عبارت دیگر بسته ارسال شده توسط فرستنده در طول مسیر در صفی معطل نمی‌ماند و RTT آن نمایانگر زمانی است که بسته بدون معطل شدن از فرستنده تا گیرنده حرکت و سپس اعلام وصول آن توسط فرستنده دریافت می‌گردد. فرستنده با داشتن RTT_{min}، می‌تواند با محاسبه RTT هر بسته به مقایسه آن با RTT_{min} بپردازد. اختلاف بین این دو مقدار، زمان تاخیری است که با توجه به میزان آن می‌توان میزان وجود ترافیک را تخمین زد. آستانه شروع آهسته ایده‌آل بر اساس رابطه زیر بیان می‌شود:

$$ssthresh_{ideal} = \frac{RTT_{min}}{interACK} + 1 \quad (۴)$$

همان طور که می‌توان دید هر قدر که interAck افزایش یابد، نمایانگر این است که شبکه زیاد خلوت نیست و به این دلیل $ssthresh_{ideal}$ نیز کاهش می‌یابد و شبکه زودتر وارد فاز اجتناب از ازدحام می‌شود. از

سوی دیگر زیاد بودن RTT_{min} به این معنی است که شبکه ظرفیت پذیرش تعداد زیادی از بسته‌ها را دارد؛ بسته‌هایی که بر روی لینک‌های مختلف شبکه در حال حرکت می‌باشند.

به عبارت دیگر آستانه ایده‌آل، حد بالای اندازه پنجره ازدحام را نشان می‌دهد و این یعنی پنجره ازدحام در بهترین شرایط می‌تواند آن مقدار را داشته باشد و به نوعی با بدست آوردن این مقدار از ظرفیت شبکه می‌توان مطلع شد. به عنوان مثال، هنگامی که TCP بداند که فاصله زمانی فرستنده تا گیرنده مثلاً ۱۲ میلی‌ثانیه است و هر ۲ میلی‌ثانیه یکبار یک ACK جدید می‌رسد پس می‌تواند محاسبه کند که در شبکه‌ای با چنین وضعیتی می‌توان به طور هم‌زمان حداکثر ۷ بسته را در شبکه انتقال داد تا صفی برای پردازش بسته‌ها تشکیل نشود و ازدحام بوجود نیاید. از این رو، هنگامی که شبکه دارای کم‌ترین میزان ترافیک بسته‌ای و جریانی باشد مثلاً در ابتدای شروع بکار ارسال توسط فرستنده می‌توان برای جلوگیری از ازدحام این مقدار را به عنوان حد آستانه در نظر گرفت.

با بدست آمدن $ssthresh_{ideal}$ که به نوعی ظرفیت شبکه را بر اساس تعداد بسته‌ها نشان می‌دهد، پهنای باند انطباقی شبکه نیز از رابطه (۵) بدست می‌آید.

$$Adaptive\_Bandwidth\_Network=\frac{ssthresh_{indeal}}{RTT} \qquad (۵)$$

با بدست آوردن پهنای باند انطباقی شبکه، می‌توان آستانه شروع آهسته شبکه را نیز از رابطه زیر بدست آورد:

$$ssthresh_{net}=\left(Adaptive\_Bandwidth\_Newtork\right).\left(RTT_{min}\right) \qquad (۶)$$

در روش LIAD و با استفاده از روابط بالا آستانه شروع آهسته شبکه از روابط (۷) و (۸) محاسبه می‌شود:

$$ssthresh_{net}=\left(\frac{ssthresh_{ideal}}{RTT}\right).\left(RTT_{min}\right) \qquad (۷)$$

$$ssthresh_{net}=\left(\frac{RTT_{min}}{interACK}+1\right).\left(\frac{RTT_{min}}{RTT}\right) \qquad (۸)$$

در رابطه فوق RTT_{min}/RTT عامل کاهنده آستانه شروع آهسته ایده‌آل نامیده می‌شود. از آن جایی که با گذشت زمان و افزایش تعداد جریان‌ها در شبکه ممکن است، صف‌هایی در مسیر ایجاد و منجر به افزایش زمان RTT شود، مقدار عامل کاهنده همواره کوچک‌تر از ۱ خواهد بود. البته در بهترین شرایط که $RTT=RTT_{min}$ است این مقدار ۱ خواهد شد.

محاسبه آستانه شروع آهسته شبکه از این طریق و بر مبنای RTT ممکن است از مسائل زیر رنج ببرد:

۱. از آن جایی که RTT مجموع زمان رفت یک بسته تا گیرنده و برگشت ACK آن به فرستنده است نمی‌تواند تحلیل درستی از ترافیک مسیر رفت را به فرستنده بدهد [۲۷]. برای مثال از بین دو بسته ممکن است RTT بسته اول از بسته دوم بیشتر باشد ولی مسیر رفت به سمت گیرنده برای بسته اول دارای ترافیک کم‌تری باشد و این به دلیل این است که ACK بسته اول به علت شلوغی مسیر برگشت دیرتر به فرستنده رسیده است.

۲. در کل رسانه‌های بی‌سیم در مقایسه با رسانه‌های سیمی از تاخیر و زمان انتقال طولانی‌تری برخوردارند [۲]. این تاخیر باعث می‌شود که RTT بسیار افزایش یابد و به دنبال آن در عملکرد پروتکل‌ها در شبکه‌های ناهمگن اختلال ایجاد شود. علت این اختلال تفاوت زیاد مقدار RTT در بخش‌های مختلف شبکه است که باعث می‌شود پروتکل نتواند بر مبنای RTT تحلیل درستی از شرایط ترافیکی شبکه داشته باشد.

برای تشریح حالت بعدی دو وضعیت را در نظر می‌گیریم. در وضعیت اول ترافیک شبکه به طور تقریباً مساوی بر روی مسیریاب‌های مختلف توزیع شده است و هر مسیریاب بخش کوچکی از بافر خود را اشغال شده می‌بیند و ازدحام خاصی در هیچ مسیریابی مشاهده نمی‌شود. در وضعیت دوم، ترافیکی حتی کمتر از ترافیک قبلی را در نظر بگیرید که بطور نامساوی بین مسیریاب‌ها پخش شده است، به گونه‌ای که یک مسیریاب به علت ضعیف بودن پردازنده‌اش و یا تحمیل بار اضافی از سوی جریانی متقاطع، مسیر را دچار اختلال نموده و بافر آن در آستانه پر شدن باشد. RTT محاسبه شده در وضعیت اول بیشتر از RTT وضعیت دوم می‌باشد، ترافیک بیشتری در شبکه وجود دارد، درحالی که شبکه وضعیت عادی خود را دارد. اما در وضعیت دوم، اصولاً RTT کم‌تر، باید خبر از بهتر بودن شرایط شبکه را نسبت به وضعیت اول بدهد ولی ما می‌دانیم که اینگونه نیست و وضعیت دوم در آستانه گم‌شدن بسته‌ها به علت پر شدن بافر یک مسیریاب قرار دارد.

بر اساس سناریوی فوق می‌توان دید، که RTT بعضاً نشانه‌ی دقیقی برای اندازه‌گیری شدت ازدحام شبکه نیست. در چنین مواقعی می‌توان به کمک عوامل دیگری نظیر interACK، حالت‌های خاصی را که ممکن است، از RTT برداشت درستی نشود، تصحیح نمود.

برای این منظور پیشنهاد می‌کنیم از نرخ تغییرات interACK در محاسبه انطباقی آستانه شروع آهسته شبکه استفاده شود. براساس رابطه (۹) این عامل را که $interACK_{changerate}$ می‌نامیم، میزان تغییرات interACK را نسبت به مقدار کمینه ($interACK_{min}$) آن که اغلب در آغاز کار فرستنده بدست می‌آید بیان می‌دارد.

$$interACK_{changerate}=\frac{interACK_{min}}{interACK} \qquad (۹)$$

برای به دست آوردن آستانه شروع آهسته انطباقی شبکه، رابطه مربوط به عامل کاهنده را در رابطه (۸) و با کمک گرفتن از $\text{interACK}_{\text{changerate}}$، بازنویسی می‌کنیم. به گونه‌ای که داریم:

$$\text{ssthresh}_{\text{net}}=\left(\frac{\text{RTT}_{\min}}{\text{interACK}}+1\right).\left(0.5\times\frac{\text{RTT}_{\min}}{\text{RTT}}+0.5\times\frac{\text{interACK}_{\min}}{\text{interACK}}\right) \quad (۱۰)$$

در رابطه (۱۰) مشاهده می‌شود که با کمک معیار دیگری نظیر interACK می‌توان در کنار استفاده از RTT به درک درست‌تری از شرایط ترافیک شبکه رسیده و متناسب با آن آستانه شروع آهسته با دقت بیشتری نسبت به وضعیت شبکه بدست آمده است. اگر شرایط شبکه خلوت باشد که مقدار عامل کاهنده برابر با یک می‌شود و به عبارت دیگر آستانه شروع آهسته انطباقی شبکه برابر است، با آستانه شروع آهسته ایده‌آل. با استفاده از $\text{ssthresh}_{\text{net}}$ می‌توان زمان مناسب برای تغییر فاز انتقال از شروع آهسته به اجتناب از ازدحام را بدست آورد. می‌توان با استفاده از چنین ساز و کاری و با تقریب خوبی منشا گم‌شدن بسته‌ها در طول فاز شروع آهسته را رخ‌دادن خطای رسانه انتقالی دانست و برخورد مناسب با آن را انجام داد. شیوه دقیق این کار در قسمت بعدی به تفصیل شرح داده شده است.

۴-۲- مدیریت تغییرات پنجره

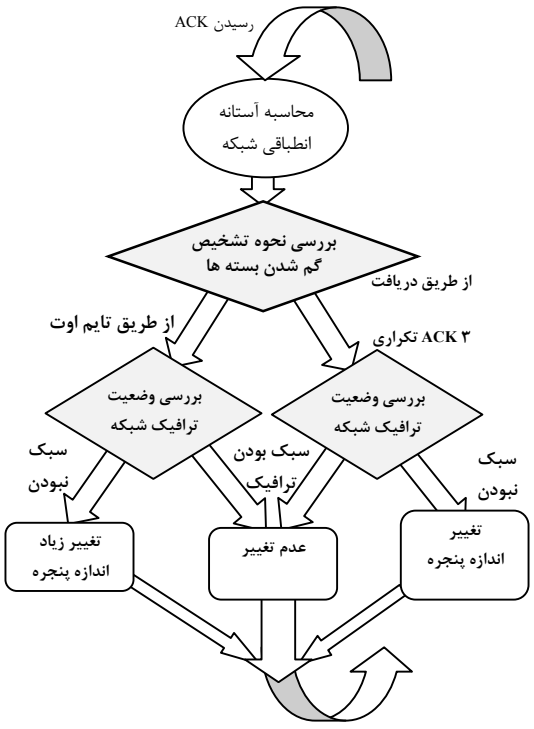
همان طور که عنوان شد مساله‌ای که در هنگام بکارگیری بسیاری از روش‌های کنترل ازدحام در شبکه‌های بی‌سیم مساله‌ساز می‌شود، این است که این روش‌ها در صورت اطلاع از گم‌شدن بسته‌ای چه از طریق دریافت سه dupack و چه از طریق خطای تایم‌اوت، اندازه پنجره ازدحام را بسته به نوع روش به اندازه‌های مختلف و تا یک کاهش می‌دهند. در صورتی‌که علت گم‌شدن، خطاهای شبکه بی‌سیم باشد نباید این کاهش انجام پذیرد.

خطاهای بی‌سیم می‌تواند در هر فاز انتقال رخ دهد، از طرفی ما می‌دانیم که به احتمال بسیار زیاد در فاز شروع آهسته بسته‌ای به علت ازدحام گم نخواهد شد پس می‌توان در هنگام تشخیص گم‌شدن بسته با هر یک از دو روش، ابتدا بررسی کرد که اگر انتقال در فاز شروع آهسته باشد پس اندازه پنجره بدون تغییر باقی بماند. در این تحقیق به منظور طراحی الگوریتم پیشنهادی، از TCP-Vegas به عنوان الگوریتم پایه بهره می‌بریم. بر این اساس روش TCP-Vegas را به گونه‌ای تغییر می‌دهیم که اگر گم‌شدن بسته‌ای تشخیص داده شد به شرطی پنجره ازدحام کاهش می‌یابد که ازدحام در فاز شروع آهسته نباشد و $\Delta>\beta$ باشد. به عبارتی ما منطقه‌ای که Δ مقداری کمتر از β دارد را منطقه امن به‌حساب می‌آوریم، به طوری که در آن منطقه ازدحام به حدی نیست که بسته‌ها به دلیل آن از بین روند.

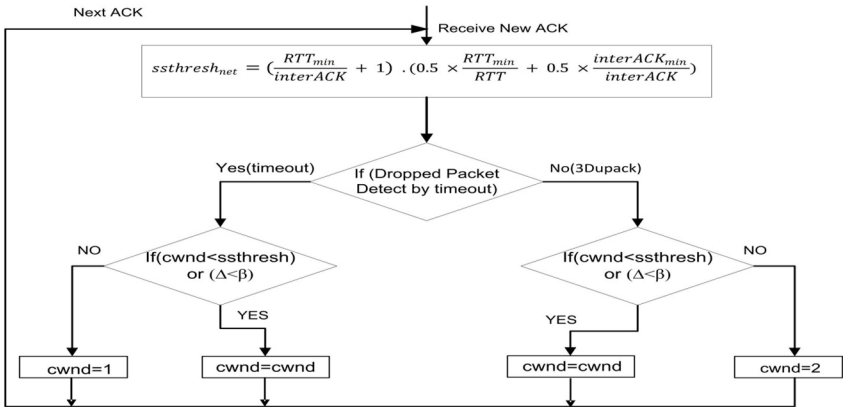
در مرحله پایانی و برای انجام واکنش مناسب به بسته گم‌شده، از آستانه شروع آهسته انطباقی ($\text{ssthresh}_{\text{net}}$) که در قسمت قبل محاسبه نمودیم استفاده می‌کنیم. همان طور که عنوان شد $\text{ssthresh}_{\text{net}}$ نشان‌دهنده آستانه انطباقی شبکه و مرز بین فازهای شروع آهسته و اجتناب از ازدحام است که ظرفیت شبکه را با توجه به ترافیکی که در آن وجود دارد به فرستنده اعلام می‌دارد. هرگاه فرستنده گم‌شدن بسته‌ای را تشخیص داد ظرفیت فعلی شبکه را مورد بررسی قرار می‌دهد. اگر شبکه شلوغ نبود و ظرفیت تحمل داده‌های ورودی بیشتری را داشت، پس به احتمال زیاد بسته به علت خطای انتقال گم شده است. در چنین شرایطی دیگر نیاز به کاهش اندازه پنجره نیست و بهتر است اندازه پنجره ثابت نگاه داشته شود تا از کاهش توان عملیاتی شبکه جلوگیری شود.

همان طور که در شکل (۱) می‌توان مشاهده نمود طرز کار کلی و مفهومی روش پیشنهادی از قرار زیر است. پروتکل کنترل انتقال با رسیدن هر ACK جدید به محاسبه آستانه انطباقی شبکه می‌پردازد. هرگاه که خطای dupack سه‌گانه رخ داد وضعیت ترافیکی شبکه مورد بررسی قرار می‌گیرد. اگر اندازه cwnd کمتر از $\text{ssthresh}_{\text{net}}$ بود و یا نرخ اختلاف توان‌عملیاتی مورد انتظار و واقعی کوچکتر از متغیر آستانه‌ای با مقدار ۳ بود، پس ترافیک سبک است و نیازی به کاهش اندازه پنجره نیست. ولی اگر این شرایط برقرار نبودند طبق عملکرد TCP-Vegas اندازه پنجره به عدد ۲ کاهش می‌یابد.

حال اگر خطای تایم‌اوت رخ داده باشد وضعیت ترافیکی شبکه مورد بررسی قرار می‌گیرد. اگر شرایط بالا برقرار بود مشابه برخورد با



شکل (۱): نمودار مفهومی طرز کار روش پیشنهادی



شکل (۲): نمودار طرز کار روش پیشنهادی‌با جزئیات

dupack سه‌گانه عمل می‌شود. ولی اگر این شرایط برقرار نبودند پس به احتمال زیاد در شبکه ازدحام رخ‌داده است و از اینرو اندازه پنجره به عدد ۱ کاهش می‌یابد (شکل (۲)). لازم به ذکر است که ما از این به بعد روش پیشنهادی خود را در این متن TCP-WANET می‌نامیم.

۵- ارزیابی روش پیشنهادی

در اینجا روش پیشنهادی با نسخه‌های پرکاربرد TCP مانند TCP-Tahoe، TCP-Vegas، TCP-New ReNo و TCP-SACK مورد مقایسه قرار می‌گیرد.

۵-۱- معیارهای ارزیابی

یکی از اصلی‌ترین معیارها در مقایسه‌ی روش‌های مختلف کنترل ازدحام، توان عملیاتی یا توان گذردهی نام دارد. منظور از توان عملیاتی در این بخش میزان ترافیک دریافتی توسط یک گره در مدت زمان انجام شبیه‌سازی است. الگوریتم‌های اجتناب از ازدحام به دلیل سیاست محتاطانه‌ای که دارند باعث کوچک نگاه‌داشتن اندازه پنجره ازدحام و در پی آن کاهش نرخ انتقال می‌شوند. معیارهای ارزیابی دیگری نیز در این نوع شبکه‌ها وجود دارند که برخی از مهمترین آن‌ها عبارتند از *نسبت بسته‌های از بین رفته*[۲۳] که عبارت است از تعداد بسته‌های از بین رفته بر تعداد بسته‌های ارسالی، و نسبت سربار انتقال که عبارت است از تعداد بسته‌های مسیریابی به بسته‌های TCP و میانگین زمان تاخیر انتها به انتها تمام بسته‌های TCP، و میزان انرژی مصرفی توسط گره‌های بی‌سیم.

۵-۲- محیط شبیه‌سازی و طراحی سناریو

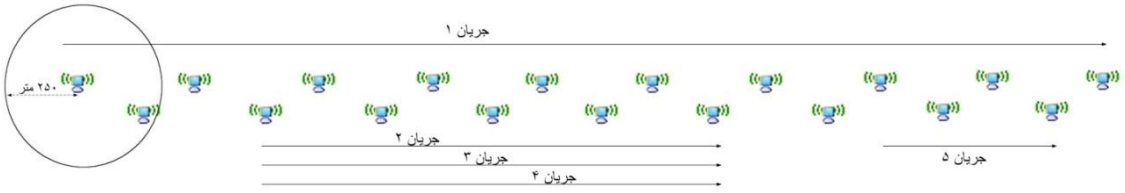
برای ارزیابی و تحلیل کارآیی روش‌های مختلف کنترل ازدحام در محیط شبکه‌های سیار و مقایسه عملکرد روش پیشنهادی از نسخه شماره ۲ نرم‌افزار شبیه‌ساز NS [28][۲۸] استفاده کرده‌ایم. در این پژوهش الگوریتم پیشنهادی با تغییرات اعمال شده در کد روش TCP-Vegas، موجود در شبیه‌ساز شبکه ns-2، پیاده‌سازی شده است. جهت ارزیابی این الگوریتم سناریویی در قالب شبکه‌های موردی بی‌سیم طراحی و با زبان TCL پیاده‌سازی شده است.

۵-۲-۱- سناریو شبیه‌سازی

برای شبیه‌سازی نحوه کار پروتکل‌های مختلف کنترل انتقال، ما از دو سناریوی پرکاربرد یعنی زنجیره‌ای و ماتریسی در شبکه‌های بی‌سیم استفاده می‌کنیم.

سناریو اول، شبکه زنجیره‌ای: در این سناریو گره‌های سیار بطور زنجیره‌ای و توسط اتصالات بی‌سیم به هم متصل شده‌اند و به نوعی ارتباطات در این شبکه چندگامی یا چند مرحله‌ای است. بدین گونه که همواره گره‌ها حداکثر با دو گره مجاور خود در ارتباط هستند و ارتباط با گره‌های دیگر به کمک گره‌های میانی انجام می‌گیرد (شکل (۳)). در این سناریو ما کنترل بیشتری روی جریان‌های مختلف داشته و می‌توانیم چندین جریان را بر روی یک لینک خاص متمرکز کنیم تا باعث بوجود آمدن ازدحام در شبکه شویم.

در این نوع سناریو و برخلاف سایر سناریوهای رایج شبیه‌سازی، تمام



شکل (۳): وضعیت گره‌ها و جریان‌ها در سناریوی اول

لینک‌های ارتباطی از پهنای باند یکسانی بهره می‌برند و لینک

گلوگاهی با پهنای باند کم وجود ندارد. از سوی دیگر در این سناریو با ایجاد چندین جریان بر روی یک لینک، ترافیک بسته‌ها در آن ناحیه افزایش یافته و به دنبال آن ازدحام ایجاد می‌شود، سپس عملکرد روش‌های مختلف کنترل ازدحام بررسی و با یکدیگر مقایسه می‌گردند.

جریان ترافیکی اصلی، یک جریان FTP از گره اول به گره آخر است، که از روش ارائه شده ما به عنوان پروتکل انتقال استفاده می‌کند. سه ترافیک FTP دیگر نیز به عنوان ترافیک پس‌زمینه مورد استفاده قرار می‌گیرد که از گره ۴ به گره ۱۲ ارسال می‌گردد. یک جریان FTP نیز از گره ۱۶ به گره ۱۹ وجود دارد. اندازه بسته‌های TCP برابر با ۱۴۶۰ بایت در نظر گرفته شده و پروتکل مسیریابی مورد استفاده AODV می‌باشد.

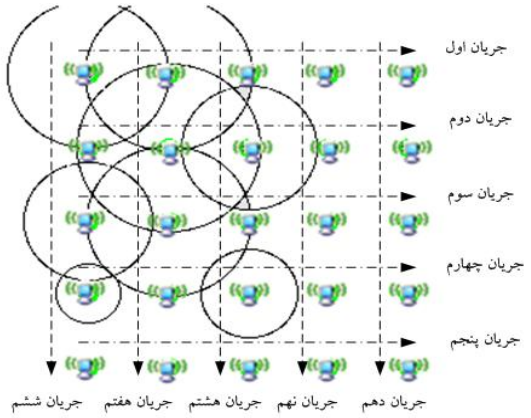
بسته‌های ارسالی روی پیوند بی‌سیم در معرض خطاهای انتقال بی‌سیم قرار دارند. مقدار نرخ خطای بسته بین ۰/۰۰۱ تا ۰/۰۷ متغیر است. انرژی اولیه گره‌ها ۸۰ ژول در نظر گرفته شده است، بطوریکه تمامی گره‌های موجود در شبکه و تحت هر کدام از پروتکل‌های موجود در شبیه‌سازی، تا پایان زمان شبیه‌سازی انرژی داشته و به عبارتی زنده می‌مانند. جدول (۱) نشان دهنده پارامترهای مربوط به سناریو اول

جدول (۱): پارامترهای مربوط به سناریوی شبیه سازی

پارامتر	مقدار
نوع گره‌ها (Nodes Type)	بی‌سیم
تعداد گره‌ها (Nodes Numbers)	۲۰
انرژی اولیه گره‌ها (Init Energy)	۸۰ ژول
سرعت حرکت گره‌ها (Nodes Speed)	۱۵ متر بر ثانیه
میزان فضای حرکتی	۱۲۰۰ × ۴۰۰۰
نرخ خطا (Error Rate)	۰/۰۰۱ و ۰/۰۱ و ۰/۰۵ و ۰/۰۷
الگوریتم مسیریابی (Routing Algorithm)	AODV
تعداد جریان (Flow Number)	۵
پهنای باند (Link Bandwidth)	۱۰ مگابیت‌برثانیه
محدوده انتقال (Transmission Range)	۲۵۰ متر
الگوریتم مدیریت صف (Queue Management)	Drop Tail
بیشینه اندازه صف (Queue Max Size)	۵۰
اندازه بسته (Packet Size)	۱۴۶۰ بایت

شبیه‌سازی مورد استفاده است.

سناریو دوم، شبکه ماتریسی: سناریوی دوم بکار گرفته شده نیز یک شبکه بی‌سیم موردی است که در آن گره‌های سیار بطور ماتریسی و توسط اتصالات بی‌سیم به هم متصل شده‌اند و به عبارت دیگر ارتباطات در این شبکه نیز چندگامی یا چند مرحله‌ای است. بدین گونه که همواره گره‌ها حداکثر با چهار گره مجاور خود در طرفین و بالا و پایین مرتبط هستند و ارتباط با گره‌های غیرمجاور با کمک گره‌های میانی انجام می‌گیرد (شکل ۴). در این سناریو پنج جریان افقی از گره اول هر سطر ماتریس به گره آخر همان سطر ماتریس و پنج جریان عمودی از گره اول هر ستون ماتریس به گره آخر همان ستون ماتریس



شکل (۴) وضعیت گره‌ها و جریان‌ها در سناریوی دوم

جدول (۲): پارامترهای مربوط به سناریوی شبیه سازی دوم

پارامتر	مقدار
نوع گره‌ها (Nodes Type)	بی‌سیم
تعداد گره‌ها (Nodes Numbers)	۲۵
انرژی اولیه گره‌ها (Init Energy)	۸۰ ژول
میزان فضای حرکتی	۱۰۰۰ × ۱۰۰۰
نرخ خطا (Error Rate)	۰/۰۰۱ و ۰/۰۱ و ۰/۰۵ و ۰/۰۷
الگوریتم مسیریابی (Routing Algorithm)	AODV
تعداد جریان عمودی (Vertical Flow Number)	۵
تعداد جریان افقی (Horizontal Flow Number)	۵
پهنای باند (Link Bandwidth)	۱۰ مگابیت برثانیه
محدوده انتقال (Transmission Range)	۲۵۰ متر
الگوریتم مدیریت صف (Queue Management)	Drop Tail
بیشینه اندازه صف (Queue Max Size)	۵۰
اندازه بسته (Packet Size)	۱۴۶۰ بایت

برقرار است. به عبارت دیگر از تمامی گره‌ها دو جریان که یکی عمودی و دیگری افقی است عبور می‌کند. جدول (۲) نیز نشان‌دهنده پارامترهای مربوط به سناریو دوم شبیه‌سازی مورد استفاده است.

۵-۳- نتایج حاصل از شبیه‌سازی

در این قسمت نتایج حاصل از شبیه‌سازی بر اساس دو سناریوی مذکور ارائه شده و سپس آن نتایج مورد تحلیل و بررسی قرار گرفته است.

۵-۳-۱- نتایج سناریوی اول

در شبکه منطبق با سناریوی اول، همواره گره‌ها بطور زنجیره‌ای با یکدیگر در ارتباط هستند. هدف از این امر بوجود آوردن شرایطی است که در آن امکان ایجاد ازدحام وجود داشته باشد. چرا که در شبکه‌های سیار ممکن است گره‌های فرستنده و گیرنده همسایه بوده و احتمال ایجاد ازدحام بسیار ضعیف باشد.

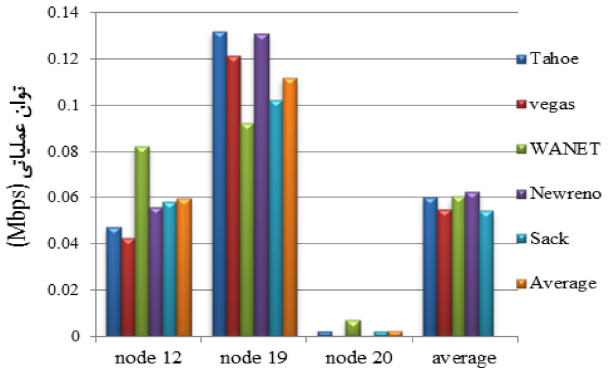
بیشتر گره‌ها در این سناریو ثابت هستند و فقط ۵ گره متحرک وجود دارد. برای بررسی تاثیرات تحرک گره‌ها، ما آن‌ها را با سرعت و مقصد دلخواه حرکت داده‌ایم. به طوری که در زمانی خاص، گره‌های ۵ با ۶ و ۳ با ۴ اقدام به تعویض مکان‌هایشان می‌کنند تا پس از این تغییر نیز همچنان زنجیره‌ی گره‌ها برقرار باشد. در این سناریو با اعمال چندین جریان بر روی تک مسیر زنجیره‌ای، احتمال به وجود آمدن ازدحام در برخی از اتصالات افزایش می‌یابد. ما در ادامه این قسمت، نتایج حاصل از شبیه‌سازی این سناریو را بر اساس معیارهای مذکور مورد بررسی قرار می‌دهیم.

در شکل (۵) نتایج مربوط به توان‌عملیاتی در سه گره ۱۲، ۱۹ و ۲۰ را نشان داده شده است. این گره‌ها مقصد یک تا سه جریان مختلف هستند، و نتایج حاصله تحت نرخ خطا ۰/۰۰۱ بدست آمده است. می‌توان مشاهده نمود در گره ۱۲ یعنی در جائیکه بیشترین جریان ورودی و به دنبال آن بیشترین احتمال ایجاد ازدحام وجود دارد توان عملیاتی TCP-WANET در حدود ۳۰ تا ۵۰ درصد بالاتر از روش‌های دیگر است. ولی در گره ۱۹ که فقط یک جریان ورودی به آن و یک جریان گذری از روی آن به گره بعدی وجود دارد اوضاع کمی متفاوت است و در آن روش‌های TCP-Tahoe و TCP-NewReNo بهتر از روش‌های دیگر عمل می‌کنند. در گره ۲۰ نیز که به علت وجود جریان‌های مختلف در میانه زنجیره، دارای ورودی کمی است، توان عملیاتی TCP-WANET بیشترین مقدار را در مقایسه با روش‌های دیگر دارد. اما در ستون چهارم و با مقایسه میانگین توان عملیاتی مربوط به سه گره یاد شده، مشاهده می‌شود که TCP-NewReNo و TCP-WANET با تفاوت اندکی بیشترین توان عملیاتی را دارند.

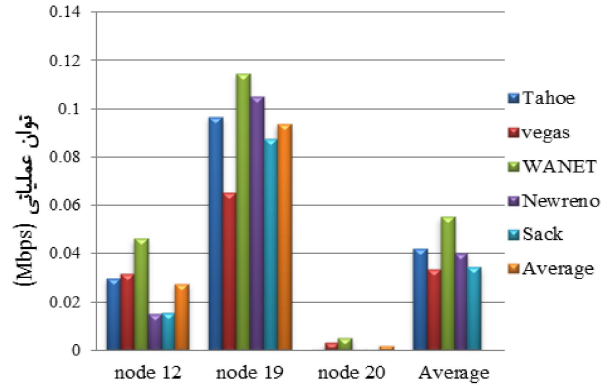
شکل (۶) دارای اطلاعات مشابه شکل قبلی است با این تفاوت که نرخ خطا در اینجا ۰/۰۵ در نظر گرفته شده است. در این شکل و تحت شرایطی که احتمال رخ دادن خطا در آن زیادتر از حالت قبل است توان عملیاتی TCP-WANET در تمامی گره‌ها بیشتر از توان عملیاتی روش‌های دیگر است.

شکل (۷) میانگین توان عملیاتی محاسبه شده گره‌ها را تحت نرخ‌های خطا مختلف و در کنار هم نشان می‌دهد. این نمودار به روشنی بالاتر بودن کارایی TCP-WANET را در محیط‌های دارای نرخ خطای بالا نظیر شبکه‌های بی‌سیم، و نسبت به سایر روش‌ها نشان می‌دهد.

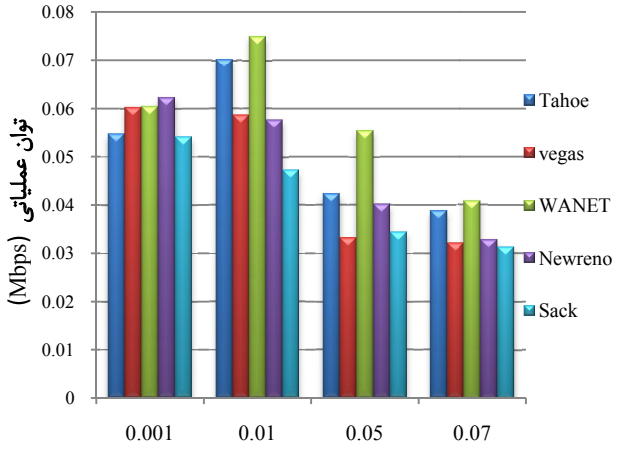
نسبت گم‌شدن بسته TCP-WANET در این سناریو و تحت هر دو نرخ خطا کمتر از سایر روش‌ها است. شکل (۸) نشان می‌دهد که



شکل (۵): توان عملیاتی مربوط به سه گره مختلف گیرنده برای روش‌های مختلف با نرخ خطا ۰/۰۰۱



شکل (۶): توان عملیاتی مربوط به سه گره گیرنده برای روشهای مختلف با نرخ خطا ۰/۰۵



شکل (۷): مقایسه میانگین توان-عملیاتی مربوط به شبکه با چهار نرخ خطای مختلف در سناریو اول

این روش نسبت به TCP-Vegas نیز در شرایط بسیار بهتری قرار دارد. روش TCP-WANET به دلیل ماهیتی که دارد سعی در اجتناب از ایجاد ازدحام می‌کند تا کنترل آن، بسته‌های کمتری را دور می‌اندازد.

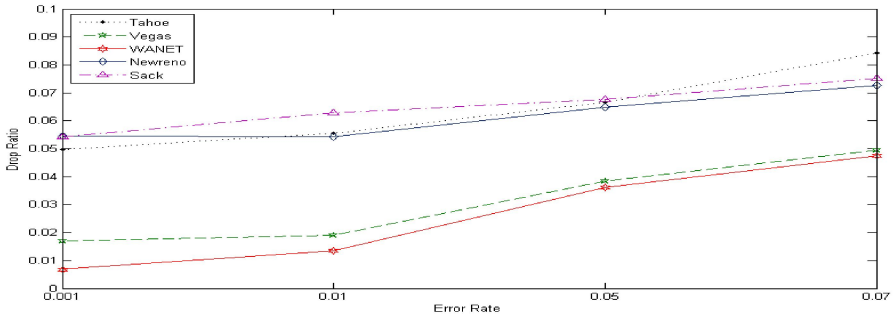
عملکرد مناسب روش‌های مختلف در شبکه‌های بی‌سیم باعث بهبود بسیاری از معیارهای دیگر مرتبط با وضعیت این نوع شبکه‌ها از قبیل نسبت بسته‌های سربار می‌شود. شکل (۹) نشان می‌دهد که TCP-WANET در شرایطی که خطای کمی به میزان ۰/۰۰۱ در شبکه وجود دارد با اندکی اختلاف پس از TCP-NewReNo دارای پایین‌ترین نسبت بسته‌های سربار است. از طرف دیگر در شرایطی که خطا در حد ۰/۰۱ یا ۰/۰۵ و یا ۰/۰۷ است و به محیط واقعی شبکه‌های بی‌سیم نزدیک‌تر است TCP-WANET نسبت به سایر روش‌ها بسته‌های سربار کم‌تری را به شبکه تحمیل می‌کند.

پایین‌تر بودن نسبت بسته‌های سربار در روش پیشنهادی بدین دلیل است که با کاهش نسبت گم‌شدن بسته و کاهش تعداد دفعات ارسال مجدد، نیاز به بررسی اطلاعات و برقراری مسیر در روش مسیریابی AODV نیز کاهش یافته و در نتیجه نسبت سربار کمتر می‌شود. شکل (۹) حاوی این نکته نیز هست که با افزایش نرخ خطا، نسبت بسته‌های سربار نیز افزایش می‌یابد. این افزایش در TCP-WANET تقریباً مشابه با TCP-Vegas انجام می‌پذیرد.

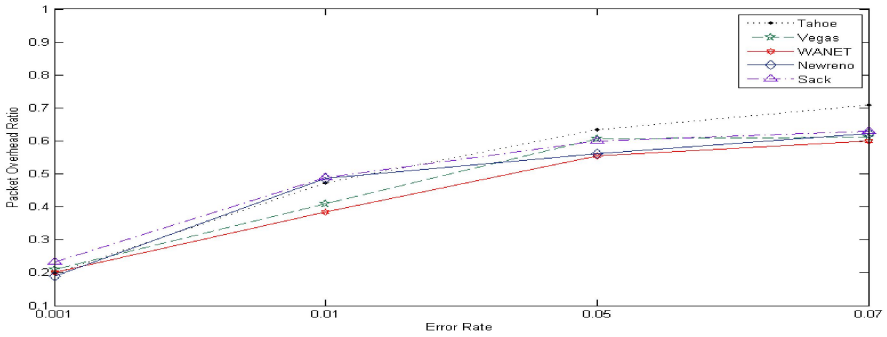
در شکل (۱۰) نشان داده شده است که میزان تاخیر در شبکه با استفاده از TCP-WANET به همراه TCP-Vegas کمتر از سایر روش‌هاست و به مانند دیگر روش‌ها با افزایش نرخ خطا تاخیر نیز افزایش می‌یابد. هنگامی که شرایط ترافیکی شبکه حالت مناسبی داشته باشد صفی طولانی در مسیریاب‌ها به وجود نخواهد آمد و در نتیجه میانگین زمان تاخیر بسته‌ها کاهش خواهد یافت.

۵-۳-۲- نتایج سناریوی دوم

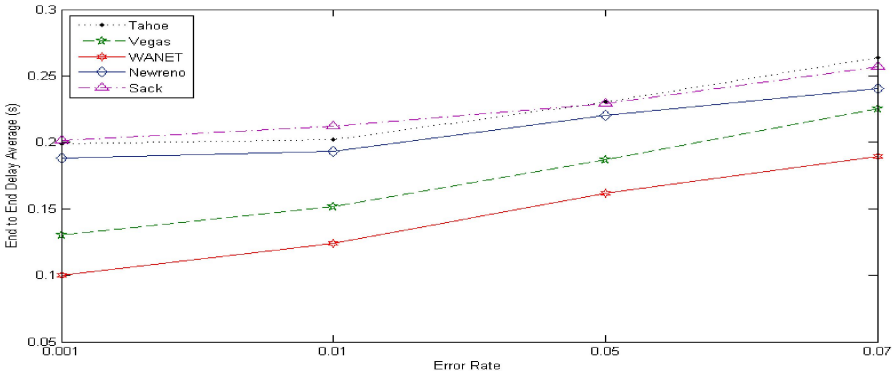
سناریوی دوم که مشخصات مربوط به آن در جدول (۲) به تفصیل بیان شد شبکه‌ای را با ۲۵ گره و پیکربندی ماتریسی شکل به‌وجود می‌آورد. در این شبکه پنج جریان افقی و پنج جریان عمودی در شبکه



شکل (۸): نسبت گم شدن بسته با نرخ‌های مختلف خطا



شکل (۹): نسبت بسته‌های سربار با نرخ‌های مختلف خطا



شکل (۱۰): زمان تاخیر انتها به انتها

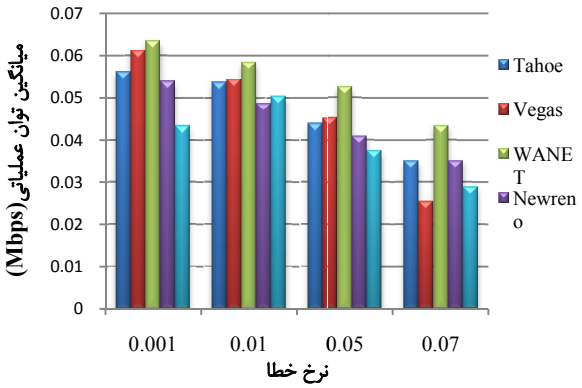
برقرار است و گره‌ها متحرک نیستند تا در اینجا نیز بتوان ترافیک بر روی مسیرهای خاص را اعمال و مورد ارزیابی قرار داد. بیشتر گره‌ها در این سناریو ثابت هستند و فقط ۵ گره متحرک وجود دارد. در ادامه، نتایج حاصل از شبیه‌سازی این سناریو بر اساس معیارهای مذکور مورد بررسی قرار می‌گیرد.

شکل (۱۱) توان عملیاتی حاصله از میانگین توان‌های عملیاتی مربوط به چند گره مختلف را نمایش می‌دهد. به علت رعایت اختصار در اینجا نمودارهای توان عملیاتی تفصیلی هر گره آورده نشده است و فقط به رسم نمودار میانگین توان عملیاتی اکتفا شده است. نتایج نشان می‌دهد در پیکربندی ماتریسی و تحت نرخ‌های خطا متفاوت، TCP-WANET دارای بالاترین میزان توان عملیاتی در مقایسه با روش‌های دیگر است. از سوی دیگر با افزایش نرخ خطا، توان عملیاتی روش TCP-WANET با شیب ملایم‌تری به نسبت روش‌های دیگر کاهش می‌یابد که چنین امری نشان‌دهنده این است که این روش تا حدودی مدیریت بهتری در مواجه با خطاهای غیر از ازدحام دارد و افزایش خطاهای بی‌سیم به نسبت روش‌های دیگر تاثیر کم‌تری بر روی کارآیی آن دارد.

با توجه به اطلاعات موجود در شکل (۱۲) مشاهده می‌شود که در سناریوی دوم TCP-WANET در حدود ۱۵۰٪ نسبت گم‌شدن بسته‌ها را نسبت به TCP-Vegas کاهش داده است.

شکل (۱۳) مربوط به این نسبت بسته‌های سربار به کل بسته‌ها است. در سناریوی دوم نیز TCP-WANET دارای کم‌ترین میزان سربار در بین روش‌های آزموده شده است.

در شکل (۱۴) نیز نشان داده شده است که در سناریوی دوم میزان تاخیر در شبکه با استفاده از TCP-WANET در حدود ۶۵٪ تاخیر TCP-Vegas است. همان طور که بیان شد روش TCP-Vegas در هنگام اجرا به نوعی طول صف موجود در مسیریاب‌ها را کنترل می‌نماید تا از به وجود آمدن ازدحام که باعث بالارفتن میزان تاخیر انتها به انتها می‌شود جلوگیری نماید. از این رو میزان تاخیر در آن نسبت به سایر روش‌ها کم‌تر است. از طرفی چون در روش TCP-WANET از یک آستانه شبکه پویا استفاده می‌شود طول صف دقیق‌تر کنترل می‌شود و



شکل (۱۱): میانگین توان عملیاتی گره‌ها بر حسب نرخ‌های خطای متفاوت در سناریو دوم

با توجه به این نسبت به TCP-Vegas نیز تاخیر کم‌تری وجود خواهد داشت.

۶- جمع‌بندی و کارهای آینده

پروتکل‌های TCP مبنا از الگوریتم‌های مختلفی جهت حل مشکلات مربوط به انتقال داده در محیط‌های بی‌سیم بهره می‌برند. بیشتر الگوریتم‌های کنترل ازدحام در شبکه‌های سیمی عملکرد بهتری را به نسبت شبکه‌های بی‌سیم از خود نشان می‌دهند. علت این امر آن است که TCP در محیط‌های با نرخ خطای بالا در رسانه انتقال، توانایی تشخیص ماهیت خطای رخداده را ندارد و تمام خطاها را ناشی از وجود ازدحام در شبکه می‌داند.

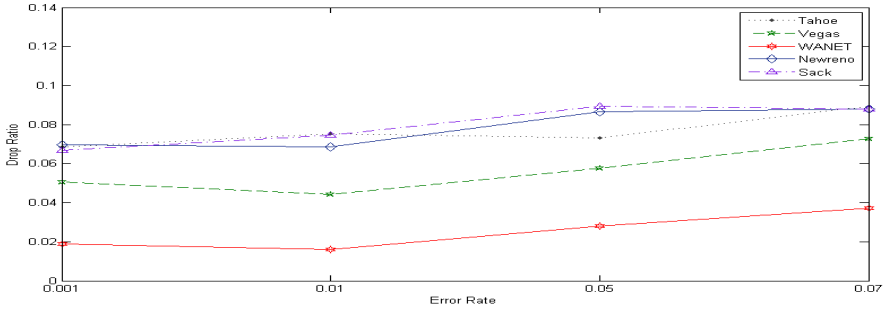
برای حل این مشکل روشی ارائه شد که با انجام برآورد دقیقی از وضعیت ترافیک، می‌تواند نرخ ارسال فرستنده را به‌گونه‌ای تغییر دهد که کم‌ترین رخداد ازدحام و بیشترین بازدهی را به دنبال داشته باشد. الگوریتم پیشنهادی که به نام TCP-WANET ارائه شده خطاهای رخداده را به گونه‌ای مدیریت می‌کند که با هر خطا متناسب با علت آن برخورد می‌شود. بدین صورت که اگر خطا ناشی از ازدحام است نرخ ارسال را کاهش می‌دهد و اگر ناشی از مسائلی غیر از ازدحام نظیر خطاهای بی‌سیم است نرخ ارسال را ثابت نگاه داشته و یا به مقدار کمی کاهش دهد. مشاهده شد که روش پیشنهادی با در نظر گرفتن شرایط بالا به میزان بالاتری از کارآیی در مقایسه با سایر روش‌ها می‌رسد. از نقطه نظر کارایی، روش پیشنهادی موفق شده است تا از اکثر انتقال جددها و تایم‌اوت‌ها و افت پنجره غیرضروری که از خطای بی‌سیم نشات می‌گیرد جلوگیری کند و بدین ترتیب کارایی روی پیوندهای بی‌سیم را بهبود بخشد.

با توجه به گوناگونی زیاد شبکه‌های بی‌سیم از منظر فناوری به کار رفته و پارامترهای مرتبط با آن، ارائه روشی که بتواند بهترین عملکرد را در تمامی شرایط داشته باشد مستلزم مطالعه و بررسی جامعی است. از این رو، برای بهبود کارایی روش ارائه شده در شرایط کلی و نیز در شرایط خاص، مواردی به عنوان محورهای تحقیقاتی آینده و بصورت زیر پیشنهادات می‌گردد:

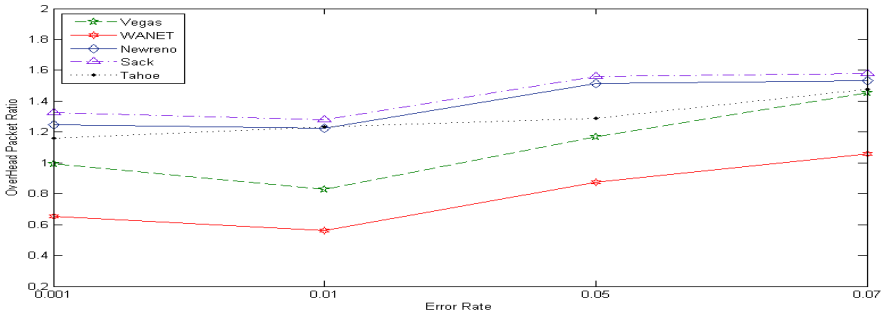
۱. یافتن راهی که بتوان منشا بروز خطا را در تمامی فازهای انتقال تشخیص داد.

۲. با توجه به اینکه این روش در شبکه‌های موردی بی‌سیم مورد بررسی قرار گرفته است آزمایش آن در انواع دیگر شبکه نظیر شبکه‌های ناهمگن می‌تواند حاوی نتایج جالب توجهی باشد.

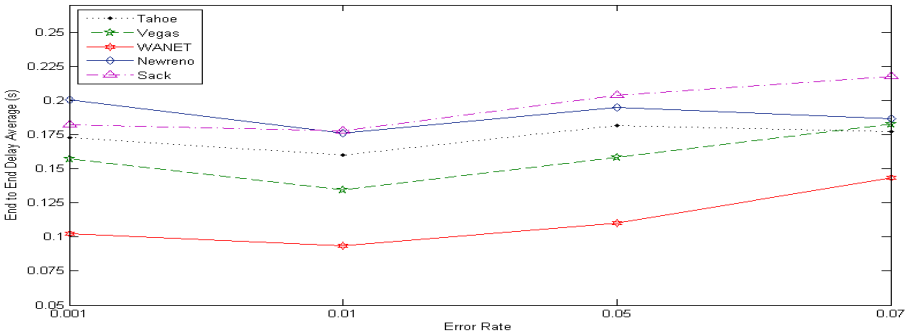
۳. در این مقاله کارایی روش‌های مختلف بر روی شبکه‌های بی‌سیم موردی و با استفاده از الگوریتم مسیریابی AODV مورد ارزیابی قرار گرفته است از این رو می‌توان در آینده عملکرد روش پیشنهادی را در



شکل (۱۲): نسبت گم شدن بسته با نرخ‌های مختلف خطا در سناریو دوم



شکل (۱۳): نسبت بسته‌های سربار در سناریو دوم



شکل (۱۴): میانگین زمان تاخیر انتها به انتها در سناریو دوم

شبکه‌هایی که از سایر الگوریتم‌های مسیریابی بهره می‌برند، مورد آزمون قرار داده و با روش‌های دیگر مقایسه نمود.

مراجع

[1] J. Postel, “Transmission Control Protocol”, RFC 793, 1981.

[2] W. C. Y. Lee, *Mobile Communications Design Fundamentals*, 2nd Edition, John Wiley & Sons, 1993.

[3] K. Dina, M. Handley and C. Rohrs, “Congestion Control for High Bandwidth-Delay Product Networks,” ACM SIGCOMM, 2002.

[4] X. Li and H. Yousefi’zadeh, “Analysis, Simulation, and Implementation of VCP: A Wireless Profiling,” IEEE/ACM Transactions on Networking, Vol. 18, No. 5, pp. 1345-1358, 2010.

[5] S. Kopparty, S. Krishnamurthy, M. Falouts and S. Tripath, “Split-TCP for Mobile Ad-hoc Networks”, IEEE GLOBECOM, Vol. 1, pp. 138-142, 2002.

[6] D. Chiu and R. Jain, “Analysis of the Increase and Decrease Algorithms for Congestion Avoidance in Computer Networks,” Computer Networks and ISDN Systems, Vol. 17, No. 1, 1989.

[7] V. Jacobson, “Congestion Avoidance and Control,” ACM SIGCOMM, pp. 314-329, 1988.

[8] W. Stevens, “TCP Slow Start, Congestion Avoidance, Fast Retransmit and Fast Recovery Algorithms,” RFC 2001, 1997.

[9] S. Floyd, “The New Reno Modification to TCP’s Fast Recovery Algorithm,” RFC 2582, 1999.

[10] M. Mathis, J. Mahdavi, S. Floyd and A. RomaNow, “TCP Selective Acknowledgement Options,” RFC 2018, 1996.

[11] C. Casetti, M. Gerla, S. Mascolo, M. Y. Sanadidi and R. Wang,”TCP Westwood: Bandwidth Estimation for Enhanced Transport over Wireless Links,” ACM MOBICOM, pp. 287-297, 2001.

[12] L. A. Grieco and S. Mascolo,"Performance Evaluation of Westwood+ TCP over WLANs with Local Error Control," IEEE LCN, pp. 440-448, 2003.
[13] X. Kai, T. Ye and N. Ansari, "TCP-Jersey for Wireless IP Communications," IEEE Journal on Selected Areas in Communications, Vol. 22, No. 4, pp. 747-756, 2004.
[14] C. P. Fu and S. C. Liew, "TCP VeNo: TCP Enhancement for Transmission over Wireless Access Networks," IEEE Journal on Selected Areas in Communications, Vol. 21, No. 2, pp. 216-228, 2003.
[15] Brakmo. Lawrence. S, O'Malley. Sean W, Peterson. Larry.L, "TCP Vegas: New Techniques for Congestion Detection and Avoidance," ACM SIGCOMM, pp. 24-35, 1994.
[16] B. Chang, Sh. Lin and J. Jin, "LIAD: Adaptive Bandwidth Prediction Based Logarithmic Increase Adaptive Decrease for TCP Congestion Control in Heterogeneous Wireless Networks," Computer Networks, Vol. 53, No. 14, pp. 2566-2585, 2009.
[17] S. Floyd, "HighSpeed TCP for Large Congestion Windows," RFC 3649, 2003.
[18] T. Kelly, "Scalable TCP: Improving Performance in High Speed Wide Area Networks," ACM SIGCOMM, Vol. 32, No. 2, 2003.
[19] C. Jin, D. X. Wei, S. H. Low, "FAST TCP: Motivation, Architecture, Algorithms, Performance," IEEE INFOCOM, Vol. 4, pp. 2490-2501, 2004.
[20] L. Xu, K. Harfoush, I. Rhee, "Binary Increase Congestion Control for Fast, Long Distance Networks", IEEE INFOCOM, Vol. 4, pp. 2514-2524, 2004.
[21] I. Rhee, L. Xu, "CUBIC: A New TCP-Friendly High-Speed TCP Variants," International Workshop on Protocols for Fast Long-Distance Networks, 2005.
[22] D. Kliazovich, F. Granelli and D. Miorandi, "TCP Westwood+ Enhancement in High-Speed Long-Distance Networks", IEEE ICC, Vol. 2, pp. 710-715, 2006.
[23] K. Tan, J. Song, Q. Zhang and M. Sridharan, "A Compound TCP Approach for High-Speed and Long Distance Networks", IEEE INFOCOM, 2006.
[24] S. M. ElRakabawy, A. Klemm and C. Lindemann, "TCP with Adaptive Pacing for Multihop Wireless Networks," MobiHoc'05, 2005.
[25] J. Chen, Y. Z. Lee, M. Gerla, and M. Y. Sanadidi, "TCP with Delayed ACK for Wireless Networks," Ad Hoc Networks, Vol. 6, pp. 1098–1116, 2008.
[26] R. Oliveira and T. Braun, "A Smart TCP Acknowledgment Approach for Multihop Wireless Networks," IEEE Transactions on Mobile Computing, Vol. 6, No. 2, 2007.
[27] C. Parsa and J. J. Garcia-Luna-Aceves, "Improving TCP Congestion Control over Internets with Heterogeneous Transmission Media," IEEE ICNP, 1999.
[28] The Network Simulator ns-2. <http://www.isi.edu/nsnam/ns/>.

زیرنویس‌ها

[1] Transmission Control Protocol
[2] Bit Error Rate
[3] Explicit Congestion Notification
[4] Active Queue Management
[5] Congestion Experienced
[6] Explicit Congestion control Protocol
[7] Variable-Structure Congestion control Protocol
[8] Sliding Window
[9] Additive Increase Multiplicative Decrease
[10] Duplicative AckNowledgment
[11] Congestion Window
[12] Round Trip Time
[13] Selective AckNowledgment
[14] Available Bandwidth Estimation
[15] Additive Increase ADaptive Decrease
[16] Grieco & Moscolo
[17] Friendly
[18] Lgarithmic Increase Adaptive Decrease
[19] Scalable
[20] Network Bandwidth
[21] Modified Exponential Weight Moving Average (M-EWMA)
[22] RTO (Retransmission Timeout)
[23] Packet Drop Ratio