

بنام خدا - محمد مجتبی صادقی - پروژه دوم هوش مصنوعی - شماره دانشجویی: ۱۹۹۴۵۳۸

نیمت ترکیب:

۱- با جمعیت اولیه بسیار کم، احتمال رسیدن به جواب بهینه کاهش می یابد.

با زیاد شدن سرعت محاسبه هر دوره، اختلاط می یابد و شاید نتوانم دوره های زیادی را اجرا کنم تا به نتیجه برسیم.

۲- دقت افزایش خواهد یافت اما سرعت به طرز چشمگیری در هر مرحله کاهش می یابد و این روش اصلاً پیشنهاد نمی شود.

۳- در Crossover، در پدر چشم ترکیب شده و دو فرزند جدید بوجود می آورند.



در mutation، به احتمالی برخی از کروموزوم ها جهش ژنتیکی دهند.



اگر فقط از یکی از این دو استفاده کنیم، ممکن است سنبل به یک لو سوتق دار شود و دیگر نتوان به جواب بهینه دست یافت.
داشتن هر دو نه الگوریتم های ترکیب حیاتی است.

۴- استفاده از heuristic مناسب، تعریف کروموزوم ها و در سرعت تأثیر دارند.

Scanned

۵- طه با وجود تدابیر که اندیشیده ام، این امکان ندارد تقریباً:

۱۵٪ جمعیت را از ۱۵٪ بهترین جمعیت هر یک بدر در نظر می گیریم.
۵۰٪ " " بصورت رندوم از کل جمعیت بدر در نظر می گیریم.
۳۰٪ جمعیت را بصورت رندوم از ۳۰٪ بهترین جمعیت هر یک بدر در نظر می گیریم.
۱۰٪ را هم خودم بصورت رندوم تولید و وارد جمعیت می کنم.

اگر تنوع را بین کورسوزم ها رعایت نکنیم، ممکن است تمام کورسوزم ها به یک سمت تمایل پیدا کنند و دیگر نتوانیم به جواب بهینه برسیم.

تابع $h_{unistic}$ که در نظر گرفته ام، برابر اختلاف جواب هر کورسوزم با جواب بهینه است و

۶- در هر راند، بهترین کورسوزم نزدیکتر به جواب بهینه را ذخیره می کنیم.

در صورت پیداشدن جواب، می توانیم از نزدیکترین کورسوزم استفاده کنیم.

پیدا شده به جواب استفاده می کنیم.

نمایش بازه

۱- heuristic خوب باید بتواند پیش بینی هایی کند که واقعیت نزدیکتر باشد.

همچنین باید بتواند Order نیز خوب کند بین State ها

مثلاً در State زیر را در نظر بگیرید:



State A باید heuristic بالاتر نسبت به B داشته باشد، چرا که حرف اول احتمالاً در دست ما است

همچنین State هایی که حرف اولی بزرگتر باید heuristic بسیار زیاد و اگر صاف برابریم بسیار کم باشد:

$$heuristic = (\text{blue-leaf-nodes} - 6 \times \text{red-leaf-nodes}) + 1000$$

$$\text{if (we win)} \quad heuristic \pm 50$$

$$\text{if (ble win)} \quad heuristic - 50$$

که heuristic من می تواند این را به خوبی توضیح کند.

۲- با افزایش حجم الگوریتم، تعداد گره ها که دیده شده در زمان Compute

افزایش می یابد.

نتایج شش حرکت پیروز بر یک ۱۰۰ بار با یک را هم با هم حرکت مختلف می‌تواند بسنید؟

$$Depth = 7 \Rightarrow \begin{cases} Red: 91 \\ Blue: 9 \end{cases}$$

$$Depth = 5 \Rightarrow \begin{cases} Red: 84 \\ Blue: 16 \end{cases}$$

$$Depth = 3 \Rightarrow \begin{cases} Red: 81 \\ Blue: 19 \end{cases}$$

$$Depth = 1 \Rightarrow \begin{cases} Red: 91 \\ Blue: 9 \end{cases}$$

تقریباً می‌توان گفت با افزایش عمق، درصد پیروز افزایش می‌یابد ولی این تقریبی است. چرا که مرتب ما دارو به صورت شش حرکت می‌کنیم. اگر خریف به صورت optimal اشیاء می‌گیریم، این روزها دست می‌شود.

معتبر بود برای درخت $min-max$ ، از درخت $alphaMax$ استفاده می‌کنیم. در اسیفیت به طور قطع می‌توانیم درصد پیروز را بسیار بهبود بخشیم.

۳- بدون استفاده از $Prune$ ، زمان خیلی طول می‌کشد به طور مثال برای $Depth = 5$ و 20 بار

اجرا، با $Prune$ 17s و بدون $Prune$ تقریباً 6 دقیقه اجرا طول می‌کشد.

شش پیروز با اضافه کردن $Prune$ هیچ تغییر نخواهد یافت و فقط سرعت \uparrow می‌یابد.

در هنگام $Prune$ ، ترتیب دیدن گره‌ها بسیار مهم است. بعد از انجام کارها، بسیار

موجب شدم بهترین حالت زمانی است که در گره‌ها Δ (Max) است.

Successor داریم بهترین $huristic$ و در گره‌ها Δ از کمترین شروع به بالا بیشترین رفتن

کنیم. دقت کنید هر دو با هم در یک هوس را در یک تابع بسیار ساده نمودم

Sort و Sort کردن در بین هوس است یعنی چون نیاز است و همه را خلاص می‌شود!