به نام خدا

هوش مصنوعی

فصل پنجم

مسائل ارضای محدودیت

CONSTRAINT SATISFACTION PROBLEM

مقدمه :

- این فصل مسائل ارضای محدودیت را بررسی میکند. در این مسائل، حالتها و آزمون هدف از یک نمایش استاندارد، ساختیافته و ساده پیروی می کند
- میتوان الگوریتمهای جستجویی تعریف کرد که از امتیاز این نمایش خاص بهره برده و از توابع هیورستیک همه منظوره به جای توابع هیورستیک خاص مسئله در حل مسایل استفاده کنند.
- نمایش استاندارد آزمون هدف، ساختار مسئله را آشکار می سازد
- به این ترتیب روش هایی برای تجزیه مسئله و نیز روش هایی برای درک میزان ارتباط بین ساختار یک مسئله و سختی حل آن به دست می آیند.

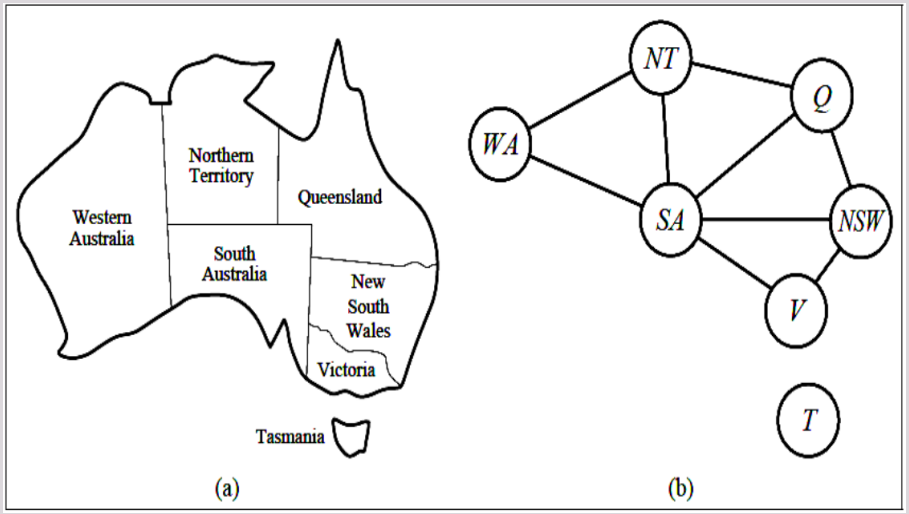
مسایل ارضای محدودیت :

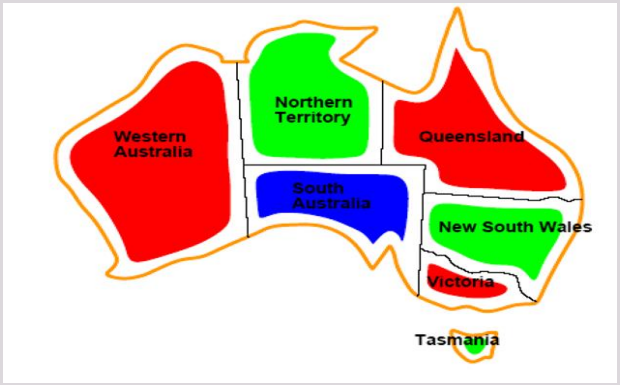
- مسائل ارضای محدودیت(csp) توسط مجموعه متناهی ای از متغیر ها : x1 ,x2 , x3 , … ,xn و مجموعه متناهی ای از محدودیت ها : c1 , c2 , c3 ,… , cn تعریف می شوند .
- هریک از متغیر ها دارای دامنه غیرتهی Di از مقادیر ممکن می باشد .
- هر محدودیت Ci شامل تعدادی زیرمجموعه از متغیر هاست . به طوریکه آن محدودیت، مقادیر ترکیبات مجاز این زیر مجموعه ها را مشخص میکند.
- هر حالت با انتساب مقادیر به چند متغیر یا تمام آنها تعریف می شود بنابراین در حالت اولیه هیچ یک از متغیرها مقدار ندارند .

مسایل ارضای محدودیت :

- انتسابی که هیچ محدودیتی را نقض نکند، انتساب قانونی یا سازگار نامیده میشود
- انتساب کامل انتسابی است که در آن همه متغیرها مقدار گرفته باشند.
- راه حل در مسئله CSPیک انتساب کامل است که تمام محدودیت ها را ارضا نموده است. یعنی بعضی از CSPها به راه حل هایی نیاز دارند که تابع هدف را بیشینه می کند .

مسئله نقشه استرالیا:





- در فرموله سازی این مسئله به صورت CSP برای هرناحیه یک متغیر درنظرگرفته میشود. (). WA, NT, Q, SW,N, V, SA, T
- دامنه هرمتغیر، مجموعه { red , green , blue} است .
- محدودیت این است که نواحی همجوار همرنگ نباشند
- مثال برای ترکیب های مجاز برای NT , WA :
{(red , green) , (red , blue) , (green , blue) , (blue , red) , (blue , green) , (green , red)}
- در گراف راس ها با متغیر های مسئله و یال ها با محدودیت ها متناظر هستند .

مسایل ارضای محدودیت :

- یک مسئله CSPمیتواند با استفاده از فرمولسازی افزایش مانند یک مسئله جستجوی استاندارد ارائه شود:
  - حالت اولیه: انتساب خالی { } که در آن هیچ متغیری مقدار ندارد.
  - تابع پسین: یک مقدار میتواند در هر متغیر فاقد مقدار نسبت داده شود اگر با متغیرهایی که قبلاً مقدار گرفتهاند، تضاد نداشته باشد.
  - آزمون هدف: آیا انتساب فعلی، کامل است؟
  - هزینه مسیر: یک هزینه ثابت (مثلاً ۱ (برای هر مرحله
- هر راه حل یک انتساب کامل است. اگر در مسئله nمتغیر وجود داشته باشد، راه حل در عمق nخواهد بود و درخت جستجو نیز دارای عمق nمیباشد. با توجه به دلایل فوق جستجوهای عمقی برای CSPها مناسب اند .
- ساده ترین نوع CSPشامل متغیرهای گسسته با دامنه های متناهی می باشد.

مسایل ارضای محدودیت :

- اگر در یک مسئله CSP،حداکثر اندازه دامنه هر متغیر برابر با dباشد، تعداد انتسابهای کامل ممکن ($O(d^n)$ است (n تعداد متغیر هاست) .
- مسایل CSPبا دامنه متناهی، مسایل CSPبولی را شامل میشود. در مسایل CSPبولی، متغیرها درست یا نادرست (false/true)هستند. CSPهای بولی شامل مسایل (NP-complete)هستند. لذا انتظار نمیرود در بدترین حالت پیچیدگی زمانی حل مسایل CSPبا دامنه متناهی غیرنمایی باشد.

انواع محدودیت ها در csp:

- **محدودیت یکتا (یکانی) :**
  - سادهترین نوع، محدودیت
  - به مقدار یک متغیر محدود میشود.
  - هر محدودیت یکتا میتواند به سادگی با انجام پیشپردازشی روی دامنه متغیر متناظر، حذف شود
  - برای مثال استرالیاییهای جنوبی رنگ سبز را دوست ندارند SA != GREEN

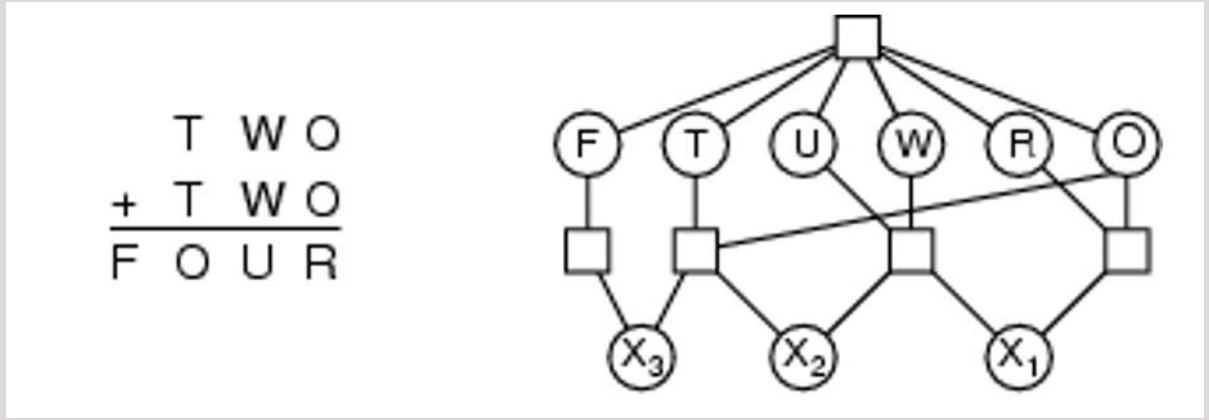
- **محدودیت دودویی :**
  - با دو متغیر مرتبط است.
  - بهطورمثال SA =! NSW یک محدودیت دودویی است
  - CSP دودویی:آن است که فقط محدودیت دودویی دارد و می توانند مانند گراف موجود در اسلاید ۶ نمایش داده شود.

انواع محدودیت ها در csp:

- محدودیت های مرتبه ی بالاتر :
  - شامل ۳ یا چند متغیر.
  - اگر متغیرهای کمکی کافی معرفی شوند،هر محدودیت مرتبه ی بالاتر و با دامنه ی متناهی می تواند به مجموعه ای از محدودیت ها ی دودویی کاهش یابد.

- محدودیت اولویت :
  - نشان می دهد کدام راه حل ارجح تر است.

مسائل csp (مثال رمزنگاری):



$$O + O = R + 10 . X_1$$
$$X_1 + W + W = U + 10 . X_1$$
$$X_2 + T + T = O + 10 . X_2$$
$$X_3 = F$$

متغیر ها : F,T,U,W,R,O,X1,X2,X3

متغیر های کمکی : X1,X2,X3

محدودیتها: F,T,U,R,O,W مخالفند (یک محدودیت ۶ متغیره)

جستجوی عقبگرد مسایل ارضای محدودیت :

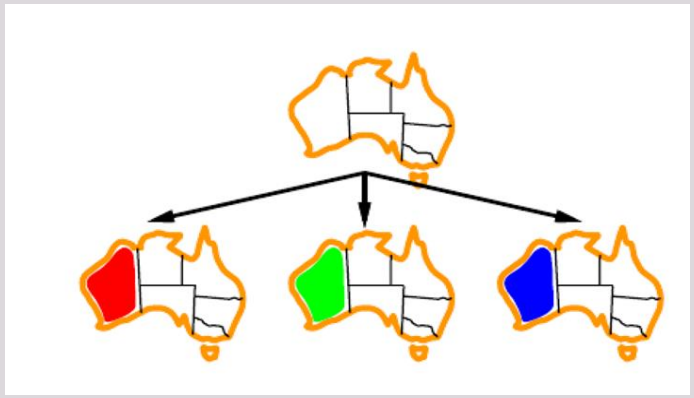
- اصطلاح جستجوی عقبگرد برای جستجوی عمقیای بهکار میرود که در هر سطح یک متغیر را مقدار میدهد و وقتی مقدار معتبری برای انتساب یک متغیر وجود نداشته باشد، به عقب برمیگردد.
- یک الگوریتم نا آگاهانه است و برای مسائل بزرگ کارآمد نیست .
- یکی از خواص مهم مسائل ارضای محدودیت ، خاصیت جا به جایی پذیری است .
- **جابه جایی پذیری :** یک مسئله وقتی جابجاییپذیر است که ترتیب بهکارگیری اعمال تأثیری درنتیجه حل ندارد. بنابراین تمام الگوریتمهای خاص CSP،در هر سطح فقط یک متغیر را مقداردهی میکنند. بنابراین فاکتور انشعاب در هر سطح d خواهد بود

```
function: BCACKTRACKING – SEARCH (CSP) returns a solution, or failure
return RECURSIVE – BACKTRACKING ({ } , CSP)
function: RECURSIVE – BACKTRACKING (assigument, CSP) returns a solution, or failure if assignment is complete then
return assignment
var ← SELECT – UNASSIGNED – VARIABLE (VARIABLES [CSP], assignment, CSP)
for each value in ORDER – DOMAIN – VALUSE (var, assignment, CSP) do
add {var = value} to assignmetn
result ← RECURSIVE – BACKTRACKING (assignment, CSP)
if result ≠ failure then retur result
remove {var = value} from assignment
return failur
```

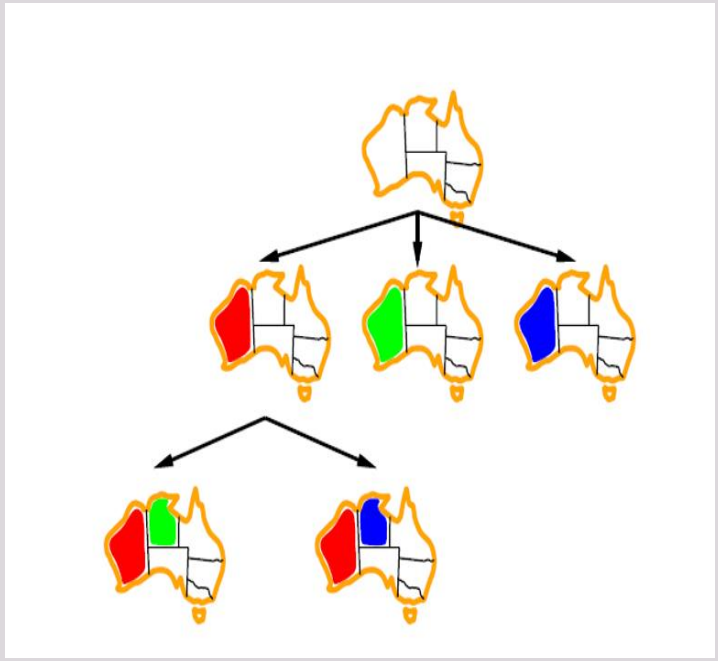
شکل ۵-۳: الگوریتم جستجوی عقبگرد ساده برای مسایل ارضای محدودیت

مثال جستوجو عقبگرد برای csp :

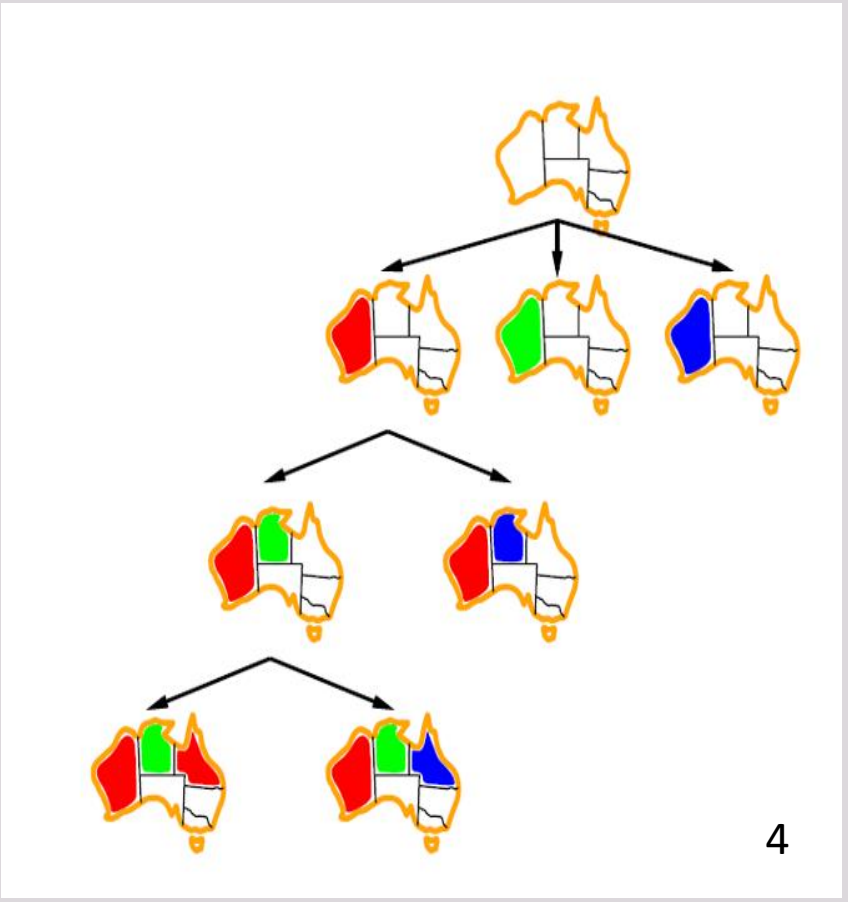
1



2



3



4

سه تابع اکتشاف برای csp :

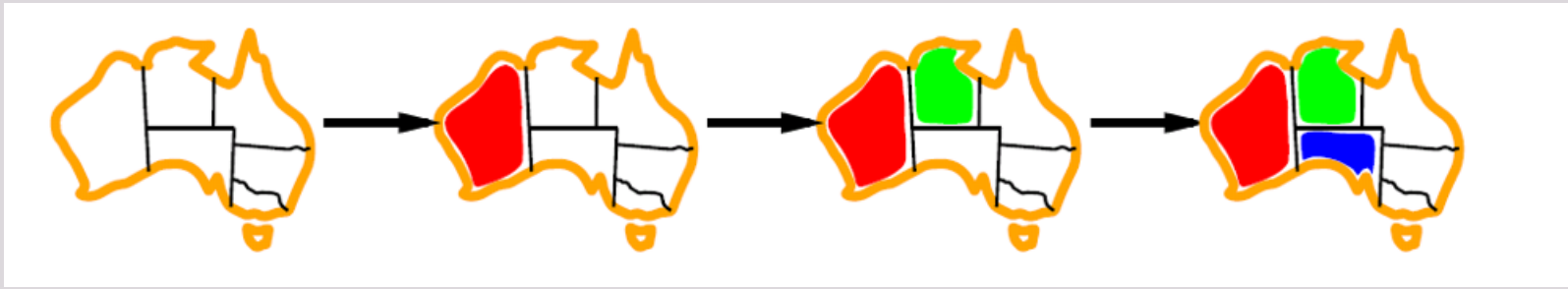
- اکتشاف مقادیر باقی مانده کمینه (MRV)
- اکتشاف درجه ای
- اکتشاف مقداری با کمترین محدودیت

اکتشاف مقادیر باقی مانده کمینه (MRV):

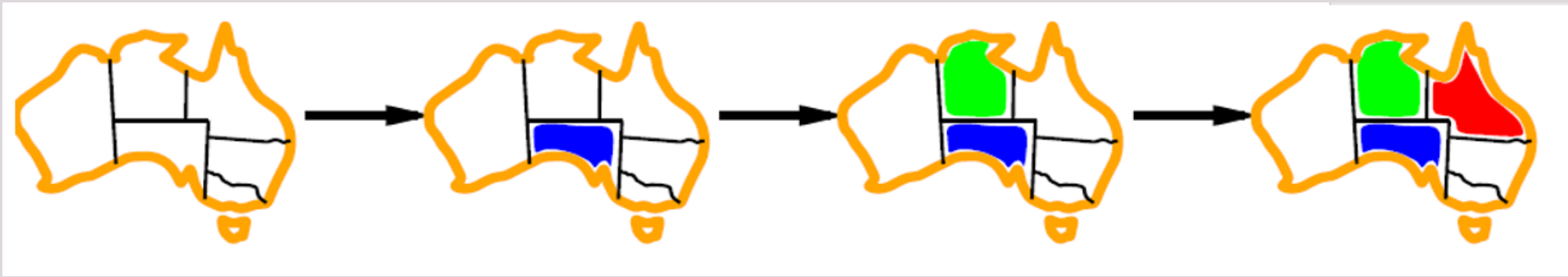
- انتخاب متغیری با کمترین مقادیر معتبر
- هیورستیک محدودترین متغیر یا هیورستیک اولین شکست نیز نامیده میشود
- درواقع محدودترین متغیر، متغیری است که کوچکترین دامنه را داراست
- متغیری انتخاب میشود که به احتمال زیاد، بزودی با شکست مواجه شده و در نتیجه درخت جست و جو را هرس میکند
- اگر برای متغیر Xمقدار معتبری باقی نمانده باشد، هیورستکی MRVمتغیر Xرا انتخاب میکند و با شکست مواجه میشود.
- هیورستیک MRVدر انتخاب اولین ناحیهای که باید رنگ شود (برای مثال در رنگآمیزی نقشه استرالیا) کمکی نمیکند زیرا در آغاز هر ناحیه سه رنگ معتبر دارد (در ابتدا همه متغیرها یکسان است).

اکتشاف درجه ای :

- هیورستیک درجه سعی میکند فاکتور انشعاب را برای انتخابهای آینده کم کند.
- بدینمنظور متغیری را انتخاب مینماید که بیشترین محدودیت را روی متغیرهای انتساب نیافته ایجاد میکند
- درجه هر متغیر تعداد محدودیت هاییست که روی دیگر متغیرها ایجاد می کند .
- در مسئله نقشه استرالیا متغیر SA دارای درجه ۵ است یعنی روی ۵ متغیر دیگر محدودیت ایجاد می کند، متغیر T دارای درجه صفر و سایر متغیرها دارای درجه ۲ یا ۳ هستند. درواقع اگر ابتدا SAانتخاب شود، بکارگیری هیورستیک درجه، این مسئله را بدون هیچ مرحله نادرستی حل میکند



اکتشاف مقادیر باقی مانده کمینه



اکتشاف درجه ای

اکتشاف مقداری با کمترین محدودیت :

- این روش مقداری را برای یک متغیرترجیح میدهد که نسبت به سایر مقادیر، از همه کمتر منجر به کاهش انتخابهای متغیرهای همسایه آن متغیر در گراف محدودیت میشود.
- به عنوان مثال فرض کنید green = NT و red = WA انتخاب بعدی Q است. مقداردهی Q با رنگ آبی مناسب نیست زیرا آخرین مقدار معتبری که برای همسایه Q یعنی SA باقی میماند، حذف میکند. هیورستیک مقدار با کمترین محدودکنندگی، قرمز را بر آبی ترجیح میدهد.
- سعی میکند بیشترین قابلیت انعطاف را برای انتساب متغیرهای بعدی ایجاد کند

پخش اطلاعات از طریق محدودیتها :

- بررسی پیشرو
- پخش محدودیت سازگاری یال
- پخش محدودیت سازگاری k

بررسی پیشرو (forward checking):

- وقتی متغیر Xمقدار میگیرد، فرآیند بررسی پیشرو، هر متغیر مثل Yرا که از طریق یک محدودیت به Xمتصل است، درنظر میگیرد و هر مقداری را که با مقدار انتخاب شد ه برای Xتضاد دارد، از دامنه Yحذف میکند.
- با هر سه اکتشاف ذکر شده می توان در نظر گرفت.

نمونه : مراحل جستجوی بررسی پیشرو در مسئله رنگآمیزی نقشه. ابتدا WA = redانجام میشود سپس بررسی پیشرو redرا از دامنه متغیرهای همسایه یعنی SA، NTحذف میکند. پس از Q = greenمقدار greenاز دامنه، SA، SW, NTو پس از V = blueاز دامنه های SA، SWحذف میشود. درنتیجه SAفاقد مقدار میگردد.

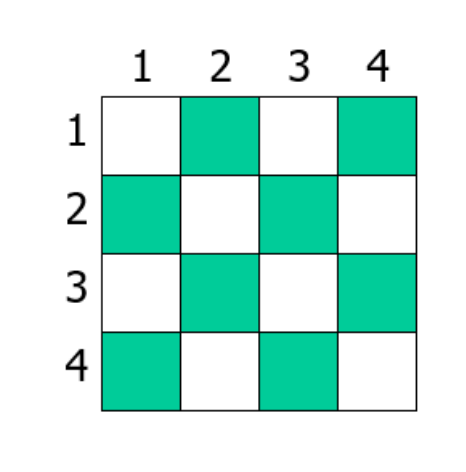
	WA	NT	Q	NSW	V	SA	T
Initial domains	R G B	R G B	R G B	R G B	R G B	R G B	R G B
After WA=red	(R)	G B	R G B	R G B	R G B	G B	R G B
After Q=green	(R)	B	(G)	R B	R G B	B	R G B
After V=blue	(R)	B	(G)	R	(B)		R G B

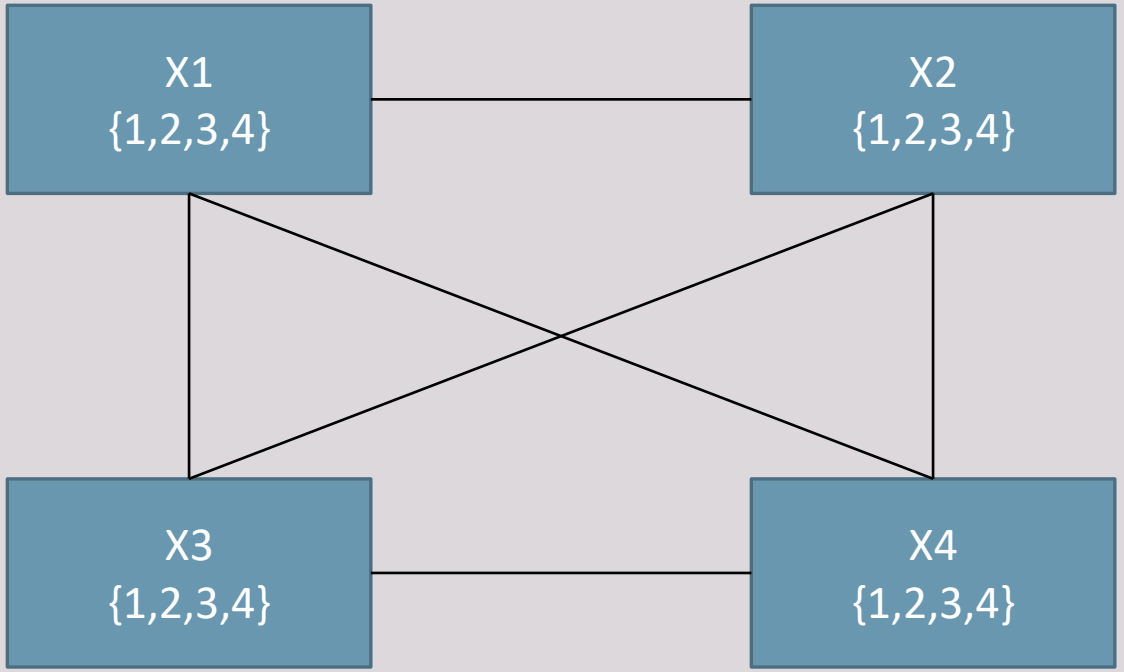
بررسی پیشرو (forward checking):

- هیروستیک MRVکه شریک خوبی برای جستجوی بررسی پیشرو است، SAو NTرا انتخاب میکند
- جستجوی بررسی پیشرو، یک روش کارآمد برای محاسبه اطلاعات موردنیاز هیورستیک MRVاست

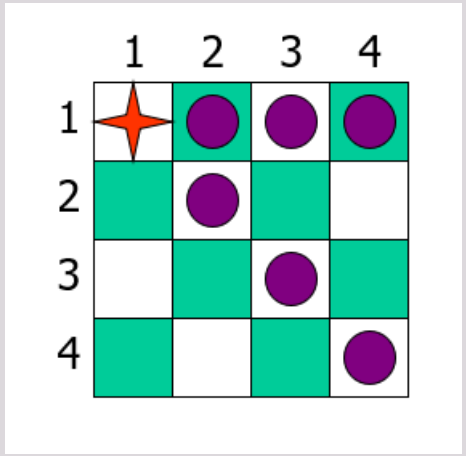
- . نکته دوم اینکه پس از انتساب blue = Vدامنه SAخالی میشود. لذا جستجوی بررسی پیشرو تشخیص میدهد که انتساب {v=blue , q=green , wa = red}با محدودیتهای مسئله سازگار نیست و الگوریتم به سرعت عقبگرد میکند. این جستجو وجود تناقض را سریعتر از جستجوی عقبگرد ساده تشخیص . میدهد

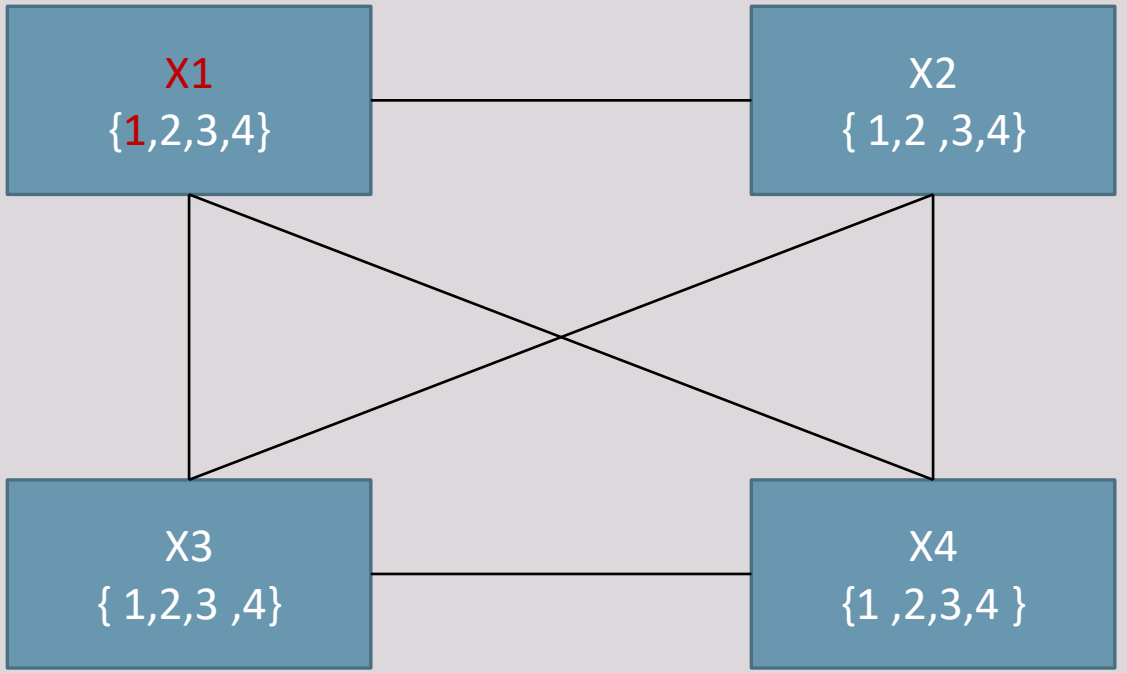
مثال ۸ وزیر (بررسی پیشرو) :



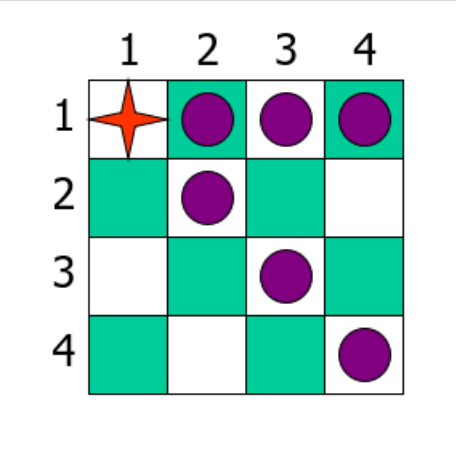


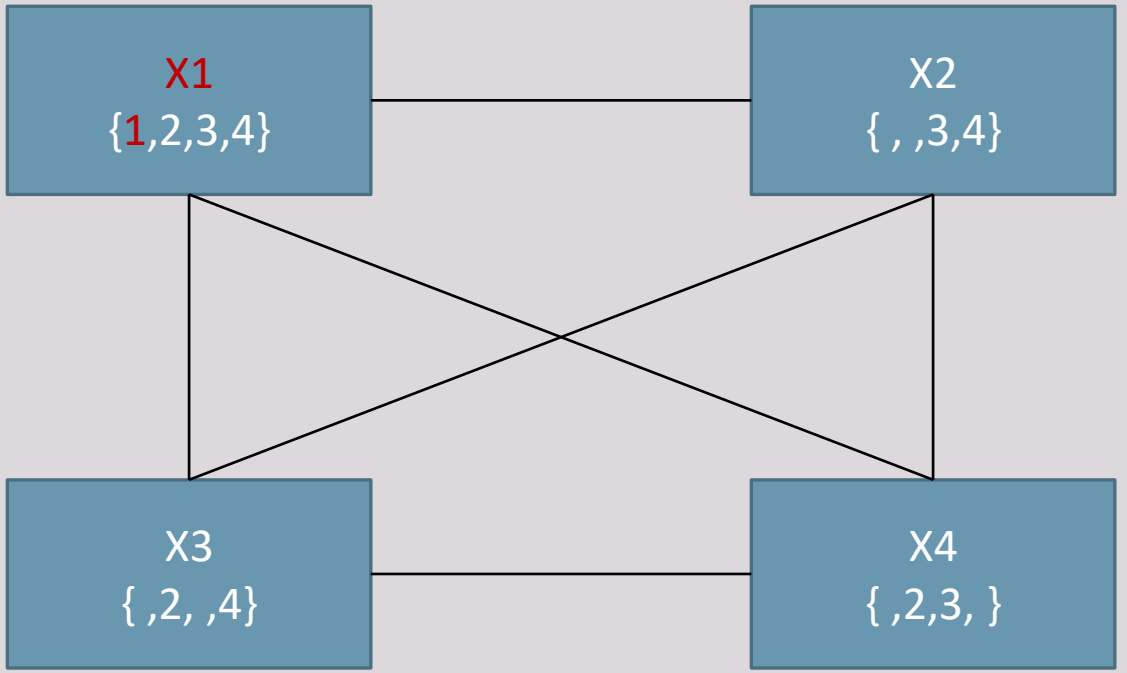
مثال ۸ وزیر (بررسی پیشرو) :



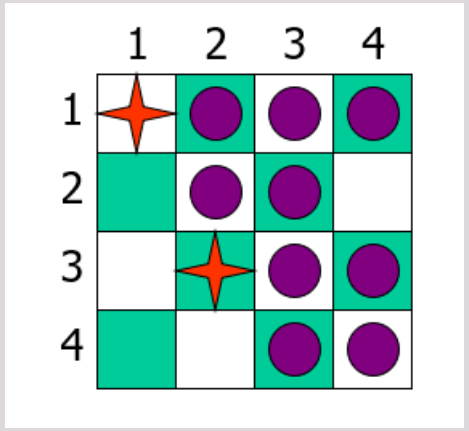


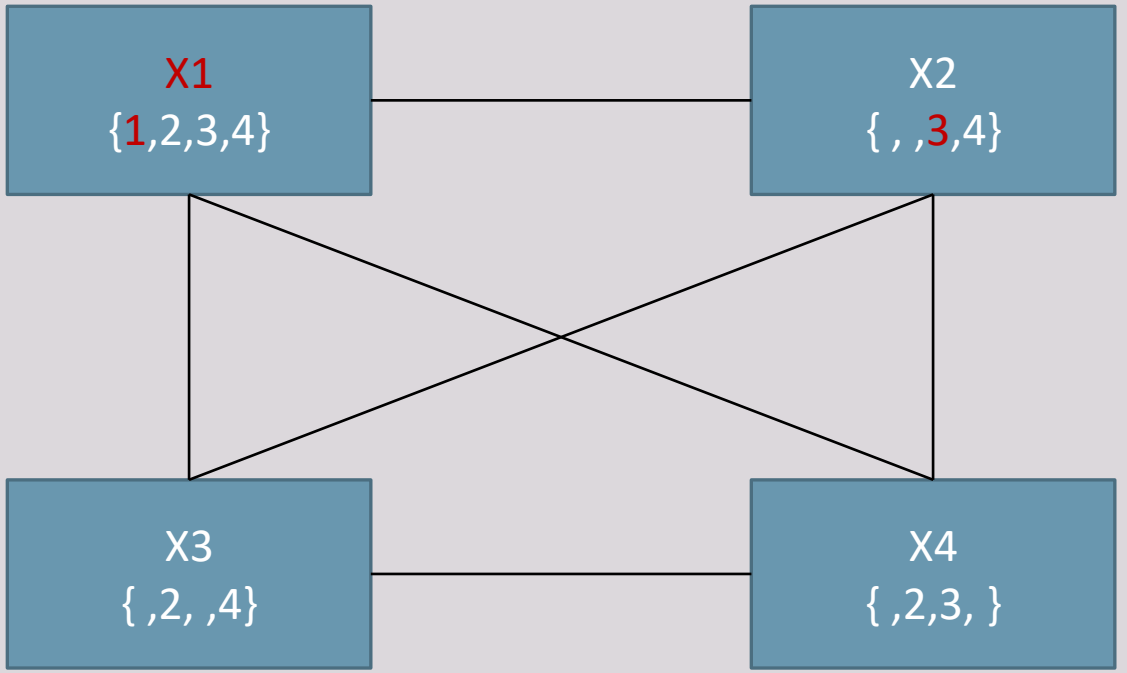
مثال ۸ وزیر (بررسی پیشرو) :



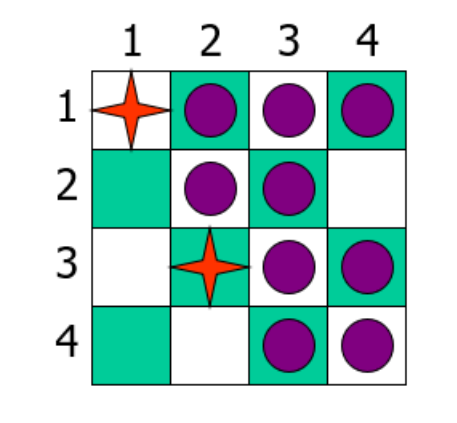


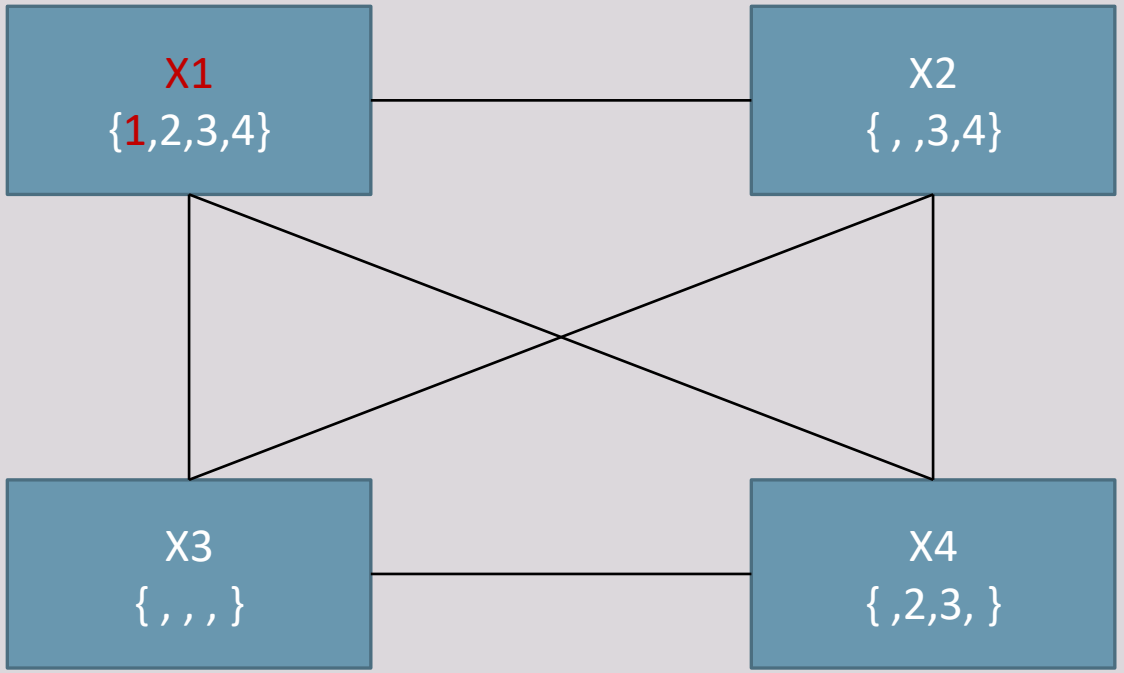
مثال ۸ وزیر (بررسی پیشرو) :



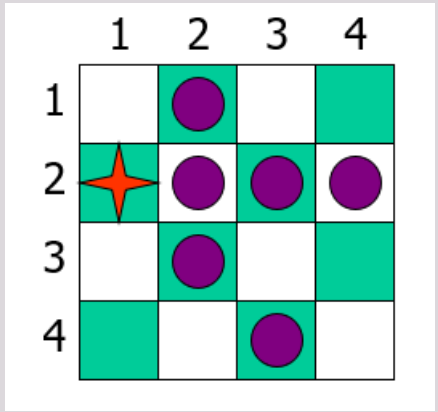


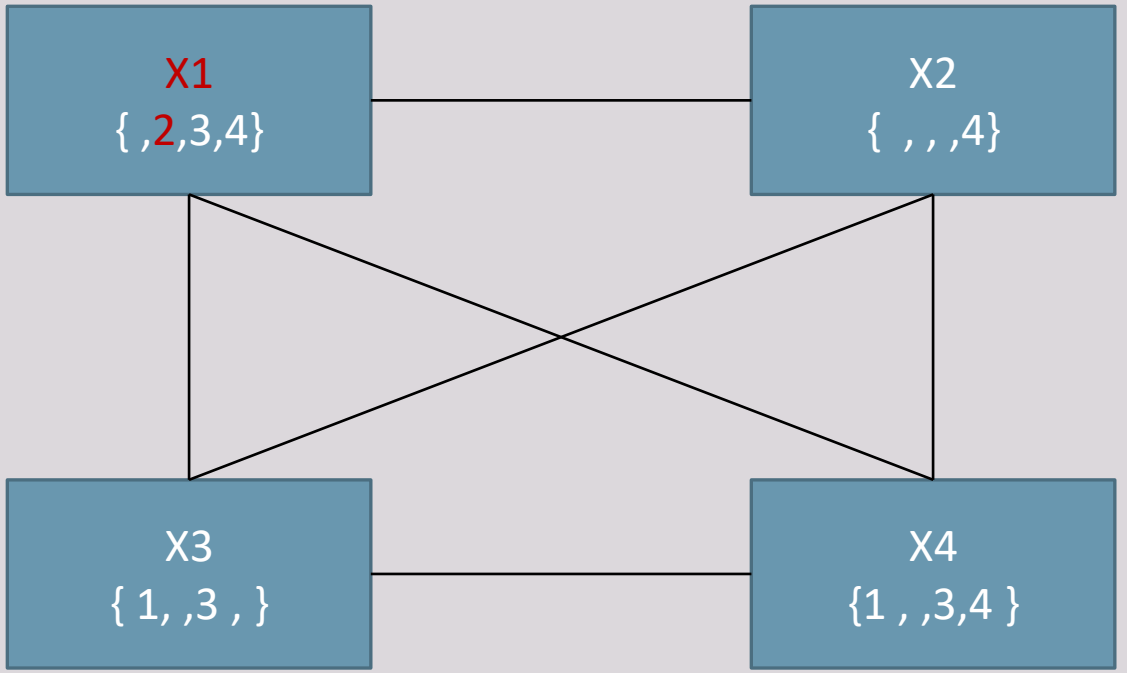
مثال ۸ وزیر (بررسی پیشرو) :



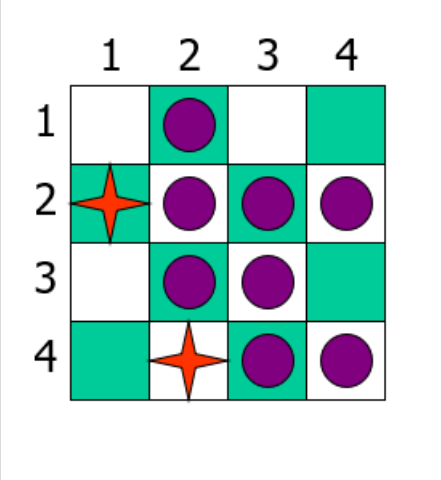


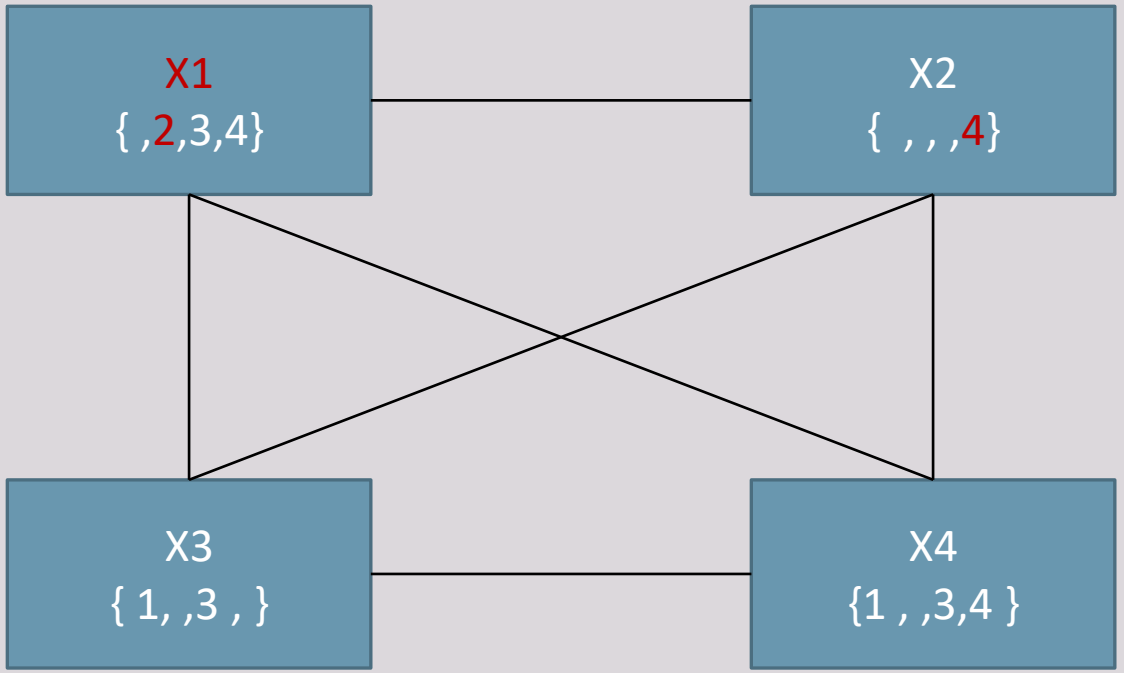
مثال ۸ وزیر (بررسی پیشرو) :



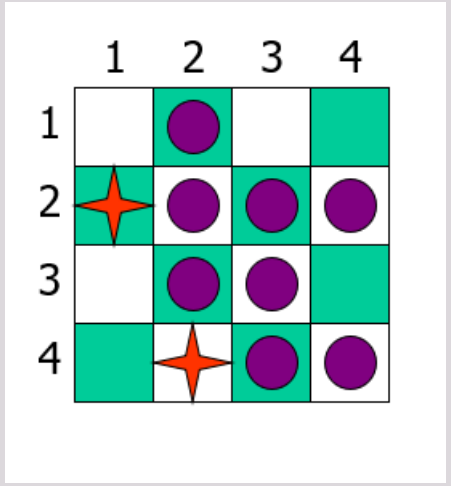


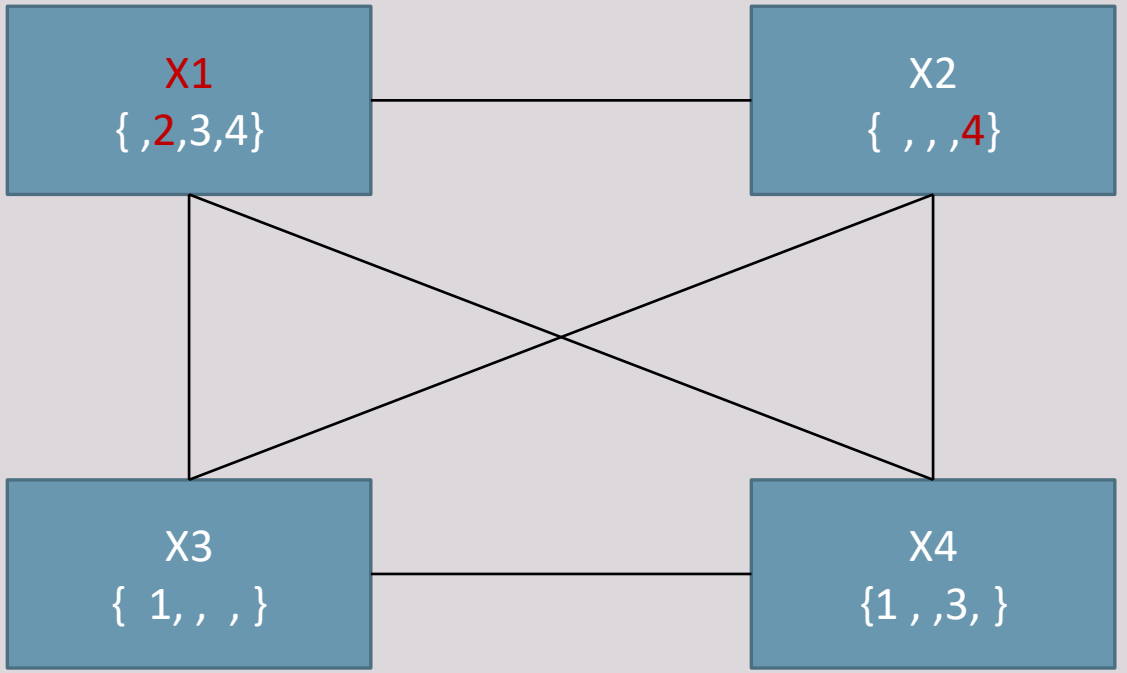
مثال ۸ وزیر (بررسی پیشرو) :



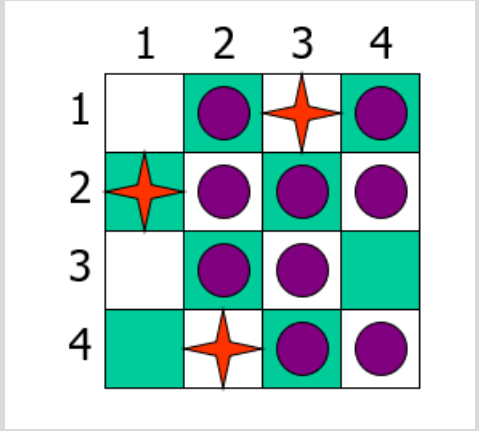


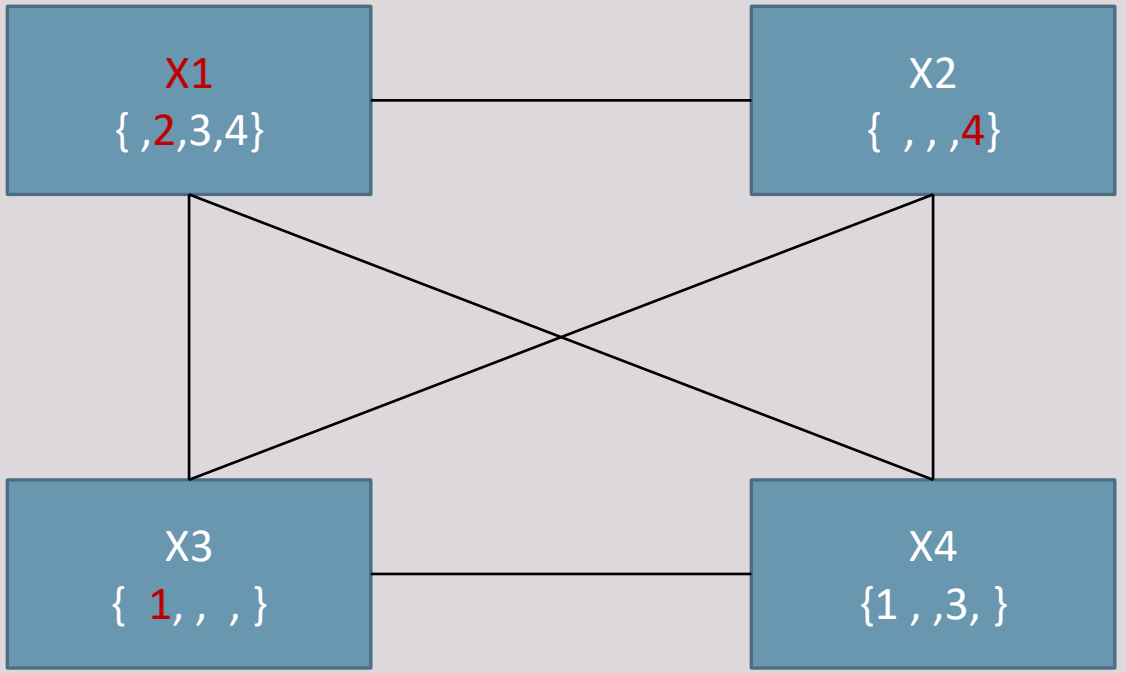
مثال ۸ وزیر (بررسی پیشرو) :



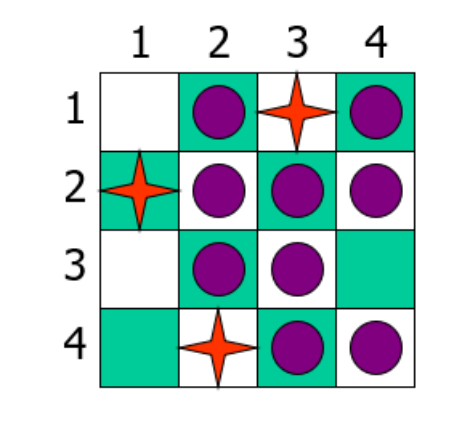


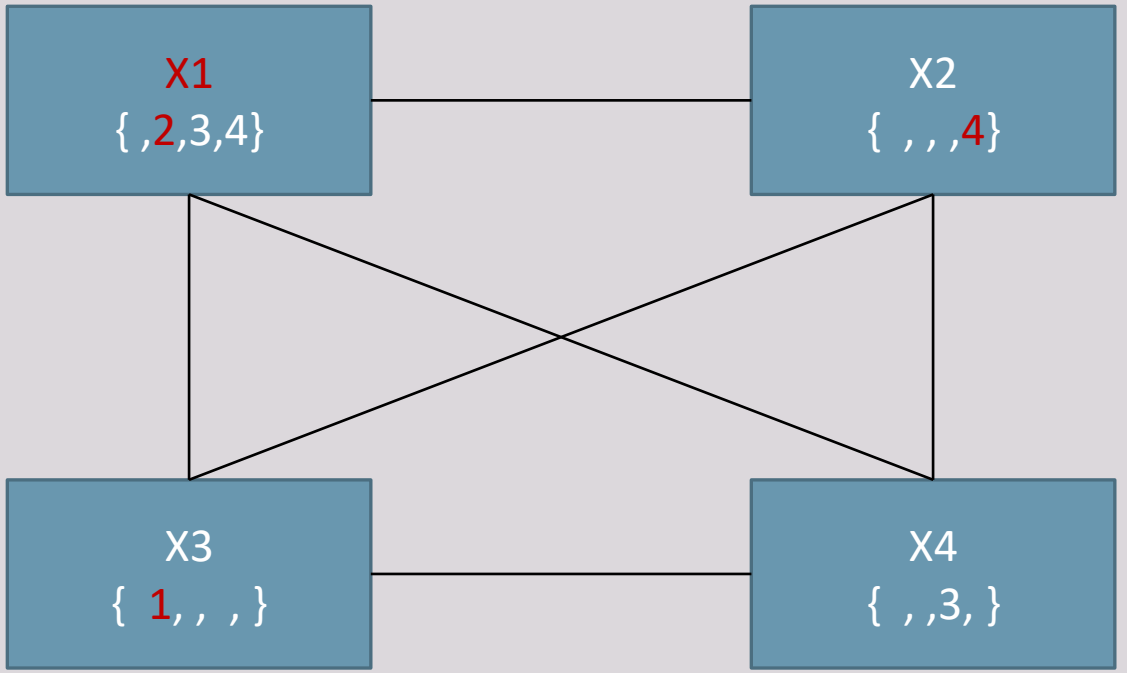
مثال ۸ وزیر (بررسی پیشرو) :



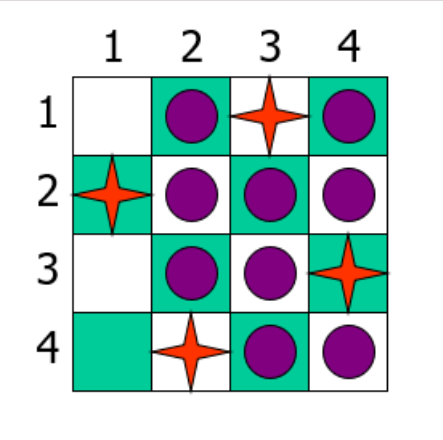


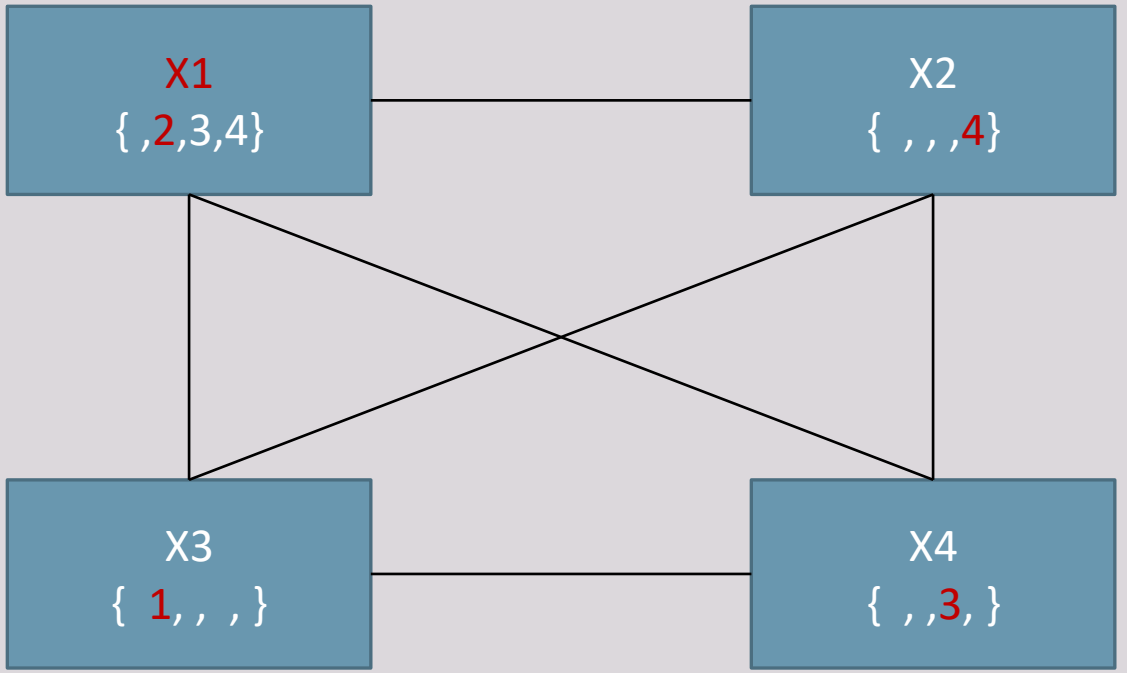
مثال ۸ وزیر (بررسی پیشرو) :





مثال ۸ وزیر (بررسی پیشرو) :





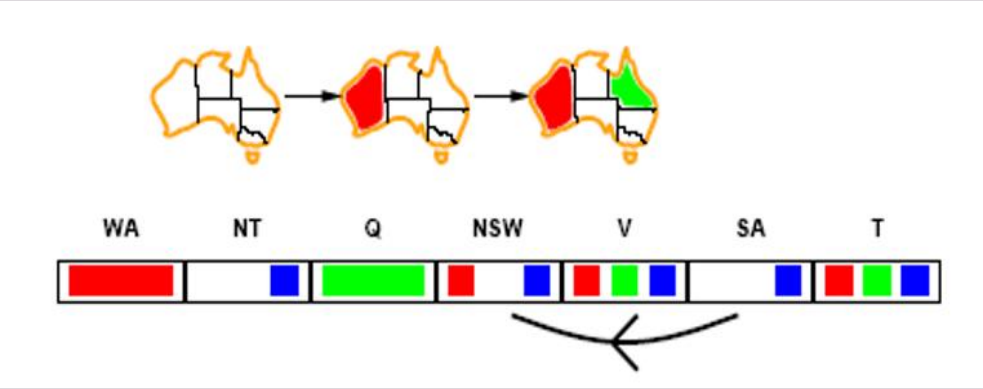
انتشار محدودیت:

- گرچه بررسی پیشرو بسیاری ازناسازگاریها را کشف میکند اما همه آن را تشخیص نمیدهد.
- برای نمونه در مثال رنگ آمیزی ، وقتی WAقرمز و Qسبز است NTو SAفقط میتواند آبی باشند. اما این دو همجوارند و نمیتوانند رنگ یکسانی داشته باشند. جستجوی بررسی پیشرو این مورد را بهعنوان ناسازگاری تشخیص نمیدهد، زیرا به اندازه کافی به جلو توجه نمیکند.
- پخش محدودیت درواقع به معنی تأثیر محدودیت روی یک متغیر، بر سایر متغیرها است.
- در مثال قبل ابتدا اطلاعات محدودیت باید از WAو Qبه NTو SAو سپس محدودیت باید بین NTو SAمنتشر شود تا ناسازگاری تشخیص داده شود
- اگر زمانی که برای انتشار محدودیتها صرف میشود بیشتر از زمان جستجوی ساده باشد، این روش ترجیح داده نمیشود

سازگاری یال :

❖یک روش سریع برای انتشار محدودیتها فراهم میکند و درنتیجه بسیار قویتر از جستجو بررسی پیشرو است

❖در روش سازگاری کمان با گراف محدودیت جهتدار روبرو هستیم



❖ دو یال جهتدار با سوهای مختلف بین SA NSWفرض میشود. ممکن است سازگاری کمان تنها در یک جهت برقرار باشد. میتوان گفت با توجه به دامنه های SAو NSWاین یال وقتی سازگار است که به ازای انتساب هر مقدار Xاز SA،مقداری مثل yبرای NSWوجود دارد که با X سازگار است

❖ دامنه های فعلی SAو NSWبه ترتیب {blue}و {red , blue}میباشد.

❖ به ازایSA =BLUE انتساب سازگاری برای NSW وجود دارد که NSW = RED است .

❖ لذا یال SAو NSWسازگار است. اما یال معکوس یعنی یال NSWبه SAسازگار نیست. زیرا به ازای blue = NSW،انتساب سازگاری برای SAوجود ندارد. با حذف مقدار blueاز دامنه NSW،یال سازگار میشود

سازگاری یال :

function: AC – S (CSP) returns the CSP, possibly with reduced domains

inputs: CSP, a binary CSP with Variables $\{X_1 , X_2 , \dots , X_n\}$

local variables: queue, a queue of arcs, initially all the arcs in CSP while queue is not empty do

$(X_i , X_j) \leftarrow$ REMOVE – FIRST (queue)

if REMOVE – INCONSISTENT – VALUES (X_i , X_j) then for each X_k in NEIGHBORS $[X_i]$ do add (X_k , X_i) to queue

function: REMOVE – INCOSISTENT – VALUES (X_i , X_j) returns true if we remove a value

removed ← false

for each x in DOMAIN $[X_i]$ do

if not value y on DOMAIN $[X_j]$ allows (x,y) to satisfy the cosntraint between X_i and X_j

then delete x from DOMAIN $[X_i]$; removed ← true

return removed

شکل ۵-۷: الگوریتم arc consistency (AC-S نسخه سوم این روش است که توسط طراح آن انتخاب شده است).

سازگاری یال :

- فرآیند حذف ناسازگاری ها باید مکررا انجام شود تا هیچ ناسازگاری باقی نماند. زیرا هر زمان که به منظور رفع ناسازگاری یال، یک مقدار از دامنه متغیری حذف میشود، ممکن است در یالهایی که به آن متغیر اشاره میکنند، ناسازگاری جدید ایجاد شود.
- الگوریتم کامل ناسازگاری یال (AC-3)یال هایی را که باید ناسازگاری آن ها بررسی شوند در یک صف قرار می دهد.
- اگر مقادیری از دامنه ی Xi حذف شود آنگاه هر یال که به Xi اشاره می کند دوباره به صف مذکور اضافه می شود تا سازگاری اش بررسی گردد.
- پیچیدگی زمانی در بدترین وضعیت O(n^2d^3) است.

سازگاری k :

- سازگاری یال تمام ناسازگاریهای ممکن را مشخص نمیکند
- نمونه ی قویتری برای انتشار محدودیت است.
- یک مسئله ارضای محدودیت سازگاری مرتبه ی kام دارد، اگر برای هر ۱ – kمتغیر و برای هر انتساب سازگار آن متغیرها، همیشه یک مقدار سازگار برای هر kامین متغیر وجود داشته باشد.
- بهعنوان مثال معنای 1 -consistentاین است که هر متغیر با خودش سازگار است (این سازگاری، سازگاری گره نیز نامیده میشود). سازگاری مرتبه ی ۲ مشابه سازگاری کمان است. معنای سازگاری مرتبه ی ۳ این است که هر جفت از متغیرهای همجوار میتوانند به سومین متغیر همسایه گسترش یابد. (نام دیگر آن سازگاری مسیر است.)

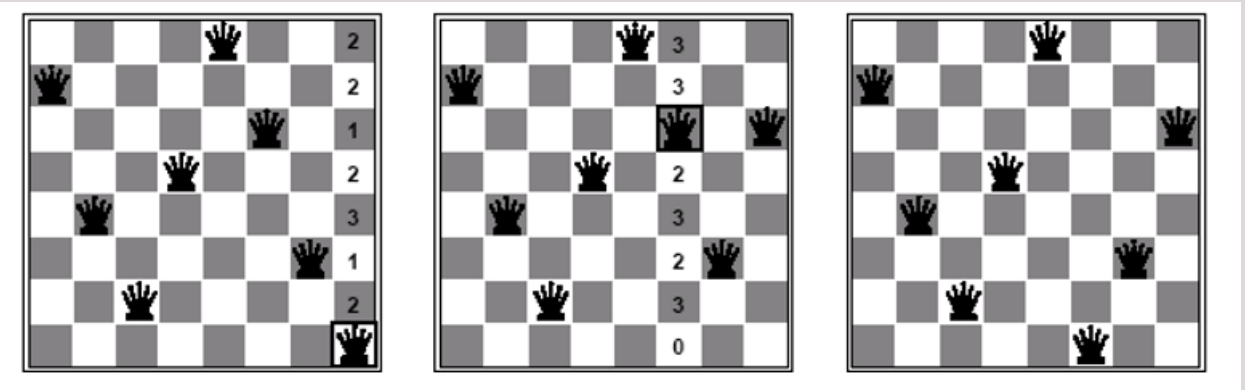
سازگاری k :

- یک گراف دارای سازگاری قوی است اگر سازگاری مرتبهی اول، مرتبهی دوم و ... و سازگاری مرتبـ هی kام را داشـته باشـد.
- این صورت، مسئله را بدون عقبگرد میتوان حل کرد
- پیچیدگی زمانی آن O(nd) است

کاربرد جستجوهای محلی در حل مسائل ارضای محدودیتها:

- الگوریتم های جستجوی محلی بسیاری از CSPها را بطور کارآمد حل می کنند
  - حالت اولیه مقداری را به هر متغیر نسبت می دهد .
  - تابع جانشین ، تغییر مقدار یک متغیر در هر زمان
- انتخاب مقدار جدید برای یک متغیر
  - انتخاب مقداری که کمترین برخورد را با متغیرهای دیگر ایجاد کند(اکتشاف برخورد کم)
  - زمان اجرای برخورد کم مستقل از اندازه مسئله است
  - برخورد کم، برای مسئله های سخت نیز کار میکند
- جست و جوی محلی میتواند در صورت تغییر مسئله، تنظیمات Online را انجام دهد

مسائل ارضای محدودیت مثال ۸ وزیر:



- ✓ استفاده از مینیمم تناقضات برای مسئله ۸– وزیر در مرحله، یک وزیر برای انتساب مجدد در ستون خودش انتخاب میگردد. تعداد تناقضات (تعداد وزیرهایی که به هم حمله میکنند) در هر مربع نشان داده شده است. الگوریتم وزیر را به مربعی با کمترین تناقض انتقال میدهد.

پایان فصل ۵