جستجوی ممنوع

Tabu Search

Dr. Mostafa Zandieh
Shahid Beheshti University

منشاء TS

- TS در سال ۱۹۸۶توسط Fred Glover برای غلبه بر مشکل روشهای موسوم به جستجوی محلی[1] (LS) ارایه شد.
  - LS را می توان یک رویه جستجوی تکرارشونده دانست که با یک جواب شدنی شروع شده و با انجام حرکت هایی[2]، به سمت یک بهینه ی محلی هدایت می شود.
  - در اغلب مواقع، این بهینه محلی چیزی بیش از یک جواب با کیفیت متوسط نیست.
  - در LS معمولا کیفیت جواب بدست آمده به میزان زیادی بستگی به غنای حرکت های تعریف شده دارد (مشکل اساسی در رویکردهای مبتنی بر LS).

1- Local Search 2- Move

مفاهیم و اصول اساسی در TS

- برای اجتناب از دور زدن و رسیدن به جوابهایی که پیش از این بدست آمده، از حافظه ای بنام لیست ممنوع[1] استفاده می شود.
  - در این حافظه حرکات اخیر (نه جوابهای اخیر) ثبت می شود.
- در واقع یک TS ساده را می توان ترکیبی از یک حافظه کوتاه مدت[2] با LS دانست.

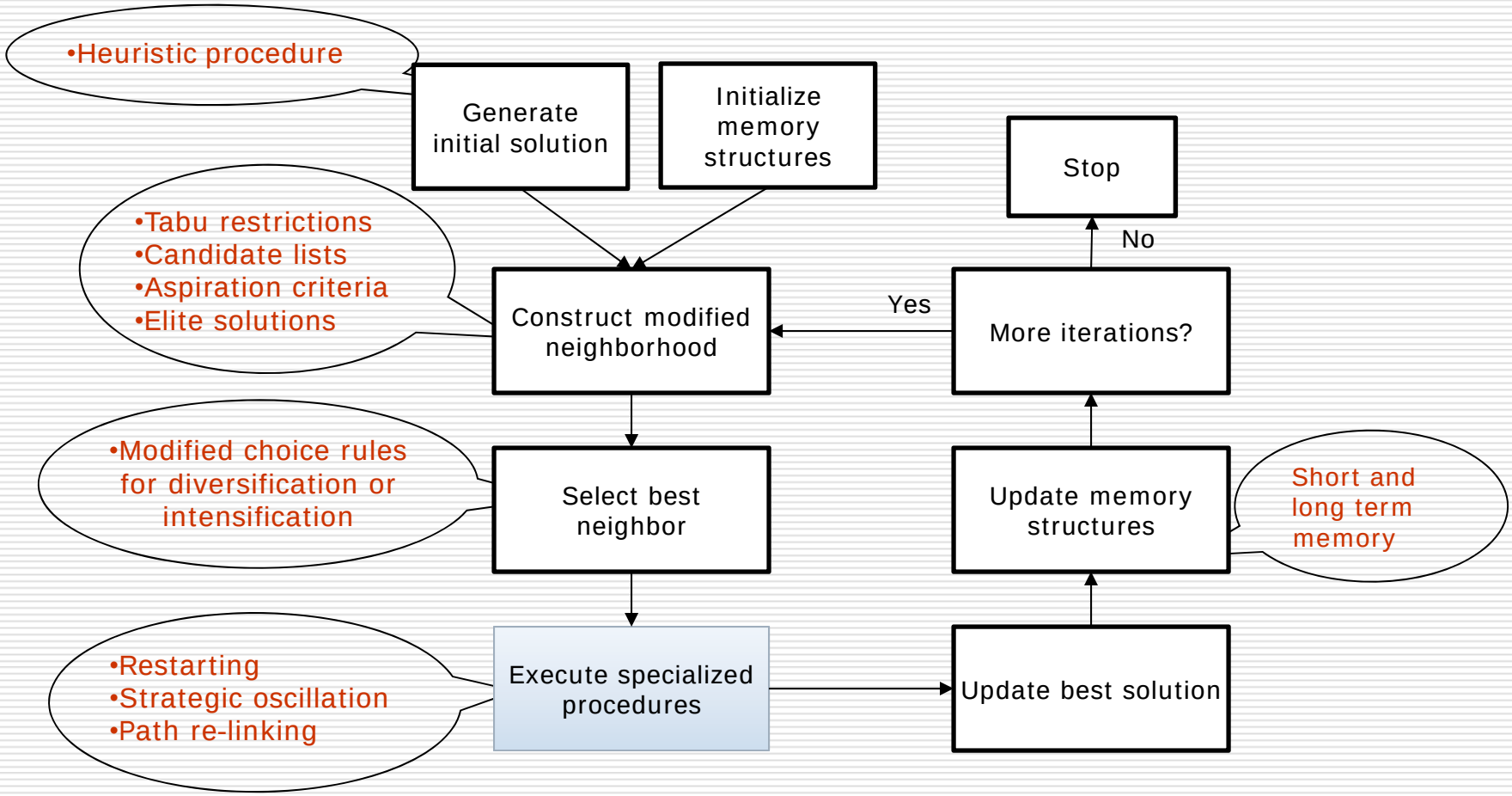
1- Tabu List 2- STM: Short Term Memory

اصول اساسی در TS

- تضمین همگرایی (تمرکزدهی) با استفاده از یک LS به عنوان زیربنای اصلی فرآیند جستجو (مثل روش HC).
- تضمین تنوع بخشی (استراتژی فرار از بهینه ی محلی) با
  - استفاده از یک لیست ممنوع[1]
  - مجاز دانستن حرکت هایی که بهبودی به همراه ندارند(تنوع بخشی)، با این شـــرط که در لیست مـمنوع قرار نداشته باشند.
- تضمین تعادل بین استرتژیهای تمرکزدهی و تنوع بخشی با آزادسازی ممنوعیت بر اساس معیارهای آرمانی

1- Tabu List

چارچوب کلی TS



اجزاء الگوریتم TS

- روشی برای نمایش جواب[1]
- تابعی برای ارزیابی جوابها (تابع هدف)
- عملگر حرکت
- مجموعه ای از حرکات کاندیدا[2]
- لیست ممنوع
- معیارهای آرمانی برای آزادسازی ممنوعیتها[3]
- شرط/شرایط توقف

1- Encoding 2- Neighborhood of Candidate Moves 3- Aspiration Criteria

برخی از علائم قراردادی

- فضای جواب

مجموعه تمام جوابهای موجه مساله را با $\boldsymbol{S}$ نمایش می دهیم.

- جواب موجه
  - یک جواب موجه متعلق به فضای جواب را با s نمایش می دهیم.
  - بهترین جواب یافت شده تاکنون را $\hat{s}$ با نمایش می دهیم.
- همسایگی

یک همسایگی برای جواب s را با $\boldsymbol{N}(s)$ نمایش می دهیم($\boldsymbol{N}(s) \subseteq \boldsymbol{S}$).

  - بعبارت دیگر مجموعه جوابهایی که با استفاده از یک انتقال، از جواب s بدست می آیند.
  - ساختار همسایه، میتواند حتی شامل تمامی فضای جواب نیز باشد. برای یک مساله خاص، نوع انتقال تعریف شده، نقشی اساسی در وسعت همسایگی بوجود آمده دارد.

- مجموعه جوابهای کاندیدا[1]

مجموعه ای از جوابها که براساس لیست ممنوع اصلاح شده است:

$$N^*(s, k)=N(s, k)\ominus TabuList$$

1- Candidate Set

فرم ابتدایی الگوریتم TS

Set *Aspiration-Criteria*; *TabuList*={ }; *TabuTenure*=*TT*; k=1;
Choose an initial solution $s \in S$
$\hat{s}=s$
Repeat
  Generate $N(s, k) \subseteq N(s)$
  Evaluate each $s \in N(s, k)$

  Modify the neighborhood $N^*(s, k)=N(s, k) \ominus$ TabuList
  If the *Aspiration-Criterion* is met **then**
    Choose the best solution $s' \in N(s, k)$
  Else
    Choose the best solution $s' \in N^*(s, k)$
  End If
  Move to $s=s'$
  If solution s is better than $\hat{s}$ **then** $\hat{s}=s$
  Update the TabuList based on *TT*
  $k=k+1$
Until stopping condition is met

جواب اولیه[1]

- تصادفی[2]
- مبتنی بر یک روش ابتکاری سازنده[3]

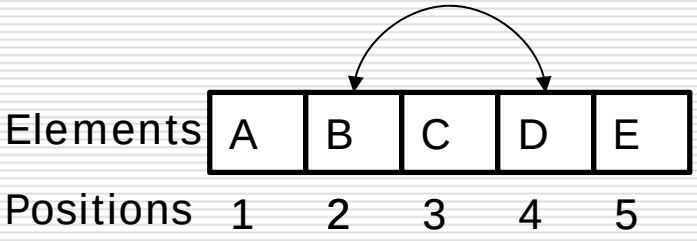
1- Initial Solution 2- Random 3- Constructive

انعطاف پذیری جستجو[1]

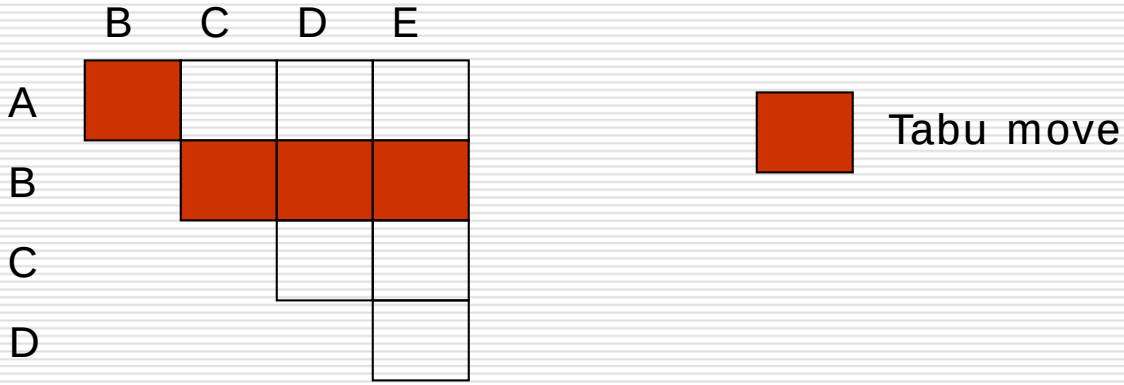
- تعداد حرکتهای قابل قبول در همسایگی جواب فعلی به عوامل زیر بستگی دارد:
  - نوع حرکت
  - قاعده فعال سازی ممنوعیت[2]
  - دوره ممنوعیت
  - معیارهای آرمانی

1- Search Flexibility 2- Tabu Activation Rule

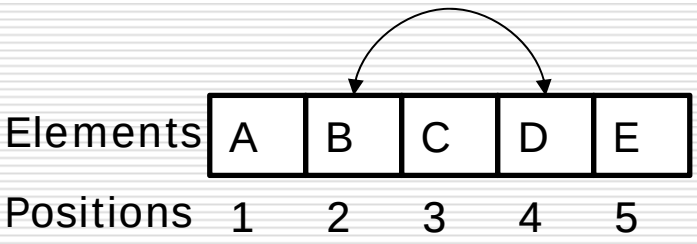
مثال ۱



Tabu Activation Rule: move (B ↔ *) is tabu

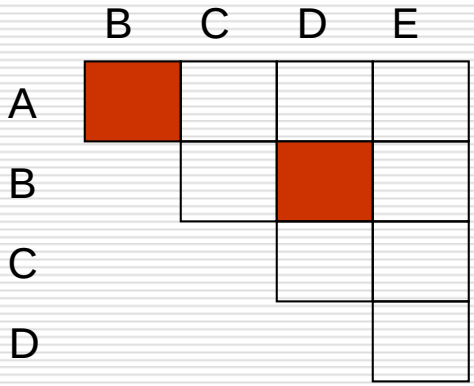


مثال ۲



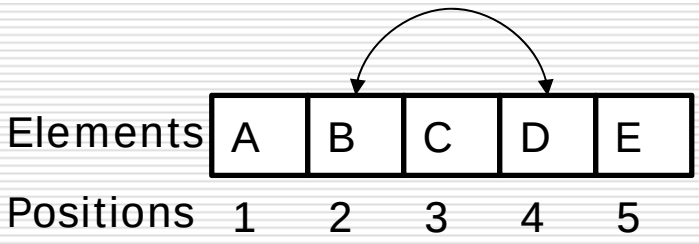
Tabu Activation Rule: move (B ↔ *) is tabu if B moves to position 2 or earlier

اگر B جابجا شود، برگشت B به
مکان قبلی اش یا ماقبل آن ممنوع است

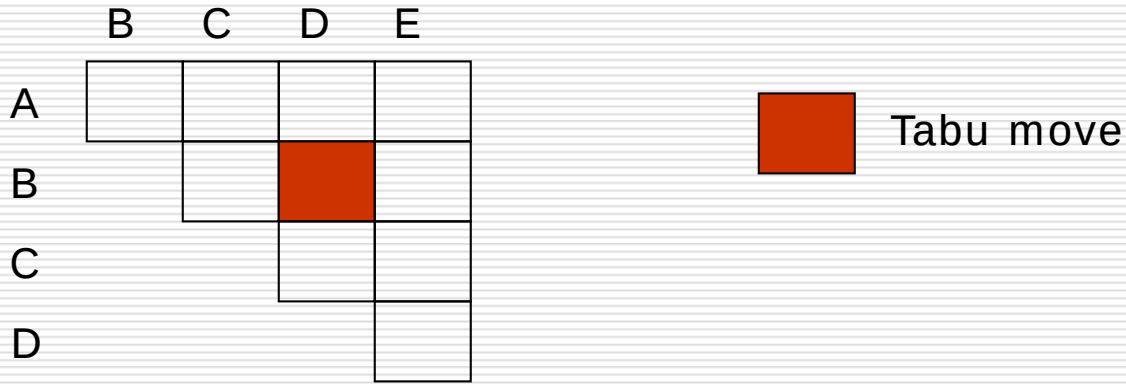


Tabu move

مثال ۳



Tabu Activation Rule: move (B ↔ D) is tabu



برخی از معیارهای آرمانی

- مبتنی بر تابع هدف
  - حرکتی که جواب حاصل از آن از بهترین جواب فعلی بهتر است[1](مرسومترین).
  - حرکتی که جواب حاصل از آن از بهترین چند جواب اخیر، بهتر است[2].
  - حرکتی که جواب حاصل از آن در بین جوابهای با خاصیتی معین، بهترین است[3].
  - حرکتی که جواب حاصل از آن بیشترین شباهت را با بهترین جواب فعلی دارد[4].
  - حرکتی که جواب حاصل از آن کمترین شباهت را با جواب فعلی دارد[5].
- مبتنی بر جهت جستجو(در مواقعی که امکان انجام هیچ حرکتی وجود ندارد)
  - حرکتی که در صف خروج از لیست ممنوع از همه جلوتر است.

1- Global Best 2- Recent Best 3- Region Best 4- Intensification 5- Diversification

دوره ممنوعیت[1]

- تعداد تکرارهایی که یک حرکت ممنوع است.
- این تعداد ممکن است استاتیک/پویا باشد.
  - معمولا بکارگیری دوره ممنوعیت استاتیک ساده تر است.
  - درحالتی که جوابهایی خاص دائم تکرار می شوند، دوره ممنوعیت باید افزایش یابد.

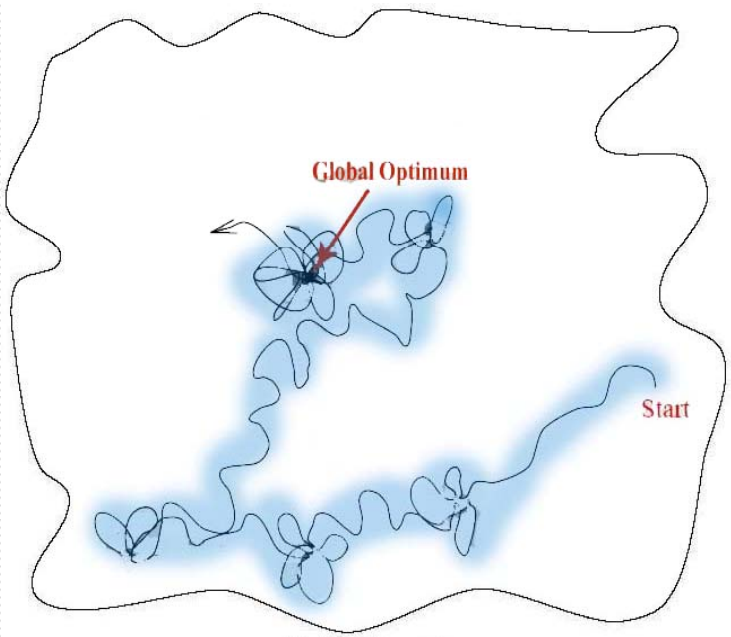
1- Tabu Tenure

شرط توقف

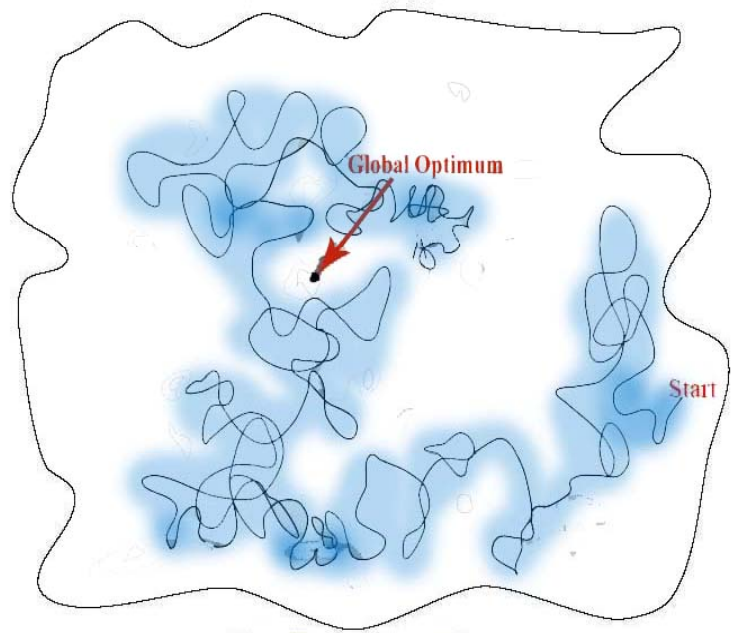
- توقف بعد از تعداد تکرارهای از قبل تعیین شده (*MaxIte*).
- توقف بعد از زمان محاسباتی از قبل تعیین شده (*MaxCput*).
- توقف بعد از تعدادی از تکرارهای بدون بهبود (*MaxIteWithoutImp*).
- توقف بعد از رسیدن به نقطه خاصی از فضای جستجو (*OFThreshold*).

1- Tabu Tenure

TS vs. SA



Tabu Search



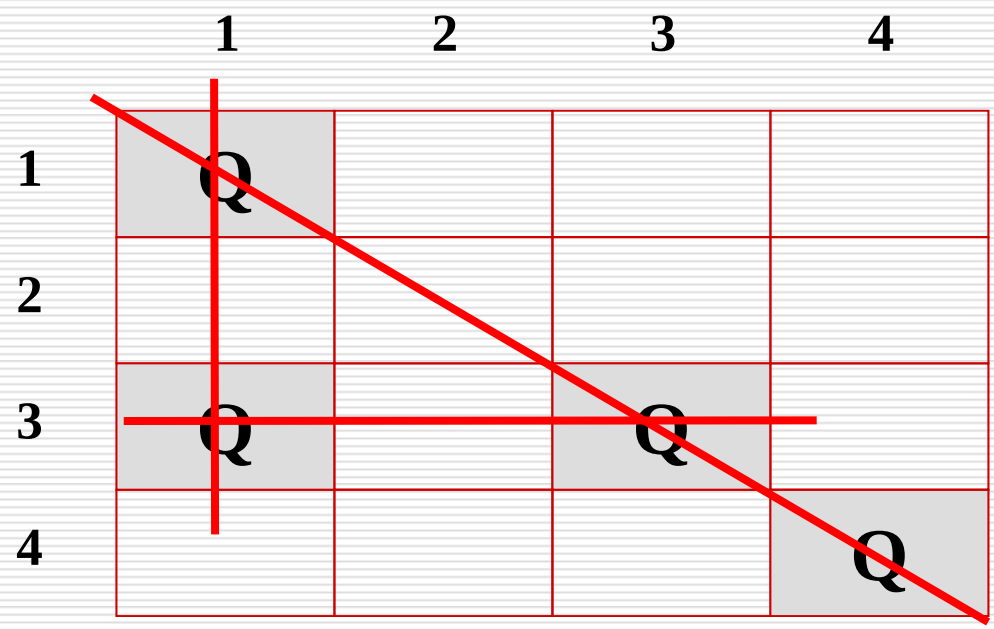
Simulated Annealing

- پراکندگی بیشتر
- سرعت همگرایی کمتر

- معمولا سرعت همگرایی بیشتر
- امکان گیر افتادن در دام بهینه محلی

مثال ٤: مساله n وزیر

- میخواهیم n وزیر را در یک صفحه شطرنج $n \times n$ بگونه ای بچینیم که هیچ دو وزیری نتوانند یکدیگر را بزنند.



روش نمایش جواب

- مساله n وزیر را می توان بصورت یک مساله جایگشتی[1] نمایش داد.
- فرض کنید که وزیرها توسط حرف i شماره گذاری شده باشند.
- همچنین فرض کنید که وزیر i در ردیف i قرار گرفته باشد.
- فرض کنید شماره ستونی که وزیر ردیف i در آن قرار گرفته را با $\pi(i)$ نمایش دهیم. آنگاه یک جواب بصورت $\Pi = \{\pi(1),\pi(2),...\pi(n)\}$ قابل نمایش است.

$\Pi = \{3,1,4,2\}$

	1	2	3	4
1			Q	
2	Q			
3				Q
4		Q		

1- Permutation

همسایگی

- جابجایی دوتایی[1]، یکی از مرسومترین روشهای تعریف همسایگی در مسائل جایگشتی است.

Current solution

Π =

4	5	3	6	7	1	2

Move operator

Swap queens 2 and 6

Π' =

4	1	3	6	7	5	2

1- Pairwise Swap

ساختار لیست ممنوع

- برای جلوگیری از حرکات تکراری، همه جابجایی های دوتایی برای ۳ تکرار در لیست ممنوع قرار می گیرند.

	2	3	4	5	6	7
1						
2					*	
3						
4						
5						
6						

Tabu Memory

* تعداد تکرارهای باقیمانده ای که امکان جابجایی موقعیت وزیرهای متناظر با این سلول وجود ندارد.

Iteration - 0

Current solution

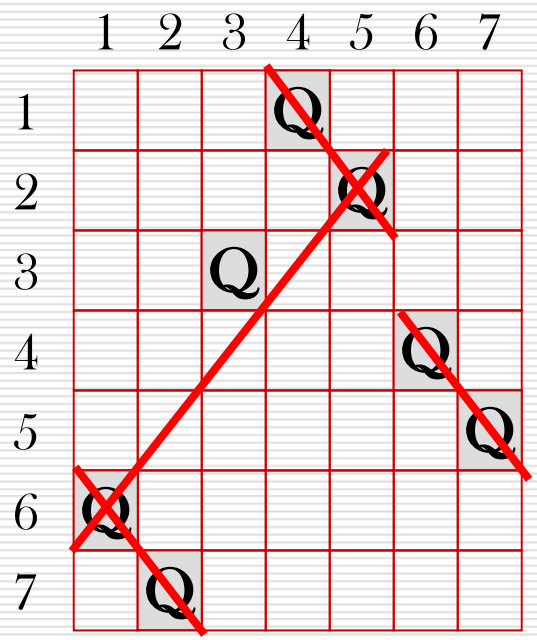
4	5	3	6	7	1	2

Tabu Memory

	2	3	4	5	6	7
1						
2						
3						
4						
5						
6						

Swap		Value
1	7	-2
2	4	-2
2	6	-2
5	6	-2
1	5	1

Top 5 (of 21) Candidates



تعداد برخوردها = 4

Iteration - 1

Current solution

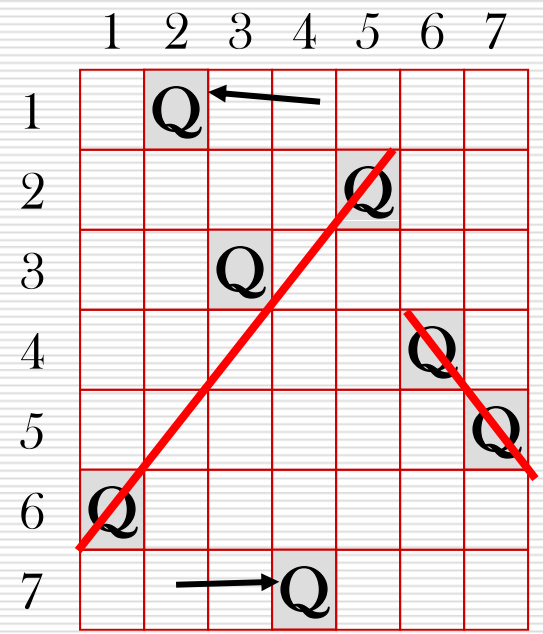
2	5	3	6	7	1	4

Tabu Memory

	2	3	4	5	6	7
1						3
2						
3						
4						
5						
6						

Swap		Value
2	4	-1
1	6	0
2	5	0
1	2	1
1	3	1

Top 5 (of 20) Candidates



2 = تعداد برخوردها

Iteration - 2

Current solution

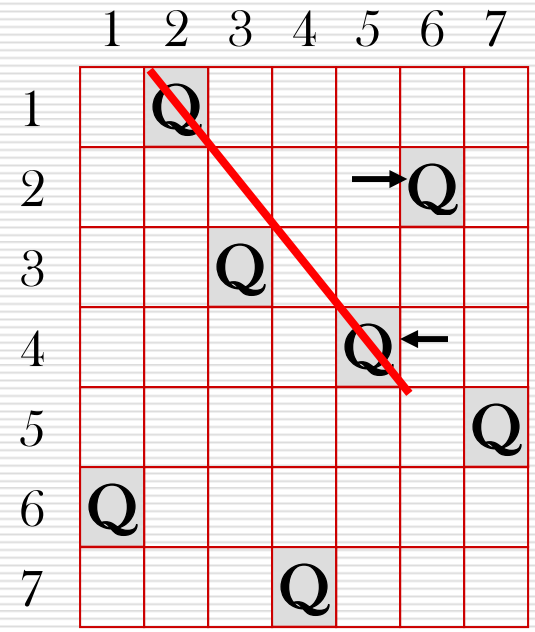
2	6	3	5	7	1	4

Tabu Memory

	2	3	4	5	6	7
1						2
2			3			
3						
4						
5						
6						

Swap		Value	
1	3	0	
1	7	1	Tabu
2	4	1	Tabu
4	5	1	
6	7	1	

Top 5 (of 19) Candidates



1 = تعداد برخوردها

Iteration - 3

Current solution

3	6	2	5	7	1	4

Tabu Memory

	2	3	4	5	6	7
1		3				1
2			2			
3						
4						
5						
6						

Top 5 (of 18) Candidates

Swap		Value	
1	3	0	Tabu
1	7	0	Tabu
5	7	1	
6	7	1	
1	2	2	

	1	2	3	4	5	6	7
1		→	Q				
2						Q	
3		Q	←				
4					Q		
5							Q
6	Q						
7				Q			

تعداد برخوردها = 1

Iteration - 4

Current solution

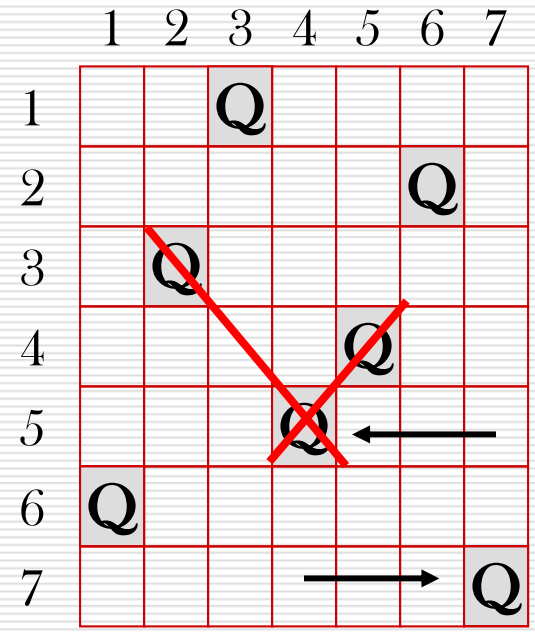
3	6	2	5	4	1	7

Tabu Memory

	2	3	4	5	6	7
1		2				
2			1			
3						
4						
5						3
6						

Swap		Value	
4	7	-1	
5	7	-1	Tabu
1	5	0	
2	5	0	
2	4	2	Tabu

Top 5 (of 18) Candidates



تعداد برخوردها = 2

Iteration - 5

Current solution

3	6	2	7	4	1	5

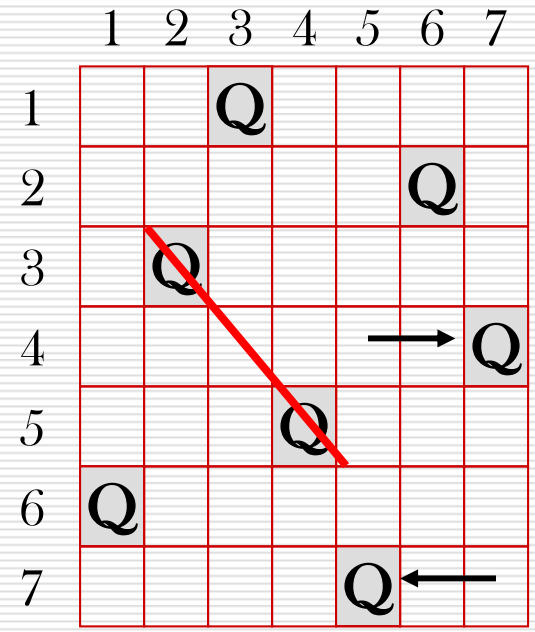
* Satisfies Aspiration Criterion

Tabu Memory

	2	3	4	5	6	7
1		1				
2						
3						
4						3
5						2
6						

Top 5 (of 18) Candidates

Swap		Value	
1	3	-1	*Tabu
1	6	0	
5	7	0	Tabu
5	6	1	
1	7	2	



1 = تعداد برخوردها

Iteration - 6

Current solution

2	6	3	7	4	1	5

Tabu Memory

	2	3	4	5	6	7
1						
2						
3						
4						2
5						1
6						

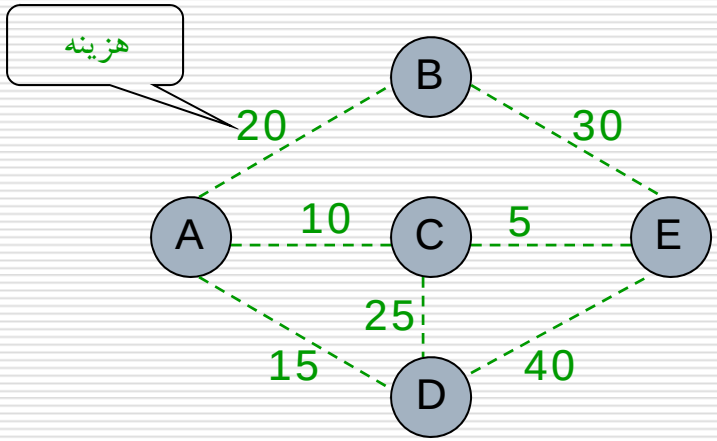
	1	2	3	4	5	6	7
1		Q	←				
2						Q	
3		→	Q				
4							Q
5				Q			
6	Q						
7					Q		

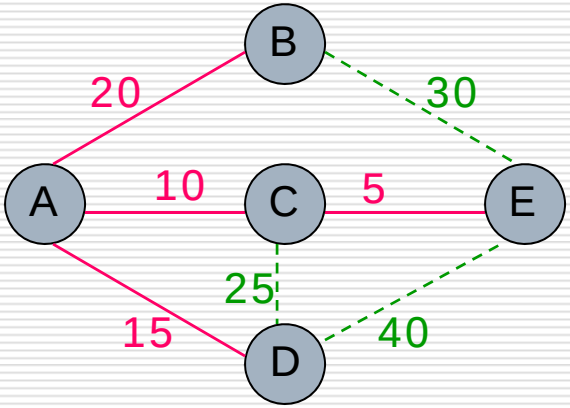
تعداد برخوردها = 0

Search Terminates

مثال ۵: مساله کوتاهترین درخت گسترده با محدودیت

هدف: تمام گرهها با کمترین هزینه به یکدیگر متصل شوند.



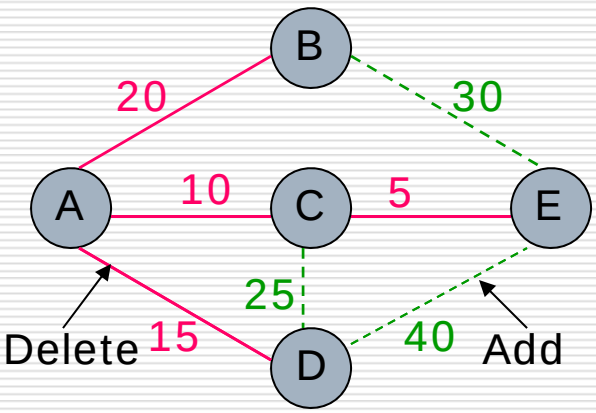


یک جواب بهینه بدون توجه به محدودیتها

محدودیت ۱: ارتباط AD فقط در صورتی که ارتباط DE برقرار باشد، میتواند وجود داشته باشد(جریمه ۱۰۰).
محدودیت ۲: حداکثر یکی از ارتباطات AD ، CD و AB میتواند وجود داشته باشد(جریمه ۱۰۰واحدی برای انتخاب ۲ ارتباط از ۳ ارتباط – و جریمه ۲۰۰ واحدی برای انتخاب ۳ ارتباط).

Iteration 1

Cost = 50 + 200 (constraint penalties)



Add	Delete	Cost
BE	CE	75+200=275
BE	AC	70+200=270
BE	AB	60+100=160
CD	AD	60+100=160
CD	AC	65+300=365
DE	CE	85+100=185
DE	AC	80+100=180
DE	AD	**75+0=75**

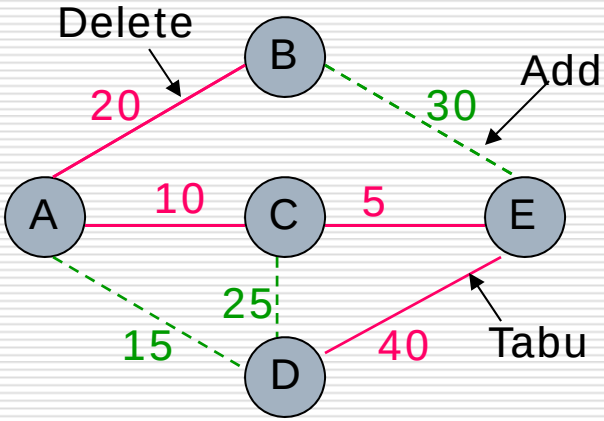
New cost = 75 (iteration 2)

(local optimum)

Constraints 1: Link AD can be included only if link DE also is included. (penalty:100)
Constraints 2: At most one of the three links – AD, CD, and AB – can be included. (Penalty of 100 if selected two of the three, 200 if all three are selected.)

Tabu list: DE

Iteration 2 Cost = 75



Add	Delete	Cost
AD	DE*	Tabu move
AD	CE	85+100=185
AD	AC	80+100=180
BE	CE	100+0=100
BE	AC	95+0=95
BE	AB	**85+0=85**
CD	DE*	60+100=160
CD	CE	95+100=195

**** A tabu move will be considered only if it would result in a better solution than the best trial solution found previously (Aspiration Condition)***

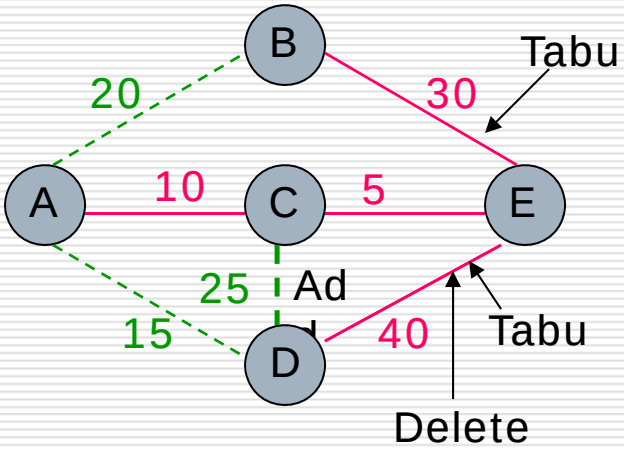
Iteration 3 new cost = 85 Escape local optimum

Constraints 1: Link AD can be included only if link DE also is included. (penalty: 100)
Constraints 2: At most one of the three links – AD, CD, and AB – can be included. (Penalty of 100 if selected two of the three, 200 if all three are selected.)

Tabu list: DE & BE

Iteration 3

Cost = 85

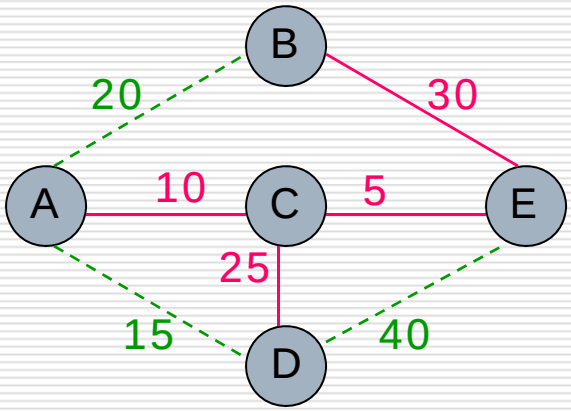


Add	Delete	Cost
AB	BE*	Tabu move
AB	CE	100+0=100
AB	AC	95+0=95
AD	DE*	60+100=160
AD	CE	95+0=95
AD	AC	90+0=90
CD	DE*	**70+0=70**
CD	CE	105+0=105

**** A tabu move will be considered only if it would result in a better solution than the best trial solution found previously (Aspiration Condition)***

Iteration 4 new cost = **70** Override tabu status

Constraints 1: Link AD can be included only if link DE also is included. (penalty:100)
Constraints 2: At most one of the three links – AD, CD, and AB – can be included. (Penalty of 100 if selected two of the three, 200 if all three are selected.)



Optimal Solution

Cost = 70

Additional iterations only find

inferior solutions

References

[1] Glover, F., Kelly, J. P., and Laguna, M. 1995. Genetic Algorithms and Tabu Search: Hybrids for Optimization. *Computers and Operations Research*. Vol. 22, No. 1, pp. 111 – 134.

[2] Glover, F. and Laguna, M. 1997. Tabu Search. Norwell, MA: Kluwer Academic Publishers.

[3] Hanafi, S. 2001. On the Convergence of Tabu Search. Journal of Heuristics. Vol. 7, pp. 47 – 58.

[4] Hillier, F.S. and Lieberman, G.J. 2005. Introduction to Operations Research. New York, NY: McGraw-Hill. 8th Ed.

[5] Ji, M. and Tang, H. 2004. Global Optimizations and Tabu Search Based on Mamory. *Applied Mathematics and Computation*. Vol. 159, pp. 449 – 457.

[6] Pham, D.T. and Karaboga, D. 2000. Intelligent Optimisation Techniques – Genetic Algorithms, Tabu Search, Simulated Annealing and Neural Networks. London: Springer-Verlag.

[7] Reeves, C.R. 1993. Modern Heuristic Techniques for Combinatorial Problems. John Wiley & Sons, Inc.