علوم و مهندسی کامپیوتر
نشریه علمی پژوهشی انجمن کامپیوتر ایران
مجلد ۸، شماره ۱ و ۳ (الف)، ۱۳۸۹
صفحات ۱-۷
مقاله کوتاه

بهبود پروتکل مسیریابی Tora در شبکه‌های ویژه سیار بکمک الگوریتم کولونی مورچه‌ها

رعنا غزالی مجتبی صفاری علی موقر رحیم‌آبادی

دانشکده مهندسی کامپیوتر، دانشگاه صنعتی شریف، تهران، ایران

چکیده

الگوریتم TORA یکی از پروتکل‌های بر حسب نیاز مسیریابی در شبکه ویژه سیار می‌باشد. ما پیشنهاد ترکیب پروتکل TORA با الگوریتم Ant را داده و هدف از ارائه و بررسی پروتکل Ant-TORA، استفاده از الگوریتم هوشمند Ant برای تشخیص و استفاده از مسیرهای کوتاه‌تر از نظر زمانی است. همچنین به‌کارگیری چنین الگوریتمی که یک راه حل چندمسیره[1] برای مسیریابی در شبکه‌های ویژه سیار فراهم می‌کند، تحمل‌پذیری در برابر خطا را نیز افزایش خواهد داد و در نتیجه موجب بهبود کارایی شبکه می‌شود. کارایی پروتکل جدید Ant-TORA را با پروتکل پایه TORA در قالب سناریوهای شبیه‌سازی بررسی کرده‌ایم. همانطور که خواهیم دید پروتکل Ant-TORA کارایی قابل توجهی نسبت به TORA نشان می‌دهد.

کلمات کلیدی: شبکه، ویژه سیار، Ant_TORA-TORA.

۱- مقدمه

Mobile ad hoc network (MANET) مجموعه‌ای از نودهای متحرک بی‌سیم است که بصورت پویا و بدون مدیر مرکزی می‌باشد.

۱-۱- الگوریتم TORA

خصوصیت اصلی TORA متمرکز ساختن پیام‌های کنترلی در یک مجموعه بسیار کوچک از نودهای نزدیک به محلی که تغییرات توپولوژی در آن اتفاق افتاده است می‌باشد. برای دستیابی به این موضوع نودها اطلاعات مسیریابی در بازه نودهای مجاور را نگهداری می‌کنند. پروتکل سه وظیفه اساسی ایجاد مسیر و ترمیم مسیر و پاک کردن مسیر را دارد. ایجاد مسیر با استفاده از بسته‌های QRY، UPD انجام می‌شود. الگوریتم ایجاد مسیر با قرار دادن یک مجموعه صفر برای ارتفاع نود مقصد و مجموعه خالی برای ارتفاع نودهای دیگر شروع می‌شود. مبدا بسته QRY که شناسه نود مقصد در آن قرار دارد را پخش می‌کند. یک نود با ارتفاع غیرخالی با استفاده از یک بسته UPD که ارتفاع خودش را در آن قرار می‌دهد پاسخ می‌دهد. نودی که یک بسته UPD را دریافت می‌کند ارتفاع خودش را به یکی بیشتر از ارتفاع تولید کننده UPD تنظیم می‌کند. نود با بیشترین ارتفاع به عنوان نود ابتدایی مسیر و نود با کمترین ارتفاع به عنوان نود انتهایی مسیر مطرح می‌شود. در این روش یگ گراف غیرحلقوی (DAG) از مبدا به مقصد ساخته می‌شود. وقتی که یک نود حرکت می‌کند و مسیر DAG شکسته می‌شود ترمیم مسیر برای دوباره ساختن DAG در نود مقصد مورد نیاز است. در فاز از بین بردن مسیر TORA یک بسته پاکسازی (CLR) را در کل شبکه برای از بین بردن مسیرهای غیرمعتبر بصورت Flooding ارسال می‌نماید.

فواید TORA کاهش پیام‌های کنترل به همسایه‌ها زمانیکه تغییرات در توپولوژی رخ می‌دهد می‌باشد. مزیت دیگر آن این است که multicasting را حمایت می‌نماید هر چند که جز عملیات اصلی نمی‌باشد. TORA برای اتصالات با وزن سبک و سازگار با multicast استفاده می‌شود. عیب این روش این است که ممکن است مسیرهای موقت نامعتبر ایجاد نماید.

۱-۲- الگوریتم کولونی مورچه

الگوریتم کولونی مورچه‌ها بر پایه کولونی مورچه‌ها در طبیعت می‌باشد. مورچه‌ها قادر هستند کوتاهترین مسیر بین لانه و منبع غذا را بکمک ترشح فرومون بیابند. در واقع مورچه‌ها در حین راه رفتن ماده‌ای بنام فرومون ترشح می‌نمایند و سایر مورچه‌ها نیز از طریق فرومون ترشح شده توسط مورچه‌های دیگر مسیر را می‌یابند. فرض کنیم مورچه‌ها به نقطه‌ای برسندکه باید تصمیم بگیرند سمت چپ بروند یا سمت راست و چون نمی‌دانند کوتاهترین مسیر کدام است در ابتدا بصورت تصادفی سمت چپ یا راست را انتخاب می‌کنند و فرض کنیم ۵۰٪ سمت چپ و ۵۰٪ سمت راست را انتخاب نمایند و تمام مورچه‌ها با سرعت ثابت حرکت می‌کنند، کوتاهترین مسیر توسط مورچه‌های بیشتری پیموده می‌شود لذا فرومون ترشح شده در آن مسیر بیشتر است. پس از دوره زمانی کوتاه تفاوت فرومون موجود در دو مسیر زیاد شده بطوریکه در دفعات بعدی مورچه‌ها مسیری را که فرومون بیشتری دارد با احتمال بالاتری انتخاب می‌نمایند.

۱-۳- الگوریتم Ant-TORA

مولفه اصلی پروتکل Ant-TORA، جدولی به نام جدول Ant[۲] است. از جدول Ant برای نگهداری مسیرهای مختلف و تعیین مسیرهای مناسب به سمت مقصد خاص در مواقع لازم استفاده می‌شود. هر سطر این جدول شامل چهار عنصر است: Destination Node: این عنصر، گرهی را مشخص می‌کند که دانستن مسیری به آن مد نظر است. باید توجه داشت که این عنصر در بین سطرهای جدول Ant یکتا نیست. یعنی ممکن است در جدول ما چندین سطر مختلف با Destination Node یکسان وجود داشته باشند که هر کدام، مسیر خاصی را به سمت این گره مشخص می‌کنند.

Next Node: این عنصر، اولین گره از یک مسیر به سمت گره Destination Node را نشان می‌دهد. به عبارت دیگر ما با دیدن یک سطر از جدول Ant در می‌یابیم که برای رسیدن به گره Destination Node باید از گره Next Node بگذریم. به این ترتیب، دو عنصر Destination Node و Next Node با یکدیگر یک سطر یکتا را در جدول مشخص می‌کنند.

Pheromone: این عنصر، مقدار فرومون موجود در مسیر Destination Node از گره Next Node را در زمان Last Update Time مشخص می‌نماید. به عبارت دیگر، مقدار فرومون معیاری برای سنجش میزان تازگی و میزان استفاده از مسیر به دست می‌دهد. این متغیر از نوع حقیقی مثبت است.

Last Update Time: این عنصر، آخرین زمانی را نشان می‌دهد که این سطر از جدول به‌روز رسانده شده و مقدار فرومون آن تغییر کرده است. به عبارت دیگر با نگاه کردن به یک سطر ما می‌فهمیم که در لحظه Last Update Time، میزان فرومون فلان مقدار بوده و پس از آن تغییری داده نشده است. این متغیر یک عدد صحیح مثبت است که زمان را بر حسب میلی ثانیه نشان می‌دهد.

ما از دو ثابت دیگر به نام‌های آهنگ کاهش[۳] یا DR و بزرگترین زمان ممکن[۴] یا MT در کنار جدول Ant استفاده می‌کنیم. با به کارگیری این دو ثابت ما می‌توانیم مقدار فعلی فرومون را در هر لحظه تعیین کنیم. ثابت DR، مقداری بزرگتر از یک دارد. ثابت MT هم هر مقداری می‌تواند داشته باشد؛ اما برای عملیاتی شدن سیستم باید مقدار آن را در محدوده طول زمان شبیه‌سازی در نظر گرفت. مقدار MT,DR به ترتیب 1.05 و 200000 فرض شده است.

به عنوان یک قانون کلی، مقدار فرومون یک مسیر در هر لحظه برابر است با:

$$\text{Current Pheromone} = \frac{\text{Pheromone}}{\text{DR}} \times \text{MAX}\left(0, \left(1 - \frac{\text{Current Time - Last Update Time}}{\text{MT}}\right)\right) \quad (۱)$$

با دقیق‌تر شدن در فرمول بالا در می‌یابیم که هر چه آهنگ کاهش و حداکثر زمان بیشتر باشند، مقدار فرومون در طول زمان با سرعت بیشتری کاهش می‌یابد. همچنین در صورتی که فاصله زمانی زمان فعلی و زمان آخرین به‌روزرسانی (Last Update Time) بیشتر از حداکثر زمان (MT) شود، مقدار فرومون صفر خواهد شد.

نکته قابل توجهی که باید در نظر گرفته شود این است که پس از هر بار دسترسی به یک سطر و محاسبه مقدار فعلی فرومون، می‌توان محتوای آن سطر را در جدول Ant به‌روز رساند.

۲- مکانیزم واردسازی وبروزرسانی اطلاعات در جدول Ant

اطلاعات مسیریابی در پروتکل Ant-TORA در سه زمان وارد جدول Ant و بروزرسانی می‌شوند. در ادامه این سه وضعیت شرح داده می‌شوند:

دریافت پیغام درخواست مسیر: هنگامی که گره میانی C بسته درخواست مسیری را که متعلق به گره A است از گره B دریافت می‌کند، در صورتیکه قبلا زوج (A, B) را به عنوان (Destination Node, Next Node) در جدول Ant نیابد، چهارتایی زیر را در جدول اضافه می‌کند:

(A, B, 1, Current Time)

در صورتیکه این زوج مرتب قبلا وجود داشته باشد، چهارتایی زیر را در جدول اضافه می‌کند:

(A, B, 1+Δ, Current Time)

که Δ مقدار پیشین فرومون مسیر C به A از طریق B است و همانند فرمول بالا قابل حصول می‌باشد.

دریافت پیغام بروزرسانی مسیر: هر گاه گره C پیغام بروزرسانی مسیری را که از گره مقصد X صادر شده است، از گره D دریافت دارد، در صورتیکه قبلا زوج (X, D) را به عنوان (Destination Node, Next Node) در جدول Ant نیابد، چهارتایی زیر را در جدول اضافه می‌کند:

(X, D, 1, Current Time)

در صورتیکه این زوج مرتب قبلا وجود داشته باشد، چهارتایی زیر را در جدول اضافه می‌کند:

(X, D, 1+Δ, Current Time)

که Δ مقدار پیشین فرومون مسیر C به X از طریق D است و همانند فرمول بالا قابل حصول می‌باشد.

همچنین اگر این پیغام بروزرسانی مسیر از طرف یکی از گره‌های میانی صادر شده باشد نیز چنین درج اطلاعاتی در جدول Ant انجام می‌شود؛ با این تفاوت که در این حالت از آنجا که یک گره میانی این بروزرسانی را صادر کرده است، باید جلوی اشتباه در فهم کوتاه‌ترین مسیر را بگیریم. یعنی ممکن است پیغام درخواست و به تبع آن پیغام بروزرسانی متناظر با آن درخواست از مسیری کوتاه‌تر، دیرتر از پیغام بروزرسانی که مربوط به مسیر طولانی‌تری بوده، ولی از گره میانی صادر شده است به گره C برسد. در نتیجه ما می‌توانیم از پارامتر طول مسیر هم برای محاسبه متغیر فرومون استفاده کنیم.

دریافت بسته پاک کننده مسیرهای نامعتبر: هر گاه گره C بسته پاک کننده را که متعلق به گره A است و به مقصد گره X ارسال شده است را برای تحویل به

گره میانی بعدی D از گره B تحویل بگیرد در صورتیکه هر یک از زوج‌های (X, D) و (A, B) را در جدول Ant بیابد، چهارتایی‌های زیر را در جدول اضافه می‌کند:

$\Delta_1=\Delta_1 *0.1$
(A, B, Δ_1, Current Time)
$\Delta_2=\Delta_2 *0.1$
(X, D, $1+\Delta_2$, Current Time)

Δ_1 وΔ_2 مقادیر پیشین مسیرهای متناظر هستند. هرگاه میزان فرومون در جدول Ant صفر شود چهارتایی متناظر با آن از جدول حذف می‌گردد.

۳- مکانیزم بازیابی اطلاعات از جدول Ant برای یافتن مسیر

گره نوعی C در چند حالت به جدول Ant خود مراجعه می‌کند و از اطلاعات آن استفاده می‌کند. این حالات به ترتیب شرح داده خواهند شد.

هنگامی که گره C بخواهد بسته‌ای به گره D ارسال نماید. در این حالت در پروتکل TORA، گره C شروع به ایجاد و ارسال پیغام درخواست مسیر می‌کند. اما در پروتکل پیشنهادی Ant-TORA، ابتدا جدول Ant برای یافتن تمامی سطرهایی که Destination Node آن‌ها برابر D است، جستجو می‌شود به این ترتیب، جستجوی بسیار وقت‌گیر حجم زیاد اطلاعات سازمان‌نیافته، با جستجوی سریع یک ستون از جدول Ant با اندیس D جایگزین شده است.

اگر یک سطر در جدول Ant یافت شود، باید توجه داشت که با توجه به مکانیزم درج در جدول Ant، اگر مسیری به یک مقصد خاص در این جدول یافت شود، به احتمال زیاد مسیر متناظر با آن سطر وجود خواهد داشت.

پس از یافتن مسیری یا مسیرهای جایگزینی به سمت گره D (مثلا از طریق گره E) در جدول Ant، حافظه نهان را جستجو نکنیم. به جای آن، آدرس گره بعدی را در بسته برابر گره E قرار داده و بسته را ارسال نماییم. گره E پس از دریافت این بسته به دلیل ناقص بودن مسیر مبدا آن، دوباره جدول Ant خود را جستجو می‌نماید و نظیر گره قبل، بسته را به گره بعدی تحویل می‌دهد. این کار آنقدر تکرار می‌شود تا بسته به مقصد برسد.

اما اگر چند سطر در جدول Ant یافت شود، فرومون فعلی هر یک از این مسیرها بر اساس فرمول ۱ محاسبه می‌شود. مسیری که بیشترین فرومون فعلی را دارد، برای ارسال بسته‌ها انتخاب می‌شود.

ما برای بهبود بیشتر کارایی TORA در هنگام شکستگی پیوند، Ant-TORA را بدین گونه طراحی کرده‌ایم که گره C پس از تشخیص شکستگی پیوند جدول Ant خود را جستجو می‌کند. اگر مسیر دیگری به سمت مقصد با فرومون بزرگتر از صفر یافت شد. اگر چند مسیر به سمت مقصد یافت شد، فرومون‌های فعلی آن‌ها را محاسبه می‌نماییم و مسیری را که بیشترین فرومون را دارد و در نتیجه تازه‌ترین مسیر است و بیشترین استفاده از آن شده است را انتخاب نماید. در صورتی که هیچ مسیری در جدول Ant یافت نشود، فرایند درخواست مسیر برای یافتن مسیری برای بسته داده‌ای فراخوانی می‌شود.

هنگامی که گره میانی C، پیغام درخواست مسیر گره A را دریافت کرده است و مسیرهای مختلفی به گره مقصد X در جدول Ant خود سراغ دارد. در این حالت گره C، فرومون این مسیرها را محاسبه می‌نماید و مسیرهای متناظر با هر یک از این سطرها را پیدا می‌کند. برای هر مسیر، یک پیغام بروزرسانی مسیر به گره مبدا فرستاده می‌شود. طبیعی است که متناظر با هر پیغام بروزرسانی، وزن جاری مسیرها هم بروزرسانی خواهد شد. گره مبدا پس از دریافت این بروزرسانی‌ها هم جدول Ant خود را به‌روز می‌رساند و هم بهترین مسیر فعلی را جهت ارسال بسته‌های داده‌ای انتخاب می‌کند.

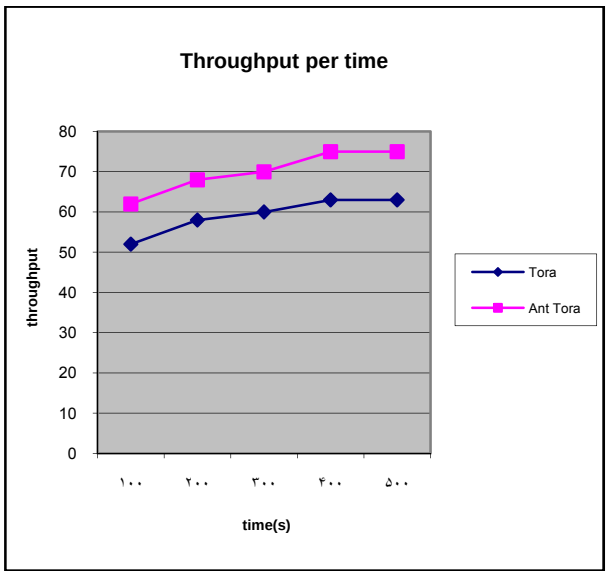
۴- نتایج شبیه‌سازی

از نرم‌افزار شبیه‌ساز شبکه NS2 (Network Simulator) نسخه ۲.۲۷ انجام شده است. برای شبیه‌سازی پروتکل جدید و مقایسه آن با پروتکل TORA از چندین سناریو استفاده شده است. در این سناریوها محدوده محیط شبیه‌سازی مربعی به طول ۱۰۰۰ متر فرض شده است که شامل ۳۰ نود متحرک است. همچنین نحوه حرکت نودها از الگوی Random Waypoint پیروی می‌کند. سرعت حرکت نودها دارای توزیع یکنواخت در یک بازه معین صفر تا m/s۲۰ است. نود با سرعت مشخص شروع به حرکت به سمت مقصد می‌کند، پس از رسیدن به مقصد برای مدتی در آنجا می‌ماند که به این زمان Pause time (زمان توقف) می‌گویند؛ سپس نقطه دیگری را به عنوان مقصد بعدی انتخاب می‌نماید. نودها این کار را تا پایان شبیه‌سازی تکرار می‌کنند. زمان توقف m/s۲۵۰ فرض شده است و از مدل CBR جهت ترافیک استفاده شده است.

۴-۱- توان عملیاتی

توان عملیاتی (Throughput) عبارت است از نسبت بسته‌های دریافت شده به بسته‌های ارسال شده.

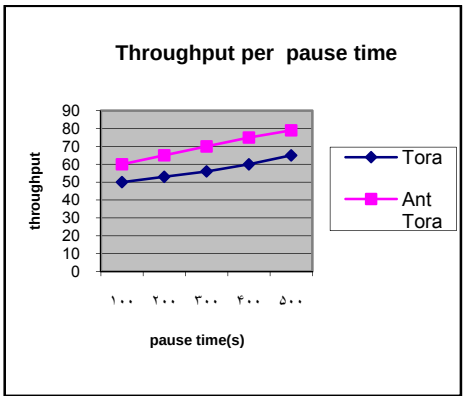
در نمودار زیر توان عملیاتی دو پروتکل TORA، Ant-TORA در طول زمان شبیه‌سازی با هم مقایسه شده‌اند. همانطور که در شکل پیداست توان عملیاتی دو پروتکل TORA و Ant-TORA با گذشت زمان افزایش پیدا می‌کند؛ علت آن است که به مرور زمان تعداد مسیرهای موجود افزایش یافته و نودها بسته‌های کمتری را از دست می‌دهند.



شکل ۱- نمودار توان عملیاتی بر حسب زمان

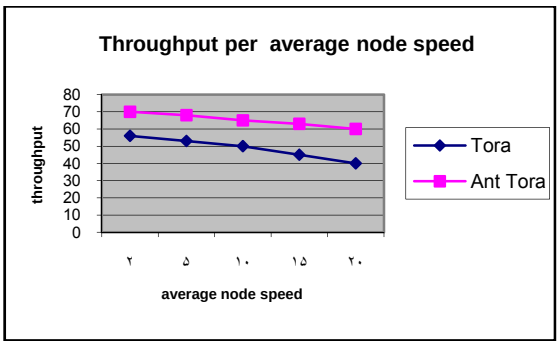
توان عملیاتی با افزایش زمان توقف نودها، افزایش می‌یابد. زیرا با افزایش زمان توقف، نودها تحرک کمتری دارند و در نتیجه شکست پیوند کمتر پیش می‌آید پس توان عملیاتی بالا می‌رود. همانگونه که نمودار مشخص می‌کند در حالتی که توقف نودها کم است توان عملیاتی Ant-TORA بهتر از TORA است. زیرا در این

حالت شکست پیوند به تناوب رخ می‌دهد و Ant-TORA چون مسیرهای دیگری به مقصد دارد نیاز به ارسال مجدد بسته درخواست ندارد. بنابراین توان عملیاتی آن بالاتر است ولی در حالتی که نودها زمان توقف بیشتری دارند شکست پیوند کم اتفاق می‌افتد. بنابراین عملکرد دو پروتکل شبیه به هم می‌شود.



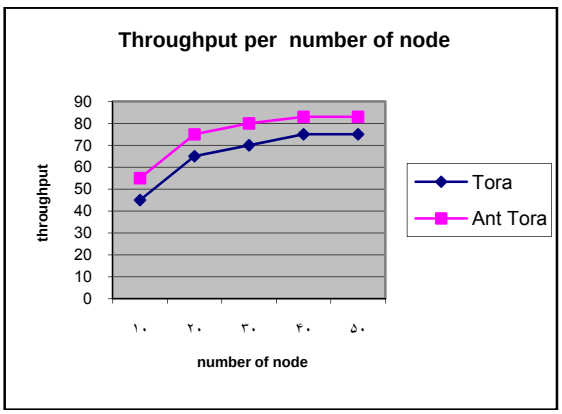
شکل۲- نمودار توان عملیاتی بر حسب زمان توقف

طبق نمودار، توان عملیاتی با افزایش سرعت نودها، کاهش می‌یابد زیرا با افزایش سرعت، شکست پیوند بیشتر رخ می‌دهد. پس توان عملیاتی پایین می‌آید. در حالتی که سرعت نودها زیاد شود توان عملیاتی Ant-TORA بهتر از TORA است. زیرا Ant-TORA مسیرهای مختلفی به مقصد دارد.



شکل ۳- نمودار توان عملیاتی برحسب سرعت نودها

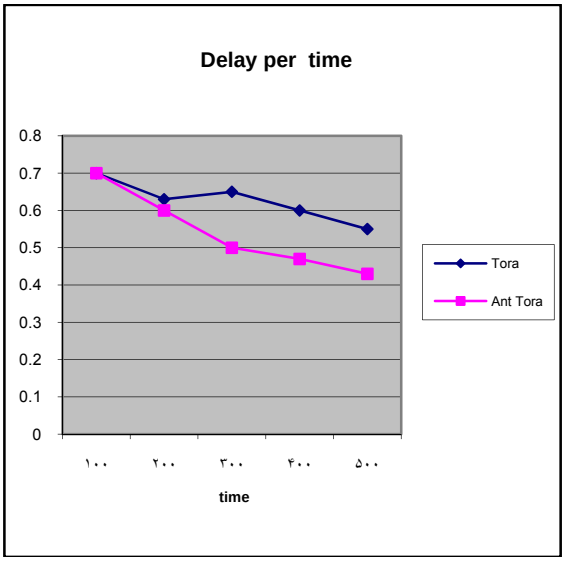
توان عملیاتی با افزایش تعداد نودها، افزایش می‌یابد زیرا احتمال وجود مسیرهای بیشتر و کوتاه‌تر افزایش می‌یابد. براساس نمودار، توان عملیاتی Ant-TORA و TORA تقریباً یکسان است.



شکل ۴- نمودار توان عملیاتی بر حسب تعداد نودها

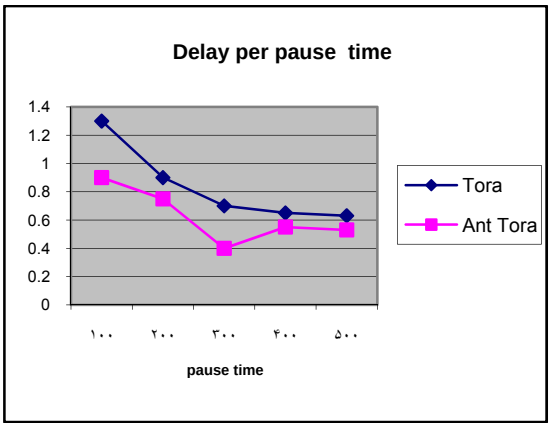
۴-۲- تاخیر

تاخیر (Delay) عبارت است از متوسط اختلاف زمانی بین ارسال بسته‌ها در نود مبدا تا دریافت بسته‌ها در نود مقصد که به آن اصطلاحاً تاخیر انتها به انتها (End-to-End Delay) نیز می‌گویند. تاخیر ارسال بسته‌ها با افزایش زمان کاهش می‌یابد (البته نه به صورت یکنواخت). علتی که می‌توان برای آن متصور بود این است در جدول Ant به مرور زمان مسیرهای اصلی مشخص می‌گردند. در نتیجه نودها سریعتر می‌توانند مسیری به مقصد بیابند و واضح است که هر چه تعداد مسیرها بیشتر باشد احتمال یافتن مسیر کوتاه‌تر افزایش می‌یابد؛ این موضوع باعث کاهش تاخیر می‌شود. آنچه از نمودار پیداست این است که تاخیر Ant-TORA از TORA کمتر است.



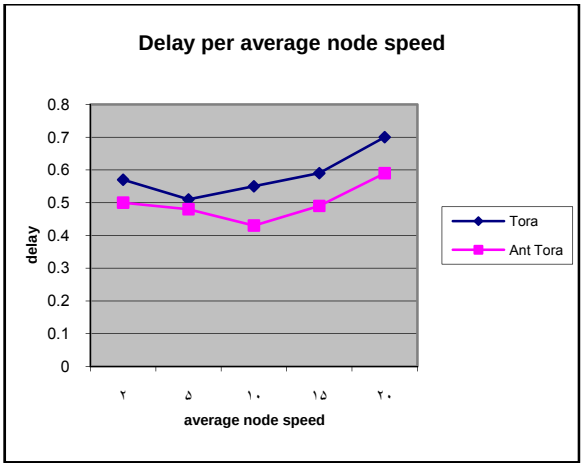
شکل ۵- نمودار تاخیر بر حسب زمان

نمودار تاخیر بر حسب زمان توقف نودها به صورت زیر است. تاخیر ارسال بسته‌ها با افزایش زمان توقف نودها، کاهش می‌یابد. زیرا با افزایش زمان توقف، نودها تحرک کمتری دارند و در نتیجه گسست ارتباط کمتر پیش می‌آید. پس متوسط زمان تاخیر کم می‌شود. همانگونه که نمودار مشخص می‌کند تاخیر Ant-TORA بهتر از TORA است (کمتر از آن است) زیرا Ant-TORA مسیرهای دیگری به مقصد دارد و نیاز به ارسال مجدد پیام درخواست ندارد. به جای آن مستقیما مسیر دیگری را جایگزین کرده و باعث صرفه‌جویی در وقت می‌شود.



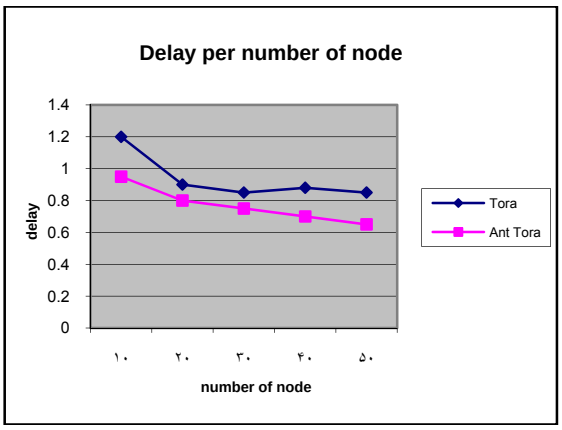
شکل ۶- نمودار تاخیر بر حسب زمان توقف

نمودار تاخیر بر حسب متوسط سرعت نودها در زیر آمده است. طبق نمودار، تاخیر با افزایش سرعت نودها، افزایش می‌یابد زیرا با افزایش سرعت، شکست پیوند بیشتر رخ می‌دهد پس تاخیر در ارسال بسته‌ها بیشتر می‌شود. همانگونه که نمودار مشخص می‌کند تاخیر Ant-TORA کمتر از TORA است زیرا Ant-TORA مسیرهای مختلفی به مقصد دارد و همچنین احتمال یافتن مسیر کوتاهتری به مقصد در آن بیشتر است؛ همین امر باعث می‌شود بسته‌ها مدت زمان کمتری در صف ارسال یا در طول مسیر ارسال باشند.



شکل ۷- نمودار تاخیر بر حسب میانگین سرعت نودها

تاخیر ارسال بسته‌ها بر حسب تعداد نودها در نمودار زیر آورده شده است تاخیر ارسال بسته‌ها با افزایش تعداد نودها، کاهش می‌یابد. زیرا در این حالت احتمال وجود مسیرهای مختلف و کوتاه بیشتر است. بنابراین بسته‌ها زیاد معطل نمی‌مانند. همانند نمودارهای قبل، در این نمودار نیز مشخص است که تاخیر Ant-TORA از TORA کمتر است. در حالت کلی می‌توان گفت که تاخیر ارسال بسته‌ها در پروتکل Ant-TORA از پروتکل TORA کمتر است زیرا داشتن مسیرهای مختلف به مقصد باعث کاهش زمان انتظار ارتباط می‌شود.

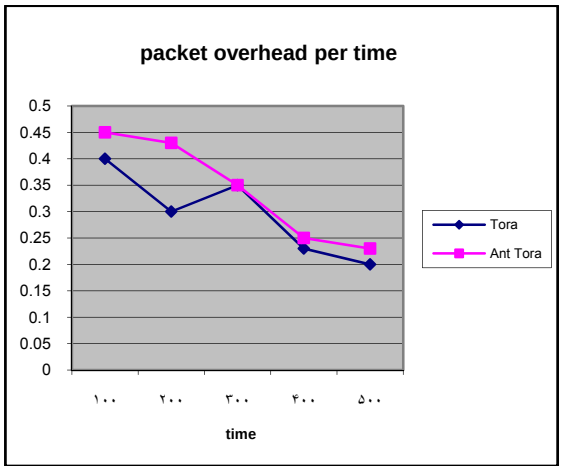


شکل ۸- نمودار تاخیر بر حسب تعداد نودها

۴-۳- سربار بسته‌ها

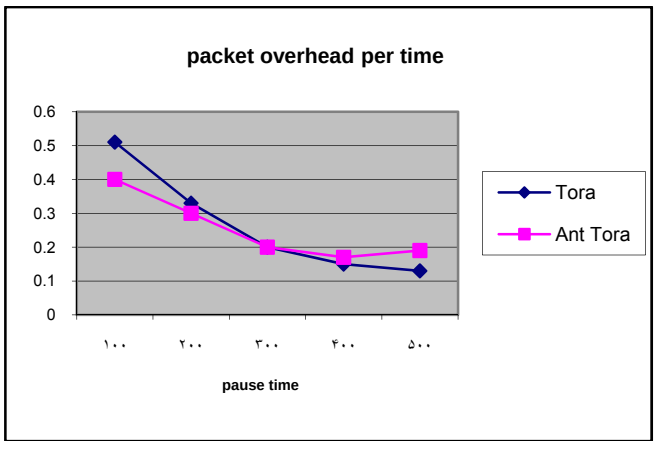
سربار بسته‌ها (Packet Overhead) عبارت است از نسبت بسته‌های کنترلی ارسال شده توسط نودها به کل بسته‌های ارسالی. بسته‌های کنترلی به بسته‌هایی گفته می‌شود که برای ساخت و ترمیم مسیر استفاده می‌شوند. بسته‌های درخواست مسیر، بروزرسانی مسیر، پاک کننده مسیرهای نامعتبر و بسته‌های Ant نمونه‌هایی از بسته‌های کنترلی هستند.

همانطور که در شکل پیداست سربار بسته‌ها با گذشت زمان کاهش می‌یابد؛ علت آن است که به مرور زمان تعداد مسیرهای موجود در جدول Ant افزایش پیدا می‌کند. بنابراین نیاز کمتری به ارسال بسته‌های کنترلی برای یافتن مسیر دارند. نکته دیگری که مشاهده می‌گردد این است که سربار بسته‌ها در Ant-TORA اندکی بیشتر از TORA است. علت این امر، ارسال بسته‌های Ant در شبکه است که باعث افزایش نسبی بسته‌های کنترلی می‌شود.



شکل ۹- نمودار سربار بر حسب زمان

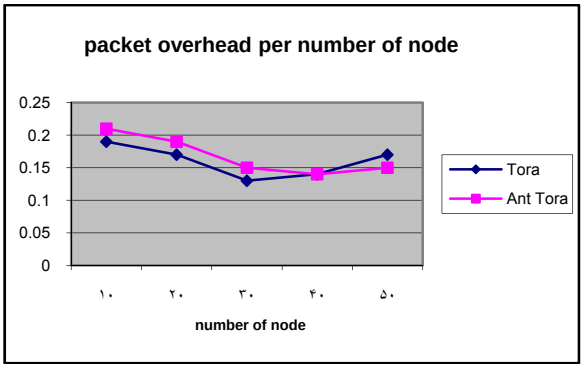
نمودار سربار بسته‌ها بر حسب زمان توقف نودها به صورت زیر است سربار بسته‌ها با افزایش زمان توقف نودها، کاهش می‌یابد. زیرا با افزایش زمان توقف، نودها تحرک کمتری دارند و در نتیجه شکست پیوند کمتر پیش می‌آید. پس نیاز به ترمیم مسیر یا درخواست مسیر جدید کم می‌شود و این یعنی کاهش ارسال بسته‌های کنترلی. همانگونه که نمودار مشخص می‌کند وقتی زمان توقف نودها کم است، سربار بسته‌ها در Ant-TORA کمتر از TORA است زیرا Ant-TORA مسیرهای دیگری به مقصد دارد و نیاز به ارسال مجدد درخواست مسیر ندارد.



شکل ۱۰- نمودار سربار بر حسب زمان توقف

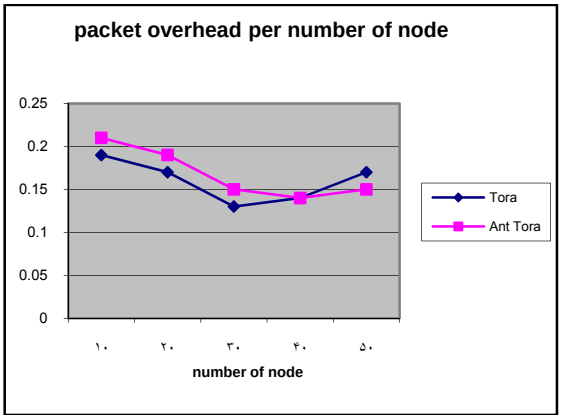
سربار بسته‌ها با افزایش سرعت نودها، افزایش می‌یابد؛ همچنین بر اساس نمودار، وقتی که سرعت حرکت نودها زیاد است، سربار بسته‌ها در پروتکل Ant-TORA کمتر از پروتکل TORA است. ولی وقتی که سرعت حرکت نودها کم است، سربار بسته‌ها در هر دو پروتکل تقریبا یکسان می‌باشد. علت این امر، شکست

سریع پیوندها در سرعت زیاد است. وقتی که شکست پیوند رخ می‌دهد، برای ترمیم مسیر اگر مسیر جایگزینی موجود نباشد، نیاز است که بسته‌های کنترلی ارسال شوند.



شکل ۱۱- نمودار سربار بر حسب میانگین سرعت نودها

در اینجا نمودار سربار بسته‌ها برای تعداد متغیر نودها رسم شده است. از نمودار می‌توان استنباط کرد که با افزایش تعداد نودها از ۱۰ نود به ۳۰ نود، سربار بسته‌ها کاهش می‌یابد. ولی با افزایش بیشتر نودها به ۶۰ نود، سربار بسته‌ها اندکی افزایش می‌یابد. این مطلب برای هر دو پروتکل تقریبا یکسان است. علتی که برای این امر وجود دارد این است که وقتی تعداد نودها کم است، تراکم نودها کم است. بنابراین احتمال اینکه نودها در فضای قابل دسترس یکدیگر قرار گیرند، کم می‌شود و برای ایجاد ارتباط نیاز به ارسال بسته‌های کنترلی بیشتری است. وقتی تعداد نودها از حد خاصی بیشتر می‌شود، هم شکست در پیوندها زیاد اتفاق می‌افتد هم بسته‌های کنترلی در محدوده گسترده‌تری پخش می‌شوند که همین امر باعث افزایش سربار بسته‌ای می‌شود.



شکل ۱۲- نمودار سربار بر حسب تعداد نودها

۵- نتیجه‌گیری

برای بهبود توان عملیاتی و تاخیر ارسال بسته‌ها ، ما پیشنهاد ترکیب پروتکل TORA با الگوریتم Ant را داده و ضمن معرفی این الگوریتم مکانیزم درخواست مسیر و بروزرسانی اطلاعات جداول مسیریابی و روند مسیریابی بکمک این روش را شرح دادیم.

نتایج شبیه‌سازی نشان داده است که روش ANT_TORA نسبت به TORA درحالتی که تحرک نودها بالاست کارایی بهتری داشته و همچنین میزان تاخیر مسیریابی درANT_TORA کمتر از TORA می‌باشد و سربار مسیریابی ANT _TORA کمتر از TORA می‌باشد.

مراجع

[1] E. Bonabeau, M. Dorigo, and G. Theraulaz, *Swarm intelligence: from natural to artificial intelligence*, Oxford University Press, 1999.

[2] J. Broch, D. A. Maltz, D. B. Johnson, Y.-C. Hu, and J. Jetcheva, "A performance comparison of multihop wireless ad hoc network routing protocols," *Proc. of the 4th Annual ACM/IEEE International Conference on Mobile Computing and Networking*, pp. 85-97, 1998.

[3] M. Dorigo, and G. D. Caro, "The ant colony optimization meta-heuristic," *New Ideas in Optimization*, pp. 11-32. McGraw-Hill, London, 1999.

[4] K. Fall, and K. Varadhan, *The ns Manual*, Nov 2000.

[5] V. Park, and M. S. Corson, "Temporally-Ordered Routing Algorithm (TORA) Version 1 Functional Specification," http://tools.ietf.org/html/draft-ietf-manet-tora-spec-04.

[6] J. P. Macker, and M. S. Corson, "Mobile ad hoc networking and the IETF," *Mobile Computing and Communications Review*, vol. 2, no. 1, pp. 9-14, 1998.

رعنا غزالی در حال حاضر مدرس دانشگاه جامع علمی کاربردی می‌باشد که از سال ۱۳۸۴ مشغول به تدریس می‌باشد. مدرک کارشناسی در رشته مهندسی کامپیوتر نرم‌افزار را در سال ۱۳۷۹ از دانشگاه آزاد اسلامی واحد تهران جنوب اخذ نموده و مدرک کارشناسی‌ارشد در رشته مهندسی کامپیوتر نرم‌افزار را در سال ۱۳۸۴ از دانشگاه آزاد واحد اراک اخذ نموده است، همچنین سه مقاله در کنفرانس بین‌المللی انجمن کامپیوتر ایران به چاپ رسانده است و دو کتاب تحت عنوان سیستم عامل‌ها و ریاضی در علم کامپیوتر را تالیف نموده است. زمینه‌های مورد علاقه ایشان شبکه‌های بی‌سیم سیار و هوش مصنوعی می‌باشد.

آدرس پست‌الکترونیکی ایشان عبارت است از:

ra_arpanet2000@yahoo.com

مجتبی صفاری تا زمان چاپ مقاله بیوگرافی نویسنده دریافت نشده است.

آدرس پست‌الکترونیکی ایشان عبارت است از:

m_saffari@yahoo.com

علی موقر رحیم‌آبادی در حال حاضر استاد دانشکده مهندسی کامپیوتر در دانشگاه صنعتی شریف است که در سال ۱۳۷۳ در این دانشگاه به عنوان استادیار شروع به کار کرد. او مدرک کارشناسی را در رشته مهندسی برق از دانشکده فنی دانشگاه تهران در سال ۱۳۵۶ و مدارک کارشناسی‌ارشد و دکتری را در رشته مهندسی کامپیوتر، اطلاعات و کنترل از دانشگاه میشیگان به ترتیب در سال‌های ۱۳۵۸ و ۱۳۶۴ دریافت نمود. او در

موسسه تحقیقاتی INRIA فرانسه به عنوان محقق در سال ۱۳۶۳، در آزمایشگاه‌های بل به عنوان محقق فنی در سال‌های ۱۳۶۴ و ۱۳۶۵ و در دانشگاه میشیگان به عنوان مدرس در سال‌های ۱۳۶۶ الی ۱۳۶۸ مشغول به کار بود. زمینه‌های مورد علاقه حرفه‌ای ایشان مدلسازی کارایی و اتکاپذیری، درستی‌یابی و اعتبار سنجی، شبکه‌های بی‌سیم سیار، و سیستم‌های بی‌درنگ توزیع شده می‌باشد. ایشان عضو ارشد سازمان‌های IEEE و ACM نیز هستند.
آدرس پست‌الکترونیکی ایشان عبارت است از:
movaghar@sharif.edu

اطلاعات بررسی مقاله:

تاریخ ارسال: ۸۸/۶/۴
تاریخ اصلاح: ۸۹/۶/۱۶
تاریخ قبول شدن: ۸۹/۷/۱۸
نویسنده مرتبط: دکتر علی موقر رحیم‌آبادی، دانشکده مهندسی کامپیوتر، دانشگاه صنعتی شریف، تهران، ایران.

[1] Multipath
[2] Ant Table
[3] Decline Rate
[4] Max Time