Semiring Algebraic Structure for Metarouting with Automatic Tunneling

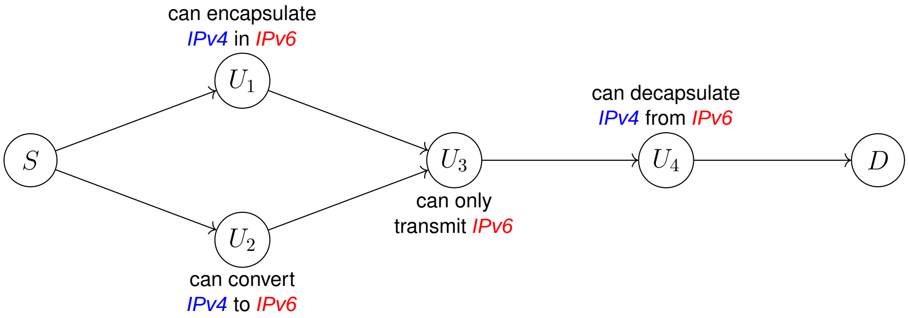
Noureddine Mouhoub
Mohamed Lamine Lamali, and Damien Magoni

LaBRI - CNRS, University of Bordeaux, France

New IP and Beyond Workshop (IEEE ICNP)
October 30-November 2, 2022
Lexington, Kentucky, USA

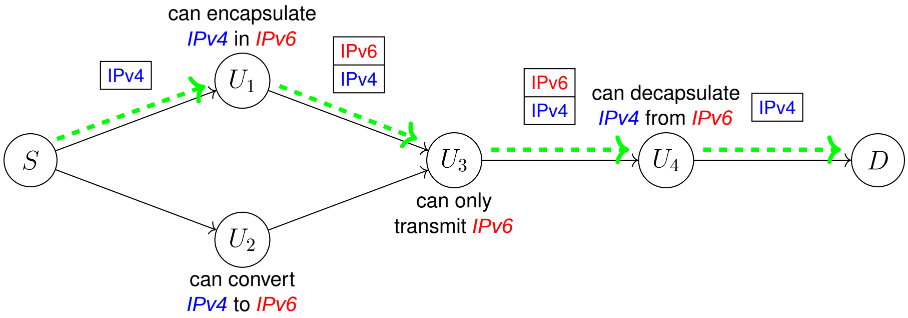
Outline

Routing with Automatic Tunneling



Example of a network encompassing IPv4 and IPv6 protocols.

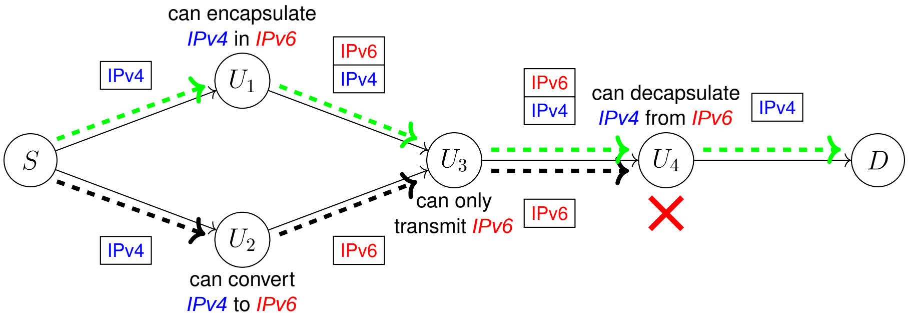
Routing with Automatic Tunneling



Example of a network encompassing IPv4 and IPv6 protocols.

- The top path (in green) is valid with a tunnel from U_1 to U_4.

Routing with Automatic Tunneling



Example of a network encompassing IPv4 and IPv6 protocols.

- The top path (in green) is valid with a tunnel from U_1 to U_4.
- The bottom path (in black) is invalid.

Routing with Automatic Tunneling

- The computation of paths (routing) in networks with tunnels is not yet fully automated. (*e.g.*, Teredo, 6over4, 6to4, ISATAP, etc.)
- The path computation in multi-protocol networks with conversions and encapsulations, cannot be performed by using classical path computation algorithms. (*e.g.*, Dijkstra, Bellman-Ford, etc.)
- There are a few path computation algorithms which take into account encapsulations and conversions. (*e.g.*, Stack-Vector, etc.)
- The valid path problem under bandwidth constraints is NP-hard.

Metarouting

- Metarouting is an algebraic model of routing protocols.
  (semiring, Sobrinho's algebra and algebra of endomorphisms.)

Metarouting

- Metarouting is an algebraic model of routing protocols.
  (semiring, Sobrinho's algebra and algebra of endomorphisms.)

Routing Algebra = $(S, \oplus, \otimes, \overline{0}, \overline{1})$

- S: set of routes
- $\oplus$: route selection
- $\otimes$: route extension
- $\overline{0}$: initial route
- $\overline{1}$: invalid route

Metarouting

- Metarouting is an algebraic model of routing protocols.
  (semiring, Sobrinho's algebra and algebra of endomorphisms.)

Routing Algebra = $(S, \oplus, \otimes, \overline{0}, \overline{1})$

- S: set of routes
- $\oplus$: route selection
- $\otimes$: route extension
- $\overline{0}$: initial route
- $\overline{1}$: invalid route

- Separates the routing data and algorithm.
  (route exchange and route update.)

- Study the properties of routing protocols.
  (convergence and the set of optimal paths.)

Routing Algebras with Tunnels

- Most research work on routing algebras have been applied to routing protocols used in networks having a single addressing and forwarding protocol.

- There is no proof of convergence of the asynchronous distributed stack-vector algorithm.

Routing Algebras with Tunnels

- Most research work on routing algebras have been applied to routing protocols used in networks having a single addressing and forwarding protocol.

- There is no proof of convergence of the asynchronous distributed stack-vector algorithm.

Our contribution is to generalize the semiring structure for modeling the routing problem with automatic tunneling and to study some algebraic properties of convergence.

Network Model

A multi-layer network $\mathcal{N}$ is modeled by:

Network Model

A multi-layer network $\mathcal{N}$ is modeled by:

- A directed graph of size n, $\mathcal{G} = (\mathcal{V}, \mathcal{E})$

Network Model

A multi-layer network $\mathcal{N}$ is modeled by:

- A directed graph of size n, $\mathcal{G} = (\mathcal{V}, \mathcal{E})$
- A set of λ communication protocols, $\mathcal{A} = \{x, y, \dots\}$
  - The set of protocols that a node U can receive is $In(U)$
  - The set of protocols that a node U can send is $Out(U)$

Network Model

A multi-layer network $\mathcal{N}$ is modeled by:

- A directed graph of size n, $\mathcal{G} = (\mathcal{V}, \mathcal{E})$
- A set of λ communication protocols, $\mathcal{A} = \{x, y, \dots\}$
  - The set of protocols that a node U can receive is $In(U)$
  - The set of protocols that a node U can send is $Out(U)$
- A set of adaptation functions, $\mathcal{F}$ of type:
  - $(x \rightarrow x)$ is the retransmission of the protocol x
  - $(x \rightarrow y)$ is the conversion of the protocol x to y
  - $(x \rightarrow xy)$ is the encapsulation of the protocol x in y
  - $(\overline{x \rightarrow xy})$ is the decapsulation of the protocol x from y

Network Model

A multi-layer network $\mathcal{N}$ is modeled by:

- A directed graph of size n, $\mathcal{G} = (\mathcal{V}, \mathcal{E})$
- A set of λ communication protocols, $\mathcal{A} = \{x, y, \dots\}$
  - The set of protocols that a node U can receive is $In(U)$
  - The set of protocols that a node U can send is $Out(U)$
- A set of adaptation functions, $\mathcal{F}$ of type:
  - $(x \to x)$ is the retransmission of the protocol x
  - $(x \to y)$ is the conversion of the protocol x to y
  - $(x \to xy)$ is the encapsulation of the protocol x in y
  - $(\overline{x \to xy})$ is the decapsulation of the protocol x from y
- A weight function, $\omega : \mathcal{V} \times \mathcal{F} \times \mathcal{V} \to \mathcal{R}^+$

Protocol Stack and Functions Composition

Let f be an adaptation function and h a protocol stack (a sequence of encapsulated - nested protocols).

Protocol Stack and Functions Composition

Let f be an adaptation function and h a protocol stack (a sequence of encapsulated - nested protocols).

- The valid application of f on h is $f(h) = h'$. Otherwise $f(h) = \phi$.
- The protocol at the top of the stack h is denoted by $Top(h)$.

Protocol Stack and Functions Composition

Let f be an adaptation function and h a protocol stack (a sequence of encapsulated - nested protocols).

- The valid application of f on h is $f(h) = h'$. Otherwise $f(h) = \phi$.
- The protocol at the top of the stack h is denoted by $Top(h)$.

Let $f_i f_{i+1} \dots f_{j-1} f_j$ be a sequence of functions and h_i an initial stack.

Protocol Stack and Functions Composition

Let f be an adaptation function and h a protocol stack (a sequence of encapsulated - nested protocols).

- The valid application of f on h is $f(h) = h'$. Otherwise $f(h) = \phi$.
- The protocol at the top of the stack h is denoted by $Top(h)$.

Let $f_i f_{i+1} \ldots f_{j-1} f_j$ be a sequence of functions and h_i an initial stack.

- The induced protocol stacks are defined recursively:

$$h_k = f_{k-1}(h_{k-1}), \quad i+1 \leq k \leq j+1$$

Protocol Stack and Functions Composition

Let f be an adaptation function and h a protocol stack (a sequence of encapsulated - nested protocols).

- The valid application of f on h is $f(h) = h'$. Otherwise $f(h) = \phi$.
- The protocol at the top of the stack h is denoted by $Top(h)$.

Let $f_i f_{i+1} \dots f_{j-1} f_j$ be a sequence of functions and h_i an initial stack.

- The induced protocol stacks are defined recursively:

$$h_k = f_{k-1}(h_{k-1}), \quad i + 1 \leq k \leq j + 1$$

Let $\mathcal{H} = \{\phi, x, y, xy, xyx, \dots\}$ be the set of all possible protocol stacks.

Protocol Stack and Functions Composition

Let f be an adaptation function and h a protocol stack (a sequence of encapsulated - nested protocols).

- The valid application of f on h is $f(h) = h'$. Otherwise $f(h) = \phi$.
- The protocol at the top of the stack h is denoted by $Top(h)$.

Let $f_i f_{i+1} \dots f_{j-1} f_j$ be a sequence of functions and h_i an initial stack.

- The induced protocol stacks are defined recursively:

$$h_k = f_{k-1}(h_{k-1}), \quad i+1 \leq k \leq j+1$$

Let $\mathcal{H} = \{\phi, x, y, xy, xyx, \dots\}$ be the set of all possible protocol stacks.

- The set of all stacks starting with the sub-stack xy is $H_{xy} \in \mathcal{H}$.
- An adaptation function $f = (x \to xy)$ is defined as $f : H_x \to H_{xy}$.

Protocol Stack and Functions Composition

- The composition of two functions $f : H_a \rightarrow H_b$ and $f' : H'_a \rightarrow H'_b$ is the new function, $f'' = f' \odot f$, where:

Protocol Stack and Functions Composition

- The composition of two functions $f : H_a \rightarrow H_b$ and $f' : H'_a \rightarrow H'_b$ is the new function, $f'' = f' \odot f$, where:

$$f'' = \begin{cases} H''_a \rightarrow H''_b & \text{if } (H''_a \neq \emptyset) \wedge (H''_b \neq \emptyset) \\ \{\phi\} \rightarrow \{\phi\} & \text{otherwise} \end{cases}$$

And

$$H''_a = \Big\{h \in H_a \mid f(h) \in H'_a\Big\} \qquad H''_b = \Big\{f'(h) \mid h \in (H_b \cap H'_a)\Big\}$$

Protocol Stack and Functions Composition

- The composition of two functions $f : H_a \to H_b$ and $f' : H'_a \to H'_b$ is the new function, $f'' = f' \odot f$, where:

$$f'' = \begin{cases} H''_a \to H''_b & \text{if } (H''_a \neq \emptyset) \land (H''_b \neq \emptyset) \\ \{\phi\} \to \{\phi\} & \text{otherwise} \end{cases}$$

And

$$H''_a = \Big\{ h \in H_a \mid f(h) \in H'_a \Big\} \qquad H''_b = \Big\{ f'(h) \mid h \in (H_b \cap H'_a) \Big\}$$

- The set of all adaptation functions close under composition is:

$$\hat{\mathcal{F}} = \Big\{ \phi \to \phi, x \to x, x \to y, x \to xxx, xyx \to x, \ldots \Big\}$$

Protocol Stack and Functions Composition

- The composition of two functions $f : H_a \to H_b$ and $f' : H'_a \to H'_b$ is the new function, $f'' = f' \odot f$, where:

$$f'' = \begin{cases} H''_a \to H''_b & \text{if } (H''_a \neq \emptyset) \wedge (H''_b \neq \emptyset) \\ \{\phi\} \to \{\phi\} & \text{otherwise} \end{cases}$$

And

$$H''_a = \left\{ h \in H_a \mid f(h) \in H'_a \right\} \qquad H''_b = \left\{ f'(h) \mid h \in (H_b \cap H'_a) \right\}$$

- The set of all adaptation functions close under composition is:

$$\hat{\mathcal{F}} = \left\{ \phi \to \phi, x \to x, x \to y, x \to xxx, xyx \to x, \ldots \right\}$$

- The set of all identity functions (classical retransmission) is:

$$\mathcal{F}_{id} = \left\{ x \to x, y \to y, \ldots \right\}$$

Path Validity

- A path $p = h_i v_i f_i v_{i+1} f_{i+1} \dots v_{j-1} f_{j-1} h_j v_j$ from v_i to v_j is a mixed sequence of nodes and adaptation functions and it is valid iff:

Path Validity

- A path $p = h_i v_i f_i v_{i+1} f_{i+1} \ldots v_{j-1} f_{j-1} h_j v_j$ from v_i to v_j is a mixed sequence of nodes and adaptation functions and it is valid iff:
  - The starting stack, $h_i \neq \phi$ and $Top(h_i) \in In(v_i)$
  - The ending stack, $h_j \neq \phi$ and $Top(h_j) \in In(v_j)$

Path Validity

- A path $p = h_i v_i f_i v_{i+1} f_{i+1} \ldots v_{j-1} f_{j-1} h_j v_j$ from v_i to v_j is a mixed sequence of nodes and adaptation functions and it is valid iff:
  - The starting stack, $h_i \neq \phi$ and $Top(h_i) \in In(v_i)$
  - The ending stack, $h_j \neq \phi$ and $Top(h_j) \in In(v_j)$
  - The sequence $v_i v_{i+1} \ldots v_{j-1} v_j$ is a classical path in $\mathcal{G}$

Path Validity

- A path $p = h_i v_i f_i v_{i+1} f_{i+1} \ldots v_{j-1} f_{j-1} h_j v_j$ from v_i to v_j is a mixed sequence of nodes and adaptation functions and it is valid iff:
  - The starting stack, $h_i \neq \phi$ and $Top(h_i) \in In(v_i)$
  - The ending stack, $h_j \neq \phi$ and $Top(h_j) \in In(v_j)$
  - The sequence $v_i v_{i+1} \ldots v_{j-1} v_j$ is a classical path in $\mathcal{G}$
  - The sequence $f_i f_{i+1} \ldots f_{j-1}$ induces the protocol stack, $h_j = f_{j-1}\Big(f_{j-2}\big(\ldots f_{i+1}\big(f_i(h_i)\big)\big)\Big)$

Path Validity

- A path $p = h_i v_i f_i v_{i+1} f_{i+1} \ldots v_{j-1} f_{j-1} h_j v_j$ from v_i to v_j is a mixed sequence of nodes and adaptation functions and it is valid iff:
  - The starting stack, $h_i \neq \phi$ and $Top(h_i) \in In(v_i)$
  - The ending stack, $h_j \neq \phi$ and $Top(h_j) \in In(v_j)$
  - The sequence $v_i v_{i+1} \ldots v_{j-1} v_j$ is a classical path in $\mathcal{G}$
  - The sequence $f_i f_{i+1} \ldots f_{j-1}$ induces the protocol stack, $h_j = f_{j-1}\Big(f_{j-2}\big(\ldots f_{i+1}\big(f_i(h_i)\big)\big)\Big)$

Note that a valid path $p = h_i v_i f_i \ldots v_{j-1} f_{j-1} h_j v_j$ can be represented by the composition $f_{j-1} \odot \cdots \odot f_{i+1} \odot f_i$

Path Validity

- A path $p = h_i v_i f_i v_{i+1} f_{i+1} \dots v_{j-1} f_{j-1} h_j v_j$ from v_i to v_j is a mixed sequence of nodes and adaptation functions and it is valid iff:
  - The starting stack, $h_i \neq \phi$ and $Top(h_i) \in In(v_i)$
  - The ending stack, $h_j \neq \phi$ and $Top(h_j) \in In(v_j)$
  - The sequence $v_i v_{i+1} \dots v_{j-1} v_j$ is a classical path in $\mathcal{G}$
  - The sequence $f_i f_{i+1} \dots f_{j-1}$ induces the protocol stack, $h_j = f_{j-1}\Big(f_{j-2}\big(\dots f_{i+1}\big(f_i(h_i)\big)\big)\Big)$

  Note that a valid path $p = h_i v_i f_i \dots v_{j-1} f_{j-1} h_j v_j$ can be represented by the composition $f_{j-1} \odot \dots \odot f_{i+1} \odot f_i$

- The weight of a valid path p from v_i to v_j is the sum of the weights of its links and its adaptation functions,

$$\omega(p) \stackrel{def}{=} \sum_{k=i}^{j-1} \omega(v_k, f_i, v_{k+1})$$

Semiring and Properties

A semiring is a structure $SM = (S, \oplus, \otimes, \overline{0}, \overline{1})$ where:

Semiring and Properties

A semiring is a structure $SM = (S, \oplus, \otimes, \overline{0}, \overline{1})$ where:

- $\oplus$ and $\otimes$ are associative binary operations over S
- $\oplus$ is commutative

Semiring and Properties

A semiring is a structure $SM = (S, \oplus, \otimes, \overline{0}, \overline{1})$ where:

- $\oplus$ and $\otimes$ are associative binary operations over S
- $\oplus$ is commutative
- $\overline{0}$ is an identity element for $\oplus$ and an annihilator for $\otimes$
- $\overline{1}$ is an identity for $\otimes$

Semiring and Properties

A semiring is a structure $SM = (S, \oplus, \otimes, \overline{0}, \overline{1})$ where:

- $\oplus$ and $\otimes$ are associative binary operations over S
- $\oplus$ is commutative
- $\overline{0}$ is an identity element for $\oplus$ and an annihilator for $\otimes$
- $\overline{1}$ is an identity for $\otimes$
- $\otimes$ is left and right distributive on $\oplus$

Semiring and Properties

A semiring is a structure $SM = (S, \oplus, \otimes, \overline{0}, \overline{1})$ where:

- $\oplus$ and $\otimes$ are associative binary operations over S
- $\oplus$ is commutative
- $\overline{0}$ is an identity element for $\oplus$ and an annihilator for $\otimes$
- $\overline{1}$ is an identity for $\otimes$
- $\otimes$ is left and right distributive on $\oplus$

Problem	S	$\oplus$	$\otimes$	$\overline{0}$	$\overline{1}$
shortest paths (SM_{sp})	$\mathbb{N}^{\infty}$	min	+	$+\infty$	0
widest paths	$\mathbb{N}^{\infty}$	max	min	0	$+\infty$
most reliable paths	$[0, 1]$	max	$\times$	0	1
accessible paths	{0,1}	max	min	0	1

Semiring and Properties

A semiring is a structure $SM = (S, \oplus, \otimes, \overline{0}, \overline{1})$ where:

- $\oplus$ and $\otimes$ are associative binary operations over S
- $\oplus$ is commutative
- $\overline{0}$ is an identity element for $\oplus$ and an annihilator for $\otimes$
- $\overline{1}$ is an identity for $\otimes$
- $\otimes$ is left and right distributive on $\oplus$

Semiring and Properties

A semiring is a structure $SM = (S, \oplus, \otimes, \overline{0}, \overline{1})$ where:

- $\oplus$ and $\otimes$ are associative binary operations over S
- $\oplus$ is commutative
- $\overline{0}$ is an identity element for $\oplus$ and an annihilator for $\otimes$
- $\overline{1}$ is an identity for $\otimes$
- $\otimes$ is left and right distributive on $\oplus$

If $\oplus$ is idempotent then the relation $\leq_{\oplus}$ is a partial order over S:

$$(a \leq_{\oplus} b) \equiv (a = a \oplus b)$$
$$(a <_{\oplus} b) \equiv (a = a \oplus b \neq b)$$

Note that this order is total when the operation $\oplus$ is selective

Matrix Semirings

Given a semiring $(S, \oplus, \otimes, \overline{0}, \overline{1})$, the semiring of $n \times n$ matrix is $\big(\mathbf{M}_n(S), \oplus, \otimes, \mathbf{N}, \mathbf{I}\big)$, where:

- $\mathbf{N}_{i,j} = \overline{0}$
- $\mathbf{I}_{i,j} = \begin{cases} \overline{1} & \text{if } (i = j) \\ \overline{0} & \text{otherwise} \end{cases}$

Matrix Semirings

Given a semiring $(S, \oplus, \otimes, \overline{0}, \overline{1})$, the semiring of $n \times n$ matrix is $\big(\mathbf{M}_n(S), \oplus, \otimes, \mathbf{N}, \mathbf{I}\big)$, where:

- $\mathbf{N}_{i,j} = \overline{0}$
- $\mathbf{I}_{i,j} = \begin{cases} \overline{1} & \text{if } (i = j) \\ \overline{0} & \text{otherwise} \end{cases}$

And for any two matrices $\mathbf{X}, \mathbf{Y} \in \mathbf{M}_n(S)$, the two operations $\oplus$ and $\otimes$ are defined as follow:

- $(\mathbf{X} \oplus \mathbf{Y})_{i,j} = \mathbf{X}_{i,j} \oplus \mathbf{Y}_{i,j}$
- $(\mathbf{X} \otimes \mathbf{Y})_{i,j} = \bigoplus_{k=1}^{n} \mathbf{X}_{i,k} \otimes \mathbf{Y}_{k,j}$

The General Path Computation Problem

Given a semiring $(S, \oplus, \otimes, \overline{0}, \overline{1})$, a directed graph of n nodes $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ and a weight function $\omega : \mathcal{E} \to S$

The General Path Computation Problem

Given a semiring $(S, \oplus, \otimes, \overline{0}, \overline{1})$, a directed graph of n nodes $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ and a weight function $\omega : \mathcal{E} \rightarrow S$

- The weight of a path $p = v_0, v_1, v_2, ..., v_{k-1}, v_k$ is:

$$\omega(p) = \omega_{0,1} \otimes \omega_{1,2} \otimes ... \otimes \omega_{k-1,k} = \bigotimes_{i=0}^{k-1} \omega_{i,i+1}$$

The General Path Computation Problem

Given a semiring $(S, \oplus, \otimes, \overline{0}, \overline{1})$, a directed graph of n nodes $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ and a weight function $\omega : \mathcal{E} \rightarrow S$

- The weight of a path $p = v_0, v_1, v_2, ..., v_{k-1}, v_k$ is:

$$\omega(p) = \omega_{0,1} \otimes \omega_{1,2} \otimes ... \otimes \omega_{k-1,k} = \bigotimes_{i=0}^{k-1} \omega_{i,i+1}$$

- The weighted adjacency matrix $\mathbf{A} \in \mathbf{M}_n(S)$ of the graph $\mathcal{G}$ is:

$$\mathbf{A}_{i,j} = \begin{cases} \omega_{i,j} & \text{if } e_{i,j} \in \mathcal{E} \\ \overline{0} & \text{otherwise} \end{cases}$$

The General Path Computation Problem

Given a semiring $(S, \oplus, \otimes, \overline{0}, \overline{1})$, a directed graph of n nodes $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ and a weight function $\omega : \mathcal{E} \rightarrow S$

- The weight of a path $p = v_0, v_1, v_2, ..., v_{k-1}, v_k$ is:

$$\omega(p) \;=\; \omega_{0,1} \otimes \omega_{1,2} \otimes ... \otimes \omega_{k-1,k} \;=\; \bigotimes_{i=0}^{k-1} \omega_{i,i+1}$$

- The weighted adjacency matrix $\mathbf{A} \in \mathbf{M}_n(S)$ of the graph $\mathcal{G}$ is:

$$\mathbf{A}_{i,j} = \begin{cases} \omega_{i,j} & \text{if } e_{i,j} \in \mathcal{E} \\ \overline{0} & \text{otherwise} \end{cases}$$

- The power of a matrix $\mathbf{A} \in \mathbf{M}_n(S)$ is:

$$\mathbf{A}^{\mathbf{k}} = \begin{cases} \mathbf{I} & \text{if } k = 0 \\ \mathbf{A} \otimes \mathbf{A}^{\mathbf{k-1}} & \text{otherwise} \end{cases}$$

The General Path Computation Problem

Let $\mathcal{P}_{i,j}^{k}$ be the set of all paths from node v_i to node v_j of size k
Let $\mathcal{P}_{i,j}^{(k)}$ be the set of all paths from v_i to v_j of size at most k.

The General Path Computation Problem

Let $\mathcal{P}_{i,j}^{k}$ be the set of all paths from node v_i to node v_j of size k

Let $\mathcal{P}_{i,j}^{(k)}$ be the set of all paths from v_i to v_j of size at most k.

- The weight of an optimal path from v_i to v_j of size k is:

$$\mathbf{A}_{i,j}^{k} = \omega_{i,j}^{k} = \bigoplus_{p \in \mathcal{P}_{i,j}^{k}} \omega(p)$$

The General Path Computation Problem

Let $\mathcal{P}^k_{i,j}$ be the set of all paths from node v_i to node v_j of size k

Let $\mathcal{P}^{(k)}_{i,j}$ be the set of all paths from v_i to v_j of size at most k.

- The weight of an optimal path from v_i to v_j of size k is:

$$\mathbf{A}^k_{i,j} = \omega^k_{i,j} = \bigoplus_{p \in \mathcal{P}^k_{i,j}} \omega(p)$$

- The weight of an optimal path from v_i to v_j of size at most k is:

$$\mathbf{A}^{(k)}_{i,j} = \omega^{(k)}_{i,j} = \bigoplus_{p \in \mathcal{P}^{(k)}_{i,j}} \omega(p)$$

The General Path Computation Problem

Let $\mathcal{P}^k_{i,j}$ be the set of all paths from node v_i to node v_j of size k
Let $\mathcal{P}^{(k)}_{i,j}$ be the set of all paths from v_i to v_j of size at most k.

- The weight of an optimal path from v_i to v_j of size k is:

$$\mathbf{A}^k_{i,j} = \omega^k_{i,j} = \bigoplus_{p \in \mathcal{P}^k_{i,j}} \omega(p)$$

- The weight of an optimal path from v_i to v_j of size at most k is:

$$\mathbf{A}^{(k)}_{i,j} = \omega^{(k)}_{i,j} = \bigoplus_{p \in \mathcal{P}^{(k)}_{i,j}} \omega(p)$$

- The global optimal solution for the generalized path computation problem consists in finding (if it exists) the matrix $\mathbf{A}^*$,

$$\mathbf{A}^* = \bigoplus_{k \geq 0} A^{(k)} = \bigoplus_{p \in \mathcal{P}^{(k)}} \omega(p)$$

Valid Shortest Paths Semiring

Main idea: Construct a semiring that enumerates the set of all valid shortest paths between each pair of nodes:

Valid Shortest Paths Semiring

Main idea: Construct a semiring that enumerates the set of all valid shortest paths between each pair of nodes:

$$SM_{sp} = \left(\mathbb{N}^{\infty}, min, +, +\infty, 0\right) \qquad (shortest\ paths\ semiring)$$

$$SM_{vp} = \left(\mathcal{P}(\hat{\mathcal{F}}), \cup, \odot, \emptyset, \mathcal{F}_{id}\right) \qquad (valid\ paths\ semiring)$$

Valid Shortest Paths Semiring

Main idea: Construct a semiring that enumerates the set of all valid shortest paths between each pair of nodes:

$$SM_{sp} = \Big(\mathbb{N}^{\infty}, min, +, +\infty, 0\Big) \qquad (shortest\ paths\ semiring)$$

$$SM_{vp} = \Big(\mathcal{P}(\hat{\mathcal{F}}), \cup, \odot, \emptyset, \mathcal{F}_{id}\Big) \qquad (valid\ paths\ semiring)$$

Where:

- $\mathcal{P}(\hat{\mathcal{F}})$ is the power set of all functions closed under compositions.
- $\odot$ is the composition operation of subsets in $\mathcal{P}(\hat{\mathcal{F}})$:

$$\hat{F}_1 \odot \hat{F}_2 = \Big\{\hat{f}_1 \odot \hat{f}_2 \,|\, \hat{f}_1 \in \hat{F}_1 \text{ and } \hat{f}_2 \in \hat{F}_2\Big\}$$

- $\mathcal{F}_{id}$ is the set of identity adaptation functions $\{x \rightarrow x, y \rightarrow y, \dots\}$.

Valid Shortest Paths Semiring

- The valid shortest paths semiring is the semi-directed product:

$$SM_{vsp} = SM_{vp} \rtimes SM_{sp} = \Big(\mathcal{P}\big(\hat{\mathcal{F}} \times \mathbb{N}^{\infty}\big), \underset{min}{\cup}, (\odot \times +), \emptyset, (\mathcal{F}_{id} \times 0)\Big)$$

Valid Shortest Paths Semiring

- The valid shortest paths semiring is the semi-directed product:

$$SM_{vsp} = SM_{vp} \rtimes SM_{sp} = \Big(\mathcal{P}\big(\hat{\mathcal{F}} \times \mathbb{N}^{\infty}\big), \underset{min}{\cup}, (\odot \times +), \emptyset, (\mathcal{F}_{id} \times 0)\Big)$$

- The extension operator $(\odot \times +)$ is a direct product.
- The selection operator $(\underset{min}{\cup})$ is an union-min product:

$$S_1 \underset{min}{\cup} S_2 = \Big\{(\hat{f}, \omega_{\hat{f}}) \mid (1) \vee (2)\Big\} \qquad \forall S_1, S_2 \in \mathcal{P}\big(\hat{\mathcal{F}} \times \mathbb{N}^{\infty}\big)$$

$$(\hat{f}, \omega_{\hat{f}}) \in S_1 \wedge \forall\, (\hat{g}, \omega_{\hat{g}}) \in S_2,\ (\hat{f} = \hat{g}) \Rightarrow \omega_{\hat{f}} = min[\omega_{\hat{f}}, \omega_{\hat{g}}] \qquad (1)$$

$$(\hat{f}, \omega_{\hat{f}}) \in S_2 \wedge \forall\, (\hat{g}, \omega_{\hat{g}}) \in S_1,\ (\hat{f} = \hat{g}) \Rightarrow \omega_{\hat{f}} = min[\omega_{\hat{f}}, \omega_{\hat{g}}] \qquad (2)$$

Valid Shortest Paths Semiring

- The valid shortest paths semiring is the semi-directed product:

$$SM_{vsp} = SM_{vp} \rtimes SM_{sp} = \Big(\mathcal{P}\big(\hat{\mathcal{F}} \times \mathbb{N}^{\infty}\big), \underset{min}{\cup}, (\odot \times +), \emptyset, (\mathcal{F}_{id} \times 0)\Big)$$

- The extension operator $(\odot \times +)$ is a direct product.
- The selection operator $(\underset{min}{\cup})$ is an union-min product:

$$S_1 \underset{min}{\cup} S_2 = \Big\{(\hat{f}, \omega_{\hat{f}}) \mid (1) \vee (2)\Big\} \qquad \forall S_1, S_2 \in \mathcal{P}\big(\hat{\mathcal{F}} \times \mathbb{N}^{\infty}\big)$$

$$(\hat{f}, \omega_{\hat{f}}) \in S_1 \wedge \forall (\hat{g}, \omega_{\hat{g}}) \in S_2,\ (\hat{f} = \hat{g}) \Rightarrow \omega_{\hat{f}} = min[\omega_{\hat{f}}, \omega_{\hat{g}}] \qquad (1)$$

$$(\hat{f}, \omega_{\hat{f}}) \in S_2 \wedge \forall (\hat{g}, \omega_{\hat{g}}) \in S_1,\ (\hat{f} = \hat{g}) \Rightarrow \omega_{\hat{f}} = min[\omega_{\hat{f}}, \omega_{\hat{g}}] \qquad (2)$$

- The partial order relation over elements of $\mathcal{P}\big(\hat{\mathcal{F}} \times \mathbb{N}^{\infty}\big)$:

$$S_1 \subseteq S_2 \equiv \forall (\hat{f}, \omega_{\hat{f}}) \in S_1 \Rightarrow \exists (\hat{g}, \omega_{\hat{g}}) \in S_2,\ (\hat{f} = \hat{g}) \wedge (\omega_{\hat{f}} \leq \omega_{\hat{g}})$$

Algebraic Convergence Properties

- The convergence of routing protocols, e.g., distance-vector and path-vector protocols, is based on two algebraic properties:

Algebraic Convergence Properties

- The convergence of routing protocols, e.g., distance-vector and path-vector protocols, is based on two algebraic properties:

$$a \leq_{\oplus} a \otimes b \equiv a = a \oplus (a \otimes b) \qquad \forall\, a, b \in S \qquad (monotonicity)$$

$$a \leq_{\oplus} b \implies a \otimes c \leq_{\oplus} b \otimes c \qquad \forall\, a, b, c \in S \qquad (isotonicity)$$

Algebraic Convergence Properties

- The convergence of routing protocols, e.g., distance-vector and path-vector protocols, is based on two algebraic properties:

$$a \leq_{\oplus} a \otimes b \equiv a = a \oplus (a \otimes b) \qquad \forall\, a, b \in S \qquad (monotonicity)$$
$$a \leq_{\oplus} b \implies a \otimes c \leq_{\oplus} b \otimes c \qquad \forall\, a, b, c \in S \qquad (isotonicity)$$

- The monotonicity guarantees that the routing protocol converges in any network, but not necessarily to a global optimal solution.
- The isotonicity guarantees that the routing protocol converges to a global optimal solution.

Algebraic Convergence Properties

- The convergence of routing protocols, e.g., distance-vector and path-vector protocols, is based on two algebraic properties:

$$a \leq_{\oplus} a \otimes b \equiv a = a \oplus (a \otimes b) \qquad \forall\, a, b \in S \qquad (monotonicity)$$
$$a \leq_{\oplus} b \implies a \otimes c \leq_{\oplus} b \otimes c \qquad \forall\, a, b, c \in S \qquad (isotonicity)$$

- The monotonicity guarantees that the routing protocol converges in any network, but not necessarily to a global optimal solution.
- The isotonicity guarantees that the routing protocol converges to a global optimal solution.

Proposition

The direct product operator $(\odot \times +)$ over the power set $\mathcal{P}(\hat{\mathcal{F}} \times \mathbb{N}^{\infty})$ is isotonic and not monotonic.

Iterative Convergence Theorem

Theorem [B. A. Carré, 71]

In a free network (without absorbing circuits) we have:

$$\mathbf{A}^* = \mathbf{A}^{(n-1)} = \mathbf{I} \oplus \mathbf{A} \oplus \mathbf{A}^2 \oplus \cdots \oplus \mathbf{A}^{n-1}$$

Iterative Convergence Theorem

Theorem [B. A. Carré, 71]

In a free network (without absorbing circuits) we have:

$$\mathbf{A}^* = \mathbf{A}^{(n-1)} = \mathbf{I} \oplus \mathbf{A} \oplus \mathbf{A}^2 \oplus \cdots \oplus \mathbf{A}^{n-1}$$

- A multilayer circuit is a valid path $p = h_i v_i f_i \dots f_{j-1} h_j v_j$ where:
  - The node v_i is the same node v_j, *i.e.*, $v_i = v_j$
  - The starting and the ending stacks are the same, *i.e.*, $h_i = h_j$

Iterative Convergence Theorem

Theorem [B. A. Carré, 71]

In a free network (without absorbing circuits) we have:

$$\mathbf{A}^* = \mathbf{A}^{(n-1)} = \mathbf{I} \oplus \mathbf{A} \oplus \mathbf{A}^2 \oplus \cdots \oplus \mathbf{A}^{n-1}$$

- A multilayer circuit is a valid path $p = h_i v_i f_i \ldots f_{j-1} h_j v_j$ where:
  - The node v_i is the same node v_j, *i.e.*, $v_i = v_j$
  - The starting and the ending stacks are the same, *i.e.*, $h_i = h_j$
- A multilayer elementary path is a valid path in which its circuits (if it exists) are non multilayer circuits.

Iterative Convergence Theorem

Theorem [B. A. Carré, 71]

In a free network (without absorbing circuits) we have:

$$\mathbf{A}^* = \mathbf{A}^{(n-1)} = \mathbf{I} \oplus \mathbf{A} \oplus \mathbf{A}^2 \oplus \cdots \oplus \mathbf{A}^{n-1}$$

- A multilayer circuit is a valid path $p = h_i v_i f_i \dots f_{j-1} h_j v_j$ where:
  - The node v_i is the same node v_j, *i.e.*, $v_i = v_j$
  - The starting and the ending stacks are the same, *i.e.*, $h_i = h_j$
- A multilayer elementary path is a valid path in which its circuits (if it exists) are non multilayer circuits.
- A free multilayer network is a network in which all of its multilayer circuits have positive weights.

Iterative Convergence Theorem

Theorem

In a free multilayer network $\mathcal{N}$ we have:

$$\mathbf{A}^* = \mathbf{A}^{(k)} = \mathbf{I} \oplus \mathbf{A} \oplus \mathbf{A}^2 \oplus \cdots \oplus \mathbf{A}^k$$

Where k is the maximum length of the multilayer elementary paths in $\mathcal{N}$, and it is equal to $2^{(\lambda+1)\lambda^2 n^2} - 1$.

Iterative Convergence Theorem

Theorem

In a free multilayer network $\mathcal{N}$ we have:

$$\mathbf{A}^* = \mathbf{A}^{(k)} = \mathbf{I} \oplus \mathbf{A} \oplus \mathbf{A}^2 \oplus \cdots \oplus \mathbf{A}^k$$

Where k is the maximum length of the multilayer elementary paths in $\mathcal{N}$, and it is equal to $2^{(\lambda+1)\lambda^2 n^2} - 1$.

Proposition [M. L. Lamali, S. Lassourreuille, S. Kunne, and J. Cohen, 19]

For any multilayer network $\mathcal{N}$, the valid shortest path (if any) between two nodes is upper bounded by $2^{(\lambda+1)\lambda^2 n^2}$.

Conclusion and Future Work

- New routing algebra based on semirings for path computation with automatic tunneling: an isotonic and non-monotonic algebra with a partial order.
- A generalization of the iterative convergence theorem for the optimal solution of the valid shortest paths problem.

Conclusion and Future Work

- New routing algebra based on semirings for path computation with automatic tunneling: an isotonic and non-monotonic algebra with a partial order.
- A generalization of the iterative convergence theorem for the optimal solution of the valid shortest paths problem.
- The adaptation of the other existing algebraic structures (algebra of endomorphisms and Sobrinho's algebra) to the valid shortest paths problem.
- The generalization of the asynchronous convergence theorem for the stack-vector protocol.

References

Mehryar Mohri.
Semiring frameworks and algorithms for shortest-distance problems.
J. Autom. Lang. Comb., 7(3):321–350, 2002.

Timothy G. Griffin and João Luís Sobrinho.
Metarouting.
In *Proc. of ACM SIGCOMM*, pages 1–12, 2005.

Matthew L. Daggitt, Alexander J. T. Gurney, and Timothy G. Griffin.
Asynchronous convergence of policy-rich distributed bellman-ford routing protocols.
In *Proc. of ACM SIGCOMM*, pages 103–116, 2018.

Fernando Kuipers and Freek Dijkstra.
Path selection in multi-layer networks.
Comput. Commun., 32(1):78–85, 2009.

Mohamed Lamine Lamali, Hélia Pouyllau, and Dominique Barth.
Path computation in multi-layer multi-domain networks: A language theoretic approach.
Computer Communications, 36(5):589–599, 2013.

M. L. Lamali, S. Lassourreuille, S. Kunne, and J. Cohen.
A stack-vector routing protocol for automatic tunneling.
In *IEEE INFOCOM*, pages 1675–1683, 2019.

Noureddine Mouhoub, Mohamed Lamine Lamali, and Damien Magoni.
A highly parallelizable algorithm for routing with automatic tunneling.
In *2022 IFIP Networking Conference (IFIP Networking)*, pages 1–9, 2022.