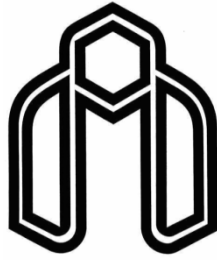


بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ



دانشگاه صنعتی شاهرود

دانشکده ریاضی

گروه ریاضی کاربردی

مکانیابی با جاذبه و دافعه

دانشجو : علی جمالیان

استاد راهنما :

دکتر جعفر فتحعلی

استاد مشاور :

دکتر احمد نزاکتی

پایان نامه جهت اخذ درجه کارشناسی ارشد ریاضی کاربردی

بهمن ماه ۱۳۸۸

تقدیم به پدرم و مادرم

به پاس محبت ها و فداکاری هایشان

تشکر و قدردانی

خداوند یکتا و متعال را شاکرم که توفیق یافتم تا این پایان نامه را با راهنمایی و یاری اساتید و دیگر بزرگواران، تدوین نمایم و به زعم خود از همه این عزیزان تشکر و قدردانی می کنم و برای ایشان از درگاه ایزد مهربان آرزوی سعادت و عاقبت به خیری دارم.

در ابتدا صمیمانه ترین تشکر ها تقدیم به پدر و مادر عزیزم که همواره راهنما و مشوقم بوده اند و گذر از مشکلات زندگی ام به حول و قوه الهی و بدون دعای خیر و برکت وجودشان، غیر ممکن بود.

از استاد راهنمای ارجمندم، **جناب آقای دکتر فتحعلی** که سعه صدر و صبوری مرا راهنمایی نموده و در پیشبرد این پایان نامه سعی تمام مبذول داشتند کمال تشکر و قدردانی می نمایم.

از استاد مشاورگرمی ام، **جناب آقای دکتر نزاکتی** که مرا در انجام این پایان نامه از نظرات و راهنمایی های سازنده، راهگشا و ارزنده شان بهره مند ساختند، کمال سپاسگزاری را دارم.

از اساتید محترم، **جناب آقای دکتر میرحسینی و جناب آقای دکتر امینی خواه** که زحمت داوری این پایان نامه را به عهده داشتند صمیمانه تشکر و قدردانی می نمایم.

بر خود لازم می دانم تا از کلیه اساتید گرانقدر داشکده ریاضی که در دوران تحصیل از محضرشان کسب فیض نمودم، تشکر نمایم.

در نهایت از تمامی دوستان و همکلاسی ها و هم دانشکده ای های گرمی ام که در طول این مدت افتخار مصاحبت و همفکری با آنها را داشتم، صمیمانه سپاسگزاری نمایم.

دانشجو تأیید می نماید که مطالب مندرج در این پایان نامه (رساله) نتیجه تحقیقات خودش می باشد و در صورت استفاده از نتایج دیگران مرجع آن را ذکر نموده است.

کلیه حقوق مادی مترتب از نتایج مطالعات ، آزمایشات و نوآوری ناشی از تحقیق موضوع این پایان نامه (رساله) متعلق به دانشگاه صنعتی شاهرود می باشد .

بهمن ماه ۱۳۸۸

چکیده

دردو سه دهه اخیر، توجه و علاقه مندی به مسائل محیطی و دنیای واقعی کاربردی، افزایش پیدا کرده است. این مسائل عمدتاً در نتیجه پیشرفت های صنعتی و تکنولوژی ایجاد شده اند و غالباً تأثیرات منفی در محیط را نیز به دنبال دارند و مشکلات زیست محیطی به بار می آورند. مسائلی مانند استقرار یک سرویس دهنده، فرآیند یا وسیله که در عین تامین خدمات و تقاضای مشتریان اثراتی مثل سرو صدا، گردوخاک، دود، گاز های سمی، بوی بد، شلوغی و مزاحمت هایی از این نوع را بر مشتریان تحمیل می کنند، از این دسته هستند. طراحان شهری، و در حالت کلی تصمیم گیرندگان در عمل با مسائل پیچیده ای نظیر انتخاب محل استقرار یک فرودگاه، کارخانه ها، نیروگاه ها، آزمایشگاه های شیمیایی، مراکز هسته ای، فروشگاه های زنجیره ای، مراکز اورژانس، بیمارستان ها، اماکن دفن زباله و زباله سوزی، پایانه های حمل و نقل، تصفیه خانه های آب، و غیره روبرو هستند که همگی در حوزه مکانیابی سرویس دهنده نیمه ناخوشایند قابل بررسی و مطالعه هستند. مطالعات مهمی در زمینه مکانیابی سرویس دهنده نیمه نامطلوب انجام شده است.

مدل های مکانیابی نیمه نامطلوب اساساً چند مدی هستند و لذا روش های استاندارد که بر مبنای آنالیز محدب برای مسائل مکانیابی سرویس دهنده مطلوب در صفحه به کار می روند، ممکن است به دام بهینه های موضعی بیافتند. از طرف دیگر به خاطر پیچیدگی هندسی این مسائل ممکن است روش های هندسه محاسباتی که برای حل مسائل مکانیابی سرویس دهنده نامطلوب به کار می روند، برای این مسائل جوابگو نباشند.

با توجه به اهمیت موضوع، مدل های مختلفی که در این زمینه تا کنون ارائه شده اند را مورد مطالعه قرار داده و دیدگاهی نوین جهت مدلسازی مسائل مکانیابی با جاذبه و دافعه ارائه می کنیم. این مسائل را مسائل مکانیابی آرمانی نامیده و برای حل چنین مسائلی در حالات مختلف الگوریتم ها و روش های

مناسب را طراحی یا اجرا کرده ایم. برای مسئله وبر آرمانی یک روش گرادیانی شبه وایزفلد با همگرایی
سراسری بدست آمده و نتایج آن با روشهای دیگر مورد مقایسه قرار گرفته است.

لیست مقالات مستخرج از پایان نامه

[1] A. Jamalian, J. Fathali, (2009), "Linear programming for the location problem with minimum absolute error", *World Journal of Applied Science*, to appear.

[2] J. Fathali, A. Jamalian, "On the Goal Weber Problem", *submitted*.

[3] A. Jamalian, J. Fathali, H. Ahsani Tehrani "Locating Multiple Facilities in Convex Sets With Fuzzy Data and Block Norms", *submitted*.

[۴] علی جمالیان، احمد نزاکتی، جعفر فتحعلی، (۱۳۸۸). "حل مسئله حداقل مربعات خطای مکانیابی تک وسیله ای با الگوریتم PSO". دومین کنفرانس بین المللی تحقیق در عملیات، بابلسر.

[۵] علی جمالیان، شهزاد رادکانی، مصطفی حیدری، جعفر فتحعلی، (۱۳۸۸). "مسئله مکانیابی تک وسیله ای آرمانی با پارامترهای فازی و کاربرد های آن". " دومین همایش علوم منطقه ای و کاربردهای ریاضی در علوم مهندسی، دانشگاه آزاد اسلامی جویبار.

فهرست مندرجات

۱	مقدمه ای بر مسائل مکانیابی	۱
۲	۱-۱ مقدمه.....	۲
۵	۲-۲ اجزا مسائل مکانیابی.....	۵
۵	۱-۲-۱ فضا.....	۵
۷	۲-۲-۱ تعداد سرویس دهنده (یا فعالیت) هایی که باید مستقر شوند (P).....	۷
۸	۳-۲-۱ تعداد سرویس دهندگان موجود.....	۸
۸	۴-۲-۱ اهداف تصمیم گیرنده.....	۸
۹	۵-۲-۱ اندازه گیری فاصله.....	۹
۱۱	۶-۲-۱ محدودیت های فاصله.....	۱۱
۱۱	۷-۲-۱ وزن.....	۱۱
۱۲	۸-۲-۱ ارتباط بین سرویس دهندگان.....	۱۲
۱۲	۹-۲-۱ مشتریان.....	۱۲
۱۲	۱۰-۲-۱ تابع هدف.....	۱۲
۱۴	۳-۱ تابع هدف های جاذب.....	۱۴

۴-۱ تابع هدف های دافع ۱۶

۱-۴-۱ مسائل مکانیابی سرویس دهندگان نامطلوب یا مضر ۱۸

۲-۴-۱ مدل های پراکندگی ۱۸

۵-۱ تابع هدفهای پرکاربرد مکانیابی ۱۹

۲ الگوریتمهای شاخه و کران برای مسائل مکانیابی پیوسته ۲۳

۱-۲ مقدمه ۲۴

۲-۲ روش مستطیل بزرگ مستطیل کوچک ۲۵

۱-۲-۲ الگوریتم مستطیل بزرگ مستطیل کوچک ۲۵

۳-۲ روش مثلث بزرگ مثلث کوچک ۲۷

۱-۳-۲ گراف ورونوی ۲۸

۲-۳-۲ مثلث بندی دلانی ۳۵

۳-۳-۲ الگوریتم مثلث بزرگ مثلث کوچک ۳۵

۴-۳-۲ محاسبه کران پایین در یک مثلث ۳۷

۳ الگوریتم بهینه سازی پرندگان ۴۱

۱-۳ مقدمه ۴۲

۲-۳ الگوریتم بهینه سازی دسته ذرات ۴۳

۴۸	شبه کد الگوریتم PSO	۱-۲-۳
۵۰	نسخه های اصلاح شده الگوریتم PSO	۳-۳
۵۱	الگوریتم PSO استاندارد (PSO-S)	۱-۳-۳
۵۳	الگوریتم PSO با عامل انقباض	۲-۳-۳
۵۳	الگوریتم ترکیبی PSO و فرآیند شبیه سازی تبریدی	۳-۳-۳
۵۶	شبه کد الگوریتم ترکیبی PSO و فرآیند شبیه سازی تبریدی	۴-۳-۳
۵۷	مکانیابی سرویس دهنده نیمه ناخوشایند	۴
۵۸	مقدمه	۱-۴
۶۵	مدل های مکانیابی گسسته با جاذبه-دافعه	۲-۴
۶۵	مسئله مکانیابی سرویس دهنده با ظرفیت نامحدود و اثرات منفی جمعی	۱-۲-۴
۶۷	مسئله مکانیابی سرویس دهنده با ظرفیت نامحدود و کمترین پوشش اثرات منفی	۲-۲-۴
۶۹	مدل کمترین مجموع-کمترین مجموع	۳-۲-۴
۶۹	مدل کمترین مجموع-ماکزیمین	۴-۲-۴
۷۰	مدل های شبه پراکنده	۳-۴
۷۰	مسئله p -دفاع-مجموع	۱-۳-۴
۷۱	مسئله p -پراکنده	۲-۳-۴

۷۴	مسائل قابل حل در زمان چند جمله ای	۴-۴
۷۵	مدل های مکانیابی پیوسته با جاذبه و دافعه	۵-۴
۷۵	مدل کمینه کردن مجموع فاصله ها	۱-۵-۴
۸۵	مکانیابی چند وسیله ای در مجموعه های محدب با نرم بلوکی	۶-۴
۸۶	مسئله مکانیابی چندوسیله ای در مجموعه های محدب	۱-۶-۴
۹۰	مسئله وبر با جاذبه و دافعه	۵
۹۱	مقدمه	۱-۵
۹۴	مسئله وبر با جاذبه و دافعه و نرم مستطیلی	۲-۵
۹۶	مسئله وبر با جاذبه و دافعه و مجذور نرم اقلیدسی	۳-۵
۹۷	مسئله وبر با جاذبه و دافعه و نرم اقلیدسی	۴-۵
۱۰۲	قضیه موضعی سازی	۵-۵
۱۰۴	مسئله مکانیابی آرمانی	۶
۱۰۵	مقدمه	۱-۶
۱۰۷	خواص مسئله	۲-۶
۱۱۳	مسئله مکانیابی تک وسیله ای با کمترین خطای مطلق	۳-۶
۷۶	نتیجه گیری و پیشنهادات	
۷۹	مراجع	

پیوست ۱ متن برنامه ها

۸۳

واژه نامه

۸۶

فهرست اشکال

- شکل ۱-۱: مدل لانهارد با دو منبع و یک بازار..... ۳
- شکل ۱-۲: گراف ورونوی و دوگان آن..... ۳۰
- شکل ۲-۲: یالهای متناهی و نیمه متناهی گراف ورونوی..... ۳۱
- شکل ۳-۲: گراف همبند و مسطح شده گراف ورونوی با یالهای نیمه متناهی و
راس بینهایت..... ۳۳
- شکل ۱-۳: نمایش چگونگی حرکت ذرات در الگوریتم PSO..... ۴۷
- شکل ۱-۴: تابع تاثیرات منفی بر حسب فاصله..... ۶۴
- شکل ۲-۴: تابع هزینه حمل و نقل بر حسب فاصله..... ۶۴
- شکل ۳-۴: نقاط تقاضا و همسایگی های حول نقاط دارای اثر منفی..... ۸۰
- شکل ۱-۵: تابع هدف با سه نقطه..... ۹۸
- شکل ۱-۶: نمایش گرافیکی مسئله مکانیابی آرمانی..... ۱۰۶

فهرست جداول

جدول ۴-۱: مختصات ماشینها..... ۸۸

جدول ۴-۲: مختصات نواحی برای استقرار ماشینهای جدید..... ۸۹

جدول ۶-۱: نتایج الگوریتم های PSO، WLA استاندارد و نسخه اصلاح شده

آن و نیز الگوریتم BSSS..... ۱۱۷

جدول ۶-۲: زمان اجرای الگوریتم های PSO، WLA استاندارد و نسخه اصلاح

شده آن و نیز الگوریتم BSSS..... ۱۱۸

فصل ۱

مقدمه ای بر مسائل مکانیابی

۱-۱ مقدمه

مسائل مکانیابی مسائلی هستند که وقتی به دنبال یافتن محل و نحوه استقرار بهینه یک یا چند فعالیت معین (سرویس دهنده)^۱ بر اساس عوامل و متغیرهای موثر بر مکانیابی هستیم با آنها روبه رو می شویم. معمولاً در این مسائل فرض بر این است که تعدادی مشتری (سرویس گیرنده، نقطه ثابت، نقطه تقاضا) موجودند و می خواهیم بهترین مکان برای استقرار سرویس دهندگان جدید را به گونه ای بیابیم که هزینه ی حمل و نقل، تاثیرات محیطی نامطلوب، زمان سرویس دهی، سود، کیفیت، نحوه ی توزیع سرویس و رعایت انصاف و برابری در آن، کسب بیشترین سهم در بازار و ... بسته به شرایط مسئله، بهینه شوند.

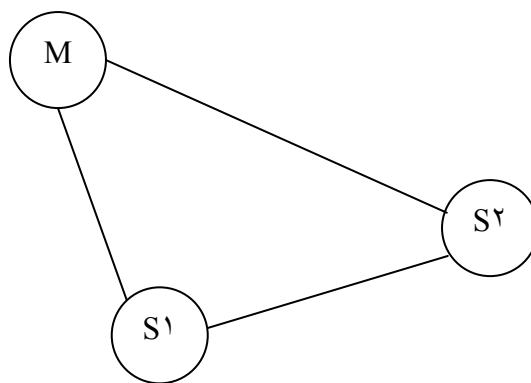
مکانیابی تاسیسات و تسهیلات از کاربردی ترین زمینه های تحقیقاتی است که در بسیاری از شاخه های علوم گسترده شده است. از جمله این شاخه ها می توان به تحقیق در عملیات، علوم مدیریت، مهندسی صنایع، جغرافیا، اقتصاد، علوم کامپیوتر، ریاضیات، بازاریابی، مهندسی برق، برنامه ریزی شهری، علوم نظامی و دیگر شاخه های مرتبط اشاره کرد. از کاربرد های معروف مسائل مکانیابی می توان مکانیابی انبارها، کارخانه ها، ماشین آلات کارخانه، بیمارستان ها، نمایندگی های فروش، آتش نشانی ها، پاسگاه های پلیس، پایانه های مسافری، اجزا الکترونیکی در مدارها، آژیر خطر، چاه های نفت، تاسیسات نظامی، نیروگاه ها، آزمایشگاه ها و غیره را نام برد. مسائل ترکیبی از مسیریابی و مکانیابی نیز در این حوزه مورد مطالعه و بررسی قرار می گیرند.

زمان پیدایش مسائل مکانیابی به اوایل قرن هفدهم و مسئله ای که فرما^۲ مطرح کرد بر می گردد. مسئله ای که فرما مطرح کرد به این صورت است: "یک مثلث را در نظر بگیرید که وزن هر یک از رئوس آن ۱+ باشد. نقطه چهارم را به گونه ای پیدا کنید که مجموع فواصل وزنی آن تا رئوس مثلث

۱ Facility

۲ Fermat

کمینه گردد." تورپچلی^۳ مشاهده کرد که محل برخورد دایره های محیطی مثلث های متساوی الاضلاعی که بر روی اضلاع مثلث اصلی و رو به بیرون آن ساخته می شوند، جواب مسئله است. نظریه مکانیابی به شکلی که امروزه مورد استفاده قرار می گیرد، با منتشر شدن کتاب آلفرد وبر^۴ در سال ۱۹۰۹ با عنوان *Vber den standortder indvstrien* متولد شد **Error! Reference source not found.** [۱]. وبر در این کتاب نتایج تحقیقات خود در مورد صنایع کارخانه ای را ارائه کرد. بیست سال قبل از ارائه تئوری مکانیابی صنعتی توسط وبر، لانهارد^۵ به طور جزئی تر به این امر همت گماشته بود. لانهارد کوشش کرد نشان دهد که چگونه می توان مکانیابی بهینه را در شرایط ساده ای که دو منبع مواد اولیه وجود دارند و یک بازار به وسیله رئوس یک مثلث نشان داد (شکل ۱-۱) [۱].



شکل ۱-۱: مدل لانهارد با دو منبع و یک بازار

نظریه وبر دارای سه اصل اساسی است. اول اینکه محل مواد اولیه معلوم است. دوم اینکه مکان و میزان مصرف مشتریان مشخص است. تعداد این مکان ها متناهی است و جدا از هم هستند. هر تولید کننده بازار نامحدودی در اختیار دارد و امتیازات انحصاری از نظر انتخاب مکان وجود ندارد. سوم اینکه چندین مکان ثابت برای عرضه نیروی کار وجود دارد به طوری که این نیروی کار دارای قابلیت تحرک و جابجایی نبوده و عرضه آن نامحدود است.

مطالعات جدی بر روی مکانیابی از زمانی شروع شد که در سال ۱۹۶۴ حکیمی [۴]، تابع هدف را به دو صورت کمترین مجموع و مینیماکس مطرح کرد. او همچنین مطالعاتی بر روی مسائل مکانیابی روی شبکه انجام داد که نقطه شروع مکانیابی گسسته مدرن است. معمولاً در مسائل مکانیابی سرویس دهندگانی که می‌خواهیم مکان استقرار آنها را بیابیم، مطلوب در نظر گرفته می‌شوند و تنها تاثیرات مثبت شان را به حساب می‌آوریم. بنابراین نزدیک بودن به آنها نوعی جاذبه و دوری از آنها دافعه، تلقی می‌شود. در جوامع مدرن و پیشرفته امروزی با فعالیت ها و سرویس دهندگانی رو برو هستیم که علاوه بر ارائه خدمات، تاثیرات منفی و مخربی نیز بر محیط پیرامون خود دارند. کارگاه‌های تولیدی مثال آشکار این مورد هستند. کارگاه کالای مورد نیاز جامعه را تولید کرده و از طرف دیگر آلاینده محیط زیست است. مراکز ورزشی در عین حال که باید در دسترس باشند و امکانات ورزشی و تفریحی مناسب را ارائه دهند، می‌توانند تاثیرات منفی همچون سر و صدا، ازدحام جمعیت و یا مشکلاتی از این قبیل را داشته باشند.

مطالعه مدل‌هایی که هر دو ویژگی را به طور همزمان در خود دارند، انطباق بیشتری با مسائل واقعی و کاربردی دارد و به سرعت در حال پیشرفت است. از دهه ۱۹۶۰ تا اواسط ۱۹۷۰ مدل‌های کمترین مجموع در بخش‌های خصوصی و مدل‌های مینیماکس در بخش خصوصی به کار گرفته می‌شد. از اواسط ۱۹۷۰ این موضوع تغییر کرد. گلدمن و دیرینگ [۵]، چرچ و گارفینکل [۶] اولین کسانی بودند که مواردی را پیشنهاد کردند که بعد ها مدل‌های مکانیابی مضر^۱ نام گرفتند. در این دسته مدل‌ها، مشتریان فرض مطلوب بودن سرویس دهنده (فعالیت) ها را کنار می‌گذارند و دیگر سعی در نزدیکتر بودن به آنها را نداشته و تا حدی از آنها دوری می‌کنند. به عنوان کاربردی از این مدل‌ها می‌توان به مکانیابی راکتورهای هسته‌ای، مراکز تصفیه آب و اماکن دفن زباله اشاره کرد. در حقیقت ویژگی‌های این نوع سرویس دهندگان نوعی دافعه ایجاد می‌کند و سبب می‌شود که مشتریان آنها را از خود دور کنند.

اولین طبقه بندی مدل های مکانیابی توسط هندلر و میرچندانی [۷] ارائه شد. پس از آن طبقه بندی های دیگری توسط هاماکر و نیکل [۸] و نیز ارکوت و نیومن [۹] انجام شده است. در ادامه فصل مجموعه ای از کارهای انجام شده در این زمینه را ارائه می دهیم.

۲-۱ اجزا مسائل مکانیابی

در این بخش ترکیبی از طبقه بندی هایی که از مسائل مکانیابی و تعیین ویژگی های آنها و اجزایشان صورت گرفته است، ارائه می شود.

۱-۲-۱ فضا^۷

معمولا فضای مسائل مکانیابی یکی از سه دسته زیر است :

(۱) پیوسته : هرگاه فضای مکان تاسیسات و نقاط تقاضا پیوسته باشند، یعنی توسط متغیر هایی که به صورت پیوسته تغییر می کنند، نظیر مختصات مشخص شود، مدل را پیوسته می گویند. در این مدل ها مکان سرویس دهنده را در فضای d بعدی R^d پیدا می کنیم. اکثر مسائل مکانیابی در فضای حداقل دو بعدی تعریف می شوند. مسائل دو بعدی به خاطر ویژگی های هندسی و قابلیت ادراک بصری شان بیشتر مورد توجه قرار می گیرند. در مواجهه با مسائل پیوسته از ابزار های هندسه و آنالیز (آنالیز تابعی، که معمولا از ویژگی های تحدب و تقعر استفاده می شود) بهره می بریم. عمدتا مسائل مکانیابی طبیعت و توصیف هندسی دارند ولی مطالعه و روند حل آنها هم شامل بررسی های هندسی و هم آنالیزی است.

در کاربرد های عملی از فضای پیوسته به دو شکل استفاده می شود. مسطح، مثل مدارهای مجتمع در مدار های الکتریکی و الکترونیکی، یک تکه کاغذ، میز کار، و یا یک تکه زمین و کروی برای نواحی مثل قاره ها و حتی کره زمین. در این حالت ها برای مشخص کردن نقاط احتیاج به

دو مولفه داریم. در مسائلی که با مکانیابی در ساختمان زیر آب و یا در هوا رو به رو هستیم، باید مشخصه سومی که ارتفاع یا عمق را نشان می دهد را هم به حساب آوریم. مسائل یک بعدی عمدتاً زمانی رخ می دهند که مکانیابی را روی یک خط (مستقیم یا دارای انحنا) انجام می دهیم، مثل کشیدن یک بزرگراه یا خط آهن.

(۲) گسسته : مدل های مکانیابی گسسته مدل هایی هستند که در آنها باید مکان سرویس دهندگان را تنها روی نقاط از پیش تعیین شده پیدا کنیم. در این مدل ها مجموعه ای متناهی از نقاط کاندیدا داده شده است که توسط متغیر های گسسته نشان داده می شوند. در مدل های مکانیابی پیوسته بر خلاف مدل های گسسته نمی توان لیست جامع و کاملی از نقاط در دسترس را ارائه داد. از این جهت مدل های پیوسته را مدل های ایجاد مکان و مدل های گسسته را مدل های انتخاب مکان نیز خوانده اند. یکی از ابزار مفید در ریاضیات در مواجهه با مسائل بزرگ و پیوسته ایده گسسته سازی فضا است. یکی از دلایل استفاده از این ایده ساده تر شدن و کاهش فضای جواب به تعداد متناهی مجموعه و کاربرد های ویژه در بعضی حالات خاص از مسئله است. در مدل های گسسته، در حقیقت هر نقطه نماینده یک ناحیه از فضا است. قابل توجه است که در حالت پیوسته باید یک تابع برای تخمین فاصله بین نقاط انتخاب شود که نحوه انتخاب یک تابع مناسب، خود موضوع مقالات زیادی در زمینه مکانیابی پیوسته بوده است ولی در حالت گسسته، فاصله واقعی پیموده شده و یا از قبل محاسبه شده، مورد استفاده قرار می گیرد. بنابراین مدل های پیوسته به راحتی موقعیت های عملی و واقعی را مدل می کنند و به سرعت اجرا می شوند زیرا به بانک های اطلاعاتی عظیم مسافت ها و همچنین محاسبه کوتاه ترین مسیر احتیاجی ندارند. این نکته قابل ذکر است که به خاطر استفاده از تابع تخمین از دقت کمتری نسبت به انواع گسسته دارند. البته انتخاب نوع فضا به نظر تصمیم گیرنده بستگی دارد. بعضی اوقات ممکن است مزایای مدل های پیوسته سبب شود که از دقت آن تا حدی صرف نظر کرد. مثلاً در یک شهر

ممکن است که انتخاب فاصله مستطیلی^۸ (منهتن) دقت بسیار زیادی در تخمین فاصله واقعی بین دو نقطه از آن را داشته باشد. عمدتاً شهر هایی که دارای خیابان های عمود بر هم و توری شکل هستند در این دسته قرار می گیرند .

(۳) شبکه ای : مدل های شبکه ای مدل هایی هستند که در آن سرویس دهندگان روی رئوس و یا یال های یک شبکه مستقر می شوند. در حوزه مکانیابی با فضای شبکه ای، مسائلی وجود دارند که در حالت کلی NP - سخت هستند ولی در گرافهای خاص مثل درخت ها یا k - درختها در زمان های حداکثر از درجه چند جمله ای قابل حل هستند. علاوه بر موارد ذکر شده ممکن است فضایی که سرویس دهنده باید در آن قرار گیرد با تعریف نواحی ممنوع محدود شود. نواحی ممنوع نواحی هستند که سرویس دهنده نباید در آن قرار گیرد. این مناطق می توانند قوانین منطقه بندی یا سایر محدودیت را مدل کنند.

۲-۲-۱ تعداد سرویس دهنده (یا فعالیت) هایی که باید مستقر شوند (P)

فرض کنید p تعداد سرویس دهنده ها باشد. حالت های $p = 1$ و $p > 1$ را در نظر می گیریم:

(۱) $p = 1$ ؛ در این حالت فقط یک فعالیت (سرویس دهنده) باید مستقر شود .

(۲) $p > 1$ ؛ در این حالت علاوه بر پیدا کردن محل استقرار سرویس دهنده ها باید چگونگی تخصیص مشتریان به سرویس دهندگان را نیز مشخص کنیم .

درحالتی که چند سرویس دهنده را مکانیابی می کنیم، ممکن است ارتباط بین سرویس دهندگان جدید نیز دارای اهمیت باشد. مدل هایی که این هدف را بر آورده می کنند شامل فاصله وزندار بین سرویس دهندگان جدید می باشند. به عنوان مثالی از این مورد می توان به مکانیابی تاسیسات نظامی اشاره کرد که در آن پراکنده شدن و فاصله داشتن سرویس دهندگان جدید میزان ضرر و خسارت و آسیب پذیری را کاهش می دهد. در مکانیابی پمپ های گاز و یا فست

فود ها تصمیم گیرنده سعی در پراکنده کردن آنها دارد تا از ایجاد رقابت مستقیم آنها با یکدیگر جلوگیری کند.

۱-۲-۳ تعداد سرویس دهندگان موجود

بر مبنای این پارامتر دو نوع مدل مختلف وجود دارد:

(۱) مدلهایی که تعداد سرویس دهندگان جدید در آن ثابت است.

(۲) مدلهایی که تعداد سرویس دهندگان جدید در آن متغیر است.

مدل های نوع دوم پیچیده تر از مدل های نوع اول هستند، زیرا باید متغیری دیگر را که نشان دهنده تعداد سرویس دهندگان است را به مدل اضافه کنیم. در مدل های مکانیابی دو حالت رخ می دهد. در بسیاری از مدل ها فرض بر این است که تصمیم گیرنده با یک فضای خالی رو به رو می شود که هیچ سرویس دهنده ای در آن وجود ندارد. در حالتی دیگر که آن را با نام مکانیابی شرطی^۹ می شناسیم، تعدادی سرویس دهنده موجود هستند و هدف اضافه کردن تعدادی دیگر به آنهاست.

۱-۲-۴ اهداف تصمیم گیرنده

مدل می تواند یک یا چند هدفه باشد. یعنی تصمیم گیرنده می تواند به دنبال مکان بهینه برای استقرار سرویس دهنده تحت یک هدف یا چند هدف به طور همزمان باشد. در بین مدلهای تک هدفه حداقل سه دسته وجود دارند :

(۱) مینیم سازی تابعی از فاصله: کمترین مجموع^{۱۰}، مینیماکس^{۱۱}

(۲) ماکزیمم سازی تابعی از فاصله: بیشترین مجموع^{۱۲}، ماکزیمین^{۱۳}

Conditional ۹

Minisum ۱۰

Minimax ۱۱

Maxisum ۱۲

Maximin ۱۳

(۳) تابع های ترکیبی از فاصله

۵-۲-۱ اندازه گیری فاصله

فاصله یک معیار عددی برای تشخیص میزان دور بودن اشیا است. در مسائل مکانیابی فرض می کنیم که ارتباط بین تاسیسات مستقیما وابسته به موقعیت مکانی آنها باشد و معمولا به شکل تابعی از فاصله بیان می شود. بنابراین مفهوم فاصله نقشی محوری در مکانیابی پیوسته دارد. با توجه به موقعیت و شرایط مسئله اندازه های مختلفی برای ارزیابی فاصله به کار می روند. استفاده از روند^{۱۴} نرم ها به عنوان تابع تخمین فاصله در یک مسئله مکانیابی با فضای پیوسته اولین بار توسط لاو و موریس [۱۰] پیشنهاد شد. وارد و وندل [۱۱-۱۲] نیز استفاده از نرم های بلوکی^{۱۵} را مورد مطالعه قرار دادند. برای اندازه گیری فاصله بین مشتریان و سرویس دهندگان معمولا از مترهای زیر استفاده می شود.

(۱) فاصله مینکوفسکی^{۱۶} (یا نرم L_p): این فاصله بین دو نقطه $x = (x_1, x_r), y = (y_1, y_r)$ در صفحه به صورت زیر تعریف می شود:

$$L_p(x, y) = \left(k_1 |x_1 - y_1|^p + k_r |x_r - y_r|^p \right)^{\frac{1}{p}} \quad 1 \leq p \leq \infty \quad (1-1)$$

فاصله مینکوفسکی برای مدل کردن مسائلی که فاصله در جهت های مختلف تقارن ندارد، مناسب است. در مسائل مکانیابی از بین خانواده L_p ها، فاصله ی اقلیدسی با $k_1 = k_r = 1, p = 2$ و فاصله ی مستطیلی (منهتن^{۱۷}) با $k_1 = k_r = 1, p = 1$ بیشتر مورد استفاده هستند. فاصله اقلیدسی برای مدل کردن فاصله های هوایی یا دریایی و فاصله مستطیلی برای مدل کردن فاصله های خیابانی (به شرط اینکه خیابانها یکطرفه نباشند) مناسب هستند.

Round norm ۱۴

Block norms ۱۵

Minkowsky ۱۶

Manhattan ۱۷

(۲) فاصله شبکه ای: برابر با کوتاه ترین مسیر بین دو نقطه روی شبکه است. برای محاسبه این فاصله الگوریتم های مختلفی پیشنهاد شده است که از آن جمله می توان الگوریتم های دایکسترا و فلوید را نام برد.

(۳) نرم بلوکی: نرم های بلوکی نرم هایی هستند که کانتورشان چندضلعی است. نرم های L_1, L_∞ دو مثال از نرم های بلوکی هستند. نرم های بلوکی اولین بار توسط وارد و وندل [۱۱-۱۲] برای حل مسائل مکانیابی به کار برده شد. این نرمها برای مدل کردن فاصله در مسائلی که محدود به حرکت در جهت های از پیش تعیین شده هستیم، مناسب اند. همچنین کاربرد های زیادی در مسائل مکانیابی با مانع دارند. نرم بلوکی بردار x را می توان به صورت زیر نشان داد:

$$\|x\|_B = \min \left\{ \sum_{g=1}^r |\lambda_g| : x = \sum_{g=1}^r \lambda_g b_g \right\} \quad (2-1)$$

که نقاط $g = 1, \dots, r$ و b_g و $-b_g$ ، نقاط گوشه ای چند ضلعی کانتور واحد آن نرم هستند. همچنین یک نوع مشخصه سازی دیگر از نرم های بلوکی را بر مبنای مجموعه قطبی پیشنهاد شده که به صورت زیر است:

$$\|x\|_B = \max \{ |x b_g^*| : g = 1, \dots, r \} \quad (3-1)$$

که در آن b_g^0 و $-b_g^0$ با $g = 1, \dots, r$ نقاط گوشه ای مجموعه قطبی $B^* = \{v : b_g^* v \leq 1, g = \pm 1, \pm 2, \dots, \pm r\}$ وارد و وندل با استفاده از این مشخصه سازی مدل های برنامه ریزی خطی کمترین مجموع و مینیماکس را برای مسئله مکانیابی تک وسیله ای پیشنهاد کردند.

اگر از نرم بلوکی B برای اندازه گیری فاصله در صفحه استفاده کنیم آنگاه به دو صورت می توان یک مدل برنامه ریزی خطی برای مسئله نوشت:

(۱) برای هر $i = 1, \dots, n$ قرار می دهیم:

$$\|X - P_i\|_B = \sum_{g=1}^r (\lambda_{gi}^+ + \lambda_{gi}^-) \quad (۴-۱)$$

$$X - P_i = \sum_{g=1}^r (\lambda_{gi}^+ - \lambda_{gi}^-) b_g$$

(۲) برای هر $i = 1, \dots, n$ قرار می دهیم:

$$z_i = \|X - P_i\|_B = \max \left\{ |(X - P_i) b_g| : g = 1, \dots, r \right\} \quad (۵-۱)$$

با جایگذاری این معادلات در مدل یک مسئله برنامه ریزی خطی خواهیم داشت.

۶-۲-۱ محدودیت های فاصله

در بسیاری از مدلها محدودیت فاصله وجود دارد. محدودیت های فاصله بر دو نوع هستند :

(۱) *کران بالا برای فاصله*: در این مدلها سرویس دهنده (ها) به اندازه کافی دور از هم قرار می

گیرند ولی تا حدی که در دسترس نیز باشند.

(۲) *کران پایین برای فاصله*: در این مدلها سرویس دهنده (ها) تا حد امکان نزدیک هم هستند

اما نه بیش از یک حد معین.

۷-۲-۱ وزن

فاصله ها در تابع هدف می توانند دارای وزن های ثابت یا متغیر، یکسان یا مختلف باشند. گاهی

برای مشخص کردن میزان تقاضا در هر نقطه ثابت (مشتریان) از وزن ها استفاده می شود. در مسائل

شبکه ای، وزن یال می تواند نشاندهنده طول مسیر، زمان پیمایش یال، میزان مخاطره و هزینه

استفاده از یال و ارائه خدمات و مواردی از این قبیل باشد.

۸-۲-۱ ارتباط بین سرویس دهندگان

حداقل سه نوع ارتباط مختلف بین سرویس دهندگان می تواند وجود داشته باشد:

(۱) فقط فاصله بین مشتریان و سرویس دهندگان در نظر گرفته شود، مثلا در مکانیابی چند

محل دفن زباله در یک کشور.

(۲) فقط فاصله بین سرویس دهندگان محاسبه شود، مثلا مکانیابی سایت های موشکی در یک

ناحیه بدون جمعیت.

(۳) هر دو نوع فاصله ذکر شده در بالا در نظر گرفته شود، مثلا مکانیابی تعدادی سایت هسته ای

در یک ناحیه دارای جمعیت.

۹-۲-۱ مشتریان

در تعدادی از مدل های مکانیابی نقش مشتریان نیز منظور شده است. لذا لازم است که نحوه توزیع و

میزان تقاضای آنها را بدانیم. برای مثال ممکن است فرض کنیم مشتریان به طور یکنوخت در یک

ناحیه توزیع شده اند و یا در نقاط مخصوصی از فضا، مثلا رئوس یک شبکه قرار دارند و میزان تقاضا

تابعی از فاصله باشد.

مشتریان ممکن است به یک سرویس دهنده اختصاص یابند و یا ممکن است در انتخاب آن آزاد باشند

که در آن صورت سوال اینست که آیا همیشه مشتریان به نزدیک ترین سرویس دهنده مراجعه می

کنند یا عوامل دیگری در انتخاب سرویس دهنده توسط مشتری دخیل هستند.

۱۰-۲-۱ تابع هدف

در این قسمت دسته های اصلی تابع های هدف در مدل های موجود مکانیابی را معرفی می کنیم. اکثر این توابع پیشنهادی هستند. یک راه، تفکیک سرویس دهندگان بر اساس نوع سرویس شان به سرویس دهنده ها است که به سرویس دهنده های عمومی و خصوصی تقسیم می شوند. اغلب موارد خصوصی متناظر با هدف کمترین مجموع و موارد عمومی متناظر با مینیماکس است، زیرا هدف کمترین مجموع سرویس دهنده را به گونه ای مکانیابی می کند که مجموع هزینه های حمل و نقل مینیمم شود. در مکانیابی سرویس دهنده های عمومی مانند استخر یا کتابخانه، می خواهیم مکانی را پیدا کنیم که به همه مشتریان نزدیک باشد.

مورد دیگری که در تابع هدف مد نظر قرار می گیرد، میزان مطلوبیت سرویس دهنده برای مشتریان است. سرویس دهندگان از دید مشتریان یکی از چهار حالت زیر را دارند :

(۱) مطلوب : در این حالت سرویس گیرنده می خواهد تا جایی که امکان دارد سرویس دهنده به او نزدیک باشد، یعنی سعی در جذب سرویس دهنده دارد .

(۲) نامطلوب (مضر)^{۱۸} : سرویس گیرنده می خواهد تا جایی که امکان دارد از سرویس دهنده فاصله بگیرد، یعنی سعی در دفع آن دارد .

(۳) بی تفاوت : سرویس گیرندگانی که برایشان اثرات مثبت و منفی سرویس دهنده اهمیت ندارد را می توان از مدل کنار گذاشت چرا که میزان مطلوبیت سرویس دهنده برای آنها محاسبه نمی شود و در نهایت در جواب مسئله نیز تاثیر گذار نیستند.

(۴) تا حدی مطلوب و تا حدی نا مطلوب^{۱۹} : در این حالت سرویس گیرنده می خواهد که سرویس گیرنده نزدیک باشد ولی نه خیلی نزدیک. مثلاً یک سوپر مارکت را در نظر بگیرید، از یک طرف مشتریان می خواهند که برای راحتی در امر خرید به فروشگاه نزدیک باشند و از طرف دیگر ترافیک،

شلوغی و سر و صدای ناشی از فروشگاه، آن را نامطلوب و ناخوشایند می سازد. در چنین موقعیتی، سرویس گیرنده سه انتخاب دارد:

- (۱) با اعمال یک محدودیت روی فاصله ما بین شان سرویس دهنده را دفع کند، یعنی سرویس دهنده تا حد امکان نزدیک باشد ولی حداقل به اندازه از پیش تعیین شده فاصله داشته باشد.
- (۲) با اعمال یک محدودیت روی فاصله ما بین شان سرویس دهنده را جذب کند، یعنی سرویس دهنده تا حد امکان دور باشد ولی حداکثر به اندازه از پیش تعیین شده از آن فاصله داشته باشد.
- (۳) ساخت یک هدف ترکیبی جاذبه-دافعه^{۲۰}، مثلاً ترکیب محدبی از دو هدف جاذب و دافع.

۳-۱ تابع هدف های جاذب^{۲۱}

در تمام توابع هدفی که در این قسمت در موردشان بحث می شود، فرض بر این است که سرویس دهنده هایی که قرار است مستقر شوند، همگی مطلوب و خوشایند هستند. بنابراین تصمیم گیرنده می خواهد که آنها را نزدیک مشتریان مستقر کند. از این جهت این مدل ها را دارای خاصیت جاذبه می نامیم. مدل های مکانیابی از قدیمی ترین آنها یعنی مسائل وبر، میانه، مرکز و پوشش همگی در این دسته قرار می گیرند. از جمله هدف های جاذب می توان می نیم سازی هزینه حمل و نقل یا حداکثر زمان سفر و ماکزیمم کردن جمعیت در دسترس یا منطقه تحت پوشش بازار را نام برد. در میان مدل های فراوانی که در زمینه مکانیابی گسسته ارائه شده است چهار دسته از مدل ها نقش اساسی و محوری را بازی می کنند: مسائل p -میانه^{۲۲}، p -مرکز^{۲۳}، مکانیابی سرویس دهندگان با

Push-Pull ۲۰

Pull objectives ۲۱

P-median ۲۲

P-center ۲۳

ظرفیت نامحدود^{۲۴} و مسئله تخصیص درجه دو^{۲۵}. برای مسائل p -میانه، مکانیابی سرویس دهندگان با ظرفیت نامحدود و مسئله تخصیص درجه دو، تابع هدف از نوع کمترین مجموع است و در مسئله p -مرکز از نوع مینیماکس است و همگی به شکلی، تمایل به نزدیکی بین سرویس دهندگان و مشتریان یا بین سرویس دهندگان دارند، یعنی جاذب هستند. بیشتر کاربرد های بهینه سازی ترکیباتی در مکانیابی، در این حوزه قرار می گیرد. سه دسته معروف از مسائل یعنی پوشش مجموعه^{۲۶}، افراز مجموعه^{۲۷} و بسته بندی کردن مجموعه^{۲۸} همگی حالت های خاصی از مسئله مکانیابی سرویس دهندگان با ظرفیت نامحدود هستند.

مینیمم سازی مجموع وزنی فاصله که به هدف کمترین مجموع معروف است یکی از بهترین گزینه ها برای استفاده در مسائل بخشهای عمومی و خصوصی است. حکیمی اولین کسی بود که مسائل مکانیابی با هدف کمترین مجموع را مورد بررسی قرار داد [۴]. اولین دسته از مسائل پیوسته با هدف کمترین مجموع تعمیمی از مسئله مطرح شده توسط فرماست. فرض کنید که تصمیم گیرنده قصد دارد مکان سرویس دهنده های جدیدی را پیدا کند. مکان این سرویس دهندگان را با $x_1, \dots, x_p$ نشان می دهیم. همچنین فرض می کنیم که مشتریان در نقاط $v_1, \dots, v_m$ مستقر هستند. اگر وزن های متناظر با مشتریان باشند، مدل مسئله به صورت زیر است :

$$\text{Min} \quad f_{\text{sum}}(x) = \sum_{i=1}^n w_i \min_k \{d(v_i, x_k)\} \quad (6-1)$$

اگر به دنبال پیدا کردن مکان یک وسیله باشیم آنگاه مسئله را مسئله مکانیابی تک وسیله ای^{۲۹} و در حالتی که فاصله اقلیدسی باشد مسئله وبر^{۳۰} می نامند. برای مرور کارهای انجام شده در مورد این مسئله به [۱۳-۱۶] مراجعه کنید.

اگر $p > 1$ یعنی بخواهیم چند وسیله را مستقر کنیم آنگاه مسئله را مسئله چند وسیله^{۳۱} می نامند. این مسئله اولین بار در [۱۳] مطرح شد. در این مسئله علاوه بر مکانیابی سرویس دهندگان، با تخصیص آنها به مشتریان نیز رو به رو هستیم.

دسته ای دیگر از مسائل سعی در استقرار یک سرویس دهنده دارند به گونه ای که بیشترین فاصله بین سرویس دهنده و مشتریان حداقل ممکن باشد. این مسائل را مینیماکس می گویند که در [۱۷] بحث شده اند. مسئله p - مرکز در این دسته قرار می گیرد.

$$\text{Min} \quad f_{\max}(x) = \max_i \{w_i \min_k \{d(v_i, x_k)\}\} \quad (7-1)$$

بعضی از محققان سعی کرده اند کارایی مسئله میانه را با مطلوبیت مسئله مرکز ترکیب کنند. یکی از راه های انجام این کار این است که مسئله p - میانه را محدود کنیم. یعنی یک مسئله با هدف کمترین مجموع که محدودیت کران بالا برای فاصله بین تاسیسات دارد. این مسئله اولین بار در [۱۶ و ۱۸] مطرح شد و به مسئله میانه-مرکز معروف است. یک راه دیگر آن است که از یک تابع هدف مینیماکس با یک کران روی مجموع وزن دار فاصله ها استفاده کنیم. در [۱۹] ترکیب خطی محدبی از هدفهای کمترین مجموع و مینیماکس ارائه شده است. این مسائل را مرکز-میانه می نامند.

۴-۱ تابع هدف های دافع^{۳۲}

Single Facility Location Problem^{۲۹}

Weber Problem^{۳۰}

Multi Facility Location Problem^{۳۱}

Push objectives^{۳۲}

اگرچه نظریه مکانیابی ریشه در قرن هفدهم دارد ولی مسائل مکانیابی که شامل سرویس دهندگان نامطلوب می باشند. تقریباً از اوایل دهه ۱۹۷۰ مورد بررسی قرار گرفته اند دلیل این امر ناشی از پیشرفت های تکنولوژیکی و صنعتی سازی است.

در مسائل مکانیابی سرویس دهندگان ناخوشایند (نامطلوب) مجموعه ای از نقاط وجود دارند که تحت تاثیرات منفی سرویس دهندگان قرار دارند و هدف مینیم کردن این تاثیرات است. مکانیابی راکتورهای هسته ای، نیروگاه ها، محل دفن زباله و فرودگاه های بزرگ از مسائل قرن بیستم است. از طرف دیگر سرویس دهنده های مطلوب مثل پاسگاه های پلیس، بیمارستان، مدرسه، انبار برای آنها وجود داشته اند. اولین تلاشی که برای فرمولبندی این مسائل صورت گرفت توسط چرچ و گارفینکل [۶] و گولدمن و دیرینگ [۵] در سال های ۱۹۷۵ و ۱۹۷۸ بود که مدل ماکزین^{۳۳} را پیشنهاد کردند. این مدل هم ارز و هم تراز با مدل میانه است با این تفاوت که هدف آنها بیشترین مجموع است. یک منبع قدیمی تر در مورد این دسته از مسائل، مسئله ایست که در کتاب "ریاضیات چیست"^{۳۴} نوشته کورانت و رابینز [۲۰] در سال ۱۹۴۱ مطرح شده است. در این کتاب ابتدا مسئله ای که فرما در قرن هفدهم مطرح کرد فرموله شد سپس همراه آن مسئله ای مطرح شده است که به مسئله متمم^{۳۵} (comp) معروف است. مسئله متمم همان مسئله فرماست با این تفاوت که وزن یکی از رئوس مثلث را در این حالت برابر با ۱- در نظر می گیریم. مسئله متمم اولین مسئله مکانیابی است که صریحاً وزن های مثبت و منفی را به طور همزمان به کار می گیرد. متأسفانه راه حلی که در [۲۰] ارائه شده اشتباه است و کرارپ در مقاله ای در سال ۱۹۱۸ [۲۱] راه حل درست آن را ارائه داده است.

یک راه برای مدل سازی مکانیابی یک سرویس دهنده نامطلوب این است که به مسئله مینیم سازی یک کران پایین برای فاصله بین مشتری و سرویس دهنده اضافه کنیم. هر مشتری یا دسته ای از مشتریان، یک ناحیه ممنوع به فضای جواب تحمیل می کنند. این مدل ها در مقایسه با مدل های

maxian ۳۳

What's Mathematics ۳۴

Complementary Problem ۳۵

دارای هدف ماکزیمم سازی کمتر مورد توجه قرار گرفته اند و مقالات کمی در این زمینه نوشته شده است. مون و چودهاری [۲۲] یک مدل برنامه ریزی عدد صحیح برای مسائل p -میان با کران پایین بین هر جفت از سرویس دهنده ها ارائه کردند مدلی هایی از این نوع بعد ها مدل های نامطلوب یا ناخوشایند خوانده شدند. مدلی هایی که در زمینه توابع هدف دافع بیشتر مورد توجه قرار گرفته اند عبارتند از ماکزیمم کردن فاصله، پراکندگی و کمترین پوشش. هدف های دافع به طور کلی در [۲۴و۲۳و۹] مرور و بیان شده اند. ما این دسته از مدل ها را به دو بخش تقسیم می کنیم :

۱-۴-۱ مسائل مکانیابی سرویس دهندگان نامطلوب^{۳۶} یا مضر^{۳۷}

در این مسائل مجموعه ای از نقاط ثابت وجود دارند و تحت تاثیرات منفی سرویس دهندگان قرار دارند. هدف مینیمم کردن این تاثیرات است. تفاوت بین سرویس دهندگان نامطلوب و مضر، بیشتر مربوط به حوزه معنایی آنهاست تا تابع هدفشان. مدلی های نوع دوم را عموماً به سرویس دهندگان خطرناک نسبت می دهند. مثلاً آنهایی که به خاطر نشتی محیط اطراف را سمی می کنند. سرویس دهندگان نوع اول برای مشتریان و سرویس دهندگان دیگر مزاحمت و ناراحتی ایجاد می کنند. مثلاً احداث ورزشگاه ها در مراکز شهر که با ایجاد سروصدا و شلوغی برای ساکنان اطراف آن مزاحمت ایجاد می کنند. در هر دو حالت مشتریان سعی دارند تا حد امکان از آنها فاصله بگیرند.

۲-۴-۱ مدل های پراکندگی

در این مسائل، سرویس دهندگان به طریقی مستقر می شوند که کمترین تاثیر را روی هم داشته باشند.

در مسائل مکانیابی تک وسیله تنها مدلی های نوع اول رخ می دهند و قسمت قابل توجهی از مقالات ارائه شده، از مدلی در محیط پیوسته استفاده کرده اند. دلیل عمده این امر اینست که تاثیرات نامطلوب مثل گرما، صدا، تشعشع و آلودگی معمولاً به طور پیوسته در فضا پراکنده می شوند مدلی های

Obnoxious^{۳۶}

Noxious^{۳۷}

مکانیابی نامطلوب چند وسیله ای و مدل های پراکندگی عموماً در حالت گسسته بررسی شده اند و لذا روش های بهینه سازی ترکیبیاتی برای حل آنها به کار گرفته شده اند.

حال برای درک تفاوت بین تابع هدف های کمترین مجموع و بیشترین مجموع، مثال زیر را مطرح می کنیم. فرض کنیم می خواهیم مکان یک سرویس دهنده جدید را با توجه به مشتریان موجود در رئوس یک مربع که دارای وزنهای منفی هستند، پیدا کنیم. مکان بهینه سرویس دهنده جدید تا حد امکان دور از مربع است و به سمت بینهایت میل می کند و این به وضوح نشان دهنده نیروی دافع سرویس دهندگان است. برای دوری از جواب های بهینه بیکران، لازم است مجموعه ای تعریف کنیم که سرویس دهندگان در آن قرار گیرند. برای تابع های کمترین مجموع و وزنهای مثبت، می توان به راحتی ثابت کرد که سرویس دهنده جدید در پوسته محدب نقاط موجود قرار می گیرد. اگر در مثال بالا ناحیه جواب را به پوسته محدب نقاط موجود محدود کنیم، هدف بیشترین مجموع در مکان هر یک از مشتریان بهینه خواهد شد.

استقرار یک سرویس دهنده به قسمی که فاصله بین آنها و هر کدام از مشتریان ماکزیمم شود، معادل با یک مسئله هندسی یعنی پیدا کردن بزرگترین دایره خالی است. یعنی بزرگترین دایره ای شامل هیچ کدام از نقاط داده شده نباشد. مرکز این دایره جواب بهینه مسئله است. این مسائل را بعداً مسائل ۱-ماکزین^{۳۸} نامیدند. در [۲۴] مروری بر کار های انجام شده با توابع بیشترین مجموع و ماکزیمین و فاصله های اقلیدسی و مستطیلی ارائه شده است. مسائل پاد-وبر^{۳۹} و پاد-رائول^{۴۰} از دیگر مسائل این حوزه هستند که مورد بررسی قرار گرفته اند.

۵-۱ تابع هدفهای پر کاربرد مکانیابی

۳۸-maxian ۱

۳۹-Anti-Weber

۴۰-Anti-Rawl

در این بخش لیستی از توابع هدف معروف مکانیابی با توضیح مختصری درباره هر کدام ارائه می دهیم:

پاد پوشش^{۴۱}: طرح استقرار حداکثر تعداد سرویس دهندگان به قسمی که فاصله بین هر جفت سرویس دهنده بزرگتر یا مساوی با مقدار مشخص a باشد.

پاد مرکز^{۴۲}: نقطه ای که کوتاه ترین فاصله وزن دار بین سرویس دهنده و مشتری را ماکزیمم کند.

پاد مرکز پراکنده^{۴۳}: استقرار p وسیله جدید در بین تعداد وسیله موجود به قسمی که کمترین فاصله های بین سرویس دهندگان جدید و قدیم، ماکزیمم شود.

پاد مرکز-ماکزین^{۴۴}: ترکیب خطی محدب از توابع ماکزین و پاد مرکز است.

پاد میانه^{۴۵}: نقطه ای روی شبکه که مجموع فواصل را ماکزیمم کند.

پاد-ویر^{۴۶}: نقطه ای روی صفحه که مجموع فواصل بین سرویس دهنده و مشتری را ماکزیمم می کند.

پراکنده^{۴۷}: طرح استقرار p سرویس دهنده به گونه ای که کمترین فاصله بین سرویس دهندگان ماکزیمم شود.

پراکنده-دفاع^{۴۸}: ترکیب خطی محدب از دفاع و پراکنده است.

پراکندگی با بیشترین مجموع^{۴۹}: مجموع فاصله های بین سرویس دهندگان را ماکزیمم می کند.

Anti covering ۴۱

Anticenter ۴۲

Anticenter Dispersion ۴۳

Anticenter-Maxian ۴۴

Antimedial ۴۵

Anti-Weber ۴۶

Dispersion ۴۷

Dispersion-Defense ۴۸

حد اکثر پوشش^{۵۰}: استقرار p نقطه بگونه ای که مجموع وزن دار مناطق تحت پوشش با فاصله a را ماکزیمم کنند.

کم هزینه ترین پوشش^{۵۱}: کمترین تعداد سرویس دهندگانی که تمام مشتریان را در فاصله a تحت پوشش قرار می دهند.

ماکزین: نقطه ای روی شبکه که مجموع فاصله های وزن دار را ماکزیمم کند.

مرکز: نقطه ای روی شبکه که بیشترین فاصله بین سرویس دهنده و مشتری را مینیمم کند.

مرکز کلی^{۵۲}: نقطه ای روی شبکه که ماکزیمم فواصل بین آن و هر نقطه ای روی شبکه را مینیمم کند. تفاوت این مسئله با مسئله مرکز در این است که جواب مسئله مرکز کلی لزوماً یکی از رئوس شبکه نیست و می تواند هر جای آن از جمله روی یالها نیز قرار گیرد.

مرکز-میانه^{۵۳}: ترکیب خطی محدب از مرکز و میانه است.

مرکز جزیی^{۵۴}: p نقطه که ماکزیمم فاصله بین سرویس دهندگان و مشتریان را مینیمم می کند. تنها فاصله هایی در نظر گرفته می شوند که مشتری مزبور تحت پوشش باشند.

مسئله چند وسیله ای وبر^{۵۵}: p نقطه روی صفحه که مجموع فواصل بین مشتری و نزدیکترین سرویس دهنده را می نیمم می کند.

میانه: نقطه ای روی شبکه که مجموع فواصل بین سرویس دهنده و مشتری را مینیمم کند.

Maxisum Dispersion ۴۹

Max Cover ۵۰

Min cost cover ۵۱

General Center ۵۲

Cent-dian ۵۳

Partial Center ۵۴

Multi Weber problem ۵۵

میان‌ه کلی^{۵۶}: نقطه ای روی شبکه که مجموع فواصل بین آن و هر نقطه‌های روی شبکه را مینیمم کند. تفاوت این مسئله با مسئله میان‌ه در این است که جواب مسئله مرکز کلی لزوماً یکی از رئوس شبکه نیست و می‌تواند هر جای آن از جمله روی یالها نیز قرار گیرد.

میان‌ه-مرکز^{۵۷}: نقطه ای روی شبکه که میان‌ه فاصله‌ها را می‌نیمم کند.

p -دفاع^{۵۸}: طرح استقرار p سرویس دهنده به قسمی که مجموع نزدیکترین فاصله بین سرویس دهندگان ماکزیمم شود.

p -میان‌ه مقید^{۵۹}: استقرار p نقطه روی شبکه که مجموع فاصله مشتریان تا نزدیکترین سرویس دهنده مینیمم شود.

k -مرکزی^{۶۰}: نقطه ای روی شبکه که مجموع فواصل k تا از دورترین سرویس دهنده‌ها را مینیمم کند.

p -میان‌ه^{۶۱}: استقرار p نقطه روی شبکه بگونه ای که مجموع فواصل بین مشتری تا نزدیکترین سرویس دهنده می‌نیمم گردد.

p -مرکز^{۶۲}: استقرار p نقطه روی شبکه بگونه ای که بیشترین فاصله بین مشتری سرویس دهنده گان می‌نیمم گردد.

General Median ۵۶

Medi-Center ۵۷

p-Defense ۵۸

Constrained p-median ۵۹

K-centrum ۶۰

p-Median ۶۱

P-Center ۶۲

فصل ۲

الگوریتم‌های شاخه و کران برای مسائل مکانیابی پیوسته

۱-۲ مقدمه

مسائل مکانیابی در دهه های گذشته توجه بسیاری از پژوهشگران را بخود جلب کرده است. در نظریه مکانیابی، گاهی با مسائلی برخورد می کنیم که نامحدب و مشتق ناپذیرند و حتی دارای بهینه های محلی^۱ زیادی هستند و لذا باید از روشهای بهینه سازی سراسری^۲ برای حل مسائل استفاده کنیم. نوعی روش بهینه سازی سراسری بر مبنای افراز ناحیه موجود و ایجاد شاخه و کران توسط هرست [۲۵] و توی و هرست [۲۶]، به ترتیب در سالهای ۱۹۸۶ و ۱۹۸۸ ارائه شد. الگوریتمی که در سال ۱۹۸۵ توسط هانسن و همکارانش [۲۷] برای حل مسئله تک وسیله ای ارائه شد، یعنی مستطیل بزرگ مستطیل کوچک^۳، از این نوع است. مزیت اصلی این الگوریتم، سادگی مفهوم آن و توانایی و قدرت زیاد آن برای حل تقریباً تمام مسائل مکانیابی تک وسیله ای پیوسته که تا بحال مطرح شده است، می باشد. روش های شاخه و کران^۴، مسائل سخت را با افراز ناحیه شدنی به زیر ناحیه ها (شاخه زدن) و ارزیابی زیر ناحیه ها که شامل محاسبه تابع هدف و یا محدودیت هاست (ایجاد کران) حل می کند. ارزیابی ها تمام اطلاعات مورد نیاز از زیر مسائل نظیر اینکه آیا یک جواب خاص یک جواب بهینه برای زیر مسئله است و یا آیا برای یک زیر مسئله جوابی بهتر از جواب فعلی وجود دارد و یا آیا اصلاً مسئله جواب دارد، را در اختیار قرار می دهند.

روش های شاخه و کران از آغاز به طور گسترده در مسائل برنامه ریزی عدد صحیح به کار گرفته شده اند. روشهایی که در این بخش معرفی می شوند از روشهای شاخه و کران برای مسائل پیوسته و

Local optima ۶۵

Global optimization ۶۶

Big Square Small Square (BSSS) ۶۸

Branch and Bound (B&B) ۶۹

بخصوص مسائل مکانیابی پیوسته هستند. در ادامه به بررسی الگوریتمهای مستطیل بزرگ مستطیل کوچک و نیز مثلث بزرگ مثلث کوچک^۵ می پردازیم.

۲-۲ روش مستطیل بزرگ مستطیل کوچک

روش مستطیل بزرگ مستطیل کوچک یک روش شاخه و کران هندسی است که جوابهای مسئله را با دقت قابل قبولی بدست می آورد. این روش ملهم از ایده ای است که فرانسیس [۲۸] در سال ۱۹۶۷ ارائه داد. این روش بطور رسمی در سال ۱۹۸۵ توسط هانسن و همکاران برای حل مسئله p -میانه مقید مورد استفاده قرار گرفت و در سال ۱۹۹۲ توسط پلستریا [۲۹] برای سرعت بخشیدن به همگرایی آن اصلاح شد. این الگوریتم برای حل مسائل مکانیابی پیوسته مختلفی همچون ۱-میانه با وجود مانع و نواحی ممنوع، مکانیابی رقابتی و مسئله وبر با جاذبه و دافعه بکار گرفته شده است. در سال ۲۰۰۸ تعمیمی از این الگوریتم برای حل مسائل چند مدی، به نام الگوریتم مکعب بزرگ مکعب کوچک ارائه شده است [۳۰].

از اولین کارهایی که از روش های شاخه و کران برای حل مسائل مکانیابی پیوسته استفاده شده است می توان به مقالات [۳۱ و ۳۲] اشاره کرد که در آن مکانیابی یک سرویس دهنده نامطلوب مورد مطالعه بوده است و الگوریتم پیشنهادی آنها به روش مستطیل بزرگ مستطیل کوچک، گسترش پیدا کرد.

۱-۲-۲ الگوریتم مستطیل بزرگ مستطیل کوچک

(۱) کوچکترین مستطیلی که شامل نقاط شدنی (پیوسته مستطیلی) است را به چهار ناحیه مستطیلی با مساحت های مساوی افراز می کنیم .

^۵ Big Triangle Small Triangle (BTST)

(۲) مقدار کران پایین تابع هدف F را روی زیر مستطیل هایی که با ناحیه شدنی اشتراک دارند، حساب می کنیم .

(۳) زیر مستطیل هایی که کران پایین آنها از بهترین مقدار به دست آمده بزرگتر است را کنار می گذاریم و آن را بروز می کنیم .

(۴) این مراحل را آن قدر تکرار می کنیم تا ماکزیمم طول و عرض مستطیل کمتر از یک مقدار داده شده شود.

قواعد الگوریتم مستطیل بزرگ مستطیل کوچک به صورت زیر است :

الف) مقدمات.

فرض کنید Q_1 کوچکترین مستطیلی باشد که شامل ناحیه شدنی (G) است و اضلاعش موازی محور مختصات است. فرض که طول اضلاع Q_1 برابر با l است و ε حداکثر طول ضلع مستطیل در شرط خاتمه الگوریتم باشند. مقدار F را در چند نقطه شدنی محاسبه و F_{opt} که کوچکترین آنهاست را به دست می آوریم. S_{opt} نقطه متناظر با F_{opt} است .

ب) شاخه زدن و تست شدنی بودن.

هر یک از مستطیل های Q_i از لیست فعلی مستطیل ها را به ترتیب به چهار زیر مستطیل $Q_{k+1}, Q_{k+2}, Q_{k+3}, Q_{k+4}$ افراز کنید. برای هر یک از این زیر مستطیل های Q_j اگر $Q_j \cap G \neq \emptyset$ آنگاه آن را به لیست جدید از مستطیل ها اضافه کنید. بعد از اینکه تمام Q_i ها بررسی شدند، لیست جدید را جایگزین لیست قدیم کنید.

ج) بهبود جواب ها.

برای مستطیل های Q_i موجود در لیست فعلی به ترتیب فرض که s_i مرکز Q_i باشد. اگر s_i شدنی

$$\text{بود و } F(s_i) < F_{opt}, \quad S_{opt} \leftarrow s_i, \quad F_{opt} \leftarrow F(s_i).$$

د) ایجاد کران و تست بهینگی.

برای هر کدام از مستطیل ها Q_i درون لیست فعلی، برای نقطه تقاضا j ام، نزدیک ترین نقطه به

آن را روی مستطیل Q_i پیدا کنید و کران پایین $\underline{F}_i$ را حساب کنید.

اگر $\underline{F}_i \geq F_{opt}$ مستطیل Q_i را از لیست حذف کنید.

ه) تست خاتمه الگوریتم.

اگر $l < \varepsilon$ الگوریتم متوقف است و S_{opt} با مقدار F_{opt} جواب بهینه است در غیر این صورت

$$l \leftarrow \frac{l}{2} \text{ و به مرحله } b \text{ برگرد.}$$

۳-۲ روش مثلث بزرگ مثلث کوچک

این روش اصلاح شده روش مستطیل بزرگ مستطیل کوچک است که در آن به جای مستطیل از مثلث استفاده شده است [۳۳]. این روش مزایایی دارد که در ادامه بحث خواهد شد و برای مسائل مکانیابی غیر محدب و پیوسته پیشنهاد شده است. این روش برای مسائل مختلفی مثل مکانیابی سرویس دهنده مضر، مسئله وبر با جاذبه ودافعه، p - میانه مرتب و... اجرا شده و نتایج خوبی حاصل شده است. این الگوریتم دو مزیت عمده دارد. اول اینکه وقتی ناحیه شدنی اجتماعی از چند ضلعی هاست، مثلث ها تمام ناحیه شدنی را می پوشانند و شامل هیچ نقطه نشدنی نمی باشد بنابراین تست شدنی بودن که کاری وقت گیر و نامنظم است، انجام نمی گیرد، لذا فقط به یک مثلث بندی اولیه احتیاج داریم. مزیت دوم اینست که محاسبات کران ها برای خانه ها ساده تر شده اند و حتی در

مواردی (وقتی هیچ نقطه تقاضایی در مثلث نباشد) دقیق تر هم شده است. بنابراین در مثلث بندی ها نقاط تقاضا به عنوان رئوس مثلث های آن هستند. برای مثلث بندی ناحیه شدنی از ساختارهای هندسی گراف ورونوی^۶ و مثلث بندی دلانی^۷ استفاده می شود که در ادامه این فصل این ساختارها را معرفی می کنیم.

۱-۳-۲ گراف ورونوی

فرض کنید که هیئت مرکزی یک فروشگاه زنجیره ای برای خدمات رسانی بهتر و تامین نیاز مشتریان تصمیم به احداث یک شعبه جدید دارند. مهمترین عاملی که در انتخاب محل احداث فروشگاه جدید باید مد نظر قرار دهیم، تعداد مشتریانی است که فروشگاه می تواند جذب کند. برای این منظور باید رفتار مشتریان را مدلسازی کنیم، یعنی بدانیم که مشتریان برای انتخاب محل خرید خود، بر چه اساسی تصمیم گیری می کنند. به عبارت دیگر به دنبال قلمرو تجاری یک سرویس دهنده هستیم. برای بررسی این سوالات فرضیات زیر را در نظر می گیریم:

- ۱- قیمت هر کالا در تمام نقاط ارائه خدمات ، مشابه است.
 - ۲- هزینه تهیه خدمات/کالا برابر است با مجموع هزینه های خرید و حمل و نقل
 - ۳- هزینه حمل و نقل تا یک سرویس دهنده برابر است با فاصله اقلیدسی ضربدر کرایه هر واحد فاصله پیموده شده.
 - ۴- هدف مشتریان دریافت خدمات با کمترین هزینه است.
- فرضیات بالا ناحیه موجود را به نواحی کوچکتر تقسیم می کنند - قلمرو تجاری هر سرویس دهنده-
 بگونه ای که مشتریانی که در یک ناحیه زندگی می کنند، همگی از یک سرویس دهنده استفاده می

^۶ Voronoi Diagram
^۷ Delaunay Triangulation

کنند. طبق فرضیات بالا، مشتریان نزدیکترین سرویس دهنده را برای دریافت خدمات، انتخاب می کنند. بنابراین هر ناحیه دارای یک سرویس دهنده است که فاصله نقاط ناحیه تا آن کمتر از سایر سرویس دهندگان است.

مدلی که در آن، نقاط به نزدیکترین سرویس دهنده اختصاص می یابند را مدل تخصیص ورونوی می نامند و افراز بندی و گراف حاصل از این مدل را گراف ورونوی می نامند. گراف ورونوی یک ساختار هندسی است که کاربرد های زیادی در فیزیک، نجوم، رباتیک، در زیست شناسی، جغرافیا، محیط زیست، باستان شناسی و بلورشناسی دارد [۳۴]. همچنین گراف ورونوی با مثلث بندی دلانی که یکی از مهمترین ساختار های هندسی است ارتباط زیادی دارد از اینرو یکی از ابزار های معروف و پر کاربرد در مسائل مکانیابی پیوسته گراف ورونوی است.

فرض کنید که مجموعه $P = \{p_1, \dots, p_p\}$ شامل p نقطه p نقطه در صفحه موجود باشد، و هر نقطه از صفحه را به نزدیک ترین نقطه از اعضای P اختصاص می دهیم. بنابراین صفحه به چند ضلعی های $V_1, \dots, V_p$ افراز می شود (شکل ۱-۲) [۳۳]. این افراز را گراف ورونوی، چند ضلعی های $V_1, \dots, V_p$ را چند ضلعی های ورونوی و P را مجموعه مولد می نامیم. طبق تعریف چند ضلعی ورونوی، اگر نقطه q متعلق به V_i باشد آنگاه p_i نزدیکترین نقطه مولد به نقطه q است. به همین سبب گاهی V_i را ناحیه یا قلمرو p_i می نامیم.

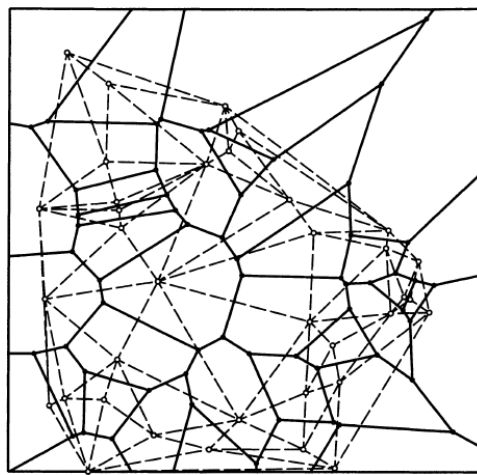
حال تعاریف بالا را به صورت ریاضی فرمول بندی می کنیم. فرض کنید $p_i = (x_i, y_i)$ نشان دهنده مکان p_i در فضای R^2 باشد و $\|q - p_i\|$ فاصله اقلیدسی بین q, p_i باشد. چند ضلعی متناظر با p_i را به صورت زیر تعریف می کنیم :

$$V_i = \bigcap_{j: j \neq i} \{q \in R^2 \mid \|q - p_i\| < \|q - p_j\|\} \quad (1-2)$$

و همچنین

$$\begin{aligned} w_{ij} &= \partial V_i \cap \partial V_j \\ U_{ijk} &= w_{ij} \cap w_{jk} \cap w_{ki} \end{aligned} \quad (2-2)$$

که w_{ij} (مرز مشترک ناحیه های V_i, V_j) را یال ورونوی و U_{ijk} (نقطه هم راسی حداقل سه یال ورونوی) را یک نقطه ورونوی می نامیم.



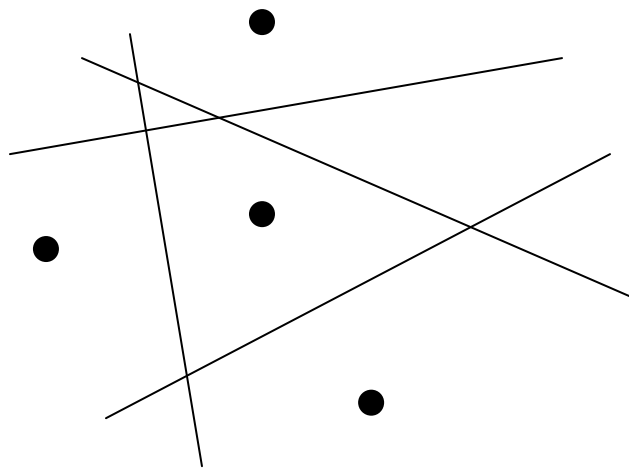
شکل ۱-۲: گراف ورونوی و دوگان آن

برای ساخت گراف ورونوی الگوریتم های زیادی در شاخه هندسه محاسباتی پیشنهاد شده است [۳۷-۳۵]. دو تا از بهترین این روشها، روش تقسیم و گسترش و روش افزایشی است [۳۶]. پیچیدگی محاسباتی روش اول از مرتبه $O(p \log p)$ در بدترین حالت و همچنین در حالت میانگین است و روش دوم از مرتبه $O(p^2)$ در بدترین حالت و از مرتبه $O(p)$ در حالت میانگین است.

برای هر دو نقطه p, q در صفحه، عمود منصف پاره خط $\overline{pq}$ ، صفحه را به دو نیم صفحه تقسیم می کند. نیم صفحه ای که شامل p است را با $h(p, q)$ و دیگری را که شامل q است را با $h(q, p)$ نشان می دهیم. لذا اگر $r \in h(p, q)$ و تنها اگر $d(r, p) < d(r, q)$.

$$\text{گزاره ۱-۲ ([۳۸]): } V_i = V(p_i) = \bigcap_{\substack{1 \leq j \leq n \\ j \neq i}} h(p_i, p_j)$$

بنابر گزاره ۱-۲، $V_i = V(p_i)$ اشتراک $n-1$ نیم صفحه است و از اینرو یک ناحیه چند ضلعی محدب (و احتمالاً بیکران) با حداکثر $n-1$ یال و $n-1$ رأس است. نتیجه دیگر اینکه گراف ورونوی یک افراز و تقسیم بندی مسطح است که یالهایش مستقیم هستند. بعضی یالها پاره خط و بعضی نیم خط هستند (شکل ۲-۲).



شکل ۲-۲: یالهای متناهی و نیمه متناهی گراف ورونوی

قضیه ۲-۲ ([۳۸]): فرض کنیم P مجموعه ای از n نقطه در صفحه باشد. اگر این نقاط هم خط باشند آنگاه $V(P) = \bigcup_{1 \leq i \leq n} V_i$ شامل $n-1$ خط موازی است در غیر اینصورت $V(P)$ همبند است و یالهایش پاره خط یا نیم خط هستند.

برهان. اثبات قسمت اول به سادگی و از گزاره ۱-۲ نتیجه می شود. حال فرض کنیم که تمام نقاط هم خط نباشند. ابتدا نشان می دهیم که یالهای $V(P)$ پاره خط یا نیم خط هستند. تا اینجا بنابر اطلاعاتی که داریم، می دانیم یالهای $V(P)$ قسمت هایی از خط های مستقیم یعنی عمود منصف های هر جفت از نقاط هستند. به برهان خلف فرض می کنیم که $V(P)$ دارای یک یال مثل e است که خط است. فرض کنیم e روی مرز خانه های ورونوی $V(p_i), V(p_j)$ و $p_k \in P$ نقطه غیر هم

خط با p_i, p_j باشد. عمود منصف $\overline{p_j p_k}$ با e موازی نیست و لذا e را قطع می کند. قسمتی از e که درون $h(p_k, p_j)$ قرار می گیرد، نمی تواند روی مرز $V(p_j)$ باشد، زیرا به نقطه p_k نسبت به p_j نزدیکتر است و این تناقض است. حال ثابت می کنیم که $V(P)$ همبند است. اگر چنین نباشد آنگاه باید چند ضلعی $V(p_i)$ وجود باشد که صفحه را به دو قسمت تقسیم کند. چون چند ضلعی های ورونوی محدب هستند، $V(p_i)$ می بایست شامل یک ناحیه باشد که با دو خط موازی کراندار شده است. قبلاً ثابت کردیم که یالهای گراف ورونوی نمی توانند خط باشند و این تناقض است. بنابراین حکم برقرار است. $\square$

تعریف (۳۹). گراف مسطح گرافی است که بتوان آن را روی صفحه نشان داد، یعنی بتوان آن را طوری رسم کرد که یالهایش همدیگر را قطع نکنند.

لم ۲-۳. (فرمول اویلر) ([۳۹]). اگر G یک گراف همبند و مسطح روی صفحه باشد تعداد یالهایش e ، تعداد راس هایش v و تعداد وجه هایش f است، آنگاه:

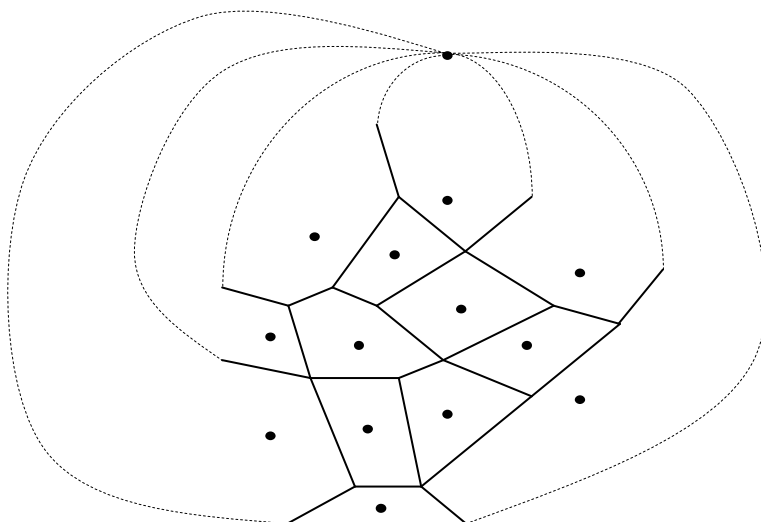
$$v - e + f = 2$$

قضیه ۲-۴: ([۳۸]) برای $n \geq 3$ ، تعداد رئوس گراف ورونوی n نقطه در صفحه، حداکثر $2n - 5$ و تعداد یالها حداکثر $3n - 6$ است.

برهان. اگر نقاط هم خط باشند، نتیجه از قضیه ۲-۲ بدست می آید. این قضیه را با استفاده از فرمول اویلر و با فرض هم خط نبودن نقاط اثبات می کنیم. بنابر فرمول اویلر، برای هر گراف همبند مسطح که بر روی صفحه نشانده شده است و در آن m_v بیانگر تعداد گره ها، m_e تعداد یالها و m_f تعداد وجه ها باشند رابطه زیر برقرار است:

$$m_v - m_e + m_f = 2 \quad (3-2)$$

اما چون $V(P)$ شامل یالهای نیمه نامتناهی است نمی توانیم از فرمول اویلر برای آن استفاده کنیم. به این منظور یک راس فرضی به نام v_∞ را در بینهایت در نظر می گیریم و به مجموعه رئوس اضافه می کنیم سپس تمام یالهای نیمه نامتناهی را به آن متصل می کنیم. حال یک گراف مسطح همبند داریم و می توانیم از فرمول اویلر استفاده کنیم (شکل ۲-۲).



شکل ۲-۳: گراف همبند و مسطح شده گراف ورونوی با یالهای نیمه متناهی و راس بینهایت

رابطه زیر بین n_v ، تعداد رئوس گراف ورونوی، n_e تعداد یالهای $V(P)$ و n ، تعداد سرویس دهنده ها، برقرار است:

$$(n_v + 1) - n_e + n = 2 \quad (2-4)$$

علاوه بر این هر یال در گراف مورد بحث دارای دقیقاً دو راس است. بنابراین اگر درجه همه رئوس را با هم جمع کنیم برابر خواهد شد با دو برابر تعداد یالها. چون هر راس (به انضمام راس v_∞)، حداقل از درجه ۳ است، لذا داریم:

$$2n_e \geq 3(n_v + 1) \quad (2-5)$$

لذا با استفاده از رابطه (۲-۵) داریم:

بنابراین $n_v \leq 2n - 5$ و $2n_e \geq 2(2 - n + n_e)$ پس $n_e \leq 3n - 6$ لذا

قضیه حاصل می شود. $\square$

قضیه ۲-۵ ([۳۸]): نتایج زیر برای گراف ورونوی $V(P)$ از مجموعه نقاط P برقرارند:

(i) نقطه q یک راس از $V(P)$ است اگر و تنها اگر حداقل سه نقطه روی محیط بزرگترین دایره خالی $C_p(q)$ درون $V(P)$ قرار گیرند.

(ii) عمود منصف p_i, p_j یک یال $V(P)$ است اگر و تنها اگر نقطه ای مانند q روی عمود منصف موجود باشد که هیچ نقطه دیگری از P غیر از p_i, p_j روی محیط آن قرار نگیرد.

برهان. (i) فرض می کنیم که q نقطه ای باشد که $C_p(q)$ شامل حداقل سه عضو از P روی محیط اش باشد. این نقاط را p_i, p_j, p_k در نظر می گیریم. از آنجائیکه درون $C_p(q)$ خالی است، q باید روی مرز هر یک از $V(p_i), V(p_j), V(p_k)$ قرار گیرد و لذا q یک راس از $V(P)$ است.

برای اثبات عکس، هر راس q از $V(P)$ با حداقل سه یال مجاور است یعنی با حداقل سه خانه گراف ورونوی، $V(p_i), V(p_j), V(p_k)$ مجاور است. راس q باید از سه نقطه p_i, p_j, p_k به یک فاصله باشد زیرا در غیر اینصورت $V(p_i), V(p_j), V(p_k)$ در q مشترک نخواهند شد. بنابراین دایره ای که p_i, p_j, p_k روی محیط آن قرار می گیرند شامل هیچ عضو دیگری از P در درون خود نیست.

(ii) فرض می کنیم که q دارای ویژگی های بیان شده در قضیه باشد چون $C_p(q)$ شامل هیچ عضوی از P در درونش نیست و p_i, p_j روی محیط آن قرار دارند، داریم $d(q, p_i) = d(q, p_j) \leq d(p_i, p_k)$ برای $1 \leq k \leq n$. لذا نتیجه می شود که q روی یک یال یا راس از $V(P)$ است. قسمت اول قضیه نشان می دهد که q نمی تواند یک راس باشد. پس q روی یک یال از $V(P)$ قرار دارد که همان عمود منصف p_i, p_j است.

برای عکس فرض می کنیم که عمود منصف p_i, p_j یک یال ورونوی را تشکیل دهد. بزرگترین دایره خالی از هر نقطه ای مثل q در درون این یال باید شامل p_i, p_j روی محیط خود باشد و هیچ عضو دیگری از P روی آن قرار نمی گیرد. $\square$

۲-۳-۲ مثلث بندی دلانی

فرض کنیم $P = \{p_1, \dots, p_n\}$ مجموعه ای از نقاط روی صفحه باشد. افراز S را افراز مسطح ماکزیمال می گویند هرگاه با افزودن یک یال به S گراف حاصل مسطح نباشد. یک مثلث بندی از P را یک افراز مسطح ماکزیمال که رئوس آن از P باشند، تعریف می کنیم. دوگان گراف ورونوی را با G نشان می دهیم. دوگان گراف G ، گرافی است که متناظر با هر وجه از G یک راس و متناظر با هر دو وجه مجاور در G ، یک یال دارد. اگر گراف G را با یالهای مستقیم (پاره خط های راست و بدون انحناء) در نظر بگیریم، گراف حاصل را گراف دلانی نامیده و با $DG(P)$ نشان می دهیم. نقاط و یال های مثلث بندی دلانی به ترتیب متناظر با مولد های گراف ورونوی و پاره خط های واصل مولد های همسایه است. در حقیقت یک تناظر یک به یک بین رئوس $V(P)$ و وجوه کراندار G وجود دارد. در شکل ۱-۲ نمودار ورونوی (خطوط ضخیم) و مثلث بندی دلانی (خط چین ها) تولید شده توسط نقاط تصادفی در مستطیل واحد را مشاهده می کنید.

۳-۳-۲ الگوریتم مثلث بزرگ مثلث کوچک

الگوریتمی که در ادامه می آید را می توان برای نواحی که اجتماعی از چند ضلعی های محدب هستند به کار برد. هر ناحیه ای را می توان با دقت دلخواه به وسیله اجتماعی از چند ضلعی های محدب تقریب زد. ممکن است بعضی از نقاط تقاضا بیرون ناحیه شدنی باشد که از آنها صرف نظر می شود.

۱- مرحله مثلث بندی (فاز ۱)

ناحیه شدنی اجتماعی از چند ضلعی های محدب است. هرچند ضلعی را با استفاده از رئوس و نقاط تقاضایی که درون آن قرار می گیرند مثلث بندی می کنیم. نقاط تقاضایی که درون چند ضلعی ها قرار ندارند را در مثلث بندی استفاده نمی کنیم.

۲- مرحله ارزیابی (فاز ۲)

مقدار تابع هدف را در مرکز (مرکز ثقل سه راس با وزن های برابر) هر مثلث محاسبه می شود. کمترین مقدار تابع هدف در مراکز مثلث ها را به عنوان کران بالای اولیه UB منظور می کنیم.

۳- یک کران پایین برای تابع هدف در هر مثلث LB محاسبه می شود.

۴- از تمام مثلث هایی که کران پایین شان بزرگتر از $\frac{UB}{1+\varepsilon}$ باشد صرف نظر می کنیم. مثلث های باقیمانده، مجموعه اولیه مثلث ها T_0 را تشکیل می دهند.

۵- مرحله شاخه و کران (فاز ۳)

کوچکترین کران پایین را پیدا می کنیم ($LB_{\min}$). اگر $LB_{\min} \geq \frac{UB}{1+\varepsilon}$ ، الگوریتم متوقف می شود، UB جواب است.

۶- مثلی که کوچکترین کران پایین را دارد، $LB_{\min}$ ، به چهار مثلث مجزا افراز می شود. رئوس این چهار مثلث شامل رئوس مثلث اصلی و مرکز اضلاع آن است. این چهار مثلث همگی متشابه با مثلث اصلی هستند و اضلاعشان نصف اضلاع متناظر در مثلث اصلی است.

۷- یکی از مثلث ها جایگزین مثلث اصلی شده و سه مثلث دیگر به لیست مثلث ها اضافه می شوند.

۸- مقدار تابع هدف در مرکز هر یک از چهار مثلث کوچک محاسبه می شود و کران بالا در صورت نیاز بروز می شود.

۹- یک کران پایین برای هر مثلث کوچک محاسبه شود.

۱۰- تمام مثلث های درون مجموعه T که کران پایین شان بزرگتر از $\frac{UB}{1+\epsilon}$ است را کنار می گذاریم. (اگر UB تغییر کرد، تمام مثلث ها باید مورد بررسی قرار گیرند) به مرحله ۵ بازگرد.

❖ نکاتی در مورد الگوریتم که باید به آنها توجه کنیم :

۱- مثلث بندی هر چند ضلعی، مجموعه ای از مثلث ها را به دست می دهد.

۲- اجتماع همه مجموعه های مثلث ها، T ، افزای از ناحیه شدنی به مثلث را تشکیل می دهد که هیچ نقطه تقاضایی در درون مثلث ها وجود ندارد. بعضی از رئوس و اضلاع ممکن است به چند مثلث تعلق داشته باشند.

۳- در مثلث بندی ممکن است بعضی از مثلث ها دراز و باریک باشند. کران ها در درون چنین مثلث هایی ممکن است دارای دقت کافی نباشند در این حالت می توان مثلث های دراز و باریک را (یک زاویه بزرگتر از ۱۲۰ درجه دارند) به چهار مثلث کوچکتر تقسیم کرد. برای این منظور بلند ترین ضلع را به چهار قسمت مساوی تقسیم کرده و از راس مقابل آن به این نقاط وصل می کنیم.

۴- بیشترین تعداد مثلث های ایجاد شده در طول فرآیند شاخه و کران را با $T_{\max}$ نشان می دهیم.

۵- چند ضلعی های نامحدب را به راحتی با اجتماعی از چند ضلعی های محدب می توان به دست آورد و لذا به سادگی می توان آن ها را مثلث بندی کرد.

۴-۳-۲ محاسبه کران پایین در یک مثلث

مثلی را در نظر بگیرید که رئوس مجموعه $T = \{T_1, T_2, T_3\}$ را می سازند. می خواهیم یک کران پایین برای تابع هدف در این مثلث پیدا کنیم. مسائل و فرمولبندی آنها در ادامه آمده اند.

(۱) مسئله مکانیابی سرویس دهنده مضر

۱-۱ کران پایین اول: فرض که $D_i^T = \max_{1 \leq j \leq 3} \{d_i(T_j)\}$ ، که در آن $d_i(T_j)$ فاصله راس j ام از

مشتری (نقطه تقاضا) i ام است. کران پایین LB_i ، برای تمام نقاط مثلث T به صورت

زیر است :

$$LB_i = \sum_{i=1}^n \frac{w_i}{[D_i^T]} \quad (۶-۲)$$

۱-۲ کران پایین دوم، تابع $\frac{1}{x}$ را برای $x > 0$ نزدیک $x = a$ در نظر بگیرید. چون $\frac{1}{x}$ تابعی

محدب است لذا خط مماس در $x = a$ زیر تابع قرار می گیرد.

$$\frac{1}{x} \geq \frac{1}{a} - \frac{1}{a^2}(x-a) = \frac{1}{a} \left(2 - \frac{x}{a} \right) \quad (۷-۲)$$

$$\frac{1}{x} \geq \frac{1}{a} \left(2 - \frac{x}{a} \right) \quad \text{لم ۶ : ([۳۳])}$$

اثبات . چون $\frac{1}{x} - \frac{2}{a} + \frac{x}{a^2} \geq 0$ و $x > 0$ لذا داریم $\frac{1}{x} \left(2 - \frac{x}{a} \right) \geq 0$. با باز کردن جمله دوم نتیجه

$$\frac{1}{x} \geq \frac{2}{a} - \frac{x}{a^2} = \frac{1}{a} \left(2 - \frac{x}{a} \right) \quad \text{حاصل می شود:}$$

فرض که a_i فاصله بین نقطه تقاضای i ام و مرکز ثقل مثلث باشد. توجه داریم که هر نقطه درون مثلث می تواند مانند مرکز، سرویس دریافت کند. مرکز را برای راحتی بدون احتساب وزن محاسبه

می کنیم. با به کار بردن لم ۶ برای $x = d_i^*(x)$ ، $a = a_i^*$ داریم :

$$F(x) = \sum_{i=1}^n \frac{w_i}{d_i^r(x)} \geq \sum_{i=1}^n \frac{w_i}{a_i^r} \left(r - \frac{d_i^r(x)}{a_i^r} \right) \quad (۸-۲)$$

سمت راست معادله بالا یک تابع مقعر است بنابراین مینیمم خود را روی یکی از رئوس مثلث می گیرد. تعریف می کنیم :

$$L_j = \sum_{i=1}^n \frac{w_i}{a_i^r} \left(r - \frac{d_i^r(T_j)}{a_i^r} \right) \quad j = ۱, ۲, ۳ \quad (۹-۲)$$

$$LB_r = \min_{1 \leq j \leq 3} \{L_j\} \quad \text{و کران پایین را به دست می آوریم}$$

کران پایین LB_r کارا تر است و به طور چشمگیری زمان اجرا و تعداد مراحل اجرا و تعداد مثلث ها را کاهش می دهد. در اکثر مواقع LB_r بهتر از LB_1 عمل می کند. اما به دلیل سادگی محاسبه LB_1 پیشنهاد می شود که از $\max\{LB_1, LB_r\}$ به عنوان کران پایین استفاده شود.

۲) مسئله مکانیابی وبر با جاذبه و دافعه

برای این مسئله چندین کران پایین ساخته می شود. یک کران پایین برای F^+ و یک کران بالا برای F^- معمولاً مینیمم F^+ و ماکزیمم F^- در نقاط متفاوتی رخ می دهند و استفاده از دو کران نیز موثر و دارای دقت کافی نیست. در مسئله ای که با $n = ۲۰$ نقطه و دقت ۱۰^{-4} به بیش از ۵۰۰,۰۰۰ مثلث در طی فرآیند شاخه و کران احتیاج است. نتایج حاصل از اجرای الگوریتم نشان می دهند که استفاده از یک کران پایین برای تفاضل دو تابع محدب در مقایسه با استفاده از کران پایین های به دست آمده از هر کدام بهتر است. هر دو تابع F^+, F^- محدب هستند. نقطه (a, b) در مثلث را در نظر بگیرید. هر کدام از این توابع می توانند از پایین با خط مماس کران دار شوند که چون این کران خطی است، هم محدب است و هم مقعر. هدف ساختن یک تابع مقعر به عنوان کران پایین است. ساختن یک کران پایین محدب فرآیند را پیچیده تر می کند، زیرا مینیمم یک تابع محدب در مثلث

می تواند هر جایی در درون آن قرار گیرد در حالی که ساخت یک تابع مقعر برای کران پایین، کار را ساده ترمی کند زیرا می نیمم در یکی از سه راس مثلث به دست می آید (همانند LB_p).

چون F^+ تابعی محدب است لذا بنابر بسط تیلر تابع داریم :

$$F^+(x, y) \geq \bar{F}(x, y) = F^+(a, b) + \frac{\partial F}{\partial x} \Big|_{(a,b)} [x - a] + \frac{\partial F}{\partial y} \Big|_{(a,b)} [y - b] \quad (10-2)$$

با جایگذاری مشتقات در فرمول داریم :

$$\bar{F}(x, y) = F^+(a, b) + \sum_{i=1}^n w_i \frac{(x - a_i)}{d_i(x, y)} \Big|_{(a,b)} [x - a] + \sum_{i=1}^n w_i \frac{(y - b_i)}{d_i(x, y)} \Big|_{(a,b)} [y - b] \quad (11-2)$$

بنابراین $F(x, y) = F^+(x, y) - F^-(x, y) \geq \bar{F}(x, y) - F^-(x, y)$ پس $F(x, y) \geq \bar{F} - F^-$.

سمت راست نامعادله بالا یک تابع مقعر است بنابراین مینیمم روی مرز قرار دارد. چون نقاط روی اضلاع مثلث ها ترکیب محدب رئوس متناظر با آن هستند لذا تابع هدف می نیمم خود را روی رئوس مثلث اختیار می کند. تعریف می کنیم : $j = 1, 2, 3$ و $L_j = \bar{F}(T_j) - F^-(T_j)$ و کران پایین زیر را به دست می آوریم:

$$LB_p = \min_{1 \leq j \leq 3} \{L_j\} \quad (12-2)$$

فصل ۳

الگوریتم بهینه سازی پرندگان

۱-۳ مقدمه

طبیعت و روند تکامل موجودات در آن همواره انگیزه ای برای پژوهشگران بوده تا با الگو گرفتن از آن روشهایی را به منظور حل مسائل بهینه سازی طراحی کنند. بهینه سازی بخش جدایی ناپذیر از زندگی انسانهاست به طوری که بسیاری از شاخه های علمی نظیر اقتصاد، مهندسی، علوم پایه و حتی برخی از گرایشهای علوم اجتماعی برای رسیدن به اهداف خود، ناگزیر به استفاده از روش های بهینه سازی و جست و جو می باشند. روش های بهینه کلاسیک به دو دسته تقسیم می شوند:

(۱) روش های جست و جوی مستقیم

(۲) روش های مبتنی بر گرادیان

روش های جست و جوی مستقیم از تابع هدف و محدودیت ها برای جست و جو استفاده می کنند. این روش ها معمولاً حرکت و روند آهسته ای دارند و برای همگرایی احتیاج به محاسباتی زیادی دارند. اما با وجود این می توان از این روش ها بدون تغییر عمده در ساختارشان در طیف گسترده ای از مسائل استفاده کرد. روش های مبتنی بر گرادیان از مشتق مرتبه اول یا دوم تابع هدف و قیود استفاده می کنند. این روش ها بر خلاف روش های مستقیم در اطراف جواب بهینه، همگرایی سریعی دارند ولی با این حال در مسائل مشتق ناپذیر ناپیوسته کارایی مناسبی ندارند.

با گسترش نظریات تکامل در طبیعت در سال های نه چندان دور، علاوه بر روش های کلاسیک روش های دیگری با تقلید از اصول حاکم بر طبیعت، به منظور حل مسائل بهینه سازی ابداع شدند. این روش ها که الگوریتم های تکاملی (EAs) نامیده می شوند، روش هایی تصادفی و مبتنی بر جمعیت و غیر کلاسیک می باشند که با تکیه بر ترکیب تصادفی دو عامل تغییر و انتخاب به حل مسائل بهینه سازی می پردازند. الگوریتم های تکاملی بر خلاف روش های کلاسیک، به جای استفاده از یک جواب،

جمعیتی از جواب ها را در فرآیند جست و جوی خود مورد استفاده قرار می دهند و با استفاده از عملگر های خود سعی در رسیدن به جواب بهینه دارند. علاوه بر تمام موارد ذکر شده، مزیت اصلی (EAs) این است که در فرآیند خود، احتیاجی به مشتق پذیری و یا حتی پیوستگی تابع هدف و قیود مسئله بهینه سازی ندارند.

یکی از اعضای خانواده (EAs)، الگوریتم بهینه سازی پرندگان^۱ (PSO) است که با الهام گرفتن از اصول حاکم بر رفتار گونه های اجتماعی موجود در طبیعت از قبیل پرندگان و ماهی ها، در سال ۱۹۹۵ توسط دکتر کندی و ابرهارت [۴۰] ابداع گردید.

همانطور که قبلاً اشاره شد PSO یک روش بهینه سازی تصادفی مبتنی بر جمعیت است که براساس همکاری اجتماعی و رقابت گروهی و بهبود اعضای جمعیت عمل می کند. جمعیت PSO را گروه^۲ و اعضای آن را ذره^۳ می نامیم. هر ذره در آغاز فرآیند PSO به صورت تصادفی در فضای جست و جو تولید می شود. گروه PSO دارای یک حافظه جمعی است به طوریکه هر ذره همواره بهترین موقعیتی را که خود کسب کرده و همچنین بهترین موقعیتی را که گروه پیدا کرده، به یاد داشته و بر اساس ترکیبی از این دو به عنوان عامل جهت دهی خود (سرعت) به حرکت در فضا و جست و جو می پردازد. بنابراین "هوش اجتماعی"^۴ بهترین تعبیری است که می توان راجع به PSO عنوان کرد.

PSO به خاطر داشتن قواعد پیاده سازی راحت و همچنین عدم استفاده از اطلاعات گرادیان تابع هدف و قیود و سرعت در محاسبات، محبوبیت فراوانی کسب کرده است.

۲-۳ الگوریتم بهینه سازی دسته ذرات

Particle Swarm Optimization ۷۱

Swarm^۲

Particle^۳

Social Intelligence^۴

فرض کنیم که X ، فضای جست و جو PSO، یک فضای جست و جوی n -بعدی و پیوسته باشد. هر ذره در تکرار t ام الگوریتم PSO دارای سه ویژگی می باشد:

$x(t)$: موقعیت فعلی ذره در تکرار t ام

$v(t)$: سرعت فعلی ذره در تکرار t ام

$y(t)$: بهترین موقعیت فردی ذره تا تکرار t ام

بنابراین با توجه به فضای جست و جوی n بعدی، $x(t)$ و $v(t)$ به صورت زیر می باشند:

$$x(t) = (x_1(t), x_2(t), \dots, x_n(t)) \quad (۱-۳)$$

$$v(t) = (v_1(t), v_2(t), \dots, v_n(t)) \quad (۲-۳)$$

در حالت کلی $X = \prod_{i=1}^n [l_i, u_i]$ که l_i, u_i به ترتیب کران های بالا و پایین بعد i ام فضای جستجو هستند.

بردار های اولیه مکان و سرعت را به صورت زیر تعریف می کنیم:

$$x(\cdot) = (x_1(\cdot), x_2(\cdot), \dots, x_n(\cdot)) \quad (۳-۳)$$

$$v(\cdot) = (v_1(\cdot), v_2(\cdot), \dots, v_n(\cdot)) \quad (۴-۳)$$

که در آن

$$\begin{aligned} x_j(\cdot) &\sim U[l_j, u_j] & j = 1, \dots, n \\ v_j(\cdot) &\sim U[V_{\min j}, V_{\max j}] & j = 1, \dots, n \end{aligned} \quad (۵-۳)$$

و $U[l_j, u_j]$ توزیع یکنواخت روی بازه $[l_j, u_j]$ است.

میزان شایستگی هر ذره را برابر با مقدار تابع هدف به ازای آن تعریف می کنیم و سپس هر ذره با سرعت v که مبتنی بر شایستگی ذره و سایر ذرات گروه می باشد، در فضای جست و جو حرکت می کند. بهترین موقعیت فردی ذره تا تکرار t ام، $y(t)$ بهترین مقدار شایستگی برای ذره را نتیجه می دهد و بهترین موقعیتی می باشد که ذره از ابتدا تا تکرار t ام مشاهده کرده است. فرض کنید که در تکرار t ام PSO قرار داریم. با توجه به اینکه اندازه گروه برابر با m می باشد، می توان مجموعه های $S(t)$ و $V(t)$ را به صورت زیر تعریف کرد :

$$\begin{aligned} S(t) &= \{x_1(t), x_2(t), \dots, x_m(t)\} \\ V(t) &= \{v_1(t), v_2(t), \dots, v_m(t)\} \end{aligned} \quad (۶-۳)$$

حال بر اساس تابع هدف، شایستگی هر ذره در $S(t)$ سنجیده می شود. مجموعه $F(t)$ به صورت زیر ساخته می شود :

$$F(t) = \{F(x_1(t)), F(x_2(t)), \dots, F(x_m(t))\} \quad (۷-۳)$$

حال بر اساس مجموعه $F(t)$ به راحتی می توان بهترین موقعیت فردی ذره i ام تا تکرار t ام، یعنی $y_i(t)$ را بر اساس رابطه زیر مشخص کرد :

$$y_i(t) = \begin{cases} x_i(t) & F(x_i(t)) < F(y_i(t-1)) \\ y_i(t-1) & o.w. \end{cases} \quad (۸-۳)$$

بعد از مشخص شدن $y_i(t)$ ها می توان مجموعه $P(t)$ را به صورت زیر تعریف کرد :

$$P(t) = \{y_1(t), y_2(t), \dots, y_m(t)\} \quad (۹-۳)$$

بنابراین $\hat{y}(t) = y_g(t) = \arg \min_{i=1, \dots, m} F(y_i(t))$ بهترین موقعیت سراسری پیدا شده بین تمام ذرات گروه

تا تکرار t ام می باشد.

پس از مشخص شدن مجموعه های $V(t), S(t), P(t)$ و همچنین $\hat{y}(t)$ می توان موقعیت هر ذره در $S(t)$ را بر اساس روابط زیر به روز رسانی کرد :

$$v_i(t+1) = v_i(t) + r_1 c_1 (y_i(t) - x_i(t)) + r_2 c_2 (\hat{y}(t) - x_i(t)) \quad (10-3)$$

$$x_i(t+1) = x_i(t) + v_i(t+1) \quad (11-3)$$

که در آن $r_1, r_2 \sim U(0,1)$ به منظور حفظ پراکندگی ذرات، القای خاصیت تصادفی بدون ذرات و جلوگیری از جهش ناگهانی ذرات PSO به کار برده می شوند.

c_1, c_2 ضرایب شتاب فردی و اجتماعی می باشند که مقادیر آنها بر اساس نتایج تجربی، از راه آزمون و خطا تعیین می شوند.

شای و ابرهارت [۴۱ و ۴۲] با فرض اینکه $X = \prod_{i=1}^n [l_i, u_i]$ باشد $V_{\max}, V_{\min}$ را به صورت زیر تعریف کردند،

$$\begin{aligned} V_{\max j} &= \gamma(u_j - l_j) \\ V_{\min} &= -V_{\max j} \end{aligned}$$

که γ با توزیع یکنواخت از بازه $[0,1]$ انتخاب می شود. الگوریتم PSO بر اساس روابط (۱۰-۳) و (۳-۱۱) فرآیند خود را بروز رسانی می کند تا اینکه معیار توقف حاصل شود.

هنگام بروز رسانی موقعیت ذرات PSO ممکن است حالتی اتفاق بیافتد که ذره امکان خروج از فضای جست و جوی X را پیدا کند. شای و ابرهارت به منظور جلوگیری از خروج ذرات از فضای X مقید کردن v به $[V_{\min}, V_{\max}]$ را پیشنهاد کردند [۴۳]، به طوریکه :

$$v_{ij}(t+1) = \begin{cases} V_{\max j} & v_{ij}(t+1) > V_{\max j} \\ V_{\min j} & v_{ij}(t+1) < V_{\min j} \end{cases}$$

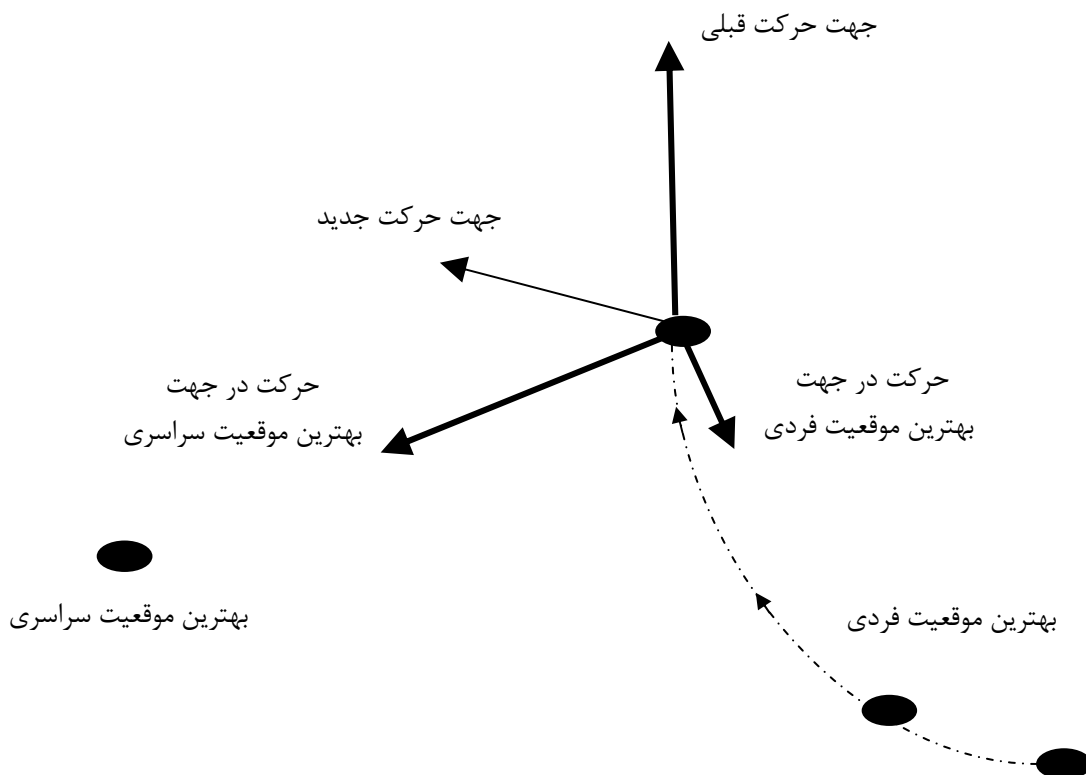
اما اگر پس از مقید کردن v ذره همچنان تمایل به خروج از فضای جست وجود داشته باشد، این بار از مقید کردن x استفاده می کنیم، به این صورت که

$$x_{ij}(t+1) = \begin{cases} x_{ij}(t+1) - \alpha & x_{ij}(t+1) > u_j \\ x_{ij}(t+1) + \beta & x_{ij}(t+1) < l_j \end{cases} \quad (12-3)$$

به طوریکه

$$\begin{aligned} \alpha &\sim U[x_{ij}(t+1) - u_j, x_{ij}(t+1) - l_j] \\ \beta &\sim U[l_j - x_{ij}(t+1), u_j - x_{ij}(t+1)] \end{aligned} \quad (13-3)$$

به منظور درک بهتر معادلات (۱۰-۳) و (۱۱-۳) حرکت ذره i ام و تکرار $t+1$ ام، $x_i(t+1)$ در شکل (۱-۳) نشان داده شده است.



شکل ۱-۳: نمایش چگونگی حرکت ذرات در الگوریتم PSO

سرعت $v_i(t)$ متناسب با جهت حرکت پیشین ذره می باشد که به طور چشمگیری از تغییر ناگهانی ذره جلوگیری کرده و حرکت ذره را به سمت جهت فعلی سوق می دهد.

$r_1 c_1 (y_i(t) - x_i(t))$ مولفه فردی می باشد که میزان بازدهی ذره i ام را نسبت به مراحل قبل تعیین می کند .

$r_2 c_2 (\hat{y}(t) - x_i(t))$ مولفه اجتماعی می باشد که میزان بازدهی ذره i ام را نسبت به کل گروه تعیین می کند .

مرحله آغازین از الگوریتم PSO شامل گام های زیر می باشد :

۱- در شروع الگوریتم، مولفه j ام از ذره i ام را با x_{ij} نشان می دهیم و با توزیع تصادفی در بازه

$$[l_j, u_j] \text{ مشخص کنید. و این روند را برای } j = 1, \dots, n, \quad i = 1, \dots, m \text{ تکرار نمایید.}$$

۲- در شروع الگوریتم، مولفه j ام بردار سرعت i ام را v_{ij} ، با توزیع تصادفی در بازه

$$[V_{\min}, V_{\max}] \text{ مشخص کنید و این روند را برای } j = 1, \dots, n, \quad i = 1, \dots, m \text{ تکرار نمایید. به}$$

جای این روش می توان سرعت اولیه همه ذرات را صفر در نظر گرفت .

۳- در شروع الگوریتم $(i = 1, \dots, m)$ $y_i = x_i$ در نظر بگیرید. همچنین می توان به جای این

روش، دو بردار تصادفی را برای هر یک از اعضا تولید کرده و برداری که شایستگی بهتری دارد

را به y_i نسبت داد . به وضوح روش دوم به محاسبات بیشتری نیاز دارد .

۱-۲-۳ شبه کد الگوریتم PSO

شبه کد الگوریتم PSO به صورت زیر است :

مرحله ۱ آغاز الگوریتم

$$t = 0 : 1-1$$

$$i = 1, \dots, m : 1-2 \quad m \text{ ذره تصادفی تولید شوند } x_i(t)$$

$$i = 1, \dots, m : 1-3 \quad m \text{ بردار سرعت تصادفی تولید شوند } v_i(t)$$

$$i = 1, \dots, m : 1-4 \quad \text{بهترین موقعیت فردی ذرات تعیین شوند } y_i(t)$$

$$\hat{y}(t) = y_g(t) = \arg \min_{i=1, \dots, m} F(y_i(t)) : 1-5$$

مرحله ۲ تا زمانی که معیار توقف حاصل نشده است مراحل زیر تکرار شوند :

۲-۱ : برای ذره i ام، محاسبات زیر انجام شود :

$$v_i(t+1) \text{ از رابطه (۳-۱۰) محاسبه شود.}$$

$$x_i(t+1) \text{ از رابطه (۳-۱۱) محاسبه شود.}$$

۲-۲ : شرط $x \in [L, U]$ بررسی شده و در صورت لزوم از رابطه های اصلاح شدنی بودن

استفاده شود و سپس $y_i(t)$ ها به روز شوند.

$$\hat{y}(t+1) = \arg \min_{i=1, \dots, m} F(y_i(t+1)) : 2-3$$

$$t = t + 1 : 2-4$$

یکی از مهمترین بخش های الگوریتم PSO معیار توقف آن است. در حالت کلی معیار توقف PSO به مسئله در دست حل بستگی دارد. در بعضی از مسائل، معیار توقف دستیابی به بیشینه تکرار مجاز از قبل تعیین شده T می باشد و حال آنکه در برخی دیگر از مسائل معیار توقف، دستیابی به آستانه خطا از پیش تعیین شده ی ε در میزان شایستگی است .

اگر فقط از معیار T به منظور توقف PSO استفاده شود، ممکن است قبل از آنکه PSO به معیار T برسد، آستانه خطا برای مقدار شایستگی $\hat{y}(t)$ حاصل شود. بنابراین استفاده تنها از معیار T ، تنها محاسبات را افزایش می دهد.

همچنین اگر فقط از معیار آستانه خطا استفاده شود ممکن است حالتی رخ دهد که الگوریتم دچار همگرایی زودرس شود و لذا اگر از معیار T استفاده نشود، آنگاه الگوریتم PSO فرصت بهبود حالت رخ داده را ندارد. معیار دیگری که در شرط توقف الگوریتم می توان استفاده کرد، تعریف یک کران برای اجرای الگوریتم بعد از آخرین بهبود است. یعنی بعد از آخرین بروز رسانی $\hat{y}(t)$ ، چند مرحله الگوریتم اجرا شد و در $\hat{y}(t)$ بهبودی حاصل نشد، الگوریتم متوقف شود.

بنابراین در طیف وسیعی از مسائل، معیار توقف بر اساس ترکیب بیشینه تکرار مجاز T یا $|F_{\min} - F_{\max}| < \varepsilon$ یا $|F_{\min}| < \varepsilon$ می باشد که در آن

$$\begin{aligned} F_{\min} &= F(\hat{y}(t)) \\ F_{\max} &= F(\hat{z}(t)) \quad \hat{z}(t) = \arg \max_{i=1, \dots, m} F(y_i(t)) \end{aligned} \quad (3-14)$$

۳-۳ نسخه های اصلاح شده الگوریتم PSO

برای الگوریتم بهینه سازی پرندگان گونه های اصلاحی مختلفی پیشنهاد شده است که برخی از آنها در زیر آمده اند:

۱- الگوریتم PSO با وزن اینرسی ثابت، [۴۲ و ۴۴].

۲- الگوریتم PSO با وزن اینرسی متغیر با زمان، [۴۳ و ۴۵ و ۴۶].

۳- الگوریتم PSO با وزن اینرسی پویا و سرعت بیشینه، [۴۷ و ۴۸].

۴- الگوریتم PSO با ضرایب شتاب متغیر با زمان، [۴۹].

۵- الگوریتم PSO با ضرایب شتاب پویا، [۵۰].

۶- الگوریتم PSO با عامل انقباض، [۵۰ و ۵۱].

۷- الگوریتم PSO مبتنی بر توپولوژی های خاص، [۵۱ و ۵۲].

۸- الگوریتم PSO با همگرایی تضمین شده، [۵۳].

۹- الگوریتم PSO با مشارکت، [۵۴].

۱۰- الگوریتم PSO برای پیدا کردن بهینه چندگانه، [۵۵-۵۷].

۱۱- الگوریتم PSO برای مسائل بهینه سازی چند هدفه، [۵۸-۶۰].

۱۲- الگوریتم PSO برای مسائل گسسته، [۶۱].

۱۳- الگوریتم PSO برای مسائل بهینه سازی مقید، [۶۲-۶۷].

۱۴- الگوریتم PSO ترکیبی با فرآیند شبیه سازی تبریدی، [۶۵].

در این پایان نامه، تنها الگوریتم PSO استاندارد و الگوریتم PSO با عامل انقباض را که از مهمترین این نسخه ها هستند، بیان می کنیم و در ادامه توضیح مختصری از روش ترکیبی بهینه سازی ذرات و فرآیند شبیه سازی تبریدی ارائه می شود.

۳-۳-۱ الگوریتم PSO استاندارد (PSO-S)

در الگوریتم PSO اولیه، جملات $(\hat{y}(t) - x(t))$ ، $(y(t) - x(t))$ دو معادله بروز رسانی سرعت، جملاتی وزن دار بوده و به ترتیب از r_1c_1 ، r_2c_2 به منظور کنترل میزان بازدهی ذره نسبت به y و $\hat{y}$ استفاده

می شود. از طرف دیگر همانطور که در رابطه (۳-۱۰) مشاهده می شود، $v(t)$ هیچ گونه عامل وزنی ندارد. شای و ابرهات [۴۲ و ۴۴]. به منظور کنترل تاثیر سرعت ذره i ام از تکرار قبلی در رابطه بروزرسانی سرعت، از ضریبی تحت عنوان وزن اینرسی، w ، برای $v(t)$ استفاده کردند. w یک مقیاس اسکالری می باشد که در ابتدا به صورت ثابت معرفی شد. حال با در نظر گرفتن w رابطه بروزرسانی سرعت به صورت زیر تبدیل می شود:

$$v_i(t+1) = wv_i(t) + r_1c_1(y_i(t) - x_i(t)) + r_2c_2(\hat{y}(t) - x_i(t)) \quad (3-15)$$

جمله $wv_i(t)$ ، مولفه اینرسی نام دارد و الگوریتم PSO با رابطه (۳-۱۵) را الگوریتم PSO استاندارد می نامند. اگر $w > 1$ آنگاه با افزایش t ، ذره تمایل به سرعت بیشینه پیدا می کند و در حرکت خود به جهت حرکت قبلی توجه بیشتری می کند. لذا ذره حرکتی یکنواخت پیدا کرده و دچار رکود رفتاری می شود. اگر $w < 1$ ، با افزایش t ، ذره تمایل به کاهش سرعت پیدا می کند و لذا عامل پراکندگی و جست وجوی گسترده آن ضعیف شده و احتمال اینکه همگرایی زودرس رخ دهد، بسیار زیاد می باشد.

اگر در آغاز الگوریتم مقادیر c_1, w بزرگ بوده و c_2 کوچک باشد آنگاه ذره به جای حرکت به سوی $\hat{y}(t)$ به جست وجو در فضا می پردازد. اگر در ادامه روند حرکت ذرات، از مقادیر اولیه c_1, w کاسته شود و به c_2 افزوده شود آنگاه مولفه فردی و اینرسی سعی در کاهش خود داشته و ذرات حرکت خود را برای رسیدن به $\hat{y}$ تنظیم می کنند.

شای و ابرهات [۴۶]، وزن اینرسی. متغیر با زمان (TVIW) و ضرایب شتاب متغیر با زمان (TVAC) را به صورت زیر معرفی کردند.

$$w = w(t) = w_i - (w_i - w_r) \frac{t}{T}$$

$$c_i = c_i(t) = c_{vi} - (c_{vi} - c_{vf}) \frac{t}{T}$$

$$c_r = c_r(t) = c_{ri} - (c_{ri} - c_{rf}) \frac{t}{T}$$

w_i, w_r مقادیر آغازی و پایانی w ، $c_{vi}, c_{vf}, c_{ri}, c_{rf}$ مقادیر آغازی و پایانی c_i, c_r هستند t نشان دهنده تکرار فعلی و T برابر با بیشینه تکرار مجاز در الگوریتم است.

۲-۳-۳ الگوریتم PSO با عامل انقباض

کلرک و کندی [۵۰]، معادلات بروزرسانی دیگری را پیشنهاد کردند که بر آنها دامنه حرکت ذرات کنترل شده و همگرایی حرکت ذرات PSO تضمین می شود. در معادلات آنها از پارامتر χ به صورت زیر استفاده شده است:

$$\chi = \frac{2}{|2 - \phi - \sqrt{\phi^2 - 4\phi}|}$$

که در آن $\phi > 4$ یک پارامتر است. χ هر سه عامل دخیل در بروزرسانی سرعت را بطور همزمان کنترل می کند. χ را عامل انقباض و الگوریتم PSO یا این پارامتر را الگوریتم PSO با عامل انقباض می نامند و با PSOC نشان می دهند. این نسخه از الگوریتم کارایی زیادی در حل مسائل دارد ولی بخاطر محاسبات زیادی که دارد زمان بیشتری را مصرف می کند.

۳-۳-۳ الگوریتم ترکیبی PSO و فرآیند شبیه سازی تبریدی

شبیه سازی تبریدی فلزات، یک الگوریتم بهینه سازی کارا می باشد که بر اساس شبیه سازی گرم و سرد کردن فلزات و استراتژی حل مسائل بهینه سازی ترکیبیاتی شکل گرفته است. این فرآیند مکانیزمی دارد که از به دام افتادن در بهینه موضعی جلوگیری می کند. از لحاظ تئوری می توان ثابت

کرد که فرآیند شبیه سازی تبریدی با برقراری شرایط خاصی با احتمال ۱ به طور سراسری همگرا می شود.

فرآیند شبیه سازی تبریدی دارای خاصیت به نام پرش^۵ می باشد که به آن کمک می کند تا از دام بهینه موضعی فرار کند. در روش ترکیبی، به منظور جستجوی بهتر و کاراتر، از فرآیند شبیه سازی تبریدی برای جستجوی موضعی $\hat{y}$ استفاده می شود. جستجوی موضعی به کمک فرآیند شبیه سازی تبریدی برای $\hat{y}$ به صورت زیر می باشد:

مرحله ۱: فرض کنید $i = 1$, $p'_g = \hat{y}(k)$

مرحله ۲: بر اساس معادله زیر، یک جواب جدید تولید کنید:

$$x' = p'_g + \eta(x_{\max} - x_{\min}) \times N(0,1) \quad (16-3)$$

مرحله ۳: p_a را بر اساس قاعده زیر محاسبه کنید:

(۱) اگر x' شدنی و p'_g نشدنی باشند آنگاه $p_a = 1$

(۲) اگر p'_g شدنی و x' نشدنی باشند آنگاه $p_a = 0$

(۳) اگر p'_g و x' شدنی باشند آنگاه

$$p_a = \min \left\{ 1, \exp \left(\frac{f(p'_g) - f(x')}{t(k)} \right) \right\} \quad (17-3)$$

(۴) اگر p'_g و x' نشدنی باشند آنگاه

$$p_a = \min \left\{ 1, \exp \left(\frac{viol(p'_g) - viol(x')}{t(k)} \right) \right\} \quad (18-3)$$

مرحله ۴: اگر $p_a \geq U[0,1]$ آنگاه $p'_g = x'$

مرحله ۵: $i = i + 1$

مرحله ۶: اگر $i > L$ آنگاه توقف کنید و p'_g را به عنوان $\hat{r}$ در نظر بگیرید، در غیر اینصورت برو به مرحله ۲.

در معادله (۳-۴)، η پارامتر طول گام می باشد و $x_{\min}, x_{\max}$ کرانهای بالا و پایین جوابهای تعریف شده در مسئله می باشند. $N(0,1)$ متغیر تصادفی نرمال استاندارد است.

در معادله (۳-۱۷)، $t(k)$ دمای تکرار k ام می باشد. در مرحله ۴، $U[0,1]$ توزیع یکنواخت در بازه $[0,1]$ است. در مرحله ۶، L تعداد تکرارهای مورد استفاده جهت توقف جستجوی موضعی می باشد که توسط کاربر تعیین می شود. معمولاً مقادیر زیر را برای آنها در نظر می گیرند:

$$L = 20, \quad \lambda = 0.94, \quad \eta = 0.001$$

در این الگوریتم، دمای اولیه بر اساس رابطه زیر مشخص می شود:

$$t(0) = -\frac{f_{\max} - f_{\min}}{\ln(0.1)}$$

که در آن

$$\begin{aligned} f_{\max} &= \max\{f(x_i(0)) \mid i = 1, \dots, m\} \\ f_{\min} &= \min\{f(x_i(0)) \mid i = 1, \dots, m\} \end{aligned}$$

بعلاوه

$$t(k+1) = \lambda t(k), \quad k = 0, 1, 2, \dots$$

که λ نرخ سرد و گرم کردن می باشد و $0 < \lambda < 1$

کد برنامه این روش برای حل مسئله مکانیابی آرمانی و یافتن مقادیر ویژه ماتریسهای غیر هرمیتی و با مرتبه بالا، در ضمیمه متن برنامه ها آمده است.

۳-۳-۴ شبه کد الگوریتم ترکیبی PSO و فرآیند شبیه سازی تبریدی

مرحله ۱ آغاز الگوریتم

$$t(k) = t, \quad k = 0 : 1-1$$

$$i = 1, \dots, m \quad 1-2: \text{ذره تصادفی تولید شوند } x_i(k)$$

$$i = 1, \dots, m \quad 1-3: \text{م بردار سرعت تصادفی تولید شوند } v_i(k)$$

$$i = 1, \dots, m \quad 1-4: \text{بهترین موقعیت فردی ذرات تعیین شوند } y_i(k)$$

$$\hat{y}(k) = y_g(k) = \arg \min_{i=1, \dots, m} F(y_i(k)) \quad 1-5:$$

مرحله ۲ تا زمانی که معیار توقف حاصل نشده است مراحل زیر تکرار شوند:

۲-۱: برای ذره i ام، محاسبات زیر انجام شود:

$$v_i(k+1) \text{ از رابطه (۳-۱۰) محاسبه شود.}$$

$$x_i(k+1) \text{ از رابطه (۳-۱۱) محاسبه شود.}$$

۲-۲: شرط $x \in [L, U]$ بررسی شده و در صورت لزوم از رابطه های اصلاح شدنی بودن

استفاده شود و سپس $y_i(k)$ ها به روز شوند.

۲-۳: جستجوی موضعی را برای $\hat{y}(k+1)$ بر اساس فرآیند شبیه سازی تبریدی به کار گیرید.

$$k = k + 1, \quad t(k+1) = \lambda t(k) \quad 2-4:$$

فصل ۴

مکانیابی سرویس دهنده نیمه ناخوشایند

۱-۴ مقدمه

دردو سه دهه اخیر، توجه و علاقه مندی به مسائل محیطی و دنیای واقعی افزایش پیدا کرده است. این مسائل عمدتاً در نتیجه پیشرفت های صنعتی و تکنولوژی ایجاد شده اند و غالباً تأثیرات منفی در محیط را نیز به دنبال دارند و مشکلات زیست محیطی به بار می آورند. مسائلی مانند استقرار یک سرویس دهنده، فرآیند یا وسیله که در عین تأمین خدمات و تقاضای مشتریان اثراتی مثل سرو صدا، گرد و خاک، دود، گاز های سمی، بوی بد، شلوغی و مزاحمت هایی از این نوع را بر مشتریان تحمیل می کنند، از این دسته هستند. طراحان شهری، و در حالت کلی تصمیم گیرندگان در عمل با مسائل پیچیده ای نظیر انتخاب محل استقرار یک مرکز آتش نشانی، پاسگاه پلیس، فرودگاه، کارخانه ها، نیروگاه ها، آزمایشگاه های شیمیایی، مراکز هسته ای، فروشگاه های زنجیره ای، مراکز اورژانس، بیمارستان ها، اماکن دفن زباله و زباله سوزی، پایانه های حمل و نقل، تصفیه خانه های آب، و غیره روبرو هستند که همگی در حوزه مکانیابی سرویس دهنده نیمه ناخوشایند قابل بررسی و مطالعه هستند. مقالات مهمی در زمینه مکانیابی سرویس دهنده نیمه نامطلوب به چاپ رسیده است که از آن جمله می توان به [۶۹و۶۸و۹۶] اشاره کرد. در مسائل مکانیابی کلاسیک غالباً سرویس دهنده ها مفید فرض می شوند و دارای خاصیت جاذبه هستند و لذا در آنها مکان بهینه سرویس دهنده جدید، تا حد امکان نزدیک به مکان مشتریان است. در این مدل ها از اثرات منفی و محدودیت های ایجاد شده توسط سرویس دهنده صرف نظر می شود. در یک دسته بندی این اثرات منفی را به دو دسته نامطلوب (مانند شلوغی و سروصدای ناشی از مراکز تجاری، ورزشگاه ها، پارکینگ ها و از این قبیل) و مضر (مراکز شیمیایی، نیروگاه های هسته ای) تقسیم می کنند.

در سال های اخیر تلاش هایی برای وارد کردن این جنبه های نامطلوب (مضر) در مدل های مکانیابی صورت گرفته است که از جمله می توان به کارهای ارکوت و نیومن [۹] و یا پلستریا [۶۹] اشاره کرد. بیشتر مدل هایی که در ابتدا ارائه شدند فقط یکی از دو جنبه را در نظر گرفته اند یعنی فقط به جنبه تاثیرات محیطی و یا هزینه های حمل و نقل توجه داشته اند و از دیگری صرف نظر کرده اند فلذا مدل های حاصل مطابقت کمتری با واقعیت دارند و تصمیم گیری را با اشکالاتی مواجه می سازند. مقالاتی در دهه اخیر ارائه شده اند که هر دو عامل را با هم در مدل لحاظ کرده اند و به حل مسئله پرداخته اند.

مدل های مکانیابی نیمه نامطلوب اساسا چند مدی هستند و لذا روش های استاندارد که بر مبنای آنالیز محدب برای مسائل مکانیابی سرویس دهنده مطلوب در صفحه به کار می روند، ممکن است به دام بهینه های موضعی بیافتند. از طرف دیگر به خاطر پیچیدگی هندسی این مسائل ممکن است روش های هندسه محاسباتی که برای حل مسائل مکانیابی سرویس دهنده نامطلوب به کار می روند، برای این مسائل جوابگو نباشند. برای توضیح بیشتر در این زمینه به مقاله پلستریا [۶۹] مراجعه کنید. حال به بررسی مدل های مکانیابی گسسته که در آنها هر دو جنبه مثبت و منفی سرویس دهنده در نظر گرفته شده است، می پردازیم. این مدل ها در حوزه بهینه سازی ترکیبیاتی قرار می گیرند [۷۰]. برای مرور تاریخی کارهای ابتدایی انجام شده در این باره به فصل اول مراجعه کنید.

از یک نظر مسائل مکانیابی جاذبه و دافعه را می توان به دو دسته کلی تقسیم کرد:

الف) مسائلی که در آن ها تنها ارتباط بین سرویس دهنده ها مد نظر است. بعضی سرویس دهندگان جاذب و بعضی دیگر دافع هستند. این مدل ها را مدل های شبه پراکنده^۱ می نامیم.

ب) مسائلی که دسته ای از مشتریان (I)، سرویس دهنده را به سوی خود جذب می کنند و دسته ای دیگر (K)، سرویس دهنده را از خود دور می کنند. ساده ترین راه برای مدلسازی این مسائل اینست که مسئله را به صورت دو هدفه با اهداف متضاد جاذب و دافع فرموله کنیم. در اینجا قبل از اینکه حالت‌های مختلف را بررسی کنیم متصری درباره برنامه ریزی چند هدفه و مفاهیم مرتبط با آن توضیح داده می شود.

در مدل‌های کلاسیک بهینه سازی شامل برنامه ریزی خطی، عدد صحیح، غیر خطی و یا مخلوط، تنها از یک معیار برای سنجش بهینگی استفاده می شود که هدف انتخاب شده دارای اولویت بیشتری نسبت به سایر اهداف است و از بقیه صرف نظر می شود. در دهه های اخیر توجه محققین معطوف به مدل‌های چند معیاره^۲ برای تصمیم گیری های پیچیده گردیده است. در این تصمیم گیری ها به جای استفاده از یک معیار سنجش بهینگی، از چند معیار استفاده می شود که معیار گاه هم راستا و گاه متضاد و متقابل اند. مدل‌های تصمیم گیری به دو دسته عمده تقسیم می شوند. مدل‌های چند هدفه^۳ و مدل‌های چند شاخصه^۴. مدل‌های چند هدفه به منظور طراحی بکار گرفته می شوند در حالی که مدل‌های چند شاخصه به منظور انتخاب گزینه برتر استفاده می گردند. مدل‌های چند هدفه را که مدل‌های بهینه سازی برداری نیز نامیده می شود، به صورت زیر می توان نوشت:

$$\text{Min(Max)} \quad F(X) = [f_1(X), \dots, f_k(X)]$$

S.t.

$$g_i(X) \begin{pmatrix} \leq \\ \geq \end{pmatrix} \cdot \quad i = 1, \dots, m$$

$$X \in E^n$$

^۲ Multiple Criteria Decision Making (MCDM)

^۳ Multiple Objective Decision Making (MODM)

^۴ Multiple Attribute Decision Making (MADM)

مدل فوق دارای k تابع هدف، n متغیر تصمیم و m محدودیت است. جواب ایده آل برای این مدل مجموعه مقادیر اختصاص یافته متغیرهای تصمیم x_j ($j=1, \dots, n$) است که در تمامی m محدودیت صدق کرده و همزمان k تبع هدف را بهینه کند. در اغلب مسائل واقعی به علت تقابل اهداف مدل جواب موجهی که یک تابع هدف را بهینه می کند نمی تواند موجب بهینه کردن سایر توابع هدف گردد. لذا تصمیم گیری در این وضعیت نیازمند یک جواب کارا و ترجیحی است. مفهوم جواب در مباحث تصمیم گیری با چندین هدف از اهمیت بسزایی برخوردار است و دارای تنوع بیشتری نسبت به مسائل تک هدفه می باشد. مجموعه مقادیری که به متغیرهای تصمیم اختصاص داده می شود، اعم از اینکه در مجموعه محدودیت ها صدق کند یا صدق نکند جواب نامیده می شود. جوابی که در تمامی محدودیت ها صدق کند را جواب موجه می نامیم. جواب موجهی که تمامی اهداف را همزمان در بهترین مقدار خود قرار می دهد، جواب ایده آل نامیده می شود. در مسائل چند هدفه، جواب موجه A را بر جواب موجه B مسلط می گوئیم هر گاه حداقل بر مبنای تمامی اهداف به خوبی B بوده و دست کم بر اساس یک هدف بهتر از B باشد. جواب موجه A را کارا گوئیم هرگاه جواب موجه دیگری وجود داشته باشد که حداقل بر اساس یک هدف بهتر از A بوده و بر اساس سایر اهداف به خوبی A نباشد. جواب کارا را جواب غیر مسلط، جواب بهینه پارتو و جواب موثر نیز می نامند. جواب غیر مسلطی که توسط تصمیم گیرنده و با توجه به معیارهای دیگر انتخاب و به عنوان جواب نهایی بکار برده می شود را جواب ترجیحی می نامند. برای حل و ارزیابی مدلهای چند هدفه روشهای مختلفی وجود دارد که بستگی به زمان و نوع اطلاعاتی دارند که از تصمیم گیرنده کسب می گردد. اینگونه اطلاعات همگی به منظور بر آورد تابع مطلوبیت برای تصمیم گیرنده است بطوری که از طریق تعامل بین تصمیم گیرنده و تحلیلگر کسب می گردد. راههای مختلف کسب اطلاعات از تصمیم گیرنده برای ارزیابی یک مدل چند هدفه بصورت زیر است:

الف) عدم دسترسی به کسب اطلاعات از تصمیم گیرنده

ب) گرفتن اطلاعات اولیه از تصمیم گیرنده قبل از حل مسئله

ج) گرفتن اطلاعات به صورت تعاملی در ضمن حل مسئله

د) گرفتن اطلاعات نهایی از تصمیم گیرنده بعد از حل مسئله.

برای هر یک از این حالات روشهای مختلفی برای حل مسئله چند هدفه پیشنهاد شده است. برای مطالعه بیشتر در این زمینه به [۲-۳] مراجعه شود.

حالت های مختلفی که در مسائل مکانیابی با جاذبه و دافعه پیش رو داریم به قرار زیر است:

۱) برای مسئله دو هدفه، مجموعه پارتو^۵ را مشخص کنیم که برای مسائل مکانیابی چند وسیله ای کار ساده ای نیست. برای مسئله تک وسیله ای حالت پیوسته به نتایج کارهای دیاگ [۷۱] و کریزوسا [۷۲] مراجعه کنید.

۲) یک کران برای اثرات منفی در نظر گرفت و آن را به صورت قید به مسئله اضافه کرد و سپس هدف جاذب را بهینه کنیم. این مدلها تبدیل به مسائل استاندارد با اهداف جاذب می شوند که تعدادی محدودیت اضافی دارند. ممکن است این محدودیت ها فقط روی سرویس دهندگان [۷۳ و ۷۲] یا فقط روی مشتریان باشند [۷۴-۷۶].

۳) یک کران برای قسمت جاذب هدف در نظر گرفته و قسمت دافع را بهینه کنیم. این مدل ها تبدیل به مسائل استاندارد با هدف دافع می شوند که یک محدودیت روی میزان جاذبه دارند.

۴) قسمت های جاذب و دافع را در یک تابع ترکیب کنیم.

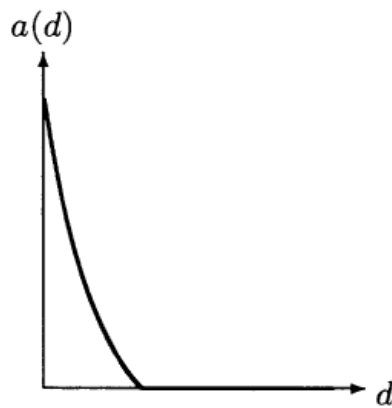
۵) وقتی $I = K$ مدل هایی به دست می آیند که ایسلت و لاپورت [۷۷] آن ها را مدل های متعادل یا متوازن^۶ نامیده اند. این مدلها واریانس فاصله مشتریان تا سرویس دهنده ها را مینیمم می کنند. برای مطالعه بیشتر به [۷۸-۸۰] مراجعه کنید.

۶) در مدل های کلی جاذبه و دافعه مجموعه های I, K ارتباطی با هم ندارند. این مدلها کاربرد های فراوانی دارند. در فرمولبندی مدل های ترکیبیاتی که در ادامه می آیند، متغیر صفر و یک x_j را به این صورت تعریف می کنیم که مقدارش برابر با ۱ است هرگاه سرویس دهنده $j \in J$ مستقر شود، در غیر این صورت $x_j = 0$ که در آن J مجموعه سرویس دهندگان است. در تمام مسائل فرض بر این است که ظرفیت سرویس دهنده نامحدود است. این فرض وجود جواب بهینه را تضمین می کند و آن جواب این است که همه مشتریان از یک سرویس دهنده خدمات دریافت کنند. فرض دیگر اینست که مشتریان تمام تقاضای خود را تنها از یک سرویس دهنده تامین می کنند. لذا متغیر های تخصیص از نوع صفر و یک هستند. فرض کنید متغیر های y_{ij} و z_{kj} به این صورت تعریف شوند که $y_{ij} = 1$ هرگاه مشتری $i \in I$ از سرویس دهنده $j \in J$ خدمات دریافت کند، در غیر این صورت $y_{ij} = 0$ و همچنین $z_{kj} = 1$ هرگاه مشتری $k \in K$ تحت تاثیر منفی سرویس دهنده $j \in J$ قرار گیرد، در غیر این صورت $z_{kj} = 0$. هزینه استقرار سرویس دهنده j را با f_j ، هزینه دریافت خدمات مشتری $i \in I$ از سرویس دهنده $j \in J$ را با c_{ij} و میزان دافعه سرویس دهنده $j \in J$ برای مشتری $k \in K$ را با a_{kj} نشان می دهیم.

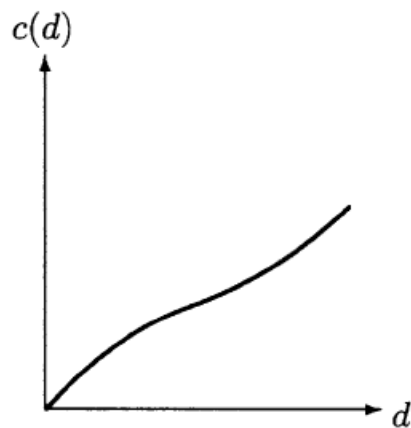
چون غالباً توابع هدف بر مبنای فاصله بیان می شوند، از دو روش برای بخش های جاذب و دافع تابع هدف مسائل مکانیابی سرویس دهنده نامطلوب می توان استفاده کرد. معمولاً جاذبه با در دسترس بودن و دافعه با اثر منفی که به طور پیوسته در فضا پخش می شود، متناظر است. بنابراین از کوتاه ترین مسیر برای قسمت جاذب و از فاصله اقلیدسی برای قسمت دافع استفاده می شود. قابل توجه

است که استفاده از کوتاه ترین مسیر برای قسمت دافع همیشه امکان پذیر نیست. مثلاً فرض کنید که قرار است یک ماده خطرناک جا به جا شود لذا کوتاه ترین مسیر لزوماً امن ترین مسیر نیست.

تأثیرات منفی یک سرویس دهنده بصورت یک تابع نزولی از فاصله خواهد بود. این نزول تا جایی ادامه پیدا می کند که از آنجا به بعد بتوان از اثر زیان بار سرویس دهنده صرف نظر کرد (شکل ۴-۱). همچنین قسمت جاذب هدف را می توان در قالب هزینه حمل و نقل که تابعی صعودی از فاصله است بیان کرد (شکل ۴-۲).



شکل ۴-۱: تابع تأثیرات منفی بر حسب فاصله



شکل ۴-۲: تابع هزینه حمل و نقل بر حسب فاصله

۲-۴ مدل های مکانیابی گسسته با جاذبه - دافعه

در این بخش به بررسی مدل های مکانیابی گسسته با جاذبه و دافعه می پردازیم. در این مسائل برخی مشتریان تحت تاثیر منفی سرویس دهندگان قرار دارند. تمام مدل هایی که در این بخش معرفی می شوند NP -سخت هستند و برای حل آنها می توان از روش های فوق ابتکاری نیز بهره برد.

۱-۲-۴ مسئله مکانیابی سرویس دهنده با ظرفیت نامحدود و اثرات منفی

جمعی

در مسئله مکانیابی سرویس دهنده با ظرفیت نامحدود^۷، یک مجموعه از نقاط کاندیدا برای احداث سرویس دهنده وجود دارند و هدف پیدا کردن یک زیرمجموعه از آن ها برای استقرار سرویس دهنده هاست به گونه ای که هر مشتری دقیقاً به یکی از آن ها تخصیص داده شود و هزینه کل شامل هزینه احداث و هزینه حمل و نقل کمینه گردد. در این مدل بر خلاف مدل های p -میانه و p -مرکز تعداد سرویس دهنده ها از قبل مشخص نیست و جواب بهینه تعداد آن را مشخص می کند. حال مسئله را به حالتی که بعضی از مشتریان تحت تاثیرات منفی سرویس دهندگان قرار دارند گسترش می دهیم. این مشتریان را با K نمایش می دهیم. اگر $a_{kj} \geq 0$ بیانگر میزان مخاطره سرویس دهنده $j \in J$ روی مشتری $k \in K$ باشد و فرض کنیم که میزان مخاطره کلی جمعی باشد یعنی برابر با مجموع مخاطره هر جفت مشتری-سرویس دهنده باشد با تعریف متغیر های $x_j = 1$ هرگاه سرویس دهنده $j \in J$ احداث شود، در غیر این صورت $x_j = 0$ و همچنین $y_{ij} = 1$ هرگاه مشتری i از سرویس دهنده j خدمات دریافت کند، در غیر این صورت $y_{ij} = 0$ ، مدل زیر به دست می آید:

$$\begin{aligned}
 & \text{Min} \quad \left(\sum_{i \in I} \sum_{j \in J} c_{ij} y_{ij} + \sum_{j \in J} f_j x_j \right) + \sum_{k \in K} \sum_{j \in J} a_{kj} x_j \\
 & \text{S.t.} \quad \sum_{j \in J} y_{ij} = 1 \quad i \in I \quad (1-4) \\
 & \quad y_{ij} \leq x_j \quad i \in I, j \in J \\
 & \quad x_j \in \{0, 1\}, y_{ij} \in [0, 1] \quad i \in I, j \in J
 \end{aligned}$$

توجه کنید که به مدل مسئله $UFLP$ در حالت جاذبه-دافعه فقط جمله میزان مخاطره کل در تابع هدف نسبت به $UFLP$ اصلی اضافه شده است. قابل ذکر است چون x_j یک متغیر صفر/یک است، محدودیت های مسئله ایجاب می کنند که y_{ij} یک مقدار صحیح را در جواب بهینه داشته باشد لذا محدودیت $y_{ij} \in \{0, 1\}$ را به صورت $y_{ij} \in [0, 1]$ نوشته ایم^۱ تا از پیچیدگی مسئله تا حد امکان کاسته شود. برای توضیح علت، بدون اینکه به کلیت موضوع خللی وارد شود، فرض کنیم $i \in I$ داشته باشیم و به ازای آن $0 < y_{i,j} < 1$ بنابر محدودیت های مسئله $j_1, \dots, j_q \in J$ وجود دارند بطوری که $x_{j_k} = 1, \sum_{k=1}^q y_{i,k} = 1$ لذا تابع هدف مسئله به ازای این جواب برابر است با $\sum_{k=1}^q c_{i,j_k} y_{i,j_k} + \sum_{k=1}^q f_{j_k}$ اما $c_{i,j_p} = \min_{j_k} \{c_{i,j_k}\} \leq \sum_{k=1}^q c_{i,j_k} y_{i,j_k}$ زیرا عبارت $\sum_{k=1}^q c_{i,j_k} y_{i,j_k}$ ترکیب محدبی از c_{i,j_k} هاست، بنابراین داریم: $c_{i,j_p} + f_{j_p} \leq c_{i,j_p} + \sum_{k=1}^q f_{j_k} \leq \sum_{k=1}^q c_{i,j_k} y_{i,j_k} + \sum_{k=1}^q f_{j_k}$ و این یعنی جواب در حالت $0 < y_{i,j} < 1$ بهینه نیست، پس $y_{ij} \in \{0, 1\}$.

در این مدل f_j بیانگر هزینه احداث سرویس دهنده j ام است. هزینه c_{ij} می تواند شامل فاصله مشتری i تا سرویس دهنده j ، زمان یا هزینه دریافت خدمات باشد مثلاً c_{ij} را می توان به صورت $c_{ij} = w_i (h_j + t_{ij})$ نوشت که w_i میزان تقاضای مشتری i ام، h_j هزینه هر واحد خدمات سرویس دهنده j (هزینه های تولید و عملیات) و t_{ij} هزینه حمل و نقل ارائه یک واحد خدمات از سرویس دهنده j به مشتری i است.

^۱ Relaxation

حال اگر در مدل (۱-۴) به تاثیرات منفی سرویس دهنده j را با هزینه ثابت آن ادغام کنیم، خواهیم داشت:

$$\tilde{f}_j = f_j + \sum_{k \in K} a_{kj}$$

لذا تابع هدف (۱-۴) به تابع زیر تبدیل می شود:

$$\text{Min} \left(\sum_{i \in I} \sum_{j \in J} c_{ij} y_{ij} + \sum_{j \in J} \tilde{f}_j x_j \right) \quad (۲-۴)$$

همانطور که دیده می شود مسئله $UFLP$ با جاذبه و دافعه با تکنیک فوق به مسئله $UFLP$ معمولی تبدیل می شود.

۲-۲-۴ مسئله مکانیابی سرویس دهنده با ظرفیت نامحدود و کمترین پوشش

اثرات منفی

در حالت های مختلفی جمعاً فرض کردن اثرات منفی منطقی به نظر نمی رسد مثلاً سرویس دهندگانی که آب های زیرزمینی را آلوده می کنند تا یک شعاع معین، همه چیز را تحت تاثیر می گذارند و در حقیقت به تعداد سرویس دهندگان بستگی ندارد زیرا در هر صورت یک یا چند سرویس دهنده که در فاصله بحرانی آن قرار داشته باشند، تحت تاثیرات منفی قرار می گیرد. البته تاثیر تعداد سرویس دهندگان را در میزان آلودگی باید دانست. این نوع اثرات از نوع اثرات منفی با کمترین پوشش هستند و مدل آن به صورت زیر است.

$$\begin{aligned}
 & \text{Min} \quad \left(\sum_{i \in I} \sum_{j \in J} c_{ij} y_{ij} + \sum_{j \in J} f_j x_j \right) + \sum_{k \in K} a_k z_k \\
 & \text{S.t.} \quad \sum_{j \in J} y_{ij} = 1 \quad i \in I \\
 & \quad y_{ij} \leq x_j \quad i \in I, j \in J \\
 & \quad x_j \leq z_k \quad j \in C_k, k \in K \\
 & \quad x_j \in \{0, 1\}, \quad y_{ij}, z_k \in [0, 1] \quad i \in I, j \in J, k \in K
 \end{aligned} \tag{۳-۴}$$

متغیر z_k نشان می دهد که آیا مشتری k تحت تاثیر منفی سرویس دهندگان قرار گرفته است یا خیر و a_k بیان گر میزان هزینه یا مخاطره ی تاثیرات ایجاد شده روی مشتری است. مجموعه مکان های کاندیدا برای استقرار سرویس دهندگانی که در فاصله بحرانی مشتری k قرار دارند را با C_k نشان می دهیم. بنابراین اگر در هر یک از مکان های مجموعه C_k ، سرویس دهنده ای مستقر شود آنگاه متغیر z مربوطه یعنی $z_k = 1$ و مقدار a_k واحد به تابع هدف اضافه می شود. اگر مشتری k ام تحت تاثیر سرویس دهندگان نباشد یعنی $x_j = 0 \quad \forall j \in C_k$ لذا خواهیم داشت $0 \leq z_k$ و هیچ کدام از محدودیت های مسئله شامل z_k نخواهد بود (هیچ محدودیتی روی z_k نداریم). از طرفی اگر $a_k \geq 0$ در جواب بهینه باید داشته باشیم $z_k = 0$. این موضوع برای حالتی که z یک متغیر پیوسته باشد نیز برقرار است، بنابراین می توانیم بنویسیم $z_k \in [0, 1]$. اگر $a_k < 0$ محدودیت دیگری به صورت $z_k \leq \sum_{j \in C_k} x_j$ را می توانیم به مدل اضافه کنیم. این قید تضمین می کند که هرگاه $x_j = 0 \quad \forall j \in C_k$ آنگاه $z_k = 0$ شود. به عبارت دیگر هرگاه هیچ سرویس دهنده ای در اطراف مشتری k مستقر نشده باشد آنگاه این مشتری از محاسبات تابع هدف خارج شود.

ارکوت و نیومن [۷] در مطالعه تاثیرات منفی در مسائل مکانیابی به این نتیجه رسیدند که بهتر است تاثیرات منفی و مثبت را به طور جداگانه مورد توجه قرار داد لذا بررسی مدل های چند هدفه اهمیت زیادی در این حوزه مسائل مکانیابی دارد. حال مسائل بیان شده را به صورت دو هدفه فرموله می کنیم.

۳-۲-۴ مدل کمترین مجموع - کمترین مجموع

در مدل (۳-۴) قسمت های جاذب و دافع در یک تابع و صورت خطی ترکیب شده اند. مدل دو هدفه (۳-۴) به قرار زیر است:

$$\begin{aligned}
 & \text{Min} \quad \sum_{i \in I} \sum_{j \in J} c_{ij} y_{ij} + \sum_{j \in J} f_j x_j \\
 & \text{Min} \quad \sum_{k \in K} a_k z_k \\
 & \text{S.t.} \quad \sum_{j \in J} y_{ij} = 1 \quad i \in I \\
 & \quad y_{ij} \leq x_j \quad i \in I, j \in J \\
 & \quad x_j \leq z_k \quad j \in C_k, k \in K \\
 & \quad x_j \in \{0, 1\}, y_{ij}, z_k \in [0, 1] \quad i \in I, j \in J, k \in K
 \end{aligned} \tag{۴-۴}$$

برای حل این مدل باید مجموعه جواب های پارتو را که منجر به تبادل کامل اطلاعات بین هدف ها می شود پیدا کرد و نهایتا تصمیم گیرنده جوابها را با هم مقایسه کرده و جواب بهینه (برتر) را انتخاب می کند.

۴-۲-۴ مدل کمترین مجموع - ماکزیمین

یکی دیگر از توابعی که برای اهداف دافع مورد استفاده قرار می گیرد تابع ماکزیمین است. این تابع زمانی استفاده می شود که بخواهیم بیشترین تاثیری که یک مشتری از سرویس دهندگان دریافت کرده را کمینه کنیم. تعریف می کنیم $\tilde{a}_j = \max_{k \in K} a_{kj}$. مدل حاصل به صورت زیر است:

$$\begin{aligned}
 & \text{Min} \quad \sum_{i \in I} \sum_{j \in J} c_{ij} y_{ij} + \sum_{j \in J} f_j x_j \\
 & \text{Min} \quad Z \\
 & \text{S.t.} \quad \sum_{j \in J} y_{ij} = 1 \quad i \in I \\
 & \quad y_{ij} \leq x_j \quad i \in I, j \in J \\
 & \quad \tilde{a}_j x_j \leq Z \quad j \in J \\
 & \quad Z \geq 0 \\
 & \quad x_j \in \{0, 1\}, y_{ij}, z_k \in [0, 1] \quad i \in I, j \in J, k \in K
 \end{aligned} \tag{۵-۴}$$

۳-۴ مدل های شبه پراکنده

در مدل های پراکنده هیچ مشتری وجود ندارد و هدف مکانیابی سرویس دهندگان به گونه ای است که کمترین تاثیر را روی هم داشته باشند. در این بخش به بررسی مسائل p -پراکنده و p -دفاع با توجه به مفهوم جاذبه و دافعه می پردازیم.

۱-۳-۴ مسئله p -دفاع-مجموع

در مسئله p -دفاع-مجموع^۹ ($PDSP$) هم مشتری وجود ندارد و قرار است p تا از $|J|$ سرویس دهنده به گونه ای مستقر شوند تا کمترین مزاحمت و تاثیر را بر هم داشته باشند. مثلاً در استقرار فرستنده های رادیویی، اگر فرستنده ها دارای فرکانس های مشابه باشند باید تا حد امکان دور از هم قرار گیرند تا روی عملکرد هم تاثیر کمتری داشته باشند و ایجاد اختلال نکنند. فرستنده هایی که فرکانس کاری متفاوت دارند می توانند نزدیک هم قرار گیرند. محل استقرار آن ها با کمترین معیار هزینه پیدا می شود. فرض کنیم که $a_{hj} \geq 0$ نشان دهنده فاصله بین سرویس دهنده های j, h است واضح است که $a_{jj} = 0$. مسئله p -دفاع-مجموع مکان استقرار p سرویس دهنده را به گونه ای پیدا می کند که مجموع فاصله های بین هر جفت از سرویس دهنده های مستقر شده ماکزیمم شود.

$$\begin{aligned} \text{Max} \quad & \sum_j \sum_h a_{hj} x_h x_j \\ \text{S.t.} \quad & \sum_j x_j = p \\ & x_j \in \{0, 1\} \end{aligned} \quad (۴-۶)$$

مسئله p -دفاع-مجموع از نوع NP -سخت است که برای مطالعه جزئیات آن به [۸۱] مراجعه شود. کرتسارز و پلگ [۸۲] یک الگوریتم تقریبی از مرتبه $O(|J|^{۰.۳۸۸۵})$ برای حل این مسئله پیشنهاد

^۹ p-Diffence-Sum Problem(PDSP)

کرده اند. تعداد الگوریتم های دقیقی که برای حل مسئله p -دفاع-مجموع پیشنهاد شده، کم است.

در [۸۳] نحوه استفاده از کران بالا نشان داده شده است، که بقرار زیر است:

$$\bar{a}_h = \max \left\{ \sum_{j \in J} a_{hj} x_j : \sum_{j \in J} x_j = p; x_j \in \{0, 1\}, j \in J \right\} \quad (۷-۴)$$

حال کران بالای u را به صورت زیر انتخاب می کنیم:

$$u = \max \left\{ \sum_{h \in J} \bar{a}_h x_h : \sum_{h \in J} x_h = p; x_h \in \{0, 1\}, h \in J \right\} \quad (۸-۴)$$

مسائل (۷-۴) و (۸-۴) را به راحتی در $O(n)$ می توان محاسبه کرد زیرا از بین n متغیر می خواهیم

p تا را به گونه ای انتخاب کنیم که تابع هدف مربوطه ماکزیمم شود. کران به دست آمده را می توان

در یک روش شاخه و کرانه ای برای پیدا کردن جواب بهینه استفاده کرد.

۲-۳-۴ مسئله p -پراکنده

یکی از مسائل پراکندگی پرکاربرد، مسئله p -پراکنده (PDP) است که در آن به دنبال پیدا کردن

مکان p سرویس دهنده هستیم به گونه ای که کمترین فاصله بین هر جفت از سرویس دهندگان

ماکزیمم شود. اگر a_{hj} منفی باشد مسئله به صورت زیر فرموله می شود:

$$\begin{aligned} \text{Max} \quad & r \\ \text{S.t.} \quad & r \leq a_{hj} x_h x_j \quad j, h \in J \\ & \sum_{j \in J} x_j = p \\ & x_j \in \{0, 1\} \quad j \in J \end{aligned} \quad (۹-۴)$$

مسئله p -پراکنده نیز NP -سخت است [۸۴]. راوی^{۱۰} و همکاران [۸۵] نشان دادند که در حالت

کلی مسئله p -پراکنده را نمی توان با نرخ ثابت به صورت تقریبی حل کرد مگر اینکه $NP = P$.

ارکوت [۸۴] چند کران بالای ساده برای حل مسئله p -پراکنده پیشنهاد کرده است. او این کران ها را در یک الگوریتم شاخه و کرانه به کار برد و مسائلی از اندازه حداکثر ۴۰ را با آن حل کرد. در [۸۵] نشان داده شده است که مدل (۹-۴) را می توان به وسیله مدل (۶-۴) حل کرد. برای یک مقدار ثابت در نظر می گیریم. حال تعریف می کنیم اگر $a_{hj} \geq r$ آنگاه $\tilde{a}_{hj} = 1$ ، در غیر اینصورت $\tilde{a}_{hj} = 0$. بنابراین مسئله به صورت زیر نوشته می شود:

$$\begin{aligned} \text{Max} \quad & \sum_h \sum_j \tilde{a}_{hj} x_h x_j \\ \text{S.t.} \quad & \sum_{j \in J} x_j = p \\ & x_j \in \{0, 1\} \quad j \in J \end{aligned} \quad (10-4)$$

اگر مقدار جواب به دست آمده برابر با $n(n-1)$ بود آنگاه (۹-۴) به ازای این مقدار از r شدنی است. زیرا در حالتی که $j \in J$ و $x_j = 1$ و برای هر $h, j \in J$ داشته باشیم $r \leq a_{hj}$ داریم $\tilde{a}_{hj} = 1$ بنابراین $n(n-1) = \sum_h \sum_j \tilde{a}_{hj} x_h x_j$ و این همان چیزی است که از ساده کردن مدل به دنبالش بودیم. در هر صورت اگر $h, j \in J$ وجود داشته باشند که $a_{hj} < r$ لذا مقدار بهینه (۱۰-۴) کمتر از $n(n-1)$ خواهد شد. بنابراین مقدار بهینه (۱۰-۴) شدنی بودن را به ازای هر مقدار مشخص شده r نشان خواهد داد. چون r باید با یکی از مقادیر a_{hj} (با بزرگترین آن ها) برابر شود لذا حداکثر $O(n^2)$ مقدار متفاوت از r باید ارزیابی شوند و جست و جوی دودویی را برای پیدا کردن مقدار بهینه r می توان به کار برد. نتایج محاسباتی نشان می دهند که تجزیه مسئله به زیر مسئله ها سرعت الگوریتم را نسبت به روش پیشنهادی ارکوت افزایش می دهد.

در مسائل p -پراکنده و p -دفاع مجموع اگر ضرایب منفی باشند، با افزودن یک مقدار به اندازه کافی بزرگ M به همه ضرایب، می توان مسئله را به حالتی که تمام ضرایب مثبت باشند تبدیل کرد. در این صورت در مسئله PDP به اندازه M واحد و در مسئله $PDSP$ به اندازه Mp^2 واحد به مقدار

بهینه اضافه می شود. مطالعات دیگری که بر روی مسئله p -پراکنده انجام شده است روابط زیر را نشان می دهند:

$$\text{مقدار تابع هدف مسئله } (p - \text{مرکز}) = \text{مقدار تابع هدف مسئله } ((p+1) - \text{پراکنده}) \cdot \frac{1}{2}$$

این رابطه را شیر [۸۶] برای مسئله p -پراکنده روی درخت به دست آورد. مطالعه فوق رابطه دوگانی بین دو مسئله از دو دسته مختلف مکانیابی را نشان می دهد که یکی از نوع جاذب و دیگری از نوع دافع است. کبی [۸۷] رابطه زیر را برای مسئله p -پراکنده روی شبکه به دست آورده است:

$$\text{مقدار تابع هدف مسئله } (p - \text{مرکز}) \leq \text{مقدار تابع هدف مسئله } ((p+1) - \text{پراکنده}) \cdot \frac{1}{2}$$

چاندراکان و داتی [۸۸] با استفاده از این خاصیت دوگانی دو الگوریتم مشابه از مرتبه $O(n^2 \log p \log n)$ برای مسائل p -پراکنده و p -مرکز روی درخت ارائه دادند. کینکید [۸۹]،

الگوریتم شبیه سازی تبریدی و جست وجوی ممنوع^{۱۱} را برای مسئله p -پراکنده گسسته بدون وزن پیاده کرد و نتایج محاسبات نشان می دهد که روش جست وجوی ممنوع از روش شبیه سازی تبریدی و الگوریتم ابتکاری ارکوت [۸۴]، عملکرد بهتری دارد.

تمیر [۹۰] معیار دیگری را برای مکانیابی سرویس دهنده مضر روی گراف در نظر گرفت و آن مسئله p -بیشترین مجموع است. او مسئله را به صورت زیر فرموله کرد:

$$\text{Max} \left\{ \sum_{i=1}^n \sum_{j=1}^p \alpha_{ij} d(v_i, y_j) + \sum_{j=1}^p \sum_{k=1}^p \beta_{jk} d(y_j, y_k) \right\} \quad (11-4)$$

او نشان داد که این مسئله روی گراف یک مسئله NP -سخت است و برای حالتی که گراف درخت باشد یک روش حل با مرتبه $O(np)$ با برنامه ریزی پویا ارائه کرد. مسئله p -دفاع-مجموع در این دسته قرار می گیرد.

۴-۴ مسائل قابل حل در زمان چند جمله ای

برای مسائل مکانیابی که روی شبکه تعریف می شوند، حالت های خاصی در زمان چند جمله ای قابل حل هستند را می توان مشخص کرد حالت هایی که شبکه دارای ساختار خاصی مثل مسیر، دور، ستاره یا درخت باشد. در [۹۱] روشی برای حل مسئله ۱-میان روی شبکه با وزن مثبت و منفی ارائه شد. همچنین نشان داده شده است که مسئله ۱-میان روی کاکتوس در $O(|V|n)$ قابل حل است که در آن V مجموعه رئوس گراف است. تحقیقی که توسط کرارپ^{۱۲} و بورکارد^{۱۳} [۹۱]، پایه گذاری شد، در [۹۲] برای مسئله ۲-میان روی درخت با وزن مثبت و منفی ادامه پیدا کرد.

مسئله p -میان با وزن مثبت دارای خاصیت بهینگی راسی است که باعث می شود حداقل یک جواب بهینه که شامل p راس گراف است، موجود باشد. بر خلاف آنچه که در مسائل با وزن مثبت داریم، در مسائل مکانیابی با وزن مثبت و منفی و استقرار بیش از یک وسیله دقیقاً مشخص نیست که چه چیزی باید بهینه شود به این منظور دو تابع در [۹۲] برای مسئله p -میان پیشنهاد شده است.

$$f_l(X) = \sum_{i=1}^n \min_{1 \leq j \leq p} (w_i d(v_i, x_j))$$

$$f_r(X) = \sum_{i=1}^n w_i \min_{1 \leq j \leq p} d(v_i, x_j)$$
(۱۲-۴)

Krarup^{۱۲}

Burkard^{۱۳}

بورکارد و همکاران [۹۲]، برای f_1 خواصی را ثابت کردند و یک الگوریتم از مرتبه $O(n^2)$ برای مسئله ۲-میانه روی یک درخت با n راس و یک الگوریتم از مرتبه $O(n \log n)$ برای ستاره و یک الگوریتم خطی برای مسیرها ارائه دادند. همچنین برای f_2 یک الگوریتم از مرتبه $O(n^2)$ برای ۲-میانه روی یک درخت ارائه دادند. اگر میانه ها را فقط روی رئوس مستقر کنیم، پیچیدگی به $O(n^2)$ کاهش می یابد. بورکارد و فتحعلی [۹۳] نیز الگوریتمی برای پیدا کردن ۳-میانه روی یک درخت با تابع هدف f_2 ارائه کردند. از روش های ابتکاری که برای پیدا کردن p -میانه با وزن مثبت و منفی روی شبکه ارائه شده است می توان به روش جست وجوی همسایگی متغیر از فتحعلی و تقی زاده [۹۴]، الگوریتم مورچه فتحعلی و همکاران [۹۵] و الگوریتم ژنتیک فتحعلی [۹۶]، اشاره کرد.

۵-۴ مدل های مکانیابی پیوسته با جاذبه و دافعه

در این بخش به بررسی مدل های مکانیابی نیمه نامطلوب (با جاذبه و دافعه) در فضای پیوسته می پردازیم.

۴-۵-۱ مدل کمینه کردن مجموع فاصله ها

مدل زیر توسط مورالس و همکاران [۹۷] ارائه شده است:

$$\inf z(x) = \sum_{a \in A} \left[f_a(\|x - a\|_-^a) + g_a(\|x - a\|_+^a) \right] \quad (۱۳-۴)$$

که در آن $A \subseteq R^2$ ، یک مجموعه متناهی و ناتهی است و شامل مختصات مشتریان (مراکز جمعیتی) است. $S \subseteq R^2$ مجموعه جوابهای شدنی است و می توان آن را به صورت اجتماعی از نواحی چند ضلعی نوشت. معمولاً در مسائل مکانیابی پیوسته فرض بر این است که $S \setminus A \neq \emptyset$.

توابع فاصله دلخواه متناظر با مشتری a در فضای R^2 هستند. برای مطالعه بیشتر در مورد توابع فاصله و نحوه انتخاب آنها به [۹۸-۱۰۰ و ۶۸] مراجعه کنید.

تعریف. فرض کنید $(X, d_X), (Y, d_Y)$ دو فضای متریک باشند، تابع $f: X \rightarrow Y$ لپ شیتس گویند اگر $k \geq 0$ وجود داشته باشد که $\forall x_1, x_2 \in X \quad d_Y(f(x_1), f(x_2)) \leq k d_X(x_1, x_2)$. همچنین گوییم تابع f در شرط هلدر از مرتبه α صدق می کند هرگاه

$$\forall x_1, x_2 \in X \quad d_Y(f(x_1), f(x_2)) \leq k d_X(x_1, x_2)^\alpha$$

$f_a: (0, +\infty) \rightarrow R$ تابعی غیر صعودی و محدب است و تاثیرات محیطی سرویس دهنده روی مرکز جمعیتی a و در فاصله $\|x - a\|_-^a$ را نشان می دهد. f_a با تعریف در نقطه $x = 0$ به صورت زیر گسترش می یابد:

$$f_a(\cdot) = \lim_{t \downarrow 0} f_a(t) \in R \cup \{+\infty\}$$

یک تابع پیوسته لپ شیتس غیر نزولی است که هزینه حمل و نقل از مرکز a تا سرویس دهنده را بر اساس فاصله $\|x - a\|_+^a$ اندازه می گیرد.

قابل ذکر است که حالت های خاصی از (۴-۱۳) در [۱۰۱ و ۱۰۲] بررسی شده اند و در آن ها توابع f_a, g_a را خطی در نظر گرفته اند و تنها از یک نرم خاص برای فاصله استفاده شده است یعنی

$$\forall a \in A: \quad \|\cdot\|_-^a = \|\cdot\|_+^a = \|\cdot\|$$

در [۱۰۳] ناحیه شدنی را اجتماعی متناهی از چند وجهی ها در نظر گرفته و از نرم اقلیدسی برای فاصله استفاده شده است. همچنین توابع f_a, g_a خطی در نظر گرفته شده اند و مسئله را با روش های بهینه سازی $d.c.$ ^{۱۴} (روش های بهینه سازی مربوط به تفاضل دو تابع محدب [۱۰۰]) حل کرده

^{۱۴} Difference of Convex functions

اند. آن ها همچنین یک حالت غیر خطی را نیز مورد بررسی قرار داده اند که در آن $f_a(t) = w_a e^{-t}$ با $w_a > 0$ است که مجدداً تبدیل به یک مسئله بهینه سازی $d.c.$ می شود. تابع هدف این مسئله از نوع $d.c.$ است و محدودیت هایش مجموعه های تراز تابع $d.c.$ هستند.

در [۱۰۲] حالت های خاص دیگری از (۴-۱۳) بررسی شده است و آن ها ناحیه شدنی را یک چندضلعی محدب در نظر گرفته و توابع $f_a, g_a -$ روی R_+ غیر صعودی و محدب فرض شده اند. آن ها نشان دادند که تحت شرایط معینی تابع هدف به نوع $d.c.$ تبدیل می شود و با روش های بهینه سازی $d.c.$ قابل حل خواهد بود.

یکی از موثرترین روش هایی که برای حل مسائل مکانیابی سرویس دهنده نیمه ناخوشایند پیشنهاد شده است روش شاخه و کرانه ای پیوسته مستطیل بزرگ مستطیل کوچک است که در [۱۰۷] برای حل حالت خاصی از (۴-۱۳) ارائه شده است. آنها فرض کردند که برای هر $a \in A$ تابع f_a غیرصعودی و پیوسته است و $g_a = 0$.

حال به بررسی شرایط وجود جواب و شرط کافی بهینگی را بررسی می کنیم.

در مدل (۴-۱۳) وقتی فاصله بین سرویس دهنده و مرکز a کم می شود، تابع f_a به سمت $+\infty$ میل می کند. بنابراین نقاط نزدیک a نمی توانند بهینه باشند. می دانیم که هر دو نرمی در فضای متناهی، معادلند یعنی ثابتهای $c_+, c_- > 0$ موجوداند به گونه ای که :

$$c_- \|x\| \leq \|x\|_r \leq c_+ \|x\|, \quad \forall x \in R^r \quad (۴-۱۴)$$

با توجه به تعریف فوق ثابت های $k_+, k_-, c_+, c_- > 0$ موجوداند به قسمی که برای هر $x \in R^r$ و $a, b \in A$ داریم:

$$\|x\|_+^a \leq c_+ \|x\|_+^b \quad (۴-۱۵)$$

$$\|x\|_-^a \leq c_- \|x\|_-^b \quad (۱۶-۴)$$

$$k_- \|x\|_-^a \leq \|x\|_+^a \quad (۱۷-۴)$$

$$k_+ \|x\|_+^a \leq \|x\|_-^a \quad (۱۸-۴)$$

تعریف می کنیم:

$$A_\infty = \{b \in A : \lim_{t \rightarrow +\infty} f_b(t) = +\infty\} \quad (۱۹-۴)$$

قضیه ۴-۱: ([۹۷]) مجموعه ای مانند $H = \bigcup_{b \in A_\infty} B_-^b$ موجود است که B_-^b همسایگی های دو به دو

مجزای باز از A_∞ هستند و

$$\inf_{x \in S} z(x) = \inf_{x \in S \setminus H} z(x) \quad (۲۰-۴)$$

اثبات. فرض کنیم $x \in S \setminus A_\infty$ و $c_- > 0$ در (۱۴-۴) صدق می کند. طبق تعریف A_∞ می توان

$t \cdot (b) \in (0, 1)$ را به گونه ای انتخاب کرد که

$$f_b(t) > z(x) - \sum_{\substack{a \in A \\ a \neq b}} f_a(c_- + \|b - a\|_-^a) - \sum_{a \in A} l_a \quad \forall t \in (0, t \cdot (b))$$

$$H = \bigcup_{b \in A_\infty} B_-^b \text{ و } B_-^b = \{x \in R^r : \|x - b\|_-^b < t \cdot (b)\}$$

بدون از دست دادن کلیت، این همسایگی ها را باز و دو به دو مجزا در نظر می گیریم (توجه داریم که

به خاطر در اختیار بودن انتخاب $t \cdot (b)$ این کار قابل انجام است) اگر $x \in S \cap H$ آنگاه دقیقا یک

$b \in A_\infty$ موجود است که $\|x - b\|_-^b < t \cdot (b)$. چون B_-^b ها مجزایند لذا به ازای هر $a \in A$ که $a \neq b$

است، طبق خاصیت نامساوی مثلث برای نرم ها داریم:

$$\begin{aligned}
\|x - a\|_-^a &\leq \|x - b\|_-^a + \|b - a\|_-^a \\
&\leq c_- \|x - b\|_-^b + \|b - a\|_-^a < c_- t(b) + \|b - a\|_-^a \\
&< c_- + \|b - a\|_-^a
\end{aligned}$$

بنابراین

$$\begin{aligned}
z(x) &= \sum_{a \in A} [f_a(\|x - a\|_-^a) + g_a(\|x - a\|_+^a)] \\
&\geq \sum_{a \in A} [f_a(\|x - a\|_-^a) + l_a] \\
&= f_b(\|x - b\|_-^b) + \sum_{\substack{a \in A \\ a \neq b}} f_a(\|x - a\|_-^a) + \sum_{a \in A} l_a \\
&\geq f_b(\|x - b\|_-^b) + \sum_{\substack{a \in A \\ a \neq b}} f_a(c_- + \|b - a\|_-^a) + \sum_{a \in A} l_a \\
&> z(x_-) - \sum_{\substack{a \in A \\ a \neq b}} f_a(c_- + \|b - a\|_-^a) - \sum_{a \in A} l_a \\
&\quad + \sum_{\substack{a \in A \\ a \neq b}} f_a(c_- + \|b - a\|_-^a) + \sum_{a \in A} l_a = z(x)
\end{aligned}$$

در این رابطه هم ناتهی بودن $S \setminus H$ را می‌رساند زیرا $x \in S \setminus H$ و هم اینکه هیچ نقطه‌ای در

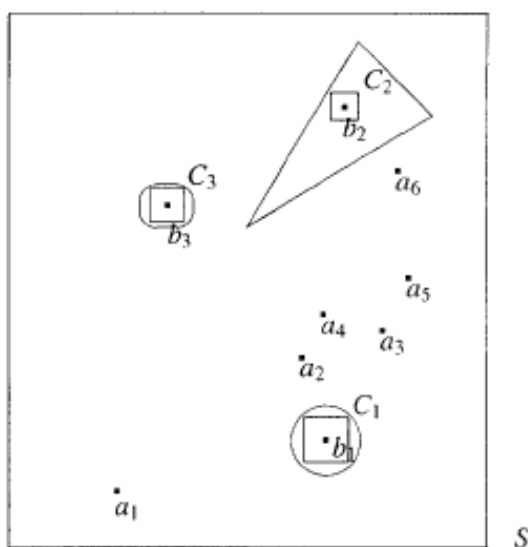
$S \cap H$ بهتر از x نیست. $\square$

نکته ۱: همانطور که در قضیه فوق دیده می‌شود می‌توان مجموعه‌هایی از نقاط شدنی از اطراف هر

نقطه در A_∞ را حذف کرد و این مجموعه‌ها را می‌توان مستطیل‌هایی به مرکز هر $b \in A$ در نظر

گرفت که در B_-^b محاط شده‌اند. لذا بر این اساس مجموعه $S \setminus H$ را می‌توان به صورت اجتماعی

متناهی از ناحیه‌های چند ضلعی نوشت.



شکل ۳-۴: نقاط تقاضا و همسایگی های حول نقاط دارای اثر منفی

همانطور که در شکل ۳-۴ نشان داده شده است، $A = \{a_1, a_2, a_3, a_4, a_5, a_6, b_1, b_2, b_3\}$ و $A_m = \{b_1, b_2, b_3\}$ حول هر نقطه در A_m همسایگی متناظرش با استفاده از نرم $\| \cdot \|_{b_i}$ و شعاع $t(b_i)$ رسم شده است. درون هر همسایگی یک مستطیل c_i محاط شده است. بنابراین جواب بهینه (۱۳-۴) در $S \setminus \bigcup_{i=1}^3 c_i$ قرار دارد. اگر S کران دار باشد باید یک تابع پیوسته را روی یک مجموعه فشرده کمینه کنیم و جواب از قضیه ۱-۴ و قضیه وایرستراس به دست می آید. پس قضیه زیر را داریم.

قضیه ۲-۴: ([۹۷]). اگر ناحیه شدنی S کران دار باشد، آن گاه مسئله (۱۳-۴) جواب بهینه دارد.

اگرچه در مسائل عملی، ناحیه شدنی همیشه کران دار است ولی مسائل با ناحیه شدنی جنبه تئوریک دارند و از آن ها برای محاسبه کران ها استفاده می شود. جواب بهینه مسئله کران دار، بهتر از جواب مسئله بیکران نیست یعنی مسئله بیکران یک کران پایین برای آن به دست می دهد.

در شرایط کافی که برای وجود جواب بهینه (۱۳-۴) در [۱۰۳] داده شده است، هزینه حمل و نقل به صورت یک تابع غیر نزولی و مقعر از فاصله فرض شده است:

قضیه ۳-۴: ([۹۷]). فرض کنیم K_1, C_- اعداد مثبتی باشند که به ترتیب در (۴-۱۶) و (۴-۱۷)

$$\lim_{t \uparrow \infty} \sum_{a \in A} [f_a(C_-t) + g_a(K_1t)] = +\infty \text{ و } \forall a \in A \text{ اگر } g_a \text{ یک تابع مقعر باشد و}$$

آنگاه مسئله (۴-۱۳) جواب بهینه دارد.

نکته ۲. آنطور که در [۱۰۴] آمده است تحت فرضیات قضیه ۳-۴ همیشه می توان مستطیلی ساخت که شامل نقطه بهینه باشد و چون g_a مقعر هستند، بدون از دست دادن کلیت می توان فرض کرد که ناحیه شدنی کراندار است. چون مسئله (۴-۱۳) در حالت کلی نامحدب است برای یافتن جواب باید از روشهای بهینه سازی سراسری [۱۰۴ و ۱۰۵] استفاده کنیم. الگوریتمی که مورالس و همکاران [۹۷]، برای این مسئله در نظر گرفتند روش مستطیل بزرگ مستطیل کوچک است. در الگوریتم آنها مجموعه $S \setminus H$ را به عنوان ناحیه شدنی اولیه در نظر می گیریم و کار با مستطیل R که شامل جواب بهینه است آغاز می شود. در آنچه در ادامه می آید R نشان دهنده یک مستطیل و $z^*(R)$ مقدار بهینه $z(x)$ روی R است.

با استفاده از خواص یکنوایی f_a, g_a کران های زیر به دست می آیند.

گزاره ۴-۴: ([۹۷]). فرض کنیم $z.(R)$ به صورت زیر باشد

$$z.(R) = \sum_{a \in A} \left[f_a \left(\max \|x - a\|_-^a \right) + g_a \left(\min \|x - a\|_+^a \right) \right] \quad (۴-۲۱)$$

$$z.(R) \leq z^*(R) \text{ آنگاه}$$

این نتیجه با توجه به مثبت بودن f_a, g_a و یکنوایی آنها و نیز توجه به این موضوع که در حالت کلی

$$\min(f+g)(x) \geq \min f(x) + \min g(x), \text{ به دست می آید.}$$

نکته ۳. چون $\| \cdot \|_-^a$ یک تابع محدب است، ماکزیمم خود را در R روی رؤوس آن اختیار می کند. بنابراین پیدا کردن $\max_{x \in R} \|x - a\|_-^a$ به ارزیابی چهار راس مستطیل R تبدیل می شود. محاسبه $\min_{x \in R} \|x - a\|_+^a$ یک مسئله برنامه ریزی محدب است. در حالتی که $\| \cdot \|_+^a$ یک نرم L_p باشد، روش خاصی برای حل آن پیشنهاد شده است. برای مطالعه بیشتر به [۲۹ و ۲۷] مراجعه شود. بنابراین $z(R)$ را به راحتی با محاسبه بیشترین فاصله هر $a \in A$ تا مستطیل R و تصویر کردن نقاط $a \in A$ روی R می توان محاسبه کرد.

حال کران پایین دیگری برای $z^*(R)$ به دست می آوریم و نشان می دهیم که کارایی آن حداقل به اندازه $z(R)$ است.

حال از (۴-۲۳) جهت پیدا کردن یک کران پایین بهتر برای $z^*(R)$ استفاده می کنیم و با استفاده از روش ضرایب لاگرانژ [۱۰۶] یک فرم معادل برای آن پیدا می کنیم.

گزاره ۴-۵: ([۹۷]). برای هر $a \in A$ فرض می کنیم I_a به ازای از R_+ و شامل بازه زیر باشد:

$$\left[\min_{x \in R} \|x - a\|_-^a, \max_{x \in R} \|x - a\|_-^a \right]$$

آنگاه (۴-۲۳) معادل است با

$$\begin{aligned} \text{Min} \quad & \sum_{a \in A} f_a(t_a) \\ \text{S.t.} \quad & x \in R \\ & t_a \in I_a \quad a \in A \\ & t_a \leq \|x - a\|_-^a \quad a \in A \end{aligned} \quad (۴-۲۴)$$

اثبات. (i) فرض می کنیم x^* جواب بهینه (۴-۲۳) باشد. تعریف می کنیم $t^* = (\|x - a\|_-^a)_{a \in A}$ و لذا

(x^*, t^*) جواب بهینه (۴-۲۴) است چون برای هر جواب شدنی $(\hat{x}, \hat{t})$ داریم:

$$\sum_{a \in A} f_a(t_a^*) = \sum_{a \in A} f_a(\|x^* - a\|_-^a) \leq \sum_{a \in A} f_a(\|\hat{x} - a\|_-^a) \leq \sum_{a \in A} f_a(\hat{t}_a) \quad \square$$

با استفاده از روش ضرایب لاگرانژ [۱۰۶] برای محدودیت $t_a \leq \|x - a\|_-^a$ گزاره زیر به دست می آید.

گزاره ۴-۶: ([۹۷]). برای هر $\lambda = (\lambda_a)_{a \in A} \geq 0$ مقدار $RL_-(\lambda)$ که به صورت زیر تعریف می شود

$$RL_-(\lambda) = \min_{\substack{x \in R \\ t_a \in I_a}} \sum_{a \in A} [f_a(t_a) + \lambda_a(t_a - \|x - a\|_-^a)] = \left(\sum_{a \in A} \min_{t_a \in I_a} [f_a(t_a) + \lambda_a t_a] \right) - \max_{x \in R} \sum_{a \in A} \lambda_a \|x - a\|_-^a$$

یک کران پایین برای (۴-۲۳) است.

نکته ۴. ([۹۷]). برای محاسبه $RL_-(\lambda)$ باید جواب بهینه یک مسئله بهینه سازی محدب به ازای

هر $a \in A$ را به دست آورد.

نکته ۵. ([۹۷]). اگر $\left[\cdot, \max_{x \in R} \|x - a\|_-^a \right]$ باشد داریم:

$$\max_{\lambda \geq 0} RL_-(\lambda) \geq RL_-(\cdot) = \sum_{a \in A} f_a \left(\max_{x \in R} \|x - a\|_-^a \right) \quad (۴-۲۵)$$

گزاره ۴-۷: ([۹۷]). $z(R) \leq \max_{\lambda \geq 0} L(\lambda)$ که در آن

$$L(\lambda) = RL_-(\lambda) + \sum_{a \in A} g_a(\min \|x - a\|_+^a) \quad (۴-۲۶)$$

تابع $z(\cdot)$ یک تابع لیپ شیتس روی $S \setminus H$ است [۹۷]، و از هر کدام از موارد ذکر شده می توان به

عنوان کران پایین استفاده کرد. با به کار بردن روش شاخه و کرانه ای مستطیل بزرگ مستطیل

کوچک یا مثلث بزرگ مثلث کوچک می توان مسئله را حل کرد.

محاسبه تاثیرات منفی و انتخاب تابعی که نشان دهنده این تاثیر بر سرویس گیرنده باشد کار پیچیده

و مشکلی است. در این خصوص معمولاً از نرم های اقلیدسی، اریب (در حالت هایی که نقش باد و

عوامل نظیر آن مهم است) و یا بلوکی استفاده شده است. استفاده از نرم های بلوکی پیچیدگی مدل را کمتر می کند و لذا حل آن ساده تر می شود. چگال بودن نرم های چند ضلعی در خانواده نرم ها سبب می شود تا هر نرم دلخواه را بتوان با دقت مورد نیاز تقریب زد. هرچه تعداد جهت های راسی بیشتر باشد دقت نرم مزبور نیز بیشتر خواهد بود.

همانطور که قبلا گفته شد تابع هزینه های نامطلوب به صورت یک تابع نزولی از فاصله است. کیفیت و چگونگی نزول این تابع به نوع سرویس دهنده و خواص و ویژگی های آن بستگی دارد. تا به حال ۵ نوع تابع پیشنهاد شده است [۱۰۷] که به اختصار به آن ها اشاره می کنیم.

ساده ترین و معمول ترین تابعی که در نظر گرفته شده است تابع خطی است.

$$f_a(d) = \alpha_a - \beta_a d$$

که α_a, β_a اعداد حقیقی نامنفی هستند.

تابع دیگری که مورد استفاده قرار گرفته است، تابع پتانسیل می باشد. این تابع توسط ملاچرینودیس و کلینا [۲۴] پیشنهاد شده است.

$$f_a(d) = \frac{\alpha_a}{d^{k_a}}$$

که α_a یک پارامتر حقیقی نامنفی است. حالت خاصی از این تابع ($k=2$) را نوع گرانشی می نامند. گسترش برخی از آلودگی ها مثل صوت، نور و تشعشع با معکوس مجذور فاصله نسبت مستقیم دارند، لذا توی، الخیال و ژو در سال ۱۹۹۵ تابع نمایی برای مدل کردن اثر نامطلوب سرویس دهنده به صورت زیر معرفی کردند [۱۰۰]

$$f_a(d) = \alpha_a e^{(-\beta_a d)}$$

کیسر و مارین [۱۰۸] نیز تابع لگاریتمی زیر را برای این منظور به کار بردند

$$f_a(d) = -\alpha_a \log(d)$$

توابع گسسته ای نیز برای این منظور پیشنهاد شده است که به صورت زیر می باشد:

$$f_a(d) = \begin{cases} w_a & d < d_a^* \\ \cdot & o.w. \end{cases}$$

که در آن d_a^* مرزی است که از آن به بعد از اثر نامطلوب سرویس دهنده صرف نظر می شود.

۴-۶ مکانیابی چند وسیله ای در مجموعه های محدب با نرم بلوکی

در مسائل مکانیابی سرویس دهنده نیمه نامطلوب در عمل با موقعیت هایی رو به رو می شویم که فضای شدنی اجتماعی از مجموعه های محدب است و تشکیل یک مجموعه احتمالا نامحدب را می دهند. در این مسئله ما به بررسی مسئله چند وسیله ای در محیطی می پردازیم که فضای شدنی آن اجتماعی از مستطیل هاست. تابع فاصله در نظر گرفته شده نرم های بلوکی هستند. برای مسئله فوق الذکر مدل های برنامه ریزی خطی ارائه می شود.

در مسئله مکانیابی چند وسیله ای فرض می کنیم که m سرویس دهنده $P_1, \dots, P_m$ موجودند و می خواهیم مکان n سرویس دهنده جدید یعنی $X_1, \dots, X_m$ را در صفحه پیدا کنیم. اگر $d(X_j, X_k)$ بیانگر فاصله بین دو سرویس دهنده جدید و $d(X_j, P_i)$ نیز بیانگر فاصله بین سرویس دهنده جدید j و سرویس دهنده موجود i باشد و همچنین هزینه هر واحد حمل و نقل بین X_j, P_i برابر w_{ji} و هزینه بین X_j, X_k برابر با v_{jk} باشد تابع هدف مسئله به صورت زیر است:

$$f(X_1, \dots, X_n) = \sum_{1 \leq j < k \leq n} v_{jk} d(X_j, X_k) + \sum_{j=1}^n \sum_{i=1}^m w_{ji} d(X_j, P_i) \quad (52-4)$$

در مسئله مکانیابی چند وسیله ای به دنبال پیدا کردن مکان n سرویس دهنده جدید یعنی $X_1^*, \dots, X_n^*$ هستیم به گونه ای که هزینه کل مینیمم گردد. مسئله مکانیابی چند وسیله ای با نرم بلوکی به صورت زیر فرموله می شود:

$$f(X_1, \dots, X_n) = \sum_{1 \leq j < k \leq n} v_{jk} \|X_j - X_k\|_B + \sum_{j=1}^n \sum_{i=1}^m w_{ji} \|X_j - P_i\|_B \quad (53-4)$$

۱-۶-۴ مسئله مکانیابی چندوسیله ای در مجموعه های محدب

فرض کنیم $P_1, \dots, P_m$ مختصات m ماشین در یک کارگاه باشند و می خواهیم مکان n ماشین جدید را پیدا کنیم. فرض می کنیم که ماشین ها یک فضای مستطیلی را اشغال می کنند و مختصات آنها به صورت (px_j, py_j) , $j = 1, \dots, m$ است.

بنابراین فضای باقیمانده از کارگاه که قابل استفاده برای مستقر کردن ماشین های جدید هستند را به K مستطیل (بازه) تقسیم می کنیم به گونه ای که با اندازه ماشین ها تطابق داشته باشد و مکانیابی را روی چنین مجموعه هایی انجام می دهیم. این مجموعه ها را برای $j = 1, \dots, K$ به صورت زیر نمایش می دهیم:

$$S_j = \{(x, y) | a_j \leq x \leq b_j, c_j \leq y \leq d_j\}$$

بنابراین فرمول بندی مسئله عبارت است از

$$\begin{aligned} \text{Min} \quad & \sum_{1 \leq j < k \leq n} v_{jk} \|X_j - X_k\|_B + \sum_{j=1}^n \sum_{i=1}^m w_{ji} \|X_j - P_i\|_B \\ \text{S.t.} \quad & X_j \in S_j \quad j = 1, \dots, n \end{aligned} \quad (54-4)$$

در این مسئله علاوه بر مکانیابی ماشین های جدید با نوعی مسئله تخصیص نیز رو به رو هستیم. هر ماشین باید فقط به یک ناحیه تخصیص داده شود و هر ناحیه حداکثر می تواند شامل یک ماشین باشد. بنابراین متغیر های صفر و یک γ_{ij} را به این صورت در نظر می گیریم که $\gamma_{ij} = 1$ هرگاه ماشین j به ناحیه i تخصیص داده شود و در غیر اینصورت $\gamma_{ij} = 0$. لذا مدل زیر را خواهیم داشت:

$$\begin{aligned}
 & \text{Min} \quad \sum_{1 \leq j, k \leq n} v_{jk} \|X_j - X_k\|_B + \sum_{j=1}^n \sum_{i=1}^m w_{ji} \|X_j - P_i\|_B \\
 & S/t/ \\
 & x_j \leq b_i + M(1 - \gamma_{ij}) \quad i = 1, \dots, K \quad j = 1, \dots, n \\
 & x_j \geq a_i - M(1 - \gamma_{ij}) \quad i = 1, \dots, K \quad j = 1, \dots, n \\
 & y_j \leq d_i + M(1 - \gamma_{ij}) \quad i = 1, \dots, K \quad j = 1, \dots, n \\
 & y_j \geq c_i - M(1 - \gamma_{ij}) \quad i = 1, \dots, K \quad j = 1, \dots, n \\
 & \sum_{i=1}^K \gamma_{ij} = 1 \quad j = 1, \dots, n \\
 & \sum_{j=1}^n \gamma_{ij} \leq 1 \quad i = 1, \dots, K \\
 & \gamma_{ij} \in \{0, 1\}
 \end{aligned} \tag{۵۵-۴}$$

چهار محدودیت اول مدل بیانگر این موضوع می باشند که اگر ماشین j به ناحیه i ام تخصیص داده شد، در درون این ناحیه قرار گیرد و بیرون آن نیافتد، در غیر اینصورت به این ناحیه محدود نشود. مثلاً وقتی $\gamma_{ij} = 1$ ، محدودیت های مسئله ایجاب می کنند که $a_i \leq x_j \leq b_i$ ، $c_i \leq y_j \leq d_i$ یعنی ماشین در ناحیه مربوطه قرار گیرد. اگر $\gamma_{ij} = 0$ آنگاه محدودیت ها ایجاب می کنند که $a_i - M \leq x_j \leq b_i + M$ ، $c_i - M \leq y_j \leq d_i + M$ یعنی ماشین j ام محدوده مستطیل i نمی شود. دو محدودیت دیگر مدل تضمین می کنند که هر ماشین فقط به یک ناحیه تخصیص داده شود و هر ناحیه حداکثر شامل یک ماشین باشد. مدل برنامه ریزی خطی زیر را داریم:

$$\begin{aligned}
 \text{Min} \quad & \sum_{1 \leq j < k \leq n} \sum_{g=1}^r v_{jk} (\lambda_{gjk}^+ + \lambda_{gjk}^-) + \sum_{j=1}^n \sum_{i=1}^m \sum_{g=1}^r w_{ji} (\beta_{gji}^+ + \beta_{gji}^-) \\
 \text{S.t.} \quad & X_j - X_k = \sum_{g=1}^r (\lambda_{gjk}^+ - \lambda_{gjk}^-) b_g \\
 & X_j - P_i = \sum_{g=1}^r (\beta_{gji}^+ - \beta_{gji}^-) b_g \\
 & \left. \begin{aligned} x_j &\leq b_i + M(1 - \gamma_{ij}) & x_j &\geq a_i - M(1 - \gamma_{ij}) \\ y_j &\leq d_i + M(1 - \gamma_{ij}) & y_j &\geq c_i - M(1 - \gamma_{ij}) \end{aligned} \right\} i = 1, \dots, K \quad j = 1, \dots, n \\
 & \sum_{i=1}^K \gamma_{ij} = 1 \quad j = 1, \dots, n \\
 & \sum_{j=1}^n \gamma_{ij} \leq 1 \quad i = 1, \dots, K \\
 & \gamma_{ij} \in \{0, 1\}, \lambda_{gjk}^+, \lambda_{gjk}^-, \beta_{gji}^+, \beta_{gji}^- \geq 0.
 \end{aligned} \tag{۵۶-۴}$$

که این مدل یک مدل برنامه ریزی خطی است.

مثال ۴-۱. فرض کنید می خواهیم مکان استقرار دو ماشین جدید در یک کارگاه بیابیم. این کارگاه شامل ۵ ماشین در مکانهای $P_1, \dots, P_5$ است. مختصات ماشینها در جدول ۴-۱ آمده است. همچنین چهار ناحیه برای استقرار ماشینهای جدید در دست است که مشخصات این نواحی در جدول ۴-۲ آمده است. هدف یافتن مکان این ماشینها بگونه ای است که مجموع فواصل مستطیلی ماشینهای جدید تا ماشینهای موجود کمینه گردد.

حل. اگر مدل (۵۶-۴) را برای نرم مستطیلی بنویسیم، مسئله (۵۷-۴) بدست می آید که یک مسئله

برنامه ریزی خطی است و به راحتی می توان آن را حل کرد. مدل مسئله فوق بصورت زیر است:

جدول ۴-۱. مختصات ماشینها

	P_1	P_2	P_3	P_4	P_5
x	۴	۲	۲	۴	۶
y	۴	۲	۵	۱	۵

جدول ۴-۲. مختصات نواحی برای استقرار ماشینهای جدید

y^U	y^L	x^U	x^L	
۷	۶	۲	۱	S_1
۴	۳	۳	۲	S_2
۷	۵	۵	۴	S_3
۳	۲	۶	۵	S_4

$$\min \sum_{1 \leq j < k \leq n} v_{jk} (p_{jk} + q_{jk}) + \sum_{j=1}^n \sum_{i=1}^m w_{ji} (r_{ji} + s_{ji})$$

$$+ \sum_{1 \leq j < k \leq n} v_{jk} (p'_{jk} + q'_{jk}) + \sum_{j=1}^n \sum_{i=1}^m w_{ji} (r'_{ji} + s'_{ji})$$

$St.$

$$x_j - x_k - p_{jk} + q_{jk} = 0$$

$$y_j - y_k - p'_{jk} + q'_{jk} = 0$$

$$x_j - r_{ji} + s_{ji} = px_i$$

$$y_j - r'_{ji} + s'_{ji} = py_i$$

$$x_j \leq b_i + M(1 - \gamma_{ij}) \quad i = 1, \dots, K \quad j = 1, \dots, n$$

$$x_j \geq a_i - M(1 - \gamma_{ij}) \quad i = 1, \dots, K \quad j = 1, \dots, n$$

$$y_j \leq d_i + M(1 - \gamma_{ij}) \quad i = 1, \dots, K \quad j = 1, \dots, n$$

$$y_j \geq c_i - M(1 - \gamma_{ij}) \quad i = 1, \dots, K \quad j = 1, \dots, n$$

$$\sum_{i=1}^K \gamma_{ij} = 1 \quad j = 1, \dots, n$$

$$\sum_{j=1}^n \gamma_{ij} \leq 1 \quad i = 1, \dots, K$$

$$\gamma_{ij} \in \{0, 1\}$$

$$p_{jk}, q_{jk}, p'_{jk}, q'_{jk}, r_{ji}, s_{ji}, r'_{ji}, s'_{ji} \geq 0$$

(۵۷-۴)

از حل این مدل جواب $X_1^* = (4, 5) \in S_3$, $X_2^* = (3, 4) \in S_2$ با مقدار تابع هدف $Z^* = 30$ بدست می

آید.

فصل ۵

مسئله وبر با جاذبه و دافعه

۱-۵ مقدمه

مسئله وبر یکی از پرکاربردترین مسائل نظریه مکانیابی است که مطالعات زیادی روی آن انجام شده است. برای مطالعه کامل این مسئله و انواع آن می توانید به [۱۳ و ۱۰۹ و ۱۱۰] مراجعه کنید. مسئله وبر به این صورت است که فرض کنید n نقطه در صفحه داده شده است و متناظر با هر راس یک وزن تخصیص داده شده است. اگر وزن مثبت باشد یعنی هزینه با افزایش فاصله، زیاد می شود و اگر منفی باشد یعنی با افزایش فاصله، هزینه کم می شود. هدف پیدا کردن مکان یک سرویس دهنده جدید است به قسمی که مجموع وزنی فاصله بین سرویس دهنده جدید و نقاط موجود کمینه گردد، یعنی

$$\text{Min } F(X) = \sum_{i=1}^n w_i d_i(X) \quad (1-5)$$

که در آن X ، مکان سرویس دهنده است.

w_i ، وزن متناظر با نقطه i است.

$d_i(X)$ فاصله L_p بین نقطه i و سرویس دهنده است.

در ادامه با نرم های مستطیلی و اقلیدسی، مسئله را مورد بررسی قرار می دهیم که حالت خاصی از نرم

L_p هستند. اگر $P_i = (x_i, y_i)$ مختصات نقطه i باشد، فاصله $d_i(X)$ برابر است با:

$$d_i(X) = d_i(x, y) = \left[|x_i - x|^p + |y_i - y|^p \right]^{\frac{1}{p}} \quad (2-5)$$

اگر وزن متناظر با راس منفی باشد معنی اش این است که نزدیک شدن به مشتری هزینه را افزایش می دهد. لذا می توان این طور تعبیر کرد که نقاط تقاضا، سرویس دهنده را جذب یا دفع می کنند و

در جواب بهینه این دو نیرو به تعادل می رسند. این تعبیر اولین بار توسط تلیر [۱۱۰] و تلیر و ککالدی [۱۱۱] ارائه شد. حال وقتی بعضی وزن ها مثبت و بعضی دیگر منفی هستند، مسئله را مسئله وبر با جاذبه و دافعه^۱ نامیده و وضعیت را به این صورت تعبیر می کنند که سرویس دهنده برای بعضی مشتریان نامطلوب است و به برخی دیگر خدمات می رساند. به عنوان مثال فرض کنید که می خواهیم یک کارخانه احداث کنیم و نقاط تقاضا نواحی مسکونی باشد. ساکنان نواحی مسکونی دوست دارند تا حد امکان از کارخانه دور باشند در حالی که برخی واحد های صنعتی از نزدیک بودن به آن سود می برند و برایشان مفید است.

تلیر [۱۱۲] مسئله وبر با جاذبه و دافعه را با سه نقطه تقاضا حل کرد. بعدا تلیر و پلانسکی در سال ۱۹۸۹ [۱۱۳] درزنر و وسلوسکی در سال ۱۹۹۱ [۱۰۴] و کرارپ در سال ۱۹۹۸ [۲۱] مسئله را در حالت کلی و با هر تعداد نقطه تقاضا حل کردند. ماراناس و فلوداس [۱۱۴] و توی و همکاران [۱۰۰] مثال های بزرگ مقیاس از مسئله وبر با جاذبه و دافعه را حل کردند. این نوع به کارگیری وزن های مثبت و منفی برای مسائل جایابی نیز به کار برده شده است که در این مورد می توانید به [۱۱۵] مراجعه کنید.

قضیه ای که در زیر بیان می شود برای هر تابع فاصله ای که در نامساوی مثلث صدق کند درست است (مجذور فاصله اقلیدسی در نامساوی صدق نمی کند). قرار می دهیم:

$$W = \sum_{i=1}^n w_i$$

قضیه ۵-۱: ([۱۰۴]). اگر $W > 0$ آنگاه مکان بهینه متناهی است و اگر $W < 0$ مکان بهینه نامتناهی است.

اثبات. نقطه (x, y) را در نظر بگیرید. اگر $D(x, y)$ فاصله (x, y) تا مبدا باشد، بنابر نامساوی مثلث و برای یک i داریم:

$$D(x, y) - d_i(.,.) \leq d_i(x, y) \leq D(x, y) + d_i(.,.)$$

حال اگر $w_i \geq 0$ برای این i داریم:

$$w_i D(x, y) - w_i d_i(.,.) \leq w_i d_i(x, y) \leq w_i D(x, y) + w_i d_i(.,.)$$

و اگر $w_i < 0$ آنگاه

$$w_i D(x, y) + w_i d_i(.,.) \leq w_i d_i(x, y) \leq w_i D(x, y) - w_i d_i(.,.)$$

که این ها معادل اند با

$$w_i D(x, y) - |w_i| d_i(.,.) \leq w_i d_i(x, y) \leq w_i D(x, y) + |w_i| d_i(.,.)$$

و این رابطه برای هر i برقرار است. حال با جمع زدن طرفین نامساوی ها روی i داریم:

$$WD(x, y) - U \leq F(x, y) \leq WD(x, y) + U \quad (3-5)$$

که در آن $U = \sum_{i=1}^n |w_i| d_i(.,.)$ ، اگر $(x, y) \rightarrow \infty$ آنگاه $D(x, y) \rightarrow \infty$. اگر $W > 0$ آنگاه

$F(x, y) \rightarrow \infty$ چون حد نامساوی سمت چپ ∞ است. این نشان می دهد که جواب بهینه متناهی

موجود است. اگر $W < 0$ آنگاه $F(x, y) \rightarrow -\infty$ زیرا نامساوی راست به $-\infty$ میل می کند و این یعنی

جواب بهینه نامتناهی است. $\square$

تعریف. درجه منفی بودن یک مسئله، α ، را به صورت زیر تعریف می کنیم:

$$W^+ = \sum_{w_i > 0} w_i$$

$$W^- = - \sum_{w_i < 0} w_i$$

$$\alpha = \frac{W^-}{W^+}$$

که $W = W^+ + W^-$ حال قضیه ۵-۱ را در قالب درجه منفی بودن به این صورت می توان مطرح کرد که اگر $\alpha < 1$ آنگاه جواب متناهی است و اگر $\alpha > 1$ آنگاه جواب نامتناهی است.

۲-۵ مسئله وبر با جاذبه و دافعه و نرم مستطیلی

مسئله وبر با جاذبه و دافعه با نرم مستطیلی را به روش های مشابه با آنچه برای مسئله وبر با وزن مثبت آمده است می توان حل کرد. توجه می کنیم که وقتی بعضی وزن ها منفی می شوند، مسئله ممکن است محدب نباشد و ممکن است شامل چندین بهینه موضعی باشد. بنابراین ویژگی های مسئله به صورت زیر خلاصه می کنیم:

(۱) مسئله را می توان به دو مسئله یک بعدی تجزیه کرد و جواب بهینه از ترکیب جواب بهینه هر کدام از مسائل بدست می آید. بنابراین از این به بعد مسئله یک بعدی را در نظر می گیریم و $F(x, y)$ به $F(x)$ تبدیل می شود.

(۲) تابع هدف و مشتق آن در نقطه x که یک نقطه تقاضا نیست به صورت زیراند:

$$F(x) = \sum_{i=1}^n w_i |x - x_i| = \sum_{x > x_i} w_i |x - x_i| + \sum_{x < x_i} w_i |x - x_i| = \sum_{x > x_i} w_i (x - x_i) - \sum_{x < x_i} w_i (x - x_i)$$

$$F'(x) = \sum_{x > x_i} w_i - \sum_{x < x_i} w_i$$

در نتیجه مشتق برای نقاط بین دو نقطه تقاضای متوالی ثابت است.

(۳) ویژگی قبل نشان می دهد که در مسائل ۱- بعدی ممکن است بیش از یک جواب بهینه داشته باشیم، اما حداقل یکی از جواب ها نقطه تقاضاست. نقطه تقاضا x_i یک بهینه موضعی است اگر و تنها

اگر $\left| \sum_{x > x_i} w_i - \sum_{x < x_i} w_i \right| \leq w_i$ نتیجه اینکه تنها نقاطی که دارای وزن منفی نیستند می توانند مینیمم

باشند. برای حل مسئله الگوریتم زیر را به کار می بریم

الگوریتم ۱:

$$(۱) \quad x_i \leq x_{i+1} \text{ یعنی } x_i \text{ مرتب می کنیم}$$

$$(۲) \quad \text{قرار می دهیم } W_1 = -\sum_{i=1}^n w_i \text{ اگر } W_1 > 0 \text{ آنگاه توقف کن جواب نامتناهی است.}$$

$$(۳) \quad \text{مقدار تابع هدف را در } x_1 \text{ محاسبه کن، } F_1 = \sum_{i=2}^n w_i (x_i - x_1) \text{ قرار می دهیم}$$

$$i=1, \quad x^* = x_1, \quad F^* = F_1$$

$$(۴) \quad \text{مقادیر } F_{i+1} = f_i + W_{i+1} (x_{i+1} - x_i), \quad W_{i+1} = W_i + 2w_i \text{ محاسبه می کنیم. } W_i \text{ برابر است با}$$

مشتق چپ $F(x)$ و W_{i+1} برابر است با مشتق راست $F(x)$.

$$(۵) \quad \text{اگر } F_{i+1} < F^*, x^* \text{ آنگاه } F^*, x^* \text{ را بروز می کنیم.}$$

$$(۶) \quad \text{قرار می دهیم } i=i+1 \text{ و اگر } i \leq n \text{ برو به گام ۴.}$$

$$(۷) \quad \text{در غیر صورت توقف کن } x^* \text{ جواب بهینه و } F^* \text{ مقدار بهینه تابع هدف است.}$$

پیچیدگی الگوریتم فوق از مرتبه $O(n \log n)$ است. زیرا یک مرحله مرتب سازی در گام ۲ داریم و

بقیه الگوریتم از مرتبه $O(n)$ است. لذا اگر بردار x مرتب شده باشد، پیچیدگی الگوریتم به $O(n)$

کاهش می یابد.

۳-۵ مسئله وبر با جاذبه و دافعه و مجذور نرم اقلیدسی

ویژگی قابل توجهی که مسئله با مجذور نرم اقلیدسی دارد اینست که جواب آن را می توان به عنوان نقطه شروعی برای مسئله وبر با جاذبه و دافعه و نرم اقلیدسی منظور کرد. در حالتی که وزن ها مثبت هستند برای حل مسئله باید مرکز ثقل نقاط را پیدا کنیم و این نقطه جواب مسئله است. لذا از (۵-۱) داریم:

$$F(x, y) = \sum_{i=1}^n w_i [(x - x_i)^2 + (y - y_i)^2]$$

$$F(x, y) = W(x^2 + y^2) - 2x \sum_{i=1}^n w_i x_i - 2y \sum_{i=1}^n w_i y_i + \sum_{i=1}^n w_i (x_i^2 + y_i^2) \quad (۴-۵)$$

لم ۲-۵: ([۱۰۴]). $F(x, y)$ محدب است اگر $W > 0$ و مقعر است اگر $W < 0$ و لذا خطی خواهد بود هرگاه $W = 0$.

حال اگر مشتقات جزئی (۴-۵) را برابر صفر قرار دهیم، خواهیم داشت:

$$\begin{aligned} x^* &= \frac{\sum_{i=1}^n w_i x_i}{W} \\ y^* &= \frac{\sum_{i=1}^n w_i y_i}{W} \end{aligned} \quad (۵-۵)$$

برای اثبات لم ۲-۵ از ماتریس هسین $F(x, y)$ استفاده می کنیم.

$$H(F(x, y)) = \begin{bmatrix} 2W & 0 \\ 0 & 2W \end{bmatrix} \text{ و در آن } \det(H(F(x, y))) = 4W^2. \text{ به وضوح اگر } W > 0 \text{ آنگاه } F$$

محدب است و اگر $W < 0$ آنگاه F مقعر است. $\square$

لم ۵-۳: ([۱۰۴]). اگر $W > 0$ مینیمم $F(x, y)$ در (x^*, y^*) به دست می آید و اگر $W < 0$ جواب

نامتناهی خواهد بود. اگر $W = 0$ آنگاه جواب نامتناهی است. اگر $W = 0$ و همچنین

$$\sum_{i=1}^n w_i x_i = \sum_{i=1}^n w_i y_i = 0$$

آنگاه تابع هزینه در سراسر صفحه ثابت است و هر نقطه ای جواب بهینه است

اثبات. اثبات قسمت اول از قضیه ۵-۱ به دست می آید. اگر $W = 0$ آنگاه طبق (۵-۵)، (x^*, y^*)

نامتناهی خواهند بود. اگر $W = 0$ و $\sum_{i=1}^n w_i x_i = \sum_{i=1}^n w_i y_i = 0$ آنگاه از (۴-۵) داریم

$$F(x, y) = \sum_{i=1}^n w_i (x_i^2 + y_i^2) \quad \square$$

که مقداری ثابت است و لذا هر نقطه در صفحه بهینه است.

۴-۵ مسئله وبر با جاذبه و دافعه و نرم اقلیدسی

در حالتی که نرم فاصله را نرم اقلیدسی در نظر بگیریم، داریم:

$$F(x, y) = \sum_{i=1}^n w_i \sqrt{(x - x_i)^2 + (y - y_i)^2}$$

اگر تمام وزن ها مثبت باشند آنگاه تابع هدف محدب است و اگر همگی منفی باشند مقعر است. اگر

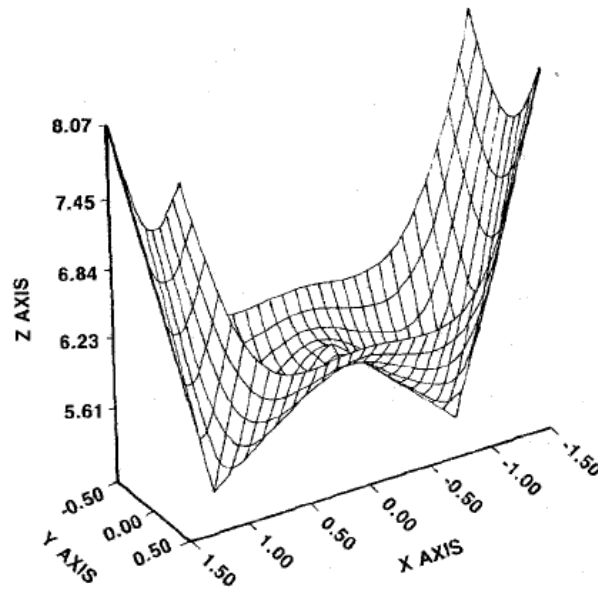
بعضی از وزن ها منفی باشند خاصیت تحدب تابع هدف از بین می رود. برای مثال نقاط

$$(1, 0), (0, 0), (-1, 0)$$

با وزن های $3, -1, 3$ در نظر بگیرید.

تابع هدف در شکل (۵-۱) رسم شده است [۱۰۴]. همانطور که از شکل پیداست تابع هدف نه محدب

است نه مقعر و دو مینیمم موضعی در $(1, 0), (-1, 0)$ دارد.



شکل ۵-۱. تابع هدف با سه نقطه

یکی از روش هایی که برای حل خانواده مسائل وبر به کار می رود، روش وایزفلد^۲ است. الگوریتم با نقطه شروع $(x^{(0)}, y^{(0)})$ آغاز می شود و روند تکراری زیر را دنبال می کند:

$$\begin{aligned}
 x^{(k+1)} &= \frac{\sum_{i=1}^n \frac{w_i x_i}{d_i(x^{(k)}, y^{(k)})}}{\sum_{i=1}^n \frac{w_i}{d_i(x^{(k)}, y^{(k)})}} \\
 y^{(k+1)} &= \frac{\sum_{i=1}^n \frac{w_i y_i}{d_i(x^{(k)}, y^{(k)})}}{\sum_{i=1}^n \frac{w_i}{d_i(x^{(k)}, y^{(k)})}}
 \end{aligned}
 \tag{۶-۵}$$

حال تعریف می کنیم $W' = \sum_{i=1}^n \frac{w_i}{d_i(x^{(k)}, y^{(k)})}$ به سادگی می توان بررسی کرد که

$$(x^{(k+1)}, y^{(k+1)}) = (x^{(k)}, y^{(k)}) + \frac{1}{W'} \nabla F(x^{(k)}, y^{(k)})$$

لم ۴-۵: ([۱۰۴]). اگر $W' > 0$ آنگاه $(x^{(k+1)}, y^{(k+1)})$ روی خط تندترین کاهش قرار دارد و اگر $W' < 0$ آنگاه روی خط تندترین افزایش است.

لذا اگر $W' > 0$ از $(x^{(k+1)}, y^{(k+1)})$ و اگر $W' < 0$ از $(x^{(k)}, y^{(k)}) - (x^{(k+1)}, y^{(k+1)})$ به عنوان نقطه شروع تکرار بعد استفاده می کنیم.

وقتی $W' = 0$ تکرار بعدی را نمی توانیم اجرا کنیم و تعریف نشده است. طول گام روش وایزفلد در هر تکرار $\frac{1}{W'}$ برابر مقدار گرادیان است لذا اگر مقدار W' کوچک باشد جواب تکرار بعد بزرگ می شود و ممکن است به واگرایی روش بیانجامد. برای کنترل مقدار طول گام تعریف می کنیم $W'_i = \frac{w_i}{d_i(x^{(k)}, y^{(k)})}$ طبق محاسباتی که در [۱۰۴] انجام شده است نتایج زیر برای هر نقطه (x, y) در صفحه به دست می آید،

$$\begin{aligned} F(x, y) = F(x^{(k)}, y^{(k)}) + G^* + W' \left[(x - x^{(k+1)})^2 + (y - y^{(k+1)})^2 \right] \\ - \sum_{i=1}^n W'_i [d_i(x, y) - d_i(x^{(k)}, y^{(k)})]^2 \end{aligned} \quad (۷-۵)$$

$$G^* = \sum_{i=1}^n W'_i [x_i^2 + y_i^2 - \{x^{(k+1)}\}^2 - \{y^{(k+1)}\}^2]$$

که در آن

فرض کنیم که در تکرار بعد الگوریتم به نقطه $(x(\lambda), y(\lambda))$ حرکت کنیم که

$$(x(\lambda), y(\lambda)) = (x^{(k)}, y^{(k)}) + \lambda [(x^{(k+1)}, y^{(k+1)}) - (x^{(k)}, y^{(k)})] \quad (۸-۵)$$

با به کار بردن تقریب های

$$d_i(x(\lambda), y(\lambda)) = d_i(x^{(k)}, y^{(k)}) \approx \lambda [d_i(x^{(k+1)}, y^{(k+1)}) - d_i(x^{(k)}, y^{(k)})]$$

$$\gamma F(x(\lambda), y(\lambda)) \approx \delta + (\lambda - 1)^\gamma \beta - \lambda^\gamma \gamma \quad (9-5)$$

که در آن

$$\begin{aligned} \delta &= F(x^{(k)}, y^{(k)}) + G^* \\ \beta &= W' \left[(x^{(k+1)} - x^{(k)})^\gamma + (y^{(k+1)} - y^{(k)})^\gamma \right] \\ \gamma &= \sum_{i=1}^n W_i \left[d_i(x^{(k+1)}, y^{(k+1)}) - d_i(x^{(k)}, y^{(k)}) \right]^\gamma \end{aligned}$$

و با صفر قرار دادن مشتق $F(x(\lambda), y(\lambda))$ نسبت به λ در (۹-۵) خواهیم داشت: $d = \frac{\beta}{\beta - \gamma}$

که نتیجه می دهد تکرار بعدی روش باید به صورت زیر باشد:

$$(x', y') = \frac{[\beta(x^{(k+1)}, y^{(k+1)}) - \gamma(x^{(k)}, y^{(k)})]}{\beta - \gamma} \quad (10-5)$$

برای برخورد نکردن با مشکل $W' = 0$ نیز W_x, W_y را به صورت زیر تعریف می کنیم

$$(W_x, W_y) = W'(x^{(k+1)}, y^{(k+1)})$$

لذا با جایگذاری تعریف فوق در رابطه (۱۰-۵) داریم:

$$(x', y') = \frac{[\beta(W_x, W_y) - \gamma'(x^{(k)}, y^{(k)})]}{W'\beta' - \gamma'} \quad (11-5)$$

که

$$\begin{aligned} \beta' &= W'\beta = \text{sgn}(W') \left[(W_x - W'x^{(k)})^\gamma + (W_y - W'y^{(k)})^\gamma \right] \\ \gamma' &= W''\gamma = \sum_{i=1}^n W_i \left[\sqrt{(W_x - W'x^{(k)})^\gamma + (W_y - W'y^{(k)})^\gamma} - W'd_i(x^{(k)}, y^{(k)}) \right]^\gamma \end{aligned}$$

قابل ذکر است که اگر مشتق دوم $F(x(\lambda), y(\lambda))$ نسبت به λ ، یعنی $\beta - \lambda$ (یا $W'\beta' - \gamma'$) مثبت باشد آنگاه λ فوق مینیمم F را می دهد. اگر $\beta - \lambda$ منفی باشد آنگاه مینیمم F در رابطه (۵-۹) به ازای $\lambda = \pm\infty$ به دست می آید.

نتایج فوق را در الگوریتمی که از خانواده روش های گرادیانی است خلاصه می کنیم. توجه داریم که در روش گرادیانی استاندارد نیاز به محاسبه تابع هدف و گرادیان در نقاط زیادی می باشد. پس روی خط گرادیان به جست و جو پرداخته مقداری که به ازای آن می نیمم تابع تقریب زده می شود را می یابیم و آن را به عنوان طول گام استفاده می کنیم.

الگوریتم ۲: روش وایزفلد اصلاح شده ([۱۰۴])

(۱) مقدار W را محاسبه می کنیم. اگر $W < 0$ توقف کن، جواب نامتناهی است. نقطه شروع $(x^{(0)}, y^{(0)})$ را انتخاب می کنیم. مثلاً می توانیم مرکز ثقل را انتخاب کنیم و شمارنده تکرارها را قرار می دهیم $k = 0$.

(۲) از (۳۹-۴) نقطه (x', y') را محاسبه می کنیم. اگر $(x^{(k)}, y^{(k)})$ یک نقطه تقاضا بود، آنگاه در جهت تند ترین کاهش به اندازه بسیار کمی فاصله بگیریم.

(۳) اگر $W'\beta' - \gamma' > 0$ قرار دهید $(x', y') = (x^{(k+1)}, y^{(k+1)})$ و به گام ۷ بروید.

(۴) و اگر نه $(x(\lambda), y(\lambda))$ را با $\lambda = 1$ ، طبق (۵-۸) محاسبه کنید.

(۵) اگر $F(x(\lambda), y(\lambda)) < F(x^{(k)}, y^{(k)})$ قرار دهید $(x(\lambda), y(\lambda)) = (x^{(k+1)}, y^{(k+1)})$ و به گام ۷ بروید.

(۶) و اگر نه قرار دهید $\lambda = \frac{\lambda}{\gamma}$ و به گام ۵ برگردید.

(۷) قرار دهید $k = k + 1$ اگر فاصله بین $(x^{(k)}, y^{(k)})$ و $(x^{(k+1)}, y^{(k+1)})$ کمتر از خطای از پیش تعیین شده ε بود، توقف کنید، $(x^{(k+1)}, y^{(k+1)})$ جواب بهینه است و اگر نه به گام ۲ بروید.

روش های دیگری برای این مسئله تا الان به کار برده شده است که از آن جمله می توان به روش های مستطیل بزرگ مستطیل کوچک مثلث بزرگ مثلث کوچک و اخیر روش مکعب بزرگ مکعب کوچک [۳۰] را نام برد.

۵-۵ قضیه موضعی سازی

در این قسمت ناحیه ای که شامل جواب بهینه است را مشخص می کنیم. اگر $\alpha = 0$ (یعنی تمام وزن ها مثبت باشند) به راحتی می توان نشان داد که جواب بهینه در پوسته محدب نقاط تقاضا قرار دارد. این واقعیت را می توان از رابطه تکراری وایزفلد به دست آورد. در هر صورت برای مطالعه بیشتر در این زمینه به [۱۱۶ و ۱۱۷ و ۱۱۸] مراجعه کنید.

حال کوچکترین دایره ای که شامل تمام نقاط تقاضا با مرکز c و شعاع R است را در نظر بگیرید.

اگر $0 < \alpha < 1$ در قضیه ۱۲ ثابت می شود که جواب بهینه در دایره ای با مرکز c و شعاع $\frac{R}{\sqrt{1-\alpha^2}}$

است. اگر $\alpha \geq 1$ آنگاه جواب نامتناهی خواهد بود، این قضیه در مرجع [۱۰۴] آمده است.

قضیه ۱۲: مجموعه جواب های بهینه شامل موضعی و سراسری در دایره ای به مرکز c و شعاع

$$\frac{R}{\sqrt{1-\alpha^2}} \text{ قرار دارند.}$$

اثبات. نقطه $(x, y) \neq (x_i, y_i)$ را از نقطه (x_i, y_i) به اندازه $d_i(x, y)$ واحد فاصله دارد در نظر می

گیریم. فرض کنید نقطه (x, y) را به اندازه ناچیز ε در جهتی که با پاره خط واصل (x, y) تا

(x_i, y_i) زاویه θ می سازد، جا به جا کنیم. واضح است که اثر این در $d_i(x, y)$ به صورت زیر محاسبه می شود:

$$\Delta d_i(x, y) = -\varepsilon \cos \theta \quad (۱۳-۵)$$

حال نقطه (x, y) را به گونه ای در نظر بگیرید که فاصله اش از c بیشتر از $\frac{R}{\sin \theta}$ باشد. اگر به اندازه بسیار کوچک ε از نقطه (x, y) به طرف c حرکت کنیم، چون زاویه بین حرکت در جهت هر نقطه تقاضا بین صفر و θ است، بنا بر (۱۳-۵) خواهیم داشت:

$$-\varepsilon \leq \Delta d_i(x, y) \leq -\varepsilon \cos \theta \quad (۱۴-۵)$$

بنابراین تغییرات ایجاد شده در تابع هدف در اثر این حرکت به صورت زیر است:

$$\begin{aligned} \Delta F(x, y) &= \sum_{i=1}^n w_i \Delta d_i(x, y) < -\sum_{w_i > 0} w_i \varepsilon \cos \theta - \sum_{w_i < 0} w_i \varepsilon \\ &= \varepsilon [-W^+ \cos \theta + W^-] = \varepsilon [W^+ \alpha - W^+ \cos \theta] = \varepsilon W^+ [\alpha - \cos \theta] \end{aligned} \quad (۱۵-۵)$$

بنابراین برای نقطه ای که در آن $\cos \theta > \alpha$ آنگاه $\Delta WF(x, y) < 0$ و این یعنی (x, y) نمی تواند می نیمم موضعی باشد، بنابراین در بهینه موضعی (بهینه سراسری) باید شرط $\cos \theta \leq \alpha$ برقرار باشد. اما چون $\sin \theta = \sqrt{1 - \cos^2 \theta}$ آنگاه $\cos \theta \leq \alpha$ معادل است با $\sin \theta \geq \sqrt{1 - \alpha^2}$ لذا جواب بهینه در

دایره ای با مرکز c و شعاع $\frac{R}{\sqrt{1 - \alpha^2}}$ قرار دارد. $\square$

فصل ۶

مسئله مکانیابی آرمانی

۱-۶ مقدمه

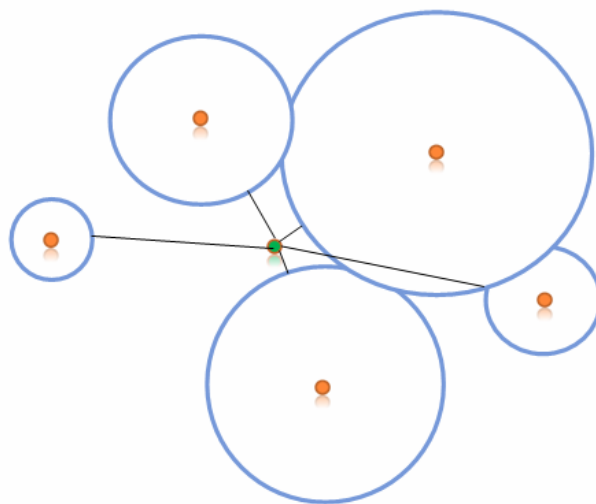
فرض کنید که می خواهیم فروشگاه‌های را در یک شهر احداث کنیم. با توجه به ساختار زندگی شهری نحوه و توزیع جمعیت، هر چه فروشگاه به مرکز شهر و مناطق پر جمعیت نزدیک تر باشد، از یک طرف دسترسی ساکنان افزایش یافته و متوسط هزینه حمل و نقل آن ها کاسته شده و تقاضای مشتریان به خوبی ارضا می شود و از طرف دیگر احداث فروشگاه در چنین جایی هزینه های زیادی در پی دارد، مالیات که افزایش فروشگاه باید پرداخت کند، میزان فضایی که در این منطقه باید به آن اختصاص داده شود، شلوغی، سرو صدا و ترافیک ناشی از آن که تردد را دچار مشکل می کند همگی از جمله اثرات نامطلوب مشکلات نزدیکی فروشگاه به مرکز شهر است. اگر این فروشگاه را تا حد امکان دور از مرکز شهر احداث کنیم، ضمن کاهش اثرات نامطلوب آن بر مشتریان، اثرات مثبت آن هم کاهش می یابد، دوری از مرکز شهر، افزایش هزینه و زمان رفت و آمد، کاهش دسترسی را در پی دارد. در حالت اخیر اثرات مطلوب نزدیکی به مرکز شهر به عنوان اثرات نامطلوب و اثرات نامطلوب آن به عنوان اثرات مطلوب دوری از مرکز شهر تلقی می شوند. تصمیم گیری برای احداث چنین سرویس دهندگانی که در دنیای واقعی و زندگی شهری با آنها رو به رو هستیم و تعدادشان هم کم نیست، کار پیچیده و دشواری است.

ایده ای که برای مدل سازی این مسئله در [۱۱۹] پیشنهاد شده است، این است که برای دوری و نزدیکی از مرکز شهر یک کران مشترک در نظر گرفته شود. حال مسئله را از دو دیدگاه می توان مورد بررسی قرار داد و مدل سازی کرد. یکی از دید مدیران فروشگاه و دیگری از دید مشتریان و سرویس گیرندگان. از نظر مدیران، حالتی که سود آوری، جذب مشتری بیشتری و پوشش سهم بیشتری از بازار با کمترین هزینه را در پی داشته باشد، مطلوب تر است و از دید مشتریان، حالتی که دسترسی بهتر و آسان تر، قیمت های کمتر و نیز اثرات نامطلوب نظیر سروصدا و شلوغی و ترافیک کمتری برای

آنها داشته باشد مناسب تر و مطلوب تر است. بنابراین برای جلب رضایت مشتریان و مدیران از یک کران مشترک استفاده می کنیم. یعنی اگر فروشگاه در فاصله نزدیک تری از کران مزبور به مرکز شهر احداث کنیم هزینه ناشی از اثرات نامطلوب افزایش می یابد و اگر در فاصله بیشتری از آن احداث شود رضایتمندی مشتریان و سود آوری فروشگاه کاهش می یابد، لذا با تعیین یک کران مشخص برای فاصله از مرکز شهر مسئله را با هدف کمترین مجموع مجذور فاصله از کران های مذکور مدل می کنیم. در حقیقت میزان خطای دوری از این کران را مینیمم می کنیم و به عبارتی مسئله تعمیمی از مسئله وبر خواهد بود. مدل مسئله در حالت کلی به صورت زیر است:

$$\text{Min } F(X) = \sum_{i=1}^n (d(X, P_i) - r_i)^2 \quad (1-6)$$

که P_i نقاط و مناطق موجود، r_i ها شعاع های متناظر با هر منطقه و $d(X, P_i)$ فاصله نقطه مجهول تا سرویس گیرنده P_i است. این مسئله در شکل (۱-۶) برای پنج نقطه تقاضا با شعاع های متفاوت نشان داده شده است.



شکل ۱-۶: نمایش گرافیکی مسئله مکانیابی آرمانی

اگر d فاصله اقلیدسی باشد مسئله را مسئله تک وسیله ای و بر آرمانی می نامیم و یک روش تکراری نقطه ثابت شبه وایزفلد برای آن ارائه می کنیم و همگرایی سراسری آن را ثابت می کنیم. همچنین مسئله را برای حالتی که $d(.)$ یک نرم بلوکی است بررسی و یک مدل برنامه ریزی خطی برای آن ارائه می کنیم.

به عنوان مثالی از این مسئله، اگر تمام نقاط تقاضا روی دایره ای به شعاع r قرار داشته باشند و شعاع آرمانی مفروض برای هر سرویس گیرنده برابر r باشد یعنی $\forall i \quad r_i = r$ ، آنگاه مرکز دایره فوق جواب بهینه مسئله خواهد بود و مقدار تابع هدف برابر صفر است. قابل ذکر است که این مسئله با مسئله پوشش متفاوت است زیرا در مسئله پوشش تعداد سرویس دهنده ها مشخص نیست و در آن به دنبال کمترین تعداد سرویس دهنده هستیم به گونه ای که فاصله هر مشتری با نزدیکترین سرویس دهنده به آن کمتر از شعاع مربوطه باشد.

با توجه به توجیه و تعبیر مسئله مکانیابی آرمانی، این مسئله کاربردهای زیادی در استقرار استادیوم ها، کارخانه ها، ماشین آلات، نیروگاه ها، اماکن دفن زباله و سرویس دهندگانی از این قبیل که هم دارای تاثیرات مثبت و هم تاثیرات منفی هستند، می توانند داشته باشند.

۲-۶ خواص مسئله

مدلبندی مسئله در بخش قبل انجام شد. در این بخش به بررسی خواص این مسئله می پردازیم. فتحعلی و همکاران [۱۱۹] نشان دادند که تابع هدف محدب نیست و لذا ممکن است دارای بهینه موضعی زیادی باشد. آنها همچنین نشان دادند که جواب بهینه در پوسته مستطیل گسترش یافته قرار دارد. پوسته مستطیلی گسترش یافته کوچکترین مستطیل است که شامل نقاط تقاضا و دایره های متناظرشان است. زعفرانی و همکاران [۱۲۰] از لم فارکاس^۱ در حل مسئله مکانیابی تک

^۱ Farkas's Lemma

وسیله ای در صفحه با دو نرم متفاوت استفاده کردند. در این قسمت از لم فارکاس که از قضایای پر کاربرد بهینه سازی است، برای پیدا کردن ناحیه شامل جواب بهینه استفاده می کنیم. برای مطالعه بیشتر به [۱۲۱ و ۱۲۲] مراجعه شود.

لم ۱-۶ (فارکاس (۱۹۰۲)). اگر A یک ماتریس $m \times n$ و C یک n بردار باشد آنگاه تنها یکی از دو دستگاه زیر جواب دارد:

$$\begin{aligned} Cx < \bullet, \quad Ax &\geq \bullet \\ W &\geq \bullet, \quad WA = C \end{aligned}$$

قضیه ۲-۶. جواب بهینه مسئله (۱-۶) در پوسته مستطیلی گسترش یافته نقاط تقاضا قرار دارد.

برهان. ابتدا چهار نقطه زیر را تعریف می کنیم.

$$\begin{aligned} RH_{\downarrow} &= \min\{a_i - r_i \mid i = 1, \dots, n\} \\ RH_{\uparrow} &= \max\{b_i + r_i \mid i = 1, \dots, n\} \\ RH_{\downarrow} &= \max\{a_i + r_i \mid i = 1, \dots, n\} \\ RH_{\uparrow} &= \min\{b_i - r_i \mid i = 1, \dots, n\} \end{aligned}$$

نشان می دهیم که جواب بهینه از مستطیل تولید شده با این چهار نقطه به عنوان رئوس آن قرار دارد.

فرض خلف می کنیم که $X_b = \begin{pmatrix} x_b \\ y_b \end{pmatrix}$ جواب بهینه مسئله، خارج از مستطیل قرار داشته باشد.

بنابراین باید خارج از یکی از مخروط های تولید شده با $(RH_{\downarrow} - RH_{\uparrow}, \bullet), (\bullet, RH_{\downarrow} - RH_{\uparrow})$ یا $(RH_{\downarrow} - RH_{\uparrow}, \bullet), (\bullet, RH_{\downarrow} - RH_{\uparrow})$ باشد.

حالت ۱: فرض کنیم جواب بهینه خارج از مخروط تولید شده با $(RH_{\downarrow} - RH_{\uparrow}, \bullet), (\bullet, RH_{\downarrow} - RH_{\uparrow})$ قرار داشته باشد.

بنابراین بردار $X_b - (RH_{\downarrow}, RH_{\uparrow})^t$ در مخروط تولید شده قرار ندارد. یعنی دستگاه زیر جواب ندارد:

$$\begin{cases} w_{\lambda}(RH_{\lambda} - RH_{\tau}, \cdot) + w_{\tau}(\cdot, RH_{\lambda} - RH_{\tau}) = (x_b - RH_{\tau}, y_b - RH_{\tau}) \\ w_{\lambda}, w_{\tau} \geq \cdot \end{cases}$$

لذا بنا به لم فارکاس دستگاه زیر باید جواب داشته باشد:

$$\begin{cases} \begin{bmatrix} RH_{\lambda} - RH_{\tau} & \cdot \\ \cdot & RH_{\tau} - RH_{\lambda} \end{bmatrix} \begin{bmatrix} d_{\lambda} \\ d_{\tau} \end{bmatrix} \geq \cdot \\ \begin{bmatrix} x_b - RH_{\tau} & y_b - RH_{\tau} \end{bmatrix} \begin{bmatrix} d_{\lambda} \\ d_{\tau} \end{bmatrix} < \cdot \end{cases}$$

بنابراین داریم:

$$\begin{cases} (RH_{\lambda} - RH_{\tau})d_{\lambda} \geq \cdot \Rightarrow d_{\lambda} \leq \cdot \\ (RH_{\tau} - RH_{\lambda})d_{\tau} \geq \cdot \Rightarrow d_{\tau} \leq \cdot \end{cases}$$

بنابراین

$$(x_b - RH_{\tau})d_{\lambda} + (y_b - RH_{\tau})d_{\tau} < \cdot$$

از طرفی چون $i = 1, \dots, n$ $a_i \leq RH_{\tau}$ $b_i \leq RH_{\tau}$ لذا

$$(x_b - a_i)d_{\lambda} + (y_b - b_i)d_{\tau} \leq (x_b - RH_{\tau})d_{\lambda} + (y_b - RH_{\tau})d_{\tau} < \cdot$$

حال اگر $x_b - RH_{\tau} < \cdot$ و $y_b - RH_{\tau} > \cdot$ قرار می دهیم $d = \begin{pmatrix} \cdot \\ d_{\tau} \end{pmatrix}$ و اگر $x_b - RH_{\tau} > \cdot$ و $y_b - RH_{\tau} < \cdot$ قرار می دهیم

$y_b - RH_{\tau} < \cdot$ قرار می دهیم $d = \begin{pmatrix} d_{\lambda} \\ \cdot \end{pmatrix}$ و اگر $x_b - RH_{\tau} > \cdot$ و $y_b - RH_{\tau} > \cdot$ قرار می دهیم

$$d = \begin{pmatrix} d_{\lambda} \\ d_{\tau} \end{pmatrix} \text{ بنابراین}$$

$$\nabla d(X_b, P_i)d = \frac{1}{d(X_b, P_i)}(x_b - a_i, y_b - b_i)d < \cdot$$

در نتیجه داریم:

$$\begin{aligned}\nabla F(X_b)d &= \sum_{i=1}^n w_i \left(1 - \frac{r_i}{d(X_b, P_i)} \right) (x_b - a_i)d + \sum_{i=1}^n w_i \left(1 - \frac{r_i}{d(X_b, P_i)} \right) (y_b - b_i)d \\ &= \sum_{i=1}^n w_i \left(1 - \frac{r_i}{d(X_b, P_i)} \right) [(x_b - a_i)d + (y_b - b_i)d] < 0.\end{aligned}$$

و این یعنی d یک جهت کاهش است لذا X_b نمی تواند بهینه باشد. توجه داریم که در نامعادله آخر

عبارت $1 - \frac{r_i}{d(X_b, P_i)}$ برای تمام i ها مثبت است زیرا X_b خارج از پوسته مستطیل گسترش یافته

قرار دارد یعنی $r_i \leq d(X_b, P_i)$ بنابراین $1 - \frac{r_i}{d(X_b, P_i)} > 0$.

حالت ۲: فرض می کنیم که جواب بهینه خارج از مخروط تولید شده با بردار های

$(RH_1, \cdot), (RH_2, \cdot), (RH_3, \cdot), (RH_4, \cdot)$ باشد. اثبات این حالت مشابه با اثبات حالت ۱ است.

بنابراین جواب بهینه در پوسته مستطیلی تولید شده با RH_1, RH_2, RH_3, RH_4 قرار دارد. $\square$

برای شرط لازم بهینگی باید مشتقات جزئی $F(X)$ را برابر صفر قرار دهیم یعنی $\frac{\partial F(X^*)}{\partial X} = 0$ ، لذا،

$$\frac{\partial F}{\partial X} = \sum_{i=1}^n w_i \left(1 - \frac{r_i}{d(X, P_i)} \right) (X - P_i) = 0. \quad (2-6)$$

با برابر صفر قرار دادن ساده کردن معادله فوق، به دست می آوریم:

$$X = \frac{\sum_{i=1}^n w_i \left(1 - \frac{r_i}{d(X, P_i)} \right) P_i}{\sum_{i=1}^n w_i \left(1 - \frac{r_i}{d(X, P_i)} \right)} \quad (3-6)$$

حال اگر در معادله (۳-۶) سمت راست معادله را برحسب X_k و سمت چپ را X_{k+1} در نظر بگیریم، یعنی در حقیقت با استفاده از تکنیک نقطه ثابت، خواهیم داشت:

$$X_{k+1} = \frac{\sum_{i=1}^n w_i \left(1 - \frac{r_i}{d(X_k, P_i)} \right) P_i}{\sum_{i=1}^n w_i \left(1 - \frac{r_i}{d(X_k, P_i)} \right)} \quad (۴-۶)$$

پس با انتخاب یک جواب اولیه X و تغییر جزئی در معادله (۴-۶) رابطه زیر را داریم:

$$X_{k+1} = X_k - \frac{1}{\sum_{i=1}^n w_i \left(1 - \frac{r_i}{d(X_k, P_i)} \right)} \left[\sum_{i=1}^n w_i \left(1 - \frac{r_i}{d(X_k, P_i)} \right) (X_k - P_i) \right] \quad (۵-۶)$$

اگر قرار دهیم

$$d_k = -\frac{1}{\sum_{i=1}^n w_i \left(1 - \frac{r_i}{d(X_k, P_i)} \right)} (X_k - P_i) \quad (۶-۶)$$

واضح است که d_k یک جهت کاهشی است یعنی $d_k^T \cdot \nabla F(X_k) < 0$ پس معادله (۵-۶) یک روش گرادیانی است و دنباله $\{F(X_k)\}$ غیر صعودی است.

حال عبارت زیر را تعریف می کنیم:

$$h(X) = X - \frac{1}{\sum_{i=1}^n w_i \left(1 - \frac{r_i}{d(X, P_i)} \right)} \cdot \frac{\partial F(X)}{\partial X} \quad (۷-۶)$$

لم ۳-۶. تابع $h(X)$ به ازای $i = 1, \dots, n$ در نقطه $X \neq P_i$ پیوسته است.

برهان. تابع F دوبار مشتق پذیر است. طبق معادلات (۶-۷) و (۶-۵) و مشتق پذیری F ، نتیجه برقرار می شود. $\square$

لم ۴-۶. دنباله تولید شده از معادله (۶-۵) در پوسته آفینی نقاط P_i قرار دارد.

برهان. در هر تکرار از معادله (۶-۵) داریم، $X_{k+1} = \frac{\sum_{j=1}^n \alpha_j P_j}{\sum_{j=1}^n \alpha_j}$ که در آن $\alpha_i = w_i \left(1 - \frac{r_i}{d(X, P_i)} \right)$

قرار می دهیم $\gamma_i = \frac{\alpha_i}{\sum_{j=1}^n \alpha_j}$ که $\gamma_i \in [-1, 1]$ و $\sum_{i=1}^n \gamma_i = 1$. لذا دنباله تولید شده $\{X_k\}$ از معادله (۶-۶)

(۵) ترکیب آفینی از نقاط تقاضا هستند و این یعنی دنباله $\{X_k\}$ در پوسته آفینی نقاط P_i قرار دارد. $\square$

لم ۵-۶. حد رو به رو موجود است $\lim_{k \rightarrow \infty} F(X_k)$.

برهان. قبلا بیان شد که جهت d_k در تکرار k ام از روش، یک جهت کاهشی است و لذا دنباله

$\{F(X_k)\}$ غیر صعودی است. از طرفی چون این دنباله از پایین با صفر کران دار است (به وضوح از

صورت تابع به دست می آید) پس حد $\lim_{k \rightarrow \infty} F(X_k)$ موجود است. $\square$

لم ۶-۶. اگر $X_k = X_{k+1}$ آنگاه $\frac{\partial F(X_k)}{\partial X} = 0$.

برهان. از معادله (۶-۶) داریم $X_{k+1} = h(X_k) = X_k - \frac{1}{\gamma} \lambda_k \frac{\partial F(X_k)}{\partial X}$ که در آن

$\lambda_k = \frac{1}{\sum_{i=1}^n w_i \left(1 - \frac{r_i}{d(X_k, P_i)} \right)}$ اگر $X_k = X_{k+1}$ آنگاه $-\frac{1}{\gamma} \lambda_k \frac{\partial F(X_k)}{\partial X} = 0$ یعنی $\frac{\partial F(X_k)}{\partial X} = 0$. $\square$

لم ۶-۶. اگر $X_k \neq X_{k+1}$ آنگاه $F(X_{k+1}) < F(X_k)$.

برهان. اگر $X_k \neq X_{k+1}$ آنگاه از معادله (۶-۶) داریم که $\frac{\partial F(X_k)}{\partial X} \neq 0$. قرار می دهیم

$$\square \quad d = -\frac{\partial F(X_k)}{\partial X}, \text{ چون } d^T \nabla F(X_k) < 0 \text{ لذا } d \text{ یک جهت کاهشی است یعنی } F(X_{k+1}) < F(X_k).$$

لم ۶-۷. اگر $F(X_{k+1}) = F(X_k)$ آنگاه $X_K = X_{K+1}$.

برهان. چون جهت d_k برای هر k ، یک جهت غیرافزایشی است، طبق لم ۶-۶ داریم $X_K = X_{K+1}$.

قضیه ۶-۸. دنباله $\{X_k\}$ همگرا به یک بهینه موضعی یا نقطه زینی از تابع $F(X)$ است.

برهان. طبق لم ۶-۴ دنباله $\{X_k\}$ در پوسته آفینی نقاط تقاضا قرار دارد و چون جهت d_k در معادله

(۶-۶) یک جهت کاهشی است لذا دنباله $\{X_k\}$ به نقطه ای مثل $\hat{X}$ همگرا می شود. از طرفی حد

$$\lim_{k \rightarrow \infty} F(X_k) \text{ موجود است در نتیجه } \lim_{k \rightarrow \infty} F(X_{k+1}) - F(X_k) = 0 \text{ و } F(h(\hat{X})) = F(\hat{X}) \text{ طبق لم ۶-۷}$$

داریم $h(\hat{X}) = \hat{X}$. اگر $X_k \neq X_{k+1}$ آنگاه نامساوی $F(X_{k+1}) \leq F(X_k)$ به صورت اکید برقرار می

شود. از پیوستگی تابع $h(X)$ نتیجه می گیریم که اگر X_k به $\hat{X}$ بگراید آنگاه X_{k+1} به $\hat{X}$ $h(\hat{X}) = \hat{X}$

همگرا می شود و این یعنی $\{X_k\}$ به $\hat{X}$ همگرا می شود. بنا به لم ۶-۶، $\frac{\partial F(\hat{X})}{\partial X} = 0$ و چون

$\{F(X_k)\}$ غیر صعودی است و نیز از پایین کران دار است، نقطه $\hat{X}$ نمی تواند یک ماکزیمم موضعی

باشد یعنی یا مینیمم موضعی است یا یک نقطه زینی است. $\square$

۳-۶ مسئله مکانیابی تک وسیله ای با کمترین خطای مطلق

این مسئله در واقع همان مسئله وبر آرمانی است با این تفاوت که در جمله خطا بجای مجذور جملات

از قدر مطلق آنها استفاده می شود. در این بخش به بررسی مسئله با خطای مطلق و نرم بلوکی به

عنوان تابع فاصله می پردازیم. علت این امر اینست که این حالت را می توان به مسئله برنامه ریزی خطی تبدیل نموده و از الگوریتمهای مناسب برای حل آن استفاده کرد. فرضیات مسئله همان فرضیات مسئله وبر آرمانی است و برای جلوگیری از تکرار از ذکر آنها خودداری می کنیم و تنها به فرمولبندی مسئله و روش خطی سازی آن می پردازیم.

مسئله تک وسیله ای با حداقل خطای مطلق را به صورت زیر فرموله می کنیم:

$$\min_{x \in S} \sum_{i=1}^n w_i e(X, P_i)$$

که در آن P_i ها نقاط تقاضا، w_i ها میزان تقاضای متناظر با هر نقطه، S فضای شدنی و $e(X, P_i) = |d(X, P_i) - r_i|$ و r_i شعاع متناظر با هر نقطه می باشند.

جواب این مسئله دارای خواصی است که از آن جمله، جواب در پوسته مستطیلی نقاط تقاضا قرار دارد و اثبات از طریق لم فارکاس قبلا ارائه شد و خاصیت دیگر در قضیه زیر آمده است:

قضیه ۶-۹. اگر X^* جواب بهینه برای مسئله تک وسیله ای با نرم $\|\cdot\|_a$ باشد و داشته باشیم

$$\|X^* - P_i\|_a \geq r_i \quad \text{آنگاه } X^* \text{ جواب بهینه } \sum_{i=1}^n w_i \left| \|X - P_i\|_a - r_i \right| \text{ نیز هست، که در آن } w_j \geq 0 \text{ برای}$$

$$j = 1, \dots, n.$$

برهان. با توجه به تعریف مسئله داریم:

$$\begin{aligned}
\sum_{i=1}^n w_i \left| \|X^* - P_i\|_a - r_i \right| &= \sum_{i=1}^n w_i \left(\|X^* - P_i\|_a - r_i \right) \\
&= \sum_{i=1}^n w_i \|X^* - P_i\|_a - \sum_{i=1}^n w_i r_i \\
&\leq \sum_{i=1}^n w_i \|X - P_i\|_a - \sum_{i=1}^n w_i r_i \\
&= \sum_{i=1}^n w_i \left(\|X - P_i\|_a - r_i \right) \\
&\leq \sum_{i=1}^n w_i \left| \|X - P_i\|_a - r_i \right|
\end{aligned}$$

□

مدل مسئله با نرم بلوکی بقرار زیر است:

$$\text{Min } F(X) = \sum_{i=1}^n w_i \left| \|X - P_i\|_B - r_i \right|$$

حال طبق روشهایی که در بخش نرمهای بلوکی گفته شد، به دو طریق می توان مدل را بصورت خطی

بیان کرد:

(۱) برای هر $i = 1, \dots, n$ قرار می دهیم:

$$\begin{aligned}
\|X - P_i\|_B &= \sum_{g=1}^r (\lambda_{gi}^+ + \lambda_{gi}^-) \\
X - P_i &= \sum_{g=1}^r (\lambda_{gi}^+ - \lambda_{gi}^-) b_g
\end{aligned}$$

و قرار می دهیم $z_i^+ - z_i^- = \|X - P_i\|_B - r_i$ بنابر این مسئله بصورت زیر نوشته می شود:

$$\begin{aligned}
\text{Min } & \sum_{i=1}^n w_i (z_i^+ + z_i^-) \\
\text{S.t. } & \sum_{g=1}^r (\lambda_{gi}^+ + \lambda_{gi}^-) - z_i^+ + z_i^- = r_i \quad i = 1, \dots, n \\
& X - P_i = \sum_{g=1}^r (\lambda_{gi}^+ - \lambda_{gi}^-) b_g \quad i = 1, \dots, n \\
& z_i^+, z_i^- \geq 0 \quad i = 1, \dots, n \\
& \lambda_{gi}^+, \lambda_{gi}^- \geq 0 \quad g = 1, \dots, r
\end{aligned}$$

(۸-۶)

(۲) برای هر $i = 1, \dots, n$ قرار می دهیم:

$$y_i = \|X - P_i\|_B = \max \left\{ |(X - P_i)b_g| : g = 1, \dots, r \right\}$$

قرار می دهیم $z_i^+ - z_i^- = y_i - r_i$ لذا داریم:

$$\begin{aligned} \text{Min} \quad & \sum_{i=1}^n w_i (z_i^+ + z_i^-) \\ \text{S.t.} \quad & z_i^+ - z_i^- = y_i - r_i \quad i = 1, \dots, n \\ & (X - P_i)b_g \leq y_i \quad g = 1, \dots, r, i = 1, \dots, n \\ & -(X - P_i)b_g \leq y_i \quad g = 1, \dots, r, i = 1, \dots, n \\ & z_i^+, z_i^-, y_i \geq 0 \quad i = 1, \dots, n \end{aligned}$$

قابل توجه است که مدل اول شامل $2n + 2rn$ متغیر و $2n$ محدودیت است و مسئله دوم دارای $2n + 2$ متغیر و $n(4r + 1)$ محدودیت است.

مثال ۶-۱. نقاط $p_1 = (0, 0), p_2 = (0, 1), p_3 = (1, 0), p_4 = (1, 1)$ با وزنهای $w_1 = w_2 = w_3 = w_4 = 1$ و شعاع های $r_1 = r_2 = r_3 = r_4 = 1$ در نظر می گیریم. فرض کنیم فاصله بین نقاط با نرم بلوکی که نقاط گوشه ای کانتور واحد آن بقرار زیر است، اندازه گیری می شود.

$b_1 = -b_{-1} = (0, 1), b_2 = -b_{-2} = (\sqrt{3}/2, 1/2), b_3 = -b_{-3} = (1, 0), b_4 = -b_{-4} = (\sqrt{3}/2, -1/2)$ حال با استفاده از مدل (۶-۸) بدست می آید:

$$\begin{aligned} \text{Min} \quad & \sum_{i=1}^4 (z_i^+ + z_i^-) \\ \text{S.t.} \quad & \sum_{g=1}^4 (\lambda_{gi}^+ + \lambda_{gi}^-) - z_i^+ + z_i^- = 1 \quad i = 1, \dots, 4 \\ & X - P_i = \sum_{g=1}^4 (\lambda_{gi}^+ + \lambda_{gi}^-) b_g \quad i = 1, \dots, 4 \\ & z_i^+, z_i^- \geq 0 \quad i = 1, \dots, 4 \\ & \lambda_{gi}^+, \lambda_{gi}^- \geq 0 \quad g = 1, \dots, 4 \end{aligned}$$

از حل این مدل در نرم افزار لینگو ۱۱,۰ نتیجه $X^* = (۰/۵,۰/۷۱۱۳۳)$ با مقدار تابع هدف $F^* = ۰$ حاصل شده است.

مثال ۶-۲. مسئله وبر آرمانی را برای نمونه هایی که با اندازه مختلف تولید شده است حل می کنیم. برای حل آن از زوش پیشنهادی WLA، PSO استاندارد و نسخه اصلاح شده آن و نیز الگوریتم BSSS استفاده کرده ایم. مکان سرویس دهندگان، میزان تقاضا و شعاع ها همگی اعداد مثبتی هستند که بطور تصادفی به ترتیب در بازه های $[۱,۶۰]$ ، $[۱,۳]$ ، $[۱,۱۰]$ تولید شده اند. نتایج حاصل، در جدول های ۱-۶ و ۲-۶ خلاصه شده است.

جدول ۱-۶: نتایج الگوریتم های WLA، PSO استاندارد و نسخه اصلاح شده آن و نیز الگوریتم BSSS

n	PSOCGWP		PSOGWP	
	fitness	% Error	fitness	% Error
100	73813.4667	0.4179	73813.8262	0.4184
200	151348.9234	0.4326	151348.9276	0.4326
300	228498.2687	0.4453	228498.2952	0.4453
400	300238.9688	0.4516	300238.9830	0.4516
500	381777.8140	0.4541	381777.8176	0.4541
1000	2607804.4829	0.1922	2607804.4886	0.1922
2000	23496462.5451	0.0837	23496462.6927	0.0837
3000	82145115.6903	0.0810	82145123.8949	0.0810

n	WLA		BSSS	
	fitness	% Error	fitness	% Error
100	73506.2916	0	73813.4667	0.4179
200	150696.9426	0	151348.9234	0.4326
300	227485.2900	0	228498.2687	0.4453
400	298889.2018	0	300238.9688	0.4516
500	380052.1115	0	381777.8140	0.4541
1000	2602802.1689	0	2607804.4829	0.1922
2000	23476806.3317	0	23496462.5451	0.0837
3000	82078641.1077	0	82145115.6903	0.0810

جدول ۶-۲: : زمان اجرای الگوریتم های WLA، PSO استاندارد و نسخه اصلاح شده آن و نیز الگوریتم BSSS

n	WLA	PSOCGWP	PSOGWP	BSSS
100	0.002	0.650	0.628	4.204
200	0.004	1.213	1.145	7.767
300	0.005	1.720	1.683	11.374
400	0.007	2.257	2.190	15.065
500	0.008	2.799	2.708	18.211
1000	0.014	2.487	5.350	83.617
2000	0.027	10.708	10.566	389.984
3000	0.040	15.881	15.799	580.316

نتیجه گیری و پیشنهادات

در این پایان نامه مسائل مکانیابی با جاذبه و دافعه مورد مطالعه و بررسی قرار گرفتند و همچنین دیدگاه و روشی جدید برای مدلسازی مسائل این حوزه پیشنهاد شده است. این نگرش به مکانیابی با جاذبه و دافعه را مکانیابی آرمانی می‌نامیم. در حالتی که فاصله نقاط با نرم‌های بلوکی مورد سنجش قرار گیرد، مدل برنامه ریزی خطی برای مسئله ارائه شد. البته خطا در این شرایط با تابع قدر مطلق اندازه گرفته می‌شود. در شرایطی که با تابع مجذور اختلافات اندازه گرفته شود و فاصله نقاط با نرم اقلیدسی مورد سنجش قرار گیرد، یک الگوریتم گرادیانی شبه وایزفلد با همگرایی سراسری برای حل آن پیشنهاد شد و جواب آن با روشهای دیگر نظیر BTST, BSSS, PSO مقایسه شد.

به دلیل فازی بودن ماهیت داده‌ها و نظر تصمیم‌گیرندگان و شرایطی که در عمل پیش می‌آید، داده‌ها و متغیرهای مسئله را می‌توان فازی در نظر گرفته و آن را در حالت فازی نیز بررسی کرد و یک مدل برنامه ریزی فازی برای آن به کار بریم. در این راستا می‌توان از توابع رتبه‌بندی یا حساب ديفرانسیل فازی بهره برد. همچنین می‌توان مسئله را در حالتی که داده‌ها بازه‌ای یا تصادفی باشند مورد بررسی قرار داد. نگرش دیگری که می‌تواند مورد مطالعه قرار گیرد، استفاده از تابع لینکس برای مدلسازی مسئله است. مطالعه الگوریتم‌های مناسب با بررسی همگرایی آنها می‌تواند موضوع مطالعات بعدی باشد. موضوع دیگری که در مکانیابی کارا با جاذبه و دافعه می‌تواند موثر و راهگشا باشد، بحث تحلیل تناظری است که به نظر می‌رسد می‌توان از آن در کنار تحلیل پوششی داده‌ها برای مکانیابی کارا در محیط پیوسته استفاده کرد.

مراجع

کتابنامه

[۱] سهامی ح، (۱۳۸۶) "آمایش و مکانیابی" چاپ اول، دانشگاه صنعتی مالک اشتر، پژوهشکده مهندسی پدافند غیر عامل.

[۲] اصغر پور م.ج، (۱۳۸۷) "تصمیم گیری های چند معیاره"، چاپ ششم، موسسه انتشارات دانشگاه تهران.

[۳] مهرگان م، (۱۳۸۶) "تصمیم گیری با چندین هدف"، چاپ اول، موسسه انتشارات دانشگاه تهران.

[4] Hakimi, S. L., (1964), "Optimum locations of switching centers and absolute centers and medians of a graph", **Operations Research**, 12, 450-459.

[5] Goldman A.J., Dearing P.M., (1975) "Concepts of optimal location for partially noxious facilities", **ORSA Bull.** 23 , 1, B31.

[6] Church R.L., Garfinkel R.S., (1978) "Locating an obnoxious facility on a network," **Trans. Sci.** 12 pp 107-118.

[7] Handler G. Y. , Mirchandani P. B., (1979) "Location on networks: Theory and algorithms". **MIT Press, Cambridge.**

[8] Hamacher H.W. , Nickel S., (1998) "Classification of location models.", **Location Sci.**, 6, 229-242.

[9] Erkut E., Neuman S., (1989) "Analytical models for locating undesirable facilities," **European Journal of Operational Research.** 40 , 275-291.

[10] Drezner Z., (1995) "Facility Location: A Survey of Applications and Methods", **Springer-Verlag**, New York, Inc.

-
- [11] Ward J. E. , Wendell R. E., (1980) "A new norm for measuring distance which yields linear location problems," **Operations Research**, 28, 836-844.
 - [12] Ward J. E. , Wendell R. E., (1985) "Using block norms for location modelling," **Operations Research**, 33, 1074-1090.
 - [13] Francis R. L. , White J. A., (1974) "Facility Layout and Location," **Prentice-Hall**, Englewood Cliffs NJ.
 - [14] Rosing K. E., (1992) "An optimal method for solving the (generalized) multi Weber problem," **European Journal of Operational Research**, 58, 414-426.
 - [15] Wesolowsky G. O. (1993) "The Weber problem: History and Perspectives." **Location.Sci**, 1, 5-23.
 - [16] Toregas C., Swain R., Revelle C. and Berman L., (1971) "The location of emergency service facilities," **Operations Research**, 19, 1363-1373.
 - [17] Cooper L. (1963), "Location-allocation problems," **Operations Research**, 11, 331-343.
 - [18] Khumawala B. M., (1973) "An efficient algorithm for the P-median problem with maximum distance constraint," **Geographical Analysis** 5, 309- 321.
 - [19] Halpern J., (1976) "The location of a center-median convex combination on a undirected Tree," **J. Regional. Sci.**, 16, 237-245.
 - [20] Courant R., Robbins H., (1941) "What is mathematics?," **Oxford University Press**, Oxford,.
 - [21] Krarup J., (1998) "On a complementary problem of Courant and Robbins," **Location Sci.** 6 , 337–354.
 - [22] Moon I. D. , Chaudhry S. S., (1984) "An analysis of network location Problems with distance constraints," **Management. Sci.** , 30, 290-307.
 - [23] Plastria F., (1996) "Optimal location of undesirable facilities: a selective overview," **JORBEL** 36 , 109–127.
 - [24] Melachrinoudis E. , Cullinane T. P., (1986) "Locating an obnoxious facility within a polygonal region," **Annals of Oper. Res.**, 6, 137-145.
 - [25] Horst R., (1986), "A general class of branch and bound method in global optimization with some new approaches for concave minimization" **J. Optimization Theory and Application**, 51, 271-291.

- [26] Tuy H., Horst R., (1988), "Convergence and restart in branch and bound algorithms for global optimization and application to concave minimization and D.C. optimization" **Mathematical Programming**, 41, 161-183.
- [27] P. Hansen, D. Peeters, D. Richard, J.E Thisse, (1985) "The minisum and minimax location problems revisited," **Operations Research** 33 , 1251-1265.
- [28] Francis R. L., (1967) "Some aspects of a minimax location problem," **Operations Research**, 15, 1163-1169.
- [29] Plastria E., (1992) "GBSSS: The generalized big square small square method for planar single-facility location," **European Journal of Operational Research** 62, 163-174.
- [30] Scholz D., (2009), " The multicriteria big cube small cube method," **TOP**, DOI 10.1007/s11750-009-0105-4, accepted May 2009.
- [31] Hansen P., Peeters D. , Thisse J. F. , (1981) "On the location of an obnoxious facility", **Sistemi Urbani**, 3, 229-317.
- [32] Hansen P. , Thisse J. F., (1981) "The generalized Weber-Rawls problem," **Operations Research** Humburg July 1981, J.P. Brans , Editor , North Hooland, Amsterdam.
- [33] Drezner Z., (2004), "The big triangle small triangle method for the solution of nonconvex facility location problems," **Operations Research , INFORMS** , 52, 1, 128-135.
- [34] Okabe A., Boots B. , Sugihara K., (1992) "Spatial Tesselations. Concepts and applications of voronoi diagrams," **John Wiley & Sons** , Chichester.
- [35] Fortune S., (1987) "A sweepline algorithm for voronoi diagrams," **Algorithmica**, 2, 153-174.
- [36] Ohya T., Iri M. and Murota T., (1984) "Improvements of the incremental of method for the voronoi diagram with computational comparisons of various algorithms," **Journal of Operations Research Society of Japan** , 27, 306-336.
- [37] Preparata F. P. , Shamos M. I., (1985) "Computational Geometry – An Introduction," **Springer Verlag**.
- [38] overmars
- [39] Bader D.A. , Sreshta S., "A new parallel algorithm for planarity testing", UNM-ECE Technical Report 03-002, October 2003.

-
- [40] Eberhart R. C. , Kennedy J., (1995) "A new optimizer using particle swarm theory" in **proc. 6th int. Symp. Micro Machine and Human science**, Nagoya, Japan, 39-43.
 - [41] Fan H., Shi Y., (2001) "Study on Vmax of particle swarm optimization." **Proceedings of the Workshop on Particle Swarm Optimization**. Indianapolis, IN: Purdue School of Engineering and Technology, IUPUI . April.
 - [42] Shi Y., Eberhart R.C., (1998), "A modified particle swarm optimizer." In: **Proceedings of the IEEE international conference on evolutionary computation**. IEEE Press, Piscataway, NJ, 69–73.
 - [43] Ali M.M., Kaelo P., (2008), "Improved particle swarm algorithms for global optimization," **Applied Mathematics and Computation**, 196, 578-593.
 - [44] Shi Y., Eberhart R.C., (1998), "Parameter selection in particle swarm optimization," *Evolutionary Programming VII. Lecture Notes in Computer Science*, Vol. 1447, **Springer, Berlin**, 591–600.
 - [45] Senthil Arumugam M., Rao M.V.C., (2008), "On the improved performances of the particle swarm optimization algorithms with adaptive parameters, cross-over operators and root mean square (RMS) variants for computing optimal control of a class of hybrid systems," **Applied Soft Computing**, 8, 324–336.
 - [46] Shi Y., Eberhart R.C., (1999), "Empirical study of particle swarm optimization," in: **IEEE International Congress on Evolutionary Computation**, 3, 101–106.
 - [47] Fourie P.C., Groenwold A.A., (2002), "The particle swarm optimization algorithm in size and shape optimization," **Structural and Multidisciplinary Optimization** 23, (4), 259-267.
 - [48] Jiao B., Lian Z., Gu X., (2008), "A dynamic inertia weight particle swarm optimization algorithm," **Chaos, Solitons and Fractals**, 37, 698-705.
 - [49] Ratnaweera A., Halgamuge S.K., Watson H.C., (2004), "Self-organizing hierarchical particle swarm optimizer with time-varying acceleration coefficients," **IEEE Transactions On Evolutionary Computation**, 8, (3), 240-255.
 - [50] Clerc M., Kennedy J., (2002) "The particle swarm-explosion, stability, and convergence in a multidimensional complex space," **IEEE Transactions on Evolutionary Computation**, 6, (1), 58-73.

-
- [51] Van den Bergh F., (2002), "An analysis of particle swarm optimizers." **Ph.D. thesis**, Department of Computer Science, University of Pretoria, Pretoria, South Africa.
 - [52] Suganthan P.N., (1999), "Particle swarm optimizer with neighborhood operator," In: **proceedings of the Congress on Evolutionary Computation**, Piscataway, NJ: IEEE Service Center, 1958-1962.
 - [53] Van den Bergh F., Engelbrecht A.P., (2002), "A new locally convergent particle swarm optimizer," In: **Proceedings of the IEEE Conference on Systems, Men and Cybernetics**, 96-101.
 - [54] Van den bergh F., Engelbercht A.P., (2004), "A cooperative approach to particle swarm optimization," **IEEE Transactions on Evolutionary Computation**, 8, (3), 225-239.
 - [55] Brits R., Engelbrecht A.P., Van den Bergh F., (2007), "Locating multiple optima using particle swarm optimization," **Applied Mathematics and Computation**, 189, 1859–1883.
 - [56] Parsopoulos K.E., Plagianakos V.P., Magoulas G.D., Vrahatis M.N., (2001), "Stretching technique for obtaining global minimizers through particle swarm optimization," In: **Proceedings of the Particle Swarm Optimization Workshop**, Indianapolis, USA, 22-29.
 - [57] Sevkli Z., Sevilgen F.E., (2008), "A hybrid particle swarm optimization for function optimization," **Applications of Evolutionary Computing, Springer-Verlag, Berlin**, 585-595.
 - [58] Tripathi P.K., Bandyopadhyay S., Pal S.K., (2007), "Multi-objective particle swarm optimization with time variant inertia and acceleration coefficients," **Information Sciences**, 177, 5033-5049.
 - [59] Parsopoulos K.E., Vrahatis M.N., (2002) "Particle swarm optimization method in multi-objective problems." **Proceedings ACM Symposium on Applied Computing**, 603-607.
 - [60] Deb K., (2001), "Multi-objective optimization using evolutionary algorithms," **Wiley**, New York.
 - [61] Jin Y.X., Cheng H.Z., Yan J., Zhang L., (2007), "New discrete method for particle swarm optimization and its application in transmission network expansion planning," **Electric Power Systems Research**, 77, 227-233.

-
- [62] De Frietas A.I., (2006) "Optimization of nonlinear constrained particle swarm," **Technological and Economic Development of Economy**, Vol. 12, No. 1, pp. 30-36.
- [63] Coello C.A.C., (2002), "Theoretical and numerical constraint-handling techniques used with evolutionary algorithms: a survey of the state of the art," **Comput. Methods Appl. Mech. Engrg.**, 191, 1245-1287.
- [64] Deb K., (2002), "An efficient constraint handling method for genetic algorithms," **Comput. Methods Appl. Mech. Engrg.**, 186, 311-338.
- [65] He Q., Wang L., 2007, "A hybrid particle swarm optimization with a feasibility-based rule for constrained optimization," **Applied Mathematics and Computation**, 186, 1407-1422.
- [66] Parsopoulos K.E., Vrahatis M.N., (2002), "Particle swarm optimization method for constrained optimization problems," In: V. Kvasnicka, P. Sincak, J. Vascak, J. Pospichal (Eds.), **Proceedings of the 2nd Euro-International Symposium on Computational Intelligence**, Kosice, Slovakia, 214-220.
- [67] Yang J.M., Chen Y.P., Horng J.T., C.Y. Kao, (1997), "Applying family competition to evolution strategies for constrained optimization," *Lecture Notes in Computer Science*, Vol. 1213, **Springer-Verlag, Berlin Heidelberg New York**, 201-211.
- [68] Plastria E., (1995), "Continuous location problems, in: Z. Drezner (Ed.), *Facility Location: A Survey of Applications and Methods*," **Springer-Verlag**, Berlin, 225-262.
- [69] Plastria E., (1995), "Continuous location of attracting and undesirable facilities," *Semi-Plenary Papers Book*, **EURO XIV Conference Jerusalem**, 131-148.
- [70] Krarup J., Pisinger D., Plastria F., (2002) "Discrete location problems with push-pull objectives," **Discrete Applied Mathematics**, 123, 363-378.
- [71] Ndiaye M., (1996), "**ETcience sous contraintes et thZeorie de la localisation**", Ph.D., UniversitZe de Bourgogne, Dijon, France, unpublished.
- [72] Carrizosa E., Conde E., Romero-Morales M.D., (1997) "Location of a semi-obnoxious facility, a biobjective approach, in: R. Caballero, F. Ruiz, R.E. Steuer (Eds.), *Advances in Multiple Objective and Goal Programming*", **Springer**, Berlin.

-
- [73] Murray A.T., Church R.L., (1999) "Using proximity restrictions for locating undesirable facilities," **Stud. Locational Anal.** 12 , 81–99.
 - [74] Brimberg J., Juel H., (1998) "A minisum model with forbidden regions for locating a semi-desirable facility in the plane," **Location Sci.** 6 , 109–120.
 - [75] Hamacher H.W., Nickel S., (1995) "Restricted planar location problems and applications," **Nav. Res. Logistics** , 42 , 967–992.
 - [76] Nickel S., (1997) "Bicriteria and restricted 2-facility Weber problems," **Math. Methods Oper. Res.** 45 , 167–195.
 - [77] Eiselt H.A., Laporte G., (1995), "Objectives in location problems, in: Z. Drezner (Ed.), Facility Location. A Survey of Application and Methods," **Springer**, New York, 151–180.
 - [78] Carrizosa E., (1999) "Minimizing the variance of euclidean distances," **Stud. Locational Anal.** 12 , 101–118.
 - [79] Halpern J., Maimon O., (1983), "Accord and conflict among several objectives in locational decisions on tree networks," in: J.F. Thisse, H.G. Zoller (Eds.), **Locational Analysis of Public Facilities**, North-Holland, Amsterdam, 301–314.
 - [80] Morrill R.L., Symons J., (1977) "Efficiency and inequality aspects of optimum location," **Geogr. Anal.** 9 , 215–225.
 - [81] Hansen P., Moon I.D., , (1988) "Dispersing facilities on a network," Presentation at the TIMS ORSA Joint National Meeting, Washington D.C.
 - [82] Kortsarz G., Peleg D., (1993) "On choosing a dense subgraph," in: Proceedings of the 34th Annual IEEE Symposium on Foundation of Computer Science, 692–701.
 - [83] Pisinger D., (1999) " Exact solution of p-dispersion problems," Technical Report 99/14, DIKU, University of Copenhagen, Denmark.
 - [84] Erkut E., (1990) "The discrete p-dispersion problem," **European J. Oper. Res.** 46, 48–60.
 - [85] Ravi S.S., Rosenkrantz D.J., Tayi G.K., (1994) "Heuristics and special case algorithms for dispersion problems," **Oper. Res.** 42 , 299–310.
 - [86] Shier D. R., (1977) "A Min-Max theorem for p-center problem on a Tree," **Trans.Sci.**, 11, 243-252.

-
- [87] Kuby M. J., (1987) "Programming models for facility dispersion: the p-dispersion and maxisum dispersion models," **Geographical analysis**, 19, 315-329.
- [88] Chandrasekaran R. , Daughety A., (1981) "Location on tree networks: p-centre and n-dispersion problems," **Mathematics of Operations Research**, 6, 50-57.
- [89] Kincaid, R. K. (1992) "Good Solutions to Discrete Noxious Location problems via Metaheuristics", **Annls of Operations Research**, 40, 265-281.
- [90] Tamir A., (1991) "Obnoxious facility location on graphs," **SIAM Journal on Discrete Mathematics**, 4, 550-567.
- [91] Burkard R.E., Krarup J., (1998) "A linear algorithm for the pos/neg-weighted 1-median problem on a cactus," **Computing** 60 , 193–215.
- [92] Burkard R.E., Cela E., Dollani H., (2000) "2-medians in trees with pos/neg weights," **Discrete Appl. Math.** 105 , 51–71.
- [93] Burkard R.E., Fathali J., (2007) "A polynomial method for the pos/neg weighted 3-median problem on a tree," **Mathematical Methods of Operations Research**, 65, 229-238.
- [94] Fathali J., Kakhki H.T., (2006) "Solving the p-median problem with pos/neg weights by variable neighborhood search and some results for special cases," **Eur. J. Oper. Res.** 170 , 440–462.
- [95] Fathali J., Kakhki H.T., Burkard R.E., (2006) "An ant colony algorithm for the pos/neg weighted p-median problem," **Central European J. Oper. Res.** 14, 229-246.
- [96] Fathali J., (2006) "A genetic algorithm for the p-median problem with pos/neg weights," **Applied Mathematics and Computation**, 183, 1071-1083.
- [97] Romero-Morales D., Carrizosa E. , Conde E., (1997) "Semi-obnoxious location models: a global optimization approach." **European Journal of Operational Research**, 102, 295-301.
- [98] Duffer R., Michelot C., (1985) "Geometrical properties of the Fermat-Weber problem," **European Journal of Operational Research** 20, 332-343.
- [99] Rockafellar R.T., (1970) "Convex Analysis," **Princeton University Press**, Princeton, NJ.

- [100] Tuy H., Al Khayyal EA., Zhou E, (1995) "A d.c. optimization method for single facility location problems," **Journal of Global Optimization** 7, 209-227.
- [101] Hansen P., Jaumard B., Tuy H., (1995) "Global optimization in location, in: Z. Drezner (Ed.), Facility Location: A Survey of Applications and Methods," **Springer-Verlag**, Berlin, 43-68.
- [102] Chen RC., Hansen R, Jaumard B., Tuy H., (1992) "Weber's problem with attraction and repulsion," **Journal of Regional Science** 32, 467-486.
- [103] Horst R., Tuy H., (1993) "Global Optimization (Deterministic Approaches)," 2nd ed., **Springer-Verlag**, Berlin.
- [104] Drezner Z., Wesolowsky G.O., (1991) "The Weber problem on the plane with some negative weights," **INFOR** 29 , 87-99.
- [105] Carrizosa E., Romero-Morales D., (1995) "Attainment of optimal solution in a semiobnoxious location problem," Prepublicacion No. 31 de la Facultad de Matemáticas de la Universidad de Sevilla, Seccion de Estadística e Investigacion Operativa.
- [106] Minoux M.,(1986) "Mathematical Programming: Theory and Algorithms," **Wiley**, Chichester, UK.
- [107] Carrizosa E., Plastria F., (1999) "Location of semi-obnoxious facilities," **Stud. Locational Anal.** 12 , 1-27.
- [108] Kaiser, M.J. , Morin T.L., (1993) "Center points, equilibrium positions, and the obnoxious location problem", **Journal of Regional Science** 33 , 237—249.
- [109] Love R. F., Morris J. G. , Wesolosky G.O., (1988) "Facilities location models and methods," Elsevier Science Publishing Co.
- [110] Tellier L. N., (1972) "The Weber Problem: Solution and Interpretation" **Geographical Analysis** 4, 215 -233.
- [111] Tellier L. N. and Ceccaldi X., (1983) "Phenomenes de polarization et de repulsion dans le contexte du probleme de Weber", **Canadian Regional Science Association**, Vancouver.
- [112] Tellier, L. N. (1987) "Points d'attraction, points de repulsion," centre et peripherie in association de science regionale de langue Francaise. Espace Et Peripherie. Laboratorio Nacional De Engenharia Civil, Lisbon, Portugal.

-
- [113] Tellier, L. N., B. Polanski. (1989) "The Weber problem: Frequency and different solution types and extension to repulsive forces and dynamic processes." **J. Regional Sci.**, 29, 387-405.
- [114] Maranas, C. D., C. A. Floudas. (1994) "A global optimization method for Weber's problem with attraction and repulsion" W. W. Hager, D. W. Hearn, P. M. Pardalos, eds. *Large Scale Optimization: State of the Art*. **Kluwer**, Dordrecht, The Netherlands.
- [115] Drezner Z., (1980) "DISCON – A new method for the layout problem," **Operations Research**, 28, 1375-1384.
- [116] Drezner Z., (1988) "An Improved Weiszfeld iteration for the Weber locatoion problem ,"
- [117] Drezner Z., (1988) "On the set of optimal points to the Weber problem" **Transportation Science**.
- [118] Kuhn H. W., (1976) "On a Pair of Dual Non-Linear Problems," in *Non Linear Programming*, J. Abadie (Ed.), **John Wiley**, NY.
- [119] Fathali J., Zaferanieh M. and Nezakati A., (2009) "A BSSS algorithm for the location Problem with minimal square error", **To appear in Advances in Oper. Res.**
- [120] Zaferanieh M., Taghizadeh Kakhki H., Brimberg G.O., Wesolowsky G. O., "A BSSS algorithm for single facility location problem in two regions with different norms", **European Journal of Operational Research**, 190, 79-89.
- [121] Bazaraa, M.S., Sherali, H., Shetty, C.M., (1993) "Nonlinear Programming," second ed. **John Wiley & Sons**.
- [122] Bazaraa, M.S., Sherali, H., Jarvis, J., (2005) "Linear Programming and Network Flows," third ed. **John Wiley & Sons**.

پیوست ۱

متن برنامه ها

متن برنامه الگوریتم ترکیبی PSO با شبیه سازی تبریدی

```
%% Solvin Goal Location Problem using Particle Swarm Optimization

%% Initialization
clear all
close all
clc
format long

exe_step = 1 ; % number of Run for algorithm
swarm_size = 50 ; % Size of the swarm
swich=0 ;
max_iter = 100; % Maximum number of iteration for every
run
w_max= 1.2 ;
w_min = 0.4 ;
w_iter = floor(0.75*max_iter) ; % w decreasing step
w_dec = (w_max-w_min)/w_iter ;
w = w_max ;
c2 = 2.05 ; % PSO parameter C1
c1 = 2.05 ; % PSO parameter C2
R1 = rand(2, swarm_size) ;
R2 = rand(2, swarm_size) ;

eps = 0.001 ;
lastb_max = 10 ;
lastbest=0;
fit = 0;
landa = 0.94 ;
eta = 0.001;
l = 20;

load('r100.txt')
load('weight100.txt')
load('fac_pos100.txt')

r=r100;
weight=weight100;
fac_pos=fac_pos100;

%%
*****
**
%-----
%
% Initializing swarm and velocities and position
%
%-----
%

[fac_size,pp]= size(fac_pos) ;
lower = min(fac_pos) ;
upper = max(fac_pos) ;
a1=lower(1) ;
```

```

a2=upper(1) ;
b1=lower(2) ;
b2=upper(2) ;

vmax = [(a2-a1)*(rand) , (b2-b1)*(rand)] ;
vmax = vmax' ;
vmin = -vmax ;

tic ;
*****

%-----%
% Initializing swarm and velocities and position %
%-----%

position = [a1*ones(1,swarm_size) + (a2-a1)*(rand(1, swarm_size)) ;
b1*ones(1,swarm_size) + (b2-b1)*(rand(1, swarm_size))];

for i = 1 : swarm_size
    position_fitness(i) = fitness(weight , fac_pos , r , fac_size ,
position(:,i)) ;
end

fmin=min(position_fitness);
fmax=max(position_fitness);
t(1)=-(fmax-fmin)/(log(0.1)) ;

p_best_position = position ;
p_best_fitness = position_fitness ;

[g_best_fitness,g] = min(p_best_fitness) ;
for i = 1 : swarm_size
    g_best_position(:,i) = p_best_position(:,g) ;
end

% velocity = zeros(2,swarm_size);
for i = 1 : swarm_size
    velocity(:,i) = (vmax - vmin).*rand(2,1) + vmin ;
end
*****

%-----%
% VELOCITY UPDATE %
%-----%

velocity = w*velocity + c1*(R1.*(p_best_position - position))+
c2*(R2.*(g_best_position - position)) ;

*****

%-----%
% SWARM UPDATE %
%-----%

position = position + velocity ;
*****

%% Main Loop
for j = 1 : exe_step

```

```

iter = 0 ;           % Iterations' counter

while ( iter < max_iter ) && (iter-lastbest<5)
%   && (abs(g_best_fitness - fit)>= eps)
iter = iter + 1 ;
% updating value of w
if (iter<= w_iter)
    w=w_max-(iter-1)*w_dec ;
end

R1 = rand(2, swarm_size) ;
R2 = rand(2, swarm_size) ;

velocity = rand * velocity + c1*(R1.*(p_best_position - position)) +
c2*(R2.*(g_best_position - position)) ;
position = position + velocity ;

for i = 1 : swarm_size
    position_fitness(i) = fitness(weight , fac_pos , r , fac_size
,position(:,i)) ;
end

for i = 1 : swarm_size
    if position_fitness(i) < p_best_fitness(i)
        p_best_fitness(i) = position_fitness(i) ;
        p_best_position(:,i) = position(:,i) ;
    end
end

[global_best_fitness,g] = min(p_best_fitness) ;

if global_best_fitness < g_best_fitness
    g_best_fitness = global_best_fitness ;
    lastbest=iter;
    for i = 1 : swarm_size
        g_best_position(:,i) = p_best_position(:,g) ;
    end
end

sai=1;
while (sai<=1)
    ppg = g_best_position(:,1) ;
    xp = ppg + eta*(max(a2,b2)-min(a1,b1))*randn ;
    pa = min(1,exp((fitness(weight , fac_pos , r , fac_size ,ppg)-
fitness(weight , fac_pos , r , fac_size ,xp))/t(iter))) ;
    if pa>=rand
        ppg = xp ;
    end
    sai=sai+1;
    for i = 1 : swarm_size
        g_best_position(:,i) = ppg ;
    end
    g_best_fitness = fitness(weight , fac_pos , r , fac_size ,ppg);
end % while
t(iter+1) = landa*t(iter);
rrr(j) = g_best_fitness ;
qqq(iter) = g_best_fitness ;

```

```

    sprintf(' iteration %3.0f .....', iter )

end          % end of while loop

j=j+1 ;
end % end of for j <= exe
toc ;
t = toc ;
sprintf('best fitness %3.4f: ', min(qqq) )
% g_best_position (:,1)
% % r
t/exe_step

plot(qqq);
% hold on;
%      h=figure;
% plot(rrr);
*****

%% this M-file evaluate the fitness function

function f = fitness(e, r, weight, fac_pos, fac)
sum = 0;
for col = 1 : fac
    s = weight(col) * (norm(e - fac_pos(col,:)) - r(col))^2 ;
    sum = sum + s ;
end
f = sum ;

```

متن برنامه مثلث بزرگ مثلث کوچک

```
%% Solvin %%Goal Location Problem using BTST

%% Initialization
clear all
close all
clc
format short

exe_step = 1 ; % number of Run for algorithm
swich=0 ;
max_iter = 500; % Maximum number of iteration for every
run
eps = 0.1 ;
lastb_max = 10 ;
lastbest=0;

load('r100.txt')
load('weight100.txt')
load('fac_pos100.txt')

r=r100;
weight=weight100;
fac_pos=fac_pos100;

*****
%-----%
%      Initialization      %
%-----%

[fac_size,pp]= size(fac_pos) ;
lower = min(fac_pos) ;
upper = max(fac_pos) ;
a1=lower(1) ;
a2=upper(1) ;
b1=lower(2) ;
b2=upper(2) ;
facility = fac_pos;
[fac,pp2]= size(facility) ;
fac_pos(fac_size+1,:)= [a1,b1];
fac_pos(fac_size+2,:)= [a2,b2];
fac_pos(fac_size+3,:)= [a1,b2];
fac_pos(fac_size+4,:)= [a2,b1];
f11=fac_pos(:,1);f1=f11';
f22=fac_pos(:,2);f2=f22';
td=delaunay(f1,f2);
    triplot(td,f1,f2);
tic ;

%% Main Loop
% ** Phase 2 : Evaluating **

[t1,t2] = size(td);
for i = 1 : t1
    q = td(i,:);
```



```

        center(i,1) = (f1(q(1)) + f1(q(2)) + f1(q(3)))/3;
        center(i,2) = (f2(q(1)) + f2(q(2)) + f2(q(3)))/3;
        center_fitness(i) = fitness(center(i,:), r, weight, fac_pos,
fac_size);
end
[UB,g] = min(center_fitness) ;

%% calc lower bound for each tri    (LB)
for i = 1 : t1
    q = td(i,:);
    for j=1 : fac_size
        u(j,:) = [norm(fac_pos(j,:)-fac_pos(q(1),:)), norm(fac_pos(j,:)-
fac_pos(q(2),:)), norm(fac_pos(j,:)-fac_pos(q(3),:))] ;
        DT(j) = max(u(j,:)) ;
    end
    sum=0;
    for k=1 : fac_size
        sum = sum + weight(k) * (DT(k)-r(k))^2 ;
    end;
    LB(i) = sum ;
end

for i=1 : t1
    discard(i) = 0 ;
end

for i=1 : t1
    if LB(i)>UB/(1+eps)
        discard(i) = 1 ;
    end
end

%% ** Phase 3 : Branch and Bound **

[LB_min,h] = min(LB) ;
for i = 1 : t1
    td_new(i,:) = td(i,:);
end

iter = 0 ;           % Iterations' counter
while ( LB_min < UB/(1+eps) ) && (iter < max_iter)
    iter = iter + 1 ;
    q = td_new(h,:);
    q
    new(1,1) = (f1(q(1))+f1(q(2)))/2;
    new(1,2) = (f2(q(1))+f2(q(2)))/2;
    new(2,1) = (f1(q(1))+f1(q(3)))/2;
    new(2,2) = (f2(q(1))+f2(q(3)))/2;
    new(3,1) = (f1(q(3))+f1(q(2)))/2;
    new(3,2) = (f2(q(3))+f2(q(2)))/2;

    [fac_size,pp]= size(fac_pos) ;
    [t1,t2] = size(td_new);

    td_new(t1+1,:) = [q(1),fac_size+1,fac_size+2] ;
    td_new(t1+2,:) = [q(2),fac_size+3,fac_size+1] ;
    td_new(t1+3,:) = [q(3),fac_size+2,fac_size+3] ;
    td_new(h,:) = [fac_size+1,fac_size+3,fac_size+2] ;

```

```

f1=fac_pos(:,1); f2=fac_pos(:,2);
[t1,t2] = size(td_new);

for i = 1 : t1
    q = td_new(i,:);
    center(i,1) = (f1(q(1)) + f1(q(2)) + f1(q(3)))/3;
    center(i,2) = (f2(q(1)) + f2(q(2)) + f2(q(3)))/3;
    center_fitness(i) = fitness(center(i,:), r, weight ,
facility , fac_size) ;
end
[UB,g] = min(center_fitness)

%% calc lower bound for each tri LB
for i = 1 : t1
    if discard(i) == 0
        q = td_new(i,:);
        for j=1 : fac_size
            u= [norm(fac_pos(j,:)-
fac_pos(q(1),:)),norm(fac_pos(j,:)-fac_pos(q(2),:)),norm(fac_pos(j,:)-
fac_pos(q(3),:)))] ;
            DT(j) = max(u);
        end
        sum=0;
        for k=1 : fac_size
            sum = sum + weight(k)*(DT(k)-r(k))^2 ;
        end;
        LB(i) = sum ;
    end %% if
end

for i=1 : t1
    if LB(i)>UB/(1+eps)
        discard(i) = 1 ;
    end
end
triplot(td_new,f1,f2);
sprintf(' iteration %3.0f .....', iter )

end % while
toc ;
t = toc ;
t/exe_step
sprintf(' The Solution is = %3.7f ', UB )
sprintf(' Number of Iterations = %3.0f ', iter )
% sprintf(' CPU Time = %3.5f ', t )

%%
*****

```

متن برنامه وایزفیلد اصلاح شده برای وبر آرمانی

```

%% Solvin %%Goal Location Problem using a Weiszfeld like Algorithm
%
%

clear all
close all
clc

max_iter = 300 ;           % Maximum number of iteration for every
run
exe_step = 30 ;           % number of Run for algorithm
eps = 10^-5 ;

load('r100.txt')
load('weight100.txt')
load('fac_pos100.txt')

r=r100;
weight=weight100;
fac_pos=fac_pos100;

[fac_size,pp]= size(fac_pos) ;
x(:,1) = [-1;-1] ;
x(:,2) = x(:,1) - (1/(g(weight , fac_pos , r , fac_size , x(:,1)
    )))*(h(weight , fac_pos , r , fac_size , x(:,1) )) ;

k=2;
tic ;
for j = 1 : exe_step
    % for k = 2 : max_iter

        while (k < max_iter) && (abs(fitness(weight, fac_pos, r, fac_size,
            x(:,k))-fitness(weight, fac_pos, r, fac_size, x(:,k-1)))
            >= eps)

            k=k+1;
            sprintf(' iteration %3.0f .....', k )
            x(:,k) = x(:,k-1) - (1/(g(weight , fac_pos , r , fac_size ,
                x(:,k-1) )))*(h(weight , fac_pos , r , fac_size ,
                x(:,k-1) )) ;

        end
    % end
toc ;
t(j) = toc ;
end
sprintf('xbest %3.4f ', x(:,k) )
sprintf('best fitness %3.4f ', fitness(weight , fac_pos , r ,
    fac_size , x(:,k) ) )
sum(t)/exe_step
k

```

```

%% Solvin %%Calculating eigenvalues of a nonsymmetric matrix using
Particle Swarm Optimization
%% Initialization
clc
format long

exe_step = 1 ;           % number of Run for algorithm
swarm_size = 3 ;        % Size of the swarm
swich=0 ;
max_iter = 4;           % Maximum number of iteration for every run
n = 1024;
m = 20 ;
step = m ;
k_max=50 ;
k = 200 ;
w_max= 1.2 ;
w_min = 0.4 ;
w_iter = floor(0.75*max_iter) ; % w decreasing step
w_dec = (w_max-w_min)/w_iter ;
w= w_max ;
c2 = 2.4 ;              % PSO parameter C1
c1 = 0.4 ;              % PSO parameter C2
R1 = rand(n, swarm_size) ;
R2 = rand(n, swarm_size) ;

eps = 0.0001 ;
lastb_max = 10 ;
fun_eval=0;

lastbest=0;

%% *****
%%*****          DATA
%data2
% %
% for i=1:8
%     for j=1:8
%         t1(i,j)=0;
%     end
% end
% for i=2:8
%     t1(i,i)=-1;
% end
% t1(1,1)=-2;
% t1(8,8)=-1;
% for i=2:8
%     t1(i,i-1)=-1;
% end
% for i=1:8
%     for j=1:8
%         t2(i,j)=0;
%     end
% end
% for i=2:8
%     t2(i,i)=-1;
% end

```

```

% for i=2:8
%     t2(i,i-1)=-1;
% end
% t2(8,8)=-2;
% t2(1,1)=-1;
%
% I=eye(8) ;
%
b1=blkdiag(4*I,4*I,4*I,4*I,4*I,4*I,4*I,4*I,4*I,4*I,4*I,4*I,4*I,4*I,4*I);
%
% s=[t1 -2*I zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) zeros(8)
zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) zeros(8)
zeros(8); -I t2 -I zeros(8) zeros(8) zeros(8) zeros(8) zeros(8)
zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) zeros(8)
zeros(8); zeros(8) -I t1 -I zeros(8) zeros(8) zeros(8) zeros(8)
zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) zeros(8)
zeros(8); zeros(8) zeros(8) -I t2 -I zeros(8) zeros(8) zeros(8)
zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) zeros(8)
zeros(8); zeros(8) zeros(8) zeros(8) -I t1 -I zeros(8) zeros(8)
zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) zeros(8)
zeros(8); zeros(8) zeros(8) zeros(8) zeros(8) -I t2 -I zeros(8)
zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) zeros(8)
zeros(8); zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) zeros(8)
zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) zeros(8)
-I t1 -I zeros(8) zeros(8); zeros(8) zeros(8) zeros(8) zeros(8)
zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) zeros(8)
-I t2 -I zeros(8) zeros(8) zeros(8); zeros(8) zeros(8) zeros(8)
zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) zeros(8)
-I t1 -I zeros(8) zeros(8); zeros(8) zeros(8) zeros(8) zeros(8)
zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) zeros(8)
-I t2 -I zeros(8); zeros(8) zeros(8) zeros(8) zeros(8) zeros(8)
zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) zeros(8)
-I t1 -I; zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) zeros(8)
zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) zeros(8)
-2*I t2;];
%
% ss=[t2 -2*I zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) zeros(8)
zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) zeros(8)
zeros(8); -I t1 -I zeros(8) zeros(8) zeros(8) zeros(8) zeros(8)
zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) zeros(8)
zeros(8); zeros(8) -I t2 -I zeros(8) zeros(8) zeros(8) zeros(8)
zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) zeros(8)
zeros(8); zeros(8) zeros(8) -I t1 -I zeros(8) zeros(8) zeros(8)
zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) zeros(8)
zeros(8); zeros(8) zeros(8) zeros(8) -I t2 -I zeros(8) zeros(8)
zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) zeros(8)
zeros(8); zeros(8) zeros(8) zeros(8) zeros(8) -I t1 -I zeros(8)
zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) zeros(8)
zeros(8); zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) zeros(8)
zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) zeros(8)
-I t2 -I zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) zeros(8)
zeros(8); zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) zeros(8)
-I t1 -I zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) zeros(8)
zeros(8);

```

```

zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) -I t2 -
I zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) zeros(8); zeros(8)
zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) -I t1 -
I zeros(8) zeros(8) zeros(8) zeros(8) zeros(8); zeros(8) zeros(8)
zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) -I t2 -
I zeros(8) zeros(8) zeros(8) zeros(8); zeros(8) zeros(8) zeros(8)
zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) -I t1 -
I zeros(8) zeros(8) zeros(8); zeros(8) zeros(8) zeros(8) zeros(8)
zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) -I t2 -
I zeros(8) zeros(8); zeros(8) zeros(8) zeros(8) zeros(8) zeros(8)
zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) -I t1 -
I zeros(8); zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) zeros(8)
zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) -I t2 -
I; zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) zeros(8)
zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) zeros(8) -2*I
t1;];
%
% a=[b1 s; ss b1];
% a=newnew ;
*****
%-----%
% Initializing swarm and velocities and position %
%-----%

tic ;

position = ones(n, swarm_size) ;
for i = 1 : swarm_size
position(:,i) = position(:,i)/norm(position(:,i));
end

for i = 1 : swarm_size
[ff,landa(:,i),z(:, :,i)] = ra(k, position(:,i), m ,n ,a);
position(:,i) = ff ;
position_fitness(i) = fitness(m,n,landa(:,i),z(:, :,i),a ) ;
end

p_best_position = position ;
p_landa = landa ;
p_eigv = z ;
p_best_fitness = position_fitness ;

[g_best_fitness,g] = min(p_best_fitness) ;
last_best_fitness = g_best_fitness ;

for i = 1 : swarm_size
g_best_position(:,i) = p_best_position(:,g) ;
end
g_landa = p_landa(:,g);
g_eigv = p_eigv(:, :,g);

velocity = zeros(n,swarm_size);

*****
%% Main Loop
for j = 1 : exe_step

iter = 0 ;          % Iteration's counter

```

```

while ( iter < max_iter ) & (swich==0)
% (g_best_fitness > eps)
iter = iter + 1 ;
% updating value of w
if (iter<= w_iter)
    w=w_max-(iter-1)*w_dec ;
end

R1 = rand(n, swarm_size) ;
R2 = rand(n, swarm_size) ;

velocity = w * velocity + c1*(R1.*(p_best_position - position)) +
c2*(R2.*(g_best_position - position)) ;
position = position + velocity ;

for i = 1 : swarm_size
    [ff,landa(:,i),z(:, :,i)] = ra(k, position(:,i), m ,n, a ) ;
    position(:,i) = ff ;
    position_fitness(i) = fitness(m,n,landa(:,i),z(:, :,i),a ) ;
    fun_eval = fun_eval+1 ;
end

for i = 1 : swarm_size
    if position_fitness(i) < p_best_fitness(i)
        p_best_fitness(i) = position_fitness(i) ;
        p_best_position(:,i) = position(:,i) ;
        p_landa(:,i) = landa(:,i) ;
        p_eigv(:, :,i) = z(:, :,i) ;
    end
end

[global_best_fitness,g] = min(p_best_fitness) ;

if global_best_fitness < g_best_fitness
    lastbest=iter;
    g_best_fitness = global_best_fitness ;
    for i = 1 : swarm_size
        g_best_position(:,i) = p_best_position(:,g) ;
    end
    g_landa = p_landa(:,g);
    g_eigv = p_eigv(:, :,g);

    for i=1:m
        g_eigv(:,i)=g_eigv(:,i)/norm(g_eigv(:,i));
    end
end

rrr(iter) = g_best_fitness ;
sprintf(' iteration %3.0f .....', iter )

if abs(g_best_fitness)<=eps
    swich=1;
end

end % end of while loop

j=j+1 ;
end % end of for j <= exe

```

```

for i=1:m
    error(i) = norm(a*g_eigv(:,i)-g_landa(i)*g_eigv(:,i));
end
% rrr

toc ;
% t/exe_step
g_landa
g_best_fitness
error'
figure ;
plot(rrr) ;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

```

function [ff,ft,ev] = ra(k, x, m ,n , a )

%% %%%%%%%%% Restarnted Arnoldi MGS() %%%%%%%%%

v(:,1) = ones(n,1) ;
v(:,1) = v(:,1)*(1/norm(v))
vv(:,1)=x;
for s = 1 : k      %% ,..... other conditions
    vvv = vv(:,s) ;
    [h,v] = arnoldi(vvv,m,n,a) ;
    [g , l] = eig(h);
    %     for j = 1 : m
    %         lambda(j) = l(j,j) ;
    %     end
    lambda = diag(l);
    y = v*g ;
    vvv = zeros(n,1);
    %%%%% detemining c(i) %%%%
    c = ones(1,m) ;
    %     c(1)=0;
    %     for i = 1 : m
    %         c(i) = norm(a*y(:,i)-lambda(i)*y(:,i)) ;
    %     end
    %%%%%%%%%
    for i = 1 : m
        vvv = vvv + c(i)*y(:,i) ;
    end
    vv(:,s+1) = vvv ;
end
ff = vv(:,k+1);
ft = lambda ;
ev = y ;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

```

function [f,rr] = arnoldi(q, m, n, am )

%% %%%%%%%%% Arnoldi MGS() %%%%%%%%%
% tic;
% v(:,1) = ones(n,1) ;
v(:,1) = q ;
v(:,1) = v(:,1)*(1/norm(q)) ;

```

```

for j = 1 : m
    w = am*v(:,j) ;
    for i = 1 : j
        h(i,j) = w'*v(:,i) ;
        w = w - h(i,j)*v(:,i) ;
    end
    if j ~= m
        h(j+1,j) = norm(w) ;
        v(:,j+1) = (1/h(j+1,j))*w ;
    end
end
for j = 1 : m
    for i = j+2 : m
        h(i,j)=0;
    end
end
f = h ;
rr = v;
% toc;
% h
% v
% t = toc
% am*v-v*h;
*****

%% this M-file evaluate the fitness function

function f = fitness(m,n,lam,zz,a )
sum = 0;
for j = 1 : m
    sum = sum + norm(a*zz(:,j) - lam(j)*zz(:,j)) ;
end
f = sum ;

```

واژه‌نامه

واژه نامه انگلیسی به فارسی

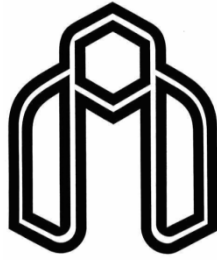
Attraction	جاذبه
Barrier	نواحی ممنوع
Bound	کران
Lower	پایین
Upper	بالا
Branch and Bound	شاخه و کران
Center of gravity	مرکز ثقل
Combinatorial optimization	بهینه سازی ترکیبیاتی
Computational geometry	هندسه محاسباتی
Concave	مقعر
Connected	همبند
Convergence	همگرایی
Convex	محدب
Convex hull	پوسته محدب
Covering	پوشش
Customer	مشتری
Delauney graph	گراف دلانی
Demand point	نقطه تقاضا
Descent direction	جهت کاهشی
Distance	فاصله
Evolutionary	تکاملی
Facility	سرویس دهنده
Global optimum	بهینه سراسری
Goal single facility problem	مسئله تک وسیله ای آرمانی
Hazard	خطرناک
Heuristic	ابتکاری
Iterative	تکراری
Lagrangean relaxation	روش ضرایب لاگرانژ
Layout	جایابی
Local optimum	بهینه موضعی
Location	مکانیابی

Minisum	کمترین مجموع
Multi facility location problem	مسئله مکانیابی چند وسیله ای
Norm	نرم
Block	بلوکی
Minkowsky	مینکوفسکی
Noxious	مضر
Obnoxious	نامطلوب
Pareto set	مجموعه پارتو
Particle	ذره
Polygone	چندضلعی
Position	موقعیت
Power station plant	نیروگاه
Pull	دافعه
Push	جاذبه
Rectangle hull	پوسته مستطیلی
Repulsion	دافعه
Saddle point	نقطه زینی
Semi-dispersion	شبه پراکنده
Semi-obnoxious, semi-undesirable	نیمه نامطلوب
Space	فضا
Continuous	پیوسته
Discrete	گسسته
Network	شبکه
Swarm	دسته ، گروه
Triangulation	مثلث بندی
Undesirable	نامطلوب
Velocity	سرعت
Voronoi diagram	گراف ورونوی
Warehouse	انبار
Weight	وزن
Zoning	منطقه بندی

واژه نامه فارسی به انگلیسی

Heuristic	ابتکاری
Warehouse	انبار
Combinatorial optimization	بهینه سازی ترکیبیاتی
Global optimum	بهینه سراسری
Local optimum	بهینه موضعی
Convex hull	پوسته محدب
Rectangle hull	پوسته مستطیلی
Covering	پوشش
Evolutionary	تکاملی
Iterative	تکراری
Push, Attraction	جاذبه
Layout	جایابی
Descent direction	جهت کاهش
Polygone	چندضلعی
Hazard	خطرناک
Pull, Repulsion	دافعه
Swarm	دسته ، گروه
Particle	ذره
Lagrangean relaxation	روش ضرایب لاگرانژ
Velocity	سرعت
Facility	سرویس دهنده
Branch and Bound	شاخه و کران
Semi-dispersion	شبه پراکنده
Distance	فاصله
Space	فضا
Continuous	پیوسته
Network	شبکه
Discrete	گسسته
Bound	کران
Upper	بالا
Lower	پایین

Minisum	کمترین مجموع
Delauney graph	گراف دلانی
Voronoi diagram	گراف ورونوی
Triangulation	مثلث بندی
Pareto set	مجموعه پارتو
Convex	محدب
Center of gravity	مرکز ثقل
Goal single facility problem	مسئله تک وسیله ای آرمانی
Multi facility location problem	مسئله مکانیابی چند وسیله ای
Customer	مشتری
Noxious	مضر
Concave	مقعر
Location	مکانیابی
Zoning	منطقه بندی
Position	موقعیت
Obnoxious, Undesirable	نامطلوب
Norm	نرم
Block	بلوکی
Minkowsky	مینکوفسکی
Demand point	نقطه تقاضا
Saddle point	نقطه زینی
Barrier	نواحی ممنوع
Power station plant	نیروگاه
Semi-obnoxious, semi-undesirable	نیمه نامطلوب
Weight	وزن
Connected	همبند
Convergence	همگرایی
Computational geometry	هندسه محاسباتی



Shahrood University of Technology
Faculty of Mathematics

Location Problems with Push-Pull objectives

Student:

Ali Jamalian

Supervisor:

Dr. Jafar Fathali

Advisos:

Dr. Ahmad Nezakati

A Thesis Submitted for Master of Science Degree in Applied Mathematics

Febuary 2010

Abstract