بسمه تعالی

فصل دوم

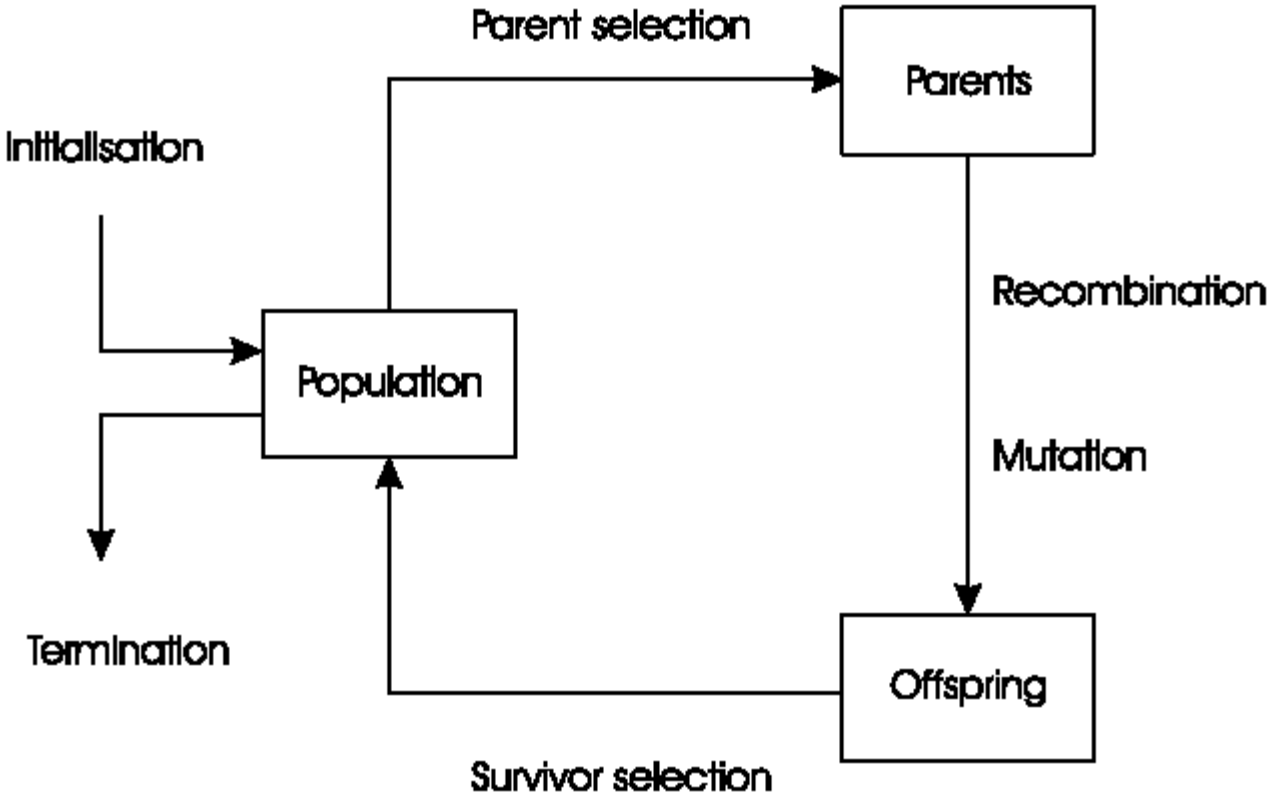
مفاهیم اولیه پردازش تکاملی

ارائه دهنده : محمد فرجی مهماندار

یک الگوریتم تکاملی چیست؟ ایده اصلی

- یک محیط از جمعیتی از افراد با منابع محدود تشکیل می شود
- رقابت برای این منابع باعث انتخاب افرادی می شودکه بهتر از بقیه با محیط تطبیق یافته اند
- این افراد به عنوان والد به منظور ایجاد افراد جدید از طریق آمیزش و جهش عمل می کنند
- برازندگی افراد جدید ارزیابی می شود و سپس این افراد جدید به منظور بقاء به رقابت می پردازند
- در طول زمان انتخاب طبیعی باعث افزایش برازندگی جمعیت می شود
- الگوریتم های تکاملی در دسته الگوریتم های «تولید و تست» قرار می گیرند
- ویژگی های الگوریتم های تکاملی
  - الگوریتم های تکاملی مبتنی بر جمعیت می باشند: مجموعه ای از راه حل های کاندیدا را به طور همزمان پردازش می کنند.
  - الگوریتم های تکاملی اغلب از عملگر آمیزش به منظور ترکیب اطلاعات موجود در چندین راه حل در یک راه حل جدید استفاده می کنند
  - الگوریتم های تکاملی اتفاقی (غیرقطعی) می باشند.

شمای کلی الگوریتم های تکاملی



شبه کد یک الگوریتم تکاملی

```
BEGIN
  INITIALISE population with random candidate solutions;
  EVALUATE each candidate;
  REPEAT UNTIL ( TERMINATION CONDITION is satisfied ) DO
    1 SELECT parents;
    2 RECOMBINE pairs of parents;
    3 MUTATE the resulting offspring;
    4 EVALUATE new candidates;
    5 SELECT individuals for the next generation;
  OD
END
```

مولفه های یک الگوریتم تکاملی

❑مولفه های یک الگوریتم الگوریتم تکاملی:

به منظور تعریف یک الگوریتم تکاملی خاص باید مؤلفه های زیر مشخص شوند

- ❖ بازنمایی (نحوه تعریف افراد)
- ❖ تابع ارزیابی (یا تابع برازندگی)
- ❖ جمعیت
- ❖ مکانیزم انتخاب والد
- ❖ عملگرها ی ژنتیکی (آمیزش و جهش)
- ❖مکانیزم انتخاب بازمانده (استراتژی جایگزینی)

علاوه بر این موارد، باید نحوه تولید جمعیت اولیه و شرط (شرایط) توقف الگوریتم نیزمشخص شوند

مولفه های الگوریتم تکاملی: روش های بازنمایی (تعریف افراد)

- راه حل های کاندیدا (افراد) در فضای فنوتایپ قرار دارند
- این راه حل ها به صورت کروموزوم ها کد می شوند که در فضای ژنوتایپ قرار دارند

Encoding : phenotype=> genotype (لزوما یک به یک نمی باشد)

Decoding : genotype=> phenotype (باید یک یه یک باشد)

- کروموزم ها حاوی ژن ها می باشند
  - محل قرار گرفتن ژن در کروموزوم: Locus
  - مقدار ژن: Allele

برای یافتن بهینه سراسری، هر راه حل امکان پذیر باید بتواند در فضای ژنوتایپ بازنمایی شود

مولفه های الگوریتم تکاملی: جمعیت

- نقش جمعیت، نگهداری راه حل های ممکن برای مسأله می باشد
- معمولاً دارای اندازه ثابتی می باشد و یک چندمجموعه از ژنو تا یپ ها می باشد
- معمولاً عملگرهای انتخاب کل جمعیت را در نظر می گیرند، یعنی احتمال تکثیر هر کروموزوم نسبت به نسل فعلی محاسبه می شود
- تعداد برازندگی ها/ فنوتایپ ها/ ژنوتایپ های متفاوت در جمعیت معرف تنوع آن جمعیت می باشد.

مولفه های الگوریتم تکاملی: تابع ارزیابی (برازندگی)

- بیانگرنیازمندی هایی است که جمعیت باید با آن ها تطبیق یابد
- نوعی تابع کیفیت یا تابع هدف
- به هر فنوتایپ یک مقدار برازندگی (عدد حقیقی) نسبت می دهد و اساس انتخاب را تشکیل می دهد
  - بنابراین هرقدر مقادیرمتفاوت تری را نسبت دهد، بهتر است
- نوعاً برازندگی کمیتی است که باید بیشینه شود.
  - تبدیل یک مسأله کمینه سازی به یک مسأله بیشینه سازی و (برعکس) کار ساده ای می باشد.

مولفه های الگوریتم تکاملی: مکانیزم انتخاب

- افراد را بر اساس مقدار برازندگی آنها به عنوان والد برای تولید نسل انتخاب می کند

معمولا احتمالاتی می باشد

  - راه حل هایی که دارای کیفیت بالاتری هستند نسبت به راه حل هایی که دارای کیفیت پایین تری می باشند، شانس بیشتری برای انتخاب شدن دارند
  - اما چنین چیزی تضمین شده نیست
  - حتی بدترین فرد حاضردر جمعیت نیز شانس انتخاب شدن دارد

- همین طبیعت غیر قطعی به فرار از بهینه های محلی کمک کند.

مولفه های الگوریتم تکاملی: عملگرهای ژنتیکی

- نقش آنها تولید راه حل های کاندیدای جدید می باشد
- این عملگرها معمولا بر اساس چندی (تعداد ورودی) در دو دسته قرارمی گیرند:
  - چندی = ۱ :عملگرهای جهش
  - چندی>1 : عملگرها ی آمیزش
  - چندی = ۲ :عملگر crossover

بحث های بسیاری در مورد اهمیت نسبی عملگرهای ژنتیکی (آمیزش و جهش) وجود دارد

- امروزه اغلب الگوریتم های تکاملی از هردو استفاده می کنند
- انتخاب عملگرهای ژنتیکی خاص، وابسته به نحوه بازنمایی می باشد

مولفه های الگوریتم تکاملی: عملگر جهش و عملگر آمیزش

عملگر جهش

- برروی یک ژنوتایپ عمل می کند و یک ژنوتایپ جدید ایجاد می کند
- در جهش، عنصر تصادفی بودن یک عنصر ضروری می باشد و باعث تمایز آن از دیگرعملگرهای هیوریستیک می شود
- اهمیت آن بستگی به روش بازنمایی و نوع الگوریتم تکاملی دارد
- عملگر جهش می تواند تضمین کننده پیوستگی فضای جستجو باشد(اثبات همگرایی)

عملگر آمیزش

- اطلاعات والدین را در فرزندان ادغام می کند
- انتخاب اینکه چه اطلاعاتی باید ادغام شوند، اتفاقی می باشد
- اغلب فرزندان ممکن است بدترو یا مانند والدینشان باشند
- این عملگر با این امید انجام می شود که برخی از فرزندان با ترکیب عناصر ژنوتایپ ها که منجر به ایجاد ترکیب های بهتری از ویژگی ها می شوند، از والدینشان بهترباشند
- این اصل هزاران سال توسط پرورش دهندگان گیاهان و حیوانات استفاده شده است

مولفه های الگوریتم تکاملی: جایگزینی و توقف

جایگزینی

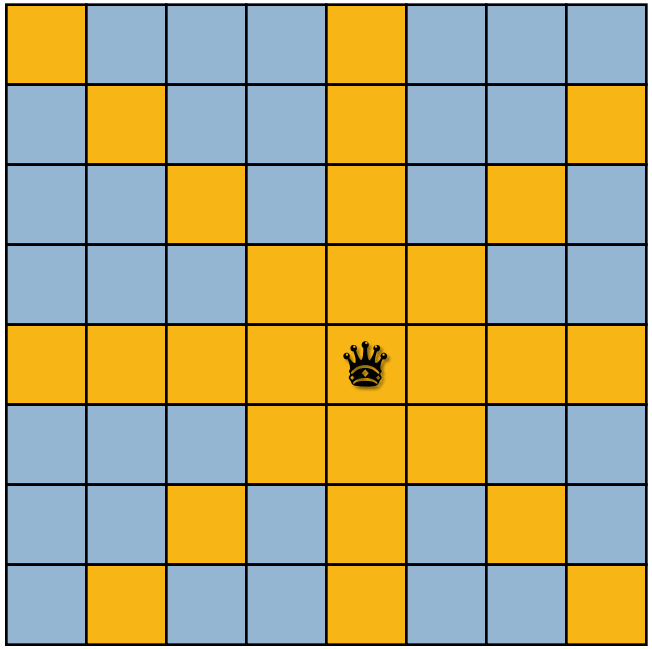
- اغلب الگوریتم های تکاملی از یک اندازه ثابت برای جمعیت استفاده می کنندو بنابراین به روشی برای رفتن به نسل بعدی (با انتخاب افراد از میان والدین و فرزندان) نیاز دارند
- اغلب قطعی می باشد:
  - مبتنی بر برازندگی : مثلاً رتبه بندی والدین+ فرزندان براساس برازندگی ها و انتخاب بهترین ها
  - مبتنی بر سن: به تعداد والدین فرزند ایجاد می شود و سپس تمامی والدین حذف می شوند
  - برخی مواقع به صورت ترکیبی (elitism)

مقدار دهی جمعیت اولیه و توقف

- مقدار دهی جمعیت اولیه معمولاً به صورت تصادفی می باشد
- می تواند شا مل راه حل های موجود باشد و یا از هیوریستیک های خاص مسأله برای ایجاد جمعیت اولیه استفاده کند
- شرایط توقف در هرنسل بررسی می شود:
  - رسیدن به یک مقدار خاص از برازندگی
  - رسیدن به یک حداکثر تعداد مجاز از نسل ها
  - رسیدن به یک سطح حداقل از نظرتنوع
  - رسیدن به یک تعداد مشخص از نسل های متوالی بدون بهبود برازندگی

مثال: مسأله ۸–وزیر

هشت وزیر را بر روی یک صفحه شطرنج هشت در هشت قرار دهید به طوری که هیچ دو وزیری نتوانند یکدیگر را تهدید کنند.



مسأله ۸–وزیر: بازنمایی

Phenotype:
a board configuration

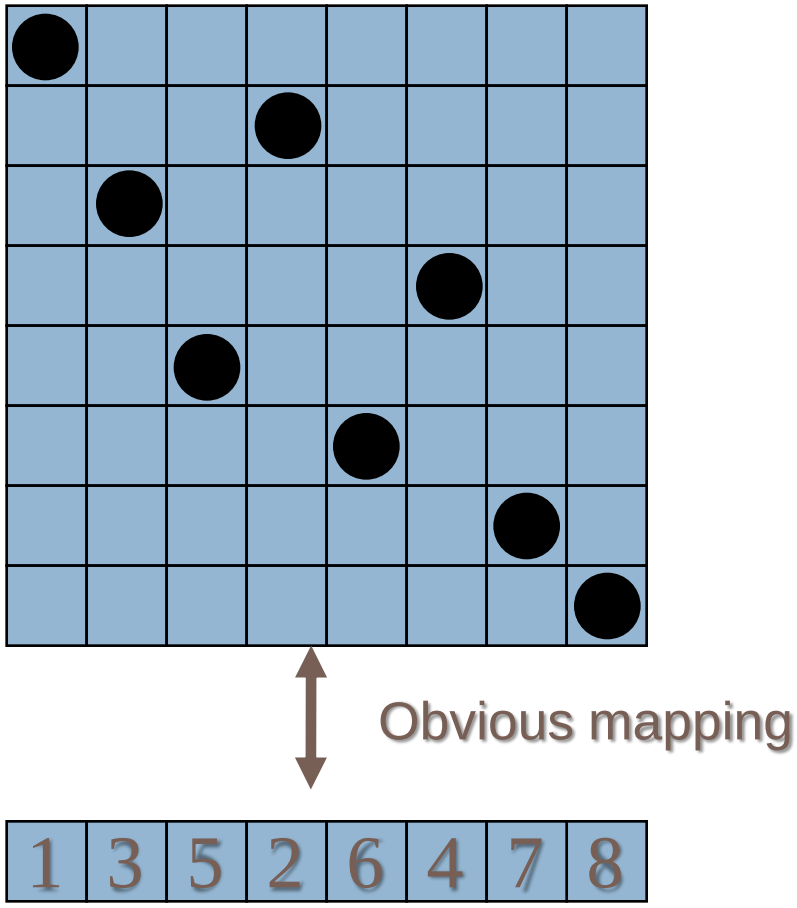
فنوتایپ :

یک پیکره بندی از صفحه

ژنوتایپ :

یک جایگشت از اعداد ۱ تا ۸

Genotype:
a permutation of the numbers 1 - 8



مسأله هشت وزیر: ارزیابی برازندگی

- محاسبه خطا برای یک وزیر
  - تعداد وزیرهای تحت حمله توسط آن وزیر
- محاسبه خطا برای یک پیکره بندی
  - مجموع خطای تمامی وزیرها
- توجه: خطا باید کمینه شود
- محاسبه برازندگی یک پیکره بندی
  - بیشینه سازی عکس خطا

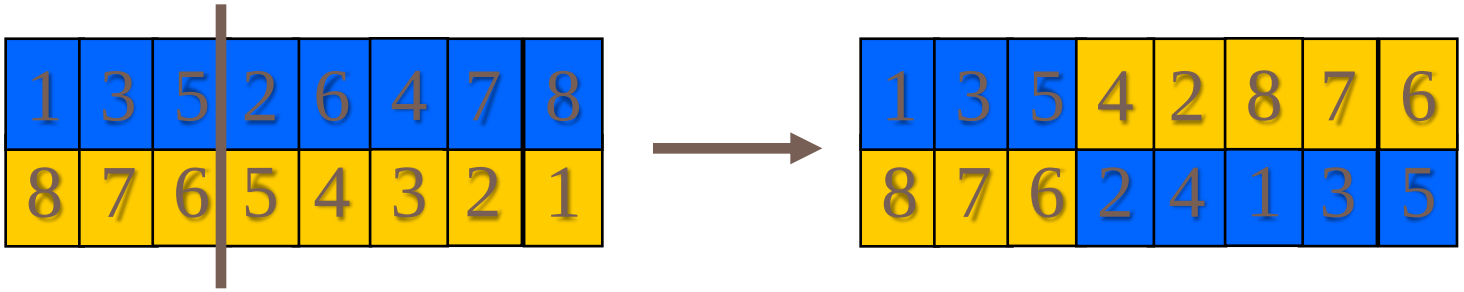
مسأله ۸–وزیر: جهش

تغییر کوچک در یک جایگشت

–مثلاً: جابجا نمودن مقادیردو مکان به صورت تصادفی

مسأله ۸–وزیر: آمیزش

- ترکیب نمودن دو جایگشت در دو جایگشت جدید
- یک نقطه crossover به صورت تصادفی انتخاب کن
- بخش های اول را در فرزندان کپی کن
- بخش دوم هرفرزند را با درج مقادیراز والد دیگرایجاد کن
  - به ترتیب وقوع مقادیر
  - با شروع از مکان بعد از نقطهcrossover
  - با پرش از روی مقادیر موجود در فرزند



مسأله ۸–وزیر: انتخاب

انتخاب والد:

–۵ والد را انتخاب کن و ۲ تای بهتررا برای انجام crossover برگزین

انتخاب بازمانده (جایگزینی)

– هنگام درج یک فرزند جدید در جمعیت، به روش زیر یک عضو موجود را برای جایگزینی انتخاب کن

کل جمعیت را براساس مقادیربرازندگی به صورت نزولی مرتب کن

لیست را به ترتیب از بالا به پایین پیمایش کن

فرزند جدید را با اولین عنصری که برازندگی آن کمتر از برازندگی فرزند داده شده می باشد جایگزین کن

مسأله ۸–وزیر: خلاصه

توجه کنید که جدول بالا تنها یک مجموعه از انتخاب های ممکن را برای عملگرها و پارامترها نشان می دهد.

Representation	Permutations
Recombination	"Cut-and-crossfill" crossover
Recombination probability	100%
Mutation	Swap
Mutation probability	80%
Parent selection	Best 2 out of random 5
Survival selection	Replace worst
Population size	100
Number of Offspring	2
Initialisation	Random
Termination condition	Solution or 10,000 fitness evaluation

مسأله کوله پشتی

یک کوله پشتی با گنجایش W و مجموعه ای از n شیء که هر کدام دارای ارزش v_i و وزن w_i می باشند، موجود است. می خواهیم زیرمجموعه ای با حداکثر ارزش از این n شی ء تعیین کنیم به طوری که مجموع وزنشان از گنجایش کوله پشتی بیشتر نشود.

- مسأله کوله پشتی: بازنمایی

  رشته های باینری به طول n : وجود ۱ در یک مکان مشخص به معنی انتخاب و وجود ۰ در آن مکان به معنی عدم انتخاب یک شیء مشخص می باشد

- فضای ژنوتایپ :
  - مجموعه تمام رشته های باینری به طول n
  - اندازه آن به طور نمایی با افزایش n افزا یش می یابد.

مسأله کوله پشتی : روش های بازنمایی

مسأله کوله پشتی: روش بازنمایی اول

- فضای فنوتایپ = فضای ژنوتایپ
- تبدیل ژنوتایپ به فنوتایپ: یک به یک
- ارزیابی برازندگی
  - جمع ارزش اشیاء موجود در کوله پشتی
- مشکل این روش بازنمایی
  - راه حل های نامعتبربا وزن بیشتراز گنجایش کوله پشتی

مسأله کوله پشتی: روش بازنمایی دوم

- رشته باینری به طول n
- تبدیل ژنوتایپ به فنوتایپ: چند به یک

–رشته باینری را از چپ به را ست می پیماییم تا به اولین قطعه (اولین ۱) برسیم که اگر وزن آن به وزن مجموع فعلی اضافه شود، وزن مجموع از وزن کوله پشتی بیشترمی شود

–این نگاشت تضمین می کند که تمام رشته های باینری بیانگر یک راه حل معتبربا برازندگی منحصربه فرد می باشد

مسأله کوله پشتی : عملگرهای ژنتیکی

❑آمیزش :

– Crossover تک نقطه ای با احتمال ۷۰ درصد

❑جهش (bit-flipping) :

–با احتمالی برابر عکس اندازه کروموزوم (n/1)مقدار هر مکان تغییر می کند

مسأله کوله پشتی: انتخاب

- انتخاب والد
  - kبار و هر بار دو کروموزوم را به طور تصادفی از جمعیت انتخاب کن و از بین این دو، کروموزوم با برازندگی بیشتر را انتخاب کن
- انتخاب باز مانده: (جایگزینی)
  - کل جمعیت در هرتکرار با فرزندان آنها جانشین می شود

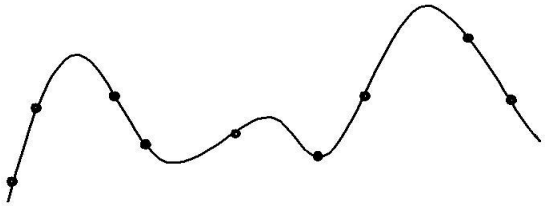
مسأله کوله پشتی: ایجاد جمعیت اولیه و توقف

- تولید (k اندازه جمعیت) رشته باینری به طول n به طور تصادفی
  - هرمکان از رشته باینری با احتمال برابر، با ۱ یا ۰ پرمی شود
- توقف :
  - الگوریتم تا زمانی که در ۲۵ نسل متوالی بهبودی در برازندگی بهترین عضو جمعیت مشاهده نشود، اجرا می شود

مسأله کوله پشتی: خلاصه

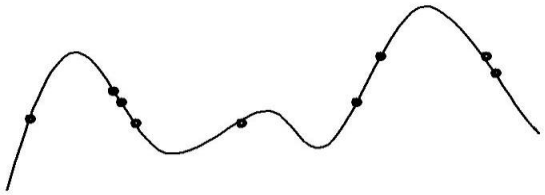
Representation	Binary string of length n
Recombination	One point crossover
Recombination probability P_c	70%
Mutation	Flip mutation
Mutation probability p_m	$1/n$
Parent selection	Best out of 2 random member
Survival Selection	Generational
Population size	500
Number of offspring	500
Initialization	Random
Termination condition	No improvement in last 25 generations

فازهای مختلف در بهینه سازی یک تابع برازندگی یک بعدی

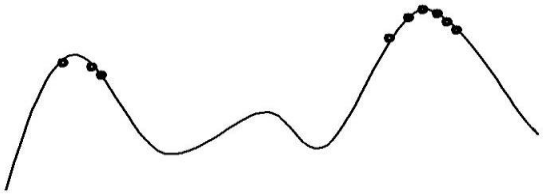


فاز اولیه

- توزیع تصادفی افراد در کل فضای جستجو



- فاز میانی
  - جمعیت اطراف قله ها پراکنده است



- فاز نهایی
  - جمعیت در اطراف قله های مرتفع پراکنده است

Exploration Vs. Exploitation

فازهای مختلف جستجو اغلب برحسب موارد زیر دسته بندی می شوند

- ❑ (Exploration – :اکتشاف): تولید افراد جدید در نواحی آزمایش نشده از فضای جستجو
- ❑ (Exploitation –بهره برداری): تمرکز جستجو در نزدیکی راه حل های خوب

در فرآیند جستجوی تکاملی باید بین این دو توازن برقرار شود

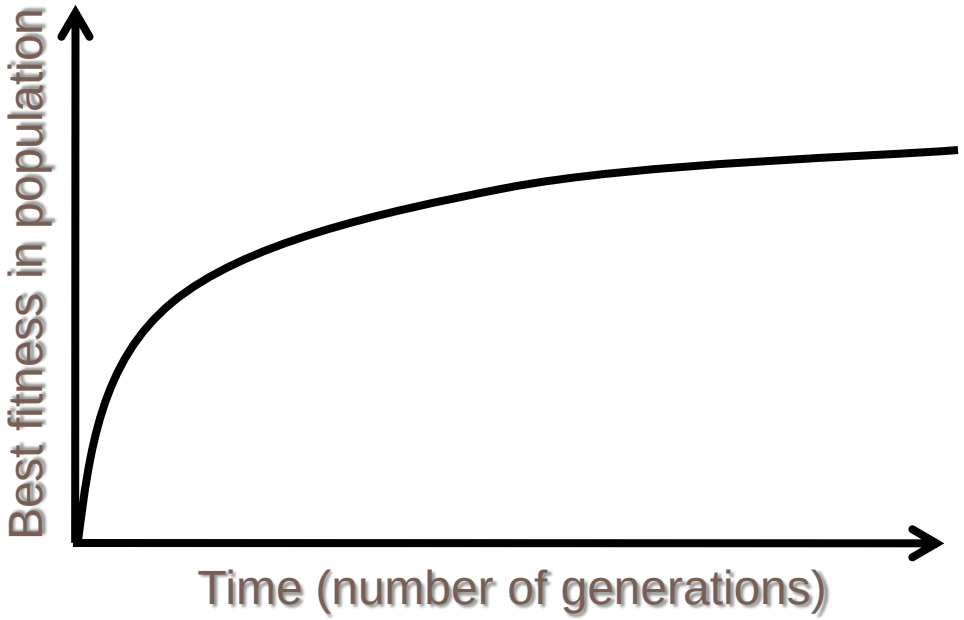
- ❖ Explorationزیاد: منجر به عدم کارآیی جستجو، پاسخ بهینه
- ❖ Exploitationزیاد: منجر به افزایش میزان حریصانه بودن جستجو، سریع، پاسخ غیربهینه

- ❑ همگرایی زودرس (convergence premature): از دست دادن سریع تنوع جمعیت و افتادن در دام بهینه های محلی

یک نمودار اجرای نوعی

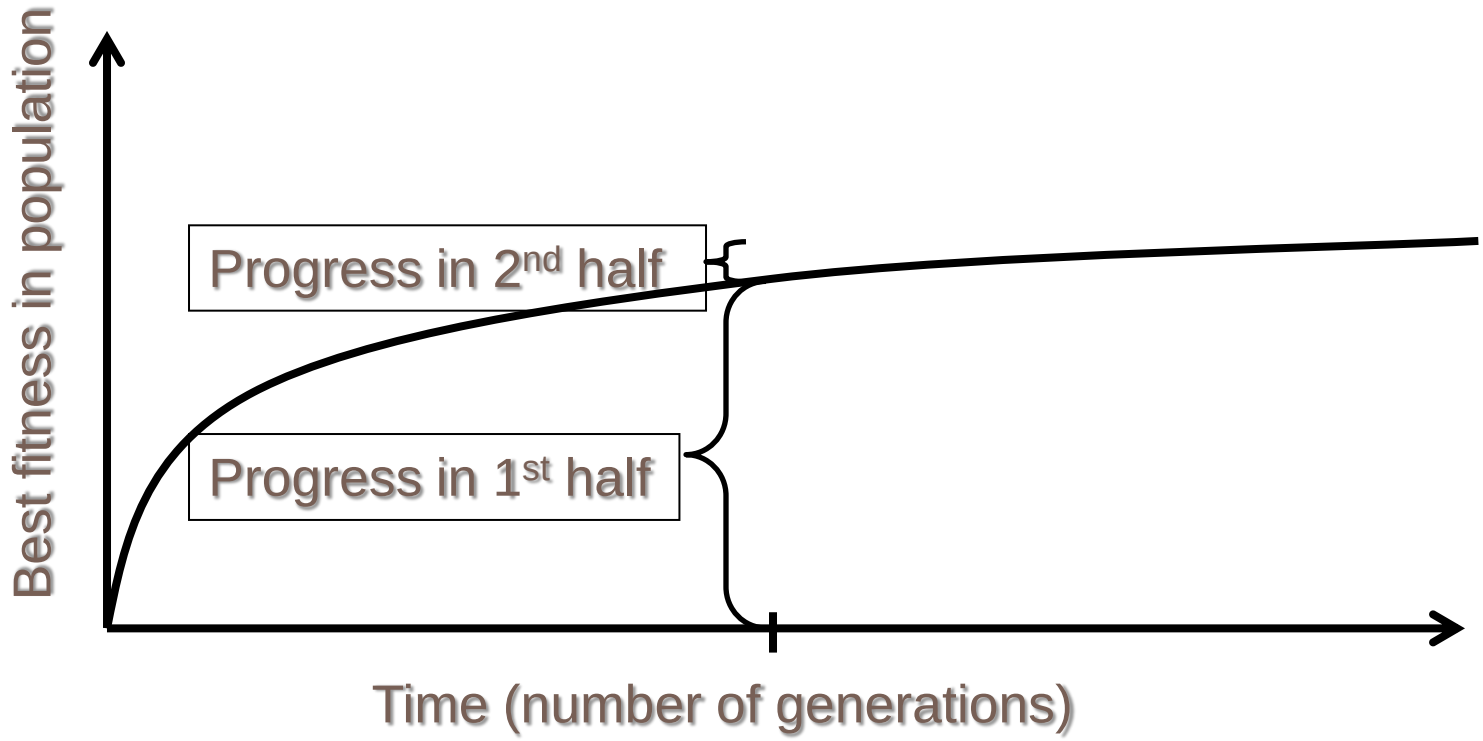
- نشانگر "anytime behavior"

معنی anytime : جستجو را می توان در هر زمان دلخواه متوقف نمود و الگوریتم یک راه حل می یابد (اگرچه زیربهینه)



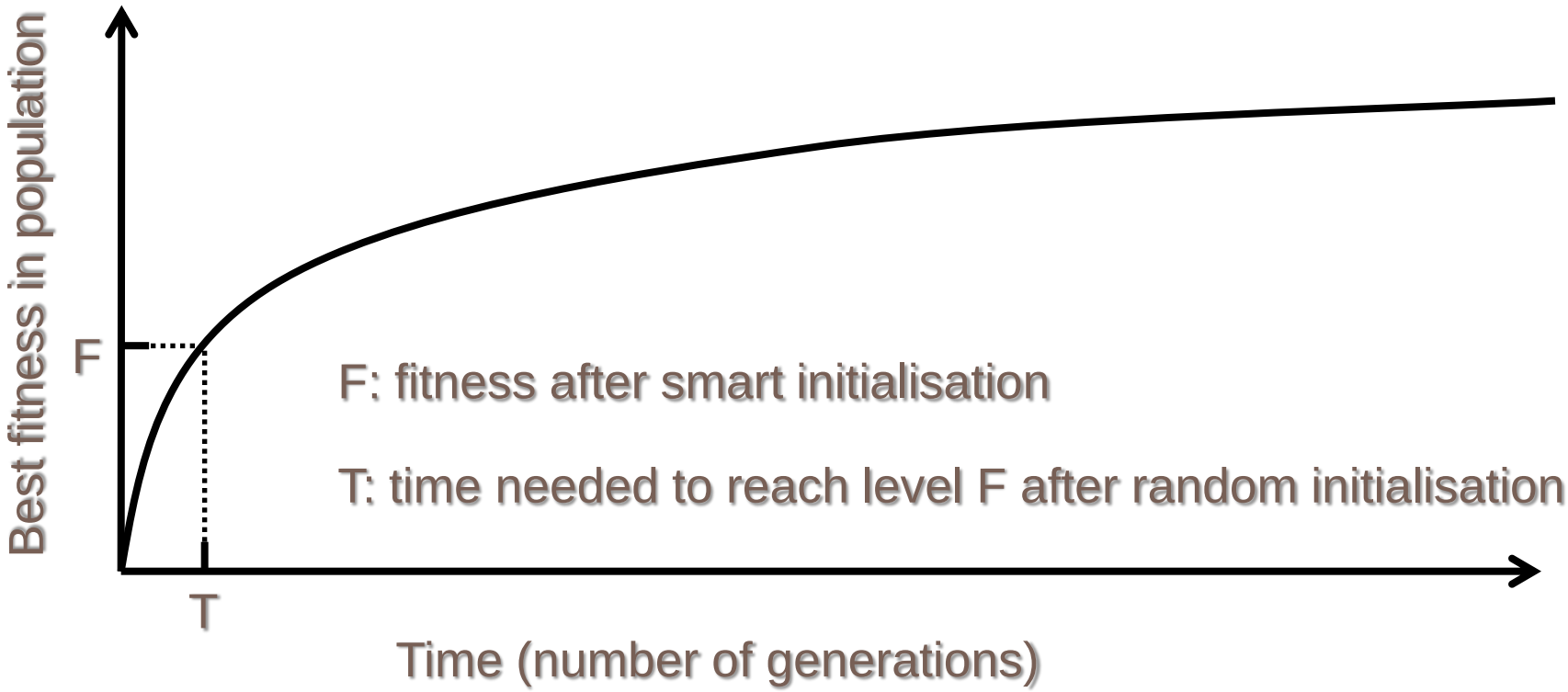
آیا اجراهای اجراهای طولانی مفید می باشند؟

- پاسخ: بستگی به مقدار مورد نیاز برازندگی دارد
  - بهتر است اجراهای کوتاه تربیشتری انجام شود



آیا تلاش برای ایجاد جمعیت اولیه به صورت هوشمند ارزش دارد؟

- پاسخ: بستگی دارد:
  - احتمالاً، اگرراه حل ها یا روش های خوبی وجود دارد
  - به دقت فراوانی نیاز است.



الگوریتم های تکاملی در مفهوم

دیدگاه های زیادی در مورد استفاده از الگوریتم های تکاملی به عنوان یک ابزار قوی در حل مسأله وجود دارد:

برای اغلب مسائل یک ابزار خاص مسأله ممکن است:

– از یک الگوریتم جستجوی عمومی برروی اغلب نمونه ها بهترعمل کند

–کاربرد محدودی داشته باشد

–برروی تمام نمونه ها به خوبی عمل نکند

هدف فراهم کردن یک ابزار قوی با عملکرد خوب بر روی گستره وسیعی از مسائل و نمونه ها می باشد

- بهینه سازی سراسری: جستجو برای یافتن بهترین راه حل x از یک مجموعه ثابت S
- روش های قطعی
  - مانند شاخه و حد
  - یافتن x را تضمین می کند اما ممکن است در یک زمان بیش از چند جمله ای اجرا شود
- روش های هیوریستیک (تولید و تست)
  - قوانینی برای تصمیم گیری در مورد $S \in x$ بعدی که باید تولید شود
  - تضمینی برای اینکه بهترین راه حل یافته شده راه حل بهینه سراسری باشد وجود ندارد

پردازش تکاملی و جستجوی همسایگی

- بسیاری از هیوریستیک ها یک ساختار همسایگی را بر روی S تحمیل می کنند(روش های جستجوی محلی)

- چنین هیوریستیک هایی تضمین می کنند که بهترین نقطه یافته شده بهینه محلی باشد(مانند تپه نوردی)
  - اما در اغلب مسائل، بهینه های محلی فراوانی وجود دارد
  - اغلب در یافتن یک راه حل خوب (نزدیک به بهینه سراسری) بسیار سریع می باشند

ویژگی های الگوریتم های تکاملی

- مبتنی بر جمعیت ====> فرار از بهینه های محلی – قابل استفاده در فضاهای حالت بزرگ
- استفاده از چندین عملگر جستجوی اتفاقی
- عملگرهای ژنتیکی ویژه با چندی بزرگتر از ۱
- انتخاب اتفاقی