



دانشگاه صنعتی شریف
دانشکده مهندسی کامپیوتر

رساله‌ی دکتری

مهندسی کامپیوتر

عنوان

پوشش موزه‌های هنر متعامد توسط
 k -فرستنده‌های لغزان

نگارش

سلما سادات مهدوی

استاد راهنما

دکتر محمد قدسی

مهر ۱۳۹۹

اللَّهُمَّ صَلِّ وَسَلِّمْ وَبَارِكْ عَلَى سَيِّدِنَا مُحَمَّدٍ

تقدیر و تشکر

وظیفه‌ی خود می‌دانم از زحمات استاد ارجمندم، جناب آقای دکتر محمد قدسی، که با کمک‌ها و راهنمایی‌های بی‌دریغشان، بنده را در انجام این پروژه یاری داده‌اند، تشکر و قدردانی کنم. متشکرم که قبول زحمت فرمودند. شاگردی ایشان از بزرگترین افتخارات علمی زندگی من است. از اساتید محترم آقایان دکتر ایزدی، دکتر رزازی، دکتر زارعی، دکتر ضرابی‌زاده و دکتر محدث که تقبل زحمت مطالعه و داوری را فرمودند و با پیشنهادات خود باعث بهبود این رساله شدند، کمال امتنان را دارم.

از آقایان سعید صدیقین، مسعود صدیقین، شادروان محمدامین بیروتی و علی نارنجی به دلیل یاری‌ها و راهنمایی‌هایشان که بسیاری از سختی‌ها را برایم آسان‌تر نمودند سپاس‌گزارم. از اعضای آزمایشگاه الگوریتم و تمامی دوستانی که در طول انجام رساله همراه من بودند، تشکر می‌کنم. از خانواده عزیز و مهربانم که پشتیبان همیشگی من بوده‌اند کمال سپاس و تشکر را دارم و توفیق جبران محبت این عزیزان را از خداوند خواستارم.

چکیده

مسئله‌ی نگهداری موزه‌های هنر متعامد با استفاده از دوربین‌های لغزان حالت خاصی از مسئله موزه‌ی هنر است که در آن هدف، قرار دادن کمترین تعداد نگهبان برای نگهداری یک موزه است. هر نگهبان در یک نقطه قرار دارد و همه نقاطی را که در ناحیه دیدش هستند می‌تواند نگهداری کند. در مدل دوربین‌های لغزان هر نگهبان به صورت یک پاره‌خط افقی یا عمودی که به طور کامل درون موزه قرار دارد نشان داده می‌شود. ناحیه دید هر دوربین لغزان براساس پاره‌خطی که روی آن حرکت می‌کند تعریف می‌شود. به دنبال پیشرفت در تکنولوژی بی‌سیم و نیاز به ارائه سرویس‌های بی‌سیم به مشتریان، نوع جدیدی از مسائل برای پوشش یک ناحیه مورد بررسی قرار گرفتند. در این مسائل یک نگهبان به صورت یک فرستنده بی‌سیم که دارای محدوده نامحدود پخش امواج و قدرت نفوذ درون k دیوار برای رسیدن به مقصد است، مدل می‌شود. یکی از مسائلی که ما بررسی کردیم، مسئله پوشش موزه‌های هنر متعامد توسط دوربین‌های لغزانی که قدرت نفوذ تا k دیوار را دارند. هدف در این مسئله قرار دادن تعدادی k -فرستنده لغزان درون یک چندضلعی است به طوری که تعداد آن‌ها و یا مجموع طول پاره‌خط‌هایی که روی آن‌ها حرکت می‌کنند، کمینه شود. مسئله مهم پس از بدست آوردن مجموعه‌ای از دوربین‌های لغزان که بتوانند یک موزه‌ی هنر متعامد را نگهداری کنند، این است که چگونه این دوربین‌ها در راستای پاره‌خط‌های متناظرشان حرکت کنند تا بتوانند موزه‌ی هنر را نگهداری کنند. بنابراین در ادامه به بررسی مسئله طراحی مسیر حرکت برای دسته‌ای از ربات‌های لغزان برای نگهداری یک موزه‌ی هنر پرداختیم. در این مسئله فرض بر این است که تعدادی دزد که دارای سرعت حرکت بسیار زیاد هستند داخل یک چندضلعی متعامد در حال حرکتند که دارای مکان نامشخص هستند. هدف، یافتن مسیر حرکت برای یک یا چند ربات لغزان است که بتوانند دزدها را ببینند. ما این مسئله را درون یک چندضلعی متعامد و x -یکنوا و همچنین یک چندضلعی متعامد ساده بررسی کرده و الگوریتم‌های چندجمله‌ای برای نگهداری کردن آن‌ها ارائه دادیم.

کلمات کلیدی: ۱- هندسه محاسباتی، ۲- مسئله‌ی موزه‌ی هنر، ۳- مسئله‌ی طراحی مسیر حرکت،

۴- k -فرستنده، ۵- پوشش.

فهرست مطالب

۱۱	۱	مقدمه
۱۱	۱.۱	مسئله‌ی موزه‌ی هنر
۱۴	۲.۱	مدل k -فرستنده
۱۴	۳.۱	مدل دوربین‌های لغزان
۱۸	۴.۱	مسئله‌ی دزد و پلیس
۱۹	۵.۱	ساختار گزارش
۲۱	۲	تاریخچه
۲۱	۱.۲	موزه‌ی هنر
۲۳	۲.۲	دوربین‌های لغزان
۲۴	۳.۲	پوشش موزه‌های هنر متعامد توسط کمترین تعداد دوربین‌های لغزان
۲۵	۴.۲	پوشش موزه‌های هنر متعامد توسط کمترین مجموع طول دوربین‌های لغزان
۲۷	۵.۲	تمیز کردن چندضلعی‌ها یا مسئله‌ی دزد و پلیس
۲۸	۳	پوشش موزه‌های هنر متعامد توسط کمترین مجموع طول k -فرستنده‌های لغزان
۲۹	۱.۳	مفاهیم مقدماتی
۲۹	۱.۱.۳	مسئله‌ی پوشش رأسی کمینه
۳۰	۲.۱.۳	برنامه‌ریزی خطی
۳۱	۳.۱.۳	برنامه‌ریزی تفاضلی
۳۲	۴.۱.۳	مسئله‌ی کاشی‌کاری

۳۴	تعریف مسئله	۲.۳
۳۶	ان پی-سخت بودن مسئله	۳.۳
۴۰	الگوریتم ۲-تقریب	۴.۳
۴۵	۱.۴.۳ تعمیم مسئله	
۴۸	نتیجه‌گیری	۵.۳
۴۹	۴ پوشش موزه‌های هنر متعامد توسط کمترین تعداد ۲-فرستنده‌های لغزان	
۵۰	۱.۴ مفاهیم مقدماتی	
۵۰	۱.۱.۴ مسئله برخورد با بازه‌ها	
۵۱	تعریف مسئله	۲.۴
۵۲	پوشش چندضلعی توسط ۲-فرستنده‌های عمودی	۳.۴
۵۹	الگوریتم برای حل مسئله ۲ - <i>MCSC</i>	۴.۴
۶۱	۱.۴.۴ الگوریتم	
۶۲	تحلیل ضریب تقریب الگوریتم	۲.۴.۴
۶۵	تحلیل پیچیدگی زمانی الگوریتم	۳.۴.۴
۶۷	نتیجه‌گیری	۵.۴
۶۹	۵ تمیز کردن چندضلعی‌های متعامد توسط ربات‌های لغزان	
۶۹	تعریف مسئله	۱.۵
۷۰	۱.۱.۵ کارهای انجام شده	
۷۱	تمیز کردن یک موزهی هنر متعامد توسط ربات‌های لغزان	۲.۵
۷۱	تعریف مسئله	۱.۲.۵
۷۲	الگوریتم ارائه شده	۲.۲.۵
۷۴	تحلیل الگوریتم	۳.۲.۵
۷۷	پیاده‌سازی الگوریتم	۴.۲.۵

۳.۵	تمیز کردن یک موزهی هنر متعامد و x -یکنوا توسط کمترین تعداد ربات‌های لغزان	۷۹
۱.۳.۵	تعریف مسئله	۷۹
۲.۳.۵	الگوریتم ارائه شده	۷۹
۳.۳.۵	تحلیل زمان و درستی الگوریتم	۸۲
۴.۵	نتیجه‌گیری	۸۴
۶ نتیجه‌گیری و کارهای آینده		
۱.۶	نگهبانی موزهی هنر متعامد توسط دوربین‌های لغزان	۸۶
۲.۶	پوشش موزه‌های هنر متعامد توسط k -فرستنده‌های لغزان	۸۶
۳.۶	تمیز کردن موزه‌های هنر توسط دوربین‌های لغزان	۸۷

فهرست شکل‌ها

- ۱.۱ نمونه ای از یک چندضلعی با ۱۵ یال که برای نگهبانی کردن آن باید حداقل ۵ نگهبان نقطه‌ای قرار داد. ۱۲
- ۲.۱ ناحیه‌ی دید یک نگهبان در مدل ۲- فرستنده ۱۴
- ۳.۱ دوربین لغزانی که در راستای s حرکت می‌کند نقاط a و c را می‌بیند و نقاط e و d را نمی‌بیند. چندضلعی رویت‌پذیر s به رنگ خاکستری نشان داده شده است. ۱۵
- ۴.۱ نمونه ای از چندضلعی‌های ساده و دارای حفره ۱۶
- ۵.۱ مقایسه ناحیه‌ی دید یک دوربین لغزان در سه حالت (الف) ۰- فرستنده ، (ب) ۲- فرستنده و (ج) ۴- فرستنده ۱۶
- ۶.۱ در شکل (الف) جواب بهینه مسئله کمترین تعداد دوربین‌های لغزان و در شکل (ب) جواب بهینه مسئله‌ی کمترین مجموع طول دوربین‌های لغزان برای یک چندضلعی نشان داده شده است. ۱۷
- ۱.۲ افراز چندضلعی متعامد به نواحی مستطیل شکل ۲۴
- ۲.۲ یک نمونه چندضلعی متعامد و گراف بدست آمده از روی آن [۲۱] ۲۶
- ۱.۳ یک ناحیه در مسئله‌ی کاشی‌کاری ۳۲
- ۲.۳ کاشی‌کاری یک ناحیه توسط تعدادی کاشی داده شده ۳۳
- ۳.۳ طریقه ساختن P از روی R ۳۶
- ۴.۳ قسمت‌هایی که برای اجتناب از داشتن (الف) فرستنده‌های با طول بیشتر از $1 + \epsilon$ و (ب) فرستنده‌هایی که دو قسمت ناهمبند از چندضلعی را پوشش دهند، اضافه می‌شوند. ۳۷

- ۵.۳ چندضلعی ورودی مسئله‌ی کاشی‌کاری (R) و چندضلعی ساخته شده از روی آن (P) . . . ۳۸
- ۶.۳ محل قرار دادن یک فرستنده برای هر کاشی ۳۹
- ۷.۳ افراز چندضلعی به مستطیل‌ها در الگوریتم تقریبی ۴۱
- ۸.۳ تقسیم مسئله به مسئله‌های کوچکتر در الگوریتم تقریبی ۴۴
- ۱.۴ تعدادی بازه روی محور اعداد حقیقی به همراه کمترین تعداد خط عمودی برخورد کننده با آن‌ها نشان داده شده‌اند. ۵۰
- ۲.۴ اعضای هر ۲-بازه یه سورت متمایز نشان داده شده‌اند. برای پوشش این ۲-بازه‌ها به حداقل ۲ خط عمودی نیاز است. ۵۱
- ۳.۴ مستطیل‌بندی $S(P)$ و $S'(P)$ متناظر با چندضلعی P نشان داده شده‌اند. ۵۱
- ۴.۴ پاره‌خط‌های H با رنگ قرمز نشان داده شده‌اند. ۵۲
- ۵.۴ مجموعه I متناظر با $S'(P)$ به صورت خط‌چین مشخص شده است [۷]. ۵۳
- ۶.۴ شکل الف: مجموعه $I(h_i)$ متناظر با فرستنده h_i و شکل ب: اعضای از مجموعه Q که بخاطر h_i به این مجموعه اضافه شده‌اند [۷]. ۵۴
- ۷.۴ در این شکل اعضای q_i با خط و اعضای متناظر آن‌ها در Q' به صورت خط‌چین مشخص شده است [۷]. ۵۴
- ۸.۴ رابطه بین جواب مسئله‌ی پوشش ۳-بازه و مسئله‌ی پوشش چندضلعی متعامد و x - یکنوا توسط فرستنده‌های فقط عمودی نشان داده شده است [۷]. ۵۵
- ۹.۴ در این شکل ناحیه s با رنگ خاکستری و پاره‌خط متناظر با این ناحیه با حرف z مشخص شده است. همان‌طور که در این تصویر نیز مشاهده می‌شود در صورتی که x_i با تصویر z بر محور x ها برخورد نداشته باشد، باز هم فرستنده متناظر با آن (c_t) حتما توانایی دیدن ناحیه s را خواهد داشت [۷]. ۵۶
- ۱۰.۴ با توجه به قابل قبول بودن مجموعه C برای پوشش چندضلعی، حتما امتداد فرستنده‌ای که ناحیه s_2 را پوشش می‌دهد، با q_i برخورد خواهد داشت [۷]. ۵۸

۱۱.۴	با توجه به قابل قبول بودن مجموعه C ، حتما فرستنده‌ای در این مجموعه قرار دارد که درون
۵۸	یکی از s_1 یا s_2 باشد [۷].
۱۲.۴	در این شکل مجموعه δ متناظر با فرستنده افقی c_i به صورت خطچین به نشان داده شده
۶۰	است [۷].
۱۳.۴	مجموعه‌ی C_H قبل و بعد از خلاصه شدن [۷].
۶۰	در این شکل c_i و c_{i+1} به صورت خطچین و مجموعه t_i متناظر با این دو با خطوط کامل
۶۲	مشخص شده است.
۱۵.۴	نمونه‌ای از یک چندضلعی که دارای جواب بهینه ۱ است اما الگوریتم ما سه فرستنده را
۶۲	گزارش می‌دهد.
۱۶.۴	الگوریتم دو فرستنده افقی را به جای یک فرستنده عمودی گزارش می‌کند.
۶۴	
۱.۵	دو پنجره رأس v_j به صورت خطچین نشان داده شده‌اند.
۷۰	
۲.۵	ریات‌های r_1, r_2 و r_3 به ترتیب در امتداد s_1, s_2 و s_3 حرکت می‌کنند.
۷۳	
۳.۵	برای تمیز کردن چندضلعی، ربات s_1 و یا s_2 باید $O(n)$ مرتبه صدا زده شوند.
۷۷	
۴.۵	مسیر ربات‌ها، نقطه‌ی ابتدایی و نقطه‌ی انتهایی هر گام با فلش‌های نقطه‌چین، ضربدر و
۷۸	دایره به ترتیب نشان داده شده‌اند.
۵.۵	روی r_i حرکت می‌کند.
۸۱	
۶.۵	روی r_i حرکت می‌کند.
۸۲	

فصل ۱

مقدمه

۱.۱ مسئله‌ی موزه‌ی هنر

مسئله‌ی موزه‌ی هنر، یکی از مسائل کلاسیک در حوزه‌ی هندسه‌ی محاسباتی است. هدف در این مسئله تعیین تعداد نگهبانانی است که برای دیدن تمام نقاط داخل یک اتاق موزه‌ی هنر کافی می‌باشند. به عبارت دیگر هر نقطه داخلی اتاق باید توسط حداقل یکی از نگهبانان دیده شود. دیدپذیری^۱ یک پدیده طبیعی و مفهوم اساسی در زندگی روزمره است. مسئله دیدپذیری یکی از مسائل پرکاربرد و گسترده در زمینه‌های مختلف علوم کامپیوتر از جمله رباتیک^۲، بینایی کامپیوتر^۳ و گرافیک کامپیوتری^۴ است. مطالعه دیدپذیری در سال ۱۹۱۳ زمانی که برن^۵ [۱۰] قضیه‌ای در خصوص هسته^۶ یک مجموعه ثابت کرد، شروع شد. پس از آن الگوریتم‌های دیدپذیری از اواخر دهه ۱۹۷۰ وارد حوزه هندسه محاسباتی شدند.

مسئله‌ی موزه‌ی هنر، تعیین تعداد نگهبانان مورد نیاز برای دیدن تمام نقاط داخل یک اتاق موزه‌ی هنر است. به عبارت دیگر هر نقطه داخلی اتاق باید توسط حداقل یکی از نگهبانان دیده شود. در طول

^۱Visibility

^۲Robotics

^۳Computer Vision

^۴Computer Graphics

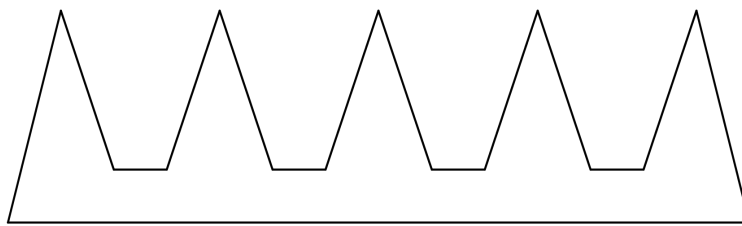
^۵Brunn

^۶Kernel

سال‌ها حالت‌های مختلفی از مسئله‌ی موزه‌ی هنر مورد بررسی قرار گرفته‌اند. تفاوت این مسائل در محیط‌های متفاوت، انواع متفاوت نگهبانان مورد استفاده قرار گرفته و همچنین انواع متفاوت تعریف دیدپذیری است. بسیاری از این مسائل *ان‌پی*-سخت هستند، از جمله حالتی از مسئله که محیط مورد بررسی یک چندضلعی متعامد ساده است و نگهبانان باید روی رأس‌ها قرار گیرند.

فرض می‌کنیم مساحت موزه به وسیله چندضلعی P مشخص شده است. یال‌های این چندضلعی یکدیگر را قطع نمی‌کنند اما این چندضلعی می‌تواند دارای حفره^۷ یا مانع باشد. ناحیه‌ی دید هر نگهبان در نقطه p را برابر با تمام نقاط q تعریف می‌کنیم که پاره‌خط واصل p و q به طور کامل درون P قرار دارد. هدف مسئله قرار دادن کمترین تعداد نگهبان درون P است به طوری که اجتماع ناحیه‌ی دید آن‌ها برابر با کل مساحت P است.

با مطرح شدن مسئله‌ی موزه‌ی هنر اولین سوالی که به ذهن می‌رسد این است که حداقل تعداد نگهبان‌هایی را که با آن‌ها بتوان همه موزه‌هایی که دارای n یال هستند نگهبانی کرد چقدر است. برای این سوال الگوریتم‌هایی مطرح شده است که نشان می‌دهد هر موزه‌ی هنر را می‌توان با $\lfloor \frac{n}{3} \rfloor$ نگهبان، نگهبانی کرد [۵۳]. شکل ۱.۱ نشان می‌دهد که این مقدار برای برخی از چندضلعی‌ها لازم است.



شکل ۱.۱: نمونه‌ای از یک چندضلعی با ۱۵ یال که برای نگهبانی کردن آن باید حداقل ۵ نگهبان نقطه‌ای قرار داد.

برخی از مسائل موزه‌ی هنر را که در آن‌ها دید نگهبان محدود است، می‌توان در زمان چندجمله‌ای حل کرد. موتوانی و همکارانش^۸ [۳۷] برای حل مسئله‌ی موزه‌ی هنر با دید محدود ”روش گراف

^۷Hole

^۸Motwani et al.

کامل^۹ را ارائه دادند. در این روش یک گراف G به چندضلعی P اختصاص داده شده و مسئله‌ی مورد نظر روی P به یک مسئله روی گراف تبدیل می‌شود. آن‌ها با کمک این روش مسئله‌ی موزه‌ی هنر در مدل s -دیدپذیری^{۱۰} را حل کردند. در این مدل یک نگهبان قرار گرفته در یک نقطه p (درون چندضلعی متعامد P) می‌تواند تمام نقاطی را که توسط یک مسیر پلکانی متعامد (مسیر کاملاً درون P است) به p وصل می‌شوند، ببیند. این روش قبلاً توسط ورمن و کیل^{۱۱} [۵۲] برای حل مسئله‌ی مشابهی در مدل r -دیدپذیری^{۱۲} استفاده شده است. در این مدل یک نگهبان قرار گرفته در یک نقطه p تمام نقاطی از P مانند q را، که مستطیل موازی محورهای مختصات با قطر \overline{pq} کاملاً درون P قرار دارد، می‌بیند.

از مسائل پایه‌ای دیگری که در این زمینه مطرح شده است، می‌توان به بررسی پیچیدگی محاسباتی این مسئله اشاره کرد. نشان داده شده است که این مسئله در حالت تصمیم‌گیری یک مسئله NP -سخت^{۱۳} است. به عبارت دیگر، در صورتی که الگوریتمی چندجمله‌ای طراحی شود که تصمیم بگیرد آیا می‌توان موزه P را با k نگهبان نگهبانی کرد یا نه، پی^{۱۴} برابر با NP خواهد بود. همچنین در ادامه نشان داده شده است که این مسئله تقریب-سخت^{۱۶} نیز هست به عبارت دیگر نمی‌توان برای این مسئله الگوریتم چندجمله‌ای تقریبی با ضریب تقریب ثابت دلخواه بیشتر از ϵ پیدا کرد مگر این که پی با NP برابر باشد.

^۹Perfect Graph Approach

^{۱۰} s -visibility

^{۱۱}Worman and Keil

^{۱۲} r -visibility

^{۱۳} NP -Hard

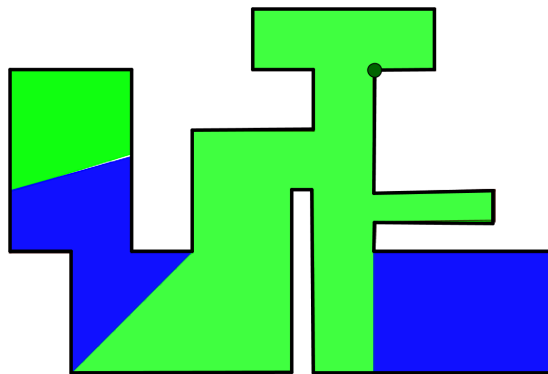
^{۱۴} P

^{۱۵} NP

^{۱۶} APX -Hard

۲.۱ مدل k -فرستنده

در این قسمت به بررسی نسخه‌ی خاصی از مسئله‌ی موزه‌ی هنر می‌پردازیم که در آن ناحیه‌ی دید هر نگهبان افزایش پیدا کرده است. در این مدل فرض بر این است که هر نگهبان می‌تواند نواحی پشت حداکثر k دیوار را نگهبانی کند [۳]. به طور مثال در شکل ۲.۱ همه نقاط سبز نواحی دید نگهبان را در مدل ۲-فرستنده نشان می‌دهند. نقاط آبی نقاطی هستند که داخل ناحیه‌ی دید این نگهبان قرار نمی‌گیرند. با قرار دادن $k = 0$ این مسئله تبدیل به مسئله‌ی معروف موزه‌ی هنر می‌شود، پس بدیهی است که این مسئله نیز یک مسئله‌ی ان‌پی-سخت است.

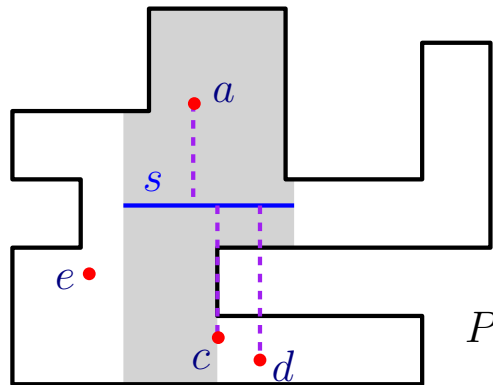


شکل ۲.۱: ناحیه‌ی دید یک نگهبان در مدل ۲-فرستنده

۳.۱ مدل دوربین‌های لغزان

در این قسمت، مدل دوربین‌های لغزان را که برای مسئله‌ی نگهبانی موزه ارائه شده است معرفی می‌کنیم. در این مدل معمولاً فرض بر این است که مساحت موزه به صورت یک چندضلعی متعامد^{۱۷} که می‌تواند دارای حفره نیز باشد نشان داده می‌شود. در این رساله فرض می‌کنیم نگهبان‌ها به صورت دوربین‌های لغزان مدل شده‌اند که هر یک روی یک پاره‌خط حرکت می‌کند. به این صورت که هر دوربین ناحیه‌ای از صفحه را می‌تواند نگهبانی کند که بتوان از آن نقاط پاره‌خطی عمود به پاره‌خط دوربین رسم کرد که به طور کامل درون چندضلعی قرار داشته باشد. چندضلعی رویت‌پذیر دوربین لغزانی که در راستای

^{۱۷}Orthogonal



شکل ۳.۱: دوربین لغزانی که در راستای s حرکت می‌کند نقاط a و c را می‌بیند و نقاط d و e را نمی‌بیند. چندضلعی رویت‌پذیر s به رنگ خاکستری نشان داده شده است.

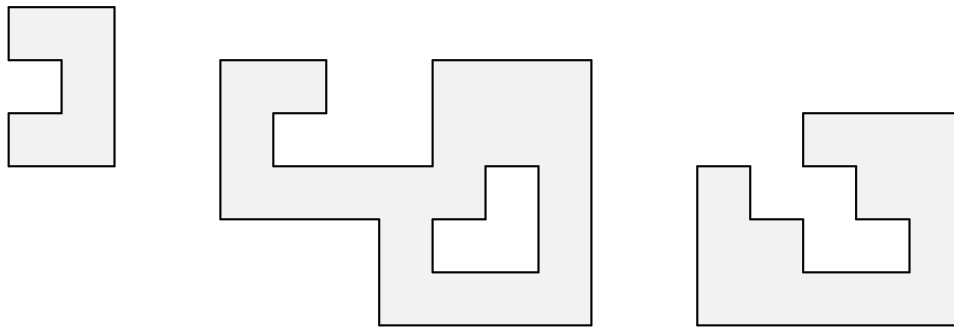
یک پاره‌خط افقی یا عمودی مانند s حرکت می‌کند شامل تمام نقاطی از P است که پاره‌خط عمود از آن نقاط بر s به طور کامل درون P قرار داشته باشد (شکل ۵.۱ را ملاحظه کنید). در صورتی که دوربین‌های ما k -فرستنده در نظر گرفته شوند، این پاره‌خط می‌تواند در k نقطه با یال‌های چندضلعی برخورد داشته باشد. فرض ما بر این است که دوربین‌ها می‌توانند تنها به صورت افقی یا عمودی در چندضلعی حرکت کنند.

حالت‌های مختلف مسئله

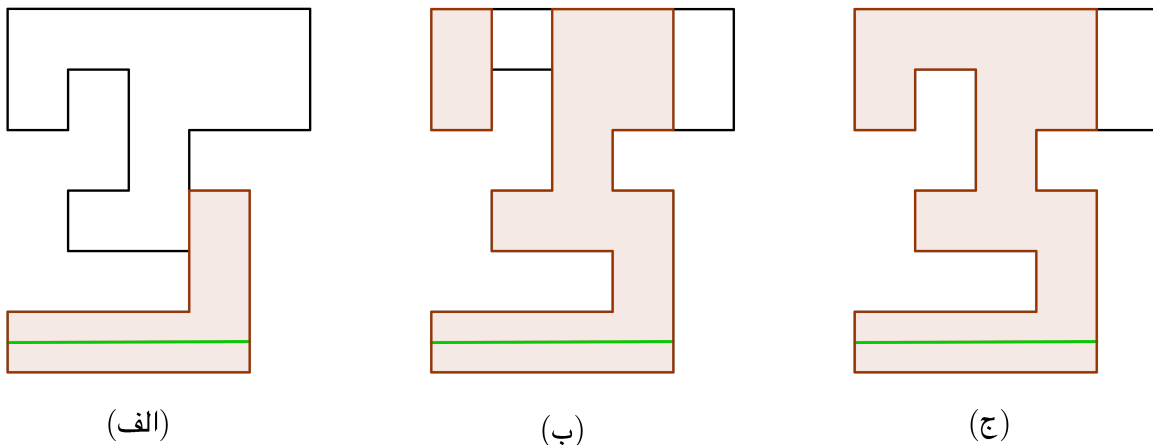
حالت‌های متفاوت مسئله براساس نوع چندضلعی، قدرت نفوذ دوربین‌ها و هدف مورد نظر، دسته‌بندی می‌شوند.

۱. **نوع چندضلعی:** چندضلعی می‌تواند دارای مانع یا بدون مانع باشد. هرچند در نگاه اول به نظر می‌رسد دارا بودن یا نبودن مانع درون چندضلعی تفاوت زیادی بر پیچیدگی مسئله ندارد، اما در ادامه شرح می‌دهیم که سختی مسئله به میزان زیادی به این عامل بستگی دارد. در شکل ۴.۱ چندضلعی وسط دارای حفره است اما دو چندضلعی دیگر بدون حفره هستند.

۲. **قدرت دوربین‌ها:** قدرت دوربین‌ها نشان می‌دهد که دوربین‌ها توانایی نفوذ تا حداکثر چند دیوار را دارند. در مدل ساده فرض بر این است که این قدرت برابر با صفر است. به عبارت



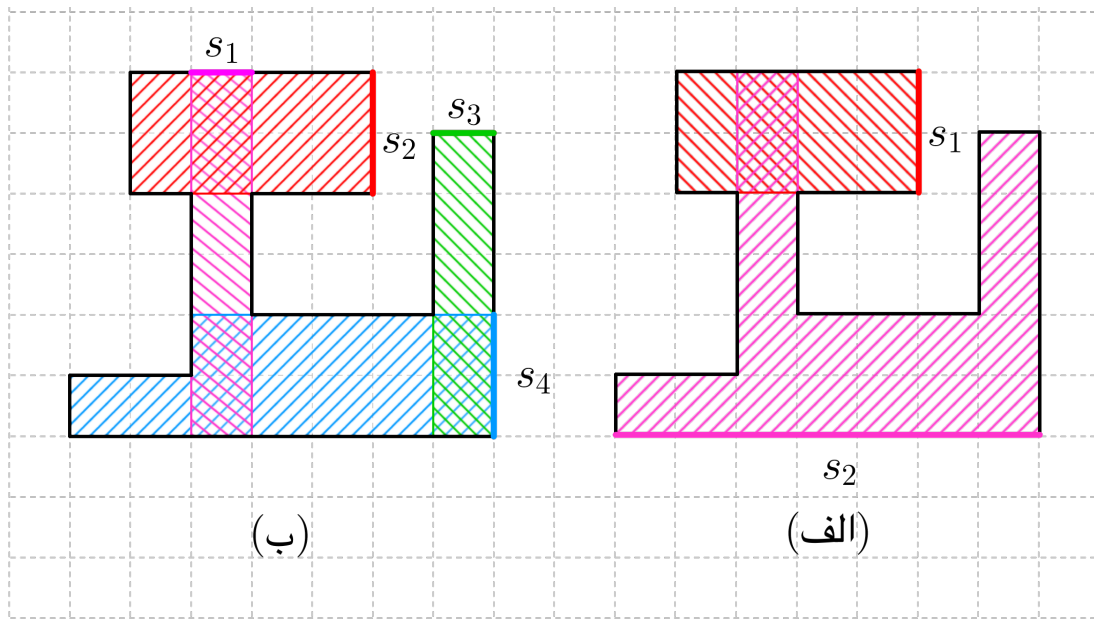
شکل ۴.۱: نمونه ای از چندضلعی‌های ساده و دارای حفره



شکل ۵.۱: مقایسه ناحیه‌ی دید یک دوربین لغزان در سه حالت (الف) ۰- فرستنده ، (ب) ۲- فرستنده و (ج) ۴- فرستنده

دیگر، ناحیه‌ی دید دوربین‌ها به اولین دیواری که جلوی آن‌ها قرار می‌گیرد محدود می‌شود. این حالت ۰- فرستنده نامیده می‌شود. در حالت پیچیده‌تر می‌توان فرض کرد که دوربین‌ها قادر هستند ناحیه پشت حداکثر k دیوار را مشاهده کنند. بدیهی است که چون هدف نگهبانی مساحت چندضلعی ساده است و دوربین‌ها درون چندضلعی قرار دارند، حالت $2k$ - فرستنده و حالت $(2k + 1)$ - فرستنده تفاوتی ندارند. در شکل ۵.۱ ناحیه‌ی دید یک دوربین افقی در سه حالت ۰- فرستنده ، ۲- فرستنده و ۴- فرستنده نشان داده شده است.

۳. **هدف مسئله:** هدف مسئله می‌تواند به دو صورت تعریف شود. در حالت اول هدف کمینه کردن تعداد دوربین‌ها و در حالت دوم هدف کمینه کردن مجموع طول پاره‌خط‌هایی است که



شکل ۶.۱: در شکل (الف) جواب بهینه مسئله کمترین تعداد دوربین‌های لغزان و در شکل (ب) جواب بهینه مسئله کمترین مجموع طول دوربین‌های لغزان برای یک چندضلعی نشان داده شده است.

دوربین‌ها روی آن‌ها حرکت می‌کنند. همان‌گونه که در شکل ۶.۱ قابل ملاحظه است، بدیهی است که در حالتی که طول دوربین‌ها کمینه است لزوماً تعداد دوربین‌ها کمینه نیست و برعکس.

ما در این رساله مدل خاصی از این مسئله را مورد بررسی قرار می‌دهیم. در این مدل به جای نگهداری از دوربین لغزان استفاده می‌کنیم و هدف مسئله این است که طوری دوربین‌ها را درون موزه قرار دهیم که علاوه بر نگهداری کردن کل مساحت موزه هزینه قرار دادن دوربین‌ها هم کمینه شود. این هزینه می‌تواند برابر با تعداد دوربین‌ها یا مجموع طول پیموده شده توسط دوربین‌ها باشد. همچنین مساحتی که هر دوربین می‌تواند نگهداری کند با توجه به نوع دوربین متغیر است.

با توجه به این که مسئله نگهداری موزه هنر دارای تاریخچه طولانی است و کاربردهای زیادی در صنعت دارد، این ایده که نگهداری می‌تواند با استفاده از امواج رادیویی نواحی موزه را نگهداری کنند دور از ذهن نیست. این مسئله اولین بار توسط فابیل^{۱۸} مطرح شد [۴۵]. در مدل آن‌ها هر نگهدارنده به صورت یک گیرنده امواج رادیویی در نظر گرفته شده، که امواج آن‌ها می‌تواند از حداکثر k دیوار عبور کند. آن‌ها با استفاده از مدل پیشنهادی خود حد پایین و بالا را برای تعیین تعداد نگهدارنده‌هایی

^{۱۸}Fabila

که برای نگهبانی موزه با n یال لازم است، به دست آوردند.

۴.۱ مسئله دزد و پلیس

در این قسمت به معرفی مسئله دزد و پلیس^{۱۹} می‌پردازیم. این مفهوم اولین بار توسط پارسون [۴۶] مطرح شد. در این مسئله هدف، یافتن تمام دزدهایی است که درون یک چندضلعی قرار دارند. لاوال و همکارانش^{۲۰} [۳۶] در سال ۱۹۹۷ مسئله طراحی مسیر حرکت^{۲۱} را براساس مفهوم رویت‌پذیری تعریف کردند. هدف در این مسئله یافتن مسیر حرکت برای یک یا تعداد بیشتری ربات (پلیس) است که دارای سنسورهای همه‌جانبه^{۲۲} برای دیدن یک دزد هستند، به طوری که دزد غیرقابل پیش‌بینی بوده و دارای مکان شروع نامعلوم است و می‌تواند با سرعت دلخواه زیاد حرکت کند. روند پیدا کردن تمام دزدها را تمیز کردن^{۲۳} چندضلعی نیز می‌نامند. مسئله دزد و پلیس دارای کاربردهای زیادی از جمله کنترل ترافیک، استراتژی نظامی و پیگیری مسیر^{۲۴} [۳۶] است.

ما در این رساله براساس مفهوم دوربین‌های لغزان و مسئله تمیز کردن یک چندضلعی، یک مسئله جدید را مطرح کردیم. فرض کنید یک چندضلعی متعامد ساده که دارای تعدادی دزد است به عنوان ورودی داده شده است و هدف طراحی مسیر حرکت برای تعدادی ربات لغزان است که بتوانند تمام دزدها را پیدا کنند. یک ربات لغزان مانند یک دوربین لغزان است به طوری که ربات در راستای یک پاره‌خط حرکت می‌کند و در هر لحظه تمام نقاط در راستای عمود بر مسیر حرکتش را می‌بیند. دلیل مهم برای معرفی این مسئله این است که برخلاف تعریف دوربین لغزان، در تمام مسائل حل شده با این مفهوم [۳۲، ۲۱، ۱۱] فرض شده است که یک دوربین لغزان تمام نقاط درون چندضلعی رویت‌پذیرش را به طور همزمان می‌بیند (که یک تناقض است). از آنجایی که هر دوربین لغزان در

^{۱۹}Pursuit-Evasion

^{۲۰}Lavalle et al.

^{۲۱}Motion-Path-Planning

^{۲۲}Omnidirectional

^{۲۳}Clearing

^{۲۴}Trajectory Tracking

هر لحظه نقاط در راستای عمود بر مسیر حرکتش را می‌بیند و نمی‌تواند به طور همزمان تمام نقاط چندضلعی رویت‌پذیر مسیر حرکتش را ببیند، ما این مفهومی مهم را در نظر گرفته و دو مسئله در این زمینه را در فصل ۵ مطرح کرده و آن‌ها را حل می‌کنیم.

۵.۱ ساختار گزارش

در فصل ۱ به معرفی مسئله‌ی نگهبانی موزه‌ی هنر پرداخته و سپس مسئله‌های مورد نظر در این رساله را تعریف کردیم. در فصل ۲ برخی از نتایج بدست آمده را برای مسئله‌ی موزه‌ی هنر در مدل دوربین‌های لغزان مطرح می‌کنیم. در فصل ۳ مسئله‌ی پوشش چندضلعی‌های متعامد با استفاده از کمترین مجموع طول دوربین‌های لغزان را در حالت k -فرستنده بررسی می‌کنیم و نشان می‌دهیم که این مسئله برخلاف حالت ۰-فرستنده یک مسئله ان‌پی-سخت است. سپس یک الگوریتم با ضریب تقریب ۲ برای این مسئله ارائه می‌دهیم. نتیجه بدست آمده از این فصل در کنفرانس Canadian Conference on Computational Geometry [۴۰] و مجله Theoretical Computer Science [۴۱] به چاپ رسیده‌اند.

- Salma Sadat Mahdavi, Saeed Seddighin and Mohammad Ghodsi *Covering Orthogonal Polygons with Sliding k -Transmitters*, In the Proceedings of the 26th Canadian Conference on Computational Geometry, 2014
- Salma Sadat Mahdavi, Saeed Seddighin and Mohammad Ghodsi, *Covering Orthogonal Polygons with Sliding k -Transmitters*, Theoretical Computer Science, 85: 163–181, 2020.

در فصل ۴ مسئله‌ی پوشش موزه‌های هنر متعامد و x -یکنوا توسط کمترین تعداد دوربین‌های لغزان در حالت ۲-فرستنده را مطرح کرده و یک الگوریتم تقریبی با ضریب تقریب ۶ برای حل این مسئله ارائه می‌دهیم که دارای مثال محکم^{۲۵} است.

^{۲۵}Tigh

در فصل ۵ مسئله‌ی تمیز کردن چندضلعی‌های متعامد توسط ربات‌های لغزان را بررسی می‌کنیم. این مسئله را در دو حالت چندضلعی متعامد و x -یکنوا و همچنین چندضلعی‌های متعامد بررسی کرده و الگوریتم‌های چند جمله‌ای برای حل آن‌ها ارائه می‌دهیم. یک نتیجه‌ی بدست آمده از این فصل در مجله Theoretical Computer Science [۳۸] به چاپ رسیده و یک نتیجه‌ی دیگر به Information Processing Letters [۴۰] ارسال شده‌است.

- Salma Sadat Mahdavi and Mohammad Ghodsi, *Clearing an Orthogonal Polygon to Find the Evaders*, Theoretical Computer Science, 2020.
- Salma Sadat Mahdavi and Mohammad Ghodsi. *Clearing an x -monotone Orthogonal Polygon to Find Evaders*, Submitted to Information Processing Letters, December 2018.

در فصل ۶ مطالب ارائه شده را جمع‌بندی کرده و مسائل پیش‌رو را مطرح می‌کنیم. لازم به ذکر است کار انجام شده در فصل ۳ این رساله کاری مشترک با آقای سعید صدیقین و همچنین کار ارائه شده در فصل ۴ کاری مشترک با مرحوم آقای محمدامین بیروتی است. در نگارش این رساله پایان‌نامه‌های کارشناسی ارشدشان [۴۸، ۷] نیز مورد استفاده قرار گرفته است.

فصل ۲

تاریخچه

در این فصل به بررسی تاریخچه و کارهای انجام شده قبلی، در حوزه‌ی نگهداری موزه‌های هنر متعامد توسط دوربین‌های لگزان و همچنین مسئله‌ی تمیز کردن چندضلعی‌ها، می‌پردازیم.

۱.۲ موزه‌ی هنر

در سال ۱۹۶۹ کلی^۱ [۲۳] این سوال را مطرح کرد: برای محافظت از نقاشی‌ها و کارهای هنری درون یک موزه‌ی هنر که دارای n دیوار است، چه تعداد نگهبان لازم و چه تعداد نگهبان کافی هستند؟ این سوال جالب و ساده هندسه ترکیباتی منجر به نوشتن مقالات زیبا و یک کتاب شد، که اکثر آنها در ۲۵ سال گذشته نوشته شده‌اند. تجربه نشان داده هنگامی که به دنبال یک مکان مناسب برای قرار دادن یک فرستنده بی‌سیم در یک ساختمان هستیم، عامل مهم که باعث ضعیف شدن سیگنال یک فرستنده می‌شود، تعداد دیوارهایی است که سیگنال باید از آنها عبور کند. یک فرستنده بی‌سیم را یک k -فرستنده می‌نامیم اگر قادر به عبور دادن سیگنال از k دیوار ساختمان بوده و سیگنال پس از عبور از دیوارها به اندازه کافی قوی باشد. حالت جامع‌تر سوال کلی می‌تواند به این صورت مطرح شود که: چند k -فرستنده لازم و چند تا کافی هستند تا بتوانند فضای داخلی یک موزه‌ی هنر با n دیوار را پوشش دهند.

^۱Klee

برای $k = 0$ ، مسئله ۰ - فرستنده برای چندضلعی‌های ساده توسط قضیه‌ی موزه‌ی هنر حل شده است. این قضیه ثابت می‌کند که $\lfloor \frac{n}{3} \rfloor$ نگهبان کافی و برخی موارد لازم است تا بتوان یک چندضلعی با n رأس را پوشش داد. یافتن کمترین تعداد ۰ - فرستنده که بتوانند یک چندضلعی را پوشش دهند این‌پی - سخت است [۳۵، ۴۴]. برای $k > 0$ ، آیکهولزر و همکارانش^۲ [۱] و همچنین فابیلای مونروی و همکارانش^۳ [۲۳] مسئله k - فرستنده را زمانی که ناحیه به صورت یک چندضلعی یکنوا یا یک چندضلعی متعامد یکنوا با n رأس است، بررسی کردند. آنها نشان دادند که $(\frac{n}{2k})$ تا k - فرستنده کافی و $(\lceil \frac{n}{2k+4} \rceil)$ تا k - فرستنده بعضی مواقع لازمند تا یک چندضلعی یکنوا پوشش داده شود. آنها همچنین چندضلعی‌های ساده، چندضلعی‌های متعامد و چیدمانی از خطوط برای فرستنده‌های خیلی قدرتمند، k - فرستنده‌ی که براساس یک تابع از n ممکن است افزایش یابد، را بررسی کردند. به طور مثال، نشان دادند هر چندضلعی ساده با n رأس همیشه می‌تواند توسط یک فرستنده با قدرت $\lceil \frac{2n+1}{3} \rceil$ پوشیده شود و این کران تا یک ثابت افزودنی بسته است. برای چندضلعی‌های متعامد یک $\lfloor \frac{n}{3} \rfloor$ - فرستنده برای پوشش تمام چندضلعی کافی است. مسئله‌ی پوشش صفحه توسط یک k - فرستنده در [۲۶] در نظر گرفته شده است. در این مقاله ثابت شده که یک مجموعه‌ای از n پاره‌خط هم اندازه و دو به دو مجزا در فضای اقلیدسی وجود دارد به طوری که، برای هر نقطه درون صفحه یک اشعه با حداقل $2n/3$ برخورد با پاره‌خط‌ها، وجود دارد. در حالیکه در [۱، ۲۳، ۲۶] تمرکز روی یافتن کمترین تعداد از فرستنده‌های با قدرت بالاست و در [۳] هدف یافتن فرستنده‌هایی با قدرت پایین است.

مفهوم دیدپذیری گذرنده از k پاره‌خط در کارهای دیگر نیز ظاهر شده است. دین و همکارانش^۴ [۱۵، ۲۴، ۲۷] دیدپذیری نوار عمودی^۵ را که در آن k - دیدپذیری از k پاره‌خط عبور می‌کند مطالعه کردند. همچنین آیکهولزر و همکارانش [۲] مفهوم دیگری به نام k - تحدب^۶ را معرفی کردند که در آن قطر ممکن است مرز چندضلعی را در حداکثر $2(k-1)$ مرتبه قطع کند. برای بررسی

^۲Aichholzer et al.^۳Fabila-monroy et al.^۴Dean et al.^۵Vertical Bar k -Visibility^۶ k -Convexity

جامع در زمینه مسئله‌ی موزه‌ی هنر و انواع مختلف آن می‌توانید به [۴۴، ۵۰] مراجعه کنید. همچنین برای نتایج روی مسئله‌ی محلی‌سازی بی‌سیم^۷ می‌توانید به [۲۲، ۱۳، ۱۲] مراجعه کنید، که در این مقالات در مورد یک مجموعه از ۰- فرستنده‌ها سوال مطرح می‌شود. این مجموعه نه تنها نیاز به پوشش یک ناحیه داده شده دارد بلکه دستگاه‌های ارتباطی تلفن همراه را فعال می‌سازد تا مشخص شود در داخل محدوده قرار دارند یا خارج آن.

۲.۲ دوربین‌های لغزان

مفهوم دوربین‌های لغزان، اولین بار در سال ۲۰۱۱ توسط کاتز و مورگنسترن^۸ معرفی شد [۳۲]. آن‌ها به بررسی مسئله‌ی نگهبانی چندضلعی‌های متعامد توسط کمترین تعداد دوربین‌های لغزان که آن‌را با $MCSC$ ^۹ نشان دادند، پرداختند. ابتدا فرض کردند که دوربین‌ها فقط می‌توانند به صورت افقی حرکت کنند و ثابت کردند این مسئله به طور دقیق قابل حل در زمان چندجمله‌ای است. سپس برای مسئله‌ی کلی‌تر زمانی که دوربین‌ها مجاز به حرکت افقی و عمودی باشند و همچنین چندضلعی متعامد x -یکنوا باشد، یک الگوریتم با ضریب تقریب ۲ ارائه دادند.

پس از آن در سال ۲۰۱۳ دروچر و مهرابی^{۱۰} ثابت کردند زمانی که یک چندضلعی متعامد دارای حفره باشد، مسئله $MSCS$ این‌پی-سخت است [۲۱]. اما پیچیدگی این مسئله در حالتی که چندضلعی بدون مانع است تا به امروز نامشخص باقی مانده است. آن‌ها همچنین یک مسئله‌ی دیگر را در نظر گرفتند که در آن هدف نگهبانی چندضلعی متعامد توسط کمترین مجموع طول دوربین‌های لغزان است که آن‌را با $MLSC$ ^{۱۱} نشان دادند. به عبارت دیگر، مجموع طول پاره‌خط‌های متعامدی که دوربین‌ها روی آن‌ها حرکت می‌کنند، کمینه شود. آن‌ها نشان دادند یک الگوریتم قطعی برای مسئله $MLSC$ وجود دارد که پیچیدگی زمانی چندجمله‌ای دارد. یک سال بعد، یک الگوریتم تقریبی با ضریب

^۷Wireless Localization Problem

^۸Katz and Morgenstern

^۹Minimum Cardinality Sliding Camera

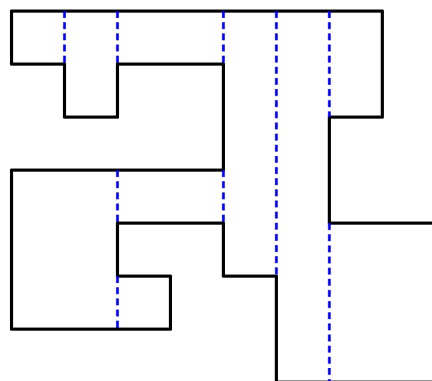
^{۱۰}Durocher and Mehrabi

^{۱۱}Minimum Length Sliding Camera

تقریب ۳,۵ برای مسئله $MSCS$ ارائه شد [۲۰]. در سال ۲۰۱۴ دیبرگ و همکارانش^{۱۲} [۱۱] یک الگوریتم خطی برای حل مسئله $MCSC$ درون یک چندضلعی متعامد و x -یکنوا ارائه دادند. بعد از آن مقالات دیگری برای حل مسئله‌های متفاوت توسط دوربین‌های لغزان ارائه شدند [۳۶][۹][۸].

۳.۲ پوشش موزه‌های هنر متعامد توسط کمترین تعداد دوربین‌های لغزان

در این قسمت مسئله‌ی پوشش موزه‌های هنر متعامد توسط کمترین تعداد دوربین‌های لغزان را به طور کلی شرح می‌دهیم [۳۲]. هدف در این مسئله قرار دادن کمترین تعداد دوربین‌های متعامد برای نگهداری کردن تمام موزه‌ی هنر (چندضلعی) است. در این مقاله ابتدا فرض کردند که نگهبانان همگی می‌توانند افقی باشند و یک الگوریتم چندجمله‌ای برای حل این مسئله ارائه دادند. سپس با فرض n -پی-سخت بودن مسئله $MCSC$ یک الگوریتم ۲-تقریب برای حالتی که چندضلعی متعامد و x -یکنوا است، ارائه دادند. روش استفاده شده در این مقاله برای حل مسئله $MCSC$ زمانی که تنها دوربین‌های افقی مجاز باشند، کاهش آن به مسئله‌ی پیدا کردن افزایش یک گراف وتری به کمترین تعداد خوشه است. فرض کنید P یک چندضلعی متعامد و ساده با n یال باشد. همان‌طور که در شکل



شکل ۱.۲: افزایش چندضلعی متعامد به نواحی مستطیل شکل

^{۱۲}de Berg et al.

۱.۲ ملاحظه می‌کنید، یک تقسیم‌بندی از P برای حل این مسئله ارائه دادند. یال‌های P و خط‌های اضافه شده، P را به تعدادی ناحیه‌های مستطیل شکل افراز می‌کنند. برای هر مستطیل یک رأس در گراف G قرار دادند. دو رأس v و u در G را با یک یال به هم وصل کردند اگر و تنها اگر بتوان یک دوربین افقی لغزان s درون P قرار داد که u و v درون ناحیه دیدپذیر s باشند. سپس ثابت کردند که جواب مسئله $MCSC$ زمانی که تنها دوربین‌های افقی مجاز باشند برابر است با افراز گراف متناظر G به کمترین تعداد خوشه. الگوریتم افراز یک گراف به کمترین خوشه، IN -پی-سخت است. اما در این مقاله نشان دادند که گراف G یک گراف وتری است.

گراف‌های وتری زیرمجموعه‌ای از گراف‌های تام^{۱۳} هستند. در این گراف‌ها عدد رنگی هر زیرمجموعه از رئوس با اندازه بزرگترین خوشه از آن زیرمجموعه برابر است. نشان داده شده است که می‌توان عدد رنگی، بزرگترین خوشه و افراز گراف به کمترین تعداد خوشه را برای این گراف‌ها در زمان خطی به دست آورد.

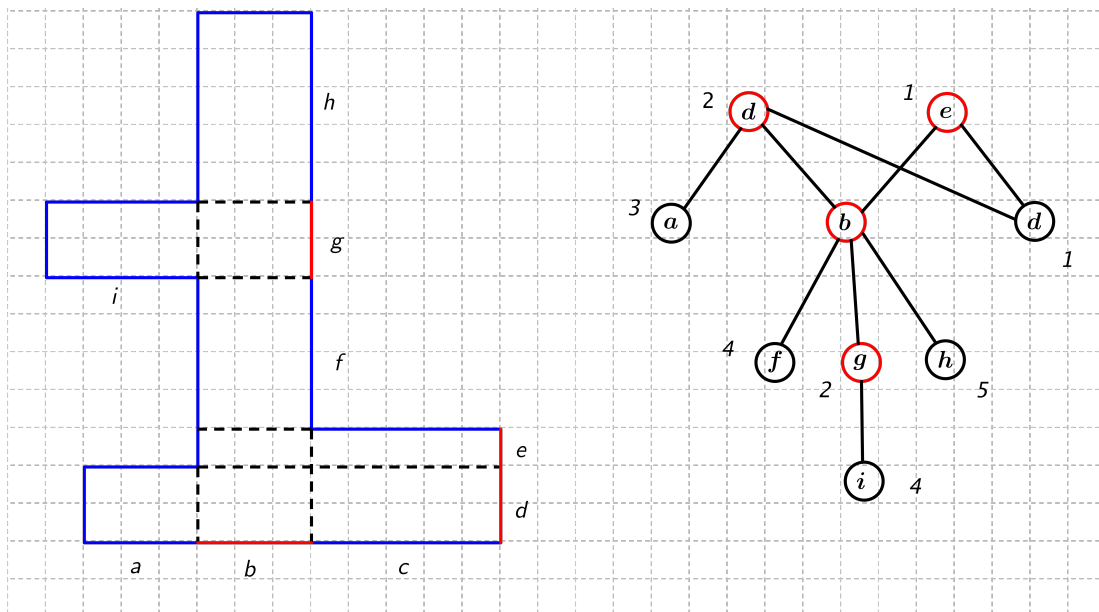
بنابراین مسئله‌ی مورد نظر قابل حل به صورت قطعی در زمان چندجمله‌ای است. برای مشاهده جزئیات اثبات می‌توانید به [۳۲] مراجعه کنید.

۴.۲ پوشش موزه‌های هنر متعامد توسط کمترین مجموع طول دوربین‌های لغزان

در این قسمت الگوریتم و نتایج سختی مسئله‌ی پوشش موزه‌های هنر متعامد با استفاده از دوربین‌های لغزان که توسط دروچر و مهربابی [۳۲] ارائه شده است را به طور کلی شرح می‌دهیم. هدف در این مقاله قرار دادن تعدادی دوربین لغزان درون P است به طوری که مجموع طول پاره‌خط‌هایی که روی آن‌ها حرکت می‌کنند کمینه شود. در این مقاله یک الگوریتم دقیق با زمان اجرای چندجمله‌ای برای مسئله $MLSC$ ارائه شده است.

^{۱۳}Perfect

در این مقاله نشان داده شده است که برای هر جواب بهینه این مسئله همیشه می‌توان یک حالتی را در نظر گرفت که دوربین‌ها روی یال‌های P قرار دارند. مانند [۳۲]، یال‌ها را امتداد دهید تا با برخورد با مرز P رأس‌های جدید بوجود آیند. به عبارت دیگر، هر یال را به یال‌های کوچک‌تر تقسیم کنید. می‌توان نشان داد برای این چندضلعی جدید یک جواب وجود دارد که دوربین‌ها به یال‌ها چسبیده‌اند و هر دوربینی که روی یک یال قرار دارد کل آن یال را شامل می‌شود.



شکل ۲.۲: یک نمونه چندضلعی متعامد و گراف بدست آمده از روی آن [۲۱]

به ازای هر یال از چندضلعی جدید که قسمت بالا یا چپ آن یال داخل P است، یک رأس در G قرار دهید. در صورتی که با قرار دادن دو دوربین روی دو رأس از G ناحیه دیدشان اشتراک داشته باشد، آن دو رأس را با یک یال به هم وصل کنید. سپس وزن هر رأس را برابر طول یال P متناظر با آن قرار دهید. شکل ۲.۲ را ملاحظه کنید.

می‌توان نشان داد که جواب مسئله $MLSC$ متناظر با جواب مسئله‌ی کم وزن‌ترین پوشش رأسی گراف G است که این مسئله در حالت کلی NP -سخت است. همچنین می‌توان نشان داد که گراف ساخته شده G یک گراف دوبخشی است. از آنجایی که یک الگوریتم چندجمله‌ای برای مسئله‌ی کم وزن‌ترین پوشش رأسی وجود دارد، پس یک الگوریتم چندجمله‌ای برای حل مسئله $MLSC$ خواهیم

داشت. این الگوریتم به وسیله برنامه‌ریزی خطی حل می‌شود. همچنین می‌توان نشان داد همیشه برای برنامه‌ریزی خطی ارائه شده یک جواب وجود دارد که متغیرهای آن ۰ یا ۱ هستند. به همین دلیل می‌توان از جواب برنامه‌ریزی خطی استفاده کرد و از روی آن جواب اصلی برنامه را مشخص کرد. برای مشاهده جزئیات می‌توانید به [۲۱] مراجعه کنید.

۵.۲ تمیز کردن چندضلعی‌ها یا مسئله‌ی دزد و پلیس

در این قسمت مروری بر کارهای انجام شده در زمینه‌ی تمیز کردن چندضلعی‌ها انجام می‌دهیم. به طور کلی در مسئله‌ی دزد و پلیس، یک پلیس یا جستجوگر^{۱۴} به عنوان یک l -جوینده^{۱۵} با l چراغ قوه^{۱۶} در نظر گرفته می‌شود که می‌تواند چراغ قوه‌هایش را مرتباً با یک سرعت محدود بچرخاند [۴۹]. بنابراین یک ∞ -جوینده (که جوینده همه‌جانبه^{۱۷} نیز نامیده می‌شود) یک ربات متحرک است که دارای سنسور با دید ۳۶۰ درجه برای دیدن دزدها می‌باشد. لاوال و همکارانش^{۱۸} [۳۶] اولین الگوریتم را برای حل مسئله‌ی دزد و پلیس توسط یک l -جوینده ارائه دادند. آن‌ها چندضلعی را براساس خاصیت رویت‌پذیری به سلول‌هایی تقسیم کردند و مسئله را به جستجو روی یک گراف اطلاعات^{۱۹} با سایز نمایی تبدیل کردند. درهام و همکارانش^{۲۰} [۱۹] مسئله‌ی طراحی مسیر حرکت برای یک گروه از ربات‌های متحرک درون یک محیط مسطح ناشناخته در نظر گرفتند. آن‌ها فرض کردند که ربات‌ها دارای قدرت دید محدود هستند و دارای قابلیت تبادل اطلاعات با یکدیگر را دارند. یک الگوریتم ارائه دادند که تضمین می‌کند تمام دزدها را پیدا کند.

^{۱۴}Pursuit

^{۱۵} l -searcher

^{۱۶}Flashlights

^{۱۷}Omnidirectional

^{۱۸}Lavalle et al.

^{۱۹}Information Graph

^{۲۰}Durham et al.

فصل ۳

پوشش موزه‌های هنر متعامد توسط کمترین مجموع طول k -فرستنده‌های لغزان

در این فصل، ابتدا در قسمت ۱.۳ چند مفهوم مقدماتی را که در این فصل مورد نیاز هستند، شرح می‌دهیم. سپس در قسمت ۲.۳ به تعریف مسئله‌ی پوشش موزه‌های هنر متعامد توسط k -فرستنده‌های لغزان با کمترین مجموع طول ($MLSCk$) پرداخته و با توجه به اثبات IN-پی -سخت بودن این مسئله که در قسمت ۳.۳ ارائه شده است، در قسمت ۴.۳ یک الگوریتم در زمان چندجمله‌ای با ضریب تقریب ۲ ارائه می‌دهیم. همچنین مسئله را در حالت تعمیم داده شده بیان می‌کنیم. در قسمت ۵.۳ نتیجه‌ی بدست آمده در این فصل و همچنین مسائلی را که می‌توان در آینده در این رابطه بررسی کرد، مطرح می‌کنیم. نتایج بدست آمده از این فصل در مقالات زیر به چاپ رسیده‌اند که یکی نسخه کنفرانس [۴۰] و دیگری نسخه تعمیم یافته مجله [۴۱] است.

- Salma Sadat Mahdavi, Saeed Seddighin and Mohammad Ghodsi *Covering Orthogonal Polygons with Sliding k -Transmitters*, In the Proceedings of the 26th Canadian Conference on Computational Geometry, 2014

- Salma Sadat Mahdavi, Saeed Seddighin and Mohammad Ghodsi, *Covering Orthogonal Polygons with Sliding k -Transmitters*, Theoretical Computer Science, 85: 163–181, 2020.

۱.۳ مفاهیم مقدماتی

۱.۱.۳ مسئله‌ی پوشش رأسی کمینه

یکی از مسائل معروف در نظریه گراف، مسئله‌ی پوشش رأسی کمینه است. در این مسئله هدف انتخاب کمترین تعداد رأس از گراف است طوری که حداقل یک رأس از دو سر هر یال در این مجموعه قرار گرفته باشد. این مسئله را می‌توان در مدل گراف‌های وزن دار نیز بررسی کرد. در این مدل هر رأس دارای وزن مربوط به خودش است و هدف مسئله انتخاب کم‌وزن‌ترین زیرمجموعه از رأس‌ها است که حداقل یک سر از هر یال درون آن قرار داشته باشد.

نشان داده شده است که هر دو مدل از این مسئله برای گراف‌های کلی مسائل IN-PI -سخت هستند [۲۹]. برای این مسئله راه‌حل‌های تقریبی ارائه شده است که آن‌را در زمان چندجمله‌ای حل می‌کنند. ضریب تقریب بهترین الگوریتم ارائه شده برابر است با $2 - \frac{c}{\log V}$ [۳۱]. همچنین ثابت شده که نمی‌توان هیچ الگوریتمی با تقریب بهتر از $1/36$ برای این مسئله ارائه کرد، مگر این که پی با IN-PI برابر باشد [۱۷].

این مسئله یکی از کلیدی‌ترین مسائل در زمینه الگوریتم‌های با پارامتر ثابت^۱ است. راه‌حلی که در این زمینه برای این مسئله ارائه شده است دارای زمان اجرای $1/273^k \cdot n^{O(1)}$ است [۲۵]. عدد k برابر با پاسخ سوال است.

^۱Fixed Parameter Tractable

۲.۱.۳ برنامه‌ریزی خطی

برنامه‌ریزی خطی، یا همان بهینه‌سازی خطی، روشی در ریاضیات است که به پیدا کردن مقدار کمینه یا بیشینه از یک تابع خطی روی یک چندضلعی محدب می‌پردازد [۱۴]. این چندضلعی محدب در حقیقت نمایش نموداری تعدادی محدودیت از نوع نامعادله روی متغیرهای تابع است. به بیان ساده‌تر به وسیله برنامه‌سازی خطی می‌توان بهترین نتیجه (مثلاً بیشترین سود یا کمترین هزینه) را در شرایط خاص و با محدودیت‌های خاص به دست آورد. کاربرد اصلی استفاده برنامه‌ریزی خطی در مدیریت و اقتصاد است، اما در مهندسی نیز کاربردهای فراوانی دارد. در واقع برنامه‌ریزی خطی بخشی از تحقیق در عملیات و موسوم به علم مدیریت است که اولین بار توسط نیروی هوایی ارتش آمریکا بکار گرفته شد [۲۸].

نخستین بار در سال ۱۹۷۹ لئونید خاچیان^۲ نشان داد که مسئله‌ی برنامه‌ریزی خطی در مرتبه زمانی چندجمله‌ای قابل حل است [۳۴]. اما پیشرفت اساسی‌تر زمانی حاصل شد که نراندرا کارمارکار^۳ یک روش نقطه داخلی^۴ جدید برای حل این مسائل معرفی کرد [۴۳]. مثال دانترینگ برای منتسب کردن هفتاد نفر به هفتاد شغل متمایز، کارآمدی برنامه‌ریزی خطی را به نمایش می‌گذارد. توان محاسباتی لازم برای آزمودن همه جایگشت‌های ممکن این مسئله بسیار بالاست. این تعداد از تعداد ذرات موجود در عالم بیشتر است. با این حال، پیدا کردن پاسخ بهینه با تبدیل مسئله به یک مسئله‌ی برنامه‌ریزی خطی و حل آن با روش سیمپلکس تنها لحظه‌ای طول می‌کشد.

^۲Leonid Khachiyan^۳Narendra Karmarkar^۴Interior Point

۳.۱.۳ برنامه‌ریزی تفاضلی

برنامه‌ریزی تفاضلی نوع خاصی از مسئله‌ی امکان‌سنجی^۵ خطی است که در آن همه شرط‌ها به صورت زیر اند:

$$x_i - y_i \geq c_i$$

که در آن x_i و y_i متغیرهای برنامه هستند و c_i عدد ثابتی است که برای هر شرط متفاوت است. هدف این مسئله این است که تشخیص دهد آیا می‌توان طوری به متغیرها مقدار نسبت داد که همه شرط‌های داده شده برقرار باشند؟

راه‌حلی که برای این مسئله به دست آمده ساده و زیباست. به این صورت که اگر n متغیر و m شرط در برنامه باشد، یک گراف با $n + 1$ رأس و $m + n$ یال می‌سازیم. به ازای هر متغیر یک رأس قرار می‌دهیم و رأس شماره $n + 1$ را به صورت جدا قرار می‌دهیم. به ازای هر شرط یک یال جهت‌دار با وزن $-c_i$ از رأس متناظر با x_i به y_i می‌کشیم. همچنین از رأس $n + 1$ ، n یال جهت‌دار به دیگر رئوس با وزن ۰ قرار می‌دهیم.

اثبات شده برنامه داده شده دارای جواب است اگر و فقط اگر در گراف داده شده هیچ دور با مجموع وزن منفی وجود نداشته باشد. همچنین در صورتی که گراف دور منفی نداشته باشد اگر مقدار هر متغیر را برابر با فاصله رأس متناظر با آن از رأس آخر قرار دهیم، همه شرایط مسئله برقرار می‌شود. به همین دلیل می‌توان این برنامه را با استفاده از الگوریتم بلمن-فورد^۶ حل کرد.

الگوریتم بلمن-فورد الگوریتمی است که تشخیص می‌دهد آیا گراف داده شده دور با طول منفی دارد یا نه. همچنین فاصله یک رأس از بقیه رئوس را در زمان $\mathcal{O}(|V(G)||E(G)|)$ پیدا می‌کند [۵۱]. یکی از مهمترین نکاتی که در چنین برنامه‌هایی وجود دارد این است که در صورتی که همه مقادیر c_i صحیح باشند، پاسخی صحیح برای این برنامه‌ها وجود دارد. همچنین می‌توان شبیه به کاری که در

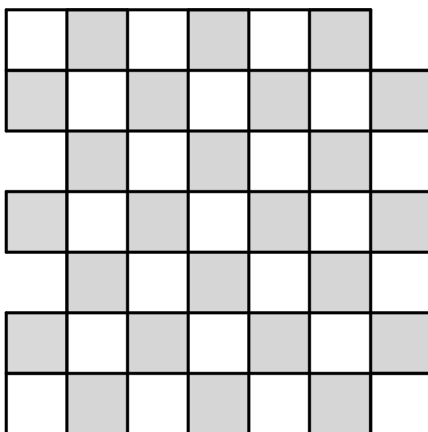
^۵Feasibility Problem

^۶Bellman-ford

برنامه ریزی خطی انجام می‌دهیم برای این برنامه‌ها یک تابع هدف معین کنیم و نشان دهیم که از بین تمامی پاسخ‌هایی که برای این برنامه وجود دارد، پاسخی مطلوب ماست که مقدار یک تابع را بیشینه یا کمینه می‌کند. البته این تابع بایستی به صورت تفاضل دو تا از متغیرها باشد.

۴.۱.۳ مسئله‌ی کاشی‌کاری

مسئله‌ی کاشی‌کاری^۷ در ترکیبیات از اهمیت فراوانی برخوردار است [۶]. یکی از ابتدایی‌ترین سوال‌هایی که در زمینه ترکیبیات مطرح است، این است که آیا می‌توان ناحیه‌ای که در شکل ۱.۳ نمایش داده شده است را با کاشی‌های ۱×۲ کاشی‌کاری کرد؟ به طور کلی روش کاشی‌کاری به صورت زیر تعریف می‌شود:

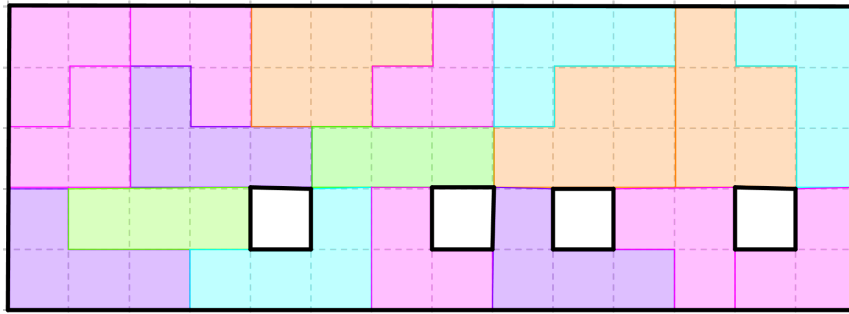


شکل ۱.۳: یک ناحیه در مسئله‌ی کاشی‌کاری

تعریف ۱.۳ پر کردن یک ناحیه با تعدادی کاشی، طوری که هیچ دو کاشی اشتراک نداشته باشند و کل ناحیه پوشیده شود کاشی‌کاری نام دارد.

پاسخ سوال بالا منفی است، زیرا تعداد خانه‌های سیاه این شکل بیشتر از تعداد خانه‌های سفید آن است و هر کاشی ۱×۲ بر روی یک خانه سیاه و یک خانه سفید قرار می‌گیرد. کاشی‌کاری لزوماً توسط کاشی‌های ۱×۲ انجام نمی‌شود. برای مثال در شکل ۲.۳ نمونه‌ای از کاشی‌کاری شدن یک ناحیه

^۷Tiling



شکل ۲.۳: کاشی‌کاری یک ناحیه توسط تعدادی کاشی داده شده

با استفاده از تعدادی کاشی آمده است. یکی از مسائل معروف حوزه کاشی‌کاری، کاشی کردن یک ناحیه با دومینو^۸ (یک کاشی ۱×۲) است. مسئله به صورت دقیق در زیر آمده است:

مسئله ۲.۳ ناحیه C به مساحت $۲k$ داده شده است. آیا می‌توان این ناحیه را با k دومینو کاشی‌کاری کرد؟

می‌توان مسئله‌ی بالا را به مسئله‌ی تطابق بیشینه^۹ در گراف دوبخشی کاهش داد. به این صورت که به ازای هر مربع ۱×۱ در ناحیه داده شده یک رأس در گراف قرار می‌دهیم و در صورتی که دو مربع دارای ضلع مشترک هستند، بین آن‌ها یال قرار می‌دهیم. ثابت شده است می‌توان ناحیه C را با دومینوها کاشی‌کاری کرد اگر و فقط اگر گراف داده شده یک تطابق کامل^{۱۰} داشته باشد [۱۸].

به نظر می‌رسد بتوان راه‌حل مشابهی برای مسئله‌ی تصمیم‌گیری که آیا می‌توان یک ناحیه را با تعدادی کاشی داده شده کاشی‌کاری کرد یا نه ارائه داد. اما این تصور کاملاً اشتباه است. به عبارت دیگر حتی ساده‌ترین مسئله بعد از مسئله‌ی کاشی‌کاری با دومینو را نیز نمی‌توان به صورت مناسبی حل کرد. مسئله‌ی زیر را در نظر بگیرید:

مسئله ۳.۳ ناحیه C به مساحت $۳k$ داده شده است. آیا می‌توان این ناحیه را با k کاشی ۱×۳ کاشی‌کاری کرد؟

^۸Domino

^۹Matching

^{۱۰}Perfect Matching

می‌توان گفت این مسئله ساده‌ترین مسئله‌ی کاشی‌کاری بعد از مسئله‌ی کاشی‌کاری با دومینو است. برخلاف مسئله‌ی قبلی این مسئله بسیار سخت است. در حقیقت این مسئله یک مسئله NP -سخت است و نباید انتظار داشته باشیم برای آن یک راه‌حل چندجمله‌ای پیدا شود [۵].

از جمله جذابیت‌های مسئله‌های حوزه کاشی‌کاری این است که ثابت شده برای برخی از این مسائل هیچ الگوریتمی وجود ندارد. در حقیقت این مسئله‌ها تصمیم‌ناپذیر^{۱۱} هستند [۴۷]. مسئله‌ی زیر یکی از این مسائل است:

مسئله ۴.۳ کاشی‌های c_1, c_2, \dots, c_n داده شده‌اند. اگر از هر کدام از این کاشی‌ها به مقدار بی‌نهایت داشته باشیم، آیا می‌توان کل صفحه دوبعدی را با آن‌ها کاشی‌کاری کرد؟

نشان داده شده، اگر بتوان الگوریتمی برای این مسئله ارائه داد، می‌توان برای مسئله A^{TM} نیز الگوریتمی ارائه داد که امکان‌پذیر نیست!

۲.۳ تعریف مسئله

مسئله‌ی محافظت^{۱۲} و مسئله‌ی روشن‌سازی^{۱۳} تعمیم داده شده مسئله‌ی معروف موزه‌ی هنر در هندسه‌ی محاسباتی هستند [۴۴، ۵۰]. هدف یافتن کمترین تعداد نگهبان کافی برای محافظت یا روشن کردن یک ناحیه براساس محدودیت‌های مشخص است. ناحیه‌ی تحت مراقبت ممکن است یک چندضلعی یا یک صفحه دارای موانعی به صورت چندضلعی یا پاره‌خط باشد. مکان نگهبان ممکن است محدود به رأس‌ها (نگهبان رأسی) یا یال‌های (نگهبان یالی) یک چندضلعی ورودی، یا بدون محدودیت (نگهبان نقطه‌ای) باشد. یک نگهبان ممکن است مانند یک فرستنده به طور یکسان در تمام جهات روشن‌سازی کند، یا مانند یک نورافکن در یک زاویه و در یک جهت مشخص نورافشانی کند.

به دنبال پیشرفت در تکنولوژی بی‌سیم و نیاز به ارائه سرویس‌های بی‌سیم به مشتریان، فابریلا مونرویی

^{۱۱} Undecidable

^{۱۲} Guarding

^{۱۳} Illumination

و همکارانش^{۱۴} [۲۳] و آیکهولزر و همکارانش^{۱۵} [۱] نوع جدیدی از مسائل روشن‌سازی را به نام روشن‌سازی فرستنده^{۱۶} معرفی کردند. در این مسئله یک نگهبان به صورت یک فرستنده بی‌سیم که دارای محدوده نامحدود پخش امواج و قدرت نفوذ درون k دیوار برای رسیدن به مقصد (مشتری) است، مدل می‌شود (برای عدد ثابت $k \geq 0$). به صورت هندسی، دیوارها اغلب به صورت پاره‌خط در صفحه نشان داده می‌شوند. زمانی که یک فرستنده درون یک چندضلعی قرار می‌گیرد برای دسترسی به خارج چندضلعی باید از یک دیوار عبور کند.

در مسئله‌ی پوشش چندضلعی‌ها توسط k - فرستنده‌ها فرض می‌شود که یال‌های چندضلعی دارای ضخامت ϵ هستند (دیوارها مانند کاغذ در نظر گرفته می‌شوند). با این فرض ضخامت دیوار که یک عامل مهم در انتقال انرژی، صدا و ... است نادیده گرفته می‌شود. ما نیز در مرحله اول دیوارها را مانند کاغذ در نظر گرفته و پیچیدگی مسئله $MLSC_k$ را ثابت می‌کنیم. سپس با توجه به این پیچیدگی بودن آن، یک الگوریتم چندجمله‌ای با ضریب تقریب ۲ ارائه می‌دهیم. در مرحله دوم مسئله را تعمیم داده و عامل مهم ضخامت را در نظر می‌گیریم. سپس ثابت می‌کنیم مسئله‌ی تعمیم یافته همچنان یک الگوریتم با ضریب تقریب ۲ دارد.

حال فرض کنید که دیوارها مانند کاغذ هستند. همچنین فرض کنید چندضلعی ساده و متعامد P و همچنین یک عدد صحیح و مثبت k به ما داده شده است. هدف قرار دادن تعدادی k - فرستنده لغزان درون P است به طوری که تمام چندضلعی را پوشش داده و مجموع طول آن‌ها کمترین باشد. محیط و یال‌های P را به ترتیب با $\mathcal{E}(P)$ و $A(P)$ نشان می‌دهیم. نقاط انتهایی هر یال $e \in \mathcal{E}(P)$ را امتداد می‌دهیم تا به یک خط شامل e دست یابیم. فرض کنید L مجموعه این خطوط باشد. واضح است که L فضای $A(P)$ را به تعدادی مستطیل افراز می‌کند که آن‌ها را با $\mathcal{P}(P)$ نشان می‌دهیم.

^{۱۴}Fabila-Monroy et al.^{۱۵}Aichholzer et al.^{۱۶}Modem Illumination

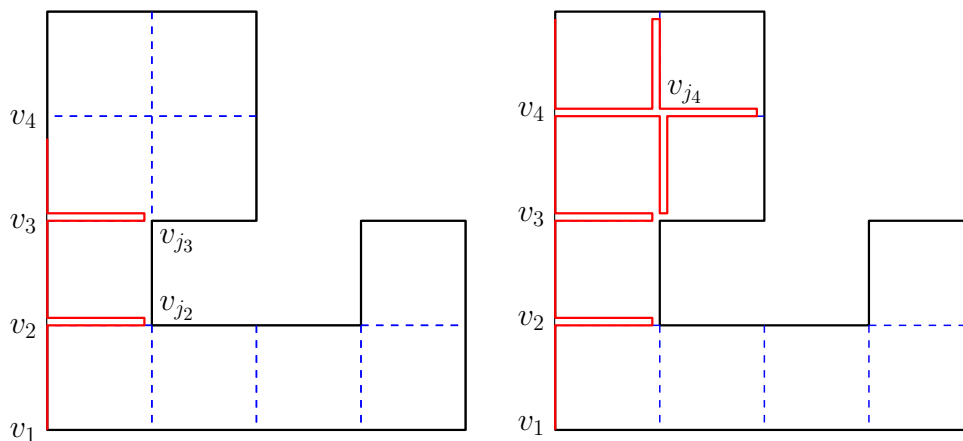
۳.۳. ان‌پی-سخت بودن مسئله

با استفاده از کاهش مسئله $MLSCk$ به مسئله‌ی کاشی‌کاری توسط مستطیل‌های ۱×۳ که در ۴.۱.۳ توضیح دادیم، ثابت می‌کنیم که این مسئله ان‌پی-سخت است. در مسئله‌ی کاشی‌کاری، یک چندضلعی متعامد R که روی شبکه G رسم شده است به عنوان ورودی داده می‌شوند. هدف قرار دادن مستطیل‌های ۱×۳ غیرهمپوشان برای کاشی کردن تمام R است.

اثبات ما برای ان‌پی-سخت بودن مسئله $MLSCk$ شامل دو قسمت است. ابتدا، یک چندضلعی متعامد جدید P از روی R می‌سازیم. سپس ثابت می‌کنیم برای هر جواب مسئله $MLSCk$ روی P ، یک جواب متناظر برای مسئله‌ی کاشی‌کاری روی R وجود دارد و برعکس.

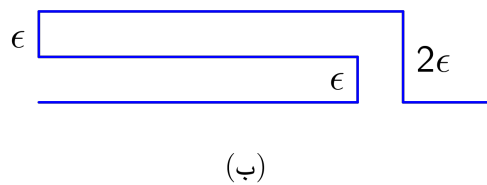
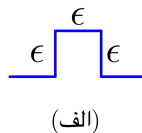
ساختن کاهش

برای ساختن این کاهش R و G به ما داده می‌شوند. رأس‌های R را با V_R نشان می‌دهیم. فرض کنید n تعداد نقاطی از شبکه است که درون و روی مرز R قرار دارند. E_G ، را مجموعه یال‌های G که درون (نه روی مرز) هستند قرار می‌دهیم. EE_G را به این صورت تعریف می‌کنیم که به ازای هر $e \in E_G$ دو یال e, e' درون EE_G قرار می‌دهیم (به عبارت دیگر از هر یال e دو نسخه نگه می‌داریم). رأس‌های G که روی مرز R قرار دارند را با $V_{BR} = \{v_1, \dots, v_n\}$ (در جهت ساعتگرد) نشان می‌دهیم. فرض



شکل ۳.۳: طریقه ساختن P از روی R

کنید VE_P مجموعه رأس‌ها و یال‌های P در جهت ساعتگرد باشد که در ابتدا تهی است. برای ساختن چندضلعی P ، مجموعه VB_R را پیمایش می‌کنیم. این پیمایش را از رأس v_1 شروع کرده و ادامه می‌دهیم تا به اولین رأس مانند v_i برسیم به طوری که $v_i \in VB_R$ و $v_i \notin V_R$ (مانند v_2 در شکل ۳.۳). حال یال $e_i \in E_G$ که مجاور v_i است را پیمایش کرده و به رأس v_j می‌رسیم. اگر $v_j \notin VB_R$ ، مانند رأس v_{j_2} در شکل ۳.۳، آنگاه سمت راست‌ترین یال مجاور v_j در جهت ساعتگرد که هنوز پیمایش نشده را پیمایش می‌کنیم. اگر تمام یال‌های مجاور v_j پیمایش شده باشند یا $v_j \in VB_R$ باشد (به طور مثال رأس v_{j_2} یا v_{j_3} در شکل ۳.۳)، آنگاه e'_j نسخه دوم یال e_i را پیمایش می‌کنیم. این طریقه پیمایش را ادامه می‌دهیم تا زمانی که دوباره به رأس v_i بازگردیم. حال پیمایش VB_R را از رأس v_i ادامه می‌دهیم تا زمانی که دوباره به رأس پیمایش نشده‌ای مانند v_x برسیم که رأسی از G است و $v_x \notin V_R$. سپس همان مراحل را که برای رأس v_i انجام دادیم تکرار می‌کنیم تا اینکه دوباره به رأس v_x بازگردیم. روند پیمایش R و G را ادامه می‌دهیم تا زمانی که تمام رأس‌های VB_R پیمایش شوند. در حین انجام روند پیمایش، هر رأس یا یالی را که پیمایش می‌کنیم به مجموعه VE_P اضافه می‌کنیم.

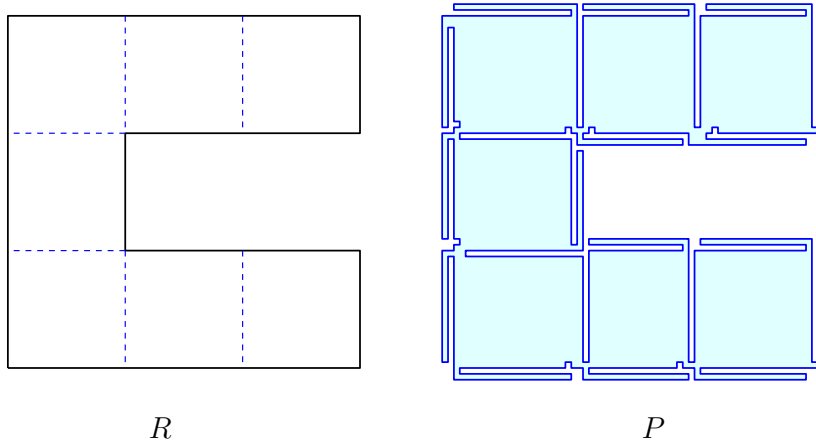


شکل ۴.۳: قسمت‌هایی که برای اجتناب از داشتن (الف) فرستنده‌های با طول بیشتر از $1 + \epsilon$ و (ب) فرستنده‌هایی که دو قسمت ناهمبند از چندضلعی را پوشش دهند، اضافه می‌شوند.

همان‌طور که در شکل ۳.۳ ملاحظه می‌کنید، طول یال‌های کوچک درونی را بسیار کوچک (ϵ) در نظر می‌گیریم. پس از پایان روند پیمایش، برای اجتناب از داشتن k - فرستنده‌های لغزان با طول بیشتر از $1 + \epsilon$ قسمت‌های کوچکی که در شکل ۴.۳ (الف) ملاحظه می‌کنید و همچنین برای هر یال از

شبکه که روی مرز R قرار دارد، برای اجتناب از داشتن فرستنده‌هایی که بخواهند دو قسمت ناهمبند از P را ببینند، قسمت نشان داده شده در شکل ۴.۳. (ب) را به P اضافه می‌کنیم. در نتیجه شکل ۵.۳ را خواهیم داشت.

از آنجایی که تمام رأس‌ها و یال‌های G را که درون R قرار دارند پیمایش می‌کنیم، بنابراین پیچیدگی ساختن P $O(n)$ خواهد بود.



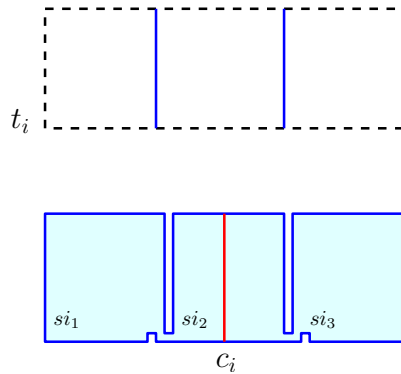
شکل ۵.۳: چندضلعی ورودی مسئله‌ی کاشی‌کاری (R) و چندضلعی ساخته شده از روی آن (P)

اثبات درستی کاهش

حال پس از ساختن P ، ثابت می‌کنیم برای هر جواب مسئله $MLSCk$ روی P ، یک جواب متناظر برای مسئله‌ی کاشی‌کاری روی R وجود دارد. فرض کنید g تعداد سلول‌های شبکه G درون R باشد که مضربی از ۳ است. اگر g مضربی از ۳ نباشد آنگاه R نمی‌تواند توسط مستطیل‌های ۱×۳ پوشیده شود. همچنین فرض می‌کنیم $k = ۲$ باشد به عبارت دیگر اشعه فرستنده‌های لغزان می‌توانند از میان دو دیوار عبور کنند.

ابتدا فرض کنید مسئله $MLSCk$ را برای چندضلعی P حل کرده و جواب را با $C = \{c_1, c_2, \dots, c_x\}$ نشان می‌دهیم. مقدار m را برابر مجموع طول این دوربین‌ها قرار دهید. چون P را طوری ساخته‌ایم که نمی‌تواند دوربین متعامد با طول بیشتر از $۱ + \epsilon$ داشته باشد، پس به ازای هر c_i داریم: $|c_i| \leq ۱ + \epsilon$. از طرفی برای پوشش دادن اگر $m = g/۳$ باشد، چون ۱ تمام C را پوشش می‌دهد، ۲ فضای

پوشش فرستنده‌ها با هم پوشانی ندارند، (۳) هر c_i یک ۲-فرستنده است و (۴) براساس طرز ساخته شدن P از روی R ، نتیجه می‌گیریم که پاسخ مسئله‌ی کاشی‌کاری برای R بله خواهد بود. در غیر اینصورت جواب خیر است.



شکل ۶.۳: محل قرار دادن یک فرستنده برای هر کاشی

حال فرض کنید مسئله‌ی کاشی‌کاری را برای R حل کرده و جواب $T = \{t_1, t_2, \dots, t_m\}$ را بدست آوردیم. براساس T مجموعه‌ای از k -فرستنده‌ها، $C = \{c_1, c_2, \dots, c_m\}$ ، را برای پوشش کامل درون P قرار می‌دهیم. براساس طریقه ساختن کاشش، هر مستطیل $t_i \in T$ در R به سه عدد مربع si_1, si_2, si_3 و P در si_2 افزایش می‌شود. یک k -فرستنده لغزان را در وسط مربع si_2 قرار می‌دهیم (شکل ۶.۳ را ملاحظه کنید). به دلیل آنکه c_i فقط تا نقاط پشت ۲ دیوار را می‌تواند پوشش دهد بنابراین فقط می‌تواند si_2, si_1 و si_3 را پوشش دهد و مجموعه C تمام P را پوشش می‌دهد. برای پوشش دادن قسمت‌های کوچکی با طول ϵ که به R اضافه کردیم و توسط C پوشیده نمی‌شوند نیز فرستنده‌های متعامد با طول ϵ قرار می‌دهیم. از آنجاییکه مستطیل‌ها با هم پوشانی ندارند پس $|C| = m$ که مجموع طول تمام فرستنده‌های C است، کمترین خواهد بود. در نتیجه مجموع طول تمام فرستنده‌ها $m + \epsilon$ و کمینه خواهد بود.

بنابراین قضیه زیر را نتیجه می‌گیریم.

قضیه ۵.۳ مسئله‌ی پوشش یک چندضلعی ساده و متعامد توسط کمترین مجموع طول k -فرستنده‌های لغزان حتی برای $k = 2$ ان‌پی-سخت است.

برای مشاهده‌ی جزئیات بیشتر این کاهش می‌توانید به مقاله [۴۱] مراجعه کنید.

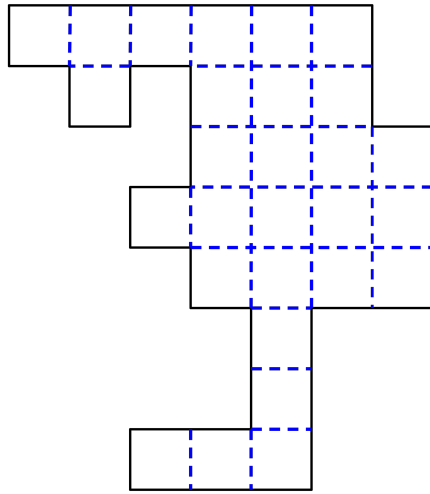
۴.۳ الگوریتم ۲ - تقریب

همان‌طور که توضیح دادیم مسئله $MLSC_k$ ان‌پی-سخت است. بنابراین یک الگوریتم تقریبی با ضریب تقریب ۲ برای این مسئله ارائه می‌دهیم. الگوریتم ما شامل دو قسمت است. ابتدا یک برنامه خطی نوشته و آن را در زمان چندجمله‌ای حل می‌کنیم. اگر جواب برنامه خطی صحیح باشد، آنگاه مسئله حل شده است. اما جواب همیشه صحیح نیست. بنابراین با توجه به پاسخ برنامه خطی تعدادی محدودیت به مسئله اضافه کرده و نشان می‌دهیم که همیشه یک جواب صحیح بهینه برای برنامه خطی جدید وجود دارد که در زمان چندجمله‌ای بدست می‌آید. همچنین نشان خواهیم داد جواب بدست آمده حداکثر دو برابر جواب مسئله‌ی اصلی است، پس یک الگوریتم با ضریب تقریب ۲ بدست آورده‌ایم. ابتدا مانند روشی که در حالت ۰ - فرستنده [۲۱] ارائه شد را انجام داده و یال‌های چندضلعی را به قسمت‌های کوچک‌تر تقسیم می‌کنیم به طوری که حتماً یک جواب بهینه برای مسئله وجود داشته باشد که در شرایط زیر صدق کند:

- تمام فرستنده‌ها به یال‌ها چسبیده باشند.

- هر فرستنده که دربرگیرنده یک قسمت از یک یال است کل آن یال را شامل شود.

همان‌طور که در [۲۱] ثابت شد، با در نظر گرفتن شرایط بالا می‌توان فرض کرد که در جواب بهینه دوربین‌ها روی یال‌هایی قرار دارند که بالا یا سمت چپ آن‌ها درون چندضلعی قرار دارد. خطوطی را که یال‌های P روی آن‌ها قرار دارند در نظر بگیرید. این خطوط P را به تعدادی مستطیل افراز می‌کنند (شکل ۷.۳ را ملاحظه کنید). مجموعه این مستطیل‌ها را S می‌نامیم. یک برنامه خطی برای این مسئله ارائه می‌دهیم. به ازای هر یال که فرستنده می‌تواند روی آن قرار گیرد یک متغیر a_i تعریف می‌کنیم. فرض کنید E ، مجموعه یال‌هایی از P باشد که سمت بالا یا چپ آن‌ها درون چندضلعی قرار



شکل ۷.۳: افراز چندضلعی به مستطیل‌ها در الگوریتم تقریبی

دارد. به ازای هر یال $e \in E$ مقدار $l(e)$ برابر طول آن یال آن است. همچنین این شرط را قرار می‌دهیم که به ازای هر ناحیه از چندضلعی فرستنده‌ها باید طوری روی یال‌ها قرار گیرند که آن ناحیه را پوشش دهند. بنابراین برای هر ناحیه $s \in S$ ، $V(s)$ برابر مجموعه تمام یال‌هایی از E است که اگر فرستنده روی آن‌ها قرار گیرد تمام آن ناحیه را پوشش می‌دهد. در نتیجه برنامه خطی زیر را خواهیم داشت.

$$\min. \sum_{e \in E} a_e \cdot l(e) \quad (1.3)$$

$$s.t. \sum_{e \in V(s)} a_e \geq 1 \quad \forall s \in S \quad (2.3)$$

$$a_e \geq 0 \quad \forall e \in E \quad (3.3)$$

$$a_e \leq 1 \quad \forall e \in E \quad (4.3)$$

واضح است که اگر جواب این برنامه خطی صحیح باشد، آنگاه می‌توان یک جواب بهینه برای مسئله بدست آورد و براساس آن روی هر یال e که $a_e = 1$ است، یک فرستنده قرار داد. بنابراین شرط‌های

ذکر شده این فرستنده‌ها تمام P را پوشش داده و مجموع طول آن‌ها کمینه است. به علاوه می‌توان از روی جواب بدست آمده یک جواب صحیح غیر بهینه ساخت که در آن هر متغیر را که مقدار آن بیشتر یا مساوی $\frac{1}{2k}$ است برابر ۱ و هر متغیر را که مقدار آن کمتر است برابر ۰ قرار می‌دهیم. به دلیل آنکه هر ناحیه از S در محدوده تحت پوشش حداکثر $2k$ فرستنده قرار دارد، پس در جواب جدید متغیر متناظر حداقل یکی از فرستنده‌ها برابر ۱ است. به علاوه مقدار تابع هدف در جواب صحیح جدید حداکثر $2k$ برابر شده است. بنابراین با این روش گرد کردن^{۱۷} یک الگوریتم با ضریب تقریب $2k$ خواهیم داشت که با هدف ضریب تقریب ۲ ما در تناقض است. پس به روش دیگری جواب مسئله را از روی جواب برنامه خطی قبلی بدست می‌آوریم. در هر جواب (نه لزوماً صحیح) برنامه خطی که بدست بیاوریم، داریم:

$$\forall s \in S : \sum_{e \in V(s)} a_e \geq 1$$

یال‌هایی که در این شرط صدق می‌کنند می‌توانند افقی یا عمودی باشند. واضح است که چون مجموع متغیر نسبت داده شده به این یال‌ها بیشتر یا مساوی ۱ است، حداقل یکی از حالت‌های زیر برقرار است:

۱. مقدار $\sum a_e$ برای یال‌های افقی $e \in V(s)$ بیشتر یا مساوی ۰,۵ است.

۲. مقدار $\sum a_e$ برای یال‌های عمودی $e \in V(s)$ بیشتر یا مساوی ۰,۵ است.

بنابراین محدودیت‌های جدیدی را اضافه می‌کنیم. به ازای همه ناحیه‌هایی که حالت ۱ برای آن‌ها برقرار است شرط می‌کنیم که باید توسط فرستنده‌های افقی پوشیده شوند. همچنین به ازای همه ناحیه‌هایی که حالت ۲ برای آن‌ها برقرار است شرط می‌کنیم که باید توسط فرستنده‌های عمودی پوشیده شوند. با اضافه کردن این محدودیت‌ها جواب جدید بیشتر یا مساوی جواب قبلی خواهد بود.

برای هر $s \in S$ ، $V'(s)$ را برابر همه یال‌هایی قرار می‌دهیم که در $V(s)$ هستند و در شرط جدید صدق می‌کنند. حال یک برنامه‌ریزی جدید ارائه می‌دهیم که شبیه قبلی است اما به جای $V(s)$ از $V'(s)$

^{۱۷}Rounding

استفاده می‌کنیم.

$$\min. \sum_{e \in E} b_e \cdot l(e) \quad (5.3)$$

$$s.t. \sum_{e \in V'(s)} b_e \geq 1 \quad \forall s \in S \quad (6.3)$$

$$b_e \geq 0 \quad \forall e \in E \quad (7.3)$$

$$b_e \leq 1 \quad \forall e \in E \quad (8.3)$$

به طور مشابه ثابت می‌کنیم که هر جواب صحیح برای این برنامه معادل یک جواب برای مسئله‌ی اصلی است که دیگر لزوماً بهینه نیست. فرض کنید a' جواب برنامه خطی اولیه باشد. یک جواب قابل قبول برای برنامه خطی جدید به این صورت است که به ازای هر $e \in E$ قرار دهیم $b, b_e = 2a_e$. بنابراین جواب این برنامه حداکثر دو برابر جواب برنامه قبلی است. حال نشان می‌دهیم که یک جواب بهینه صحیح برای برنامه خطی جدید وجود دارد و اینکه این جواب را می‌توان در زمان چندجمله‌ای بدست آورد. بنابراین یک جواب با ضریب تقریب ۲ برای مسئله $MLSCk$ خواهیم داشت. بدون کم شدن از کلیت برنامه خطی، فرض کنید یال‌ها براساس ترتیب زیر به صورت $e_1, e_2, \dots, e_{|V|}$ اندیس گذاری شده‌اند:

۱. ابتدا یال‌های افقی از چپ به راست و در صورت برابری از بالا به پایین قرار دارند.

۲. سپس یال‌های عمودی از بالا به پایین و در صورت برابری از چپ به راست قرار دارند.

علت این اندیس گذاری اینست که به ازای هر $s \in S$ ، اعضای $V'(s)$ شامل دقیقاً یک بازه پشت سر هم از یال‌ها می‌شوند. بنابراین مسئله را به صورت زیر می‌نویسیم. متغیر جدید b_i^* را تعریف می‌کنیم

که تاثیری در جواب بهینه ندارد و مقدار آن برابر $\sum_{j=1}^i b_j$ است.

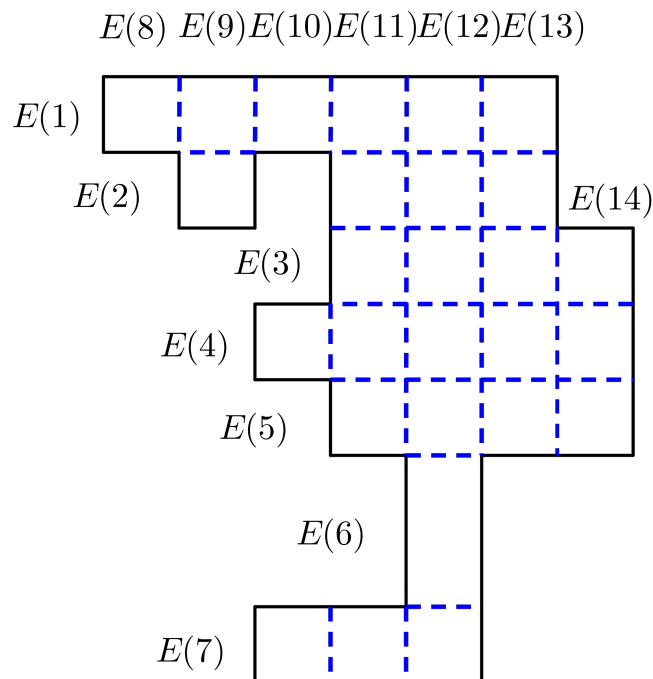
$$\min. \sum_{i=1}^{|E|} (b_i^* - b_{i-1}^*) l(e_i) \quad (9.3)$$

$$b_{r(s)}^* - b_{l(s)-1}^* \geq 1 \quad \forall s \in S \quad (10.3)$$

$$b_i^* - b_{i-1}^* \geq 0 \quad \forall 1 \leq i \leq |E| \quad (11.3)$$

$$b_i^* - b_{i-1}^* \leq 1 \quad \forall 1 \leq i \leq |E| \quad (12.3)$$

که $l(s)$ و $r(s)$ به ترتیب نشان دهنده شماره شروع و پایان بازه $V'(s)$ هستند. این برنامه را می‌توان به تعدادی برنامه کوچکتر تقسیم کرد به طوری که در هر برنامه تنها یال‌های میان دو خط افقی یا عمودی را بررسی می‌کنیم و به صورت زیر خواهند بود:



شکل ۸.۳: تقسیم مسئله به مسئله‌های کوچکتر در الگوریتم تقریبی

$$\min. \quad b_{|E(t)|}^* - b^* \quad (۱۳.۳)$$

$$s.t. \quad b_{r(s)}^* - b_{l(s)-1}^* \geq 1 \quad \forall s \in S \quad (۱۴.۳)$$

$$b_i^* - b_{i-1}^* \geq 0 \quad \forall 1 \leq i \leq |E(t)| \quad (۱۵.۳)$$

$$b_i^* - b_{i-1}^* \leq 1 \quad \forall 1 \leq i \leq |E(t)| \quad (۱۶.۳)$$

که $E(t)$ نشان دهنده مجموعه همه یال‌هایی است که در زیر مسئله بالا بررسی می‌کنیم. همان‌طور که ملاحظه می‌کنید تابع هدف و شرط‌های این برنامه به صورت تفاضل دو متغیر هستند. با توجه به آنچه در ۳.۱.۳ توضیح دادیم، این برنامه‌ها همیشه یک جواب بهینه صحیح دارند که می‌توان آنرا توسط الگوریتم بلمن فورد بدست آورد.

با توجه به آنچه در بالا مطرح کردیم، روش ارائه شده قابل پیاده‌سازی در زمان چندجمله‌ای است و قضیه زیر نتیجه می‌شود.

قضیه ۶.۳ مسئله‌ی پوشش یک چندضلعی متعامد ساده با استفاده از کمترین مجموع طول k - فرستنده‌های لغزان دارای یک الگوریتم چندجمله‌ای با ضریب تقریب ۲ است.

۱.۴.۳ تعمیم مسئله

حال در مرحله دوم فرض می‌کنیم که چندضلعی متعامد و ساده P دارای n رأس و n یال می‌باشد که هر یال e_i دارای ضخامت $1 < t_i \leq 0$ است. یک فرستنده p و یک نقطه q درون P در نظر بگیرید. فرض کنید پاره‌خط واصل pq و مرز P در مجموعه یال‌های E_q تلاقی داشته باشند. آنگاه p می‌تواند q را پوشش دهد اگر و تنها اگر داشته باشیم:

$$\sum_{j: e_j \in E_q} t_j \leq 1$$

حال هدف یافتن کمترین مجموع طول فرستنده لغزان برای پوشش تمام چندضلعی است به طوری که هر فرستنده نقاطی از محیط را پوشش می‌دهد که مجموع ضخامت دیوارهای عبور کرده تا آن نقطه کمتر مساوی ۱ باشد. بنابراین مسئله‌ی تعمیم داده شده به صورت زیر تعریف می‌شود.

مسئله ۷.۳ چندضلعی متعامد P داده شده است. P می‌تواند دارا یا بدون مانع باشد. هر یال این چندضلعی دارای ضخامت t_i است. همچنین این چندضلعی به نواحی متعامدی افراز شده است. هدف قرار دادن تعدادی دوربین لغزان در این چندضلعی است که نواحی این چندضلعی را به میزان لازم نگهبانی کند. به عبارت دقیق‌تر به ازای هر ناحیه از این چندضلعی می‌دانیم که حداقل چند دوربین باید آن ناحیه را نگهبانی کنند. ناحیه دید دوربین‌های لغزان تمام نقاطی از چندضلعی است که پاره‌خط عمود آن‌ها به پاره‌خط دوربین با یال‌هایی از P برخورد دارد که مجموع ضخامت آن‌ها بیشتر از ۱ نیست. هدف قرار دادن این دوربین‌هاست به قسمی که مجموع طول آن‌ها کمینه باشد.

حالت خاصی از این مسئله زمانی که ضخامت همه یال‌ها برابر $\frac{1}{k}$ باشد، و همه نواحی آن نیاز به یک دوربین داشته باشند، همان مسئله‌ی پوشش چندضلعی متعامد با کمترین مجموع طول k - فرستنده‌های لغزان است که در ۲.۳ مطرح کرده و ثابت کردیم k - پی - سخت است. بنابراین مسئله ۷.۳ یک مسئله k - پی - سخت خواهد بود. حال نشان می‌دهیم که الگوریتم ارائه شده در قسمت ۷.۳ با اندکی تغییر برای این مسئله نیز قابل استفاده است.

فرض کنید مانند قسمت قبل خطوط افقی و عمودی را طوری رسم کنیم که مساحت P را به تعدادی ناحیه افراز کند. این نواحی را در مجموعه S نگه می‌داریم. به ازای هر ناحیه $s \in S$ ، مقدار $N(s)$ را برابر حداقل تعداد فرستنده‌هایی که باید این ناحیه را پوشش دهند، قرار می‌دهیم. E و $V(s)$ را نیز مانند قبل تعریف می‌کنیم.

$$\min. \sum_{e \in E} a_e \cdot l(e) \quad (17.3)$$

$$s.t. \sum_{e \in V(s)} a_e \geq N(s) \quad \forall s \in S \quad (18.3)$$

$$a_e \geq 0 \quad \forall e \in E \quad (19.3)$$

شبهه به قبل، به ازای هر $s \in S$ یکی از شرط‌های زیر برقرار است.

۱. مقدار $\sum a_e$ برای یال‌های افقی $e \in V(s)$ بیشتر یا مساوی $\frac{N(s)}{4}$ است.

۲. مقدار $\sum a_e$ برای یال‌های عمودی $e \in V(s)$ بیشتر یا مساوی $\frac{N(s)}{4}$ است.

حال محدودیت‌های جدیدی را اضافه می‌کنیم. به ازای همه ناحیه‌هایی که حالت ۱ برای آن‌ها برقرار است شرط می‌کنیم که باید توسط فرستنده‌های افقی پوشیده شوند. همچنین به ازای همه ناحیه‌هایی که حالت ۲ برای آن‌ها برقرار است شرط می‌کنیم که باید توسط فرستنده‌های عمودی پوشیده شوند. در نتیجه برنامه خطی جدیدی را خواهیم داشت.

$$\min. \sum_{e \in E} b_e \cdot l(e) \quad (20.3)$$

$$s.t. \sum_{e \in V'(s)} b_e \geq N(s) \quad \forall s \in S \quad (21.3)$$

$$b_e \geq 0 \quad \forall e \in E \quad (22.3)$$

با اثباتی مانند اثبات قسمت ۴.۳ می‌توان نتیجه گرفت که برنامه جدید دارای یک جواب صحیح بهینه است که در زمان چندجمله‌ای به دست می‌آید. در نتیجه یک الگوریتم تقریبی با ضریب تقریب ۲ برای این مسئله خواهیم داشت.

۵.۳ نتیجه‌گیری

بنا بر آنچه در این فصل توضیح دادیم، مسئله‌ی پوشش چندضلعی‌های متعامد با کمترین مجموع طول فرستنده‌های لغزان در حالت k -فرستنده یک مسئله NP -سخت است. سپس یک الگوریتم تقریبی با ضریب تقریب ۲ ارائه دادیم. با توجه به این که این الگوریتم اولین نتیجه در این زمینه است امکان دارد بتوان با روش‌ها و تحقیقات جدید، یک الگوریتم تقریبی با ضریب تقریب کمتر از ۲ یا حتی یک الگوریتم $PTAS$ ^{۱۸} برای این مسئله ارائه داد. زمانی که برای یک مسئله یک الگوریتم تقریبی ارائه می‌شود، اولین سوال که به ذهن می‌رسد این است که آیا این بهترین الگوریتم تقریبی ارائه شده است یا نه؟ به همین دلیل برای ادامه کار در این زمینه می‌توانیم به بررسی یک حد پایین برای بهترین ضریب تقریب الگوریتم تقریبی چندجمله‌ای پردازیم.

همچنین می‌توان به بررسی این مسئله در زمینه الگوریتم‌های تصادفی^{۱۹} و الگوریتم‌های با پارامتر ثابت^{۲۰} پرداخت و نتایج جدیدی در این زمینه بدست آورد.

^{۱۸}PTAS

^{۱۹}Randomized Algorithm

^{۲۰}Fixed Parameter Algorithm

فصل ۴

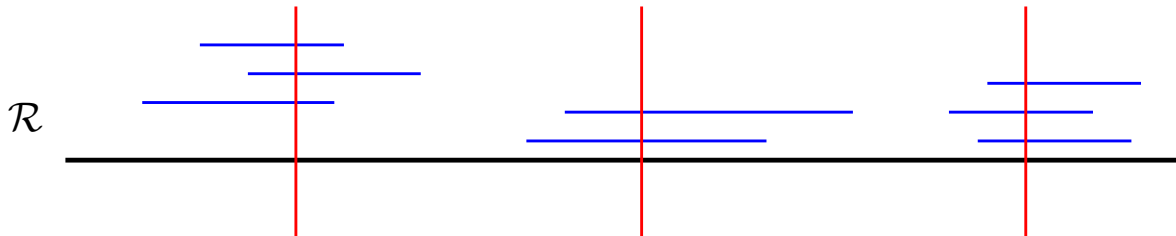
پوشش موزه‌های هنر متعامد توسط کمترین تعداد ۲- فرستنده‌های لغزان

در این فصل، ابتدا در قسمت ۱.۴ یک مفهوم مقدماتی را که در این فصل مورد نیاز است، شرح می‌دهیم. سپس در قسمت ۲.۴ به تعریف مسئله‌ی پوشش موزه‌های هنر متعامد و x -یکنوا توسط کمترین تعداد ۲- فرستنده‌های لغزان پرداخته و در قسمت ۳.۴ یک الگوریتم چندجمله‌ای با ضریب تقریب ۳ برای حل این مسئله زمانی که فقط مجاز به استفاده از فرستنده‌های عمودی لغزان هستیم، ارائه می‌دهیم. سپس در قسمت ۴.۴ یک الگوریتم چندجمله‌ای با ضریب تقریب ۷ برای حل مسئله‌ی پوشش موزه‌های هنر متعامد و x -یکنوا توسط کمترین تعداد ۲- فرستنده‌های لغزان ارائه می‌دهیم. در قسمت ۵.۴ نتیجه بدست آمده در این فصل و همچنین مسائلی را که می‌توان در آینده در این رابطه بررسی کرد، مطرح می‌کنیم.

۱.۴ مفاهیم مقدماتی

۱.۱.۴ مسئله برخورد با بازه‌ها

مسئله برخورد با بازه‌ها^۱ و یا پوشش بازه‌ها^۲ به این صورت است که مجموعه I از بازه‌ها در فضای \mathbb{R} داده شده است و هدف، تعیین کمترین تعداد خطوط عمودی است که بتوانند با تمام بازه‌ها برخورد داشته باشند. به عنوان مثال شکل ۱.۴ را در نظر بگیرید. این نمونه از مسئله با یک مرتبه پیمایش بازه‌ها از چپ به راست و در زمان خطی نسبت به تعداد کل بازه‌ها، حل می‌شود.



شکل ۱.۴: تعدادی بازه روی محور اعداد حقیقی به همراه کمترین تعداد خط عمودی برخورد کننده با آن‌ها نشان داده شده‌اند.

حالت کلی‌تر این مسئله به این صورت است که به ازای $d \in \mathbb{N}$ ، یک d -بازه^۳ مانند I عبارت است از اجتماع d بازه غیر تهی مانند $I^1, I^2, \dots, I^d \subset \mathbb{R}$. ورودی این مسئله، مجموعه‌ای مانند \mathcal{H} است که هر عضو آن یک d -بازه است و هدف، پیدا کردن مجموعه‌ای از نقاط مانند $X \subset \mathbb{R}$ است به طوری که با تمام اعضای \mathcal{H} برخورد داشته و اندازه آن کمینه است. همچنین منظور از برخورد یک نقطه (و یا خط عمود گذرنده از آن نقطه) با یک d -بازه مانند I این است که آن نقطه باید با حداقل یکی از d بازه‌ی موجود در I برخورد داشته باشد [۱۶]. نمونه‌ای از مسئله در حالت ۲-بازه در شکل ۲.۴ نشان داده شده است.

برخلاف سادگی مسئله در حالت ۱-بازه، برای حالت ۲-بازه، این مسئله تقریب-سخت است

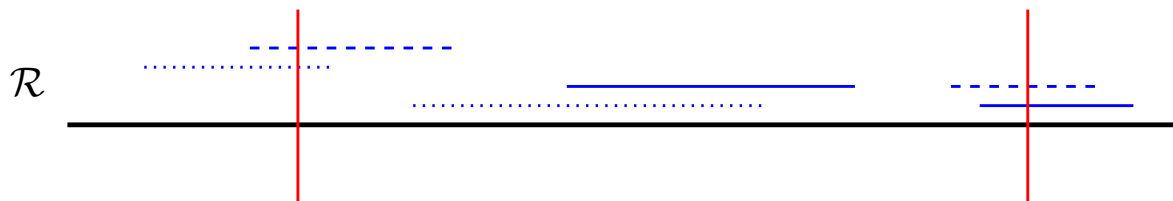
[۴۲، ۴]. بهترین نتیجه موجود برای حل مسئله‌ی پوشش d -بازه مربوط به کار بن دیوید و همکارانش^۴

^۱Interval Hitting

^۲Interval Covering

^۳d-interval

^۴Ben-David et al.



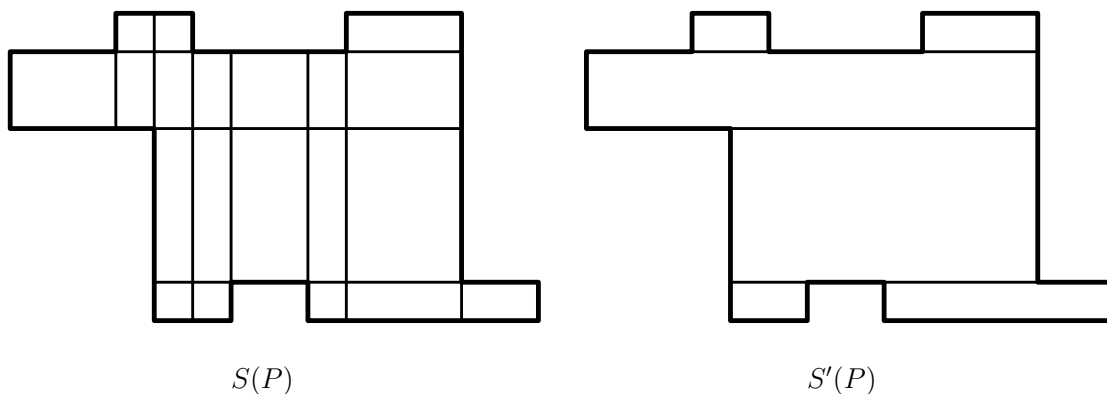
شکل ۲.۴: اعضای هر ۲- بازه یه سورت متمایز نشان داده شده‌اند. برای پوشش این ۲- بازه‌ها به حداقل ۲ خط عمودی نیاز است.

در سال ۲۰۱۲ است [۱۶]. آن‌ها با استفاده از یک دستگاه برنامه‌ریزی خطی، یک الگوریتم d -تقریب برای حل مسئله ارائه دادند.

۲.۴ تعریف مسئله

فرض کنید یک چندضلعی ساده، متعامد و x -یکنوا، P ، داده شده است. هدف قرار دادن کمترین تعداد ۲- فرستنده لغزان درون P است به طوری که تمام آن‌را پوشش دهند.

محیط و یال‌های P را به ترتیب با $\mathcal{E}(P)$ و $A(P)$ نشان می‌دهیم. نقاط انتهایی هر یال $e \in \mathcal{E}(P)$ را امتداد می‌دهیم تا به یک خط شامل e دست یابیم. فرض کنید L مجموعه این خطوط باشد. واضح است که L فضای $A(P)$ را به تعدادی مستطیل افراز می‌کند که آن‌ها را با $S(P)$ نشان می‌دهیم. فرض کنید تعداد مستطیل‌ها m است و داریم: $S(P) = \{s_1, \dots, s_m\}$. به ازای هر پاره‌خط s ، نقطه انتهایی سمت راست و نقطه انتهایی سمت چپ آن‌را به ترتیب با lr_s و lp_s نشان می‌دهیم.

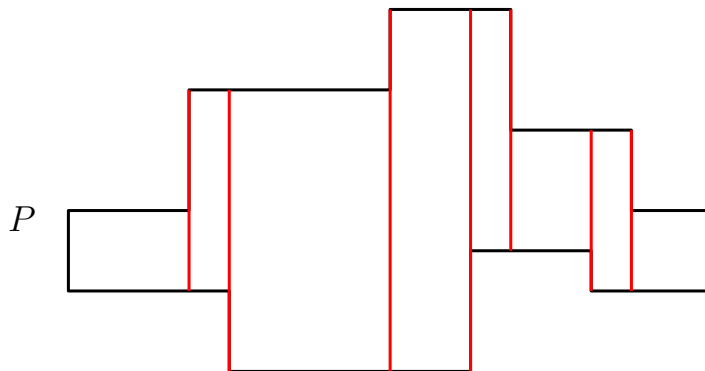


شکل ۳.۴: مستطیل‌بندی $S(P)$ و $S'(P)$ متناظر با چندضلعی P نشان داده شده‌اند.

نوع دیگری از مستطیل‌بندی را در نظر بگیرید که فقط از طریق امتداد دادن یال‌های افقی به دست

می‌آید و آن را با $S'(P)$ نمایش می‌دهیم. تفاوت این دو مستطیل‌بندی در شکل ۳.۴ نشان داده شده است.

هر یال عمودی عضو $\mathcal{E}(P)$ را از دو سمت امتداد می‌دهیم تا جایی که به مرز P برخورد کنند (به عبارت دیگر یال‌های عمودی را بیشینه^۵ می‌کنیم). مجموعه‌ی تمام پاره‌خط‌های عمودی بیشینه را با $H = \{h_1, \dots, h_m\}$ نشان می‌دهیم. در شکل ۴.۴ این مجموعه نشان داده شده است.



شکل ۴.۴: پاره‌خط‌های H با رنگ قرمز نشان داده شده‌اند.

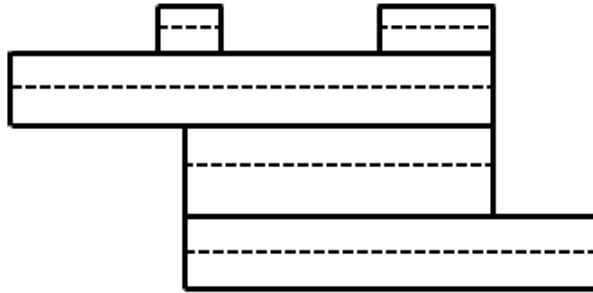
در قسمت ۴.۴ مسئله ۲- $MCSC$ در چندضلعی‌های متعامد و x -یکنوا را بررسی کرده و برای این مسئله الگوریتمی تقریبی با ضریب تقریب ۷ ارائه می‌دهیم. همچنین در قسمت ۳.۴ که در صورتی که بتوان حالت خاصی از این مسئله را که فقط مجاز به استفاده از فرستنده‌های عمودی است به صورت بهینه حل کرد، ضریب تقریب مسئله اصلی از ۷ به ۳ کاهش پیدا خواهد کرد. الگوریتم ما به این صورت است که ابتدا فرستنده‌های افقی که از نظر الگوریتم مناسب است را پیدا کرده و سپس نواحی باقی‌مانده را با فرستنده‌های عمودی می‌پوشانیم.

۳.۴ پوشش چندضلعی توسط ۲- فرستنده‌های عمودی

به منظور پوشش چندضلعی توسط ۲- فرستنده‌های عمودی، می‌توان از کاهش مسئله به مسئله‌ی پوشش d -بازه (قسمت ۱.۴) استفاده کرد. روش کار به این صورت است که ابتدا مستطیل‌بندی $S'(P)$ برای

^۵Maximal

چندضلعی متعامد و x -یکنوای P را تشکیل می‌دهیم و متناظر با هر $s \in S'(P)$ یک پاره‌خط افقی دقیقاً موازی و هم اندازه با ضلع بالایی مستطیل s و درون آن قرار می‌دهیم. مجموعه تمام این پاره‌خط‌ها را با I نشان می‌دهیم. شکل ۵.۴ این مجموعه را نمایش داده است.

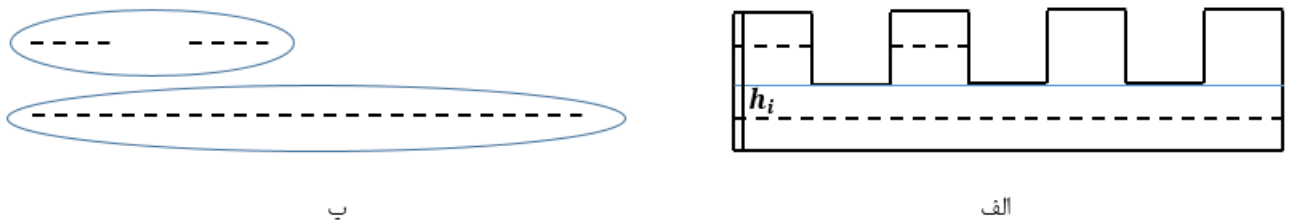


شکل ۵.۴: مجموعه I متناظر با $S'(P)$ به صورت خط‌چین مشخص شده است [۷].

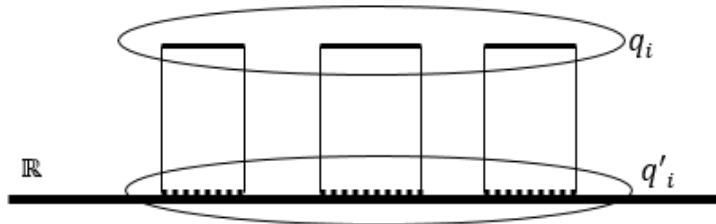
سپس مجموعه H (فرستنده‌های عمودی ماکزیمال) متناظر با چندضلعی P را تشکیل داده و به ازای هر $h_i \in H$ ، پاره‌خط‌های متناظر با نواحی که از $S'(P)$ می‌بیند را بدست آورده و در مجموعه $I(h_i)$ قرار می‌دهیم. به عنوان مثال شکل ۶.۴-الف مجموعه $I(h_i)$ متناظر با فرستنده h_i را نشان می‌دهد. به این ترتیب می‌توان مجموعه‌های $I(h_i)$ را مقدار دهی کرد.

مجموعه Q را در نظر بگیرید، این مجموعه در ابتدا تهی می‌باشد و اعضای آن مجموعه‌هایی با حداکثر سه عضو می‌باشند. به ازای هر $h_i \in H$ مجموعه $I(h_i)$ را از بالا به پایین پیمایش کرده و در هر سطح، اعضایی که با خط پیمایشگر برخورد دارند (بخاطر قدرت $k = 2$ فرستنده‌ها، در هر سطح حداکثر با سه عضو از $I(h_i)$ برخورد می‌کنیم) را در یک مجموعه دسته‌بندی کرده (مثلاً q_j) و به Q اضافه می‌کنیم. به عنوان مثال شکل ۶.۴ نشان می‌دهد که چگونه به دلیل وجود h_i دو عضو به مجموعه Q اضافه می‌شود.

به منظور استفاده از مسئله‌ی پوشش d -بازه، کافی است که با توجه به Q یک ورودی معتبر مانند Q' تشکیل دهیم. فرض کنید که در ابتدا Q' تهی می‌باشد. اعضای Q را یکی یکی بررسی کرده و به ازای هر $q_i \in Q$ یک ۳-بازه مانند q'_i می‌سازیم و به Q' اضافه می‌کنیم. به این ترتیب که مانند شکل ۷.۴، اعضای q'_i تصویر متناظر اعضای q_i بر روی محور x -ها می‌باشند و در صورتی که تعداد اعضای



شکل ۶.۴: شکل الف: مجموعه $I(h_i)$ متناظر با فرستنده h_i و شکل ب: اعضای از مجموعه Q که بخاطر h_i به این مجموعه اضافه شده‌اند [۷].



شکل ۷.۴: در این شکل اعضای q_i با خط و اعضای متناظر آن‌ها در Q' به صورت خطچین مشخص شده است [۷].

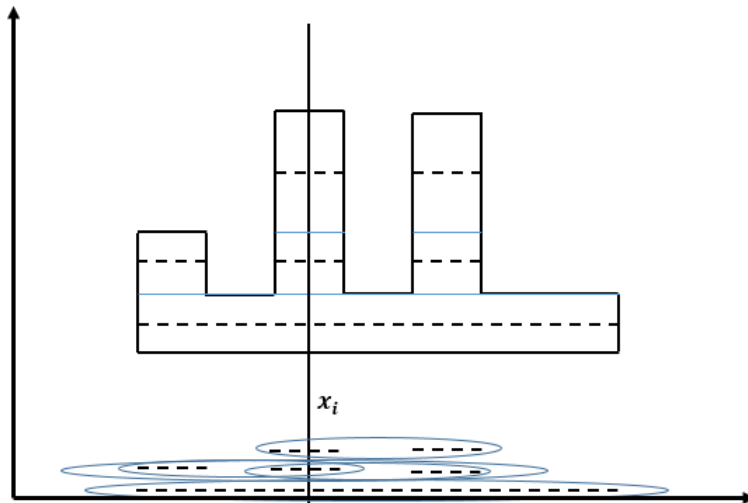
$q_i \in Q$ کمتر از سه باشد، هنگام ساخت عضو متناظر آن، به وسیله کپی کردن یکی از اعضا، تعداد بازه‌های موجود در Q' را به سه می‌رسانیم.

لم ۱.۴ فرض کنید مطابق آنچه در بالا گفته شد مجموعه Q' را با توجه به چندضلعی متعامد و x -یکنوای P تشکیل دهیم. به ازای هر جواب قابل قبول برای مسئله پوشش ۳- بازه با ورودی Q' ، می‌توان یک جواب قابل قبول با همان تعداد اعضا برای مسئله پوشش چندضلعی P توسط فرستنده‌های فقط عمودی ارائه کرد.

اثبات. فرض کنید که خروجی مسئله پوشش ۳- بازه تعدادی خطوط عمودی مانند $X = \{x_1, \dots, x_m\}$ باشد، به طوری که با تمام q'_i ها برخورد دارند (x_i با q'_j برخورد دارد در صورتی که با حداقل یکی از سه بازه موجود در q'_j برخورد داشته باشد).

با توجه به نحوه ساخت Q' ، بازه‌های موجود در هر q'_i دقیقاً تصویر متناظر با q_i بوده و تنها تفاوت این دو در مختصه y شان می‌باشد. بنابراین با توجه به مجموعه X ، جواب مسئله ما اشتراک این خطوط با چندضلعی می‌باشد. به عنوان مثال همان‌طور که در شکل ۸.۴ ملاحظه می‌کنید جواب بهینه مسئله پوشش ۳- بازه، خط x_i می‌باشد که در صورتی که در محل اشتراک آن با چندضلعی P یک ۲- فرستنده

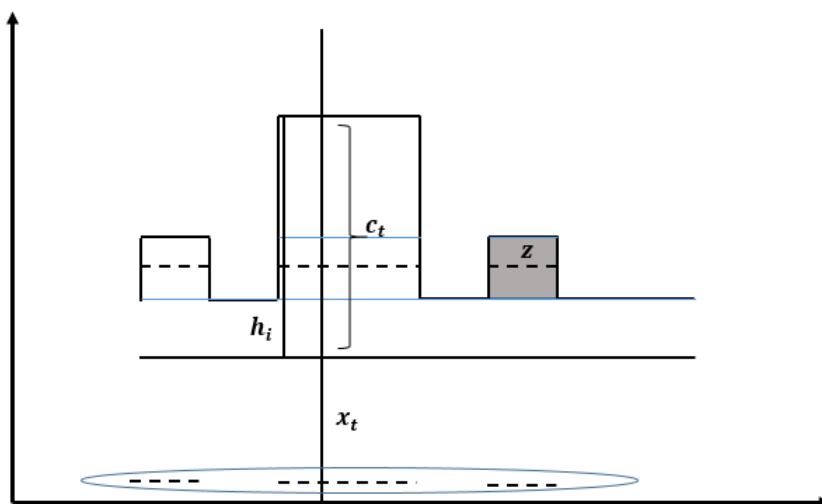
قرار دهیم، تمام چندضلعی پوشش داده می‌شود. فرض کنید جوابی که به این نحو از روی مجموعه X می‌سازیم، $C = \{c_1, \dots, c_m\}$ باشد.



شکل ۸.۴: رابطه بین جواب مسئله پوشش ۳-بازه و مسئله پوشش چندضلعی متعامد و x -یکنوا توسط فرستنده‌های فقط عمودی نشان داده شده است [۷].

حال ثابت می‌کنیم که برای مسئله پوشش چندضلعی متعامد و x -یکنوا توسط فرستنده‌های فقط عمودی یک جواب قابل قبول می‌باشد. فرض کنید که این طور نباشد (اثبات از طریق برهان خلف)، به این معنا که حداقل ناحیه‌ای مانند $s \in S(P)$ وجود دارد که توسط هیچ کدام از فرستنده‌ها پوشیده نشود. می‌دانیم که حتما فرستنده عمودی بیشینه‌ای مانند $h_i \in H$ وجود دارد که بتواند s را ببیند؛ بنابراین پاره خط افقی متناظر با s (این پاره خط را z می‌نامیم) عضو $I(h_i)$ می‌باشد. از آنجا که در هنگام ساخت Q تمام اعضای H را بررسی و سطح به سطح پیمایش کردیم، بنابراین حتما q_t وجود دارد که $z \in q_t$ باشد. طبق فرض، مجموعه X با هر $q'_i \in Q'$ برخورد دارد. بنابراین حتما با q'_i (که شامل تصویر اعضای q_t روی محور x می‌باشد) نیز برخورد دارند. خط گذرنده از q'_i را با x_t نشان می‌دهیم. در صورتی که x_t از تصویر پاره خط z بگذرد، مطمئنان امتداد آن نیز از z می‌گذرد و بنابراین فرستنده عمودی که حاصل اشتراک x_t و چندضلعی P می‌باشد (فرستنده c_t) نیز از ناحیه s می‌گذرد. در نتیجه ناحیه s حتما پوشش داده می‌شود. حال فرض کنید که x_t با تصویر z برخورد نداشته

باشد بلکه از یکی دیگر از اعضای q'_t بگذرد، در این صورت با توجه به نحوه ساخت Q ، مطمئناً بین ناحیه s و فرستنده c_t بیش از ۲ دیوار وجود ندارد، بنابراین باز هم این ناحیه توسط c_t دیده خواهد شد (شکل ۹.۴ را ملاحظه کنید) و این فرض که ناحیه دیده نشده‌ای مانند s وجود دارد نقض می‌شود. ■



شکل ۹.۴: در این شکل ناحیه s با رنگ خاکستری و پاره‌خط متناظر با این ناحیه با حرف z مشخص شده است. همان‌طور که در این تصویر نیز مشاهده می‌شود در صورتی که x_t با تصویر z بر محور x برخورد نداشته باشد، باز هم فرستنده متناظر با آن (c_t) حتماً توانایی دیدن ناحیه s را خواهد داشت [۷].

لم ۲.۴ فرض کنید که چندضلعی متعامد x - یکنوای P داده شده است و متناسب با آن و بر اساس آنچه در بالا گفته شد مجموعه Q' را تشکیل دهیم. در این صورت اگر $C = \{c_1, \dots, c_m\}$ یک جواب قابل قبول برای مسئله پوشش چندضلعی P توسط فرستنده‌های فقط عمودی باشد، می‌توان یک جواب قابل قبول با همان تعداد اعضا برای مسئله پوشش ۳- بازه روی مجموعه Q' ارائه کرد.

اثبات. فرض کنید که Q' و چندضلعی P هر دو در یک دستگاه مختصات حضور داشته باشند، به طوری که بازه‌های موجود در هر $q'_i \in Q'$ دقیقاً تصویر بازه‌های متناظر در $q_i \in Q$ باشد (مانند شکل ۸.۴). در این صورت اگر اعضای مجموعه C را امتداد دهیم تا محور x ها را قطع کنند (این مجموعه خطوط را X می‌نامیم)، حتماً با هر $q'_i \in Q'$ برخورد خواهد داشت. اثبات این ادعا از طریق برهان خلف امکان پذیر می‌باشد.

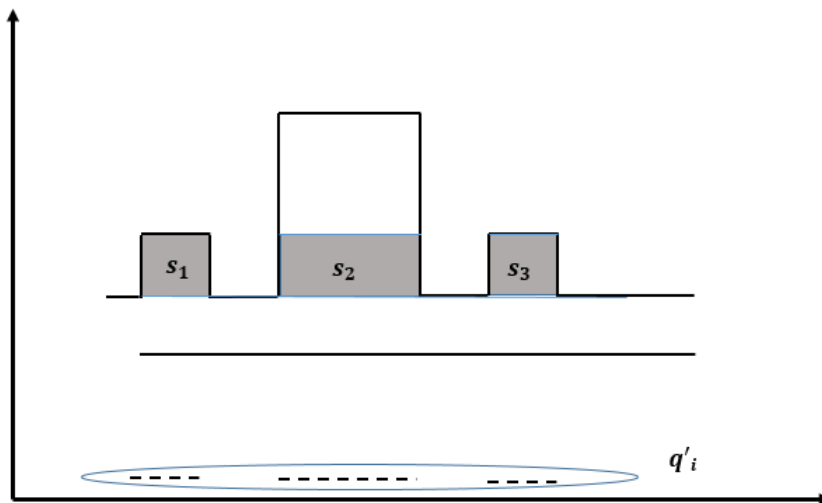
فرض کنید که C تمام چندضلعی P را پوشش دهد ولی مجموعه X متناظر با آن، با تمامی اعضای Q' برخورد نداشته باشد. فرض کنید $q'_i \in Q'$ عضوی از Q' با این خاصیت باشد. نواحی متناظر با بازه‌های q'_i که درون $S(P)$ قرار دارند را در نظر بگیرید:

۱. اگر تعداد این نواحی ۳ باشد، با توجه به نحوه مستطیل‌بندی $S'(P)$ می‌دانیم که یال‌های عمودی هر ناحیه مستطیل شکل، بخشی از یال‌های عمودی چندضلعی P می‌باشد. بنابراین با توجه به دید $k_i = 2$ فرستنده‌ها و این که بین هر دو ناحیه مستطیل شکل هم‌سطح حتماً دو دیوار (بخشی از یال‌های عمودی چندضلعی) وجود دارد، پس برای پوشش ناحیه وسط (ناحیه s_2 درون شکل ۱۰.۴)، یا باید درون این ناحیه و یا یکی از نواحی کناری فرستنده‌ای واقع شده باشد. در هر دو حالت، امتداد این فرستنده مطمئناً با یکی از سه بازه موجود در q'_i برخورد خواهد داشت.

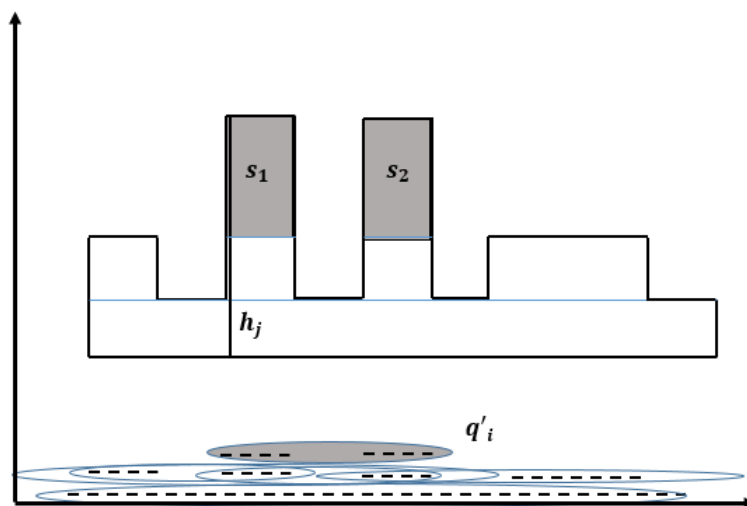
۲. اگر تعداد این نواحی ۲ باشد، با توجه به نحوه ساخت q_i ها از روی نواحی قابل دید فرستنده‌های عمودی عضو H ، در صورتی که h_j در یک سطح فقط دو ناحیه را ببیند، به این معناست که حداقل یکی از این دو ناحیه حتماً نسبت به نواحی این سطح، سمت چپ‌ترین یا سمت راست‌ترین ناحیه می‌باشند (شکل ۱۱.۴ را ملاحظه کنید). فرض کنید ناحیه سمت چپ را با s_1 و سمت راست را با s_2 نشان دهیم. در صورتی که s_1 سمت چپ‌ترین ناحیه این سطح باشد، برای دیدن این ناحیه حتماً باید فرستنده‌ای عمودی در یکی از نواحی s_1 یا s_2 حضور داشته باشد که این امر تضمین می‌کند که امتداد این فرستنده با حتماً با q'_i برخورد دارد که خلاف فرض است. در صورتی که s_2 سمت راست‌ترین ناحیه این سطح باشد نیز با استدلال مشابه فرض اولیه نقض می‌شود.

۳. اگر تعداد این نواحی ۱ باشد، این حالت نیز صرفاً حالت خاصی از ۲ ناحیه می‌باشد و به همان ترتیب می‌توان استدلال کرد.

بنابراین فرض اولیه نقض می‌شود و مجموعه X یک جواب قابل قبول برای مسئله ۳- بازه می‌باشد. ■



شکل ۱۰.۴: با توجه به قابل قبول بودن مجموعه C برای پوشش چندضلعی، حتما امتداد فرستنده‌ای که ناحیه s_2 را پوشش می‌دهد، با q_i برخورد خواهد داشت [۷].



شکل ۱۱.۴: با توجه به قابل قبول بودن مجموعه C ، حتما فرستنده‌ای در این مجموعه قرار دارد که درون یکی از s_1 یا s_2 باشد [۷].

با توجه به لم ۲.۴، به ازای هر جواب قابل قبول که برای مسئله پوشش چندضلعی متعامد و x -یکنوای P وجود دارد، یک جواب با همان تعداد می‌توان برای مسئله پوشش ۳-بازه با ورودی Q' (توجه کنید که روش ساخت این مجموعه همان‌گونه که ذکر شد براساس چندضلعی P و مجموعه فرستنده‌های عمودی H می‌باشد) ارائه کرد. بنابراین داریم که جواب بهینه مسئله پوشش ۳-بازه قطعا کمتر مساوی جواب بهینه مسئله پوشش چندضلعی توسط فرستنده‌های فقط عمودی می‌باشد

و در نتیجه از آنجا که برای مسئله ۳- بازه یک راه حل ۳- تقریب وجود دارد [۱۶]، بنابراین برای مسئله‌ی مورد نظر، نیز با استناد به لم ۱.۴ یک جواب ۳ تقریب وجود خواهد داشت.

۴.۴ الگوریتم برای حل مسئله ۲- MCSC

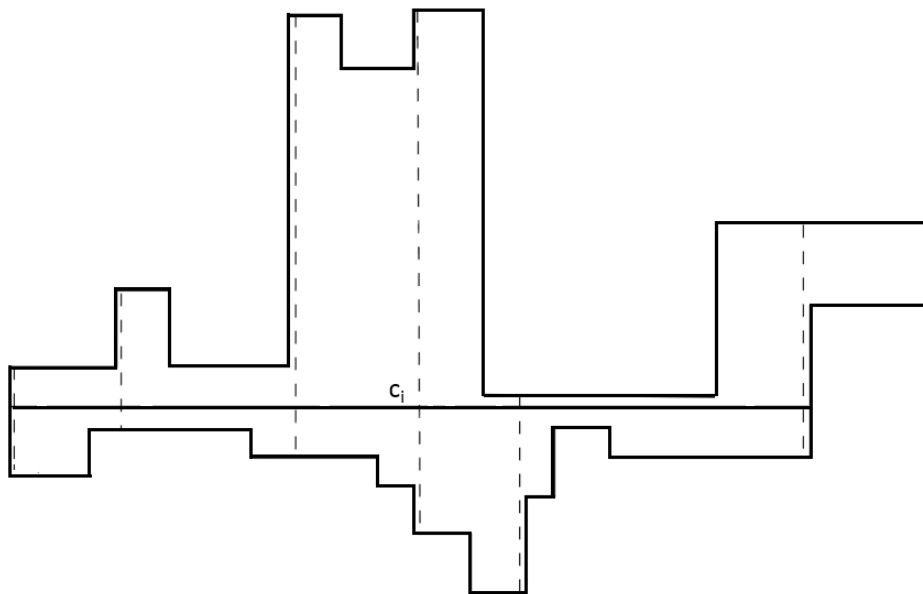
در این قسمت یک الگوریتم تقریبی با ضریب تقریب ۳ برای حل مسئله‌ی تعریف شده، در صورتی که هر دو نوع فرستنده‌ی افقی و عمودی مجاز باشند، ارائه می‌دهیم. این الگوریتم ۳- تقریب برای حالتی است که جواب بهینه مسئله‌ی پوشش توسط فقط فرستنده‌های عمودی را داشته باشیم. در صورتی که از الگوریتم تقریبی ارائه شده در ۳.۴ استفاده کنیم، ضریب تقریب الگوریتم پوشش توسط فرستنده‌های لغزان ۷ خواهد بود.

فرض کنید یک چندضلعی متعامد P و عدد ثابت $k = 2$ به عنوان ورودی داده شده اند. هدف پوشش P توسط کمترین تعداد ۲- فرستنده لغزان است. این مسئله را ۲- MCSC می‌نامیم. چندضلعی رویت‌پذیری فرستنده c در حالت k - فرستنده را با $V^k(c)$ نشان می‌دهیم. دو فرستنده c_1 و c_2 را در نظر بگیرید. c_1 توسط c_2 مغلوب می‌شود اگر تنها اگر $V^k(c_1) \subset V^k(c_2)$ باشد.

به ازای هر فرستنده‌ی افقی c_i درون P مجموعه‌ی $\delta(c_i)$ را مجموعه‌ی فرستنده‌های عمودی درون P که تعداد آن‌ها کمینه بوده و اجتماع چندضلعی رویت‌پذیری آن‌ها با قدرت $k = 0$ ، $V^0(c_i)$ را مغلوب می‌کند، می‌نامیم. به عبارت دیگر، مجموعه $\delta(c_i)$ را به عنوان زیر مجموعه‌ای از H مانند H' در نظر می‌گیریم که:

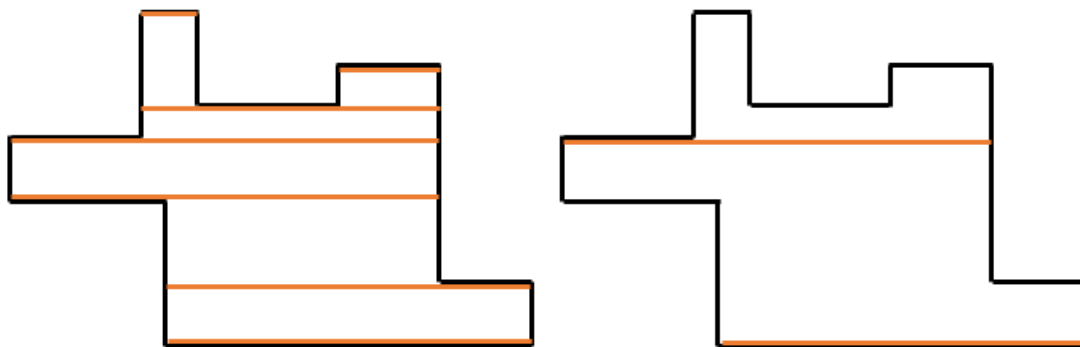
$$V^0(c) \subseteq \bigcup_{c' \in H'} V^0(c') \quad |H'| \text{ is minimized} \quad (1.4)$$

فرض کنید C_H مجموعه‌ی پاره‌خط‌های افقی بیشینه درون P است که از امتداد دادن یال‌های افقی به دست آمده‌اند. پاره‌خط‌های درون C_H را از چپ به راست براساس نقطه‌ی انتهایی سمت چپشان مرتب کنید. در صورتی که بیش از دو پاره‌خط درون C_H وجود داشته باشند که اشتراک چندضلعی



شکل ۱۲.۴: در این شکل مجموعه δ متناظر با فرستنده افقی c_i به صورت خط‌چین به نشان داده شده است [۷].

دیدپذیر آن‌ها غیر تهی باشد، به جز سمت چپ‌ترین و سمت راست‌ترین آن‌ها، بقیه را از C_H حذف کنید (شکل ۱۳.۴ را ملاحظه کنید).



شکل ۱۳.۴: مجموعه C_H قبل و بعد از خلاصه شدن [۷].

در الگوریتم ارائه شده، ابتدا با استفاده از مجموعه C_H قسمتی از P را پوشش داده و سپس نواحی پوشیده نشده را توسط پاره‌خط‌های فقط عمودی پوشش می‌دهیم. منظور از پوشش دادن، پوشش دادن در حالت ۲- فرستنده است.

۱.۴.۴ الگوریتم

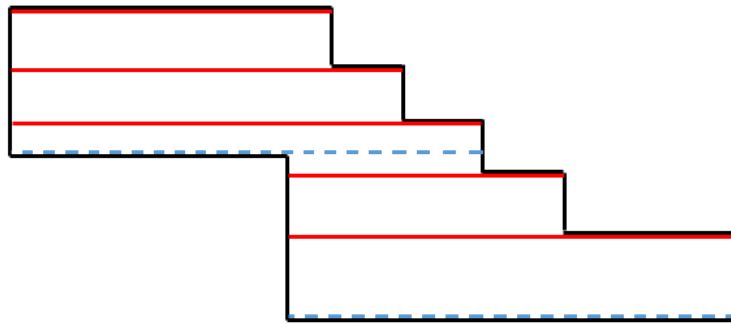
۱. مجموعه‌ی C_H را تشکیل دهید.

۲. CH را از چپ به راست پیمایش کنید. به ازای هر $c_i \in C_H$ ، $\delta(c_i)$ را تشکیل دهید. اگر ناحیه‌ای مانند $r \in S(P)$ وجود داشته باشد که پوشیده نشده و توسط فقط c_i و برخی از پاره‌خط‌های عمودی $\delta(c_i)$ پوشش داده شود، آنگاه c_i را گزارش کنید. همچنین اگر r فقط توسط c_i و c_{i+1} و برخی از اعضای $\delta(c_i)$ پوشش داده شود، آنگاه هر دو c_i و c_{i+1} را گزارش کنید.

۳. CH را از چپ به راست پیمایش کنید. در صورتی که c_i و c_{i+1} وجود داشته باشند که هیچ کدامشان در گام ۲ گزارش نشده‌اند، مجموعه t_i را تشکیل دهید (شکل ۱۴.۴ را ملاحظه کنید). t_i شامل تمام پاره‌خط‌های افقی بیشینه‌ای است که بین نقطه‌ی انتهایی سمت چپ c_i و نقطه‌ی انتهایی سمت راست c_{i+1} وجود دارند. (در این جا نیز مانند تشکیل مجموعه C_H عمل می‌کنیم، به این منظور که اعضای t_i را از طریق امتداد یال‌های افقی بدست می‌آوریم). پس از تشکیل t_i این مجموعه را خلاصه می‌کنیم، به این صورت که هر عضو این مجموعه که مغلوب c_i یا c_{i+1} شود را از مجموعه حذف می‌کنیم. پس از خلاصه کردن t_i به ازای هر $c \in t_i$ مجموعه $\delta(c)$ را تشکیل می‌دهیم و بررسی می‌کنیم که آیا ناحیه پوشش داده نشده‌ای مثل $r \in V^2(c)$ وجود دارد که توسط هیچ فرستنده عمودی خارج از مجموعه $\delta(c)$ دیده نشود. در صورت وجود چنین ناحیه‌ای c را گزارش می‌دهیم و نیازی به چک کردن بقیه اعضای t_i نیز نخواهد بود.

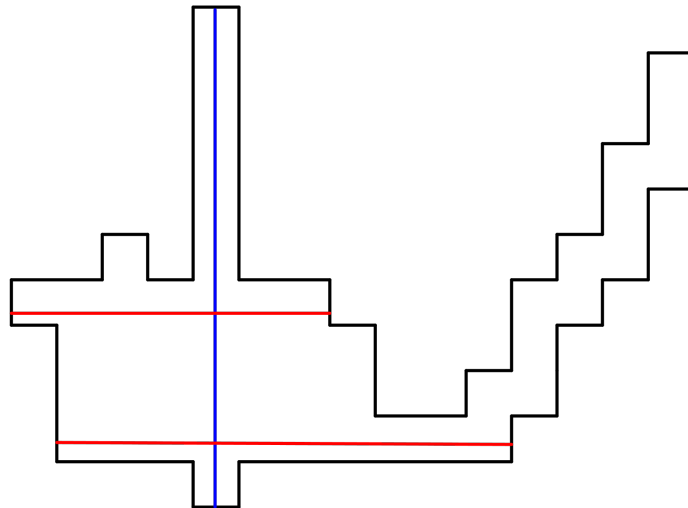
۴. بقیه‌ی نواحی پوشش داده نشده را توسط فرستنده‌های فقط عمودی پوشش دهید.

با استفاده از این الگوریتم یک جواب برای مسئله ۲ - $MCSC$ بدست می‌آید. در صورت بهینه بودن جواب پوشش چندضلعی توسط ۲- فرستنده‌های فقط عمودی، جواب الگوریتم بالا حداکثر ۳ برابر جواب بهینه است. مثال محکم برای این الگوریتم در شکل ۱۵.۴ آمده است. جواب بهینه این



شکل ۱۴.۴: در این شکل c_i و c_{i+1} به صورت خط‌چین و مجموعه t_i متناظر با این دو با خطوط کامل مشخص شده است.

چندضلعی تنها فرستنده آبی رنگ است اما الگوریتم دو فرستنده قرمز رنگ را به همراه فرستنده آبی رنگ گزارش می‌کند.



شکل ۱۵.۴: نمونه‌ای از یک چندضلعی که دارای جواب بهینه ۱ است اما الگوریتم ما سه فرستنده را گزارش می‌دهد.

۲.۴.۴ تحلیل ضریب تقریب الگوریتم

حال ضریب تقریب الگوریتم را بررسی می‌کنیم. فرض کنید جواب بهینه مسئله OPT است. فرستنده‌های OPT را به دو دسته فرستنده‌های افقی بهینه H_{OPT} و فرستنده‌های عمودی بهینه V_{OPT} تقسیم می‌کنیم. از آنجایی که فرستنده‌های افقی گزارش شده در هر مرحله مستقل از یکدیگرند، فرستنده‌های افقی جواب بهینه را به صورت $H_{OPT} = H_{OPT}^1 + H_{OPT}^2 + H_{OPT}^3 + H_{OPT}^4$ در نظر گرفته و جواب بهینه

عمودی را به صورت $V_{OPT} = V_{OPT}^{۳۱} + V_{OPT}^{۳۲} + V_{OPT}^{۴} + V_{OPT}^{۵}$ در نظر می‌گیریم، که به صورت زیر تعریف می‌شوند.

برای H_{OPT} داریم:

۱. $H_{OPT}^۱$ ، فرستنده‌های افقی جواب بهینه‌ای که در گام ۱ هنگام ساختن C_H حذف شده و در گام

۲ به دلیل گزارش c_i و $c_{i+۱}$ به‌طور کامل پوشش داده شده‌اند.

۲. $H_{OPT}^۲$ ، فرستنده‌های افقی جواب بهینه‌ای که در گام ۱ هنگام ساختن C_H حذف شده و در گام

۲ قسمتی از آن‌ها پوشش داده شده‌اند.

۳. $H_{OPT}^۳$ ، فرستنده‌های افقی که عضو جواب بهینه هستند و در گام ۲ نیز گزارش می‌شوند.

۴. $H_{OPT}^۴$ ، فرستنده‌های افقی بهینه‌ای که نه خودشان در گام ۲ گزارش شده‌اند و نه هیچ قسمتی از

آن‌ها در گام ۲ پوشش داده شده است. بلکه در گام ۴ توسط فرستنده‌های عمودی پوشش داده

شده‌اند.

برای V_{OPT} داریم:

۱. $V_{OPT}^{۳۱}$ ، فرستنده‌های عمودی جواب بهینه‌ای که در گام ۲ توسط یک دوربین افقی پوشش داده

شده‌اند.

۲. $V_{OPT}^{۳۲}$ ، فرستنده‌های عمودی جواب بهینه‌ای که در گام ۲ توسط دو دوربین افقی پوشش داده

شده‌اند.

۳. $V_{OPT}^۴$ ، فرستنده‌های عمودی که در گام ۳ توسط یک فرستنده افقی پوشش داده شده‌اند.

۴. $V_{OPT}^۵$ ، فرستنده‌های عمودی جواب بهینه‌ای که در گام ۴ گزارش داده می‌شوند.

بنابراین جواب بدست آمده الگوریتم ارائه شده به شرح زیر است.

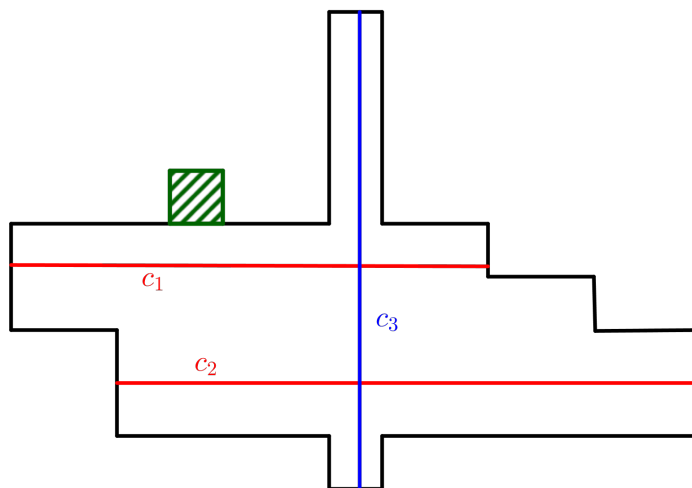
• زمانی که ناحیه‌ی r در جواب بهینه توسط یک فرستنده افقی c پوشیده می‌شود.

- اگر برای پوشش r در گام ۲ به جای c ، دو فرستنده افقی گزارش شوند، آنگاه تعداد فرستنده‌های گزارش شده حداکثر $2H_{OPT}^r + 2H_{OPT}^l$ است.

- اگر در گام ۲ تنها قسمتی از r پوشیده شود، آنگاه تعداد فرستنده گزارش شده حداکثر H_{OPT}^r است.

- اگر در گام ۳ توسط یک افقی غیر از c پوشیده شود، بدلیل آنکه c در گام ۲ گزارش نشده و قسمتی از آن هم پوشیده نشده است، آنگاه تعداد فرستنده گزارش شده حداکثر H_{OPT}^f است.

- اگر در r در گام ۴ توسط دو فرستنده عمودی پوشیده شود، آنگاه در بدترین حالت تعداد فرستنده‌های گزارش شده $2H_{OPT}^f + 2H_{OPT}^v$ است.



شکل ۱۶.۴: الگوریتم دو فرستنده افقی را به جای یک فرستنده عمودی گزارش می‌کند.

• زمانی که ناحیه‌ی r در جواب بهینه توسط یک فرستنده عمودی c پوشیده می‌شود.

- اگر برای پوشش r در گام ۲ دو فرستنده افقی گزارش شوند، آنگاه حداکثر تعداد فرستنده‌های گزارش شده $2V_{OPT}^r$ است.

- اگر برای پوشش r در گام ۲ یک فرستنده افقی گزارش شود، آنگاه حداکثر تعداد فرستنده‌های گزارش شده V_{OPT}^{r1} است.

- اگر به جای c در گام ۳، یک فرستنده افقی گزارش شود، آنگاه تعداد دوربین گزارش شده V_{OPT}^c است.

- اگر در گام ۴، c گزارش شود آنگاه تعداد فرستنده گزارش شده V_{OPT}^c است.

در نتیجه تعداد فرستنده‌های گزارش شده در گام ۲ حداکثر برابر با $2(H_{OPT}^1 + H_{OPT}^2 + V_{OPT}^{22})$ و تعداد فرستنده‌های گزارش شده در گام ۳ حداکثر برابر با $H_{OPT}^3 + V_{OPT}^{31}$ و تعداد فرستنده‌های گزارش شده در گام ۴ حداکثر برابر با $2(H_{OPT}^4 + H_{OPT}^5) + V_{OPT}^5$ هستند. بنابراین داریم:

$$\begin{aligned} \text{Out Put} &\leq (2H_{OPT}^1 + H_{OPT}^2 + 2H_{OPT}^3 + V_{OPT}^{31} + 2V_{OPT}^{32}) + (H_{OPT}^4 + V_{OPT}^4) + (2H_{OPT}^5 + \\ &2H_{OPT}^4 + V_{OPT}^5) \leq (2H_{OPT}^1 + 3H_{OPT}^2 + 2H_{OPT}^3 + 3H_{OPT}^4) + (V_{OPT}^{31} + 2V_{OPT}^{32} + V_{OPT}^4 + V_{OPT}^5) \\ &\leq 3H_{OPT} + 2V_{OPT} \leq 3OPT \end{aligned}$$

۳.۴.۴ تحلیل پیچیدگی زمانی الگوریتم

برای تشکیل C_H کافی است، اضلاع عمودی و افقی چندضلعی را به ترتیب از چپ به راست و از بالا به پایین مرتب کنیم. سپس با یکبار پیمایش کردن فرستنده‌های ماکزیمال بدست می‌آیند. بنابراین مجموعه C_H در زمان $O(n \log n)$ قابل محاسبه است. برای حذف کردن فرستنده‌های افقی که حداقل یک نقطه مشترک را پوشش می‌دهند، کافی است نقاط انتهایی پاره‌خط‌های افقی را بررسی کنیم. این کار در زمان $O(n)$ قابل انجام است. بنابراین C_H در زمان $O(n \log n)$ بدست می‌آید.

برای نگه داشتن نواحی پوشش داده شده، چندضلعی را با امتداد دادن اضلاع آن مستطیل‌بندی

می‌کنیم. نواحی بدست آمده را در نظر بگیرید. تعداد این نواحی $O(n^2)$ است. حال یک ماتریس را در نظر بگیرید که دارای $O(n^2)$ ستون و $O(n)$ سطر است. هر درایه می‌تواند ۰ یا ۱ باشد. اگر فرستنده i -ام ناحیه j -ام را پوشش دهد، پس درایه $(i, j) = 1$ و در غیر اینصورت $(i, j) = 0$ است. به علاوه برای هر ناحیه یک برچسب قرار می‌دهیم که یک است اگر پوشش داده شده باشد و در غیر اینصورت صفر است. در ابتدا همه برچسب‌ها صفر هستند و با پیدا کردن چندضلعی رویت‌پذیری هر فرستنده در حالت ۲- فرستنده، درایه‌های ماتریس را پر می‌کنیم. حال با کمک این ماتریس و برچسب‌ها گام ۲ را در زمان $O(n^3)$ محاسبه می‌کنیم. در ابتدا تمام درایه‌های ماتریس ۰ هستند. در هر مرحله که فرستنده‌ای گزارش می‌شود، سطر متناسب با آن به روز خواهد شد. برای گام ۲ کافی است چندضلعی را از چپ به راست پیمایش کرده و در پیچیدگی زمانی این الگوریتم $O(n^3)$ است.

با توجه به x -یکنوا بودن P ، داریم که $\sum_{c_i \in C_H} |t_i| = O(n)$ و تمام این‌ها در زمان $O(n \log n)$ توسط یک‌بار پیمایش از بالا به پایین بدست می‌آیند. فرض کنید که a_i و b_i به ترتیب سمت چپ‌ترین و سمت راست‌ترین نقاط نواحی پوشش داده نشده مجموعه t_i باشند. همچنین دو خط عمودی که از a_i و b_i می‌گذرند را به ترتیب با v_{a_i} و v_{b_i} نشان می‌دهیم. در این صورت به ازای هر $r \in V^2(c_i)$ ، پاره خط افقی گذرنده از آن مانند L_r را در نظر گرفته و آن را تا جایی امتداد می‌دهیم که در هر سمت با دو یال عمودی برخورد داشته باشد. اگر فرستنده‌ای مثل $c' \in t_i$ وجود داشته باشد که بتواند L_r را پوشش دهد (به این معنا که $lp_{c'}$ سمت چپ L_r و $lr_{c'}$ سمت راست L_r باشد)، c' را گزارش می‌کنیم.

با توجه به این‌که تعداد کل t_i ‌ها از مرتبه $O(n)$ و تعداد کل نواحی از مرتبه $O(n^2)$ می‌باشد، بنابراین اگر به ازای هر ناحیه مثل r ، L_r مربوطه را بدست آوریم، بنابراین در زمان $O(n^3)$ می‌توان تمام L_r ‌ها (تعدادشان حداکثر برابر با تعداد نواحی می‌باشد) را استخراج کرده و سپس در زمان $O(n^3)$ می‌توان آن‌ها را با تمام t_i ‌ها بررسی کرده و گام ۳ در زمان چندجمله‌ای انجام می‌شود. در نتیجه بدلیل آنکه در قسمت ۳.۴ ثابت کردیم، گام ۴ در زمان چندجمله‌ای انجام می‌شود می‌توانیم قضیه ۳.۴ را نتیجه

بگیریم.

قضیه ۳.۴ یک الگوریتم تقریبی با ضریب تقریب ۳ (با فرض داشتن جواب بهینه مسئله پوشش توسط فقط فرستنده‌های عمودی) برای حل مسئله پوشش موزه‌های هنر متعامد و x -یکنوا توسط کمترین تعداد ۲- فرستنده‌های لغزان وجود دارد.

بنابر قضیه ۳.۴ در صورت داشتن جواب بهینه برای مسئله پوشش چندضلعی متعامد و x -یکنوا توسط فرستنده‌های فقط عمودی، می‌توان به ضریب تقریب ۳ رسید. با استفاده از قسمت ۳.۴ مسئله پوشش توسط فرستنده‌های فقط عمودی با ضریب تقریب ۳ حل می‌شود، بنابراین هزینه گام ۴ سه برابر شده و به ضریب تقریب ۷ می‌رسیم.

$$\begin{aligned} \text{Out Put} &\leq (2H_{OPT}^1 + H_{OPT}^2 + 2H_{OPT}^3 + V_{OPT}^{31} + 2V_{OPT}^{32}) + (H_{OPT}^4 + V_{OPT}^4) + 3(2H_{OPT}^2 + \\ &2H_{OPT}^4 + V_{OPT}^5) \leq (2H_{OPT}^1 + 7H_{OPT}^2 + 2H_{OPT}^3 + 6H_{OPT}^4) + (V_{OPT}^{31} + 2V_{OPT}^{32} + V_{OPT}^4 + 3V_{OPT}^5) \\ &\leq 7H_{OPT} + 3V_{OPT} \leq 7OPT \end{aligned}$$

لازم به ذکر است کار انجام شده در این گزارش پیشرفت، کاری مشترک با شادروان آقای محمدامین بیروتی است و از رساله‌ی ایشان [۷] در نوشتن استفاده شده است. لازم به ذکر است ایشان الگوریتم ۷- تقریب را به الگوریتم ۶- تقریب بهبود بخشیدند.

۵.۴ نتیجه‌گیری

در این فصل مسئله پوشش یک چندضلعی متعامد و x -یکنوا توسط کمترین تعداد ۲- فرستنده لغزان را مطرح کرده و یک الگوریتم تقریبی برای آن ارائه دادیم. بنابر نتایجی که پس از این الگوریتم بدست آمدند این مسئله برای $k > 0$ درون یک چندضلعی متعامد و یکنوا ان پی سخت است و یک الگوریتم با ضریب تقریب ثابت برای آن ارائه شده است. برای کار آینده می‌توان به بررسی این مسئله در زمینه

الگوریتم‌های تصادفی^۶ و الگوریتم‌های با پارامتر ثابت^۷ پرداخت. همچنین سوال جالب و مهم در این زمینه این است که آیا می‌توان یک چندضلعی متعامد ساده را در زمان چندجمله‌ای توسط کمترین تعداد ۰- فرستنده لغزان پوشاند.

^۶Randomized Algorithm

^۷Fixed Parameter Algorithm

فصل ۵

تمیز کردن چندضلعی‌های متعامد توسط ربات‌های لغزان

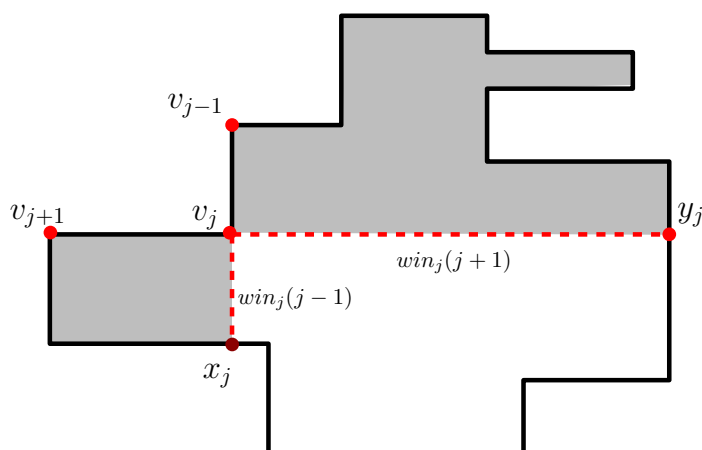
۱.۵ تعریف مسئله

در این فصل به بررسی مسئله‌ی تمیزکردن موزه‌های هنر متعامد توسط ربات‌های لغزان می‌پردازیم. مسئله‌ی مهم پس از بدست آوردن مجموعه‌ای از دوربین‌های لغزان که بتوانند یک موزه‌ی هنر متعامد را نگهداری کنند، این است که چگونه این دوربین‌ها در راستای پاره‌خط‌های متناظرشان حرکت کنند تا بتوانند موزه‌ی هنر را نگهداری کنند. این نکته مهم که در کارهای انجام شده قبلی در نظر گرفته نشده‌است باعث بررسی مسئله‌ی طراحی مسیر حرکت برای دسته‌ای از ربات‌های لغزان برای نگهداری کردن یک موزه‌ی هنر شد. در این مسئله فرض بر این است که موزه‌ی هنر دارای تعدادی دزد است (به عبارت دیگر آلوده است) و هدف بدست آوردن دنباله‌ای از مسیرهای حرکت برای ربات‌هاست به گونه‌ای که در پایان حرکت آن‌ها، تمام دزدهای داخل موزه‌ی هنر پیدا شوند (به عبارت دیگر موزه‌ی هنر تمیز شود). لازم به ذکر است که پیدا شدن دزد به معنای دیده شدن آن توسط یک ربات است. یک چندضلعی متعامد به عنوان ورودی داده شده است. فرض بر این است که تعدادی دزد که دارای سرعت حرکت بسیار زیاد هستند داخل این چندضلعی در حال حرکتند که دارای مکان نامشخص

هستند. هدف یافتن مسیر حرکت برای یک یا چند ربات لغزان است که بتوانند دزدها را ببینند. در مسئله‌ی طراحی مسیر حرکت برای بیش از یک ربات که قرار است یک کار مشترک انجام دهند، نکته مهم پیچیدگی ارتباط برقرار کردن میان ربات‌ها است. هر چه ارتباط میان ربات‌ها کمتر باشد الگوریتم کاراتری خواهیم داشت.

تعریف زیر در الگوریتم‌های ارائه شده در این فصل مورد نیاز است.

تعریف ۱.۵ فرض کنید v_i یک رأس انعکاسی از P بوده و دو یال متصل به آن $\overline{v_{j-1}v_j} = e_{j-1}$ و $\overline{v_jv_{j+1}} = e_j$ باشند. این دو یال می‌توانند در داخل P امتداد پیدا کنند تا به مرز P برخورد کنند. امتداد این دو یال را پنجره‌های v_j می‌نامیم. در شکل ۱.۵ دو پاره‌خط $\overline{v_jx_j}$ و $\overline{v_jy_j}$ پنجره‌های رأس v_j هستند.



شکل ۱.۵: دو پنجره رأس v_j به صورت خط‌چین نشان داده شده‌اند.

۱.۱.۵ کارهای انجام شده

در این فصل مسئله‌ی تمیز کردن یک موزه‌ی هنر توسط تعدادی ربات لغزان را در دو حالت تمیز کردن یک موزه‌ی هنر متعامد ساده توسط تعدادی ربات‌های لغزان (۲.۵) و تمیز کردن یک موزه‌ی هنر متعامد و x -یکنوا توسط کمترین تعداد ربات‌های لغزان (۳.۵) بررسی می‌کنیم. نتیجه‌ی بدست آمده از ۲.۵ در مجله Theoretical Computer Science [۳۸] به چاپ رسیده و نتیجه‌ی بدست آمده

از ۳.۵ به Information Processing Letters [۳۹] ارسال شده است.

- Salma Sadat Mahdavi and Mohammad Ghodsi. *Clearing an Orthogonal Polygon to Find the Evaders*, Theoretical Computer Science, 2020.
- Salma Sadat Mahdavi and Mohammad Ghodsi. *Clearing an x -monotone Orthogonal Polygon to Find Evaders*, Submitted to IPL 2018.

۲.۵ تمیز کردن یک موزه‌ی هنر متعامد توسط ربات‌های لغزان

۱.۲.۵ تعریف مسئله

فرض کنید یک چندضلعی متعامد و ساده، P ، و یک مجموعه از پاره‌خط‌ها، S ، به عنوان ورودی داده شده است. مجموعه S به گونه‌ای است که اجتماع چندضلعی‌های رویت‌پذیر لغزان پاره‌خط‌های آن برابر با P می‌باشد. P دارای تعدادی دزد است که با سرعت زیاد در حال حرکت بوده و دارای مکان مشخصی نیستند. هدف طراحی مسیر حرکت برای دسته‌ای از ربات‌های لغزان است که روی اعضای S حرکت کنند و بتوانند P را تمیز کنند (به عبارت دیگر، تمام دزدها را پیدا کنند).

ما فرض می‌کنیم که محیط برای ربات‌ها شناخته شده است و ربات‌ها توانایی فرستادن پیغام به تمام ربات‌های دیگر را دارند. بنابراین ربات‌ها می‌توانند با یکدیگر ارتباط داشته باشند. بهترین نتیجه الگوریتم این قسمت این است که اگر S مجموعه $MCSC$ (کمترین تعداد دوربین لغزان) باشد که تمام P را پوشش می‌دهد، آنگاه الگوریتم ما P را توسط کمترین تعداد ربات‌های لغزان که تنها در راستای پاره‌خط‌های S حرکت می‌کنند، تمیز می‌کند. ما این الگوریتم را پیاده‌سازی کرده و یک نمونه اجرای آن را در قسمت ۴.۲.۵ قرار دادیم.

۲.۲.۵ الگوریتم ارائه شده

مجموعه‌ی تمام رأس‌های P در جهت پادساعتگرد را $V(P) = \{v_1, v_2, \dots, v_n\}$ و مجموعه‌ی تمام رأس‌های انعکاسی P را $V_{ref}(P)$ می‌نامیم. فرض کنید در حالت کلی هیچ چهار رأس انعکاسی همراستا نیستند.

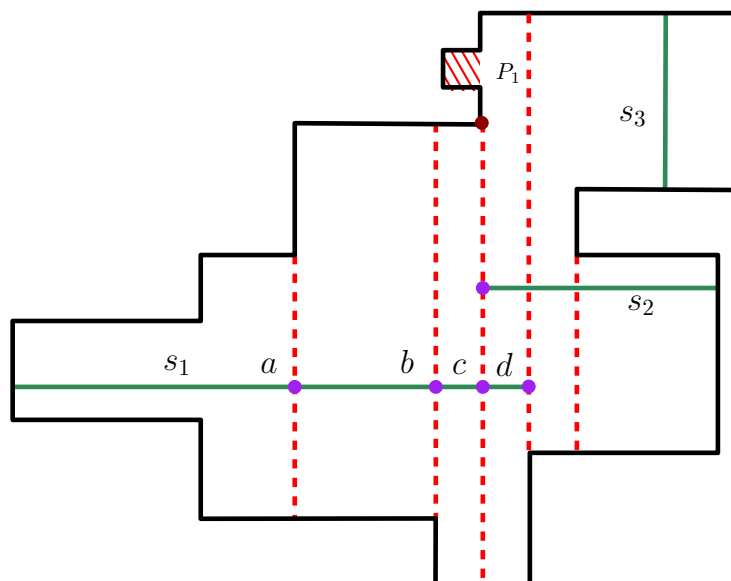
به ازای هر $s_i \in S$ یک ربات $r_i \in R$ در نظر می‌گیریم که در راستای s_i حرکت می‌کند. هر ربات لغزان r_i دارای سه حافظه $D_i(1), D_i(2), D_i(3)$ است. $D_i(1)$ بیانگر زیرچندضلعی تمیزی از P است که r_i قسمتی از آن را تمیز کرده است. $D_i(2)$ بیانگر زیرچندضلعی از P است که باید توسط r_i به طور کامل و یا قسمتی از آن تمیز شود. $D_i(3)$ بیانگر زیرچندضلعی از P است که باید تمیز شود تا r_i بتواند به حرکت خود ادامه دهد. برای ارتباط میان ربات‌ها، این قابلیت را در نظر می‌گیریم که می‌توانند برای یکدیگر پیامی توسط سیگنال ارسال کنند.

تعریف ۲.۵ یک نقطه‌ی بحرانی زمانی اتفاق می‌افتد که یک نقطه‌ی بحرانی زمانی اتفاق می‌افتد که ربات r_i یک رأس انعکاسی را ببیند، به یکی از نقاط انتهایی پاره‌خط s_i برسد و یا یک ربات دیگر را که منتظر است، ببیند.

در الگوریتم ما ربات‌ها بر اساس نقاط بحرانی حرکت می‌کنند. در هر لحظه تنها یک ربات لغزان حرکت می‌کند. الگوریتم ارائه شده این مزیت را داراست که با شروع از هر ربات دلخواه یک طراحی مسیر حرکت برای تمیز کردن P ارائه می‌دهد.

الگوریتم ما دارای شش مرحله اصلی است که عبارتند از: گام شروع، گام تصمیم‌گیری، گام فرستادن سیگنال، گام حرکت به عقب، گام به‌روز رسانی و گام پایانی.

برای ارائه مسیر حرکت از یک ربات لغزان دلخواه $r_i \in R$ که روی $s_i \in S$ قرار دارد شروع می‌کنیم. r_i حرکتش را از روی یکی از دو نقطه‌ی انتهایی s_i به صورت دلخواه آغاز می‌کند. شکل ۲.۵ را ملاحظه کنید. فرض کنید $r_i = r_1$ از نقطه‌ی انتهایی سمت چپ s_1 شروع به حرکت می‌کند. زمانی که r_i به یک نقطه‌ی بحرانی می‌رسد (یک رأس انعکاسی را می‌بیند) مانند نقاط a, b و c و



شکل ۲.۵: ربات‌های r_1, r_2, r_3 به ترتیب در امتداد s_1, s_2, s_3 حرکت می‌کنند.

یا به یک نقطه‌ی انتهایی s_1 می‌رسد (مانند d) قسمت‌هایی از چندضلعی را که تمیز کرده است به‌روز رسانی می‌کند ($D_i(1)$ را اضافه کرده و $D_i(2)$ را کاهش می‌دهد). به محض اینکه $D_i(2)$ تهی شود در صورتی که $D_i(1)$ غیرتهی باشد، حرکت خود را در جهت عکس حرکت قبلی شروع می‌کند (حرکت به عقب). به علاوه در هر نقطه‌ی بحرانی r_i می‌ایستد و براساس زیرچندضلعی‌های تمیز P ، حرکت خود را ادامه می‌دهد (زمانی که r_1 به a و b می‌رسد به حرکت خود ادامه می‌دهد) و یا صبر کرده و یک سیگنال برای بقیه ربات‌ها می‌فرستد تا زیرچندضلعی مورد نظرش را تمیز کنند (زمانی که r_1 به نقطه c می‌رسد باید منتظر باشد تا P_1 تمیز شود) و پس از آن r_i بتواند به حرکت خود ادامه دهد (گام تصمیم‌گیری). بنابراین $D_i(3)$ برابر با P_1 می‌شود. زمانی که r_i یک سیگنال می‌فرستد یک ربات لغزان مانند r_j که می‌تواند حداقل یک قسمت از P_1 را تمیز کند حرکتش را در راستای پاره‌خط متناظرش شروع می‌کند (زمانی که r_1 می‌ایستد، حرکت خود را برای تمیز کردن P_1 شروع می‌کند و $D_1(3) = D_3(2)$). زمانی که تمام قسمت‌های P تمیز شوند، الگوریتم به پایان می‌رسد. به عبارت دیگر، از آنجایی که در ابتدا فرض می‌کنیم که تمام P آلوده است پس برای تمام $r_i \in R$ داریم $D_i(1) = \emptyset$. بنابراین زمانی که اجتماع $D_i(1)$ برای تمام $r_i \in R$ برابر با P شود، الگوریتم طراحی مسیر حرکت به پایان می‌رسد. بنابراین الگوریتم ارائه شده از تعداد ربات‌های لغزان برابر با

تعداد اعضای S استفاده می‌کند. نتیجه مهم این است که اگر مجموعه S شامل پاره‌خط‌هایی باشد که جواب مسئله $MCSC$ است آنگاه الگوریتم ما چندضلعی متعامد را توسط کمترین تعداد ربات‌های لغزان تمیز می‌کند.

۳.۲.۵ تحلیل الگوریتم

برای اثبات درستی الگوریتم ابتدا در لم ۳.۵ ثابت می‌کنیم الگوریتم دارای حلقه نمی‌باشد. به دلیل آنکه (۱) در هر لحظه یک ربات حرکت می‌کند، (۲) مجموعه‌ی S تمام P را پوشش می‌دهد و (۳) این نکته که الگوریتم دارای حلقه نمی‌باشد، بنابراین الگوریتم متوقف می‌شود.

لم ۳.۵ الگوریتم ارائه شده دارای حلقه نمی‌باشد (در دور گیر نمی‌کند).

اثبات. فرض کنید r_i منتظر است تا زیرچندضلعی P_i تمیز شود. درون P_i ممکن است r_j منتظر باشد تا زیرچندضلعی P_j تمیز شود. بنابراین یک زنجیر از ربات‌های منتظر درون P_i وجود دارند. این زنجیر را $r_{seq(i)} = \langle r_j, r_t, \dots, r_m \rangle$ می‌نامیم. اگر $r_i \in r_{seq(i)}$ باشد، آنگاه یک حلقه اتفاق می‌افتد و الگوریتم تمام نمی‌شود. بنابراین ما نشان می‌دهیم $r_i \in r_{seq(i)}$ هرگز رخ نمی‌دهد.

بنابر تعریف پنجره و زیرچندضلعی‌های آن، زمانی که r_i برای تمیز شدن P_i منتظر می‌ماند هرگز نمی‌تواند هیچ نقطه‌ی داخلی P_i (مگر پنجره آن) را ببیند. به دلیل آنکه زیرچندضلعی‌های متناظر با ربات‌های $r_{seq(i)}$ درون P_i هستند، هیچ کدام از آن‌ها نمی‌توانند منتظر r_i بمانند. بنابراین الگوریتم در حلقه گیر نمی‌کند. ■

لم ۴.۵ یک چندضلعی متعامد و ساده می‌تواند با شروع از یک ربات لغزان دلخواه به طور کامل تمیز شد.

اثبات. فرض کنید از یک ربات دلخواه r_i شروع می‌کنیم. بنابر لم ۳.۵ الگوریتم ارائه شده در حلقه گیر نمی‌کند. به علاوه به دلیل آنکه S تمام P را پوشش می‌دهد، گام پایانی اتفاق می‌افتد. بنابر گام

پایانی، $P = \bigcup_{i=1}^{|S|} D_i(1)$ اتفاق می‌افتد. در نتیجه تمام نقاط درون P دیده شده و P تمیز می‌شود. ■

نکته مهم دیگر، بدست آوردن مجموع طول پیموده شده توسط تمام ربات‌ها برای تمیز کردن چندضلعی است. این مجموع طول را X می‌نامیم. فرض کنید m مجموع طول تمام یال‌های P و $S_{out} \subset S$ مجموعه‌ی پاره‌خط‌های خروجی الگوریتم باشد که ربات‌ها روی آن‌ها حرکت کرده و P را تمیز می‌کنند. ابتدا ثابت می‌کنیم دوباره آلوده شدن (به عبارت دیگر، زمانی که یک قسمت از P تمیز می‌شود دیگر توسط هیچ دزدی آلوده نمی‌شود) نداریم. سپس ثابت می‌کنیم X حداکثر برابر با $|e_{max}| \cdot (2m + n)$ است، که $e_{max} = \max_{e_i \in S} e_i$.

لم ۵.۵ دوباره آلوده شدن در الگوریتم ارائه شده وجود ندارد.

اثبات. بنابر الگوریتم ارائه شده زمانی که یک ربات تمیز کردنش را تمام می‌کند در همان نقطه می‌ایستد. بنابراین از قسمتی که تمیز کرده محافظت می‌کند. در حالت‌های خاصی یک ربات مانند r_i ممکن است کار تمیز کردن را بیش از یک مرتبه انجام دهد. در این حالت‌ها یک ربات دیگر وظیفه محافظت از قسمت تمیز شده را برعهده می‌گیرد. بنابر لم ۳.۵ الگوریتم در حلقه گیر نمی‌کند، بنابراین r_i هرگز هیچ قسمتی از P_i را تمیز نمی‌کند و هیچ نقطه‌ای از $D_i(1)$ دوباره آلوده نمی‌شود. در نتیجه دوباره آلوده شدن در الگوریتم نداریم. ■

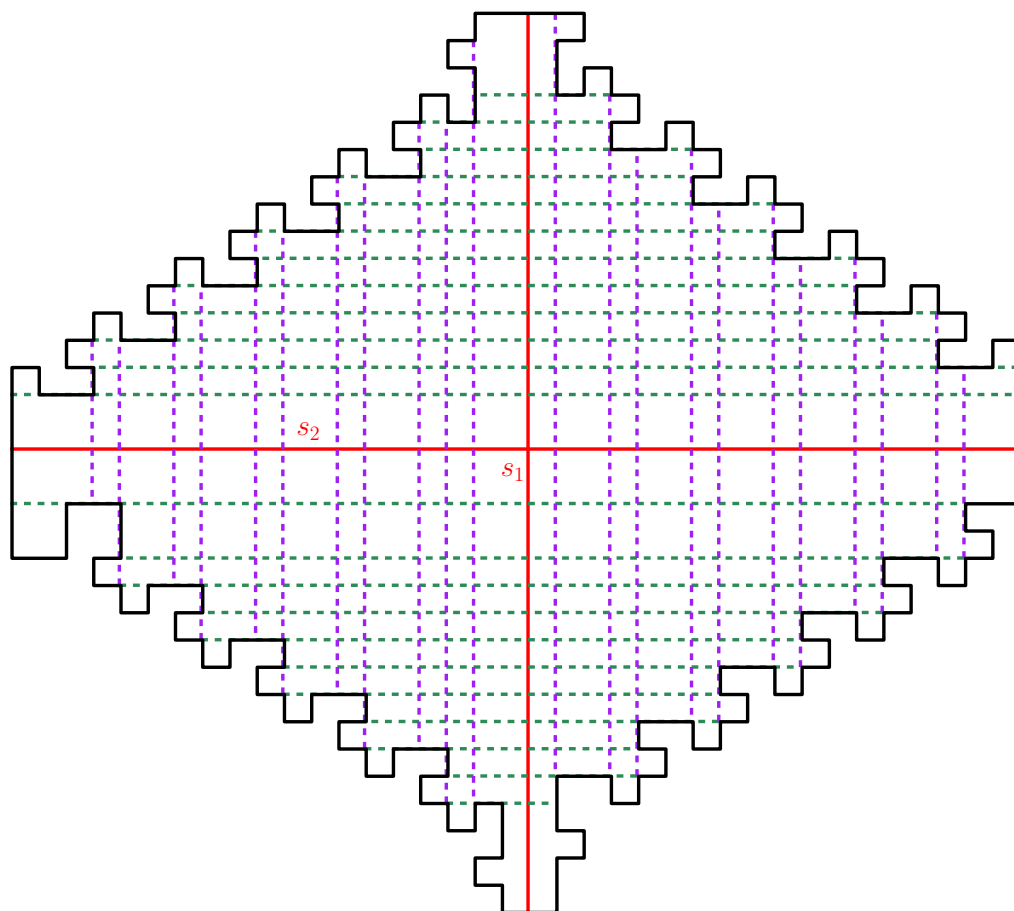
لم ۶.۵ مجموع طول تمام مسیرهایی که ربات‌ها طی می‌کنند تا P را تمیز کنند برابر است با $2m + |e_{max}| \cdot n$.

اثبات. مجموع طول پیموده شده توسط ربات‌ها (X) می‌تواند به سه مجموعه از حرکت‌ها تقسیم شود. مجموعه‌ای از حرکت‌ها که ربات همزمان با حرکت کردنش قسمتی از P را نیز تمیز می‌کند، که آن‌ها را Y می‌نامیم. مجموعه‌ای از حرکت‌ها در گام حرکت به عقب ربات‌ها، که آن‌ها را Z می‌نامیم.

مجموعه‌ای از حرکت‌ها زمانی که یک ربات سیگنالی را دریافت می‌کند و حرکت می‌کند تا به نقطه‌ی شروع مسیر تمیز کردنش برسد، که آن‌ها را W می‌نامیم. بنابر الگوریتم ارائه شده، قسمت‌های تمیز شده همیشه اضافه شده و هرگز کم نمی‌شوند. فقط زمانی که یک ربات حرکت به عقب دارد قسمت تمیز شده تغییری نمی‌کند (نه افزایش و نه کاهش). بنابر لم ۵.۵ و این نکته که قسمت تمیز شده هرگز کاهش نمی‌یابد، الگوریتم هر قسمت از P را فقط یک مرتبه تمیز می‌کند. بنابراین برای تمیز کردن P تنها کافی‌ست مرزهای P را یک مرتبه تمیز کرده و خواهیم داشت $Y \leq m$. به دلیل آنکه مسیرهایی که یک ربات حرکت می‌کند و همزمان قسمتی از P را تمیز می‌کند و مسیرهایی که برگشت به عقب دارد برابر هستند، داریم $Z \leq m$. همان‌طور که بیان کردیم الگوریتم مرز P را یک مرتبه تمیز می‌کند. بنابراین در بدترین حالت برای تمیز کردن هر رأس از P یک ربات باید صدا زده شود (گام فرستادن سیگنال). در بدترین حالت در هر مرتبه صدا زدن یک ربات r_i باید در طول s_i حرکت کرده و به نقطه‌ی شروع برای تمیز کردن برسد. در نتیجه مجموع طول حرکت‌ها در هر مرتبه صدا زدن حداکثر برابر با s_i است. به دلیل آنکه بلندترین یال مجموعه‌ی S_{out} است، داریم $|s_i| \leq |e_{max}|$. پس هر ربات در هر مرتبه صدا زدن حداکثر $|e_{max}|$ مقدار حرکت می‌کند. به دلیل آنکه تعداد رأس‌ها n است، برای تمام صدا زدن‌ها داریم $W \leq n|e_{max}|$. بنابراین نتیجه می‌گیریم $X \leq 2m + n|e_{max}|$. بدترین حالت تعداد صدا زدن‌ها در شکل ۳.۵ نشان داده شده است. ■

بنابر لم ۴.۵ با شروع از یک ربات دلخواه الگوریتم تمام P را تمیز می‌کند. بنابراین الگوریتم ما درست است و می‌توانیم قضیه زیر را نتیجه بگیریم.

قضیه ۷.۵ برای یک چندضلعی متعامد P که دارای تعداد نامحدودی دزد است، یک الگوریتم چندجمله‌ای برای طراحی مسیر حرکت دسته‌ای از ربات‌های لغزان وجود دارد که ربات‌ها می‌توانند تمام دزدها را پیدا کنند. همچنین مجموع طول مسیری که تمام ربات‌ها برای تمیز کردن P طی می‌کنند، حداکثر $2m + n|e_{max}|$ است.



شکل ۳.۵: برای تمیز کردن چندضلعی، ربات s_1 و یا s_2 باید $O(n)$ مرتبه صدا زده شوند.

نتیجه ۸.۵ اگر S مجموعه‌ی شامل کمترین تعداد دوربین‌های لغزان باشد، آنگاه الگوریتم ما P را توسط کمترین تعداد ربات‌های لغزان، با فرض اینکه ربات‌ها تنها مجاز به حرکت در امتداد پاره‌خط‌های S باشند، تمیز می‌کند.

برای مشاهده‌ی جزئیات بیشتر الگوریتم می‌توانید به مقاله [۳۸] مراجعه کنید.

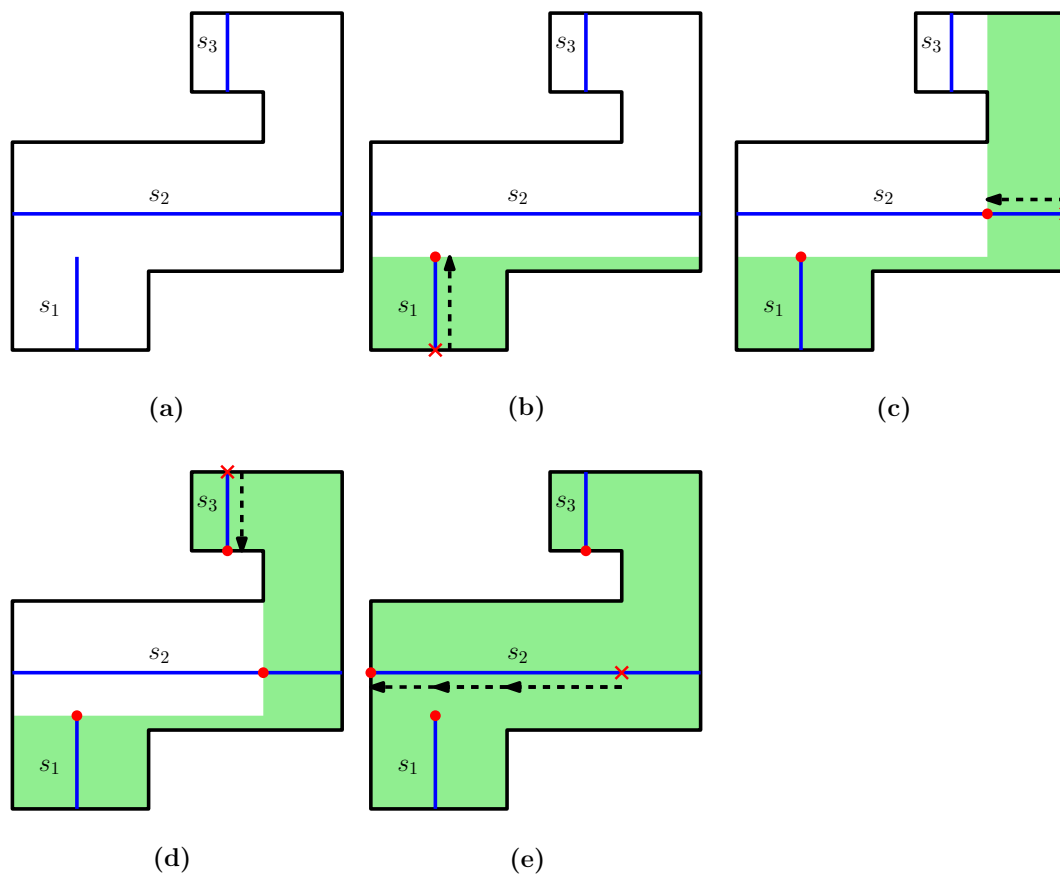
۴.۲.۵ پیاده‌سازی الگوریتم

الگوریتم در زبان جاوا با استفاده از کامپیوتر اپل MF۸۴۰ با پردازشگر ۲/۷ گیگاهرتز، رم ۸ گیگا بایت پیاده‌سازی شده است. یک مثال از پیاده‌سازی در شکل ۴.۵ نشان داده شده است. برای دیدن پویانمایی از پیاده‌سازی الگوریتم روی تعدادی چندضلعی متعامد و ساده و همچنین گزارش مجموع طول پیموده شده توسط ربات‌ها، تعداد گام‌های تصمیم‌گیری و تعداد صدا زدن ربات‌های لینک زیر

را مشاهده نمایید.

https://www.dropbox.com/sh/wskztugum1s411u/AAApfhkIjC1WkjRxOr_9xlPAa?dl=

0



شکل ۴.۵: مسیر ربات‌ها، نقطه‌ی ابتدایی و نقطه‌ی انتهایی هر گام با فلش‌های نقطه‌چین، ضربدر و دایره به ترتیب نشان داده شده‌اند.

۳.۵ تمیز کردن یک موزه‌ی هنر متعامد و x -یکنوا توسط کمترین تعداد ربات‌های لغزان

۱.۳.۵ تعریف مسئله

فرض کنید یک چندضلعی متعامد و x -یکنوا، P ، به عنوان ورودی داده شده است. P دارای تعدادی دزد است که با سرعت زیاد در حال حرکت بوده و دارای مکان نامشخص هستند. هدف طراحی مسیر حرکت برای کمترین تعداد ربات‌های لغزان است که بتوانند P را تمیز کنند (به عبارت دیگر، تمام دزدها را پیدا کنند). ابتدا یک مجموعه از کمترین تعداد دوربین‌های لغزان را پیدا می‌کنیم که بتوانند P را پوشش دهند و آن را $S = \{s_1, s_1, \dots, s_k\}$ می‌نامیم. سپس دنباله‌ای از حرکت‌ها برای ربات‌های لغزان ارائه می‌دهیم که روی این دوربین‌های لغزان حرکت کنند و P را تمیز کنند. با توجه به پیچیدگی ارتباطی میان بیش از یک ربات که می‌خواهند یک کار مشترک را انجام دهند، ما یک روش جدید با ذخیره‌سازی اطلاعات روی رأس‌های انعکاسی، ارائه می‌دهیم. فرض کنید چندضلعی P برای ربات‌ها شناخته شده است و ربات‌ها هیچ ارتباطی با یکدیگر ندارند. تنها هدایت‌کننده ربات‌ها یک نشانه‌گر^۱ است که روی مجموعه‌ی S حرکت می‌کند.

بهترین نتیجه‌ی الگوریتم ارائه شده این است که می‌تواند یک چندضلعی متعامد و x -یکنوا را توسط کمترین تعداد ربات لغزان در زمان حساس به خروجی $\mathcal{O}(n \log n + nm_{out})$ تمیز کند، که m_{out} تعداد ربات‌های خروجی است.

۲.۳.۵ الگوریتم ارائه شده

برای بدست آوردن مجموعه‌ی S می‌توانیم از الگوریتم ارائه شده در [۱۱] استفاده کنیم. با استفاده از این الگوریتم، S در زمان خطی قابل محاسبه است. برای استفاده کردن از S به عنوان مجموعه‌ی هدایتگر ربات‌ها، آن را کمی تغییر می‌دهیم. تمام پاره‌خط‌های عمودی درون S را به سمت چپان

^۱Pointer

حرکت می‌دهیم تا به مرز P برخورد کنند و سپس آن‌ها را از دو طرف امتداد می‌دهیم تا به مرز برخورد کنند (پاره‌خط‌ها بیشترین عمودی^۲ می‌شوند). با این تغییرات چندضلعی رویت‌پذیر این پاره‌خط‌ها ممکن است اضافه شوند اما تعداد آن‌ها تغییر نمی‌کند. سپس براساس نقطه‌ی انتهایی چپ تمام اعضای S را از چپ به راست مرتب می‌کنیم (برای پاره‌خط‌های عمودی نقطه‌ی انتهایی بالایی را به عنوان نقطه‌ی انتهایی چپ در نظر بگیرید). در صورتی‌که نقطه‌ی انتهایی چپ یک پاره‌خط افقی و عمودی برابر باشند، ابتدا پاره‌خط عمودی را قرار دهید. به دلیل آنکه S کمترین تعداد دورین لغزان را دارد که P را پوشش می‌دهند، هیچ دو پاره‌خط افقی با مولفه x یکسان در نقطه‌ی انتهایی چپشان و هیچ دو پاره‌خط عمودی با مولفه x یکسان وجود ندارند.

حال با استفاده از S یک طراحی مسیر حرکت برای ربات‌های لغزان $R = \{r_1, r_2, \dots, r_k\}$ ارائه می‌دهیم که در آن ربات r_i روی پاره‌خط s_i حرکت می‌کند. فرض کنید p_s یک اشاره‌گر است که روی S حرکت می‌کند که در ابتدا $p_s = s_1$. همچنین فرض کنید $p_s(next)$ و $p_s(previous)$ به ترتیب به پاره‌خط‌های بعد و قبل از p_s در مجموعه‌ی S اشاره می‌کنند.

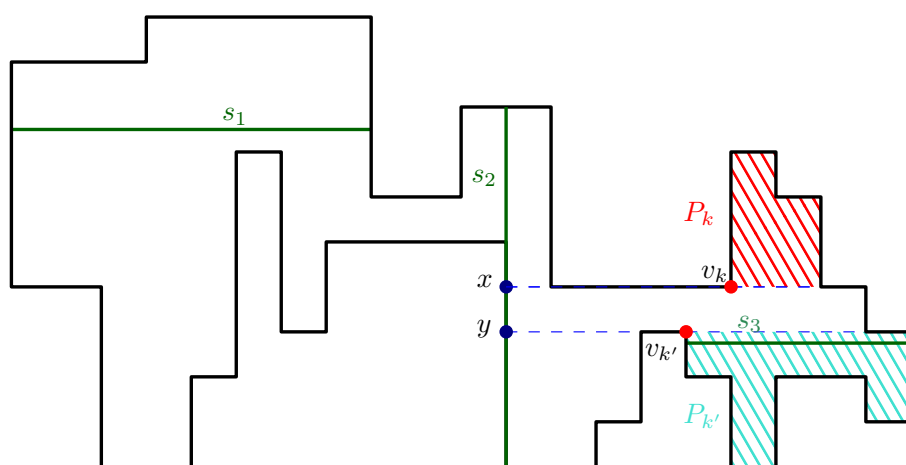
هر ربات لغزان r_i دارای سه حافظه $D_i(1), D_i(2), D_i(3)$ است. $D_i(1)$ بیانگر زیرچندضلعی تمیزی از P است که r_i قسمتی از آن را تمیز کرده است. $D_i(2)$ بیانگر زیرچندضلعی از P است که باید توسط r_i به طور کامل و یا قسمتی از آن تمیز شود. $D_i(3)$ بیانگر زیرچندضلعی از P است که باید تمیز شود تا r_i بتواند به حرکت خود ادامه دهد. به دلیل آنکه فرض کردیم تمام P آلوده است بنابراین داریم: $\forall r_i \in R D_i(1) = \emptyset$.

در الگوریتم ما ربات‌ها بر اساس نقاط بحرانی حرکت می‌کنند. همانطور که در ۱.۵ تعریف کردیم، یک نقطه‌ی بحرانی زمانی اتفاق می‌افتد که ربات r_i یک رأس انعکاسی را ببیند، به یکی از نقاط انتهایی پاره‌خط s_i برسد و یا یک ربات دیگر را که منتظر است ببیند. در هر لحظه تنها یک ربات لغزان حرکت می‌کند. الگوریتم ما دارای پنج مرحله اصلی است که عبارتند از: گام شروع، گام

^۲Vertically Maximal

تصمیم‌گیری، گام به‌روز رسانی، گام منتظر ماندن و گام پایانی.

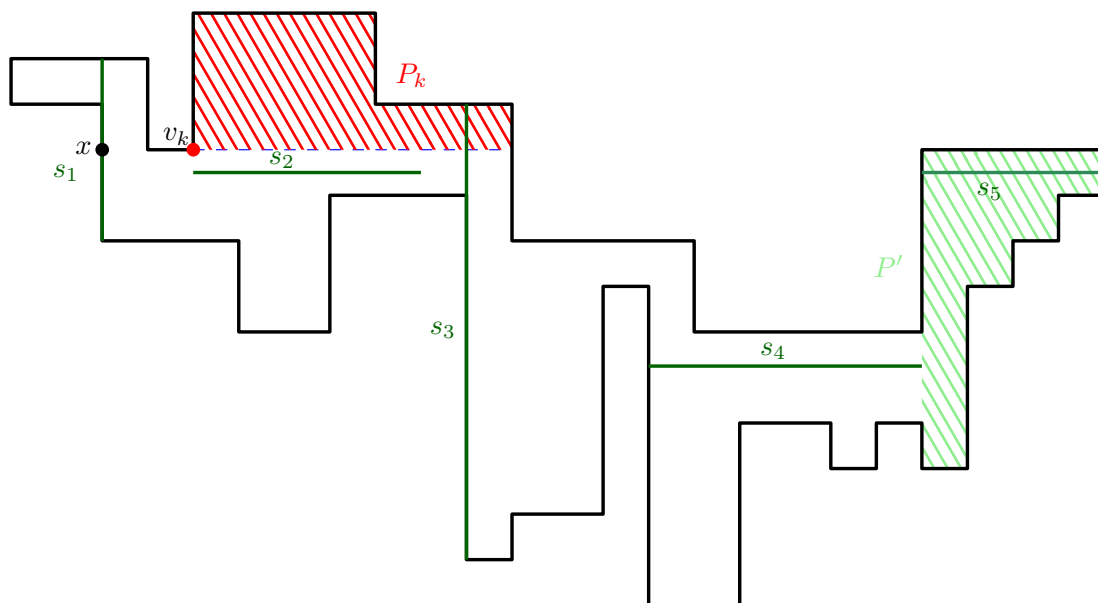
گام شروع الگوریتم از نقطه‌ی انتهایی سمت چپ پاره‌خطی است که p_s به آن اشاره دارد که در ابتدا $p_s = s_1$ است. فرض کنید در حین اجرای الگوریتم بوده و $p_s = s_j$ است. حرکت خود را از نقطه‌ی انتهایی سمت چپ s_j شروع می‌کند. اگر s_j افقی باشد، آنگاه r_j حرکت خود را تا رسیدن به نقطه‌ی انتهایی سمت راست s_j ادامه می‌دهد و در انتها p_s را با $p_s(next)$ به‌روز رسانی می‌کند (پاره‌خط s_1 را در شکل ۵.۵ ملاحظه کنید). اگر s_j عمودی باشد، حرکت خود را تا دیدن یک رأس



شکل ۵.۵: r_i روی s_i حرکت می‌کند.

انعکاسی و یا رسیدن به نقطه‌ی انتهایی دیگر s_j ادامه داده و سپس قسمت‌هایی را که تمیز کرده به‌روز رسانی می‌کند ($D_j(1)$ را اضافه کرده و $D_j(2)$ را کاهش می‌دهد). پاره‌خط s_1 یا s_3 را در شکل ۶.۵ ملاحظه کنید. به دلیل آنکه S از چپ به راست مرتب شده است، تغییر p_s و $D_1(1)$ در زمان $O(1)$ قابل انجام است. بنابراین گام شروع در زمان $O(1)$ انجام می‌شود.

دو گام منتظر ماندن و تصمیم‌گیری به این صورت هستند که زمانی که r_j یک رأس انعکاسی را می‌بیند، منتظر می‌ماند تا زیر چندضلعی متناظر آن تمیز شود. به طور مثال، زمانی که r_1 در شکل ۶.۵ به نقطه‌ی x (به‌طور مشابه زمانی که r_2 در شکل ۵.۵ به نقطه‌ی x و یا نقطه‌ی y می‌رسد) رسیده و رأس v_k را می‌بیند، منتظر می‌ماند تا زیر چندضلعی P_k تمیز شود. برای این کار p_s را به $p_s(next)$ تغییر داده و تمیز کردن P_k توسط رباتی که p_s به پاره‌خط متناظر آن اشاره می‌کند، شروع می‌شود. زمانی که

شکل ۵.۶: r_i روی s_i حرکت می‌کند.

r_j به نقطه‌ی انتهایی s_j می‌رسد، کار تمیز کردنش به پایان می‌رسد. بنابراین p_s را به $p_s(next)$ تغییر می‌دهد تا زیرچندضلعی باقی مانده از P را شروع به تمیز کردن کند. به طور مثال زمانی که r_f به نقطه‌ی انتهایی s_f می‌رسد و p_s را به s_5 تغییر داده و P' باید تمیز شود.

زمانی که (۱) هیچ ربات منتظری وجود نداشته باشد، (۲) تمام ربات‌ها قسمت‌های متناظرشان را تمیز کرده باشند $(\forall r_i \in R D_i(2) = \emptyset)$ ، و (۳) تمام قسمت‌های P تمیز شده باشند $(\bigcup_{i=1}^{|R|} D_i(1) = P)$ ، آنگاه الگوریتم تمام می‌شود. این سه شرط می‌توانند در هر زمان که r_i روی نقطه‌ی انتهایی s_i و روی مرز P قرار دارد بررسی شود. این بررسی کردن در زمان $O(1)$ قابل انجام است.

۳.۳.۵ تحلیل زمان و درستی الگوریتم

برای اثبات درستی الگوریتم ابتدا در لم ۹.۵ ثابت می‌کنیم الگوریتم دارای حلقه نمی‌باشد. به دلیل آنکه (۱) در هر لحظه یک ربات حرکت می‌کند، (۲) مجموعه‌ی S تمام P را پوشش می‌دهد و (۳) این نکته که الگوریتم دارای حلقه نمی‌باشد، بنابراین الگوریتم متوقف می‌شود.

لم ۹.۵ الگوریتم ارائه شده دارای حلقه نمی‌باشد (در دور گیر نمی‌کند).

اثبات. فرض کنید r_i منتظر است تا زیرچندضلعی P_i تمیز شود. درون P_i ممکن است r_j منتظر باشد تا زیرچندضلعی P_j تمیز شود. بنابراین یک زنجیر از ربات‌های منتظر درون P_i وجود دارند. این زنجیر را $r_{seq(i)} = \langle r_j, r_t, \dots, r_m \rangle$ می‌نامیم. اگر $r_i \in r_{seq(i)}$ باشد، آنگاه یک حلقه اتفاق می‌افتد و الگوریتم تمام نمی‌شود. بنابراین ما نشان می‌دهیم $r_i \in r_{seq(i)}$ هرگز رخ نمی‌دهد. بنابر تعریف پنجره و زیرچندضلعی‌های آن، زمانی که r_i برای تمیز شدن P_i منتظر می‌ماند هرگز نمی‌تواند هیچ نقطه‌ی داخلی P_i (مگر پنجره آن) را ببیند. به دلیل آنکه زیرچندضلعی‌های متناظر با ربات‌های $r_{seq(i)}$ درون P_i هستند، هیچ کدام از آن‌ها نمی‌توانند منتظر r_i بمانند. بنابراین الگوریتم در حلقه گیر نمی‌کند. ■

لم ۱۰.۵ دوباره آلوده شدن در الگوریتم ارائه شده وجود ندارد.

اثبات. بنابر الگوریتم ارائه شده زمانی که یک ربات تمیز کردنش را تمام می‌کند در همان نقطه می‌ایستد. بنابراین از قسمتی که تمیز کرده محافظت می‌کند. در حالت‌های خاصی یک ربات مانند r_i ممکن است کار تمیز کردن را بیش از یک مرتبه انجام دهد. در این حالت‌ها یک ربات دیگر وظیفه محافظت از قسمت تمیز شده را برعهده می‌گیرد. بنابر لم ۹.۵ الگوریتم در حلقه گیر نمی‌کند، بنابراین r_i هرگز هیچ قسمتی از P_i را تمیز نمی‌کند و هیچ نقطه‌ای از $D_i(1)$ دوباره آلوده نمی‌شود. در نتیجه دوباره آلوده شدن در الگوریتم نداریم. ■

حال ثابت می‌کنیم پیچیدگی زمانی الگوریتم ارائه شده $\mathcal{O}(n \log n + nm_{out})$ است، که در آن m_{out} بیانگر تعداد ربات‌های لغزان استفاده شده در الگوریتم است. به دلیل آنکه پیچیدگی زمانی گام تصمیم‌گیری و گام منتظر ماندن به هم وابسته هستند، پیچیدگی زمانی مجموع این دو گام را با هم محاسبه می‌کنیم. بنابر الگوریتم، قسمت‌های تمیز شده همیشه افزایش می‌یابند و هرگز کاهش نمی‌یابند. با توجه به لم ۱۰.۵ و این حقیقت که قسمت‌های تمیز شده همواره افزایش می‌یابند، الگوریتم هر قسمت

P را یک مرتبه تمیز می‌کند. بنابراین گام منتظر ماندن برای هر زیرچندضلعی فقط یک مرتبه صدا زده می‌شود. در بدترین حالت در هر رأس انعکاسی گام تصمیم‌گیری گام منتظر ماندن را صدا می‌زند. بنابراین پیچیدگی زمانی این دو گام در تمام الگوریتم $O(nm_{out})$ است. همانطور که توضیح دادیم پیچیدگی زمانی گام شروع $O(1)$ است. به دلیل آنکه مرتب کردن S دارای پیچیدگی زمانی $O(n \log n)$ است، پیچیدگی زمانی الگوریتم ارائه شده $O(n \log n + nm_{out})$ است.

قضیه ۱۱.۵ یک الگوریتم طراحی مسیری برای تمیزکردن یک چندضلعی متعامد و x -یکنوا توسط کمترین تعداد ربات‌های لغزان وجود دارد که دارای مرتبه‌ی زمانی $O(n \log n + nm_{out})$ است. در اینجا مقصود از m_{out} تعداد ربات‌های لغزان است که در الگوریتم مورد استفاده قرار می‌گیرند.

برای مشاهده‌ی جزئیات بیشتر الگوریتم می‌توانید به مقاله [۳۹] مراجعه کنید.

۴.۵ نتیجه‌گیری

در ادامه کارهای انجام شده، قصد بررسی مسئله‌ی طراحی مسیری برای حرکت دسته‌ای از ربات‌های لغزان را برای تمیزکردن چندضلعی‌های متعامد داریم در صورتی که محیط برای ربات‌ها ناشناخته باشد. همچنین بررسی مسئله در حالتی که کمترین رابطه میان ربات‌ها را داشته باشیم، قابل بررسی است. این مسئله از لحاظ کاربردی حائض اهمیت است زیرا استفاده از ربات‌های با قدرت کم از نظر اقتصادی بسیار به صرفه است. بررسی مسئله‌ی طراحی مسیری حرکت برای ربات‌های لغزان زمانی که ربات‌ها دارای قدرت پوشش با گذر از k دیوار را داشته باشند، از دیگر مسائل مطرح قابل بررسی در این زمینه خواهد بود.

فصل ۶

نتیجه‌گیری و کارهای آینده

ما در این رساله مسئله‌ی پوشش موزه‌های هنر متعامد توسط k -فرستنده‌های لغزان را مورد بررسی قرار دادیم. در فصل ۱ مقدمه و مفاهیم مرتبط با مسئله را شرح دادیم. سپس در فصل ۲ تاریخچه و کارهای انجام شده قبلی برای مسئله نگرهبانی موزه‌های هنر متعامد توسط دوربین‌های لغزان را مطرح کردیم.

در فصل ۳ به بررسی مسئله‌ی پوشش موزه‌های هنر متعامد توسط k -فرستنده‌های لغزان پرداختیم. در این مسئله فرض کردیم که هدف کمینه کردن مجموع طول k -فرستنده‌های لغزان است. ابتدا با استفاده از کاهش این مسئله به مسئله کاشی‌کاری، نشان دادیم که این مسئله برخلاف حالت ۰-فرستنده، یک مسئله ان‌پی-سخت است. سپس یک الگوریتم تقریبی با ضریب تقریب ۲ برای این مسئله ارائه دادیم که در زمان چندجمله‌ای قابل حل است. نتیجه بدست آمده برای این مسئله در کنفرانس CCCG'14 [۴۰] چاپ شده است. همچنین نسخه تکمیل شده‌ی این مقاله در مجله Theoretica Computer Science [۴۱] به چاپ رسیده است.

در فصل ۴ به بررسی مسئله‌ی پوشش موزه‌های هنر متعامد و x -یکنوا توسط کمترین تعداد ۲-فرستنده‌های لغزان پرداخته و یک الگوریتم چندجمله‌ای با ضریب تقریب ۳ برای حل این مسئله زمانی که فقط مجاز به استفاده از فرستنده‌های عمودی لغزان هستیم، ارائه دادیم. سپس یک الگوریتم

چندجمله‌ای با ضریب تقریب ۷ برای حل مسئله‌ی پوشش موزه‌های هنر متعامد و x -یکنوا توسط کمترین تعداد ۲- فرستنده‌های لغزان ارائه دادیم.

در فصل ۵ به بررسی مسئله‌ی تمیز کردن یک موزه‌ی هنر متعامد توسط دوربین‌های لغزان پرداخته و این مسئله را در دو حالت مختلف حل کردیم. ابتدا فرض کردیم موزه‌ی هنر یک چندضلعی متعامد و x -یکنوا است و یک الگوریتم چندجمله‌ای برای تمیز کردن آن ارائه دادیم. سپس حالت کلی‌تری را فرض کرده و شرط یکنوا بودن را حذف کردیم. در این حالت نیز یک الگوریتم چندجمله‌ای برای تمیز کردن هر موزه‌ی هنر متعامد ارائه دادیم. نتیجه‌ی بدست آمده از ۲.۵ در مجله Theoretical Computer Science [۳۸] به چاپ رسیده و نتیجه‌ی بدست آمده از ۳.۵ به Information Processing Letters [۳۹] ارسال شده است.

۱.۶ نگهبانی موزه‌ی هنر متعامد توسط دوربین‌های لغزان

مسائلی را که در ادامه پژوهش می‌توان بررسی کرد در پایان هر فصل شرح دادیم. همان‌طور که توضیح دادیم پیچیدگی مسئله‌ی نگهبانی موزه‌های هنر توسط کمترین تعداد دوربین‌های لغزان هنوز ثابت نشده است. این پیچیدگی مدت زیادی است که توسط محققان بررسی می‌شود، اما تنها برای چندضلعی‌های دارای حفره نتیجه‌ای بدست آمده است. بنابراین یکی از مسائلی که در ادامه پژوهش می‌توان بررسی کرد همین مسئله است. همچنین تاکنون الگوریتمی با ضریب تقریب ۳.۵ برای این مسئله ارائه شده است. چون این، اولین الگوریتم تقریبی است، یک مسئله‌ی مطرح دیگر اثبات یک حد پایین برای تقریب این مسئله خواهد بود.

۲.۶ پوشش موزه‌های هنر متعامد توسط k -فرستنده‌های لغزان

در مورد مسئله‌ی پوشش چندضلعی‌های متعامد توسط کمترین مجموع طول k -فرستنده‌های لغزان، ثابت کردیم که این مسئله IN-PI -سخت است و یک الگوریتم تقریبی با ضریب تقریب ۲ برای آن

ارائه دادیم که این اولین الگوریتم تقریبی در این زمینه است. برای ادامه پژوهش در این زمینه می‌توان به بررسی بدست آوردن یک حد پایین برای بهترین ضریب تقریب ممکن برای این مسئله پرداخت. همچنین با توجه به این پی-سخت بودن این مسئله بدست آوردن یک الگوریتم تصادفی و الگوریتم با پارامتر ثابت می‌تواند موضوعی برای ادامه پژوهش باشد. برای این مسئله هم ورودی می‌تواند تعداد حداکثر فرستنده‌ای باشد که قرار است کل چندضلعی را پوشش دهند.

۳.۶ تمیز کردن موزه‌های هنر توسط دوربین‌های لغزان

در فصل ۵ به بررسی مسئله‌ی تمیز کردن موزه‌های هنر متعامد توسط دوربین‌های لغزان پرداختیم. در این مسئله فرض بر این است که موزه‌ی هنر دارای تعدادی دزد است (به عبارت دیگر آلوده است) و هدف بدست آوردن دنباله‌ای از مسیرهای حرکت برای ربات‌هاست به گونه‌ای که در پایان حرکت آن‌ها، تمام دزدهای داخل موزه‌ی هنر پیدا شوند (به عبارت دیگر موزه‌ی هنر تمیز شود). مسئله‌ی تمیز کردن یک موزه‌ی هنر توسط تعدادی ربات لغزان را در دو حالت تمیز کردن یک موزه‌ی هنر متعامد و x -یکنوا توسط کمترین تعداد ربات‌های لغزان و تمیز کردن یک موزه‌ی هنر متعامد توسط تعدادی ربات‌های لغزان بررسی کرده و الگوریتم‌های چندجمله‌ای برای حل آن‌ها ارائه دادیم. در ادامه کارهای انجام شده، می‌توان به بررسی مسئله‌ی طراحی مسیر برای حرکت دسته‌ای از ربات‌های لغزان را برای تمیز کردن چندضلعی‌های متعامد داریم در صورتی که محیط برای ربات‌ها ناشناخته باشد. همچنین بررسی مسئله در حالتی که کمترین رابطه میان ربات‌ها را داشته باشیم، قابل بررسی است. این مسئله از لحاظ کاربردی حائز اهمیت است زیرا استفاده از ربات‌های با قدرت کم از نظر اقتصادی بسیار به صرفه است. بررسی مسئله‌ی طراحی مسیر حرکت برای ربات‌های لغزان زمانی که ربات‌ها دارای قدرت پوشش با گذر از k دیوار را داشته باشند، از دیگر مسائل مطرح قابل بررسی در این زمینه خواهد بود.

فهرست مراجع

- [1] O. Aichholzer, R. Fabila-Monroy, D. Flores-Pealoza, T. Hackl, C. Huemer, J. Urrutia, and B. Vogtenhuber. *Modern illumination of monotone polygons*. In Proceeding of 25th European Conference on Computational Geometry, 167–170, 2009. [22](#), [35](#)
- [2] O. Aichholzer, F. Aurenhammer, F. Hurtado, P. Ramos, and J. Urrutia. *k-convex polygons*. In Proceeding of 25th European Conference on Computational Geometry, 117–120, 2009. [22](#)
- [3] B. Ballinger, N. Benbernou, P. Bose, M. Damian, E. D. Demaine, V. Dujmovi´c, R. Flatland, F. Hurtado, J. Iacono, A. Lubiw, P. Morin, V. Sacrist´an, D. Souvaine, and R. Uehara. *Coverage with k-transmitters in the presence of obstacles*, Journal of Combinatorial Optimization, Springer, 25(2), 208–233, 2013. [14](#), [22](#)
- [4] R. Bar-Yehuda, M. M. Halldórsson, J. Naor, H. Shachnai, and I. Shapira. *Scheduling split intervals*. SIAM Journal on Computing, 36(1):1–15, 2006. [50](#)
- [5] D. Beauquier, M. Nivat, E. Rémila, and M. Robson. *Tiling figures of the plane with two bars*. Computational Geometry, 5(1): 1–25, 1995. [34](#)
- [6] A. T. Benjamin and J. J. Quinn. *Proofs that really count: the art of combinatorial proof*. Number 27. MAA, 2003. [32](#)
- [7] M. A. Beiruti. *Guarding polygons with sliding cameras*. Master’s thesis, Sharif University of Technology, 2015. [9](#), [10](#), [20](#), [53](#), [54](#), [55](#), [56](#), [58](#), [60](#), [67](#)
- [8] T. Biedl, T. M. Chan, S. Lee, S. Mehrabi, F. Montecchiani, H. Vosoughpour, and Z. Yu. *Guarding Orthogonal Art Galleries with Sliding k-Transmitters: Hardness and Approximation*. Algorithmica, 1–29, 2018. [24](#)
- [9] T. Biedl, S. Mehrabi, and Z. Yu. *Sliding k-transmitters: Hardness and approximation*. arXiv preprint arXiv:1607.07364, 2016. [24](#)
- [10] H. Brunn. *Über kernegebiete*. Mathematische Annalen, 73: 436–440, 1913. [11](#)
- [11] M. de Berg, S. Durocher, S. Mehrabi. *Guarding monotone art galleries with sliding cameras in linear time*. Journal of Discrete Algorithms, 44: 39–47, 2017. [18](#), [24](#), [79](#)

- [12] T. Christ, M. Hoffmann, Y. Okamoto, and T. Uno. *Improved bounds for wireless localization*. In SWAT '08: Proceedings of the 11th Scandinavian Workshop on Algorithm Theory, 77–89, Berlin, Heidelberg. Springer-Verlag, 2008. 23
- [13] M. Damian, R. Flatland, J. O'Rourke, and S. Ramaswami. *A new lower bound on guard placement for wireless localization*. In FWCG 07: Proceedings of the 17th Fall Workshop on Computational Geometry, 21–24, 2007. 23
- [14] G. B. Dantzig. *Linear programming and extensions*. Princeton university press, 1998. 30
- [15] A. M. Dean, W. Evans, E. Gethner, J. Laison, M. A. Safari, and W. T. Trotter. *Bar k -visibility graphs: Bounds on the number of edges, chromatic number, and thickness*. In Proceeding of Graph Drawing, LNCS 3843, 73–82, 2005. 22
- [16] S.B. David, E. Grant, W. Ma, and M. Sharpe. *The approximability and integrality gap of interval stabbing and independence problems*. 2012. 50, 51, 59
- [17] I. Dinur and S. Safra. *On the hardness of approximating minimum vertex cover*. Annals of Mathematics, 439–485, 2005. 29
- [18] M. Donaldson. *Tiling with dominoes*. PhD thesis, Citeseer, 1996. 33
- [19] J.W. Durham, A. Franchi, and F. Bullo. *Distributed pursuit-evasion without mapping or global localization via local frontiers*. Autonomous Robots, 32(1), 81–95, 2012. 27
- [20] S. Durocher, O. Filtser, R. Fraser, A. D. Mehrabi and S. Mehrabi. *A (7/2)-Approximation Algorithm for Guarding Orthogonal Art Galleries with Sliding Cameras*. In *Proceedings of the 11th Latin American Symposium*, 294–305, 2014. 24
- [21] S. Durocher and S. Mehrabi. *Guarding orthogonal art galleries using sliding cameras: algorithmic and hardness results*. In Proceeding of MFCS, 314–324, LNCS 8087, 2013. 8, 18, 23, 26, 27, 40
- [22] D. Eppstein, M. T. Goodrich, and N. Sitchinava. *Guard placement for efficient point- n -polygon proofs*. In SoCG, 27–36, 2007. 23
- [23] R. Fabila-Monroy, A. R. Vargas, and J. Urrutia. *On modern illumination problems*. In XIII Encuentros de Geometria Computacional, Zaragoza, Spain 2009. 22, 35
- [24] S. Felsner and M. Massow. *Parameters of bar k -visibility graphs*. Journal of Graph Algorithms and Applications, 12(1): 5–27, 2008. 22
- [25] J. Flum and M. Grohe. *Parameterized complexity theory*. volume 3. Springer Heidelberg, 2006. 29
- [26] R. Fulek, A. F. Holmsen, and J. Pach. *Intersecting convex sets by rays*. Discrete Computational Geometry, 42(3): 343–358, 2009. 22

- [27] S. G. Hartke, J. Vandenbussche, and P. Wenger. *Further results on bar k -visibility graphs*. SIAM Journal of Discrete Mathematics, 21(2): 523–531, 2007. [22](#)
- [28] F. Hillier and G. Lieberman. *Introduction to mathematical programming*. 1977. [30](#)
- [29] S. Hochbaum. *Approximation algorithms for the set covering and vertex cover problems*. SIAM Journal on Computing, 11(3): 555–556, 1982. [29](#)
- [30] J. Kahn, M. M. Klawe, and D. J. Kleitman. *Traditional galleries require fewer watchmen*. SIAM J. on Algebraic Disc. Methods, 4(2): 194–206, 1983. [12](#)
- [31] G. Karakostas. *A better approximation ratio for the vertex cover problem*. In Automata, Languages and Programming, Springer, 1043–1050, 2005. [29](#)
- [32] M. J. Katz and G. Morgenstern. *Guarding orthogonal art galleries with sliding cameras*. International Journal of Computational Geometry & Application, 21(2): 241–250, 2011. [18](#), [23](#), [24](#), [25](#), [26](#)
- [33] V. Klee. *Is every polygonal region illuminable from some point?* Computational Geometry: Amer. Math. Monthly, (76) 180, 1969. [21](#)
- [34] L. G. Khachiyan. *Polynomial algorithms in linear programming*. USSR Computational Mathematics and Mathematical Physics, 20(1): 53–72, 1980. [30](#)
- [35] D. T. Lee and A. K. Lin. *Computational complexity of art gallery problems*. IEEE Trans. on Inf. Theory, 32(2): 276–282, 1986. [22](#)
- [36] S.M. Lavalle, D. Lin, L.J. Guibas, J.C. Latombe and R. Motwani. *Finding an unpredictable target in a workspace with obstacles*. In Proc. IEEE int. Conf. Robotics and Automation, (1): 737–742, 1997. [18](#), [24](#), [27](#)
- [37] R. Motwani, A. Raghunathan, and H. Saran. *Covering orthogonal polygons with star polygons: the perfect graph approach*. In Proceeding of ACM SOCG, 211–223, 1988. [12](#)
- [38] S.S.Mahdavi and M. Ghodsi. *Clearing an Orthogonal Polygon to Find the Evaders*, Theoretical Computer Science, 2020. [20](#), [70](#), [77](#), [86](#)
- [39] S.S.Mahdavi and M. Ghodsi. *Clearing an x -monotone Orthogonal Polygon to Find Evaders*, Information Processing Letters, Submitted, 2018. [71](#), [84](#), [86](#)
- [40] S.S.Mahdavi, S. Seddighin and M. Ghodsi. *Covering Orthogonal Polygons with Sliding k -Transmitters*, In the Proceedings of the 26th Canadian Conference on Computational Geometry, 2014. [19](#), [20](#), [28](#), [85](#)
- [41] S.S.Mahdavi, S. Seddighin and M. Ghodsi. *Covering Orthogonal Polygons with Sliding k -Transmitters*, Theoretical Computer Science, 815:163–181, 2020. [19](#), [28](#), [40](#), [85](#)

- [42] H. Nagashima and K. Yamazaki. *Hardness of approximation for non-overlapping local alignments*. Discrete applied mathematics, 137(3):293–309, 2004. 50
- [43] Y. Nesterov, A. S. Nemirovskii, and Y. Ye. *Interior-point polynomial algorithms in convex programming*. volume 13. SIAM, 1994. 30
- [44] J. O’Rourke, *Art Gallery Theorems and Algorithms*, The International Series of Monographs on Computer Science, Oxford University Press, New York, NY, 1987. 22, 23, 34
- [45] J. O’Rourke. *Computational geometry column 52*, ACM SIGACT News, 43(1): 82–85, 2012. 17
- [46] T. D. Parsons. *Pursuit-evasion in a graph*. In Theory and Application of Graphs, Y. Alavi and D. R. Lick, Eds. Berlin, Germany: Springer-Verlag, 426–441, 1976. 18
- [47] R. M. Robinson. *Undecidability and nonperiodicity for tilings of the plane*. Inventiones Mathematicae, 12(3): 177–209, 1971. 34
- [48] S. Seddighin. *Guarding polygons with sliding cameras*. Master’s thesis, Sharif University of Technology, 2014. 20
- [49] I. Suzuki and M. Yamashita. *Visibility Searching for a Mobile Intruder in a Polygonal Region*. SIAM J. Comput. 21(5), 863–888, 1992. 27
- [50] J. Urrutia. *Art gallery and illumination problems*. In Handbook of Computational Geometry, 973–1027. North-Holland, 2000. 23, 34
- [51] D. B. West et al. *Introduction to graph theory*. volume 2. Prentice hall Englewood Cliffs, 2001. 31
- [52] C. Worman and J. M. Keil. *Polygon decomposition and the orthogonal art gallery problem*. International Journal of Computational Geometry & Application, 17(2): 105–138, 2007. 13
- [53] J. Kahn, M. M. Klawe, and D. J. Kleitman. *Traditional galleries require fewer watchmen*. SIAM J. on Algebraic Disc. Methods, 4(2): 194–206, 1983. 12

Abstract

The problem of guarding orthogonal art galleries with sliding cameras is a special case of the well-known art gallery problem when the goal is to minimize the number of guards. Each guard is considered as a point, which can guard all points that are in its visibility area. In the sliding camera model, each guard is specified by an orthogonal line segment which is completely inside the polygon. The visibility area of each sliding camera is defined by its line segment.

Inspired by advancements in wireless technologies and the need to offer wireless services to clients, a new variant of the problems for covering the regions has been studied. In this problem, a guard is modeled as an omnidirectional wireless modem with an infinite broadcast range and the power to penetrate up to k walls to reach a client. One of the problems that we study is the problem of covering orthogonal polygons with sliding cameras with the power to penetrate up to k walls. The goal in this problem is to place some k -transmitters in the polygon such that their cardinality or the total length of their line segments is minimized.

After finding the set of sliding cameras which can guard an orthogonal art gallery, the important problem is how to move these sliding cameras along their line segments to guard the art gallery. So, we study the problem of motion path planning for a group of sliding robots to guard an art gallery. In this problem, suppose that some unpredictable, moving evaders that have unbounded speed are inside the orthogonal polygon. The goal is to find the motion path planning for one or more sliding robots to find the evaders. We study this problem in an x -monotone orthogonal polygon and also in a simple orthogonal polygon, and present polynomial time algorithms for guarding them.

Keywords: 1-Computational Geometry, 2-Art Gallery Problem, 3-Motion Path Planning Problem, 4- k -transmitters, 5-Covering



Sharif University of Technology
Computer engineering department

PhD Thesis

Topic

**Covering Orthogonal Art Galleries with
Sliding k -transmitters**

By

Salma Sadat Mahdavi

Supervisor

Mohammad Ghodsi

October 2020