



# بخش اول

## پیشینه تحقیق

## ۱.۰ پیشینه تحقیق

### ۱.۱.۰ روش‌های ریاضی و فراابتکاری

دو روش کلی جهت بهینه‌سازی یک تابع وجود دارد؛ روش‌های برنامه‌نویسی ریاضی<sup>۱</sup> و روش‌های فراابتکاری<sup>۲</sup>. روش‌های برنامه‌نویسی ریاضی متنوعی از قبیل برنامه‌نویسی خطی<sup>۳</sup>، برنامه‌نویسی خطی همگن<sup>۴</sup>، برنامه‌نویسی صحیح<sup>۵</sup>، برنامه‌نویسی پویا<sup>۶</sup> و برنامه‌نویسی غیر خطی<sup>۷</sup> به منظور حل مسایل بهینه‌سازی بکار برده شده‌اند. این روش‌ها از اطلاعات مربوط به تغییر شیب<sup>۸</sup> در اطراف نقطه آغاز جستجو، در فضای جستجوی جواب استفاده می‌کنند. در عمل، روش‌های مبتنی بر شیب، سریع‌تر همگرا می‌شوند و همچنین می‌توانند جواب‌های با دقت بالاتری را نسبت به روش‌های تصادفی<sup>۹</sup> در هنگام جستجوی محلی<sup>۱۰</sup> بدست آورند. برای کاربرد مفید این روش‌ها، متغیرها و تابع هزینه باید پیوسته باشند. علاوه بر این، برای اجرای موفق این روش‌ها، یک نقطه آغازین خوب، خیلی مهم است. در بسیاری از مسایل بهینه‌سازی، نواحی بحرانی<sup>۱۱</sup>، حدود مرزی توابع هزینه غیر یکنواخت<sup>۱۲</sup> یا غیر محدب<sup>۱۳</sup> باید در نظر گرفته شوند. در نتیجه مسایل بهینه‌سازی غیر محدب که فاقد شیب کافی در نواحی بحرانی می‌باشند، با روش‌های برنامه‌نویسی ریاضی مرسوم قابل حل نیستند. اگرچه برنامه‌نویسی پویا یا برنامه‌نویسی صحیح غیرخطی ترکیبی و اصلاحات آن، ترفندهایی را برای حل مسایل غیر محدب پیشنهاد می‌کنند، ولی در حالت کلی، این روش‌ها نیازمند کارهای محاسباتی قابل توجه است. در مقابل روش‌های ریاضیاتی مرسوم، تکنیک‌های بهینه‌سازی فراابتکاری به منظور بدست آوردن جواب‌های بهینه کلی یا نزدیک به کلی<sup>۱۴</sup> استفاده می‌شوند. این روش‌ها به دلیل داشتن قابلیت‌های اکتشافی و پیدا کردن نواحی محتمل در فضای جستجو در مدت زمان ممکن، به منظور جستجوهای کلی و همچنین کاهش نیاز به متغیرها و توابع هزینه پیوسته، کاملاً مناسب می‌باشند. اگرچه این روش‌ها، روش‌های تقریبی هستند، بدین معنی که جواب‌های آن‌ها خوب اما ضرورتاً بهینه نیستند، لیکن به مشتق تابع هدف و قیدها نیاز ندارند و قواعد انتقالی احتمالاتی را به جای قوانین معین و قطعی بکار می‌برند. طبیعت همواره به عنوان یک مرجع مهم جهت الهام گرفتن، برای مهندسين مطرح بوده است، و فلسفه‌های طبیعی و روش‌های فراابتکاری زیادی از روش‌هایی که به نظر می‌رسد خود طبیعت برای حل مسایل سخت انتخاب می‌کند، الهام گرفته شده‌اند.

<sup>۱</sup> Mathematical Programming

<sup>۲</sup> Meta heuristic Methods

<sup>۳</sup> Linear Programming

<sup>۴</sup> Homogenous Linear Programmin

<sup>۵</sup> Integer Programming

<sup>۶</sup> Dynamic Programming

<sup>۷</sup> Nonlinear Programming

<sup>۸</sup> Gradient information

<sup>۹</sup> Stochastic Approach

<sup>۱۰</sup> Local Search

<sup>۱۱</sup> Prohibited Zones

<sup>۱۲</sup> Non Smooth

<sup>۱۳</sup> Non Convex

<sup>۱۴</sup> Near Global Optimum Solution

الگوریتم تدریجی<sup>۱</sup> (EA) که توسط فوگل<sup>۲</sup> و همکاران[۳]، دی جانگ<sup>۳</sup>[۴] و کوزا<sup>۴</sup>[۵] معرفی شد و الگوریتم ژنتیک<sup>۵</sup> (GA) که توسط هالند[۶] و گولد برگ<sup>۷</sup>[۸] پیشنهاد شد، از فرایندهای تدریجی زیستی الهام گرفته شده‌اند. مطالعات بر روی رفتار و نحوه زندگی حیوانات، منجر به کشف روش‌های بهینه‌سازی دیگری شده که عبارتند از: روش جستجوی تابو<sup>۸</sup> (TS) که توسط گلوور[۹] ارائه شد، بهینه‌سازی بر اساس کلونی مورچه‌ها<sup>۱۰</sup> (ACO) که توسط دوریگو<sup>۱۱</sup> و همکاران[۱۲] عرضه شد، بهینه‌سازی بر اساس دسته حیوانات<sup>۱۲</sup> (PSO) که بوسیله ابره‌ارت<sup>۱۳</sup> و کندی<sup>۱۴</sup>[۱۵] پیشنهاد شد. شبیه‌سازی گرم و سرد کردن<sup>۱۵</sup> (SA) فلزات معرفی شده توسط کرکپاتریک<sup>۱۶</sup> و همکاران[۱۷]، الگوریتم جستجوی گرانشی<sup>۱۷</sup> (GSA) ارئه شده توسط راشدی[۱۸]، همگی از پدیده‌های فیزیکی الهام گرفته شده‌اند.

---

<sup>۱</sup> Evolutionary Algorithm

<sup>۲</sup> Fogel

<sup>۳</sup> De Jong

<sup>۴</sup> Koza

<sup>۵</sup> Genetic Algorithm

<sup>۶</sup> Holland

<sup>۷</sup> Goldberg

<sup>۸</sup> Tabu Search

<sup>۹</sup> Glover

<sup>۱۰</sup> Ant Colony Optimization

<sup>۱۱</sup> Dorigo

<sup>۱۲</sup> Particle Swarm Optimizer

<sup>۱۳</sup> Eberhart

<sup>۱۴</sup> Kennedy

<sup>۱۵</sup> Simulated Annealing

<sup>۱۶</sup> Kirkpatrick

<sup>۱۷</sup> Gravitational Search Algorithm

<sup>۱۸</sup> Rashedi