

1 - سیستمی که مانند انسان رفتار می کند را با ذکر مثال تشریح کنید؟

نیاز به تئوری های علمی در خصوص فعالیت های مغزی دارد مثال تست تورینگ .

2 - هدف از تفکر عاقلانه چیست و چه آورده ای در پی خواهد داشت؟

• رفتار عاقلانه: انجام دادن عمل درست

• عمل درست یعنی دستیابی به بهترین هدف

• دو مزیت نسبت به تعاریف قبلی:

– عمومیت بیشتر نسبت به تفکر

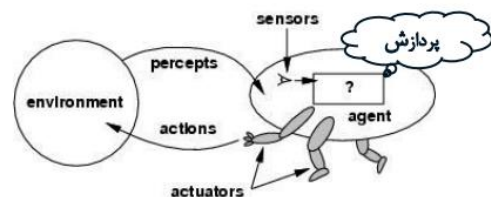
– عدم نیاز به تئوریهای پیشرفته علمی

3 - اجزای عامل و وظیفه عامل را با رسم شکل و تابع نویسی بررسی کنید؟

اجزای عامل

1. سنسور: Sensor وظیفه دریافت مشخصه هایی از محیط Percept

2. عملگر: Actuator وظیفه انجام اعمال بر روی محیط Action



• عامل وظیفه دارد رشته دریافت های ورودی را به دنباله هایی از اعمال نگاشت نماید

• بنابراین میتوان گفت عامل مانند یک تابع است

$$f : P^* \rightarrow A \quad \text{Action} = \text{Agent}(\text{Percept})$$

عمل (ها)

4 - Peas را برای ربات فضانورد و فوتبالیست تشریح کنید؟

معیارکارایی: برد بازی ، گل زدن بیشتر از تیم حریف ، بیرون نرفتن از چهارچوب بازی ، خطا نکردن روی بازیکن حریف ، سرعت جابجایی قابل قبول برای پشت سر گذاشتن بازیکن تیم حریف ، توانایی شناسایی و تشخیصی عوامل محیطی و فیزیکی بازی به راحتی مانند (حریف ، بازیکن خودی ، دروازه ها ، توپ) ، استفاده نکردن دست بجز دروازه بان

محیط : زمین چمن (مصنوعی،طبیعی) ، فوتسال ، ساحلی (شن)

عملگرها : شوت ، چپ ، سانتر کردن ،

پاس دادن ، گرفتن توپ توسط دست (دروازه بان) ، هد زدن ، تکل زدن

سنسور: سنسور تشخیص توپ ، سرعت توپ و جهت آن ، سنسور تشخیص فاصله ، سنسور آنالیز بازیکنان حریف ، سنسور خطوط (برای بیرون نرفتن توپ از محیط زمین) ، سنسور تشخیص بازیکن حریف/خودی / دروازه ها ، سنسور شناسایی عوامل بیرونی محیطی / فیزیکی ، سنسور دوربین ۳۶۰ درجه ، سنسور آب و هوا (برای لیز خوردن و تشخیص جهت باد قبل از شوت زدن)، سنسور افساید

5 - طبق شبکه کد زیر چرا عامل مبتنی بر جدول به شکست مواجه می شود؟ راهکارهای پیشنهادی خود را نام برده و مختصری در خصوص هر کدام توضیح دهید؟

عامل های مبتنی بر جدول Agents Based-Table ممکن است در مواجه با محیطهای پیچیده و پویا به شکست بخورند. این موضوع به دلیل محدودیتهایی است که در طراحی و استفاده از جداول به عنوان مکانیسم اصلی تصمیمگیری ایجاد میشود.

برای پیشگیری از این مشکلات و افزایش عملکرد عامل های مبتنی بر جدول، راهکارهای زیر پیشنهاد میشود :

1. استفاده از الگوریتم های یادگیری -جایگزین کردن جداول با الگوریتم های یادگیری مانند Learning-Q یا Deep Networks-Q که به عامل امکان یادگیری از تجربه و تطبیق با محیط را میدهند .

2. استفاده از الگوریتم های پویا - استفاده از الگوریتم هایی که قابلیت تطبیق با تغییرات محیط را داشته باشند، مانند الگوریتم های بازگشت به عقب (Backpropagation) در شبکه های عصبی .

3. ترکیب با الگوریتم های هوشمند - ترکیب جداول با الگوریتم های هوشمند به صورتی که جداول تنها به عنوان یک منبع اطلاعات مکمل استفاده شوند و عامل توانایی یادگیری و تطبیق را نیز داشته باشد .

4. استفاده از روش های بهینه سازی - استفاده از روش های بهینه سازی برای مدیریت بهتر جداول و کاهش حجم داده ها مانند فشرده سازی داده ها یا الگوریتم های بهینه سازی مانند تکنیک های مدیریت حافظه

6 - دنیای جارو برقی را با توجه به فرموله سازی مساله تشریح کنید؟

• حالت: 8 حالت مختلف

• حالت شروع: هریک از حالت

• اعمال: چپ، راست، مکش یا هیچ کار

• آزمون هدف: حالت 7 و 8

• هزینه مسیر: تعداد اعمال انجام شده تا رسیدن به هدف

7 - جستجوی عمقی را با رسم مرحله به مرحله شرح دهید و در نهایت کارایی الگوریتم را با چهار معیاراندازی گیری بیان کنید؟

• عمیق ترین گره بسط داده میشود .

• پیاده سازی fringe: از طریق یک صف LIFO پشته

• کامل بودن؟ خیر مگر اینکه فضای حالت محدود باشد و حلقه تکرار وجود نداشته باشد .

• بهینه بودن؟ خیر چون کامل نیست.

• پیچیدگی زمانی؟ $O(b^m)$

✓ اگر m خیلی بزرگتر از d باشد به مراتب بدتر است.

✓ در بسیاری از مسایل سریعتر از جستجوی BF است.

• پیچیدگی حافظه؟ $O(bm + 1)$

8 - ضمن بررسی الگوریتم جستجوی درختی شبکه کد زیر بررسی کنید که استراتژی در کدام از 4 توابع، پیاده سازی شده است، توابع را نام برده و عملکرد هر یک را بیان کنید؟

- روش کلی برای جستجوی آگاهانه
- جستجوی سطحی: گره برای بسط دادن بر اساس تابع ارزیابی $n(f)$ انتخاب میشود.
- ایده: تابع ارزیابی فاصله تا هدف را اندازه گیری میکند. یعنی گره ایی را برای بسط دادن انتخاب میکند که به نظر بهترین باشد.
- پیاده سازی fringe: یک صف مرتب بر اساس n

9 - شبکه کد زیر مربوط به کدام جستجوی ناآگاهانه می باشد و از مزایای کدام جستجوهای دیگر بهره برده است با ترسیم شکل توضیح دهید؟

- جستجوی عمقی تکراری
- یک استراتژی برای یافتن بهترین مقدار L
 - روشی برای ترکیب مزایای جستجوی BF و D
 - کامل بودن؟ بلی | به شرطی که حلقه تکرار وجود نداشته باشد.
 - بهینه بودن؟ بلی
 - اگر هزینه مسیرها باهم برابر باشد.
 - مثال: تعداد گره های تولید شده با جستجوی سطری و عمقی تکراری در یک مساله فرضی با فاکتور انشعاب 10 و عمق جواب بهینه 5 را بدست آورید.

10 - شش نوع جستجوهای ناآگاهانه جدول زیر را به تفکیک، با چهار معیار مربوطه به اختصار شرح دهید ؟

11 - جستجوی A^* را با توجه به جدول hSLD با جستجوی حریصانه search Greedy با رسم درختی به طور کامل توضیح داده و تفاوت ها را با دلیل ذکر کنید؟

جستجوی حریصانه: (Greedy Search)
این الگوریتم به هدف نزدیکترین گزینه در هر مرحله را انتخاب می‌کند بدون توجه به مسیری که تا این مرحله طی شده است.
از یک تابع هیوریستیک (heuristic function) استفاده می‌کند تا گزینه‌ای را انتخاب کند که به نظر می‌رسد به هدف نزدیکترین باشد. این انتخاب بر اساس تخمین فاصله تا هدف انجام می‌شود. بدون در نظر گرفتن گذشته یا مسیر کلی، تنها به دنبال نزدیکترین گزینه به هدف است.

A^* (A-star):
این الگوریتم از یک ترکیبی از هزینه و هیوریستیک برای انتخاب گزینه استفاده می‌کند.
از دو عامل استفاده می‌کند: هزینه واقعی از شروع تا هر گره و هیوریستیک که یک تخمین از هزینه باقی‌مانده تا هدف است.
با جمع کردن هزینه واقعی و تخمین هیوریستیکی، الگوریتم A^* می‌تواند به صورت بهینه‌تری به هدف برسد و در نتیجه مسیر بهینه‌ای را پیدا کند

12 - الگوریتم زیر را شرح دهید و با توجه به جدول و شکل سوال 11 با رسم درخت جستجو توضیح دهید؟

هر حالت state در حقیقت بیانگر یک حالت فیزیکی است .
• هر گره node نیز در حقیقت یک ساختمان داده متناظر با درخت است. هر گره دارای فرزند، پدر ، عمق و سایر مفاهیم درختان است .
• آرایه برگها FRINGE شامل گره‌های تولید شده اما بسط داده نشده در درخت است. گره های سفید

13 - چند نوع تابع هیوریستیک را می توان برای پازل اعداد معرفی کرد، با رسم شکل بررسی کنید؟

ابداع تابع هیوریستیک قابل پذیرش 1

• از طریق نسخه ساده شده از مساله `version relax`

h_1 هر کاشی می تواند به هر جایی منتقل شود.

h_2 هر کاشی می تواند به هر خانه همسایه منتقل شود `ABSolver`. هزینه راه حل برای مکعب روبیک را تخمین میزند

ابداع تابع هیوریستیک قابل پذیرش 2

• از طریق نسخه کوچکتر از مساله `subproblem`

ابداع تابع هیوریستیک قابل پذیرش 3

• از طریق یادگیری از `experience from learning` تجربه : حل تعداد بسیار زیادی از مساله

14 - سه راه حل جهت ابداع تابع هیوریستیک نام برده و شرح دهید؟

ابداع تابع هیوریستیک قابل پذیرش (1):

از طریق نسخه ساده شده از مساله (`relax version`)

h_1 هر کاشی میتواند به هر جایی منتقل شود.

h_2 هر کاشی می تواند به هر خانه همسایه منتقل شود.

`ABSolver` هزینه راه حل برای مکعب روبیک را تخمین میزند.

ابداع تابع هیوریستیک قابل پذیرش (2):

از طریق نسخه کوچکتر از مساله (`subproblem`)

ابداع تابع هیوریستیک قابل پذیرش(3):

از طریق یادگیری از تجربه (learning from experience)

تجربه : حل تعداد بسیار زیادی از مساله

15 - انواع جستجوی محلی را نام برده و ایده هر یک را بیان کنید؟

الگوریتم تپه نوردی - Simulated annealing - Local beam search - Genetic algorithms

الگوریتم تپه نوردی:

- به طور متناوب در جهت بهبود حرکت میکند. زمانی که به قله برسد متوقف میشود.
- تپه نوردی به آینده گره های برگت وجه نمیکند، به همین دلیل به الگوریتم جستجوی محلی حریصانه هم مشهور است.
- تپه نوردی در صورتیکه بیشتر از یک گره بهترین وجود داشته باشد، بهترین گره را به صورت تصادفی انتخاب میکند.

Simulated annealing:

اجتناب از گیر کردن در بیشینه های محلی با اجازه دادن به انجام حرکت های فرعی (نامناسب)، که در حین گذشت زمان احتمال و تعداد آن کاهش می یابد.

پیشینه الگوریتم به علوم متالورژی بر میگردد.

Local beam search:

از k حالت شروع به جای یک حالت شروع بهره میبرد.

حالت شروع : k حالت تصادفی

حالت بعدی : انتخاب k تا بهترین حالت از بین تمام برگ ها
حالت خاتمه : پیدا شدن هدف یا بررسی تمام حالات
تفاوت با تپه نوردی با شروع مجدد تصادفی این است که اطلاعات به اشتراک گذاشته میشود.
ممکن است الگوریتم از عدم تنوع کافی برخوردار باشد.

:Genetic algorithms

الگوریتم‌های ژنتیک یک نوع الگوریتم هوش مصنوعی هستند که از فرآیندهای تکاملی برگرفته شده‌اند و برای بهینه‌سازی و حل مسائل بهینه‌سازی یا جستجوی مسیر استفاده می‌شوند. این الگوریتم‌ها از ایده‌ها و اصول موجود در تکامل طبیعی برای بهبود جواب‌های مسئله استفاده می‌کنند.

16 - الگوریتم زیر را شرح داده و انواع آن را نام برده و بررسی کنید؟

الگوریتم تپه نوردی :

- به طور متناوب در جهت بهبود حرکت میکند. زمانی که به قله برسد متوقف میشود.
- تپه نوردی به آینده گره های برگت وجه نمیکند، به همین دلیل به الگوریتم جستجوی محلی حریصانه هم مشهور است.
- تپه نوردی در صورتیکه بیشتر از یک گره بهترین وجود داشته باشد، بهترین گره را به صورت تصادفی انتخاب میکند.

انواع الگوریتم تپه نوردی:

تپه نوردی غیر قطعی (Stochastic hill-climbing):

در بین حرکت های روبه بالا یکی به صورت تصادفی انتخاب شود. البته احتمال انتخاب با شیب متناسب است.

تپه نوردی با انتخاب اولین گزینه (First-choice hill-climbing):

گره ها تا حصول یک گره بهتر بسط داده میشوند.

تپه نوردی تصادفی (Random-restart hill-climbing):
از حالت شروع مجدد تصادفی تا حصول جواب مجدداً شروع خواهد نمود.

