

در محیط های تصادفی عملکرد با علامت گذاری تعریف شود که عامل بر اساس متوسط نتایج با احتمالات و وضعیت مختلف روبروی بکند. نوعی این محیط چون عوامل تصادفی و عدم قطعی است عامل نمی تواند همیشه به نتیجه درست برسد. برای همین معیار عملکرد به دست سنجش میزان دستاورد بعد از شرایط مختلف متغیر می شود

(۱)

دلیل این تعریف هم این است که عامل باید در جای انتظار یک نتیجه قطعی بماند و نه همه احتمالات در حالت مارو در نظر بگیرد و با وجود عدم قطعیت بهترین تصمیم را بگیرد

Tani Driving

- Partially ممکن است بهیچ وجه
- Single agent (در محیط های شلوغ multi)
- Stochastic (عامل تصادفی، شرایط نامعین)
- Continuous (حرکت پیوسته)
- Dynamic (محیط تغییر می کند)
- Sequential (تصمیمات متوالی با تأثیر)
- پوشش جبهه ها

chess with a clock

- fully observable همه داده ها می شوند
- multi-agent دوبازیکن
- Deterministic حرکات مشخص
- Discrete محدود و دورن
- Semi-d زمان پایان ولی خود بازی بیست
- sequential حرکات پیوسته

pacman (optimally)

- partially همه داده ها نمی شود
- multi agent خوش و روح
- Stochastic احتمال حرکت
- Discrete حرکت محدود
- Dynamic
- Sequential

pacman (Randomly)

- مدل PEAS ۱- محیط: میدان جنگ، یوگا با مدافع و دشمنان متحرک (۳) ارتباطات
- ۲- حرکتها: تصمیم برداری، راه، GPS، حاکمهای خود ۳- عاملها: مزبور، سنج، اجنبی - از منابع
- ۴- سنجها: عملکرد، وقت شناسایی اجنبی - از منابع، کارایی (نویزی)، فشار و زمان برای مدیریت

هنگام محیط طراحی قابل مشاهده اطلاعاتی رو کپی می کند که شاید خیلی دقیق باشد ولی
 از صفا با استفاده از آنکه خیلی قابل اعتماد نیست ولی استفاده می کنیم و وضعیت را از بالا
 می بینیم با استفاده از داده های تاریخی می توان الگوها و احتمالات رونمایی کنیم
 (۴) برای سنج بین هر تغییرات محیط و تصمیم گیری های آگاهانه تر به این ترتیب عامل بهتر است

(۱) F-1 ۲- فرزند و گمان (T) T-3 F-4 ۵- آنکه تابع consistency باشد

F-4 -v -

UCS: a c c e r f r y (b) DFS: a b c c e r f r y (a)
 BFS: a c d r y

(c) h_۲ نیست، b فرزند واقعی است که از تخمین گرفته (d)
 h_۱ قابل قبول
 a c c e r f r y c e

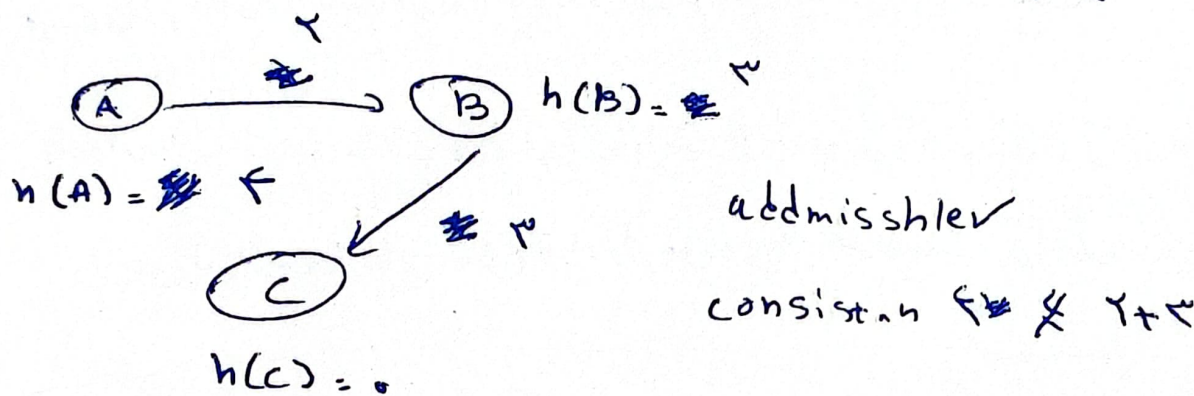
$$h(n) \leq c(n, n') + h(n')$$

(۸)

$$h(n) - h(n') \leq c(n, n')$$

آنگاه این نابرابری برای همگروه‌ها بنویسیم و ترکیب کنیم داریم:

$$\left. \begin{array}{l} h(n_1) - h(n_2) \leq c(n_1, n_2) \\ \vdots \\ h(n_{n-1}) - h(n_n) \leq c(n_{n-1}, n_n) \end{array} \right\} \Rightarrow h(n_1) - h(n_n) \leq c(n_1, n_n) \quad \checkmark$$

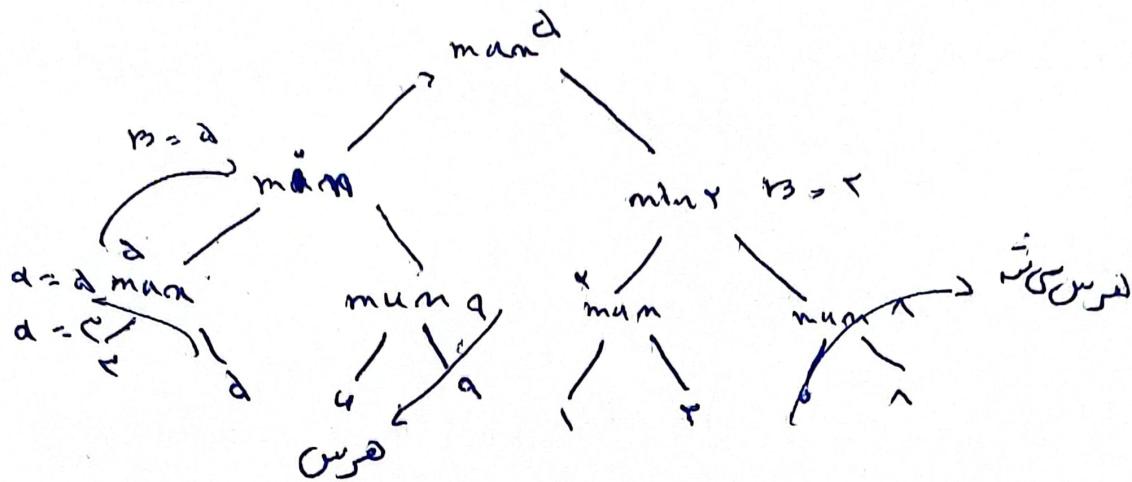


۱) الگوریتم فرض می‌کند که طرف مقابل منطقی فکر می‌کند اگر یک بازیکن باشد، الگوریتم ممکن است حرکات ضعیف انجام دهد و درجا خطا بشود

۲) در این بازی ممکن است به جای کاهش حریف بهترین حرکت افزایش دهد و باشد
نیاز این ~~minimum~~ ^{maximum} به جای کاهش باید به فکر سود خود باشد

۳) به جای حداکثر کردن سود یکی و کمتر کردن دیگری باید هر دو زیاد شوند ^{maximum}

۴) امکان پذیر نیست ولی با تغییراتی چون تصمیم‌گیری بر اساس منصفانه بودن است و نیاز به هوش متفاوتی دارد



11 minimum وقتی از حرکت ممکن در نظریه برای بار هرگز هم یک وضعیت بازی روشن می ده و اگر هم باید با وجود به به و نشود بران باز یک حرکت یابی و محاسبه کند

تابع ارزیابی چون حالت ماضی را یاد است برای همین نیست تمام حرکت ها و محاسبه کرد ولی می شه از معیارهایی مثل تعداد وارزش سوها وضعیت و کنترل مرکز خسته و ... از برای

Expectimax minimum با احتمال ادن ماهیت قطعی که دارد و در باتی های احتمالی نفی خورده ولی

باز از احتمالات استفاده می کند و تصمیم می گیره مثلا در همین باتی Expectimax در هرگز استاز حالت ممکن رو با احتمالات و قوش فرسی کند و باز یک می گیره این باز یک

12 تدی تصمیم های یابی می کنه

Expectimax در شرایط نامطمئن ~~خوب~~ و با نتایج احتمالی محاسبه می کنه

minimum پیچیدگی بیشتری داره چون باید همه حالت ها و محاسبه کند