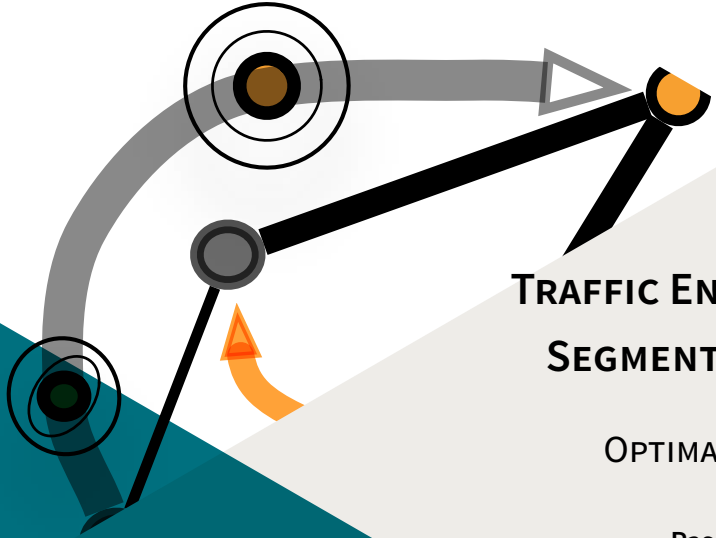
Université de Strasbourg



Traffic Engineering with Segment Routing (SR)

Optimally wrap SR!?

Pascal Mérindol

23/6/2023

A brief presentation: who is this guy?

My main Research Contributions

Université de Strasbourg

Two Main Research Topics to Improve the Internet

Both about Routing but from two distinct perspectives

1. **Routing Algorithms & Protocols**: **Compute and Deploy Valuable IP Routes**
   - Fast and correct convergence for several change models: link and node failures, maintenances or BGP hot potato
   - Multi-metric and multi-path for traffic engineering
2. **IP Measurements** in Transit Networks: **Topology Discovery & Large Scale Monitoring**
   - Reveal Hidden MPLS Tunnels & Forwarding Detours
   - Troubleshoot ISP (Internet Service Provider) Networks

A Short Introduction & Context

Internet: the Worldwide Distributed System

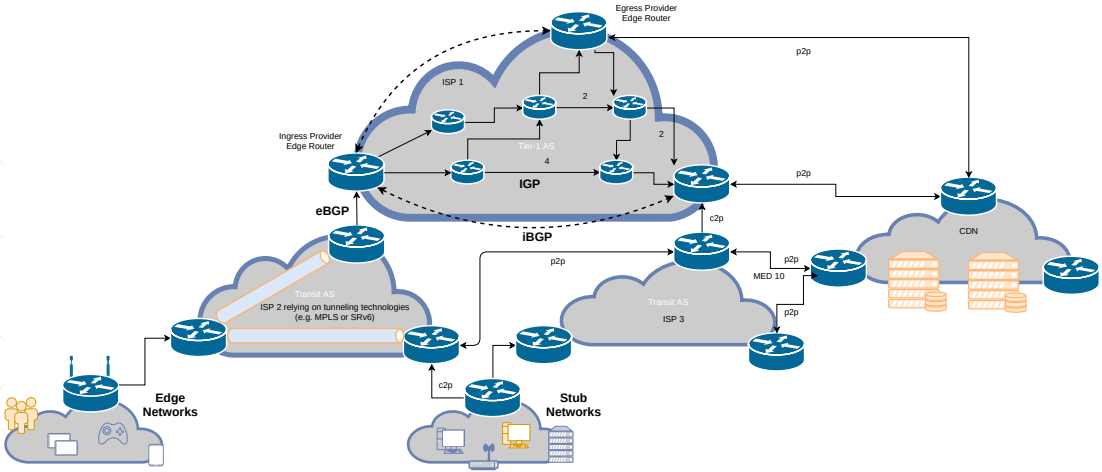
- **Large scale**: soon $100,000$ Domains (including $\approx [100...10,000]$ routers), $1M$ IP prefixes and much more end-devices than human beings!
- **Complex**: numerous heterogeneous hardware and software components

Internet: the Worldwide Distributed System

- **Large scale**: soon $100,000$ Domains (including $\approx [100 \ldots 10,000]$ routers), $1M$ IP prefixes and much more end-devices than human beings!
- **Complex**: numerous heterogeneous hardware and software components
- **Innovation is difficult and challenging**:
  - The universal common technology for connecting all devices lies in the **IP data-plane**
  - The **IP control-plane (i.e. the routing)** provides means to install and manage forwarding routes, e.g. with MPLS or Segment Routing (SR) w.r.t. the type of service

The Big Picture

Complex ecosystems



Two Scales and Several Technologies

Two scales of routing operations:

- inter-domain or AS-wide level: **BGP**
- intra-domain or router level within an AS: **IGP**

Two Scales and Several Technologies

Two scales of routing operations:

- inter-domain or AS-wide level: **BGP**
- intra-domain or router level within an AS: **IGP**

Several technologies and paradigms:

- **Hop-by-hop** packet forwarding (for best-effort traffic), or **loose source routing**, e.g. with **SR or MPLS tunnels**
- With packet encapsulation, one can enable **Traffic Engineering (TE)** for premium flows, or deploy **Load Balancing (LB) and Fast-ReRoute** (FRR) for all services. **A single deviation from the best-effort usual IP forwarding scheme** is often enough

Background: Internet Routing Protocols

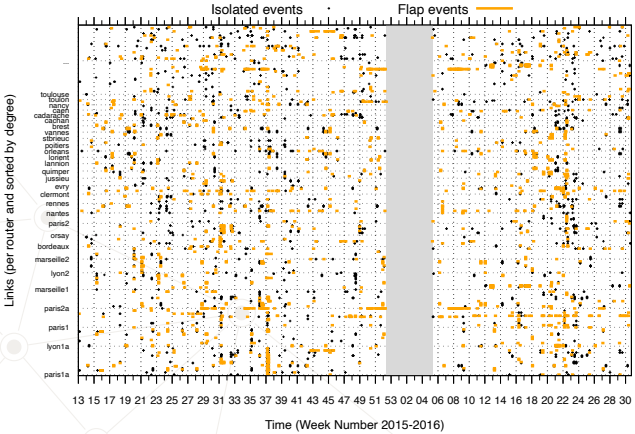
Two respective paradigms for inter- and intra-domain routing:

- with BGP, preferences are local and selfish: the path vector protocol may diverge
  - sufficient conditions for convergence a priori met in practice (*valley free* routes and usual economical incentives)
- within IGP link-state routing protocols (e.g. OSPF, IS-IS), the objective is globally consistent, typically a *minimum function* applied on an *additive metric*: $(min, +)$
  - convergence is granted (thanks to the *isotonicity* & *monotony* of routing operations), but **anomalies, e.g. forwarding loops and path sub-optimality,** still occur during transient periods of change

Routing Changes: Illustration in RENATER

Université de Strasbourg

The **DCART** project deployed in the French Educational & Research Network

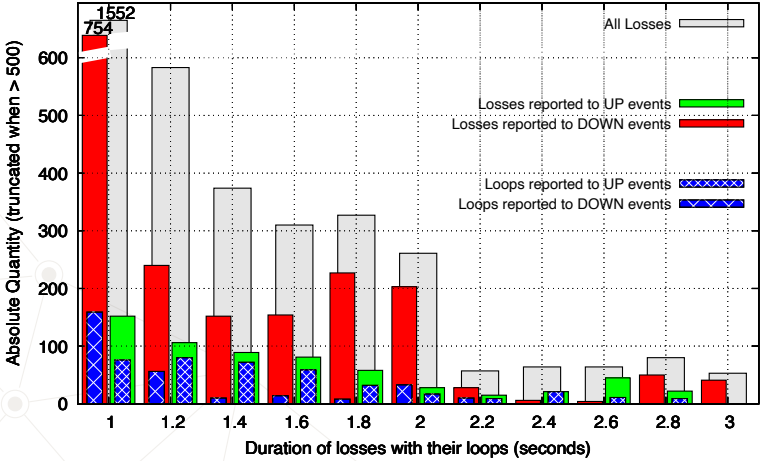


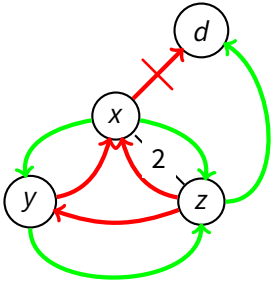
RENATER

GÉANT
Networks • Services • People

Topological changes are common and links are often **flapping**

Routing Changes: Illustration in RENATER





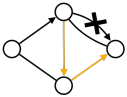
A not so simple ECMP forwarding loop

Forwarding loops do occur, in particular when routers reboot!

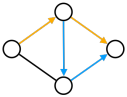
SR basics (I)

Use-cases & Context

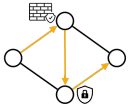
- Optimal Fast Re-Routing



- Load Balancing



- Service Chaining



Deployment of Delay-Constrained, Least-Cost paths (DCLC)

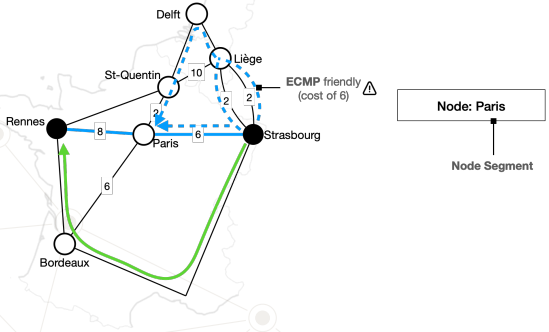
- Upper delay constraint
- Optimised IGP cost

MPLS is the legacy technology: as a data-plane layer, it is valuable, but its control-planes (e.g., with LDP or RSVP-TE) imply many forwarding states at routers :(

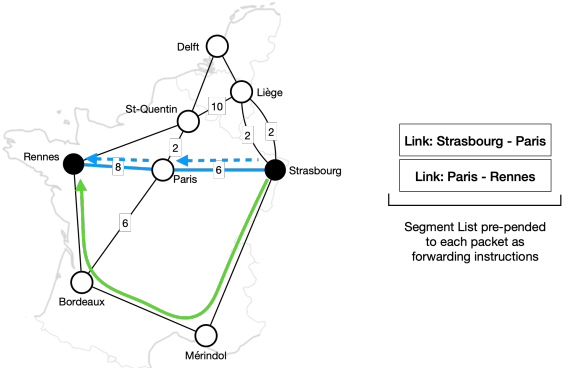
On the contrary, SR implies to **push a SR stack on packets at the source**, i.e. at the ingress PER.

SR basics (II)

Two kinds of Segments



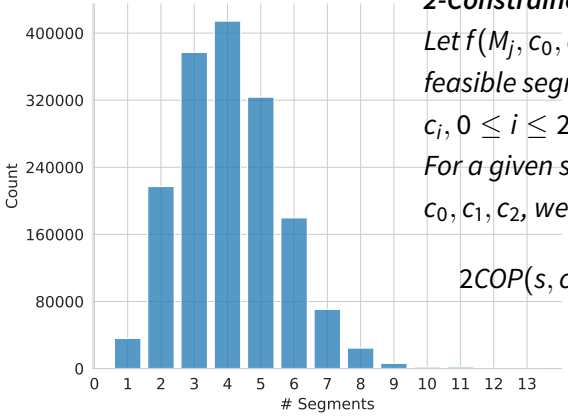
Node segment



Adjacency segment

Limitations: MSD, the maximal segment depth, that is the maximal number of segments (#seg, the SR operational metric) an ingress PER can append to each packet at line rate.

Pushing SR to its limit with DCLC



2-Constrained Optimal Paths (2COP)

Let $f(M_j, c_0, c_1, c_2, s, d)$ be a function returning a feasible segment path from s to d, verifying constraints $c_i, 0 \leq i \leq 2$ and optimizing $M_j, j \in 0, 1, 2$.

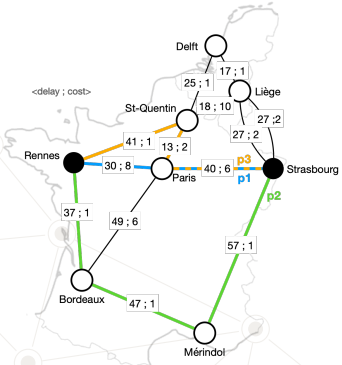
For a given source s and given upper constraints c_0, c_1, c_2, we have

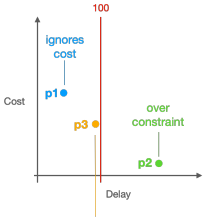
$$2COP(s, c_0, c_1, c_2) = \bigcup_{\substack{\forall d \in V, \\ \forall j \in \{0,1,2\}, \\ \forall c'_j \leq c_j}} f(M_j, c'_0, c'_1, c'_2, s, d)$$

With extreme use-case such as DCLC (2COP in particular), the number of required segments is not negligible: it can reach and even exceed the MSD limit!

2COP IS EASIER THAN YOU THINK BUT...

...#SEG IS NOT YET ANOTHER METRIC!

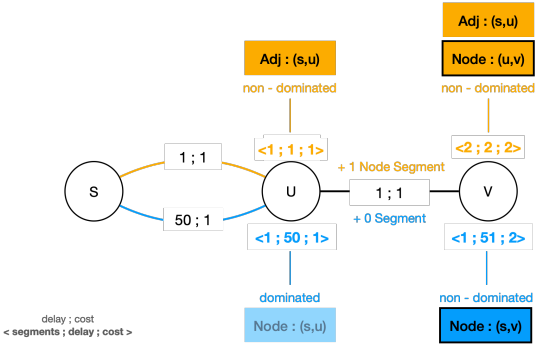




Link: Strasbourg - Paris

Node: Paris - Rennes

The Pareto Front can be discretized but...



...# seg is a *weird* metric!

Isotony (I)

A sub-path of a best path is also a best path!

More precisely, using a routing algebra formalism, let me denote $\mathcal{A} = \langle S, \oplus, \otimes, \mathcal{O} \rangle$ with $S = L \uplus \Sigma$ a non empty finite partitionned set with two operations:

$$\oplus : \Sigma \times \Sigma \longrightarrow \Sigma \qquad \otimes : L \times \Sigma \longrightarrow \Sigma$$

Where the first, $\oplus$, is the best path selector, and the second, $\otimes$, the path extension through an edge. The set Σ is about paths and L about edges. Finally, $\mathcal{O}$ depicts special paths when initially advertised and, with a total order, we can define $\preccurlyeq$ such as $\beta \preccurlyeq \alpha \Leftrightarrow \beta \oplus \alpha = \beta$. Then, formally, **isotony** is defined as follows:

$$\forall (\lambda, \sigma, \tau) \in L \times \Sigma^2 \ : \sigma \preccurlyeq \tau \Longrightarrow \lambda \otimes \sigma \preccurlyeq \lambda \otimes \tau$$

Or equivalently as:

$$\lambda \otimes (\sigma \oplus \tau) = (\lambda \otimes \sigma) \oplus (\lambda \otimes \tau)$$

Isotony (II)

Université de Strasbourg

In (other) words, isotony (and strict monotony) is a sufficient condition for convergence, and it enables greedy algorithms like *Dijkstra* to be efficient! Relying on an algebra with operators such as $(min, +)$, like in IGP networks, it is easy (but another story with BGP...).

In case of multi-metric IGP routing problems, the definition of isotony can be generalized to **non dominated paths** (the *Pareto Front* with a controllable size of Γ and with $\gamma = \frac{\Gamma}{c_1}$) as optimal paths. It is then enough to extend them to retrieve **all** optimal paths!

Although the 2COP problem is (weakly) NP-hard, one can easily reach almost optimal solutions. For example, with guarantees such as (p_d is the returned path):

$$d_0(p_d) \leq c_0; \quad d_1(p_d) < c + \frac{c_0}{\gamma}; \quad d_2(p_d) \leq d_2(p_d^*)$$

with d_i providing distance for mettic i, $c \leq c_1$, the delay constraint, and p_d^* the optimal constrained path towards d.

Improving Routing Systems with SR

Best Exact Segment Tracks for 2 Constrained Optimal Paths (BEST2COP)

Université de Strasbourg

Goal

Compute Delay Constrained Least Cost (DCLC) paths in SR domains

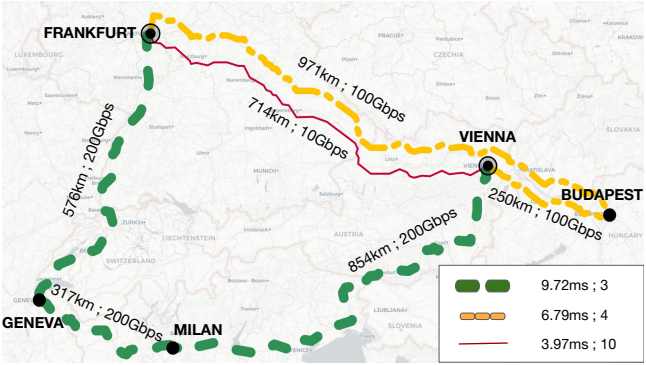
Challenge and Originality

With a third SR operational constraint (MSD), solve DCLC in *massive scale networks*

The Algorithm in a Nutshell

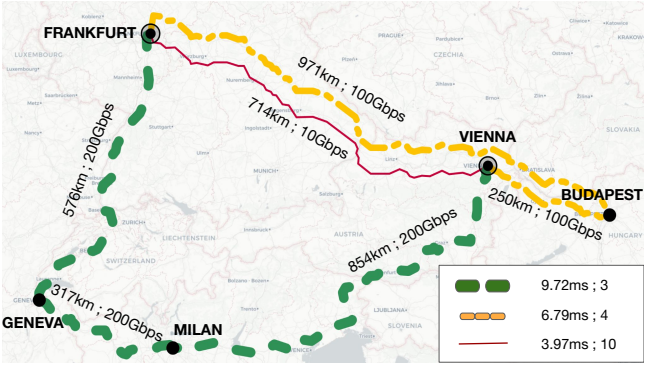
- Transform the raw graph into a multi-metric SR graph enabling ECMP with min-max DAGs: (best IGP cost, worst delay) to satisfy the delay constraint
- Use a Constrained Bellmann-Ford (BF) variant limited to MSD steps on the SR graph
- Group the *Pareto Front* updates at each iteration of ***the parallel not in place*** BF variant

Problem Statement: DCLC in SR domains



A simple DCLC illustration

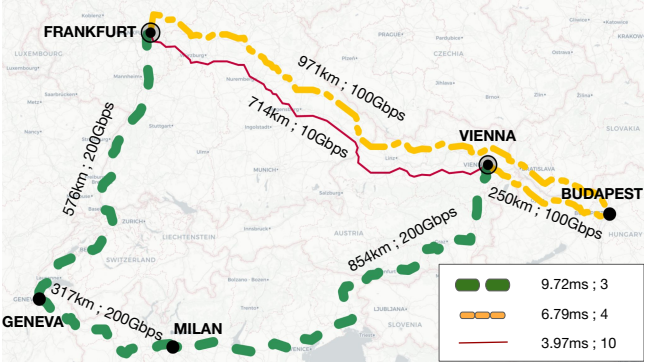
Problem Statement: DCLC in SR domains



A simple DCLC illustration

- DCLC is only weakly NP-hard
- Delays are measured (trueness)
- Approximation with guarantees!

Problem Statement: DCLC in SR domains

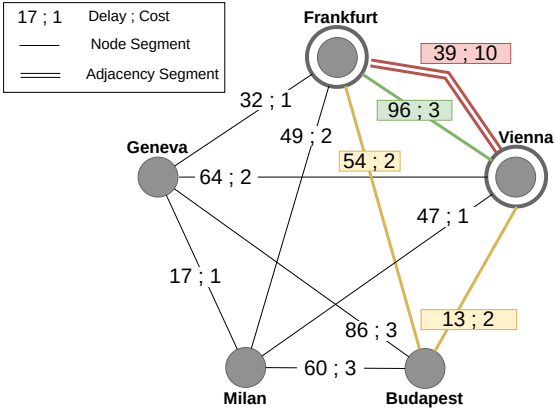


A simple DCLC illustration

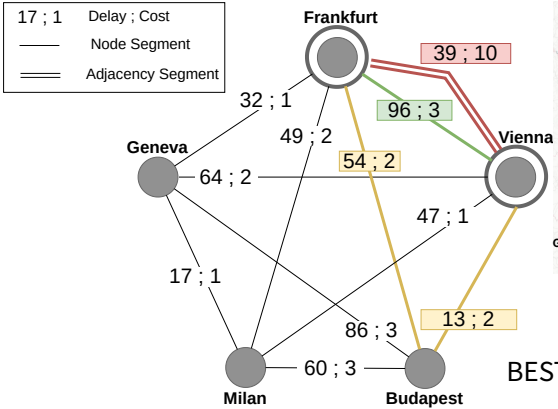
- DCLC is only weakly NP-hard
- Delays are measured (trueness)
- Approximation with guarantees!

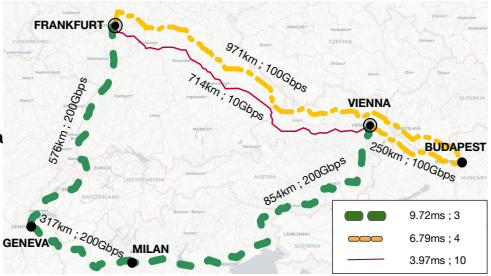
- The SR constraint (MSD) can be natively handle (SR graph)
- Our BF implementation can be efficiently multi-threaded...
- And Pareto-front updates can be grouped at each iteration!

The Data-Structures and our Model



The Data-Structures and our Model



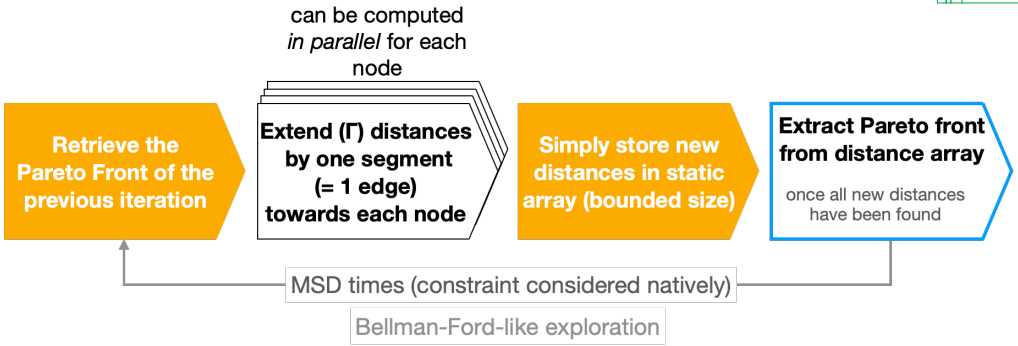


← The multi-metric SR graph

BEST2COP iterates on it MSD times

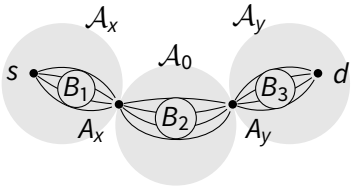
The SR encoding is either embedded natively or can be used as a third dimension with a live-conversion algorithm (GoFoR-SR) as we will see afterwards...

The Core Algorithm



$$O(\underbrace{MSD}_{\text{max. iterations}} \times \underbrace{\Gamma}_{\text{max. distances}} \times \underbrace{V^2}_{\text{\#edges on SR Graph}})$$

Extensions for Massive Scale Networks



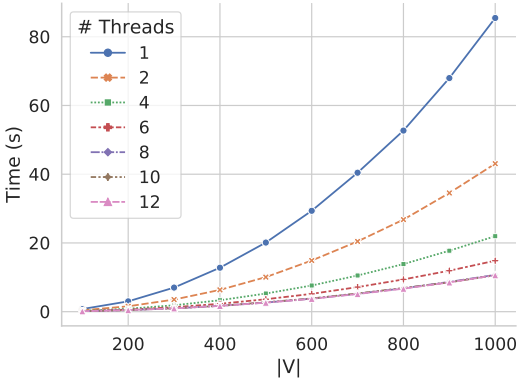
$\text{b2cop}(s, d) \subset B_1 \times B_2 \times B_3$

1. Perform a flat B2COP within each area (to obtain each $B_i, i \geq 0$);
2. ABRs A_x share their Pareto Fronts (for area $\mathcal{A}_x$ and others in the opposite direction) among areas and perform their Cartesian products;
3. A post-processing is applied to refine computations (i.e. to remove superfluous segments and clean merged Pareto Front), in particular when several ABRs exist between the backbone (area 0) and other areas.

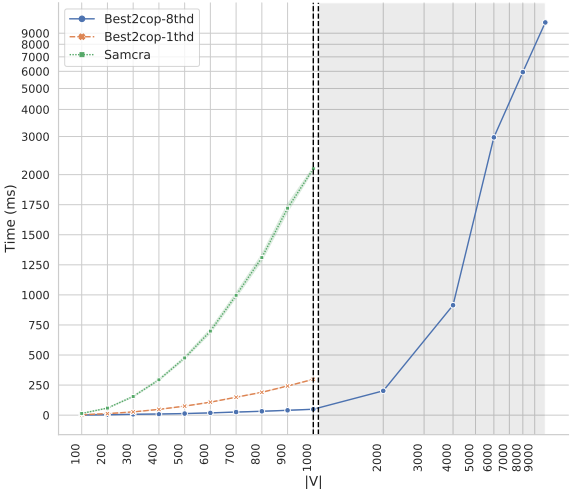
B2COP can be extended into BEST2COPE to deal with multiple areas and save computing, memory and message ressources.

Performance Evaluation: First Results

Flat Core IP Networks

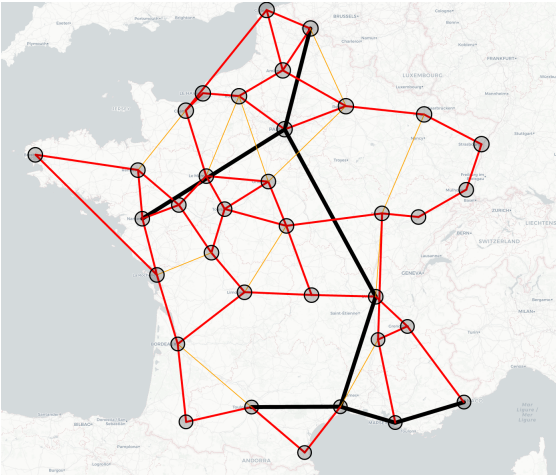


Crash test for BEST2COP

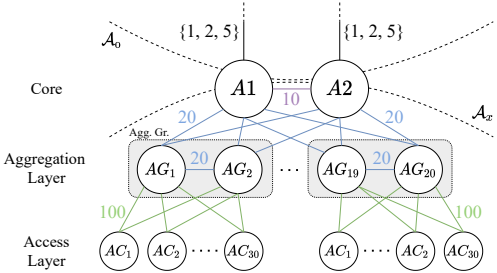


Comparison with SAMCRA

Extended Performance Evaluation: Setup



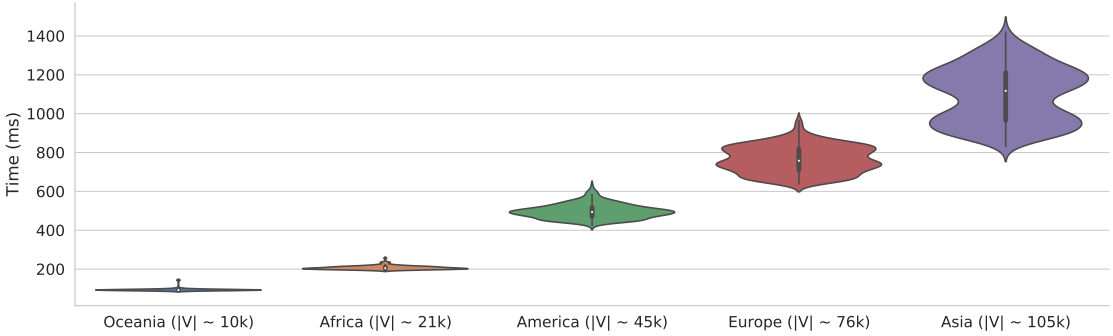
YARGG@core (Backbone)



YARGG@edges (Area > 0)

Performance of BEST2COPE

IP Graphs such that $10.000 < n = |V| < 105.000$



BEST2COPE scales very well for massive networks!

Wrap SR around any use-cases?!

Wrap SR around any routing problems

Université de Strasbourg

GoFoR-SR – submitted to CoNEXT'23 – a general and optimal framework

Goal: Optimally Deploy Source Routing (SR)

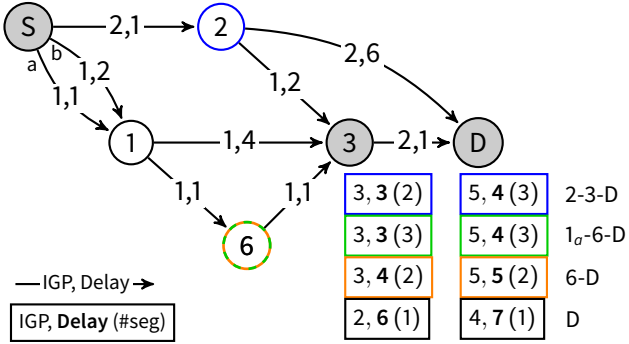
For **usual shortest path computation & routing models**, such as:

- best path on an orthogonal metric (non IGP, e.g., the latency);
- best path avoiding a link / node;
- and their generalization: set of optimal paths, i.e. non dominated, on multi-weighted graphs avoiding a link / node!

- Several strategies: lexicographic, constrained (Pareto) vs. 1Best, AllBest, All (diverse multi-path options);
- **Two main challenges**: (*i*) *truly* optimal encoding and, (*ii*), extension of the comparaison function (initially denoted as $\leqslant_{\setminus 0}$ without SR)

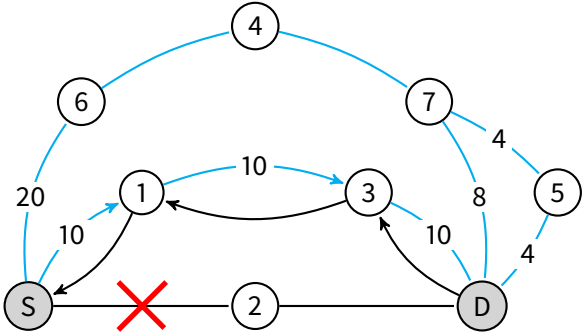
GoFor-SR: minimize delays (first example)

Relying only on IGP segments!



Loss of Isotony: how to extend **the comparison operator** $\leqslant_{\backslash 0}$ to **sort paths** while efficicently integrating SR?

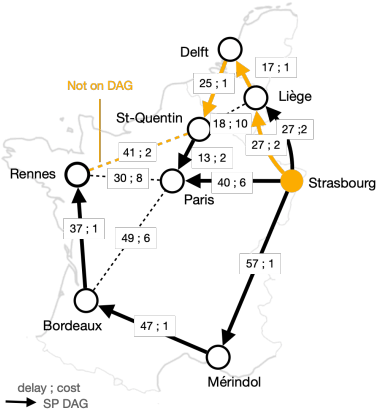
A SECOND EXAMPLE: AVOID A FAILED COMPONENT



How to **encode (all) paths verifying the objectives** (i.e., with desired structural and logical properties)? Towards a *loose* (and efficient) approach to translate raw paths into logical ones – not a *strict* encoding (as often seen in the literrature).

A GREEDY ENCODING APPROACH...

...BUT WITH A LOOSE MODEL!



Strasbourg - Liège (27 ; 2)
Liège - Delft (17 ; 1)
Delft- St Quentin (25 ; 1)
St Quentin - Rennes (41 ; 1)

What is the longest sub-path that can be encoded in one segment?

Node: Strasbourg - St Quentin
Node: Saint Quentin - Rennes

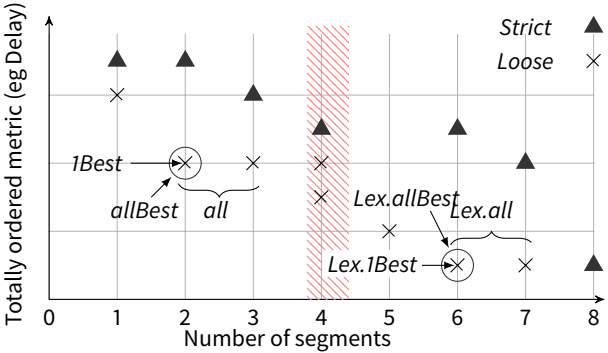
Key insights:

- Go as far as possible (until the link is not on the DAG)!
- Check that all paths in the DAG have better or equal properties (typically on the delays);
- Support multiple types of segments (IGP, Delay, Flex-algo in general...)!

Several Strategies/options to extend $\leqslant_{\setminus 0}$

Option	$\mathcal{R}$	Segment list x $\mathcal{R}$-dominates y, iff:
Constrained-SR		
oneBest	$\preccurlyeq$	$d(x) \leqslant_{\setminus 0} d(y) \wedge d_0(x) \leq d_0(y)$
allBest	$\prec$	$x \preccurlyeq y \wedge d(x) \neq d(y)$
all	$\prec\!\!\!

Illustration of our set of strategies



Distribution of optimal distances toward a given destination, depending on the strategy and the encoding scheme.

A simplified recipe: last source is the key!

Université de Strasbourg

Definition of the extended relation

Let $\mathcal{R}$ be a dominance relation over SR-distances. The extended relation $\overset{\Diamond}{\mathcal{R}}$ associated with $\mathcal{R}$ is defined over segment lists as follows. For any segment lists L and L', we have $L \overset{\Diamond}{\mathcal{R}} L'$, *i.e.*, L' is $\overset{\Diamond}{\mathcal{R}}$-dominated by L, if either

(*i*) $d(L)\,\mathcal{R}\,d(L')$, and $L^{type}_{last} = L'^{type}_{last} \wedge L^{src}_{last} \in L'_{last}$;

(*ii*) or $d(L)\,\mathcal{R}\,(d_0(L') - 1, d_{\setminus 0}(L'))$

where L_{last} denotes the last segment of L and L^{src}_{last} its source. Notation $d_{\setminus 0}$ refers to the exclusion of metric d_0 in the vector of considered metrics.

Apply the recipe: Wrap SR around any $\mathcal{A}$

Université de Strasbourg

Theorem: the extended dominance $\overset{\Diamond}{\mathcal{R}}$ is isotonic

The extension of a $\overset{\Diamond}{\mathcal{R}}$-dominated path remains $\mathcal{R}$-dominated, even after extensions.

Formally, from algorithm $\mathcal{A}$ ($\leqslant_{\setminus 0}$), every call to $\hookrightarrow$ (insert path p in the PQ)

$$p \overset{\leqslant_{\setminus 0}}{\hookrightarrow} \mathcal{P}$$

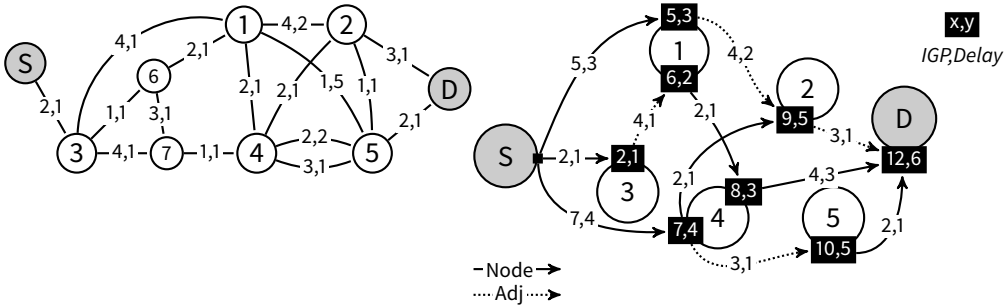
is replaced by a call (generalized for non dominated paths)

$$\textsc{Encode}(p) \overset{\overset{\Diamond}{\mathcal{R}}}{\hookrightarrow} \mathcal{L}$$

with $L \overset{\mathcal{R}}{\hookrightarrow} \mathcal{L} \Leftrightarrow \nexists L' \in \mathcal{L}$ such that $L' \mathcal{R} L$

A last example: 2COP is back!

Delay Constrained (<7 ms) – Least Cost



How to implement such a logical metaDAG in a constrained architecture with limited ressource (e.g., # stages in the pipeline and memory limitations)?

Our goal is to enable **Load Balancing driven at the source!**

Overall Complexity of GoFoR-SR

Université de Strasbourg

$$n\Gamma.MSD \times log(n\Gamma.MSD) + mn(\Gamma.MSD)^2$$

instead of

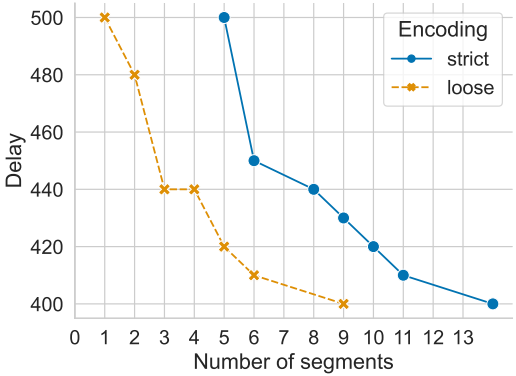
$$MSD \cdot \Gamma \cdot n^2$$

with:

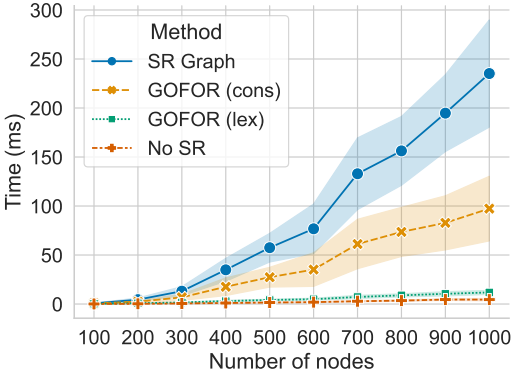
- n and m, the cardinal fo nodes and edges respectively;
- Γ, the maximal number of non dominated paths per node (typically $\Gamma \approx 1000$);
- MSD, the maximal number of segments ($MSD \leq 10$).

But $m \approx n$ in the raw graph and $n \approx 1$ in $mn(\Gamma.MSD)^2$ in practice..!

Performance Evaluation



The *power* of loose encoding on *Min-Delay*



2COP: a marginal overhead

GoFoR-SR is efficient, both in terms of encoding and computing time.

Conclusion & Perspectives

Summary

Université de Strasbourg

SR enables various and complex tactical TE: from FRR up to DCLC paths...

- BEST2COP for worst-case graphs (e.g. dense ones)
- GoFoR-SR for usual graphs (e.g. sparse metric-aligned ones)
- in addition, and overall, GoFoR-SR is more versatile!

With GoFoR-SR, the maximal number of segments, MSD, can be handled with several options: **whatever the use-case and the strategy**, GoFoR-SR is efficient and easy to wrap around any path computation algorithm.
Moreover, it deals with **any type of segments** and can combined them (i.e. not only IGP base ones).

Future works

Université de Strasbourg

We aim to implement GoFoR-SR in P4!

A prototype to inspect all corner aspects of an almost real deployment such as:

- implementation of logical (meta)DAG in Match Action tables?
- fast-reroute implementation: how to quickly switch on failovers?
- load-balancing at the source: use 2COP and its end-to-end delay estimation to balance TCP flows?
- generally speaking, how to optimize the TCAM memory?

Finally, perform experiments and performance evaluation on real hardware!

Acknowledgements

Université de Strasbourg

Here is the list of people involved in this work:

- Jean-Romain Luttringer (the best-ex PhD student)
- Quentin Bramas (the Great folk)
- François Clad (the Cisco folk)
- Cristel Pelsser (the PhD supervisor)
- ...and the internships: Thomas Alfroy, Adrien Rodau, etc.

And the three papers on which this talk is based:

- **NCA 2020**: *Computing Delay-Constrained Least-Cost Paths for Segment Routing is Easier Than You Think*
- **ComNet 2022**: *Deploying Near-Optimal Delay-Constrained Paths with Segment Routing in Massive-Scale Networks*
- **...CoNext 2023?** *GoFOR-SR: a simple General Operational Framework to deploy Optimal Routes with Source Routing*