



مسئله‌ی مرکز با وزن مثبت و منفی روی شبکه

دانشجو: فرشته السادات میرجلالی

استاد راهنما

دکتر جعفر فتحعلی

استاد مشاور

دکتر احمد نزاکتی

پایان نامه ارشد جهت اخذ درجه کارشناسی ارشد

دی ماه ۱۳۸۷

صلى الله عليه وسلم

تقدیم به

پدر فرزانه و مادر فداکار و همسر دلسوزم

سپاسگزاری

در ابتدا از استاد بزرگوارم جناب آقای دکتر فتحعلی که برای من راهنمایی بسیار صبور و راهگشا بودند و همچنین آقای دکتر نزاکتی که ایده‌ها و نظرات حکیمانه‌ی ایشان راهگشای بنده بود، کمال سپاس و امتنان را دارم. از پدر و مادرم که هیچگاه تنه‌ایم نگذاشتند و بی‌دریغ هرآنچه در توان داشتند نثار زندگی من کردند سپاسگزارم و از همسرم که قدم به قدم همراه واقعی و همگام دلسوز من بود متشکرم. همچنین لازم می‌دانم که مراتب سپاس خود را از کارشناسان محترم برنامه‌ریزی استانداری سمنان و مدیران ستادی این نهاد که صادقانه مرا در انجام این پایان‌نامه یاری نمودند اعلام نمایم. برای تمامی این عزیزان آرزوی موفقیت و کامیابی همیشگی دارم.

پیشگفتار

بی‌شک یکی از مسائل مهم دنیای امروز بهره‌وری در کلیه‌ی امور چه در زمینه‌های علمی و چه در مسائل کاربردی می‌باشد. دقت و ثبات در تصمیم‌گیری، پایه‌ی اثربخشی و در نهایت کارایی آن می‌باشد. یکی از راه‌های مهم و منطقی در تصمیم‌گیری مکان دقیق پروژه‌های عمرانی، استفاده از علم مکان‌یابی است. علم مکان‌یابی به عنوان یک روش منطقی تصمیم‌گیری که نزدیک به ۳ قرن از پیدایش آن می‌گذرد تا به امروز نتایج قابل قبولی ارائه نموده است. در این تحقیق که قریب به یک سال پژوهش را به همراه خود داشته است پس از بیان مقدمات علم مکان‌یابی مسئله‌ی مکان‌یابی مرکز در ابعاد مختلف بررسی شده است و با استفاده از اطلاعات دقیق سرشماری سال ۱۳۸۵ و همچنین دیدگاه و نقطه‌نظرات کارشناسان مربوطه، به مکان‌یابی دقیق فرودگاه استان سمنان پرداخته شده است. امید است نتایج حاصل از این تحقیق بتواند قدم کوچکی در رشد استان داشته و پایه‌گذار کاربرد و استفاده‌ی عملی علوم نوین در تمامی عرصه‌های توسعه‌ی استان باشد.

فرشته السادات میرجلالی

دی ماه یکهزار و سیصد و هشتاد و هفت

چکیده

اولین بار در قرن هفدهم علم مکان‌یابی توسط فرما بیان شد. به این صورت که ۳ نقطه در صفحه داده شده است، نقطه‌ی چهارم به گونه‌ای اختیار شود که مجموع فاصله‌ی آن تا سه نقطه‌ی داده شده، کمینه گردد. توسعه‌ی این علم از اوایل قرن بیستم آغاز شد که امروزه راهگشای بسیاری از تصمیم‌گیری‌ها شده است. در این حیطه یکی از شاخه‌هایی که بسیار مورد توجه واقع گردیده است مکان‌یابی مرکز می‌باشد. منتها پیدا کردن شاخص‌های مؤثر و کارا در تصمیم‌گیری، یکی از نقاط ضعف این علم بود که با روش‌های دقیق ریاضی از جمله روش آنتروپی این مشکل برطرف شد. تعیین وزن رئوس شبکه‌ی تصمیم، نیازمند روشی دقیق و علمی می‌باشد که در این تحقیق از روش تاپسیس استفاده شده است. در نهایت آنچه که از نتایج این تحقیق بر طبق آمار و داده‌های سالنامه‌ی آماری سال ۱۳۸۶ استان سمنان استخراج گردید موقعیت پیشنهادی فرودگاه در ۱۶ کیلومتری غرب شهر دامغان است که این منطقه مناسب‌ترین محل جهت احداث فرودگاه می‌باشد.

کلید واژه‌ها: مکان‌یابی، مکان‌یابی مرکز، آنتروپی، تاپسیس، فرودگاه، استان سمنان.

فهرست مطالب

صفحه	عنوان
۵	چکیده
۱	فصل اول: مقدمات
2	۱ - ۱ - مقدمه
3	۱ - ۲ - تعاریف
۶	۱ - ۳ - انواع مدل‌های مکان‌یابی
۶	۱ - ۳ - ۱ - مدل‌های مکان‌یابی بیشترین فاصله
6	۱ - ۳ - ۱ - ۱ - مدل مکان‌یابی مجموعه پوششی
۷	۱ - ۳ - ۱ - ۲ - مدل مکان‌یابی مجموعه‌ی پوششی ماکسیمال
۷	۱ - ۳ - ۱ - ۳ - مدل مکان‌یابی p - مرکز
۸	۱ - ۳ - ۱ - ۴ - مدل مکان‌یابی p - پراکنده
۸	۱ - ۳ - ۲ - مدل‌های مکان‌یابی مجموع فواصل
۸	۱ - ۳ - ۲ - ۱ - مدل مکان‌یابی p - میانه
۹	۱ - ۳ - ۲ - ۲ - مدل مکان‌یابی هزینه ثابت
۹	۱ - ۳ - ۲ - ۳ - مدل مکان‌یابی نامطلوب
۹	۱ - ۳ - ۲ - ۴ - مدل مکان‌یابی واسطه‌ای
۱۰	۱ - ۳ - ۳ - مدل‌های مکان‌یابی پویا
۱۰	۱ - ۳ - ۴ - مدل‌های مکان‌یابی تصادفی
۱۱	۱ - ۳ - ۵ - فرمول‌بندی مدل مرکز و مدل p - مرکز

فصل دوم: مکان‌یابی مرکز ۱۳

- ۲ - ۱ - مسئله‌ی مرکز ۱۴
- ۲ - ۲ - مکان‌یابی مرکز روی شبکه ۱۴
- ۲-۲-۱ - مرکز نسبی ۱۴
- ۲-۲-۲ - مرکز شبکه‌ی رأس - وزین ۱۷
- ۲-۲-۲-۱ - الگوریتم مرکز نسبی در یک شبکه رأس - وزین ۲۰
- ۲-۲-۲-۲ - پیچیدگی زمانی ۲۱
- ۲-۲-۲-۳ - الگوریتم مرکز روی شبکه‌ی رأس - وزین ۲۲
- ۲-۲-۲-۴ - پیچیدگی زمانی ۲۲
- ۲-۲-۳ - مرکز شبکه‌ی رأس - غیروزین ۲۲
- ۲-۲-۳-۱ - الگوریتم مرکز نسبی روی یک شبکه‌ی رأس - غیر وزین ۲۷
- ۲-۲-۳-۲ - الگوریتم مرکز روی شبکه رأس - غیر وزین ۲۹
- ۲ - ۳ - مکان‌یابی مرکز روی درخت ۲۹
- ۲-۳-۱ - مکان‌یابی مرکز روی درخت غیر وزین ۲۹
- ۲-۳-۱-۱ - الگوریتم مکان‌یابی مرکز روی درخت غیر وزین ۲۹
- ۲-۳-۱-۲ - پیچیدگی زمانی ۳۰
- ۲-۳-۱-۳ - مثال ۳۰
- ۲-۳-۲ - مکان‌یابی ۲ - مرکز روی درخت غیر وزین ۳۱
- ۲-۳-۲-۱ - الگوریتم مکان‌یابی ۲ - مرکز روی درخت غیر وزین ۳۱
- ۲-۳-۲-۲ - پیچیدگی زمانی ۳۲

- ۳۲.....مثال ۳-۲-۳-۲
- ۳۳.....مکان‌یابی مرکز روی درخت وزین ۳-۳-۲
- ۳۳.....روش مرکز ثقل برای یافتن مرکز روی درخت وزین ۱-۳-۳-۲
- ۳۴.....الگوریتم پیشنهادی ۲-۳-۳-۲
- ۳۴.....پیچیدگی زمانی ۳-۳-۳-۲
- ۳۵.....مرکز ثقل درخت ۴-۳-۳-۲
- ۳۶.....الگوریتم مرکز ثقل درخت ۵-۳-۳-۲
- ۳۶.....مثال ۶-۳-۳-۲
- ۳۸.....الگوریتم مکان‌یابی مرکز روی درخت وزین(روش مرکز ثقل) ۷-۳-۳-۲
- ۳۹.....پیچیدگی زمانی ۸-۳-۳-۲
- ۴۱.....**فصل سوم : مکان‌یابی P - مرکز**
- ۴۲.....۱ - ۳ مکان‌یابی P - مرکز
- ۴۲.....۱-۱-۳ مکان‌یابی P - مرکز در یک شبکه‌ی رأس- وزین و شبکه‌ی رأس- غیر وزین
- ۴۴.....۱-۱-۳-۱ الگوریتم مکان‌یابی P - مرکز در یک شبکه‌ی رأس- وزین و شبکه‌ی رأس- غیر وزین
- ۴۵.....۲-۱-۳-۱ پیچیدگی زمانی الگوریتم
- ۴۶.....۲-۱-۳ مجموعه‌ی احاطه‌گر با شعاع r روی درخت رأس- وزین
- ۴۸.....۱-۲-۱-۳ الگوریتم مجموعه‌ی احاطه‌گر با شعاع r روی یک درخت رأس- وزین
- ۵۱.....۳-۱-۳ P - مرکز درخت رأس- وزین
- ۵۳.....۱-۳-۱-۳ الگوریتم P - مرکز یک درخت رأس- وزین
- ۵۴.....۲-۳-۱-۳ پیچیدگی زمانی الگوریتم

۵۴	۳-۱-۴ مکان‌یابی P - مرکز روی درخت رأس- غیر وزین
۵۵	۳-۱-۴-۱ الگوریتم P - مرکز مطلق روی درخت رأس- غیر وزین
۵۶	۳-۱-۴-۲ پیچیدگی زمانی الگوریتم
۵۷	۳-۱-۴-۳ الگوریتم P - مرکز رأسی روی درخت رأس- غیر وزین
۵۸	۳-۱-۴-۴ پیچیدگی زمانی الگوریتم
۶۰	فصل چهارم : مکان‌یابی مرکز با وزن مثبت و منفی در شبکه
۶۱	۴-۱ مکان‌یابی مرکز با وزن مثبت / منفی
۶۲	۴-۱-۱ مکان‌یابی مرکز با وزن مثبت و منفی روی شبکه‌ای کلی
۶۳	۴-۱-۲ مکان‌یابی مرکز نامطلوب روی مسیر
۶۴	۴-۱-۲-۱ الگوریتم بدست آوردن ساختار g_j^+ با استفاده از g_{j-1}^+
۶۵	۴-۱-۲-۲ پیچیدگی زمانی الگوریتم
۶۷	فصل پنجم : کاربرد مدل ترکیبی تاپسیس و مکان‌یابی مرکز درخت وزین
۶۸	۵-۱ مقدمات
۷۲	۵-۲ مفاهیم به کار رفته در این فصل
۷۲	۵-۲-۱ تعریف شهرستان
۷۲	۵-۲-۲ تعریف شهر
۷۲	۵-۲-۳ ایستگاه‌های سینوپتیک (ایستگاه‌های هواگزاری همزمان)
۷۲	۵-۳ روش تحقیق
۷۳	۵-۴ هدف تحقیق
۷۳	۵-۵ تصمیم‌گیری چند شاخصه

- ۵ - ۶ - مراحل حل مسائل تصمیم‌گیری با استفاده از مدل‌های چند شاخصه ۷۵
- ۵-۶-۱- بی‌مقیاس کردن (یکسان‌سازی مقیاس) شاخص‌ها ۷۵
- ۵-۶-۱- روش بی‌مقیاس کردن با استفاده از « نرم » ۷۵
- ۵-۶-۲- تعیین وزن شاخص‌ها ۷۶
- ۵-۶-۲-۱- روش آنتروپی ۷۶
- ۵-۶-۳- حل مسئله تصمیم‌گیری بوسیله مدل‌های تصمیم‌گیری چند شاخصه ۷۸
- ۵-۶-۳-۱- مدل غیرجبرانی ۷۸
- ۵-۶-۳-۲- مدل جبرانی ۷۹
- ۵-۶-۳-۲-۱- روش زیر گروه سازشی ۷۹
- ۵-۶-۳-۲-۱- روش تاپسیس ۸۰
- ۵-۶-۳-۲-۱- الگوریتم تاپسیس ۸۰
- ۵-۶-۴- مراحل اجرای روش تاپسیس جهت تعیین محل احداث فرودگاه در استان سمنان ۸۲
- ۵-۶-۴-۱- انتخاب شاخص‌ها و تعیین ایده‌آل‌ها ۸۲
- ۵-۶-۴-۲- تعیین وزن شاخص‌ها به روش آنتروپی ۸۴
- ۵-۶-۴-۳- مکان‌یابی مرکز درخت رأس- وزین استان سمنان ۸۹
- منابع و مراجع ۹۳

فهرست جداول

عنوان	صفحه
جدول ۵-۱- شاخص‌های منتخب توسط کارشناسان	۸۳
جدول ۵-۲- ماتریس داده‌ها	۸۴
جدول ۵-۳- ماتریس نسبت داده‌ها	۸۵
جدول ۵-۴- بدست آوردن مقدار E_j	۸۵
جدول ۵-۵- بدست آوردن مقدار W_j	۸۵
جدول ۵-۶- تشکیل ماتریس بی‌مقیاس شده از ماتریس داده‌ها	۸۶
جدول ۵-۷- ماتریس بی‌مقیاس شده N_D	۸۶
جدول ۵-۸- تشکیل ماتریس وزین V	۸۷
جدول ۵-۹- فاصله از گزینه ایده آل مثبت (d_{i^+})	۸۷
جدول ۵-۱۰- فاصله از گزینه ایده آل منفی (d_{i^-})	۸۸
جدول ۵-۱۱- وزن شهرها	۸۹
جدول ۵-۱۲- نتیجه‌ی نهایی مدل ترکیبی تاپسیس و مرکز	۹۲

فهرست شکل‌ها

صفحه

عنوان

۱۵	 شکل ۱-۲- نمودار $D_e(v, t)$
۶۴	 شکل ۱-۴- نمودار g_{j-1}^+
۶۹	 شکل ۱-۵- موقعیت استان سمنان در کشور
۷۰	 شکل ۲-۵- موقعیت شهرهای استان سمنان
۷۱	 شکل ۳-۵- موقعیت و نوع راه‌های مواصلاتی بین شهرهای استان سمنان

فصل اول

مقدمـات

امروزه روش‌های بهینه‌سازی، از پرکاربردترین روش‌های مدیریت تخصیص منابع می‌باشد که در این عرصه، رفته‌رفته علم مکان‌یابی، جایگاه خود را مستحکم‌تر و مستدل‌تر می‌کند. به طور کلی می‌توان گفت: وقتی می‌خواهیم یک سری تجهیزات و تسهیلات را در یک منطقه به گونه‌ای مستقر نماییم تا بهترین و بیشترین استفاده از این حداقل امکانات را برده باشیم، از علم مکان‌یابی جهت رسیدن به اهداف خود سود می‌بریم. در قرن هفدهم، فرما^۱ مسئله مکان‌یابی را به صورت زیر مطرح نمود:

فرض کنید ۳ نقطه در صفحه داده شده است، نقطه‌ی چهارم را به گونه‌ای بیابید که مجموع فاصله‌های آن تا سه نقطه‌ی داده شده، کمینه شود. توریچلی^۲ در سال ۱۶۴۰ این مسئله را حل نموده، لذا نام این نقطه را نقطه‌ی توریچلی گذاشتند. در سال ۱۹۰۹ اولین نظریه‌ی مکان‌یابی مدرن با مقاله وبر^۳ [۳۸] شکل گرفت. در سال ۱۹۶۴ حکیمی^۴ تابع هدف را به دو صورت کمترین مجموع^۵ و مینیماکس^۶، طبقه‌بندی نمود [۱۵]. اولین طبقه‌بندی مدل‌های مختلف مکان‌یابی توسط هندلر^۷ و میرچندانی^۸ [۱۹] ارائه شد.

پس از آن هامچر^۹ و نیکل^{۱۰} [۱۸] و افرادی مثل تانسل^{۱۱} و همکاران [۳۶]، کراروپ^{۱۲} و پروزن^{۱۳} [۲۵]، هانسن^{۱۴} و همکاران [۲۱]، اون^{۱۵} و دسکین^{۱۶} [۳۲]، کاپانرا^{۱۷} [۵]، اسکاپارا^{۱۸} و اسکاتلا^{۱۹} [۳۳] بررسی

¹ Fermat

² Torricelli

³ Weber

⁴ Hakimi

⁵ Minisum

⁶ Minimax

⁷ Handler

⁸ Mirchandani

⁹ Hamacher

¹⁰ Nickel

¹¹ Tansel

¹² Krarup

¹³ Pruzan

¹⁴ Hansen

¹⁵ Owen

¹⁶ Daskin

مسائل مختلف مکان‌یابی را انجام داده‌اند. کارنت^{۲۰} و همکاران [۸] نیز کاربردهای مسائل مکان‌یابی را بیان نموده‌اند.

اما اخیراً، آنچه که توجه زیادی را به خود جلب نموده است، مکان‌یابی نامطلوب^{۲۱} و همچنین مکان‌یابی ترکیبی مطلوب و نامطلوب می‌باشد، که کاربردهای زیادی از جمله در تعیین محل استقرار آشغال‌های هسته‌ای دارد. در این زمینه بورکارد و دولانی^{۲۲} [۳] مقالات خوب و جامعی را ارائه نموده‌اند. در ادامه به تعریف مسائل مکان‌یابی و نیز انواع طبقه‌بندی آن می‌پردازیم.

۱-۲- تعاریف

در این پایان نامه تعاریف زیر مورد نیاز می‌باشند که قسمتی از آنها از مرجع [۴۲] گرفته شده است.

تعریف ۱-۲-۱

گراف^{۲۳} $G(V, E)$ مجموعه‌ای از رئوس V و مجموعه‌ای از یال‌های E است که به هر عضو e از مجموعه E یک زوج (نه لزوماً مرتب) $(a, b) \in V \times V$ نظیر می‌شود. a, b را نقاط انتهایی یال e گویند.

تعریف ۱-۲-۲

یک شبکه^{۲۴} گرافی است که به مجموعه رئوس آن، مجموعه گره‌ها و به مجموعه یال‌های آن، مجموعه کمان‌ها گفته می‌شود و هر کدام از کمان‌ها دارای برچسب‌های عددی هستند. این برچسب‌ها معمولاً نشان دهنده‌ی هزینه، ظرفیت یا عرضه و تقاضا هستند. در شبکه ممکن است خود رأس‌ها نیز برچسب

¹⁷ Cappanera

¹⁸ Scaparra

¹⁹ Scutella

²⁰ Current

²¹ The maxi sum location problem

²² Dollani

²³ Graph

²⁴ Network

گذاری شوند که نشان‌دهنده‌ی وزن رئوس خواهد بود. اگر رأس‌ها دارای وزن باشند به شبکه رأس-وزین^{۲۵} و در غیر این صورت رأس-غیروزین^{۲۶} اطلاق می‌شود.

تعریف ۱-۲-۳

فاصله‌ی بین دو رأس x و y در گراف G با $d(x, y)$ نمایش داده می‌شود که برابر با مجموع طول یال‌های موجود در کوتاه‌ترین مسیر بین x و y می‌باشد. همچنین فاصله‌ی بین رأس x تا زیرمجموعه‌ی X ، برابر با کمترین فاصله‌ی رأس x نسبت به رأس‌های موجود در X می‌باشد که با $d(x, X)$ نشان داده می‌شود. لذا

$$d(x, X) = \min \{d(x, y) : y \in X\}$$

تعریف ۱-۲-۴

گشت^{۲۷} یک گشت در گراف متشکل از دنباله‌ای از k یال است که به صورت $uv, vw, \dots, yz$ می‌باشد. مسیر^{۲۸}، گشتی است که علاوه بر یال‌های متمایز، رأس‌های متمایز نیز دارد. دور^{۲۹}، یک مسیری باشد که رأس ابتدا و انتهای آن یکی باشد.

تعریف ۱-۲-۵

درخت^{۳۰}، گرافی است که دور ندارد و هر دو رأس آن دقیقاً با یک مسیر به هم متصل می‌باشد. درخت را با $T(V, E)$ نمایش می‌دهند.

^{۲۵} Vertex-weighted network

^{۲۶} Vertex-unweighted network

^{۲۷} Walk

^{۲۸} Path

^{۲۹} Cycle

^{۳۰} Tree

تعریف ۱-۲-۶

از مهم‌ترین نکات طراحی یک الگوریتم، بحث پیچیدگی زمان^{۳۱} یا زمان اجرای کامل الگوریتم می‌باشد. این زمان به تعداد عملیات ریاضی و منطقی موجود در الگوریتم‌ها وابستگی مستقیم دارد. به طور ساده پیچیدگی زمانی یک الگوریتم را می‌توان تعداد دفعات اجرای دستور اصلی (یا گروهی از دستورهای اصلی) به کار رفته در الگوریتم و نیز پیش پردازش‌های الگوریتم تعریف کرد.

تعریف ۱-۲-۶-۱

از یک دیدگاه کلی هر تابعی با دامنه‌ی اعداد صحیح نامنفی و برد اعداد حقیقی نامنفی یک تابع پیچیدگی تعریف می‌شود.

تعریف ۱-۲-۶-۲

برای یک تابع پیچیدگی مفروض $f(n)$ ، عبارت $O(f(n))$ عبارت است از مجموعه‌ی همه‌ی توابع پیچیدگی مانند $g(n)$ که برای آن یک عدد ثابت حقیقی مثبت C و یک عدد صحیح نامنفی N وجود داشته باشد که

$$\forall n \geq N ; g(n) \leq C \cdot f(n)$$

در این صورت اگر $g(n) \in O(f(n))$ آنگاه گفته می‌شود $g(n)$ از مرتبه‌ی بزرگ $f(n)$ است.

تعریف ۱-۲-۶-۳

برای تابع پیچیدگی $f(n)$ ، $\Omega(f(n))$ مجموعه‌ی همه‌ی توابع پیچیدگی چون $g(n)$ است که برای آن یک ثابت حقیقی مثبت C و یک عدد صحیح نامنفی N وجود داشته باشد که

$$\forall n \geq N ; g(n) \geq C \cdot f(n)$$

در این صورت اگر $g(n) \in \Omega(f(n))$ ، گفته می‌شود $g(n)$ از مرتبه‌ی اُمگای بزرگ $f(n)$ است.

³¹ Complexity time

تعریف ۱-۲-۴

برای تابع پیچیدگی $f(n)$ ، $\theta(f(n))$ مجموعه همه توابع پیچیدگی چون $g(n)$ است که برای آن ثابت‌های حقیقی و مثبت C و D و نیز یک عدد صحیح نامنفی N چنان یافت شود که

$$\forall n \geq N \quad ; \quad C.f(n) \leq g(n) \leq D.f(n)$$

و یا به بیان دیگر

$$\theta(f(n)) = O(f(n)) \cap \Omega(f(n))$$

در این صورت اگر $g(n) \in \theta(f(n))$ باشد می‌گوییم $g(n)$ از مرتبه‌ی تتای $f(n)$ است.

۱-۳-انواع مدل‌های مکان‌یابی

در حالت کلی مدل‌های مکان‌یابی در دو دسته‌ی مکان‌یابی بیشترین فاصله و مکان‌یابی مجموع فواصل طبقه‌بندی می‌شود. مدل‌های مکان‌یابی را می‌توان در دو دسته‌ی تصادفی و پویا نیز طبقه‌بندی کرد.

۱-۳-۱-مدل‌های مکان‌یابی بیشترین فاصله^{۳۲}

مدل‌های مکان‌یابی بیشترین فاصله شامل ۴ گروه می‌باشد که در زیر به معرفی آنها پرداخته‌ایم.

۱-۳-۱-۱-مدل مکان‌یابی مجموعه پوششی^{۳۳}

اولین گروه از مدل‌های بیشترین فاصله، مکان‌یابی مجموعه پوششی می‌باشد. در این مسئله هدف کمینه کردن تعداد تسهیلات مورد نیاز جهت تجهیز و پوشش کلیه‌ی گره‌های تقاضا می‌باشد.

³² Maximum Distance Models

³³ Set covering location model

۱-۳-۲- مدل مکان‌یابی مجموعه‌ی پوششی ماکسیمال^{۳۴}

یکی از فرضیات اصلی مسئله‌ی مکان‌یابی مجموعه پوششی این بود که کلیه‌ی گره‌های تقاضا می‌بایست پوشش داده شوند و هیچ‌گونه محدودیتی در هزینه‌ها نداشتیم. اما هدف این مدل قرار دادن تعداد از پیش معینی از تسهیلات مثلاً p تا در شبکه است به گونه‌ای که بیشترین گره‌های تقاضا پوشش داده شوند. برای کسب اطلاعات بیشتر شما را به مرجع [۶] ارجاع می‌دهیم.

۱-۳-۳- مدل مکان‌یابی p -مرکز^{۳۵}

در مسئله‌ی p -مرکز [۱۵] و [۱۶] بیشترین فاصله‌ی گره‌های تقاضا را از نزدیک‌ترین محل استقرار تسهیلاتی که ما آنها را به تعداد از پیش تعیین‌شده‌ای قرار است مکان‌یابی نماییم، کمینه می‌کنیم. حالت رأسی مدل مکان‌یابی p -مرکز، مسئله را به استقرار تسهیلات جدید در گره‌های شبکه محدود می‌نماید. در حالیکه در مدل مطلق مکان‌یابی p -مرکز، تسهیلات جدید در هر نقطه از شبکه می‌توانند واقع گردند. همچنین هر دو نسخه‌ی p -مرکز رأسی^{۳۶} و p -مرکز مطلق^{۳۷}، می‌توانند رأس-وزین یا رأس-غیروزین باشند. که در حالت غیروزین، وزن کلیه‌ی گره‌ها یکسان فرض می‌شود. ولی در حالت وزین هر رأس وزن مشخصی دارد. در این پایان‌نامه منظور از شبکه و درخت وزین، شبکه و درخت رأس-وزین و همچنین منظور از شبکه و درخت غیروزین شبکه و درخت رأس-غیروزین می‌باشد. به عنوان مثال این وزن می‌تواند نشان‌دهنده‌ی اهمیت یک گره و یا سطح نیاز آن گره باشد. به مسئله‌ی p -مرکز رأسی، گسسته^{۳۸} و p -مرکز مطلق، پیوسته^{۳۹} هم می‌گویند.

³⁴ Maximum covering location problem

³⁵ P-center model

³⁶ Vertex p-center

³⁷ Absolute p-center

³⁸ Discrete

³⁹ Continuous

۱-۳-۴- مدل مکان‌یابی p - پراکنده^{۴۰}

در تمامی حالات بحث شده در بالا، آنچه که مورد توجه واقع شده بود فاصله‌ی بین نقاط تقاضا و محل استقرار تسهیلات بود. اما یک فرض ناگفته باقی می‌ماند و آن این است که نزدیکی تسهیلات جدید به یکدیگر برای ما ناخوشایند و نامطلوب باشد. این مسئله دو تفاوت با مسائل قبلی دارد. اول اینکه فاصله‌ی بین تسهیلات جدید مطرح است و دوم اینکه کمترین فاصله‌ی بین هر جفت از امکانات را بیشینه می‌نماید. مثلاً محل استقرار پایگاه‌های نظامی می‌بایست از هم دور باشند تا حمله به آنها، مشکل گردد. [۲۶]

۱-۳-۲- مدل‌های مکان‌یابی مجموع فواصل^{۴۱}

مدل‌های مکان‌یابی مجموع فواصل شامل ۴ گروه می‌باشد که در زیر به معرفی آنها پرداخته‌ایم.

۱-۳-۱- مدل مکان‌یابی p - میانه^{۴۲}

یکی از مدل‌های کلاسیک در این حیطه، مدل p - میانه [۱۵] و [۱۶] می‌باشد که مکان استقرار تسهیلات موجود به تعداد p تا را به گونه‌ای پیدا می‌نماید که مجموع فواصل وزین بین گره‌های تقاضا و تسهیلات را کمینه کند. مسئله‌ی p - میانه دارای ۳ فرض مهم می‌باشد. اولین فرض این است که هر کدام از محل‌های کاندید، جهت استقرار تسهیلات در آنها، هزینه‌ی ثابت مشابهی دارند. دومین فرض این است که تسهیلاتی را که می‌خواهیم مکان‌یابی نماییم هیچگونه ظرفیت محدودی در میزان سرویس‌دهی متقاضیان نداشته باشند و بالاخره فرض سوم این است که تعداد تسهیلات و امکاناتی را که قرار است مکان‌یابی نماییم از قبل مشخص و معلوم باشند.

⁴⁰ The p -dispersion problem

⁴¹ Total distance model

⁴² P -median model

۱-۳-۲- مدل مکان‌یابی هزینه ثابت^{۴۳}

در مدل مکان‌یابی هزینه ثابت، تمامی فرضیات مدل مکان‌یابی p - میانه نقض می‌شود. هدف این مسئله کمینه کردن تعداد تجهیزات جدید مستقر شده در شبکه و همچنین مجموع هزینه‌های حمل و نقل می‌باشد. در واقع تعداد بهین تجهیزات لازم و مکان‌های بهینه استقرار تسهیلات را معین می‌کند. همچنین برای توانایی سرویس دهی تجهیزات و تسهیلات جدید، ظرفیتی قائل شده و لذا ممکن است یک متقاضی همانند مدل‌های قبلی تحت پوشش نزدیک‌ترین پایگاه تسهیلاتی خود قرار نگیرد. [۱]

۱-۳-۲- مدل مکان‌یابی نامطلوب

در سایر مدل‌های مکان‌یابی مجموع فواصل، فرض بر این است که امکانات تا جایی که ممکن است به گره‌های تقاضا نزدیک‌تر باشد. اما برای تسهیلات نامطلوب عکس این مطلب برقرار است بدین صورت که در مسئله‌ی نامطلوب، هدف، یافتن p نقطه جهت استقرار تسهیلات و وسایل جدید می‌باشد به گونه‌ای که مجموع فواصل وزین ما بین وسایل جدید و گره‌های نیاز و تقاضا، بیشینه گردد. [۹]

۱-۳-۴- مدل مکان‌یابی واسطه‌ای^{۴۴}

در بسیاری از سیستم‌های شبکه‌های فرودگاهی و شبکه‌های حمل و نقل کالا، این نوع مدل‌بندی را به کار می‌بندند. که هدفش بیشتر کردن ظرفیت حمل و نقل و یا سریع‌تر نمودن چرخه‌ی حمل و نقل می‌باشد. در این مدل هزینه‌ی حمل و نقل کالاها بین واسطه‌ها (گره‌ها) کمینه می‌شود در حالی که میزان مجاز مبادله‌ی کالاها بین واسطه‌ها از قبل مشخص می‌باشد. این سیستم متوسط هزینه‌ی حمل و نقل در هر مایل و نیز زمان کلی حمل و نقل را کاهش می‌دهد. جهت اطلاعات بیشتر شما را به مراجع [۴] و [۲۳] و [۲۴] ارجاع می‌دهیم.

⁴³ Fixed charge location problem

⁴⁴ Hub location problems

۱-۳-۳- مدل‌های مکان‌یابی پویا^{۴۵}

دراویل بررسی مسائل مکان‌یابی، محققان به مدل‌های مکان‌یابی پویا پرداختند. در این نوع مدل‌ها تمامی پارامترهای ما یا نسبت به زمان ثابت بوده و یا به اندازه‌ی مشخصی تغییر می‌کنند. کارنت در سال ۱۹۹۸ مدل‌های مکان‌یابی را در دو دسته‌ی ضمنی^{۴۶} و صریح^{۴۷} طبقه‌بندی نمود. در مدل‌های ضمنی پارامترهای ما نسبت به زمان تغییر می‌کنند و دنبال این هستیم که ببینیم این تغییرات چه تأثیری در مکان‌یابی وسایل جدید خواهد گذاشت. برای مطالعات بیشتر در زمینه‌ی مدل‌های مکان‌یابی پویا شما را به مرجع [۱۰] ارجاع می‌دهیم.

۱-۳-۴- مدل‌های مکان‌یابی تصادفی^{۴۸}

در اکثر مدل‌های مکان‌یابی، فرض بر این است که پارامترهای مسئله به صورت قطعی شناخته شده باشند. حتی فرض می‌شود که تغییرات پارامترها در طی زمان کاملاً مشخص باشد. اما آنچه که قابل توجه می‌باشد این است که در واقعیت در بیشتر مسائل مکان‌یابی، عدم قطعیت وجود دارد. در دراز مدت تجهیزات، تقاضا، زمان طی مسافت‌ها، هزینه‌ی تجهیزات و حتی فواصل ممکن است تغییر نمایند. که این تغییرات اغلب تصادفی می‌باشند. در مورد مکان‌یابی تصادفی علاقمندان را به مرجع [۱۳] در مورد تقاضا، به مرجع [۲۹] در مورد زمان طی مسافت و به مرجع [۹] در مورد توانایی سرویس‌دهی ارجاع می‌دهیم. البته با گسترش علم مکان‌یابی در سال‌های اخیر، مکان‌یابی فازی را هم می‌توان در کنار مکان‌یابی پویا و مکان‌یابی تصادفی قرار داد.

⁴⁵ Dynamic location models

⁴⁶ implicitly

⁴⁷ Explicitly

⁴⁸ Stochastic location models

۱-۴- فرمول بندی مدل مرکز^{۴۹} و مدل p -مرکز^{۵۰}

طبق تعریفی که در [۱۲] آمده است فرض کنید n نقطه‌ی $p_1, p_2, \dots, p_n$ در شبکه $G(V, E)$ موجودند و به ترتیب دارای وزن‌های $w_1, \dots, w_n$ هستند. می‌خواهیم نقطه‌ای مانند $x \in G$ را به گونه‌ای بیابیم تا فاصله‌ی وزنی x تا دورترین نقطه‌ی موجود روی G می‌نیم شود به صورت زیر

$$\min_{x \in G} \max_{i=1, \dots, n} w_i d(x, p_i)$$

اما اگر به جای یک نقطه دنبال چند مکان برای وسایل جدید باشیم یعنی $X = \{x_1, \dots, x_p\}$ مجموعه‌ی شامل p وسیله‌ی جدید باشد که باید مکان‌یابی شوند تابع هدف به صورت زیر خواهد بود.

$$\min(F(X) = \max_{i=1, \dots, n} w_i d(X, p_i))$$

$$d(X, p) = \min_{x_i \in X} d(x_i, p) \quad \text{که در آن}$$

در فصل‌های بعدی ابتدا به مکان‌یابی مطلوب روی شبکه‌های رأس-وزین و رأس-غیرووزین می‌پردازیم. با فرض اینکه n تعداد رئوس و m تعداد یال‌های شبکه باشد، به بررسی الگوریتمی با پیچیدگی زمانی $O(m.n \log n)$ برای مکان‌یابی مرکز مطلق روی شبکه‌ی رأس-وزین خواهیم پرداخت و همچنین الگوریتمی با پیچیدگی زمانی $O(m.n + n^2 \cdot \log n)$ برای پیدا کردن مرکز مطلق یک شبکه‌ی رأس-غیروزین را تحلیل خواهیم نمود. که البته فرض بر این است که ماتریس فاصله‌ی بین رئوس از قبل داده شده است. سپس به مکان‌یابی مرکز مطلق یا رأسی روی درخت رأس-وزین و رأس-غیروزین می‌پردازیم. الگوریتم ارائه شده برای حالت وزین دارای پیچیدگی زمانی $O(n \cdot \log n)$ و در حالت غیر وزین دارای پیچیدگی زمانی $O(n)$ می‌باشد. در ادامه به مکان‌یابی p -مرکز، $p > 1$ ، می‌پردازیم و الگوریتم‌هایی با

⁴⁹ 1- center

⁵⁰ P- center

پیچیدگی زمانی $O\left(\frac{m^p \cdot n^{2p-1}}{(p-1)!} \text{Log} n\right)$ در حالت رأس-وزین و پیچیدگی زمانی $O\left(\frac{m^p \cdot n^{2p-1}}{(p-1)!}\right)$ در حالت رأس-غیروزین بررسی خواهیم نمود. همچنین در مورد شبکه‌ای که یک درخت رأس-غیروزین باشد بحث نموده و الگوریتمی با پیچیدگی زمانی $O(n^{p-1})$ را، برای حالت p - مرکز مطلق، تحلیل خواهیم کرد. و نیز الگوریتم p - مرکز رأسی درخت رأس-غیروزین قید شده دارای پیچیدگی زمانی $O(n \cdot \log^{p-1} n)$ می‌باشد. در مورد مجموعه‌ی احاطه‌گر^{۵۱} هم صحبت خواهیم نمود که الگوریتمی با پیچیدگی زمانی $O(n)$ دارد. اما آنچه که در این تحقیق حائز اهمیت است نمونه‌ی کاربردی مکان‌یابی در تعیین محل دقیق احداث فرودگاه در استان سمنان می‌باشد که با استفاده از شاخص‌های موثری که توسط کارشناسان خبره ارائه گردیده است و نیز روش‌های آنروپی و تاپسیس^{۵۲} به حل این مسئله‌ی کاربردی پرداخته می‌شود.

⁵¹ Dominating set

⁵² Topsis

فصل دوم

مکان‌یابی مرکز

۲-۱- مسئله‌ی مرکز

در این مسئله، می‌خواهیم یک وسیله‌ی جدید را به گونه‌ای در شبکه مستقر نماییم که بیشینه فاصله وزین این وسیله تا هر کدام از رئوس شبکه، کمینه گردد. به عنوان مثال شبکه‌ی بزرگراه‌ها را در نظر بگیرید می‌خواهیم یک مرکز اورژانسی را در شبکه استقرار دهیم به گونه‌ای که حداکثر زمان دسترسی دورترین نقطه‌ی نیاز، کمینه گردد.

۲-۲- مکان‌یابی مرکز روی شبکه

اگر بخواهیم مرکز یک شبکه‌ی کلی را پیدا کنیم ابتدا می‌بایست مرکز نسبی شبکه را روی تک تک یال‌ها بدست آورده و بین مرکز نسبی‌ها بهترین جواب را انتخاب نماییم. مطالب این بخش از مرجع [۲۲] گرفته شده است.

۲-۲-۱- مرکز نسبی^{۵۳}

فرض کنید $G(V, E)$ ، یک شبکه‌ی غیر جهت‌دار باشد که به هر کدام از رئوس آن عدد نامنفی $w(v)$ و همچنین به هر کدام از یال‌های آن عدد مثبت $l(e)$ نسبت داده شده است. فرض کنید $X_p = \{x_1, \dots, x_p\}$ مجموعه‌ی p نقطه روی G باشد، در حالیکه یک نقطه روی G به این مفهوم می‌باشد که می‌تواند روی یال‌های G یا روی رئوس قرار گیرد. اکنون تابع $F(X_p)$ را به صورت زیر تعریف می‌نماییم

$$F(X_p) = \max_{v \in V} \{w(v).d(v, X_p)\} \quad (۱-۲)$$

مرکز نسبی شبکه‌ی G روی یال $e \in E$ ، نقطه‌ای مانند $x^*(e)$ روی یال e می‌باشد به گونه‌ای که

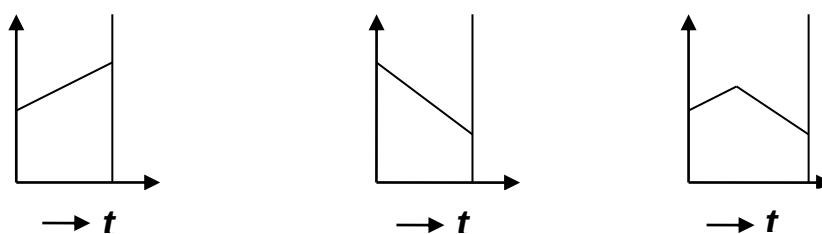
$$F(x^*(e)) = \min_{x(e) \text{ on } e} \{F(x(e))\}$$

⁵³ Local center

$F(x^*(e))$ ، شعاع نسبی^{۵۴} G روی یال e نامیده می‌شود و با $r(e)$ نمایش داده می‌شود. با توجه به تعریف ۱-۲ واضح است که اگر $r_1 = r(e_j) = \min_{e_i \in E} \{r(e_i)\}$ که $E = \{e_1, \dots, e_m\}$ باشد آنگاه r_1 ، شعاع G و نقطه‌ی متناظرش $x^*(e_j)$ ، مرکز شبکه‌ی G می‌باشد. پس برای یافتن مرکز شبکه‌ی G کافی است، مرکز نسبی شبکه را روی هر کدام از یال‌ها بیابیم. برای پیدا کردن مرکز نسبی شبکه‌ی G روی یال $e = (v_r, v_s)$ ، ابتدا فاصله‌ی وزین هر رأس $v \in V$ را تا یک نقطه دلخواه $x(e)$ روی یال e بدست می‌آوریم، بدین ترتیب که اگر $t = t(x(e))$ نشان دهنده‌ی فاصله‌ی $x(e)$ از v_r در امتداد یال e باشد آنگاه فاصله‌ی وزین نقطه‌ی دلخواه $x(e)$ از رأس v به صورت زیر، بدست خواهد آمد

$$D_e(v, t) = w(v) \cdot \min \{t + d(v_r, v), (l(e) - t + d(v_s, v))\} \quad (۲-۲)$$

نمودار $D_e(v, t)$ ، $0 \leq t \leq l(e)$ ، در شکل شماره ۱-۲ نشان داده شده است. هر نمودار شامل یک یا دو قطعه خط راست است که شیب این خط‌ها $w(v)$ می‌باشد. اگر نمودار شامل دو قطعه خط شد، نقطه‌ی شکست، جایی که دو قطعه خط همدیگر را قطع می‌کنند، همان نقطه‌ای است که $D_e(v, \cdot)$ به بیشینه مقدار خود می‌رسد.



شکل ۱-۲ - نمودار $D_e(v, t)$

اکنون تعریف می‌کنیم

⁵⁴ Local radius

$$D_e(t) = \max_{v \in V} \{D_e(v, t)\} \quad (3-2)$$

واضح است نقطه‌ای مثل $x^*(e)$ که در آن $D_e(\cdot)$ به کمینه مقدار خودش بر روی یال e برسد، مرکز نسبی روی e خواهد بود و مقدار $D_e(\cdot)$ در این نقطه شعاع نسبی روی یال e می باشد. علاوه بر این اگر $t^* = t(x^*(e))$ ، مقدار t در نقطه‌ی $x^*(e)$ باشد، آنگاه t^* مقداری است که دو شرط زیر برایش برقرار است :

(الف) $t^* = 0$ یا $t^* = l(e)$ و یا با فرض اینکه v_i و v_j دو رأس دلخواه باشند، نقطه‌ای است که دو تابع $D_e(v_i, \cdot)$ و $D_e(v_j, \cdot)$ که در t^* دارای شیب با علامت مخالف می باشند، همدیگر را قطع نمایند.

(ب) اگر T^* مجموعه همه نقاطی باشد که در شرط (الف) صدق می کنند آنگاه

$$D_e(t^*) = \min_{t' \in T^*} \{D_e(t')\}$$

از آنجایی که نمودارهای $D_e(v, t)$ خطی هستند لذا $D_e(t)$ نیز به صورت قطعه‌های خطی می باشد که هر خط از یک $D_e(v_i, t)$ گرفته می شود، تا زمانی که $D_e(v_i, t)$ جای خودش را با یک $D_e(v_j, t)$ بعد از اینکه مقدارش با آن یکی شد، عوض کند. لذا تلاقی $D_e(v_i, t)$ و $D_e(v_j, t)$ می تواند برای کمینه کردن $D_e(t)$ کاندید شود. از طرفی واضح است که $t = l(e), t = 0$ هم می توانند کاندید باشند.

در یک شبکه با n رأس، بر اساس شرط (الف)، حداکثر $\frac{n(n-1)}{2} + 2$ نقطه کاندید مرکز نسبی می -

شوند. پس با فرض اینکه ماتریس فاصله موجود باشد برای محاسبه‌ی $D_e(\cdot)$ ، $O(n)$ تکرار در هر کدام از

$O(n^2)$ نقاط کاندید، لازم است، که در نهایت با اعمال شرط (ب) در $O(n^3)$ تکرار به جواب می رسیم.

در شبکه وزین، حکیمی، اسمیشل^{۵۵} و پیرس^{۵۶}، [۱۷]، برای یافتن مرکز نسبی، الگوریتمی با پیچیدگی زمانی $O(n^2 \cdot \log n)$ ارائه کرده‌اند و همچنین کریو^{۵۷} و حکیمی الگوریتمی با پیچیدگی زمانی $O(n \cdot \log n)$ معرفی نموده‌اند، [۲۲]، که اگر این دو الگوریتم به کار گرفته شوند می‌توان مرکز شبکه وزین را به ترتیب در $O(m \cdot n^2 \cdot \log n)$ و $O(m \cdot n \cdot \log n)$ تکرار پیدا نمود. ضمناً در حالت غیر وزین، کریو و حکیمی الگوریتمی با پیچیدگی زمانی $O(n)$ برای یافتن مرکز نسبی روی یک یال ارائه کرده‌اند که در نهایت منجر به الگوریتمی با پیچیدگی زمانی $O(m \cdot n)$ جهت یافتن مرکز شبکه خواهد شد. توجه داشته باشید در شبکه‌ی G چه رأس - وزین و چه رأس - غیر وزین، مرکز رأسی را با فرض داشتن فاصله‌ها، می‌توان با $O(n^2)$ تکرار پیدا کرد.

۲-۲-۲- مرکز شبکه‌ی رأس - وزین

برای شروع فرض می‌کنیم که رئوس G به صورت $w(v_1) \leq w(v_2) < \dots \leq w(v_n)$ مرتب شده باشند که این مرتب سازی در $O(n \cdot \log n)$ تکرار انجام خواهد شد. [۴۴]

فرض کنیم e یک یال دلخواه از مجموعه E باشد. ما می‌بایست مرکز نسبی $x^*(e)$ را روی یال e پیدا

نماییم. برای هر عدد صحیح i ، $1 \leq i \leq n$ ، $D_e^{(i)}(.)$ تابعی است که به صورت زیر تعریف شده است

$$D_e^i(t) = \max_{1 \leq j \leq i} \{D_e(v_j, t)\} \quad 0 \leq t \leq l(e) \quad (4-2)$$

می‌دانیم که $D_e^{(n)}(t) = D_e(t)$. هر تابع $D_e^{(i)}(.)$ با شیب‌های متعلق به مجموعه‌ی

$\{ \pm w(v_1), \pm w(v_2), \dots, \pm w(v_i) \}$ قطعه قطعه خطی می‌باشد. فرض کنید $L^{(i)} = \langle t_0^{(i)}, t_1^{(i)}, \dots, t_{n_i}^{(i)} \rangle$

لیست مرتب طول نقاطی باشد که $D_e^{(i)}(.)$ در آن نقاط شکست دارد و نیز

پس برای هر $1 \leq j \leq n_i$ مقدار تابع $D_e^{(i)}(t)$ به ازای $t \in [t_{j-1}^{(i)}, t_j^{(i)}]$ با $0 = t_0^{(i)} < t_1^{(i)} < \dots < t_{n_i}^{(i)} = l(e)$

⁵⁵ Schmeichel

⁵⁶ Pierce

⁵⁷ Kariv

رابطه‌ی ۲-۵ داده می شود. (چون شیب خط و یک نقطه از خط را داریم لذا فرمول زیر برای معادله‌ی خط مذکور بدست می آید)

$$D_e^{(i)}(t) = D_e^{(i)}(t_{j-1}^{(i)}) + \frac{D_e^{(i)}(t_j^{(i)}) - D_e^{(i)}(t_{j-1}^{(i)})}{t_j^{(i)} - t_{j-1}^{(i)}}(t - t_{j-1}^{(i)}) \quad (5-2)$$

فرض کنیم که لیست جزئی $L^{(i)}$ را که شامل آن نقاط $t_j^{(i)}$ از لیست می شود که $t_1 \leq t_j^{(i)} \leq t_2$ با $L^{(i)}(t_1, t_2)$ نمایش دهیم. اکنون می خواهیم طریقه‌ی استخراج تابع $D_e^{(i+1)}(.)$ را از تابع $D_e^{(i)}(.)$ و نیز استخراج $L^{(i+1)}$ از $L^{(i)}$ را نشان دهیم. با توجه به رابطه‌ی ۲-۴ داریم

$$D_e^{i+1}(t) = \max\{D_e^i(t), D_e(v_{i+1}, t)\}$$

البته با توجه به ترتیب رؤس، برای هر j ، $1 \leq j \leq i$ داریم $w(v_{i+1}) \geq w(v_i)$ پس قدر مطلق شیب $D_e(v_{i+1}, .)$ بزرگ تر یا مساوی با قدر مطلق شیب قطعات $D_e^{(i)}(.)$ می باشد و یا ممکن است $D_e(v_{i+1}, .)$ و $D_e^{(i)}(.)$ در امتداد یکی از قطعات، بر هم منطبق باشند. فرض کنید $\hat{t}_{i+1}$ نقطه‌ای باشد که $D_e(v_{i+1}, .)$ در آن در بازه‌ی $[0, l(e)]$ به بیشینه مقدارش می رسد. $\hat{t}_{i+1}$ یا 0 است یا $l(e)$ و یا نقطه‌ای که تابع $D_e(v_{i+1}, .)$ در آن شکست می خورد. تابع $\hat{D}_{i+1}$ را به این صورت تعریف می کنیم

$$\hat{D}_{i+1} = D_e(v_{i+1}, \hat{t}_{i+1}) = \max_{t \in [0, l(e)]} D_e(v_{i+1}, t) \quad (6-2)$$

تنها حالات زیر، می تواند اتفاق بیفتد:

الف) اگر $\hat{D}_{i+1} \leq D_e^i(\hat{t}_{i+1})$ ، بوضوح برای همه‌ی t هایی که $t \in [0, l(e)]$ ، $D_e(v_{i+1}, t) \leq D_e^{(i)}(t)$ ، پس در

$$L^{(i+1)} = L^{(i)}, \quad D_e^{i+1}(.) = D_e^i(.)$$

این مورد داریم

ب) اگر $\hat{D}_{i+1} > D_e^i(\hat{t}_{i+1})$ ، دو بازه‌ی بسته‌ی $[0, \hat{t}_{i+1}]$ ، $[\hat{t}_{i+1}, l(e)]$ را در دو حالت جداگانه بررسی می-کنیم.

اگر دو تابع $D_e(v_{i+1}, t)$ و $D_e^{(i)}(\cdot)$ در بازه‌ی $[0, \hat{t}_{i+1}]$ (به همین ترتیب $[\hat{t}_{i+1}, l(e)]$) همدیگر را قطع ننمایند، در این بازه داریم:

$$L^{(i+1)}(0, \hat{t}_{i+1}) = \langle 0, \hat{t}_{i+1} \rangle \text{ و } D_e^{(i+1)}(t) = D_e(v_{i+1}, t)$$

(به همین ترتیب $L^{(i+1)}(\hat{t}_{i+1}, l(e)) = \langle \hat{t}_{i+1}, l(e) \rangle$) اما اگر دو تابع $D_e^{(i)}(\cdot)$ و $D_e(v_{i+1}, t)$ در بازه‌ی $[0, \hat{t}_{i+1}]$ (به همین ترتیب بازه‌ی $[\hat{t}_{i+1}, l(e)]$) همدیگر را در یک نقطه‌ی یکتا قطع نمایند مثل t_{i+1}^- (به-همین ترتیب t_{i+1}^+) و یا اینکه در امتداد یکی از قطعات خطی بر هم منطبق باشند، داریم

$$D_e^{(i+1)}(t) = \begin{cases} D_e^{(i)}(t) & 0 \leq t \leq t_{i+1}^- \\ D_e(v_{i+1}, t) & t_{i+1}^- \leq t \leq \hat{t}_{i+1} \end{cases}$$

$$L^{(i+1)}(0, \hat{t}_{i+1}) = L^{(i)}(0, t_{i+1}^-) \parallel \langle t_{i+1}^-, \hat{t}_{i+1} \rangle$$

(به همین ترتیب

$$D_e^{(i+1)}(t) = \begin{cases} D_e(v_{i+1}, t) & \hat{t}_{i+1} \leq t \leq t_{i+1}^+ \\ D_e^{(i)}(t) & t_{i+1}^+ \leq t \leq l(e) \end{cases}$$

$$L^{(i+1)}(\hat{t}_{i+1}, l(e)) = \langle \hat{t}_{i+1}, t_{i+1}^+ \rangle \parallel L^{(i)}(t_{i+1}^+, l(e))$$

با این فرض که در مواردی که $D_e^{(i)}(\cdot)$ و $D_e(v_{i+1}, \cdot)$ بر هم منطبقند t_{i+1}^- کوچک‌ترین مقدار t در بازه‌ی

$[0, \hat{t}_{i+1}]$ باشد که $D_e(v_{i+1}, t) = D_e^{(i)}(t)$ (به همین ترتیب t_{i+1}^+ ، بزرگ‌ترین مقدار t در بازه

$[\hat{t}_{i+1}, l(e)]$ باشد که $D_e(v_{i+1}, t) = D_e^{(i)}(t)$)

۲-۲-۱- الگوریتم مرکز نسبی در یک شبکه رأس - وزین [۲۲]

$$D(l(e)) \leftarrow 0, D(0) \leftarrow 0, L \leftarrow \langle 0, l(e) \rangle, i \leftarrow 0 - ۱$$

۲- اگر $i = n$ آنگاه به مرحله ۱۰ بروید و در غیر این صورت $i \leftarrow i + 1$

۳- نقطه‌ی $\hat{t}$ را بیابید به گونه‌ای که $D_e(v_i, \hat{t}) = \max_{t \in [0, l(e)]} \{D_e(v_i, t)\}$ (می‌تواند صفر یا $l(e)$ و یا نقطه‌ی شکست $D_e(v_i, .)$ باشد).

۴- در L ، نقطه‌ی t^- را به گونه‌ای پیدا کنید که $t^- = \max\{t_j \in L \mid t_j \leq \hat{t}\}$ و نیز نقطه‌ی t^+ را با فرض $t^+ = \min\{t_j \in L \mid t_j \geq \hat{t}\}$ پیدا کنید. با استفاده از رابطه‌ی ۲-۵ مقدار $D(\hat{t})$ را محاسبه نمایید. اگر $D_e(v_i, \hat{t}) \leq D(\hat{t})$ باشد در واقع نمونه‌ی (الف) رخ داده است و به مرحله ۲ برگردید. (هیچ تغییری در لیست گذشته رخ نمی‌دهد چون $D_e^{(i)}(.) = D_e^{(i+1)}(.)$ و $L^{i+1} = L^i$)

۵- بررسی رفتار نقطه‌ی $\hat{t}$:

اگر $t^- = \hat{t} = t^+$ (یعنی $\hat{t}$ در لیست L موجود باشد) آنگاه $D(\hat{t}) \leftarrow D_e(v_i, \hat{t})$ و به مرحله ۶ بروید. این در واقع حالتی است که $D_e(v_i, \hat{t}) \geq D(\hat{t})$ پس $D(\hat{t})$ را با مقدار جدید $D_e(v_i, \hat{t})$ بروزرسانی کنید و بگذارید $\hat{t}$ در لیست باقی بماند) اما اگر $t^- < \hat{t} < t^+$ ($\hat{t}$ در لیست نباشد) $\hat{t}$ را به لیست L اضافه کرده و $D(\hat{t}) \leftarrow D_e(v_i, \hat{t})$ و $t^- \leftarrow \hat{t}$ و $t^+ \leftarrow \hat{t}$ (چون در حالتی که $D_e(v_i, \hat{t}) \geq D(\hat{t})$ ، چه دو تابع $D_e(v_i, \hat{t}), D_e^{(i)}(t)$ همدیگر را قطع نمایند و چه قطع نمایند $\hat{t}$ نقطه‌ی شکست $D_e(v_i, \hat{t})$ خواهد بود) ۶- بررسی رفتار در بازه‌ی $(0, \hat{t})$:

اگر $t^- = 0$ به مرحله ۷ بروید در غیر این صورت $s^- \leftarrow t^-$ سپس جایگذاری نمایید: $t^- \leftarrow \max\{t_j \in L \mid t_j < s^-\}$ (یکی قبل‌تر از t^- را بررسی کنید چون تکلیف t^- معلوم شده است). اگر $D(t^-) < D_e(v_i, t^-)$ آنگاه t^- را از L حذف کرده و دوباره ۶ را تکرار نمایید. اگر $D(t^-) = D_e(v_i, t^-)$ به مرحله ۸ بروید. (بگذارید t^- در لیست L بماند) اما اگر $D(t^-) > D_e(v_i, t^-)$ نقطه‌ی t_0 از بازه‌ی

(t^-, s^-) را که دو تابع $D(\cdot)$ و $D_e(v_i, \cdot)$ در آن همدیگر را قطع می‌نمایند، به لیست L اضافه کرده و $D(t_0) \leftarrow D_e(v_i, t_0)$ و به مرحله ۸ بروید (بعبارتی تا جایی این مرحله را تکرار می‌کنیم که دو تابع همدیگر را قطع کنند و یا به $t^- = 0$ برسید)

۷- رفتار در نقطه‌ی صفر:

اگر $\hat{t} = 0$ به مرحله ۸ بروید در غیر این صورت $t = 0$ را به لیست L اضافه کنید و $D(0) \leftarrow D_e(v_i, 0)$
۸- بررسی رفتار در بازه‌ی $(\hat{t}, l(e))$:

اگر $t^+ = l(e)$ به مرحله ۹ بروید، در غیر این صورت $t^+ \leftarrow S^+$ و عبارت زیر را جایگزین کنید بدین صورت که $t^+ \leftarrow \min\{t_j \in L \mid t_j > s^+\}$. اگر $D(t^+) < D_e(v_i, t^+)$ آنگاه t^+ را از L حذف نمایید و ۸ را تکرار کنید. اگر $D(t^+) = D_e(v_i, t^+)$ به مرحله ۲ برگردید. اگر $D(t^+) > D_e(v_i, t^+)$ با استفاده از رابطه‌ی ۲-۵ نقطه‌ی t_0 را که در بازه‌ی (S^+, t^+) بوده و $D(\cdot)$ و $D_e(v_i, \cdot)$ در آن نقطه اشتراک دارند پیدا کرده و به L اضافه نموده و $D(t_0) \leftarrow D_e(v_i, t_0)$ و به ۲ بروید.

۹- رفتار در نقطه‌ی $l(e)$:

اگر $\hat{t} = l(e)$ به ۲ بروید و گرنه $l(e)$ را به L اضافه کرده و $D(l(e)) \leftarrow D_e(v_i, l(e))$ به ۲ برگردید.

۱۰- در لیست L ، t^* را بیابید که $D(\cdot)$ را کمینه کند. نقطه‌ی متناظر آن همان مرکز نسبی روی یال e بوده و مقدار $D(t^*)$ ، شعاع نسبی می‌باشد.

۲-۲-۲ پیچیدگی زمانی

می‌دانیم که در هر مرحله انشقاق $D_e^{(i+1)}(\cdot)$ از $D_e^{(i)}(\cdot)$ ، حداکثر ۳ نقطه‌ی $(t_{i+1}^-, \hat{t}_{i+1}, t_{i+1}^+)$ می‌تواند به لیست L اضافه شود و با توجه به مراحل الگوریتم، تعداد تکرارهای عملگرها، حداکثر $O(\log n)$ خواهد بود. [44] برای هر رأس، عملگر اضافه کردن حداکثر ۵ بار و در مجموع $5n$ بار ظاهر خواهد شد. عملگر پیدا کردن یک نقطه هم حداکثر ۴ بار (۲ بار در گام ۴ و یک بار قبل از خروج از گام‌های ۶ و ۸) و به اضافه‌ی

یک تکرار قبل از عملگر حذف انجام می‌شود. پس در مجموع تعداد عملیات‌های ما $O(n)$ خواهد بود که منجر به این می‌شود که مرکز نسبی روی هر یال در زمان $O(n \log n)$ یافت شود.

۲-۲-۳- الگوریتم مرکز روی شبکه‌ی رأس - وزین [۲۲]

- ۱- ابتدا رأس‌های شبکه را بر اساس وزن هایشان، به صورت صعودی مرتب می‌نماییم.
- ۲- برای هر یال از شبکه، الگوریتم ۱-۲-۲-۲ را جهت یافتن مرکز نسبی و شعاع نسبی روی یال مورد نظر اعمال می‌نماییم.
- ۳- کمینه‌ی شعاع‌های نسبی روی کلیه یال‌های شبکه، شعاع شبکه می‌باشد. هر مرکز نسبی‌ای که کمینه‌ی شعاع‌های نسبی در آن رخ دهد، مرکز شبکه می‌باشد.

۲-۲-۴- پیچیدگی زمانی

بدیهی است که الگوریتم فوق، بعد از m تکرار الگوریتم ۱-۲-۲-۲ به جواب می‌رسد پس پیچیدگی زمانی الگوریتم فوق، $O(m.n \log n)$ خواهد بود با این فرض که ماتریس فاصله‌ها موجود باشد.

۲-۲-۳- مرکز شبکه‌ی رأس - غیروزین

در این بخش، الگوریتمی جهت یافتن مرکز شبکه‌ی غیر وزین $G(V, E)$ ارائه می‌کنیم یعنی شبکه‌ای که در آن $w(v) = c$ برای تمامی $v \in V$ ، بدون اینکه کلیت مسئله از دست برود می‌توانیم فرض کنیم $c = 1$. مانند آنچه که در قبل عمل کردیم ابتدا الگوریتمی جهت یافتن مرکز نسبی روی یک یال دلخواه ارائه می‌کنیم. همانند قبل مرکز نسبی $x^*(e)$ روی یال $e = (v_r, v_s)$ ممکن است در نقاط انتهایی یال e یعنی v_r, v_s یا در محل تلاقی دو تابع $D_e(v_i, \cdot)$ و $D_e(v_j, \cdot)$ رخ دهد. البته با توجه به تساوی وزن رئوس، همه‌ی توابع $D_e(v_i, \cdot)$ ، $i = 1, 2, \dots, n$ ، قدر مطلق شیبشان یکسان خواهد بود. پس دو تابع $D_e(v_i, \cdot)$ و $D_e(v_j, \cdot)$ یا برهم منطبقند و یا حداکثر در یک نقطه تلاقی دارند که در آن نقطه دو تابع شیب‌هایشان دارای علامت مخالف می‌باشد.

برای هر رأس $v_i \in V$ تعریف می‌کنیم

$$V_i = \{v \in V | D_e(v, 0) \leq D_e(v_i, 0)\} \quad \bar{V}_i = V - V_i \quad (7-2)$$

اگر $\bar{V}_i \neq \emptyset$ ، رأس $\bar{v}_i$ را به گونه‌ای در نظر می‌گیریم که

$$D_e(\bar{v}_i, l(e)) = \max_{v \in \bar{V}_i} \{D_e(v, l(e))\} \quad (8-2)$$

اگر بیش از یک رأس یافتیم که در رابطه‌ی ۸-۲ صدق کرد ما رأسی که کم‌ترین اندیس را دارد انتخاب می‌کنیم.

با استفاده از رابطه‌ی ۷-۲ و ۸-۲ لم‌های زیر را ثابت می‌کنیم:

لم ۱-۲- اگر t_0 طول نقطه‌ای باشد که دو تابع $D_e(v_i, \cdot)$ و $D_e(v_j, \cdot)$ در آن تلاقی دارند به قسمی که در نقطه‌ی t_0 ، تابع $D_e(v_i, \cdot)$ ، شیب مثبت و تابع $D_e(v_j, \cdot)$ شیب منفی داشته باشد آنگاه برای

$$D_e(v_i, t) > D_e(v_j, t) \quad , \quad t_0 \leq t \leq l(e) \quad \text{و برای} \quad D_e(v_i, t) < D_e(v_j, t) \quad , \quad 0 \leq t \leq t_0$$

نتیجه ۱-۲- تحت شرایط لم ۱-۲، $v_j \in \bar{V}_i, v_i \in V_j$

زیرا اگر $t = 0$ آنگاه

$$D_e(v_i, 0) < D_e(v_j, 0)$$

و بنابراین $v_i \in V_j, v_j \in \bar{V}_i$

لم ۲-۲- مرکز نسبی G روی یال e ، ممکن است در $t = 0$ یا $t = l(e)$ رخ دهد و یا اینکه در نقطه‌ی

t_i قسمی که t_i طول نقطه‌ی اشتراک دو تابع $D_e(v_i, \cdot)$ و $D_e(\bar{v}_i, \cdot)$ باشد.

اثبات: این حقیقت که مرکز نسبی می‌تواند در نقاط 0 یا $l(e)$ رخ دهد بدیهی است منتها ما فرض می‌-

کنیم که مرکز نسبی روی یال e در نقطه‌ی t^* اتفاق بیفتد که $t^* \neq 0, l(e)$ پس همانند گذشته می‌-

بایست t^* طول نقطه‌ای باشد که دو تابع $D_e(v_i, \cdot)$ و $D_e(v_j, \cdot)$ در آن همدیگر را قطع می‌کنند و دارای

شیبی با علامت متضاد باشند. بدون اینکه کلیت مسئله از بین برود فرض می‌کنیم $D_e(v_i, \cdot)$ در نقطه‌ی

t^* ، شیب مثبت باشد. لذا با توجه به نتیجه ۲-۱ می‌بایست $v_j \in \bar{V}_i$ پس $\bar{V}_i \neq \emptyset$ و رأس $\bar{v}_i$ تعریف شده است. با توجه به تعریفی که برای مرکز نسبی داشتیم، $D_e(\bar{v}_i, t^*) \leq D_e(v_j, t^*)$ ، اگر حالت تساوی نامساوی فوق رخ دهد که لم برقرار می‌شود پس فرض می‌کنیم که $D_e(\bar{v}_i, t^*) < D_e(v_j, t^*)$ در نتیجه اگر تابع $(\bar{v}_i, \cdot)$ در t^* شیب مثبت داشته باشد آنگاه از $D_e(\bar{v}_i, t^*) < D_e(v_i, t^*)$ نتیجه می‌گیریم که $D_e(\bar{v}_i, 0) < D_e(v_i, 0)$ (چون دو تابع $D_e(\bar{v}_i, t^*)$ و $D_e(v_i, t^*)$ حداکثر در یک نقطه می‌توانند اشتراک داشته باشند) که با فرض انتخاب $\bar{v}_i$ در رابطه‌ی ۲-۷ و ۲-۸ تناقض دارد اما اگر در t^* تابع $(\bar{v}_i, \cdot)$ ، شیب منفی بود آنگاه از $D_e(\bar{v}_i, t^*) < D_e(v_j, t^*)$ نتیجه می‌گیریم $\max_{v \in \bar{V}_i} \{D_e(v, l(e))\} = D_e(\bar{v}_i, l(e)) < D_e(v_j, l(e))$. این نیز با انتخاب $\bar{v}_i$ در رابطه‌ی ۲-۸ تناقض دارد. لذا فرض $D_e(\bar{v}_i, t^*) < D_e(v_j, t^*)$ درست نبوده و $D_e(v_i, t^*) = D_e(\bar{v}_i, t^*) = D_e(v_j, t^*)$ پس لم برقرار می‌باشد.

لم ۲-۳- اگر دو تابع $D_e(v_i, \cdot)$ و $D_e(\bar{v}_i, \cdot)$ همدیگر را در نقطه‌ای به طول t_i قطع نمایند آنگاه داریم

$$D_e(v_i, t_i) = \max_{v \in V} \{D_e(v, t_i)\}$$

یا بعبارت کوتاه‌تر

$$D_e(t_i) = D_e(v_i, t_i)$$

اثبات: فرض کنیم لم درست نباشد و رأس مثل v' وجود داشته باشد که $D_e(v_i, t_i) < D_e(v', t_i)$ از

آنجایی که در نقطه‌ی t_i تابع $D_e(v_i, t_i)$ ، شیبی مثبت دارد، پس خواهیم داشت

$$D_e(v_i, 0) < D_e(v', 0)$$

یعنی $v' \in \bar{V}_i$ ، بعبارتی دیگر نامساوی $D_e(v_i, t_i) < D_e(v', t_i)$ نتیجه می‌دهد

$$D_e(\bar{v}_i, t_i) < D_e(v', t_i)$$

و از آنجایی که در نقطه‌ی t_i شیب تابع $D_e(\bar{v}_i, \cdot)$ منفی بود پس خواهیم داشت
 $D_e(\bar{v}_i, l(e)) < D_e(v', l(e))$ که با تعریف $\bar{v}_i$ در رابطه‌ی ۲-۸ تناقض دارد. مستقیماً از لم ۲-۲ و ۳-۲،
 نتیجه‌گیری زیر را خواهیم داشت:

نتیجه ۲-۲- اگر $t_0 = 0, t_{n+1} = l(e)$ و همچنین برای $1 \leq i \leq n$ ، t_i طول نقطه‌ی تقاطع $D_e(v_i, \cdot)$ و $D_e(\bar{v}_i, \cdot)$ باشد، فرض کنید داشته باشیم

$$\begin{aligned} D_0 &= \max_{v \in V} \{D_e(v, 0)\} \\ D_{n+1} &= \max_{v \in V} \{D_e(v, l(e))\} \\ D_i &= D_e(v_i, t_i) ; 1 \leq i \leq n \end{aligned}$$

ضمناً j را اندیسی بگیرد که

$$D_j = \min_{0 \leq i \leq n+1} \{D_i\}$$

به قسمی که D_i به ازای هر اندیس i تعریف شده باشد.

آنگاه D_j شعاع نسبی روی یال e خواهد بود و نقطه‌ی متناظرش t_i همان مرکز نسبی خواهد بود.
لم ۲-۴- اگر v_i و v_j دو رأس باشند که $D_e(v_i, 0) = D_e(v_j, 0)$ و $D_e(v_i, l(e)) \geq D_e(v_j, l(e))$ با
 توجه به نتیجه‌ی ۲-۲، نقطه‌ی t_j جایی که دو تابع $D_e(v_j, \cdot)$ و $D_e(\bar{v}_j, \cdot)$ اشتراک دارند، در محاسبه‌ی
 مرکز نسبی نادیده گرفته می‌شود.

اثبات: با توجه به شرایط لم داریم

برای هر t که $0 \leq t \leq l(e)$ ، $D_e(v_i, t) \geq D_e(v_j, t)$ فرض کنید t_j طول نقطه‌ی مرکز نسبی روی یال
 e باشد پس با توجه به تعریف مرکز نسبی نتیجه می‌گیریم:

$$\max_{v \in V} \{D_e(v, t_j)\} = D_e(v_j, t_j) \geq D_e(v_i, t_j)$$

پس خواهیم داشت

$$D_e(v_i, t_j) = D_e(v_j, t_j)$$

بعبارت دیگر با استفاده از تعریف 8-2 نتیجه می‌گیریم که $\bar{v}_i = \bar{v}_j$ (از آنجایی که

$D_e(v_i, 0) = D_e(v_j, 0)$ خواهیم داشت $\bar{V}_i = \bar{V}_j$ و $D_e(\bar{v}_i, t_j) = D_e(\bar{v}_j, t_j)$ نقطه‌ی تقاطع

$D_e(\bar{v}_i, \cdot)$ و $D_e(v_j, \cdot)$ نقطه تقاطع $D_e(\bar{v}_j, \cdot)$ و $D_e(v_j, \cdot)$ هم خواهد بود، چون

$$D_e(\bar{v}_i, t_j) = D_e(\bar{v}_j, t_j) = D_e(v_j, t_j) = D_e(v_i, t_j)$$

بر طبق نتایج بالا الگوریتمی ارائه می‌شود که در آن بعنوان یک پیش‌الگوریتم یک لیست مثل $L(v)$ را

برای هر $v \in V$ می‌سازیم که در آن تمامی رئوس شبکه به صورت نزولی بر اساس فاصله تا رأس v مرتب

شده باشند با این فرض که ماتریس فاصله‌ها موجود باشد. ساختن چنین لیستی نیاز به $O(n \log n)$

تکرار دارد. پس نیاز به $O(n^2 \log n)$ تکرار جهت ساخت این لیست برای تمامی رئوس شبکه خواهد بود.

یال دلخواه $e = (v_r, v_s)$ را داریم. i را i امین رأس در لیست $L(v_r)$ بگیریم. پس داریم

$$D_e(v_1, 0) \geq D_e(v_2, 0) \geq \dots \geq D_e(v_{n-1}, 0) > D_e(v_n, 0) = 0$$

که v_n همان خود v_r می‌باشد. الگوریتم در چندین مرحله به نتیجه می‌رسد که در i امین مرحله، ما

توابع $D_e(v_i, \cdot)$ و $D_e(\bar{v}_i, \cdot)$ و نقطه‌ی اشتراکشان را اگر نیاز باشد می‌آزماییم. توجه کنید که اگر برای j

و k ای دلخواه $(1 < j \leq k < n)$ داشته باشیم

$$D_e(v_{j-1}, 0) \geq D_e(v_j, 0) = \dots = D_e(v_k, 0) > D_e(v_{k+1}, 0)$$

آنگاه با توجه به رابطه‌ی ۲-۸ و ۲-۷ داریم

$$\bar{v}_j = \bar{v}_{j+1} = \dots = \bar{v}_k$$

پس اگر v_m ، $j \leq m \leq k$ رأسی با کمینه اندیس باشد که $\{D_e(v_i, l(e))\}_{j \leq i \leq k}$

آنگاه از لم ۲-۴ نتیجه می‌شود که از بین مجموعه رئوس $\{v_j, v_{j+1}, \dots, v_k\}$ تنها نیاز است رأس v_m در

نظر گرفته شود. (یعنی فقط نقطه‌ی t_m طول نقطه‌ای که دو تابع $D_e(v_m, \cdot)$ و $D_e(\bar{v}_m, \cdot)$ همدیگر را

قطع می کنند، می بایست بعنوان یکی از کاندیدهای مرکز نسبی، در نظر گرفته شود) همچنین با توجه به

ترتیب $L(v_r)$ برای رأس $\bar{v}_{k+1}$ خواهیم داشت

$$D_e(\bar{v}_{k+1}, l(e)) = \max \{D_e(\bar{v}_j, l(e)), D_e(v_m, l(e))\}.$$

همچنین اگر k عدد صحیحی باشد که

$$D_e(v_1, 0) = \dots = D_e(v_k, 0) \geq D_e(v_{k+1}, 0); \quad 1 \leq k \leq n.$$

و رأس v_m را مثل قبل به گونه ای تعریف کنیم که

$$D_e(v_m, l(e)) = \max_{1 \leq i \leq k} \{D_e(v_i, l(e))\}.$$

آنگاه با توجه به ترتیب خاص لیست $L(v_r)$ داریم $\bar{v}_{k+1} = v_m$

در الگوریتم زیر، متغیر t^* و r به ترتیب نشان دهنده مرکز نسبی و شعاع نسبی روی یال $e = (v_r, v_s)$

می باشند. رئوس $\bar{v}_i$ و v_i هم در الگوریتم به ترتیب با v^* و $\bar{v}^*$ نشان داده می شوند.

۲-۳-۱- الگوریتم مرکز نسبی روی یک شبکه ی رأس - غیر وزین [۲۲]

۱- رفتار در نقطه ی $t = 0, t = l(e)$

v^* را اولین رأس از لیست $L(v_r)$ و $\hat{v}$ را اولین رأس از لیست $L(v_s)$ بگیرید. اگر

$D_e(v^*, 0) \leq D_e(\hat{v}, l(e))$ آنگاه $D_e(v^*, 0) \leftarrow t^*, r \leftarrow D_e(v^*, 0)$ در غیر این صورت $t^* \leftarrow l(e)$ و

$r \leftarrow D_e(\hat{v}, l(e))$ اگر $\hat{v} = v^*$ آنگاه متوقف شوید. (می خواهیم ببینیم نقطه ی $t = 0$ و $t = l(e)$ کدامیک

کاندیدای بهتری هستند)

$$v_m \leftarrow v^*, \quad i \leftarrow 1 - 2$$

۳- $i \leftarrow i + 1$ و v^* را i امین رأس در $L(v_r)$ بگیرید. اگر $D_e(v^*, 0) \neq D_e(v_m, 0)$ آنگاه به گام ۴ بروید

وگرنه، اگر $D_e(v^*, l(e)) > D_e(v_m, l(e))$ آنگاه $v_m \leftarrow v^*$ و دوباره گام ۳ را تکرار کنید.

۴- $\bar{v}^* \leftarrow v_m$ اگر $i = n$ آنگاه به گام ۸ بروید و در غیر این صورت $v_m \leftarrow v^*$ و به مرحله ی ۵ بروید.

۵- تمامی رئوسی که $D_e(v,0) = D_e(v_i,0)$ و نقطه‌ی متناظرشان، v_m ، را بیابید. آنگاه $i \leftarrow i+1$ و v^* i امین رأس لیست $L(v_r)$ بگیرید. اگر $D_e(v^*,0) \neq D_e(v_m,0)$ آنگاه به گام ۶ بروید، در غیر این صورت اگر $D_e(v^*,l(e)) > D_e(v_m,l(e))$ آنگاه $v_m \leftarrow v^*$ و گام ۵ را تکرار کنید.

۶- رفتار در نقطه‌ی t_m :

اگر تابع‌های $D_e(v_m,.)$ و $D_e(\bar{v}^*,.)$ همدیگر را قطع نکردند و یا در راستای یک قطعه خط بر هم منطبق بودند آنگاه به گام ۷ بروید، در غیر این صورت t_m را طول نقطه‌ی اشتراک $D_e(v_m,.)$ و $D_e(\bar{v}^*,.)$ بگیرید. اگر $D_e(v_m,t_m) < r$ آنگاه $t_m \leftarrow t^*$ و $r \leftarrow D_e(v_m,t_m)$

۷- اگر $D_e(\bar{v}^*,l(e)) < D_e(v_m,l(e))$ آنگاه $\bar{v}^* \leftarrow v_m$ اگر $i = n$ به گام ۸ بروید و گرنه $v_m \leftarrow v^*$ و به گام ۵ بروید.

۸- رفتار در رأس v_n :

اگر تابع‌های $D_e(v^*,.)$ و $D_e(\bar{v}^*,.)$ همدیگر را قطع نکردند یا در راستای یک قطعه خط منطبق بودند متوقف شوید. در غیر این صورت t_m را طول نقطه‌ای بگیرید که دو تابع $D_e(v^*,.)$ و $D_e(\bar{v}^*,.)$ همدیگر را قطع می‌کنند. اگر $D_e(v^*,t_m) \geq r$ آنگاه دست نگه دارید و گرنه $t_m \leftarrow t^*$ و $r \leftarrow D_e(v^*,t_m)$ و توقف کنید.

۲-۲-۳-۱ پیچیدگی زمانی:

با فرض اینکه ماتریس فاصله و لیست مرتب $L(v_r)$ و $L(v_s)$ موجود باشد، پیچیدگی زمانی الگوریتم فوق $O(n)$ خواهد بود.

۲-۳-۲-۲- الگوریتم مرکز روی شبکه رأس - غیر وزین [۲۲]

۱- برای هر $v \in V$ ، لیست $L(v)$ را بسازید که بر اساس فاصله رئوس از رأس v و به صورت نزولی مرتب شده باشد.

۲- الگوریتم ۲-۳-۲-۱ را روی هر کدام از یال‌ها، جهت یافتن مرکز نسبی و شعاع نسبی بکار ببرید.

۳- کمینه مقدار شعاع نسبی‌ها روی یال‌های شبکه همان شعاع شبکه خواهد بود و نقطه‌ی متناظرش مرکز شبکه خواهد شد.

۲-۳-۲-۱- پیچیدگی زمانی

گام ۱ الگوریتم ۲-۳-۲-۲، $O(n^2 \cdot \log n)$ عملیات نیاز دارد در حالیکه گام ۲، $O(m \cdot n)$ عملیات می‌خواهد. پس با فرض داشتن ماتریس فاصله‌ها، پیچیدگی زمانی کل الگوریتم $O(m \cdot n + n^2 \cdot \log n)$ خواهد بود.

۳-۲- مکان‌یابی مرکز روی درخت

۲-۳-۱- مکان‌یابی مرکز روی درخت غیر وزین

فرض کنید درخت $T(V, E)$ غیروزی با n رأس باشد، هندلر در سال ۱۹۷۳ الگوریتمی با پیچیدگی زمانی خطی برای یافتن مرکز درخت $T(V, E)$ ارائه نموده است که در ادامه به آن می‌پردازیم. [۲۰]

۲-۳-۱-۱- الگوریتم مکان‌یابی مرکز روی درخت غیر وزین

گام ۱- یک نقطه‌ی دلخواه مثل x را روی درخت T بیابید و سپس دورترین نقطه از آن را یافته و t_1 بنامید.

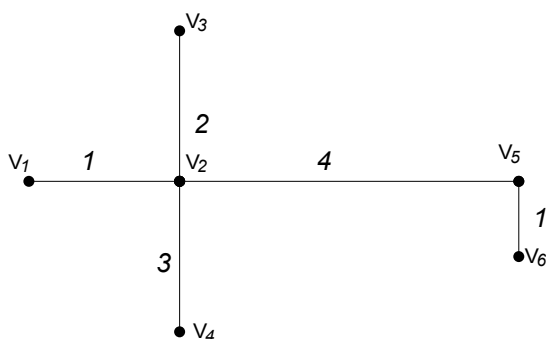
گام ۲- دورترین نقطه از t_1 را پیدا کرده و t_2 بنامید.

گام ۳- نقطه‌ی میانی مسیری که t_1 را به t_2 متصل می‌نماید همان، مرکز درخت می‌باشد.

۲-۱-۳-۲- پیچیدگی زمانی

مهم‌ترین محاسبات شامل یافتن دورترین نقطه از هر کدام از رئوس x و t_1 می‌باشد. لذا پیچیدگی زمانی الگوریتم فوق $O(n)$ خواهد بود در حالیکه n تعداد رئوس درخت باشد.

۲-۱-۳-۳- مثال

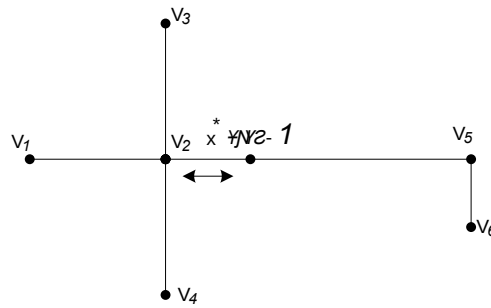


گام ۱- نقطه‌ی V_1 را انتخاب می‌کنیم.

دورترین نقطه از V_1 ، V_6 می‌باشد لذا $t_1 = V_6$.

گام ۲- حال دورترین نقطه از V_6 را پیدا می‌کنیم که V_4 می‌باشد پس $t_2 = V_4$.

نقطه‌ی میانی مسیر متصل کننده t_1 و t_2 را می‌یابیم. چون طول مسیر ۸ می‌باشد، نقطه میانی در فاصله‌ی ۴ از V_4 و V_6 واقع گردیده است که همان مرکز درخت T می‌باشد.



۲-۳-۲- مکان یابی ۲- مرکز روی درخت غیر وزین

فرض کنید درخت $T(V, E)$ غیروزی با n رأس باشد، همچنین هندلر در سال ۱۹۷۳ الگوریتمی با پیچیدگی زمانی خطی برای یافتن ۲- مرکز درخت $T(V, E)$ ارائه نموده است که در ذیل آمده است. [۲۰]

۲-۳-۲-۱- الگوریتم مکان یابی ۲- مرکز روی درخت غیر وزین

گام ۱- با استفاده از الگوریتم ۲-۳-۱-۱، مرکز درخت T را بدست می آوریم.

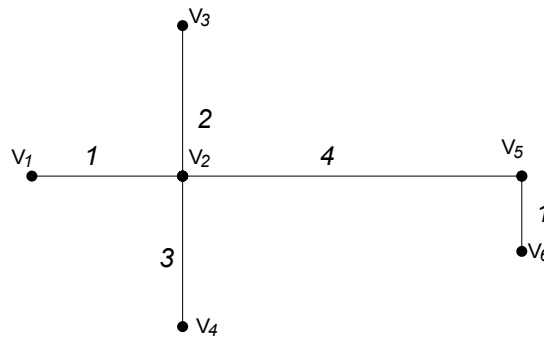
گام ۲- از درخت T ، هر یالی که شامل مرکز بوده و روی مسیری که t_1 را به t_2 متصل می نماید واقع گردیده باشد را حذف می نماییم. (همان تعریف الگوریتم ۲-۳-۱-۱ را دارند) بدین ترتیب دو زیر درخت بدست می آید. (مسیری که t_1 را به t_2 متصل می نماید یکتاست چون در غیر این صورت، دور تشکیل خواهد شد و این با درخت بودن T در تناقض خواهد بود.)

گام ۳- مرکز هر کدام از زیر درخت های حاصله را با الگوریتم ۲-۳-۱-۱ میابیم، اجتماع این دو، همان ۲- مرکز درخت T خواهد بود.

۲-۲-۳-۲- پیچیدگی زمانی

از آنجایی که این الگوریتم، الگوریتم ۲-۳-۱-۱ را سه بار تکرار نموده است پیچیدگی زمانی آن، همان $O(n)$ خواهد بود.

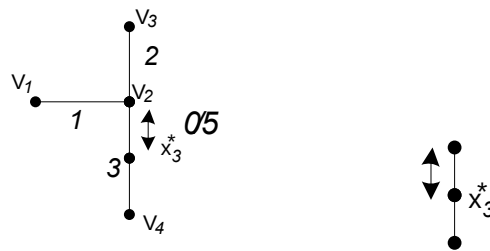
۲-۲-۳-۳- مثال



گام ۱- ابتدا با استفاده از الگوریتم ۲-۳-۱-۱، مرکز درخت T را می‌یابیم.

گام ۲- در مسیر متصل کننده $t_1 t_2$ کلیه ی کمان‌هایی که مرکز درخت T روی آن واقع گردیده است را

از درخت T حذف می‌کنیم که درخت را به دو زیردرخت افراز می‌کند به صورت زیر



گام ۳- حال مرکز هر کدام از دو زیردرخت T را می‌یابیم. X_2^* و X_3^* همان ۲- مرکز درخت T می‌باشند.

۲-۳-۳- مکان یابی مرکز روی درخت وزین

مکان یابی مرکز روی درخت شامل دو گروه کلی مکان یابی مطلق و مکان یابی رأسی خواهد بود که در زیر به آنها پرداخته شده است. [۲۲]

۲-۳-۳-۱- روش مرکز ثقل^{۵۸} برای یافتن مرکز روی درخت وزین

روشی که در این قسمت مطرح خواهد شد هم در مورد مرکز مطلق و هم در مورد مرکز رأسی درخت کاربرد دارد. فرض کنیم T یک درخت رأسی وزین باشد و v هم رأسی از درخت با درجه d_v باشد. $T - v$ گراف حاصل از حذف v و d_v یال برخورد کننده با آن از درخت T باشد. آنگاه $T - v$ شامل d_v زیر درخت همبند خواهد شد که آنها را با $T_{v,1}, T_{v,2}, \dots, T_{v,d_v}$ نمایش می دهیم. ضمناً $T_{v,i}^+$ همان زیر درخت $T_{v,i}$ است که شامل رأس v و یال متصل کننده اش به $T_{v,i}$ می باشد.

لم ۲-۵- $v \in V$ را یک رأس ثابت بگیرید و $\tilde{v}$ را رأسی بگیرید که

$$w(\tilde{v}).d(\tilde{v}, v) = \max_{v' \in V} \{w(v').d(v', v)\}$$

و نیز $T_{v,l}$ هم زیر درخت $T - v$ باشد که شامل $\tilde{v}$ است. در نتیجه مرکز درخت T بر $T_{v,l}^+$ واقع خواهد گشت.

اثبات: به برهان خلف، فرض کنیم که مرکز درخت که با x^* نمایش داده می شود بر $T_{v,l}^+$ واقع نباشد،

پس $d(\tilde{v}, x^*) > d(\tilde{v}, v)$ و از آنجایی که r_1 شعاع T می باشد، داریم :

$$r_1 \geq w(\tilde{v}).d(\tilde{v}, x^*) > w(\tilde{v}).d(\tilde{v}, v) = \max_{v' \in V} \{w(v').d(v', v)\}$$

پس v انتخاب بهتری نسبت به x^* بعنوان مرکز درخت T خواهد بود که با مفروضات ما در تناقض است.

⁵⁸The centroid

نتیجه ۲-۳- لم بالا مستقیماً به این نتیجه منجر می شود که اگر $\hat{v}$ و $\tilde{v}$ دو رأسی باشند که

$$k \neq l \quad \hat{v} \in T_{v,k} \quad \text{و} \quad \tilde{v} \in T_{v,l}$$

$$\max_{v' \in V} \{w(v').d(v', v)\} = w(\tilde{v}).d(\tilde{v}, v) = w(\hat{v}).d(\hat{v}, v)$$

لذا v ، مرکز درخت خواهد شد. چون طبق لم بالا، مرکز درخت T هم در $T_{v,l}^+$ واقع است و هم

که $T_{v,k}^+$

$$v = T_{v,k}^+ \cap T_{v,l}^+$$

۲-۳-۳- الگوریتم پیشنهادی

الگوریتم پیشنهاد شده به این صورت است که T_0 را همان درخت ابتدایی T بگیرید. رأس v_0 از T_0 را انتخاب و زیر درخت T_{v_0, l_0}^+ که در آن مرکز واقع شده است را طبق لم ۲-۵ بیابید. فرض کنید $T_1 = T_0 \cap T_{v_0, l_0}^+$ مجدداً رأس v_1 از T_1 را انتخاب و زیر درخت T_{v_1, l_1}^+ که شامل مرکز درخت T می باشد را بیابید. مسلماً مرکز می بایست بر زیر درخت $T_2 = T_1 \cap T_{v_1, l_1}^+$ واقع گردد. آنقدر روند بالا را تکرار می-نماییم که به زیر درخت T_k می رسیم که فقط شامل یک یال می باشد مرکز نسبی روی این یال همان مرکز مطلق درخت و یک یا هر دو رأس ابتدایی و انتهایی یال هم همان، مرکز رأسی درخت T خواهد بود.

۲-۳-۳- پیچیدگی زمانی

در هر مرحله، یافتن زیر درخت های T_{v_i, l_i}^+ و T_{i+1} ، $O(n)$ عملیات نیاز دارد. (چون دورترین رأس از رأس انتخابی v_i را می یابیم) پس پیچیدگی زمانی کل الگوریتم برای مرکز مطلق درخت $O(n.k + n.\log n)$ بوده و برای مرکز رأسی درخت $O(n.k)$ می باشد که k تعداد تکرارها و $n.\log n$ در قدم آخر، هزینه ی یافتن مرکز نسبی برای حالت مرکز مطلق می باشد. مقدار k به این بستگی دارد که ما در هر مرحله، چه رأسی را انتخاب می نماییم حال برای آن که انتخاب خوبی داشته باشیم، مرکز ثقل درخت را معرفی می نماییم.

۲-۳-۴- مرکز ثقل درخت

فرض کنید v یک رأس از درخت T و $T_{v,1}, T_{v,2}, \dots, T_{v,d_v}$ هم زیر درخت‌های همبند حاصل از حذف v از درخت T باشد اگر تعداد رئوس درخت T را با $|T|$ نمایش دهیم و $N(v)$ را به صورت زیر تعریف نماییم

$$N(v) = \max_{1 \leq i \leq d_v} \{|T_{v,i}|\}$$

مرکز ثقل یک درخت رأسی مانند v_c خواهد بود که $N(v)$ کم‌ترین مقدار خودش را بگیرد، یعنی

$$N(v_c) = \min_{v \in V} \{N(v)\}$$

از آنجایی که $N(v_c) \leq \lfloor n/2 \rfloor$ یعنی تعداد رئوس در هر کدام از زیر درخت‌های

$T_{v_c,1}^+, T_{v_c,2}^+, \dots, T_{v_c,d_{v_c}}^+$ از $\lfloor n/2 \rfloor + 1$ نمی‌تواند بزرگ‌تر باشد، پس در الگوریتم قبل اگر در هر مرحله

مرکز ثقل زیر درخت T_i را انتخاب نماییم و در نتیجه $|T_{i+1}| \leq \lfloor |T_i|/2 \rfloor + 1$ ، تعداد تکرارها حداکثر

$O(\log n)$ خواهد بود، زیرا در نهایت ۲ رأس برای ما باقی می‌ماند پس داریم $\frac{n}{2^k} = 2$ یا $n = 2^{k+1}$

بنابراین $(k+1)\log 2 = \log n$ یا $k+1 = \frac{\log n}{\log 2}$ و در نتیجه $k = O(\log n)$.

لذا برای بدست آوردن الگوریتمی با پیچیدگی زمانی $O(n \log n)$ برای مرکز درخت، می‌بایست الگوریتمی

با پیچیدگی زمانی $O(n)$ برای یافتن مرکز ثقل درخت داشته باشیم. با نسخه‌ای از الگوریتم گلدمن^{۵۹}

[۱۴]، بر اساس این خاصیت که $N(v_c) \leq \lfloor n/2 \rfloor$ شرط لازم و کافی برای مرکز ثقل بودن رأس v_c می-

باشد ادامه می‌دهیم. گلدمن به هر کدام از رئوس درخت T ، متغیر $n(v)$ را نسبت داده که اگر v یک برگ

از درخت T' باشد $T' - v$ در یکی از زیر درخت‌های همبند $T - v$ واقع می‌شود که $n(v)$ تعداد رئوس این

زیردرخت را می‌دهد. همچنین از کپی درخت اصلی T ، تحت عنوان درخت کمکی T' استفاده کرده است.

^{۵۹} Goldman

۲-۳-۳-۵- الگوریتم مرکز ثقل درخت

۱- در ابتدا $T' \leftarrow T$ و برای هر رأس $v \in T'$ قرار دهید، $n(v) \leftarrow n - 1$

۲- اگر درخت کمکی فقط شامل یک رأس مثل v_0 باشد دست نگه دارید، v_0 همان مرکز ثقل درخت T می باشد.

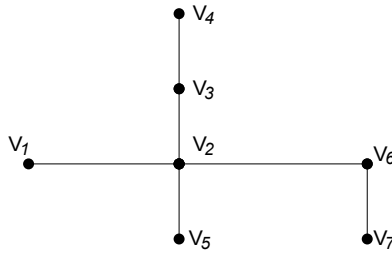
۳- اما اگر v یک برگ از درخت T' باشد و $n(v) \leq \left\lceil \frac{n}{2} \right\rceil$ دست نگه دارید، v همان مرکز ثقل

درخت T خواهد بود. در غیر این صورت برای u رأس مجاور v در درخت T' قرار دهید:

$n(u) \leftarrow n(u) - (n - n(v))$ رأس v را حذف کرده و یال (u, v) را نیز از T' برداشته و

$T' \leftarrow T' - \{v\}$ و مجدداً به ۲ بروید.

۲-۳-۳-۶- مثال



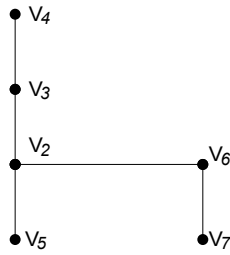
تکرار اول

$$n(V_1) = \dots = n(V_7) = 6$$

اگر برگ منتخب V_1 و V_2 رأس مجاورش باشد آنگاه

$$n(V_2) = n(V_2) - (n - n(V_1)) = 6 - (7 - 6) = 5$$

$$n(V_2) > \left\lceil \frac{n}{2} \right\rceil = \left\lceil \frac{7}{2} \right\rceil = 3$$

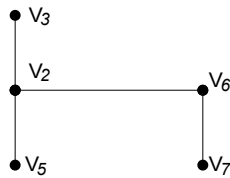


تکرار دوم

اگر برگ منتخب V_4 و V_3 رأس مجاورش باشد آنگاه

$$n(V_3) = n(V_3) - (n - n(V_4)) = 6 - (7 - 6) = 5$$

$$n(V_3) > \left\lfloor \frac{n}{2} \right\rfloor = \left\lfloor \frac{7}{2} \right\rfloor = 3$$

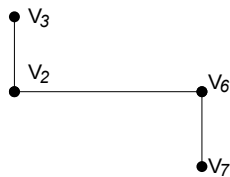


تکرار سوم

اگر V_5 رأس منتخب و V_2 رأس مجاورش باشد آنگاه

$$n(V_2) = n(V_2) - (n - n(V_5)) = 5 - (7 - 6) = 4$$

$$n(V_2) > \left\lfloor \frac{n}{2} \right\rfloor = \left\lfloor \frac{7}{2} \right\rfloor = 3$$



تکرار چهارم

اگر V_3 برگ منتخب و V_2 رأس مجاورش باشد آنگاه

$$n(V_2) = n(V_2) - (n - n(V_3)) = 4 - (7 - 5) = 2$$

و چون $n(V_2) \leq \left\lfloor \frac{n}{2} \right\rfloor = \left\lfloor \frac{7}{2} \right\rfloor = 3$ لذا V_2 مرکز ثقل ما خواهد بود.

با استفاده از الگوریتم ۲-۳-۳-۵ می‌توانیم الگوریتم مرکز درخت را فرمولبندی نماییم. در الگوریتم زیر

متغیرهای T' و T'' و T''' در هر مرحله به ترتیب نمایش دهنده‌ی T_i و T_{v_i, l_i}^+ و T_{v_i, l_i} می‌باشند.

۲-۳-۳-۷- الگوریتم مکان‌یابی مرکز روی درخت وزین (روش مرکز ثقل)

$$T' \leftarrow T - 1$$

۲- (برای حالت مرکز مطلق) اگر T' شامل تک یال مثل e باشد با الگوریتم ۲-۲-۱، مرکز نسبی

درخت T ، x^* را روی یال e یافته و متوقف می‌شویم. x^* مرکز T نیز خواهد بود. (برای حالت مرکز

رأسی) اگر T' یک یال تک مثل $e = (v_r, v_s)$ داشته آنگاه با فرض $d_r = \max_{v' \in V} \{w(v').d(v', v_r)\}$ و

$d_s = \max_{v' \in V} \{w(v').d(v', v_s)\}$ ، اگر $d_r > d_s$ ، آنگاه v_s ، مرکز رأسی می‌شود، اگر بالعکس بود v_r ، مرکز

رأسی خواهد بود و اگر $d_r = d_s$ باشد هر دو رأس v_s و v_r ، مرکز درخت T خواهند بود و متوقف شوید.

۳- با استفاده از الگوریتم ۲-۳-۳-۵ مرکز ثقل T را بیابید.

۴- فرض کنید $\tilde{v}$ رأسی از T باشد که $w(\tilde{v}).d(\tilde{v}, v_c) = \max_{v' \in V} \{w(v').d(v', v_c)\}$ آنگاه T''' را مولفه-

ای از $T - v_c$ بگیرید که شامل $\tilde{v}$ باشد و T'' را زیر درختی بگیرید که شامل T''' و رأس v_c و یال

متصل کننده v_c به T''' باشد.

۵- اگر رأسی مثل $\bar{v}$ پیدا شد که $\bar{v} \in T''$ و $w(\bar{v}).d(\bar{v}, v_c) = w(\tilde{v}).d(\tilde{v}, v_c)$ با توجه به نتیجه‌ی ۲-۳،

v_c ، مرکز T خواهد بود و متوقف شوید.

۶- $T' \leftarrow T' \cap T''$ و به ۲ برگردید.

۲-۳-۳-۸- پیچیدگی زمانی

با توجه به مراحل بالا، پیچیدگی زمانی الگوریتم $O(n \log n)$ خواهد شد. چون با توجه به اینکه، شبکه‌ی ما یک درخت است، نیاز به محاسبه‌ی ماتریس فاصله جهت یافتن مرکز نسبی و d_r, d_s در گام ۲ نداریم. مگیدو^{۶۰} [۲۷] برای یافتن مرکز درخت وزین الگوریتمی ارائه نموده است که بر اساس الگوریتم کریو و حکیمی می‌باشد. همانطور که در الگوریتم کریو و حکیمی داشتیم در هر مرحله، مرکز ثقل درخت را می‌یافتیم و با استفاده از آن زیر درخت شامل مرکز درخت را تعیین می‌نمودیم. با روش کریو و حکیمی در هر مرحله، در بدترین حالت حداقل $\frac{n}{2}$ رئوس حذف می‌گردیدند که نهایتاً بعد از $O(\log n)$ تکرار به جواب می‌رسیدیم. الگوریتم نیم‌رود مگیدو هم، از روش کریو و حکیمی استفاده نموده است منتها با این تغییر که در هر تکرار $\frac{1}{8}$ رئوس را حذف می‌کند. اما هزینه‌ی هر تکرار را کاهش می‌دهد. گرچه مثل الگوریتم کریو و حکیمی بعد از $O(\log n)$ تکرار به جواب می‌رسد، چون در هر تکرار $\frac{n}{8}$ رئوس حذف می‌شوند خواهیم داشت:

تکرار اول:
$$n - \frac{n}{8} = \frac{7n}{8} = \frac{7^1 n}{8^1}$$

تکرار:
$$\frac{7n}{8} - \frac{7n}{8 \times 8} = \frac{49n}{64} = \frac{7^2 n}{8^2}$$

دوم

پس داریم $\frac{7^k n}{8^k} = 2$ یا $7^k n = 8^k \cdot 2$

⁶⁰ Meggido

بنابراین $k \log 7 + \log n = k \log 8 + \log 2$ یا $k \log \frac{8}{7} = \log n - \log 2$ و در نتیجه $k = \frac{\log \frac{n}{2}}{\log \frac{8}{7}}$ یا

$$k = O(\log n)$$

فصل سوم

مکان‌یابی p -مرکز

در این فصل خصوصیات p - مرکز را در یک شبکه‌ی کلی بررسی کرده و الگوریتمی با پیچیدگی زمانی $O((m^p - n^{2p-1} / (p-1)!). \log n)$ جهت یافتن p - مرکز یک شبکه رأس- وزین ارائه می‌شود. همچنین برای پیدا کردن p - مرکز یک شبکه رأس- غیروزی، الگوریتمی با پیچیدگی زمانی $O(m^p \cdot \frac{n^{2p-1}}{(p-1)!})$ را بررسی خواهیم کرد.

۳-۱- مکان‌یابی p - مرکز

در این بخش مکان‌یابی p - مرکز هم در یک شبکه‌ی رأس- وزین و شبکه‌ی رأس- غیر وزین و هم در یک درخت رأس- وزین و درخت رأس- غیر وزین بررسی می‌شود. همچنین در مورد مجموعه‌ی احاطه‌گر صحبت خواهیم کرد. مطالب این بخش از مرجع [۲۲] گرفته شده است.

۳-۱-۱- مکان‌یابی p - مرکز در یک شبکه‌ی رأس- وزین و شبکه‌ی رأس- غیر وزین

فرض کنید $X^{*p} = \{x'_1, x'_2, \dots, x'_p\}$ ، p - مرکز شبکه G بوده و r_p شعاع متناظر با آن باشد که p - شعاع^{۶۱} شبکه خوانده می‌شود. رأس v از مجموعه‌ی V زمانی با نقطه‌ی x'_i از p - مرکز X^{*p} پوشش داده می‌شود که

$$w(v).d(v, x'_i) \leq r_p$$

یک رأس ممکن است با بیش از یک نقطه پوشش داده شود. برای هر نقطه‌ی $x'_i \in X^{*p}$ تعریف می‌کنیم

$$V_i = \{v \in V \mid d(v, x'_i) = d(v, X^{*p})\}$$

مجموعه‌های $V_1, V_2, \dots, V_p$ ممکن است مجزا نباشند و یا تهی باشند اما $\bigcup_{i=1}^n V_i = V$. اگر V_i تهی نباشد،

زیرگراف G که توسط V_i تولید می‌شود را G_i می‌نامیم.

⁶¹ P-radius

(G_i گرافی است که شامل تمام رئوس V_i و یال‌های متصل کننده‌ی رئوس V_i در G ، می‌باشد) فرض

کنید x_i مرکز G_i و $r^{(i)}$ شعاع آن باشد پس داریم

$$r^{(i)} = \max_{v \in V_i} \{w(v).d(v, x_i)\} \leq \max_{v \in V_i} \{w(v).d(v, x'_i)\} \leq r_p$$

اگر V_i تهی باشد G_i را با تک رأس دلخواه v_i تعریف می‌کنیم و در این حالت $x_i = v_i$ و $r^{(i)} = 0$

فرض کنید v رأسی از G باشد و همچنین متعلق به مجموعه‌ی V_i باشد، لذا با فرض $X^*_p = \{x_1, \dots, x_p\}$

داریم $w(v).d(v, x_i) \leq r^{(i)} \leq r_p$ لذا

$$\max_{v \in V} \{w(v).d(v, X^*_p)\} \leq r_p$$

پس مجموعه‌ی $X^*_p = \{x_1, \dots, x_p\}$ یک p -مرکز از G بوده و p -شعاع متناظرش با رابطه‌ی ذیل

بدست می‌آید

$$r_p = \max_{1 \leq i \leq p} \{r^{(i)}\}$$

چون اگر $r_p > \max_{1 \leq i \leq p} \{r^{(i)}\}$ آنگاه مجموعه‌ای پیدا کرده‌ایم که جواب بهتری به ما می‌دهد که این با انتخاب

X^*_p تناقض دارد.

ضمناً ما $r^{(i)}$ را برد x_i می‌نامیم. بنابراین بدون اینکه کلیت مسئله از بین برود می‌توانیم آن را به یافتن

مجموعه‌هایی به فرم $X^*_p = \{x_1, \dots, x_p\}$ محدود نماییم که در آن x_i ها، $1 \leq i \leq p$ ، مرکز زیر گراف G_i

بوده و یا اینکه رأسی با برد 0 باشند. ضمناً فرض می‌کنیم این نقاط مجزا باشند.

فرض کنیم x_i نقطه‌ای از مجموعه‌ی X^*_p باشد. اگر برد آن صفر نباشد آنگاه x_i ، مرکز زیر گرافی از G

می‌باشد و همچنین مرکز نسبی روی یالی مثل e_i از G_i خواهد بود. پس x_i ، در نقطه‌ای روی یال e_i ،

محل تقاطع دو تابع $D_{e_i} = (v_{i_1}, \dots)$ و $D_{e_i} = (v_{i_2}, \dots)$ با شیب‌هایی با علامت مخالف، واقع گردیده است و

برد $r^{(i)}$ برابر با مقدار این دو تابع در نقطه‌ی اشتراک خواهد بود. همانطور که در قبل بحث شد روی یال

e_i در مجموع حداکثر $O(n^2)$ نقطه‌ی کاندید خواهیم داشت که اعضای مجموعه‌ی X^*_p ، ممکن است در آن نقاط واقع گردند، که در کل گراف $O(m.n^2)$ کاندید خواهیم داشت. ضمناً زمانی که نقطه‌ی کاندید x_i یافت شود برد آن نیز شناخته خواهد شد.

فرض کنید $X^*_p = \{x_1, \dots, x_p\}$ یک p -مرکز از G باشد که x_i ها، همان نقاط کاندید باشند. به ازای هر $1 \leq i \leq p$ تعریف می‌کنیم

$$V_i = \{v \in V \mid w(v).d(v, x_i) \leq r^{(i)}\}$$

$$\tilde{V} = \bigcup_{i=1}^{p-1} V_i, \quad V - \tilde{V} \subseteq V_p$$

فرض کنید $\tilde{G}$ زیر شبکه‌ی حاصل از شبکه‌ی G باشد که در آن به تمامی رئوس مجموعه‌ی $\tilde{V}$ ، وزن صفر اختصاص داده شده است و سایر رئوس که در مجموعه‌ی $V - \tilde{V}$ واقع‌اند همان وزن اولیه خود را دارا می‌باشند. اگر $\tilde{x}_p$ ، مرکز $\tilde{G}$ باشد آنگاه مجموعه‌ی $X^*_p = \{x_1, x_2, \dots, x_{p-1}, \tilde{x}_p\}$ یک p -مرکز از G خواهد بود. پس بدون اینکه کلیت مسئله از دست برود جستجوی ما به یافتن مجموعه‌هایی به شکل $\tilde{X}^*_p$ ، محدود خواهد شد که در آن p امین نقطه، مرکز زیرگراف حاصل از رئوسی است که با $P-1$ نقطه‌ی قبلی پوشش داده نشده‌اند. الگوریتم زیر p -مرکز را با ساختن کلیه مجموعه‌هایی به شکل $\tilde{X}^*_p$ می‌یابد.

۳-۱-۱-۱- الگوریتم مکان‌یابی p -مرکز در یک شبکه‌ی رأس-وزین و شبکه‌ی رأس-غیروزین

۱- x_i ها و برد متناظرشان $r^{(i)}$ را از مجموعه‌ی $O(m.n^2)$ نقاط کاندید بیابید. (بدین ترتیب که نقاط

تلاقی دو تابع $D_e = (v_i, \cdot)$ و $D_e = (v_j, \cdot)$ با شیب‌هایی ناهمسان در علامت را بیابید)

۲- برای هر مجموعه‌ی ممکن $X^*_{p-1} = \{x_1, \dots, x_{p-1}\}$ از $p-1$ نقطه‌ی کاندید، حلقه‌ی شامل گام‌های ۳-

۴-۵ را انجام داده و به گام ۶ بروید. (حداکثر $\left(m.n^2\right)_{p-1}$ مجموعه خواهیم داشت.)

۳- برای هر $1 \leq i \leq p-1$ مجموعه‌ی $V_i = \{v \in V | w(v).d(v, x_i) \leq r^{(i)}\}$ را محاسبه کنید و $\tilde{V} = \bigcup_{i=1}^{p-1} V_i$

۴- مرکز $\tilde{G}$ را مثل $\tilde{X}_p$ پیدا کنید. ($\tilde{G}$ همان G می‌باشد منتها رئوس مجموعه $\tilde{V}$ در آن وزن صفر دارند.
 $r^{(p)}$ ، شعاع $\tilde{G}$ می‌باشد)

۵- جایگذاری کنید $\tilde{X}_p \leftarrow X_{p-1} U \{\tilde{x}_p\}$ و $r(\tilde{X}_p) \leftarrow \max_{1 \leq i \leq p} \{r^{(i)}\}$

۶- $\tilde{X}_p^*$ را مجموعه‌ای بگیرید که $r(\tilde{X}_p^*) = \min_{\tilde{X}_p} \{r(\tilde{X}_p)\}$ بنابراین $\tilde{X}_p^*$ ، p - مرکز G بوده و
 $r(\tilde{X}_p^*)$ ، p - شعاع خواهد بود.

۳-۱-۱-۲ پیچیدگی زمانی الگوریتم

ابتدا توجه داریم که اگر G ، رأس-وزین باشد گام ۴، $O(m.n.\log n)$ عملیات نیاز خواهد داشت و اگر رأس-غیروزین باشد گام ۴ می‌تواند با $O(m.n)$ عملیات به نتیجه برسد. در ضمن، گام ۳ نیازمند $O(np)$ عملیات و گام ۵، نیازمند $O(p)$ عملیات می‌باشند. پس پیچیدگی زمانی حلقه‌ی شامل گام‌های ۳-۴-۵ به پیچیدگی زمانی گام ۴ بستگی دارد.

حلقه‌ی شامل گام‌های ۳-۴-۵، حداکثر $O\left(\frac{m^{p-1}.n^{2p-2}}{(p-1)!}\right)$ بار تکرار می‌شود. زیرا

$$\frac{(m.n^2)!}{(p-1)!(m.n^2 - p + 1)!} = \frac{(m.n^2 - p + 1)!(m.n^2 - p + 2) \times (m.n^2 - p + 3) \times \dots}{(p-1)!(m.n^2 - p + 1)!} = \frac{O(m^{p-1}.n^{2p-1})}{(p-1)!}$$

پس می‌توانیم نتیجه بگیریم که پیچیدگی زمانی کل در حالت رأس-وزین $O\left(\frac{m^p.n^{2p-1}}{(p-1)!} \log n\right)$ خواهد

بود و در حالت رأس-غیروزین، رابطه‌ی زیر خواهد بود

$$O\left(\frac{m^p.n^{2p-1}}{(p-1)!}\right)$$

به نظر می‌رسد که جهت بهبود پیچیدگی زمانی الگوریتم می‌توان در گام ۴، به جای انتخاب یک مجموعه- X_{p-1} از $p-1$ نقطه از $O(m.n^2)$ نقطه‌ی کاندید، از مجموعه‌ی $E_{p-1} = \{e_1, e_2, \dots, e_{p-1}\}$ که شامل $p-1$ یال مجزا از مجموع m یال شبکه می‌باشد، استفاده نمود. از آنجا که روی هر یال $O(n^2)$ نقطه‌ی کاندید داریم، بنابر قانون شمارش تعداد کل حالت‌ها از رابطه‌ی زیر بدست می‌آید

$$n^2 \times n^2 \times \dots \times n^2 \times \binom{m}{p-1} = \left(n^{2(p-1)} \cdot \binom{m}{p-1} \right) = O\left(\frac{m^{p-1} \cdot n^{2p-2}}{(p-1)!} \right)$$

پس بهبودی حاصل نمی‌شود.

تمامی مراحل و ایده‌های بالا اگر در مورد مسئله‌ی p - مرکز رأسی اعمال شود، الگوریتمی با پیچیدگی زمانی بهتری حاصل نخواهد شد. برای هر زیر مجموعه‌ی V_p از p رأس گراف، ابتدا بیشینه فاصله‌ی وزین رئوس را از مجموعه V_p می‌یابیم که با فرمول زیر داده می‌شود

$$\max_{v \in V} \left[\min_{v' \in V_p} \{w(v).d(v, v')\} \right]$$

که کمینه‌ی این فاصله‌ها روی مجموعه‌ی V_p ، همان p - شعاع رأسی گراف خواهد بود. از آنجایی که محاسبه فرمول بالا، نیاز به $O(np)$ تکرار دارد لذا پیچیدگی زمانی کل الگوریتم عبارت زیر خواهد بود

$$\binom{n}{p} \cdot O(n, p) = O\left(\frac{n^{p+1}}{(p-1)!} \right)$$

۳-۱-۲- مجموعه‌ی احاطه‌گر با شعاع r روی یک درخت رأس - وزین

در این بخش می‌خواهیم به بررسی الگوریتمی برای یافتن مجموعه‌ی احاطه‌گر با شعاع r در یک درخت رأس - وزین (چه حالت رأسی - چه حالت مطلق) بپردازیم. یعنی یک عدد مثبت مثل r و یک درخت رأس - وزین مثل T داده شده است. می‌خواهیم کوچک‌ترین عدد صحیح مثبت p را بیابیم به گونه‌ای

که p - شعاع (چه رأسی - چه مطلق) درخت T ، از r بزرگ تر نشود. الگوریتم ارائه شده همچنین p - مرکز متناظر با آن را بدست می آورد.

مسئله‌ی یافتن مجموعه احاطه‌گر در یک درخت رأسی - غیروزی (پیدا کردن مجموعه‌ی احاطه‌گر در یک درخت با شعاع ۱ در حالیکه تمامی یال‌ها دارای طول ۱ و تمامی رئوس دارای وزن ۱ باشند) در زمان $O(n)$ حل شده است [۷]. همچنین مسئله محدودشده‌ی یافتن مجموعه احاطه‌گر رأسی با شعاع عدد صحیح r در یک درخت رأسی - غیر وزین، که تمامی یال‌ها طولشان ۱ می‌باشد در زمان خطی حل گردیده است. [۳۴]

الگوریتم زیر، هم مجموعه احاطه‌گر رأسی و هم مجموعه احاطه‌گر مطلق را می‌یابد. این الگوریتم با جستجو روی یال‌های T و شروع از برگ‌ها و حرکت به سمت وسط شکل می‌گیرد. درحین جستجو، نقاط مجموعه احاطه‌گر مطلوب را در مدل بهین مکان‌یابی کرده تا زمانی که همه‌ی درخت با p نقطه و شعاع r پوشش داده شود. برای انجام این امر، ابتدا یک کپی T' از درخت اصلی بعنوان درخت کمکی تهیه کرده و به هر رأس v از T' ، متغیر $R(v)$ را نسبت می‌دهیم. (تعریف متغیر $R(v)$ بعداً داده خواهد شد).

اگر v_r یک برگ از درخت کمکی T' باشد و $e = (v_r, v_s)$ یال مجاور به آن باشد آنگاه ما یال e را از v_r به v_s می‌پیماییم. متغیر $R(v_s)$ را بروز کرده و رأس v_r و یال e را از درخت T' حذف می‌کنیم و این روند را آنقدر تکرار می‌نماییم تا تمامی رئوس و یال‌های T' حذف گردند. متغیر $R(v)$ به صورت زیر تعریف میشود. حالت الف: اگر رأس v قبلاً توسط یکی از نقاط مجموعه احاطه‌گر پوشش داده شده باشد، $R(v)$ فاصله‌ی غیر وزین بین رأس v و نزدیک‌ترین نقطه‌ی مجموعه‌ی احاطه‌گر مکان‌یابی شده می‌باشد. بوضوح که در

$$\frac{r}{w(v)} \geq R(v) \geq 0 \quad \text{این حالت می‌بایست}$$

حالت ب: اگر رأس v هنوز پوشش داده نشده است آنگاه اگر $s(v)$ مجموعه رئوس درخت اولیه باشد که هنوز پوشش داده نشده‌اند و v نزدیک‌ترین رأس روی T' به آنها باشد، می‌توان گفت که v تنها رأس از

مجموعه $s(v)$ می باشد که هنوز به T' متعلق است. در حقیقت $s(v)$ ، شامل رأس v و رئوسی که قبلاً از T' حذف شده اند و می بایست با رأسی که v را پوشش می دهد، پوشش داده شوند، می باشد. در این حالت تعریف می کنیم

$$-R(v) = \min_{v' \in s(v)} \left\{ \frac{r}{w(v')} - d(v', v) \right\}$$

$-R(v)$ - بیشینه فاصله ی غیر وزین از رأس v می باشد که هر نقطه از درخت T' در طول آن، تمامی رئوس $s(v)$ را پوشش خواهد داد. خصوصاً که در این حالت

$$0 < -R(v) \leq \frac{r}{w(v)}$$

زیرا می دانیم که فاصله ی رأس v'' در مجموعه ی $s(v)$ تا نقطه ی جدید مکان یابی شده برابر با مجموع فاصله ی v'' تا رأس v و فاصله ی رأس v تا نقطه ی جدید مکان یابی شده می باشد. لذا اگر $-R(v)$ ، به ازای رأس v' بدست آید داریم:

$$d(v'', v) - R(v) = d(v'', v) + \frac{r}{w(v')} - d(v', v) \leq d(v'', v) + \frac{r}{w(v'')} - d(v'', v) \leq \frac{r}{w(v'')}$$

۳-۱-۲-۱- الگوریتم مجموعه ی احاطه گر با شعاع r روی یک درخت رأس - وزین

$$1- T' \leftarrow T \text{ و } P \leftarrow 0 \text{ و به ازای هر رأس } v \text{ از } T', R(v) \leftarrow -\frac{r}{w(v)}$$

۲- اگر هیچ برگی در درخت کمکی T' ، مثل v_r پوشش داده نشده بود یعنی $R(v_r) \geq 0$ نبود آنگاه به گام ۵ بروید. اگر درخت کمکی T' ، تک رأسی بود با تنها رأس v_r که $R(v_r) \geq 0$ ، آنگاه توقف کنید. (درخت اولیه با شعاع r توسط p نقطه ی مجموعه احاطه گر پوشش داده شده است) در غیر این صورت v_r را رأسی بگیرید که $R(v_r) \geq 0$. رأس v_r و یال مجاور به آن در T' یعنی $e = (v_r, v_s)$ را از درخت T' حذف کنید.

۳- (با کمی دقت معلوم می‌شود نقطه‌ای که رأس v_r را پوشش می‌دهد در فاصله $R(v_r) + l(e)$ از v_s قرار دارد.) اگر $R(v_r) + l(e) \leq r/w(s)$ آنگاه به ۴ بروید، اگر $R(v_r) + l(e) > r/w(s)$ آنگاه به ۲ برگردید (در این صورت نقطه‌ای که v_r را پوشش داده است، v_s را نمی‌پوشاند).

۴- اگر $R(v_r) + l(e) \leq r/w(s)$ باشد نقطه‌ای که v_r را پوشش داده، v_s را نیز می‌پوشاند و حالات زیر را داریم:
 اگر $0 \leq R(v_s) \leq R(v_r) + l(e)$ ، قبلاً با نقطه‌ی نزدیک‌تری پوشیده شده و به گام ۲ برگردید.
 اگر $0 < R(v_r) + l(e) < R(v_s)$ آنگاه نقطه‌ای که v_r را پوشانده به v_s ، از نقطه‌ای که قبلاً v_s را پوشش داده نزدیک‌تر است پس بروز رسانی انجام دهید $R(v_s) \leftarrow R(v_r) + l(e)$ و به گام ۲ بروید. همچنین اگر $0 < R(v_r) + l(e) < -R(v_s)$ آنگاه نقطه‌ای که v_r را پوشش می‌دهد اعضای مجموعه‌ی $s(v_s)$ را هم می‌پوشاند و لذا اگر $0 < -R(v_s) < R(v_r) + l(e)$ باشد نقطه‌ای که v_r را پوشش می‌دهد تمامی نقاط $s(v_s)$ را نمی‌تواند بپوشاند و به گام ۲ برگردید. در دو حالت اول رأس v_s قبلاً پوشش داده شده بود لذا با متغیر $R(v_s)$ کاری‌شود. اما در دو حالت بعدی v_s قبلاً پوشش داده نشده است لذا با متغیر $-R(v_s)$ ادامه داده می‌شود.

۵- اگر به ازای همه‌ی برگ‌های درخت T' ، $R(v_r) < 0$ باشد پس هیچ برگ‌ی پوشش داده نشده است. اگر درخت کمکی T' تک رأسی بود آنگاه $p \leftarrow p+1$ و نقطه‌ی جدید از مجموعه احاطه‌گر را در رأس v_r از T' قرار دهید و توقف کنید. در غیر این صورت v_r را برگ‌ی از T' و $e = (v_r, v_s)$ را یال مجاور به v_r در T' بگیرید. v_r و e را از T' حذف کنید. (حال نقطه‌ی جدید را در فاصله‌ی $R(v_r) -$ از v_r قرار دهید تا هم خودش و هم مجموعه‌ی $s(v_r)$ را پوشش دهد)

۶- اگر $R(v_r) \geq l(e)$ به گام ۷ بروید.

اگر $R(v_r) = l(e)$ به گام ۸ بروید.

اگر $R(v_r) < l(e)$ به گام ۹ بروید.

۷- اگر $-R(v_r) > l(e)$ باشد نیاز نیست آن نقطه‌ای از مجموعه احاطه‌گر که $s(v_r)$ را پوشش می‌دهد روی یال e واقع گردد. اگر $-R(v_r) - l(e) < -R(v_s)$ پس $R(v_s) < 0$ و هنوز پوشش داده نشده است. حذف v_r از T' را ادامه دهید. اکنون رئوس $s(v_r)$ در $s(v_s)$ گنجانده می‌شوند. پس مقدار $R(v_s)$ می‌بایست بروز شود. (چون همچنان v_s پوشش داده نشده است در حالیکه $s(v_s)$ تغییر نموده است) $R(v_s) \leftarrow R(v_r) + l(e)$ تا اگر نقطه‌ی جدید را بر اساس $-R(v_s)$ مکان‌یابی کردیم مجموعه-ی $s(v_r)$ را نیز پوشش دهد و به گام ۲ برگردید. اگر $0 < -R(v_s) \leq -R(v_r) - l(e)$ پس دیگر نیازی به تغییر $R(v_s)$ و بروزرسانی آن نیست و مجدداً به گام ۲ برگردید. (چون $-R(v_r) - l(e)$ بیشترین فاصله از رأس v_s می‌باشد که اگر نقطه‌ی جدید مجموعه‌ی احاطه‌گر در طول آن قرار گیرد $s(v_r)$ را نیز پوشش خواهد داد.) اگر $0 < R(v_s) \leq -R(v_r) - l(e)$ پس v_s قبلاً پوشش داده شده است و چون اگر نقطه‌ی جدید را در فاصله‌ی $-R(v_r)$ از v_r قرار دهیم فاصله‌ی آن تا v_s ، $-R(v_r) - l(e)$ خواهد بود، پس نقطه‌ی پوشش دهنده‌ی v_s ، مجموعه‌ی رئوس $s(v_r)$ را نیز می‌پوشاند و به گام ۲ برگردید. اما اگر $0 < -R(v_r) - l(e) < R(v_s)$ آنگاه گرچه v_s پوشش داده شده است اما نقطه‌ی پوشش دهنده‌ی آن نمی‌تواند مجموعه رئوس $s(v_r)$ را پوشش دهد. در این حالت فرض می‌کنیم که اصلاً v_s پوشش داده نشده است و دنبال نقطه‌ای می‌گردیم که هم v_s و هم $s(v_r)$ یعنی $s(v_r) \cup \{v_s\}$ را پوشش دهد و مقدار $R(v_s)$ می‌بایست بروزرسانی شود، $R(v_s) \leftarrow R(v_r) + l(e)$ و به ۲ برگردید.

۸- اگر $-R(v_r) = l(e)$ ، برای اینکه مجموعه‌ی $s(v_r)$ پوشش داده شود نقطه‌ی جدید مجموعه احاطه‌گر می‌بایست در رأس v_s واقع گردد، اگر $R(v_s) = 0$ پس قبلاً نقطه‌ای از مجموعه احاطه‌گر در رأس v_s واقع گردیده است و گرنه $P \leftarrow P + 1$ و نقطه‌ی جدید مجموعه احاطه‌گر را در رأس v_s قرار دهید $R(v_s) \leftarrow 0$ و به گام ۲ برگردید.

۹- اگر $-R(v_r) < l(e)$ به منظور پوشش $s(v_r)$ ، نقطه‌ی جدید مجموعه احاطه‌گر می‌بایست روی یال e و در فاصله‌ی $-R(v_r)$ از v_r واقع گردد یعنی در فاصله‌ی $l(e) - (-R(v_r)) = l(e) + R(v_r)$ از v_s واقع گردد پس $P \leftarrow P + 1$ و نقطه‌ی جدید مجموعه احاطه‌گر را روی یال e و در فاصله‌ی $-R(v_r)$

از v_r قرار دهید و به گام ۳ بروید. در حالت مجموعه احاطه گر رأسی، رأس v_s نمی تواند $S(v_r)$ را بپوشاند پس نقطه‌ی جدید مجموعه احاطه گر را در رأس v_r قرار داده و $R(v_r) \leftarrow 0$ و $P \leftarrow P + 1$ و به گام ۳ بروید.

p ای که پس از پایان الگوریتم بدست می آید همان عدد احاطه گر 62 و p نقطه‌ای که در طی الگوریتم بدست می آید همان p - مرکز درخت می باشد. پیچیدگی زمانی الگوریتم فوق $O(n)$ می باشد. توجه داریم که روند الگوریتم بر این اساس است که نقطه‌ی جدید به مجموعه احاطه گر زمانی اضافه شود که لازم باشد. مثلاً در گام ۷، می توانستیم نقطه‌ی جدید را در فاصله $R(v_r) -$ از v_r قرار دهیم، منتها ابتدا به بررسی وضعیت v_s پرداخته و سپس نقطه‌ی جدید را مکان یابی نمودیم. (چون می دانیم لزومی ندارد نقطه‌ی جدید روی یال e واقع گردد با آسودگی خاطر یال e را حذف می نماییم) تا از حداقل تعداد نقاط مجموعه احاطه گر بهترین بهره را ببریم. حال آنکه در گام ۸ و ۹ ناگزیر به قرار دادن نقطه - ای جدید در فاصله‌ی $R(v_r) -$ از v_r شدیم (چون می بایست نقطه‌ی جدید روی یال e واقع گردد) و بعد به مرحله‌ی ۳ بازگشتیم و به بررسی وضعیت v_s پرداختیم.

۳-۱-۳ - p - مرکز درخت رأس - وزین

در این بخش ما از الگوریتم ۳-۱-۲-۱ که برای یافتن عدد احاطه گر شعاع r و مجموعه احاطه گر متناظرش طراحی شده بود، جهت یافتن p - شعاع یک درخت رأس - وزین و p - مرکز متناظر آن استفاده می کنیم. در ابتدا نشان می دهیم که $O(n^2)$ مقدار ممکن برای p - شعاع درخت وجود دارد. برای حالت p - شعاع مطلق، X^*_p را p - مرکز مطلق یک درخت رأس - وزین مثل T بگیرید و r_p را، p - شعاع مطلق آن فرض کنید. حداقل نقطه‌ای مثل $x \in X^*_p$ و دو رأس v_1 و v_2 وجود دارند که مسیرهای از x به v_1 و از x به v_2 همدیگر را قطع نمی کنند یعنی

$$d(v_1, v_2) = d(v_1, x) + d(x, v_2)$$

⁶² Dominating number

و نیز

$$w(v_1)d(v_1, x) = w(v_2).d(v_2, x) = r_p$$

زیرا اگر فرض کنیم $w(v_2).d(v_2, x) = r_p$ دو حالت داریم:

اگر $w(v_1)d(v_1, x) > r_p$ آنگاه چون $w(v_2)d(v_2, x) = \max_{v \in V} \{w(v)d(v, X * _p)\} = r_p$ به تناقض

می‌رسیم. اما اگر $w(v_1)d(v_1, x) < r_p$ با جابه‌جا کردن x داریم $w(v_2)d(v_2, x) < r_p$ که با تعریف

p - مرکز تناقض دارد و سپس خواهیم داشت

$$\begin{aligned} r_p &= \frac{w(v_1).w(v_2)}{(w(v_1) + w(v_2))} d(v_1, v_2) = \frac{w(v_1).w(v_2)(d(v_1, x) + d(x, v_2))}{(w(v_1) + w(v_2))} \\ &= \frac{(w(v_1).w(v_2).d(v_1, x)) + (w(v_1).w(v_2).d(x, v_2))}{(w(v_1) + w(v_2))} = \frac{(w(v_1)r_p + w(v_2)r_p)}{(w(v_1) + w(v_2))} = r_p \end{aligned}$$

پس حداکثر $n(n-1)/2$ مقدار ممکن برای p - شعاع مطلق داریم زیرا

$$r_{ij} = \frac{w(v_i).w(v_j)}{w(v_i) + w(v_j)} d(v_i, v_j) \quad ; \quad 1 \leq i < j \leq n \quad (1-3)$$

از آنجایی که ماتریس فاصله‌ها در یک درخت در زمان $O(n^2)$ می‌تواند محاسبه گردد لذا ماتریس

$\|r_{ij}\|$ هم می‌تواند در همین زمان محاسبه گردد. این ماتریس تمامی مقادیر ممکن که r_p می‌تواند

بگیرد را برای هر مقدار p ، به ما می‌دهد.

برای حالت p - شعاع رأسی، فرض کنید $V * _p$ ، p - مرکز رأسی درخت رأس - وزین T و $r_p^{(v)}$ ، p -

شعاع آن باشد. پس با توجه به تعریف p - مرکز داریم

$$r_p^{(v)} = \max_{v \in V} \{w(v).d(v, V * _p)\}$$

بنابراین حداقل دو رأس v_1 و v_2 وجود دارند که

$$r_p^{(v)} = w(v_1).d(v_1, v_2)$$

لذا حداکثر $n(n-1)$ مقدار ممکن برای p - شعاع رأسی داریم یعنی به $n(n-1)$ مقدار داده شده

توسط رابطه‌ی زیر نیاز داریم:

$$r_{ij} = w(v_i).d(v_i, v_j) \quad \text{for} \quad i \neq j \quad 1 \leq i, j \leq n \quad (2-3)$$

اکنون به مسئله‌ی پیدا کردن p - شعاع درخت (مطلق یا رأسی) باز می‌گردیم. فرض کنید p - شعاع (مطلق یا رأسی) مطلوب را با r_p نمایش دهیم و r_{ij} یکی از مقادیر ممکن بدست آمده توسط رابطه‌ی ۱-۳ و ۲-۳ باشد. p_{ij} نیز عدد احاطه‌گر مربوط به شعاع r_{ij} باشد. اگر $P_{ij} \leq P$ آنگاه $r_p \leq r_{ij}$ پس داریم

$$r_p = \min_{i,j} \{r_{ij} | P_{ij} \leq P\}. \quad (3-3)$$

الگوریتم ۱-۳-۲-۱ علاوه بر اینکه برای r_{ij} ، عدد احاطه‌گر p_{ij} را بدست می‌آورد، مجموعه احاطه‌گر متناظر با آن را نیز می‌سازد. پس علاوه بر اینکه p - شعاع، r_p ، درخت T را می‌توانیم حساب کنیم مجموعه احاطه‌گر شعاع r_p را هم پیدا می‌کنیم. فرض کنید p' تعداد نقاط موجود در این مجموعه باشد اگر $p - p'$ نقطه‌ی دلخواه (که البته در حالت رأسی می‌بایست از رئوس درخت انتخاب شوند) را به مجموعه احاطه‌گر بدست آمده اضافه کنیم به p - مرکز مطلوب خود، دست یافته‌ایم.

۱-۳-۱-۳ - الگوریتم p - مرکز یک درخت رأس - وزین

۱- مقادیر r_{ij} را به صورت زیر محاسبه کنید.

برای حالت p - مرکز مطلق :

$$r_{ij} = \frac{(w(v_i).w(v_j))}{(w(v_i) + w(v_j))} d(v_i, v_j) \quad 1 \leq i < j \leq n$$

و برای حالت p - مرکز رأسی:

$$r_{ij} = w(v_i).d(v_i, v_j) \quad \text{for} \quad i \neq j \quad 1 \leq i, j \leq n$$

۲- مقادیر r_{ij} را در لیست L به صورت نزولی مرتب کنید.

۳- با یک جستجوی دودویی روی لیست L و با استفاده از الگوریتم ۱-۳-۲-۱، کوچک‌ترین r_{ij} که $p_{ij} \leq p$ و p_{ij} عدد احاطه‌گر متناظر با شعاع r_{ij} باشد را پیدا کنید. این مقدار را r_p نام گذاری کنید و عدد احاطه‌گر متناظر با آن را با p' نشان دهید.

۴- فرض کنید $X^*_{p'}$ مجموعه‌ی احاطه‌گر متناظر با شعاع r_p باشد که با الگوریتم ۱-۲-۳ بدست آمده است. $p - p'$ نقطه‌ی دلخواه را به مجموعه‌ی $X^*_{p'}$ اضافه نمایید (در حالت p - مرکز رأسی می‌بایست نقاطی اضافه شوند که رأس‌های درخت T باشند) مجموعه‌ی X^*_p که حاصل می‌شود همان p - مرکز درخت است. چون ما با تعداد p' نقطه به هدف مطلوب خود رسیده بودیم لذا اضافه کردن نقاط جدید به $X^*_{p'}$ مقدار p - شعاع را بهبود می‌بخشد.

۳-۱-۲- پیچیدگی زمانی الگوریتم

گام ۱ نیاز به $O(n^2)$ عملیات دارد. همچنین گام ۲، $O(n^2 \cdot \log n)$ عملیات می‌خواهد. پیچیدگی زمانی جستجوی دودویی‌ای که در گام ۳ به کار می‌رود $O(\log n)$ می‌باشد. [۴۴] که چون الگوریتم ۱-۲-۳ دارای پیچیدگی زمانی $O(n)$ می‌باشد پس کلاً گام ۳، $O(n \cdot \log n)$ عملیات می‌طلبد با توجه به توضیحات بالا، پیچیدگی زمانی الگوریتم با پیچیدگی زمانی گام ۲ تعیین خواهد شد و این یعنی پیچیدگی زمانی کل الگوریتم $O(n^2 \cdot \log n)$ خواهد بود.

۳-۱-۴- مکان‌یابی p - مرکز روی درخت رأس - غیر وزین

توسط هندلر [۲۰] نشان داده شده است که مرکز و ۲- مرکز یک درخت غیر وزین می‌تواند در $O(n)$ تکرار پیدا شود. حال اینجا می‌خواهیم p - مرکز یک درخت غیر وزین را بدست آوریم. الگوریتم مبتنی بر قضیه‌ی حکیمی، اسمیشل و پیرس است [17]:

«اگر x^* ، مرکز مطلق T و v_1 رأس محیطی^{۶۳} باشد (رأس محیطی رأسی مثل v_1 است که از رابطه $d(v_1, x^*) = \max_{v \in V} \{d(v, x^*)\}$ بدست می‌آید) و $p(v_1, x^*)$ مسیری باشد که x^* و v_1 را به هم متصل می‌کند. اگر x^* یک رأس نباشد، $p(v_1, x^*)$ شامل یالی خواهد بود که x^* بر آن واقع است. یک یال مثل e روی مسیر $p(v_1, x^*)$ وجود دارد که اگر از درخت T حذف گردد آن را به دو مولفه‌ی

⁶³ Peripheral

T_1 (شامل v_1) و T_2 تقسیم می‌کند که اگر x_1^* ، مرکز مطلق T_1 ، X_{p-1}^* ، $p-1$ - مرکز مطلق T_2 باشند آنگاه، $X_p^* = \{x_1^*\} \cup X_{p-1}^*$ ، p - مرکز مطلق T خواهد بود.»

۳-۱-۴-۱ - الگوریتم p - مرکز مطلق روی درخت رأس - غیر وزین

۱- مرکز مطلق x^* از درخت T و همچنین رأس محیطی v_1 را بیابید. (این گام به $O(n)$ عملیات نیازمند است.)

۲- فرض کنید $p(v_1, x^*)$ ، مسیری باشد که x^* و v_1 را به هم متصل می‌کند. اگر x^* یک رأس نباشد آنگاه $p(v_1, x^*)$ شامل یالی خواهد بود که x^* در آن واقع است. برای هر یال از $p(v_1, x^*)$ ، $T_1(e)$ را مولفه‌ای از $T - e$ بگیرید که شامل v_1 باشد و $T_2(e)$ مولفه‌ی دیگر $T - e$ باشد. مرکز (مطلق) $x_1^*(e)$ و شعاع (مطلق) $r_1^1(e)$ از $T_1(e)$ را پیدا کنید که در $O(n)$ تکرار انجام می‌پذیرد. $(p-1)$ - مرکز (مطلق) $X_{p-1}^*(e)$ و $(p-1)$ - شعاع (مطلق) $r_{p-1}^2(e)$ از $T_2(e)$ را بیابید (که با یک رابطه‌ی بازگشتی پس از $O(n^{p-2})$ عملیات بدست می‌آید، یعنی فعلاً فقط یک یال از مسیر $p(v_1, x^*)$ حذف شود.)

سپس قرار دهید:

$$r_p(e) = \max\{r_1^1(e), r_{p-1}^2(e)\}$$

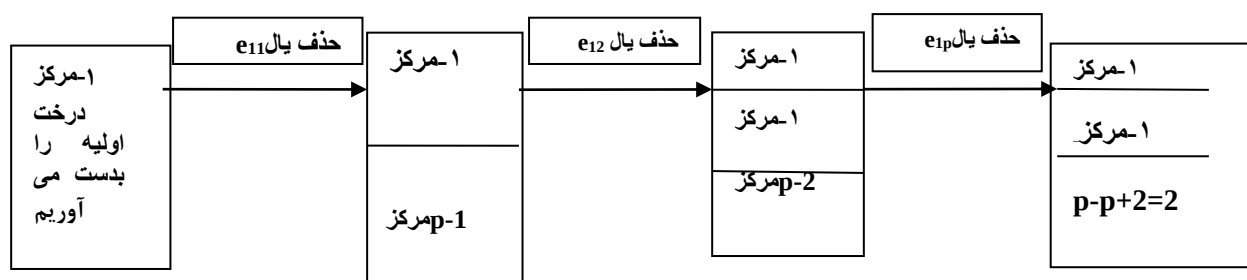
۳- قرار دهید:

$$r_p(e_0) = \min_{e \in p(v_1, x^*)} \{r_p(e)\}$$

پس $r_p(e_0)$ ، p - شعاع (مطلق) T و مجموعه‌ی $\{x_1^*(e_0)\} \cup X_{p-1}^*(e_0)$ یک p - مرکز (مطلق) T است.

۳-۱-۴-۲- پیچیدگی زمانی الگوریتم

پیچیدگی زمانی الگوریتم به پیچیدگی زمانی گام ۲ بستگی دارد. حال یال‌های حذفی در هر مرحله ممکن است متفاوت باشند. لذا با فرض اینکه تعداد یال‌ها m باشد و بسته به اینکه کدام یال از درخت را انتخاب نماییم داریم



p - از آنجایی که یافتن مرکز درخت غیر وزین پیچیدگی زمانی $O(n)$ دارد لذا با توجه به اینکه در هر مرحله حداکثر m حالت داریم پیچیدگی زمانی کل الگوریتم به صورت زیر خواهد شد

$$m.O(n) \times m.O(n) \times \dots \times m.O(n) = m^{p-2}.O(n)$$

$$m^{p-2}.O(n) = O(n^{p-1}) \quad \text{اگر } m = n-1 \text{ آنگاه}$$

لم ۳-۱- فرض کنید $T(V, E)$ یک درخت رأس-غیروزین باشد v_1 و v_2 ، رأس محیطی باشند به گونه‌ای که $d(v_1, v_2) = \max_{v, v' \in V} \{d(v, v')\}$. برای هر یال e روی مسیر $p(v_1, v_2)$ که دو رأس v_1 و v_2 را بهم متصل می‌کند دو زیر درخت همبند $T_1(e)$ و $T_2(e)$ که از حذف یال e بدست آمده‌اند و ضمناً $T_1(e)$ شامل v_1 باشد را در نظر بگیرید. پس به ازای هر $p > 1$ یک یال e_p روی $p(v_1, v_2)$ و 2 عدد مثبت صحیح p_1 و p_2 وجود دارند که $p = p_1 + p_2$ و اگر $V_{p_1}^*$ ، p_1 -مرکز $T_1(e_p)$ و $V_{p_2}^*$ ، p_2 -مرکز $T_2(e_p)$ باشند آنگاه $V_p^* = V_{p_1}^* \cup V_{p_2}^*$ یک p -مرکز از T خواهد بود.

اثبات :

فرض کنید $V_p^* = \{u_1, u_2, \dots, u_p\}$ - مرکز درخت T باشد. برای هر $1 \leq i \leq p$ را مجموعه رؤسی از درخت تعریف کنید که u_i نزدیکترین رأس در بین رؤوس V_p^* به آنها باشد، یعنی $U_i = \{v \in V \mid d(v, u_i) = d(v, V_p^*)\}$. اگر اندیسی مثل i_0 وجود داشت که تمامی رؤوس روی مسیر $p(v_1, v_2)$ به U_{i_0} تعلق داشت آنگاه u_{i_0} مرکز درخت T خواهد بود. در نتیجه p - شعاع درخت T همان شعاع متناظر u_{i_0} خواهد بود. در این حالت e_p تنها یال مجاور به v_2 است. بنابراین $T_1(e_p) = T - \{v_2\}$ و $T_2(e_p) = \{v_2\}$. اگر $p_1 = p - 1, p_2 = 1$ و $V_{p_1}^*$ - مرکز $T_1(e_p)$ و $V_{p_2}^*$ - مرکز $T_2(e_p)$ باشد آنگاه $V_p^* = V_{p_1}^* \cup V_{p_2}^*$ یک p - مرکز از T خواهد بود. پس فرض می-کنیم که تمامی رؤوس مسیر $p(v_1, v_2)$ به یک U_{i_0} متعلق نباشند و $e_p = (v_r, v_s)$ یالی روی $p(v_1, v_2)$ باشد که $v_r \in U_i$ و $v_s \notin U_i$. p_1 و p_2 را به ترتیب تعداد رؤوس V_p^* روی $T_1(e_p)$ و $T_2(e_p)$ بگیرید. بوضوح $p = p_1 + p_2$. اگر $V_{p_1}^*$ و $V_{p_2}^*$ به ترتیب p_1 - مرکز $T_1(e_p)$ و p_2 - مرکز $T_2(e_p)$ باشند آنگاه $V_p^* = V_{p_1}^* \cup V_{p_2}^*$ یک p - مرکز از T خواهد بود. اکنون به ازای هر یال روی $p(v_1, v_2)$ و هر $1 \leq p_1 < p, p_2 = p - p_1$ ، $r_{p_1}(e)$ را p_1 - شعاع $T_1(e)$ و $r_{p_2}(e)$ را p_2 - شعاع $T_2(e)$ گرفته و جایگزین کنید

$$r_{p_1, p_2}(e) = \max\{r_{p_1}(e), r_{p_2}(e)\}$$

$$r_{p_1, p_2} = \min_{e \in p(v_1, v_2)} \{r_{p_1, p_2}(e)\}$$

پس $r_p = \min\{r_{p_1, p_2} \mid 1 \leq p_1 < p, p_2 = p - p_1\}$ - شعاع درخت T خواهد بود.

۳-۴-۱- مرکز رأسی روی درخت رأس - غیر وزین

۱-۲ رأس محیطی v_1 و v_2 را بیابید که $d(v_1, v_2) = \max_{v, v' \in V} \{d(v, v')\}$ و $p(v_1, v_2)$ را مسیر

متصل کننده v_1 و v_2 بگیرید.

۲- به ازای هر $p_1 = 1, \dots, p-1$ ، گامهای ۳ و ۴ را انجام دهید.

۳- با یک جستجوی دودویی روی یال‌های مسیر $p(v_1, v_2)$ ، $\{r_{p_1, p_2}(e)\}$ را $\min_{e \in p(v_1, v_2)}$ بیابید،

درحالی‌که به ازای هر یال e جستجو شده مراحل زیر را انجام دهید:

(الف) p_1 - مرکز رأسی $V_{p_1}^*$ و p_1 - شعاع رأسی $r_{p_1}(e)$ از $T_1(e)$ را بیابید. (در حالی‌که $T_1(e)$ زیر

درخت همبند حاصل از حذف e از T می‌باشد که شامل v_1 است)

(ب) برای $p_2 = p - p_1$ ، p_2 - مرکز رأسی $V_{p_2}^*$ و p_2 - شعاع رأسی $r_{p_2}(e)$ از $T_2(e)$ بدست آورید.

(در حالی‌که $T_2(e)$ زیر درخت همبند $T - e$ شامل v_2 می‌باشد).

(ج) جایگذاری کنید

$$r_{p_1, p_2}(e) = \max\{r_{p_1}(e), r_{p_2}(e)\}$$

۴- e_{p_1, p_2} را یالی بگیرید که روی $p(v_1, v_2)$ باشد و

$$r_{p_1, p_2}(e_{p_1, p_2}) = \min_{e \in p(v_1, v_2)} \{r_{p_1, p_2}(e)\}$$

سپس فرض کنید

$$r_{p_1, p_2} = r_{p_1, p_2}(e_{p_1, p_2}), \quad V_{p_1, p_2}^* = V_{p_1}^*(e_{p_1, p_2}) \cup V_{p_2}^*(e_{p_1, p_2})$$

۵- جایگذاری کنید

$$r_p \leftarrow \min\{r_{p_1, p_2} \mid 1 \leq p_1 < p, p_2 = p - p_1\}$$

و V_p^* را مجموعه‌ی V_{p_1, p_2}^* متناظر با r_p بگیرید. پس r_p ، p - شعاع رأسی درخت T و V_p^* ،

p - مرکز رأسی T خواهند بود.

۳-۱-۴ پیچیدگی زمانی الگوریتم

پیچیدگی زمانی این الگوریتم $O(n \log^{p-1} n)$ می‌باشد که به صورت زیر می‌توانیم ادعای خود را ثابت

کنیم. فرض کنیم ادعای بالا برای m - مرکز که $m < p$ صحیح باشد (برای $p = 2$ نیز درست می‌-

باشد) هنگامیکه می‌خواهیم p - مرکز درخت T را بیابیم گام ۳- (الف) دارای پیچیدگی زمانی

$O(n \log^{p_1-1} n)$ و گام ۳- (ب) دارای پیچیدگی زمانی $O(n \log^{p_2-1} n)$ خواهد بود و بعلاوه جستجوی

دودویی که انجام دادیم گام ۳، $O(\log n)$ بار، تکرار می‌شود. پس پیچیدگی زمانی کل گام ۳،

$O(n \lg^{p_1} n + n \lg^{p_2} n)$ خواهد بود. که کل حلقه‌ی گام‌های ۳-۴ دارای پیچیدگی زمانی زیر

خواهد بود

$$\sum_{p_1=1}^{p-1} O(n \lg^{p_1} n + n \lg^{p-p_1} n) = O(2n \sum_{p_1=1}^{p-1} \log^{p_1} n) = O(2n \frac{\log^p n - \log n}{\log n - 1}) = O(n \log^{p-1} n)$$

پس الگوریتم فوق در زمان کمترین $O(n^2)$ کار می‌کند.

فصل چهارم

مکان‌یابی مرکز با وزن مثبت و منفی در شبکه

۴-۱- مکان‌یابی مرکز با وزن مثبت / منفی^{۶۴}

در یک شبکه با رئوسی با وزن مثبت و منفی، مسئله‌ی مرکز با وزن مثبت و منفی می‌خواهد ترکیب خطی بیشینه فاصله‌ی مرکز تا رئوس با وزن مثبت و نیز تا رئوس با وزن منفی را، حداقل کند. برای مطالب این بخش از مرجع [۲] استفاده شده است. فرض کنید $G(V, E)$ یک شبکه‌ی همبند با مجموعه‌ی رئوس V و $|V| = n$ و مجموعه یال‌های E و $|E| = m$ باشد. علاوه بر این طول یک یال دلخواه مثل e را با عدد مثبت $l(e)$ نمایش می‌دهیم و عدد حقیقی $w(v)$ وزن هر $v \in V$ را نشان می‌دهد. اگر رأس v مکان مطلوبی برای ما باشد $w(v) \geq 0$ و مجموعه‌ی این مکان‌های مطلوب را V_1 می‌نامیم. اما اگر رأس v مکان نامطلوبی برای ما باشد آنگاه $w(v) < 0$ و مجموعه‌ی این مکان‌های نامطلوب را V_2 می‌نامیم.

هزینه‌های سرویس‌دهی مکان‌های مطلوب در مجموعه‌ی V_1 با تابع زیر داده می‌شود

$$f(x) = \max_{v \in V_1} \{w(v)d(x, v)\}$$

و از طرف دیگر زیان‌های ناشی از سرویس‌دهی مکان‌های نامطلوب به صورت زیر خواهد بود

$$g(x) = \max_{v \in V_2} \{w(v)d(x, v)\}$$

در ادامه ما مجموع هزینه‌های سرویس‌دهی و زیان دهی را کمینه می‌نماییم یعنی

$$\min_{x \in G} E(x)$$

$$E(x) = f(x) + g(x)$$

حال اگر تابع $g(x)$ را به صورت زیر بازنویسی نماییم

$$g(x) = -\min_{v \in V_2} \{-w(v)d(x, v)\} = -g_1(x)$$

$$g_1(x) = \min_{v \in V_2} \{-w(v)d(x, v)\}$$

پس در نتیجه خواهیم داشت

⁶⁴ Center problems with pos/neg weights

$$\begin{aligned}
\min_{x \in G} E(x) &= \min_{x \in G} (f(x) + g(x)) = \min_{x \in G} (f(x) + \max_{v \in V_2} (w(v)d(x, v))) = \\
&= \min_{x \in G} (f(x) - \min_{v \in V_2} (-w(v)d(x, v))) = \\
&= \min_{x \in G} (f(x)) - \max_{x \in G} (\min_{v \in V_2} (-w(v)d(x, v))) \\
&= \min_{x \in G} f(x) - \max_{x \in G} (g_1(x))
\end{aligned}$$

با توجه به اینکه $f(x)$ که تابع هدف مرکز رأس-وزین و $g_1(x)$ تابع هدف مرکز نامطلوب رأس-وزین می‌باشد، $E(x)$ نیز هر دو مدل را تولید می‌کند. در ادامه برای حالت غیروزین دو زیر مجموعه‌ی V_1 و V_2 را مد نظر قرار داده و توابع هدف $f(x)$ و $g(x)$ را با توجه به این زیر مجموعه‌ها به صورت زیر تعریف می‌کنیم

$$f(x) = \max_{v \in V_1} (d(x, v))$$

$$g(x) = \max_{v \in V_2} (-d(x, v))$$

$$\min_{x \in G} (f(x) + g(x))$$

۴-۱-۱- مکان‌یابی مرکز با وزن مثبت و منفی روی شبکه‌ای کلی

کریو و حکیمی نشان داده‌اند که مسئله‌ی مرکز وزین روی شبکه‌ی G در حالت مثبت خالص در زمان $O(mn \log n)$ حل می‌شود [۲۲] و تامیر نشان داده که پیچیدگی زمانی مسئله‌ی مرکز نامطلوب

روی یک شبکه $O(mn)$ ، می‌باشد با این فرض که ماتریس فاصله‌ها داده شده باشد. [۳۵]

یک نظریه و پیشنهاد جهت حل مسئله‌ی مرکز با وزن مثبت و منفی روی شبکه، محاسبه کردن تحدید توابع $f(x)$ و $g(x)$ روی تک تک یال‌های شبکه می‌باشد. همانطور که در فصل دوم داشتیم کریو و حکیمی نشان داده‌اند که تحدید $f(x)$ در زمان $o(n \log n)$ قابل محاسبه بوده [۲۲] و اگر از الگوریتم مرکز نامطلوب روی یک مسیر استفاده نماییم (که در ادامه به آن اشاره شده است) [۳]، $g(x)$ هم در زمان $o(n)$ محاسبه می‌گردد.

از آنجایی که توابع $f(x)$ و $g(x)$ قطعه قطعه خطی می‌باشند، $E(x)$ نیز قطعه قطعه خطی خواهد بود. با محاسبه $f(x)$ و $g(x)$ روی هر یال، $E(x)$ هم تعیین خواهد شد که با محاسبه‌ی جواب بهین نسبی روی هر یال و مقایسه جواب‌های بهین نسبی با یکدیگر و یافتن بهین مطلق، مسئله‌ی مرکز با وزن مثبت و منفی را حل نموده‌ایم. بنابراین خواهیم داشت:

مسئله‌ی مرکز با وزن مثبت و منفی روی یک شبکه در زمان $O(mn \log n)$ قابل حل می‌باشد.

۴-۱-۲- مکان‌یابی مرکز نامطلوب روی مسیر

بورکارد [۳] نشان داده است که اگر گراف تحت بررسی جهت یافتن مرکز نامطلوب یک مسیر باشد با n رأس، به صورت زیر باید به حل آن پرداخت. رئوس را روی محور اعداد حقیقی قرار داده و به صورت زیر اعداد حقیقی به آنها نسبت می‌دهیم

$$0 = x_1 < x_2 < \dots < x_n$$

$$d(x_i, x_j) = |x_i - x_j|$$

بنابر این تابع هدف به صورت زیر خواهد بود

$$g(z) = \min\{w_i |z - x_i|, i = 1, 2, \dots, n\} = \min\{g^+(z), g^-(z)\}$$

که

$$g^+(z) = \min\{w_i(z - x_i) : x_i \leq z\}$$

$$g^-(z) = \min\{w_i(x_i - z) : x_i \geq z\}$$

در این قسمت یک الگوریتم خطی جهت یافتن بیشینه مقدار تابع $g(z)$ در طول یک مسیر ارائه می‌شود. ابتدا توضیح می‌دهیم که چگونه تابع $g^+(z)$ از چپ به راست محاسبه نموده از اینرو به طریق مشابه خواهیم توانست تابع $g^-(z)$ را محاسبه کرده در نتیجه محاسبه $g(z)$ و یافتن بیشینه مقدار آن آسان خواهد شد.

در ابتدا توابع $g_1^+, g_2^+, \dots, g_{n-1}^+$ که به صورت زیر تعریف می‌شوند را محاسبه می‌نماییم

$$g_j^+ : [x_j, x_n] \rightarrow R^+, \quad g_j^+(z) = \min\{w_i(z - x_i), i = 1, 2, \dots, j\}$$

بلافاصله از تعریف خواص زیر نتیجه می‌شود:

لم ۴-۱-

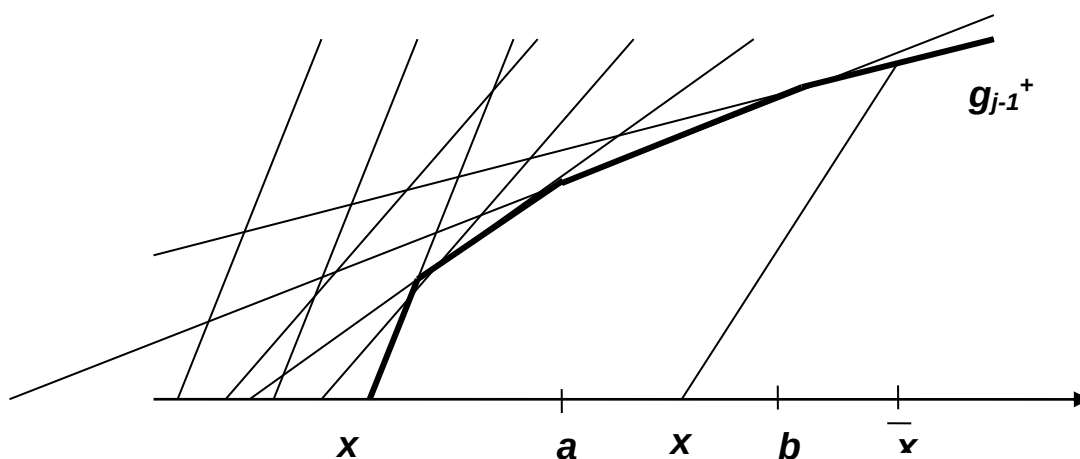
$$\forall z; x_j \leq z \leq x_{j+1}; g^+(z) = g_j^+(z) \quad (\text{الف})$$

(ب) $g_j^+(z)$ یک تابع قطعه قطعه خطی و صعودی می باشد.

$$g_1^+(z) = w_1(z - x_1) \quad (\text{ج})$$

$$\forall j = 2, \dots, n-1, g_j^+(z) = \min\{w_j(z - x_j), g_{j-1}^+(z)\} \quad ; \quad z \geq x_j \quad (\text{د})$$

(ه) تابع $g^+(z)$ قطعه قطعه خطی بوده و حداکثر $2n - 3$ قطعه ی خطی دارد.



شکل ۴-۱- نمودار g_{j-1}^+

۴-۱-۲- الگوریتم بدست آوردن ساختار g_j^+ با استفاده از g_{j-1}^+

گام اول -

ابتدا لیست بازه های g_{j-1}^+ از x_{j-1} تا زمانی که به x_j برسیم پویش می کنیم. طبق لم ۴-۱- (الف) این لیست از بازه ها را حذف و به لیست بازه هایی که تابع g^+ را تعریف می نمایند کپی می نماییم. به طور کلی نقطه ی x_j در میان یک بازه واقع می گردد که این بازه همانطور که در شکل دیده می شود به دو قسمت تقسیم گردیده است. قسمت چپ بازه جهت تخمین مقدار تابع g^+ بکار می رود و قسمت راست آن به عنوان بخشی از تعریف تابع g_{j-1}^+ باقی می ماند.

گام دوم -

بعد از این تبدیل دامنه‌ی تابع g_{j-1}^+ به $[x_j, x_n]$ کاهش می‌یابد، که همان دامنه‌ی تابع g_j^+ می‌باشد که به دنبال محاسبه‌ی آن می‌باشیم. اکنون لم ۴-۱-۵ (د) را بکار می‌بریم. تابع g_j^+ با بازه‌ای شروع می‌شود که $g_j^+ = w_j(z - x_j)$ ، این بازه تا زمانی که تابع g_j^+ نمودار $g_{j-1}^+(z)$ را قطع می‌کند امتداد می‌یابد و از نقطه‌ی تلاقی به بعد با تابع $g_{j-1}^+(z)$ منطبق می‌گردد. اگر بخواهیم جامع‌تر توضیح دهیم بدین‌صورت است که لیست بازه‌های باقی مانده‌ی g_{j-1}^+ را از سمت چپ پویش کرده و مقدار تابع g_{j-1}^+ را در هر بازه با مقدار تابع $w_j(z - x_j)$ مقایسه می‌کنیم، تا زمانی که که این تابع در طول یک بازه کوچک‌تر از g_{j-1}^+ می‌باشد آن بازه را از لیست بازه‌ها حذف می‌نماییم. زمانی که دو تابع همدیگر در $\bar{x}$ را قطع نمایند بازه را از محل تقاطع شان به دو قسمت تقسیم نموده نیمه چپ بازه را با بازه‌ی $[x_j, \bar{x}]$ جایگزین می‌نماییم. در واقع در گام اول فقط بازه‌ی $[a', b']$ برایمان باقی ماند. چون هر بازه نمایشگر یک قطعه خط می‌باشد و قسمتی از این بازه بعد از x_j واقع گردید لذا بازه‌ی $[a', b']$ حذف نمی‌شود. در گام دوم هم فقط یک بازه برایمان باقی ماند و آن هم بازه‌ای بود که تابع $w_j(z - x_j)$ جهت تعریف g_j^+ مورد استفاده قرار گرفت که با کلیه‌ی بازه‌های قبلی جایگزین شد. پس برای هر g_j^+ دو بازه (قطعه خط) باقی ماند. اکنون می‌توانیم لم ۴-۱-۵ (ه) را اثبات نماییم. از آنجایی که هر کدام از این بازه‌های باقی مانده نشان دهنده‌ی یک قطعه خط صعودی می‌باشند و تابع g_1^+ هم فقط یک قطعه خط دارد، لذا چون $n - 2$ تابع به این شکل داریم که اگر بخواهیم با توجه به روند بالا به شکل دهی ساختارشان بپردازیم در مجموع تابع g^+ تعداد $1 + (n - 2)(1 + 1)$ قطعه خط خواهد داشت. مشابه روند بالا تابع g^- را نیز از راست به چپ محاسبه می‌نماییم.

۴-۲-۱-۲- پیچیدگی زمانی الگوریتم

از آنجایی که در گام اول هر کدام از بازه‌هایی که پویش می‌شوند یک قطعه خط از تابع g^+ نمایش می‌دهند لذا با توجه به لم ۴-۱-۵ (ه) چون در مجموع $1 + (n - 2)(1 + 1)$ قطعه داریم زمان کلی جهت

انجام گام اول $O(n)$ خواهد بود. همچنین زمان لازم جهت انجام گام دوم $O(1+k)$ خواهد بود در حالیکه k تعداد بازه‌های حذف شده در گام دوم می‌باشد. از آنجایی که تعداد بازه‌های حذف شده از تعداد اولیه‌ی بازه‌ها که $O(n)$ تا بوده نمی‌تواند بیشتر شود پس ادعای ما درست خواهد بود. ضمناً مقایسه‌ی توابع در بازه‌ها و همچنین محاسبه نقطه‌ی اشتراکشان زمان ثابتی نیاز دارد.

فصل پنجم

کاربرد مدل ترکیبی تاپسیس و مکان‌یابی مرکز درخت وزین

۵-۱- مقدمات

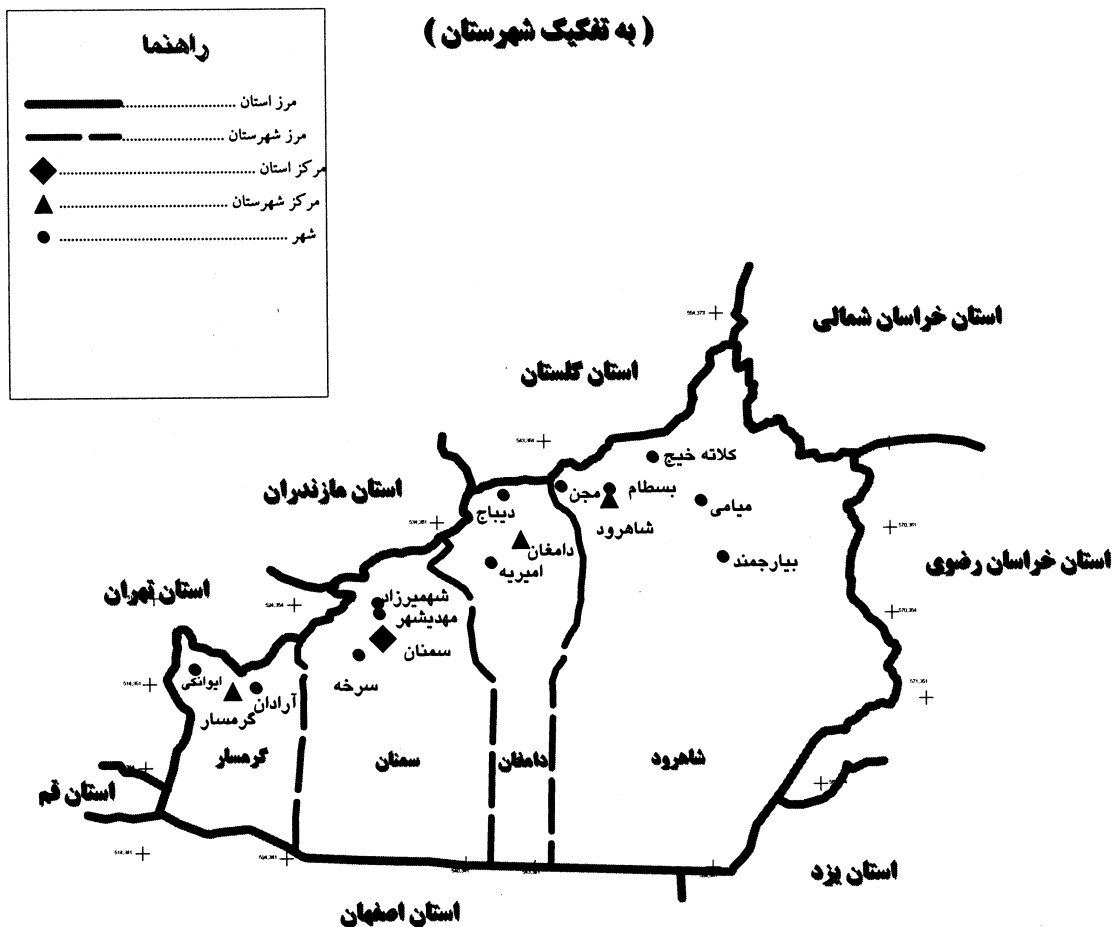
استان سمنان بر اساس آخرین تقسیمات کشوری با داشتن ۵ شهرستان (دامغان، سمنان، شاهرود، گرمسار و مهدیشهر)، ۱۳ بخش، ۱۷ شهر و ۲۹ دهستان، دارای مساحتی معادل ۹۷۴۹۱ کیلومترمربع می‌باشد که بین ۵۱ درجه و ۵۱ دقیقه تا ۵۷ درجه و ۳ دقیقه طول شرقی و ۳۴ درجه و ۱۳ دقیقه تا ۳۷ درجه و ۲۰ دقیقه عرض شمالی از نصف‌النهار گرینویچ قرار گرفته‌است.

این استان از جانب شمال به استان‌های خراسان شمالی، گلستان و مازندران، از جنوب به استان‌های یزد و اصفهان، از مشرق به استان خراسان رضوی و از مغرب به استان‌های تهران و قم محدود است و مرکز آن شهر سمنان می‌باشد. به شکل ۵-۱ و ۵-۲ رجوع کنید.

براساس نتایج سرشماری عمومی نفوس و مسکن، جمعیت استان در سال ۱۳۸۵ برابر ۵۸۹۷۴۲ نفر بوده که از این تعداد ۴۴۰۵۵۹ نفر در مناطق شهری و ۱۴۹۱۸۳ نفر در مناطق روستایی ساکن بوده‌اند و تراکم نسبی جمعیت در همین سال ۶/۰۵ نفر در هر کیلومترمربع است. [40]

استان سمنان از لحاظ وسعت ششمین استان کشور می‌باشد، این استان در مسیر ترانزیت کالا از غرب به شرق کشور است. به شکل ۵-۳ نگاه کنید. استان سمنان در چند سال اخیر رشد قابل توجهی در زمینه مبادلات کالاهای صنعتی و همچنین خدمات علمی و فنی داشته است. یکی از مشکلات موجود در این عرصه عدم وجود فرودگاه فعال در این استان است، با توجه به گسترش علمی و صنعتی این استان، احداث فرودگاه امری ضروری می‌باشد. مکان‌یابی دقیق و علمی فرودگاه با توجه به وسعت استان از اهمیت زیادی برخوردار است. تعیین دقیق محل احداث فرودگاه با توجه به شاخص‌های معتبر و موثر و همچنین روش‌های علمی می‌تواند در رشد و توسعه‌ی استان تأثیر گذار باشد و بالعکس تعیین غیر علمی و بدون مطالعه‌ی محل فرودگاه علاوه بر به هدر دادن بودجه‌ای کلان، رشد و توسعه‌ای را نیز برای استان

نقشه پراکندگی نقاط شهری استان سمنان (به تفکیک شهرستان)



سازمان مدیریت و برنامه ریزی استان سمنان
معاونت آمار و اطلاعات
گروه نقشه و GIS

N
W E
S
مقیاس ۱:۳۲۰۵۱۲۹

شکل ۵-۲- موقعیت شهرهای استان سمنان

۵-۲- مفاهیم به کار رفته در این فصل

۵-۲-۱- تعریف شهرستان

واحدی از تقسیمات کشوری است با محدوده جغرافیایی معین که از به هم پیوستن چند بخش همجوار که از نظر عوامل طبیعی و اوضاع اجتماعی، اقتصادی، سیاسی و فرهنگی واحد متناسب و همگنی را به وجود آورده‌اند، تشکیل شده است.

۵-۲-۲- تعریف شهر

محلی است با محدوده قانونی مشخص که در محدوده جغرافیایی بخش واقع شده و از نظر بافت ساختمانی، اشتغال و سایر عوامل، دارای سیمایی با ویژگی‌های شهر می‌باشد.

۵-۲-۳- ایستگاه‌های سینوپتیک (ایستگاه‌های هواگزاری همزمان)

ایستگاه‌هایی است که در آن، عوامل مختلف جوی از قبیل درجه حرارت خشک و تر، مقدار و نوع ابر، مقدار بارندگی، نوع پدیده جوی، سمت و سرعت باد، مقدار تبخیر، فشار، ساعت‌های آفتابی، نحوه تغییرات فشار، میزان تشعشع خورشید و ... بر اساس دستورالعمل‌های مشخص و در رأس ساعت‌های معین (به طور همزمان در سراسر جهان)، اندازه‌گیری و پس از ثبت در یک دفتر مخصوص، توسط دستگاه‌های مخابراتی از قبیل **SSB**، تلفن و تلکس، سیستم سوئیچ و **VPN** به مراکز هواشناسی در سراسر جهان مخابره می‌شود.

۵-۳- روش تحقیق

در این تحقیق ۵ شهر را که به عنوان مراکز ۵ شهرستان استان سمنان می‌باشند به همراه شهر بیارجمند به عنوان ۶ رأس درختی انتخاب می‌نماییم که یال‌های آن جاده‌های اصلی بین این شهرها می‌باشد. با استفاده از شاخص‌هایی که توسط کارشناسان مربوطه ارائه شده است می‌توان توسط روش آنالیز

شاخص‌های موثرتر را اولویت داد. ضمناً برای وزن‌دهی به رؤس از روش تاپسیس استفاده می‌شود، پس از وزن‌دهی به رؤس، با استفاده از روش‌های مکان‌یابی بهترین نقطه برای احداث فرودگاه انتخاب می‌شود.

۵-۴- هدف تحقیق

انتخاب مناسب‌ترین محل جهت احداث فرودگاه در استان سمنان توسط روش مکان‌یابی، هدف اصلی این تحقیق است. در این تحقیق از روش آنتروپی و تاپسیس نیز جهت وزن‌دهی به شاخص‌های موثر و رؤس استفاده شده است. شاخص‌های استفاده شده در این تحقیق با توجه به نظرات کارشناسان و مسئولان ذیربط در مجموعه معاونت برنامه‌ریزی استانداری سمنان (سازمان مدیریت و برنامه‌ریزی سابق) و همچنین سازمان هواپیمایی کشور استخراج شده است.

۵-۵-تصمیم‌گیری چند شاخصه

تصمیم‌گیری را می‌توان حرکت در مسیر خاص تعریف نمود که با تأمل و آگاهی از میان راه‌های مختلف برای نیل به یک هدف مطلوب، انتخاب شده است. [۳۰] با اتخاذ هر تصمیم، جریان یا فرآیندی به پایان رسیده است ولی مشکل در این است که سرانجام یک فرآیند را می‌توان شروع فرآیندی دیگر تلقی نمود از این‌رو نگرستن به تصمیم‌گیری از این زاویه مانند قرار دادن یک آینه در مقابل آینه‌ای دیگر است. [۲۸] تصمیم‌گیری به تنهایی مهم‌ترین وظیفه مدیر است، زیرا کیفیت طرح و برنامه‌ها اثربخشی و کارآمدی راهبردها و کیفیت نتایجی که از اعمال آن‌ها بدست می‌آید، همگی تابع کیفیت تصمیماتی است که مدیر اتخاذ می‌کند. [۳۱] مدل‌های بهینه‌سازی از دوران نهضت صنعتی در جهان و بخصوص از زمان جنگ دوم جهانی همواره مورد توجه ریاضی‌دانان و دست‌اندرکاران صنعت بوده است. تأکید اصلی بر مدل‌های کلاسیک بهینه‌سازی، داشتن یک معیار سنجش (یا یک تابع هدف) می‌باشد. اما توجه محققین در دهه‌های اخیر معطوف به مدل‌های چند معیاره برای تصمیم‌گیری‌های پیچیده گردیده است. در این

تصمیم‌گیری‌ها به جای استفاده از یک معیار سنجش بهینگی از چندین معیار سنجش ممکن است استفاده گردد. مدل‌های تصمیم‌گیری به دو دسته عمده تقسیم می‌گردند

الف (مدل‌های چند هدفه (MODM) ^{۶۵}

ب (مدل‌های چند شاخصه (MADM) ^{۶۶}

مدل‌های چند هدفه به منظور طراحی بکارگرفته می‌شوند در حالی که مدل‌های چند شاخصه بمنظور انتخاب گزینه برتر استفاده می‌گردند. به بیانی دیگر برای انتخاب مناسب‌ترین گزینه از بین m گزینه موجود به کار می‌روند. [۳۹]

شاخص عبارت است از ویژگی‌ها یا پارامترهای عملکردی که برای انتخاب راهبردهای تصمیم‌گیری مطرح است. شاخص‌ها ممکن است کمی یا کیفی باشند و شاخص‌های کیفی ممکن است دارای مطلوبیت مثبت یا مطلوبیت منفی بوده و با عباراتی مانند خیلی کم، کم، متوسط، زیاد، خیلی زیاد بیان شوند ولی شاخص‌های کمی با مقدار و با واحد مربوطه بیان می‌شوند. [۴۳]

تصمیم‌گیری‌های چند شاخصه معمولاً توسط ماتریس زیر (ماتریس تصمیم‌گیری) فرموله می‌گردند

شاخص \ گزینه	X_1	X_2	X_n		X_n
A_1	r_{11}	r_{12}	r_{13}		r_{1n}
A_2	.	.	.		.
.	.	.	.		.
.	.	.	.		.
.	.	.	.		.
A_m	r_{m1}	r_{m2}	r_{m3}		r_{mn}

به طوری که A_i نشان دهنده گزینه i ام، X_j نشان‌دهنده شاخص j ام و r_{ij} نشان‌دهنده مقدار شاخص j ام برای گزینه i ام می‌باشد.

⁶⁵ - Multiple Objective Decision Making

⁶⁶ - Multiple attribute Decision Making

۵-۶- مراحل حل مسائل تصمیم‌گیری با استفاده از مدل‌های چند شاخصه

۱- بی‌مقیاس کردن (یکسان‌سازی مقیاس)^{۶۷} .

۲- تعیین وزن شاخص‌ها.

۳- حل مسئله تصمیم‌گیری بوسیله مدل‌ها .

۵-۶-۱- بی‌مقیاس کردن (یکسان‌سازی مقیاس) شاخص‌ها

به طور کلی یک گزینه A_i در مدل‌های چند شاخصه ممکن است توسط دو نوع شاخص کیفی و کمی توصیف شود که در این بررسی از شاخص اول (شاخص کیفی) استفاده نشده و شاخص دوم (شاخص کمی) مورد بحث قرار می‌گیرد. باید توجه داشت که مقیاس اندازه‌گیری شاخص‌های کمی می‌توانند با یکدیگر متفاوت باشند. اما انجام عملیات ریاضی بعد از بی‌مقیاس کردن یا یکسان‌سازی مقیاس‌ها می‌تواند انجام پذیرد.

به منظور یکسان‌سازی مقیاس (بی‌مقیاس کردن) شاخص‌ها، روش‌های مختلفی مانند : «بی‌مقیاس کردن با استفاده از نرم»، «بی‌مقیاس کردن خطی» و «بی‌مقیاسی فازی» وجود دارد.

۵-۶-۱-۱- روش بی‌مقیاس کردن با استفاده از «نرم»

در این روش هر عنصر (r_{ij}) از ماتریس تصمیم‌گیری مفروض را بر نرم موجود از ستون j ام (به ازای شاخص X_i) تقسیم می‌کنیم، یعنی :

$$n_{ij} = \frac{r_{ij}}{\sqrt{\sum_{i=1}^m r_{ij}^2}}$$

⁶⁷ - Dimensionless

۵-۶-۲- تعیین وزن شاخص‌ها

در اکثر مدل‌های چند شاخصه نیاز به داشتن و دانستن اهمیت نسبی از شاخص‌های (اهداف) موجود داریم، به طوری که مجموع آنها برابر با واحد (نرمالیزه)^{۶۸} شده و این اهمیت نسبی درجه ارجحیت هر شاخص (هدف) را نسبت به بقیه برای تصمیم‌گیری مورد نظر می‌سنجد. ارزیابی اوزان برای شاخص‌ها به چهار روش انجام می‌گیرد که عبارتند از:

۱- روش آنتروپی

۲- روش *Linmap*^{۶۹}

۳- روش کم‌ترین مجذورات وزین شده

۴- روش بردار ویژه

روش‌های آنتروپی و *Linmap* نیاز به ماتریس تصمیم‌گیری داشته و در محیط‌های مدل‌های چند شاخصه قابل استفاده نیستند. درحالی که دو روش دیگر نیاز به ماتریس تصمیم‌گیری ندارند.

۵-۶-۲-۱- روش آنتروپی

آنتروپی یک مفهوم عمده در علوم فیزیکی، علوم اجتماعی و تئوری اطلاعات می‌باشد به طوری که نشان‌دهنده میزان عدم اطمینان موجود از محتوای مورد انتظار اطلاعاتی از یک پیام است. به بیان دیگر آنتروپی در تئوری اطلاعات، معیاری است برای میزان عدم اطمینان بیان شده توسط یک توزیع احتمال گسسته به طوری که این عدم اطمینان، در صورت پخش بودن^{۷۰} توزیع، بیشتر از موردی است که توزیع فراوانی تیزتر باشد این عدم اطمینان به صورت ذیل تشریح می‌شود:

ابتدا ارزشی با نماد E محاسبه می‌شود

⁶⁸ - Normalization

⁶⁹ - Linear Programming for Multidimensional Analysis of Preference

⁷⁰ - Broad

$$E = -K \sum_{i=1}^n [p_i \times \ln p_i]$$

بطوریکه K یک عدد ثابت مثبت است و به گونه‌ای تعیین می‌شود که داشته باشیم $0 \leq E \leq 1$

E از توزیع احتمال P_i بر اساس مکانیزم آماری محاسبه شده و مقدار آن در صورت تساوی P_i ها با

یکدیگر (یعنی $P_i = \frac{1}{n}$) ماکزیمم مقدار ممکن خواهد بود، بدینصورت که

$$\begin{aligned} -K \sum_{i=1}^n [p_i \times \ln p_i] &= -K \left\{ \frac{1}{n} \ln \frac{1}{n} + \frac{1}{n} \ln \frac{1}{n} + \dots + \frac{1}{n} \ln \frac{1}{n} \right\} \\ &= -K \left\{ \left(\ln \frac{1}{n} \right) \left(\frac{n}{n} \right) \right\} = -K \ln \frac{1}{n} \end{aligned}$$

ماتریس تصمیم‌گیری از مدل‌های چند شاخصه که قبلاً گفته شد، حاوی اطلاعاتی است که آنتروپی

می‌تواند بعنوان معیاری برای ارزیابی آن بکار رود. محتوای اطلاعاتی موجود از این ماتریس را ابتدا به

صورت P_{ij} در ذیل محاسبه می‌کنیم

$$; P_{ij} = \frac{r_{ij}}{\sum_{i=1}^m r_{ij}} \quad \forall i, j$$

و برای E_j ها به ازای هر مشخصه خواهیم داشت

$$; E_j = -k \sum_{i=1}^m [P_{ij} \times \ln P_{ij}] \quad \forall j$$

بطوریکه $K = \frac{1}{\ln(m)}$ است .

اینک عدم اطمینان یا درجه انحراف (d_j) از اطلاعات ایجادشده به ازای شاخص j ام بدین قرار است

$$d_j = 1 - E_j ; \quad \forall j$$

و سرانجام برای اوزان (w_j) از شاخص‌های موجود خواهیم داشت

$$w_j = \frac{d_j}{\sum_{j=1}^n d_j} ; \forall j$$

چنانچه تصمیم گیرنده ^{۷۱} از قبل دارای یک قضاوت ذهنی (λ_j) به عنوان اهمیت نسبی برای شاخص j باشد، آنگاه می توان w_j محاسبه شده از طریق آنتروپی را به صورت زیر تعدیل نمود

$$W_j' = \frac{\lambda_j \cdot w_j}{\sum_{j=1}^n \lambda_j \cdot w_j} ; \forall j$$

۵-۶-۳- حل مسئله تصمیم گیری بوسیله مدل های تصمیم گیری چند شاخصه

دو دسته عمده از روش های مختلف در تحلیل کردن اطلاعات موجود مدل های چند شاخصه مطرح شده است

۱- مدل غیر جبرانی

۲- مدل جبرانی

۵-۶-۳-۱- مدل غیر جبرانی

شامل روش هایی می شود که در آنها مبادله بین شاخص ها مجاز نیست، یعنی مثلاً نقطه ضعیف موجود در یک شاخص توسط مزیت موجود از شاخص دیگر جبران نمی شود. بنابراین هر شاخص در این روش - ها به تنهایی مطرح بوده و مقایسات بر اساس شاخص به شاخص انجام می پذیرد.

مزیت روش های متعلق به این مدل نیز سادگی آنهاست که به رفتار تصمیم گیرنده و محدود بودن اطلاعات او بستگی دارد. در برخی از این روش ها ممکن است نیازی به کسب اطلاعات از تصمیم گیرنده

نباشد. این مدل شامل ۸ روش به شرح ذیل است

۱- روش رضایت بخش

⁷¹ - Decision maker

۲- روش ماکسی ماکس

۳- روش تسلط

۴- روش رضایت بخش خاص

۵- روش ماکسی مین

۶- روش حذف

۷- روش لکسیکو گراف

۸- روش جایگشت

۵-۶-۳-۲- مدل جبرانی

مشمتمل بر روش‌هایی است که اجازه مبادله در بین شاخص‌ها در آنها مجاز است، یعنی مثلاً تغییری (احتمالاً کوچک) در یک شاخص می‌تواند توسط تغییری مخالف در شاخص (یا شاخص‌های) دیگر جبران شود. این مدل شامل سه زیر گروه می‌گردد

۱- زیر گروه هم آهنگ

۲- زیر گروه سازشی

۳- زیر گروه نمره‌گذاری

۵-۶-۳-۲-۱- روش زیر گروه سازشی^{۷۲}

به طور خلاصه در این روش گزینه‌ای ارجح خواهد بود که نزدیک‌ترین گزینه به نقطه ایده‌آل باشد.

روش‌های متداول در زیر گروه سازشی عبارتند از

۱- روش تاپسیس^{۷۳}

۲- روش **MRS**

⁷² -Compromising Subgroup

⁷³ - Technique for Order Preference by Similarity to Ideal Solution

۳- روش LINMAP

۴- روش MDS

۵-۶-۳-۲-۱-۱- روش تاپسیس

در این روش علاوه بر در نظر گرفتن فاصله یک گزینه A_i از نقطه‌ای ایده‌آل مثبت، فاصله آن از نقطه‌ای ایده‌آل منفی هم در نظر گرفته می‌شود. بدان معنی که گزینه انتخابی دارای کم‌ترین فاصله از نقطه‌ای ایده‌آل مثبت بوده و در عین حال دارای دورترین فاصله از نقطه‌ای ایده‌آل منفی باشد. برای ارزیابی و بررسی مدل‌های چند شاخصه در این تحقیق از مدل جبرانی، زیرگروه سازشی و از روش تاپسیس استفاده شده است. این روش به دلایل زیر نسبت به سایر روش‌های تصمیم‌گیری چند معیاره ارجح دانسته شده است.

اولاً : نسبت به سایر روش‌ها از مبنای ریاضی قوی‌تری برخوردار است.

ثانیاً : نوع تکنیک جبرانی است و ارتباط شاخص‌ها را مدنظر دارد.

ثالثاً : دو نقطه‌ای ایده‌آل مثبت و منفی را در نظر می‌گیرند و گزینه‌ها را با آنها می‌سنجد.

رابعاً : طراحی پرسشنامه و گردآوری اطلاعات برای آن آسان‌تر است.

۵-۶-۳-۲-۱- الگوریتم تاپسیس

قدم اول - تبدیل ماتریس تصمیم‌گیری موجود به یک ماتریس « بی مقیاس شده »

با استفاده از فرمول

$$n_{ij} = \frac{r_{ij}}{\sqrt{\sum_{i=1}^m r_{ij}^2}}$$

قدم دوم - ایجاد ماتریس « بی مقیاس وزین » که با روش آنتروپی جهت وزن دهی به شاخص‌ها بدست آمده است.

یعنی

$$W' = \{w'_1, w'_2, \dots, w'_n\}$$

$$V = N_D \cdot W'_{n \times n} = \text{ماتریس بی مقیاس وزین}$$

بطوری که $N_D = |n_{ij}|$ ماتریسی است که شاخص‌ها در آن « بی مقیاس » و قابل مقایسه شده‌اند و $W'_{n \times n}$ ماتریسی است قطری که فقط عناصر قطر اصلی آن غیر صفر خواهد بود و عناصر روی قطر اصلی همان $\{w'_1, w'_2, \dots, w'_n\}$ خواهند بود.

قدم سوم - مشخص نمودن مجموعه‌ی نقاط ایده‌آل مثبت و ایده‌آل منفی

به صورت زیر مجموعه‌ی نقاط ایده‌آل مثبت و ایده‌آل منفی را مشخص می‌نماییم:

$$A^+ = \{(\max_{1 \leq i \leq m} V_{ij} | j \in J)\} \cup \{(\min_{1 \leq i \leq m} V_{ij} | j \in J')\}$$

$$= \{V_1^+, V_2^+, \dots, V_j^+, \dots, V_n^+\}$$

$$A^- = \{(\min_{1 \leq i \leq m} V_{ij} | j \in J)\} \cup \{(\max_{1 \leq i \leq m} V_{ij} | j \in J')\}$$

$$= \{V_1^-, V_2^-, \dots, V_j^-, \dots, V_n^-\}$$

بطوری که :

$$J = \{j = 1, 2, 3, \dots, n | \text{جمعیت با جنبه مثبت مثل جمعیت}\}$$

$$J' = \{j = 1, 2, 3, \dots, n | \text{های مربوط به شاخص‌های با جنبه منفی مثل سرعت ورزش باد}\}$$

لازم به توضیح است که با توجه به شاخص‌های مورد بررسی در این تحقیق، مجموعه‌ی نقاط ایده‌آل مثبت مشخص می‌شود.

قدم چهارم - محاسبه اندازه جدائی (فاصله)

فاصله گزینه i ام با ایده‌آل‌ها با استفاده از روش اقلیدسی بدین قرار است

$$d_{i+} = \text{فاصله گزینه } i \text{ ام از ایده‌آل مثبت} = \sqrt{\sum_{j=1}^n (V_{ij} - V_j^+)^2} \quad ; i = 1, 2, \dots, m$$

$$d_{i-} = \text{فاصله گزینه } i \text{ ام از ایده‌آل منفی} = \sqrt{\sum_{j=1}^n (V_{ij} - V_j^-)^2} \quad ; i = 1, 2, \dots, m$$

قدم پنجم - محاسبه نزدیکی نسبی گزینه A_i به نقطه‌ی ایده‌آل مثبت

این نزدیکی نسبی را به صورت زیر تعریف می‌کنیم

$$cL_i = \frac{d_{i-}}{(d_{i+} + d_{i-})} \quad ; \quad 0 \leq cL_i \leq 1 \quad ; \quad i = 1, 2, \dots, m$$

اگر A^+ ایده‌آل مثبت و A^- ایده‌آل منفی باشد، ملاحظه می‌شود که چنانچه $A_i = A^+$ گردد، آنگاه

$d_{i+} = 0$ بوده و خواهیم داشت: $cL_i = 1$ و در صورتی که $A_i = A^-$ شود، آنگاه $d_{i-} = 0$ بوده و $cL_i = 0$

خواهد شد. بنابراین هر اندازه گزینه A_i به ایده‌آل مثبت (A^+) نزدیک‌تر باشد، ارزش cL_i به واحد نزدیک‌تر خواهد بود.

قدم ششم - رتبه‌بندی گزینه‌ها

بر اساس ترتیب نزولی cL_i می‌توان گزینه‌های موجود از مسئله مفروض را رتبه‌بندی کرد.

۵-۶-۴- مراحل اجرای روش تاپسیس جهت تعیین محل احداث فرودگاه در استان سمنان

۵-۶-۴-۱- انتخاب شاخص‌ها و تعیین ایده‌آل‌ها

ایده‌آل مثبت (+): هرچه مقدار شاخص افزایش یابد، میزان مطلوبیت شهر جهت احداث فرودگاه افزایش خواهد یافت.

ایده آل منفی (-): هرچه مقدار شاخص افزایش یابد، میزان مطلوبیت شهر جهت احداث فرودگاه کاهش خواهد یافت.

جدول ۵-۱- شاخص‌های منتخب توسط کارشناسان

وزن کارشناسی (λ_j)	نوع شاخص	مقیاس	عنوان شاخص	شاخص
0.18	مثبت	نفر	جمعیت	X_1
0.08	منفی	روز	تعداد روزهای یخبندان در سال	X_2
0.05	مثبت	عدد	تعداد مرکز هواشناسی سینوپتیک	X_3
0.33	مثبت	کیلومتر	فاصله از نزدیک‌ترین فرودگاه	X_4
0.05	مثبت	دو قطبی*	کیفیت راه ارتباطی	X_5
0.08	مثبت	نفر	جمعیت شاغلان غیر بومی غیر ساکن	X_6
0.06	منفی	میلیمتر	متوسط بارندگی طی یک ماه	X_7
0.08	مثبت	ریال	متوسط درآمد خانوار در سال	X_8
0.07	مثبت	نفر	متوسط تعداد جهانگردان و زائران به خارج از کشور در سال	X_9
0.02	منفی	متر در ثانیه	میانگین سرعت وزش باد	X_{10}

* مقیاس دو قطبی عبارت است از تخصیص عددی بین ۰ تا ۱۰ که عدد بیشتر نشانه مطلوبیت بیشتر است

جدول ۵-۲- ماتریس داده ها

شاخص/شهر	X ₁	X ₂	X ₃	X ₄	X ₅	X ₆	X ₇	X ₈	X ₉	X ₁₀
بیارجمند	۲۵۰۴	۸۲	۱	۵۱۸	۴	۸۹	۱۳,۸	۱۴۴۴۴۲۴۰	۱۲۹	۱۲,۵
دامغان	۵۹۳۰۰	۶۳	۱	۳۲۰	۷	۲۵۱۴	۹,۱	۲۰۶۹۷۳۴۲	۱۸۲۶	۲۱,۳
سمنان	۱۲۶۷۸۰	۵۵	۱	۲۱۶	۸	۷۲۸۴	۱۴,۷	۲۵۳۹۸۷۳۲	۳۵۴۱	۱۳,۸
شاهرود	۱۳۲۳۷۹	۶۵	۱	۴۰۰	۷	۶۵۲۳	۱۴	۲۴۳۶۹۷۸۵	۵۴۹۷	۱۴,۶
گرمسار	۳۹۵۲۳	۴۵	۱	۱۰۰	۷	۳۰۲۱	۱۱,۲	۱۸۳۹۸۵۴۱	۱۱۳۰	۱۶,۸
مهدیشهر	۲۱۰۰۶	۷۶	۰	۲۳۳	۶	۱۶۴	۱۵,۲	۱۶۹۸۲۵۹۸	۵۶۳	۱۳,۵
جمع	۳۸۱۴۹۲	۳۸۶	۵	۱۷۸۷	۳۹	۱۹۵۹۵	۷۸	۱۲۰۲۹۱۲۳۸	۱۲۶۸۶	۹۲,۵

۵-۶-۴-۲- تعیین وزن شاخص ها به روش آنتروپی

دلیل استفاده از این روش این است که هیچ کدام از شاخص ها را حذف نمی نماید و برای هر یک از شاخص ها براساس میزان پراکندگی وزن داده می شود. ضمناً می توان از وزن های کارشناسی (قضاوت ذهنی λ_j) نیز به عنوان اهمیت نسبی برای شاخص استفاده نمود و وزن محاسبه شده به این روش را تعدیل نمود. در این تحقیق براساس نظرات کارشناسان گروه های مختلف از جمله گروه های اقتصادی، آمایش سرزمین، متخصصین امور فرودگاه ها و همچنین کارشناسان گروه نقشه و GIS استفاده شده است. ماتریس داده های شاخص های انتخابی بر اساس اطلاعات سالنامه ی آماری سال ۱۳۸۶ و همچنین سرشماری نفوس و مسکن سال ۱۳۸۵ که جدیدترین سرشماری کشور محسوب می شود، تشکیل شده است.

انجام این روش احتیاجی به نرم افزار خاصی ندارد و مثلاً این مطالعه بوسیله نرم افزار **EXCEL** انجام

شده است که از مجموعه نرم افزارهای استاندارد **OFFICE** می باشد که بر روی همه ی کامپیوترها

وجود دارد.

جدول ۵-۳- ماتریس نسبت داده ها

شاخص/شهر	X_1	X_2	X_3	X_4	X_5	X_6	X_7	X_8	X_9	X_{10}
بیارجمند	۰,۰۰۶۶	۰,۲۱۲۴	۰,۲۰۰۰	۰,۲۸۹۹	۰,۱۰۲۶	۰,۰۰۴۵	۰,۱۷۶۹	۰,۱۲۰۱	۰,۰۱۰۲	۰,۱۳۵۱
دامغان	۰,۱۵۵۴	۰,۱۶۳۲	۰,۲۰۰۰	۰,۱۷۹۱	۰,۱۷۹۵	۰,۱۲۸۳	۰,۱۱۶۷	۰,۱۷۲۱	۰,۱۴۳۹	۰,۲۳۰۳
سمنان	۰,۳۳۲۳	۰,۱۴۲۵	۰,۲۰۰۰	۰,۱۲۰۹	۰,۲۰۵۱	۰,۳۷۱۷	۰,۱۸۸۵	۰,۲۱۱۱	۰,۲۷۹۱	۰,۱۴۹۲
شاهرود	۰,۳۴۷۰	۰,۱۶۸۴	۰,۲۰۰۰	۰,۲۲۳۸	۰,۱۷۹۵	۰,۳۳۲۹	۰,۱۷۹۵	۰,۲۰۲۶	۰,۴۳۳۳	۰,۱۵۷۸
گرمسار	۰,۱۰۳۶	۰,۱۱۶۶	۰,۲۰۰۰	۰,۰۵۶۰	۰,۱۷۹۵	۰,۱۵۴۲	۰,۱۴۳۶	۰,۱۵۲۹	۰,۰۸۹۱	۰,۱۸۱۶
مهدیشهر	۰,۰۵۵۱	۰,۱۹۶۹	*۰,۰۰۰۰	۰,۱۳۰۴	۰,۱۵۳۸	۰,۰۰۸۴	۰,۱۹۴۹	۰,۱۴۱۲	۰,۰۴۴۴	۰,۱۴۵۹

* در سومین شاخص از شهر مهدیشهر به دلیل آنکه ln صفر مقدار بینهایت دارد بنابر این به جای صفر از یک عدد نزدیک به صفر استفاده می نماییم.

جدول ۵-۴- بدست آوردن مقدار E_j

شاخص/شهر	X_1	X_2	X_3	X_4	X_5	X_6	X_7	X_8	X_9	X_{10}
بیارجمند	-۰,۰۳۳۰	-۰,۳۲۹۱	-۰,۳۲۱۹	-۰,۳۵۹۰	-۰,۲۳۳۶	-۰,۰۲۴۵	-۰,۳۰۶۴	-۰,۲۵۴۵	-۰,۰۴۶۷	-۰,۲۷۰۵
دامغان	-۰,۲۸۹۴	-۰,۲۹۵۹	-۰,۳۲۱۹	-۰,۳۰۸۰	-۰,۳۰۸۳	-۰,۲۶۲۴	-۰,۲۵۰۷	-۰,۳۰۲۸	-۰,۲۷۹۰	-۰,۳۳۸۲
سمنان	-۰,۳۶۶۱	-۰,۲۷۷۶	-۰,۳۲۱۹	-۰,۲۵۵۴	-۰,۳۲۴۹	-۰,۳۶۷۹	-۰,۳۱۴۵	-۰,۳۲۸۴	-۰,۳۵۶۲	-۰,۲۸۳۸
شاهرود	-۰,۳۶۷۳	-۰,۳۰۰۰	-۰,۳۲۱۹	-۰,۳۳۵۰	-۰,۳۰۸۳	-۰,۳۶۶۲	-۰,۳۰۸۳	-۰,۳۲۳۴	-۰,۳۶۲۴	-۰,۲۹۱۴
گرمسار	-۰,۲۳۴۹	-۰,۲۵۰۶	-۰,۳۲۱۹	-۰,۱۶۱۳	-۰,۳۰۸۳	-۰,۲۸۸۳	-۰,۲۷۸۷	-۰,۲۸۷۲	-۰,۲۱۵۴	-۰,۳۰۹۸
مهدیشهر	-۰,۱۵۹۶	-۰,۳۲۰۰	۰,۰۰۰۰	-۰,۲۶۵۶	-۰,۲۸۸۰	-۰,۰۴۰۰	-۰,۳۱۸۷	-۰,۲۷۶۴	-۰,۱۳۸۲	-۰,۲۸۰۹
جمع	-۱,۴۵۰۲	-۱,۷۷۳۱	-۱,۶۰۹۴	-۱,۶۸۴۴	-۱,۷۷۱۴	-۱,۳۵۰۳	-۱,۷۷۷۳	-۱,۷۷۲۷	-۱,۳۹۷۹	-۱,۷۷۴۶
E_j	۰,۸۰۹۴	۰,۹۸۹۶	۰,۸۹۸۲	۰,۹۴۰۱	۰,۹۸۸۶	۰,۷۵۳۶	۰,۹۹۱۹	۰,۹۸۹۴	۰,۷۸۰۲	۰,۹۹۰۴

جدول ۵-۵- بدست آوردن مقدار W'_j

شاخص	X_1	X_2	X_3	X_4	X_5	X_6	X_7	X_8	X_9	X_{10}
d_j	۰,۱۹۰۶	۰,۰۱۰۴	۰,۱۰۱۸	۰,۰۵۹۹	۰,۰۱۱۴	۰,۲۴۶۴	۰,۰۰۸۱	۰,۰۱۰۶	۰,۲۱۹۸	۰,۰۰۹۶
$d_j * \lambda_j$	۰,۰۳۴۳	۰,۰۰۰۸	۰,۰۰۵۱	۰,۰۱۹۸	۰,۰۰۰۶	۰,۰۱۹۷	۰,۰۰۰۵	۰,۰۰۰۸	۰,۰۱۵۴	۰,۰۰۰۲
$\sum d_j * \lambda_j$	۰,۰۹۷۲									
W'_j	۰,۳۵۲۹	۰,۰۰۸۶	۰,۰۵۲۳	۰,۲۰۳۵	۰,۰۰۵۹	۰,۲۰۲۸	۰,۰۰۵۰	۰,۰۰۸۷	۰,۱۵۸۳	۰,۰۰۲۰

جدول ۵-۶- تشکیل ماتریس بی مقیاس شده از ماتریس داده ها

شاخص/شهر	X_1	X_2	X_3	X_4	X_5	X_6	X_7	X_8	X_9	X_{10}
بیجار جمند	6270016	6724	1	268324	16	7921	190.44	E+142.08636	16641	156.25
دامغان	3516490000	3969	1	102400	49	6320196	82.81	E+144.2838	3334276	453.69
سمنان	16073168400	3025	1	46656	64	53056656	216.09	E+146.45096	12538681	190.44
شاهرود	17524199641	4225	1	160000	49	42549529	196	E+145.93886	30217009	213.16
گرمسار	1562067529	2025	1	10000	49	9126441	125.44	E+143.38506	1276900	282.24
مهدیشهر	441252036	5776	0	54289	36	26896	231.04	E+142.88409	316969	182.25
جمع	39123447622	25744	5	641669	263	E+081.11	1041.82	E+152.50291	47700476	1478.03
SQRT	197796.48	160.45	2.24	801.04	16.22	10539.81	32.28	50029121.41	6906.55	38.45

جدول ۵-۷- ماتریس بی مقیاس شده N_D

شاخص/شهر	X_1	X_2	X_3	X_4	X_5	X_6	X_7	X_8	X_9	X_{10}
بیجار جمند	۰.۰۱۳	۰.۵۱۱	۰.۴۴۷	۰.۶۴۷	۰.۲۴۷	۰.۰۰۸	۰.۴۲۸	۰.۲۸۹	۰.۰۱۹	۰.۳۲۵
دامغان	۰.۳۰۰	۰.۳۹۳	۰.۴۴۷	۰.۳۹۹	۰.۴۳۲	۰.۲۳۹	۰.۲۸۲	۰.۴۱۴	۰.۲۶۴	۰.۵۵۴
سمنان	۰.۶۴۱	۰.۳۴۳	۰.۴۴۷	۰.۲۷۰	۰.۴۹۳	۰.۶۹۱	۰.۴۵۵	۰.۵۰۸	۰.۵۱۳	۰.۳۵۹
شاهرود	۰.۶۶۹	۰.۴۰۵	۰.۴۴۷	۰.۴۹۹	۰.۴۳۲	۰.۶۱۹	۰.۴۳۴	۰.۴۸۷	۰.۷۹۶	۰.۳۸۰
گرمسار	۰.۲۰۰	۰.۲۸۰	۰.۴۴۷	۰.۱۲۵	۰.۴۳۲	۰.۲۸۷	۰.۳۴۷	۰.۳۶۸	۰.۱۶۴	۰.۴۳۷
مهدیشهر	۰.۱۰۶	۰.۴۷۴	۰.۰۰۰	۰.۲۹۱	۰.۳۷۰	۰.۰۱۶	۰.۴۷۱	۰.۳۳۹	۰.۰۸۲	۰.۳۵۱

جدول ۵-۸- تشکیل ماتریس وزین V

شاخص/شهر	X_1	X_2	X_3	X_4	X_5	X_6	X_7	X_8	X_9	X_{10}
بیارجمند	0.0045	0.0044	0.0234	0.1316	0.0014	0.0017	0.0021	0.0025	0.0030	0.0006
دامغان	0.1058	0.0034	0.0234	0.0813	0.0025	0.0484	0.0014	0.0036	0.0419	0.0011
سمنان	0.2262	0.0029	0.0234	0.0549	0.0029	0.1402	0.0023	0.0044	0.0812	0.0007
شاهرود	0.2362	0.0035	0.0234	0.1016	0.0025	0.1255	0.0022	0.0043	0.1260	0.0008
گرمسار	0.0705	0.0024	0.0234	0.0254	0.0025	0.0581	0.0017	0.0032	0.0259	0.0009
مهدیشهر	0.0375	0.0041	0.0000	0.0592	0.0022	0.0032	0.0023	0.0030	0.0129	0.0007
نوع شاخص	+	-	+	+	+	+	-	+	+	-
گزینه ایده آل مثبت	0.2362	0.0024	0.0234	0.1316	0.0029	0.1402	0.0014	0.0044	0.1260	0.0006
گزینه ایده آل منفی	0.0045	0.0044	0.0000	0.0254	0.0014	0.0017	0.0023	0.0025	0.0030	0.0011

جدول ۵-۹- فاصله از ایده آل مثبت (d_{i+})

شاخص/شهر	X_1	X_2	X_3	X_4	X_5	X_6	X_7	X_8	X_9	X_{10}
بیارجمند	0.053707834	$E-3.913$ 06	0	0	$E-2.08$ 06	0.019166	$E-5.28$ 07	$E-3.66395$ 06	0.015139	0
دامغان	0.017004807	$E-9.26$ 07	0	0.002529	$E-1.3$ 07	0.008424	0	$E-6.74866$ 07	0.00708	$E-2.05$ 07
سمنان	$E-059.98177$	$E-2.858$ 07	0	0.005884	0	0	$E-7.49$ 07	0	0.00201	$E-4.47$ 09
شاهرود	0	$E-1.143$ 06	0	0.000898	$E-1.3$ 07	0.000214	$E-5.74$ 07	$E-3.23259$ 08	0	$E-1.17$ 08
گرمسار	0.027454052	0	0	0.011273	$E-1.3$ 07	0.006728	$E-1.05$ 07	$E-1.49618$ 06	0.010019	$E-4.89$ 08
مهدیشهر	0.039495385	$E-2.747$ 06	0.000548	0.005241	$E-5.21$ 07	0.018768	$E-8.89$ 07	$E-2.16267$ 06	0.01279	$E-2.64$ 09

دنباله جدول ۵-۹

شاخص/شهر	جمع فاصله ها	d_{i+}
بیارجمند	۰,۰۸۸۰۲۳	۰,۲۹۶۶۸۶
دامغان	۰,۰۳۵۰۴	۰,۱۸۷۱۸۹
سمنان	۰,۰۰۷۹۹۵	۰,۰۸۹۴۱۶
شاهرود	۰,۰۰۱۱۱۵	۰,۰۳۳۳۸۶
گرمسار	۰,۰۵۵۴۷۶	۰,۲۳۵۵۳۴
مهدیشهر	۰,۰۷۶۸۴۸	۰,۲۷۷۲۱۵

جدول ۵-۱۰ - فاصله از ایده آل منفی (d_{i-})

شاخص/شهر	X_1	X_2	X_3	X_4	X_5	X_6	X_7	X_8	X_9	X_{10}
بیارجمند	0	0	0.000548	0.011273	0	0	E-4.68 08	0	0	E-2.05 07
دامغان	0.01027122	E-1.032 06	0.000548	0.003123	E-1.17 06	0.002177	E-8.89 07	E-1.19387 06	0.001513	0
سمنان	0.049176889	E-2.084 06	0.000548	0.000868	E-2.08 06	0.019166	E-5.97 09	E-3.66395 06	0.006116	E-1.49 07
شاهرود	0.053707834	E-8.26 07	0.000548	0.005807	E-1.17 06	0.015326	E-3.44 08	E-3.00797 06	0.015139	E-1.19 07
گرمسار	0.004363509	E-3.913 06	0.000548	0	E-1.17 06	0.003183	E-3.82 07	E-4.77424 07	0.000526	E-5.35 08
مهدیشهر	0.001089993	E-1.029 07	0	0.001141	E-5.21 07	E-062.08	0	E-1.9673 07	E-059.9	E-1.61 07

دنباله جدول ۵-۱۰

شاخص/شهر	جمع فاصله ها	d_{i-}
بیارجمند	۰,۰۱۱۸۲۱	۰,۱۰۸۷۲۵
دامغان	۰,۰۱۷۶۳۶	۰,۱۳۲۸۰۱
سمنان	۰,۰۷۵۸۸۳	۰,۲۷۵۴۶۸
شاهرود	۰,۰۹۰۵۳۲	۰,۳۰۰۸۸۶
گرمسار	۰,۰۰۸۶۲۶	۰,۰۹۲۸۷۹
مهدیشهر	۰,۰۰۲۳۳۳	۰,۰۴۸۳۰۴

جدول ۵-۱۱- وزن شهرها

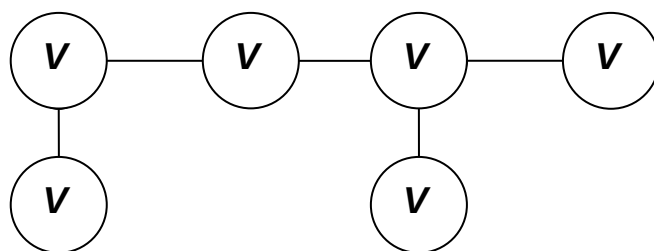
شهر	$d_{i-} + d_{i+}$	cl_i	$w_i = cl_i$
بیجارجمند	۰,۴۰۵۴۱۰۶۵۷	۰,۲۶۸۱۸۴۳	۰,۲۷
دامغان	۰,۳۱۹۹۹۰۶۱۲	۰,۴۱۵۰۱۶۶	۰,۴۲
سمنان	۰,۳۶۴۸۸۴۴۷۴	۰,۷۵۴۹۴۶۷	۰,۷۵
شاهرود	۰,۳۳۴۲۷۲۱۱۸	۰,۹۰۰۱۲۲۳	۰,۹۰
گرمسار	۰,۳۲۸۴۱۲۵۴	۰,۲۸۲۸۱۱۶	۰,۲۸
مهدیشهر	۰,۳۲۵۵۱۹۰۲۸	۰,۱۴۸۳۹۰۹	۰,۱۵

۵-۶-۳- مکان‌یابی مرکز درخت رأس- وزین استان سمنان

در مکان‌یابی مرکز مطلوب یک شبکه یا درخت مشخص شدن وزن رئوس امری لازم و ضروری می‌باشد و آنچه که امروز کاربرد عملی مسائل p - مرکز را با مشکل مواجه نموده است همان نداشتن وزن رئوس شبکه می‌باشد. که ما در این تحقیق این مشکل را با روش آنترופی و تاپسیس که از مفاهیم جدید علم مدیریت می‌باشند حل نمودیم. همانطور که در قسمت‌های قبلی دیدیم با استفاده از روش آنترופی و تاپسیس وزن‌دهی شهرهای منتخب استان به طریقی کاملاً معقول و با در نظر گرفتن اهمیت و تاثیر شاخص‌ها بر هدف مطلوب، صورت گرفت. اکنون با استفاده از الگوریتم‌های ۱-۲-۲-۲ و ۱-۲-۳-۲-۲-۶ و وزن‌هایی که در قسمت قبل برای رئوس بدست آوردیم محل احداث فرودگاه در استان سمنان را تعیین خواهیم نمود.

درخت اولیه‌ی استان سمنان به صورت زیر خواهد بود:

با فرض اینکه V_1 = بیارجمند V_2 = شاهرود V_3 = دامغان V_4 = سمنان V_5 = مهدیشهر V_6 = گرمسار



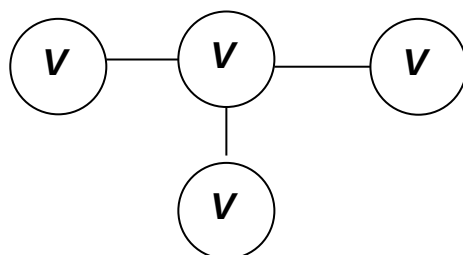
با بکار بستن الگوریتم ۲-۳-۳-۵ در تکرار اول مرکز ثقل درخت استان شهرستان دامغان خواهد شد.

بیشینه فاصله‌ی وزین از شهر دامغان را پیدا می‌کنیم

$$w(\tilde{V}).d(\tilde{V}, V_3) = \max_{V' \in V} \{w(V')d(V', V_3)\} = \max\{5940, 4968, 8700, 2025, 6552\} = w(V_4)d(V_4, V_3)$$

که شهر سمنان می‌باشد. اکنون دامغان را از درخت اولیه حذف کرده زیر درخت حاصل از حذف که

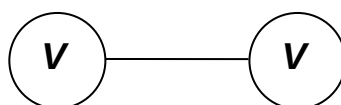
شامل سمنان می‌باشد شامل جواب هم خواهد بود. درخت جدید به صورت زیر شکل می‌گیرد



مرکز ثقل درخت این بار سمنان خواهد بود و بیشینه فاصله‌ی وزین از آن در شهر دامغان رخ خواهد

داد لذا در نهایت به یال زیر می‌رسیم و می‌بایست مرکز نسبی درخت استان را روی این یال محاسبه

نماییم



$e = (\text{سمنان و دامغان})$

$$D_e(V_1, t) = 27(184 + t) = 27t + 4968$$

$$D_e(V_2, t) = 90(66 + t) = 90t + 5940$$

$$D_e(V_3, t) = 42t = 42t$$

$$D_e(V_4, t) = (116 - t) \times 75 = -75t + 8700$$

$$D_e(V_5, t) = 15(135 - t) = -15t + 2025$$

$$D_e(V_6, t) = 28(234 - t) = -28t + 6552$$

حال باید محل تلاقی ها را بیابیم یعنی

$$D_e(v_i, t) = D_e(v_j, t) \quad 1 \leq i < j \leq 6$$

$$27t + 4968 = 90t + 5940 \Rightarrow 63t = -972 \Rightarrow t = -15.4$$

$$27t + 4968 = 42t \Rightarrow 15t = 4968 \Rightarrow t = 331.2$$

$$27t + 4968 = -75t + 8700 \Rightarrow 102t = 3732 \Rightarrow t = 36.5$$

$$27t + 4968 = -15t + 2025 \Rightarrow 42t = -2943 \Rightarrow t = -70$$

$$27t + 4968 = -28t + 6552 \Rightarrow 55t = 1584 \Rightarrow t = 28.8$$

$$90t + 5940 = 42t \Rightarrow 48t = -5940 \Rightarrow t = -123.7$$

$$90t + 5940 = -75t + 8700 \Rightarrow 165t = 2760 \Rightarrow t = 16.7$$

$$90t + 5940 = -15t + 2025 \Rightarrow 105t = -3915 \Rightarrow t = -37.2$$

$$90t + 5940 = -28t + 6552 \Rightarrow 118t = 612 \Rightarrow t = 5.1$$

$$42t = -75t + 8700 \Rightarrow 117t = 8700 \Rightarrow t = 74.3$$

$$42t = -15t + 2025 \Rightarrow 57t = 2025 \Rightarrow t = 35.5$$

$$42t = -28t + 6552 \Rightarrow 70t = 6552 \Rightarrow t = 93.6$$

$$-75t + 8700 = -15t + 2025 \Rightarrow 60t = 6675 \Rightarrow t = 111.25$$

$$-75t + 8700 = -28t + 6552 \Rightarrow 47t = 2148 \Rightarrow t = 45.7$$

$$-1t + 2025 = -28t + 6552 \Rightarrow 13t = 4527 \Rightarrow t = 348.2$$

با توجه به اینکه $0 \leq t \leq 116$ می باشد لذا فقط نقاط کاندید زیر را بررسی می نماییم:

$$k = \{36.5, 28.8, 16.7, 5.1, 74.3, 35.5, 93.6, 111.25, 45.7\}$$

همانطور که داشتیم

$$D_e(t) = \max\{27t + 4968, 90t + 5940, 42t, -75t + 8700, -15t + 2025, -28t + 6552\}$$

طول نقطه‌ی مرکز درخت را به ما می‌دهد $\min_{t \in k} D_e(t) \Rightarrow$

جدول ۵-12- نتیجه‌ی نهایی مدل ترکیبی تاپسیس و مرکز

k	$27t+4968$	$90t+5940$	$42t$	$-75t+8700$	$-15t+2025$	$-28t+6552$	max
۵,۱	۵۱۰۵,۷	۶۳۹۹	۲۱۴,۲	۸۳۱۷,۵	۱۹۴۸,۵	۶۴۰۹,۲	۸۳۱۷,۵
16.7	۵۴۱۸,۹	۷۴۴۳	۷۰۱,۴	۷۴۴۷,۵	۱۷۷۴,۵	۶۰۸۴,۴	۷۴۴۷,۵
28.8	۵۷۴۵,۶	۸۵۳۲	۱۲۰۹,۶	۶۵۴۰	۱۵۹۳	۵۷۴۵,۶	۸۵۳۲
35.5	۵۹۲۶,۵	۹۱۳۵	۱۴۹۱	۶۰۳۷,۵	۱۴۹۲,۵	۵۵۵۸	۹۱۳۵
36.5	۵۹۵۳,۵	۹۲۲۵	۱۵۳۳	۵۹۶۲,۵	۱۴۷۷,۵	۵۵۳۰	۹۲۲۵
45.7	۶۲۰۱,۹	۱۰۰۵۳	۱۹۱۹,۴	۵۲۷۲,۵	۱۳۳۹,۵	۵۲۷۲,۴	۱۰۰۵۳
74.3	۶۹۷۴,۱	۱۲۶۲۷	۳۱۲۰,۶	۳۱۲۷,۵	۹۱۰,۵	۴۴۷۱,۶	۱۲۶۲۷
93.6	۷۴۹۵,۲	۱۴۳۶۴	۳۹۳۱,۲	۱۶۸۰	۶۲۱	۳۹۳۱,۲	۱۴۳۶۴
111.25	۷۹۷۱,۷۵	۱۵۹۵۲,۵	۴۶۷۲,۵	۳۵۶,۲۵	۳۵۶,۲۵	۳۴۳۷	۱۵۹۵۲,۵
min							7447.5

بنابراین $t = 16.7$ طول مرکز درخت استان می باشد. پس فرودگاه باید در ۱۶,۷ کیلومتری شهر دامغان (از سمت غرب) یعنی حوالی شهر امیریه احداث گردد. در حالیکه اگر فقط به روش تاپسیس اکتفا می- کردیم به اشتباه شاهرود به عنوان بهترین محل جهت احداث فرودگاه استان سمنان تعیین می گردید.

- [1] Balinski M. L. (1965) "Integer programming: methods, uses, computation" *Management Science*, 12, pp 253 -313.
- [2] Burkard R. and Dollani H. (2002) "Center problems with pos/neg weights on trees" *European Journal of Operational Research*, 145, pp 483-495
- [3] Burkard R. E. and Dollani H. and Lin Y. and Rote G. (2001) "The obnoxious center problem on a tree" *SIAM J. Discrete Math*, 14 , pp 498-509.
- [4] Campbell J. F. (1990) "Locating transportation terminals to serve an expanding demand" *Transportation Research*, 24B, pp 173 -192.
- [5] Cappanera P. (1999) "A survey on obnoxious facility location problems" Technical Report: Tr-99-11, Dipartimento di informatica, Università di Pisa.
- [6] Church R. L. and ReVelle C. S. (1974) "The maximal covering location problem" *Papers of the Regional Science Association*, 32, pp 101-118.
- [7] Cockayne E. J. Goodman S. E. Hedetniemi S. T. (1975) "A linear algorithm for the domination number of a tree" *Information Processing Letters*, 4, pp 41-44.
- [8] Current J. and Daskin M. and Schilling D. (1992) "Discrete networks location models" Chapter 3 in: *Facility Location Theory and Methods*.
- [9] Daskin M. S. (1982) "Application of an expected covering model to emergency medical service system design" *Decision Sciences*, 13, pp 416-439.
- [10] Drezner Z. and Hamacher H. (2001). Chicago University Press, Chicago, Illinois.
- [11] Drezner Z. and Wesolowsky G. O. (1991) "Facility location when demand is time dependent" *Naval Research Logistics*, 38, pp 763 -777.
- [12] Francis R. and McGinnis Jr. L. F. and White J. A. (1992) "Facility Layout and Location" An Analytical Approach. Prentice Hall.
- [13] Frank H. (1966) "Optimum locations on a graph with probabilistic demands" *Operations Research*, 14, pp 409-421.
- [14] Goldman A. J. (1971) "Optimal center location in simple networks" *Transportation Science*, 5, pp 212-221.
- [15] Hakimi S. (1964) "Optimum location of switching centers and the absolute centers and medians of a graph" *Operations Research*, 12, pp 450 -459.
- [16] Hakimi S. (1965) "Optimum location of switching centers in a communications network and some related graph theoretic problems" *Operations Research*, 13, pp 462-475.
- [17] Hakimi S. L. and Schmeichel E.F. and Pierce J. G. (1978) "On p-centers in networks" *Ibid*, 12, pp 1-15.
- [18] Hamacher H. W. and Nickel S. (1998) "Classification of location models" *Location Science*, 6, pp 229-242.
- [19] Handler G. Y. and Mirchandani P. B. (1979) "Location on networks: theory and algorithms" MIT-press, Cambridge.
- [20] Handler G.Y. (1973) "Minimax location of a facility in an undirected tree graph" *Ibid*, 7, pp 287-293.
- [21] Hansen P. and Labbe M. and Peters D. (1987) "Single facility location on networks" *Annals of Discrete Math*, 3, pp 113-146.

- [22] Kariv O. Hakimi S. L. (1979) "An algorithmic approach to network location problems. Part I: The p-center" *SIAM J. Appl. Math*, 37, pp 513-537
- [23] Kelly M. E. (1986a) "Activity levels at hub facilities in interacting networks" *Geographical Analysis*, 18, pp 343-356.
- [24] Kelly M. E. (1986b) "The location of interacting hub facilities" *Transportation Science*, 20, pp 92- 106.
- [25] Krarup J. and Pruzan P. M. (1983) "The simple plant location problem: survey and synthesis" *European Journal of Operational Research*, 12, pp 36-81.
- [26] Kuby M. (1987) "The p-dispersion and maximum dispersion problems" *Geographical Analysis*, 19, pp315 -329.
- [27] Meggido N. (1983) "Linear time algorithms for linear programming in R^3 and related problems" *SIAM Journal of Discrete Computing*, 12, pp 751-758.
- [28] Miller D. and Starr M. (2000) "The structure of human decisions", *prentice Hall*.
- [29] Mirchandani P. B. (1980) "Locational decisions on stochastic networks" *Geographical Analysis*, 12, pp172 -183.
- [30] Mossie J. (2001) "Essentials of management" *prentice Hall*.
- [31] Newman W. and McGill A. (1987) "The process of Management" *prentice Hall*.
- [32] Owen S. H. and Daskin M. S. (1998) "Strategic facility location: a review" *European Journal of Operational Research*, 111, pp 423-477.
- [33] Scaparra M. P. and Scutella M. G. (2001) "*Facilities, locations, customers: building blocks of location models: A survey*" Thechnical Report: tr-01-18, University of pizza, Italy.
- [34] Slater P. J. (1976) "R-domination in graphs" *J. A. Comput. Mach*, 23, pp 446-450.
- [35] Tamir A. (1988) "Obnoxious facility location on graphs" *SIAM Journal of Discrete Mathematics*, 4, pp550-567
- [36] Tansel B. C. and Francis R. L. and Lowe T. J. (1983) "Location on networks a survey" *Management Science*, 29, pp482-511.
- [37] Toregas C. and Swain R. and Reville C. and Bergman L. (1971) "The location of emergency service facilities" *Operations Research*, 19, pp1363-1373.
- [38] Weber A. (1929) "*Uber den standort-der industrient. Tübingen*" English Trans Theory of Location of Industries, Chicago University press, Chicago, Illinois.

منابع فارسی

[۳۹] اصغرپور م، (۱۳۸۵) "تصمیم‌گیریهای چند معیاره" چاپ چهارم، انتشارات دانشگاه تهران، ۲۳۰-۲۵۰.

[۴۰] عبدالله زاده ر، (۱۳۸۶) "سالنامه‌ی آماری استان سمنان" معاونت برنامه ریزی استانداری سمنان، دفتر آمار و اطلاعات.

[۴۱] فتحعلی ج، (۱۳۸۳)، رساله دکترا: "مسئله P - میانه با وزنهای مثبت و منفی" دانشکده علوم ریاضی و آمار، دانشگاه فردوسی مشهد.

[۴۲] فلسفی م، (۱۳۸۶)، پایان نامه کارشناسی ارشد: "بهینه سازی زمان سفر مورد انتظار با استفاده از اطلاعات بهنگام در شبکه های تصادفی وابسته به زمان" دانشکده ریاضی، دانشگاه صنعتی امیرکبیر.

[۴۳] مهرگان م، (۱۳۷۹) "پژوهش عملیاتی: برنامه‌ریزی خطی و کاربردهای آن" جلد اول، چاپ دوازدهم، انتشارات سالکان

[۴۴] هورویتز آ، (۱۳۷۹) "مفاهیم بنیادی ساختمان داده‌ها در $C++$ " جعفرنژاد قمی ع، جلد اول، چاپ سوم، انتشارات شایگان

Abstract

We discussed about Problems of finding p -centers and dominating sets of radius r in networks. Let n be the number of vertices and m be the number of edges of a network. With the assumption that the distance-matrix of the network is available, we analyze an $O(m.n.\log n)$ algorithm for finding an absolute center of a vertex weighted network and $O(m.n + n^2.\log n)$ algorithm for finding absolute center of a vertex-unweighted network. We describe an algorithm of complexity $O((m^p.n^{2p-1}/(p-1)!) \log n)$ (respectively, $O(m^p.n^{2p-1}/(p-1)!)$ for finding an absolute p -center in a vertex-weighted (respectively, vertex-unweighted) network. We discussed about the problems of finding p -centers and dominating sets of networks whose underlying graphs are trees. And then we talked about obnoxious center problems. At last section we introduce topsis method and review topsis algorithm for determining the weights of vertices. At last we combine topsis method with center problem to find the best location for locating the airport of semnan state.

Keywords: center problems, dominating set, topsis method