Data Transfer for Distributed Analytics with Latency Constraints

Stephen Pasteris (UCL), **Shiqiang Wang** (IBM US)

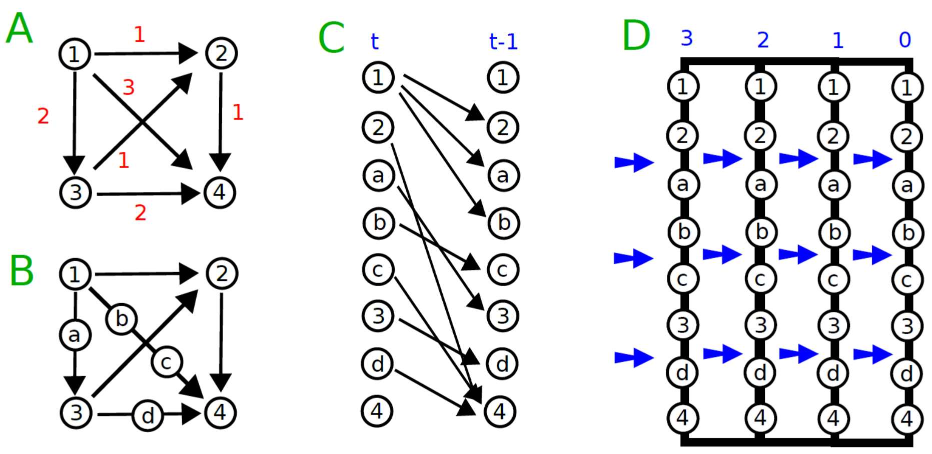
Problem Description

- Directed graph $G = (V, E)$
  - The vertices of the graph represent routers and the edges of the graph represent (direct) communication pipelines
- For every directed edge $(\mathrm{v}, \mathrm{w}) \in E$, we have an associated bandwidth, $\beta(\mathrm{v}, \mathrm{w}) \in \mathbb{R}^+$, and transmission latency, $\lambda(\mathrm{v}, \mathrm{w}) \in \mathbb{R}^+$

The problem: We have Δ bits of data stored at a "source'" vertex, $\perp \in V$, which need to be transferred to a "sink" vertex, $\top \in V$. We need to compute how to route the data so that this data-transfer takes the minimum possible time.

Frame Graph

Graph Definition



A: The original graph G with latencies in red
B: Intermediate graph
C: Frame t of the frame graph
D: The first three frames of the frame graph (blue arrows represent the edges in diagram C)

Dynamic Flow

A "dynamic flow", F, on the frame graph H is a function $F: D \to \mathbb{R}^+$ such that $\sum_{u \in U:(u,v) \in D} F(u,v) = \sum_{w \in U:(v,w) \in D} F(v,w)$ for all $v \in U$ with $v \notin \bigcup_{t \in \mathbb{N}} \{\perp(t), \top(t)\}$.

Constructing a Dynamic Flow

For $i \in \mathbb{N}$, at the start of round i, we have a dynamic flow F_i, initialised such that $F_1(e) = 0$ for all $e \in D$. On round i we do the following:

- Let s be equal to the minimum t such that that there exists an $\mathrm{F_i}$-augmenting path from $\perp(\mathrm{t})$ to $\top(1)$. Let P be such a path. If there is no such F_i-augmenting path then quit the algorithm with F_i being the required dynamic flow.
- Set $F' \leftarrow F_\mathrm{i} \oplus P$
- Write P as a sequence $\{x_1, x_2, x_3, \dots, x_l\}$, of vertices in U
- For all $i < l$:
  - If $\psi_P(x_i) = 1$ set $e \leftarrow (x_i, x_{i+1})$. Else set $e \leftarrow (x_{i+1}, x_i)$
  - Let r be such that $e \in Z_r$
  - For all $t > r$ set $F'(\uparrow_t(e)) \leftarrow F'(e)$
- Set $F_{i+1} \leftarrow F'$

Implicit Construction

To reduce computational complexity

Definition: Given a path $P \coloneqq \{v_1, v_2, \dots, v_l\}$ in G', its "effective length" is defined equal to

$$\sum_{i<l:\psi_P(v_i)=1} \lambda(v_i, v_{i+1}) - \sum_{i<l:\psi_P(v_i)=-1} \lambda(v_{i+1}, v_i)$$

Algorithm: On round i the algorithm does the following:

- Let P be an f_i-augmenting path in G' from $\perp$ to $\top$ of minimum effective length. If there is no such path then the algorithm terminates with the required scheduling.
- Set $f_{i+1} \leftarrow f_i \oplus P$
- Let $(\perp = v_1, v_2, \dots, v_l = \top) \coloneqq P$
- For all $i < l$:
  - If $\psi_P(v_i) = 1$ let $e \leftarrow (v_i, v_{i+1})$; else let $e \leftarrow (v_{i+1}, v_i)$
  - Let ζ be the effective length of the path $(v_i, v_{i+1}, v_{i+2}, \dots, v_l = \top)$
  - $\mu(e) \leftarrow \mu(e) + 1$
  - $\delta_e(\mu(e)) \leftarrow \zeta$
  - $\epsilon_e(\mu(e)) \leftarrow f_{i+1}(e)$

Distributed Analytics and Information Science
International Technology Alliance
Annual Fall Meeting, September 2017