



Original Article: ALGORITMO DI RICERCA APPLICAZIONE PERCORSI PIÙ BREVI IN UN GRAFO PESATO PER COSTI DI TRENO A LUNGA PERCORRENZA

Citation

Grishin E.V., Irinina Yu.S., Dodonova N.L. Algoritmo di Ricerca Applicazione Percorsi Più Brevi in un Grafo Pesato per Costi di Treno a Lunga Percorrenza. *Italian Science Review*. 2015; 1(22). PP. 218-221. Available at URL: <http://www.ias-journal.org/archive/2015/january/Grishin.pdf>

Authors

Evgeniy V. Grishin, Samara State Aerospace University named after academician S.P. Korolyov, Russia.
Yulia S. Irinina, Samara State Aerospace University named after academician S.P. Korolyov, Russia.
Natalia L. Dodonova, Samara State Aerospace University named after academician S.P. Korolyov, Russia.

Submitted: January 04, 2015; Accepted: January 17, 2015; Published: January 30, 2015

Attualmente l'azienda è in una fase quando si viaggia in un altro paese o in un'altra città non è difficile. Fatturato del commercio presentata entro non un particolare della città, e il paese o anche in tutto il mondo. Trasporto di merci fanno parte della vita di tutti i giorni, non solo le aziende professionali, ma anche semplici cittadini: mandano pacchi ad altre regioni, beni acquistati da altre città e dall'estero. La gente spesso viaggiare da una città all'altra. Anche diverse aziende hanno a spendere un sacco di soldi per pagare i propri dipendenti viaggiano ad altre città. In questo lavoro, consideriamo questo problema urgente in questo momento, come risparmiare sui viaggi di lungo percorso civili, in particolare sui treni. Come continuazione e il miglioramento della applicabilità pratica di questo lavoro in futuro abbiamo in programma di prendere in considerazione non solo viaggio civile, ma anche il trasporto merci. Per studiare il percorso intrapreso da Samara a Mosca. Tutte le possibili vie di treni sono rappresentati come un grafico, in cui i vertici - stazione bordi - percorso tra le stazioni, il peso della

nervatura - il prezzo di un biglietto da una stazione all'altra.

Accessibile da Samara a Mosca, selezionando uno dei tre treni: il numero 55, il numero 121E e 131U [3]. Considerato nella stazione sono mostrati in tabella 1.

Schematicamente, un grafico è il seguente. Percorso Verde - 55 del treno, giallo - 131U, arancione - 121E.

Collegato da un numero vertici bordo 1 (Samara), con tutti (Tabella 2).

E effettuata da tutti i vertici del bordo di sommità 33 (Moscow) (Tabella 3).

Tutti i dati inseriti. Come metodo di trovare un algoritmo soluzione Bellman - Ford: un algoritmo per trovare il percorso più breve in un grafo pesato [1]. Egli trova il percorso più breve da un vertice a tutti gli altri. Implementazione software viene eseguito in modo indipendente. La pseudocodice dell'algoritmo è il seguente:

```
for v ∈ V
do d[v] ← ∞
d[s] ← 0
for i ← 1 to |V| - 1
do for (u, v) ∈ E
```

```

if  $d[v] > d[u] + \omega(u, v)$  then  $d[v] \leftarrow$ 
 $d[u] + \omega(u, v)$ 
return  $d$ ; [2]

```

Applicando l'algoritmo al grafico del problema, la soluzione risulta che il viaggio più economica costa 2770 rubli (Figura 2). Va notato che una delle linee del biglietto 33 picchi sono 3250, 2800 e 2950, rispettivamente.

Percorso la soluzione risultante sarà il seguente:

Cioè, è necessario ottenere 1-21 vertici per cambiare treno lì e arrivare in finale.

Riassumendo, è necessario dire che era chiaramente considerato l'uso della teoria grafico in pratica. In particolare, è stato trovato il modo più economico per viaggiare in treno da Samara a Mosca. Soluzioni Stroke dimostra pienamente l'efficacia del modello di vista come un grafico e l'applicazione pratica della algoritmo per trovare il percorso più breve, in particolare algoritmo Bellman-Ford. Salvataggio ricevuto relativamente piccola,

ma con un aumento del numero di persone che hanno bisogno di ottenere da un punto all'altro, il suo valore aumenta e diventa apprezzabile, che è importante per le grandi aziende, in quanto consente loro di ridurre i costi e le spese. In futuro, il problema sarà discusso in relazione al trasporto. Ovviamente, per un chilogrammo di risparmio di trasporto merci sarà piccolo, ma aumentando il peso fino a decine o centinaia di tonnellate, saranno ottenuti sufficientemente grandi valori che le aziende possono risparmiare denaro applicando l'unica rappresentazione del modello come un grafico e un algoritmo elementare per trovare il percorso più breve.

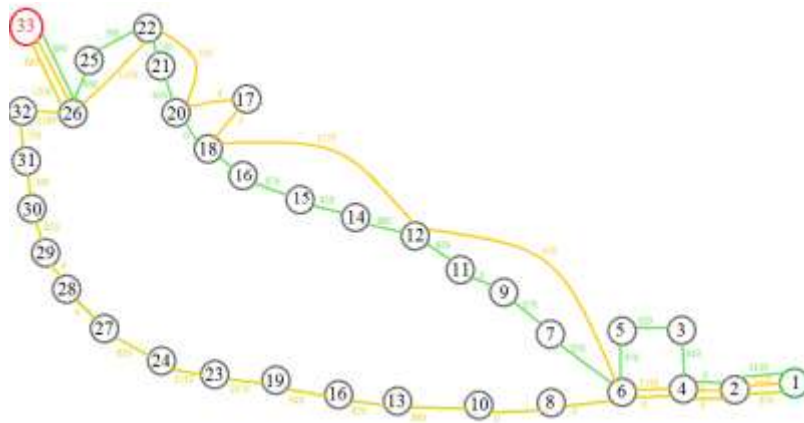
References:

1. Basaker R., Saati T. 1974. Finite graphs and networks. 368p.
2. Belov V.V., Vorobyev E.M., Shatalov V.E. 1976. Theory graphs. 392p.
3. Site "Russian Railways".

La stazione

1. Samara	10. Klyuchiki	19. Penza	28. Sosedka
2. Novokuybyshevsk	11. Bazarnaya	20. Kovytkino	29. Vernadovka
3. Bezenchuk	12. Inza	21. Pot'ma	30. Morshansk
4. Chapaevsk	13. Kuznetsk	22. Zubov Glade	31. Verda
5. Obsharovki	14. Nochka	23. Belinskaya	32. Ryazhsk
6. Sizran	15. Sura	24. Pachelma	33. Mosca
7. Kuzovatovo	16. Chaadaevka	25. Sasovo	
8. Novospasskoe	17. Kadoshkino	26. Ryazan	
9. Barysh	18. Ruzhevka	27. Bashmakovo	

Figura 1. Tratte ferroviarie grafico



Pesi dei bordi del primo vertice

1-2: 1130	1-8: 1280	1-18: 1240	1-26: 2510
1-2: 1220	1-9: 1220	1-19: 1630	1-27: 1860
1-2: 920	1-10: 1210	1-20: 1370	1-28: 1820
1-3: 1130	1-11: 1170	1-20: 1590	1-29: 1910
1-4: 1090	1-12: 1000	1-21: 1420	1-30: 2020
1-4: 1140	1-12: 1220	1-22: 1440	1-31: 2140
1-4: 890	1-13: 1450	1-22: 2030	1-32: 2170
1-5: 900	1-14: 1170	1-23: 1690	1-33: 3250
1-6: 930	1-15: 1120	1-24: 1970	1-33: 2800
1-6: 1000	1-16: 1440	1-25: 1490	1-33: 2950
1-6: 620	1-17: 1340	1-26: 2120	
1-7: 930	1-18: 1650	1-26: 2030	

Pesi di bordi l'ultimo vertice

1-33: 3250	6-33: 2700	16-33: 2400	25-33: 1310
1-33: 2800	6-33: 2770	17-33: 1430	26-33: 890
1-33: 2950	7-33: 2700	18-33: 1810	26-33: 860
2-33: 3250	8-33: 2770	18-33: 1900	26-33: 1200
2-33: 2730	9-33: 2800	19-33: 2400	27-33: 1910
2-33: 2950	10-33: 2590	20-33: 1810	28-33: 1910
3-33: 3250	11-33: 2230	20-33: 1420	29-33: 1660
4-33: 3250	12-33: 2100	21-33: 1520	30-33: 1730
4-33: 2710	12-33: 1950	22-33: 1360	31-33: 1520
4-33: 2950	13-33: 2590	22-33: 1400	32-33: 1770
5-33: 2730	14-33: 2010	23-33: 2100	
6-33: 3250	15-33: 3100	24-33: 1840	

Figura 2. Cerca soluzione software

```

vertex 0> 0
vertex 1> 870
vertex 2> 1130
vertex 3> 870
vertex 4> 900
vertex 5> 620
vertex 6> 930
vertex 7> 620
vertex 8> 1220
vertex 9> 620
vertex 10> 1170
vertex 11> 1000
vertex 12> 1450
vertex 13> 1170
vertex 14> 1120
vertex 15> 1440
vertex 16> 1340
vertex 17> 1240
vertex 18> 1630
vertex 19> 1370
vertex 20> 1420
vertex 21> 1440
vertex 22> 1690
vertex 23> 1970
vertex 24> 1490
vertex 25> 2030
vertex 26> 1860
vertex 27> 1820
vertex 28> 1910
vertex 29> 2020
vertex 30> 2140
vertex 31> 2170
vertex 32> 2770
    
```

Figura 3. Il percorso risultante

