Spider: High-Efficiency Cryptocurrency Routing for Payment Channel Networks

Vibhaalakshmi Sivaraman[1], Shaileshh Bojja Venkatakrishnan[1], Kathleen Ruan[2], Lei Yang[1], Parimarjan Negi[1], Radhika Mittal[1,3], Mohammad Alizadeh[1], Giulia Fanti[2], Pramod Viswanath[3]

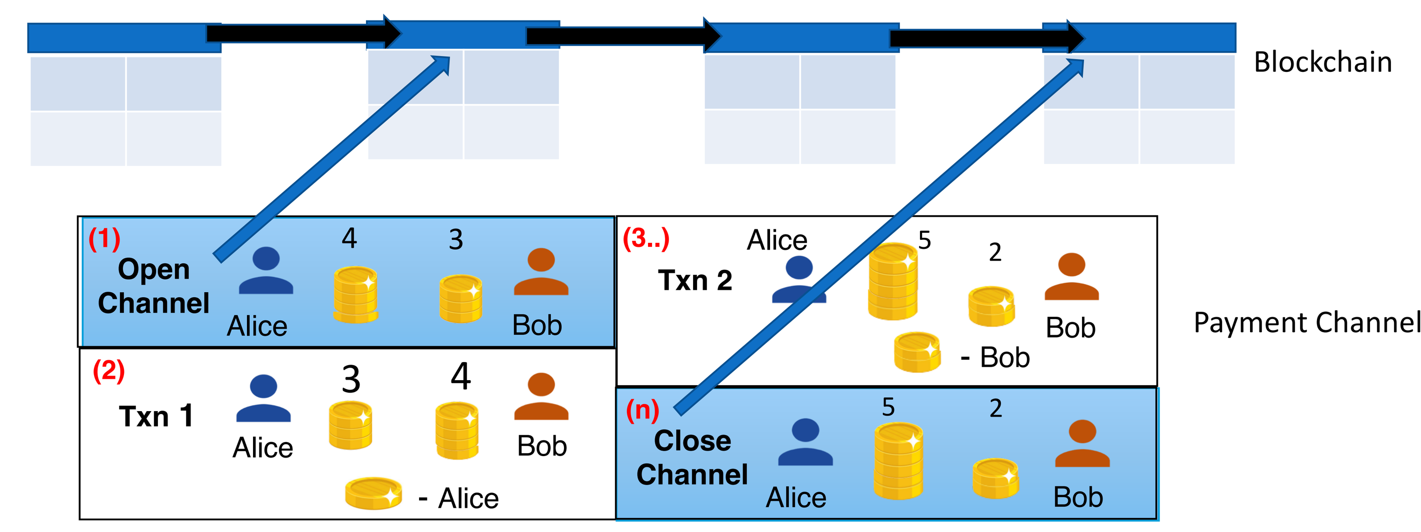
spider.network@mit.edu

[1]MIT CSAIL [2]CMU [3]UIUC

Motivation

- Blockchains are not scalable.
  - slow confirmation times, high transaction costs.
- Leading solution: Payment channel networks (PCNs)
  - speed up transactions by reducing use of blockchain.
- Inefficient transaction routing in PCNs degrades transaction throughput and latency.

Goal: Routing for high transaction throughput on PCNs with small network capital.



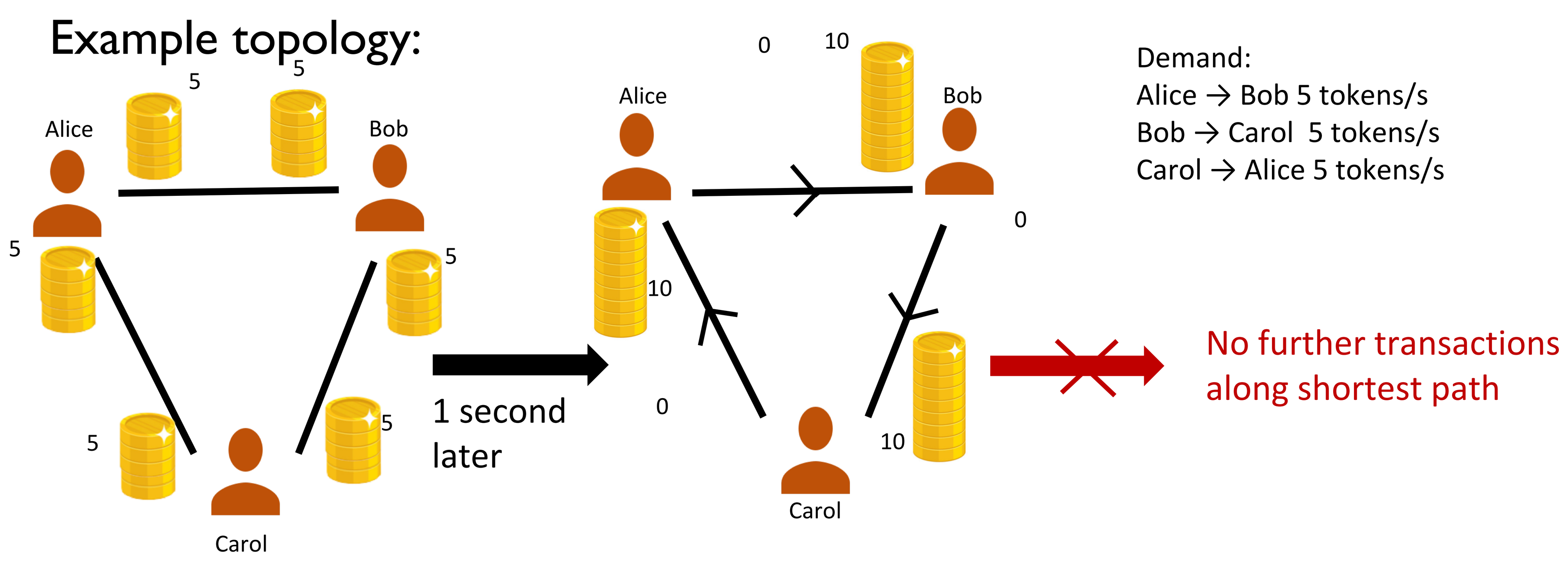
Routing in Payment Channel Networks

State of the art: atomic shortest path routing

Problems:

- Cannot route large payments.
- Causes channel imbalance.

Example topology:



Spider:

- **Packetized payments:** Split payments into "transaction units" that are routed independently on multiple paths over time.
- **Balance aware-routing:** Route based on real time channel balance information.

LP under fluid model of transactions

Total Throughput

$$\text{maximize} \quad \sum_{i,j \in V} \sum_{p \in \mathcal{P}_{i,j}} x_p$$

demand

$$\text{s.t.} \quad \sum_{p \in \mathcal{P}_{i,j}} x_p \le d_{i,j} \quad \forall i,j \in V$$

balance

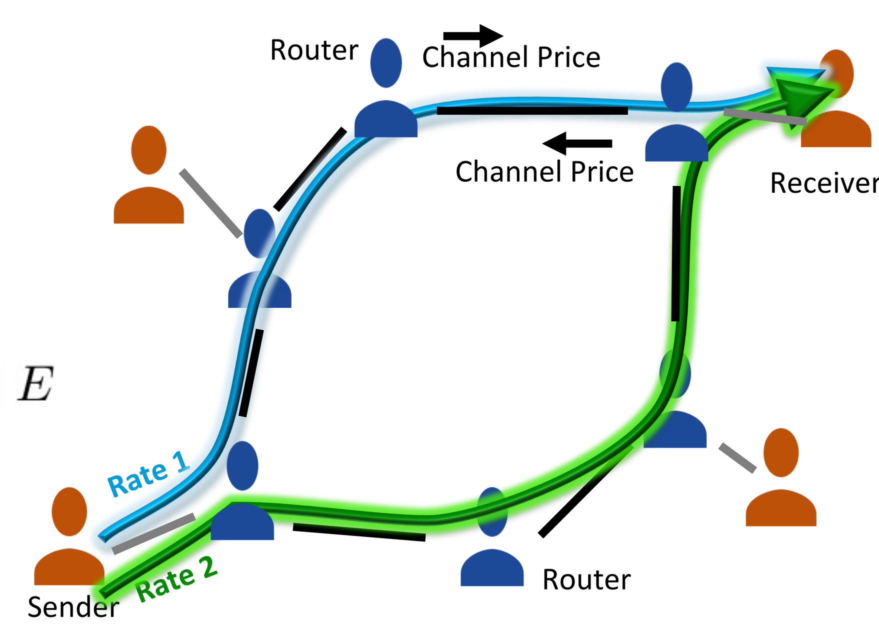
$$x_{(u,v)} - x_{(v,u)} \le 0 \quad \forall (u,v) \in E$$

capacity

$$x_{(u,v)} + x_{(v,u)} \le \frac{c_{(u,v)}}{\Delta} \quad \forall (u,v) \in E$$

$$x_p \ge 0 \quad \forall p \in P$$

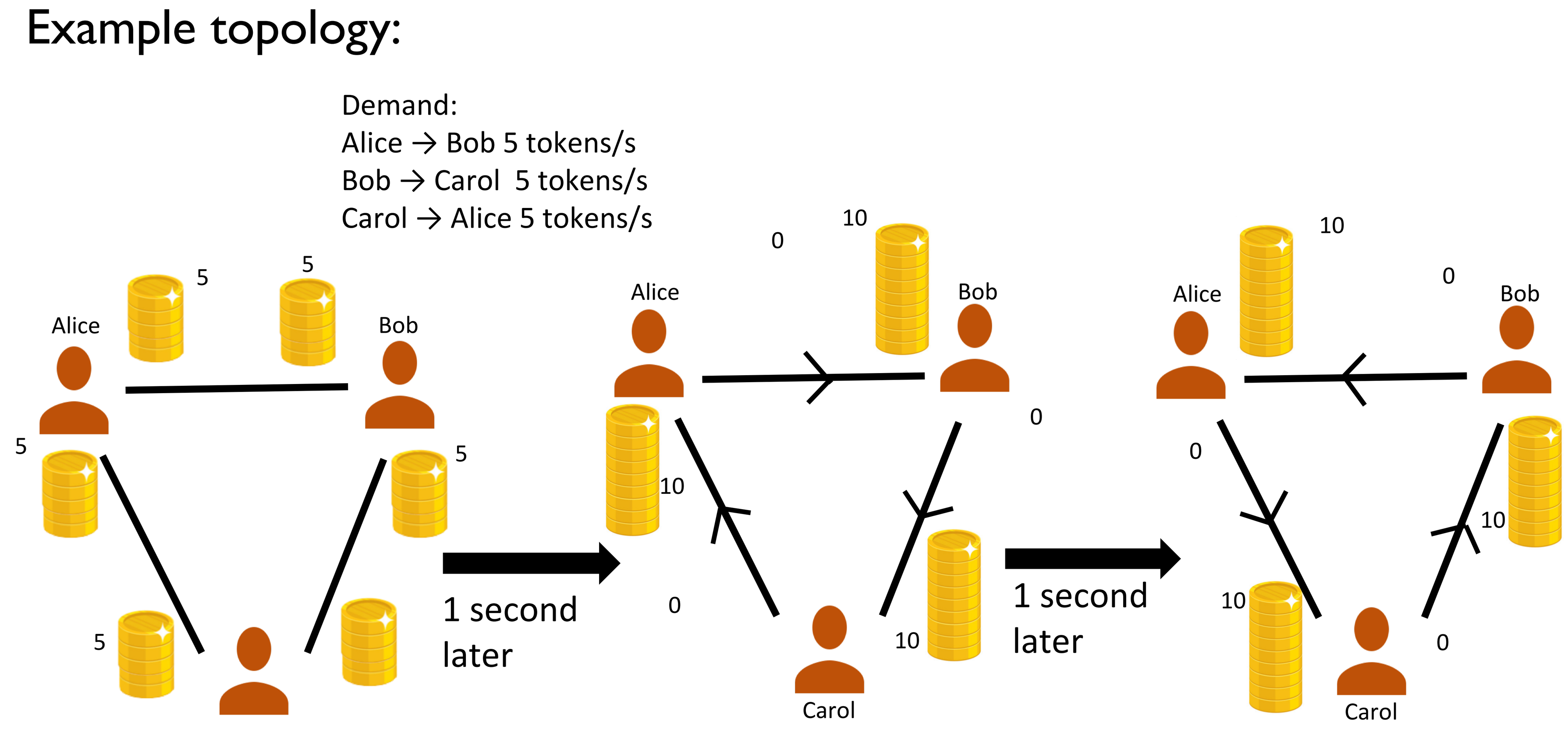
Routers: Set channel prices based on congestion and imbalance.



Senders: Pick rates to send on K candidate paths based on network feedback.

Primal-dual solution to LP leads to a multi-path transport protocol.

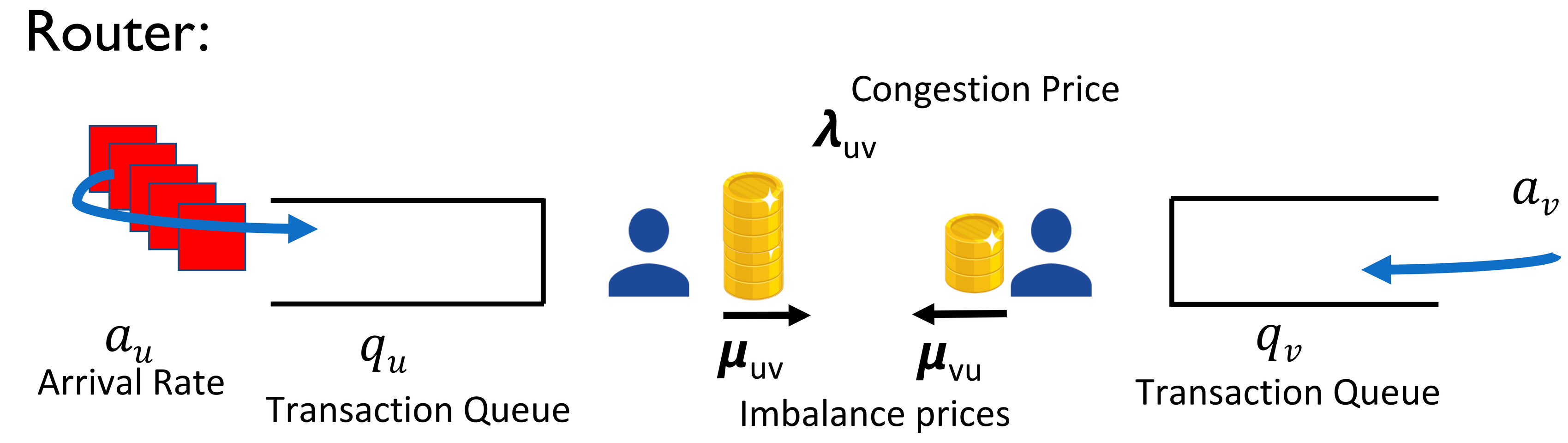
Example topology:



Entire demand can be satisfied indefinitely by alternating between the shorter and longer paths.

Spider Design

Router:



Imbalance price: $\mu_{uv} \leftarrow [\mu_{uv} + \kappa\underbrace{(a_u - a_v)}_{\text{Difference in arrival rates}}]_+$

Congestion price:

Channel capacity

$$\lambda_{uv} \leftarrow [\lambda_{uv} + \eta(a_u\Delta_u + a_v\Delta_v - c_{uv})]_+$$

Funds needed from v->u

Funds needed from v->u

Δ = Transaction confirmation delay

End-Host:

Periodic probes to compute path price for candidate K paths.

Path Price

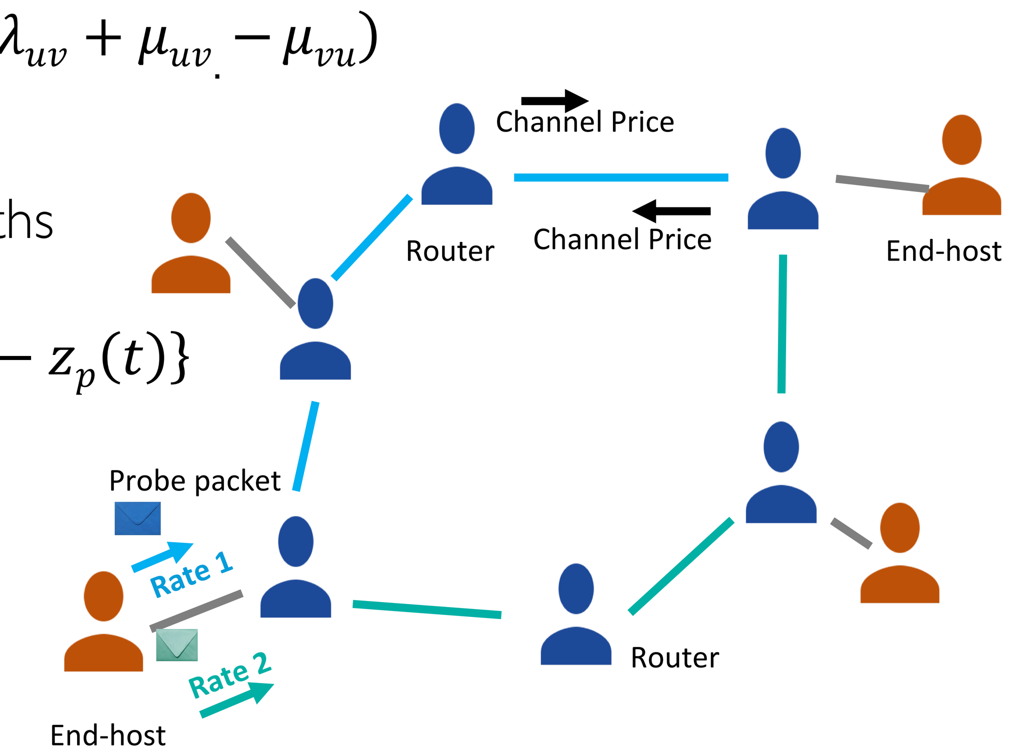
$$z_p = \sum_{(u,v):(u,v)\in p} (\lambda_{uv} + \mu_{uv} - \mu_{vu})$$

Update rates for candidate paths based on path price.

$$x_p(t+1) = x_p(t) + \alpha\{1 - z_p(t)\}$$

Rate

Window based flow control to control sending rate.



Results