مجله علمی - پژوهشی "پدافند الکترونیکی و سایبری"
سال چهارم، شماره ۳، پاییز ۱۳۹۵، ص ۸۲-۷۱

کمینه‌سازی تعداد تحویل‌ها در شبکه‌های بی‌سیم با حرکت گروهی کاربران

منصور داودی منفرد[1*]، اسماعیل دلفراز پهلوانلو[2]، سجاد قبادی بابی[3]

۱- استادیار، ۲ و ۳- دانشجوی کارشناسی ارشد، دانشکده علوم رایانه و فناوری اطلاعات، دانشگاه تحصیلات تکمیلی علوم پایه، زنجان

(دریافت: ۹۵/۰۲/۱۹، پذیرش: ۹۵/۰۵/۱۱)

چکیده

هنگامی که یک کاربر در محیط تحت پوشش یک شبکه بی‌سیم حرکت می‌کند، برای دریافت سرویس‌های مورد نظر خود ممکن است پیوسته به نقاط دسترسی متعددی متصل شود و عملیات تحویل را موجب شود. وقوع تحویل‌ها می‌تواند باعث ایجاد اختلال در ارتباط کاربر با شبکه شود. هدف ما در این مقاله کمینه‌سازی برخط تکرار تحویل‌ها در شبکه‌های بی‌سیم با ظرفیت سرویس‌دهی محدود نقاط دسترسی است. ما این مسئله را با در نظر گرفتن دو حالت روی حرکت کاربران تحلیل می‌کنیم: ۱- هر کاربر بتواند درون شبکه مسیر حرکت دلخواه خود را داشته باشد. ۲- کاربران به صورت گروهی و با هم حرکت کنند. در حالت اول با فرض اینکه اگر کاربری به نقطه دسترسی متصل شود تا هنگامی که این نقطه دسترسی برای کاربر مذکور در دسترس است باید اتصال خود را به آن ادامه دهد، ثابت می‌کنیم که هیچ الگوریتم رقابتی نمی‌تواند در حالت برخط این مسئله را با ضریب رقابتی محدود حل کند. در حالت دوم ما یک الگوریتم بهینه در حالت برون‌خط ارائه می‌دهیم و همچنین در حالت برخط ما یک الگوریتم جدید برای کاهش تعداد تحویل‌هایی که برای تمام کاربران در شبکه بی‌سیم رخ می‌دهد، ارائه می‌دهیم و ثابت می‌کنیم ضریب رقابتی الگوریتم ارائه شده، یک حد پایین برای تمامی الگوریتم‌های رقابتی در حالت برخط می‌باشد.

واژه‌های کلیدی: شبکه‌های بی‌سیم، الگوریتم برون‌خط، الگوریتم برخط، بهینه‌سازی، ضریب رقابتی

۱- مقدمه

سرعت گسترش شبکه‌های بی‌سیم[1] در سال‌های اخیر با توجه به پهنای باند[2] مناسب، قیمت ارزان و سهولت استفاده به دلیل رهایی از قید اتصال باسیم[3]، موجب جذب مطالعات گوناگون در بهبود کارآیی آنها شده است. شبکه داده بی‌سیم معمولا از یک سیستم باسیم پشتیبانی اصلی[4] و یک یا چند نقطه دسترسی[5] بی‌سیم که خدمات‌دهی بدون نیاز به استفاده از کابل را برای کاربران فراهم می‌کنند، تشکیل شده است. هنگامی که کاربر در محیط تحت پوشش یک شبکه بی‌سیم حرکت می‌کند برای دریافت سرویس‌های مورد نیاز خود ممکن است به نقاط دسترسی متعددی متصل شود. تعویض اتصال کاربر به نقاط دسترسی نیازمند انجام عملیاتی است که تحویل[6] نامیده می‌شود. با وقوع هر تحویل ارتباط کاربر با شبکه دچار وقفه شده و در نتیجه باعث بروز تداخل در خدماتی نظیر کاربردهای چند رسانه‌ای[7]، تلفن اینترنتی[8] و همچنین افزایش مصرف انرژی در شبکه می‌شود [۱]. در یک شبکه فشرده[9] با تعداد زیاد نقطه دسترسی، یک نقطه از شبکه ممکن است توسط چندین نقطه دسترسی پوشیده شده باشد. بنابراین کاربر در این مکان‌ها چندین انتخاب برای اتصال به نقاط دسترسی دارد. چگونگی انتخاب نقاط دسترسی توسط کاربر فرایند کنترل اتصال[10] شناخته می‌شود که نقش ویژه‌ای در کیفیت سرویس دریافتی[11] کاربر بازی می‌کند. کنترل اتصال یکی از مولفه‌های مهم در شبکه‌های بی‌سیم است که تأثیر بسزایی در کارایی سیستم دارد، به ویژه هنگامی که نقاط دسترسی به صورت فشرده در محیط توزیع شده باشند. اگر چه تاکنون مطالعات بسیاری برای این منظور انجام شده است [۳-۲]، اما بسیاری از تولیدکنندگانی که از محصولات *WLAN* استفاده می‌کنند نیز روش خود را در این جهت به کار برده‌اند [۵-۴]. به صورت کلی تصمیماتی که هنگام کنترل اتصال گرفته می‌شوند از جهات گوناگون روی کارآیی شبکه تأثیر مستقیم می‌گذارند. از جمله این موارد می‌توان به تعادل بار[12] میان نقاط دسترسی، توزیع عادلانه ظرفیت مفید

*رایانامه نویسنده مسئول: mdmonfared@iasbs.ac.ir

1- Wireless Networks
2- Bandwidth
3- Wires Connection
4- Infrastructure Wired Network
5- Access Point
6- Handoff or Handover
7- Multimedia Applications
8- Voice over Internet Protocol (VoIP)
9- Dense Networks
10- Association Control
11- Quality of Service
12- Load Balancing

شبکه میان کاربران[1] و نیز بروز وقفه در ارتباط کاربر با شبکه اشاره کرد. تمامی روش‌های کنونی تلاش می‌کنند که یکی و یا چند هدف از این موارد را بهینه کنند.

با توجه به این‌که در استانداردهای $IEEE$ 802.11 هیچ مکانیسم ویژه‌ای برای کنترل اتصال در نظر گرفته نشده است، بیشتر تولیدکنندگان در محصولات خود، نقطه دسترسی با بیشترین قدرت سیگنال دریافتی از آن را برای اتصال انتخاب می‌کنند [۴]. به این معنی که هنگامی که قدرت سیگنال دریافتی از نقطه دسترسی که کاربر هم اکنون به آن متصل است از یک حد آستانه کمتر می‌شود، با سنجش سیگنال نقاط دسترسی، کاربر به نقطه دسترسی با بیشترین قدرت سیگنال متصل می‌شود. قدرت سیگنال نقاط دسترسی تابع عوامل متعددی مانند وجود موانع و چگونگی حرکت است. بنابراین استفاده از این روش ممکن است موجب بروز تحویل‌های تکراری زیادی شود [۶]. در حال حاضر دو روش کلی متمرکز[2] و غیرمتمرکز[3] برای کنترل اتصال در شبکه‌های بی‌سیم وجود دارد [۸-۷]. در روش متمرکز فرض می‌شود که یک ایستگاه مرکزی وظیفه کنترل اتصال کاربران به نقاط دسترسی را بر عهده دارد. در این حالت فرض می‌شود که این ایستگاه مرکزی با داشتن دیدی از بالا و تسلط به کل مجموعه، بسته به اهداف و نوع مسئله با در اختیار داشتن دانش کافی از تعداد کاربران حاضر در سیستم، چگونگی اتصال آنها به نقاط دسترسی و دیگر اطلاعات مورد نیاز، تصمیمات اتصال هر کاربر را اتخاذ می‌کند. از طرف دیگر در روش غیرمتمرکز هر کاربر به طور مستقل و بدون آگاهی از اینکه دیگر کاربران به چه صورت به نقاط دسترسی متصل شده‌اند در هر زمان تصمیم می‌گیرد که کدام نقطه دسترسی را برای اتصال انتخاب نماید. با توجه به این مطلب، به منظور ایجاد تعادل بار نقاط دسترسی، هر نقطه دسترسی می‌تواند بار کنونی خود را از طریق پیام رادیویی انتشار دهد و هر کاربر می‌تواند نقطه دسترسی با حداقل مقدار بار که در نزدیکی آن قرارد دارد را برای اتصال انتخاب کند [۵]. برای رسیدن به یک کیفیت سرویس دریافتی(QoS) معین، روش کنترل اتصال می‌تواند به این صورت باشد که کاربر یک مقدار پهنای باند معین مورد نیاز خود را از شبکه درخواست کند و به دنبال آن به نقطه دسترسی که می‌تواند این مقدار را برای آن فراهم کند متصل شود [۳-۲]. همچنین برای رسیدن به عدالت پهنای باند میان کاربران مختلف، $Bejerano$ و همکارانش [۹] الگوریتم‌های کنترل اتصال ارائه داده‌اند که با مدیریت یک ایستگاه مرکزی، عدالت $max-min$ را به صورت بهینه میان کاربران فراهم کنند. sun و همکارانش [۱۰] مسئله بهینه‌سازی همزمان تعداد تحویل‌ها و توان عملیاتی سیستم در شبکه‌های بی‌سیم مورد استفاده در سیستم حمل‌ونقل را مورد بررسی قرار داده‌اند و برای اثبات $NP-hard$ بودن آن، یک الگوریتم متمرکز تقریبی برای آن ارائه داده‌اند.

تاکنون مطالعات زیادی برای کاستن تأثیر تحویل‌ها از طریق کاهش تأخیر هر تحویل به صورت جداگانه صورت گرفته است [۱و ۱۳ -۱۱]. اگر چه این روش‌ها صرفه‌جویی مناسبی برای تأخیر هر تحویل فراهم می‌کنند، اما تحویل‌های تکراری هنوز چالشی برای کاربردهای تعاملی در شبکه‌های بی‌سیم هستند. کمینه‌سازی تکرار تحویل ها[4] به فرآیندی گفته می‌شود که در آن اتصال کاربران به نقاط دسترسی به گونه‌ای کنترل می‌شود که از بروز تحویل‌های تکراری و غیرلازم جلوگیری می‌شود. در حال حاضر، اکثر مکانیسم های کنونی کاهش تعداد تحویل‌ها از روش‌های اکتشافی[5] برای انتخاب نقطه دسترسی مناسب در کنترل اتصال استفاده می‌کنند. $sung$ و همکارانش [۱۴] با ارائه روش‌هایی تلاش می‌کنند به‌گونه‌ای مسیر حرکت کاربر را پیش‌بینی کرده و نقطه دسترسی مناسب را برای آن انتخاب نمایند که از وقوع تحویل‌های غیرلازم جلوگیری شود. به‌گونه‌ای مشابه، با استفاده از روش GPC[6]، می‌تواند اطلاعات حرکتی کاربر را در مدت زمان طولانی جمع‌آوری کرده تا مسیر حرکت آن در آینده پیش‌بینی شود و با استفاده از این مسیر، اتصال کاربر به گونه‌ای انجام شود که هم سرعت انجام تحویل افزایش یابد و هم تعداد تحویل‌ها کاهش یابد [۱۵].

sun و همکارانش [۱۶] مسئله کمینه‌سازی تحویل‌ها و متعادل کردن بار نقاط دسترسی را که به صورت یک مسئله $min-max$ است، مطالعه کرده‌اند و با اثبات $NP-Hard$ بودن آن، الگوریتم‌های تقریبی برای دو حالت متمرکز و غیر متمرکز این مسئله ارائه داده‌اند. برای کاهش تکرار تحویل‌ها Kim و همکارانش [۶] با ارائه روش‌هایی هوشمندانه، سعی داشته‌اند کاربران به گونه‌ای نقاط دسترسی را برای اتصال انتخاب کنند که مدت زمان اتصال آنها حداکثر شود و بدین وسیله تعداد تحویل‌های به وجود آمده را کاهش دهند. همچنین به طور مشابه $Eppstein$ و همکارانش [۱۷] مسئله اختصاص یک شی در حال حرکت به ناحیه‌های بیانگر محدوده تحت پوشش دستگاه‌های سرویس‌دهنده ارتباط بی‌سیم را با هدف کاهش تعداد تحویل‌ها مورد مطالعه قرار داده‌اند. اگر چه این روش‌ها نتایج خوبی برای بهینه‌سازی تعداد تحویل‌ها هنگامی که ظرفیت نقاط دسترسی محدود نباشد را به دست می‌دهند، اما در عمل هر نقطه دسترسی در هر زمان می‌تواند به تعداد محدودی کاربر

1- Bandwith Fairness
2- Centralized
3- Deentralized
4- Handoff Frequency Minimization
5- Heuristic methods
6- Global path catching

خدمات دهد، بنابراین این روش‌ها ممکن است در یک شبکه واقعی کارآیی لازم را نداشته باشند. اگر چه مکانیسم‌های [۶ و ۱۷] نتایج بسیار مناسبی برای بهینه کردن تعداد تحویل‌ها را به‌دست آورده‌اند اما در تمام این مطالعات فرض می‌شود که هر نقطه دسترسی می‌تواند در هر زمان به تعداد نامتناهی کاربر خدمات دهد. در عمل دو عامل محدود کننده برای اینکه چه تعداد کاربر بطه صورت همزمان می‌توانند به نقطه دسترسی واحد متصل شوند وجود دارد: ۱- بعضی از تولیدکنندگان روی تعداد کاربرانی که می‌توانند همزمان به یک نقطه دسترسی متصل شوند محدودیت‌های فنی قرار می‌دهند، ۲- مقدار ترافیک داده‌ای کاربران (مقدار زیاد یا کم دانلود و بارگذاری)، می‌تواند یک محدودیت عملی برای اینکه چه تعداد کاربر همزمان می‌توانند به صورت مفید از یک نقطه دسترسی بهره ببرند، باشد [۱۹-۱۸]. با در نظر گرفتن این محدودیت، نقاط دسترسی با ظرفیت نامحدود در عمل در $WLAN$ها در دسترس نیستند. در این مقاله ما روش‌هایی برای کمینه‌سازی تعداد تحویل‌ها در شرایطی که نقاط دسترسی در هر زمان می‌توانند به تعداد محدودی کاربر خدمات دهند را ارائه می‌کنیم. همچنین مسئله رایج در شبکه‌های حسگر بی‌سیم، ردیابی برخط اجسام در حال حرکت می‌باشد [۲۴-۲۰]. به طور مشابه تحویل‌ها در خدمات تلفن همراه برای پیگیری مشتریان در حال حرکت مورد بررسی قرار می‌گیرد [۲۵]. در دو موضوع ردیابی حسگر و برنامه‌های کاربردی تلفن همراه، تحویل، سربار قابل توجهی روی سیستم اعمال می‌کند [۲۲-۲۰، ۲۴ و ۲۵]، و ما به دنبال کاهش تعداد تحویل‌هایی که رخ می‌دهد، هستیم.

تاکنون هیچ مطالعه‌ای با هدف مدیریت برخط[1] تکرار تحویل‌ها هنگامی که هر نقطه دسترسی در هر زمان تنها به تعداد محدودی کاربر قابلیت سرویس‌دهی داشته باشد انجام نشده است. در این مقاله، ما مسئله کمینه‌سازی برخط تکرار تحویل‌ها با ظرفیت سرویس‌دهی محدود نقاط دسترسی در شبکه بی‌سیم را مورد مطالعه قرار داده‌ایم و به منظور ارائه و تحلیل استراتژی‌های کنترل تحویل، از چارچوب تحلیل رقابتی در سنجش کیفیت راه‌حل‌های ارائه شده بهره می‌بریم. بر این اساس، با در نظر گرفتن دو حالت روی چگونگی حرکت کاربران به بررسی این مسئله پرداخته‌ایم: ۱- هر کاربر بتواند درون شبکه مسیر دلخواه خود را داشته باشد، ۲- حرکت گروهی کاربران که در آن کاربران به صورت گروهی با هم حرکت می‌کنند. در حالت اول با فرض اینکه بعد از اختصاص یک کاربر به یک نقطه دسترسی که توانایی ارائه سرویس به آن را دارد، تا زمانی که این نقطه دسترسی در دسترس این کاربر می‌باشد اتصال آن به این نقطه دسترسی ادامه دارد، ثابت می‌کنیم که هیچ الگوریتمی با ضریب رقابتی محدود نمی‌تواند به صورت برخط این مسئله را حل کند. در حالت گروهی، ابتدا یک الگوریتم حریصانه برای حالتی که مسیر حرکت کاربران از قبل مشخص باشد (برون‌خط) ارائه و بهینگی آن را ثابت می‌کنیم و سپس برای حالت برخط با حرکت گروهی، الگوریتمی ارائه داده و آن را در دو حالت متمرکز و غیرمتمرکز بررسی می‌کنیم که اجرای آن در این دو حالت بدون نیاز به داشتن هیچ دانشی از چگونگی حرکت کاربران انجام می‌شود. در تحلیل این الگوریتم در دو حالت متمرکز و غیرمتمرکز ثابت می‌کنیم که ضریب رقابتی آن حداکثر Δ است که Δ حداکثر تعداد نقاط دسترسی است که کاربران در طول مسیر حرکت خود می‌توانند به آنها متصل شوند. یکی از نتایج مهم که در این مطالعه به دست آوردیم، ضریب رقابتی این الگوریتم در حالت مورد انتظار $O(log\,\Delta)$ می‌باشد که ضریب رقابتی این الگوریتم در حالت مورد انتظار به صورت قابل توجهی نسبت به حالت قطعی کاهش پیدا می‌کند. یکی دیگر از نتایج این مقاله، بهینگی الگوریتم ارائه شده در حالت برخط می‌باشد. ما ثابت کردیم هیچ الگوریتم قطعی در حالت برخط با حرکت گروهی کاربران، ضریب رقابتی بهتر از Δ ندارد.

۲- تعریف مسئله

۲-۱- مدل شبکه

در این مطالعه، ما یک شبکه بی‌سیم منطبق با استاندارد $IEEE$ 802.11 را در نظر گرفته‌ایم که دسترسی به یک شبکه بی‌سیم که همه‌جا و در هر زمان در دسترس است را برای دستگاه‌های قابل حمل فراهم می سازد. هم اکنون بسیاری از سازمان‌ها به دلیل ارزان بودن و پهنای باند مناسب، در فضای خود از چنین شبکه‌ای برای ارتباطات داخلی و جهانی استفاده می‌کنند. در آینده‌ای نزدیک، شبکه‌های بی‌سیم در هم‌بافته[2] حتی مقیاس‌های گسترده‌تری مانند مراکز تجاری شهرها و حتی محیط یک شهر را نیز پوشش خواهند داد. با توجه به پوشش محدود هر نقطه دسترسی، برای خدمات‌رسانی به کل محیط، معمولا شبکه از تعداد زیادی (صدها و یا حتی هزاران) نقطه دسترسی تشکیل می‌شود. همچنین در بیشتر مواقع قرارگیری نقطه دسترسی به گونه‌ای است که در هر زمان یک دستگاه موبایل ممکن است در محدوده تحت پوشش چندین نقطه دسترسی قرار گیرد و باید برای دریافت سرویس مورد نیاز خود

1- Online

2- Mesh Networks

یکی از آنها را انتخاب کرده و به آن متصل شود. در حال حاضر معمولا دستگاه‌های موبایل از میان نقاط دسترسی انتخاب خود را بر اساس قدرت سیگنال دریافتی انجام می‌دهند. بر این اساس، هنگامی که قدرت سیگنال نقطه دسترسی که کاربر به آن متصل شده از یک حد آستانه کمتر شود، اتصال خود را با نقطه دسترسی با بیشترین سیگنال به‌روز رسانی می‌کند. از آن جا که قدرت سیگنال تحت تأثیر عوامل متعددی مانند موانع و چگونگی حرکت است، ممکن است کاربر تحویل‌ها تکراری و غیرلازم زیادی را در طول مسیر حرکت خود تجربه کند [۱]. یک نمونه از مسئله کمینه‌سازی تعداد تحویل‌ها به صورت زیر تعریف می شود:

یک مجموعه از نقاط دسترسی A و مجموعه‌ای از کاربران U داده شده‌اند. زمان به برش‌های مجزای $1,2,\dots,T$ تقسیم شده است. در هر برش زمانی t، هر کاربر $u \in U$ مجموعه‌ای از نقاط دسترسی کاندید $a \in C_u(t)$ می‌گوییم که a برای کاربر u در برش زمانی t در دسترس است. هر نقطه دسترسی a دارای ظرفیت سرویس‌دهی l_a است که برابر حداکثر تعداد کاربرانی است که می‌توانند در هر برش زمانی به آن متصل شده و از آن سرویس بگیرند. با توجه به این محدودیت، ما فرض می‌کنیم که پوشش نقاط دسترسی یک پوشش کامل است، یعنی ظرفیت و پوشش نقاط دسترسی به گونه‌ای است که در هر نقطه از شبکه مجموع ظرفیت نقاط دسترسی پوشش‌دهنده آن نقطه بیشتر یا مساوی تعداد کاربران حاضر در شبکه است. همچنین اگر Pمجموعه‌ای از نقاط دسترسی باشد، تعداد نقاط دسترسی P را با $\#P$ و مجموع ظرفیت نقاط دسترسی آن را با $|P|$نشان می دهیم، یعنی:

$$|P| = \sum_{a\in P} l_a. \qquad (۱)$$

هر کاربر u در ابتدای هر برش زمانی t باید دقیقا به یکی از نقاط دسترسی موجود در مجموعه $C_u(t)$ که ظرفیت باقیمانده آن بیشتر از صفر است متصل شود. اگر نقطه دسترسی که کاربر در برش t به آن متصل شده است متفاوت از نقطه دسترسی باشد که در برش $t-1$ به آن متصل شده باشد، می‌گوییم که کاربر در برش زمانی t متحمل یک تحویل شده است، برای راحتی در تمام طول این مقاله ارتباط کاربران در برش اول را نیز تحویل در نظر می‌گیریم. در این صورت نیاز به طراحی مکانیسمی برای اتصال کاربران به نقاط دسترسی داریم که تعداد کل تحویل‌هایی که کاربران در تمام برش‌های زمانی متحمل می‌شوند را کمینه کند .ما مسئله بالا را از چندین بعد بررسی می‌کنیم:

۱. برون‌خط[1] در مقابل برخط: در حالت برون‌خط مقادیر $C_u(t)$ برای هر برش زمانی t و هر کاربر u به عنوان ورودی مسئله داده شده است و این به معنی آن است که حرکت کاربران از قبل مشخص است. در حالت برخط، برای هر کاربر u، مجموعه $C_u(t)$ تنها در طول برش زمانی کنونی t، مشخص می شود و ما باید به صورت بلادرنگ یکی از نقاط دسترسی ممکن مجموعه $C_u(t)$ را بدون داشتن هیچ دانشی از مقادیر آینده: $C_u(t+1), C_u(t+2), \dots, C_u(T)$ انتخاب کنیم و آن را به کاربر اختصاص دهیم.

۲. حرکت دلخواه در مقابل حرکت گروهی کاربران: در حالت حرکت دلخواه هیچ فرضی روی چگونگی حرکت کاربران وجود ندارد و هر کاربر می‌تواند به دلخواه مسیر حرکت خود را در محدوده تحت پوشش نقاط دسترسی داشته باشد. در حالت حرکت گروهی فرض می‌کنیم که کاربران به صورت گروهی و با هم حرکت می‌کنند، به عبارت دیگر در هر برش زمانی t، مجموعه $C_u(t)$ برای تمام کاربران $u \in U$ برابر است، از این رو می‌توان در هر برش زمانی t، نقاط دسترسی که برای کاربران در دسترس هستند را با $C(t)$ نشان داد.

۳. متمرکز در مقابل غیرمتمرکز: در حالت متمرکز یک ایستگاه مرکزی وظیفه کنترل اتصال کاربران به نقاط دسترسی را بر عهده دارد. اما در حالت غیرمتمرکز هر کاربر، مستقل از دیگر کاربران، چگونگی اتصال خود به نقاط دسترسی را برعهده دارد.

۲-۲- ضریب رقابتی برای مدیریت برخط تکرار تحویل‌ها

در شبکه بی‌سیم ضریب رقابتی[2] یک الگوریتم برخط به صورت نسبت بین کارآیی آن الگوریتم بر الگوریتم برون‌خط بهینه که دانش کاملی از چگونگی حرکت کاربران در شبکه دارد و مسئله را با بهترین کیفیت ممکن حل می‌کند، تعیین می‌شود. به عبارت دیگر با گرفتن یک نمونه مسئله I، اگر $Cost(I)$ بیانگر هزینه الگوریتم برخط و $OPT(I)$ بیانگر هزینه الگوریتم برون‌خط بهینه روی I باشد، ضریب رقابتی الگوریتم برخط که آن را با c نشان می‌دهیم برابر است با $Cost(I) / OPT(I)$ که به ازای تمام نمونه مسئله‌های I تعریف می‌شود. به بیان دیگر:

$$c = max_I \frac{Cost(I)}{OPT(I)}. \qquad (۲)$$

یادآوری می‌کنیم که $OPT(I)$ هزینه الگوریتم بهینه برون‌خط در مسئله I است، یعنی زمانی که از حرکت کاربران در آینده، آگاهی کامل داریم و همچنین $Cost(I)$ هزینه الگوریتم برخط در مسئله I می‌باشد یعنی زمانی که دانشی نسبت به حرکت آینده کاربران نداریم و فقط با توجه به موقعیت فعلی کاربران قادر به تشخیص نقاط دسترسی هستیم که حداقل سرویس ممکن را می‌توانند به

1- Offline

2- Competitive Ratio

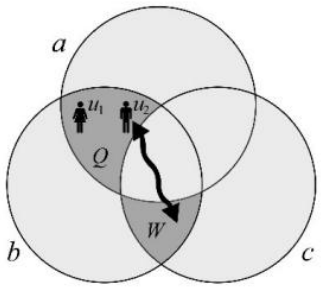
کاربران ارائه دهند.

۳- کمینه‌سازی برخط تحویل‌ها با حرکت دلخواه کاربران و ظرفیت محدود

در یک شبکه بی‌سیم هنگامی که هر کاربر بتواند مسیر دلخواه خود را مستقل از کاربران دیگر داشته باشد، و نقاط دسترسی دارای ظرفیت محدود باشند، مطابق نتایج Kim [۶] کمینه‌سازی تحویل‌ها حتی برای حالت برون‌خط نیز یک مسئله $NP-Complete$ است. در این قسمت ما با فرض "پیوستگی ارتباط" که به صورت زیر تعریف می‌شود: اگر کاربری به نقطه دسترسی متصل شود تا هنگامی که قدرت سیگنال دریافتی از این نقطه دسترسی از یک حد آستانه مشخص کمتر نشود کاربر مذکور باید اتصال خود را به آن ادامه دهد [۴، ۹]، مسئله حرکت دلخواه کاربران را بررسی می‌کنیم. در ادامه ثابت می‌کنیم که در حالت برخط اگر پیوستگی ارتباط داشته باشیم هیچ الگوریتم رقابتی نمی‌تواند این مسئله را با ضریب رقابتی محدود حل کند.

قضیه ۱: هیچ الگوریتمی با ضریب قابتی محدود برای مسئله کمینه‌سازی برخط تحویل‌ها با محدویت ظرفیت نقاط دسترسی و حرکت دلخواه کاربران و با فرض پیوستگی ارتباط وجود ندارد.

اثبات: طبق شکل (۱) یک شبکه بی‌سیم با سه نقطه دسترسی a و b و c، و دو کاربر u_1 و u_2 به صورت زیر را در نظر بگیرید: ما دو ناحیه Q و W را داخل محدوده تحت پوشش این سه نقطه دسترسی به صورت نشان داده شده در شکل (۱) در نظر می‌گیریم و فرض می‌کنیم که ظرفیت هر سه نقطه دسترسی یک است. فرض کنید که ابتدا دو کاربر در ناحیه Q و تحت پوشش دو نقطه دسترسی a و b قرار دارند، بنابراین هر کدام می‌توانند به یکی از این دو نقطه دسترسی متصل شوند. ما فرض می‌کنیم که کاربر u_1 به b و کاربر u_2 به a متصل شود. حال اگر u_2به طور متناوب بین دو ناحیه Q و W جابجا شود، در هر بار ورود و خروجش از ناحیه W نیاز به یک تغییر اتصال بین نقاط دسترسی a و c خواهد داشت، درحالی که در الگوریتم برون‌خط بهینه چون مسیر حرکت کاربران از قبل مشخص است در ابتدا کاربر u_2 به bو u_1 به a متصل می‌شود و هیچ تحویلی صورت نمی‌گیرد. با توجه به این مثال با فرض پیوستگی ارتباط برای هر کاربر، هیچ الگوریتمی چه در حالت متمرکز چه غیرمتمرکز نمی‌تواند به صورت رقابتی مسئله کمینه‌سازی برخط تحویل‌ها با حرکت دلخواه کاربران و ظرفیت محدود نقاط دسترسی را حل کند. با توجه به مثال بالا قضیه اثبات شد.



شکل (۱). با توجه به مفاهیم استفاده شده در قضیه ۱، u_1 به b متصل می‌شود و u_2 بعد از اتصال به a بین ناحیه‌های Q و W جابجا می‌شود.

۴- مدیریت تکرار تحویل‌ها با فرض حرکت گروهی کاربران

همان‌طور که در قسمت قبل نشان داده شد، مسئله کمینه‌سازی برخط تکرار تحویل‌ها به صورت رقابتی در شرایطی که ظرفیت نقاط دسترسی محدود و کاربران بتوانند حرکت دلخواه خود را داشته باشند امکان‌پذیر نیست. در این قسمت با در نظر گرفتن یک محدودیت روی چگونگی حرکت کاربران سعی در ارائه یک راه‌کار برخط برای مدیریت تحویل‌ها داریم. در این حالت خاص، فرض می‌کنیم شیوه حرکت هر کاربر تحت تأثیر چگونگی حرکت کل گروه کاربران است. به طور ویژه، در این حالت برای هر دو کاربر u_i و u_j، مجموعه نقاط دسترسی در دسترس برای u_i در طول یک برش زمانی مشخص، برابر مجموعه نقاط دسترسی در دسترس در طول همان برش زمانی برای u_j است. به عبارت دیگر:

$$\forall\, u_i, u_i \subseteq U\,, \forall t, C_{u_i}(t) = C_{u_j}(t). \qquad (۳)$$

با توجه به این ویژگی می‌توان برای سادگی مجموعه نقاط دسترسی در دسترس برای گروه کاربران در برش زمانی t را با $C(t)$ نشان داد. اگر چه Kim و همکارانش [۶] یک راه‌حل بهینه بر پایه شبکه جریان برای حالت برون‌خط این مسئله ارائه کرده‌اند اما تاکنون طراحی یک الگوریتم کارا برای حالت برخط مسئله همچنان به عنوان یک مسئله باز باقیمانده است. در این بخش ما ابتدا یک الگوریتم حریصانه برای حالت برون‌خط مسئله ارائه و بهینگی آن را ثابت می‌کنیم. سپس با ارائه راه‌حل‌های برخط، ضریب رقابتی آنها را تحلیل می‌کنیم.

۴-۱- حالت برون‌خط با فرض حرکت گروهی کاربران

همان‌طور که گفته شد در حالت برون‌خط برای تمام برش‌های زمانی $t = 1, 2, \dots, T$، مجموعه‌های $C(t)$ و تعداد کاربران در هر برش و همچنین برای تمام نقاط دسترسی $a \in A$، ظرفیت a یعنی l_a به عنوان ورودی داده شده است و در ادامه

هدف ارائه الگوریتمی است که در هر برش زمانی هر کاربر را به یکی از $a \in C(t)$ به گونه‌ای متصل کند که مجموع کل تحویل-های تمام کاربران در تمام برش‌های زمانی کمینه شود.

الگوریتم برون خط بهینه حریصانه: در این قسمت ما یک الگوریتم حریصانه برای مسئله برون‌خط کمینه‌سازی تحویل‌ها ارائه می‌دهیم و ثابت می‌کنیم این الگوریتم مسئله کمینه‌سازی تحویل‌ها را به صورت بهینه حل می‌کند و سپس با استفاده از این الگوریتم حریصانه بهینگی الگوریتم خود را در حالت برخط نمایش می‌دهیم.

الگوریتم برون‌خط به صورت زیر کار می‌کند: برای این منظور ما در هر بش زمانی t و برای هر نقطه دسترسی مانند a مقداری را به عنوان "اندازه a" به صورت زیر تعریف می‌کنیم: اندازه a در برش زمانی t برابر است با تعداد برش‌های زمانی بعد از برش زمانی t که a به صورت پیوسته برای کاربر در دسترس باشد. به عنوان مثال اگر a در برش زمانی $t = 1$ و نیز برش‌های زمانی $t = 2, 3, 4$ در دسترس کاربر باشد اما در برش زمانی $t = 5$ در دسترس کاربر نباشد، آنگاه اندازه a در برش زمانی $t = 1$ برابر ۳ است. الگوریتم برون‌خط حریصانه در هر برش زمانی t، انتخاب خود را بر اساس اندازه هر نقطه دسترسی $a \in C(t)$ انجام می‌دهد.

در هر برش زمانی t که $t = 1, 2, \dots, T$ و برای هر کاربر، فرض کنید که کاربر در برش قبلی به یک نقطه دسترسی مانند a متصل بوده است، اگر a در برش t نیز در دسترس باشد اتصال کاربر به این نقطه دسترسی در برش زمانی کنونی نیز ادامه می‌یابد، در غیر این صورت از میان مجموعه $C(t)$، یک نقطه دسترسی شدنی که دارای بیشترین اندازه است را انتخاب و کاربر را به آن متصل کرده و از ظرفیت باقیمانده آن نقطه دسترسی یک واحد کم می‌کند. یادآوری می‌کنیم که نقطه دسترسی را شدنی می‌گوییم که ظرفیت باقیمانده آن بیشتر از صفر باشد. شبه کد الگوریتم به صورت زیر می‌باشد:

Algorithm 1 Look Forward

در برش زمانی $t = 1$: هر کاربر u را به نقطه دسترسی شدنی مانند a با بیشترین اندازه متصل کن.

در هر برش زمانی $t > 1$: اگر نقطه دسترسی a برای کاربر u در دسترس بود: اتصال کاربر u به این نقطه دسترسی ادامه یابد، در غیر این صورت هر کاربر u را به نقطه دسترسی شدنی مانند $a \in C(t)$ با بیشترین اندازه متصل کن و $l_a = l_a - 1$.

یادآوری می‌کنیم که ما برای راحتی در تمام طول این مطالعه ارتباط کاربران در برش زمانی اول را نیز تحویل در نظر می‌گیریم. بنابراین الگوریتم ارائه شده در صورتی بهینه می‌باشد که تعداد کل تحویل‌های که در کل برش‌های زمانی برای تمام کاربران رخ می‌دهد را کمینه کند.

قضیه ۲: الگوریتم $Look\ Forward$ مسئله کمینه‌سازی تحویل‌ها را به صورت بهینه حل می‌کند.

اثبات: مشابه اثبات kim و همکارانش [۶] ما نیز یک راه‌حل دلخواه مانند p را برای این مسئله در نظر می‌گیریم. هنگامی که کاربر اتصال خود را به یک نقطه دسترسی از دست می‌دهد الگوریتم $Look\ Forward$ کاربر را به بزرگترین نقطه دسترسی شدنی a متصل می‌کند. در این هنگام الگوریتم p تنها دو دلیل می‌تواند داشته باشد که کاربر را به a متصل نکند: ۱- یا می‌خواهد کاربر دیگری را به این نقطه دسترسی متصل کند. ۲- و یا ممکن است بخواهد کاربر را به نقطه دسترسی دیگری متصل کند تا در آینده نقطه دسترسی بزرگتری مانند b را برای آن انتخاب کند. در دلیل اول چون هدف ما کمینه‌سازی کل تحویل‌ها برای کل کاربران در تمام برش‌های زمانی است و کاربران هیچ اولویتی نسبت به هم ندارند پس اهمیتی ندارد که این نقطه دسترسی به کدام کاربر اختصاص یابد، بنابراین p نسبت به الگوریتم $Look\ Forward$ تعداد تحویل‌های کمتری را نتیجه نخواهد داد. در دلیل دوم نیز چون الگوریتم $Look\ Forward$ بزرگترین نقطه دسترسی شدنی را انتخاب کرده است بنابراین با توجه به اندازه نقطه دسترسی، اگر الگوریتم p بعد از k تحویل کاربر را به b وصل کند، الگوریتم حریصانه با تعداد تحویل‌های کمتر یا مساوی k، کاربر را به b متصل می‌کند، پس در این حالت نیز نتیجه الگوریتم p حداکثر به خوبی نتیجه الگوریتم $Look\ Forward$ است. با توجه به گفته‌های بالا اثبات قضیه به پایان می‌رسد.

۴-۲- حالت برخط با حرکت گروهی کاربران

همان‌طور که در قسمت قبل گفته شد اگر از مسیر حرکت کاربران اطلاع داشته باشیم می‌توان اتصال کاربران را به گونه‌ای انتخاب کرد که تعداد تحویل‌های رخ داده بهینه باشد. اما در حالت برخط، ما هیچ اطلاعی از چگونگی حرکت کاربران در آینده در شبکه نداریم، بنابراین در این حالت نیاز به طراحی الگوریتمی داریم که برای اتخاذ تصمیمات خود نیاز به دانستن مسیر حرکت آینده کاربران نداشته باشد. ما در این قسمت الگوریتمی را ارائه می‌دهیم که با توجه به محل کنونی کاربران، اتصال کاربران به نقاط دسترسی موجود را تعیین می‌کند و کیفیت آن را مورد تحلیل قرار می‌دهیم. با توجه به این‌که، هنگامی که ما در یک نقطه از محیط قرار داریم آگاهی کاملی نسبت به نقاط دسترسی داریم که قادر هستند حداقل سرویس مورد نیاز کاربران را تامین کنند بنابراین، در هر نقطه از محیط مجموعه تمامی این نقاط

دسترسی را به عنوان ورودی الگوریتم در دسترس دارد. الگوریتم به این صورت عمل می‌کند که در هر برش زمانی t، یک مجموعه $B(t)$ از نقاط دسترسی که $B(t) \subseteq C(t)$ را نگه می‌دارد. برای هر کاربر u در هر برش زمانی t اگر نقطه دسترسی که در برش زمانی $B(t-1)$ به آن متصل بوده در برش زمانی t نیز در دسترس باشد، اتصال خود را به آن در برش زمانی کنونی ادامه می‌دهد در غیر این صورت یکی از نقاط دسترسی شدنی، نقطه دسترسی که در دسترس است، و ظرفیت باقیمانده آن بیشتر از صفر باشد را انتخاب می‌کند. در ادامه نحوه بروز رسانی مجموعه $B(t)$ در هر برش زمانی t که یک فرایند بازگشتی است را توضیح می‌دهیم.

Algorithm 2 Look Back

۱- در برش زمانی $t = 1$:

$$B(1) = C(1)$$

۲- در برش زمانی $t > 1$:

اگر $|B(t-1) \cap C(t)| \geq n$:

$$B(t) = B(t-1) \cap C(t)$$

در غیر این صورت: $B(t) = C(t)$.

که در آن، n تعداد کاربران که در شبکه بی‌سیم به صورت گروهی با هم حرکت می‌کنند را نشان می‌دهد و همچنین $|B(t-1) \cap C(t)|$ مجموع ظرفیت کلیه نقاط دسترسی را نشان می‌دهد که در دو برش زمانی $t-1$ و t در دسترس می‌باشند به بیان دیگر اندازه مجموع ظرفیت نقاط دسترسی $B(t-1) \cap C(t)$ است. مفهوم روابط بالا این است که اگر مجموع ظرفیت نقاط دسترسی که در مجموعه $B(t-1)$ قرار دارند و در برش کنونی t نیز در دسترس هستند، بزرگتر یا مساوی تعداد کاربران باشد، مجموعه $B(t)$ برابر همین نقاط دسترسی قرار داده می‌شوند، در غیر این صورت $B(t)$ برابر نقاط دسترسی هستند که در برش زمانی کنونی t در دسترس هستند. ما برش زمانی t را که الگوریتم در آن مجموعه $B(t)$ را برابر مجموع $C(t)$ قرار می‌دهد، اصطلاحا برش زمانی $initial$ می‌نامیم. باید متذکر شویم که مجموعه $B(t)$ در هر برش زمانی برای تمامی کاربران یکسان می‌باشد و اگر مجموع ظرفیت تمامی نقاط دسترسی که در زمان t در دسترس می‌باشند و همچنین این نقاط دسترسی در مجموعه $B(t-1)$ موجود بودند، کمتر از n باشد تمام گروه $initial$ می‌شود.

۴-۲-۱- تحلیل ضریب رقابتی الگوریتم *Look Back* در حالت متمرکز

در این قسمت هدف ما تحلیل الگوریتم $Look\ Back$ نسبت به الگوریتم برون‌خط بهینه در حالت متمرکز می‌باشد. یادآوری می‌شود که در حالت متمرکز یک ایستگاه مرکزی مسئولیت کنترل اتصال کاربران به نقاط دسترسی را بر عهده دارد. با توجه به تعریفی که از ضریب رقابتی داشتیم در ادامه الگوریتم برخط را بر اساس این مفهوم بررسی می‌کنیم و اثبات می‌کنیم ضریب رقابتی الگوریتم $Look\ Back$، Δ می‌باشد که Δ به صورت زیر تعریف می‌شود: حداکثر تعداد نقاط دسترسی که یک نقطه از محیط را پوشش می‌دهند.

در حالت برون‌خط مسیر حرکت کاربران و در نتیجه مقدار $C(t)$ برای تمام برش‌های زمانی $1, 2, \ldots, T$ از قبل مشخص است. بنابراین یک راه‌حل برای مسئله با استفاده از این ورودی این است که در هر برش زمانی t، تعدادی از نقاط دسترسی مجموعه $C(t)$ را برای اتصال کاربران انتخاب کند به گونه‌ای که تعداد کل تحویل‌ها در تمام برش‌های زمانی بهینه شود. تعداد کل تحویل‌ها در الگوریتم برون‌خط بهینه برای همه کاربران در بازه زمانی I را با $f_I(n)$ نمایش می‌دهیم که n، تعداد کاربران می‌باشد. در اینجا بازه زمانی را به صورت زیر تعریف می‌کنیم: به دو برش زمانی پشت سرهم که تمام گروه $initial$ می‌شود را یک بازه زمانی گوییم. به بیان دیگر هر بازه زمانی دلخواه برابر است با تعداد برش‌های زمانی پشت سر هم که مجموعه B در آن‌ها بروز نمی‌شود. در واقع در هر برش زمانی t، هر کاربر u مجموعه‌ای از نقاط دسترسی کاندید $C_t(n)$، که می‌توانند حداقل میزان کیفیت سرویس موردنیاز آن را فراهم کنند (نقاط دسترسی که قدرت سیگنال آنها به اندازه یک حد آستانه مشخص است) را دریافت می‌کند. به ازای هر I دلخواه در ابتدای هر بازه زمانی I، اندازه $C_I(n)$، بزرگتر مساوی n می‌باشد و همچنین طول هر بازه زمانی I به اندازه‌ای است که مجموع ظرفیت نقاط دسترسی مشاهده شده در ابتدای بازه زمانی I بزرگتر مساوی n باشد. یادآوری می‌کنیم که کاربران به صورت گروهی با هم حرکت می‌کنند و با توجه به اینکه الگوریتم به صورت متمرکز اجرا می‌شود. در این صورت مجموعه $C_I(n)$ برای کلیه کاربران مقدار یکسانی را نشان می‌دهد. متذکر می‌شویم با توجه به تعریف ما از بازه زمانی، تمامی تحویل‌ها در یک بازه زمانی دلخواه توسط نقاط دسترسی اتفاق می‌افتد که در ابتدای آن بازه زمانی ملاقات می‌شوند. واضح است که تعداد تحویل‌هایی که در الگوریتم $Look\ Back$ در بازه زمانی I رخ می‌دهد حداقل $f_I(n)$ می‌باشد. به بیان دیگر اگر تعداد تحویل‌ها در بازه زمانی I، در الگوریتم برون‌خط را با $H_{offline}(I)$ و در حالت برخط را با $H_{online}(I)$ نمایش دهیم آنگاه:

$$H_{offline}(I) = f_I(n) \leq H_{online}(I). \quad (۴)$$

فرض کنیم مجموعه‌ی $A_{opt}(I)$ همه نقاط دسترسی را که توسط الگوریتم برون‌خط بهینه در برش زمانی I برای سرویس دادن به کاربران انتخاب می‌شود را نشان می‌دهد و $k_I =$

$|A_{opt}(I)|$می‌باشد.

واضح است که $k_I \leq \Delta$ می‌باشد و همچنین فرض کردیم که حداکثر تحویل‌هایی که با اتصال کاربران به این k_I، نقطه دسترسی در بازه زمانی I رخ می‌دهد $f_I(n)$ می‌باشد.

همان‌طور که گفتیم الگوریتم برون‌خط بهینه، از مسیر حرکت کاربران اطلاع دارد پس نقاط دسترسی را انتخاب می‌کند که هیچ تحویل غیربهینه نداشته باشد ولی در الگوریتم $Look\ Back$ ما هیچ اطلاعی از مسیر آینده کاربران نداریم پس در الگوریتم $Look\ Back$ زمانی در بازه زمانی I تحویل غیربهینه داریم که برای سرویس دادن به کاربران ابتدا نقاط دسترسی که در مجموعه $A(I) - A_{opt}(I)$ قرار دارند انتخاب کند، سپس تمامی نقاط دسترسی که در مجموعه $AP_{opt}(I)$ قرار دارند انتخاب کند که $A(I)$ مجموعه تمام نقاط دسترسی است که در بازه زمانی I در دسترس کاربران هستند. یادآوری می‌کنیم که ارتباط کاربران در بازه زمانی اول را نیز تحویل در نظر می‌گیریم.

لم ۲. حداکثر تحویل‌های غیربهینه که با انتخاب یکی از نقاط دسترسی که در مجموعه $A(I) - A_{opt}(I)$ در بازه زمانی t قرار دارند رخ می‌دهد، $f_I(n)$ می‌باشد.

اثبات: با توجه به این که ما دانش کاملی نسبت به تعداد تحویل‌های بهینه‌ای که در حالت برون‌خط، برای تمامی کاربران در هر بازه زمانی I رخ می‌دهد، داریم که آن را با $f_I(n)$ نمایش می‌دهیم و می‌دانیم الگوریتم $Look\ Back$ زمانی بهینه کار نمی‌کند که ابتدا حداقل یکی از نقاط دسترسی که در مجموعه $A(I) - A_{opt}(I)$ قرار دارند را انتخاب کند و سپس نقاط دسترسی که در مجموعه $A_{opt}(I)$ قرار دارند انتخاب کند. بنابراین بدترین حالت زمانی اتفاق می‌افتد که ابتدا تمامی $f_I(n)$ تحویل بهینه توسط هر کدام از نقاط دسترسی که در مجموعه $A(I) - A_{opt}(I)$ قرار دارند رخ دهد سپس کاربران به مجموعه $A_{opt}(I)$ متصل شوند که این حالت زمانی پیش می‌آید که ظرفیت هر کدام از نقاط دسترسی مجموعه $A(I) - A_{opt}(I)$ بزرگتر مساوی $f_I(n)$ باشد. بنابراین حداکثر تحویل‌های غیربهینه که با انتخاب یکی از نقاط دسترسی که در مجموعه $A(I) - A_{opt}(I)$در بازه زمانی I رخ می‌دهند $f_I(n)$ می‌باشد. بنابراین لم اثبات شد.

قضیه ۳. ضریب رقابتی الگوریتم $Look\ Back$، Δ می‌باشد.

اثبات: با توجه به لم ۱، حداکثر تحویل‌هایی که در بازه زمانی I توسط یکی از نقاط دسترسی که در مجموعه $A(I) - A_{opt}(I)$ می‌تواند رخ دهد $f_I(n)$ می‌باشد. بنابراین به ازای هر نقطه دسترسی که توسط الگوریتم $Look\ Back$ در بازه زمانی I انتخاب می‌شود، به c که بیانگر هزینه الگوریتم $Look\ Back$ است مقدار $f_I(n)$ که بیانگر هزینه الگوریتم برون‌خط است اضافه می‌شود. بنابراین تعداد تحویل‌ها توسط الگوریتم $Look\ Back$ در بازه زمانی I به ازای هر نقطه دسترسی حداکثر $f_I(n)$ افزایش پیدا می‌کند. اگر مجموع تحویل-های بهینه تمام بازه‌های زمانی را با $f(n)$ نشان دهیم و اگر در هر کدام از بازه‌های زمانی حداکثر Δ نقطه دسترسی داشته باشیم. بنابراین ضریب رقابتی الگوریتم $LookBack$ حداکثر Δ می‌باشد. به بیان دیگر:

$$c = \frac{(f_1(n)+f_2(n)+\cdots+f_T(n))*\Delta}{f_1(n)+f_2(n)+\cdots+f_T(n)} = \frac{f(n)*\Delta}{f(n)} = \Delta. \qquad (۵)$$

۴-۲-۲- تحلیل انتخاب تصادفی الگوریتم $Look\ Back$

با توجه به قضیه ۲ در الگوریتم $Look\ Back$، کلیه کاربران در یک بازه زمانی I حداکثر Δ نقطه دسترسی برای اتصال انتخاب می‌کنند. با توجه به این که پوشش کامل داریم پس حداقل یک نقطه دسترسی وجود دارد که در کل بازه زمانی I در دسترس کاربران باشد. بنابراین اگر در الگوریتم $Look\ Back$ ما این شانس را داشته باشیم که اولین انتخاب تصادفی کاربر، نقطه دسترسی انتخاب شده توسط الگوریتم برون‌خط بهینه، یعنی نقطه دسترسی با بیشترین اندازه باشد، کمترین تعداد تحویل‌ها را خواهیم داشت.

یادآوری می‌کنیم که اندازه نقطه دسترسی a برابر تعداد برش های زمانی پشت سر هم است که a در دسترس کاربران قرار دارد و اگر نقطه دسترسی شدنی باشد یعنی ظرفیت باقیمانده بزرگتر از صفر باشد و در دسترس کاربران باشد، نقطه دسترسی a را می‌توان برای سرویس‌دهی به کاربران انتخاب کرد.

لم ۳. تعداد مورد انتظار نقاط دسترسی پذیرفته شده در یک بازه‌ی زمانی دلخواه در الگوریتم $Look\ Back$ برای یک کاربر دلخواه، $O(log\ \Delta)$ است.

فرض کنیم $X_i(I)$ متغیر تصادفی اینکه iامین نقطه دسترسی پیشنهاد شده در بازه زمانی I پذیرفته شود، باشد. آنگاه داریم: $X_i(I) = 1$ اگر نقطه دسترسی iام در بازه زمانی I پذیرفته شود. در غیر این صورت $X_i(I) = 0$.

احتمال پذیرش iامین نقطه دسترسی در یک بازه زمانی دلخواه بستگی به ظرفیت باقیمانده و اندازه آن دارد. به بیان دیگر احتمال پذیرش یک نقطه دسترسی برابر احتمال شدنی بودن آن نقطه دسترسی می‌باشد. که آن را اینگونه تعریف می‌کنیم:

احتمال پذیرش یک نقطه دسترسی برابر است با احتمال این که اندازه آن از نقاط دسترسی انتخاب شده قبلی بزرگتر باشد ضرب در احتمال این که ظرفیت باقیمانده آن بزرگتر از صفر باشد.

احتمال این که اندازه iامین نقطه دسترسی از تمام $i-1$ نقطه دسترسی قبلی که توسط الگوریتم $Look\ Back$ انتخاب

شده است بزرگتر باشد برابر است با احتمال اینکه یک عدد در بین i عدد بزرگترین باشد. بنابراین احتمال بزرگتر بودن این نقطه دسترسی از $i-1$ نقطه دسترسی برابر $\frac{1}{i}$ است.

احتمال این که ظرفیت باقیمانده یک نقطه دسترسی بزرگتر از صفر باشد بستگی به ظرفیت کل آن دارد. در واقع هر چه ظرفیت یک نقطه دسترسی بزرگتر باشد این نقطه دسترسی قادر به ارائه سرویس به تعداد کاربران بیشتری می‌باشد. به صورتی دیگر احتمال اینکه ظرفیت باقیمانده نقطه دسترسی a بزرگتر از صفر باشد به l_a که ظرفیت a را نشان می‌دهد بستگی دارد. در ادامه برای سادگی ما فرض می‌کنیم که اگر ظرفیت نقطه دسترسی iام برابر l_i باشد به جای این نقطه دسترسی l_i تا نقطه دسترسی با ظرفیت یک خواهیم داشت که هر کدام از این نقاط دسترسی فقط به یک کاربر می‌توانند اختصاص پیدا کنند. بازه زمانی I را در نظر بگیرید. اگر مقدار l_i در این بازه زمانی بزرگ باشد احتمال این که ظرفیت باقیمانده این نقطه دسترسی در این بازه زمانی بزرگتر از صفر باشد نسبت به بقیه نقاط دسترسی بیشتر می‌شود. با توجه به تحلیلی که از احتمال بزرگتر از صفر بودن باقیمانده یک نقطه دسترسی دلخواه داشتیم، احتمال این که ظرفیت باقیمانده نقطه دسترسی iام که آن را با R_i نمایش می‌دهیم بزرگتر از صفر باشد برابر احتمال این است که در بین تمام نقاط دسترسی که ظرفیت آنها را برابر یک در نظر گرفتیم، یکی از l_i، نقطه دسترسی که مربوط به نقطه دسترسی iام است، کاربر دیگری با آن در ارتباط نباشد. به بیان دیگر:

$$P\{R_i \geq 1\} = \frac{l_i}{\sum_{j=1}^{|A(I)|} l_j} \qquad (۶)$$

که $P\{R_i \geq 1\}$ نشان‌دهنده مقدار این احتمال است و $|A(I)|$ تعداد نقاط دسترسی را نشان می‌دهد که در بازه زمانی I در دسترس کاربران می‌باشد. واضح است که

$$P\{R_i \geq 1\} = \frac{l_i}{\sum_{j=1}^{|A(I)|} l_j} \leq 1. \qquad (۷)$$

بیشترین تعداد نقاط دسترسی که مجموعه $A(I)$ می‌تواند در یک بازه زمانی داشته باشد برابر Δ است. به صورتی دیگر $|A_I| = \Delta$.

بنابراین تعداد نقاط دسترسی که در یک بازه زمانی برای یک کاربر پذیرفته می‌شود برابر است با:

$$E[X_i] = \sum_{i=1}^{\Delta} \left[\frac{1}{i} * P\{R_i\}\right] \leq \sum_{i=1}^{\Delta} \frac{1}{i} = O(log\,\Delta) \qquad (۸)$$

با توجه به روابط بالا لم اثبات شد.

قضیه ۴. ضریب رقابتی الگوریتم $Look\ Back$ در حالت مورد انتظار $O(log\,\Delta)$ است.

اثبات: با توجه به اینکه تعداد کل تحویل‌هایی که در حالت بهینه در هر بازه زمانی I که برای n کاربر رخ می‌دهد برابر $f_I(n)$ می‌باشد. در این صورت $f_I(n)$ کاربر در بازه زمانی I وجود داشته‌اند که متحمل تحویل شده‌اند. طبق لم ۲، در حالت مورد انتظار ممکن است برای هر کدام از این $f_I(n)$ کاربر، در الگوریتم $Look\ Back$، $O(log\,\Delta)$ نقطه دسترسی پذیرفته شود. به بیان دیگر:

$$c = \frac{(f_1(n) + f_2(n) + \cdots + f_T(n)) * O(log\,\Delta)}{f_1(n) + f_2(n) + \cdots + f_T(n)}$$

$$= \frac{f(n) * O(log\,\Delta)}{f(n)} = O(log\,\Delta). \qquad (۹)$$

بنابراین ضریب رقابتی الگوریتم $Look\ Back$ در حالت مورد انتظار $O(log\,\Delta)$ است. در نتیجه طبق روابط بالا قضیه اثبات شد.

۴-۲-۳- تحلیل ضریب رقابتی الگوریتم $Look\ Back$ در حالت غیرمتمرکز

همان‌طور که گفته شد الگوریتم $Look\ Back$ به صورت متمرکز با توجه به تعداد کاربران و مجموعه $C(t)$ کنترل اتصال انجام می‌دهد. اما اگر این الگوریتم را توسط تک‌تک کاربران اجرا کرد و به صورت غیرمتمرکز کنترل اتصال آنها را انجام داد کاربران هیچ اطلاعی از مجموعه $B(t)$ یکدیگر نخواهند داشت و هر کاربر مشابه الگوریتم $Look\ Back$ این مجموعه را به‌روز رسانی و اتصال خود را به نقاط دسترسی انجام می‌دهد. با توجه به تعریف بازه زمانی اگر هر کاربر در یک بازه زمانی دلخواه $initial$ انجام دهد، تا انتهای بازه زمانی توسط نقاط دسترسی که در ابتدای آن بازه زمانی مشاهده شده‌اند پوشش داده می‌شوند. به ازای هر بازه زمانی I، تنها تفاوت الگوریتم غیرمتمرکز با الگوریتم متمرکز در اختلاف مجموعه $B(I)$ برای هر کاربر می‌باشد. در حالت متمرکز مجموعه $B(I)$ برای تمام کاربران یکسان می‌باشد، ولی در حالت غیرمتمرکز ممکن است کاربران مجموعه $B(I)$یکسانی نداشته باشند. یادآوری می‌کنیم که کاربران به صورت گروهی حرکت می‌کنند.

قضیه ۵. ضریب رقابتی الگوریتم $Look\ Back$ در حالت غیرمتمرکز $O(\Delta)$ می‌باشد.

اثبات: فرض کنیم در بازه زمانی دلخواه I، $f_I(n)$ کاربر $initial$می‌کنند. که ممکن است این کاربران همزمان در این بازه زمانی $initial$ انجام ندهند. به بیان دیگر هر کدام از $f_I(n)$ کاربر در یکی از برش‌های زمانی که در این بازه زمانی قرار دارند $initial$ انجام دهند. متذکر می‌شویم هنگامی که یک کاربر در یک بازه زمانی $initial$ می‌کند در واقع تمام نقاط دسترسی که قبلا در دسترس این کاربر بوده‌اند و اکنون قادر به سرویس دادن به این کاربر نمی‌باشند دو حالت دارند: ۱- این نقاط دسترسی در دسترس کاربر نمی‌باشند. یا ۲- ظرفیت این نقاط دسترسی کامل شده‌است. بنابراین تعداد کل تحویل‌هایی که در

الگوریتم برون‌خط بهینه در بازه زمانی I ممکن است رخ دهد حداقل $f_I(n)$ تحویل می‌باشد. یادآوری می‌کنیم که الگوریتم برون‌خط بهینه، از مسیر حرکت کاربران اطلاع دارد پس نقاط دسترسی را انتخاب می‌کند که هیچ تحویل غیربهینه نداشته باشد ولی در الگوریتم $Loo\ \mathrm{kBack}$ ما هیچ اطلاعی از مسیر آینده کاربران نداریم. پس در این الگوریتم زمانی در بازه زمانی I تحویل غیربهینه داریم که برای سرویس دادن به کاربران ابتدا نقاط دسترسی که در مجموعه $A(I) - \mathrm{A_{opt}}(I)$ قرار دارند انتخاب کند، سپس تمامی نقاط دسترسی که در مجموعه $AP_{opt}(I)$ قرار دارند انتخاب کند. که $A(I)$ مجموعه تمام نقاط دسترسی است که در برش زمانی I در دسترس کاربران هستند و مجموعه $A_{opt}(I)$ تمام نقاط دسترسی را نشان می‌دهد که توسط الگوریتم برون‌خط بهینه برای اتصال به کاربران در بازه زمانی I انتخاب شده است.

باتوجه به این که حداکثر تعداد نقاط دسترسی که در مجموعه $B(I)$ هر کدام از این $f_I(n)$ کاربر می تواند وجود داشته باشد Δ می‌باشد. در این صورت در بدترین حالت برای هر کدام از این $f_I(n)$ کاربر، Δ تحویل در بازه زمانی I رخ می‌دهد. بنابراین حداکثر تعداد تحویل‌هایی که در این بازه زمانی برای کلیه کاربران رخ می‌دهد $f_I(n) * \Delta$ می‌باشد. بنابراین ضریب رقابتی الگوریتم $LookBack$ برابر است با:

$$c = \frac{(f_1(n)+f_2(n)+\cdots+f_T(n))*\Delta}{f_1(n)+f_2(n)+\cdots+f_T(n)} = \frac{f(n)*\Delta}{f(n)} = \Delta. \qquad (۱۰)$$

بنابراین ضریب رقابتی الگوریتم ارائه شده در حالت غیرمتمرکز Δ می‌باشد و با توجه به رابطه بالا قضیه اثبات شد.

۵- حد پایین الگوریتم‌های قطعی در حالت برخط

در این قسمت هدف بر آن است که نشان دهیم ضریب رقابتی برای هر الگوریتم قطعی دلخواه در حالت برخط حداقل Δ می‌باشد. با توجه به اینکه ما می‌توانیم مدت زمانی که یک نقطه دسترسی در دسترس کاربران قرار دارد را به صورت یک پاره‌خط موازی با محور x در نظر بگیریم که طول هر پاره‌خط نشان دهنده مدت زمانی است که هر نقطه دسترسی می‌تواند حداقل سرویس را در اختیار کاربران قرار دهد و با توجه به تعریفی که از بازه‌های زمانی داشتیم، فرض کنیم در هر بازه زمانی Δ نقطه دسترسی در اختیار کاربران قرار دارد و نحوه سرویس دادن این نقاط دسترسی به کاربران به صورت پاره‌خط‌هایی باشد که در شکل (۲) مشاهده می‌کنید. در شکل (۲) هر کدام از پاره‌خط‌ها نشان‌دهنده یکی از نقاط دسترسی در دسترس در بازه زمانی $I-1$ می‌باشد. فرض کنیم ظرفیت هر کدام از این نقاط دسترسی برابر تعداد کاربران موجود در شبکه باشد به بیان دیگر: $\forall\, a \in A, l_a = n$. یادآوری می‌کنیم A مجموعه نقاط دسترسی موجود در شبکه و l_a ظرفیت هر نقطه دسترسی و n تعداد کاربران را نشان می‌دهد.

شکل (۲). مجموعه Δ نقطه دسترسی که بر اساس اندازه‌شان در بازه زمانی I-1 مرتب شده‌اند و Δامین نقطه دسترسی در کل این بازه زمانی در دسترس می‌باشد.

قضیه ۶. ضریب رقابتی برای هر الگوریتم قطعی دلخواه برای مسئله حرکت گروهی کاربران در حالت برخط حداقل Δ می‌باشد.

اثبات: مشابه اثبات $Eppstein$ و همکارانش [۱۷] فرض کنید در هر بازه زمانی Δ نقطه دسترسی در دسترس داریم که این نقاط دسترسی را با مجموعه $A(I)$ نشان می‌دهیم. فرض کنیم در هر بازه زمانی دلخواه نحوه سرویس‌دهی نقاط دسترسی به کاربران مانند شکل (۲) باشد. فرض کنید هیچ نقطه دسترسی در دو بازه زمانی پشت سرهم قرار نگیرد به بیان دیگر به ازای هر بازه زمانی دلخواه I، دو مجموعه $A(I-1)$ و $A(I)$ هیچ اشتراکی با هم نداشته باشند. با توجه به این که Δامین نقطه دسترسی در هر بازه زمانی I در دسترس کلیه کاربران می‌باشد و ظرفیت آن برابر تعداد کاربران می‌باشد بنابراین الگوریتم برون‌خط بهینه در هر بازه زمانی این نقطه دسترسی را برای سرویس‌دهی به کاربران انتخاب می‌کند.

حال فرض کنید A یک الگوریتم دلخواه باشد که حرکت گروهی کاربران را در حالت برخط بررسی می‌کند. توجه کنید که الگوریتم A از مسیر حرکت آینده کاربران هیچ اطلاعی ندارد. همچنین در نظر بگیرید که ما نقاط دسترسی را بر اساس اندازه آنها از یک تا Δ مرتب کرده‌ایم به بیان دیگر تمامی نقاط دسترسی را بر اساس مدت زمانی که در یک بازه‌ی زمانی دلخواه در دسترس می‌باشند به صورت صعودی مرتب کرده‌ایم که این نقاط دسترسی را به ترتیب از یک تا Δ نمایش می‌دهیم. حال اگر الگوریتم A در طول حرکت کاربران در هر بازه زمانی ابتدا اولین نقطه دسترسی و سپس دومین نقطه دسترسی و در انتها Δامین نقطه دسترسی را انتخاب کند در نتیجه در هر بازه زمانی I در الگوریتم دلخواه A برای هر کاربر، Δ تحویل رخ می‌دهد. بنابراین در هر بازه زمانی تعداد تحویل‌های رخ داده Δ برابر تعداد کاربران می‌باشد. در نتیجه ضریب رقابتی الگوریتم A، Δ می‌باشد. بنابراین هنگامی که ما از مسیر آینده کاربران اطلاعی نداریم هیچ

الگوریتم قطعی وجود ندارد که در بدترین حالت ضریب رقابتی، بهتر از Δ داشته باشد. در این صورت با توجه به مثال بالا قضیه اثبات شد.

۶- نتیجه گیری

در روش‌های ارائه شده قبلی برای کمینه‌سازی تحویل‌ها، فرض می‌شد که ظرفیت نقاط دسترسی شبکه بی‌سیم نامحدود است و تا کنون هیچ گونه مطالعه‌ای برای کمینه‌سازی تحویل‌ها در شرایطی که هر نقطه دسترسی در هر زمان بتواند به تعداد محدودی کاربر سرویس دهد، انجام نشده بود. در این مطالعه ما مسئله حرکت برخط کاربران در میان مجموعه‌ای از نقاط دسترسی را بررسی کردیم و در حالتی که حرکت کاربران به صورت دلخواه می‌باشد، با فرض پیوستگی ارتباط ثابت کردیم که هیچ الگوریتمی با ضریب رقابتی محدود نمی‌تواند این مسئله را در حالت برخط حل کند. همچنین در این مطالعه در حالتی که کاربران به صورت گروهی در شبکه در حال حرکت می‌باشند نیز بررسی کردیم و توانستیم در حالت برون‌خط الگوریتم حریصانه‌ای ارائه دهیم که این مسئله را به صورت بهینه حل کند. حرکت گروهی کاربران در حالت برخط نیز مورد بررسی قرار دادیم و توانستیم الگوریتمی ارائه دهیم که در حالت متمرکز و غیرمتمرکز بدون داشتن هیچ دانشی نسبت به مسیر آینده کاربران به صورت کارا عمل کند. ما ثابت کردیم که ضریب رقابتی این الگوریتم در حالت متمرکز و غیرمتمرکز Δ می‌باشد که Δ بیانگر حداکثر تعداد نقاط دسترسی می‌باشد که در یک نقطه از مسیر حرکت گروهی کاربران قرار دارند و کاربران می‌توانند به آنها متصل شوند. همچنین در این مقاله ثابت کردیم که ضریب رقابتی الگوریتم ارائه شده در حالت مورد انتظار $O(log\ \Delta)$ می‌باشد. یکی از نتایج مهم در این مقاله اثبات بهینگی الگوریتم ارائه شده در حالت برخط می‌باشد. ما ثابت کردیم هیچ الگوریتم قطعی با ضریب رقابتی بهتر از Δ وجود ندارد که مسئله برخط حرکت گروهی کاربران را در دو حالت متمرکز و غیرمتمرکز حل کند.

۷- مراجع

[1] C. A. Mishra, M. Shin, and W. A. Arbaush, "Context caching using neighbor graphs for fast handoffs in a wireless network," in INFOCOM 2004, Twenty-third Annual Joint Conference of the IEEE Computer and Communications Societies, vol. 1, 2004.

[2] A. Balachandran, P. Bahl, and G. M. Voelker, "Hot-spot congestion relief in public-area wireless networks," in Mobile Computing Systems and Applications, Proceedings Fourth IEEE Workshop on, pp. 70–80, 2002.

[3] T.-C. Tsai and C.-F. Lien, "IEEE 802.11 hot spot load balance and QoS-maintained seamless roaming," in Proc. National Computer Symposium (NCS), 2003.

[4] Symbol Technologies, "Wireless networker CF radio card data sheet," 2006.

[5] Cisco Systems Inc, "Aironet 802.11 a/b/g WLAN client adapter data sheet," 2006.

[6] M. Kim, Z. Liu, S. Parthasarathy, D. Pendarakis, and H. Yang, "Association control algorithms for handoff frequency minimization in mobile wireless networks," Wirel. Networks, vol. 18, no.5, pp. 535–550, Feb. 2012.

[7] Garikipati, C. Krishna, and G. S. Kang, "Distributed association control in shared wireless networks," 2013 IEEE International Conference on Sensing, Communications and Networking (SECON), IEEE, 2013.

[8] Karimi, O. Baghban, J. C. Liu, and J. Rexford, "Optimal collaborative access point association in wireless networks," IEEE INFOCOM 2014-IEEE Conference on Computer Communications," IEEE, 2014.

[9] Y. Bejerano, S.-J. Han, and L. E. Li, "Fairness and load balancing in wireless LANs using association control," Proceedings of the 10th annual international conference on Mobile computing and networking, ACM, 2004.

[10] S. Quan, L. Huang, and H. Xu, "Achieving Handoff Optimization and Throughput Efficiency in Vehicular Networks," Mobile Ad-hoc and Sensor Networks (MSN), 2013 IEEE Ninth International Conference on, IEEE, 2013.

[11] M. Shin, A. Mishra, and W. A. Arbaugh, "Improving the latency of 802.11 hand-offs using neighbor graphs," in Proceedings of the 2nd international conference on Mobile systems, applications and services, pp. 70–83, 2004.

[12] S. Pack and Y. Choi, "Fast inter-AP handoff using predictive-authentication scheme in a public wireless LAN," Networks, vol. 1, pp. 15–26, 2002.

[13] I. Ramani and S. Savage, "Sync Scan: practical fast handoff for 802.11 infrastructure networks," in INFOCOM 2005, 24th Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE, vol. 1, pp. 675–684, 2005.

[14] S. N. Woon, et al, "Base station association schemes to reduce unnecessary handovers using location awareness in femtocell networks," Wireless networks, vol. 19, no. 5, pp. 741-753, 2013.

[15] W. Wanalertlak, et al, "Scanless fast handoff technique based on global Path-Cache for WLANs," The Journal of Supercomputing, vol. 66, no. 3, pp. 1320-1349, 2013.

[16] S. Quan, et al, "Handoff optimization and load balancing in wireless LANs using association control," International Journal of Communication Systems, vol. 28, no. 4, pp. 682-704, 2015.

[17] D. Eppstein, M. T. Goodrich, and M. Löffler, "Tracking moving objects with few handovers," in Algorithms and Data Structures, Springer, pp. 362–373, 2011.

[18] Intel, "NWireless Ethernet LAN (WLAN), General 802.11a/802.11b/802.11g FAQo," 2011.

[19] A. C. Amrod and J. Woodhams, "Wireless lan design guide for high density client environments in higher education," Cisco Des. Guid., 2011.

[20] O. Ghica, G. Trajcevski, F. Zhou, R. Tamassia, and P. Scheuermann, "Selecting Tracking Principals with Epoch wareness," In: Proc. 18th ACM SIGSPATIAL Internat. Conf. on Advances in Geographic Information Systems, ACM GIS 2010.

[21] G. He and J. Hou, "Tracking targets with quality in wireless sensor networks," In: 13th IEEE Conf. on Network Protocols (ICNP), pp. 1–12, 2005.

[22] S. Pattem, S. Poduri, and B. Krishnamachari, "Energy-Quality Tradeoffs for Target Tracking in Wireless Sensor Networks," In: F. Zhao, L. Guibas (eds.) IPSN 2003, LNCS, vol. 2634, pp. 32–46, Springer, Heidelberg, 2003.

[23] K. Yi, and Q. Zhang, "Multi-dimensional online tracking," In: Proc. of the 20th ACM-SIAM Symp. On Discrete Algorithms (SODA), pp. 1098–1107, SIAM, Philadelphia, 2009.

[24] F. Zhao, J. Shin, and J. Reich, "Information-driven dynamic sensor collaboration," IEEE Signal Processing Magazine, vol. 19, no. 2, pp. 61–72, 2002.

[25] S.Tekinay and B. Jabbari, "Handover and channel assignment in mobile cellular networks, IEEE Communications Magazine vol. 29, no. 11, pp. 42–46, 1991.

Journal Of Electronical & Cyber Defence
Vol. 4, No. 3, 2016, Serial No.15

Handoff Minimization in Wireless Networks With Group Mobility

M. Davoodi Monfared*, E. Delfaraz Pahlevanloo, S. Ghobadi Babi

*Institute for Advanced Studies in Basic Sciences**,** Zanjan

(Received: 08/05/2016, Accepted: 01/08/2016)

ABSTRACT

When a user moves among a set of access points in a wireless network, the handoff operation occurs as a result of his getting constantly connected to multiple access points to get a desired service; however, frequent handoffs may disrupt his association in the network. This study aims to minimize the number of handoffs for a set of users. This problem is analyzed by considering two cases for the user's movement: (*i*) *each user can have his chosen path within the network, and (ii) all users move together and they have group mobility. In the first case, having the connectivity assumption; that is when a user "u" connects to an access point "a", while "a" is available, "u" remains connected to "a", this thesis proves that there is no competitive algorithm in an online setting for handoff minimization under connectivity assumption. However, in the second case, it proposes an optimal competitive ratio algorithm for the problem.*

Keywords: Wireless Networks, Offline Algorithm, Online Algorithm, Competitive Ratio, Handoff Minimization

* Corresponding Author Email: mdmonfared@iasbs.ac.ir