# Dijkstra's Algorithm

Timothy G. Griffin

timothy.griffin@cl.cam.ac.uk
Computer Laboratory
University of Cambridge, UK

Michaelmas Term, 2020

# Classical Dijkstra

Given adjacency matrix $\mathbf{A}$ over a selective semiring and source vertex $i \in V$, Dijkstra's algorithm will compute $\mathbf{A}^*(i, \_)$ such that

$$\mathbf{A}^*(i, j) = \bigoplus_{p \in P(i,j)} w_{\mathbf{A}}(p).$$

That is, we compule the $i - th$ row of $\mathbf{A}^*$.

# Non-Classical Dijkstra

If we drop assumptions of distributivity, then given adjacency matrix $\mathbf{A}$ and source vertex $i \in V$, Dijkstra's algorithm will compute $\mathbf{R}(i, \_)$ such that

$$\forall j \in V : \mathbf{R}(i, j) = \mathbf{I}(i, j) \oplus \bigoplus_{q \in V} \mathbf{R}(i, q) \otimes \mathbf{A}(q, j).$$

That is, we compute the $i$-the row of an $\mathbf{R}$ that solves the matrix equation

$$\mathbf{R} = (\mathbf{R} \otimes \mathbf{A}) \oplus \mathbf{I}.$$

## Read this paper

**Routing in Equilibrium**, João Luís Sobrinho and Timothy G. Griffin, MTNS 2010.

# Dijkstra's algorithm

**Input** : adjacency matrix $\mathbf{A}$ and source vertex $i \in V$,
**Output** : the $i$-th row of $\mathbf{R}$, $\mathbf{R}(i, \_)$.

**begin**
  $S \leftarrow \{i\}$
  $\mathbf{R}(i, i) \leftarrow \overline{1}$
  **for each** $q \in V - \{i\}$ : $\mathbf{R}(i, q) \leftarrow \mathbf{A}(i, q)$
  **while** $S \neq V$
    **begin**
      find $q \in V - S$ such that $\mathbf{R}(i, q)$ is $\leqslant^L_\oplus$ -minimal
      $S \leftarrow S \cup \{q\}$
      **for each** $j \in V - S$
        $\mathbf{R}(i, j) \leftarrow \mathbf{R}(i, j) \oplus (\mathbf{R}(i, q) \otimes \mathbf{A}(q, j))$
    **end**
**end**

# Classical proofs of Dijkstra's algorithm (for global optimality) assume

## Semiring Axioms

$$
\begin{array}{rcrcl}
\mathbb{AS}(\oplus) & : & a \oplus (b \oplus c) & = & (a \oplus b) \oplus c \\
\mathbb{CM}(\oplus) & : & a \oplus b & = & b \oplus a \\
\mathbb{ID}(\oplus) & : & \overline{0} \oplus a & = & a \\
\mathbb{AS}(\otimes) & : & a \otimes (b \otimes c) & = & (a \otimes b) \otimes c \\
\mathbb{IDL}(\otimes) & : & \overline{1} \otimes a & = & a \\
\mathbb{IDR}(\otimes) & : & a \otimes \overline{1} & = & a \\
\mathbb{ANL}(\otimes) & : & \overline{0} \otimes a & = & \overline{0} \\
\mathbb{ANR}(\otimes) & : & a \otimes \overline{0} & = & \overline{0} \\
\mathbb{LD} & : & a \otimes (b \oplus c) & = & (a \otimes b) \oplus (a \otimes c) \\
\mathbb{RD} & : & (a \oplus b) \otimes c & = & (a \otimes c) \oplus (b \otimes c)
\end{array}
$$

# Classical proofs of Dijkstra's algorithm assume

## Additional axioms

$$\begin{array}{lcrcl} \mathbb{SL}(\oplus) & : & a \oplus b & \in & \{a,\ b\} \\ \mathbb{AN}(\oplus) & : & \overline{1} \oplus a & = & \overline{1} \end{array}$$

Note that we can derive right absorption,

$$\mathbb{RA} \quad : \quad a \oplus (a \otimes b) \quad = \quad a$$

and this gives (right) inflationarity, $\forall a, b : a \leqslant a \otimes b$.

$$\begin{array}{lcl} a \oplus (a \otimes b) & = & (a \otimes \overline{1}) \oplus (a \otimes b) \\ & = & a \otimes (\overline{1} \oplus b) \\ & = & a \otimes \overline{1} \\ & = & a \end{array}$$

# What will we assume? Very little!

## Semiring Axioms

$$
\begin{array}{rcrcl}
\mathbb{AS}(\oplus) & : & a \oplus (b \oplus c) & = & (a \oplus b) \oplus c \\
\mathbb{CM}(\oplus) & : & a \oplus b & = & b \oplus a \\
\mathbb{ID}(\oplus) & : & \overline{0} \oplus a & = & a \\
\mathbb{AS}(\otimes) & : & a \otimes (b \otimes c) & = & (a \otimes b) \otimes c \\
\mathbb{IDL}(\otimes) & : & \overline{1} \otimes a & = & a \\
\mathbb{IDR}(\otimes) & : & a \otimes \overline{1} & = & a \\
\mathbb{ANL}(\otimes) & : & \overline{0} \otimes a & = & \overline{0} \\
\mathbb{ANR}(\otimes) & : & a \otimes \overline{0} & = & \overline{0} \\
\mathbb{LD} & : & a \otimes (b \oplus c) & = & (a \otimes b) \oplus (a \otimes c) \\
\mathbb{RD} & : & (a \oplus b) \otimes c & = & (a \otimes c) \oplus (b \otimes c)
\end{array}
$$

# What will we assume?

## Additional axioms

$$\begin{array}{rcrcl} \mathbb{SL}(\oplus) & : & a \oplus b & \in & \{a,\ b\} \\ \mathbb{ANL}(\oplus) & : & \overline{1} \oplus a & = & \overline{1} \\ \mathbb{RA} & : & a \oplus (a \otimes b) & = & a \end{array}$$

- Note that we can no longer derive $\mathbb{RA}$, so we must assume it.
- Again, $\mathbb{RA}$ says that $a \leqslant a \otimes b$.
- We don't use $\mathbb{SL}$ explicitly in the proofs, but it is implicit in the algorithm's definition of $q_k$.
- We do not use $\mathbb{AS}(\oplus)$ and $\mathbb{CM}(\oplus)$ explicitly, but these assumptions are implicit in the use of the "big-$\oplus$" notation.

# Under these weaker assumptions ...

**Theorem (Sobrinho/Griffin)**

Given adjacency matrix $\mathbf{A}$ and source vertex $i \in V$, Dijkstra's algorithm will compute $\mathbf{R}(i, \_)$ such that

$$\forall j \in V : \mathbf{R}(i, j) = \mathbf{I}(i, j) \oplus \bigoplus_{q \in V} \mathbf{R}(i, q) \otimes \mathbf{A}(q, j).$$

That is, it computes one row of the solution for the right equation

$$\mathbf{R} = \mathbf{RA} \oplus \mathbf{I}.$$

# Dijkstra's algorithm, annotated version

Subscripts make proofs by induction easier ....

**begin**
  $S_1 \leftarrow \{i\}$
  $\mathbf{R}_1(i,\ i) \leftarrow \overline{1}$
  **for each** $q \in V - S_1$ : $\mathbf{R}_1(i,\ q) \leftarrow \mathbf{A}(i,\ q)$
  **for each** $k = 2,\ 3,\ \ldots,\ \mid V \mid$
    **begin**
      find $q_k \in V - S_{k-1}$ such that $\mathbf{R}_{k-1}(i,\ q_k)$ is $\leqslant_{\oplus}^{L}$ -minimal
      $S_k \leftarrow S_{k-1} \cup \{q_k\}$
      **for each** $j \in V - S_k$
        $\mathbf{R}_k(i,\ j) \leftarrow \mathbf{R}_{k-1}(i,\ j) \oplus (\mathbf{R}_{k-1}(i,\ q_k) \otimes \mathbf{A}(q_k,\ j))$
    **end**
**end**

## Main Claim, annotated

$$\forall k : 1 \leqslant k \leqslant \mid V \mid \Longrightarrow \forall j \in S_k : \mathbf{R}_k(i,\ j) = \mathbf{I}(i, j) \oplus \bigoplus_{q \in S_k} \mathbf{R}_k(i,\ q) \otimes \mathbf{A}(q,\ j)$$

## We will use

Observation 1 (no backtracking) :

$$\forall k : 1 \leqslant k <\mid V \mid \Longrightarrow \forall j \in S_{k+1} : \mathbf{R}_{k+1}(i,\ j) = \mathbf{R}_k(i,\ j)$$

Observation 2 (Dijkstra is “greedy”):

$$\forall k : 1 \leqslant k \leqslant \mid V \mid \Longrightarrow \forall q \in S_k : \forall w \in V - S_k : \mathbf{R}_k(i,\ q) \leqslant \mathbf{R}_k(i,\ w)$$

Observation 3 (Accurate estimates):

$$\forall k : 1 \leqslant k \leqslant \mid V \mid \Longrightarrow \forall w \in V - S_k : \mathbf{R}_k(i,\ w) = \bigoplus_{q \in S_k} \mathbf{R}_k(i,\ q) \otimes \mathbf{A}(q,\ w)$$

## Observation 1

$$\forall k : 1 \leqslant k <\mid V \mid \implies \forall j \in S_{k+1} : \mathbf{R}_{k+1}(i,\ j) = \mathbf{R}_k(i,\ j)$$

Proof: This is easy to see by inspection of the algorithm. Once a node is put into $S$ its weight never changes again.

# The algorithm is "greedy"

## Observation 2

$$\forall k : 1 \leqslant k \leqslant | V | \Longrightarrow \forall q \in S_k : \forall w \in V - S_k : \mathbf{R}_k(i,\ q) \leqslant \mathbf{R}_k(i,\ w)$$

By induction.
Base : Since $S_1 = \{i\}$ and $\mathbf{R}_1(i,\ i) = \overline{1}$, we need to show that

$$\overline{1} \leqslant \mathbf{A}(i,\ w) \equiv \overline{1} = \overline{1} \oplus \mathbf{A}(i,\ w).$$

This follows from $\mathbb{ANL}(\oplus)$.
Induction: Assume $\forall q \in S_k : \forall w \in V - S_k : \mathbf{R}_k(i,\ q) \leqslant \mathbf{R}_k(i,\ w)$ and show $\forall q \in S_{k+1} : \forall w \in V - S_{k+1} : \mathbf{R}_{k+1}(i,\ q) \leqslant \mathbf{R}_{k+1}(i,\ w)$.
Since $S_{k+1} = S_k \cup \{q_{k+1}\}$, this means showing

$$\begin{array}{ll} (1) & \forall q \in S_k : \forall w \in V - S_{k+1} : \mathbf{R}_{k+1}(i,\ q) \leqslant \mathbf{R}_{k+1}(i,\ w) \\ (2) & \forall w \in V - S_{k+1} : \mathbf{R}_{k+1}(i,\ q_{k+1}) \leqslant \mathbf{R}_{k+1}(i,\ w) \end{array}$$

By Observation 1, showing $(1)$ is the same as

$$\forall q \in S_k : \forall w \in V - S_{k+1} : \mathbf{R}_k(i,\ q) \leqslant \mathbf{R}_{k+1}(i,\ w)$$

which expands to (by definition of $\mathbf{R}_{k+1}(i,\ w)$)

$$\forall q \in S_k : \forall w \in V - S_{k+1} : \mathbf{R}_k(i,\ q) \leqslant \mathbf{R}_k(i,\ w) \oplus (\mathbf{R}_k(i,\ q_{k+1}) \otimes \mathbf{A}(q_{k+1},\ w))$$

But $\mathbf{R}_k(i,\ q) \leqslant \mathbf{R}_k(i,\ w)$ by the induction hypothesis, and $\mathbf{R}_k(i,\ q) \leqslant (\mathbf{R}_k(i,\ q_{k+1}) \otimes \mathbf{A}(q_{k+1},\ w))$ by the induction hypothesis and $\mathbb{RA}$.
Since $a \leqslant^L_\oplus b \wedge a \leqslant^L_\oplus c \implies a \leqslant^L_\oplus (b \oplus c)$, we are done.

By Observation 1, showing (2) is the same as showing

$$\forall w \in V - S_{k+1} : \mathbf{R}_k(i,\ q_{k+1}) \leqslant \mathbf{R}_{k+1}(i,\ w)$$

which expands to

$$\forall w \in V - S_{k+1} : \mathbf{R}_k(i,\ q_{k+1}) \leqslant \mathbf{R}_k(i,\ w) \oplus (\mathbf{R}_k(i,\ q_{k+1}) \otimes \mathbf{A}(q_{k+1},\ w))$$

But $\mathbf{R}_k(i,\ q_{k+1}) \leqslant \mathbf{R}_k(i,\ w)$ since $q_{k+1}$ was chosen to be minimal, and $\mathbf{R}_k(i,\ q_{k+1}) \leqslant (\mathbf{R}_k(i,\ q_{k+1}) \otimes \mathbf{A}(q_{k+1},\ w))$ by $\mathbb{RA}$.
Since $a \leqslant^L_\oplus b \wedge a \leqslant^L_\oplus c \implies a \leqslant^L_\oplus (b \oplus c)$, we are done.

# Observation 3

## Observation 3

$$\forall k : 1 \leqslant k \leqslant \mid V \mid \Longrightarrow \forall w \in V - S_k : \mathbf{R}_k(i,\ w) = \bigoplus_{q \in S_k} \mathbf{R}_k(i,\ q) \otimes \mathbf{A}(q,\ w)$$

Proof: By induction:

Base : easy, since

$$\bigoplus_{q \in S_1} \mathbf{R}_1(i,\ q) \otimes \mathbf{A}(q,\ w) = \overline{1} \otimes \mathbf{A}(i,\ w) = \mathbf{A}(i,\ w) = \mathbf{R}_1(i,\ w)$$

Induction step. Assume

$$\forall w \in V - S_k : \mathbf{R}_k(i,\ w) = \bigoplus_{q \in S_k} \mathbf{R}_k(i,\ q) \otimes \mathbf{A}(q,\ w)$$

and show

$$\forall w \in V - S_{k+1} : \mathbf{R}_{k+1}(i,\ w) = \bigoplus_{q \in S_{k+1}} \mathbf{R}_{k+1}(i,\ q) \otimes \mathbf{A}(q,\ w)$$

By Observation 1, and a bit of rewriting, this means we must show

$$\forall w \in V - S_{k+1} : \mathbf{R}_{k+1}(i,\ w) = \mathbf{R}_k(i,\ q_{k+1}) \otimes \mathbf{A}(q_{k+1},\ w) \oplus \bigoplus_{q \in S_k} \mathbf{R}_k(i,\ q) \otimes \mathbf{A}($$

Using the induction hypothesis, this becomes

$$\forall w \in V - S_{k+1} : \mathbf{R}_{k+1}(i,\ w) = \mathbf{R}_k(i,\ q_{k+1}) \otimes \mathbf{A}(q_{k+1},\ w) \oplus \mathbf{R}_k(i,\ w)$$

But this is exactly how $\mathbf{R}_{k+1}(i,\ w)$ is computed in the algorithm.

# Proof of Main Claim

## Main Claim

$$\forall k : 1 \leqslant k \leqslant \mid V \mid \Longrightarrow \forall j \in S_k : \mathbf{R}_k(i,\ j) = \mathbf{I}(i,j) \oplus \bigoplus_{q \in S_k} \mathbf{R}_k(i,\ q) \otimes \mathbf{A}(q,\ j)$$

Proof : By induction on $k$.
Base case: $S_1 = \{i\}$ and the claim is easy.
Induction: Assume that

$$\forall j \in S_k : \mathbf{R}_k(i,\ j) = \mathbf{I}(i,j) \oplus \bigoplus_{q \in S_k} \mathbf{R}_k(i,\ q) \otimes \mathbf{A}(q,\ j)$$

We must show that

$$\forall j \in S_{k+1} : \mathbf{R}_{k+1}(i,\ j) = \mathbf{I}(i,j) \oplus \bigoplus_{q \in S_{k+1}} \mathbf{R}_{k+1}(i,\ q) \otimes \mathbf{A}(q,\ j)$$

Since $S_{k+1} = S_k \cup \{q_{k+1}\}$, this means we must show

$$
\begin{array}{ll}
(1) & \forall j \in S_k : \mathbf{R}_{k+1}(i,\ j) = \mathbf{I}(i,j) \oplus \bigoplus_{q \in S_{k+1}} \mathbf{R}_{k+1}(i,\ q) \otimes \mathbf{A}(q,\ j) \\
(2) & \mathbf{R}_{k+1}(i,\ q_{k+1}) = \mathbf{I}(i,q_{k+1}) \oplus \bigoplus_{q \in S_{k+1}} \mathbf{R}_{k+1}(i,\ q) \otimes \mathbf{A}(q,\ q_{k+1})
\end{array}
$$

By use Observation 1, showing (1) is the same as showing

$$\forall j \in S_k : \mathbf{R}_k(i,\ j) = \mathbf{I}(i,j) \oplus \bigoplus_{q \in S_{k+1}} \mathbf{R}_k(i,\ q) \otimes \mathbf{A}(q,\ j),$$

which is equivalent to

$$\forall j \in S_k : \mathbf{R}_k(i,\ j) = \mathbf{I}(i,j) \oplus (\mathbf{R}_k(i,\ q_{k+1}) \otimes \mathbf{A}(q_{k+1},\ j)) \oplus \bigoplus_{q \in S_k} \mathbf{R}_k(i,\ q) \otimes \mathbf{A}(q,\ j$$

By the induction hypothesis, this is equivalent to

$$\forall j \in S_k : \mathbf{R}_k(i,\ j) = \mathbf{R}_k(i,\ j) \oplus (\mathbf{R}_k(i,\ q_{k+1}) \otimes \mathbf{A}(q_{k+1},\ j)),$$

Put another way,

$$\forall j \in S_k : \mathbf{R}_k(i,\ j) \leqslant \mathbf{R}_k(i,\ q_{k+1}) \otimes \mathbf{A}(q_{k+1},\ j)$$

By observation 2 we know $\mathbf{R}_k(i,\ j) \leqslant \mathbf{R}_k(i,\ q_{k+1})$, and so

$$\mathbf{R}_k(i,\ j) \leqslant \mathbf{R}_k(i,\ q_{k+1}) \leqslant \mathbf{R}_k(i,\ q_{k+1}) \otimes \mathbf{A}(q_{k+1},\ j)$$

by $\mathbb{R}\mathbb{A}$.

To show (2), we use Observation 1 and $\mathbf{I}(i, q_{k+1}) = \overline{0}$ to obtain

$$\mathbf{R}_k(i,\ q_{k+1}) = \bigoplus_{q \in S_{k+1}} \mathbf{R}_k(i,\ q) \otimes \mathbf{A}(q,\ q_{k+1})$$

which, since $\mathbf{A}(q_{k+1},\ q_{k+1}) = \overline{0}$, is the same as

$$\mathbf{R}_k(i,\ q_{k+1}) = \bigoplus_{q \in S_k} \mathbf{R}_k(i,\ q) \otimes \mathbf{A}(q,\ q_{k+1})$$

This then follows directly from Observation 3.

# Finding Left Local Solutions?

$$\mathbf{L} = (\mathbf{A} \otimes \mathbf{L}) \oplus \mathbf{I} \quad \Longleftrightarrow \quad \mathbf{L}^T = (\mathbf{L}^T \otimes^T \mathbf{A}^T) \oplus \mathbf{I}$$

$$\mathbf{R}^T = (\mathbf{A}^T \otimes^T \mathbf{R}^T) \oplus \mathbf{I} \quad \Longleftrightarrow \quad \mathbf{R} = (\mathbf{R} \otimes \mathbf{A}) \oplus \mathbf{I}$$

where

$$a \otimes^T b = b \otimes a$$

Replace $\mathbb{RA}$ with $\mathbb{LA}$,

$$\mathbb{LA} : \forall a, b : a \leqslant b \otimes a$$