



HAL
open science

Metarouting with Automatic Tunneling in Multilayer Networks

Noureddine Mouhoub, Maria Moloney, Damien Magoni

► **To cite this version:**

Noureddine Mouhoub, Maria Moloney, Damien Magoni. Metarouting with Automatic Tunneling in Multilayer Networks. 2023. <hal-04129525>

HAL Id: hal-04129525

<https://hal.science/hal-04129525v1>

Preprint submitted on 15 Jun 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Copyright - All rights reserved

Metarouting with Automatic Tunneling in Multilayer Networks

Noureddine Mouhoub¹ Maria Moloney² Damien Magoni¹

¹ *LaBRI - CNRS, University of Bordeaux, Talence, France*

² *PEL, University College Dublin, Dublin, Ireland*

Metarouting allows for the modelling of routing protocols using an algebraic structure called routing algebra. Routing protocols requiring design or validation can, in fact, be easily modelled using this approach. To date, however, existing research on routing algebras has mainly focussed on applying this approach to routing protocols that are generally used in networks which have a single addressing and forwarding protocol. The basic algebraic structures used in such contexts are semirings, Sobrinho's algebras and algebras of endomorphisms. In this paper, we propose the modification of these existing routing algebras to deal with networks that contain multiple forwarding protocols where tunnels are omnipresent. To achieve this, we define new algebraic structures derived from the three aforementioned ones, in order to model the generalized routing problem with automatic tunneling entitled *valid paths* algebra. All of our routing algebras are defined as semi-direct products of two structures, the well-known *shortest paths* algebra and the proposed *valid paths* algebra. These new algebras are isotonic and non-monotonic with a partial order. We propose a fixed point for those new algebras and we prove the iterative convergence to the optimal solution of the *valid shortest paths* problem.

Keywords: Metarouting, routing algebra, partial order, multilayer network, automatic tunneling, path computation

1 Introduction

Routing algebras for classical routing protocols can be dated as far back as 1971. It was not until 2003, however, that they were studied in the context of Internet routing protocols such as BGP. In fact, the problem of automatic tunneling management for multilayer network routing has itself only been tackled since 2009 and has since attracted increasing attention due the advent of IPv6 and the deployment of VPNs. From this perspective, leveraging routing algebras to derive properties on multilayer networks becomes compelling. Our aim in this paper is to define such algebras and prove some properties related to network routing with automatic tunneling. Our contribution can be summarized as follow:

- We define three routing algebras for performing path computation with tunnels: one as a semiring, one as an algebra of functions, and one as a Sobrinho's algebra.
- We show that these routing algebras are isotonic and non-monotonic with a partial order.
- We propose a fixed point for these new algebras.

- Finally, we prove the iterative convergence to the optimal solution of the *valid shortest paths* problem *vsp*.

This paper is an extended version of a previous short paper by Mouhoub et al. (2022a) presented at the Workshop on New IP and Beyond held in conjunction with the International Conference on Network Protocols. The short paper only defined the semiring algebra and did not include any proofs. In this extended version, we propose and define two more routing algebras, as well as proving that all these algebras are isotonic and not monotonic. Finally, we prove the convergence to an optimal solution of the length of elementary multilayer paths in any multilayer network.

This paper is organized as follows: Section 2 reviews research on routing algebras and multilayer routing. Section 3 presents the algebraic foundations used throughout the paper, while Section 4 details the multilayer network model on which we extend our routing algebras. This is followed in Section 5 by three definitions for algebraic structures (semiring, algebra of functions, and algebra of Sobrinho) needed for the valid path problem. Finally, we prove a theorem in Section 6 which provides an upper bound on the length of every valid shortest path in a free multilayer network. Section 7 concludes the paper.

2 Related Work

2.1 Routing Algebras

Routing algebra was initially studied by Carré (1971). The author proposed an algebraic structure based on semirings for the formulation of the path computation problem. He also proposed generic solutions that were based on classical methods of linear algebra such as the Jacobi and Gauss-Seidel method, and the Gauss and Jordan elimination method. Several algebraic structures and algorithms have been described by Carré (1979); Hebisch and Weinert (1998); Gondran and Minoux (1995); Minoux (2001) to solve various optimization and routing problems in networks. Mohri (2002) proposed new algebraic frameworks and generic algorithms for the shortest path problem and the k -shortest paths problem.

More recently, work on routing algebras has been applied to existing routing protocols. A first application of this was on path vector routing protocols proposed by Sobrinho (2003, 2005). Not only did the authors propose a sufficient condition to guarantee the correctness of the iBGP protocol, they also showed that if the algebra is *monotonic*⁽ⁱ⁾ then the protocol converges in any network, but not necessarily to a global optimal solution. If the algebra is *isotonic*, however, then the protocol converges to a global optimal solution.

Griffin and Sobrinho (2005b) proposed an approach called metarouting which is based on Sobrinho's algebra and used to define routing protocols in a declarative and high-level way. They also introduced a "scoped" product to model the combined metric of iBGP and eBGP. Griffin and Sobrinho (2005b); Gurney and Griffin (2007); Dyerowicz and Griffin (2013) proposed the lexicographic product to combine several QoS metrics into a single composite metric. They also showed which properties are required to guarantee optimal global and local solutions in the case of lexicographic products. Another so-called functional product was used by Khayou and Sarakbi (2017) to model EIGRP non-lexical metrics.

An important work by Daggitt et al. (2018a) showed that Sobrinho's conditions are sufficient for a routing protocol to converge to a single solution, particularly in the case of the asynchronous Bellman-Ford

⁽ⁱ⁾ Sobrinho and Ferreira (2020) use the term *inflation*, while Daggitt et al. (2018a) use the term *increasing* to refer to the monotonicity property defined by Sobrinho (2005). In this paper, we use the term *monotonicity*.

algorithm. More recently, Sobrinho and Ferreira (2020) studied the problem of routing with non-isotonic metrics and proposed new routing protocols based on a partial order in order to preserve isotonicity.

2.2 Routing Algorithms with Automatic Tunneling

Kuipers and Dijkstra (2009) presented the first routing algorithm with automatic tunneling, which explored all possible paths using a breadth-first search and gave an exponential worst-case complexity. The related problem was shown to be NP – *hard* under bandwidth constraints. Later Lamali et al. (2013), proposed a polynomial algorithm based on a language theory approach. This approach was able to find the best path between two nodes without bandwidth constraints by minimizing either the number of hops or the number of adaptation functions. A matrix model for finding shortest paths with a length of at most k was proposed by Iqbal et al. (2015). However the related algorithms had an exponential complexity and could not compute paths containing cycles.

In 2016, an algorithm for shortest path computation based on any additive metric chosen by the user (*e.g.*, weighted links, weighted adaptation functions, etc.) was proposed by Lamali et al. (2016) as well as heuristics to compute them under bandwidth constraints. This paper also described an exponential algorithm yielding the valid shortest path under various QoS constraints by generalizing the work of Van Mieghem and Kuipers (2004). The downside to these algorithms is that they are not distributed and do not build routing tables, only paths. Source routing could, nonetheless, leverage them.

The first distributed algorithm based on a generalization of the Bellman-Ford algorithm was then proposed by Lamali et al. (2019). This algorithm propagates the protocol stack in addition to the distance. A maximum protocol stack height is enforced to ensure termination in the face of cycles. This algorithm is able to construct routing tables and is therefore suitable for hop-by-hop routing. The path computation process automatically determines tunnel endpoints in valid paths and installs them in the routing tables. Another algorithm for routing with automatic tunneling was proposed by Mouhoub et al. (2022c,b). The main idea of this algorithm is to construct routing tables by using a transitive closure operation. It is a generalization of the Floyd-Warshall algorithm in which the path concatenation depends on the compatibility of protocol stacks. As it requires shared memory, this algorithm can be used inside an SDN controller, which will then propagate the resulting routing tables to the SDN switches.

2.3 Routing Algebras with Tunnels

To date, the existing works on routing algebras and their applications, cited above, indicate that most research has focused on existing routing protocols used by networks with a single forwarding protocol (*e.g.*, IPv4). There exists, however, numerous under explored fields of application, such as path computation with protocol heterogeneity and automatic tunneling, which has not been thoroughly studied and, thus far, not fully understood. We aim to close this gap by studying the application of routing algebra in a multilayer model, where tunnels are omnipresent.

An example of such a network where IPv4 and IPv6 protocols coexist is shown in Figure 1. In the (top) path between nodes v_0 and v_5 , node v_1 encapsulates an IPv4 packet within an IPv6 packet, thus inducing a protocol stack, *i.e.*, the stack of headers (IPv4.IPv6) (from the bottom to the top of the stack). The node v_4 performs the reverse operation by decapsulating the inner IPv4 packet from the outer IPv6 one. The subpath from v_1 to v_4 is a tunnel. The whole path allows the communication from v_0 to v_5 . We call such a path a *valid path*. Another valid path is the (bottom) path, node v_2 converts an IPv4 packet into an IPv6 one. The node v_4 performs its second operation by transmitting, without any change, the IPv6 packet to

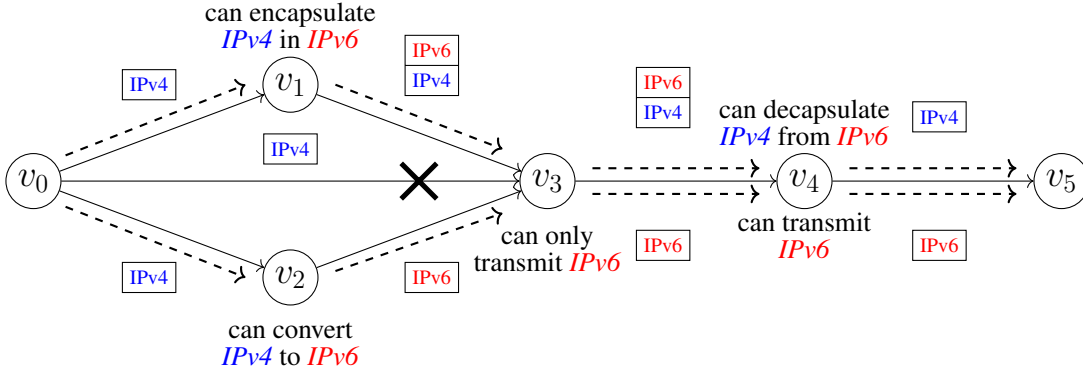


Fig. 1: Example of a network encompassing IPv4 and IPv6 protocol conversions and encapsulations.

the node v_5 . In the directed link from v_0 to v_3 , the node v_3 is not, however, able to transmit an IPv4 packet as is, v_5 cannot, thus, be reached from v_0 . We call such a path an *invalid path*.

As mentioned in the previous section, the only known algorithm for routing with automatic tunneling is the stack-vector algorithm proposed by Lamali et al. (2019). Based on this algorithm, our main idea is to define new algebraic structures based on: a semiring as described by Carré (1971); Mohri (2002); Dyerowicz and Griffin (2013), Sobrinho's algebra as proposed by Sobrinho (2003, 2005); Griffin and Sobrinho (2005b) and an algebra of functions⁽ⁱⁱ⁾ as proposed by Daggitt and Griffin (2018); Daggitt et al. (2018a). A quadrant model was proposed by Gurney and Griffin (2007) to summarize the different existing styles and their relations. These new structures must have the capacity to enumerate all sets of valid shortest paths between every pair of nodes in the network. These paths can be represented by pairs of protocol stacks (or a compositions of adaptation functions that make a path as valid) and weights.

In order to define these new structures, we aim to construct two operations on the *valid shortest paths* set. The first operation is used to concatenate valid shortest paths and the second one is used to choose between them. The concatenation operation must make it possible to calculate the weight sum of a path under the condition of preserving its validity. However, the second operation must keep all valid paths. In equality cases, *i.e.*, valid paths having the same protocol stack (tunnel), the second operation must keep the shortest one.

These proposed algebraic structures can be used to prove some convergence properties of the stack-vector protocol proposed by Lamali et al. (2019). They can also be used to find the optimal solution for networks running ECMP when using QoS metrics as shown by Dyerowicz and Griffin (2013).

3 Algebraic Foundations

We use the same algebraic structures as those defined by Mohri (2002); Sobrinho (2005); Dyerowicz and Griffin (2013); Daggitt et al. (2018a). Table 1 summarizes the main algebraic axioms.

⁽ⁱⁱ⁾ An algebra of functions is a generalization of an algebra of monoid endomorphisms, see Minoux and Gondran (2008) for more information.

Tab. 1: Algebraic properties.

Property	Definition
Selectivity (\oplus)	$a \oplus b \in \{a, b\}$
Idempotence (\oplus)	$a \oplus a = a$
Commutativity (\oplus)	$a \oplus b = b \oplus a$
Associativity (\oplus)	$a \oplus (b \oplus c) = (a \oplus b) \oplus c$
Left-Distributivity (\oplus, \otimes)	$a \otimes (b \oplus c) = (a \otimes b) \oplus (a \otimes c)$
Right-Distributivity (\oplus, \otimes)	$(b \oplus c) \otimes a = (b \otimes a) \oplus (c \otimes a)$
Identity ($\oplus, \bar{0}$)	$a \oplus \bar{0} = \bar{0} \oplus a = a$
Annihilator ($\oplus, \bar{1}$)	$a \oplus \bar{1} = \bar{1} \oplus a = \bar{1}$

3.1 Semiring

Definition 1. A semiring is a structure $SR = (S, \oplus, \otimes, \bar{0}, \bar{1})$, such that:

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

The generalized shortest paths problem is modeled by the semiring, $SR_{sp} = (\mathbb{R}^\infty, \min, +, +\infty, 0)$ ⁽ⁱⁱⁱ⁾. In the idempotent semiring (\oplus is idempotent) the relation \leq_\oplus is a partial order over S where,

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

This order is total when the operation \oplus is selective.

Definition 2. Given a semiring $(S, \oplus, \otimes, \bar{0}, \bar{1})$, the semiring of $n \times n$ matrix is $(\mathbb{M}_n(S), \oplus, \otimes, \mathbf{N}, \mathbf{I})$, where:

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

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

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

⁽ⁱⁱⁱ⁾ The notation \mathbb{R}^∞ represents the set of real numbers plus the infinity elements $\{+\infty, -\infty\}$.

Given a semiring $(S, \oplus, \otimes, \bar{0}, \bar{1})$ and a directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with n nodes. We define a weight function on the links of \mathcal{G} , $\omega : \mathcal{E} \rightarrow S$ and we denote with ω_{ij} the weight of the link $e_{ij} = (v_i, v_j)$. Let $\mathbf{A} \in \mathbb{M}_n(S)$ the $n \times n$ weighted adjacency matrix of G such that,

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

For a path $p = v_0, v_1, v_2, \dots, v_{k-1}, v_k$, the weight $\omega(p)$ is,

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

We define recursively the power of a matrix $\mathbf{A} \in \mathbb{M}_n(S)$ with,

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

Let $\mathcal{P}_{i,j}^k$ be the set of all paths from node v_i to node v_j of size k . We denote by $\mathcal{P}_{i,j}^{(k)}$ the set of all paths from v_i to v_j of size at most k . We define the weight of an optimal path from v_i to v_j of size at most k as follows,

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

Where:

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

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

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

3.2 Algebra of Functions

Definition 3. An algebra of functions is a structure $AF = (S, \oplus, F, \bar{0}, \bar{1})$, where:

- \oplus is an associative, commutative and selective binary operation over S
- F is a set of functions^(iv) from S to S
- $\bar{0}$ is a fixed point for all $f \in F$, $f(\bar{0}) = \bar{0}$
- $\bar{0}$ is an identity of \oplus and $\bar{1}$ is its annihilator

^(iv) In the case of an arbitrary operation \otimes , we can define the set of functions F by $F_{\otimes} = \{f_a(b) = a \otimes b \mid a \in S\}$.

The generalized shortest paths problem is modeled by the algebra of functions AF_{sp} defined by the tuple $AF_{sp} = (\mathbb{R}^\infty, \min, F_+, +\infty, -\infty)$. In the case where all the weights are positive, this structure is of the form $AF_{sp} = (\mathbb{R}_+^\infty, \min, F_+, +\infty, 0)$.

Given an algebra of functions $(S, \oplus, F, \bar{0}, \bar{1})$ and a directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with n nodes. We define a configuration of \mathcal{G} with a function from $\mathcal{E} \rightarrow F$. This function is represented by an $n \times n$ adjacency matrix \mathbf{F} with $\mathbf{F}_{i,j} \in F$. We define this adjacency matrix as follows:

$$\mathbf{F}_{i,j} = \begin{cases} f_{i,j} & \text{if } e_{i,j} \in \mathcal{E} \\ f_{\bar{1}} & \text{otherwise} \end{cases}$$

Where: $f_{i,j}$ represents the weight of the link $e_{i,j}$ and $f_{\bar{1}}$ represents the constant function for the missed links with $f_{\bar{1}}(a) = \bar{0}$ for all $a \in S$. We define the global computation by a matrix $\mathbf{A} \in \mathbb{M}_n(S)$ where $\mathbf{A}_{i,j}$ is the optimal weight from v_i to v_j . We define now the application of \mathbf{F} on \mathbf{A} with,

$$\mathbf{F}(\mathbf{A}_{i,j}) = \bigoplus_k \mathbf{F}_{i,k}(\mathbf{A}_{k,j})$$

3.3 Algebra of Sobrinho

Definition 4. A Sobrinho's algebra is modeled by a septet, $SA = (W, \leq, L, S, \bar{s}, \otimes, \omega)$, where:

- W is a set of weights
- \leq is a total order over W
- L is a set of labels
- S is a set of signatures with a special signature \bar{s}
- $\otimes : S \times L \rightarrow S$ is a binary operation that associates for a pair (signature, label) a signature
- $\omega : S \rightarrow W$ is a weight function that associates a signature to a weight
- $\forall l \in L, l \otimes \bar{s} = \bar{s}$ and $\forall s \in S - \{\bar{s}\}, \omega(s) \leq \omega(\bar{s})$

The generalized shortest paths problem is modeled by the Sobrinho's algebra SA_{sp} defined by the tuple $SA_{sp} = (\mathbb{R}^\infty, \leq, \mathbb{R}, \mathbb{R}^\infty, +\infty, +, id_{\mathbb{R}^\infty})$.

4 Multilayer Network Model

We use the same network model and definitions as those presented in Lamali et al. (2018, 2019) along with some additional definitions. Table 2 summarizes the main notations of the paper.

Tab. 2: Network model notations.

Notation	Description
\mathcal{N}	Multilayer network
\mathcal{A}	The set of all communication protocols
\mathcal{G}	A directed graph
\mathcal{V}	The set of all nodes
\mathcal{E}	The set of all links
\mathcal{H}	The set of protocol stacks
H	A set of protocol stacks in \mathcal{H}
\mathcal{F}	The set of all elementary adaptation functions
\mathcal{F}_{id}	The set of all passive adaptation functions
F	A set of adaptation functions in \mathcal{F}
$\hat{\mathcal{F}}$	The set of all adaptation functions closed under composition
\hat{F}	A set of adaptation functions in $\hat{\mathcal{F}}$
\odot	The composition operation of adaptation functions
\diamond	The concatenation operation of protocol stacks

4.1 Multilayer Network

Definition 5. A multilayer network is modeled by 4-tuple $\mathcal{N} = (\mathcal{G}, \mathcal{A}, \mathcal{F}, \omega)$ where:

- $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ is a directed graph modeling the network topology. The set of nodes \mathcal{V} models the routers of the network. The set of links \mathcal{E} models the physical links between the routers. The number of nodes is denoted by $|\mathcal{V}| = n$ and the number of links is denoted by $|\mathcal{E}| = m$.
- $\mathcal{A} = \{x, y, z, \dots\}$ is the set of protocols available in the network, but not necessarily at each router. The number of protocols is denoted by $|\mathcal{A}| = \lambda$. The set of protocols that a node v can receive is denoted by $In(v)$, and the set of protocols that v can send is denoted by $Out(v)$.
- \mathcal{F} is the set of elementary adaptation functions available in the network. For each node $v \in \mathcal{V}$, $\mathcal{F}(v)$ is the set of adaptation functions available on node v . There are three types of adaptation functions:
 - *Conversion*: the header of a packet of protocol x is transformed into the header of another protocol y . It is denoted by $(x \rightarrow y)$. If the received and emitted packets are of the same protocol, *i.e.*, a classical transmission, it is denoted by $(x \rightarrow x)$. Thus, we consider a classical transmission as a special type of conversion;
 - *Encapsulation*: the whole packet of protocol x is encapsulated in the data field of a packet of protocol y . It is denoted by $(x \rightarrow xy)$. Note that a packet can be encapsulated in another one of the same protocol, such as for IP-in-IP (*i.e.*, RFC 1853 and RFC 2003). In this case, the function is simply denoted by $(x \rightarrow xx)$;
 - *Decapsulation*: a packet of protocol x is extracted from the data field of a packet of protocol y , this is the reverse operation of $(x \rightarrow xy)$, and thus it is denoted by $(xy \rightarrow x)$. Note that

this operation can be performed only if the received packet of protocol y effectively contains a packet of protocol x in its data field.

- $\omega : \mathcal{V} \times \mathcal{F} \times \mathcal{V} \rightarrow \mathbb{R}$ is a weight function modeling any additive metric. The value of $\omega(v_i, f, v_j)$ is the weight of performing the adaptation function f on v_i and then transmitting the packet on the link (v_i, v_j) . Its sum over a path is the cost that we want to minimize. It is very generic as it can model the number of hops (by putting $\omega(v_i, f, v_j) = 1$ for all the possible triples), or the number of encapsulations (by putting $\omega(v_i, f, v_j) = 1$ when f is an encapsulation and $\omega(v_i, f, v_j) = 0$ otherwise), etc.

4.2 Protocol Stack

A sequence of adaptation functions induces a protocol stack. For example, the sequence:

$$(x \rightarrow y)(y \rightarrow yx)(x \rightarrow x)(x \rightarrow xx)(x \rightarrow xy)$$

induces the stack $yxxy$ (top on the right). The protocol at the top of the stack h is denoted by $Top(h)$.

Let f be an adaptation function and h a protocol stack. We denote by $f(h)$ the application of the adaptation function f to the stack h . For example, if f is the decapsulation ($xy \rightarrow x$) and $h = yxy$, then $f(h) = yx$. The outer protocol y is removed and the inner one (protocol x) is extracted, without any change to the underlying protocols that are still nested (here the bottom y).

Applying an adaptation function to a protocol stack is not always possible. For example, it is impossible to apply $f = (yx \rightarrow y)$ to stack $h = x$ since there is no nested (packet of) protocol y in h . In such cases, we denote the resulting stack by $f(h) = \phi$, where ϕ is called the *forbidden stack*. For any adaptation function f , we have $f(\phi) = \phi$. This means that there is an impossibility to apply an adaptation function at some point of a path and thus there is no way to continue through this path.

Definition 6. Let $f_i f_{i+1} \dots f_{j-1} f_j$ a sequence of adaptation functions and h_i an initial protocol stack. We define recursively the induced protocol stack h_k by,

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

More formally, let $\mathcal{H} = \{\phi, x, y, \dots, xx, yy, xy, yx, \dots\}$ be the set of all possible protocol stacks. Note that this set can be finite for a given multilayer network in which the number of protocols in any stack does not exceed λn^2 (the proof of this bound is given by Lamali et al. (2019)). We can define an elementary adaptation function as a function f from $H_a \rightarrow H_b$ where H_a (resp. H_b) is a set of protocol stacks in \mathcal{H} representing the domain (resp. co-domain) of f . For example, if the function is the decapsulation $xy \rightarrow x$, then $f : H_{xy} \rightarrow H_x$ where H_{xy} (resp. H_x) is the set of all protocol stacks starting with the sub protocol stack xy (resp. the protocol x). We define the application of a sequence of adaptation functions $f_i f_{i+1} \dots f_{j-1} f_j$ on a given protocol stack h by a composition of these adaptation functions.

Definition 7. Let $f : H_a \rightarrow H_b$ and $f' : H'_a \rightarrow H'_b$ be two elementary adaptation functions in \mathcal{F} . We define the composition by the new function $f'' = f' \odot f$ as follows:

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

Where:

$$H''_a = \{h \in H_a \mid f(h) \in H'_a\} \quad \text{and} \quad H''_b = \{f'(h) \mid h \in (H_b \cap H'_a)\}$$

Note that this composition \odot is associative and not commutative. The function $\{\phi\} \rightarrow \{\phi\}$ is denoted by the *forbidden function* $\phi \rightarrow \phi$. It is clear that the set of elementary adaptation functions \mathcal{F} is not closed under composition. For example, if the composition is $(x \rightarrow xy) \odot (y \rightarrow yx)$ then the composed function is $(y \rightarrow yxy)$. This new function is not in the set \mathcal{F} . For this, we define a new set $\hat{\mathcal{F}}$ of all adaptation functions closed under composition. Each composition of elementary adaptation functions can be represented by a new composed adaptation function. The impossibility of composition is represented by the *forbidden function* $\phi \rightarrow \phi$.

Let $\hat{\mathcal{F}} = \{\phi \rightarrow \phi, x \rightarrow x, x \rightarrow y, x \rightarrow xxx, xyx \rightarrow x, \dots\}$ be the set of all adaptation functions closed under composition. We say that two functions f' and f of $\hat{\mathcal{F}}$ are equal if and only if they have the same domain and co-domain. For example, the two functions $(x \rightarrow xx)$ and $(y \rightarrow x) \odot (x \rightarrow xy)$ are equal and they can apply on protocol stacks starting with x and give a protocol stacks starting with xx . In the following text, we denote by \mathcal{F}_{id} the set of all passive functions, *i.e.*, classical transmissions $\{x \rightarrow x, y \rightarrow y, \dots\}$ called *identity adaptation functions*.

4.3 Valid Path

A *multilayer path* is represented by a directed path in the underlined graph \mathcal{G} by taking into account the adaptation function involved at each node in the path. Thus, a path from a node v_i to a node v_j in \mathcal{N} is a mixed sequence of nodes and adaptation functions with starting and ending protocol stacks $h_i v_i f_i v_{i+1} f_{i+1} \dots v_{j-1} f_{j-1} h_j v_j$ where each v_k , $i \leq k \leq j$ is a node, and each f_k is an adaptation function. The starting stack is h_i and the ending stack is h_j .

Definition 8. 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 *valid* if and only if:

- The sequence $v_i v_{i+1} \dots v_{j-1} v_j$ is a classical path in \mathcal{G} and each $f_k \in \mathcal{F}(v_k)$;
- $h_i \neq \phi$ and $Top(h_i) \in In(v_i)$;
- $h_j \neq \phi$ and $Top(h_j) \in In(v_j)$;
- $h_j = f_{j-1}(\dots f_{i+1}(f_i(h_i)))$.

Note that a valid 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 can be represented by the valid composition of the elementary adaptation functions $f_{j-1} \odot \dots \odot f_{i-1} \odot f_i$. This latter composition is defined by the function $h_i \rightarrow h_j$ of the set $\hat{\mathcal{F}}$.

In Figure 2, the depicted path from v_0 to v_6 with the starting protocol stack yx is:

$$v_0(x \rightarrow y)v_1(y \rightarrow y)v_4(y \rightarrow y)v_5(xy \rightarrow x)v_6$$

It cannot be valid because, the node v_5 receives the protocol stack yy and it cannot decapsulate the protocol x from y , *i.e.*, it cannot perform the function $xy \rightarrow x$ on the stack yy . The composed function of this path is $\phi \rightarrow \phi$. In contrast, Figure 3 depicts a valid path v_0 to v_6 with the starting stack x and the arrival stack yx :

$$v_0(x \rightarrow y)v_1(y \rightarrow y)v_2(y \rightarrow yx)v_3(x \rightarrow xy)v_1(y \rightarrow y)v_4(y \rightarrow y)v_5(xy \rightarrow x)v_6$$

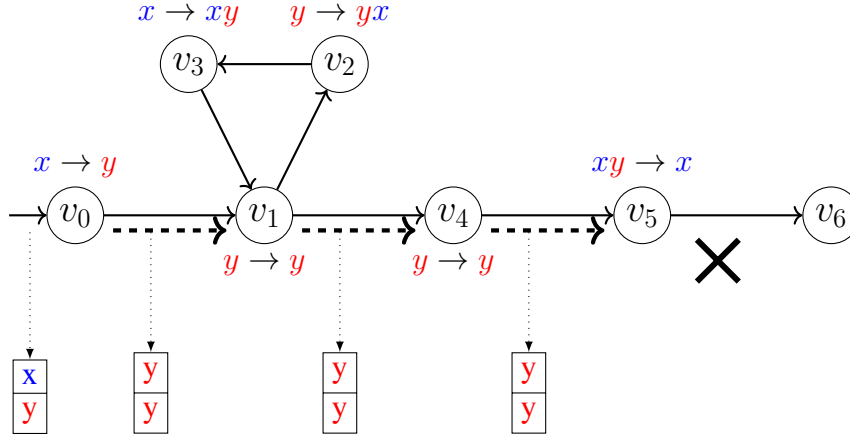


Fig. 2: An invalid multilayer path.

The composed function of this valid path is $x \rightarrow xy$. The used links in the path are in bold and the corresponding protocol stacks are below the links in Figures 2 and 3.

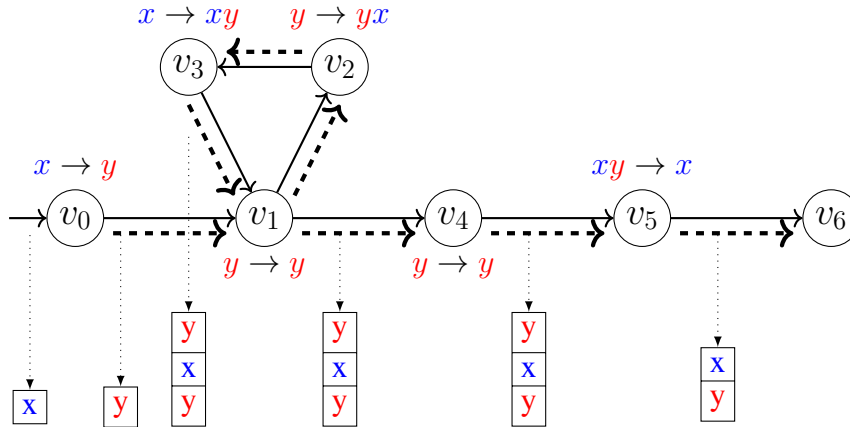


Fig. 3: A valid multilayer path with a circuit.

The weight of a valid path^(v) $p = h_i v_i f_i v_{i+1} f_{i+1} \dots v_{j-1} f_{j-1} h_j v_j$ from node v_i to node v_j with stacks h_i and h_j is the sum of the weights of its links and its adaptation functions. It is denoted by:

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

^(v) The weight of an invalid path is not defined and can be set to ∞ value.

5 Routing Algebras with Tunnels

In this section, we define three algebraic structures (semiring, Sobrinho's algebra, and algebra of functions) for the valid path problem vp . Then, we study the semi-direct product of the *valid paths* structures vp with the *shortest paths* structures sp in order to model the valid shortest paths vsp . Table 3 summarizes all *valid shortest paths* structures proposed in this paper.

5.1 Union-Min Operation

In order to enumerate the set of all valid shortest paths between each pair of nodes in a multilayer network, we introduce a semi-direct product over a set of weighted elements^(vi), in which the concatenation operation is a direct product and the choice operation is based on a new binary operation that combines the *union* and the *min* operations.

Let $S = \{a, b, c, \dots\}$ be a finite set of elements. We define the weighted set of S as the set product $S \times \mathbb{R}^\infty$. We denote $\mathcal{P}(S \times \mathbb{R}^\infty)$, the power set in which each subset contains a unique pair for each element. We define the *union-min* operation over subsets of weighted elements as follows:

Definition 9. Let S_1 and S_2 be two subsets of $\mathcal{P}(S \times \mathbb{R}^\infty)$. We define the *union-min* operation of S_1 and S_2 as follows:

$$S_1 \underset{min}{\cup} S_2 = \left\{ (a, \omega_a) \mid (a, \omega_a) \in S_i \wedge \forall (b, \omega_b) \in S_j, (a = b) \Rightarrow \omega_a = \min[\omega_a, \omega_b] \forall i, j \in \{1, 2\}; i \neq j \right\}$$

This new operation, *i.e.*, union-min, introduces the idea of enumerating different weighted paths from a source to a destination. And in the case of equality of paths (with the same elements), it keeps the path with the smallest weight. It is easy to check that the *union-min* operation is idempotent, associative and commutative, but not selective (it can return a new subset different from the two initial subsets). The identity element of this operation is the empty subset and the annihilator element is the subset of unique elements with weight $-\infty$, *i.e.*, $\{(a, -\infty), (b, -\infty), (c, -\infty), \dots\}$.

Definition 10. Let S_1 and S_2 be two subsets of $\mathcal{P}(S \times \mathbb{R}^\infty)$. We define the order relation as follows:

$$S_1 \subseteq_{min} S_2 \equiv \forall (a, \omega_a) \in S_1 \Rightarrow \exists (b, \omega_b) \in S_2, (a = b) \wedge (\omega_a \leq \omega_b)$$

It is clear that this order is a partial order, *i.e.*, there exists some incomparable subsets. For example, the two subsets $\{(a, 1), (b, 5)\}$ and $\{(a, 4), (b, 2)\}$ are incomparable.

In the next sections, we will use this operation to define our semi-direct product for semirings and algebras of functions. In the case of Sobrinho's algebra, the union-min operation is captured by the order relation.

5.2 Semiring with Tunnels

Recall that a valid multilayer path is represented by a valid composition of adaptation functions. In order to compute the set of all valid paths, *i.e.*, valid compositions, we extend the definition of adaptation function composition on sets of compositions.

^(vi) These elements can be the protocol stacks or the compositions of adaptation functions. It depends on the nature of the algebraic structure.

Let be $\hat{\mathcal{F}}$ the set of all adaptation functions closed under composition, and $\mathcal{P}(\hat{\mathcal{F}})$ its power set. If \hat{F}_1 and \hat{F}_2 are two subsets of $\mathcal{P}(\hat{\mathcal{F}})$, then we define the set of pair-wise compositions,

$$\hat{F}_1 \odot \hat{F}_2 = \{\hat{f}_1 \odot \hat{f}_2 \mid \hat{f}_1 \in \hat{F}_1 \text{ and } \hat{f}_2 \in \hat{F}_2\}$$

Note that in the case of invalid composition, the resulting forbidden function will be removed from the result set. The composition with an empty set is always an empty set. Based on this operation, we define our semiring valid paths SR_{vp} that enumerates all valid paths as follows,

$$SR_{vp} = \left(\mathcal{P}(\hat{\mathcal{F}}), \cup, \odot, \emptyset, \mathcal{F}_{id} \right)$$

Where, \emptyset is the empty set composition and \mathcal{F}_{id} is the set of identity adaptation functions $\{x \rightarrow x, y \rightarrow y, \dots\}$. It is easy to check that the composition \odot is associative and non-commutative with the \emptyset as annihilator. We check the identity set compositions \mathcal{F}_{id} and the distributivity of \odot over \cup ,

Identity of \odot

Let $\hat{F} = \{\hat{f}_i, \hat{f}_{i+1}, \dots, \hat{f}_{j-1}, \hat{f}_j\}$ be a set of composed functions in $\mathcal{P}(\hat{\mathcal{F}})$. We want to show that,

$$\hat{F} \odot \mathcal{F}_{id} = \mathcal{F}_{id} \odot \hat{F} = \hat{F} \quad \forall \hat{F} \in \mathcal{P}(\hat{\mathcal{F}})$$

Let $\hat{f}_k, i \leq k \leq j, \hat{f}_k : H_a \rightarrow H_b$ be a function in \hat{F} . There are two possible situations:

1. H_a is a set of protocol stacks starting with x . In this case, $\hat{f}_k \odot (x \rightarrow x) = \hat{f}_k$ and for all $f \in \mathcal{F}_{id} - \{(x \rightarrow x)\}$ we have $\hat{f}_k \odot f = (\phi \rightarrow \phi)$.
2. H_b is a set of protocol stacks starting with x . In this case, $(x \rightarrow x) \odot \hat{f}_k = \hat{f}_k$ and for all $f \in \mathcal{F}_{id} - \{(x \rightarrow x)\}$ we have $f \odot \hat{f}_k = (\phi \rightarrow \phi)$.

Note that, by the definition of \odot , the forbidden functions are removed from the result set.

Distributivity of \odot over \cup

Let $\hat{F}_1, \hat{F}_2, \hat{F}_3$ be three sets of composed functions in $\mathcal{P}(\hat{\mathcal{F}})$. By the definition of \odot over sets of composed functions, which compute the set of pair-wise compositions, we can see that,

$$\hat{F}_1 \odot (\hat{F}_2 \cup \hat{F}_3) = (\hat{F}_1 \odot \hat{F}_2) \cup (\hat{F}_1 \odot \hat{F}_3)$$

And,

$$(\hat{F}_2 \cup \hat{F}_3) \odot \hat{F}_1 = (\hat{F}_2 \odot \hat{F}_1) \cup (\hat{F}_3 \odot \hat{F}_1)$$

Now, we can define the *valid shortest paths* semiring SR_{vsp} by the following semi-direct product of SR_{vp} and SR_{sp} ,

$$SR_{vsp} = SR_{vp} \rtimes SR_{sp} = \left(\mathcal{P}(\hat{\mathcal{F}} \times \mathbb{R}^\infty), \underset{min}{\cup}, (\odot \times +), \emptyset, (\mathcal{F}_{id} \times 0) \right)$$

It is well known that the direct product of semigroups preserves the associativity property, the identity elements and the annihilators, as indicated by Harden (1949). In most situations where algebraic structures violate some axioms of semirings, they do not generally satisfy the distributivity of \otimes over \oplus . We check this property in order to ensure that our structure defines a semiring.

Distributivity of $(\odot \times +)$ over \bigcup_{min}

We want to show that for all $\hat{F}_1, \hat{F}_2, \hat{F}_3$ sets of weighted composed functions in $\mathcal{P}(\hat{\mathcal{F}} \times \mathbb{R}^\infty)$ we have $lhs = rhs$ where,

$$\begin{aligned} lhs &= \hat{F}_1 (\odot \times +) (\hat{F}_2 \bigcup_{min} \hat{F}_3) \\ rhs &= (\hat{F}_1 (\odot \times +) \hat{F}_2) \bigcup_{min} (\hat{F}_1 (\odot \times +) \hat{F}_3) \end{aligned}$$

Based on the definition of the union-min operation, we distinguish between the two following cases:

Case 1: The two sets \hat{F}_2 and \hat{F}_3 are strictly different and don't have any common composed function,

$$\hat{F}_2 \bigcup_{min} \hat{F}_3 = \hat{F}_2 \cup \hat{F}_3$$

In this case, we can see that,

$$lhs = rhs = (F_1 (\odot \times +) \hat{F}_2) \cup (F_1 (\odot \times +) \hat{F}_3)$$

Case 2: The two sets have some common composed functions,

$$\hat{F}_2 \bigcup_{min} \hat{F}_3 = \hat{F}_2^* \cup \hat{F}_3^*$$

Where: $\hat{F}_2 = \hat{F}_2^* \cup \hat{F}$ and $\hat{F}_3 = \hat{F}_3^* \cup \hat{F}'$ and \hat{F} (resp. \hat{F}') is the non empty set of the common non optimal composed functions of \hat{F}_2 (resp. \hat{F}_3). In this situation, we have,

$$\begin{aligned} \hat{F}_1 (\odot \times +) \hat{F}_2 &= (\hat{F}_1 (\odot \times +) \hat{F}_2^*) \cup (\hat{F}_1 (\odot \times +) \hat{F}) \\ \hat{F}_1 (\odot \times +) \hat{F}_3 &= (\hat{F}_1 (\odot \times +) \hat{F}_3^*) \cup (\hat{F}_1 (\odot \times +) \hat{F}') \end{aligned}$$

And,

$$\begin{aligned} (\hat{F}_1 (\odot \times +) \hat{F}_2^*) \bigcup_{min} (\hat{F}_1 (\odot \times +) \hat{F}') &= (\hat{F}_1 (\odot \times +) \hat{F}_2^*) \\ (\hat{F}_1 (\odot \times +) \hat{F}_3^*) \bigcup_{min} (\hat{F}_1 (\odot \times +) \hat{F}) &= (\hat{F}_1 (\odot \times +) \hat{F}_3^*) \end{aligned}$$

As we can see, the two sets \hat{F}_2^* and \hat{F}_3^* are different. So we can apply the case 1,

$$lhs = rhs = (F_1 (\odot \times +) \hat{F}_2^*) \cup (F_1 (\odot \times +) \hat{F}_3^*)$$

We can check the right distributivity in the same way.

Using this semiring, we can define the adjacency matrix \mathbf{A} of all valid paths where $\mathbf{A}_{i,j}$ is the set of valid paths from node i to node j represented by the corresponding compositions of adaptation functions and their weights involved in each path. Missing paths are simply represented by the empty set of compositions.

In order to explain how it works, we take the same network of the Figure 1 by adding weights on links. We also add the function $(x \rightarrow x)$ to model that node v_0 can emit packets of protocol x . The function

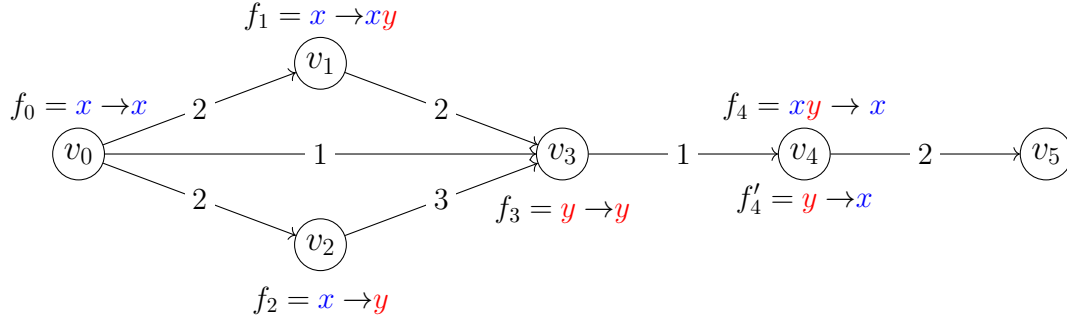


Fig. 4: Simple example network for valid shortest paths.

$(y \rightarrow y)$ will be replaced by the conversion $(y \rightarrow x)$ in order to create two equal compositions between nodes v_0 and v_5 . These modifications are presented in Figure 4.

We now define the adjacency matrix \mathbf{A} of the network presented in Figure 4,

$$\mathbf{A} = \begin{bmatrix} v_0 & v_1 & v_2 & v_3 & v_4 & v_5 \\ \emptyset & \{(f_0, 2)\} & \{(f_0, 2)\} & \{(f_0, 1)\} & \emptyset & \emptyset \\ & \emptyset & \emptyset & \{(f_1, 2)\} & \emptyset & \emptyset \\ & & \emptyset & \{(f_2, 3)\} & \emptyset & \emptyset \\ & & & \emptyset & \{(f_3, 1)\} & \emptyset \\ & & & & \emptyset & \{(f_4, 2), (f'_4, 2)\} \\ & & & & & \emptyset \end{bmatrix} \begin{matrix} v_0 \\ v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \end{matrix}$$

In this example, we are interested in computing the set of valid shortest paths, *i.e.*, the valid shortest compositions from node v_0 to node v_5 . Note that the intermediate paths will be obtained directly by calculating the \mathbf{A}^* matrix. For this, we use the formula defined in Section 3 to compute the \mathbf{A}^* matrix. The final result of the latter is shown in Figure 5.

$$\mathbf{A}^* = \begin{bmatrix} v_0 & v_1 & v_2 & v_3 & v_4 & v_5 \\ \mathcal{F}_{id} \times 0 & \{(f_0, 2)\} & \{(f_0, 2)\} & \{(f_0, 1), (f_1 \circ f_0, 4), (f_2 \circ f_0, 5)\} & \{(f_3 \circ f_1 \circ f_0, 5), (f_3 \circ f_2 \circ f_0, 6)\} & \{(f_4 \circ f_3 \circ f_1 \circ f_0, 7), (f'_4 \circ f_3 \circ f_1 \circ f_0, 7)\} \\ & \mathcal{F}_{id} \times 0 & \emptyset & \{(f_1, 2)\} & \{(f_3 \circ f_1, 3)\} & \{(f_4 \circ f_3 \circ f_1, 5), (f'_4 \circ f_3 \circ f_1, 5)\} \\ & & \mathcal{F}_{id} \times 0 & \{(f_2, 3)\} & \{(f_3 \circ f_2, 4)\} & \{(f_4 \circ f_3 \circ f_2, 6), (f'_4 \circ f_3 \circ f_2, 6)\} \\ & & & \mathcal{F}_{id} \times 0 & \{(f_3, 1)\} & \{(f_4 \circ f_3, 3), (f'_4 \circ f_3, 3)\} \\ & & & & \mathcal{F}_{id} \times 0 & \{(f_4, 2), (f'_4, 2)\} \\ & & & & & \mathcal{F}_{id} \times 0 \end{bmatrix} \begin{matrix} v_0 \\ v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \end{matrix}$$

 Fig. 5: \mathbf{A}^* matrix with all valid shortest paths of the network presented in Figure 4

5.3 Algebra of Functions with Tunnels

5.3.1 Algebra of Functions over Adaptation Functions

In order to define an algebra of functions in which the valid shortest paths are represented by the valid shortest compositions of adaptation functions, we can directly use the semiring SR_{vp} and get the following algebra,

$$AF_{vp} = (\mathcal{P}(\hat{\mathcal{F}}), \cup, \mathcal{P}(F_{\odot}), \emptyset, \hat{\mathcal{F}})$$

Now, we define the *valid shortest paths* algebra AF_{vsp} by the semi-direct product of AF_{vp} and AF_{sp} as follows,

$$AF_{vsp} = AF_{vp} \times AF_{sp} = (\mathcal{P}(\hat{\mathcal{F}} \times \mathbb{R}^{\infty}), \cup_{min}, \mathcal{P}(F_{\odot} \times F_{+}), \emptyset, (\hat{\mathcal{F}} \times -\infty))$$

Where: $F_{\odot} \times F_{+}$ is the set of functions that apply on weighted compositions defined by the product set,

$$\left\{ (\hat{f}_a, f_{\omega_a})(b, \omega_b) = (a \odot b, \omega_a + \omega_b), \mid b \in \hat{\mathcal{F}}, \omega_b \in \mathbb{R}^{\infty} \right\}$$

Note that applying a function $(\hat{f}_a, f_{\omega_a})$ on the empty set of weighted compositions is the empty set of weighted compositions. This defines a fixed point for all functions in the power set. The set of all compositions with weight $-\infty$ defines the annihilator set for the union-min operation.

5.3.2 Algebra of Functions over Protocol Stacks

We define here an algebra over protocol stacks. This algebra can generalize the idea of the Bellman-Ford algorithm with the automatic tunneling proposed by Lamali et al. (2019) in the stack-vector protocol.

Recall that our elementary adaptation functions are defined from $\mathcal{H} \rightarrow \mathcal{H}$, so we can define an algebra of functions on the set of all protocol stacks \mathcal{H} . For this, we define the application of a set of adaptation functions on a set of protocol stacks.

Let $F = \{f_1, \dots, f_{\ell}\}$ be a set of adaptation functions in $\mathcal{P}(\mathcal{F})$ and $H = \{h_1, \dots, h_k\}$ a set of protocol stacks in $\mathcal{P}(\mathcal{H})$. We define the application of F on H ,

$$F(H) = \left\{ f_i(h_j) \mid f_i \in F \text{ and } h_j \in H \right\}$$

In particular, if the set H or F is the empty set, then $F(H)$ is also the empty set. We define the AF_{vp} algebra,

$$AF_{vp} = (\mathcal{P}(\mathcal{H}), \cup, \mathcal{P}(\mathcal{F}), \emptyset, \mathcal{H})$$

Using the semi-direct product with the AF_{sp} we obtain,

$$AF_{vsp} = (\mathcal{P}(\mathcal{H} \times \mathbb{R}^{\infty}), \cup_{min}, \mathcal{P}(\mathcal{F} \times F_{+}), \emptyset, (\mathcal{H} \times -\infty))$$

Where $\mathcal{F} \times F_{+}$ is the set of functions that apply on weighted protocol stacks defined by the product set,

$$\left\{ (f, f_{\omega})(h, \omega_h) = (f(h), \omega + \omega_h), \mid h \in \mathcal{H}, \omega_h \in \mathbb{R}^{\infty} \right\}$$

Note that applying a function (\hat{f}, f_ω) on the empty set of weighted protocol stacks is the empty set of weighted protocol stacks. This defines a fixed point for all functions in the power set. The set of all protocol stacks with weight ∞ defines the annihilator set for the union-min operation.

Using this algebra, we can define the adjacency matrix \mathbf{F} of adaptation functions, where $\mathbf{F}_{i,j}$ is the set of adaptation functions of the node i with the set of corresponding weight functions. Missing links are represented by the empty set of pair-wise functions. We can define the routing state by the matrix \mathbf{A} where $\mathbf{A}_{i,j}$ is the set of all possible shortest protocol stacks that allow to route from node i to j .

Another way to define an algebra over protocol stacks is based on the idea of the generalized Floyd-Warshall algorithm for automatic tunneling proposed by Mouhoub et al. (2022c). This algorithm is based on the concatenation operation of protocol stacks' pairs.

Let $p = h_0 v_0 f_0 \dots h_k v_k$ and $p' = h'_0 v'_0 f'_0 \dots h'_k v'_k$ be two valid paths. We define the concatenation of p and p' by the new path $p'' = h''_0 v_0 f_0 \dots v_k f_k v'_0 f'_0 \dots h''_k v'_k$ where:

$$(h''_0, h''_k) = \begin{cases} (h_0, h'_k) & \text{if } h_k = h'_0 \\ (h.h_0, h'_k) & \text{if } h'_0 - h_k = h \\ (h_0, h.h'_k) & \text{if } h_k - h'_0 = h \\ (\phi, \phi) & \text{otherwise} \end{cases}$$

Note that $h.h'$ is the simple concatenation of the stack h and h' . For example, if $h = xy$ and $h' = x$ then $h.h' = xyx$.

Now we can define the binary operation \diamond that concatenates pairs of protocol stacks according to the previous conditions,

$$(h_i, h_j) \diamond (h'_i, h'_j) = (h''_i, h''_j)$$

Note that we can generalize this operation on sets of pairs of protocol stacks in order to compute the pair-wise sets. Using the previous definition, we define the following algebra,

$$AF_{vp} = \left(\mathcal{P}(\mathcal{H}^2), \cup, \mathcal{P}(F_\diamond), \emptyset, \mathcal{H}^2 \right)$$

Using the semi-direct product with the AF_{sp} we obtain,

$$AF_{vsp} = \left(\mathcal{P}(\mathcal{H}^2 \times \mathbb{R}^\infty), \underset{min}{\cup}, \mathcal{P}(F_\diamond \times F_+), \emptyset, (\mathcal{H}^2 \times -\infty) \right)$$

Where: $F_\diamond \times F_+$ is the set of functions that apply on weighted pairs of protocol stacks defined by the product set,

$$\left\{ (f_h, f_\omega)(h, \omega) = (h \diamond h', \omega + \omega'_h), \mid h' \in \mathcal{H}^2, \omega'_h \in \mathbb{R}^\infty \right\}$$

Note that applying a function (f_h, f_ω) on the empty set of weighted pairs of protocol stacks yields the empty set. This defines a fixed point for all functions in the power set. The set of all pairs of protocol stacks with weight $-\infty$ defines the annihilator set for the union-min operation.

The most common violation among all of our algebras of functions is that they lack selectivity, which is related to the nature of the union-min operation.

5.4 Sobrinho's Algebra with Tunnels

5.4.1 Sobrinho's Algebra over Adaptation Functions

We define the following Sobrinho's algebra for the valid paths problem,

$$SA_{vp} = \left(\mathcal{P}(\hat{\mathcal{F}}), \subseteq, \mathcal{P}(\mathcal{F}), \mathcal{P}(\hat{\mathcal{F}}), \emptyset, \odot, \text{id}_{\mathcal{P}(\hat{\mathcal{F}})} \right)$$

The signatures are the sets of valid compositions of adaptation functions and the labels are defined as sets of adaptation functions. The extension operation is the composition of sets of compositions. The special signature is the empty set of compositions and the weight function is the identity function of the power set of compositions. The order relation is given by the partial order \subseteq defined over subsets of compositions. Note that the two sets $\{x \rightarrow x\}$ and $\{x \rightarrow y\}$ are incomparable. Using this algebra we can define the *valid shortest paths* algebra SA_{vsp} as the directed product of SA_{vp} and SA_{sp} ,

$$SA_{vsp} = \left(\mathcal{P}(\hat{\mathcal{F}} \times \mathbb{R}^\infty), \subseteq_{min}, \mathcal{P}(\mathcal{F} \times \mathbb{R}), \mathcal{P}(\hat{\mathcal{F}} \times \mathbb{R}^\infty), \emptyset, (\odot \times +), \text{id}_{\mathcal{P}(\hat{\mathcal{F}} \times \mathbb{R}^\infty)} \right)$$

The set of signatures is the power set of all weighted composition functions and the set of labels is the power set of all weighted adaptation functions. Note that the order relation is defined on sets of weighted compositions. This order relation is the same as the partial order defined in Section 5.1 for the union-min operation.

Another idea is to define directly a Sobrinho's algebra with a specific weight function ω that maps a set of compositions to a set of weights. We define this algebra as follows,

$$SA_{vsp} = \left(\mathcal{P}(\mathbb{R}^\infty), \subseteq, \mathcal{P}(\mathcal{F}), \mathcal{P}(\hat{\mathcal{F}}), \emptyset, \odot, \omega \right)$$

Where: $\omega : \mathcal{P}(\hat{\mathcal{F}}) \rightarrow \mathcal{P}(\mathbb{R}^\infty)$ is the weight function.

5.4.2 Sobrinho's Algebra over Protocol Stacks

Recall that the application of adaptation functions on a set of protocol stacks gives a new set of protocol stacks. This application can be used to define a Sobrinho's algebra over sets of protocol stacks with a specific operation to model the application of adaptation functions on protocol stacks. We define this algebra as follows,

$$SA_{vp} = \left(\mathcal{P}(\mathcal{H}), \subseteq, \mathcal{P}(\mathcal{F}), \mathcal{P}(\mathcal{H}), \emptyset, \otimes, \text{id}_{\mathcal{P}(\mathcal{H})} \right)$$

The operation $\otimes : \mathcal{P}(\mathcal{H}) \times \mathcal{P}(\mathcal{F}) \rightarrow \mathcal{P}(\mathcal{H})$ is the binary operation that models the application of a set of adaptation functions on a set of protocol stacks. The weight function ω is the identity function of the power set of all protocol stacks. Using this algebra, we define the *valid shortest paths* algebra by the following direct product,

$$SA_{vsp} = \left(\mathcal{P}(\mathcal{H} \times \mathbb{R}^\infty), \subseteq_{min}, \mathcal{P}(\mathcal{F} \times \mathbb{R}), \mathcal{P}(\mathcal{H} \times \mathbb{R}^\infty), \emptyset, (\otimes \times +), \text{id}_{\mathcal{P}(\mathcal{H} \times \mathbb{R}^\infty)} \right)$$

The set of signatures is the power set of all weighted protocol stacks and the set of labels is the power set of all weighted adaptation functions. Note that the order relation is defined by a partial order on the sets of weighted protocol stacks.

Tab. 3: Algebraic structures for the *valid shortest paths* problem.

Algebra Style	Over Adaptation Functions	Over Protocol Stacks
Semiring (SR_{vsp})	$(\mathcal{P}(\mathcal{F} \times \mathbb{R}^\infty), \cup, (\odot \times +), \emptyset, (\mathcal{F}_{id} \times 0))$	–
Algebra of Functions (AF_{vsp})	$(\mathcal{P}(\mathcal{F} \times \mathbb{R}^\infty), \cup, \mathcal{P}(F_\odot \times F_+), \emptyset, (\mathcal{F} \times \infty))$ –	$(\mathcal{P}(\mathcal{H} \times \mathbb{R}^\infty), \cup, \mathcal{P}(\mathcal{F} \times F_+), \emptyset, (\mathcal{H} \times \infty))$ $(\mathcal{P}(\mathcal{H}^2 \times \mathbb{R}^\infty), \cup, \mathcal{P}(F_\odot \times F_+), \emptyset, (\mathcal{H}^2 \times \infty))$
Sobrinho's Algebra (SA_{vsp})	$(\mathcal{P}(\mathcal{F} \times \mathbb{R}^\infty), \subseteq, \mathcal{P}(\mathcal{F} \times \mathbb{R}), \mathcal{P}(\mathcal{F} \times \mathbb{R}^\infty), \emptyset, (\odot \times +), \text{id}_{\mathcal{P}(\mathcal{F} \times \mathbb{R}^\infty)})$ $(\mathcal{P}(\mathbb{R}^\infty), \subseteq, \mathcal{P}(\mathcal{F}), \mathcal{P}(\mathcal{H}), \emptyset, \odot, \omega)$	$(\mathcal{P}(\mathcal{H} \times \mathbb{R}^\infty), \subseteq, \mathcal{P}(\mathcal{F} \times \mathbb{R}), \mathcal{P}(\mathcal{H} \times \mathbb{R}^\infty), \emptyset, (\odot \times +), \text{id}_{\mathcal{P}(\mathcal{H} \times \mathbb{R}^\infty)})$ $(\mathcal{P}(\mathbb{R}^\infty), \subseteq, \mathcal{P}(\mathcal{F}), \mathcal{P}(\mathcal{H}), \emptyset, \otimes, \omega)$
	–	$(\mathcal{P}(\mathcal{H}^2 \times \mathbb{R}^\infty), \subseteq, \mathcal{P}(\mathcal{H}^2 \times \mathbb{R}), \mathcal{P}(\mathcal{H}^2 \times \mathbb{R}^\infty), \emptyset, (\odot \times +), \text{id}_{\mathcal{P}(\mathcal{H}^2 \times \mathbb{R}^\infty)})$
	–	$(\mathcal{P}(\mathbb{R}^\infty), \subseteq, \mathcal{P}(\mathcal{H}^2), \mathcal{P}(\mathcal{H}^2), \emptyset, \diamond, \omega)$

Another possibility is to define directly a Sobrinho's algebra with a specific weight function that maps a set of protocol stacks to a set of weights. We define this algebra as follows,

$$SA_{vsp} = (\mathcal{P}(\mathbb{R}^\infty), \subseteq, \mathcal{P}(\mathcal{F}), \mathcal{P}(\mathcal{H}), \emptyset, \otimes, \omega)$$

Where: $\omega : \mathcal{P}(\mathcal{H}) \rightarrow \mathcal{P}(\mathbb{R}^\infty)$ is the weight function.

In Section 5.3.2, we have defined a binary concatenation operation of pairs of protocol stacks in order to model the transitive closure operation to compute valid paths by transitive closure. Using this operation, we can define a Sobrinho's algebra over the sets of all pairs of protocol stacks. This algebra is defined as follows,

$$SA_{vp} = (\mathcal{P}(\mathcal{H}^2), \subseteq, \mathcal{P}(\mathcal{H}^2), \mathcal{P}(\mathcal{H}^2), \emptyset, \diamond, \text{id}_{\mathcal{P}(\mathcal{H}^2)})$$

Now we define the *valid shortest paths* Sobrinho's algebra on the pairs of protocol stacks by the following direct product,

$$SA_{vsp} = (\mathcal{P}(\mathcal{H}^2 \times \mathbb{R}^\infty), \subseteq_{min}, \mathcal{P}(\mathcal{H}^2 \times \mathbb{R}), \mathcal{P}(\mathcal{H}^2 \times \mathbb{R}^\infty), \emptyset, (\diamond \times +), \text{id}_{\mathcal{P}(\mathcal{H}^2 \times \mathbb{R}^\infty)})$$

The set of signatures and the set of labels are the power set of all weighted pairs of protocol stacks and the order relation is defined by a partial order on the sets of weighted pairs of protocol stacks.

Another idea is to define directly a Sobrinho's algebra with a specific weight function that maps a set of pairs of protocol stacks to a set of weights. We define this algebra as follows,

$$SA_{vsp} = (\mathcal{P}(\mathbb{R}^\infty), \subseteq, \mathcal{P}(\mathcal{H}^2), \mathcal{P}(\mathcal{H}^2), \emptyset, \diamond, \omega)$$

Where: $\omega : \mathcal{P}(\mathcal{H}^2) \rightarrow \mathcal{P}(\mathbb{R}^\infty)$ is the weight function.

The most common violation among all of our Sobrinho's algebras is that the order relation is not total, as it is defined over subsets of the power set.

6 Convergence Properties

In this section, we show some convergence properties on the routing algebras proposed in the previous section. Then, we prove the iterative convergence of the \mathbf{A}^* matrix using a fixed point.

6.1 Monotonicity and Isotonicity

In general, the convergence of routing protocols, *e.g.*, distance-vector and path-vector protocols, is based on the two important properties of the corresponding routing algebras.

The first property, *i.e.*, monotonicity, guarantees that the routing protocol converges in any network, but not necessarily to a global optimal solution, as shown by Sobrinho (2005).

Definition 11. Let \otimes be the extension operation and the order relation \leq_{\oplus} . The operation \otimes is monotonic if and only if,

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

We can see that all the proposed algebras in this paper do not satisfy this property. The composition of two sets can yield a new set (possibly the empty set) which is incomparable to the first sets. And the application of a set of adaptation functions on a set of protocol stacks can give a new set of protocol stacks (possibly the empty set) which is incomparable to the first set. For this, we prove the following proposition,

Proposition 1. The direct product operator $(\odot \times +)$ over the power set $\mathcal{P}(\hat{\mathcal{F}} \times \mathbb{R}^{\infty})$ is not monotonic.

Proof: We prove the proposition by using the following counterexample of composition,

$$\{(x \rightarrow xyx, 3), (y \rightarrow x, 2)\} (\odot \times +) \{(x \rightarrow xy, 1)\} = \{(x \rightarrow xx, 3)\}$$

We can see that the two sets $\{(x \rightarrow xyx, 3), (y \rightarrow x, 2)\}$ and $\{(x \rightarrow xy, 1)\}$ are incomparable by using the partial order relation of the union-min operation defined in Section 5.1. \square

We can use the same proof in the case of algebra of functions. For example,

$$F(\{(x, 1), (y, 2)\}) = \{(z, 3)\}$$

Where: $F = \{(x \rightarrow z, \omega \mapsto \omega + 2)\} \in \mathcal{P}(\mathcal{F} \times F_+)$ is a set of a pair of adaptation function and a weight function. We can observe that this two sets of stacks are incomparable.

The second property, *i.e.*, isotonicity, guarantees that the routing protocol converges to a global optimal solution, as proved by Sobrinho (2005).

Definition 12. Let \otimes be the extension operation and the order relation \leq_{\oplus} . The operation \otimes is isotonic if and only if,

$$a \leq_{\oplus} b \implies a \otimes c \leq_{\oplus} b \otimes c \quad \forall a, b, c \in S$$

In order to show that our algebras are isotonic, we prove the following proposition for semirings with composition. The same proof can be adapted to the algebras over protocol stacks.

Proposition 2. The direct product operator $(\odot \times +)$ over the power set $\mathcal{P}(\hat{\mathcal{F}} \times \mathbb{R}^{\infty})$ is isotonic.

Proof: We prove the proposition by contradiction. Suppose there exists three sets of weighted composed functions \hat{F}_1 , \hat{F}_2 and \hat{F}_3 such that the following property is true,

$$\left(\hat{F}_1 \subseteq_{\min} \hat{F}_2\right) \wedge \left(\hat{F}_1(\odot \times +)\hat{F}_3 \not\subseteq_{\min} \hat{F}_2(\odot \times +)\hat{F}_3\right)$$

By the definition of the order relation of the union-min operation,

$$\hat{F}_1 \subseteq_{\min} \hat{F}_2 \equiv \exists \hat{F}, (\hat{F}_2 = \hat{F}_1^* \cup \hat{F}) \wedge (\hat{F}_1 \underset{\min}{\cup} \hat{F}_1^* = \hat{F}_1)$$

Now, we compute the composition of \hat{F}_2 and \hat{F}_3 ,

$$\hat{F}_2(\odot \times +)\hat{F}_3 = (\hat{F}_1^* \cup \hat{F})(\odot \times +)\hat{F}_3 = (\hat{F}_1^*(\odot \times +)\hat{F}_3) \cup (\hat{F}(\odot \times +)\hat{F}_3)$$

So, we can see that,

$$(\hat{F}_1(\odot \times +)\hat{F}_3) \subseteq_{\min} (\hat{F}_1^*(\odot \times +)\hat{F}_3) \cup (\hat{F}(\odot \times +)\hat{F}_3)$$

Hence, the sets \hat{F}_1 , \hat{F}_2 and \hat{F}_3 can't exist. \square

6.2 Iterative Convergence

It is known that in classical networks without absorbing circuits (*i.e.*, with only positive weights) called free networks, the global optimal solution \mathbf{A}^* converges to the matrix $\mathbf{A}^{(n-1)}$ where $n-1$ is the maximum length of an elementary path, as proven by Carré (1971). This means that circuits only increase the weight of paths and therefore will be ignored by the computation. In our case, it is not the same situation, and as we saw in the multilayer model, circuits are allowed and sometimes necessary for some paths *i.e.*, constructing the necessary protocol stack for the path to be valid. For this reason, we firstly define the multilayer circuit and the multilayer elementary path in the multilayer network model in order to generalize the convergence theorem cited above.

Definition 13. A multilayer path $p = h_i v_i f_i \dots v_{j-1} f_{j-1} h_j v_j$ is a multilayer circuit if and only if:

- The node v_i is the same node v_j , *i.e.*, $v_i = v_j$
- The two stacks h_i and h_j received by v_i (or v_j) are the same, *i.e.*, $h_i = h_j$

Using the above definition, we now define the multilayer elementary paths.

Definition 14. A multilayer elementary path is a multilayer path in which its circuits (if they exist) are non-multilayer circuits (*i.e.*, such circuits start and end with a different protocol stack).

Definition 15. A free multilayer network is a network in which all of its multilayer circuits have positive weights.

Lamali et al. (2018) proved some bounds on the *valid shortest paths* problem in a multilayer network modeled by a Push-Down Automaton (PDA). In this paper, we focus on the following proposition whose proof is given by Lamali et al. (2018). It is a proof similar to the one showing bounds on the length of the shortest word generated by a Context-Free Grammar (CFG) in Chomsky normal form as shown by Hopcroft and Ullman (1979):

Proposition 3. For any multilayer network \mathcal{N} , the valid shortest path (if any) between two nodes contains at most $O(2^t)$ links, where $t = (\lambda + 1)\lambda^2 n^2$.

Using this proposition, we prove the following theorem,

Theorem 1. In a free multilayer network \mathcal{N} we have:

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

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

Proof: Let \mathcal{N} be a free multilayer network with elementary paths of length at most k , $k \leq 2^{(\lambda+1)\lambda^2 n^2} - 1$. Let's suppose there is a integer $\ell > 0$, where $k + \ell$ is a fixed point for \mathbf{A}^* ,

$$\mathbf{A}^* = \mathbf{A}^{(k+\ell)} = \mathbf{I} \oplus \mathbf{A} \oplus \dots \oplus \mathbf{A}^k \oplus \dots \oplus \mathbf{A}^{k+\ell}$$

Let $\mathbf{A}_{i,j}^{(k+\ell)}$ be the set of valid weighted composed functions from node i to node j of size at most $k + \ell$, i.e., with at most $k + \ell$ adaptation functions. Let's suppose there exists a valid path represented by a weighted composed function $(\hat{f}, \omega) \in \mathbf{A}_{i,j}^p$ of p adaptation functions such that $k + 1 \leq p \leq k + \ell$.

We say that (\hat{f}, ω) is optimal iff one of the following is true:

- **Hypothesis 1:** Let's assume there is no weighted composed function $(\hat{f}', \omega') \in \mathbf{A}_{i,j}^q$ with $1 \leq q \leq k$ such that $\hat{f}' = \hat{f}$.

However, according to the Proposition 3, if there is a valid shortest path with at least $k + 1$ adaptation functions then there is another valid shortest path with at most k adaptation functions. More precisely, there exists a weighted composed function $(\hat{f}', \omega') \in \mathbf{A}_{i,j}^q$, such that $\hat{f}' = \hat{f}$ and $\omega' \leq \omega$. And by the definition of the operation \oplus (union-min), the composed function \hat{f} is ignored.

$$\{(\hat{f}, \omega), \dots\} \underset{\min}{\cup} \{(\hat{f}', \omega'), \dots\} = \{(\hat{f}', \omega'), \dots\}$$

Therefore, the valid path represented by \hat{f} is not elementary and it contains a multilayer circuit.

- **Hypothesis 2:** Let's assume that for all weighted composed functions $(\hat{f}', \omega') \in \mathbf{A}_{i,j}^q$ of q adaptation functions such that $1 \leq q \leq k$ and $\hat{f}' = \hat{f}$, we have $\omega' > \omega$.

However, according to the Proposition 3, if there is a valid shortest path with at least $k + 1$ adaptation functions then there is another valid shortest path with at most k adaptation functions. More precisely, there exists a weighted composed function $(\hat{f}', \omega') \in \mathbf{A}_{i,j}^q$, such that $\hat{f}' = \hat{f}$ and $\omega' \leq \omega$. This means that the composed function \hat{f} represents a valid path which contains at most one multilayer circuit having a negative weight. Therefore, the multilayer network \mathcal{N} is not free.

The above two hypotheses do not hold, thus (\hat{f}, ω) can not be optimal. Therefore, the maximum length k of elementary multilayer paths ($k \leq 2^{(\lambda+1)\lambda^2 n^2} - 1$) in a free multilayer network \mathcal{N} is a fixed point for the global optimal solution \mathbf{A}^* . \square

This proof is made in the case of the semiring with composition, however, the same proof can be adapted to the algebra of functions in which we replace the matrix product \mathbf{A}^k by the k applications of a set of functions \mathbf{F} on an initial matrix \mathbf{A} of protocol stacks.

7 Conclusion and Future Work

Routing algebras are powerful abstractions for studying complex structures such as multilayer networks. In this paper, we defined three routing algebras for path computation with tunnels: a semiring, an algebra of functions and a so-called Sobrinho's algebra. We showed that the proposed routing algebras are isotonic and non-monotonic with a partial order. We also proposed a fixed point for these algebras and we proved the iterative convergence to the optimal solution of the *valid shortest paths* problem *vsp*.

For future work, we plan to study the asynchronous convergence of the stack-vector protocol by applying, if possible, the asynchronous convergence theorem of Daggitt et al. (2018a) and implementing it in Agda with the library of Daggitt et al. (2018b). We also aim to use the lexicographic product defined by Gurney and Griffin (2007) and refined by Dyerowicz and Griffin (2013) to study various QoS metrics for Equal-Cost Multi-Path routing (ECMP), in the case of routing with automatic tunneling.

Acknowledgements

The authors would like to thank Géraud Sénizergues for his help and explanations regarding algebraic structures and axioms. This work was funded by The French National Research Agency - HÉRA project. Grant no.: ANR-18-CE25-0002.

References

- B. A. Carré. An Algebra for Network Routing Problems. *IMA Journal of Applied Mathematics*, 7(3): 273–294, 1971. doi: 10.1093/imamat/7.3.273.
- B. A. Carré. *Graphs and Networks*. Clarendon - Oxford University Press, 1979. ISBN 0-19-859615-4.
- M. L. Daggitt and T. G. Griffin. Rate of convergence of increasing path-vector routing protocols. In *Proceedings of the IEEE International Conference on Network Protocols (ICNP)*, pages 335–345, 2018.
- M. L. Daggitt, A. J. T. Gurney, and T. G. Griffin. Asynchronous convergence of policy-rich distributed bellman-ford routing protocols. In *Proceedings of the ACM Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications (SIGCOMM)*, page 103–116, 2018a. doi: 10.1145/3230543.3230561.
- M. L. Daggitt, R. Zimrod, and T. G. Griffin. Agda-routing for SIGCOMM 2018. <https://github.com/MatthewDaggitt/agda-routing/tree/sigcomm2018>, 2018b.
- S. Dyerowicz and T. G. Griffin. On the forwarding paths produced by internet routing algorithms. In *Proceedings of the IEEE International Conference on Network Protocols (ICNP)*, pages 1–10, 2013. doi: 10.1109/ICNP.2013.6733608.
- M. Gondran. Algèbre linéaire et cheminement dans un graphe. *RAIRO - Recherche Opérationnelle*, 9 (V1):77–99, 1975. URL http://www.numdam.org/item/RO_1975__9_1_77_0/.
- M. Gondran and M. Minoux. *Graphes et algorithmes*. Eyrolles, 1995.
- T. G. Griffin and J. L. Sobrinho. Metarouting. *ACM SIGCOMM Comput. Commun. Rev.*, 35(4):1–12, 2005a. doi: 10.1145/1090191.1080094.

- T. G. Griffin and J. L. Sobrinho. Metarouting. In *Proceedings of the ACM Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications (SIGCOMM)*, page 1–12, 2005b. doi: 10.1145/1080091.1080094.
- A. J. T. Gurney and T. G. Griffin. Lexicographic products in metarouting. In *Proceedings of the IEEE International Conference on Network Protocols (ICNP)*, pages 113–122, 2007. doi: 10.1109/ICNP.2007.4375842.
- J. C. Harden. Direct and semidirect products of semigroups. Master’s thesis, University of Tennessee, 1949.
- U. Hebisch and H. Weinert. *Semirings: Algebraic Theory and Applications in Computer Science*. World Scientific, 1998. ISBN 9789810236014.
- J. E. Hopcroft and J. D. Ullman. *Introduction to Automata Theory, Languages, and Computation*. Addison-Wesley Publishing Company, 1979.
- F. Iqbal, J. van der Ham, and F. Kuipers. Technology-aware multi-domain multi-layer routing. *Computer Communications*, 62(C):85–96, 2015. doi: 10.1016/j.comcom.2015.01.010.
- H. Khayou and B. Sarakbi. A validation model for non-lexical routing protocols. *J. Netw. Comput. Appl.*, 98(C):58–64, 2017. doi: 10.1016/j.jnca.2017.09.006.
- F. Kuipers and F. Dijkstra. Path selection in multi-layer networks. *Computer Communications*, 32(1): 78–85, 2009. doi: 10.1016/j.comcom.2008.09.026.
- M. L. Lamali, H. Pouyllau, and D. Barth. Path computation in multi-layer multi-domain networks: A language theoretic approach. *Computer Communications*, 36(5):589–599, 2013.
- M. L. Lamali, N. Fergani, J. Cohen, and H. Pouyllau. Path computation in multi-layer networks: Complexity and algorithms. In *Proceedings of the IEEE International Conference on Computer Communications (INFOCOM)*, 2016.
- M. L. Lamali, N. Fergani, and J. Cohen. Algorithmic and complexity aspects of path computation in multi-layer networks. *IEEE/ACM Transactions on Networking*, 26(6):2787–2800, 2018. doi: 10.1109/TNET.2018.2878103.
- M. L. Lamali, S. Lassourreuilie, S. Kunne, and J. Cohen. A stack-vector routing protocol for automatic tunneling. In *Proceedings of the IEEE International Conference on Computer Communications (INFOCOM)*, pages 1675–1683, 2019. doi: 10.1109/INFOCOM.2019.8737536.
- M. Minoux. *Graphes, Dioides et Semi-Anneaux: Nouveaux modèles et algorithmes*. TEC and DOC, 2001. URL <https://hal.archives-ouvertes.fr/hal-01571106>.
- M. Minoux and M. Gondran. *Graphs, Dioids and Semirings. New Models and Algorithms*, volume 41. Springer, 2008. doi: 10.1007/978-0-387-75450-5.
- M. Mohri. Semiring frameworks and algorithms for shortest-distance problems. *J. Autom. Lang. Comb.*, 7(3):321–350, 2002.

- N. Mouhoub, L. Lamali, and D. Magoni. Semiring Algebraic Structure for Metarouting with Automatic Tunneling. In *Proceedings of the IEEE International Workshop on New IP and Beyond (co-located with IEEE ICNP)*, pages 1–6, 2022a. doi: 10.1109/ICNP55882.2022.9940413.
- N. Mouhoub, M. L. Lamali, and D. Magoni. A highly parallelizable algorithm for routing with automatic tunneling. In *Proceedings of the IFIP Networking Conference*, pages 1–9, 2022b. doi: 10.23919/IFIPNetworking55013.2022.9829784.
- N. Mouhoub, M. L. Lamali, and D. Magoni. (Presque) Floyd-Warshall sous stéroïdes : (encore) un algorithme de routage pour l'établissement automatique de tunnels. In *24èmes Rencontres Francophones sur les Aspects Algorithmiques des Télécommunications*, 2022c.
- T. Saitô. Note on the lexicographic product of ordered semigroups. *Proceedings of the Japan Academy*, 46(5):413 – 416, 1970. doi: 10.3792/pja/1195520304.
- J. L. Sobrinho. Network routing with path vector protocols: Theory and applications. In *Proceedings of the ACM Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications (SIGCOMM)*, page 49–60, 2003. doi: 10.1145/863955.863963.
- J. L. Sobrinho. An algebraic theory of dynamic network routing. *IEEE/ACM Transactions on Networking*, 13(5):1160–1173, 2005. doi: 10.1109/TNET.2005.857111.
- J. L. Sobrinho and M. A. Ferreira. Routing on multiple optimality criteria. In *Proceedings of the ACM Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications (SIGCOMM)*, page 211–225, 2020. doi: 10.1145/3387514.3405864.
- A. Üresin and M. Dubois. Parallel asynchronous algorithms for discrete data. *J. of the ACM*, 37(3): 588–606, 1990. doi: 10.1145/79147.79162.
- P. Van Mieghem and F. Kuipers. Concepts of exact QoS routing algorithms. *IEEE/ACM Transactions on networking*, 12(5):851–864, 2004.