

بسمه تعالی

هوش مصنوعی

مسائل ارضاء محدودیتها - ۲

نیمسال اول ۱۴۰۳-۱۴۰۲

دکتر مازیار پالهنک

آزمایشگاه هوش مصنوعی

دانشکده مهندسی برق و کامپیوتر

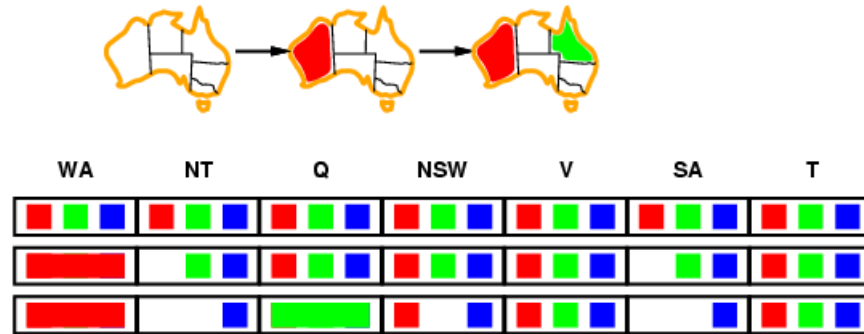
دانشگاه صنعتی اصفهان

یادآوری

- متغیرها، دامنه ها، محدودیتها
- انتساب سازگار
- انتساب کامل، جزئی
- مثال رنگ آمیزی نقشه
- گراف محدودیت
- حل بصورت یک مسئله جستجو
- تنوع متغیرها
- تنوع محدودیتها
- جستجوی عقبگرد
- متغیر محدود شده بیشینه
- متغیر محدود کن بیشینه
- مقدار محدود کن کمینه
- چک جلو

انتشار محدودیت

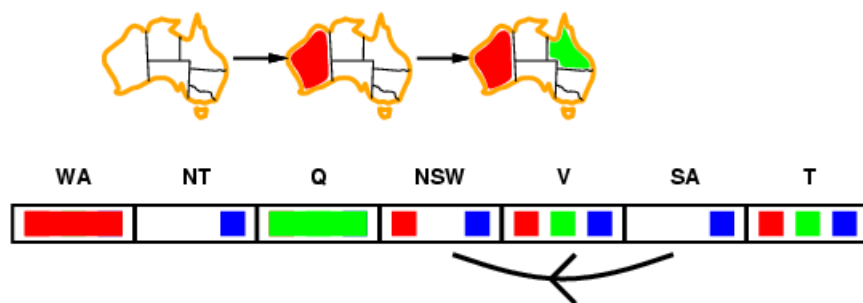
- چک جلو اطلاعات را از متغیرهای انتساب شده به نشده انتقال می دهد ولی اجازه تشخیص همه ناسازگاریهای زود هنگام را نمی دهد.



- NT و SA نمی توانند هر دو آبی باشند.

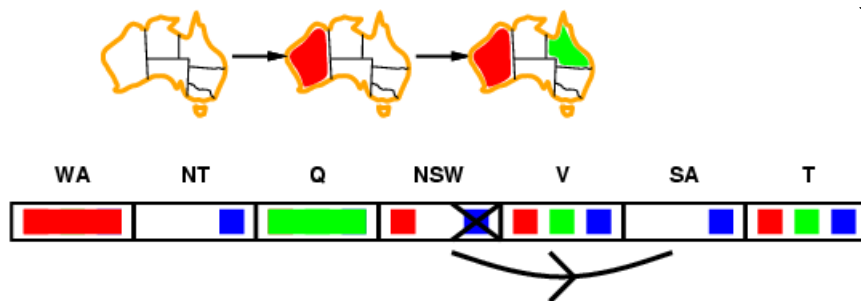
سازگاری کمان

- کمان منظور یالی است در گراف محدودیت
- کمان $X \rightarrow Y$ سازگار گفته می شود اگر برای هر مقدار X در دامنه X مقدار Y در دامنه Y وجود داشته باشد که با آن سازگار باشد.
- مثال کمان سازگار



سازگاری کمان

- کمان منظور یالی است در گراف محدودیت
- کمان $X \rightarrow Y$ سازگار گفته می شود اگر برای هر مقدار X در دامنه X مقدار Y در دامنه Y وجود داشته باشد که با آن سازگار باشد.
- مثال کمان ناسازگار

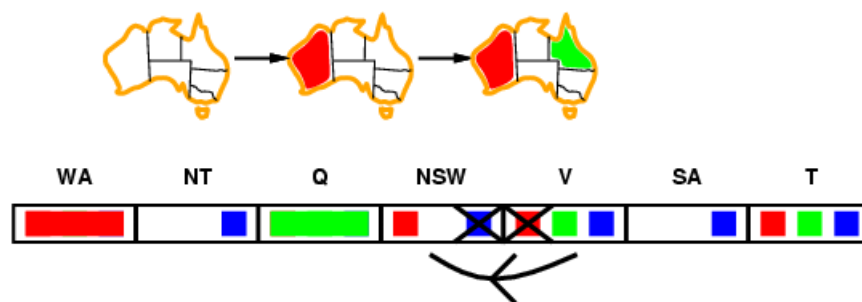


سازگاری کمان

- سعی کن همه کمانها را سازگار کنی
- اگر سعی کنیم NT و SA را سازگار کمان کنیم، به دامنه تهی می
رسیم.
- سازگاری کمان می تواند قبل از جستجو به عنوان پیش پردازش
اعمال شود.
- یا بعد از هر انتساب

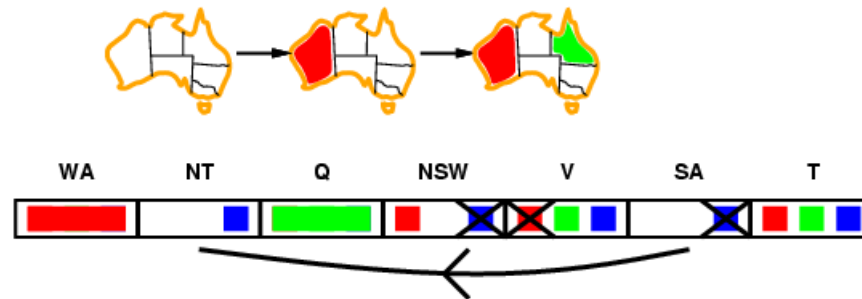
سازگاری کمان

- اگر X مقداری را از دست داد لازم است همسایگان آن چک شوند.
- چون حذف آن مقدار می تواند سازگاری کمان قبلی را از بین ببرد.



سازگاری کمان

■ سازگاری کمان شکست را زودتر از چک جلو تشخیص می دهد.



سازگاری کمان

function AC-3(*csp*) **returns** false if an inconsistency is found and true otherwise
queue \leftarrow a queue of arcs, initially all the arcs in *csp*

while *queue* is not empty **do**
 (X_i, X_j) \leftarrow POP(*queue*)
 if REVISE(*csp*, X_i, X_j) **then**
 if size of $D_i = 0$ **then return** false
 for each X_k **in** X_i .NEIGHBORS - $\{X_j\}$ **do**
 add (X_k, X_i) to *queue*
return true

function REVISE(*csp*, X_i, X_j) **returns** true iff we revise the domain of X_i
revised \leftarrow false
for each x **in** D_i **do**
 if no value y in D_j allows (x, y) to satisfy the constraint between X_i and X_j **then**
 delete x from D_i
 revised \leftarrow true
return *revised*

سازگاریهای دیگر

■ سازگاری رأس:

- یک متغیر (متناظر با یک رأس در گراف م.ا.م.) سازگار-رأس است اگر همه مقادیر دامنه آن محدودیتهای یکتائی را ارضا کنند.
- مثلاً اگر مردم SA رنگ سبز را دوست ندارند باید از دامنه آن حذف شود.
- یک گراف سازگار-رأس نامیده می شود اگر همه رئوس آن سازگار-رأس باشند.

سازگاریهای دیگر

- سازگاری مسیر:
- سازگاری کمان امکان دارد دامنه متغیرها را به یک عضو کاهش داده و پاسخ مسئله را بیابد،
- یا دامنه برخی متغیرها بدون عضو بماند و مشخص شود که مسئله بدون پاسخ است.
- اما همواره برخی مشکلات را متوجه نمی شود.
- مثلاً اگر در مسئله رنگ آمیزی نقشه دامنه هرمتغیر دارای دو رنگ {آبی، سبز} باشد.
- گراف سازگاری کمان دارد ولی مسئله حل شدنی نیست.

سازگاریهای دیگر

- دو متغیر $\{X_i, X_j\}$ نسبت به متغیر X_m سازگار-مسیر هستند اگر:
- برای هر انتساب $\{X_i=a, X_j=b\}$ که با محدودیتهای (X_i, X_j) سازگار است، انتسابی برای X_m وجود داشته باشد که محدودیتهای (X_i, X_m) و (X_m, X_j) را ارضا کند.
- بطور مثال تلاش برای سازگاری مسیر (WA, SA) نسبت به NT در حالتی که دامنه متغیرها دارای دو رنگ $\{\text{آبی و قرمز}\}$ باشد.
- دو انتساب سازگار $\{WA=\text{red}, SA=\text{blue}\}$ یا $\{WA=\text{blue}, SA=\text{red}\}$
- هیچ رنگی برای NT باقی نمی ماند.

سازگاریهای دیگر

■ سازگاری - k :

■ یک م.ا.م. سازگار- k است اگر برای هر انتساب سازگار به $k-1$ متغیر، یک انتساب سازگار برای هر متغیر k ام وجود داشته باشد.

■ سازگاری-۱ همانند سازگاری رأس

■ سازگاری-۲ همانند سازگاری کمان

■ سازگاری-۳ همانند سازگاری مسیر

■ یک م.ا.م. **سازگار- k -قوی** است اگر سازگار- k ، سازگار- $k--$ 1، سازگار- $k-2$ ،، تا سازگار-۱ باشد.

سازگاریهای دیگر

■ محدودیتهای جهانی:

■ همانگونه که قبلاً اشاره شد یک محدودیت جهانی شامل محدودیتی است که شامل چند متغیر (نه الزاماً همه) گردد.

■ بطور مثال Alldiff

■ روشی ساده برای تشخیص ناسازگاری در Alldiff:

■ اگر m متغیر در محدودیت شرکت دارند و اگر آنها روی هم دارای n مقدار متفاوت هستند و $m > n$ ، در این صورت محدودیت قابل ارضاء نیست.

سازگاریهای دیگر

- منجر به یک راهکار ساده:
- هر متغیر تک مقدار را از محدودیت حذف نمائید،
- مقدار آن متغیرها را از دامنه دیگر متغیرها حذف نمائید،
- مادامی که متغیر تک مقداری باقی مانده این کار را تکرار کنید،
- هر زمان یک دامنه تهی تشخیص داده شد، یا تعداد متغیرها بیش از تعداد مقادیر باقی مانده بود یک ناسازگاری تشخیص داده شده است.
- بطور مثال: در مسئله رنگ آمیزی WA،SA،NT با دو رنگ ناسازگاری تشخیص داده می شود (سه متغیر و دو رنگ).

جستجوی عقبگرد

function BACKTRACKING-SEARCH(*csp*) **returns** a solution or *failure*
 return BACKTRACK(*csp*, { })

function BACKTRACK(*csp*, *assignment*) **returns** a solution or *failure*
 if *assignment* is complete **then return** *assignment*
 var ← SELECT-UNASSIGNED-VARIABLE(*csp*, *assignment*)
 for each *value* **in** ORDER-DOMAIN-VALUES(*csp*, *var*, *assignment*) **do**
 if *value* is consistent with *assignment* **then**
 add {*var* = *value*} to *assignment*
 inferences ← INFERENCE(*csp*, *var*, *assignment*)
 if *inferences* ≠ *failure* **then**
 add *inferences* to *csp*
 result ← BACKTRACK(*csp*, *assignment*)
 if *result* ≠ *failure* **then return** *result*
 remove *inferences* from *csp*
 remove {*var* = *value*} from *assignment*
 return failure

جستجوی محلی برای ارضاء محدودیتها

- جستجوهای محلی از حالت کامل استفاده می کنند.
- در این حالت وضعیتی که همه متغیرها مقدار دارند.
- برای اعمال به م.ا.م.
- اجازه داشتن حالاتی که محدودیتها ارضا نشده اند.
- تغییر مقدار متغیرها (در جهت ارضاء کردن محدودیتها)
- انتخاب متغیر: بصورت تصادفی هر متغیر ناسازگار انتخاب شود.
- انتخاب مقدار با مکاشفه کمترین برخورد:
- انتخاب مقداری که کمترین محدودیتها را می شکند.

Figure 6.9

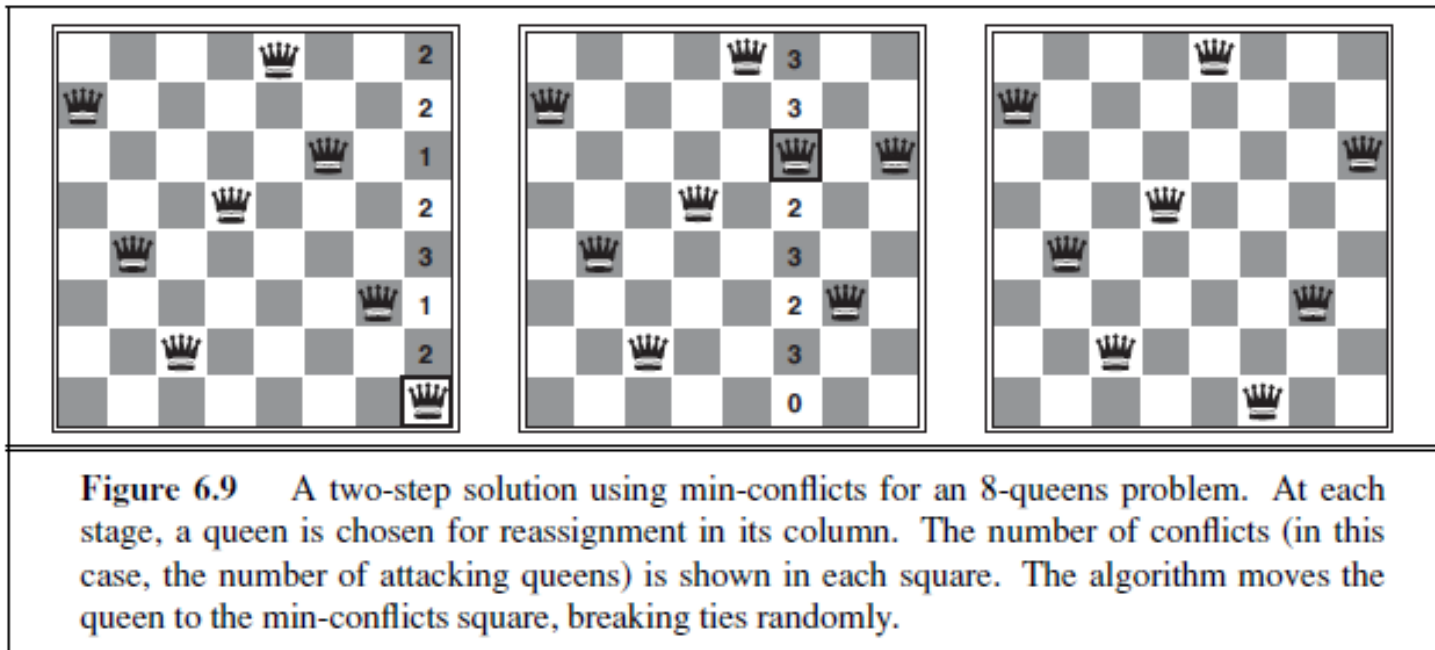
function MIN-CONFLICTS(*csp*, *max_steps*) **returns** a solution or *failure*
 inputs: *csp*, a constraint satisfaction problem
 max_steps, the number of steps allowed before giving up

 current \leftarrow an initial complete assignment for *csp*
 for *i* = 1 to *max_steps* **do**
 if *current* is a solution for *csp* **then return** *current*
 var \leftarrow a randomly chosen conflicted variable from *csp*.VARIABLES
 value \leftarrow the value *v* for *var* that minimizes CONFLICTS(*csp*, *var*, *v*, *current*)
 set *var* = *value* in *current*
 return *failure*

The MIN-CONFLICTS local search algorithm for CSPs. The initial state may be chosen randomly or by a greedy assignment process that chooses a minimal-conflict value for each variable in turn. The CONFLICTS function counts the number of constraints violated by a particular value, given the rest of the current assignment.

مثال: ۸ وزیر

- حالات: ۸ وزیر در ۸ ستون (8^8 حالت)
- انتخاب تصادفی یک متغیر
- انتخاب مقداری برای آن متغیر با کمترین تضاد
- امکان استفاده تا چند میلیون وزیر!



خلاصه

- سازگاری کمان
- معرفی سازگاریهای دیگر،
- الگوریتم جستجوی عقبگرد بصورت کاملتر
- جستجوی محلی



- دقت نمائید که پاورپوینت ابزاری جهت کمک به یک ارائه شفاهی می باشد و به هیچ وجه یک جزوه درسی نیست و شما را از خواندن مراجع درس بی نیاز نمی کند.
- لذا حتماً مراجع اصلی درس را مطالعه نمائید.
- در تهیه اسلایدها از سایت کتاب استفاده شده است.