

مسایل ارضای محدودیت
(Constraint Satisfaction Problem)

$$\begin{cases} x + 2Y + Z = 4 \\ x + Y = 1 \\ x + Z = 2 \end{cases}$$

x, Y, Z

$$2Y = 2 \rightarrow Y = 1$$

$$x = 0$$

$$Z = 2$$

مساله ارضای محدودیت (CSP)

- به طور کلی، یک مساله ارضای محدودیت (CSP) به صورت مجموعه‌ای از متغیرهای $X_1, X_2, X_3, \dots, X_n$ و مجموعه‌ای از محدودیت‌های $C_1, C_2, C_3, \dots, C_m$ تعریف می‌شود.
- هر متغیر X_i دارای یک دامنه غیرتهی D_i از مقادیر ممکن است.
- هر محدودیت C_i شامل زیرمجموعه‌ای از متغیرها است و ترکیب‌های ممکن مقادیر را برای آن زیرمجموعه مشخص می‌کند.
- یک حالت از مساله انتساب مقادیر به تعدادی از متغیرها، یا همه آن‌ها می‌باشد.
- انتسابی که هیچ محدودیتی را نقض نکند، انتساب سازگار یا مجاز گفته می‌شود.

$$\begin{cases} x + 2y + z = 4 \\ x + y = 1 \\ x + z = 2 \end{cases}$$

x, y, z

$$2y = 2 \rightarrow y = 1$$

$$x = 0$$

$$z = 2$$

متغيرها

$x \ y \ z$

محدداتها

$$x + 2y + z = 4$$

$$x + y = 1$$

$$x + z = 2$$

- درانتساب کامل، تمام متغیرها، مقداردهی شده‌اند.
- حل یک مساله CSP، انتساب کاملی است که تمام محدودیت‌ها را برآورده کند.
- دربرخی از مسائل، غیر از مجاز بودن انتساب‌ها، به راه حلی نیاز است که تابع هدف را بیشینه کند.

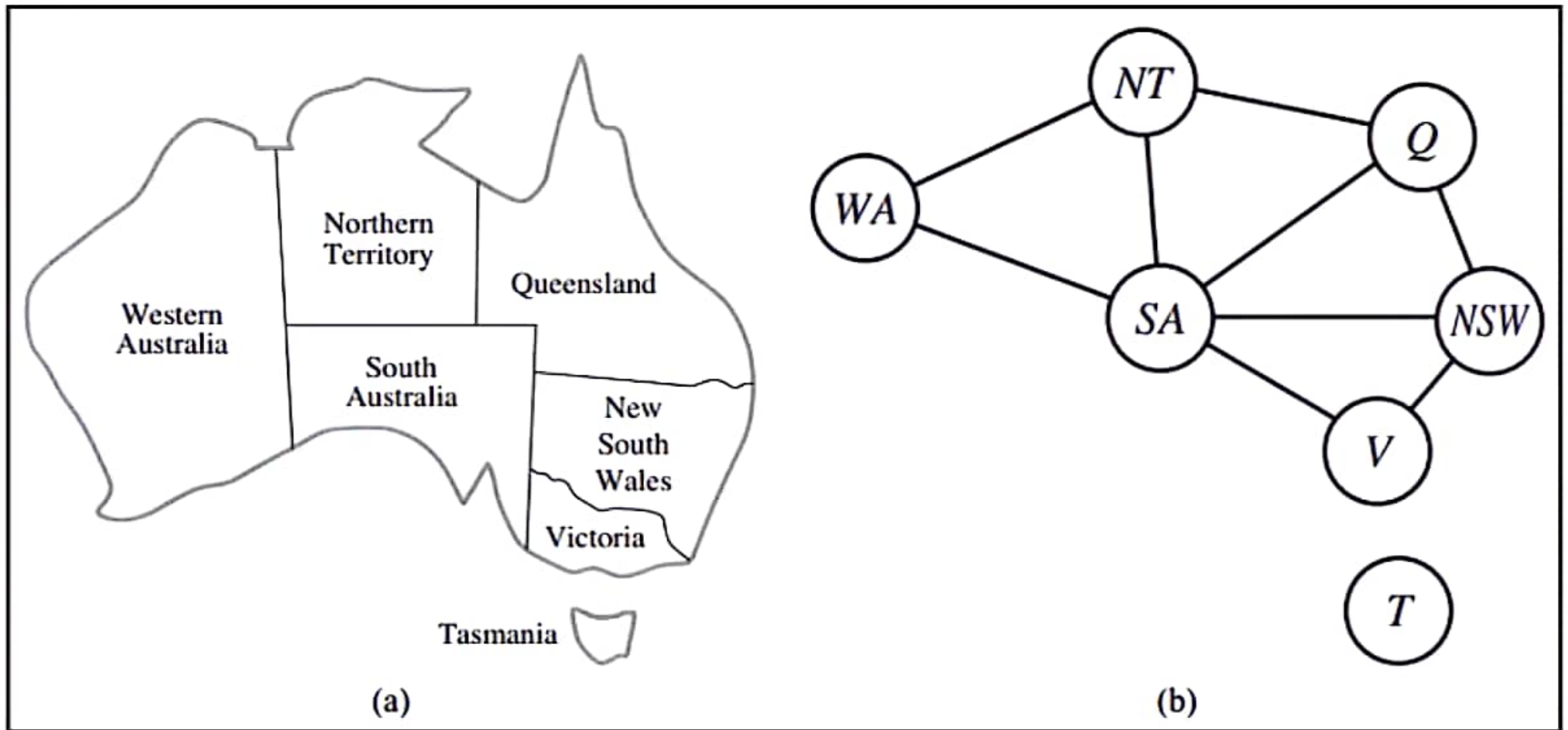
- درانتساب کامل، تمام متغیرها، مقداردهی شده‌اند.
- حل یک مساله CSP، انتساب کاملی است که تمام محدودیت‌ها را برآورده کند.
- دربرخی از مسائل، غیر از مجاز بودن انتساب‌ها، به راه حلی نیاز است که تابع هدف را بیشینه کند.

مساله زنگ آمیزی کدلف :

۱- دو راکت همایه هم رنگ نباشند .

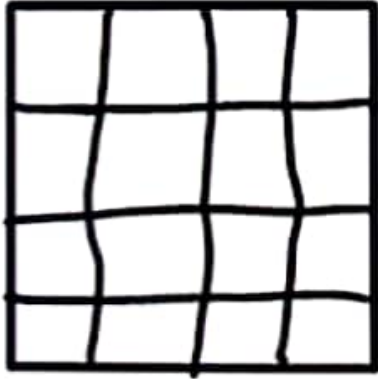
۲- صلاحی رنگ استفاده شود .

تابع هدف



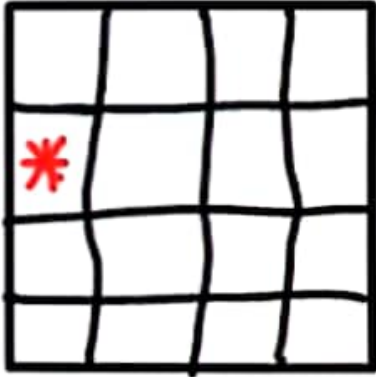
فرموله سازی افزایشی برای مساله CSP:

- حالت اولیه: انتساب خالی (هیچ متغیری مقدار نگرفته است)
- تابع مابعد (جانشین): مقداردهی یک متغیر، بدون نقض محدودیت ها
- آزمون هدف: آیا انتساب فعلی، کامل است؟
- هزینه مسیر: یک هزینه ثابت برای هر مرحله.



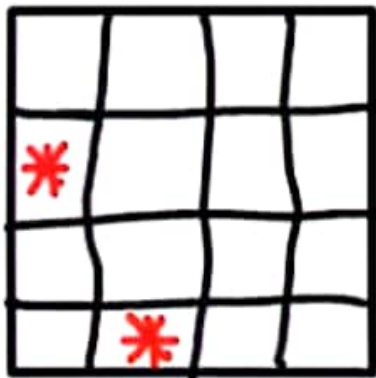
فرموله سازی افزایشی برای مساله CSP:

- حالت اولیه: انتساب خالی (هیچ متغیری مقدار نگرفته است)
- تابع مابعد (جانشین): مقداردهی یک متغیر، بدون نقض محدودیت ها
- آزمون هدف: آیا انتساب فعلی، کامل است؟
- هزینه مسیر: یک هزینه ثابت برای هر مرحله.



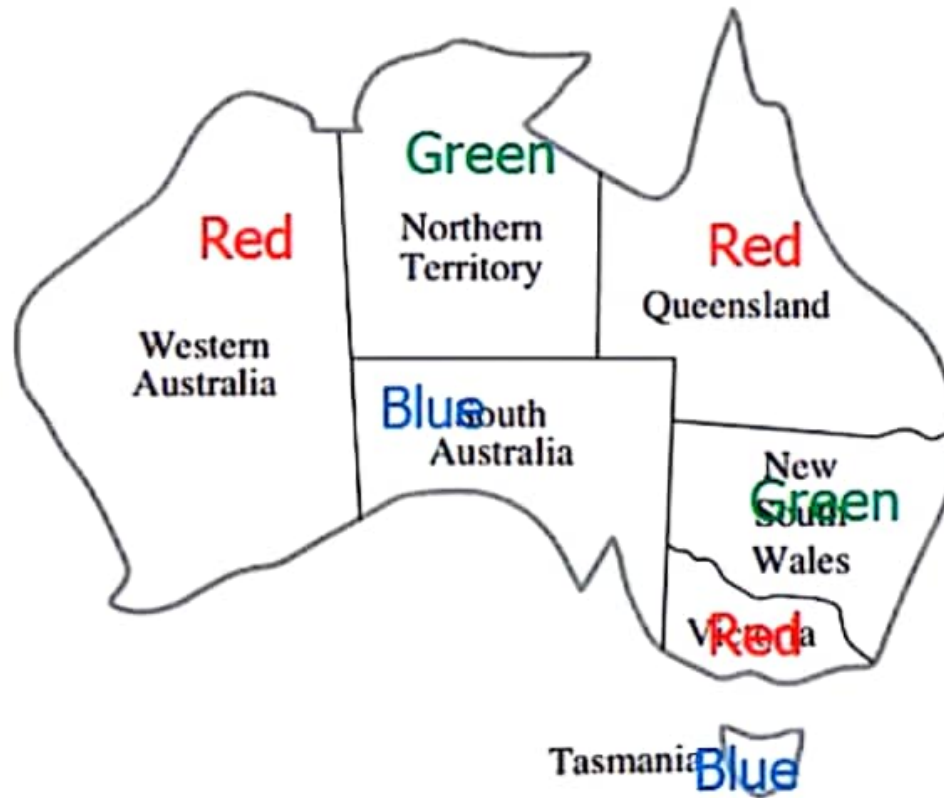
فرموله سازی افزایشی برای مساله CSP:

- حالت اولیه: انتساب خالی (هیچ متغیری مقدار نگرفته است)
- تابع مابعد (جانشین): مقداردهی یک متغیر، بدون نقض محدودیت ها
- آزمون هدف: آیا انتساب فعلی، کامل است؟
- هزینه مسیر: یک هزینه ثابت برای هر مرحله.

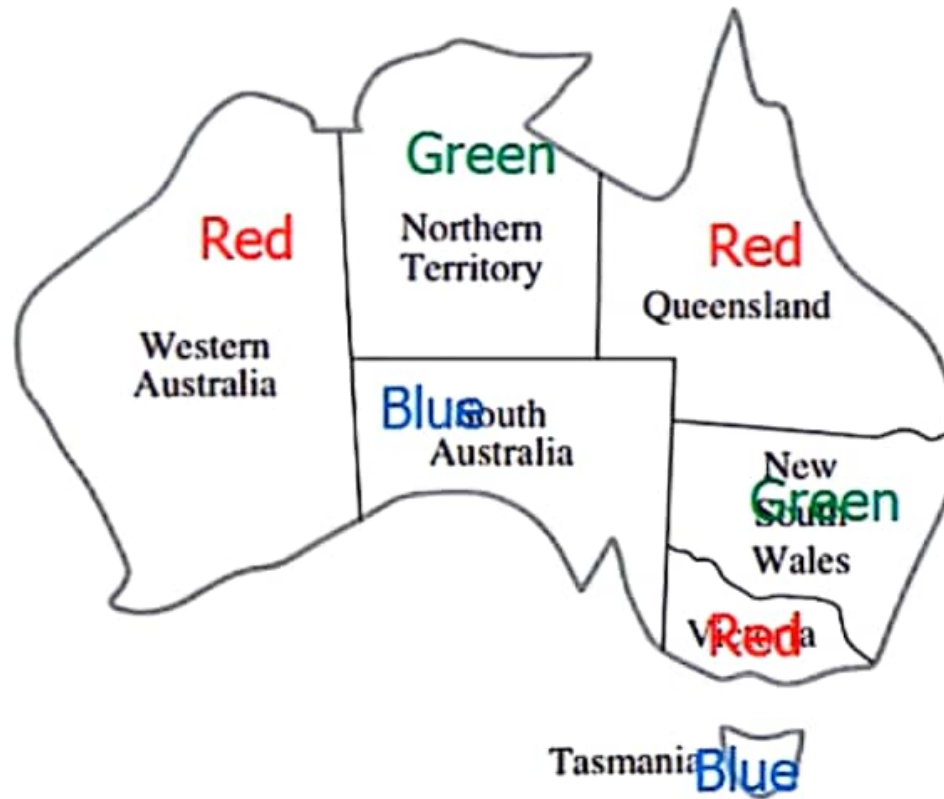


فرموله سازی افزایشی برای مساله CSP:

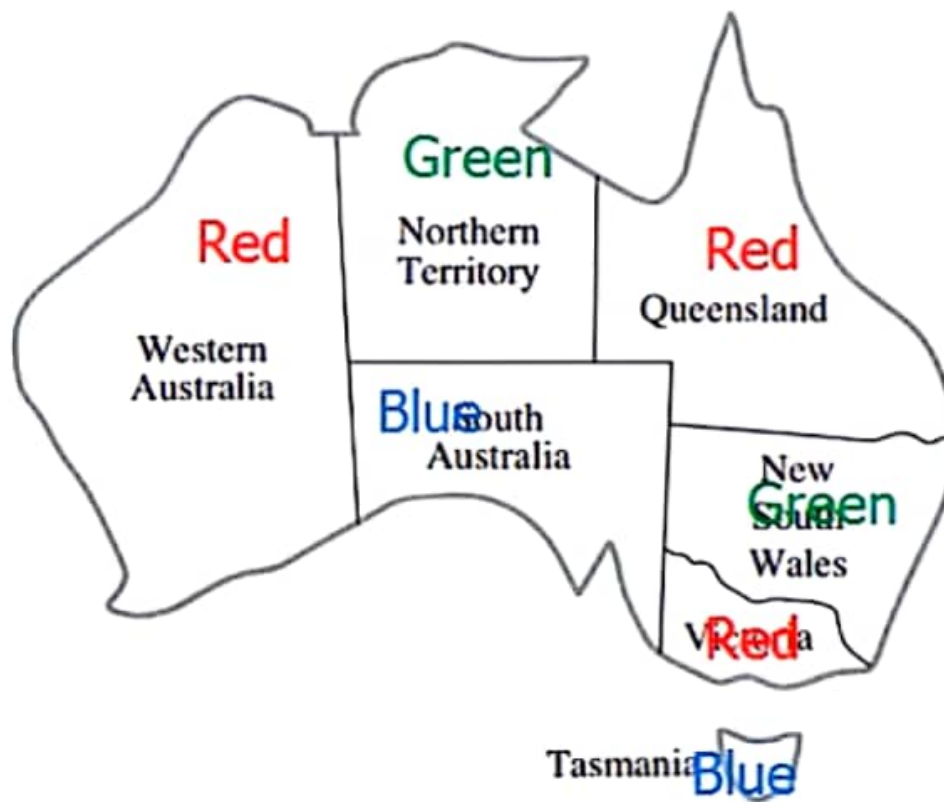
- حالت اولیه: انتساب خالی (هیچ متغیری مقدار نگرفته است)
- تابع مابعد (جانشین): مقداردهی یک متغیر، بدون نقض محدودیت ها
- آزمون هدف: آیا انتساب فعلی، کامل است؟
- هزینه مسیر: یک هزینه ثابت برای هر مرحله.

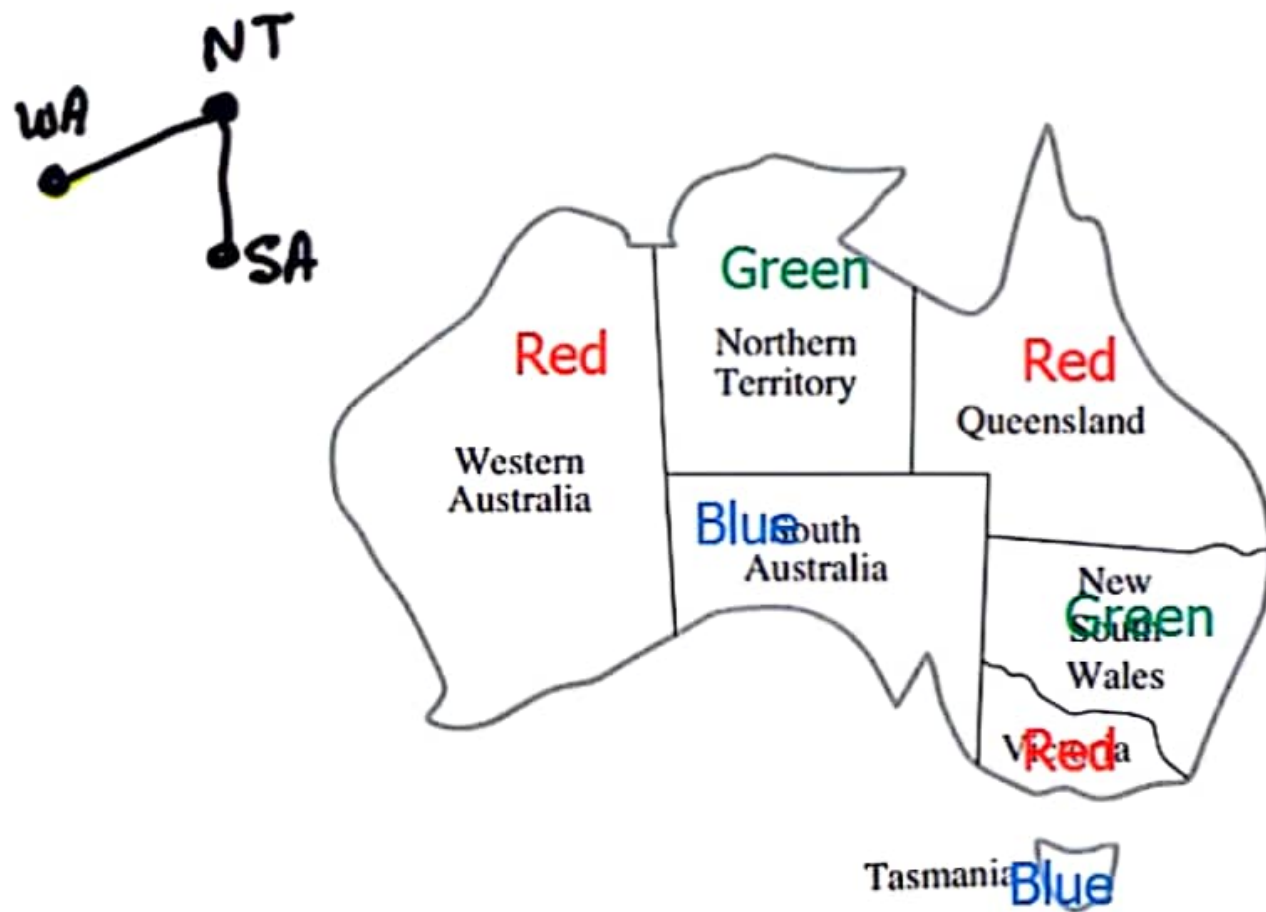


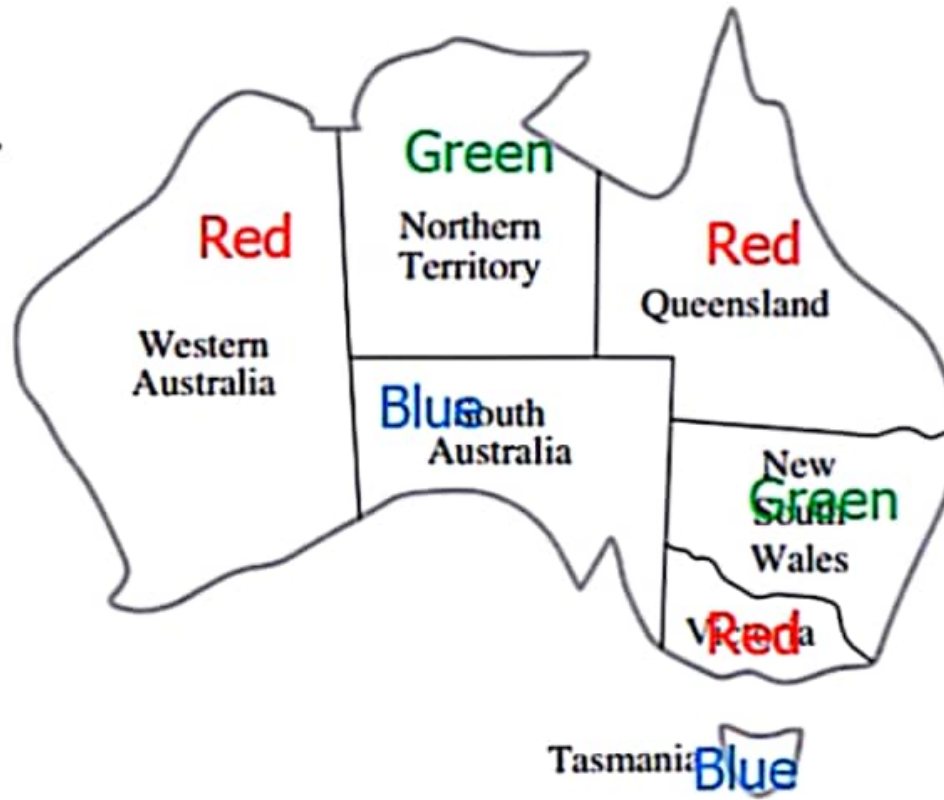
WA

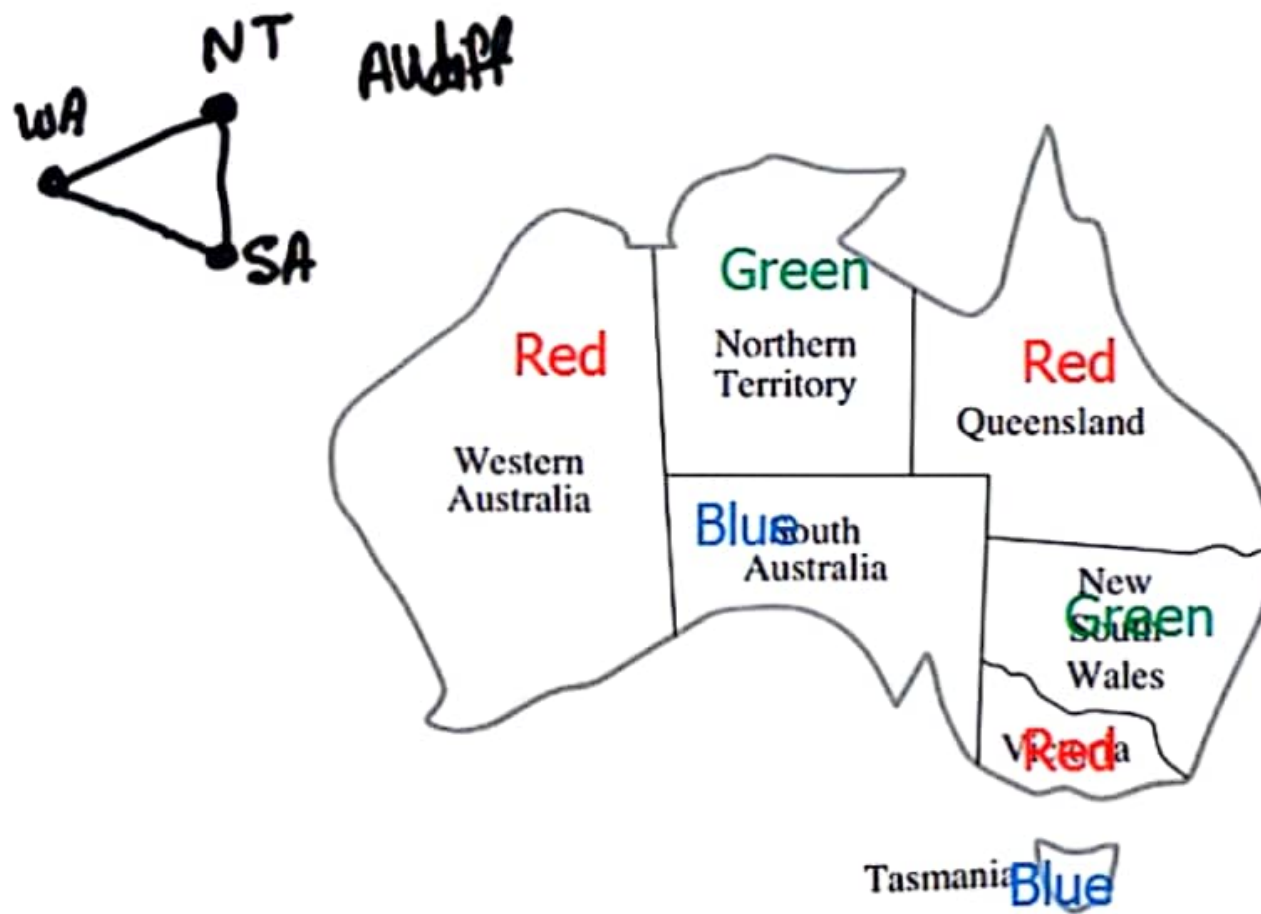


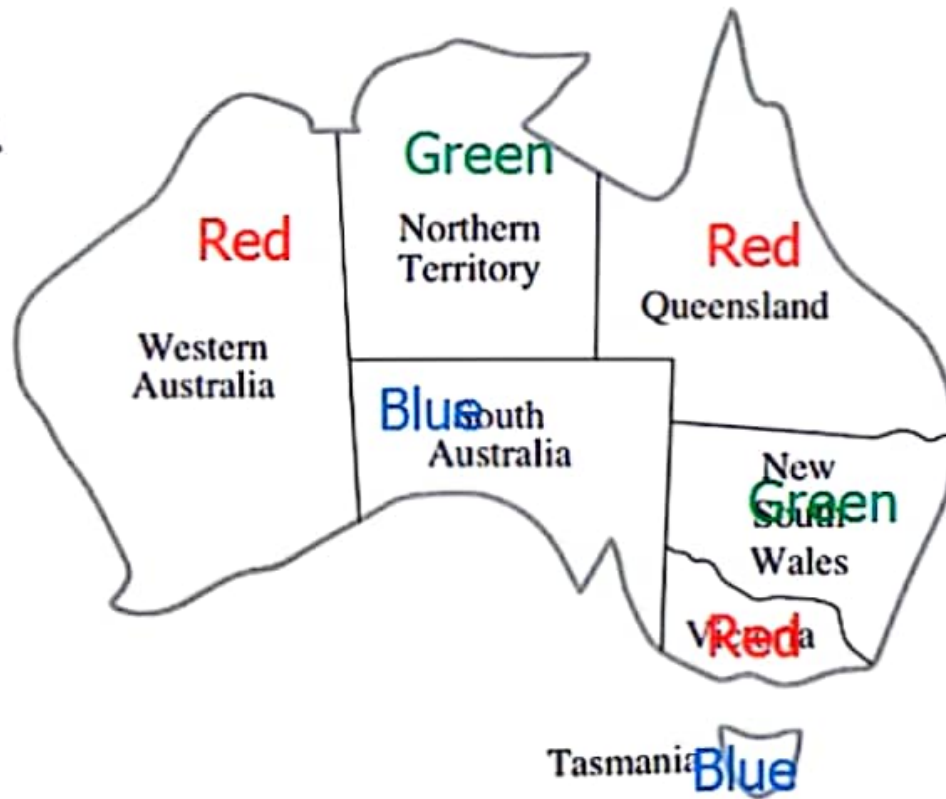
WA NT

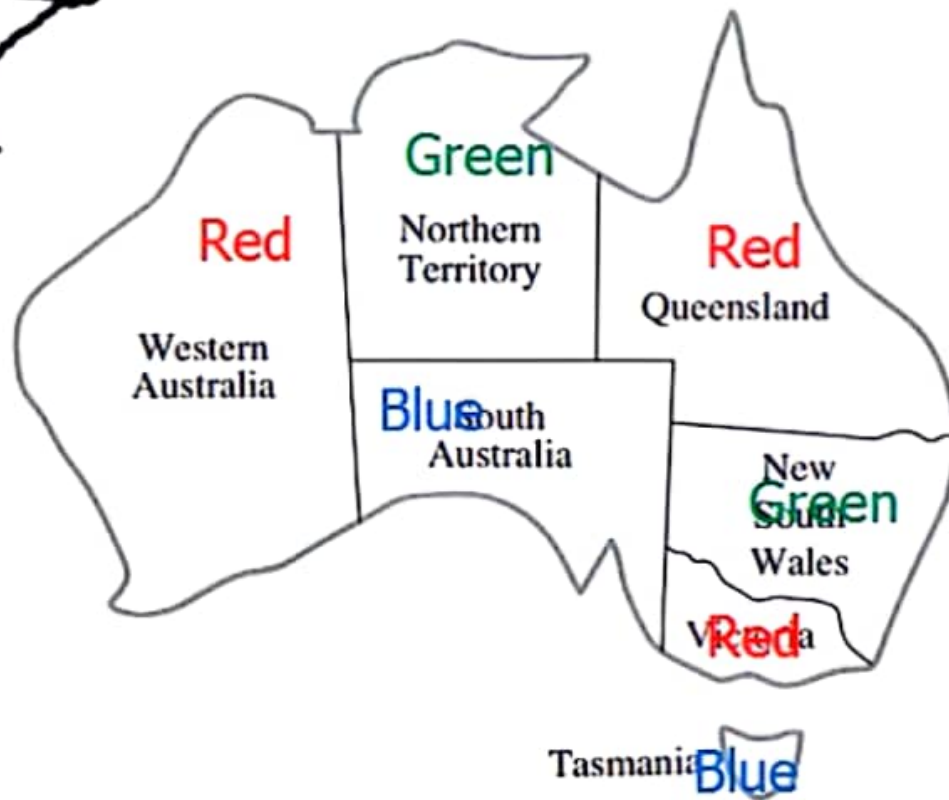
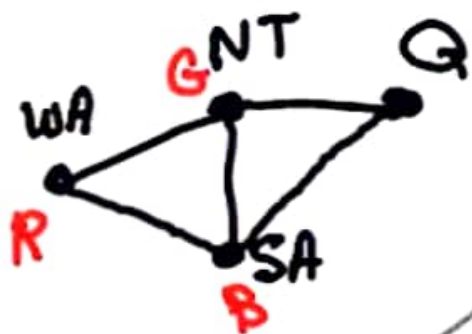


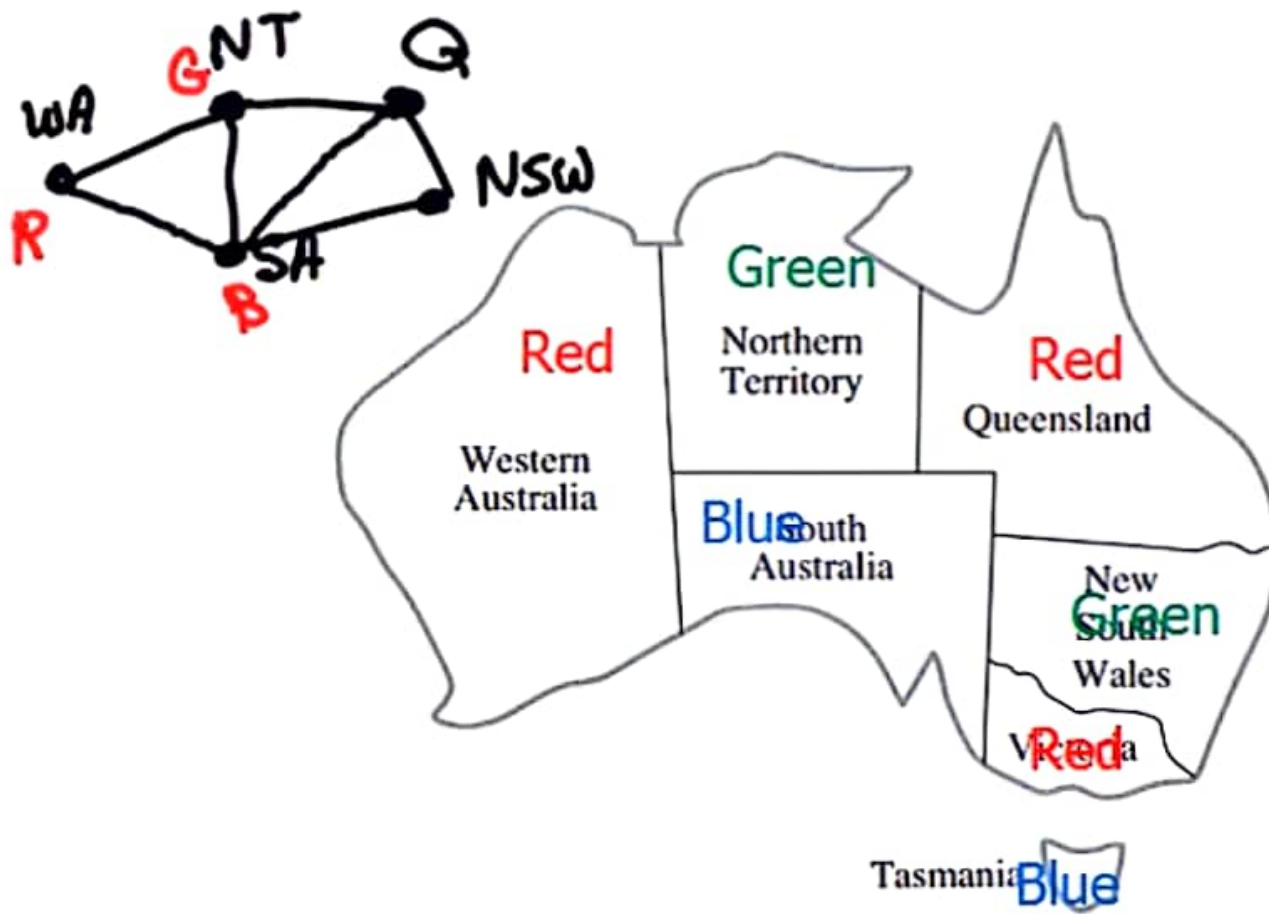


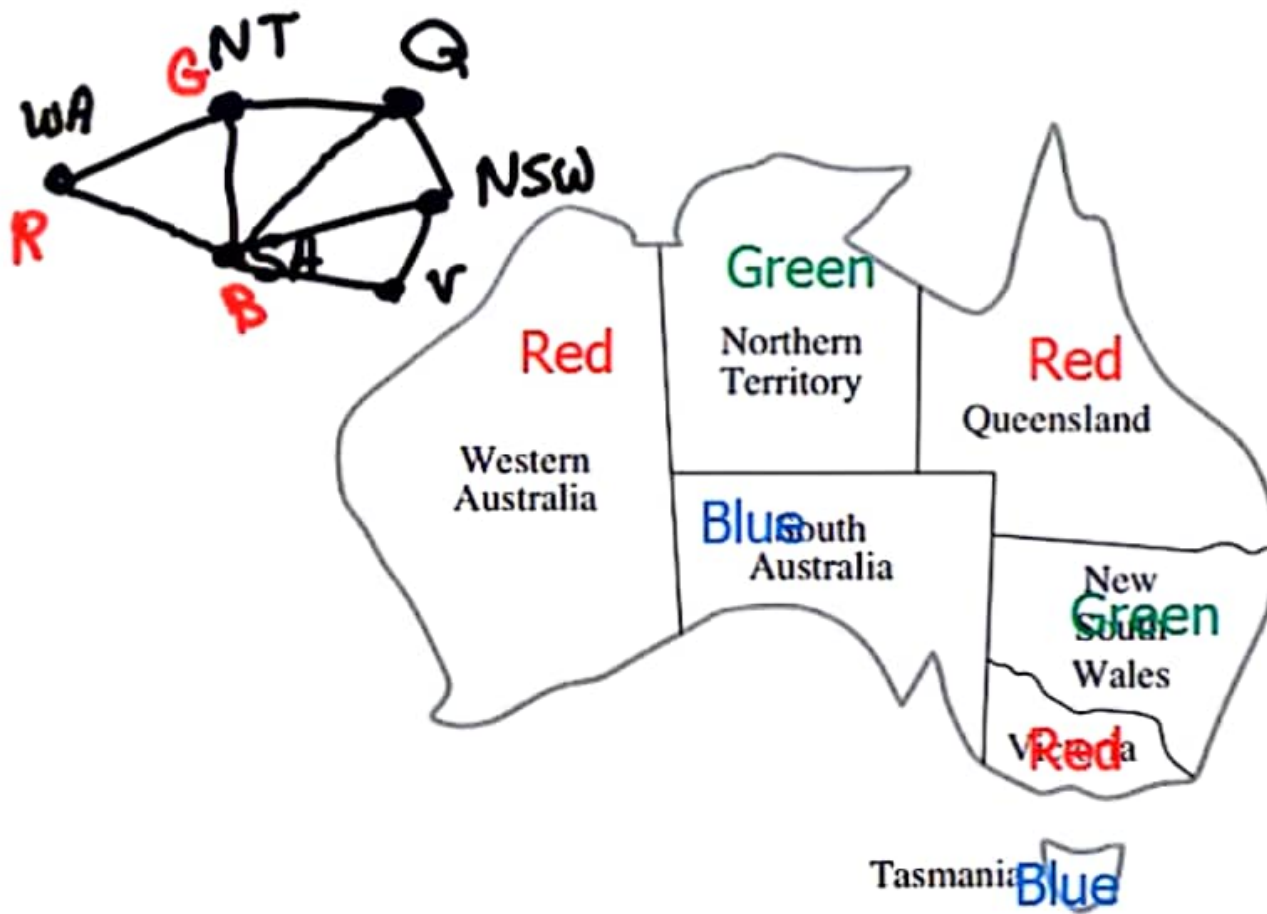


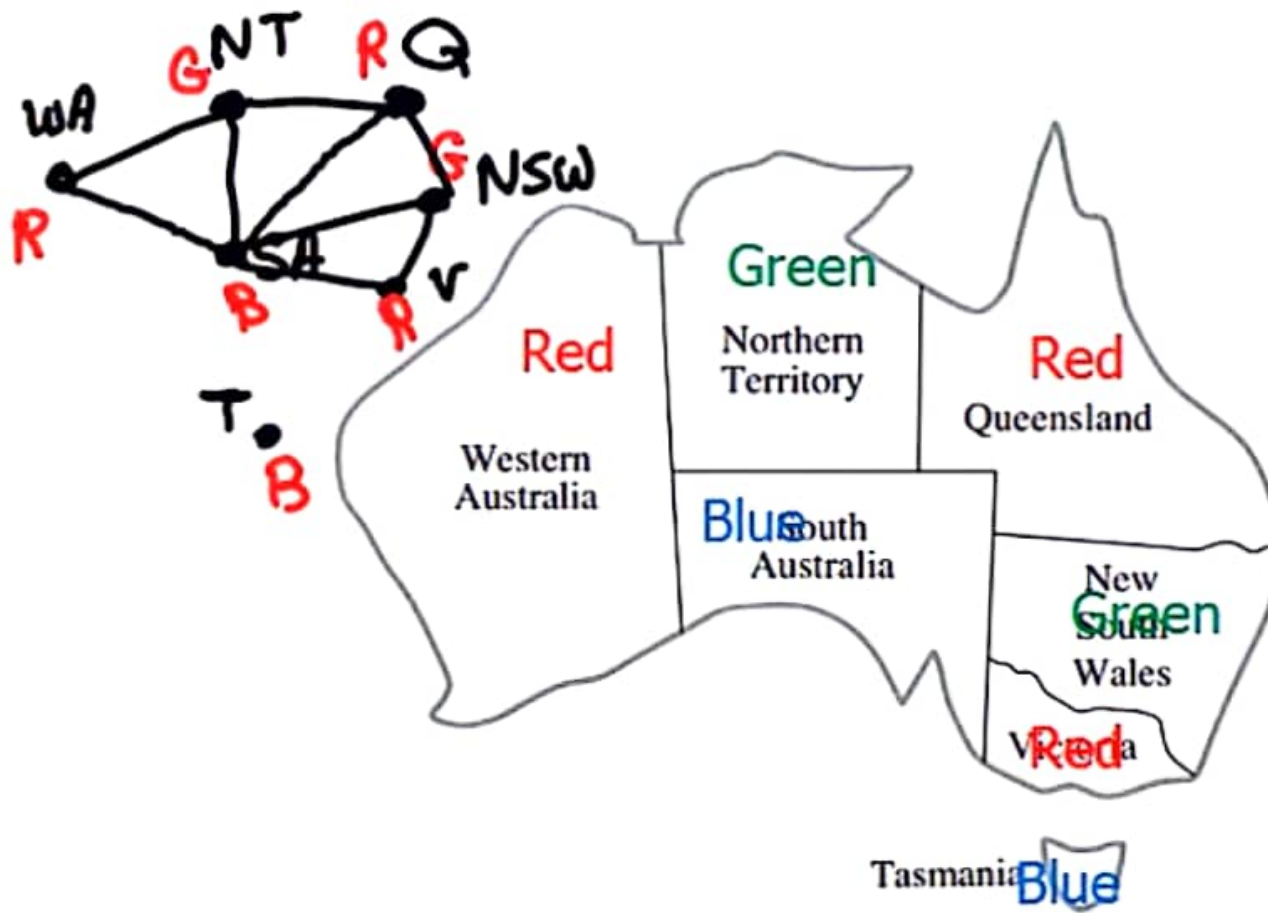






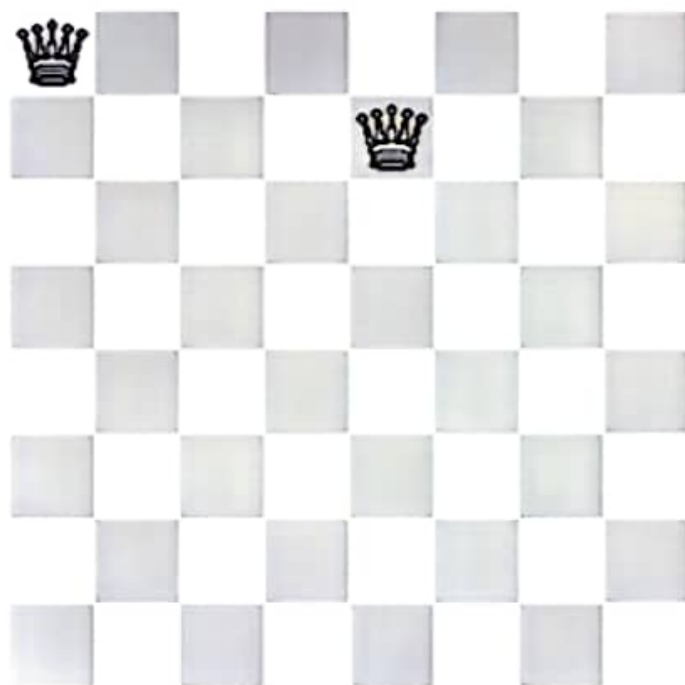






ساده ترین CSP:

○ مسایل CSP، با متغیرهای گسسته با دامنه‌های متناهی ساده ترین هستند.



$$Q_1=1$$

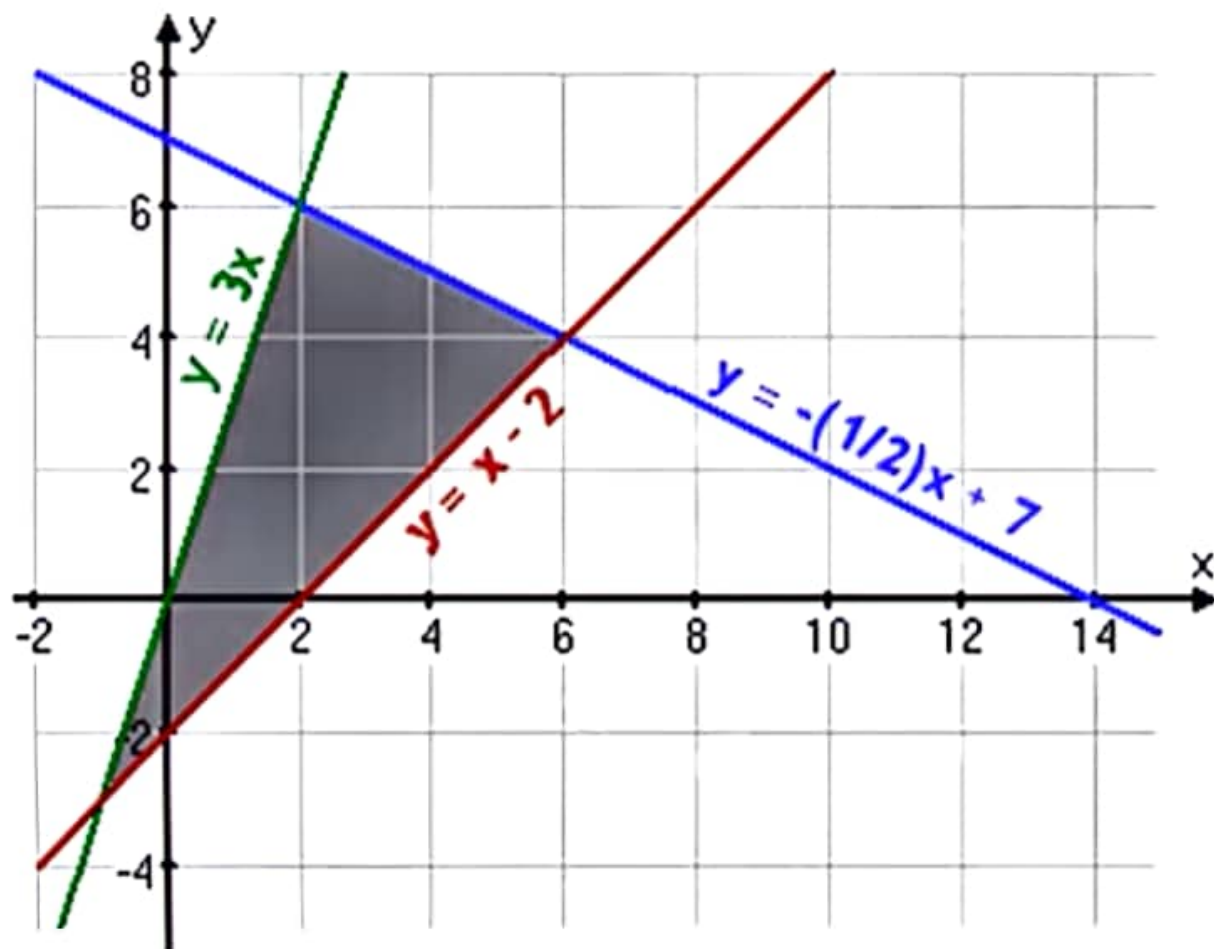
$$Q_2=5$$

$$Q_3=\{2,7,8\}$$

{Green, Blue}



مسائل CSP با دامنه پیوسته:

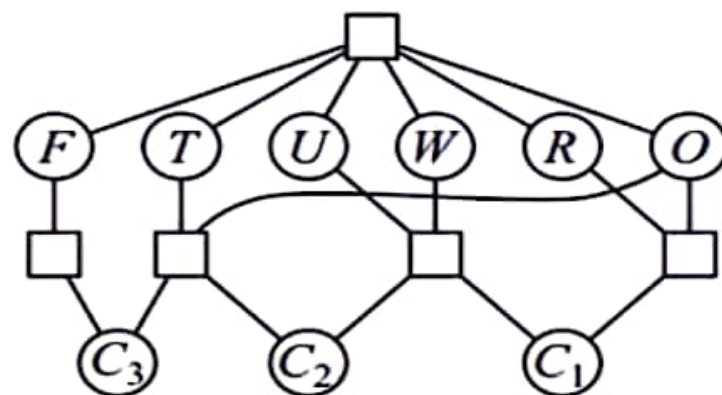


انواع محدودیت‌ها در CSP:

- محدودیت یکتا
- محدودیت دودویی
- محدودیت‌های مرتبه بالاتر مانند معمای رمزنگاری (crypt arithmetic)
- محدودیت‌های اولویت دار-محاسبه هزینه کل در یک انتساب خاص به متغیرها (مسائل بهینه سازی)

$$\begin{array}{r}
 T \ W \ O \\
 + \ T \ W \ O \\
 \hline
 F \ O \ U \ R
 \end{array}$$

(a)



(b)

متغیرها $F \ T \ U \ W \ R \ O \ X_1 \ X_2 \ X_3$

دامنه $\{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$

محدودیتها

$$\begin{aligned}
 O + O &= R + 10C_1 \\
 C_1 + W + W &= U + 10C_2 \\
 C_2 + T + T &= O + 10C_3 \\
 C_3 &= F
 \end{aligned}$$

فرموله سازی مسایل ارضای محدودیت به

صورت مسایل جستجو

○ اگر از جستجوی سطحی برای حل مساله استفاده کنیم:

- فاکتور انشعاب در اولین سطح nd است. چون برای هر متغیر d مقدار ممکن وجود دارد.

- در سطح بعدی فاکتور انشعاب $d(n-1)$ است.

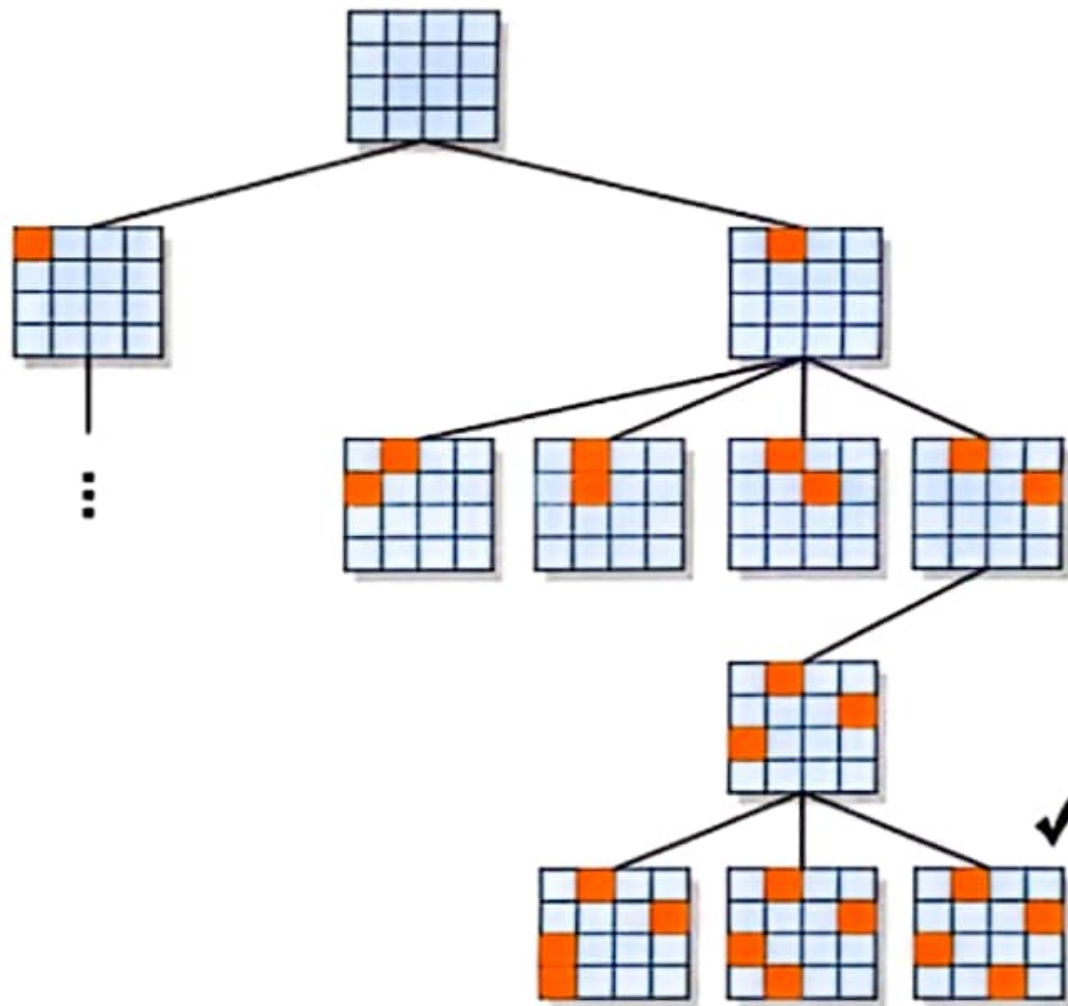
- با این فرم جستجو درخت $d^n * n!$ برگ دارد.

○ فقط d^n انتساب کامل وجود دارد!

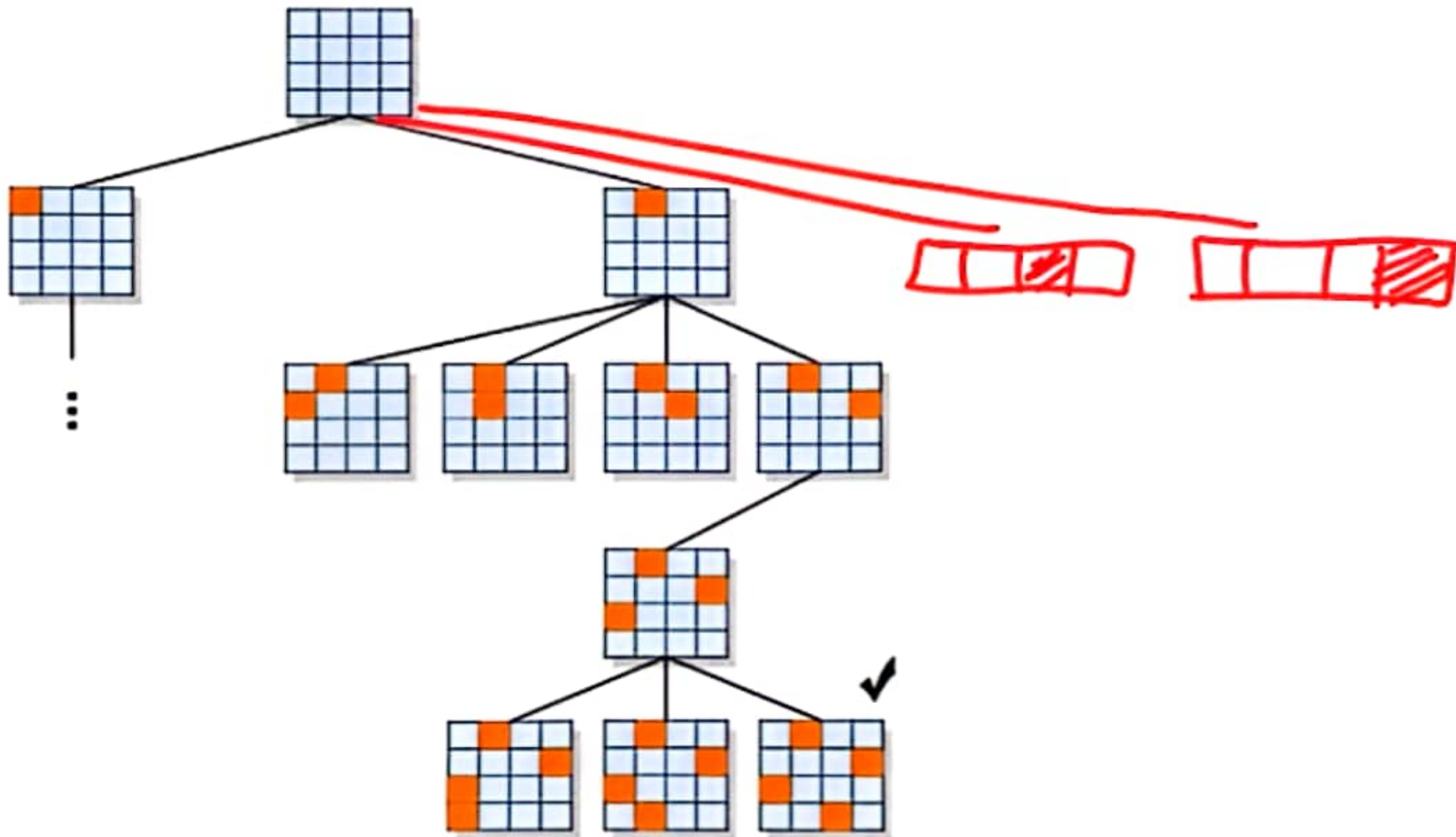
○ در جستجوی معمولی خاصیت جا به جا پذیری مربوط به مسائل ارضای محدودیت نادیده گرفته شده است.

○ جا به جا پذیری: اگر ترتیب به کارگیری اعمال تاثیری در نتیجه نهایی نداشته باشد مساله جا به جا پذیر است.

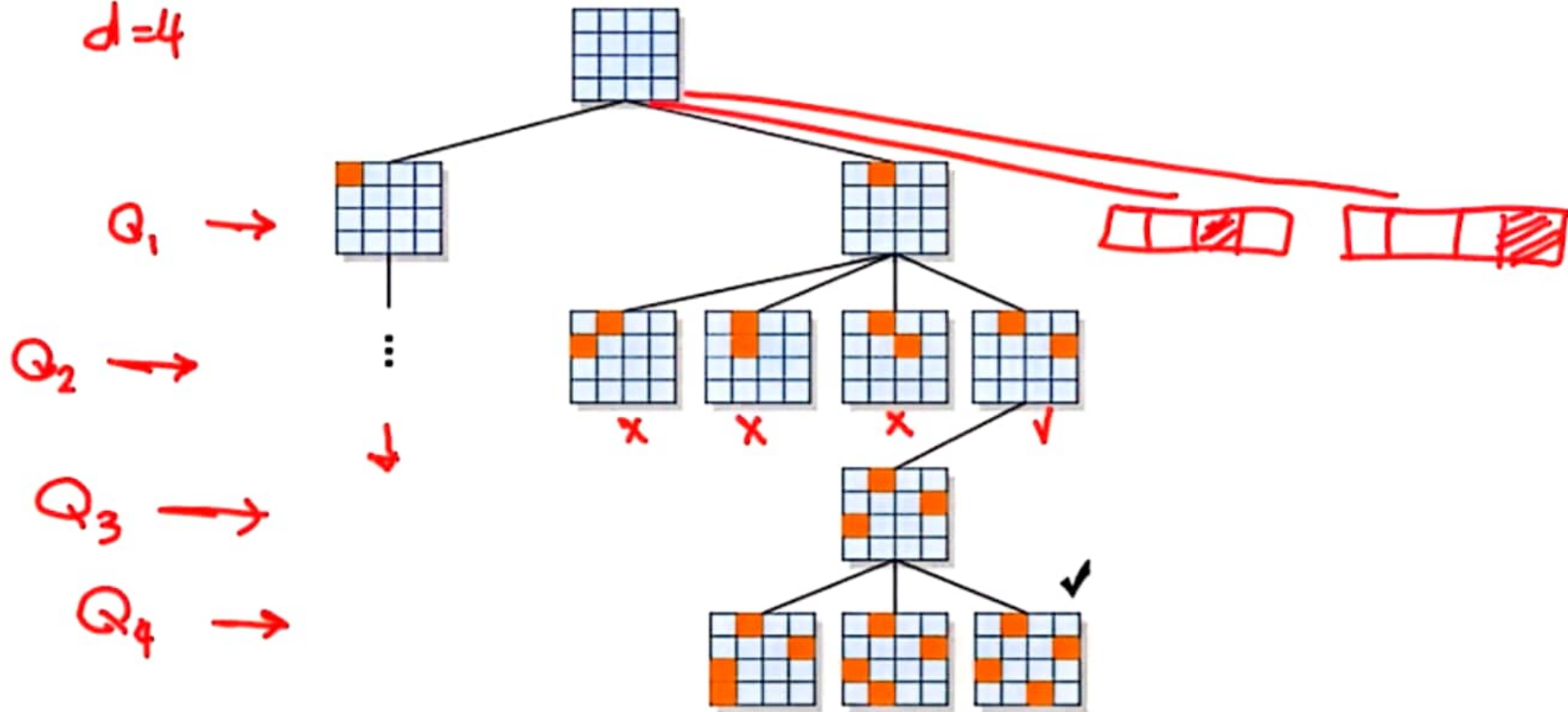
- در CSP هنگامی که مقادیر را به متغیرها نسبت می‌دهیم، صرف نظر از ترتیب آن‌ها به مقداردهی یکسانی می‌رسیم.
- در حل مسایل CSP، در هر سطح فقط یک متغیر، مقداردهی می‌شود. لذا فاکتور انشعاب آن d است.

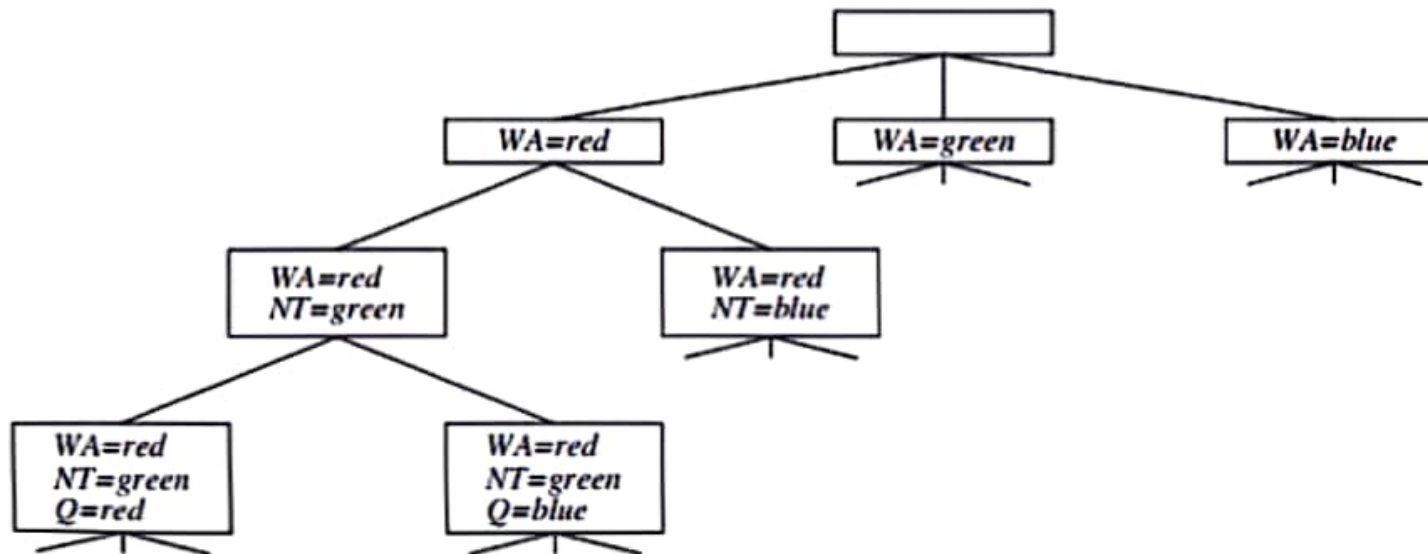


$d=4$



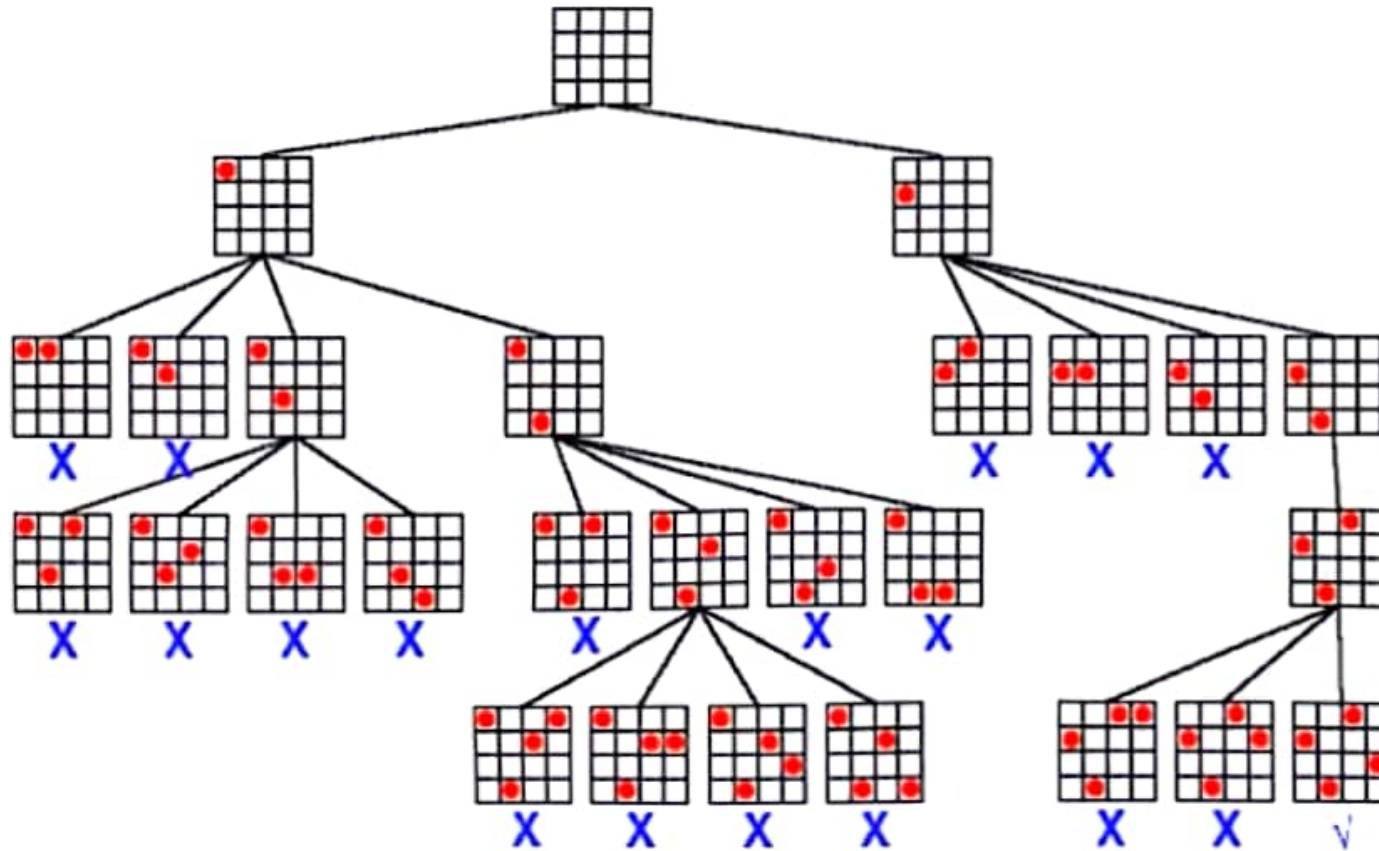
$d=4$

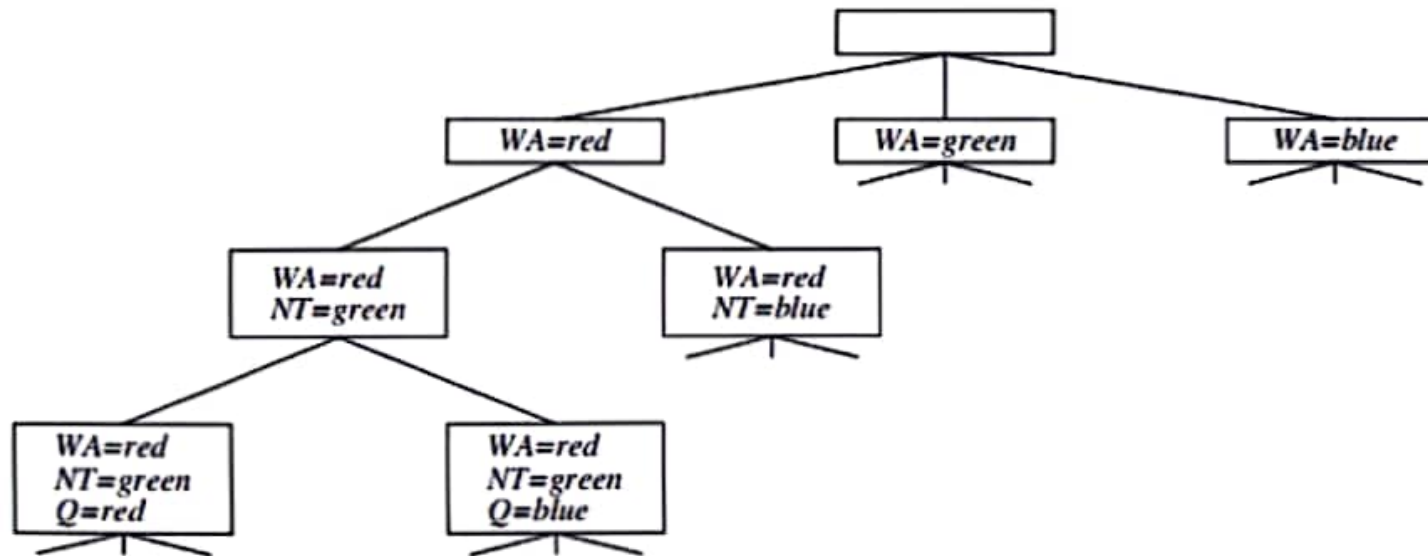




جستجوی عقب‌گرد (BT)

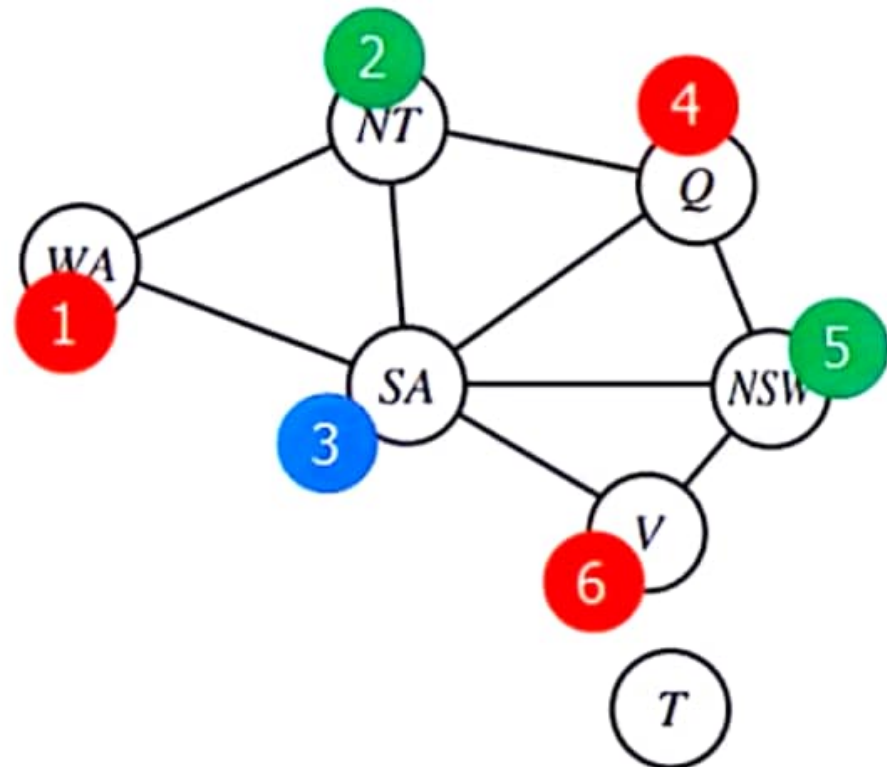
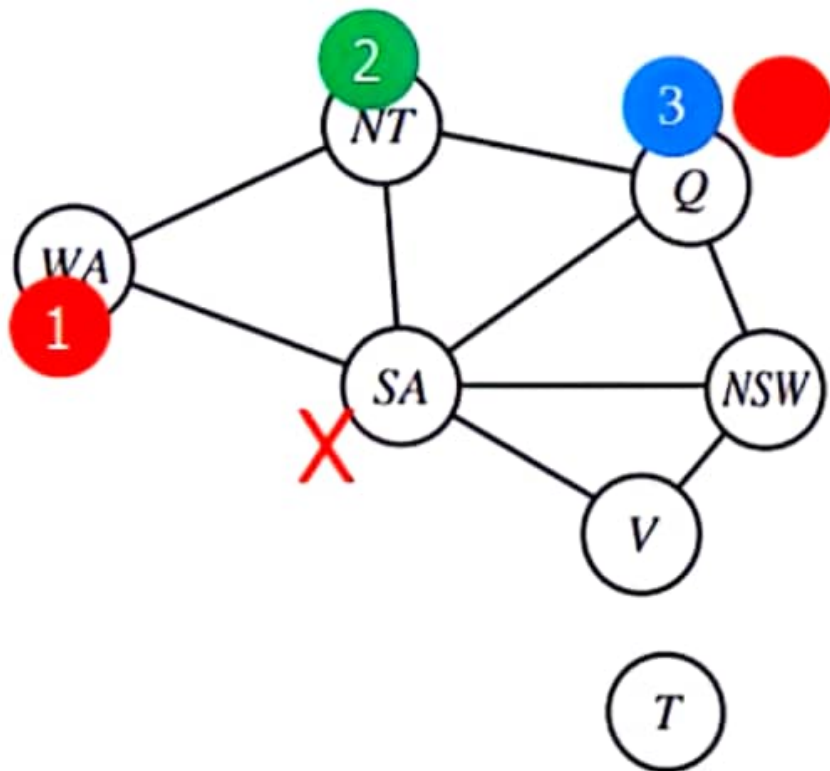
(Back Tracking)





- الگوریتم جستجوی عقبگرد، عملکردی ناآگاهانه دارد. لذا برای مسائل بزرگ ناکارآمد است
- برای حل مسایل CSP، از روش هایی استفاده می کنیم که به پرسش های زیر پاسخ دهند:
 - الف) متغیر بعدی که باید مقدار بگیرد کدام است و ترتیب بررسی مقادیر امکان پذیر برای آن چگونه باشد؟
 - ب) انتساب مقدار به متغیر فعلی روی متغیرهای بدون مقدار چه تاثیری دارد؟
 - ج) وقتی مسیری شکست می خورد، آیا جستجو می تواند از تکرار شکست در مسیرهای بعدی جلوگیری کند؟

○ انتخاب متغیر بعدی در صورتی که ایستا باشد، کارایی کمی خواهد داشت:

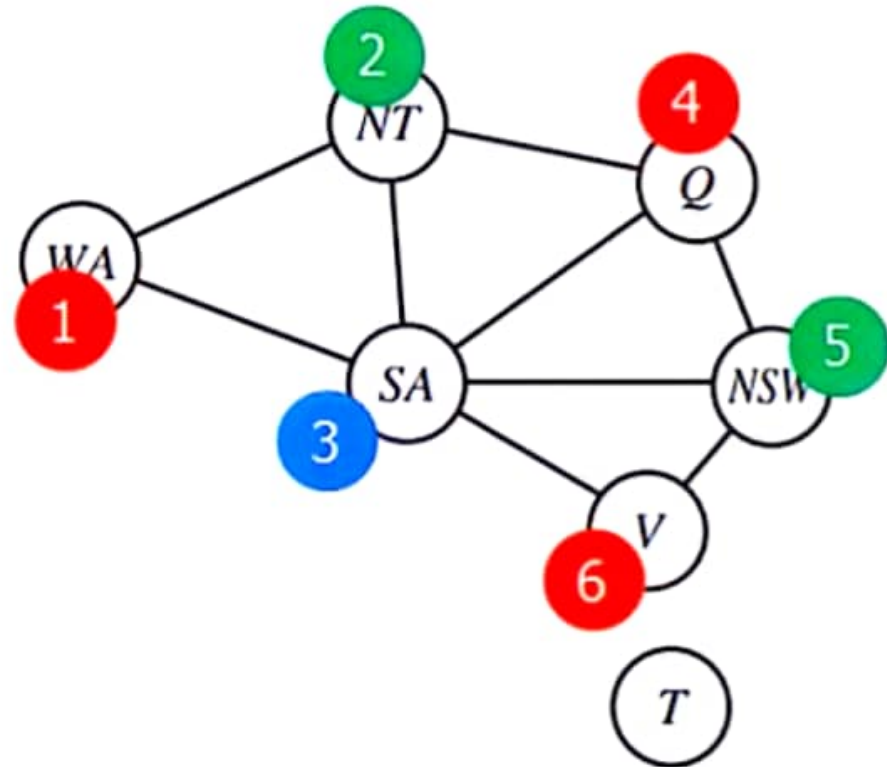
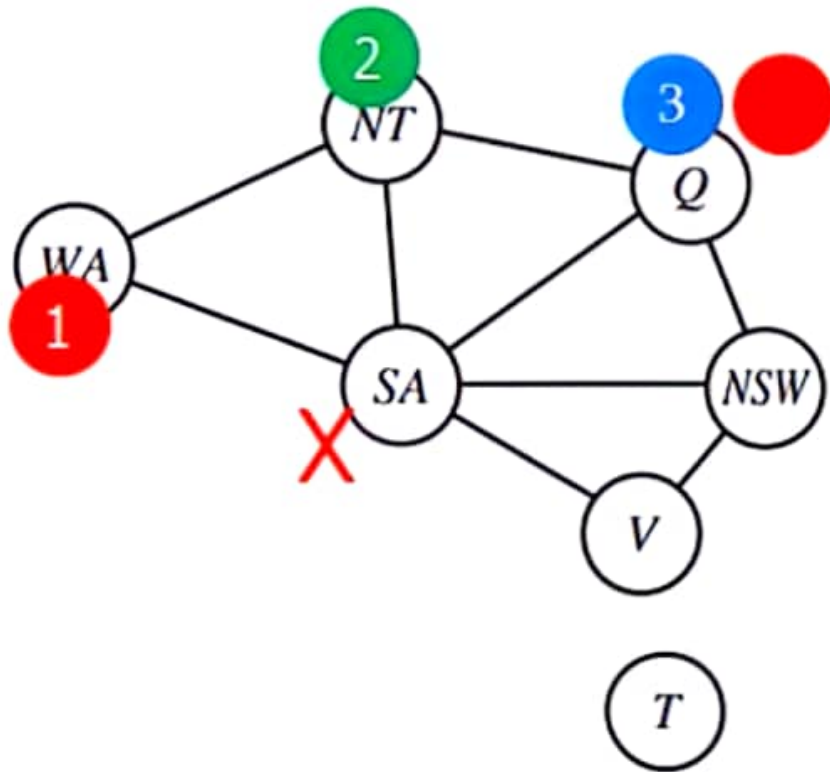


بهره‌است SA قبل از Q مقدار بگیرد. اگر $Q = \text{Blue}$ باشد آن گاه

$$D_{SA} = \{ \}$$

روال انتخاب به شکست منجر خواهد شد.

○ انتخاب متغیر بعدی در صورتی که ایستا باشد، کارایی کمی خواهد داشت:



هیوریستیک Minimum Remaining Value

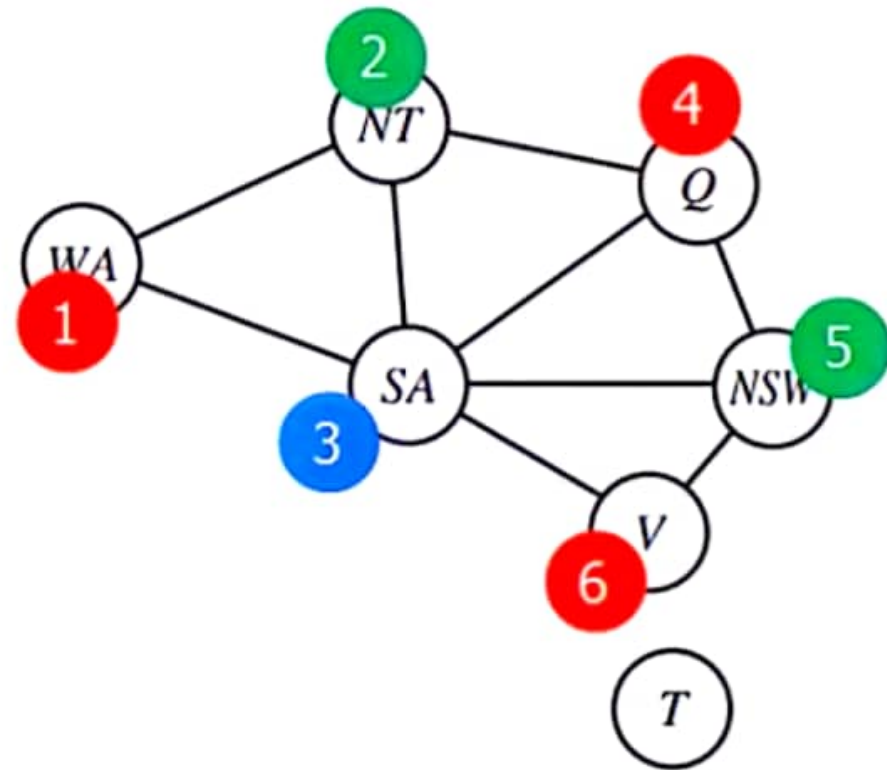
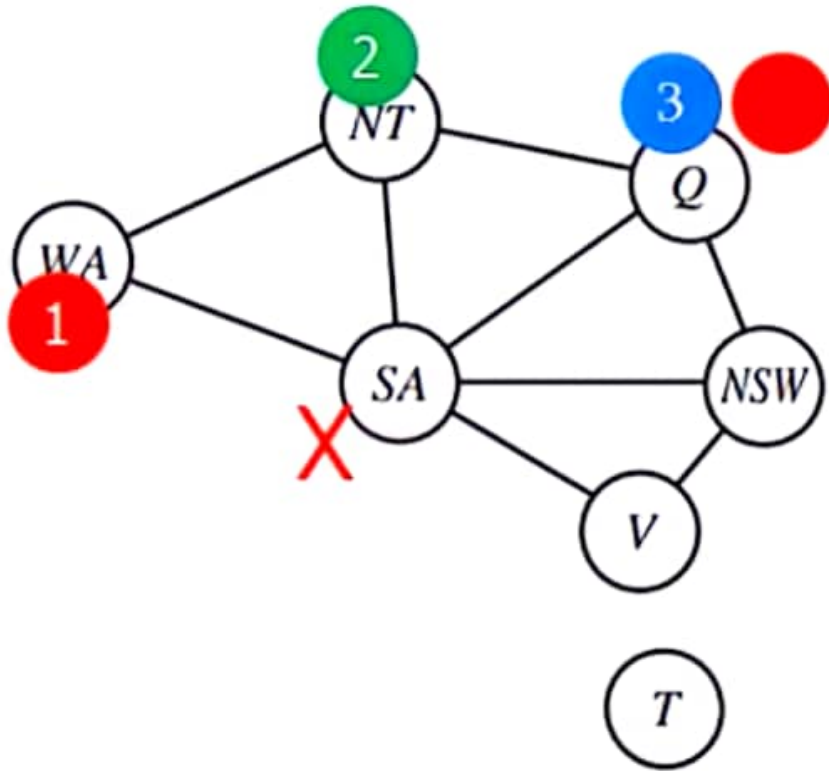
- براساس این هیوریستیک، متغیری برای انتساب انتخاب می‌شود، که دامنه مقادیر معتبر برای آن از همه کمتر باشد.
- این هیوریستیک، محدودترین متغیر (Most Constraint Variable) نیز گفته می‌شود.
- این هیوریستیک، اولین شکست (fail-first) نیز گفته می‌شود.

بهره‌است SA قبل از Q مقدار بگیرد. اگر $Q = \text{Blue}$ باشد آن گاه

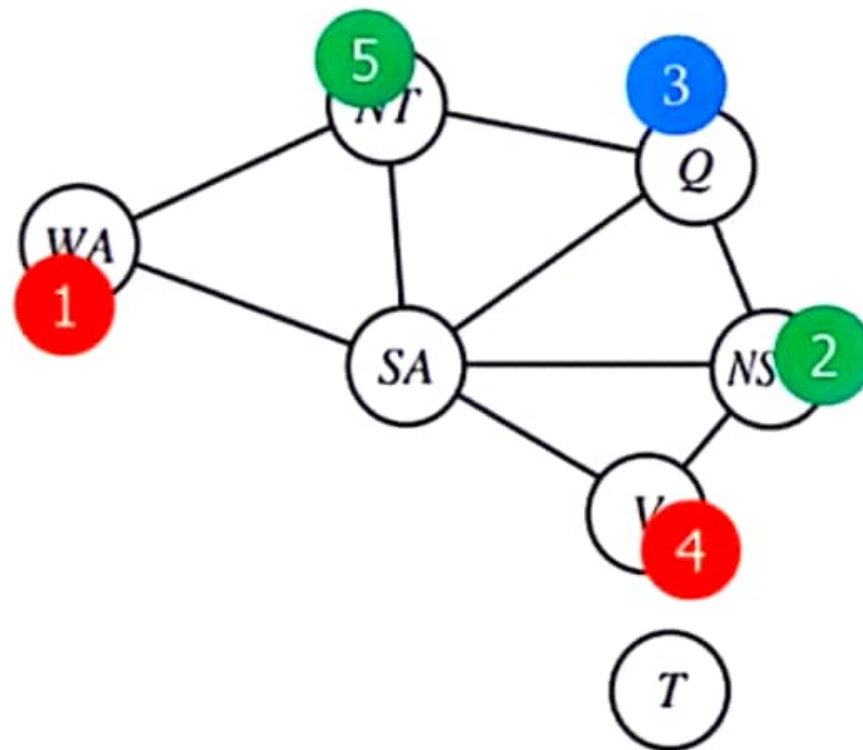
$$D_{SA} = \{ \}$$

روال انتخاب به شکست منجر خواهد شد.

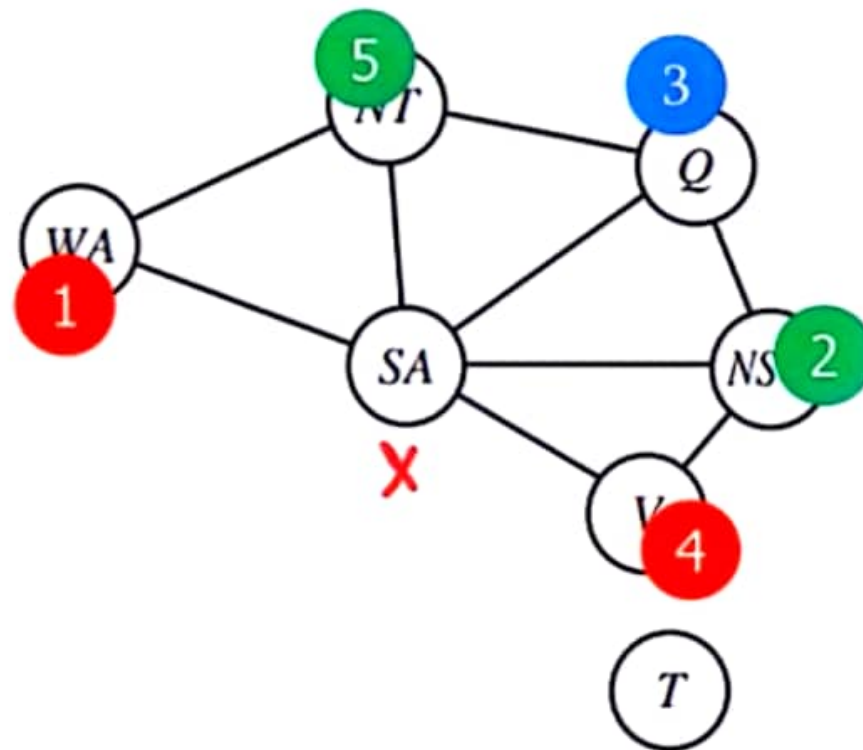
○ انتخاب متغیر بعدی در صورتی که ایستا باشد، کارایی کمی خواهد داشت:



مثال از کارایی هیوریستیک MRV



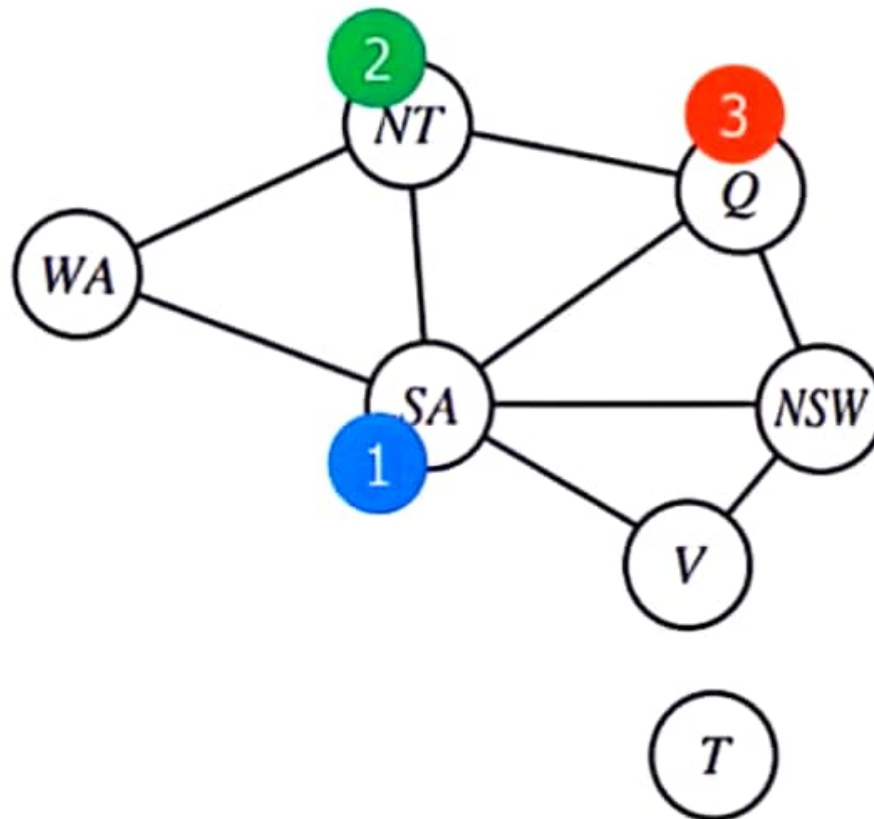
مثال از کارایی هیوریستیک MRV



Problem	Backtracking	Backtracking+MRV
USA	>1000k	>1000k
n-Queen	>40000k	13500k
zebra	3859k	1k
Random 1	415k	3k
Random 2	924k	27k

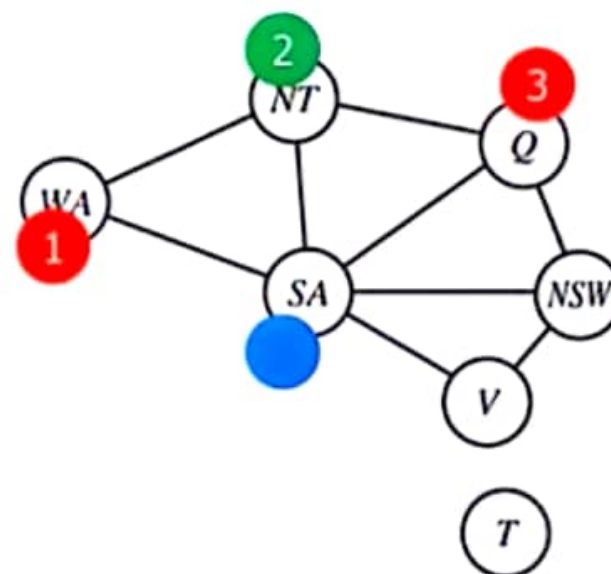
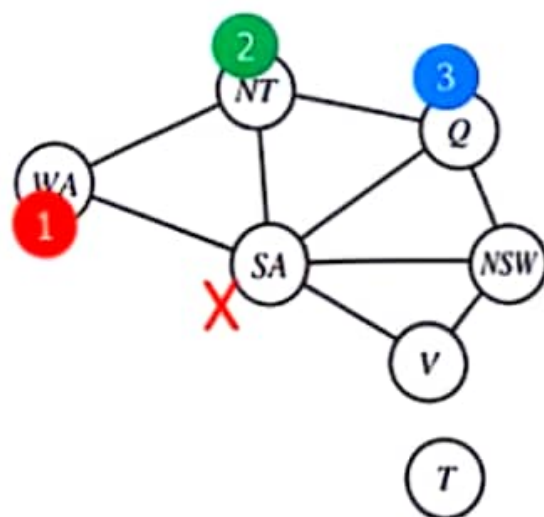
- کارایی روش عقبگرد همراه با MRV، ۳ تا ۳۰۰۰ برابر از روش عقبگرد بهتر است.
- هیوریستیک MRV، در انتخاب اولین متغیر برای انتساب، کمکی نخواهد کرد.

○ برای حل مشکل انتخاب اولین متغیر، می توان از هیوریستیک درجه استفاده نمود.



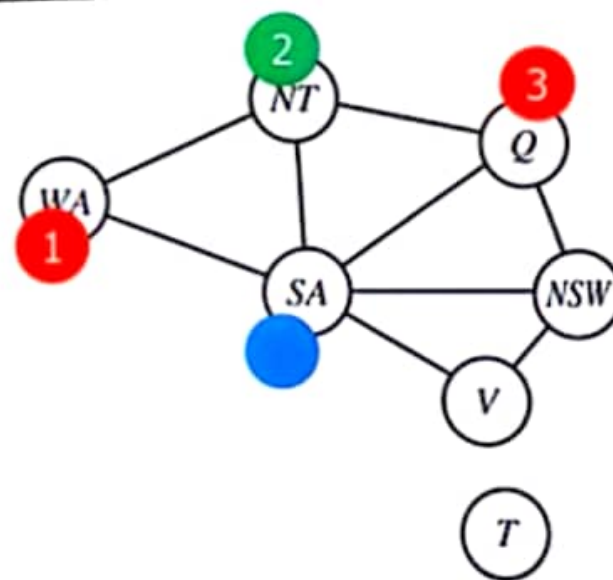
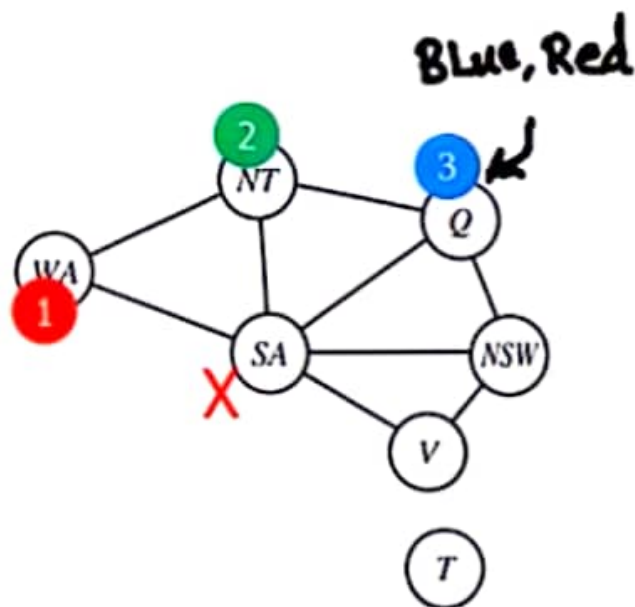
هیوریستیک مقدار با کمترین محدودکنندگی (Least Constraining Value)

- هنگام انتخاب یک متغیر، باید در مورد ترتیب امتحان کردن مقادیر برای آن تصمیم گیری نمود.
- برای یک متغیر، مقداری را انتخاب می کند، که محدودیت کمتری روی مقادیر متغیرهای همسایه آن داشته باشد.



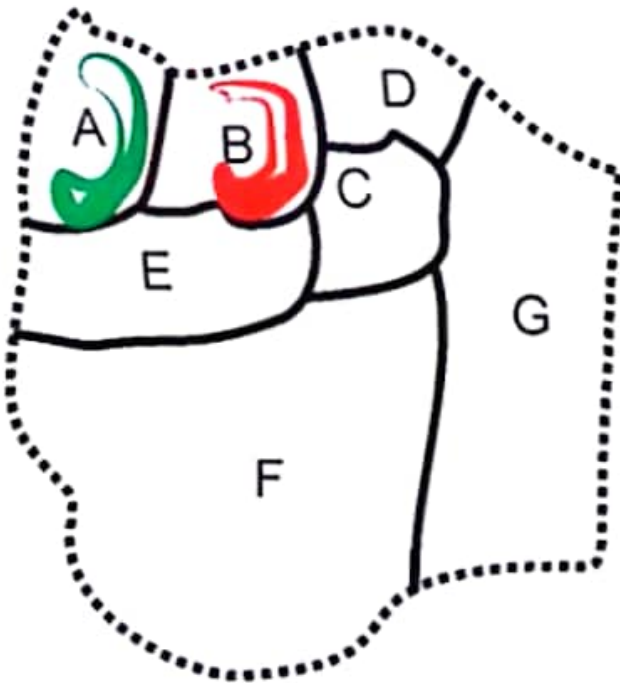
هیوریستیک مقدار با کمترین محدودکنندگی (Least Constraining Value)

- هنگام انتخاب یک متغیر، باید در مورد ترتیب امتحان کردن مقادیر برای آن تصمیم گیری نمود.
- برای یک متغیر، مقداری را انتخاب می کند، که محدودیت کمتری روی مقادیر متغیرهای همسایه آن داشته باشد.



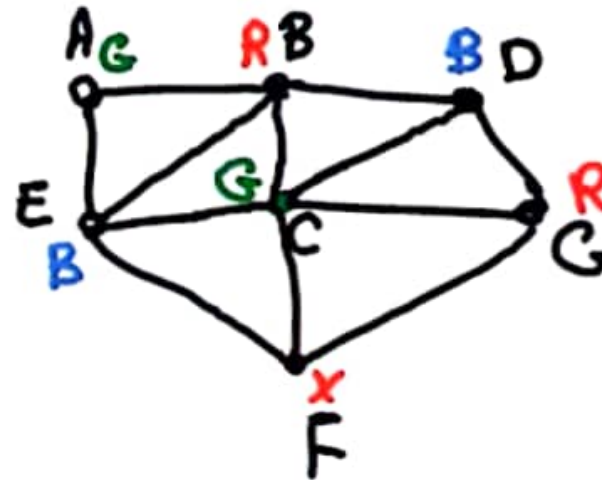
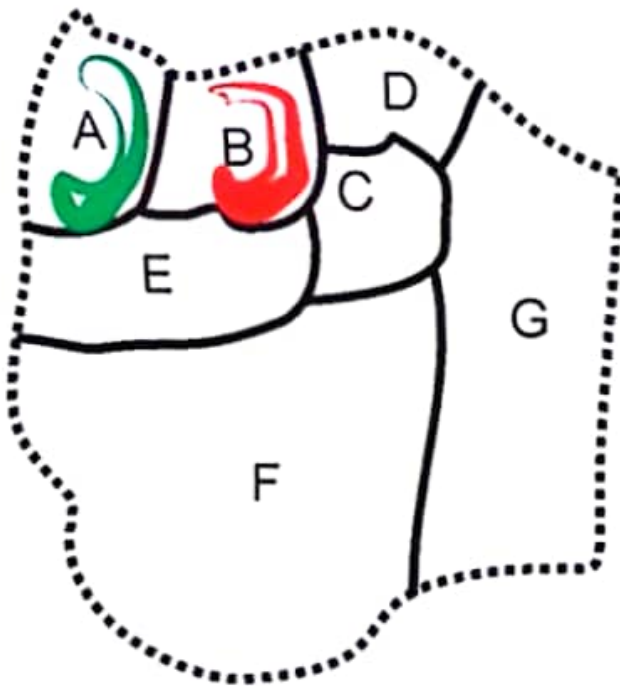
مثال

اگر بخواهیم نقشه زیر را با سه رنگ (قرمز، سبز و آبی) رنگ آمیزی کنیم، ترتیب انتخاب شهرهای باقیمانده کدام است؟



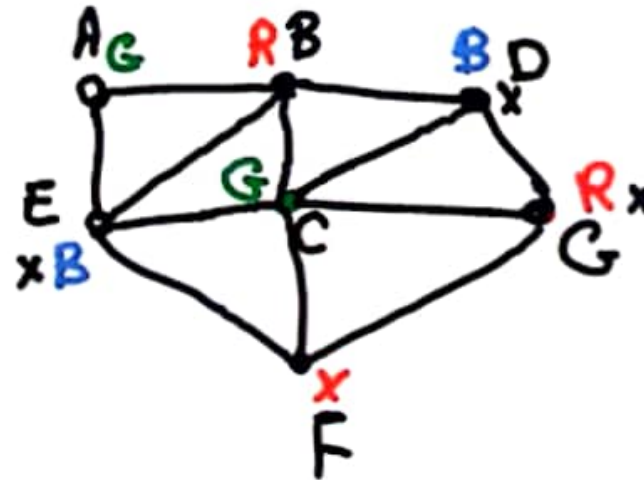
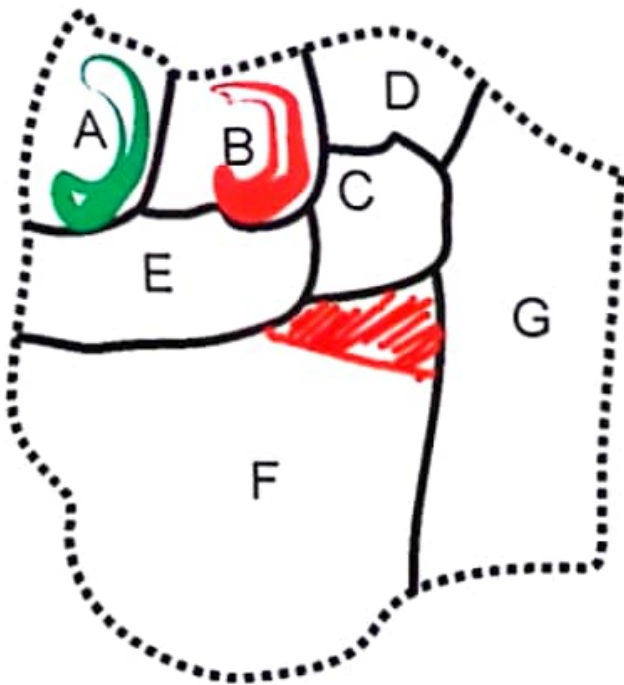
مثال

اگر بخواهیم نقشه زیر را با سه رنگ (قرمز، سبز و آبی) رنگ آمیزی کنیم، ترتیب انتخاب شهرهای باقیمانده کدام است؟



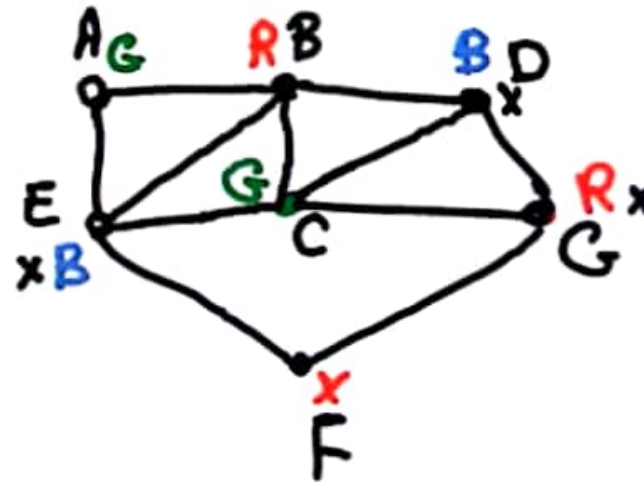
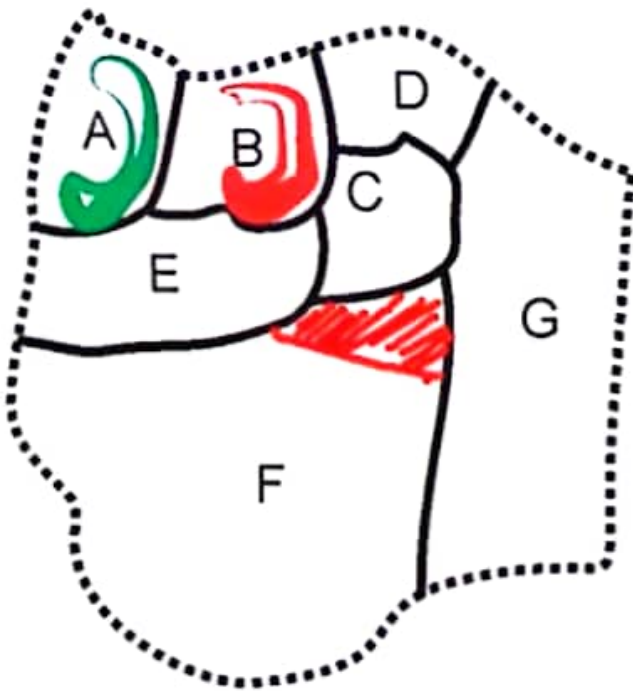
مثال

اگر بخواهیم نقشه زیر را با سه رنگ (قرمز، سبز و آبی) رنگ آمیزی کنیم، ترتیب انتخاب شهرهای باقیمانده کدام است؟



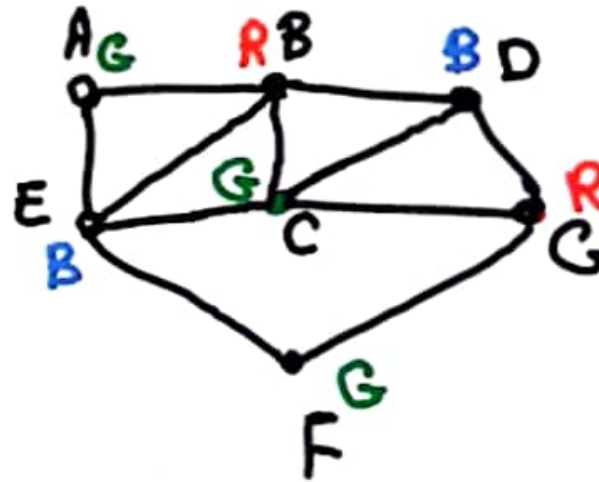
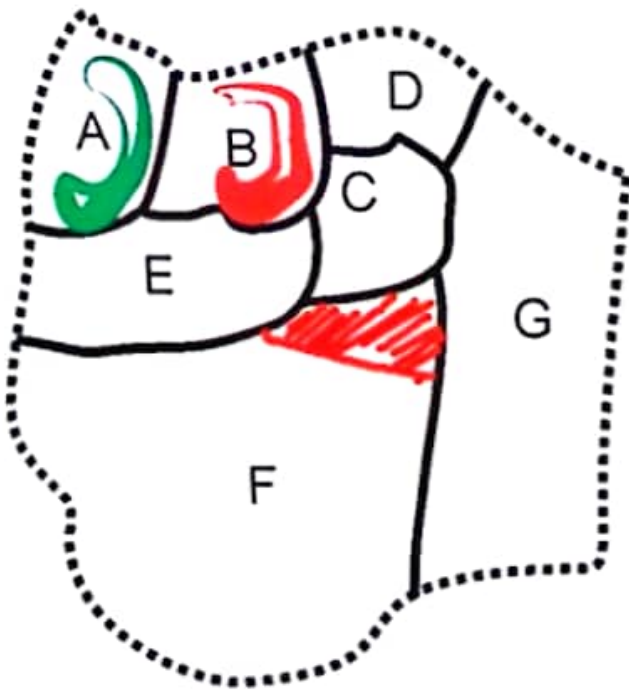
مثال

اگر بخواهیم نقشه زیر را با سه رنگ (قرمز، سبز و آبی) رنگ آمیزی کنیم، ترتیب انتخاب شهرهای باقیمانده کدام است؟



مثال

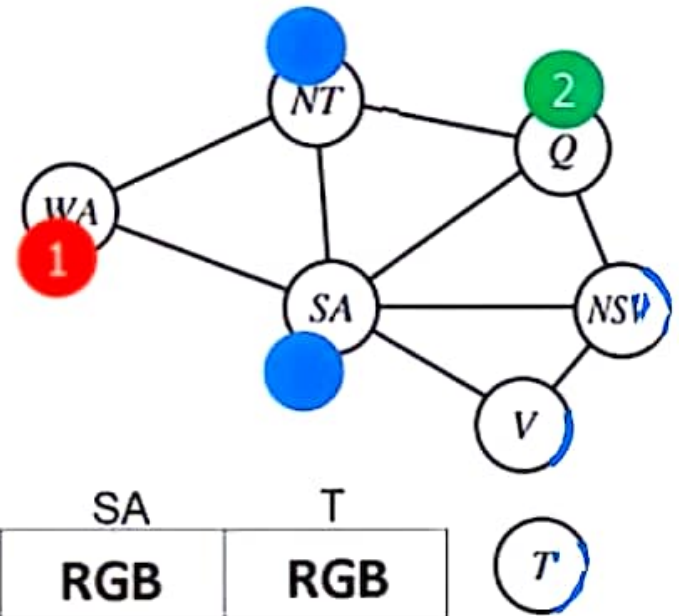
اگر بخواهیم نقشه زیر را با سه رنگ (قرمز، سبز و آبی) رنگ آمیزی کنیم، ترتیب انتخاب شهرهای باقیمانده کدام است؟



جستجوی بررسی پیشرو (Forward Checking)

- در صورتیکه محدودیت‌ها را قبل از انتخاب متغیرهای بدون مقدار بتوانیم در نظر بگیریم، آنگاه فضای جستجو کاهش می‌یابد.
- با استفاده از جستجوی بررسی پیشرو میتوان زودتر از محدودیت‌ها مطلع شد و از آنها برای کوچک کردن فضای جستجو استفاده نمود.
- در جستجوی پیشرو، هنگام مقداردهی متغیر X ، با مقدار α ، تمامی متغیرهای Y ، که از طریق یک محدودیت با X ارتباط دارند را پیدا کرده، و تمامی مقادیر ناسازگار با α را از دامنه آن‌ها حذف می‌کند.

	WA	NT	Q	NSW	V	SA	T
دامنه اولیه متغیرها	RGB	RGB	RGB	RGB	RGB	RGB	RGB
WA=Red	Ⓡ	GB	RGB	RGB	RGB	GB	RGB
Q=green	Ⓡ	B	Ⓢ	R B	RGB	B	RGB
V=blue	Ⓡ	B	Ⓢ	R	Ⓢ		RGB

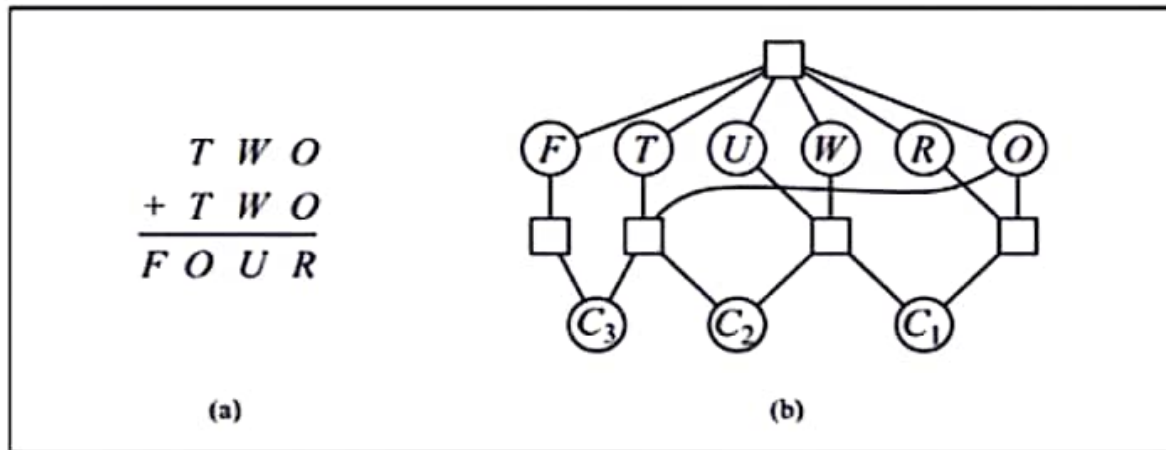


دامنه اولیه متغیرها

	WA	NT	Q	NSW	V	SA	T
	RGB	RGB	RGB	RGB	RGB	RGB	RGB
WA=Red	Ⓡ	GB	RGB	RGB	RGB	GB	RGB
Q=green	Ⓡ	B	ⓖ	R B	RGB	B	RGB
V=blue	Ⓡ	B	ⓖ	R	ⓑ		RGB

مثال

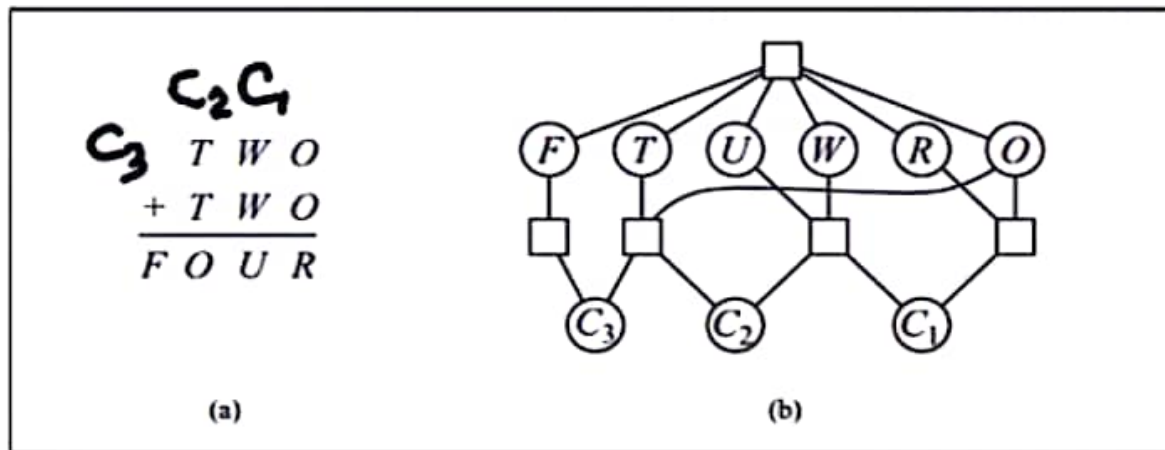
مساله معمای رمزنگاری زیر را با استفاده الگوریتم عقبگرد بکمک بررسی پیشرو و هیوریستیک های MRV و کم ترین محدود کننده مقدار حل کنید.



$$\begin{aligned}
 O+O &= R+10C_1 \\
 C_1+W+W &= U+10C_2 \\
 C_2+T+T &= O+10C_3 \\
 C_3 &= F
 \end{aligned}$$

مثال

مساله معمای رمزنگاری زیر را با استفاده الگوریتم عقبگرد بکمک بررسی پیشرو و هیوریستیک های MRV و کم ترین محدود کننده مقدار حل کنید.



$$\begin{aligned}
 O+O &= R+10C_1 \\
 C_1+W+W &= U+10C_2 \\
 C_2+T+T &= O+10C_3 \\
 C_3 &= F
 \end{aligned}$$

$$C_3 \in \{0, 1\}$$

$$C_3 = 1$$

$$F = 1$$

$$C_1, C_2 \in \{0, 1\}$$

$$C_2 = 0$$

$$C_1 = 0$$

$$O + Q = R$$

$$T + T = O + 10$$

$$\rightarrow O(\text{ع.ج.}) \rightarrow O = 4 \rightarrow R = 8$$

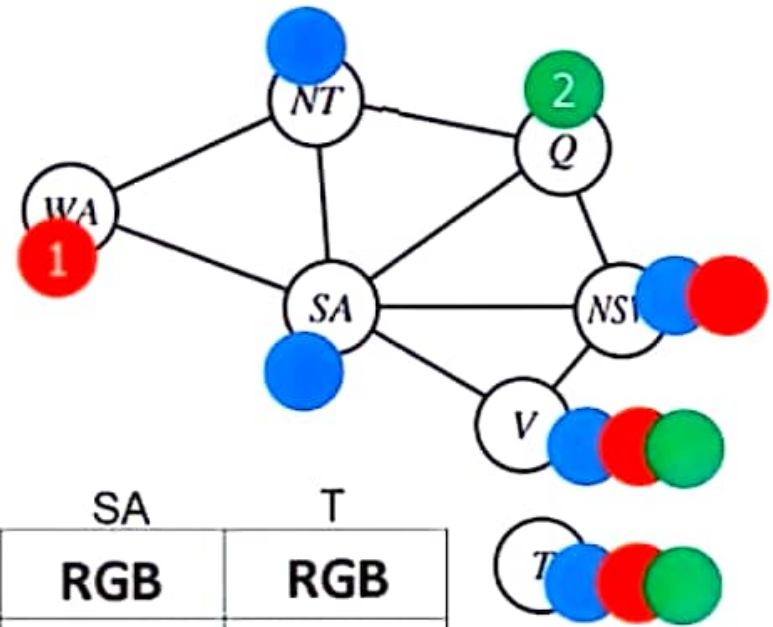
$$T + T = 4 + 10 \rightarrow T = 7$$

$$W + W = U$$

$$W = 1$$

$$U = 2$$

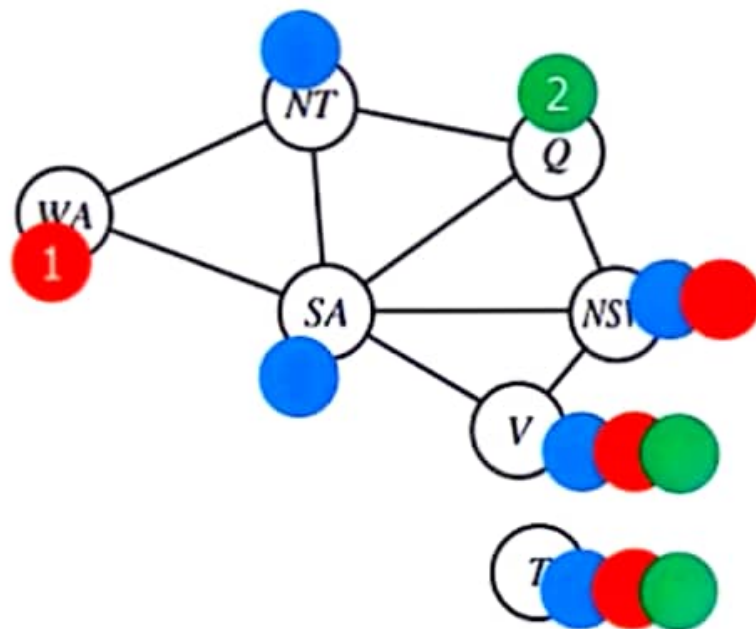
T	W	O
7	1	4



دامنه اولیه متغیرها

	WA	NT	Q	NSW	V	SA	T
دامنه اولیه متغیرها	RGB	RGB	RGB	RGB	RGB	RGB	RGB
WA=Red	Ⓡ	GB	RGB	RGB	RGB	GB	RGB
Q=green	Ⓡ	B	ⓖ	R B	RGB	B	RGB
V=blue	Ⓡ	B	ⓖ	R	ⓑ		RGB

○ در صورت انتساب زیر آیا بررسی پیشرو ناسازگاری را کشف می کند؟



Problem	Backtracking	Backtracking+MRV
USA	>1000k	>1000k
n-Queen	>40000k	13500k
zebra	3859k	1k
Random 1	415k	3k
Random 2	924k	27k

- کارایی روش عقبگرد همراه با MRV، ۳ تا ۳۰۰۰ برابر از روش عقبگرد بهتر است.
- هیوریستیک MRV، در انتخاب اولین متغیر برای انتساب، کمکی نخواهد کرد.

Problem	Backtracking	Backtracking+MRV	FC	FC+MRV
USA	>1000k	>1000k	2K	60
n-Queen	>40000k	13500k	>40000	817K
zebra	3859k	1k	35K	0.5K
Random 1	415k	3k	26K	2K
Random 2	924k	27k	77K	15K