دانشکده مهندسی برق و کامپیوتر

پاین نامه کارشناسی ارشد در رشته مهندسی کامپیوتر (هوش مصنوعی)

الگوریتم‌های موازی ناهمگون برای حل مسایل بهینه‌سازی فضای پیوسته

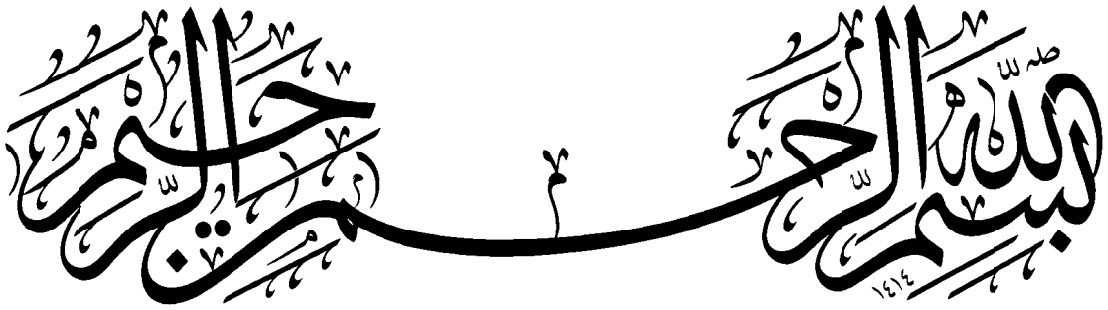
توسط:

محمود صفی خانی

استاد راهنما:

دکتر سراج الدین کاتبی

شهریور ماه ۱۳۸۸


بسم الله الرحمن الرحيم
١٤١٤

به نام خدا

الگوریتم های موازی نا همگون برای حل مسایل بهینه سازی فضای پیوسته

به وسیله ی:

محمود صفی خانی

پایان نامه

ارائه شده به تحصیلات تکمیلی دانشگاه به عنوان بخشی
از فعالیت‌های تحصیلی لازم برای اخذ درجه کارشناسی ارشد

در رشته ی:

مهندسی کامپیوتر-هوش مصنوعی

از دانشگاه شیراز

شیراز

جمهوری اسلامی ایران

ارزیابی شده توسط کمیته پایان نامه با درجه: -عالی-

دکتر سراج الدین کاتبی، استاد بخش مهندسی کامپیوتر (رئیس کمیته) . . . . . .
دکتر کورش زیارتی ، استادیار بخش مهندسی کامپیوتر . . . . . .
دکتر زهره عظیمی فر، استادیار بخش مهندسی کامپیوتر . . . . . .

شهریور ماه ۱۳۸۸

تقدیم به:

پدر، مادر

و

همسر عزیزم

سپاسگزاری

بدینوسیله مقام شامخ اساتید بزرگوار جناب آقای دکتر سراج الدین کاتبی، جناب آقای دکتر کورش زیارتی و سرکار خانم دکتر زهره عظیمی فر را که سالها خوشه چین علم و معرفتشان بودم، ارج می نهم و از راهنمایی ها و همکاری های ارزنده و خردمندانه آنها در مراحل مختلف کمال سپاسگذاری را دارم.

چکیده

الگوریتم های موازی نا همگون برای حل مسایل بهینه سازی فضای پیوسته

توسط:

محمود صفی خانی

بهینه سازی در واقع شامل حل یک مساله تصمیم گیری، شامل انتخاب مقـادیر بـرای متغیـر هـای مساله با در نظر گرفتن یک هدف است. روشها و الگوریتم های متفاوتی برای حل مسایل بهینه سازی ابداع شده اند. به کار بردن دو یا چند روش برای حل یک مساله بهینه سازی به شکلی که همکاری بـین روشـها نتیجه بهتر و سریعتری را عاید ما سازد همیشه مورد توجه بوده است. این ایده بـه خصـوص بـا ورود روش های جدید به حیطه بهینه سازی تقویت شد. برای ترکیـب الگـوریتم هـای مختلـف بـا یکـدیگر روشـهای متفاوتی به کار برده شده است استفاده از عمگرهای یک الگوریتم در دل الگوریتم دیگر، اجرای مـوازی بـر روی زیر جمعیت های مختلف و اجرای نوبتی آن ها بر روی یک جمعیت مشترک. در حالتی کـه الگـوریتم های مختلف بر روی زیر جمعیت ها جدا از هم اجرا می شود معمولا از یـک عملگـر بـرای تعـویض برخـی اعضای زیر جمعیت ها استفاده می شود. در الگوریتم های بهینـه سـازی همیشـه بایـد بـه گونـه ای بـین جستجوی محلی و سراسری تعادل ایجاد نمود. چرا که جستجوی محلـی بیشـتر منجـر بـه افتـادن در دام بهینه های محلی و تمرکز بیشتر بر روی جستجوی سراسری باعث کـاهش سـرعت همگرایـی و بعضـا بـه دست نیامدن جواب قابل قبول می باشد. در این رساله چندین الگوریتم بهینه سـازی مختلـف بـا خـواص متفاوت ترکیب شده است جواب های دست آمده نسبت بـه هـر کـدام از الگوریتمهـای قبلـی بهتـر مـی باشد.الگوریتمهایی که برای این منظور استفاده شده الگوریتم ژنتیک و ترکیب های مختلـف آن بـا دیگـر الگوریتم ها ، الگوریتم بهینه سازی تجمعـی ذرات و ترکیـب هـای مختلـف و ارتقـا یافتـه آن و همچنـین الگوریتم ژنتیک چند عاملی می باشد الگوریتم های ذکر شده به روشهای مختلفی با یکدیگر ترکیب شده و روش حاصل با توابع محک گوناگون و مسایل با ابعاد بالا تست شده است در اغلب موارد پایداری الگوریتم ترکیبی نسبت به فضاهای گو نـاگون و ابعـاد بـالا بیشـتر شـده اسـت ترکیـب الگـوریتم هـای مختلـف بـا خصوصیات مکمل باعث شده تا نقاط ضعف یک الگوریتم در یک فضا توسط الگوریتم های دیگـر پوشـش داده شود.

فهرست مطالب

عنوان صفحه

فصل اول ۱

۱- مقدمه ۲

۱-۱ بهینه سازی ۲

۱-۲ انواع مسائل بهینه سازی ۳

۱-۳ روش های حل مساله ۴

۱-۴ روشهای هیبرید برای حل مسائل بهینه سازی ۵

۱-۵ مشکلات حل مسائل بهینه سازی ۶

۱-۵-۱ انتخاب تابع ارزیابی ۶

۱-۵-۲ اندازه فضای جستجو ۷

۱-۶ مفاهیم اساسی ۷

۱-۶-۱ روش نمایش ۸

۱-۶-۲ هدف ۸

۱-۶-۳ تابع ارزیابی ۸

فصل دوم مروری بر الگوریتم ژنتیک و بهینه سازی تجمعی ذرات ۹

۲- الگوریتم ژنتیک و بهینه سازی تجمعی ذرات ۱۰

۲-۱ مقدمه ای بر الگوریتم ژنتیک ۱۰

۲-۲ ساختار الگوریتم ژنتیک ۱۰

۲-۳ عملگرهای اساسی الگوریتم ژنتیک ۱۲

۲-۳-۱ عملگر انتخاب ۱۳

۲-۳-۲ عملگر ترکیب ۱۵

۲-۳-۳ عملگر جهش ۱۵

۲-۴ الگوریتم ژنتیک با کد گذاری دودوئی ۱۶

۲-۴-۱ عملگر ترکیب در کد گذاری دودوئی ۱۶

۲-۴-۲ عملگر جهش در کد گذاری دودویی ۱۶

۲-۴-۳ معایب کد گذاری دودوئی ۱۷

۲-۵ الگوریتم ژنتیک با کد گذاری حقیقی ۱۷

۲-۵-۱ مزایای کد گذاری حقیقی ۱۹

۲-۵-۲ عملگرهای ترکیب در کد گذاری حقیقی ۲۱

۲-۵-۳ عملگرهای جهش در کد گذاری حقیقی ۲۲

۲-۶ بهینه سازی تجمعی ذرات ۲۴

فصل سوم بررسی روش های ارتقاء بهینه سازی تجمعی ذرات ۲۸

۳-۱ ارتقاء PSO با استفاده از ضرائب شتاب دهنده متغیر با زمان ۲۹

۳-۱-۱ الگوریتم ارتقاء PSO با استفاده از ضرائب شتاب دهنده متغیر با زمان ۳۱

۳-۱-۲ سه راهبرد غیر خطی متغیر با زمان برای تنظیم ضریب شناختی ۳۱

۳-۲ PSO با به کار بردن توزیع گوسی برای ضرائب ۳۳

۳-۲-۱ *PSO* با به کار بردن توزیع گوسی ۳۷

۳-۲-۲ الگوریتم CPSO-GD ۴۰

۳-۳ خود تطبیقی برای وزن میانی در بهینه سازی تجمعی ذرات ۴۳

۳-۳-۱ الگوریتم خود وفقی بهینه سازی تجمعی ذرات ۴۵

۳-۴ الگوریتم مقایسه ای بهینه سازی تجمعی ذرات ۴۷

فصل چهارم روش های هیبرید برای حل مسائل بهینه سازی پیوسته ۵۰

۴- مروری بر روش های هیبرید برای حل مسائل بهینه سازی پیوسته ۵۱

۴-۱ استفاده از الگوریتم های ترکیبی برای حل مسائل ۵۱

۴-۱-۱ روش های هیبرید همگون ۵۲

۴-۱-۲ روش های هیبرید نا همگون ۵۵

۴-۲ مروری بر چند الگوریتم بهینه سازی با ساختار هیبرید ۵۶

۴-۲-۱ الگوریتم ژنتیک چند عاملی ۵۶

۴-۲-۲ بهینه سازی ژنتیک اجتماع پرندگان ۶۸

۴-۲-۳ یک الگوریتم هیبرید با ترکیب الگوریتم ژنتیک و جستجوی هارمونی ۷۰

۴-۲-۴ هیبرید الگوریتم ژنتیک و بهینه سازی تجمعی ذرات ۷۴
۴-۲-۵ یکپارچه سازی برنامه ریزی تکاملی و بهینه سازی تجمعی ذرات ۷۸
۴-۲-۶ الگوریتم بهینه سازی تجمعی ذرات با سه زیر جمعیت ۸۱
فصل پنجم روش های پیشنهادی نتایج، مقایسه و نتیجه گیری ۸۵
۵- روش های پیشنهادی..... ۸۶
۵-۱ ترکیب الگوریتم ژنتیک و بهینه سازی تجمعی ذرات..... ۸۷
۵-۲ الگوریتم هیبرید بهینه سازی تجمعی ذرات با توزیع گوسی ضرائب ، بهینه سازی تجمعی ذرات مقایسه ای و بهینه سازی تجمعی ذرات استاندارد ۸۹
۵-۳ الگوریتم هیبرید بهینه سازی تجمعی ذرات با تنظیم وزن میانی، بهینه سازی تجمعی ذرات مقایسه ای و بهینه سازی تجمعی ذرات با همکاری اگوریتم ژنتیک ۹۱
۵-۴ ترکیب الگوریتم ژنتیک، بهینه سازی تجمعی ذرات و الگوریتم ژنتیک چند عاملی ۹۲
۵-۵ ترکیب الگوریتم ژنتیک الگوریتم ژنتیک چند عاملی بهینه سازی تجمعی ذرات با ضرائب گوسی و الگوریتم ژنتیک با همکاری جستجوی هارمونی ۹۳
۵-۶ توابع محک مورد استفاده ۹۵
۵-۶-۱ تابع محک شماره یک ۹۶
۵-۶-۲ تابع محک شماره دو ۹۷
۵-۶-۳ تابع محک شماره سه ۹۸
۵-۶-۴ تابع محک شماره چهار ۹۹
۵-۶-۵ تابع محک شماره پنج ۱۰۰
۵-۶-۶ تابع محک شماره شش ۱۰۱
۵-۷ نتایج، مقایسه و نتیجه گیری ۱۰۲
۵-۷-۱ الگوریتم هیبرید ژنتیک و بهینه سازی تجمعی ذرات ۱۰۲
۵-۷-۲ الگوریتم هیبرید بهینه سازی تجمعی ذرات با توزیع گوسی ضرائب ، بهینه سازی تجمعی ذرات مقایسه ای و بهینه سازی تجمعی ذرات استاندارد ۱۰۳
۵-۷-۳ الگوریتم هیبرید بهینه سازی تجمعی ذرات با تنظیم وزن میانی، بهینه سازی تجمعی ذرات مقایسه ای و بهینه سازی تجمعی ذرات با همکاری اگوریتم ژنتیک ۱۰۴
۵-۷-۴ ترکیب الگوریتم ژنتیک، بهینه سازی تجمعی ذرات و الگوریتم ژنتیک چند عاملی ۱۰۵
۵-۷-۵ ترکیب الگوریتم ژنتیک الگوریتم ژنتیک چند عاملی بهینه سازی تجمعی ذرات با ضرائب گوسی و الگوریتم ژنتیک با همکاری جستجوی هارمونی به صورت مرحله ای ۱۰۶
۵-۷-۶ ترکیب الگوریتم ژنتیک الگوریتم ژنتیک چند عاملی بهینه سازی تجمعی ذرات با ضرائب گوسی و الگوریتم ژنتیک با همکاری جستجوی هارمونی با زیر جمعیت های متفاوت ۱۰۷
۵-۸ سخن پایانی ۱۰۸
فصل ششم منابع ۱۰۹

۶-۱ منابع لاتین ۱۱۰

فهرست جدولها

عنوان و شماره	صفحه
جدول شماره ۱ مقایسه بین الگوریتم ژنتیک، PSOو ترکیب الگوریتم ژنتیک و PSO	۱۰۲
جدول شماره ۲ مقایسه بین PSO، PSOUg، CPSO، TS-PSOو ترکیب PSO&CPSO&PSOUg	۱۰۳
جدول شماره ۳ مقایسه بین SAPSO، GPSO، CPSOو ترکیب SAPSO&CPSO&GPSO	۱۰۴
جدول شماره ۴ مقایسه بین GA، PSO، MAGAو ترکیب MAGA&GA&PSO	۱۰۵
جدول شماره ۵ مقایسه بین GA، HSGA، MAGA، PSOUg و ترکیب HSGA&MAGA&GA&PSOUg	۱۰۶
جدول شماره ۶ مقایسه بین GA، HSGA، MAGA، PSOUg و ترکیب MAGA&GA&PSOUg&HSGA	۱۰۷

فهرست شکلها

عنوان صفحه

شکل ۱ یک تابع ارزیابی پیچیده ۷
شکل ۲ شبه کد الگوریتم ژنتیک ۱۳
شکل ۳ مثالی از جهش در یک رشته دودوئی ۱۷
شکل ۴ شبه کد الگوریتم بهینه سازی تجمعی ذرات ۲۵
شکل ۵ مجموع دو متغیر تصادفی با تابع چگالی احتمال یکنواخت ۳۹
شکل ۶ شبه کد الگوریتمCPSO-GD ۴۲
شکل ۷ شبه کد الگوریتم خود وفقی بهینه سازی تجمعی ذرات ۴۶
شکل ۸ شمای کلی globally PEAs ۵۳
شکل ۹ شمای کلی روش Coarse-grained PEA ۵۴
شکل ۱۰ شمای کلی Fine grained PEAs ۵۴
شکل ۱۱ مدل شبکه عامل ۵۹
شکل۱۲ شبه کد الگوریتم عملگر خود فراگیری ۶۵
شکل ۱۳ شبه کد الگوریتم ژنتیک چند عاملی ۶۶
شکل ۱۴ فرایند بهینه سازی NHRVGA ۷۲
شکل ۱۵ فلوچارت تولید بردار هارمونی ۷۴

شکل ۱۶ شمای الگوریتم هیبرید ژنتیک و بهینه سازی تجمعی ذرات ۷۵
شکل ۱۷ شبه کد الگوریتم هیبرید ژنتیک و بهینه سازی تجمعی ذرات ۷۷
شکل ۱۸ شبه کد الگوریتم یکپارچه سازی برنامه ریزی تکاملی و بهینه سازی تجمعی ذرات ۸۰
شکل ۱۹ شمای کلی ترکیب الگوریتم ژنتیک و PSO ۸۸
شکل ۲۰ شمای کلی الگوریتم بهینه سازی تجمعی ذرات با سه زیر جمعیت ۹۰
شکل ۲۱ شمای کلی الگوریتم هیبرید بهینه سازی تجمعی ذرات، الگوریتم ژنتیک و الگوریتم ژنتیک چند عاملی ۹۳
شکل ۲۲ شمای کلی الگوریتم hyGaPSOUgHsgaMAGA ۹۵
شکل ۲۳DeJong function ۹۶
شکل ۲۴ Ackley function ۹۷
شکل ۲۵ Griewank function ۹۸
شکل ۲۶Rastrigin function ۹۹
شکل ۲۷ Rosenbrock function ۱۰۰
شکل ۲۸ Schwefel function ۱۰۱

فصل اول

مقدمه

۱- مقدمه

در مسایل بهینه سازی انواع مختلف آن با توجه به نوع فضای جستجو و طرز فرمول بندی آنها تشخیص داده شده است.روش های حل مسایل سنتی در بهینه سازی بیشتر سعی بر تقریب مسایل غیرخطی به صورت یک مساله خطی را داشته اند که این همیشه به سادگی امکان پذیر نمی باشد. روش های جدید بر خلاف روشهای سنتی پایه ریاضی و جبری قوی نداشته ولی وابستگی کمتری به نوع مساله دارند. از دلایل مهم فراگیر شدن روش های جدید سادگی در پیاده سازی و میزان بالای انعطاف پذیری آن ها می باشد.

به کار بردن دو یا چند روش برای حل یک مساله بهینه سازی به شکلی که همکاری بین روشها نتیجه بهتر و سریعتری را عاید ما سازد همیشه مورد توجه بوده است. این ایده به خصوص با ورود روش های جدید به حیطه بهینه سازی تقویت شد. در الگوریتم های بهینه سازی همیشه باید به گونه ای بین جستجوی محلی و سراسری تعادل ایجاد نمود. چرا که جستجوی محلی بیشتر منجر به افتادن در دام بهینه های محلی و تمرکز بیشتر بر روی جستجوی سراسری باعث کاهش سرعت همگرایی و بعضا به دست نیامدن جواب قابل قبول می باشد. در اغلب الگوریتم های بهینه سازی می توان با تغییر در پارامتر های آن رفتار جستجو را تغییر داد اما ماهیت و خصوصیات کلی روش مانند قابلیت اکتشاف و بهره برداری در حالت کلی تقریبا ثابت است. با به کار بردن چند الگوریتم مختلف با خصوصیات اکتشاف و بهره برداری متفاوت بهتر می توان بین جستجوی محلی و سراسری تعادل برقرار نمود.

۱-۱ بهینه سازی[1]

تغییر و تحولات سریع در قرن حاضر و پیچیدگی محیط، اهمیت آشنایی با شیوه های صحیح تصمیم گیری را برای مدیران ضروری می سازد. این تصمیمات انواع و مقدار منابعی را که سازمان استفاده می کند، تعیین می نماید.

[1] optimization

بهینه سازی، پایه تجزیه و تحلیل بسیاری از سیستم های پیچیده و مسائل مهندسی است. بهینه سازی در واقع شامل حل یک مساله تصمیم گیری، شامل انتخاب مقادیر برای متغیر های مساله با در نظر گرفتن یک هدف[۲] است، که بنا به ماهیت مساله بیشینه[۳] یا کمینه[۴] می گردد. محدوده مسائلی که توسط بهینه سازی قابل حل هستند، بسیار وسیع است. افزایش سود یا کاهش زیان در یک سازمان یا شرکت، میزان بازگشت سرمایه در یک سیستم سرمایه گذاری اقتصادی و... همه از جمله مسائلی هستند، که در محدوده مسائل بهینه سازی قرار می گیرند.

۱-۲ انواع مسائل بهینه سازی

مسائل بهینه سازی با توجه به نوع فرمول بندی آنها، طبقه بندی می شوند. در حالت کلی یک مساله بهینه سازی را به صورت زیر می توان نمایش داد.

$$\underset{\vec{x}}{minimize(maximize)}\ \overrightarrow{Z} = (f_1(\vec{x}), f_2(\vec{x}), \dots, f_n(\vec{x})) \quad subject\ to\ \ \vec{x} \in X \qquad (۱-۱)$$

- $\overrightarrow{Z}$: بردار هدف
- $\vec{x}$: بردار متغیر های تصمیم گیری
- X : مجموعه جواب های شدنی در فضای تصمیم گیری

می توان همه مسائل بهینه سازی را بدون از دست دادن کلیت مساله، به مسائل کمینه سازی تبدیل کرد. شکل ساده تر مسائل بهینه سازی، مسائل تک هدفه[۵] می باشد. که در اینجا بحث ما در مورد این گونه مسائل می باشد.

[2] object
[3] maximize
[4] minimize
[5] Single objective

شکل یک مساله بهینه سازی تک هدفه، به صورت زیر می باشد.

$$\begin{aligned} &\underset{\vec{x}}{minimize} \quad Z = f(\vec{x}) \\ &subject\ \ to \quad \vec{x} \in X \end{aligned} \qquad (۲\text{-}۱)$$

X محدوده جواب های شدنی، که به وسیله تعدادی محدودیت[۶] (قیود) مساوی و نامساوی به شکل $h(\vec{x}) = 0\,, g(\vec{x}) \leq 0$ مشخص می گردد. بنابراین داریم:

$$\begin{aligned} &\underset{\vec{x}}{minimize} \quad Z = f(\vec{x}) \\ &subject\ \ to \quad g(\vec{x}) \leq 0 \\ &\qquad\qquad\ \ h(\vec{x}) = 0 \end{aligned} \qquad (۳\text{-}۱)$$

که:

$f(\vec{x})$ تابع هدف

$g(\vec{x}) = [g_1(\vec{x}), g_2(\vec{x}), \dots, g_k(\vec{x})]$ شامل k قید محدودیت نا مساوی[۷]

$h(\vec{x}) = [h_1(\vec{x}), h_2(\vec{x}), \dots, h_m(\vec{x})]$ شامل m قید محدودیت مساوی[۸]

در حالت کلی، هیچ نوع محدودیت خاصی برای توابع در نظر گرفته نمی شود و این توابع می‌توانند مشتق پذیر باشند یا نباشند، خطی یا غیر خطی باشند و یا ....

۱-۳ روش های حل مساله

روشهای حل مساله را می توان به دو گروه روش های سنتی و روش های مدرن طبقه بندی نمود. روش های سنتی به طور معمول دارای پایه ریاضی و جبری قوی تری می باشند. روش های

[6] constraint
[7] Inequality constraint
[8] Equality constraint

مدرن بر خلاف روشهای سنتی، عموماً به دنبال جواب دقیق نیستند و جواب بدست آمده اغلب دارای مقداری خطا می باشد.

۱-۴ روشهای هیبرید[9] برای حل مسائل بهینه سازی

به کار بردن دو یا چند روش، برای حل یک مساله، به شکلی که با همکاری بین روش ها جواب بهتری به دست آید، با ورود الگوریتم های مدرن به حیطه بهینه سازی تقویت شد. در این روش معمولا محورکار کار بر روی یکی از الگوریتم های جدید قرار می گیرد.

همکاری بین دو یا چند الگوریتم برای حل یک مساله، جنبه های گوناگونی دارد. یک روش ترکیبی[10] می تواند به صورت اعمال جند مرحله ای الگوریتم های گوناگون، بر روی فضای جواب[11] باشد. به طوری که در هر مرحله الگوریتم مورد نظر، بر روی جواب های حاصل از الگوریتم مرحله قبل کار می کند. و سعی در بهبود آن ها خواهد داشت. یا می تواند به صورت اعمال موازی چند الگوریتم، بر روی فضای جواب مورد نظر باشد. در این میان نکته حائز اهمیت، نحوه ارتباط میان این روش های حل مساله است. استفاده از قالب های گوناگون برای نمایش جواب ها در الگوریتم-های مختلف این ارتباط را تا حدودی مشکل می کند.

یکی از پر کاربرد ترین روشهای هیبریدی که تا کنون مورد استفاده قرار گرفته است، به صورت کاربرد همزمان چند الگوریتم مشابه، با پارامتر های متفاوت است. که همین تفاوت در پارامتر ها می‌تواند رفتار الگوریتم را تغییر دهد(Herrera 2000).

استفاده از الگوریتم های یکسان، کار تبادل و اشتراک اطلاعات مربوط به فضای جواب را بسیار ساده تر می کند. اما باید این نکته را مورد توجه قرار داد، که تغییر در مقادیر پارامتر های

[9] Hybrid methods
[10] combinatorial
[11] Solution space

یک الگوریتم، اگر چه رفتار جستجوی آن را می تواند تغییر دهد. اما نمی تواند ماهیت آن را عوض کند. بنابراین خصوصیات کلی روش، مانند قابلیت های اکتشاف[۱۲] و بهره برداری[۱۳] در حالت کلی، تقریبا ثابت است.

هدف کلی این رساله، یافتن راهی برای ترکیب چند الگوریتم بهینه سازی مختلف، برای استفاده از خصوصیات مثبت هر کدام و اجتناب از نقاط ضعف آن ها، با استفاده از کاربرد مکمل آن ها می باشد.

۱-۵ مشکلات حل مسائل بهینه سازی

در اغلب مسائل بهینه سازی، تعداد راه حل های شدنی[۱۴] مساله در فضای جستجو، آن قدر زیاد است، که بررسی همه آن راه حل ها و انتخاب بهترین راه حل ممکن نیست.

۱-۵-۱ انتخاب تابع ارزیابی

اغلب روشهای بهینه سازی سنتی، یک جواب ارضا کننده،[۱۵] برای حل مسائل برنامه ریزی غیرخطی ندارند.

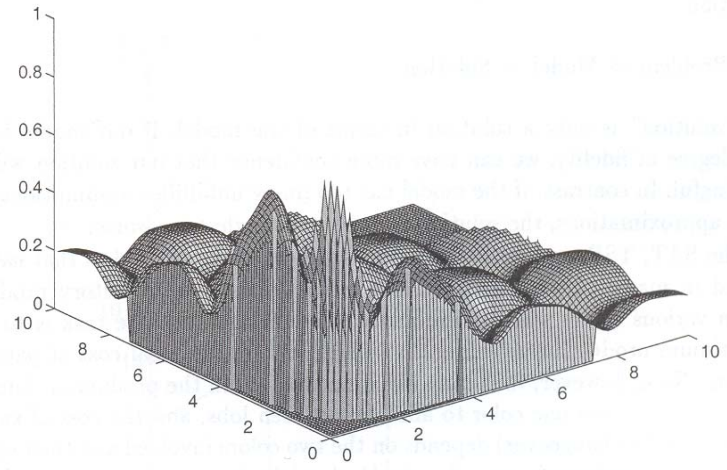
یک مساله بهینه سازی غیر خطی:

$$G2(\mathrm{x}) = \left| \frac{\sum_{i=1}^{n} \cos^4(x_i) - 2\prod_{i=1}^{n} \cos^2(x_i)}{\sqrt{\sum_{i=1}^{n} i x_i^2}} \right|, \qquad (۴-۱)$$

subject to

$\prod_{i=1}^{n} x_i \geq 0.75, \ \sum_{i=1}^{n} x_i \leq 7.5n$, and bounds $0 \leq x_i \leq 10$ for $1 \leq i \leq n$.

[12] Exploration
[13] Exploitation
[14] Feasible solution
[15] Satisfier



شکل ۱ یک تابع ارزیابی پیچیده

۱-۵-۲ اندازه فضای جستجو

در یک مساله بهینه سازی، اگر محدوده هر متغیر را بین 0 و10 قرار دهیم و حداکثر از 6 رقم اعشار برای نمایش اعداد استفاده شود. هر متغیر 10^7 مقدار متفاوت می تواند انتخاب نماید. در نتیجه برای یک مساله با n بعد، اندازه فضای جستجو برابر با 10^{7n} می شود. که این به مراتب از تعداد راه حل های مساله فروشنده دوره گرد بیشتر می باشد.

۱-۶ مفاهیم اساسی

سه مفهوم اساسی بین همه روشهای حل مساله مشترک می باشد. بدون توجه به روشی که برای حل مساله انتخاب می شود. همیشه تعریف این سه مفهوم، برای حل مساله، مورد نیاز می-باشد

- روش نمایش[۱۶]

[16] Representation