الگوریتم های جست و جوی محلی و بهینه سازی

LOCAL SEARCH AND OPTIMIZATION

www.students.madadyar.com

جستجوی متا هیورستیک

"مسائلی وجود دارند که فضای جستجوی آنها به **اندازه ای بزرگ** می باشد که استفاده از روش های جستجوی ناآگاهانه و هیوریستیکی را برایمان غیرممکن می سازند.

"در این موارد از روش های متاهیوریستیکی یا فرا اکتشافی استفاده می کنیم.

الگوریتم های اصلاح تکراری یا جستجوی محلی

"دسته ای از مسائل وجود دارند که در آنها **مسیر** **رسیدن** **به** **جواب** **مهم** **نیست** بلکه فقط **جواب نهایی** اهمیت.

“ایده اصلی در این الگوریتم ها این است که وقتی از حالت قبلی به حالت فعلی برسیم دیگر حالت قبلی را فراموش کنیم.

“لذا در این مسائل **درخت نداریم** بلکه تنها یک وضعیت فعلی داریم.

www.students.madadyar.com

“وضعیت فعلی

“شامل تمام اطلاعات مورد نیاز مسئله است که در این الگوریتم ها از یک ساختار کامل شروع شده و سپس با انجام تغییراتی در آن سعی در اصلاح کیفیت آن ساختار داریم.

www.students.madadyar.com

دو مزیت اصلی این الگوریتم ها

- "استفاده از حافظه کم.
- "ارائه راه حلهای منطقی در فضاهای بزرگ و نامتناهی.

www.students.madadyar.com

- الگوریتم های قبلی، فضای جست و جو را به طور نظامند بررسی می‌کنند.
  - تا رسیدن به هدف یک یا چند مسیر نگهداری می‌شوند.
  - مسیر رسیدن به هدف، راه حل مسئله را تشکیل می‌دهد.
- **در الگوریتم های محلی مسیر رسیدن به هدف مهم نیست خود هدف مهم است!**
  - مثال: مسئله ۸ وزیر، حل معادلات n مجهولی و ...

این الگوریتمها برای حل مسائل بهینه سازی نیز مفیدند.

یافتن بهترین حالت بر اساس تابع هدف.

www.students.madadyar.com

شعار این الگوریتم ها

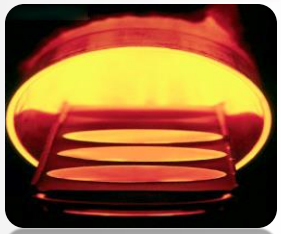
از یک حالت شروع کن

و با انجام تغییراتی در آن سعی در اصلاح کیفیت کن

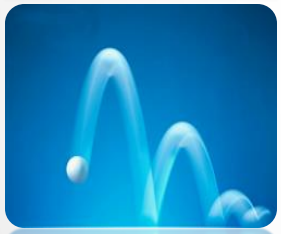
(تکامل داروینی)

انواع الگوریتم های جستجوی محلی

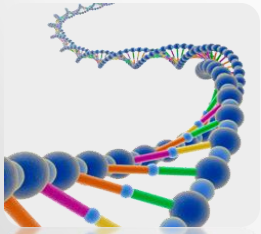
تپه نوردی



شبیه سازی حرارت

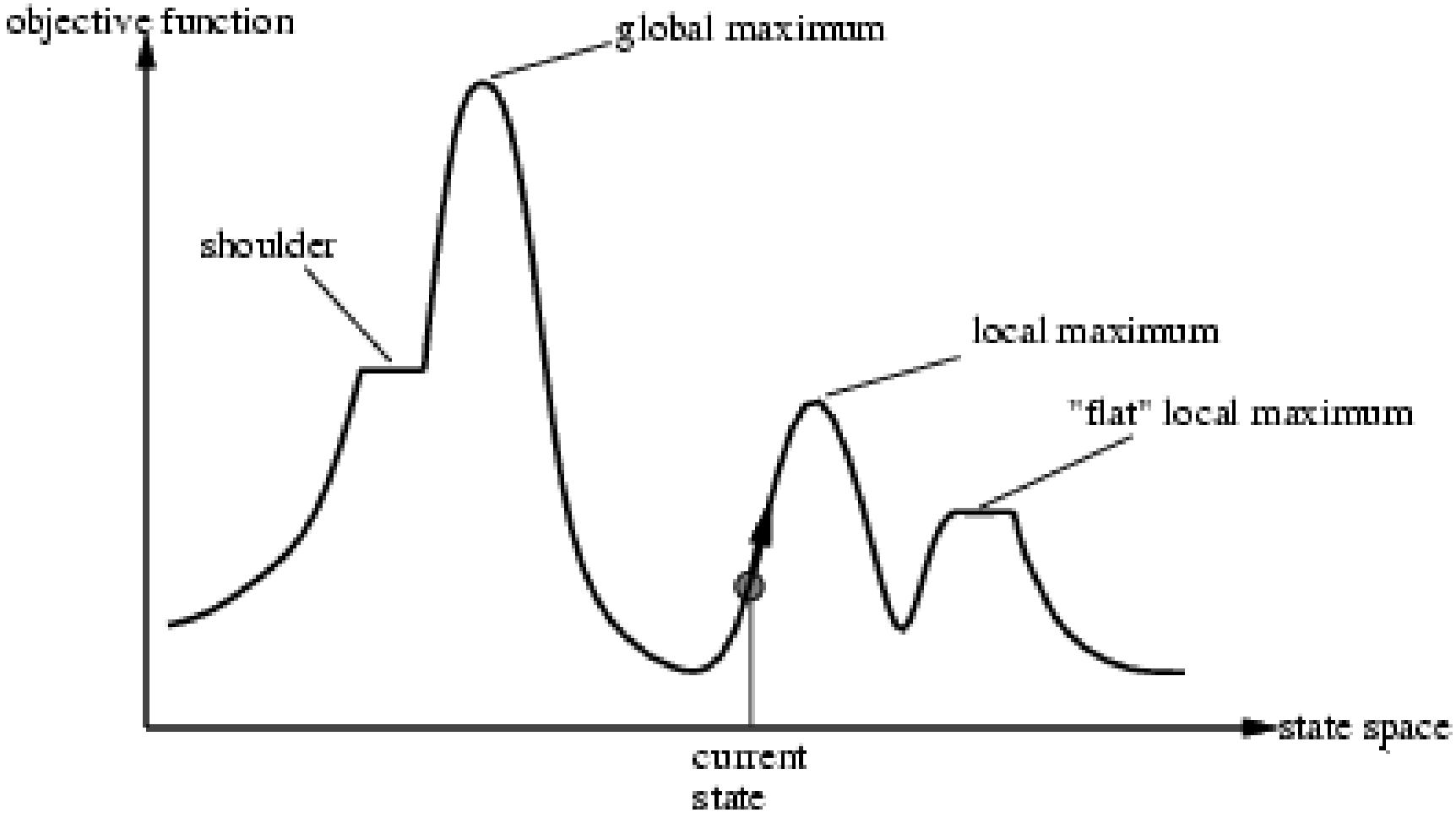


جستجوی پرتو محلی



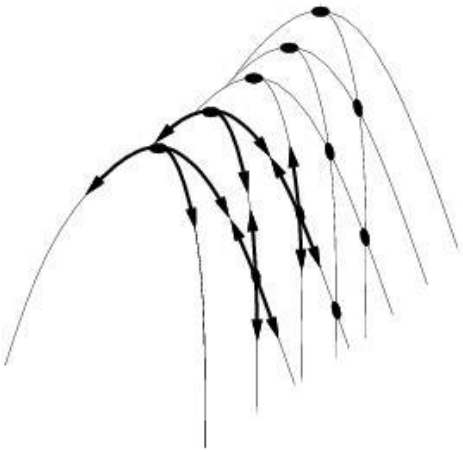
ژنتیک

www.students.madadyar.com


objective function
global maximum
shoulder
local maximum
"flat" local maximum
state space
current
state

جست و جوی تپه نوردی (Hill Climbing)

- حلقه ای که در جهت افزایش مقدار حرکت می‌کند (بطرف بالای تپه).
  - رسیدن به بلندترین قله در همسایگی حالت فعلی، شرط خاتمه است.
- ساختمان داده گره فعلی، فقط حالت و مقدار تابع هدف را نگه می‌دارد.
- جست و جوی محلی حریصانه نیز نام دارد.
  - بدون فکر قبلی حالت همسایه خوبی را انتخاب می‌کند.
- تپه نوردی به دلایل زیر می‌تواند متوقف شود:
  - بیشینه محلی
  - برآمدگی ها
  - فلات



www.students.madadyar.com

الگوریتم تپه نوردی

“ به عبارت دیگر : در الگوریتم تپه نوردی ابتدا حالت اولیه به شکل تصادفی شروع می شود.

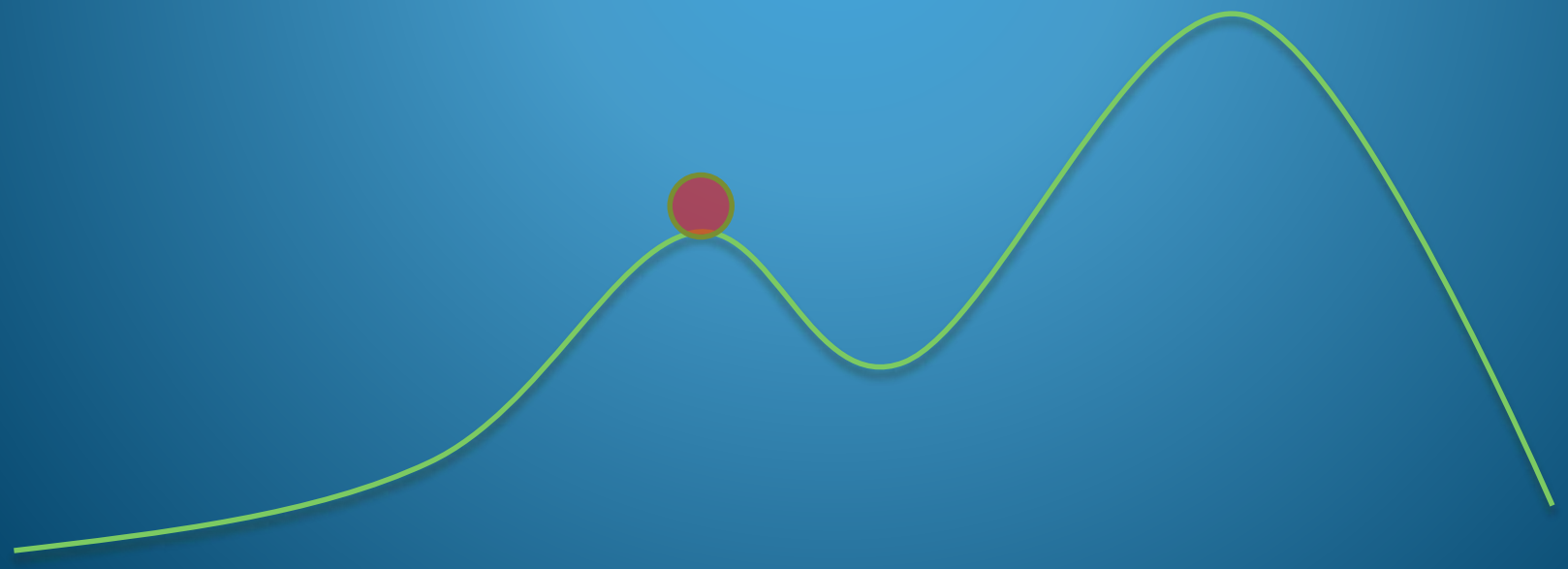
“ سپس پس از ایجاد حالت های ممکن بهترین حالت را انتخاب می کند و به آن حالت می رود .

علل ناکامی الگوریتم تپه نوردی

ماکزیمم محلی

حالتی است که از تمامی حالات همسایه خود بهتر است (حالت جاری از همسایه های هم سطح خود بهتر است) اما از چند حالت بعدتر از این همسایه ها بهتر نیست.

در یک ماکزیمم محلی هر حرکتی به نظر اشتباه به نظر می آید.



علل ناکامی الگوریتم تپه نوردی

فلات

ناحیه ای است که مقدار تابع ارز یابی در آن ثابت است .

دو نوع فلات داریم :

۱) ماکزیمم محلی صاف

۲) شانه

انواع جست و جوی تپه نوردی

- تپه نوردي اتفاقي (stochastic)
  - انتخاب یک حالت رو به بالا به صورت تصادفي.
- تپه نوردي اولین گزینه (first choice)
  - تپه نوردي اتفاقي با تولید پسین به صورت تصادفي تا تولید یک پسین بهتر.
- تپه نوردي با شروع مجدد تصادفي (random restart)
  - جستجوي تپه نوردي با حالتهاي شروع تصادفي (با تکرار در صورت شکست).
  - با احتمال نزدیک به یک کامل است!!

www.students.madadyar.com

الگوریتم جستجوی تپه نوردی

```
function HILL-CLIMBING(problem) returns a state that is a local maximum
   inputs: problem, a problem
   local variables: current, a node
                    neighbor, a node

   current ← MAKE-NODE(INITIAL-STATE[problem])
   loop do
      neighbor ← a highest-valued successor of current
      if VALUE[neighbor] ≤ VALUE[current] then return STATE[current]
      current ← neighbor
```

مثال جست و جوی تپه نوردی

مثال: مسئله ۸ وزیر

مسئله ۸ وزیر با استفاده از فرمولبندی حالت کامل.

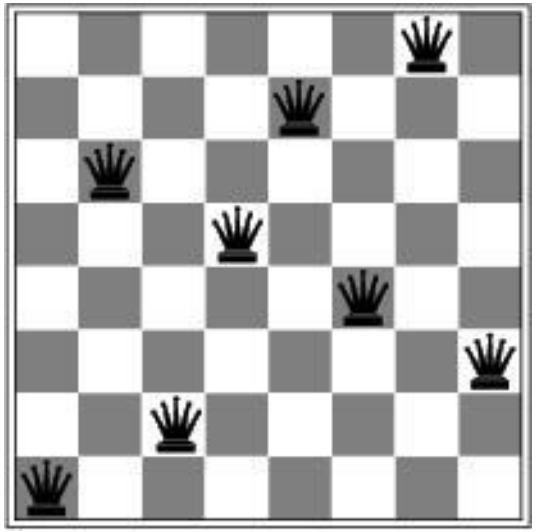
- در هر حالت ۸ وزیر در صفحه قرار دارند.
- تابع جانشین: انتقال یک وزیر به مربع دیگر در همان ستون.
- تابع اکتشاف: جفت وزیرهایی که نسبت به هم گارد دارند.

www.students.madadyar.com

مثال جست و جوی تپه نوردی

الف

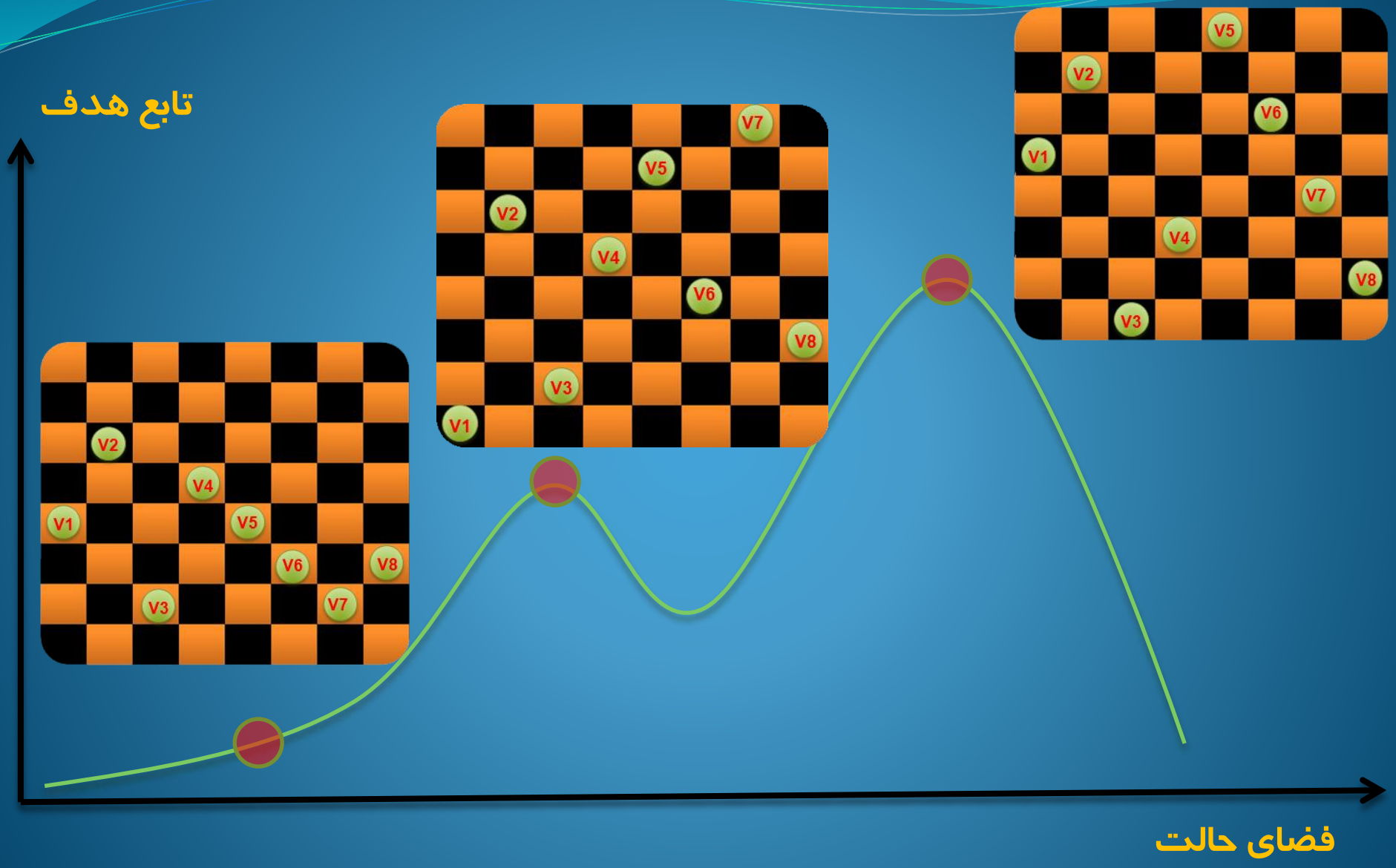
ب



الف- حالت با هزینه h=17 که مقدار h را برای هر جانشین نشان می‌دهد.

ب- کمینه محلی در فضای حالت ۸ وزیر؛ h=1.

نمودار قسمتی از فضای حالت هشت وزیر با الگوریتم تپه نوردی



نکات مهم الگوریتم تپه نوردی

الگوریتم تپه نوردی در صورتی کامل است که درگیر وضعیتهای ذکر شده نگردد.

الگوریتم تپه نوردی ممکن است در یافتن پاسخ مسئله شکست بخورد بدین شکل که هیچ کدام از حالات بعدی بهتر از حالت جاری نبوده و حالت جاری نیز یک حالت هدف نباشد.

الگوریتم تپه نوردی همانند الگوریتم حریصانه است با این تفاوت که امکان بازگشت به عقب وجود ندارد.

جست و جوی شبیه سازی حرارت

- شبیه جستجوی تپه نوردی می باشد با این تفاوت که در هر مرحله حالت بعدی، بجای انتخاب بهترین حرکت ، به طور تصادفی انتخاب می شود.

- تپه نوردی با حرکت تصادفی.
  - اگر حرکت تصادفی انتخاب شده موجب بهبود وضعیت شود، انجام می‌شود و در غیر اینصورت فقط با یک احتمال (که به صورت نمایی کاهش می یابد) آن حرکت انجام می شود.

  - **ایده اصلی:** فرار از maxهای محلی با اجازه دادن به انجام حرکتهای بد (پایین آمدن) اما به تدریج اندازه و تعداد حرکتهای بد را کاهش می دهد.

این الگوریتم اصلاح الگوریتم تپه نوردی می باشد که در آن هنگام رسیدن به max محلی به جای شروع مجدد تصادفی، اجازه پایین رفتن از تپه برای فرار از آن محل، داده می شود.

www.students.madadyar.com

SA

```
function SIMULATED-ANNEALING( problem, schedule) returns a solution state
   inputs: problem, a problem
           schedule, a mapping from time to "temperature"
   local variables: current, a node
                    next, a node
                    T, a "temperature" controlling prob. of downward steps

   current ← MAKE-NODE(INITIAL-STATE[problem])
   for t ← 1 to ∞ do
       T ← schedule[t]
       if T = 0 then return current
       next ← a randomly selected successor of current
       ΔE ← VALUE[next] – VALUE[current]
       if ΔE > 0 then current ← next
       else current ← next only with probability e^(Δ E/T)
```

جست و جوی پرتو محلی (local beam search)

- به جای یک حالت، k حالت را نگهداری می‌کند.

- **حالت اولیه:** k حالت تصادفی.
- **گام بعد:** جانشین همه k حالت تولید می‌شود.
  - اگر یکی از جانشین ها هدف بود، تمام می‌شود.
  - وگر نه **بهترین** جانشین ها را انتخاب کرده، تکرار می‌کند.

www.students.madadyar.com

جست و جوی پرتو محلی (local beam search)

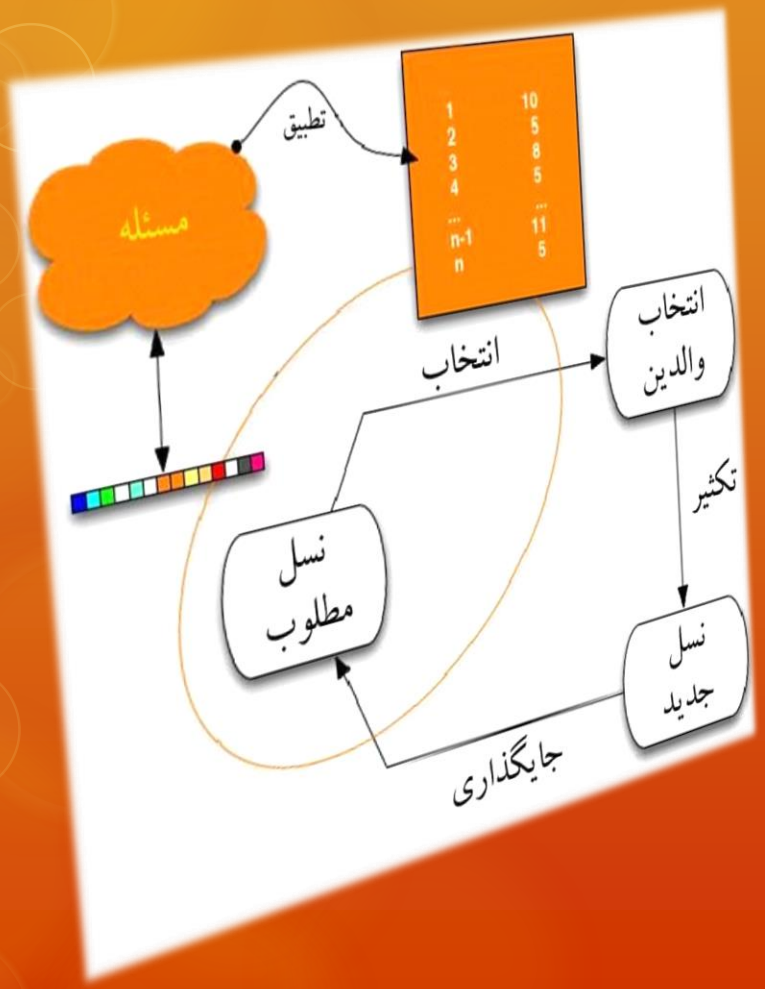
- تفاوت عمده با جستجوی شروع مجدد تصادفی
  - در جست و جوی شروع مجدد تصادفی، هر فرایند مستقل از بقیه اجرا می‌شود.
  - در جست و جوی پرتو محلی، اطلاعات مفیدی بین k فرایند موازی مبادله می‌شود!!

- جست و جوی پرتو محلی غیرقطعی

  به جای انتخاب بهترین k از جانشینها، k جانشین تصادفی را بطوریکه احتمال انتخاب یکی تابعی صعودی از مقدارش باشد، انتخاب می‌کند.

www.students.madadyar.com

الگوریتم ژنتیک
(Genetic Algorithm)



شکلی از جست و جوی پرتو محلی غیر قطعی که حالتهای جانشین از طریق ترکیب دو حالت والد تولید می‌شود.

www.students.madadyar.com

خصوصیات الگوریتم ژنتیک

- یک روش عددی و تصادفی و مبتنی بر تکرار است .
- در هر تکرار همزمان چند نقطه از فضای حالت بررسی می شود پس احتمال اینکه در ماکزیمم محلی گیر کند کم است .
- الگوریتم ژنتیک یک الگوریتم برای مسائل بهینه سازی است .
- الگوریتم ژنتیک به راحتی به مسائل پیوسته و گسسته بکار می رود.
- الگوریتم ژنتیک روندی موازی در حل مسله دارد .

فاز های اصلی الگوریتم ژنتیک

۱) کدینگ مسله (Encoding)

۲) تعیین جمعیت اولیه(Initial Population)

۳) انتخاب کردن(Selection)

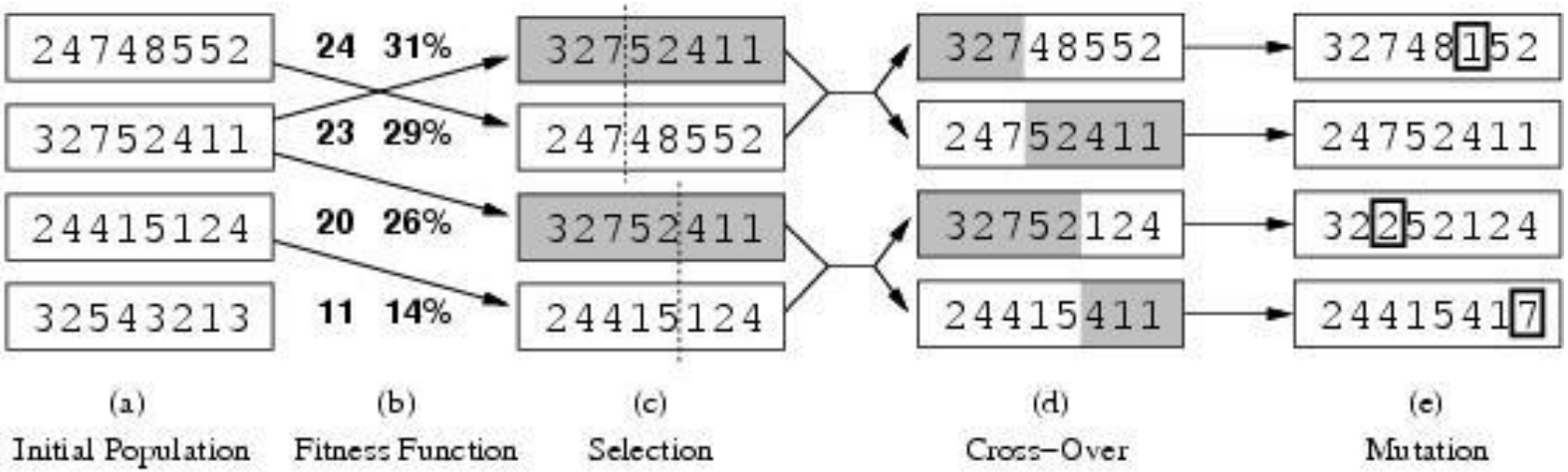
٤) ترکیب(Crossover)

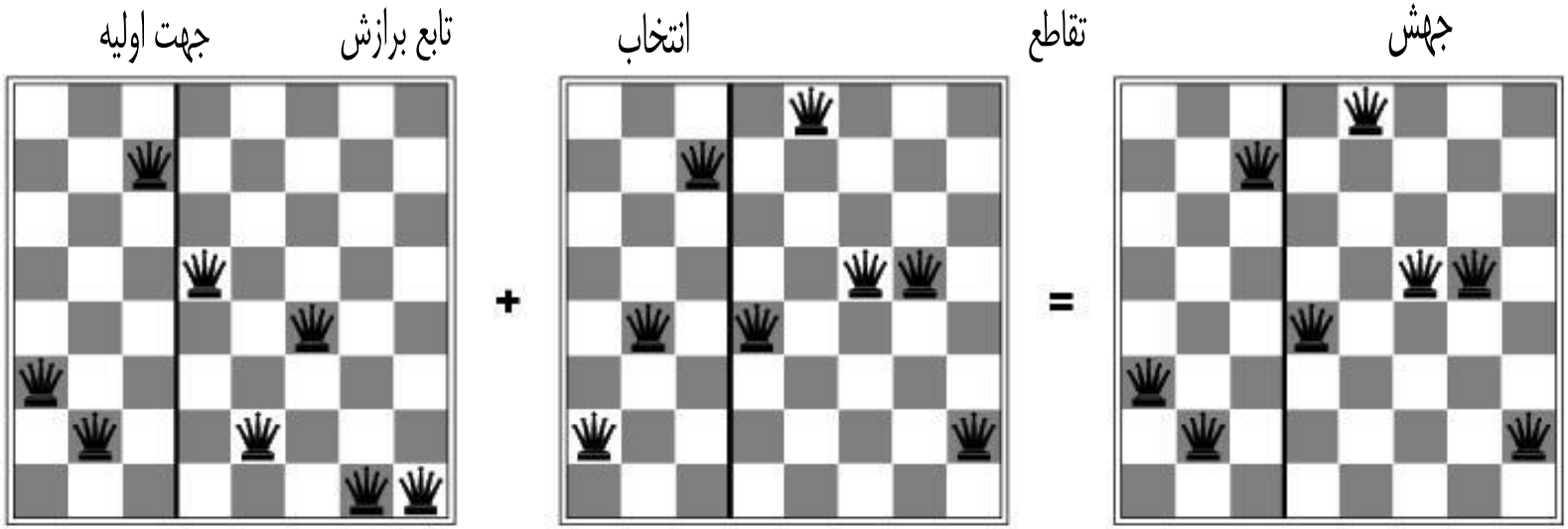
۵) جهش(Mutation)

۶) پذیرش جمعیت جدید و تست(Test & Accept)

۷) جایگزینی(Replace)

الگوریتم ژنتیک





www.students.magaayar.com

آشنایی با الگوریتم های ژنتیکی

GENETIC ALGORITHMS

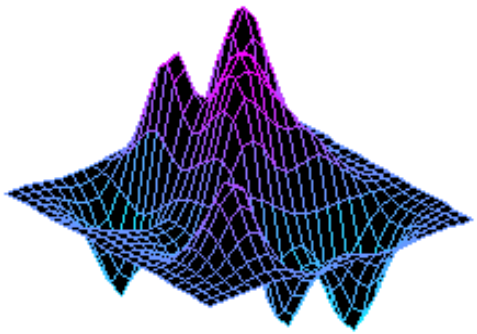
تاریخچه

- **GAs جزو شاخه خاصی از هوش مصنوعی با نام "محاسبات تکاملی" (Evolutionary Computing) هستند.**

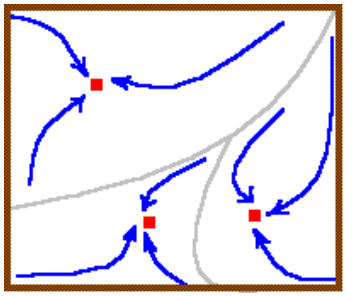
- **ایده محاسبات تکاملی اولین بار در اثر Rechenberg در سال ۱۹۶۰ با عنوان "*Evolution strategies*" معرفی شد.**

www.students.madadyar.com

Search Space فضای جستجو



- فضای تمام جوابهای ممکن یک مسئله فضای جستجو یا فضای حالت مسئله نامیده می شود.
- هر نقطه در فضای حالت، یک جواب ممکن را مشخص می کند.
- پیدا کردن جواب مساوی است با پیدا کردن نقاط ماکزیمم یا مینیمم در فضای حالت.



www.students.madadyar.com

فضای جستجو

- برای فضاهای کوچک، روشهای کلاسیک جستجو و بهینه سازی خوب و کافی هستند اما برای فضاهای بزرگ و مسائل مشکل نیازمند به تکنیکهای دیگری همچون روشهای ذیل هستیم:

- **Hill Climbing.**
- **Tabu Search.**
- **Simulated Annealing.**
- **Genetic Algorithm.**

www.students.madadyar.com

ساختار کلی الگوریتم های ژنتیکی

1. **[Start]** Generate random population of *n* chromosomes
2. **[Fitness]** Evaluate the fitness of each chromosome in the population
3. **[New population]** Create a new population by repeating following steps until the new population is complete
   - **[Selection]** Select two parent chromosomes from a population according to their fitness
   - **[Crossover]** With a crossover probability cross over the parents to form a new offspring (children). If no crossover was performed, offspring is an exact copy of parents.
   - **[Mutation]** With a mutation probability mutate new offspring at each locus (position in chromosome).
   - **[Accepting]** Place new offspring in a new population
   - **[Replace]** Use new generated population for a further run of algorithm
4. **[Test]** If the end condition is satisfied, **stop**, and return the best solution in current population
5. **[Loop]** Go to step **2**

www.students.madadyar.com

چند سؤال

چگونگی نمایش جوابهای ممکن در قالب کروموزومها.

چگونگی ایجاد جمعیت اولیه.

چگونگی تعریف اعمال crossover و mutation.

چگونگی انتخاب والدین برای crossover.

www.students.madadyar.com

نمایش کروموزومها

- **Encoding & Decoding**

- کروموزومها به نحوی باید حاوی اطلاعات جوابی باشند که نمایانگر آن هستند.

- یعنی هر کروموزوم به تنهایی می تواند یک راه حل از مساله باشد.(نه لزوما راه حل بهینه یا درست ؟!).

www.students.madadyar.com

عملگرهای GAs (Crossover)

مثال:

Chromosome1 11011|00100110110
Chromosome2 11011|11000011110

Offspring 1 11011|11000011110
Offspring 2 11011|00100110110

- Methods
  - Single Point
  - Two Point
  - ....

www.students.madadyar.com

عملگرهای GAs (Mutation)

- مثال

Original offspring 1101111000011110

Mutated offspring 1100111000011110

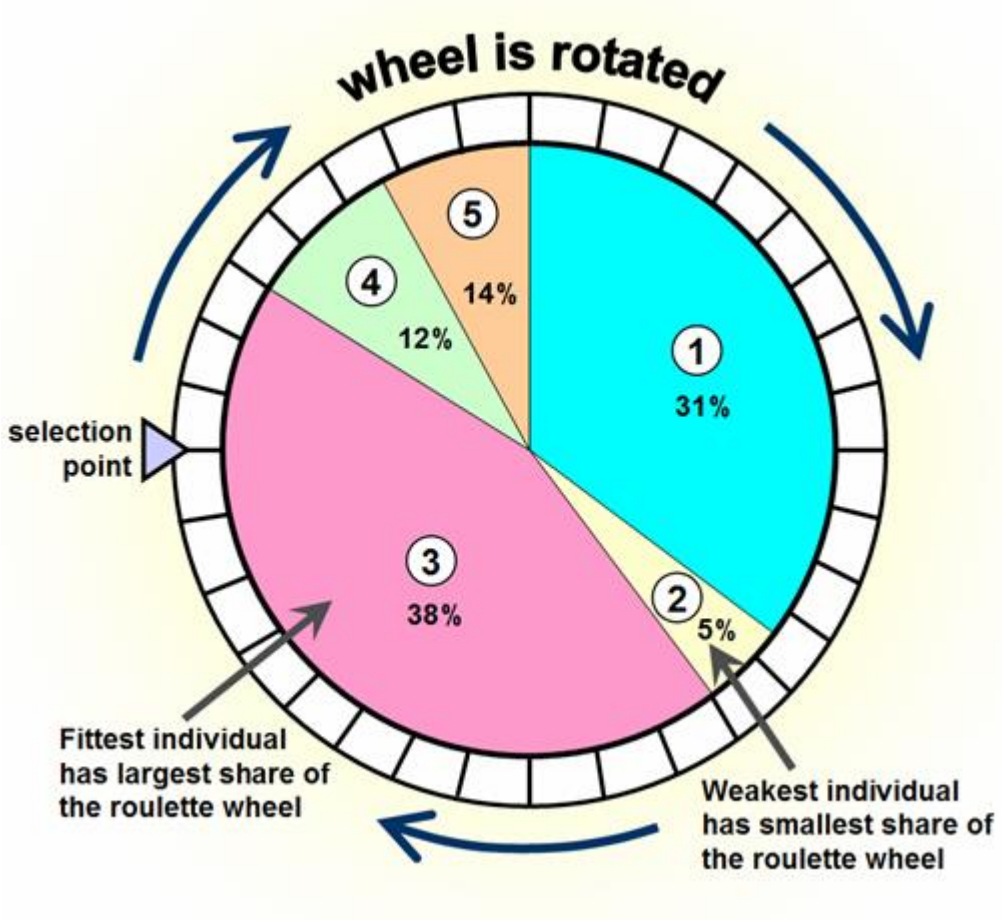
- Methods
  - Inversion
  - Exchange

www.students.madadyar.com

نحوه انتخاب والدین

- روش‌های مختلفی برای الگوریتم‌های ژنتیک وجود دارند که می‌توان برای انتخاب کروموزوم ها از آن‌ها استفاده کرد. اما روش‌های لیست شده در پایین از معمول‌ترین روش‌ها هستند.
- انتخاب **Elitist**
  - درصدي از مناسب‌ترين عضوهاي هر اجتماع انتخاب می‌شوند.
- انتخاب **Roulette**
  - یک روش انتخاب است که در آن عنصری که عدد برازش (تناسب) بیشتری داشته باشد ، انتخاب می‌شود.
- انتخاب **Tournament**
  - یک زیر مجموعه از یک جامعه انتخاب می‌شوند و اعضای آن مجموعه با هم رقابت می‌کنند و سرانجام فقط یک فرد از هر زیرگروه برای تولید انتخاب می‌شوند.

www.students.madadyar.com


wheel is rotated
5
14%
4
12%
1
31%
selection point
3
38%
2
5%
Fittest individual has largest share of the roulette wheel
Weakest individual has smallest share of the roulette wheel

پارامترهای ژنتیک

- **Parameters**
  1. Population size.
  2. Probability of crossover.
  3. Probability of mutation .

- **Experimental results**
  1. The best chromosome after n generations was.
  2. As expected .

www.students.madadyar.com

جست و جوی محلی در فضاهای پیوسته

- گسسته در مقابل محیط های پیوسته
  - در فضاهای پیوسته، تابع جانشین در اغلب موارد، حالتهای نامتناهی را بر میگرداند.
- حل مسئله:
  - گسسته کردن همسایه هر حالت.

www.students.madadyar.com

عاملهای جست و جوی Online

و محیطهای ناشناخته

www.students.madadyar.com

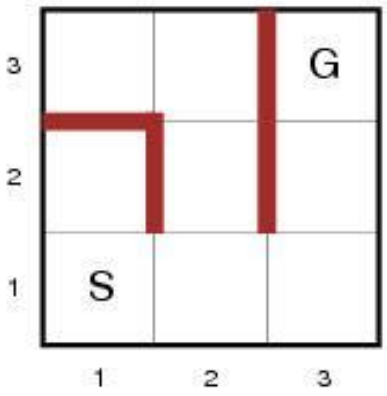
عاملهای جست و جوی Online و محیطهای ناشناخته

- الگوریتمهای مطالعه شده برون خطی بودند.
  - **برون خطی(Offline)**: راه حل قبل از اجرا مشخص است.
  - **درون خطی(Online)**: با یک در میان کردن محاسبات و فعالیت عمل می‌کند.
- عامل در حین حل جستجو در محیط واقعی است!
- جستجوی درون خطی در محیطهای پویا و نیمه پویا مفید است.
  - فعالیتها و حالتها برای عامل مشخص نیستند.
    - مثال:قرار گرفتن روبات در محیطی جدید.

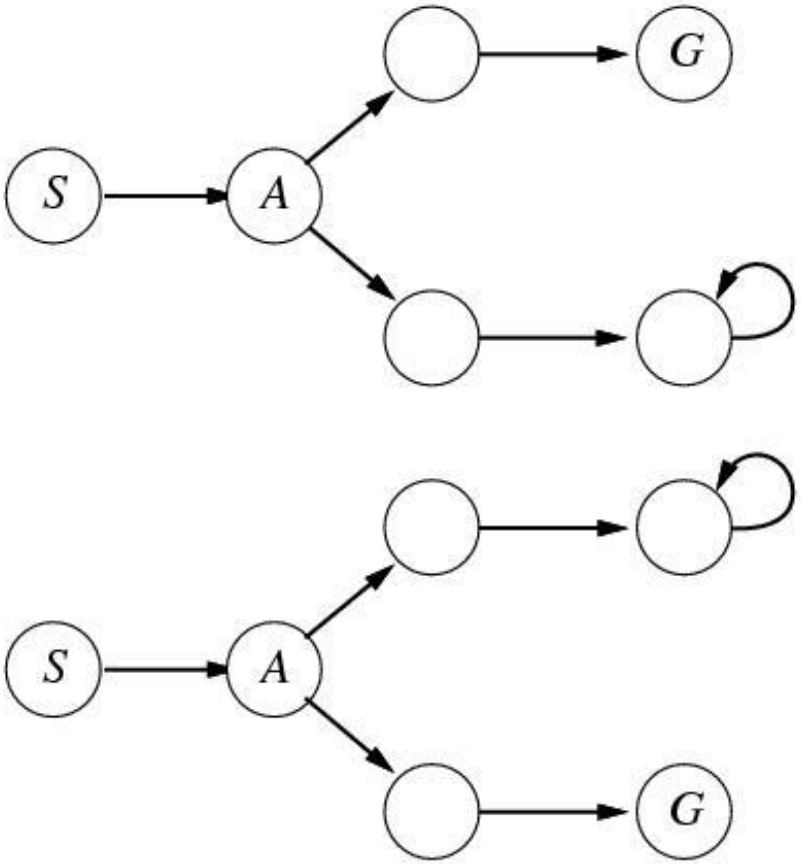
www.students.madadyar.com

مسئله های جست و جوی Online

- اطلاعات عامل
  - Actions(s): لیستی از فعالیتهای مجاز در حالت s
  - تابع هزینه مرحله ای c(s,a,s’): استفاده وقتی که بداند s’ نتیجه است.
  - Goal-Test(s): آزمون هدف.
- عامل حالت ملاقات شده قبلی را تشخیص می‌دهد.
- فرض: فعالیتها قطعی اند(محیط معین).
- دسترسی به تابع اکتشافی قابل قبول h(s).



مسئله های جست و جوی Online



- دو فضای حالت که عامل جست و جوی Online را به بن بست میرسانند. هر عاملی حداقل در یکی از این دو فضا شکست می خورد.

جستجوی محلی بر خط

- جستجوی اول عمق مانند جستجوی تپه نوردی در گسترش گره ها دارای ویژگی محلی است!

- جستجوی تپه نوردی یک الگوریتم جستجوی بر خط است!
  - عیب : بیشینه محلی ، عدم امکان استفاده از شروع مجدد تصادفی!؟

- **LRTA***
  - جستجوی تپه نوردی با امکان حافظه برای نگه داری H(s) یا بهترین برآورد فعلی.
  - با کسب تجربه عامل در فضای حالت بروزرسانی می‌شود.

www.students.madadyar.com

- $H(s)$ یا "بهترین برآورد فعلی" هزینه رسیدن به هدف از هر حالتی است که از آن گذشته ایم.

- در ابتدا $H(s)'$ برابر باهیوریستیک هر محل است ($h(s)'$).

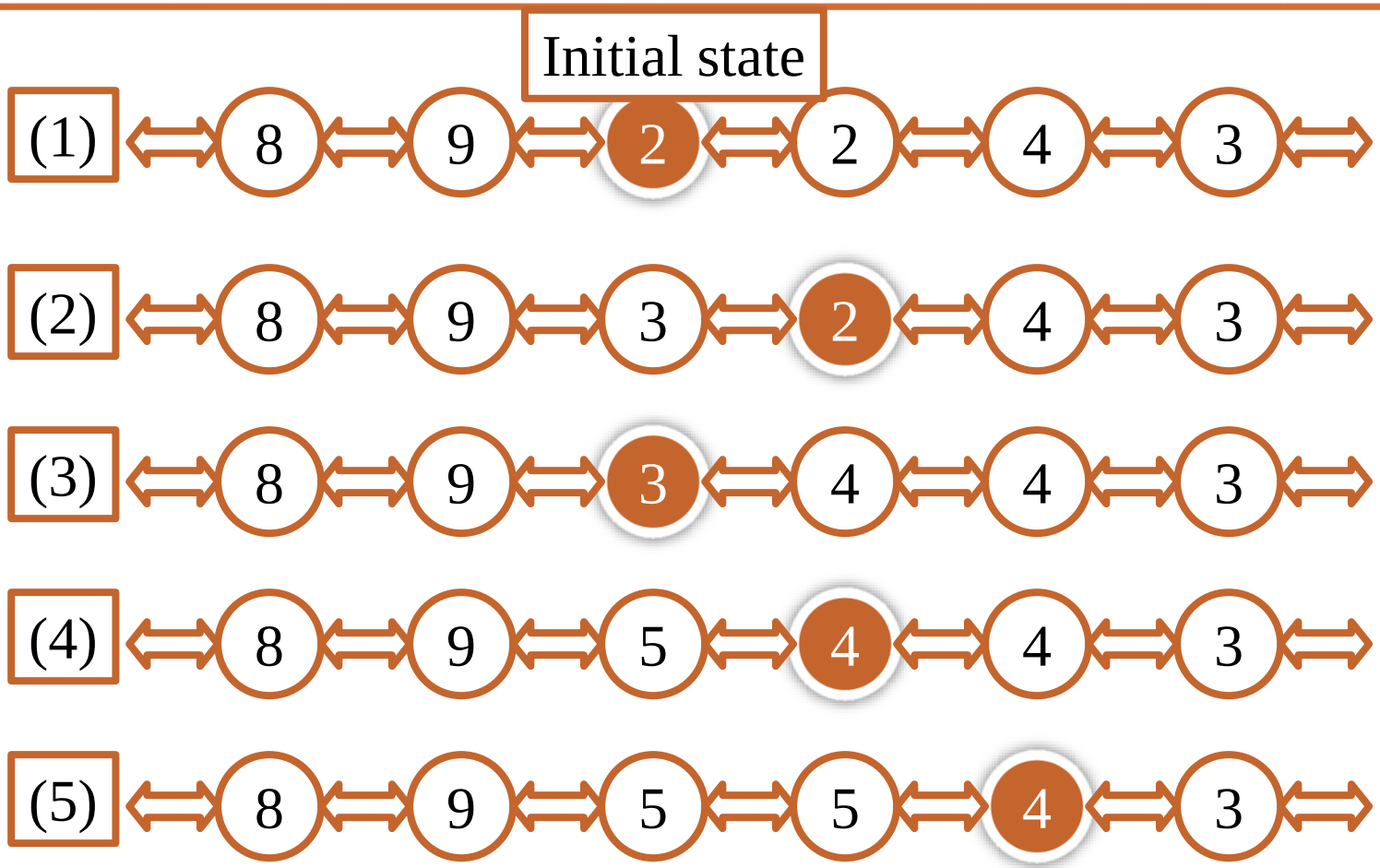
- به تدریج که عامل در فضای حالت تجربه کسب می کند، $H(s)'$ به روز آوری می شود.

- هزینه تخمینی رسیدن به هدف از مسیر گره همسایه ای مثل s' برابر است با :
  - هزینه رسیدن به s' $c(s,a,s')$ + هزینه تخمینی رسیدن به هدف از طریق s'.

- به عبارت دیگر : $H(s)' = c(s,a,s') + H(s')$

www.students.madadyar.com

این مثال را در نظر بگیرید



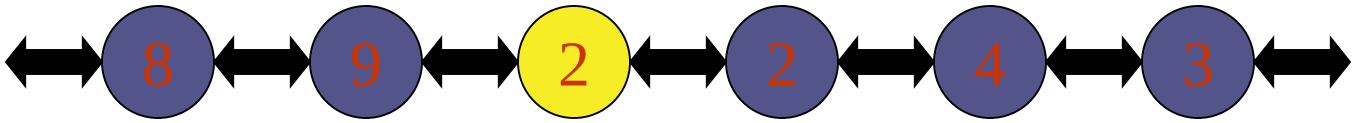
مراحل LRTA* در فضای یک بعدی

www.students.madadyar.com

✓ حالت زرد رنگ (حالت کنونی) ۲ قدم تا مسیر به هدف (گره با h=2) فاصله دارد

- در اینجا دو اقدام وجود دارد که دارای هزینه تقریبی ۱+۲ و ۱+۹ هستند.
- به نظر می آید که بهتر است به سمت راست حرکت کنیم.

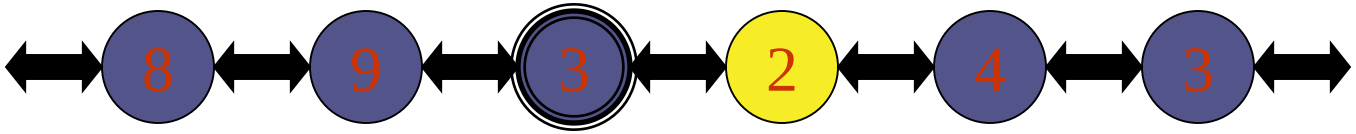
(1)



✓ اکنون تا هدف ۲ قدم وجود دارد ولی از دید عامل انتخاب چپ بهتر از حرکت به راست است.

- H باید بر اساس $H(s') = (c(s,a,s') + H(S')$ به روز آوری شود و ۳+۱ را به ۴+۱ ترجیح می دهد. $c(s,a,s')$ در اینجا برابر با ۱ است.

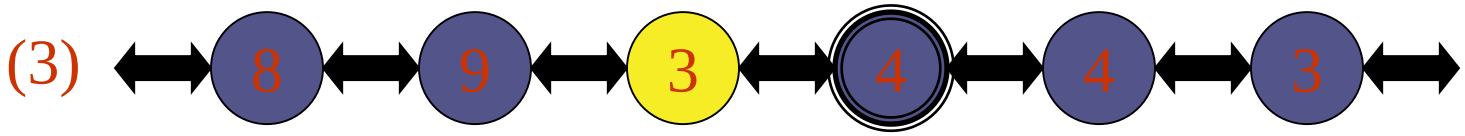
(2)



www.students.madadyar.com

✓ با ادامه این فرایند، عامل دوبرابر بیشتر به جلو و عقب حرکت می کند و با به روز آوری H کمینه محلی خود را مسطح می کند تا زمانی که نهایتاً از سمت راست فرار کند.

- عامل حالت بعدی ۴+۱ را به ۹+۱ ترجیح میدهد و دوباره به راست میرود.

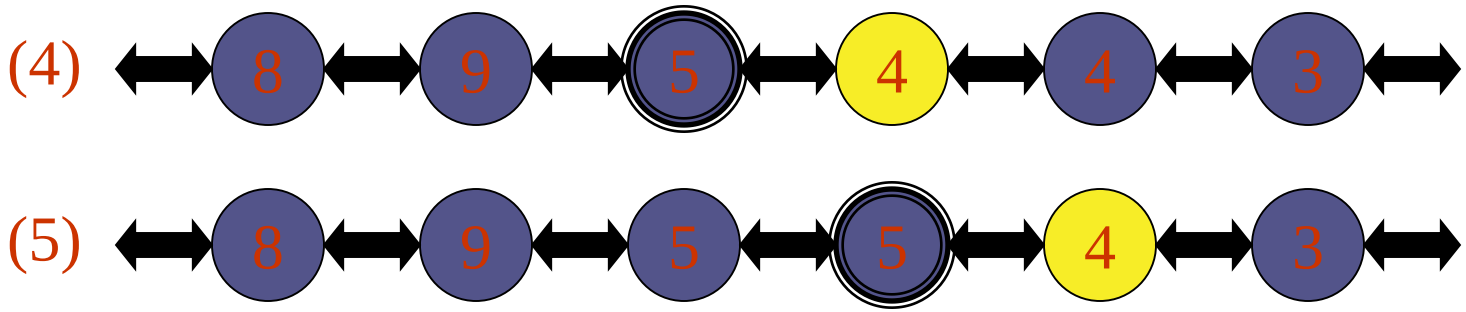


✓ اگر دقت کنیم می بینیم که عامل چندان هم باهوش نیست. مثلا پس از برگشت به راست، هنوز هیچ تصویری ندارد که اقدام چپ به حالت قبلی بر می گردد.

✓ (این در فضاهای چند بعدی خیلی تاثیر دارد).

www.students.madadyar.com

- پس از ادامه نهایتا کمینه محلی مسطح می شود و عامل از سمت راست به طرف هدف حرکت می کند.



- این عامل با استفاده از جدول result نقشه ای از محیط تهیه میکند.
- هزینه تخمینی حالتی که هم اکنون از آن خارج شده است را به روز آوری می کند.
- بر اساس تخمین های فعلی، حرکتی که به ظاهر بهترین است را انتخاب می کند.

✓ نکته : اقداماتی که تا این لحظه امتحان نشده اند، همیشه فرض می شود که با کمترین هزینه ممکن h(s) به هدف منتهی می شود. این خوش بینی عامل راتشویق می کند که به اکتشاف مسیر های جدید و احتمالاً امیدوار کننده بپردازد.

www.students.madadyar.com

بعضی از خصوصیات LRTA*

- برای یک عامل LRTA* ، پیدا کردن هدف در هر محیط متناهی مطمئناً قابل اکتشاف، تضمین شده است.

- با این حال، برخلاف A* کامل نیست (حالت هایی وجود دارد که ممکن است در آن برای همیشه سرگردان شود).

- این روش در بدترین حالت می تواند یک محیط n حالتی را در $O(n^2)$ گام اکتشاف میکند. ولی اغلب بهتر از این عمل می‌کند.

www.students.madadyar.com

END OF Chapter 4

THANK YOU

www.students.madadyar.com