به نام آن که جان را فکرت آموخت

مقدمه‌ای بر الگوریتم‌های فراابتکاری

تألیف:

دکتر علیرضا ناصر صدرآبادی (استادیار دانشگاه یزد)

نعیمه تقوی (دانشجوی دکتری دانشگاه تهران)

انتشارات پندار پارس

سرشناسه : ناصرصدرآبادی، علیرضا، ۱۳۴۹ -
عنوان و نام پدیدآور : مقدمه‌ای بر الگوریتم‌های فراابتکاری/ تالیف علیرضا ناصرصدرآبادی، نعیمه تقوی.
مشخصات نشر : تهران: پندار پارس، ۱۳۹۳.
مشخصات ظاهری : [۱۶۸] ص.: مصور، جدول، نمودار.
شابک : ۹-۶۳-۶۵۲۹-۶۰۰-۹۷۸ ۱۰۰۰۰۰ ریال
وضعیت فهرست نویسی : فیپا
موضوع : الگوریتم‌ها
موضوع : الگوریتم‌های ژنتیک
شناسه افزوده : تقوی، نعیمه، ۱۳۶۰ -
رده بندی کنگره : ۱۳۹۳۵۸/۹QA ۷م۲ن/
رده بندی دیویی : ۳۱/۰۰۶
شماره کتابشناسی ملی : ۳۶۰۶۰۰۰

انتشارات پندارپارس

دفتر فروش: انقلاب، ابتدای کارگرجنوبی، کوی رشتچی، شماره ١٤، واحد ١٦ **www.pendarepars.com**

تلفن: ٦٦٥٧٢٣٣٥ – تلفکس: ٦٦٩٢٦٥٧٨همراه: ٠٩١٢٢٤٥٢٣٤٨ **info@pendarepars.com**

نام کتاب : مقدمه‌ای بر الگوریتم‌های فراابتکاری

ناشر : انتشارات پندار پارس

تالیف : علیرضا ناصر صدرآبادی، نعیمه تقوی

چاپ نخست : مهر ماه ۹۳

شمارگان : ۵۰۰ نسخه

طرح جلد : رامین شکرالهی

لیتوگرافی، چاپ، صحافی : ترام‌سنج، فرشیوه، خیام

قیمت : ۱۰۰۰۰ تومان **شابک : ۹-۶۳-۶۵۲۹-۶۰۰-۹۷۸**

*هرگونه کپی برداری، تکثیر و چاپ کاغذی یا الکترونیکی از این کتاب بدون اجازه ناشر تخلف بوده و پیگرد قانونی دارد *

پیشگفتار

در فرایندهای تصمیم‌گیری مدیران همواره بهینه‌سازی نقش مهمی را ایفا کرده است. به همین دلیل شاخه‌های متنوعی از علم توسعه یافته‌اند که به نحوی در حل مسائل بهینه‌سازی قابل استفاده هستند. مروری بر تاریخچه معرفی روش‌های بهینه‌سازی حاکی از آن است که این حوزه از علم همواره یک مقوله جذاب برای محققان بوده است. با ظهور هر پدیده‌ای در دنیای کامپیوتر و محاسبات، جهشی تازه نیز در حوزه بهینه‌سازی صورت گرفته است. طی دهه‌های اخیر کامپیوتر باعث شده است که پیوندی ناگسستنی بین هوش مصنوعی و بهینه‌سازی برقرار گردد. امکان انجام محاسبات در زمان بسیار کوتاه و افزایش قدرت محاسباتی کامپیوترها گروه جدیدی از تکنیک‌های بهینه‌سازی موسوم به روش‌های فراابتکاری را به دنیای علم معرفی کرده است. این کتاب به معرفی تعدادی از رایج‌ترین الگوریتم‌های این حوزه یعنی جست‌وجوی ممنوع، شبیه‌سازی تبرید تدریجی، الگوریتم ژنتیک و جمعیت مورچگان می-پردازد.

در این کتاب به جای بهره جستن صرف از منابع مختلف و ایجاد گسستگی در فصول کتاب، نحوه نگارش به شکلی است که اولا چیدمان فصل‌ها به گونه‌ای باشد که هر الگوریتم مقدمه‌ای بر معرفی الگوریتم‌های بعدی باشد و ثانیا از الگویی یکسان برای معرفی الگوریتم‌ها استفاده شود. برای این منظور بخش‌های مشترک مثل خط سیر الگوریتم یا نمودار شماتیک الگوریتم در همه‌ی فصل‌ها وجود دارد و افزون بر آن از یک مثال مشترک که مربوط به حوزه زمان‌بندی تولید است استفاده شده است تا خواننده کتاب از روندی منظم در مطالعه فصل‌های کتاب سود ببرد.

فصل نخست این کتاب به مسائل بهینه‌سازی و بهینه‌سازی ترکیبیاتی و مثال‌هایی از آن پرداخته است. در فصل دوم به الگوریتم نزول به عنوان نقطه‌ی آغاز برای معرفی الگوریتم‌های فراابتکاری اشاره شده است. فصل سوم شامل معرفی الگوریتم شبیه‌سازی تبرید تدریجی و تعریف مفاهیم کاربردی است. فصل چهارم مفاهیم الگوریتم جست‌وجوی ممنوع و قوانین و سازوکارهای آن را در بر می‌گیرد. فصل پنجم شامل مقدمه‌ای بر الگوریتم ژنتیک و مباحث تکمیلی آن می‌باشد و فصل ششم الگوریتم مورچگان را به عنوان الگوریتمی الهام گرفته از زندگی مورچه‌ها برای طراحی الگوریتم‌های جدید تشریح کرده است.

این کتاب حاصل تجربه چندین ساله تدریس الگوریتم‌های فراابتکاری در مقطع کارشناسی ارشد است و هدف آن بیان پیچیدگی‌های این گروه از الگوریتم‌ها با زبانی ساده می‌باشد. با این امید که مورد استفاده علاقه‌مندان و پژوهشگران این حوزه قرار گیرد. بدیهی است که این کتاب عاری از نقص نیست، لذا مایه امتنان خواهد بود اگر خوانندگان محترم کتاب، اندیشمندان و اساتید، ما را از نقطه نظرات خود بهره‌مند نمایند.

در پایان لازم است از جناب آقای دکتر حبیب ا... زارع که نقش مهمی در شکل‌گیری این اثر داشتند تقدیر نموده و همچنین از جناب آقای دکتر علی مروتی شریف‌آباد که زحمت مطالعه نسخه اولیه کتاب را کشیدند، سپاسگزاری نماییم. همچنین مراتب تشکر خود را نسبت به تمام دانشجویانی که در کلاس‌های درس با طرح سؤالات و نقطه نظرات خود باعث پر بار شدن مطالب کتاب شده‌اند نیز ابراز می‌داریم.

علیرضا ناصر صدرآبادی

نعیمه تقوی

تابستان ۱۳۹۳

فهرست

فصل نخست: مقدمه و تعاریف ۱
۱-۱- مقدمه ۱
۱-۲- مسائل بهینه‌سازی ۱
۱-۳- مسائل بهینه‌سازی ترکیبیاتی ۲
۱-٤- مثال‌هایی از مسائل بهینه‌سازی ترکیبیاتی ٤
۱-٤-۱- مجموع زیرمجموعه ٤
۱-٤-۲- کوتاه‌ترین درخت پوشای رتبه‌ی k ٤
۱-٤-۳- رنگ‌آمیزی گراف ٥
۱-٤-٤- مسئله‌ی فروشنده‌ی دوره‌گرد ٥
۱-٤-٥- مسئله‌ی تعیین توالی سفارش‌ها ٦
۱-٥- روش‌های حل مسائل بهینه‌سازی ترکیبیاتی ۹
۱-٥-۱- شمارش کامل ۹
۱-٥-۲- روش‌های تحقیق در عملیات ۱۰
۱-٥-۳- الگوریتم‌های ابتکاری ۱۰
۱-٥-٤- الگوریتم‌های فراابتکاری ۱٦
۱-٦- شبیه‌سازی ۱۸
۱- ۷- بهینه‌سازیِ شبیه‌سازی ۲۱
فصل دوم: الگوریتم نزول **۲۳**
۲-۱- مقدمه ۲۳
۲-۲- الگوریتم نزول ۲۳
۲-۳ - حل یک مثال عددی ۲٥
۲-۳-۱- تولید اعداد تصادفی ۲٦
الف- اعداد تصادفی یکنواخت بین صفر تا یک ۲٦
ب- تولید اعداد تصادفی در فاصله a تا b ۲۷
ج- تولید اعداد تصادفی صحیح در فاصله ۱ تا n ۲۷
۲-۳-۲- انتخاب جواب اولیه ۲۷
۲-۳-۳- انتخاب یک جواب در همسایگی ۲۹
۲-۳-٤- پذیرش یا رد جواب جدید ۳۰
۲-٤- سیاست تنوع و تمرکز ۳۲
۲-٤-۱- سیاست تمرکز ۳۲
۲-٤-۲- سیاست تنوع ۳۲
فصل سوم: شبیه‌سازی تبرید تدریجی **۳۳**
۳-۱- مقدمه ۳۳
۳-۲- تاریخچه و زمینه‌ی پیدایش ۳۳
۳-۳- خط سیر الگوریتم تبرید تدریجی ۳٥
۳-٤- مفاهیم همسایگی و حرکت ۳٥

۳-۵- معیار حرکت به سمت نقطه مجاور ۳۶
۳-۶- دما یک پارامتر اساسی ۳۷
۳-۷- تنظیم پارامترها در الگوریتم‌های فراابتکاری ۳۸
۳-۸- شرط توقف الگوریتم ۳۸
۳-۹- شبه کد الگوریتم تبرید تدریجی ۳۸
۳-۱۰- مثالی از مسئله‌ی فروشنده‌ی دوره گرد ۴۰
۳-۱۱- ساز و کار تعریف جواب جدید ۴۲
۳-۱۲- حل مثال عددی ۴۳
۳-۱۳- نکات تکمیلی ۴۶
فصل چهارم: جست‌وجوی ممنوع ۴۹
۴-۱- مقدمه ۴۹
۴-۲- تاریخچه و زمینه‌ی پیدایش ۴۹
۴-۳- خط سیر الگوریتم جست‌وجوی ممنوع ۵۰
۴-۴- مفاهیم و سازوکارهای الگوریتم جست‌وجوی ممنوع ۵۰
۴-۴-۱- فهرست کاندید و طول فهرست کاندید ۵۱
۴-۴-۲- حافظه وجه تمایز جست‌وجوی ممنوع ۵۲
۴-۴-۳- حافظه‌ی کوتاه‌مدت و بلندمدت ۵۳
۴-۴-۴- حافظه‌ی ضمنی و حافظه‌ی صریح ۵۳
۴-۴-۵- انتخاب یک جواب از فهرست کاندید ۵۵
۴-۴-۶- مفهوم ممنوعیت ۵۵
۴-۴-۷- دوره‌ی ممنوعیت ۵۶
۴-۴-۸- فهرست ممنوعه ۵۶
۴-۵- حل مسئله‌ی تعیین توالی سفارش‌ها ۵۶
۴-۵-۱- تعریف ممنوعیت ۵۷
۴-۵-۲- دوره‌ی ممنوعیت ۵۸
۴-۶- رویکردهای مختلف تعیین دوره‌ی ممنوعیت ۶۰
۴-۶-۱- دوره‌ی ممنوعیت پویای تصادفی ۶۰
۴-۶-۲- دوره‌ی ممنوعیت پویای سیستماتیک ۶۱
۴-۷- معیار آستانه ۶۱
۴-۸- رابطه‌ی بین فهرست کاندید، معیار آستانه و ممنوعیت ۶۲
۴-۹- نکاتی در مورد کدنویسی ۶۴
۴-۱۰- حافظه‌ی بلندمدت ۶۴
۴-۱۰-۱- رویکرد مبتنی بر فراوانی ۶۵
۴-۱۰-۲- استراتژی‌های تمرکز ۶۸
۴-۱۰-۳- استراتژی‌های تنوع ۷۰
۴-۱۰-۴- شروع دوباره ۷۱

فصل پنجم: الگوریتم ژنتیک ۷۵
۵-۱- مقدمه ۷۵
۵-۲- تاریخچه و زمینه‌ی پیدایش ۷۵
۵-۳ خط سیر الگوریتم ژنتیک ۷۶
۵-٤- مفاهیم و سازوکارهای الگوریتم ژنتیک ۷۷
۵-٤-۱- فرد ۷۷
۵-٤-۲- کدگذاری ۷۷
۵-٤-۳- تابع برازندگی ۷۸
۵-٤-٤- جمعیت ۷۸
۵-٤-۵- فرایند جست‌وجو ۷۹
۵-٤-٦- عملگرها ۷۹
۵-٤-۷- شرط توقف ۸۲
۵-۵- الگوریتم ژنتیک در حل مسئله‌ی تعیین توالی سفارش‌ها ۸۲
۵-۵-۱- فرد و کدگذاری ۸۳
۵-۵-۲- جمعیت ۸۳
۵-۵-۳- تابع برازندگی ۸۳
۵-۵-٤- عملگر انتخاب در حل مسئله‌ی تعیین توالی سفارش‌ها ۸٤
۵-۵-۵- عملگر تقاطع در حل مسئله‌ی تعیین توالی سفارش‌ها ۸۹
۵-۵-٦- عملگر جهش در حل مسئله‌ی تعیین توالی سفارش‌ها ۹۲
۵-٦- مباحث تکمیلی ۹۳
۵-۷- انواع دیگر کدگذاری ۹٤
۵-۷-۱- کدگذاری صفر و یک ۹٤
۵-۷-۲- کدگذاری صفر و یک خاکستری ۹٦
۵-۷-۳- کدگذاری مبنای هشت ۹۹
۵-۷-٤- کدگذاری مبنای شانزده ۱۰۰
۵-۷-۵- کدگذاری اعداد حقیقی ۱۰۰
۵-۷-٦- کدگذاری کیفی ۱۰۱
۵-۸- عملگرهای تقاطع کدگذاری مبنای دو ۱۰۱
۵-۸-۱- تقاطع تک نقطه‌ای ۱۰۱
۵-۸-۲- تقاطع دو نقطه‌ای ۱۰۲
۵-۸-۳- تقاطع چندنقطه‌ای ۱۰۳
۵-۸-٤- تقاطع یکنواخت ۱۰٤
۵-۸-۵- تقاطع سه والده ۱۰٤
۵-۸-٦- تقاطع با جانشینی محدود ۱۰۵
۵-۸-۷- تقاطع برزدن ۱۰٦
۵-۹- عملگرهای جهش کدگذاری مبنای دو ۱۰۷
۵-۹-۱- جهش چرخشی ۱۰۸

۵-۹-۲- جهش جابه‌جایی ۱۰۸
۵-۹-۳- جهش معکوس ۱۰۹
۵-۱۰- عملگر تقاطع کدگذاری حقیقی ۱۰۹
۵-۱۰-۱- تقاطع ساده اعداد حقیقی ۱۰۹
۵-۱۰-۲- تقاطع گسسته‌ی اعداد حقیقی ۱۱۰
۵-۱۰-۳- تقاطع میانی اعداد حقیقی ۱۱۱
۵-۱۰-٤- تقاطع خطی اعداد حقیقی ۱۱۲
۵-۱۱- عملگر جهش کدگذاری حقیقی ۱۱۳
۵-۱۱-۱- جهش تصادفی ۱۱٤
۵-۱۱-۲- توزیع نرمال ۱۱۵
۵-۱۲- الگوریتم ژنتیک با محدودیت ۱۱٦
۵-۱۲-۱- روش تابع جریمه ۱۱۷
۵-۱۲-۲- روش مبتنی بر حفظ موجه بودن جواب‌ها ۱۱۹
فصل ششم: الگوریتم مورچگان ۱۲۱
٦-۱- مقدمه ۱۲۱
٦-۲- تاریخچه و زمینه‌ی پیدایش ۱۲۱
٦-۳- خط سیر الگوریتم مورچگان ۱۲۵
٦-٤- مفاهیم و سازوکارهای الگوریتم مورچگان ۱۲٦
٦-٤-۱- گراف متناظر با مسئله ۱۲٦
٦-٤-۲- جمعیت مورچه‌های مصنوعی ۱۲٦
٦-٤-۳- فرمون ۱۲۷
٦-٤-٤- تبخیر فرمون ۱۲۷
٦-٤-۵- فرمون‌ریزی ۱۲۸
٦-٤-٦- مقدار ابتکاری ۱۲۸
٦-٤-۷- قانون احتمال انتخاب مسیر ۱۲۹
٦-۵- گونه‌های مختلف الگوریتم مورچگان ۱۲۹
٦-۵-۱- سیستم مورچه (AS) ۱۳۰
٦-۵-۲- سیستم مورچه‌ی نخبه‌گرا (EAS) ۱۳۷
٦-۵-۳- سیستم مورچه‌ی مبتنی بر رتبه‌بندی ($ASrank$) ۱۳۷
٦-۵-٤- سیستم مورچه حداقل- حداکثر (MMAS) ۱۳۸
٦-۵-۵- سیستم جمعیت مورچه‌ها (ACS) ۱۳۹
٦-٦- توصیه‌های سایر محققان ۱٤۷
٦-۷- مقدار ابتکاری ۱٤۸
٦-۸- مقدار ابتکاری و مسئله‌ی تعیین توالی سفارش‌ها ۱٤۹

فهرست شکل‌ها

شکل ۱- ۱- مجموعه جواب‌های مسئله‌ی تعیین توالی سفارش‌ها با ٤ سفارش ۳
شکل ۱-۲- نمایش فرضی مجموعه نقاط منطقه‌ی موجه یک مسئله‌ی بهینه‌سازی ترکیبیاتی ۳
شکل ۱-۳- نمونه‌ای از یک شبکه ۵
شکل ۱-٤- مثالی از مسیرهای موجود در مسئله‌ی فروشنده‌ی دوره‌گرد ٦
شکل ۱-۵- پیکربندی ماشین‌آلات و انبارهای میان‌گیر در مسئله‌ی جریان کارگاهی ۷
شکل ۱-٦- تغییرات تابع هدف در منطقه‌ی موجه‌ی یک مسئله‌ی تعیین توالی سفارش‌ها ۹
شکل۱-۷ - جایگاه فراابتکاری‌ها در مجموعه روش‌های حل مسائل بهینه‌سازی تک‌هدفه ۱۹
شکل ۱-۸- بخشی از ورودی‌ها و خروجی‌های یک مدل شبیه‌سازی برنامه‌ریزی و زمانبندی تولید ۲۰
شکل ۱-۹- ساختار کلی بهینه‌سازی شبیه‌سازی ۲۲
شکل ۲-۱- نمایش شماتیک مجموعه‌ی نقاط همسایگی ۲٤
شکل۲-۲- مجموعه‌ی نقاط همسایگی توالی (٤،۳،۲،۱) ۲۵
شکل ۲-۳- خط سیر الگوریتم نزول ۳۱
شکل ۳-۱- فرایند تبرید و شکل گیری کریستال‌ها ۳٤
شکل ۳-۲- خط سیر الگوریتم تبرید تدریجی ۳۵
شکل ۳-۳- جواب به دست آمده برای مسئله‌ی فروشنده‌ی دوره‌گرد در دمای ۷/٦۸ ..... ٤۰
شکل ۳-٤- جواب به دست آمده برای مسئله‌ی فروشنده‌ی دوره‌گرد در دمای ۵/۳۵ ..... ٤۱
شکل ۳-۵- جواب به دست آمده برای مسئله‌ی فروشنده‌ی دوره‌گرد در دمای ۰/۱۹ ..... ٤۱
شکل ۳-٦- حرکت جابه‌جایی برای تعریف همسایگی جواب جاری ٤۲
شکل ۳-۷- انتقال ٤۳
شکل۳-۸- خط سیر پذیرش یا رد جواب ٤٦
شکل ٤-۱- خط سیر الگوریتم جست‌وجوی ممنوع ۵۰
شکل ٤-۲- مثالی از یک جواب مسئله‌ی تعیین توالی سفارش ۵۱
شکل ٤-۳- فهرست جواب‌های کاندید ۵۱
شکل ٤-٤- حافظه انطباقی ۵۳
شکل ٤-۵- اعضای فهرست کاندید با استفاده از حافظه‌ی ضمنی ۵٤
شکل٤-٦- نمایش موقعیت سفارش‌ها ۵۷
شکل ٤-۷- تهیه‌ی جواب جدید از جواب جاری ۵۷
شکل ٤-۸- رابطه‌ی بین فهرست کاندید، معیار آستانه و ممنوعیت ٦۳
شکل ۵-۱- خط سیر الگوریتم ژنتیک ۷۷
شکل ۵-۲- جریان کلی الگوریتم ژنتیک ۸۰
شکل ۵-۳- نمایشی از یک جواب ۹۵
شکل ۵-٤- مثال دو فرد در روش کدگذاری صفر و یک ۹٦

شکل ۵-۵- روش کدگذاری خاکستری ۹۸
شکل ۵-٦- کدگذاری مبنای هشت ۹۹
شکل ۵-۷- کدگذاری مبنای شانزده ۱۰۰
شکل ۵-۸- کدگذاری اعداد حقیقی ۱۰۰
شکل ۵-۹- کدگذاری کیفی ۱۰۱
شکل ۵-۱۰- عملگر تقاطع تک نقطه‌ای ۱۰۲
شکل ۵-۱۱- تقاطع دو نقطه‌ای ۱۰۲
شکل ۵-۱۲- تقاطع چند نقطه‌ای ۱۰۳
شکل ۵-۱۳- تقاطع یکنواخت ۱۰٤
شکل ۵-۱٤- تقاطع سه‌والده ۱۰۵
شکل۵-۱۵- تقاطع با جانشینی محدود ۱۰۵
شکل ۵-۱٦- تقاطع برزدن ۱۰۷
شکل ۵-۱۷- عملگر جهش چرخشی ۱۰۸
شکل ۵-۱۸- عملگر جهش جا به جایی ۱۰۸
شکل ۵-۱۹- عملگر جهش معکوس ۱۰۹
شکل ۵-۲۰- تقاطع ساده اعداد حقیقی ۱۱۰
شکل ۵-۲۱- دامنه تغییرات جواب‌های فرزند در تقاطع میانی $d > 0$ ۱۱۲
شکل ۵-۲۲- فضای تولید کروموزوم‌های نوزاد در تقاطع میانی ۱۱۲
شکل ۵-۲۳- فضای تولید کروموزوم‌های نوزاد در تقاطع خطی ۱۱۳
شکل ۵-۲٤- عملگر جهش تصادفی ۱۱۵
شکل۵-۲۵- یک قوطی نمونه و هزینه‌ی قوطی ۱۱۸
شکل ٦-۱ آزمایش پل دو طرفه ۱۲۲
شکل ٦-۲- وضعیت فرمون مسیرها پیش از حرکت مورچه‌ها ۱۲۳
شکل ٦-۳- وضعیت فرمون مسیرها پس از ۱ واحد زمان ۱۲۳
شکل ٦-٤- وضعیت فرمون مسیرها پس از ۲ واحد زمان ۱۲۳
شکل ٦-۵- وضعیت فرمون مسیرها پس از ۳ واحد زمان ۱۲٤
شکل ٦-٦- وضعیت فرمون مسیرها پس از ٤ واحد زمان ۱۲٤
شکل ٦-۷- خط سیر الگوریتم مورچگان ۱۲۵
شکل ٦-۸- نمونه‌ای از گراف متناظر با مسئله توالی چهار سفارش ۱۲٦
شکل ٦-۹- مسیرهای ساخته شده توسط مورچه‌ها ۱۲۷
شکل ٦-۱۰- نمایش حرکت مورچه‌ها ۱۲۹
شکل ٦-۱۱ ساختار حافظه مورچه مستقر در گره A ۱۳۳
شکل ٦-۱۲ ساختار حافظه مورچه مستقر در گره D ۱۳٤
شکل ٦-۱۳- تعیین مسیر حرکت مورچه در الگوریتم AS ۱۳۵

شکل ٦-١٤- تعیین مسیر حرکت مورچه در الگوریتم ACS در تکرار اول ١٤٣
کل ٦-١٥- تعیین مسیر حرکت مورچه در الگوریتم ACS در تکرار دوم ١٤٧

فهرست جدول‌ها

جدول ١-١- زمان پردازش سفارش‌ها روی یک ماشین ٧
جدول ١-٢- زمان پردازش هر کار روی هر ماشین ٨
جدول١-٣- زمان پردازش سفارش‌ها روی ماشین‌ها ١٢
جدول ١-٤- مجموع زمان انجام سفارش‌ها روی هر ماشین ١٢
جدول ١-٥- مثالی از شبیه‌سازی یک مسئله زمان‌بندی قطعی ٢١
جدول٢-١- اطلاعات مربوط به پردازش سفارش‌ها ٢٥
جدول ٢-٢ محاسبه‌ی تابع هدف برای جواب ابتدایی ٢٦
جدول٢-٣ - توالی خالی از سفارش‌ها ٢٧
جدول٢-٤- اعداد تصادفی فرضی ٢٨
جدول٢-٥- اعداد تصادفی فرضی ٢٩
جدول٣-١- اعداد تصادفی فرضی ٤٣
جدول ٤-١- جواب‌های بررسی شده طی تکرارهای ١تا ٤ ..... ٥٩
جدول ٤-٢- لیست حرکت های ممنوع در هر تکرار ٥٩
جدول ٤-٣- مثالی از معیار فراوانی ٦٧
جدول ٤-٤- مثالی از مکانیسم شروع دوباره ٧٢
جدول ٥-١ جمعیت ترکیبی از افراد مختلف ٨٣
جدول ٥-٢- جواب‌های مسئله‌ی تعیین توالی سفارش‌ها ٨٥
جدول ٥-٣- مقدار تابع برازندگی و احتمال انتخاب هر جواب ٨٦
جدول ٥-٤- اعداد تصادفی فرضی ٨٦
جدول ٥-٥- مقدار تابع برازندگی و رتبه‌ی جواب‌ها ٨٧
جدول ٥-٦- جواب‌ها و تابع برازندگی مربوط به آنها ٨٨
جدول ٥-٧- نتایج حاصل از مسابقات دوبه‌دو ٨٨
جدول ٥-٨- نتایج حاصل از مسابقات دوبه‌دو در دور دوم ٨٨
جدول ٥-٩- نسل پالایش شده ٨٩
جدول ٥-١٠- تقاطع حفظ تقدم ٩٠
جدول ٥-١١- تقاطع منظم ٩١
جدول ٥-١٢- تقاطع تطبیق یافته ناقص ٩٢
جدول ٥-١٣ طرز نوشتن اعداد صحیح در مبنای ١٠، مبنای ٢ و دستگاه خاکستری ٩٩
جدول ٥-١٤- اعداد تصادفی فرضی ١١٤
جدول ٥-١٥- اعداد تصادفی فرضی ١١٥

جدول ٦-١ اختصاص مقدار اولیه‌ی فرمون و مقدار ابتکاری به هر مسیر ١٣٣
جدول ٦-٢- تعیین احتمال انتخاب هر مسیر ١٣٣
جدول ٦-٣- تعیین احتمال انتخاب هر مسیر ١٣٤
جدول ٦-٤- تعیین احتمال انتخاب هر مسیر ١٣٤
جدول ٦-٥- مسیرهای ساخته شده توسط مورچه‌ها ١٣٥
جدول ٦-٦- میزان فرمون نهایی در هر مسیر ١٣٥
جدول ٦-٧- میزان فرمون مسیرها پس از فرمون‌ریزی ١٣٦
جدول ٦-٨- تعیین احتمال انتخاب هر مسیر در تکرار اول ١٤٢
جدول ٦-٩- تعیین احتمال انتخاب هر مسیر ١٤٢
جدول ٦-١٠- میزان فرمون پس از به‌روز رسانی محلی در هر مسیر ١٤٣
جدول٦-١١- مقدار فرمون مسیرها در ابتدای حرکت مورچه دوم ١٤٤
جدول ٦-١٢- احتمال انتخاب هر مسیر برای مورچه دوم ١٤٤
جدول ٦-١٣- مسیرهای ساخته شده توسط مورچه‌ها ١٤٥
جدول ٦-١٤- میزان فرمون پس از به‌روز رسانی سراسری در هر مسیر ١٤٥
جدول ٦-١٥- میزان فرمون نهایی در هر مسیر ١٤٦
جدول ٦-١٦- تعیین احتمال انتخاب هر مسیر در تکرار دوم ١٤٦
جدول ٦-١٧- تعیین احتمال انتخاب هر مسیر در تکرار دوم ١٤٧
جدول ٦-١٨- مقادیر پیشنهادی پارامترها ١٤٨

فصل نخست

مقدمه و تعاریف

۱-۱- مقدمه

الگوریتم‌های فرا ابتکاری[1] عموما در حل مسائلی به‌کار می‌روند که مسائل بهینه‌سازی ترکیبیاتی[2] نامیده می‌شوند. این مسائل ظاهری فریبنده دارند زیرا به سادگی بیان می‌شوند و به نظر می‌آید که پیچیدگی خاصی ندارند در حالی که حل آن‌ها مستلزم صرف زمان زیاد است. فصل نخست کتاب ضمن بیان تعاریف و مقدمات، به ویژگی‌های این گونه از مسائل خواهد پرداخت و برخی از مسائل شناخته شده‌ی این حوزه نیز در همین فصل معرفی می‌شوند. یک مسئله از حوزه‌ی تعیین توالی سفارش‌ها نیز در این مجموعه از مسائل معرفی می‌شود که روش حل این مسئله توسط هر یک از فرا ابتکاری‌ها در دیگر فصل‌های کتاب توصیف می‌شود. از آن‌جایی که حل مثال مرجع این کتاب نیازمند آشنایی با مفاهیم شبیه‌سازی است بخش پایانی فصل نخست به مرور کوتاهی بر مفاهیم شبیه‌سازی اختصاص یافته است.

۱-۲- مسائل بهینه‌سازی

علم تصمیم‌گیری روز به روز با مسائلی مواجه می‌شود که پیچیدگی آن‌ها افزایش می‌یابد. مسئله‌هایی که باید حل شوند بیشتر در قالب یک مسئله‌ی بهینه‌سازی مطرح می‌شوند. هر مسئله‌ی بهینه‌سازی دارای یک یا چند تابع هدف است که باید کمینه یا بیشینه شود. بیان هر مسئله‌ی بهینه‌سازی معمولا همراه با محدودیت‌هایی است که باید رعایت شود. هر جواب مسئله باید این محدودیت‌ها را رعایت کند تا یک جواب موجه (قابل قبول) باشد. دو نوع مسئله‌ی بهینه‌سازی قابل تشخیص است. مسائل بهینه‌سازی گسسته و مسائل بهینه‌سازی که متغیر تصمیم آن‌ها پیوسته است. مسائل بهینه‌سازی ترکیبیاتی در حوزه مسائل گسسته قرار می‌گیرند. گسسته بودن مسئله لزوما به معنای این که متغیر تصمیم مقادیر عدد صحیح را اتخاذ می‌کند نیست. ساختار متغیرهای تصمیم در مسائل گسسته تفاوت بنیادی با متغیرهای تصمیم در مسائل بهینه‌سازی پیوسته دارد. این تفاوت در مثال‌های این فصل روشن می‌شود.

---

1 . Meta Heuristic (Meta- Heuristic)

2 . Combinatorial Optimization

فرض کنید هدف از حل مسئله‌ی بهینه‌سازی کمینه کردن تابع هدف باشد که تابع هدف آن با $f(x)$ و مجموعه جواب‌های موجه (منطقه موجه) آن با X نمایش داده شده است. مدل ریاضی مسئله‌ی بهینه‌سازی، به ترتیب نشان داده شده در رابطه‌ی ۱-۱ است.

$$\begin{aligned} Min \quad & f(x) \\ s.t \quad & \\ & x \in X \end{aligned} \qquad ۱-۱$$

در تمام مطالب این کتاب فرض بر این است که تابع هدف مسئله باید کمینه شود مگر در شرایطی که خلاف این مطلب به وضوح مشخص شده باشد.

۱-۳- مسائل بهینه سازی ترکیبیاتی

یک مسئله‌ی بهینه‌سازی ترکیبیاتی مسئله‌ای است که دارای تابع هدف با مقدار حقیقی است و مجموعه جواب‌های موجه آن متناهی است. هدف از حل مسئله می‌تواند بیشینه‌سازی یا کمینه‌سازی تابع هدف باشد. با توجه به متناهی بودن جواب‌های موجه، در واقع مسئله‌ای وجود ندارد زیرا کافی است همه‌ی جواب‌ها ارزیابی شوند؛ اما هدف این است از روش‌های کاراتری به جای بررسی همه‌ی جواب‌ها استفاده شود[۱۷]. بسیاری از مسائل واقعی و حوزه‌ی تئوری از چنین ماهیتی برخوردارند. برای نمونه، می‌توان به مسئله‌ی یافتن کوتاه‌ترین مسیر در یک شبکه‌ی حمل و نقل یا مسئله‌ی تعیین توالی سفارش‌ها در یک خط تولید اشاره کرد[۱۱]. بهینه‌سازی ترکیبیاتی، بهینه‌سازی گسسته نیز نامیده می‌شود[۸].

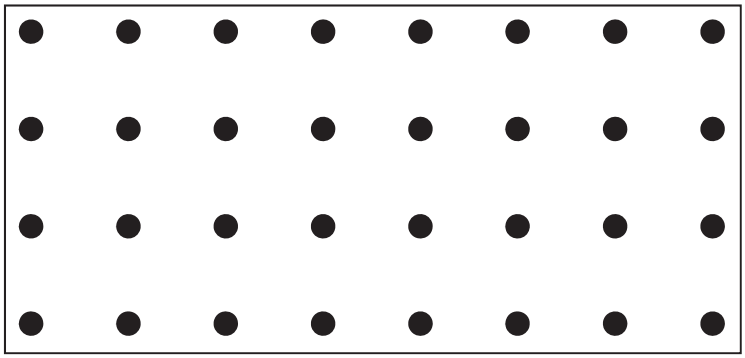
مسئله‌ای را در نظر بگیرید که هدف از حل آن تعیین بهترین ترتیب انجام ٤ کار روی یک ماشین است؛ به گونه‌ای که متوسط زمان انتظار کارها کمینه شود. تعداد جواب‌های مسئله برابر با !٤ است. در شکل ۱-۱ مجموعه متناهی جواب‌های این مسئله نشان داده شده است. هر ترتیب انجام کار، یک متوسط زمان انتظار به دست می‌دهد. جواب بهینه جوابی است که کمترین میانگین زمان انتظار را باعث شود.

•(۱،۲،۳،٤)	•(۲،۱،۳،٤)	•(۳،۱،۲،٤)	•(٤،۱،۲،۳)
•(۱،۲،٤،۳)	•(۲،۱،٤،۳)	•(۳،۱،٤،۲)	•(٤،۱،۳،۲)
•(۱،۳،۲،٤)	•(۲،۳،۱،٤)	•(۳،۲،۱،٤)	•(٤،۲،۱،۳)
•(۱،۳،٤،۲)	•(۲،۳،٤،۱)	•(۳،۲،٤،۱)	•(٤،۲،۳،۱)
•(۱،٤،۲،۳)	•(۲،٤،۱،۳)	•(۳،٤،۱،۲)	•(٤،۳،۱،۲)
•(۱،٤،۳،۲)	•(۲،٤،۳،۱)	•(۳،٤،۲،۱)	•(٤،۳،۲،۱)

شکل ۱- ۱- مجموعه جواب‌های مسئله‌ی تعیین توالی سفارش‌ها با ۴ سفارش

متغیر تصمیم x می‌تواند یکی از جایگشت‌های نشان داده شده در شکل ۱-۱ را به خود اختصاص دهد. ویژگی بارز چنین مسائلی محدود و قابل‌شمارش‌بودن تعداد جواب‌های موجه مسئله است. تعداد جواب-های موجه این مسئله و مثال‌هایی که در ادامه این فصل ارائه می‌شود را می‌توان با استفاده از اصول شمارش مثل ترتیب و ترکیب به‌دست آورد.

نکته‌ی قابل توجه این است که تعداد جواب موجه این مسئله با n سفارش برابر با $n!$ است و حتی برای تعداد محدودی سفارش نیز این تعداد بسیار چشم‌گیر خواهد بود. برای نمونه، اگر تعداد سفارش در یک مسئله برابر با ۱۵ باشد تعداد جواب‌های ممکن برابر با ۱،۳۰۷،٦۷٤،۳٦۸،۰۰۰ است که تعداد آن‌ها بسیار چشم‌گیر است. در ادامه‌ی این کتاب از تصویر شماتیک همانند شکل ۱-۲ برای نمایش فرضی مجموعه جواب‌های موجه یک مسئله‌ی بهینه‌سازی ترکیبیاتی استفاده می‌شود.



شکل ۱-۲- نمایش فرضی مجموعه نقاط منطقه‌ی موجه یک مسئله‌ی بهینه‌سازی ترکیبیاتی

۱-۴- مثال‌هایی از مسائل بهینه‌سازی ترکیبیاتی

مسائل متنوعی را می‌توان در حوزه‌ی تحقیق در عملیات مشاهده کرد که در گروه مسائل بهینه‌سازی ترکیبیاتی قرار می‌گیرند که تعدادی از آن‌ها در این بخش معرفی می‌شوند. آخرین مثالی که معرفی می‌شود مربوط به یک مسئله تعیین توالی سفارش‌ها است که مثالی مرجع در این کتاب است و جزئیات روش حل آن توسط الگوریتم‌های فراابتکاری در فصل مربوط به هر الگوریتم، توصیف خواهد شد. حل دیگر مسائل به خواننده واگذار می‌شود.

۱-۴-۱- مجموع زیرمجموعه

نخستین مثال معرفی شده از مجموعه مسائل بهینه‌سازی ترکیبیاتی یک مثال بسیار ساده است که در آن مسئله هیچ تابع هدفی ندارد و صرفا یک جست‌وجو در فضای جواب‌های مسئله قلمداد می‌شود. برای آشنایی با مسئله‌ی مجموع زیرمجموعه[1]، مجموعه‌ی S که شامل n عدد است را در نظر بگیرید:

$$S = \{A_1, A_2, ..., A_n\}$$

پرسش این است که آیا زیر مجموعه‌ای از S مثل S' می‌توان یافت که مجموع عناصر آن برابر با مقدار از پیش تعیین شده‌ی k باشد؟

$$\sum_j A_j = k \quad \forall A_j \in S'$$

مثال ۱-۱) فرض کنید مجموعه‌ی S به شرح زیر تعریف شده و مقدار k برابر با ۲۲ است:

$$S = \{۶,۱۰,۴,۱۶,۳۴,۲\}$$

یکی از جواب های مسئله عبارت است از:

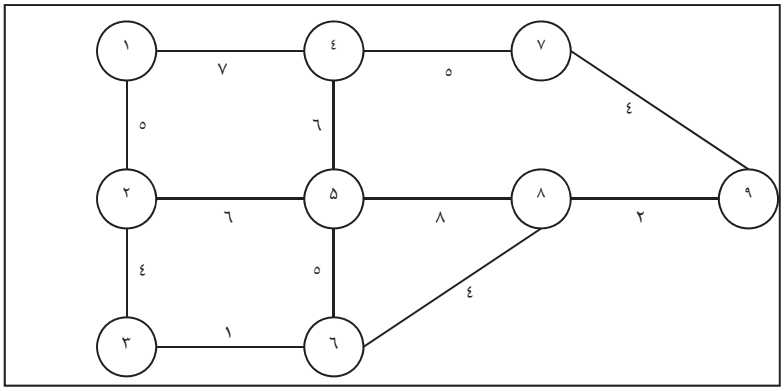
$$S' = \{۱۶,۴,۲\}$$

۱-۴-۲- کوتاه‌ترین درخت پوشای رتبه‌ی k

این مسئله یکی از سؤالات حوزه‌ی تحلیل شبکه است. در یک شبکه همانند شکل ۱-۳ که در آن طول کمان‌ها (شاخه‌ها) مشخص است درخت به مجموعه‌ای از شاخه‌ها و گره‌ها گفته می‌شود که اولا در آن

1 . Subset Sum

حلقه وجود نداشته باشد و دوم اینکه یک مسیر بین هر دو گره وجود داشته باشد. درخت پوشای رتبه‌ی k درختی است که تنها شامل k شاخه از شبکه باشد. برای نمونه، درخت متشکل از شاخه‌های ۲-۱، ۳-۲، ٤-۱ و شاخه ٥-۲ یک درخت رتبه‌ی ٤ است. سؤال کوتاه‌ترین درخت پوشای رتبه‌ی k[1] این است که به ازای k مشخص، درختی که مجموع طول شاخه‌های آن کمینه باشد، کدام است؟



شکل ۱-۳- نمونه‌ای از یک شبکه

۱-۴-۳- رنگ‌آمیزی گراف

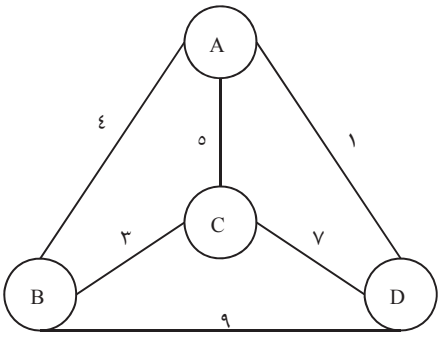
یک گراف (همانند شبکه‌ی نشان داده شده در شکل ۱-۳) که شامل مجموعه‌ای از گره‌ها و شاخه‌ها است را در نظر بگیرید. هدف از مسئله رنگ‌آمیزی گراف[2] این است که به هر گره شبکه یک رنگ اختصاص یابد به گونه‌ای که گره‌های متصل به یک شاخه دارای رنگ یکسان نباشند و تعداد رنگ‌های به کار رفته در رنگ‌آمیزی گراف کمینه شود.

۱-۴-۴- مسئله‌ی فروشنده‌ی دوره‌گرد

یکی از شناخته شده‌ترین مسائل حوزه‌ی بهینه‌سازی ترکیبیاتی، مسئله‌ی فروشنده‌ی دوره‌گرد[3] است. در این مسئله n شهر وجود دارد که بین هر دو شهر یک مسیر وجود دارد. یک فروشنده‌ی دوره‌گرد باید سفر خود را از شهر i شروع کرده و سرانجام به همین شهر باز گردد. پرسش این است که با شرط

1. Minimum k-tree
2. Graph Coloring Problem (GCP)
3. Travelling Salesman Problem (TSP)

عبور از همه‌ی شهرها و تنها یک مرتبه عبور از هر شهر، ترتیب عبور از شهرها چگونه باشد تا کل مسیر پیموده شده کمینه شود؟ در شکل ۱-٤ تصویری از مسیرهای موجود در یک مسئله‌ی فروشنده‌ی دوره-گرد نمایش داده شده است که شامل ٤ شهر است.



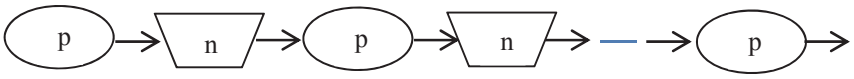
شکل ۱-۴- مثالی از مسیرهای موجود در مسئله‌ی فروشنده‌ی دوره‌گرد

۱-۴-۵- مسئله‌ی تعیین توالی سفارش‌ها

این مسئله بیانی کاربردی از مسئله‌ی فروشنده‌ی دوره‌گرد است که در حوزه‌ی مدیریت تولید و عملیات بسیار رایج است. مسئله‌ی تعیین توالی سفارش‌ها[1] شکل‌های متنوعی دارد که دو شکل متفاوت از این مسائل در این بخش معرفی می‌شود. یکی از این مسائل مثالی پایه برای نشان‌دادن چگونگی به‌کارگیری فراابتکاری‌ها است. جزئیات روش حل مسئله توسط هر یک از فراابتکاری‌ها در فصل‌های آتی خواهد آمد.

یک خط تولید را در نظر بگیرید که در آن n محصول (سفارش) باید از m ماشین عبور کند تا تکمیل شوند. سفارش‌ها باید به ترتیب از ماشین ۱، ۲، ...، m عبور کنند. بین هر دو ماشین یک انبار میان‌گیر با ظرفیت محدود یا نامحدود وجود دارد (شکل ۱-۵). زمان انجام هر کار روی هر ماشین مشخص است. زمان انجام کار j روی ماشین i را با نماد p_{ij} نشان می‌دهیم. سفارش‌ها به همان ترتیبی که روی نخستین ماشین پردازش می‌شوند روی دیگر ماشین‌ها نیز با همان توالی تکمیل می‌شوند. همه‌ی سفارش‌ها نیز در زمان صفر (زمان شروع حل مسئله) در دسترس هستند. پرسش این است که ترتیب پردازش سفارش‌ها چگونه باشد تا هدفی خاص برآورده شود.

1. Job Sequencing



شکل ۱-۵- پیکربندی ماشین‌آلات و انبارهای میانگیر در مسئله‌ی جریان کارگاهی

این نوع مسئله که جریان گارگاهی[1] نامیده می‌شود می‌تواند با مشخصه‌های دیگری نیز مطرح شود برای نمونه، ظرفیت انبارها می‌تواند محدود یا نامحدود باشد، ترتیب پردازش سفارش‌ها روی ماشین‌آلات می‌تواند از یک ماشین به ماشین دیگر تغییر کند و یا این که حل مسئله می‌تواند هدفی مثل کمینه کردن زمان تکمیل یا زمان دیرکرد را دنبال کند. علاقمندان می‌توانند به منابع [۲۲] و [۸] و [۹] مراجعه کنند. مثال‌های عددی زیر، یک شکل ساده و یک شکل پیچیده از این مسائل را نشان می‌دهد. هرچند مثال ۱-۲ به عنوان مبنای بحث دیگر فصل‌های کتاب انتخاب شده است اما ایده‌هایی که برای حل این دو مثال به کار می‌رود، ساختاری یکسان دارد.

مثال ۱-۲) یک تابع هدف رایج در مسائل تعیین توالی سفارش‌ها مجموع زمان تأخیر است. این تابع هدف زمانی کاربرد دارد که برای هر سفارش یک زمان تحویل مشخص شده باشد. فرض کنید زمان تکمیل کار j با C_j و زمان تحویل آن با d_j نمایش داده شده است. اگر زمان تکمیل پس از زمان تحویل باشد میزان تأخیر به اندازه‌ی $C_j - d_j$ است؛ وگرنه برابر با صفر است. مجموع زمان تأخیر (T) با استفاده از رابطه زیر محاسبه می‌شود:

$$T = \sum_{j=1}^{n} [cj - dj]^{+} \qquad [cj - dj]^{+} = Max(0, c_j - d_j)$$

در یک شکل ساده‌ی مسئله‌ی تعیین توالی سفارش‌ها فرض کنید ۱۰ سفارش وجود دارند که باید روی یک ماشین پردازش شوند که زمان پردازش و زمان تحویل آن‌ها در جدول زیر آورده شده است. ترتیب پردازش سفارش‌ها چگونه باشد تا مجموع زمان تأخیر کمینه شود؟

جدول ۱-۱- زمان پردازش سفارش‌ها روی یک ماشین

سفارش	J۱	J۲	J۳	J٤	J٥	J٦	J۷	J۸	J۹	J۱۰
زمان پردازش	٤	٦	۸	۲	۱	٥	۷	۳	٥	۷
موعد تحویل	۹	۱۲	۱٥	۸	۱۱	۱۰	۸	۱۲	۹	۱٥

1. Flow Shop

مثال ۱-۳) در یک مسئله‌ی پیچیده‌ی جریان کارگاهی باید ۱۰ سفارش روی ۵ ماشین پردازش شوند. بین هر دو ماشین انبارهایی وجود دارد که ظرفیت انبارهای میانگیر برابر با ۲ است. اگر انبار i خالی نباشد کار تکمیل شده روی ماشین i باقی می‌ماند (تا زمانی که انبار، دست‌کم ظرفیت یک سفارش را داشته باشد) و باعث توقف ماشین می‌شود. زمان پردازش کار J روی ماشین i به شرح جدول ۱-۲ است. ترتیب پردازش سفارش‌ها چگونه باشد تا زمان تکمیل آخرین سفارش (که با نماد C_{Max} نشان داده می‌شود) کمینه شود؟

جدول ۱-۲- زمان پردازش هر کار روی هر ماشین

سفارش / ماشین	۱	۲	۳	٤	٥	٦	۷	۸	۹	۱۰
۱	۷٥	۸۷	۱۳	۱۱	٤۱	٤۳	۹۳	٦۹	۸۰	۱۳
۲	۲٦	۳۷	۲٥	۹٥	٤۹	۱۲	٥۹	۱۷	٤٦	۲۰
۳	٤۸	٤	۹۲	۹۳	۷۲	٤٥	٥	۹۸	۹۳	۱۷
٤	۲٦	٦۷	٤	۱٤	۹۳	٥٤	۲۱	۲۰	٦	۱۸
٥	۷۷	۹٤	۹	٥۷	۲۹	۷۹	٥٥	۷۳	٦٥	۸٦

این مثال که با استفاده از اطلاعات مجموعه مسائل تایلارد [۲٦] بازسازی شده است دارای !۱۰ جواب است که تعیین بهترین جواب از بین ۳،٦۲۸،۸۰۰ جواب موجه مسئله می‌تواند به روش‌های مختلفی صورت بگیرد. ساختار منطقه‌ی موجه به گونه‌ای است که هر یک از این روش‌ها ممکن است کارایی خود را از دست بدهند.

در منطقه‌ی موجه‌ی مسائل بهینه‌سازی ترکیبیاتی بی‌نظمی‌هایی وجود دارد که حل این گروه از مسائل را با پیچیدگی مواجه می‌کند. این منطقه ممکن است شامل هزاران نقطه‌ی بهینه‌ی محلی باشد که الگوریتم حل مسئله را به دام انداخته و مانع از رسیدن به جواب بهینه شوند. در شکل ۱-٦ تغییرات تابع هدف یک مسئله‌ی تعیین توالی سفارش‌ها نشان داده شده است. در این مسئله تعداد سفارش برابر با ۷ است که تعداد کل نقاط جواب‌های مسئله برابر با ٥۰٤۰ است. مقدار تابع هدف به ازای ٤۰۰ جایگشت از جواب‌های مسئله در این شکل نمایش داده شده است. هر جایگشت در مقایسه با جایگشت مجاور آن (روی محور افقی) تنها در اولویت دو سفارش تفاوت دارد. برای نمونه، جایگشت نخست دارای توالی (۱،۲،۳،٤،٥،٦،۷) است و در جایگشت دوم سفارش ٦ و ۷ جابه‌جا شده‌اند که به (۱،۲،۳،٤،٥،۷،٦) منجر شده است. ملاحظه می‌شود که این ناحیه شامل انبوهی از نقاط بهینه‌ی محلی است. فرایند حل مسائل بهینه‌سازی ترکیبیاتی باید توان گریز از این نواحی را داشته باشد تا جواب مطلوبی به دست بدهد.