

**زمان بندی لینک ها با زمان در شبکه های مش بی سیم چند انتقالی/ دریافتی**

**2. مقدمات**

ما یک MTR WMN را همانند یک گراف جهت‌دار G(V.E) مدل‌سازی می‌کنیم. مجموعه V شامل گره‌هایی است که با $b>1 $ جهت دار شده است. هر گره u دارای یک محدوده انتقال از r و $b\_{u}\geq 1$ است. یال‌های جهت‌داری وجود دارند که گره u و گره v را اگر در محدوده‌ی انتقال یکدیگر باشند به‌هم وصل کرده‌اند. لینکی که u و v را به هم متصل کرده با (u.v) نشان می‌دهیم. توجه داشته باشید که، در عمل، گره‌ها ممکن است طیف انتقال مختلفی داشته باشند. برای این منظور، گره مورد نیاز باید از لینک‌های ورودی و خروجی به هر یک از همسایه ‌ها اطمینان داشته باشد. این را می‌‎توان از طریق پیغام‌های سلام در فرایند کشف همسایه به دست آورد که به موجب آن گره شامل همسایگانی است که پیغام را دریافت کرده‌اند. از این رو، تابع Ft :E $\rightarrow $R زمان اختصاص داده شده برای هر لینک است. بنابراین ft زمان انتقال مورد نیاز را به‌منظور روبه رویی با بار ترافیک داده مدل‌سازی می‌کند. همه گره‌ها قادر به ارسال یا دریافت امواج به صورت همزمان در همه لینک‌ها هستند. هر لینک توسط یک رادیو پشتیبانی می‌شود و ما فرض می‌کنیم $b\_{u}\geq \left|N(u)\right|$ برای هر گره است.

دو تحقق اصلی در مورد MTR WMNS وجود دارد. این موضوع برای اولین بار در [9]، که در آن هر روتر با رادیو‌های متعدد متصل به یک آنتن مجهز است بیان شده است. تمام رادیوها در فرکانس یکسانی کار می‌کنند. گره‌ها حس حامل خود را برای اجازه انتقال همزمان غیرفعال کرده‌اند و انتقال کنترل قدرتکه برای اطمینان از لینک‌های ورودی استفاده می‌شود قدرت سیگنال کافی برای اطمینان از دریافت صحیح دارد. تحقق دوم در راستای تحقق چندکاربری، چندورودی و چندخروجی است (MU-MIMO)؛ برای جزئیات بیشتر به [15] و یا [16] مراجعه کنید. گره‌ها آنتن‌های متعدد دارند که می‌توانند برای انتقال مستقل داده‌ها استفاده شوند. علاوه بر این، گره دارای اطلاعات حالت کانال است (CSI). این فرض معقول است که گره در درجه اول ایستاتیک باشد و نمادها را می‌توان برای یادگیری CSI انتقال داد.

بعبارت دیگر، محدودیت Mix-TX-Rx به شرح زیر است: برای یک گره داده شده u، فرض کنید IN(u.t)و OUT(u.t) مجموعه دریافت و انتقال لینک‌ها در زمان t باشند. این محدودیت برآورده می‌شود اگر هر دو IN(u.t)و OUT(u.t) به طور همزمان در هر زمان t برای همه گره بزرگتر از صفر نباشند.

در مدل ما، مفروضات به شرح زیر است:

• همه گره‌ها در سطح جهانی همزمان شده‌اند. با توجه به اینکه گره‌ها استاتیک هستند و می‌توانند از GPS برای همگام‌سازی زمان قفل استفاده کنند، این مسئله منطقی است.

• گره‌ها به تک فرکانس تنظیم شده‌اند. مطابق [9]، از دلایل اصلی برای به کارگیری یک کانال منفرد به شرح زیر است: (الف) استفاده از یک کانال برای پایه آن در حالی که از کانال‌های دیگر برای دسترسی محلی استفاده می‌کنیم راحت است، (ب) کانال‌های بیشتری استفاده می‌شود، هزینه‌های عملیاتی بالاتر به دلیل باندهای IEEE 802.11 است که برای استفاده در فضای باز در برخی از کشورهای در حال توسعه به مجوز نیاز دارد و (ج) برای جلوگیری از آلودگی RF همانگونه که بسیاری از شبکه ‌ای وای فای وجود دارد. همچنین توجه داشته باشید که قابلیت MTR گره‌ها می‌تواند با استفاده از MIMO [17] و یا توسط تجهیز گره با رادیو 60 گیگاهرتز به دست آید[18].

• همه گره‌ها در سطح جهانی همزمان شده‌اند. با توجه به اینکه گره‌ها استاتیک هستند و می‌توانند از GPS برای همگام‌سازی زمان قفل استفاده کنند، این مسئله منطقی است.

• گره‌ها به تک فرکانس تنظیم شده‌اند. مطابق [9]، از دلایل اصلی برای به کارگیری یک کانال منفرد به شرح زیر است: (الف) استفاده از یک کانال برای پایه آن در حالی که از کانال‌های دیگر برای دسترسی محلی استفاده می‌کنیم راحت است، (ب) کانال‌های بیشتری استفاده می‌شود، هزینه‌های عملیاتی بالاتر به دلیل باندهای IEEE 802.11 است که برای استفاده در فضای باز در برخی از کشورهای در حال توسعه به مجوز نیاز دارد و (ج) برای جلوگیری از آلودگی RF همانگونه که بسیاری از شبکه ‌ای وای فای وجود دارد. همچنین توجه داشته باشید که قابلیت MTR گره‌ها می‌تواند با استفاده از MIMO [17] و یا توسط تجهیز گره با رادیو 60 گیگاهرتز به دست آید[18].

• ما فرض می‌کنیم که بار ترافیک در هر لینک جمع شده است، به‌عنوان مثال، [14] را ببینید، و برای مقادیر غیر قابل اغماض ثابت باقی می‌ماند به‌عنوان مثال، هر ساعت. همچنین بدان معنی است که مسیریابی برای جفت مقصد منبع برای این دوره زمانی ثابت است. مورد مشترک در بهینه‌سازی مسیریابی و زمان‌بندی لینک را به‌عنوان کارهای آتی به آینده موکول می‌کنیم.

• هر واحد زمانی‌می‌تواند به یک مقدار خاص از زمان مطابقت داشته باشد، به‌عنوان مثال، یک ثانیه، و یا مدت زمان لازم برای انتقال یک بسته. این مهم است که توجه داشته باشید که یک بار که زمان به یک لینک اختصاص داده شد، انتقال زمان‌بندی شده آن، برای زمان پیش از انتقال تضمین شده است.

مشکل ما به شرح زیر است: با توجه به MTR WMN، به موجب آن که هر لینک دارای یک زمان مختلف است، طراحی الگوریتمی متمرکز که موجب کاهش طول superframe شود نیاز به فعال کردن حداقل یک بار و مطابق زمان لینک‌ها دارد. توجه داشته باشید، یک superframe را به‌صورت زیر تعریف می‌کنیم دنباله $\left(\left\{e\_{1}. 000. e\_{x}\right\}.t\_{1}\right)∪… ∪(\left\{e\_{y}. 000 . e\_{|E|}\right\}.t\_{|E|})$، که در آن $e\_{i}ϵE$ لینکی به زمان‌بندی در زمان ti است. به عنوان مثال، در شکل 2، داریم $(\left\{AB.AC\right\}.0)∪(\left\{BA.CA\right\}.10)∪(\left\{BC\right\}.15)∪(\left\{CB\right\}.24)$. مسئله یافتن superframe با حداقل مقدار $t\_{|E|}+f\_{t}(e\_{|E|})$ بنا به محدودیت‌های Mix-Tx-Rx است (شکل 1) و هر لینک $eϵE$ حداقل یک زمان دریافت می‌کند.

توجه داشته باشید، اگر تمام لینک‌ها زمان یکسانی داشته باشند، مسئله مشابه کار انجام شده در [11] خواهد شد. به طور خاص، نویسندگان [11] نشان می‌دهند که استخراج حداکثر تعداد لینک در هر اسلات متناظر با حل مسئله NP-complete و مسئله برش ماکزیمم است. برای این منظور، در بخش بعدی، ما روش تکاملی حریصانه را با هدف تعیین کوتاه ترین superframe امکان‌پذیر پیشنهاد می‌کنیم.

**3. راه‌حل: A-TxRx**

 ایده اصلی زمان‌بندی لینک‌های جدید با عدم تداخل به هنگام اتمام انتقال یک لینک است. در بخش. 3.1، ما نشان می‌دهیم که چگونه زمان‌بندی نتیجه شده می‌تواند با اضافه کردن لینک‌های به اصطلاح فرصت‌طلب بهبود یابد. علاوه براین، در بخش 3.2، ما یک گام با محاسبات سنگین را که برای محاسبه سیستم‌های مدیریت اطلاعات با یک گام حریصانه استفاده شده است با افزودن لینک‌های غیرمتضاد با توجه به زمان انتقال آنها ساده‌سازی می‌کنیم.

الگوریتم ما دارای مراحل کلیدی زیر است. در مرحله اول، یک گراف متضاد $G^{'}(V^{'}.E^{'})$ براساس توپولوژی شبکه $G^{ }\left(V^{ }.E^{ }\right)$ ساخته می‌شود. در گراف متضاد، هر راس $v^{'}ϵV^{'}$ نشان‌دهنده یک لینک در E است و یک تضاد بین دو لینک در E توسط یک $e^{'}ϵE^{'}$ بین رئوس مربوطه بیان می‌شود؛ [19] را مشاهده کنید. که تمام مجموعه‌های مستقل و حداکثر را (MISs) در $G^{'}$ نشان می‌دهد. به یاد داشته باشید که MIS زیرمجموعه‌ای از تمامی لینک‌ها در $G^{'}$ است که می‌تواند در زمان یکسانی بدون دخالت فعال شود. توجه داشته باشید که یک لینک ممکن است به‌صورت متفاوت ظاهر شود. در مرحله سوم، MIS را با بیشترین لینک‌ها انتخاب کردیم، که توان بالا را تضمین می‌کند. ما همه لینک‌ها را در MIS انتخاب شده فعال می‌کنیم و آنها را با عنوان لینک‌های فعال برچسب می‌زنیم. ما همچنین زمان آن‌ها را ضبط می‌کنیم. در میان تمام لینک‌های فعال، آنهایی که حداقل زمان را دارند با عنوان لینک‌های پایان در نظر گرفته می‌شوند. سپس زمان این لینک‌های پایان را ضبط می‌کنیم. در چنین زمانی، آنها را از $G^{'}$ حذف می‌کنیم. اگر $G^{'}$ خالی نباشد و اگر هیچ لینک فعالی وجود نداشته باشد اولین بار مشخص می‌شوند. این لینک‌ها به عنوان لینک‌های باقی‌مانده مشخص می‌شوند. اگر هیچ لینک باقی‌مانده‌ای وجود نداشته باشد، یک MIS جدید به‌طور مستقیم از $G^{'}$ به دست می‌آید. این MIS شامل لینک‌های جدیدی است که می‌تواند به superframe اضافه شود. با این حال، اگر لینک‌های باقی‌مانده‌ای وجود داشته باشد، ما چک می‌کنیم تا ببینیم که آیا می‌توانیم لینک‌های با عدم تداخل جدید اضافه کنیم. ابتدا یک گراف متضاد جدید $G^{''}$ از $G^{'}$ می‌سازیم. به‌طورخاص، لینک‌های پایانی، تمام لینک‌های باقی‌مانده و همسایگان آن را حذف می‌کنیم. سپس یک MIS جدید از $G^{''}$ به دست می‌آید، که شامل جدیدترین لینک‌هایی است که با لینک‌های باقی‌مانده مداخله‌ای ندارد. مجموعه‌ی بعدی از لینک‌های پایانی تعیین می‌شود و فرآیند را تکرار می‌کنیم. وقتی $G^{'}$ خالی است A-TxRx خاتمه می‌یابد (جدول 1).

ما نشان خواهیم داد که یک-TxRx چگونه کار می‌کند، الگوریتم 1 را ببینید، تعیین زمان‌بندی برای توپولوژی در شکل 2 نشان داده شده است. همان‌گونه که نشان داده شده است، آن یک WMN MTR با سه گره A، B او C متصل با لینک دو طرفه است. مقدار بعدی به هر یک از لینک‌ها نشان دهنده‌ی زمان اختصاص داده شده است.





* A-TxRx گراف شبکه $G^{ }\left(V^{ }.E^{ }\right)$ را به‌عنوان ورودی دریافت می‌کند و زمان $f\_{t}(e)$ را به هر یال $eϵE$ اختصاص داده می‌دهد.
* A-TxRx، SF را به‌عنوان خروجی تولید می‌کند. مجموعه SF شامل جفت $(A.t)$ است، که در آن مجموعه A نشان‌دهنده یک گروه از لینک‌های فعال است که انتقال آنها در زمان t شروع می‌شود.
* خطوط 1-6: در ابتدا، t با عدد صفر مشخص می‌شود. مجموعه SF خالی است. مجموعه‌های F و R نشان‌دهنده گروهی از لینک‌های پایان و لینک‌های باقی‌مانده هستند. مجموعه‌های A؛ F و R در ابتدا خالی هستند.
* خط 7: تابع ConflictG() گراف عدم تداخل متناظر را از محاسبه می‌کند. به شکل 3 نگاه کنید.
* خطوط 9-12: هنگامی که R یک مجموعه تهی است، تابع MaxMIS() رنگ‌آمیزی گراف $G^{'}$ را برای پیدا کردن MISs انجام می‌دهد: {eAC؛ eAB}، {eBC؛ eBA} و {eCA؛ eCB}. سپس ما MIS حاوی بیشترین لینک را انتخاب می‌کنیم. در اینجا، بدین دلیل که هر سه MISs اندازه یکسان دارند، به طور تصادفی {eAC؛ eAB} را انتخاب می‌کنیم. بنابراین، eAB و eAC به‌عنوان لینک‌های فعال انتخاب می‌شوند و در مجموعه A جای می‌گیرند، به این معنی که تمام لینک‌ها در این MIS در t=0 شروع به انتقال می‌کنند. سپس A و t مربوطه در SF=({ eAB , eAC },0) ثبت می‌شوند. گراف $G^{'}$ با از بین بردن لینک‌ها در A از $G^{'}$ به‌روز می‌شود. در این مورد، لینک‌های را eAB و eAC حذف می‌کنیم، به‌عنوان مثال، گره AB، AC و لینک‌های تلاقی آن در شکل 3.



* خطوط 13-19: در میان تمام زمان لینکها در A، کوتاه‌ترین را به‌عنوان tfinish در نظر می‌گیریم. لینک‌های با کوتاهترین زمان در مجموعه Fقرار دارند. دلیل آن این است که این لینک‌ها، در میان اعضای A، زودتر پایان می‌یابند. در مثال ما، tfinish=1 است، که زمان مربوط به لینک eAB است. از این رو، مجموعه F شامل eAB است. همچنین، زمان لینک‌ها در A با کم کردن tfinish از زمان $f\_{t}(e)$ به روز می‌شود. برای مثال، $f\_{t}\left(AC\right)=10-1=9$، به این معنی است که این نقطه زمان است، لینک eAC برای یک واحد زمان انتقال داده می‌شود، اما هنوز هم به نه واحد زمان دیگر برای تکمیل انتقال نیاز دارد.
* خط 20: متغیر t از t=0+1=1 نتیجه می‌شود زیرا یک واحد زمان برای لینک‌ها در A به کار برده شده است.
* خطوط 21-22: مجموعه R شامل لینک‌هایی است که در A هستند، اما نه در F. پس از آن، ما F را تنظیم می‌کنیم. در این مثال، مجموعه R شامل لینک‌های eAC است زیرا انتقال به پایان رسیده نیست. در این مرحله، به‌دلیل این که R مجموعه تهی نیست، A-TxRx از خط 24 مجددا راه‌اندازی می‌شود.
* خطوط 24-25: تابع Neighbor() تمام گره‌های تداخل را از لینک‌های R برمی‌گرداند. A-TxRx، $G^{'}$ را در گراف جدید $G^{''}$ کپی می‌کند و N را از $G^{''}$ حذف می‌کند؛ در مثال ما، این به این معنی است که eBA، eCA و eCB از$G^{''}$ حذف شده‌اند. این گام ضروری است، زیرا تضمین می‌کند که لینک‌های به‌تازگی فعال شده با لینک‌های R تداخل ندارند.
* خطوط 26-28: A-TxRx تابع MaxMIS() را در $G^{''}$ برای استخراج لینک‌های که می‌توانند به مجموعه A و بدون هر گونه درگیری اضافه شوند که در این مورد eBC است فراخوانی می‌کند. به‌عنوان نتیجه، $SF=(\{e\_{AB}. e\_{AC}\}.0)∪(\left\{e\_{BC}\right\}.1) $. گراف $G^{'}$ با حذف لینک‌های فعال $e\_{BC}$ از $G^{'}$ به‌روز می‌شوند.
* خط 29: همه لینک‌هایی که در این نقطه و در این زمان در حال انتقال هستند $ e\_{AC} $و $e\_{BC}$ هستند.
* خطوط 30-36: کوتاه‌ترین زمان لینک‌ها در A، ft(AC)=ft(BC)=9 است. متغیر tfinish با مقدار 9 تنظیم شده است. بنابراین، لینک $ e\_{AC}$ و $e\_{BC}$ تبدیل به لینک‌های پایان در F می‌شوند. همانند قبل، زمان لینک‌ها با کم کردن tfinish از زمان ft(e) به روز می‌شود. در این مورد، $f\_{t}\left(BC\right)=f\_{t}\left(AC\right)=9-9=0$، به این معنی است که در این نقطه در زمان، هر دو لینک انتقالشان را پایان می‌دهند.



* خطوط 37-39: متغیر t به t=1+9=10 تبدیل می‌شود. مجموعه R خالی می‌شود. پس از آن مجموعه F واضح می‌شود. در این مرحله، از آنجا که هیچ لینکی در R وجود ندارد، A-TxRx از خط 10تکرار می‌شود.
* خط 10: A-TxRx ، MaxMIS() را در $G^{'}$ اجرا می‌کند. MIS، {eCA؛ eBA} و {eCA؛ eCB}است. ما {eCA؛ eBA} را در مجموعه دلخواه A انتخاب می‌کنیم. فرآیند تا هنگامی که زمان t به 13 برسد ادامه می‌یابد. در این زمان، $G^{'}$ خالی می‌شود، به این معنی که همه لینک‌ها در شبکه زمان‌بندی می‌شوند. برنامه خاتمه می‌یابد و خروجی$ SF=(\{e\_{AB}. e\_{AC}\}.0)∪(\left\{e\_{BC}\right\}.1)∪(\{e\_{BA}. e\_{CA}\}.10)∪(\left\{e\_{CB}\right\}.13)$ است. پس از سه واحد زمانی دیگر، که است که زمان آخرین لینک فعال $e\_{CB}$ است، همه لینک‌ها در شبکه یک بار منتقل شده‌اند. بنابراین، برای شکل 2، A-TxRx یک superframe با طول 16 تولید می‌کند، کاهش طول superframe تولید شده با استفاده از 2P تقریبا 41٪ است. خط زمان زمان‌بندی در شکل 4 نشان داده شده است.

**3.1 لینک‌های فرصت طلب**

ما می‌توانیم ظرفیت را با اضافه کردن '' لینک‌های فرصت طلب '' بهبود بخشیم. این لینک‌ها به‌صورت عناصری تعریف می‌شوند که می‌توانند فرصت‌های انتقال اضافی بدون تداخل موجود در لینک‌ها اختصاص دهند. در مثال فوق، توجه داشته باشید که در زمان Time=[15,16]، تنها لینک eCB در حال انتقال است. هیچ لینک دیگری فعال نیست چرا که تمام لینک‌های یک بار منتقل شده‌اند. در واقع، ما همچنین می‌توانیم لینک eCA یا لینک $e\_{AB}$ را به‌عنوان لینک فرصت‌طلب فعال کنیم. از این رو، ما می‌توانیم $e\_{AB}$ را انتخاب کنیم چرا که آن زمان یک واحد و مهمتر از آن دارد و، اضافه کردن آن تغییر طول superframe نمی‌کند. از سوی دیگر، اگر ما eCA را در زمان Time=15 با زمانی برابر با پنج واحد زمان فعال کنیم، طول superframe را از 16 تا 20 گسترش خواهد داد.

برای اضافه کردن لینک‌های فرصت طلب، بهبود زیر را بر روی A-TxRx اعمال می‌کنیم. پس از ساخت یک MIS جدید متشکل از لینک‌های جدید که به superframe در خط 10 یا خط 26 از الگوریتم 1 اضافه می‌شوند، همه لینک‌های موجود در MIS جدید و همسایگان آن در نمودار اصلی برای به‌دست آوردن نموداری که قبلا شامل لینک‌های برنامه‌ریزی شده بود حذف می‌شوند. بعد، ما گراف جدید را برای به دست آوردن بزرگترین MIS رنگ‌آمیزی می‌کنیم. این MIS شامل تمام لینک‌هایی است که می‌توانند بدون ایجاد تداخل فعال شوند. سپس لینک‌هایی که طول superframe را گسترش نمی‌دهند انتخاب می‌شوند. از این رو، زمان‌بندی ایجاد شده تنها زمان‌بندی امکان‌پذیر است، اما دارای ظرفیت شبکه بالاتری است.

یک سوال که در اینجا مطرح می‌شود این است که چرا فقط لینک‌هایی که قبلا فعال شده‌اند می‌توانند لینک‌های فرصت طلب باشند. در خط 12 و28 از الگوریتم 1، لینکی که خیلی زود در MIS انتخاب شده قرار گیرد و برای انتقال انتخاب شود، از G’ حذف می‌شود. بنابراین، هنگامی که MaxMIS() را در گراف G’ و در خط 10 و 26 فراخوانی می‌کنیم، A نتیجه شده شامل لینک‌هایی است که هنوز فعال نشده‌اند. هیچ کدام از لینک‌های موجود در G’ نمی‌تواند برنامه‌ریزی شود چرا که حداقل با یک لینک در A تداخل دارد همان‌گونه که A یک MIS است. بنابراین، تنها با جستجوی لینکهایی که در G’ نیستند، به‌عنوان مثال، لینک‌هایی که حداقل یک بار فعال شده‌اند، می‌توانیم بعضی از لینک‌های دیگر با عدم تداخل را بیابیم و از آنها به‌عنوان لینک‌های فرصت طلب استفاده کنیم.

**3.2 A-TxRx حریص**

در A-TxRx، تابع MaxMIS() در خط 10 و 26 ابتدا رنگ‌آمیزی گراف را در گراف تداخل G’ به منظور پیدا کردن MIS انجام می‌دهد. سپس MIS با حداکثر کاردینالیتی به عنوان مجموعه‌ای از لینک‌های فعال A انتخاب می‌شود. با این حال، رنگ‌آمیزی گراف یک عملیات وقت‌گیر است. بخش 3.3 را ببینید. بنابراین، یک A-TxRx اصلاح شده را که به‌عنوان A-TxRxGreedy نشان داده شده است معرفی می‌کنیم، که در آن تابع MaxMIS() در خط 10 و 26 را با Greedy() جایگزین کرده‌ایم. تابع Greedy()، A را با تکرار از طریق تمام لینک‌ها در E که از یک و با طولانی‌ترین زمان انتقال شروع شده‌اند می‌سازد. لینک e به مجموعه A اضافه می‌شود اگر e با هر لینک حاضر در A تداخل نداشته باشد. درغیر این صورت، لینک e برای پیوستن به A انتخاب نمی‌شود.

**3.3 تجزیه و تحلیل**

هم اکنون ما پیچیدگی زمانی A-TxRx را برای یک گراف دلخواه G با |v| گره محاسبه می‌کنیم به‌طوری که تعدا یال‌های |E| در G کران بالایی مانند |v|(|v|-1) (گراف کامل) دارد. به‌طور خاص، نتیجه زیر را داریم.

**قضیه 1:** پیچیدگی زمانی A-TxRx، O(|v|5) است.

اثبات: تمام خطوط A-TxRx زمانی برابر با O(1) دارد به جزء خط 7، 8، 10، 13، 14، 24، 26، 40 و 31. در خط 7، تابع ConflictG() زمانی برابر با O(|v|2) برای تبدیل گراف اصلی G به گراف G’ نیاز دارد. دلیل آن این است که هر یال در G به‌عنوان یک راس در G’ در نظر گرفته شده است و متصل به لینک‌های مربوط به آن است. در نتیجه، تعدادراس‌ها در G’ با تعداد یال‌ها در G برابر است، که محدود به |v|(|v|-1) است. مشابه خط 8، پیچیدگی زمانی O(|v|2) است زیرا تعداد رئوس در G’ را بیان می‌کند. در خط 10 و 26، فرآیند رنگ‌آمیزی گراف انجام شده است. با توجه به [20]، الگوریتم رنگ‌آمیزی کوچکترین-آخرین گراف دارای یک پیچیدگی زمانی O(|v’|+|E’|) برای G’(|V’|,|E’|) است. برای محاسبه تعداد کل یال‌ها در G’، ما بدترین حالت را فرض می‌کنیم، به‌طوریکه G به طور کامل متصل است. برای هر لینک، eAB، $2\left|v\right|-3$ لینک از گره B به گره A که در تضاد با eAB هستند. از این رو، تعدادی از یال‌ها در G’ به صورت $\left(2\left|v\right|-3\right)×|v|(\left|v\right|-1)$ محاسبه می‌شود. در نتیجه، پیچیدگی زمانی برای خط 10 و 26، $O\left(\left|v\right| ^{2}+\left|v\right|^{3}\right)=O(|v|^{3})$ است. برای خطوط 13، 30، 14 و 31، در بدترین حالت، هر راس $v^{'}ϵV^{'}$ در G’ تنها به یک راس که نشان دهنده لینک مربوط در جهت مخالف است متصل شده و در نتیجه اندازه بزرگترین MIS، که به‌عنوان A استفاده می‌شود، با $\frac{1}{2}|v|(\left|v\right|-1)$ برابر است. به‌طور مشابه، در بدترین حالت، اندازه R برابر است با $\frac{1}{2}|v|(\left|v\right|-1)$. بنابراین، خطوط 12، 30، 14، 31 و 24 در بیشتر مواقع زمانی برابر با $O\left(\left|v\right| ^{2}\right)$ دارند. بر اساس محاسبات فوق، A-TxRx دارای پیچیدگی زمانی $O\left(\left|v\right| ^{2}\left|v\right|^{3}\right)=O(|v|^{5})$ است.

**قضیه 2 :** A-TxRx یک زمان‌بندی برخورد رایگان تولید می‌کند.

اثبات: از ثابتهای lتناقض استفاده می‌کنیم. فرض کنید دو لینک ارسال و دریافت در زمان یکسان داریم. ب‌ه‌طور خاص، eAB و exA یا eAB و eBy در حال انتقال به طور همزمان هستند، که در آن x می‌تواند هر گرهی باشد به جز A، y می‌تواند هر گرهی باشد به جز B. بنابراین دو مورد داریم:

مورد 1: این دو لینک تداخل شروع به انتقال از نقطه یکسانی در زمان‌ می‌کنند. در این مورد، دو لینک فعال در خط 10 هستند. پس از فرآیند رنگ‌آمیزی گراف، تنها لینک‌ها در یک مجموعه مستقل فعال می‌شوند. به یاد بیاورید که در یک مجموعه مستقل از یک گراف، هیچ دو راس از گراف مجاور نیستند. این با مورد دوم در تضاد است با این که این دو لینک با همدیگر تداخل دارد.

مورد 2: یکی از این دو لینک تداخل، با exA (یا eBy) شروع می‌شود در حالی که برای انتقال از لینک دیگر، انتقال آن به پایان نرسیده است. این نشان می‌دهد که لینک به تازگی اضافه شده در خط 26 فعال شده است، که رنگ‌آمیزی گراف را برای گراف جدید G’ به کار برده است. توجه داشته باشید که در خط 25، تمام لینک‌های باقی مانده از جمله eAB و لینک‌های همسایه آن از G’  حذف می‌شوند. بنابراین، هر لینک که با لینک eAB تداخل دارد در گراف G’  وجود ندارد. به عنوان نتیجه، غیر ممکن است A-TxRx برای استخراج یک زمان‌بندی با تداخل به کار گرفته شود.

**قضیه 3:** A-TxRx یک زمان‌بندی لینک تولید می‌کندکه بیشتر از $δ\left(G\right)=l\_{t}^{A}+l\_{t}^{B}$ برای یک توپولوژی گراف دوبخشی نیست.

اثبات: فرض کنید v1 و v2 دو زیرمجموعه از یک گراف دوبخشی باشند به صورتی که گره‌ها در v1(v2) تنها لینک به گره‌های در V v2(v1) دارند. خط 7 از A-TxRx گراف تداخل G’(v’,E’) را از گراف G تولید می‌کند. توجه داشته باشید که می‌توانیم دو MIS تولید کنیم برای مثال، A و B، از G’؛ A (B) شامل لینک‌های ناشی از گره‌ها در v1(v2) است. فرض کنید i1(i2) یک لینک ناشی از v1(v2) با طولانی‌ترین زمان باشد، به‌عنوان مثال، $f\_{t}\left(i\_{1}\right)=l\_{t}^{A}$ ؛ به‌طور مشابه، $f\_{t}\left(i\_{2}\right)=l\_{t}^{B}$. توجه داشته باشید که می‌تواند تعداد متعددی لینک $i\_{1}$ و $i\_{2}$ می‌تواند وجود داشته باشد. سه مورد ممکن را در نظر بگیرید: (1) تمام لینک‌ها در (B) A زمان یکسانی دارند ، به‌عنوان مثال، یک لینک در A (B) زمان $l\_{t}^{A}(l\_{t}^{B})$ را دارد، (2) لینک$i\_{1}$ و $i\_{2}$ با یکدیگر تداخل دارند و (3) لینک $i\_{1}$ و $i\_{2}$ با یکدیگر تداخل ندارند.

برای مورد (1)، A-TxRx یک زمان‌بندی تولید می‌کند که در آن لینک‌های موجود در B بعد از همه لینک‌های B فعال می‌شوند و در نتیجه $δ\left(G\right)=l\_{t}^{A}+l\_{t}^{B}$. به‌طور خاص، خط 10-12، MIS را تولید می‌کند و همه لینک‌ها را در شروع A در زمان t =0 برای $l\_{t}^{A}$ واحد زمان برنامه‌ریزی می‌کند. خط 14-19 F=A را تولید می‌کند چون تمام لینک‌ها زمان پایان $l\_{t}^{A} $ دارند و در نتیجه خط 20-21 ، t=$ l\_{t}^{A}$ را به‌دست می‌آورد. بنابراین، در تکرار دوم از حلقه در خط 8، خط 10-12، MIS A=B را تولید می‌کند و همه لینک‌ها را با شروع از B در زمان t=$ l\_{t}^{A}$ برنامه‌ریزی می‌کند. مشابه با A، این تکرار F=B را تولید می‌کند؛ بنابراین الگوریتم با یک زمان‌بندی که از t=0 شروع می‌شود کامل می‌شود. بنابراین زمان‌بندی در زمان $l\_{t}^{A}+l\_{t}^{B}$ کامل می‌شود، برای این مورد، A-TxRx زمان‌بندی با طول فریم $δ\left(G\right)=l\_{t}^{A}+l\_{t}^{B}$ تولید می‌کند.

برای مورد دوم، برخی از لینک‌های دیگر از $i\_{2}$ را می‌توان در در زمان یکسانی با لینک‌ها در A فعال کرد. با این حال، از آنجا که $i\_{2}$ و $i\_{1}$ لینک‌های تداخل هستند، $i\_{2}$ می‌تواند تنها پس از اتمام $i\_{1}$ فعال شود؛ $i\_{2} $ زمان خود را در زمان $l\_{t}^{A}+l\_{t}^{B}$ کامل می‌کند، به این معنی که طول superframe برای این مورد $δ\left(G\right)=l\_{t}^{A}+l\_{t}^{B}$ است زیرا تمام لینک‌ها در (B)A بعد از زمان $l\_{t}^{A}(l\_{t}^{A}+l\_{t}^{B})$ کامل نمی‌شوند. به طور خاص، خط 10-12، MIS A=A و زمان‌بندی تمام لینک‌ها را در زمان صفر تولید می‌کند. از آنجا که R شامل حداقل تولید است، و $G^{'}\ne ∅$ در تکرار دوم، خطوط 24-27 لینک‌های موجود در B را که با لینک‌های باقی مانده در R تداخل ندارند لینک می‌کند. برای خط 30، دو حالت ممکن را درنظر می‌گیریم: (1.1) حداقل یک لینک اضافی از B دارای زمان طولانی‌تری از باقی‌مانده زمان $i\_{1}$ است؛ (2.2) باقی‌مانده زمان $i\_{1}$ طولانی‌ترین لینک در میان MIS جدید است. برای مورد (1.1)، A-TxRx خط 23 را تکرار می‌کند و در نهایت یک MIS که فقط شامل لینک‌ها در B است، از جمله $i\_{2}$ تولید می‌کند. در این مرحله، خط 24-26، $N=∅$ را تولید می‌کند زیرا گره‌ها در $R=G^{'}⊆v\_{2}$ و گره‌های همسایه هستند. در نتیجه خط 26 لینک بیشتری به لینک SF اضافه نمی‌کند، لینک‌هایی که در زمان $l\_{t}^{A}$ شروع می‌شوند در زمان $l\_{t}^{B}$ تکمیل می‌شوند، به این معنی که طول superframe ، $δ\left(G\right)=l\_{t}^{A}+l\_{t}^{B}$است. برای مورد(2.2)، A-TxRx تکرار خود را از خط 24 تا رسیدن به مورد (1.1) ادامه خواهد داد. در هر صورت، در زمان $l\_{t}^{A}$؛ R در نهایت تنها حاوی لینک‌های B خواهد شد و همان‌گونه که قبلا شرح داده شد، لینک‌ها در زمانی بیش از $l\_{t}^{B}$ فعال نخواهند شد.

برای مورد (3)، لینک $i\_{2}$ را می‌توان در زمان یکسانی با توجه به لینک $i\_{1}$ تا زمانی که لینک تداخل خود را به اتمام نرسانده است فعال کرد. بنابراین، برای این مورد، A-TxRx زمان‌بندی با طول superframe ،$δ\left(G\right)<l\_{t}^{A}+l\_{t}^{B}$ تولید می‌کند. به طور خاص، هنگامی که A-TxRx به خط 24 می‌رسد، N شامل $i\_{2}$ نیست و در نتیجه SF در خط 27 شامل هر دو $i\_{1}$و $i\_{2}$ است، که در نهایت طول superframe زمان‎‌بندی را به کمتر از $l\_{t}^{A}+l\_{t}^{B}$ بدون تداخل لینک‌ها کاهش می‌دهد.

بر اساس تجزیه و تحلیل بالا، هنگامی که $i\_{1}$ و $i\_{2}$ لینک‌های تداخل هستند ، $i\_{2}$ می‌تواند تنها پس از فعال شود ؛ $i\_{2} $ زمان خود را در زمان $l\_{t}^{A}+l\_{t}^{B}$ که، به این معنی طول superframe برای این مورد است کامل می‌کند چرا که همه لینک‌ها در A (B) بعد از زمان $l\_{t}^{A}+l\_{t}^{B}$ کامل نمی‌شوند. به‌طور کلی، دو بخش دلخواه گراف در مورد (3) قرار می‌گیرند و در نتیجه A-TxRx باید به تولید زمان‌بندی با طول superframe ، $δ\left(G\right)<l\_{t}^{A}+l\_{t}^{B}$ مشغول باشد.

**4. روش تحقیق**

برای ارزیابی عملکرد A-TxRx، ما از MatGraph [21] استفاده می‌کنیم، یک جعبه ابزار در نرم افزار Matlab برای کار با نمودار ساده است. در آزمایشات، تمام گره‌ها ثابت و به صورت تصادفی در یک منطقه مربعی قرار گرفته‌اند. توجه داشته باشید که خطای کانال در آزمایشات درنظر گرفته نشده است. در عمل، ارسال مجدد به دلیل اشتباهات کانال می‌تواند با ابعاد اختصاص زمان انتقال در نظر گفته شود. در غیر این صورت، لینک‌های با شرایط کانال ضعیف را می‌توان از توپولوژی حذف کرد.

ما تاثیر پارامترهای زیر در عملکرد A-TxRx را مطالعه می‌کنیم: تراکم گره، شعاع انتقال، درجه گره و MIS انتخاب شده، و همچنین زمان اجرا. تعداد گره‌ها در محدوده 5 تا 40 با فاصله 5 است. شعاع انتقال در محدوده 10 تا 130 متر با فاصله 20 متر است. منطقه شبکه در محدوده از 50m\* 50m به 250m\* 250m است. درجه هر گره از 2 تا 10، با فرض مجموعا 11 گره متفاوت است. پنج آزمایش با تغییر پیکربندی شبکه در حالی که بقیه موارد ثابت هستند انجام گرفته است. نتایج به دست آمده از به‌طور متوسط ​​20 شبیه‌سازی، هر یک با توپولوژی متفاوت است.

ما A-TxRx را، که نشان‌دهنده هر دو A-TxRxGC و A-TxRxGreedy ، در برابر 2P-1slot، 2P-گره و Jazzy-MAC [14] است مقایسه می‌کنیم که A-TxRxGC الگوریتم رنگ آمیزی گراف است و A-TxRxGreedy از جستجوی حریصانه استفاده می‌کند. به‌طورخاص، برای هر دو 2P-1slot و 2P گره، برنامه‌ریزی براساس زمان انجام می‌گیرد. اندازه حافظه به طولانی‌ترین مجموعه زمانی در میان لینک‌های فعال در حافظه نگاشت می‌شود. برای 2P-1slot، گراف تداخل را برای بدست آوردن توپولوژی MIS رنگ‌آمیزی می‌کنیم. سپس، تمام لینک‌های موجود در MIS را برای انتقال در اولین شکاف زمان فعال می‌کنیم. این لینک ها پس از آن شروع به دریافت در بازه زمانی مربوطه می‌کنند. برای مثال، در حافظه i، جایی که i عدد فردی است، اگر eAB فعال باشد، لینک eBA اسلات i+1 ام را انتقال خواهد داد. برای 2P گره، رنگ‌آمیزی گراف را بر روی گراف شبکه به جای گراف تداخل انجام می‌دهیم. پس از این‌که MIS با بیشترین گره تعیین شد، تمام گره‌ها در MIS انتقال می‌یابند و در بازه زمانی بعدی تبدیل به گیرنده می‌شوند. سپس، گره انتخاب شده را از گراف شبکه حذف می‌کنیم و روند فوق را تا زمانی که همه لینک‌ها حداقل یک بار فعال شوند تکرار می‌کنیم. JazzyMAC همچنین از TDMA استفاده می‌کند اما طول شکاف پویا است. به‌صورت متمرکز مقداردهی اولیه می‌شود و پس از آن طبق قوانین اساسی زیر شروع به کار می‌کند: هر گره دارای یک نشانه برای هر یک از لینک‌های خود است. هنگامی که یک گره انتقال بر روی یک لینک را تمام می‌کند، این نشانه به انتهای دیگر آن لینک عبور می‌کند. یک گره فرستنده می‌شود هنگامی که آن نشانه را برای همه لینک‌ها (بازدید کنندگان) نگه دارد. دیگر جزئیات JazzyMAC را می‌توان در [14] یافت.

در هر آزمایش، معیارهای زیر با خطای فاصله اطمینان 95٪ جمع‌آوری شده است:

• طول Superframe: زمان کل هر لینک برای انتقال حداقل یک بار است.

• تعداد لینک‌های فعال همزمان: مربوط به میانگین تعداد لینک‌هایی است که به صورت همزمان در هر نقطه از زمان عمل می‌کنند. همچنین شامل تعداد لینک‌های فرصت طلب در بخش3.1 است. این متریک قابل توجه است زیرا نشان‌دهنده ظرفیت WMN است.

• زمان محاسبه: این زمان، زمان مورد نیاز برای هر الگوریتم جهت محاسبه برنامه‌ریزی برای یک توپولوژی داده شده بر روی یک کامپیوتر با پردازنده اینتل i7 هسته‌ای با 6 گیگابایت RAM است.

 **5. نتایج**

در بخش‌های زیر، نتایج حاصل از آزمایشات در رابطه با تراکم گره، شعاع انتقال، درجه گره، زمان در حال اجرا و تاثیر انتخاب MIS بررسی خواهد شد.

**5.1 تراکم گره**

در آزمایش اول، تاثیر تراکم گره در طول superframe و تعداد لینک‌های همزمان را مطالعه می‌کینم. تعداد گره‌ها در محدوده 5 تا 40 است. محدوده انتقال هر گره به 70 تنظیم شده است.

از شکل 5 الف، می‌بینیم که طول superframe با اضافه کردن گره‌های افزایش می‌یابد. دلیل آن این است که همان طور که تعداد گره‌ها از 5 تا 40 افزایش می‌یابد، تعداد لینک‌ها نیز از 15 تا 1159 افزایش می‌یابد. بنابراین لینک‌های فرصت طلب زیادی در هر نقطه زمانی وجود دارد (یا در طول هر زمان). علاوه بر این، در شکل 5 (الف)، ما همچنین می‌توانیم مشاهده کنیم که طول superframe در A-TxRx به طور قابل توجهی کوتاه تر از 2P-1slot و 2P گره است، به عنوان مثال، کمتر از نصف زمانی. دلیل آن این است که، در 2P-1slot و 2P گره، تا زمانی که MIS تعیین شده است، همه لینک‌ها در MIS به‌عنوان یک گروه در نظر گرفته لینک‌ها در نظر گرفته شده‌اند.



با این حال، در میان این گروه از لینک‌ها، برخی از لینک‌ها ممکن است انتقال خودشان را زودتر به پایان برسانند، به‌عنوان مثال، زمان کمتری به هنگام اختصاص اسلات دارند. به عنوان نتیجه، بخشی از کانال بیکار خواهد بود. برای A-TxRx، تا زمانی که یک لینک انتاقل خود را تمام کند، بزرگترین مجموعه از لینک‌های غیرمتضاد جدید بلافاصله با حداکثر استفاده از انتقال کانال فعال می‌شوند.

در شکل 5 (ب)، به استثناء JazzyMAC، ما 2P- 1slot، 2P-گره و A-TxRx را با اجرای برنامه‌ریزی فرصت طلب گسترش داده‌ایم. توجه داشته باشید که لینک‌های فرصت طلب در طول superframe تاثیر نمی‌گذارند. همان‌گونه که شان داده شده است، 2P-1slot حدود 25٪ بیشتر لینک همزمان در مقایسه با A-TxRx ایجاد می‌کند، به‌ویژه برای شبکه‌های با تراکم بالا، به‌عنوان مثال، زمانی که تعداد گره به 40 می‌رسد. دلیل کلیدی این است که 2P-1slot لینک فرصت طلب بیشتری با توجه به زمان اسلات اضافه می‌کند. هر دو A-TxRx و 2P-1slot به طور قابل توجهی بهتر از 2P گره و JazzyMAC هستند. دلیل این است که 2P گره، لینک همزمان کمتری با انتخاب لینک‌های گره اساسی ایجاد می‌کند. اگر یک لینک از یک گره به عنوان لینک تداخل انتخاب شود، پس از آن هیچ یک از لینک‌های دیگر را نمی‌توان برای انتقال فعال کرد. JazzyMAC، لینک همزمان کمتری دارد بنابراین گسترش آن به JazzyMAC با برنامه‌ریزی‌های فرصت طلب غیرممکن است.

برای تعیین مزایای برنامه‌ریزی‌های فرصت طلب، ما آزمایشات را برای A-TxRx تکرار می‌کنیم. شکل. 6 تعداد لینک‌های همزمان را با و بدون برنامه‌ریزی فرصت طلب برای A-TxRx مقایسه می‌کند. می‌توانید ببینیم که زمانی که برنامه‌ریزی فرصت‌طلب پیاده‌سازی شده است، حدود 20٪ لینک بیشتر برای اضافه کردن تعداد بیشتری از گره‌ها وجود دارد در‌حالی که منطقه شبکه ثابت است.

علاوه بر این، با مقایسه A-TxRxGC و A-TxRxGreedy، درمی‌یابیم که برای مقدار کمی از گره‌ها، به عنوان مثال، گره کمتر از 15، این دو الگوریتم دقیقا عملکرد یکسانی دارند.



با این حال، به عنوان تعدادی از گره افزایش 15-40، A-TxRxGC نشان می دهد یک کمی عملکرد برتر از A-TxRxGreedy از حدود 8٪ طول superframe کوتاه تر و اطراف موافقم 10٪ بیشتر لینک اجاره. دلیل این است که A-TxRxGC تمایل به فعال بزرگترین MIS در میان تمام دست ندهید تولید شده توسط رنگ آمیزی گراف. در حالی که در A-TxRx حریص، مجموعه ای از لینک فعال است که به سادگی سیستمهای اطلاعات مدیریت تصادفی. از این رو، A از A-TxRxGC است به احتمال زیاد به حاوی لینک بیشتر از A-TxRx حریص. برای این دلیل، A-TxRxGC فعال لینک های بیشتر در هر تکرار، و در نتیجه به طول superframe کوتاه تر منجر می شود.

**5.2 شعاع انتقال**

در این آزمایش، عملکرد تمام الگوریتم‌ها را زمانی که محدوده انتقال گره‌ها تغییر می‌کند ارزیابی می‌کنیم. 15 گره واقع در یک منطقه مربع 100\*100 وجود دارد. محدوده زمانی از یک تا 10 واحد زمانی است.

شکل 7 (الف) نشان می‌دهد که هنگامی که محدوده انتقال از 10 تا 90 متر افزایش می‌یابد، طول superframe تمام چهار الگوریتم را به‌صورت خطی افزایش می‌دهد. طول superframe در A-TxRx 12 واحد بعد از افزایش محدوده انتقال با اندازه مرحله 40 متر افزایش می‌یابد.



متقابلا، 2P-1slot، 2P-گره و JazzyMAC با افزایش تقریبی 30، 20، و 16 واحد زمانی مورد ارزیابی قرار می‌گیرند. برای محدوده انتقال کوچک، به عنوان مثال، 10 متر، امکان ایجاد یک لینک بین دو گره کوچک است، و در نتیجه به طور متوسط ​​فقط 6.8 لینک در مجموع وجود دارد. از سوی دیگر، افزایش دامنه انتقال به 110 متر امکان ایجاد لینک‌ها را افزایش می‌دهد. برای محدوده، به طور متوسط ​​208.7 لینک در شبکه وجود دارد و در نتیجه شبکه تقریبا به طور کامل متصل می‌شود، به موجب آن هر گره با تمام دیگر گره‌ها متصل است. بنابراین، محدوده انتقال 110-150 متر تقریبا گراف تداخل یکسانی ایجاد می‌کند. به عنوان نتیجه، طول superframe در A-TxRx، 2P-1slot، 2P-گره و Jaz-zyMAC تقریبا در مقادیر 66، 243، 145 و 99 ثابت می‌شود. ما می‌بینیم که تفاوت بین طول superframe در A-TxRx نسبت به سه الگوریتم دیگر بیش از دو برابر در محدوده انتقال از 50 و 110 متر است. این نشان می‌دهد که A-TxRx نسبت به سه الگوریتم دیگر هنگامی که محدوده انتقال بزرگ باشد به صرفه تر است.

شکل 7 (ب) میانگین تعداد لینک‌هایی که در هر نقطه از زمان تحت انتقال‌های مختلف فعال هستند نشان می‌دهد. ما می‌بینیم که تعداد لینک‌های همزمان A-TxRx افزایش می‌یابد و به مقدار ماکزیمم خود 24 می‌رسد هنگامی که محدوده انتقال 90 متر باشد. با افزایش شعاع انتقال، گراف به‌طور کامل متصل می‌شود، در نتیجه تعداد لینک‌های همزمان در 23، بدون در نظر گرفتن اتفاقی در توزیع گره ها ثابت می‌شود. درحالی که، برای 2P-1slot، پس از آن که به مقدار بیشینه خود 21 در 90 متر رسید، تعداد لینک‌های ویژه تا 15 کاهش می‌یابد، و پس از آن در اطراف 12 ثابت می‌شود. دلیل این مورد گراف کامل همبند است، به استثنای لینک‌های فرصت طلب، هنگامی که 2P- 1slot ابتدا رنگ‌آمیزی گراف را انجام می‌دهد، تنها 14 لینک می‌تواند فعال شود. به‌طور خاص، این 14 لینک از 14 گره مختلف نشات گرفته است. پس از برداشتن این 14 لینک، گره A از گراف شبکه قطع شده است. به‌عنوان نتیجه، در طول فرآیند رنگ‌آمیزی گراف دوم، تنها 13 لینک فعال وجود خواهد داشت. از این رو، تعداد لینک‌های انتقال از 14 لینک به یک لینک کاهش می‌یابد. به این ترتیب، میانگین تعداد لینک‌های همزمان کمتر از $\frac{\sum\_{i-1}^{i=14}i}{14}=7.5$ است. اگر چه لینک‌های فرصت طلب می‌توانند برای افزایش تعداد کل لینک‌ها در هر اسلات اضافه شوند، تنها لینک با زمان کمتر از طول اسلات می‌تواند انتخاب و فعال شود.

**5.3 درجه گره**

در این آزمایش، تعداد گره‌ها 11 است. درجه همه گره‌ها یکسان است و از دو تا 10 تغییر می‌یابد. ما رابطه بین درجه گره و طول superframe، و همچنین تاثیر طول superframe و تعداد کل لینک‌ها در تاخیر را مطالعه می‌کنیم.

شکل 8 (الف) نشان می‌دهد که طول superframe در A-TxRx از 18 تا 56 بنا به افزایش درجه گره به صورت خطی افزایش می‌یابد. دلیل این است که تعداد کل لینک‌های d\*n یک تابع خطی است، که در آن d نشان‌دهنده درجه گره‌هایی است که از دو تا 10 افزایش می‌یابد و n تعداد گره‌هایی است که به 11تنظیم می‌شود. جالب توجه است که، زمانی‌که تعداد درجه هر گره از 2-8 تغییر می‌کند طول superframe در JazzyMAC و A-TxRx بسیار نزدیک به هم است. سپس هنگامی‌که درجه گره از هشت تا 10 افزایش می‌یابد، طول superframe در 2P-1slot، 2P-گره و JazzyMAC به ترتیب 130، 80 و 46٪ افزایش می‌یابد. این را می‌توان با استفاده از شکل 8 (ب) توضیح داد. ما می‌بینیم که به طور قابل توجهی تعداد کمتری از لینک‌های همزمان برای 2P-1slot، 2P-گره و JazzyMAC وجود دارد که درجه گره از هشت تا 10 افزایش می‌یابد.

**5.4 زمان محاسبه**

درحال حاضر زمان محاسبه مورد نیاز برای هر الگوریتم برای محاسبه زمان‌بندی برای توپولوژی داده شده اندازه‌گیری و مقایسه می‌شود. شکل. 9 (A، B) پیکربندی شبکه یکسانی همانگونه که در بخش 5.1 و 5.3 استفاده شده است دارد.





شکل 9 (الف) زمان محاسبه هر یک از الگوریتم‌ها را تحت تعدادی گره نشان می‌دهد. توجه داشته باشید، زمان مورد نیاز برای 2P-گره و JazzyMAC کمتر از 0.1 است، که بسیار کمتر از A-TxRx است. به طور خاص، زمانی که 40 گره داریم، زمان محاسبه 2P گره و JazzyMAC تنها 0.02 درصد از A-TxRx است. دلیل اصلی آن این است که 2P-گره و JazzyMAC یک الگوریتم رنگ‌آمیزی گراف در توپولوژی شبکه اجرا می‌کنند، که شامل تا | E | گره و $\frac{|E|(\left|E\right|+1)}{2}$ یال است. علاوه بر این، 2P-1slot دارای زمان اجرای سریع‌تری از A-TxRx است، به‌عنوان مثال، سه بار سریع تر است. دلیل این است که 2P-1slot فقط بر روی هر اسلات اجرا می‌شود. به طور خاص، پس از محاسبه فرستنده‌ها از اسلات i، این فرستنده‌ها در اسلات i + 1 گیرنده می‌شوند.

شکل 9 (ب) زمان اجرا الگوریتم را با افزایش درجه گره نشان می‌دهد. ما می‌بینیم که زمان اجرا برای تمام الگوریتم‌ها با افزایش درجه گره افزایش می‌یابد. به‌طور خاص، بیشتر از22 لینک‌ با افزایش درجه گره به دو ایجاد می‌شود. جالب توجه است که، زمان محاسبه برای A-TxRx از 1 تا 3 افزایش می‌یابد. این به خاطر تعدادی یال در گراف تداخل است، که 30 بار از 31-995 افزایش می‌یابد. بنابراین، فرآیند رنگ‌آمیزی گراف نیاز به زمان بیشتری دارد، که متحمل زمان محاسبه طولانی‌تر است.



ما دریافتیم که A-TxRxGreedy  به‌طور متوسط ​​10 درصد سریعتر از A-TxRxGC است. دلیل آن این است که پیچیدگی زمانی تابع Greedy() در خط 10 و 26 از A-TxRxGreedy، نسبت به MaxMIS()، $O(|v|^{2})$ است که دارای زمان اجرا $O(|v|^{3})$ است.

**5.5 تاثیر انتخاب MIS مختلف برای A-TxRxGC**

یک سوال جالب که مطرح می‌شود این است که آیا انتخا MIS های مختلف از A-TxRxGC تاثیری بر طول superframe و تعداد لینک‌های همزمان دارد. برای پاسخ به این سوال، طول superframe و تعداد لینک‌های همزمان را برای موارد زیر مقایسه می‌کنیم: (1) بزرگترین MIS، که بزرگترین کاردینالیتی را دارد؛ (2) طولانی‌ترین MIS، که شامل لینک با طولانی‌ترین زمان است؛ (3) کوتاه‌ترین MIS؛ و (4) MIS تصادفی.

شکل. 10 طول Superframeو تعداد لینک‌های همزمان را وقتی که درجه گره‌های مختلف از 2 به 10 برای یک شبکه با 11 گره افزایش می‌یابد نشان می‌دهد. مشاهده می‌کنیم که انتخاب MIS هیچ تاثیر قابل توجهی در عملکرد A-TxRxGC ندارد. به این دلیل که A-TxRxGC بارها و بارها رنگ‌آمیزی گراف را در گراف شبکه‌ی به‌روز انجام می‌دهد. در این روش، لینک‌ها تقریبا به طور مساوی به MIS های متفاوت اختصاص داده شده‌اند. بنابراین، اختلاف بین اندازه و یا طول زمان MIS بسیار ناچیز است. بنابراین، A-TxRxGC غیرحساس به انتخاب MIS است.

**6. نتیجه‌گیری**

در این مقاله برنامه‌ریزی لینک در MTR WMNS مورد مطالعه قرار گرفته است به موجب آن گره توانایی ایجاد لینک‌های متعدد با هم را دارد. ما یک الگوریتم زمان‌بندی زمان به‌نام A-TxRx پیشنهاد می‌کنیم که تعداد انتقال همزمان در هر نقطه ای از زمان را برای افزایش ظرفیت شبکه و همچنین به حداقل رساندن طول superframe به حداکثر می‌رساند. A-TxRx اولین الگوریتم متمرکز است که لینک‌ها را با وزن‌های مختلف در یک توپولوژی شبکه زمان‌بندی می‌کند. برای WMNS MTR که در آن زمان برای هر لینک به‌عنوان وزن لینک داده شده است، A-TxRx لینک‌ها را هر زمان که لینک انتقال را به پایان برساند فعال می‌کند، به موجب آن لینک‌هایی را که در تداخل با انتقال‌ها نیست اضافه می‌کند. نتایج نشان می‌دهد که این قانون A-TxRx را قادر می‌سازد تا به طول superframe کمتر و ظرفیت شبکه بالاتری دست یابد. به‌طور خاص، نتایج ما نشان می‌دهد A-TxRx عملکرد برتری را با طول superframe 70٪ کوتاه‌تر و 60 درصد لینک‌های همزمان بیشتر در مقایسه با 2P و JazzyMAC دارد. به‌عنوان کارهای آینده، قصد داریم به بررسی سناریوها هنگامی که آنتن‌های جهت کافی برای یک گره و برقراری ارتباط با تمام همسایگان وجود دارد بپردازیم. یکی دیگر از موارد امکان‌پذیر، توسعه یک الگوریتم توزیع شده منطبق بر عملکرد A-TxRx است. در نهایت، جالب خواهد بود تا کنترل پذیرش را نیز در نظر بگیریم؛ به عنوان مثال به مرجع، [22] مراجعه کنید.

**References**

1. Shihab, E., Cai, L., Wan, F., Gulliver, A., & Tin, N. (2008). Wireless mesh networks for in-home iptv distribution. IEEE Network, 22, 52–57.

2. Nandiraju, N., Nandiraju, D., Santhanam, L., He, B., Wang, J., & Agrawal, D. (2007). Wireless mesh networks: Current challenges and future directions of web-in-the-sky. IEEE Wireless Communications, 14, 79–89.

 3. Lee, M., Zheng, J., Ko, Y.-B., & Shrestha, D. (2006). Emerging standards for wireless mesh technology. IEEE Wireless Communications, 13, 56–63.

 4. Liu, C., Gkelias, A., Hou, Y., & Leung, K. (2009). Cross-layer design for QoS in wireless mesh networks. Springer Wireless Personal Communications, 51(3), 593–613.

 5. Chin, K. (2008). A new link scheduling algorithm for concurrent Tx/Rx wireless mesh networks. In IEEE ICC (pp. 3050–3054). Beijing, China.

6. Patra, R., Nedevschi, S., Surana, S., Sheth, A., Subramanian, L., & Brewer, E. (2007). WiLDNet: design and implementation of high performance WiFi based long distance networks. In USENIX Symposium on Networked Systems Design & Implementation (pp. 87–100). Cambridge, MA,USA.

7. Dutta, P., Jaiswal, S., & Rastogi, R. (2007). Routing and channel allocation in rural wireless mesh networks. In IEEE INFOCOM (pp. 598–606) Washington, DC, USA.

8. Wu, D., Yang, S., Bao, L., & Liu, C. (2013). Joint multi-radio multi-channel assignment, scheduling and routing in wireless mesh networks. Springer Wireless Networks, 20(1), 4–16.

9. Raman, B., & Chebrolu, K. (Aug. 2005). Design and evaluation of a new MAC protocol for long-distance 802.11 mesh networks. In ACM MOBICOM. (pp. 156–169). Cologne, Germany.

10. Chin, K., Soh, S., & Meng, C. (2010). A novel spatial TDMA scheduler for concurrent transmit/receive wireless mesh networks. In IEEE AINA, Perth, WA, Australia.

11. Chin, K., Soh, S., & Meng, C. (2012). Novel scheduling algorithms for concurrent transmit/receive wireless mesh networks. Computer Networks, 56, 1200–1214.

12. Chin, K., Soh, S., & Meng, C. (2012). A novel scheduler for concurrent Tx/Rx wireless mesh networks with weighted links. IEEE Communications Letters, 16(2), 246–248.

13. Dai, H., Liew, S., & Fu, L. (Oct. 2011). Link scheduling in multitransmit-receive wireless networks. In IEEE Conference on Local Computer Networks (pp. 199–202). Bonn, Germany.

14. Nedevschi, S., Patra, R., Surana, S., Ratnasamy, S., Subramanian, L., & Brewer, E. (2008). An adaptive, high performance mac for long-distance multihop wireless networks. In ACM MOBICOM. (pp. 259–270). New York, NY, USA.

15. Sundaresan, K., Sivakumar, R., Ingram, M.-A., & Chang, T.-Y. (2004). Medium access control in ad hoc networks with mimo links: Optimization considerations and algorithms. IEEE Transactions on Mobile Computing, 3, 350–365.

16. Shepard, C., Yu, H., an dE. Li, N. A., Marzetta, T., Yang, R., & Zhong, L. (2012). Argos: Practical many-antenna base stations. In ACM MOBICOM Istanbul, Turkey.

17. Blcskei, H., Gesbert, D., Papadias, C. B., & v. d. Veen, A.-J. (2006). Space-time wireless systems: From array processing to MIMO communications. Cambridge: Cambridge University Press.

18. Mudumbai, R., Singh, S., & Madhow, U. (Apr. 2009). Medium access control for 60 GHz outdoor mesh networks with highly directional links. In 28th IEEE International Conference on Computer Communications (INFOCOM) Rio de Janeiro, Brazil.

 19. Jain, K., Padhye, J., Padmanabhan, V., & Qiu, L. (2003). Impact of interference on multi-hop wireless network performance. In ACMMOBICOM (pp. 66–80). New York, NY, USA.

20. Matula, D. W., & Beck, L. L. (1983). Smallest-last ordering and clustering and graph coloring algorithms. Journal of the ACM (JACM), 30, 417–427.

21. Scheinerman, E. R. (2008). Matgraph: a MATLAB toolbox for graph theory. Department of applied mathematics and statistics, the Johns Hopkins University, Baltimore, Maryland (pp. 1–7).

 22. Liu, C., Leung, K., & Gkelias, A. (2014). A generic admission control methodology for packet networks. IEEE Transactions on Wireless Communications, 13, 604–617.