

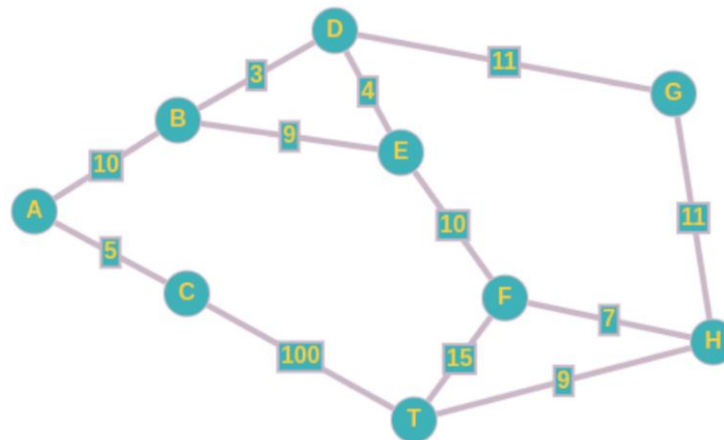
تمرین اول درس هوش مصنوعی

دانشکده مهندسی برق و کامپیوتر

فرید سیاهکلی ۸۱۰۱۹۸۵۱۰



سوال اول الگوریتم Uniform Cost Search:

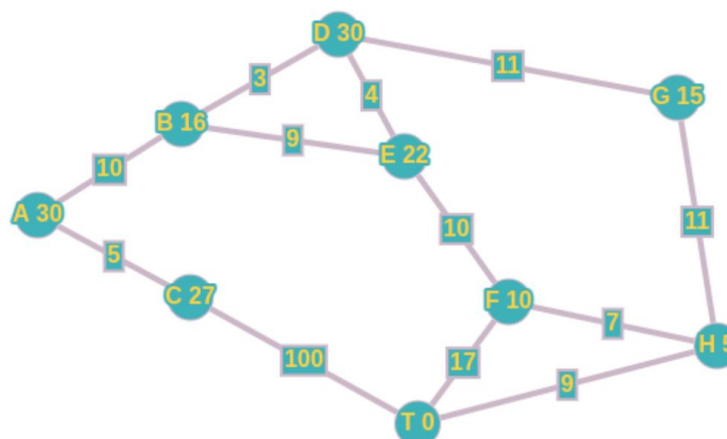


Frontier: {A, B₁₀, C₅, T_{100→44}, D₁₃, E₁₉, G₂₄, F₂₉, H₃₅}

Explored: {A, C, B, D, E, G, F, H, T}

→ **Cost:** 44, **Path:** A → B → D → G → H → T

الگوریتم (A*) Informed Search:



Frontier: $\{A, B_{26}, C_{32}, E_{41}, D_{43}, T_{105 \rightarrow 46}, F_{39}, H_{41}, G_{62 \rightarrow 39}\}$

Explored: $\{A, B, C, E, F, H, D, G, T\}$

→ **Cost:** 46, **Path:** $A \rightarrow B \rightarrow E \rightarrow F \rightarrow T$

تابع Heuristic مورد استفاده admissible است و همواره داریم:

$$Cost(Node) > h(node)$$

تابع Heuristic مورد استفاده Consistent نیست و در میان دو گره E و F داریم:

$$Cost(E, F) < h(E, F)$$

سوال دوم (Color Puzzle)

الف) یک تابع Heuristic غیربدیهی می تواند تعداد تضادهای رنگ در هر جدول باشد. تضاد رنگ زمانی رخ می دهد که دو کاشی با رنگ های مختلف در مجاورت یکدیگر قرار گیرند. این Heuristic را می توان به صورت زیر محاسبه کرد.

برای هر جدول، تعداد جفت کاشی های مجاور که رنگ های متفاوتی دارند را بشماریم. مجموع این تعداد برای همه جداول مقدار اکتشافی است. به طور شهودی، مقدار اکتشافی تخمینی از نزدیک بودن وضعیت فعلی به حالت هدف می دهد، جایی که هر جدول فقط یک رنگ کاشی دارد.

ب) برای اثبات Admissible بودن این Heuristic، باید نشان دهیم که هرگز هزینه واقعی رسیدن به حالت هدف را بیش از حد برآورد نمی کند. این به این دلیل است که یک Admissible Heuristic ضمیم می کند که راه حل بهینه توسط الگوریتم جستجو پیدا می شود. ابتدا توجه داشته باشید که هر حرکت یک کاشی حداکثر می تواند یک تضاد رنگ را حل کند. بنابراین، حداقل تعداد حرکات مورد نیاز برای حل تمام تضادهای رنگی برابر با تعداد تضاد رنگ است. همچنین توجه داشته باشید که هر بار که یک کاشی به جدول دیگری منتقل می شود، حداکثر می تواند تضاد یک رنگ را در آن جدول حل کند.

بنابراین، حداقل تعداد حرکات مورد نیاز برای حل تمام تضادهای رنگ در یک جدول برابر با حداکثر تعداد تضاد رنگ در آن جدول است. بنابراین، حداقل تعداد حرکات مورد نیاز برای حل تمام تضادهای رنگ در همه جداول با حداکثر تعداد تضاد رنگ در هر جدول برابر است.

مقدار Heuristic که ما پیشنهاد کردیم، مجموع تعداد تضادهای رنگ در هر جدول است. از آنجایی که مقدار اکتشافی یک کران بالای حداقل تعداد حرکات مورد نیاز برای حل همه تضادهای رنگی است، همچنین یک اکتشافی Admissible است.

برای اثبات Consistency، باید نشان دهیم که هزینه تخمینی انتقال از حالت فعلی به حالت جانشین به اضافه هزینه تخمینی رسیدن به حالت هدف از حالت جانشین کمتر یا برابر با هزینه تخمینی رسیدن است. وضعیت هدف از وضعیت فعلی.

فرض کنید $h(S)$ مقدار Heuristic برای حالت فعلی S باشد و $h(S')$ مقدار Heuristic برای حالت جانشین S' باشد. فرض کنید کاشی را از جدول $T1$ در حالت S به جدول $T2$ در وضعیت S' منتقل می کنیم و اجازه می دهیم " d " تعداد تضادهای رنگ در $T2$ باشد که با این حرکت حل می شود.

هزینه تخمینی حرکت از S به " S " است، زیرا این تعداد تضادهای رنگی است که با حرکت حل می شود. هزینه برآورد شده برای رسیدن به حالت هدف از $h(S')$ است.

برای تخمین هزینه رسیدن به حالت هدف از S ، توجه داشته باشید که پس از انتقال از $T1$ به $T2$ ، تعداد تضاد رنگ در $T1$ حداقل به میزان " d " کاهش می یابد. بنابراین، حداکثر تعداد تضاد رنگ در هر جدول در حالت جانشین حداکثر حداکثر تعداد تضاد رنگ در هر جدول در حالت فعلی منهای " d " است. از این رو، هزینه برآورد شده برای رسیدن به حالت هدف از S حداکثر $h(S) - d$ است.

بنابراین، ما داریم:

$$h(s) \geq d + h(s')$$

$$h(s) \geq h(s') + d$$

که تعریف سازگاری را برآورده می کند.

سوال سوم CSP (برنامه ریزی دروس دانشکده)

الف) متغیرها:

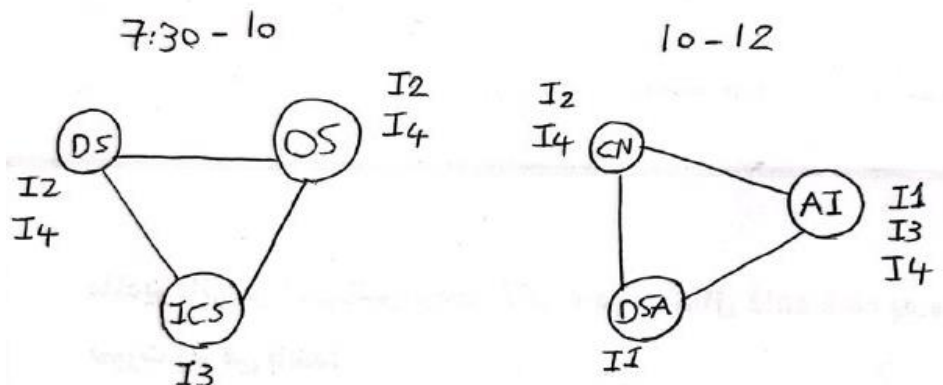
- Introduction to Computer Science (ICS)
- Data Structures and Algorithms (DSA)
- Distributed Systems (DS)
- Computer Networks (CN)
- Operating Systems (OS)
- Artificial Intelligence (AI)

دامنه متغیرها:

Time slots: 7.5-9, 8-10, 10-11.5, 10.5-12

Instructors: Dan Alistarh (L₁), Matteo Maffei (L₂), Bryan Ford (L₃), Robert Morris (L₄)

ب) گراف:



ج) دو گراف مستقل داریم چرا که می توان دروس را به دو دسته مستقل از لحاظ زمانبندی دسته بندی کرد. برای گراف اول که شامل دروس ساعت ۷:۳۰ تا ۱۰ است داریم:

- (ICS): {I3}
- (OS): {I2, I4}
- (DS): {I2, I4}

برای گراف دوم که مرتبط به دروس بین ساعات ۱۰ تا ۱۲ است داریم:

- (CN): {I2, I4}
- (DSA): {I1}
- (AI): {I1, I3, I4}

حال قیود را بررسی کرده و در نتیجه برای دو درس ICS و DSA تنها یک استاد وجود داشته که دامنه الباقی دروس را محدود می‌کند. برای گراف اول داریم:

- (ICS): {I3}
- (OS): {I2, I4}
- (DS): {I2, I4}

برای گراف دوم داریم:

- (CN): {I2, I4}
- (DSA): {I1}
- (AI): {I3, I4}

د) می‌توان دید که چند جواب برای این زمان بندی وجود دارد. یک برنامه ریزی می‌تواند به صورت زیر باشد:

- (ICS): I3
- (OS): I2
- (DS): I4
- (DSA): I1
- (CN): I4
- (AI): I3

سوال چهارم (تصحیح سوالات امتحان)

الف) اعمال قیود Unary و بدست آوردن دامنه محدود شده برای هر متغیر:

| | |
|----------------------|--------------|
| Amin (A): | {5} |
| Amir (AM): | {1, 2, 3} |
| Paria (P): | {1, 2, 3} |
| Taha (T): | {4, 6} |
| Negar (N): | {1, 2, 3} |
| Kianoush (K): | {1, 3, 5} |
| Omid (O): | {1, 2, 3, 4} |

ب) متغیری که مقدار آن ابتدا باید اختصاص متغیر امین است چرا که بیشترین قیود را میان متغیرها دارا است که باعث محدودیت دامنه شده است. در نتیجه اختصاص مقدار به امین فضای سرچ را به خوبی محدود کرده و اختصاص سوال به بقیه را ساده تر می کند. همچنین از آنجایی که سوال ۶ تنها در دامنه طاهها است، این متغیر نیز مقدار ۶ به خود می گیرد. حتی می توان استنتاج کرد که با مقداردهی طاهها، تنها متغیری که حالا می تواند مقدار ۴ بگیرد نیز مشخص شده و امید این سوال را تصحیح می کند.

ج) اگر بخواهیم به نگار یک سوال را با الگوریتم Value Constraining Least اختصاص دهیم، باید از میان سوالات ۱ و ۲ و ۳ آن سوالی را انتخاب کنیم که کمتر دامنه باقی افراد را محدود کند. سوال ۱ در دامنه ۴ فرد دیگر، سوال ۲ در دامنه ۳ فرد دیگر و سوال ۳ نیز در دامنه ۴ فرد دیگر قرار دارد. پس سوال ۲ به نگار اختصاص داده می شود.

د) راست به چپ یعنی از امین شروع کرده و هر بار متغیری که باعث محدودیت کمتری برای باقی متغیرها می شود را به هر فرد اختصاص می دهیم. به تقسیم بندی زیر می رسیم:

| | |
|----------------------|-----|
| Amin (A): | {5} |
| Amir (AM): | {2} |
| Paria (P): | {3} |
| Taha (T): | {6} |
| Negar (N): | {1} |
| Kianoush (K): | {3} |
| Omid (O): | {4} |

ه) از چپ به راست یعنی از امید این بار شروع کرده و هربار بزرگترین متغیر باقی مانده در دامنه را به فرد اختصاص می‌دهیم. به تقسیم بندی زیر خواهیم رسید:

Amin (A): {5}

Amir (AM): {1}

Paria (P): {1}

Taha (T): {6}

Negar (N): {2}

Kianoush (K): {3}

Omid (O): {4}