summary from last day — shortest paths in graphs

single source
- Dijkstra $O(m + n \log n)$
- Bellman-Ford $O(n \cdot m)$

all pairs
- Floyd-Warshall $O(n^3)$

today:

other algorithms for all-pairs shortest paths

possible papers to present

all pairs shortest path algorithms

Johnston's algorithm David Johnson 1977

idea: use Dijkstra from each node $O(n(m + n \log n)) = O(n \cdot m + n^2 \log n)$

issue: negative edge weights

Assign potential $p(v)$ $\forall v \in V$ — vertex set

Define $w_p(u,v) = w(u,v) + p(u) - p(v)$ (where $w(u,v)$ is the initial weight)

- why does this work?
- how to get $w_p \geq 0$

Claim $d_p(u,v) = d(u,v) + p(u) - p(v)$

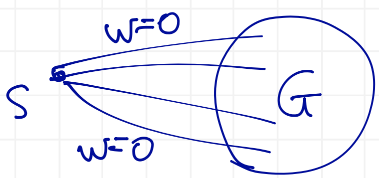
where $d_p(u,v)$ is the distance in w_p, and $d(u,v)$ is the true distance using w

Pf. true for every path $u = u_0, u_1 \cdots u_t = v$

$$\sum_{i=0}^{t-1} w_p(u_i, u_{i+1}) = \sum \left(w(u_i, u_{i+1}) + p(u_i) - p(u_{i+1}) \right)$$
$$= \left(\sum w(u_i, u_{i+1}) \right) + p(u_0) - p(u_t)$$

Johnston's algorithm

To find $p(v)$ s.t. $w_p(u,v) \geq 0 \quad \forall (u,v) \in E$



add new vertex s

define $w(s,v) = 0 \quad \forall v$

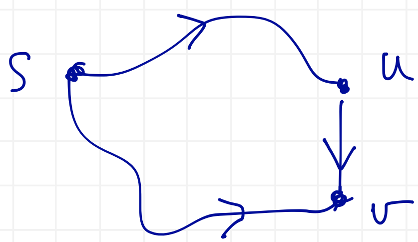
Use Bellman-Ford to find $d(s,v) \ \forall v$ $\quad O(m \cdot n)$

OR find neg. weight cycle and STOP.

Define $p(v) = d(s,v)$

Claim $w_p(u,v) \geq 0$

Pf.



look at edge (u,v)

$$\underbrace{d(s,u)}_{= p(u)} + w(u,v) \geq \underbrace{d(s,v)}_{= p(v)}$$

$$w_p(u,v) = w(u,v) + p(u) - p(v) \geq 0$$

Johnston's algorithm

wrap up

- run Bellman-Ford to find $p(v)$ — $O(n \cdot m)$
- define $w_p(u,v)$ — $O(m)$
- run Dijkstra $\forall$ sources — $O(n(m + n \log n))$
- set $d(u,v) = d_p(u,v) - p(u) + p(v)$ — $O(n^2)$

$O(n \cdot m + n^2 \log n)$

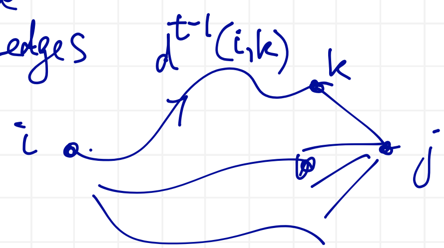
better than $O(n^3)$ for sparse graphs

all pairs shortest path algorithms

min-plus matrix multiplication — leads to good algorithms

$$d^t(i,j) = \min_{k \in V} \{ d^{t-1}(i,k) + w(k,j) \} \quad *$$

$d^t(i,j)$: shortest distance $i \to j$ using $\le t$ edges



Usual matrix mult. $A = B \times C$ $n \times n$ matrices

$$a(i,j) = \sum_k b(i,k) \cdot c(k,j)$$

in above * min · plus instead of $+, \times$

$$D^t = D^{t-1} \underline{\times} W \qquad D^1 = W$$

(D^t: $d^t(i,j)$)

want $D^{n-1} = W^{n-1}$ $\quad O(n \cdot n^3)$ ← time for one matrix "multiplication"

all pairs shortest path algorithms

min-plus matrix multiplication

Speed up — use repeated squaring

$W^1 \quad W^2 \quad W^4 \quad \ldots \quad W^{2^{\lceil \log(n-1) \rceil}}$

overshoot but ok since $W^{n-1} = W^n = W^{n+1} \ldots$ if no neg. weight cycle

$O(n^3 \log n)$

---

note: neg. weight cycle appears as neg. no. on main diagonal

Can we use ideas like Strassen fast matrix mult.?

no — they crucially use subtraction.

but see Aho Hopcroft Ullman for $O(n^3)$

special cases for shortest paths

weighted/unweighted
directed/undirected
model of computing

} allows faster algs.

Models of Computing

RAM Random Access Machine

unit cost RAM – charge 1 for arithmetic operations

for comp. geom. real RAM – even allow $\sqrt{}$

} too powerful

sensible restrictions

- strongly poly. time – treat numbers as black boxes
  - can only do (restricted) operations
- word RAM – each word has w bits
  assume $w \geq \log n$

?pointer models?

word RAM

Sorting and searching on the word RAM
T Hagerup - STACS 98, 1998 - Springer
Abstract A **word RAM** is a unit-cost random-access machine with a **word** length of w bits, for some w, and with an instruction repertoire similar to that found in present-day computers. The simple lower bounds for the problems of **sorting** and **searching** valid in the **...**
Cited by 128Related articlesAll 6 versionsCiteSave

From: http://scholar.google.ca/scholar?hl=en&q=Sorting+and+searching+on+the+word+RAM&btnG=&as_sdt=1%2C5&as_sdtp=

papers to present

more papers on single source version

Improved shortest paths on the word RAM
T Hagerup - Automata, Languages and Programming, 2000 - Springer
Abstract Thorup recently showed that single-source **shortest-paths** problems in undirected networks with n vertices, m edges, and edge weights drawn from 0,..., 2 w-1 can be solved in O (n+ m) time and space on a unit-cost random-access machine with a **word** length of w **...**
Cited by 70Related articlesAll 12 versionsCiteSave

From: http://scholar.google.ca/scholar?hl=en&q=Improved+Shortest+Paths+on+the+Word+RAM&btnG=&as_sdt=1%2C5&as_sdtp=

A simple shortest path algorithm with linear average time
AV Goldberg - **Algorithms**—ESA 2001, 2001 - Springer
Abstract We present a **simple shortest path algorithm**. If the input lengths are positive and uniformly distributed, the **algorithm** runs in **linear time**. The worst-case running **time** of the **algorithm** is O (m+ n log C), where n and m are the number of vertices and arcs of the **...**
Cited by 95Related articlesAll 21 versionsCiteSave

From: http://scholar.google.ca/scholar?q=A+Simple+Shortest+Path+Algorithm+with+Linear+Average+Time&btnG=&hl=en&as_sdt=0%2C5

[HTML] **A faster algorithm for the single source shortest path problem with few distinct positive lengths**
JB Orlin, K Madduri, K Subramani… - … of Discrete **Algorithms**, 2010 - Elsevier
In this paper, we propose an efficient method for implementing Dijkstra's **algorithm** for the **Single Source Shortest Path Problem** (SSSPP) in a graph whose edges have **positive length**, and where there are **few distinct** edge **lengths**. The SSSPP is one of the most **...**
Cited by 31Related articlesAll 14 versionsCiteSave

From: http://scholar.google.ca/scholar?hl=en&q=A+faster+algorithm+for+the+single+source+shortest+path+problem+with+few+distinct+positive+lengths&btnG=&as_sdt=1%2C5&as_sdtp=

all pairs shortest paths

* **Randomized Speedup of the Bellman-Ford Algorithm.**
MJ Bannister, D Eppstein - ANALCO, 2012 - SIAM
Abstract We describe a variant of the **Bellman–Ford algorithm** for single-source shortest paths in graphs with negative edges but no negative cycles that randomly permutes the vertices and uses this **randomized** order to process the vertices within each pass of the **...**
Cited by 5Related articlesAll 8 versionsCiteSave

From: http://scholar.google.ca/scholar?hl=en&q=Randomized+Speedup+of+the+Bellman%E2%80%93Ford+Algorithm&btnG=&as_sdt=1%2C5&as_sdtp=

Timothy will present

All-pairs shortest paths with real weights in O (n 3/log n) time ← do mini-review
TM Chan - Algorithmica, 2008 - Springer
Cited by 23Related articlesAll 7 versionsCiteSave

From: http://scholar.google.ca/scholar?q=All-Pairs+Shortest+Paths+with+Real+Weights+in+O%28n3%2F+log+n%29+Time&btnG=&hl=en&as_sdt=0%2C5

others

Finding the hidden path: Time bounds for all-pairs shortest paths ← $\Omega(m \cdot n)$ lower bound in some cases.
DR Karger, D Koller, SJ Phillips - SIAM Journal on Computing, 1993 - SIAM
Cited by 130Related articlesAll 17 versionsCiteSave

From: http://scholar.google.ca/scholar?q=Finding+the+hidden+path%3A+time+bounds+for+all-pairs+shortest+paths.&btnG=&hl=en&as_sdt=0%2C5

[PDF] **On the all-pairs-shortest-path problem in unweighted undirected graphs** ← nice!
R Seidel - Journal of computer and system sciences, 1995 - Elsevier
Cited by 168Related articlesAll 6 versionsCiteSave

From: http://scholar.google.ca/scholar?q=On+the+all-pairs-shortest-path+problem+in+unweighted+undirected+graphs&btnG=&hl=en&as_sdt=0%2C5

Zwick's survey — source of many other papers

Exact and approximate distances in graphs—a survey — not to present

U Zwick - Algorithms—ESA 2001, 2001 - Springer

Abstract W **survey** recent and not so recent results related to the computation of **exact** and **approximate distances**, and corresponding shortest, or almost shortest, paths in **graphs**. We consider many different settings and models and try to identify some remaining open **...**

Cited by 146 Related articles All 13 versions Cite Save

From: http://scholar.google.ca/scholar?hl=en&q=Exact+and+Approximate+Distances+in+Graphs+%E2%80%93+A+Survey&btnG=&as_sdt=1%2C5&as_sdtp=

All-pairs almost shortest paths — approximate

D Dor, S Halperin, U Zwick - SIAM Journal on Computing, 2000 - SIAM

Let G=(V, E) be an unweighted undirected graph on n vertices. A simple argument shows that computing **all** distances in G with an additive one-sided error of at most 1 is as hard as Boolean matrix multiplication. Building on recent work of Aingworth et al. SIAM J. Comput., **...**

Cited by 192 Related articles All 18 versions Cite Save

From: http://scholar.google.ca/scholar?hl=en&q=All+pairs+almost+shortest+paths.&btnG=&as_sdt=1%2C5&as_sdtp=

center and diameter

basic: compute all-pairs shortest paths

Fast approximation algorithms for the diameter and radius of sparse graphs
L Roditty, V Vassilevska Williams - Proceedings of the forty-fifth annual …, 2013 - dl.acm.org
Abstract The **diameter** and the **radius** of a **graph** are fundamental topological parameters that have many important practical applications in real world networks. The fastest combinatorial **algorithm** for both parameters works by solving the all-pairs shortest paths **...**
Cited by 15Related articlesAll 6 versionsCiteSave

From: http://scholar.google.ca/scholar?hl=en&q=Fast+Approximation+Algorithms+for+the+Diameter+and+Radius+of+Sparse+Graphs&btnG=&as_sdt=1%2C5&as_sdtp=

Diameter determination on restricted graph families — chordal graphs
DG Corneil, FF Dragan, M Habib, C Paul - Discrete Applied Mathematics, 2001 - Elsevier
Determining the **diameter** of a **graph** is a fundamental **graph** operation, yet no efficient (ie linear or quadratic time) algorithm is known. In this paper, we examine the **diameter** problem on chordal **graphs** and AT-free **graphs** and show that a very simple (linear time) 2-sweep **...**
Cited by 22Related articlesAll 12 versionsCiteSave

From: http://scholar.google.ca/scholar?hl=en&q=Diameter+determination+on+restricted+graph+families&btnG=&as_sdt=1%2C5&as_sdtp=

Center and diameter problems in plane triangulations and quadrangulations — mix of graphs, geometry
V Chepoi, F Dragan, Y Vaxès - … of the thirteenth annual ACM-SIAM …, 2002 - dl.acm.org
Abstract In this note, we present first linear time algorithms for computing the **center** and the **diameter** of several classes of face regular **plane** graphs: **triangulations** with inner vertices of degree≥ 6, **quadrangulations** with inner vertices of degree≥ 4 and the subgraphs of the **...**
Cited by 24Related articlesAll 9 versionsCiteSave

From: http://scholar.google.ca/scholar?hl=en&q=Center+and+diameter+problems+in+plane+triangulations+and+quadrangulations&btnG=&as_sdt=1%2C5&as_sdtp=

Scale-free characteristics of random networks: the topology of the world-wide web — diameter of www ~19
AL Barabási, R Albert, H Jeong - Physica A: Statistical Mechanics and its …, 2000 - Elsevier
The **world-wide web** forms a large directed graph, whose vertices are documents and edges are links pointing from one document to another. Here we demonstrate that despite its apparent **random** character, the **topology** of this graph has a number of universal **scale**- **...**
Cited by 1192Related articlesAll 28 versionsCiteSave

From: http://scholar.google.ca/scholar?q=Scale-free+characteristics+of+random+networks%3A+the+topology+of+the+world-wide+web&btnG=&hl=en&as_sdt=0%2C5

shortest paths in planar graphs

*

[HTML] **Faster shortest-path algorithms for planar graphs**
MR Henzinger, P Klein, S Rao… - journal of computer and …, 1997 - Elsevier
We give a linear-time **algorithm** for single-source shortest paths in **planar graphs** with nonnegative edge-lengths. Our **algorithm** also yields a linear-time **algorithm** for maximum flow in a **planar graph** with the source and sink on the same face. For the case where **...**
Cited by 295Related articlesAll 15 versionsCiteSave

From: http://scholar.google.ca/scholar?q=Faster+Shortest-Path+Algorithms+for+Planar+Graphs&btnG=&hl=en&as_sdt=0%2C5

[HTML] **Planar graphs, negative weight edges, shortest paths, and near linear time**
J Fakcharoenphol, S Rao - Journal of Computer and System Sciences, 2006 - Elsevier
In this paper, we present an O (nlog3n) **time** algorithm for finding **shortest paths** in an n-node **planar graph** with real **weights**. This can be compared to the best previous strongly polynomial **time** algorithm developed by Lipton, Rose, and Tarjan in 1978 which runs in O **...**
Cited by 86Related articlesAll 10 versionsCiteSave

From: http://scholar.google.ca/scholar?hl=en&q=Planar+graphs%2C+negative+weight+edges%2C+shortest+paths%2C+and+near+linear+time&btnG=&as_sdt=1%2C5&as_sdtp=

Compact oracles for reachability and approximate distances in planar digraphs
M Thorup - Journal of the ACM (JACM), 2004 - dl.acm.org
Abstract It is shown that a **planar digraph** can be preprocessed in near-linear time, producing a near-linear space **oracle** that can answer **reachability** queries in constant time. The **oracle** can be distributed as an O (log n) space label for each vertex and then we can **...**
Cited by 191Related articlesAll 7 versionsCiteSave

From: http://scholar.google.ca/scholar?q=Compact+oracles+for+reachability+and+approximate+distances+in+planar+digraphs&btnG=&hl=en&as_sdt=0%2C5

hierarchical approaches for road networks

Exact routing in large road networks using contraction hierarchies
R Geisberger, P Sanders, D Schultes… - Transportation …, 2012 - pubsonline.informs.org
Contraction hierarchies are a simple approach for fast **routing** in **road networks**. Our algorithm calculates **exact** shortest paths and handles **road networks** of whole continents. During a preprocessing step, we exploit the inherent hierarchical structure of **road ...**
Cited by 50Related articlesAll 6 versionsCiteSave

From: http://scholar.google.ca/scholar?q=Exact+routing+in+large+road+networks&btnG=&hl=en&as_sdt=0%2C5

Combining hierarchical and goal-directed speed-up techniques for dijkstra's algorithm
R Bauer, D Delling, P Sanders… - Journal of Experimental …, 2010 - dl.acm.org
Abstract In recent years, highly effective **hierarchical** and **goal-directed** speed-up techniques for routing in large road networks have been developed. This article makes a systematic study of combinations of such techniques. These combinations turn out to give the best **...**
Cited by 85Related articlesAll 6 versionsCiteSave

From: http://scholar.google.ca/scholar?q=Combining+hierarchical+and+goal-directed&btnG=&hl=en&as_sdt=0%2C5

Hierarchical hub labelings for shortest paths
I Abraham, D Delling, AV Goldberg, RF Werneck - Algorithms–ESA 2012, 2012 - Springer
Abstract We study **hierarchical hub labelings** for computing **shortest paths**. Our new theoretical insights into the structure of **hierarchical labels** lead to faster preprocessing algorithms, making the **labeling** approach practical for a wider class of graphs. We also **...**
Cited by 40Related articlesAll 12 versionsCiteSave

From: http://scholar.google.ca/scholar?q=Hierarchical+Hub+Labelings+for+Shortest+Paths&btnG=&hl=en&as_sdt=2005&sciodt=0%2C5&cites=12496478965946049336&scipsc=

k-th shortest paths

Finding the k shortest paths
D Eppstein - SIAM Journal on computing, 1998 - SIAM
We give algorithms for **finding** the **k shortest paths** (not required to be simple) connecting a pair of vertices in a digraph. Our algorithms output an implicit representation of these **paths** in a digraph with n vertices and m edges, in time O (m+ n log n+ **k**). We can also **find** the **k ...**
Cited by 1301Related articlesAll 20 versionsCiteSave

From: http://scholar.google.ca/scholar?q=FINDING+THE+k+SHORTEST+PATHS&btnG=&hl=en&as_sdt=0%2C5

* — Nathan

Finding the k shortest simple paths: A new algorithm and its implementation
J Hershberger, M Maxel, S Suri - ACM Transactions on **Algorithms** (…, 2007 - dl.acm.org
Abstract We describe a **new algorithm** to enumerate the **k shortest simple** (loopless) **paths** in a directed graph and report on **its implementation**. Our **algorithm** is based on a replacement **paths algorithm** proposed by Hershberger and Suri [2001], and can yield a factor Θ (n) **...**
Cited by 109Related articlesAll 9 versionsCiteSave

From: http://scholar.google.ca/scholar?q=Finding+the+k+shortest+simple+paths%3A+A+new+algorithm+and+its+implementation&btnG=&hl=en&as_sdt=0%2C5

dynamic shortest path algorithms — maintain shortest paths as the graph is updated

Fully dynamic algorithms for maintaining all-pairs shortest paths and transitive closure in digraphs
V King - Foundations of Computer Science, 1999. 40th Annual …, 1999 - ieeexplore.ieee.org
Abstract This paper presents the first **fully dynamic algorithms** for **maintaining all-pairs shortest paths** in **digraphs** with positive integer weights less than b. For approximate **shortest paths** with an error factor of (2+), for any positive constant, the amortized update time isO (**...**
Cited by 151Related articlesAll 11 versionsCiteSave

From: http://scholar.google.ca/scholar?hl=en&q=Fully+Dynamic+Algorithms+for+Maintaining+All-Pairs+Shortest+Paths+and+Transitive+Closure+in+Digraphs&btnG=&as_sdt=1%2C5&as_sdtp=

A new approach to dynamic all pairs shortest paths
C Demetrescu, GF Italiano - Journal of the ACM (JACM), 2004 - dl.acm.org
Abstract We study novel combinatorial properties of graphs that allow us to devise a completely **new approach** to **dynamic all pairs shortest paths** problems. Our **approach** yields a fully **dynamic** algorithm for general directed graphs with non-negative real-valued edge **...**
Cited by 200Related articlesAll 14 versionsCiteSave

From: http://scholar.google.ca/scholar?q=A+new+approach+to+dynamic+all+pairs+shortest+paths&btnG=&hl=en&as_sdt=0%2C5

Experimental analysis of dynamic all pairs shortest path algorithms
C Demetrescu, GF Italiano - ACM Transactions on **Algorithms** (TALG), 2006 - dl.acm.org
Abstract We present the results of an extensive computational study on **dynamic algorithms** for **all pairs shortest path** problems. We describe our implementations of the recent **dynamic algorithms** of King [1999] and of Demetrescu and Italiano [2006], and compare them to the **...**
Cited by 86Related articlesAll 18 versionsCiteSave

From: http://scholar.google.ca/scholar?q=Experimental+analysis+of+dynamic+all+pairs+shortest+path+algorithms&btnG=&hl=en&as_sdt=0%2C5

dynamic shortest path algorithms — maintain shortest paths as the graph is updated

Towards polynomial lower bounds for dynamic problems
M Patrascu - Proceedings of the 42nd ACM symposium on Theory of …, 2010 - dl.acm.org
Abstract We consider a number of **dynamic problems** with no known poly-logarithmic upper **bounds**, and show that they require n Ω (1) time per operation, unless 3SUM has strongly subquadratic algorithms. Our result is modular:(1) We describe a carefully-chosen **...**
Cited by 40Related articlesAll 9 versionsCiteSave

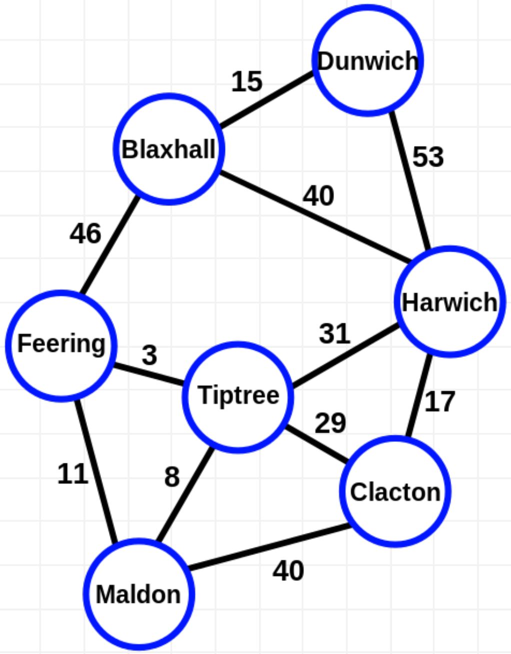
From: http://scholar.google.ca/scholar?q=Towards+polynomial+lower+bounds+for+dynamic+problems&btnG=&hl=en&as_sdt=2005&sciodt=0%2C5&cites=11369852506014706065&scipsc=

Threesomes, Degenerates, and Love Triangles
A Grønlund, S Pettie - arXiv preprint arXiv:1404.0799, 2014 - arxiv.org
Abstract: The 3SUM problem is to decide, given a set of $ n $ real numbers, whether any three sum to zero. We prove that the decision tree complexity of 3SUM is $ O (n^{3/2}\ sqrt {\ log n}) $, that there is a randomized 3SUM algorithm running in $ O (n^ 2 (\ log\ log n)^ 2/\ **...**
Cited by 2Related articlesAll 2 versionsCiteSave

— subquadratic alg. for 3SUM

From: http://scholar.google.ca/scholar?hl=en&q=Threesomes%2C+Degenerates%2C+and+Love+Triangles&btnG=&as_sdt=1%2C5&as_sdtp=

widest path problem (bottleneck shortest path problem)



find out best alg. for this.

In this graph, the widest path from Maldon to Feering has bandwidth 29, and passes through Clacton, Tiptree, Harwich, and Blaxhall.

wikipedia

bicriteria and constraints

Resource constrained shortest paths
K Mehlhorn, M Ziegelmann - Algorithms-ESA 2000, 2000 - Springer
Abstract The **resource constrained shortest path** problem (CSP) asks for the computation of a least cost **path** obeying a set of **resource constraints**. The problem is NP-complete. We give theoretical and experimental results for CSP. In the theoretical part we present the **...**
Cited by 104 Related articles All 24 versions Cite Save

From: http://scholar.google.ca/scholar?hl=en&q=Resource+constrained+shortest+paths.&btnG=&as_sdt=1%2C5&as_sdtp=