An Algebraic Approach to Internet Routing — Lectures 13, 14
Routing in Equilibrium
(presented at MTNS 2010, Budapest)

João Luís Sobrinho   Timothy G. Griffin

Instituto de Telecomunicações
Instituto Superior Técnico, Lisbon, Portugal
joao.sobrinho@lx.it.pt

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

Michaelmas Term
2010

What algebraic properties are associated with global optimality?

Distributivity

$$\begin{aligned} \text{L.D} &: & a \otimes (b \oplus c) &= (a \otimes b) \oplus (a \otimes c), \\ \text{R.D} &: & (a \oplus b) \otimes c &= (a \otimes c) \oplus (b \otimes c). \end{aligned}$$

What is this in $\mathrm{sp} = (\mathbb{N}^{\infty},\ \min,\ +)$?

$$\begin{aligned} \text{L.DIST} &: & a + (b \ \min\ c) &= (a + b) \ \min\ (a + c), \\ \text{R.DIST} &: & (a \ \min\ b) + c &= (a + c) \ \min\ (b + c). \end{aligned}$$

Left Local Optimality

Say that $\mathbf{L}$ is a left-locally optimal solution when

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

That is, for $i \neq j$ we have

$$\mathbf{L}(i,\ j) = \bigoplus_{q \in V} \mathbf{A}(i,\ q) \otimes \mathbf{L}(q,\ j)$$

- $\mathbf{L}(i,\ j)$ is the best possible value given the values $\mathbf{L}(q,\ j)$, for all out-neighbors q of source i.
- Rows $\mathbf{L}(i,\ \_)$ represents **out-trees** **from** i (think Bellman-Ford).
- Columns $\mathbf{L}(\_,\ i)$ represents **in-trees** **to** i.

Right Local Optimality

Say that $\mathbf{R}$ is a right-locally optimal solution when

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

That is, for $i \neq j$ we have

$$\mathbf{R}(i,\ j) = \bigoplus_{q \in V} \mathbf{R}(i,\ q) \otimes \mathbf{A}(q,\ j)$$

- $\mathbf{R}(i,\ j)$ is the best possible value given the values $\mathbf{R}(q,\ j)$, for all in-neighbors q of destination j.
- Rows $\mathbf{L}(i,\ \_)$ represents **out-trees from** i (think Dijkstra).
- Columns $\mathbf{L}(\_,\ i)$ represents **in-trees to** i.

With and Without Distributivity

With

For (well behaved) Semirings, the three optimality problems are essentially the same — locally optimal solutions are globally optimal solutions.

$$\mathbf{A}^* = \mathbf{L} = \mathbf{R}$$

Without

Suppose that we drop distributivity and $\mathbf{A}^*$, $\mathbf{L}$, $\mathbf{R}$ exist. It may be the case they they are all distinct.

A World Without Distributivity

Global Optimality

This has been studied, for example [**?**, **?**] in the context of circuit layout. See Chapter 5 of [**?**]. This approach does not play well with (loop-free) hop-by-hop forwarding (need tunnels!)

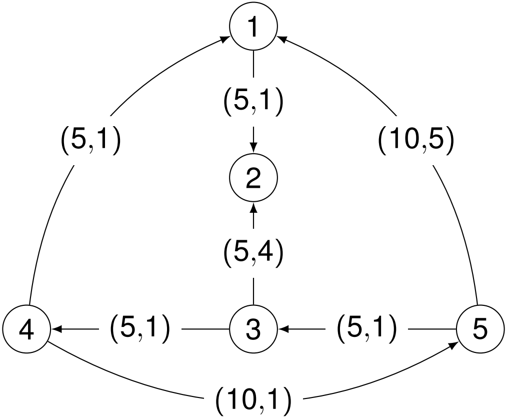
Left Local Optimality

At a very high level, this is the type of problem that BGP attempts to solve!!

Right Local Optimality

This approach does not play well with (loop-free) hop-by-hop forwarding (need tunnels!)

Example



(bandwidth, distance) with lexicographic order (bandwidth first).

Example's adjacency matrix

$$
\mathbf{A} = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{array}
\begin{array}{c}
\begin{array}{ccccc} \quad 1 \quad & \quad 2 \quad & \quad 3 \quad & \quad 4 \quad & \quad 5 \quad \end{array} \\
\left[\begin{array}{ccccc}
(0, \infty) & (5, 1) & (0, \infty) & (0, \infty) & (0, \infty) \\
(0, \infty) & (0, \infty) & (0, \infty) & (0, \infty) & (0, \infty) \\
(0, \infty) & (5, 4) & (0, \infty) & (5, 1) & (0, \infty) \\
(5, 1) & (0, \infty) & (0, \infty) & (0, \infty) & (10, 1) \\
(10, 5) & (0, \infty) & (5, 1) & (0, \infty) & (0, \infty)
\end{array} \right]
\end{array}
$$

Global optima

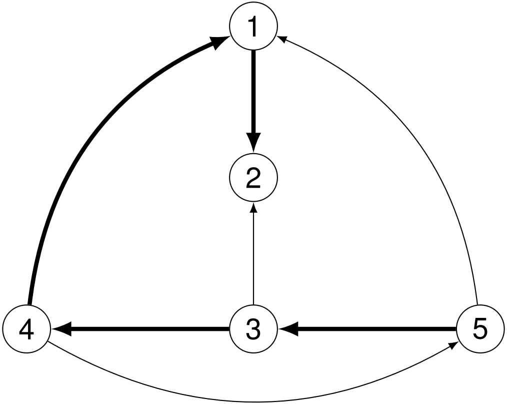
$$\mathbf{A}^* = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{array} \begin{array}{c} \begin{array}{ccccc} 1 & 2 & 3 & 4 & 5 \end{array} \\ \left[\begin{array}{ccccc} (\infty, 0) & (5, 1) & (0, \infty) & (0, \infty) & (0, \infty) \\ (0, \infty) & (\infty, 0) & (0, \infty) & (0, \infty) & (0, \infty) \\ (5, 2) & (5, 3) & (\infty, \ 0) & (5, 1) & (5, 2) \\ (10, 6) & (5, 2) & (5, 2) & (\infty, 0) & (10, 1) \\ (10, 5) & (5, 4) & (5, 1) & (5, 2) & (\infty, 0) \end{array} \right] \end{array},$$

Left local optima

$$\mathbf{L} = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{array} \begin{array}{c} \begin{array}{ccccc} 1 & 2 & 3 & 4 & 5 \end{array} \\ \left[\begin{array}{ccccc} (\infty, 0) & (5, 1) & (0, \infty) & (0, \infty) & (0, \infty) \\ (0, \infty) & (\infty, 0) & (0, \infty) & (0, \infty) & (0, \infty) \\ (\mathbf{5}, \mathbf{7}) & (5, 3) & (\infty, 0) & (5, 1) & (5, 2) \\ (10, 6) & (5, 2) & (5, 2) & (\infty, 0) & (10, 1) \\ (10, 5) & (5, 4) & (5, 1) & (5, 2) & (\infty, 0) \end{array} \right] \end{array},$$

Entries marked in **bold** indicate those values which are not globally optimal.

Left-locally optimal paths to node 2

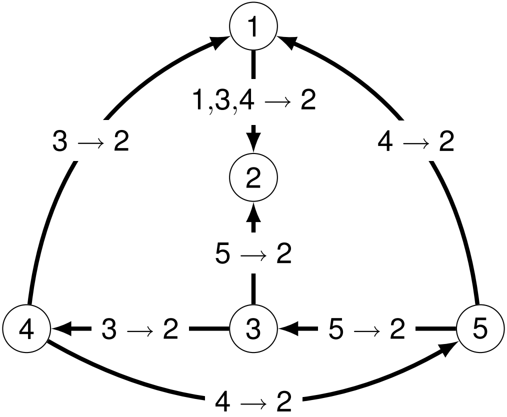


Right local optima

$$\mathbf{R} = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{array} \begin{array}{c} \begin{array}{ccccc} 1 & 2 & 3 & 4 & 5 \end{array} \\ \left[\begin{array}{ccccc} (\infty, 0) & (5, 1) & (0, \infty) & (0, \infty) & (0, \infty) \\ (0, \infty) & (\infty, 0) & (0, \infty) & (0, \infty) & (0, \infty) \\ (5, 2) & (5, 3) & (\infty, \ 0) & (5, \ 1) & (5, 2) \\ (10, 6) & (\mathbf{5}, \mathbf{6}) & (5, 2) & (\infty, 0) & (10, 1) \\ (10, 5) & (\mathbf{5}, \mathbf{5}) & (5, 1) & (5, 2) & (\infty, 0) \end{array} \right] \end{array},$$

Note : the $(\mathbf{5}, \mathbf{6})$ is $(\mathbf{5}, \mathbf{7})$ in the paper, which appears to be a bug!

Right-locally optimal paths to node 2



What are the conditions needed to guarantee existence of local optima?

For a non-distributed structure $S = (S, \oplus, \otimes, \overline{0}, \overline{1})$, can be used to find local optima when the following property holds.

Strictly Inflationary

$$\text{S.INFL} : \forall a, \; b \in S : a \neq \overline{0} \implies a < b \otimes a$$

where $a \leq b$ means $a = a \oplus b$.

We know that (a modified) Bellman-Ford iteration will converge, but we currently have no bound on the number of iterations needed!

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 $\leq_{\oplus}^{L}$ -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

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 $\mathbf{R}(i,\ q)$ is $\leq^L_{\oplus}$ -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

Assumptions on $(S, \oplus, \otimes, \overline{0}, \overline{1})$

- $(S, \oplus, \overline{0})$ is a commutative, idempotent, and selective monoid,
- $(S, \otimes, \overline{1})$ is a monoid,
- $\overline{0}$ is the annihilator for $\otimes$,
- $\overline{1}$ is the annihilator for $\oplus$,
- $\mathrm{RINF} : \forall a, b : a \leq a \otimes b$

Recall that $a \leq b \equiv a \leq_{\oplus}^{L} b \equiv a = a \oplus b$.

The goal

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

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

Main Claim

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

Observation 1

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

This is easy to see — once a node is put into S its weight never changes.

Observation 2

Observation 2

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

By induction.
Base : Need $\overline{1} \leq \mathbf{A}(i,\ w)$. OK
Induction. Assume

$$\forall q \in S_k : \forall w \in V - S_k : \mathbf{R}_k(i,\ q) \leq \mathbf{R}_k(i,\ w)$$

and show

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

Since $S_{k+1} = S_k \cup \{q_{k+1}\}$, this is means showing

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

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

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

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

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

But $\mathbf{R}_k(i,\ q) \le \mathbf{R}_k(i,\ w)$ by the induction hypothesis, and $\mathbf{R}_k(i,\ q) \le (\mathbf{R}_k(i,\ q_{k+1}) \otimes \mathbf{A}(q_{k+1},\ w))$ by the induction hypothesis and RINF.

Since $a \le_{\oplus}^{L} b \wedge a \le_{\oplus}^{L} c \implies a \le_{\oplus}^{L} (b \oplus c)$, we are done.

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

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

which expands to

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

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

Observation 3

Observation 3

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

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

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

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

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

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,

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

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

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

by RINF.

2

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?

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

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

where

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

Notice that this exchanges RINF for LINF!

$$\text{LINF} : \forall a, b : a \leq b \otimes a$$

Conclusion

- Complexity of solving for left local optima?
  - Previous work has shown that Bellman-Ford will find a solution as long as only simple paths are explored — but no time bounds are known.
  - But, now we know that $O(V^3)$ will due with Dijkstra's greedy algorithm.
  - Could do better in sparse graphs using Fibonacci heaps ...