بسمه تعالی

فصل اول

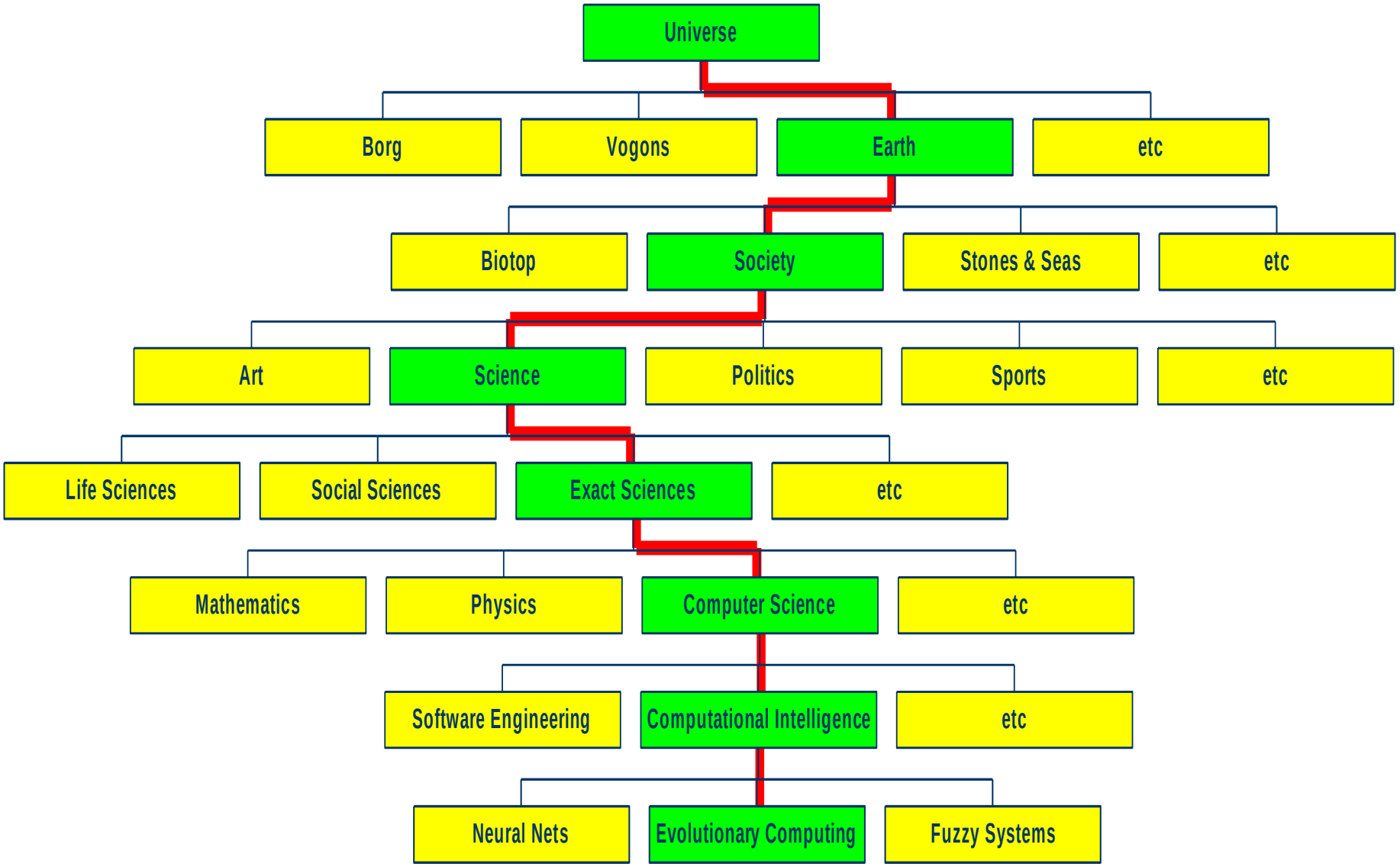
مقدمه و مروری بر پردازش تکاملی

ارائه دهنده : محمد فرجی مهماندار

جایگاه پردازش تکاملی (EC)

- EC بخشی از علوم کامپیوترمی باشد (و نه علوم زیستی)
- EC نوعی از محاسبات می باشد که از فرآیند تکامل طبیعی الهام می گیرد.
- EC می تواند در تحقیقات بیولوژیکی بکار گرفته شود.
- یکی از روش های حل مسائل بهینه سازی است.

جایگاه پردازش تکاملی (EC)



جایگاه پردازش تکاملی (EC)

EVOLUTION		PROBLEM SOLVING
(تکامل)		(حل مسئله با سعی و خطا)
Environment(محیط)	⟷	Problem(مسئله)
Individual(فرد)	⟷	Candidate Solution(راه حل کاندید)
Fitness(برازندگی)	⟷	Quality(کیفیت)

Fitness → chances for survival and reproduction

شانس برای بقاء و تکثیر→ برازندگی

Quality → chance for seeding new solutions

شانس برای ایجاد راه حل های جدید → کیفیت

انگیزه استفاده از پردازش تکاملی (EC)

- همواره طبیعت به عنوان منبع الهام برای مهندسین و دانشمندان بکار رفته است.
- بهترین حل کننده های مسائل شناخته شده در طبیعت
  - مغز (انسان): که "چرخ، شهر نیویورک، جنگ و ... " را ایجاد کرده است
  - مکانیزم تکامل: که مغز انسان را ایجاد نموده است

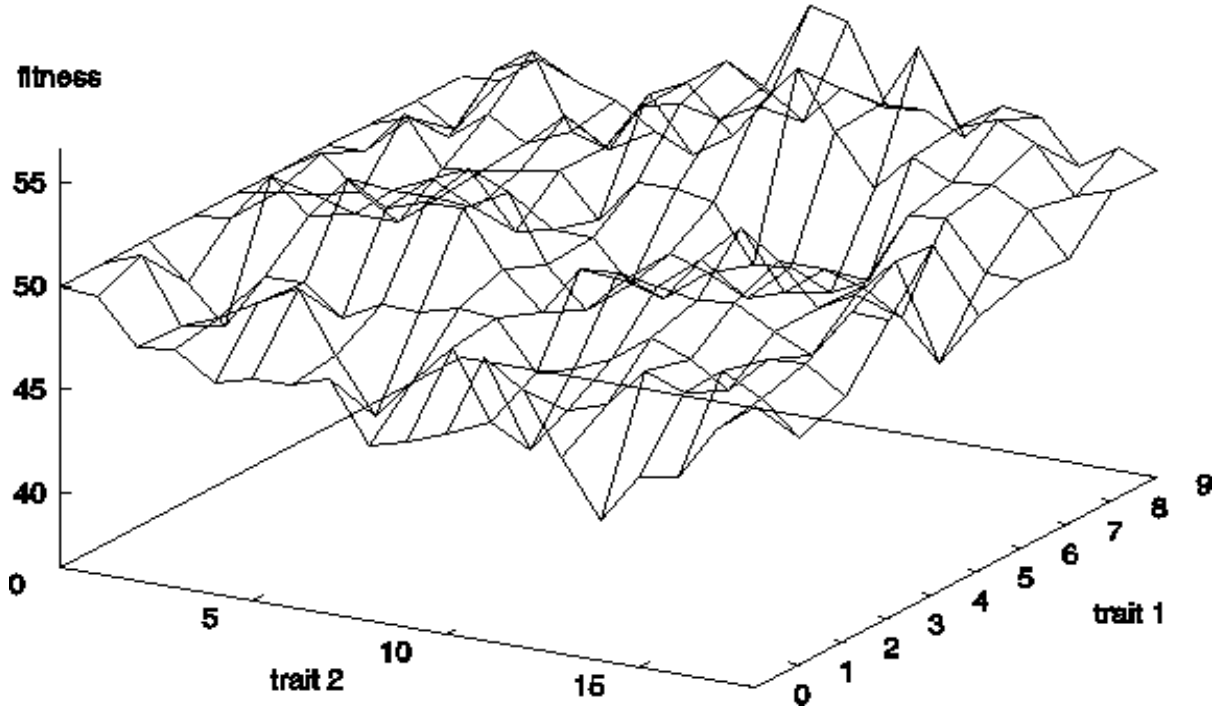
پاسخ ۱: neurocomputing

پاسخ ۲: پردازش تکاملی

- توسعه، تحلیل و اعمال روش های حل مسأله (الگوریتم ها) یک موضوع مرکزی در ریاضیات و علوم کامپیوتری می باشد
- زمان برای تحلیل دقیق و کامل مسأله کاهش می یا بد
- پیچیدگی مسائلی که باید حل شوند افزایش می یابد
- نتیجه :
  - نیاز به یک تکنولوژی قوی برای حل مسأله

سطح تطبیق پذیر (Adaptive surface)

- می توان یک جمعیت با n صفت مختلف را در یک فضای n+1بعدی (سطح) تصور کرد که در آن ارتفاع متناظر با برازندگی می باشد.
- هر فرد مختلف (فنوتایپ) در جمعیت، بیانگر یک نقطه روی این سطح می باشد.
- بنابراین جمعیت به صورت "ابری" از نقاط می باشد که در طول زمان همین طور که تکامل می یابد بر روی این سطح حرکت می کند – سازگاری (adaptation)



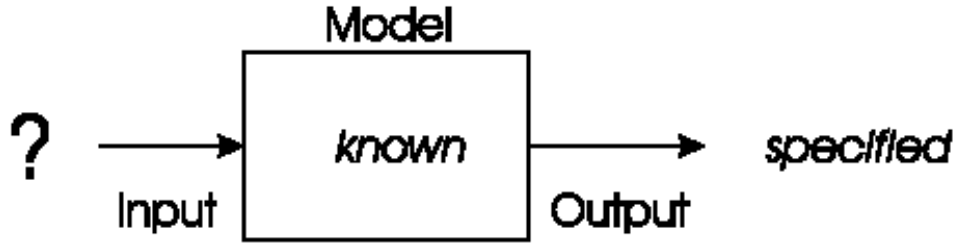
سطح تطبیق پذیر (Adaptive surface)

- یک جمعیت را می توان به صورت مجموعه ای از نقاط برروی این سطح ترسیم نمود
- هرنقطه معرف یک فرد می باشد که بیانگریک ترکیب ممکن از صفات می باشد
- بنابراین تکامل، فرآیند پیشرفت های تدریجی جمعیت به سمت نواحی مرتفع می باشد
- بهینه محلی: نقاطی برروی سطح که از همه نقاط همسایه بهترند (ارتفاع بیشتری دارند)
- بهینه سراسری: مرتفع ترین نقطه برروی سطح
- مسأله ای که در آن تنها یک نقطه (راه حل) وجود دارد که از تمام همسایگانش بهتراست (unimodal)
- مسائلی که در آنها چندین نقطه وجود دارند که از تمامی همسایگانشان بهترند (Multimodal)
- انتخاب، جمعیت را به سمت بالای سطح هل می دهد
- تغییرات تصادفی در توزیع ویژگی ها می تواند باعث شود که جمعیت به سمت پا ییین حرکت کند و بنا برا ین از بهینه های محلی فرار کند.

انواع مسائل بهینه سازی

- بهینه سازی یک فعالیت مهم و تعیین کننده در طراحی ساختاری است. طراحان زمانی قادر خواهند بود طرحهای بهتری تولید کنند که بتوانند با روشهای بهینه سازی در صرف زمان و هزینه طراحی صرفه جویی نمایند.
- بسیاری از مسایل بهینه سازی در مهندسی، طبیعتاً پیچیده تر و مشکل تر از آن هستند که با روشهای مرسوم بهینه سازی نظیر روش برنامه ریزی ریاضی و نظایر آن قابل حل باشند.
- دسته مهمی از مسایل بهینه سازی که کاربرد فراوانی دارند، بهینه سازی ترکیبی می باشند که هدف در آنها جستجو برای یافتن نقطه بهینه توابع با متغیرهای گسسته می باشد.
- با توجه به مشکالت مربوط به روش شمارش کامل، همواره بر ایجاد روش های مؤثرتر و کاراتر تأکید شده است
- مدلی از سیستم خود داریم و به دنبال ورودی هایی هستیم که به ما یک پاسخ مشخص شده می دهند

مثال : جداول زمانی برای دانشگاه یا بیمارستان



مثال بهینه سازی :جدول زمانی دانشگاه دانشگاه

- فضای جستجوی بسیار بزرگ می باشد .
- جداول زمانی باید خوب باشند .
- " خوب" بوسیله تعداد معیارهای رقابت کننده تعریف می شود:
  - دانشجویان ترجیح می دهند که در یک روز بیش از ۲ کلاس نداشته باشند
  - اساتید میل دارند که برای انجام تحقیقات بیشتر روزهایشان آزاد باشد
  - مدیریت دانشگاه می خو اهد از فضا ی موجود حداکثر استفاده را انجام دهد و یا جابجایی ها میان ساختمان های مختلف را کاهش دهد.

• جداول زمانی باید امکان پذیرباشند

  - یک دانشجو نمی تواند در یک زمان در دو کلاس حضور داشته باشد
  - در یک کلاس نمی توان به طور همزمان دو درس را ارائه نمود

• قسمت بسیار بزرگی از فضای جستجو امکان پذیرنمی باشد

انواع مسائل بهینه سازی

هدف از بهینه سازی یافتن بهترین جواب قابل قبول، با توجه به محدودیتها و نیازهای مسأله است

دسته بندی روش‌ها و الگوریتم های بهینه سازی:

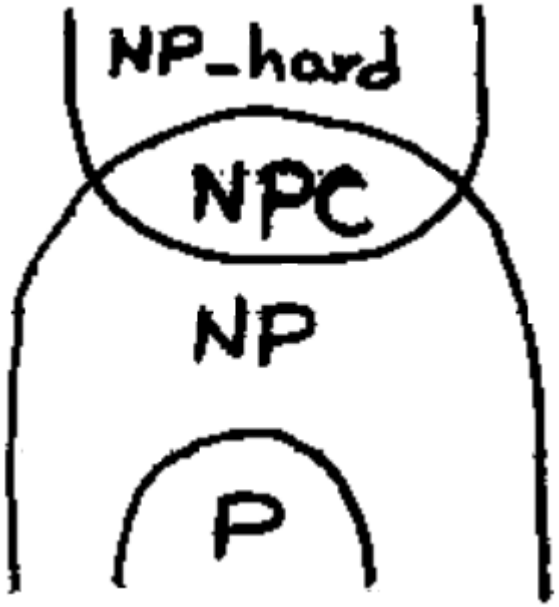
- الگوریتمهای دقیق (Exact)

الگوریتم‌های دقیق، قادر به یافتن جواب بهینه به‌صورت دقیق هستند، اما در مورد مسائل بهینه‌سازی سخت، کارایی کافی ندارند و زمان اجرای آن‌ها متناسب با ابعاد مسائل به‌صورت نمایی افزایش می‌یابد.

- الگوریتم‌های تقریبی (Approximate)

الگوریتم‌های تقریبی قادر به یافتن جواب‌های خوب و نزدیک به بهینه در زمان حل کوتاه برای مسائل بهینه‌سازی سخت هستند.

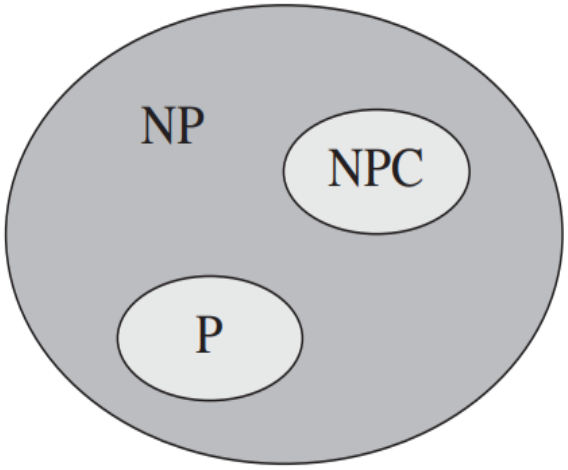
الگوریتم های دقیق



نظریه پیچیدگی محاسباتی (Computatioan Complexity):

- کلاس P (Polynomial)
- کلاس NP (Nondeterministic Polynomial time)
- کلاس NPC (NP-Complete)
- کلاس NP-Hard

روش های مختلف الگوریتم نویسی :



- **برنامه نویسی پویا**
- **روش حریصانه در طراحی الگوریتم**
- **راهبرد عقبگرد**
- **راهبرد شاخه و حد**

برنامه نویسی پویا و مسائل بهینه سازی

- ارائه یک ویژگی بازگشتی که حل بهینه ی نمونه ای از مسئله را به دست می دهد.
- محاسبه مقدار حل بهینه به شیوه ی جزء به کل.
- بنا کردن یک حل نمونه به شیوه ی جزء به کل.
- هر مسئله بهینه سازی را نمی توان با استفاده از برنامه نویسی پویا حل کرد چرا که باید اصل بهینگی در مسئله صدق کند. اصل بهینگی در یک مسئله صدق می کند اگریک حل بهینه برای نمونه ای از مسئله ، همواره حاوی حل بهینه برای همه ی زیر نمونه ها باشد

روش حریصانه در طراحی الگوریتم :

- الگوریتم حریصانه ، به ترتیب عناصر را گرفته، هر بار آن عنصری را که طبق ملاکی معین "بهترین" به نظر می رسد، بدون توجه به انتخاب هایی که قبلا انجام داده یا در آینده انجام خواهد داد، بر می دارد.
- الگوریتم حریصانه ، همانند برنامه نویسی پویا غالبا برای حل مسائل بهینه سازی به کار می روند، ولی روش حریصانه صراحت بیشتری دارد.
- در روش حریصانه ، تقسیم به نمونه های کوچک تر صورت نمی پذیرد.
- الگوریتم حریصانه با انجام یک سری انتخاب، که هر یک در لحظه ای خاص ،بهترین به نظر می رسد عمل می کند، یعنی انتخاب در جای خود بهینه است.امید این است که یک حل بهینه سرتاسری یافت شود، ولی همواره چنین نیست.
- برای یک الگوریتم مفروض باید تعیین کرد که آیا حل همواره بهینه است یا خیر.

راهبرد عقبگرد:

- از تکنیک عقبگرد برای حل مسائلی استفاده می شود که در آن ها دنباله ای از اشیاء از یک مجموعه مشخص انتخاب می شود، به طوری که این دنباله ، ملاکی را در بر می گیرد.
- یک مثال کلاسیک از عقبگرد، مسئله n وزیر است. هدف از مسئله n وزیر ، چیدن n مهره وزیر در یک صفحه شطرنج است ، به طوری که هیچ دو وزیری یکدیگر را گارد ندهند. یعنی هیچ دو مهره ای نباید در یک سطر، ستون یا قطر یکسان باشند.
- عقبگرد حالت اصلاح شده ی جست و جوی عمقی یک درخت است.
- الگوریتم عقبگرد همانند جست و جوی عمقی است، با این تفاوت که فرزندان یک گره فقط هنگامی ملاقات می شوند که گره امید بخش باشدو در آن گره حلی وجود نداشته باشد.

راهبرد شاخه و حد:

- راهبرد شاخه و حد ازآن لحاظ به عقبگرد شبا هت دارد که درآن، برای حل مسئله از درخت فضای حالت استفاده می شود.
- تفاوت این روش با عقبگرد، اولا ما را به پیمایش خاصی ازدرخت محدود نمی کندوثانیا فقط برای مسائل بهینه سازی به کار می رود.
- الگوریتم شاخه و حد ، در هر گره عددی را (حدی) رامحاسبه می کند تا تعیین شود که آیا این گره امید بخش هست یا خیر.
- اگر آن حد بهتر از مقدار بهترین حلی که تاکنون یافته شده، نباشد، گره غیر امید بخش است. در غیر این صورت، امید بخش است.

الگوریتم های تقریبی

دسته بندی الگوریتم‌های تقریبی:

- الگوریتم‌های ابتکاری (Heuristic)

دو مشکل اصلی الگوریتم‌های ابتکاری، گیر افتادن آن‌ها در نقاط بهینه محلی(Local optimum trap) و همگرایی زودرس به این نقاط است.

- فراابتکاری(Meta-heuristic)

الگوریتم‌های فراابتکاری برای حل مشکلات الگوریتم‌های ابتکاری ارائه شده‌اند. در واقع الگوریتم‌های فراابتکاری، یکی از انواع الگوریتم‌های بهینه‌سازی تقریبی هستند که دارای راهکارهای برون‌رفت از نقاط بهینه محلی هستند و قابلیت کاربرد در طیف گسترده‌ای از مسائل را دارند. رده-های گوناگونی از این نوع الگوریتم در دهه‌های اخیر توسعه‌یافته است .

الگوریتم های فرا ابتکاری

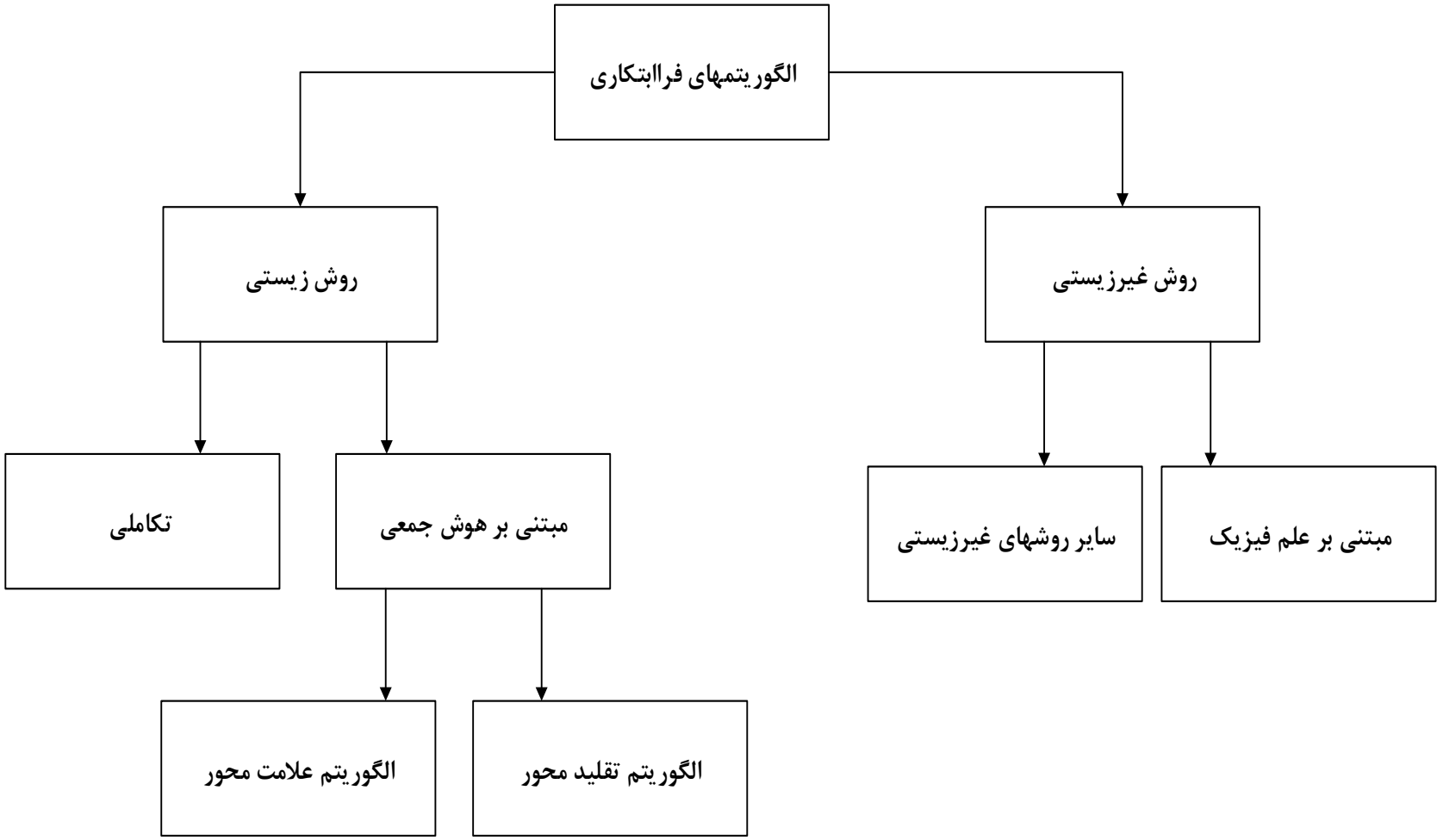
- الگوریتم های فراابتکاری به عنوان هوشمندانه ترین روش های جستجو، از یک مکاشفه طبیعی در عملیات پیمایشی خود در فضای نامنظم مسئله بهره می-برند.
- مهم‌ترین دسته الگوریتم های جستجو، الگوریتم های فراابتکاری زیستی هستند.
- دسته بندی االگوریتم های فرا ابتکاری زیستی :
  - الگوریتم های تکاملی

    ایده اصلی روش-های فراابتکاری تکاملی که سرمنشأ خلق بسیاری از روش-های زیستی هستند، مبتنی بر نظریه تکاملی داروین است(الگوریتم ژنتیک).
  - مبتنی بر هوش جمعی

    -الگوریتم های تقلید محور (الگوریتم تراکم ذرات)

    -الگوریتم های علامت محور (الگوریتم مورچه)

الگوریتم های فرا ابتکاری



الگوریتم های فرا ابتکاری

معیارهای طبقه‌بندی الگوریتم‌های فراابتکاری :

- **مبتنی بر یک جواب و مبتنی بر جمعیت:** الگوریتم‌های مبتنی بر یک جواب در حین فرایند جستجو یک جواب را در نظر گرفته و تغییر می-دهند، درحالی‌که الگوریتم‌های مبتنی بر جمعیت در حین جستجو، یک جمعیت از جواب‌ها را در نظر می-گیرند.
- **زیستی و غیرزیستی:** بسیاری از الگوریتم‌های فراابتکاری از طبیعت الهام گرفته شده‌اند و به‌عبارت‌دیگر زیستی هستند. در این میان برخی از الگوریتم‌های فراابتکاری نیز از طبیعت الهام گرفته نشده‌اند.
- **با حافظه و بدون حافظه:** برخی از الگوریتم‌های فراابتکاری فاقد حافظه می‌باشند، به این معنا که، این نوع الگوریتم‌ها از اطلاعات به‌دست‌آمده در حین جستجو استفاده نمی‌کنند (به طور مثال تبرید شبیه‌سازی شده). این در حالی است که در برخی از الگوریتم‌های فراابتکاری نظیر جستجوی ممنوعه از حافظه استفاده می‌شود. این حافظه اطلاعات به‌دست‌آمده در حین جستجو را در خود ذخیره می‌کند.
- **قطعی و احتمالی:** یک الگوریتم فراابتکاری قطعی نظیر جستجوی ممنوعه، مسئله را با استفاده از تصمیمات قطعی حل می‌کند. اما در الگوریتم‌های فراابتکاری احتمالی نظیر تبرید شبیه سازی شده، یک سری قوانین احتمالی در حین جستجو مورد استفاده قرار می‌گیرد.

نمونه مسائل بهینه سازی

مسئله ۱: مسئله n وزیر

هدف از مسئله n وزیر، چیدن n مسئله: مهره وزیر در یک صفحه شطرنج است، به طوری که هیچ دو وزیری یکدیگر را گارد ندهند. یعنی هیچ دو مهره ای نباید در یک سطر، ستون یا قطر یکسان باشند.

مسئله۲: فروشنده دوره گرد

هدف از این مسئله یافتن کوتاهترین مسیر در یک گراف جهت دار با شروع از یک راس مفروض است، مشروط بر آن که هر راس فقط یک بار ملاقات شود. چنین مسیری را یک تور می گویند

نمونه مسائل بهینه سازی

مسئله۳ : کوله پشتی صفر و یک

در این مسئله مجمو عه ای از قطعات داریم که هر یک دارای وزن و ارزش معین است.

اوزان و ارزش ها اعداد مثبتی هستند.

دزدی درنظر دارد قطعاتی که می دزدد درون یک کوله پشتی قرار دهد و اگر وزن کل قطعات قرار داده شده در آن کوله پشتی از یک عدد صحیح مثبتW فراتر رود، کوله پشتی پاره می شود.

مسئله٤: زمان بندی با مهلت معین

مسئله : تعیین زمان بندی با سود کل بیشینه ، با این فرض که هر کاری دارای سود است و فقط وقتی قابل حصول است که آن کار در مهلت مقررش انجام شود.

مسئله ۵: رنگ آمیزی گراف

مسئله رنگ آمیزی m ، عبارت از یافتن همه ی راه ها ی ممکن برای رنگ آمیزی یک گراف بدون جهت، با استفاده از حداکثر m رنگ متفاوت است، به طوری که هیچ دو راس مجاوری هم رنگ نباشند.