تمرین اول هوش مصنوعی

امیرحسین رجبی (۹۸۱۳۰۱۳)

۱۴ اسفند ۱۴۰۰

سوال اول

آ)

ب)

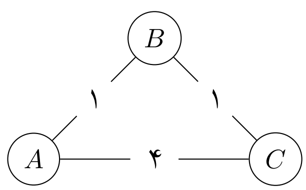
سوال دوم

آ) درست. بنابر توضیح کتاب در صفحه ۶۴ فصل دوم (از نسخه ۲۰۲۰) یک بازی ویدیویی جدید ممکن است صفحه بازی برای ما به طور کامل قابل مشاهده باشد ولی ندانیم که با فشردن هر دکمه چه اتفاقی خواهد افتد.

ب) درست. تابع transition در این مدل‌ها روی فضای متناهی تعریف می‌شود و در فضاهای پیوسته پاسخگو نیست.

ج) درست. اگر در الگوریتم A^* تابع هیوریستیک برابر تابع ثابت صفر باشد تابع ارزیابی (f) همان تابع هزینه (g) خواهد بود.

د) نادرست. گراف زیر را در نظر بگیرید. کوتاهترین مسیر از گره A به C از طریق B خواهد بود. اگر $C = ۳$ را به تمام یال‌ها اضافه کنیم، کوتاهترین مسیر یال AC خواهد بود.



ه) درست. می‌دانیم الگوریتم A^* همه گره‌ها مانند x را بسط می‌دهد که هزینه ارزیابی آنها حداکثر هزینه ارزیابی گره هدف باشد؛ یعنی اگر n گره هدف باشد، $f(x) \leq f(n) = g(n)$. اما چون $g(x) \leq g(x) + h(x) = f(x)$ پس $g(x) \leq g(n)$. بنابر شرایط گفته شده $g(n)$ بهینه است و در نتیجه گره x توسط الگوریتم Uniform Cost Search نیز بسط داده می‌شود چرا که این الگوریتم همه گره‌ها با هزینه کمتر از $g(n)$ را بسط می‌دهد.

و) نادرست. مشکل اصلی A^* پیچیدگی زمانی آن نیست بلکه پیچیدگی فضای مصرفی آن زیاد است. (همان طور که الگوریتم IDS برای کاهش فضای مصرفی الگوریتم BFS و completeness الگوریتم DFS ارائه شد.)

سوال سوم

با توجه به شماره دانشجویی مسئله چهارم را حل می‌کنم. درستی گزاره را ثابت می‌کنیم. فرض کنیم گره v مجاور گره u باشد و هزینه انتقال از u به v برابر c باشد. چون x و y تابع‌هایی consistent هستند، خواهیم داشت:

$$x(u) \leq c + x(v)$$
$$y(u) \leq c + y(v)$$

نامساوی اول را در α و نامساوی دوم را در $۱ - \alpha$ ضرب می‌کنیم: (دقت کنید به دلیل برقراری شرط $۰ < \alpha < ۱$ می‌توانیم بدون تغییر جهت نامساوی این کار را انجام دهیم.)

$$\alpha x(u) \leq \alpha c + \alpha x(v)$$
$$(۱ - \alpha)y(u) \leq (۱ - \alpha)c + (۱ - \alpha)y(v)$$

با جمع روابط بالا داریم:

$$\alpha x(u) + (۱ - \alpha)y(u) \leq c + \alpha x(v) + (۱ - \alpha)y(v)$$

که حکم نتیجه می‌شود.

سوال چهارم

مسئله اول را حل می‌کنیم.

آ) هر حالت قرارگیری n پکمن در m خانه نقشه یک حالت یا state خواهد بود. همچنین بین هر دو حالت reachable یک transition وجود دارد.

ب) با توجه به بخش قبل و فرض مسئله که امکان قرارگیری چندین پکمن در یک خانه از نقشه وجود دارد، می‌توان گفت اندازه فضای حالت برابر m^n خواهد بود چرا که هر پکمن می‌تواند در یکی از m خانه نقشه قرار گیرد.

ج) در صورتی که موقعیت همه پکمن‌ها به گونه‌ای باشد که بتوانند در هر چهار جهت بالا، پایین، چپ و راست حرکت کنند و همچنین بتوانند در جای خود باقی بمانند، تعداد یال‌های خروجی هر حالت برابر $b \leq ۵^n - ۱$ خواهد بود. (دقت کنید حداقل یک پکمن باید حرکت کند. در غیر این صورت درجا می‌زنیم. یک واحد به این دلیل کسر شده است.) به وضوح در صورت قرارگیری پکمن‌ها در حاشیه و مرز نقشه تعداد یال‌های خروجی آن حالت یا branching factor از b کمتر خواهد بود. همچنین کران پایین $b \geq ۲^n - ۱$ را نیز داریم چرا که هر پکمن یا یک خانه مجاور دارد یا می‌تواند در جای خود باقی بماند.

د) در این مسئله تابع هزینه گام‌های مسئله [1] عددی صحیح بین ۱ و n است چرا که حداقل یک پکمن یک گام حرکت کرده و همچنین حداکثر همه پکمن‌ها هر کدام یک گام حرکت می‌کنند. می‌دانیم در الگوریتم Uniform Cost Search حداکثر همه گره‌هایی بسط داده می‌شوند که مسیر بهینه آنها هزینه‌ای کمتر از پاسخ بهینه مسئله (C^*) را دارد. چون تابع هزینه حداقل $\epsilon = ۱$ است پس حداکثر همه گره‌ها تا عمق $d = \lfloor \frac{C^*}{\epsilon} \rfloor + ۱ = C^* + ۱$ [2] بسط داده می‌شوند؛ یعنی حداکثر $b + b^۲ + \cdots + b^d = \frac{b(b^d - ۱)}{b - ۱}$ گره. با توجه به بخش قبل b کران پایین بزرگی دارد و می‌توانیم فرض کنیم عبارت بالا از b^d کمتر است. از طرفی در این مسئله هر گره هدف حداکثر در عمق $m - ۱$ خواهد بود زیرا محدودیتی برای تعداد پکمن‌هایی که می‌توانند همزمان در یک خانه از نقشه قرار داشته باشند نداریم و هر پکمن با پیمایش این تعداد گام همه‌ی خانه‌های نقشه را یک بار مشاهده می‌کند. همچنین تابع هزینه هر گام حداکثر n خواهد بود. پس $C^* \leq n(m - ۱)$. به کمک بخش قبل داریم $b^d \leq (۵^n - ۱)^{n(m-۱)+۱}$.

[1] Step cost function

[2] دلیل وجود یک در این عبارت این است که الگوریتم UCS زمانی خاتمه می‌یابد که گره هدف را از صف اولویت خارج کند و قبل از بسط آن، هدف بودن آن را بررسی می‌کند و نه زمانی که آن را ایجاد می‌کند.

ه) این تابع هر دو شرایط را دارد. ابتدا ثابت می‌کنیم تابع admissible است؛ یعنی تابع هیوریستیک، از هزینه واقعی بیشتر نیست. اگر مستطیل $R = [X_۱, X_۲] \times [Y_۱, Y_۲]$ مستطیلی با کمترین مساحت باشد که همه پکمن‌ها درون یا روی آن قرار بگیرند، (چپ‌ترین پکمن روی خط $x = X_۱$، راست‌ترین پکمن روی خط $x = X_۲$، پایین‌ترین پکمن روی خط $y = Y_۱$ و بالاترین پکمن روی خط $y = Y_۲$ قرار می‌گیرند.) تابع هیوریستیک برابر $h = \frac{۱}{۲}\max(X_۲ - X_۱, Y_۲ - Y_۱)$ خواهد بود. بدون کاسته شدن از کلیت مسئله اگر $h = \frac{۱}{۲}(X_۲ - X_۱)$ و همچنین جواب بهینه برای این مسئله، ملاقات پکمن‌ها در نقطه (α, β) باشد، در این صورت اگر $\alpha \leq \frac{X_۱+X_۲}{۲}$، فاصله منهتن [3] پکمن روی خط $x = X_۲$ حداقل h خواهد بود و اگر $\alpha \geq \frac{X_۱+X_۲}{۲}$، فاصله منهتن پکمن روی خط $x = X_۱$ حداقل h خواهد بود. به طوری مشابه برای $h = \frac{۱}{۲}(Y_۲ - Y_۱)$ نیز ثابت می‌شود که h کران پایینی برای هزینه واقعی حل مسئله خواهد بود.

اکنون ثابت می‌کنیم تابع هیوریستیک consistent است. اگر v گره‌ی مجاور گره‌ی u باشد و هزینه انتقال از u به v برابر c باشد، نشان می‌دهیم $h(u) \leq h(v) + c$. مستطیل‌های R_u و R_v را مانند اثبات قبل به ترتیب برای وضعیت‌های u و v تعریف می‌کنیم. (مستطیل‌هایی با کمترین مساحت و اضلاع موازی محورهای مختصات که پکمن‌ها درون یا روی آنها قرار دارند.) همچنین اندازه طول مستطیل R را با $L(R)$ و اندازه عرض آن را با $W(R)$ نشان می‌دهیم. بدون کاسته شدن از کلیت مسئله فرض کنیم $L(R_u) \geq W(R_u)$ و در نتیجه $h(u) = \frac{L(R_u)}{۲}$. چون طول گام پکمن‌ها یک واحد است، $۲ \geq |L(R_u) - L(R_v)|$ و $۲ \geq |W(R_u) - W(R_v)|$ چرا که پکمن‌های روی طول یا عرض R_u می‌توانند به سمت یکدیگر یا خلاف همدیگر حرکت کنند. در نتیجه $۱ \geq |h(u) - h(v)|$. اگر $h(u) - h(v) = ۱$، چون $c \geq ۱$ آنگاه $h(u) \leq h(v) + c$. در غیر این صورت (مثلا $h(v) \geq h(u)$) رابطه $h(u) < h(v) + c$ برقرار خواهد بود. پس در هر صورت داریم $h(u) \leq h(v) + c$.

گزارش پروژه

[3] Manhattan distance