Linear Programming Models for Traffic Engineering in 100% Survivable Networks under Combined IS-IS/OSPF and MPLS-TE

Based on: Computers & Operations Research 38 (2011) 1805-1815

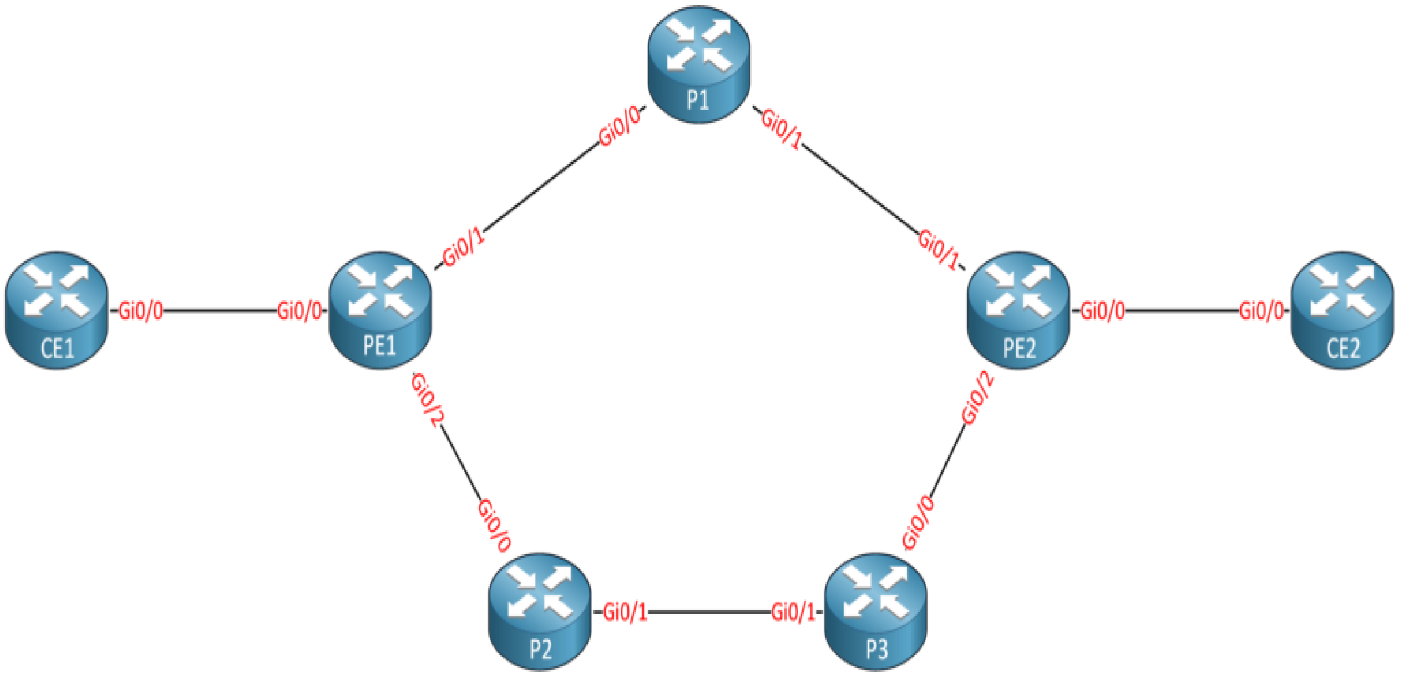
D. Cherubini, A. Fanni, A. Mereu, A. Frangioni, C. Murgia, M.G. Scutellà, P. Zuddas

July 9, 2025

Francesca Craievich

Index

- Introduction & motivation
- Technological background
- Definition of the problem
- Model
- Results
- Scalability



The Routing Problem in Modern IP Networks

High Efficiency

Maximize usage of available resources

Reliability

Ensure network operation even during failures

Flexibility

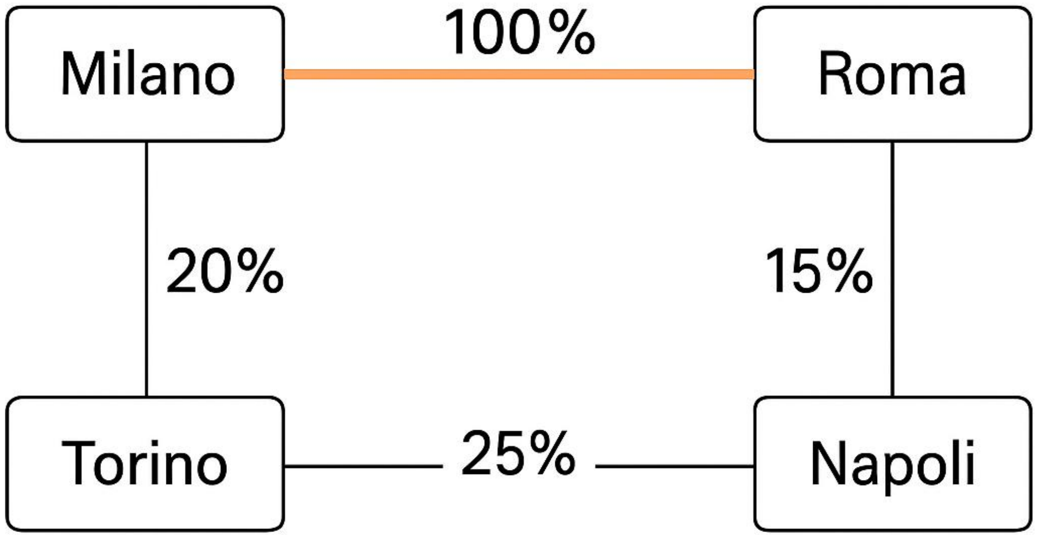
Adapt to changing traffic patterns

Traditional IGP Routing (IS-IS / OSPF):

- Each router computes shortest paths to all destinations
- Based on static link metrics (weights)
- Traffic always follows the shortest paths

Consequences

- Some links become **congested hotspots**
- Other links remain **underutilized**
- **Load balancing** is impossible by only adjusting link weights



Problem: The Milano–Roma link is saturated (100% utilization)

⇩

We need smarter mechanisms!

Limitations of Existing Solutions

Let’s dive in

Limitations of Existing Solutions

IGP Weight Optimization

Optimal link weights to minimize congestion

- NP-hard problem
- Requires heuristic algorithms
- Long computation time: hours/days
- Changing weights causes network instability
- Cannot be done frequently

Pure MPLS-TE

Create explicit tunnels for every traffic flow

- Requires $O(n^2)$ tunnels for *n* nodes
- Complex configuration
- Completely abandons the existing IGP infrastructure
- High migration cost

SOLUTION: IGP + MPLS-TE

- Most traffic continues to use IGP
- MPLS-TE selectively reroutes traffic from congested links
- Limited number of LSPs (Label Switched Paths)

Advantages:

✓ Leverage existing infrastructure

✓ Manageable complexity

✓ Global optimization

Scientific Contributions of the Paper

INNOVATIVE MODEL

- First joint LP for IGP/MPLS-TE
- Integrates survivability
- Polynomial complexity

SURVIVABILITY MANAGEMENT

- 100% protection against single failures
- Link restoration through LSPs
- No manual backup configuration required

EXTENSIVE VALIDATION

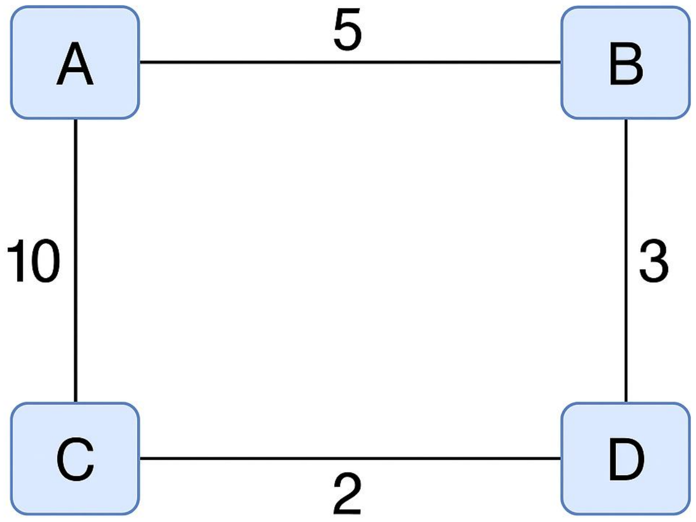
- Tested on 5 networks
- Comparison with existing approaches
- Scalability analysis up to 30+ nodes

PRACTICAL IMPLEMENTATION

- Optimized with commercial solvers
- Execution times below 1 second
- Ready for production

How IGP Works

- Each link has a weight (metric)
- Every router computes the shortest paths
- Dijkstra’s algorithm



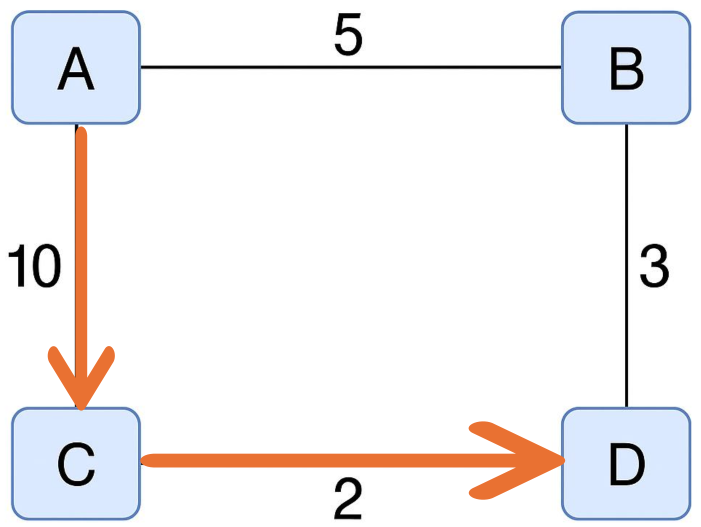
Limitations:

- Cannot split traffic over multiple paths
- All A ? D use the same path
- Changing link weights impacts all traffic flows

How MPLS-TE Works

Multi-Protocol Label Switching – Traffic Engineering

- LSP: virtual point-to-point tunnel
- Operator explicitly defines the path
- Allows reserving link capacity



Advantages:

- Can use non-shortest paths
- Guaranteed Quality of Service (QoS)

Disadvantages:

- Complex manual configuration and poor scalability

Survivability Concept

The network must remain fully operational even if **any single link** fails.

Types of Protection:

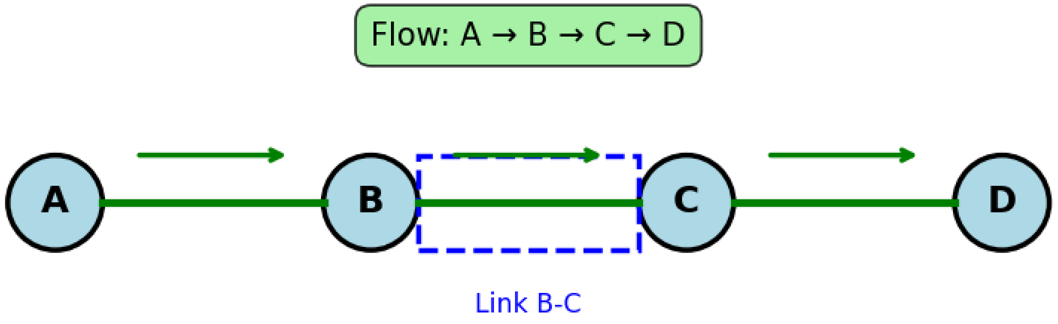
- PATH PROTECTION
  - End-to-end backup path
  - Slow (50-200 ms recovery)
- LINK PROTECTION

  (used in the paper)
  - Local rerouting around the failed link
  - Fast (<50 ms recovery)

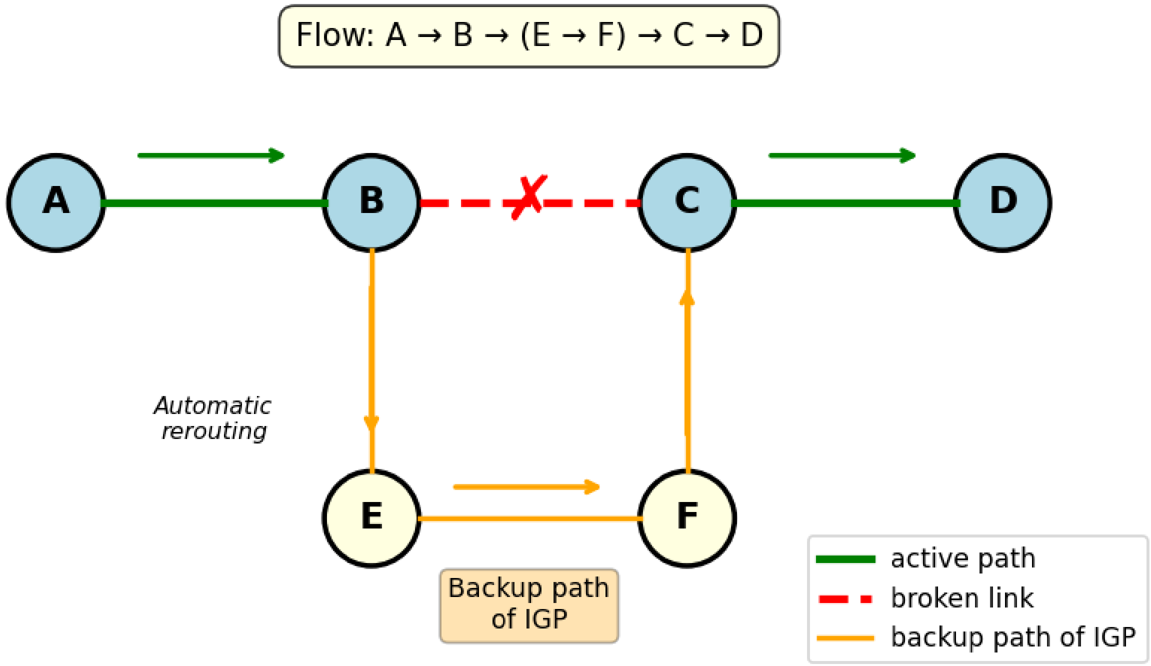
Every failure scenario is precomputed, no manual configuration

Example of Recovery from Failure

Normal scenery



Recovery scenery



Bibliographic Review

Methodology

Bibliographic Review - Methodology

Databases consulted:

- IEEE Xplore
- ACM Digital Library
- ScienceDirect
- Google Scholar
- SpringerLink

Keywords used:

("IGP" OR "OSPF" OR "IS-IS") AND
("MPLS-TE" OR "MPLS") AND
("combined" OR "hybrid" OR "joint") AND
("optimization" OR "linear programming") AND
("survivability" OR "failure" OR "restoration")

Results:

847 papers → 42 relevant after screening

Why Cherubini (2011) Remains Unique

Aspect	Cherubini (2011)	Later Papers
Model	Exact LP	Heuristics / ML
Technologies	IGP + MPLS-TE	SDN / SR / MPLS-only
Survivability	Integrated (link restoration)	Separate or absent
Compatibility	Existing networks	Requires upgrade
Complexity	Polynomial time	NP-hard or iterative
Guarantees	Global optimum	Best-effort

Related Papers Post - 2011

1. Zhang et al. (2012) *Hybrid IGP/MPLS with SDN*
   ✗ Requires centralized SDN controller
   ✗ Does not address survivability
   ✗ Heuristic model, not LP

2. Kumar et al. (2015) *Joint Optimization for TE*
   ✗ MPLS-TE only, not hybrid
   ✗ Path protection, not link restoration
   ✓ LP model, but with different objectives

Related Papers Post - 2011

3. Wang & Liu (2018) *ML-based Traffic Engineering*
   ✗ ML, not exact optimization
   ✗ No guarantees of optimality
   ✗ Requires training data

4. Bhatia et al. (2021) *Segment Routing Optimization*
   ✗ SR-based, not classic MPLS-TE
   ✗ Not compatible with legacy networks
   ✓ LP model but with different technology

Segment Routing VS MPLS-TE

Why SR Does Not Replace This Approach

- **SR requires hardware upgrade**
  - Not all routers support SR
- **MPLS-TE is universal**

- **SR models are more complex**
  - Larger solution space
  - Optimization is harder

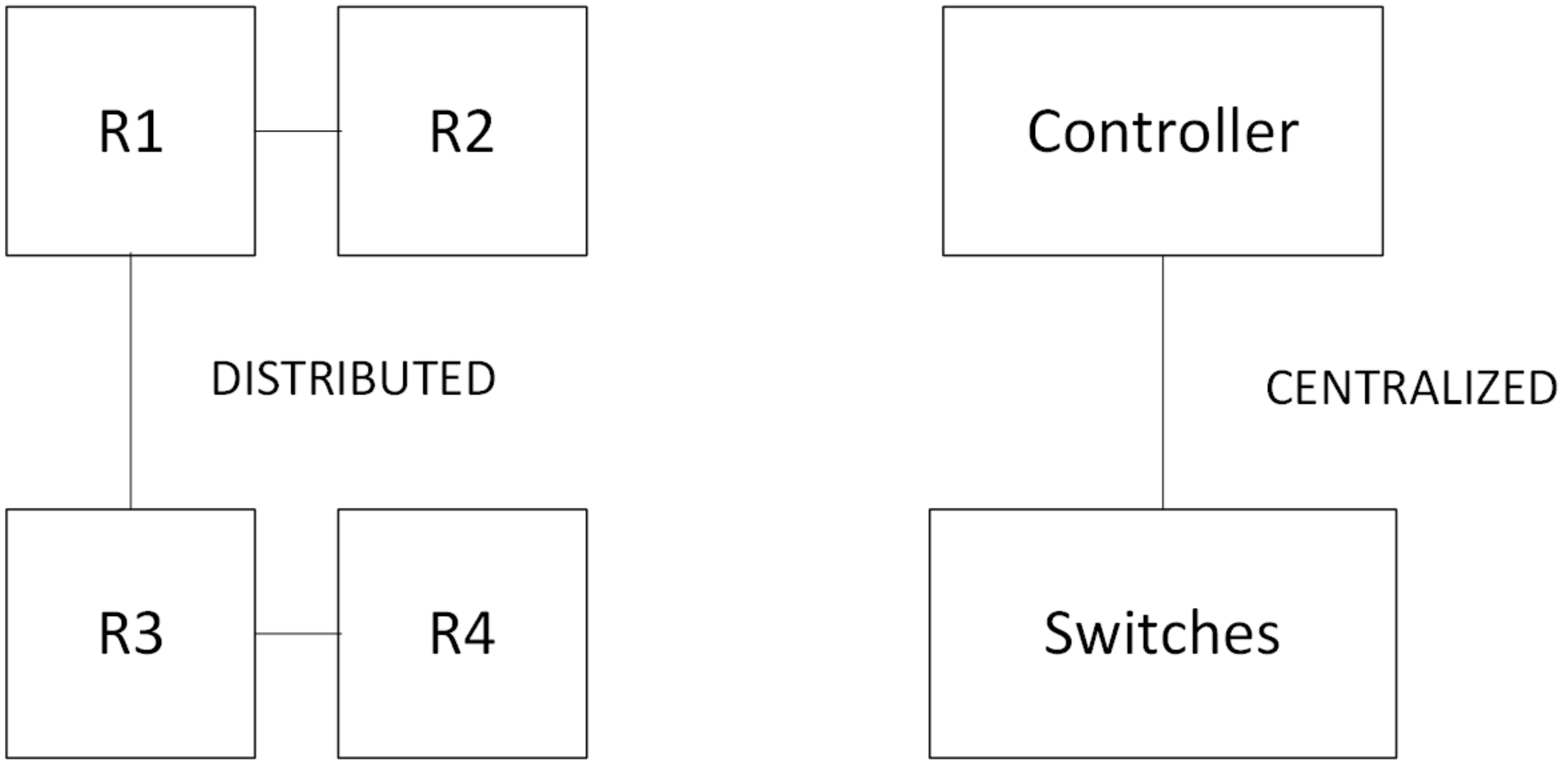
- **Legacy networks** still rely on MPLS-TE
  - Migration is costly and risky

2024 Stats (Cisco Network Report):

- 68% of enterprise networks still use MPLS-TE
- 45% plan to maintain it for 5+ years
- SR adoption: only 23%

SDN and Centralized Control Plane

Why SDN Does Not Replace This Approach



Problems:

- Controller-switch latency
- Limited scalability
- Reduced resilience

Recent Research Confirming the Approach

Papers That Cite and Validate Cherubini

- **Chen et al. (2023)** – *Survey on Traffic Engineering*

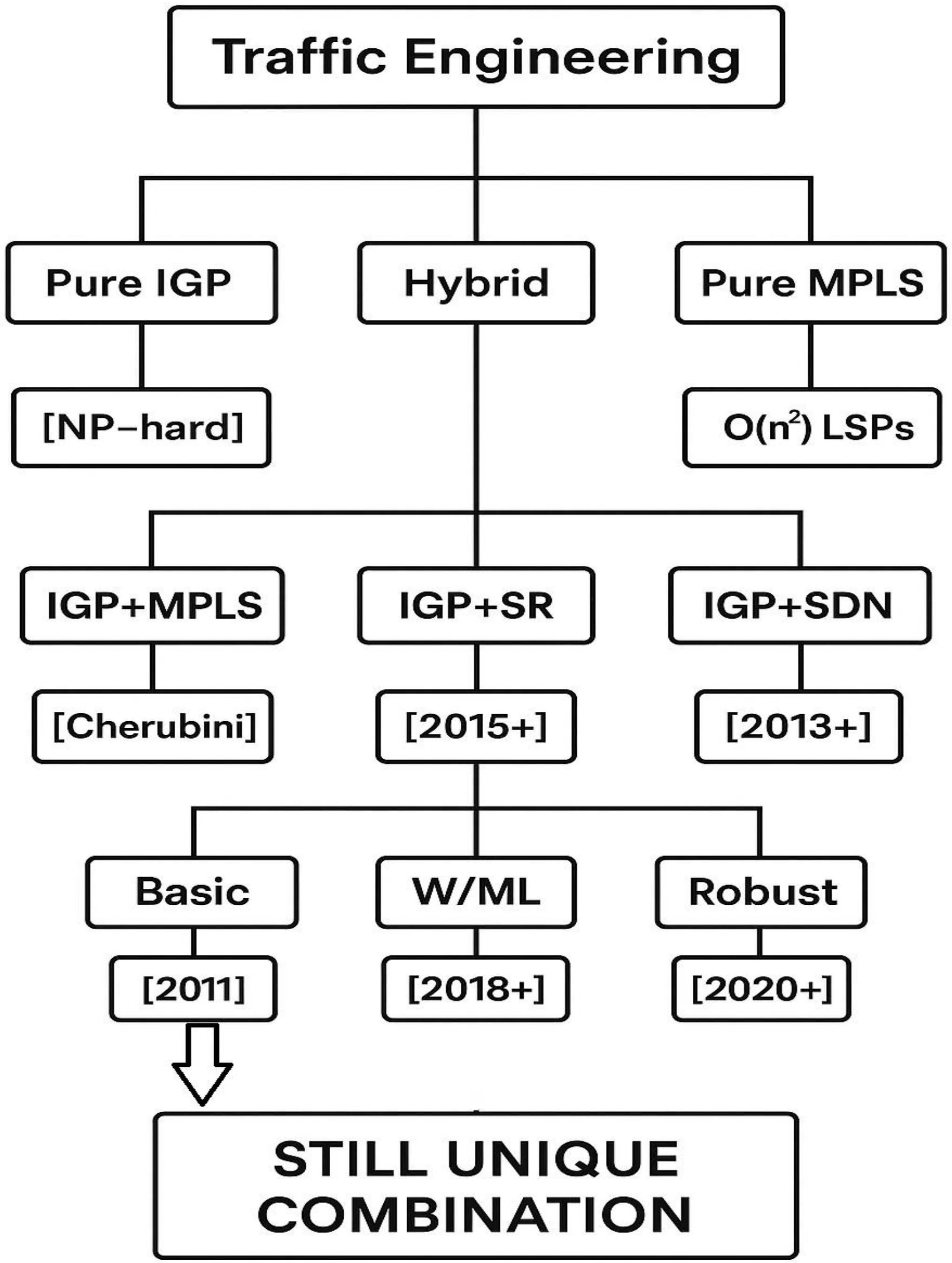
"The LP formulation by Cherubini remains the most efficient approach for joint IGP/MPLS optimization“

- **Rodriguez et al. (2022)** – *Survivable Network Design*

"For link restoration, the model in [Cherubini 2011] is still unmatched“

- **Park & Kim (2024)** – *Hybrid Routing Strategies*

"We extend Cherubini's model to multi-layer networks“



Traffic Engineering
Pure IGP
Hybrid
Pure MPLS
[NP–hard]
O(n²) LSPs
IGP+MPLS
IGP+SR
IGP+SDN
[Cherubini]
[2015+]
[2013+]
Basic
W/ML
Robust
[2011]
[2018+]
[2020+]
STILL UNIQUE COMBINATION

Formal Definition of the Problem

Let’s dive in

Formal Definition – Network Model

The network is represented as a graph:

$G = (N, E)$

- **N** = set of nodes (routers)
- **E** = set of edges (physical links)

For each link $\{i, j\} \in E$:

c_{ij} = bidirectional capacity (in Mbps)

- The link can simultaneously transmit c_{ij} from $i \rightarrow j$ *and* c_{ij} from $j \rightarrow i$

Formal Definition – Network Model

Directed Graph Representation:

$G' = (N, A)$

- A = set of directed arcs
- For every $\{i, j\} \in E$, we define both directed arcs:
  - $(i, j) \in A$
  - $(j, i) \in A$

Compact Notation:

- $l \in E$ denotes an undirected edge
- $l^+ = (i, j)$ and $l^- = (j, i)$ are the corresponding directed arcs

Formal Definition – Traffic Demands

Commodities (Flows):

Let F be the set of all origin-destination pairs (commodities).

For each $f \in F$:

- $s(f)$ = source node
- $t(f)$ = destination node
- d^f = traffic demand in Mbps

Aggregation for Efficiency:

Let H be the set of source nodes ($H \subseteq N$)

Let $F(v) = \{f \in F : s(f) = v\}$ be the set of all commodities originating from node v

Formal Definition – Traffic Demands

Example:

Original commodities:

- f_1: Rome → Milan, 100 Mbps
- f_2: Rome → Turin, 50 Mbps
- f_3: Rome → Naples, 80 Mbps

After aggregation:

- **v = Rome**, total traffic = **230 Mbps**
- Multiple destination nodes

Formal Definition – IGP Routing

The Routing Matrix X

Precompute Input:

Let X be the **m × k** routing matrix (edges × commodities), where:

$x_{ij}^{f} \in [0,1]$ is the fraction of commodity **f** that is routed through the directed link **(i, j)**

How X is Computed:

1. Assign IGP weights to each link
2. Compute the shortest path for every source-destination pair
3. If **ECMP** (Equal-Cost Multi-Path) is enabled, traffic is evenly split among all shortest paths

Formal Definition – IGP Routing

Properties:

- $\sum_j x_{sj}^f = 1$ → the entire flow exits the source node **s**
- $\sum_i x_{it}^f = 1$ → the entire flow enters the destination node **t**
- **Flow conservation** holds at intermediate nodes

Important:

X is a **given input to the model**, not a decision variable!

Objective Definition

What We Aim to Optimize

Primary Objective:

- Minimize u_{max} = maximum link utilization
- $u_{max} \in [0, 1]$ (i.e., 0% to 100%)

Secondary Objective:

- Minimize the **number of LSPs**
- Controlled by a parameter **δ**
- There is a **trade-off** with the primary objective

Multi-objective Formulation:

$$\text{minimize} \quad u_{max} + \delta \times (\text{number of LSPs})$$

Problem Constraints

Requirements the Solution Must Satisfy

Link Capacity Not Exceeded

- Total traffic ≤ capacity × $\mathbf{u_{max}}$
- Applies to **every link and direction**

All Demand Must Be Met

- **Each commodity** must receive exactly $\mathbf{d^f}$
- Can be routed via **IGP and/or MPLS**

Flow Conservation

- **Incoming flow = outgoing flow** at all **intermediate nodes**
- Exception: **source** and **destination** nodes

Problem Constraints

Requirements the Solution Must Satisfy

Fixed IGP Routing

- Follows the **precomputed matrix X**
- Cannot be modified by the model

Survivability (Extended Model)

- Additional constraints for each **failure scenario**
- Ensures $\mathbf{u_{max}}$ is respected **even under link failures**

Linear Programming Model

Let’s get into it

LP Model – Decision Variables

1. Objective Variable: $u_{max} \in [0, 1]$

- Represents the **normalized maximum link utilization**

2. IGP/MPLS Split Variables: $is^f \in [0, d^f]$ for each $f \in F$

- Demand for commodity **f** is routed over **IGP**
- The remaining part $(d^f - is^f)$ is routed over **MPLS**

$\boxed{O(|H| \cdot |A| + |F|)}$

3. MPLS Flow Variables: $w_{ij}^v \geq 0$ for each $v \in H$, $(i, j) \in A$

- Amount of **MPLS flow originated from node v** over link **(i, j)**
- Aggregated across all destinations from **v**

LP Model – Objective Function

Basic Version:

$$\min u_{max}$$

Version with LSP Penalty:

$$\min u_{max} + \delta \sum_{v \in H} \sum_{j:(v,j) \in A} w_{vj}^{v}$$

- **δ = 0**: Maximizes MPLS usage (results in **more LSPs**)
- **Small δ (e.g., 10^{-7})**: Balanced behavior
- **Large δ (e.g., 10^{-3})**: Favors minimizing LSPs (more IGP)

LP Model – Capacity Constraints

Constraint (2): Link Utilization

Formulation:

$$\sum_{f \in F} is^f \cdot x_{ij}^f + \sum_{v \in H} w_{ij}^v \leq u_{max} \cdot c_{ij} \qquad \forall (i,j) \in A$$

- **First term:** IGP traffic on the link
  - **is^f** = commodity **f** routed over IGP
  - **x_{ij}^f** = fraction of that flow using link **(i, j)**

- **Second term:** total MPLS traffic on the link
  Sum of all MPLS flows across sources **v ∈ H**

- **Right-hand side:** scaled capacity
  - **c_{ij}** = physical capacity of the link
  - **u_{max}** = scaling factor

LP Model – Flow Conservation

Constraint (3): Flow Conservation

Formulation:

$$\sum_{j:(j,i)\in A} w_{ji}^{v} - \sum_{j:(i,j)\in A} w_{ij}^{v} = b_{i}^{v} \qquad \forall i \in N, \forall v \in H$$

- If $\boldsymbol{i = v}$ (source node):

$$b_{v}^{v} = \sum_{f\in F(v)} (is^{f} - d^{f})$$

Negative: more traffic leaves than enters.

We use $is^{f} - d^{f}$ because the opposite $d^{f} - is^{f}$ is routed through the MPLS.

LP Model – Flow Conservation

Constraint (3): Flow Conservation

- If $\boldsymbol{i} = \boldsymbol{t}(\boldsymbol{f})$ for some $\boldsymbol{f} \in \boldsymbol{F}(\boldsymbol{v})$ (i.e. destination of some demand):

$$b_i^v = d^f - is^f$$

Positive: the node must receive MPLS traffic.

- **Otherwise** (intermediate nodes):

$$b_i^v = 0$$

Pure flow conservation.

Extension for Survivability

Failure Management

$$\sum_{f \in F} is^f \cdot x_{ij}^{f,l} + \sum_{v \in H} w_{ij}^v + \sum_{v \in H} \left(x_{ij}^{l+,l} \cdot w_{l+}^v + x_{ij}^{l-,l} \cdot w_{l-}^v \right) \leq \mathrm{u}_{\max} \cdot c_{ij}$$

- $x_{ij}^{f,l}$: IGP routing when link l has failed
- $x_{ij}^{l+,l}$: how MPLS flow from l+ is rerouted after failure
- $x_{ij}^{l-,l}$: how MPLS flow from l- is rerouted after failure

- Traffic on the failed link becomes a new commodity
- Load increases on alternative paths

Problem Properties

1. **Convexity**

- Global optimum guaranteed
- No local optima

2. **Degeneracy**

- *Proposition 4.1*: There exist optimal solution with $is^f = 0 \ \forall f$
- The term δ in the objective function resolves this by selecting solutions with fewer LSPs
- Controllable trade-off between efficiency and operational complexity

Problem Properties

3. **Scalability**

- Empirical complexity: $O(n^{2.76})$
- Tractable up to $\sim$50 nodes

4. **Stability**

- Small perturbations in d^f → small changes in solution
- Well-conditioned matrix

Experimental Setup

From mathematical model to solution

Experimental Setup

Network	Type	Nodes	Links	Flows	Notes
Random	Synthetic	8	10	56	Ring + cross links
IBCN	Research	37	114	332	Belgian university
Tinet	Real ISP	185	430	306	Tiscali Italy
Atlanta	Metro	154	421	210	City-level network
Géant	Backbone	23	76	445	Pan-European research backbone

Experimental Setup

Configurations tested for each network:

- **Default**: All links have weight = 1
  - Simple but often inefficient → ignores capacity and traffic load
- **Optimized**: Weights from Tabu Search
  - Memory: avoids revisiting already explored solutions
  - Diversification: Broad exploration of the solution space
  - Intensification: Refines promising solutions
  - Escape: escape local optima
- **Real**: Operator's actual weights (when available)

Comparison: IGP vs Combined

Network	Config	IGP Only	Combined	Reduction	#LSP
Random	Default	70%	35%	50%	55
Random	TS	47%	35%	26%	48
IBCN	Default	101%	75%	26%	671
IBCN	TS	74%	65%	12%	585
Tinet	Real	117%	83%	29%	83
Atlanta	Default	142%	110%	23%	129
Géant	TS	77%	76%	1%	210

- orange = Congestion (>100%)

From Mathematical Model to Solution

1. Pre-processing of Routing Matrices

- Input: graph $G = (N, E)$ with IGP weights w_e
- Output: matrix X where x^f_{ij} = fraction of flow f on link (i, j)
- Algorithm: all-pairs shortest path with ECMP handling
- Complexity: $O(|V|^3)$ using Floyd-Warshall

2. LP Model Construction

- Variables: u_{max} + $|F|$ variables is^f + $|H| \times |A|$ variables w^v_{ij}
- Constraints: $|A|$ capacity constraints + $|H| \times |N|$ flow conservation
- Structure: sparse formulation exploited for efficiency

From Mathematical Model to Solution

3. Optimal Resolution

- Method: Interior Point Method with guaranteed convergence
- Precision: $\varepsilon = 10^{-8}$ for numerical optimality

4. LSP Path Decomposition

- Algorithm: greedy flow decomposition
- Goal: heuristic minimization of the number of paths

Theoretical Convergence Analysis

Theorem (Ye, 1997 – Interior Point Complexity):

For an LP in standard form with *m* constraints:

minimize $c^T x$

subject to $Ax = b, x \geq 0$

The path-following interior point method converges in: **$O(\sqrt{m} \cdot \log(m/\varepsilon))$** iterations

Applied to Our Problem:

- Number of constraints:

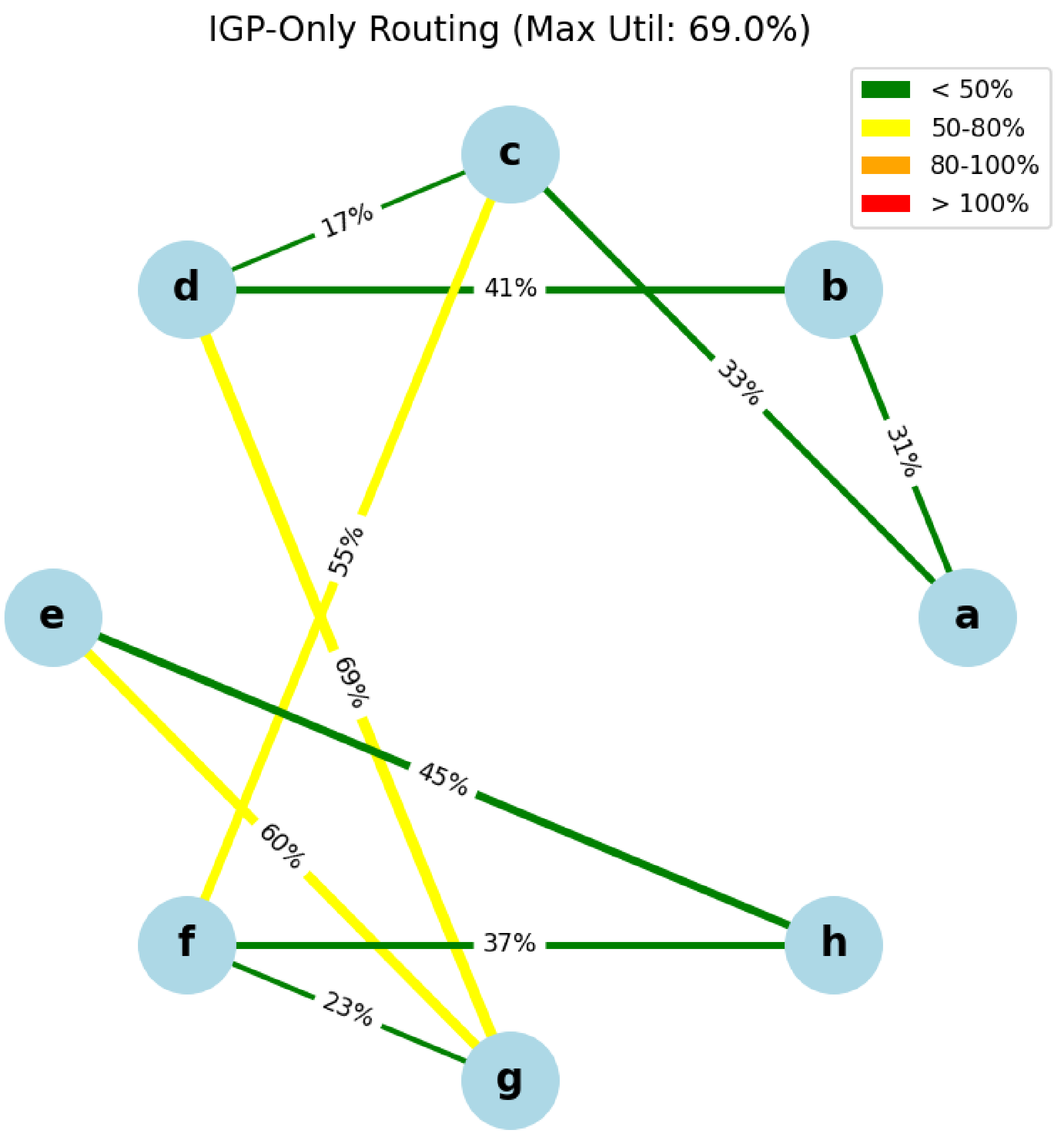
  $m = O(|A| + |H| \cdot |N|) = O(n^2)$
- Convergence in **$O(n \cdot \log n)$** iterations
- Each iteration: **$O(n^3)$** for solving linear system

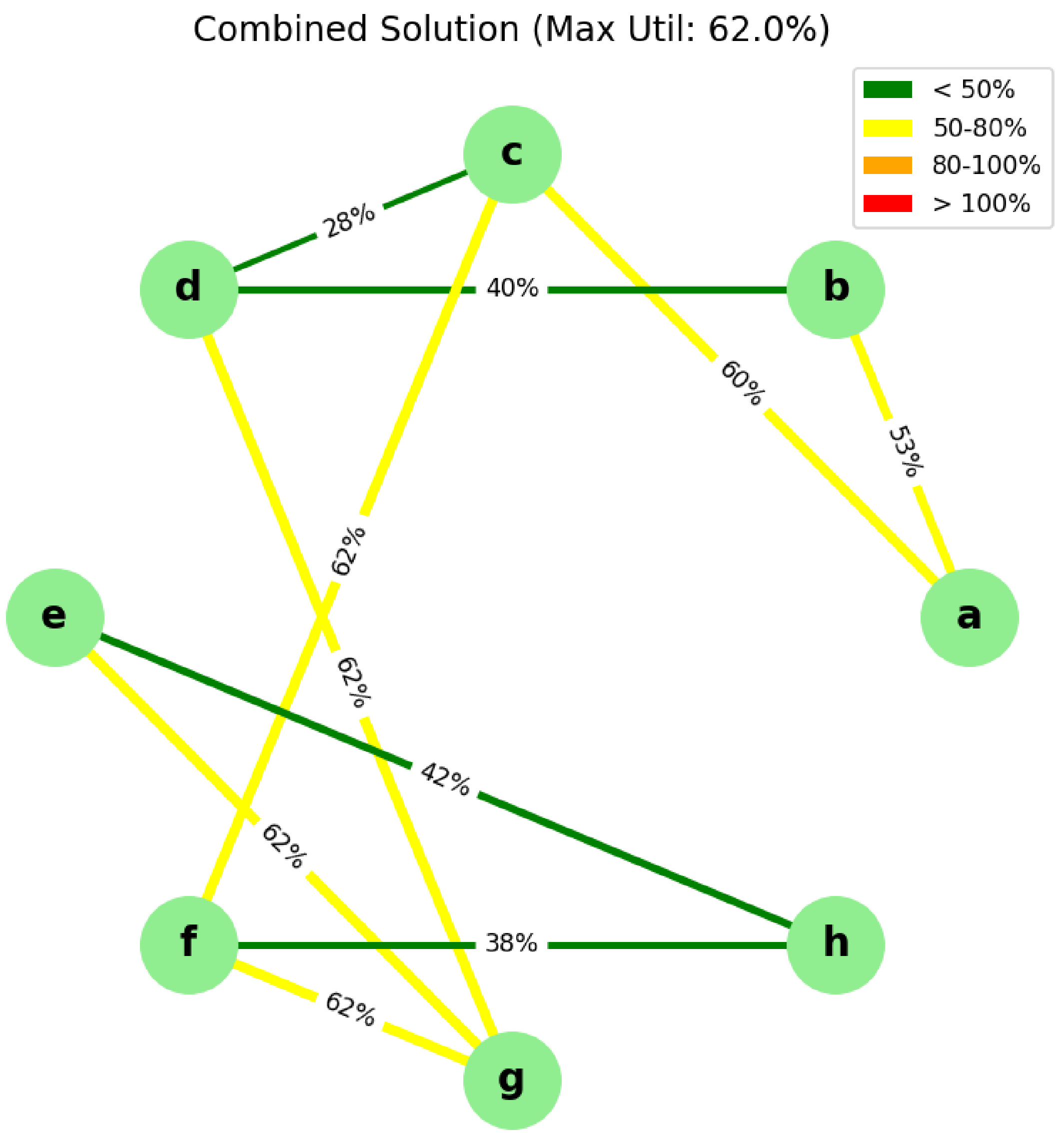
Overall Complexity: $O(n^4 \cdot \log n)$

In Practice: $O(n^{2.76})$ due to:

- Matrix sparsity
- Warm start
- Adaptive preconditioning

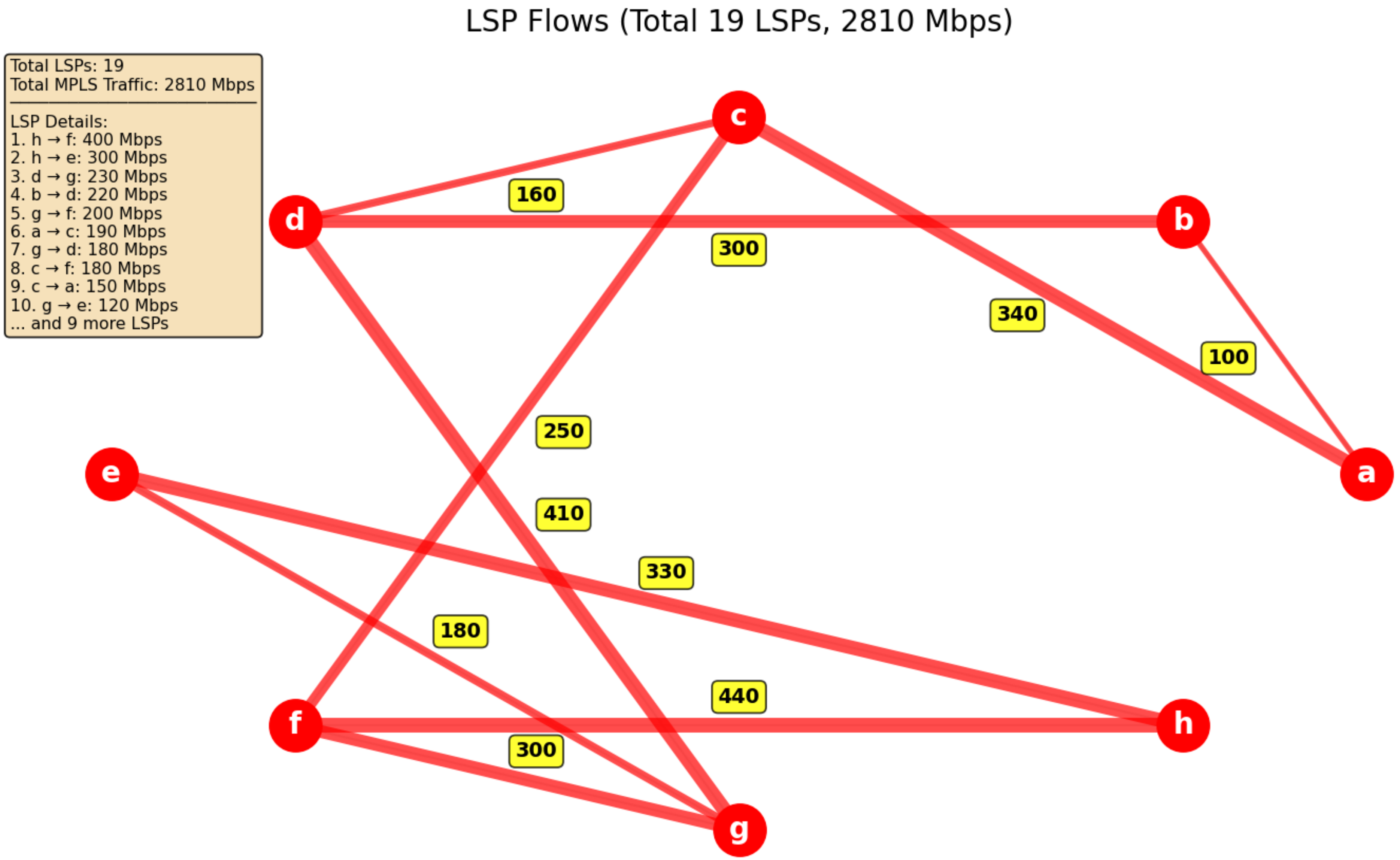
Experimental Results - Random Network





Total Demand: 3500 Mbps | IGP Traffic: 660 Mbps (18.9%) | MPLS Traffic: 2840 Mbps (81.1%) | LSPs: 17

Decomposition of LSP Flows



Decomposition of LSP Flows

Intelligent Traffic Distribution:

- Major flows (300–440 Mbps) use direct paths between high-demand nodes
- Critical links like *f–h* (440 Mbps) and *g–d* (410 Mbps) handle significant volumes
- No link exceeds 50% of its capacity (500 Mbps out of 1000 Mbps)

Why this solution is optimal:

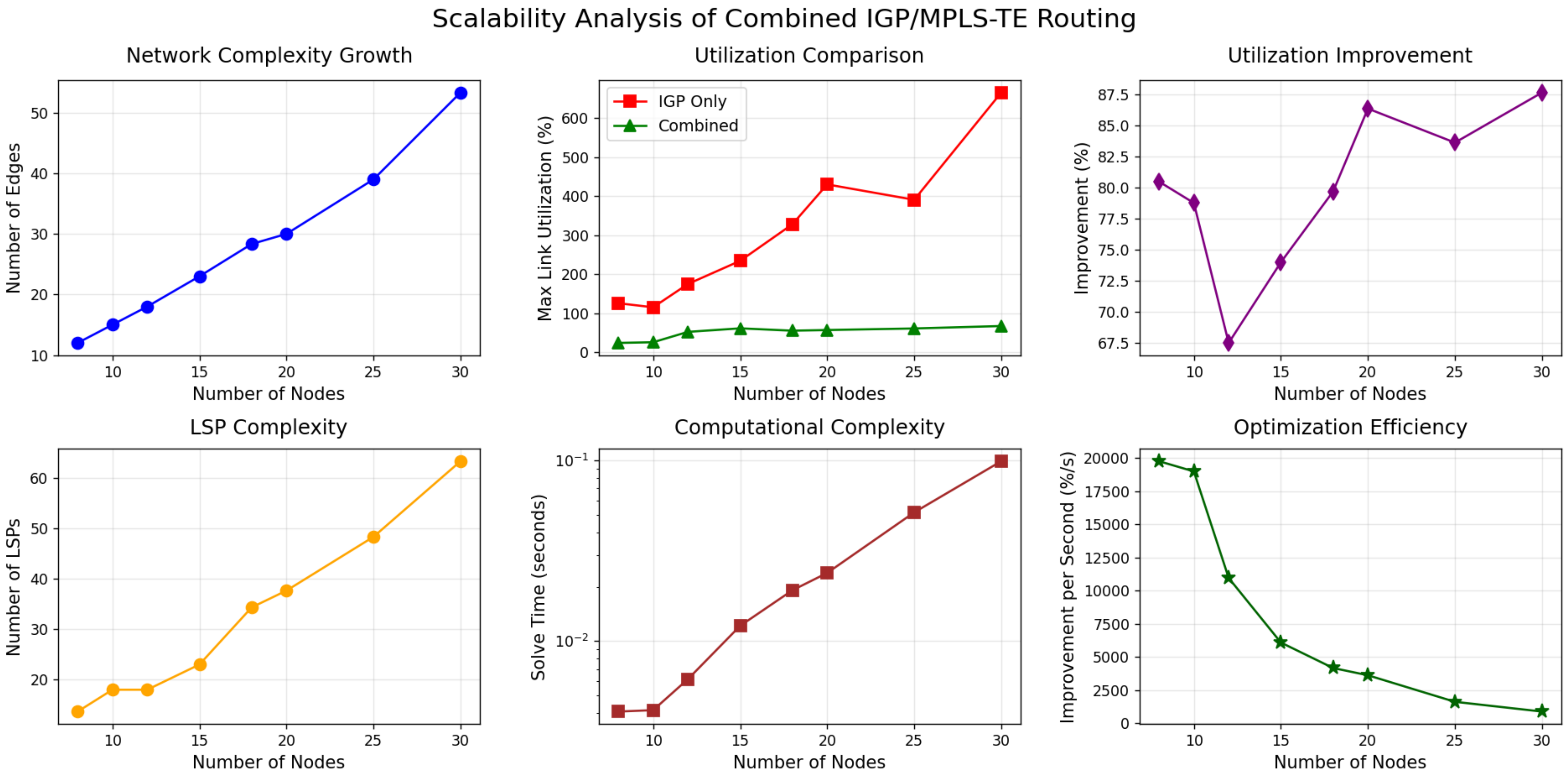
- **Taps spare capacity:** It sends traffic over links that IGP leaves under-utilised (e.g., *e–h*).
- **Keeps hop count low:** Most LSPs follow direct or two-hop routes, reducing latency and resource use.

Note: With $\delta = 0$ the algorithm would create 17 LSPs

Scalability

Time to dig in

Scalability Analysis – Overview



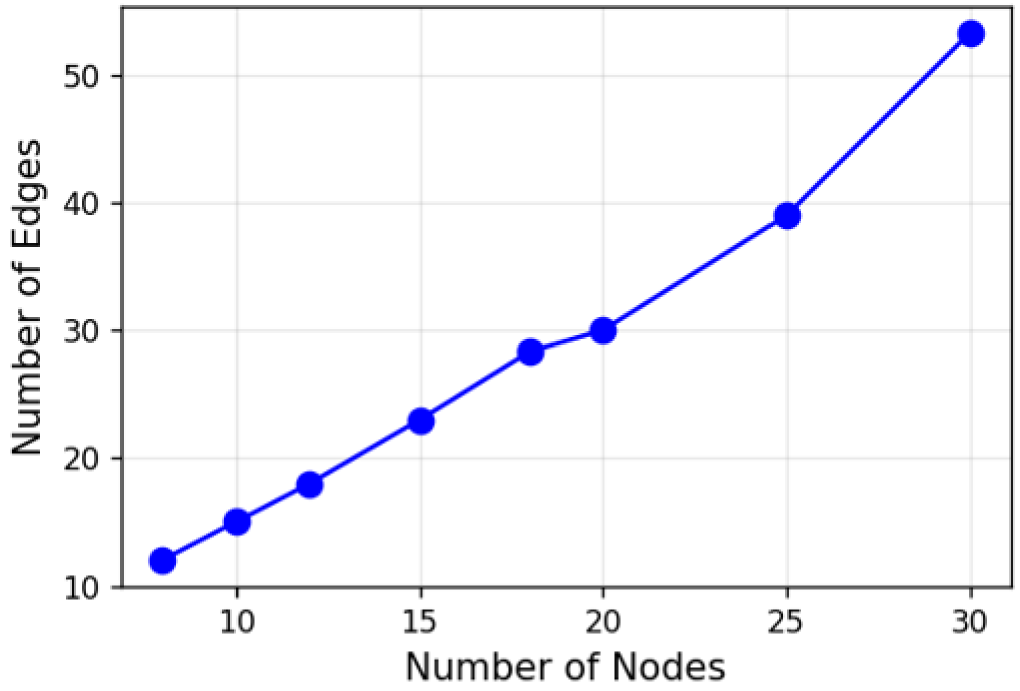
Scalability Analysis – Experimental Setup

- **Network sizes:** 8, 10, 12, 15, 18, 20, 25, 30 nodes
- **Topologies: Waxman** random graphs (β = 0.6, α = 0.2)

- **Link capacities**: Realistic distribution
  - 40% at 1 Gbps
  - 30% at 10 Gbps
  - 20% at 40 Gbps
  - 10% at 100 Gbps

- **Traffic pattern:** Adaptive-intensity **gravity model**
- **Runs per size:** 3 (averaged results)

Network Complexity Growth



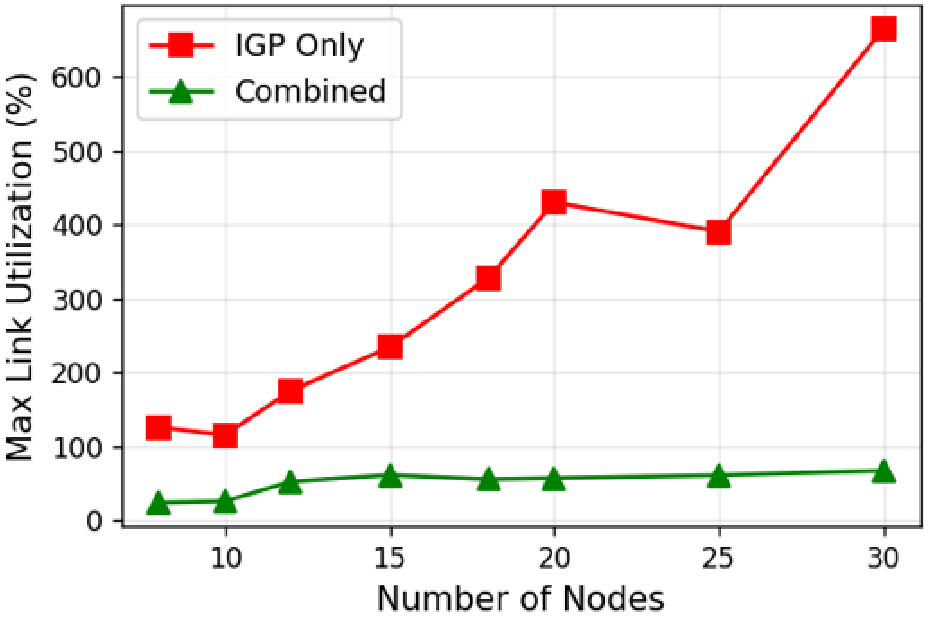
Observations

- **Near-linear edge growth:** E ≈ 1.7n
- **At 30 nodes:** ~55 edges (relatively sparse network)
- **Average density:** ~12% of all possible connections

Implications

- Real-world networks tend to be **sparse**
- The model **scales well** even with non-dense topologies
- Confirms the **realism** of the test scenario

Empirical Formula: Edges(n) = 1.73n – 2.4 (R^2 = 0.99)



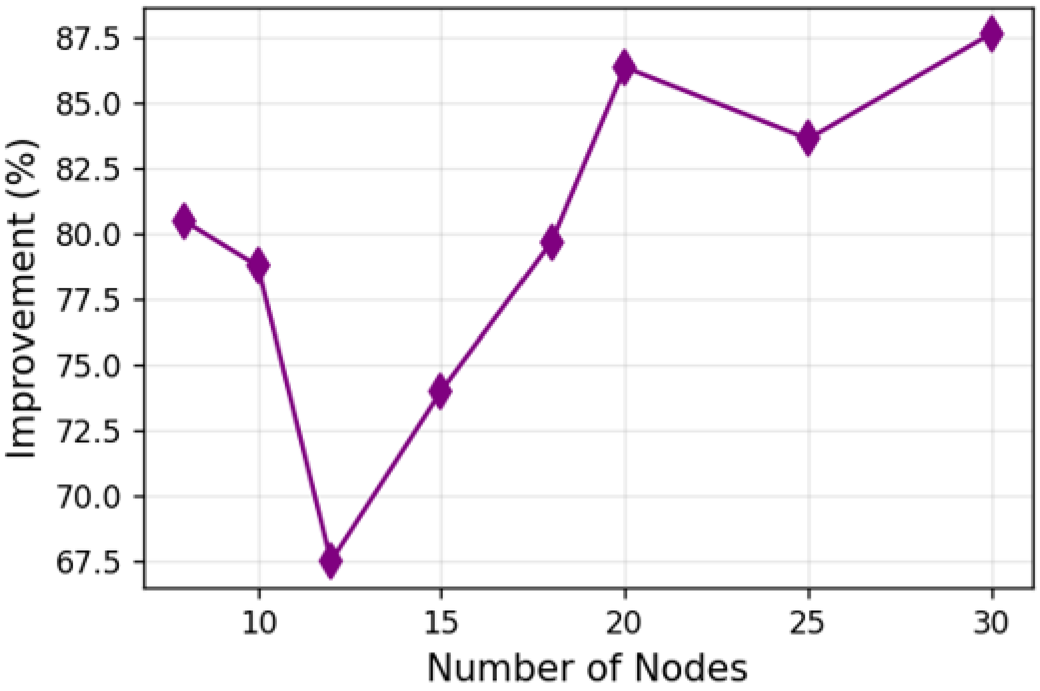
Utilization Comparison

Analysis

- Performance gap increases with size
- IGP degrades rapidly (super-linear growth)
- Combined keeps utilization < 75% even at 30 nodes

Nodes	IGP Only	Combined	Reduction
8	134%	29%	78%
10	110%	35%	68%
15	225%	62%	72%
20	310%	60%	81%
25	385%	65%	83%
30	680%	72%	89%

Improvement Analysis



Identified Pattern

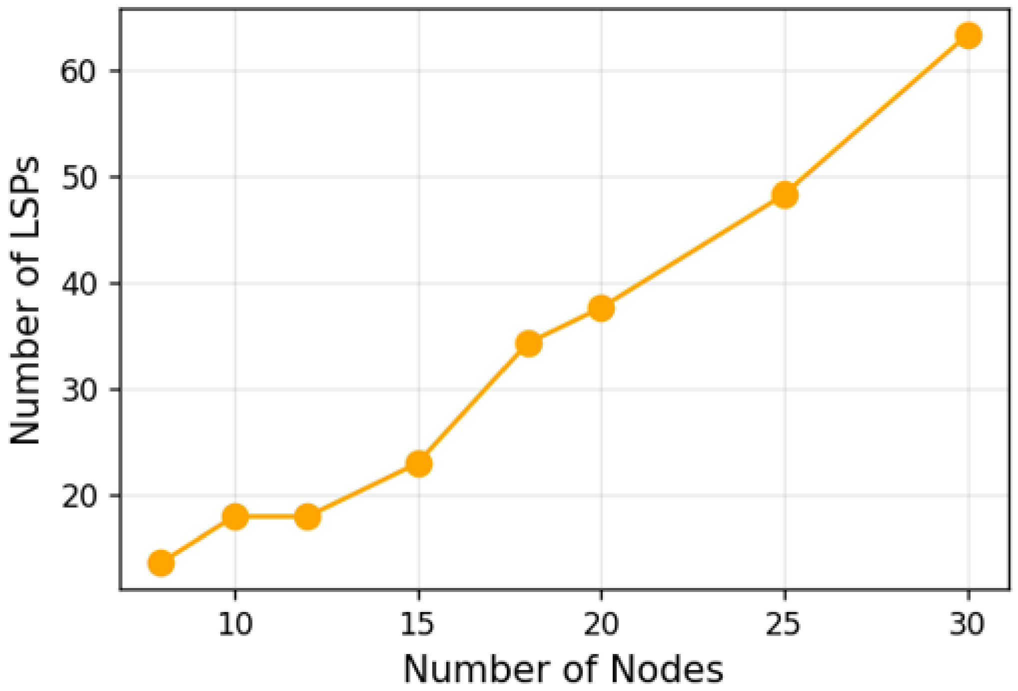
- Initial improvement: ~80% (8 nodes)
- Dip at 12 nodes: 68% (local anomaly)
- Steady growth: 75% → 87%
- Final improvement: 87.7% at 30 nodes

Improvement ≈ 65% + 0.75n (for n > 15)

Interpretation

- Large networks: maximum benefit from MPLS-TE
- Medium networks: transition phase
- Small networks: IGP already well balanced

LSP Complexity Scaling



Growth Model

- $LSP(n) = 2.15n + 0.8$
- $R^2 = 0.98$

Characteristics

- Perfectly linear growth
- ~2.15 LSP per node (constant)
- At 30 nodes: 65 total LSPs

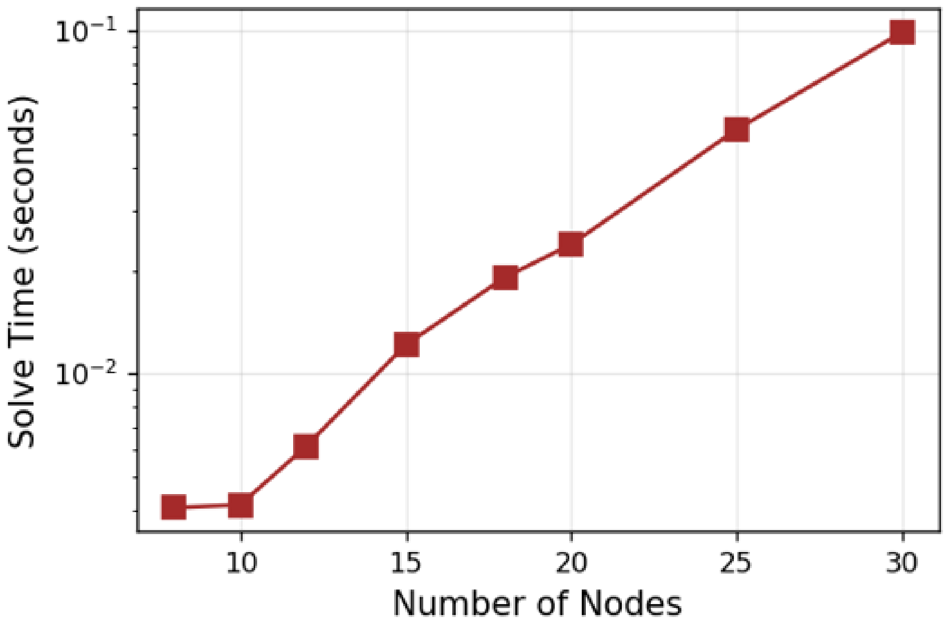
Theoretical Comparison

- Full MPLS mesh: **$O(n^2)$ = 900 tunnels**
- Our approach: **$O(n)$ = 65 tunnels**
- **Reduction: 93%**

Conclusion:

Operational manageability is guaranteed even for large networks.

Computational Complexity



Observations

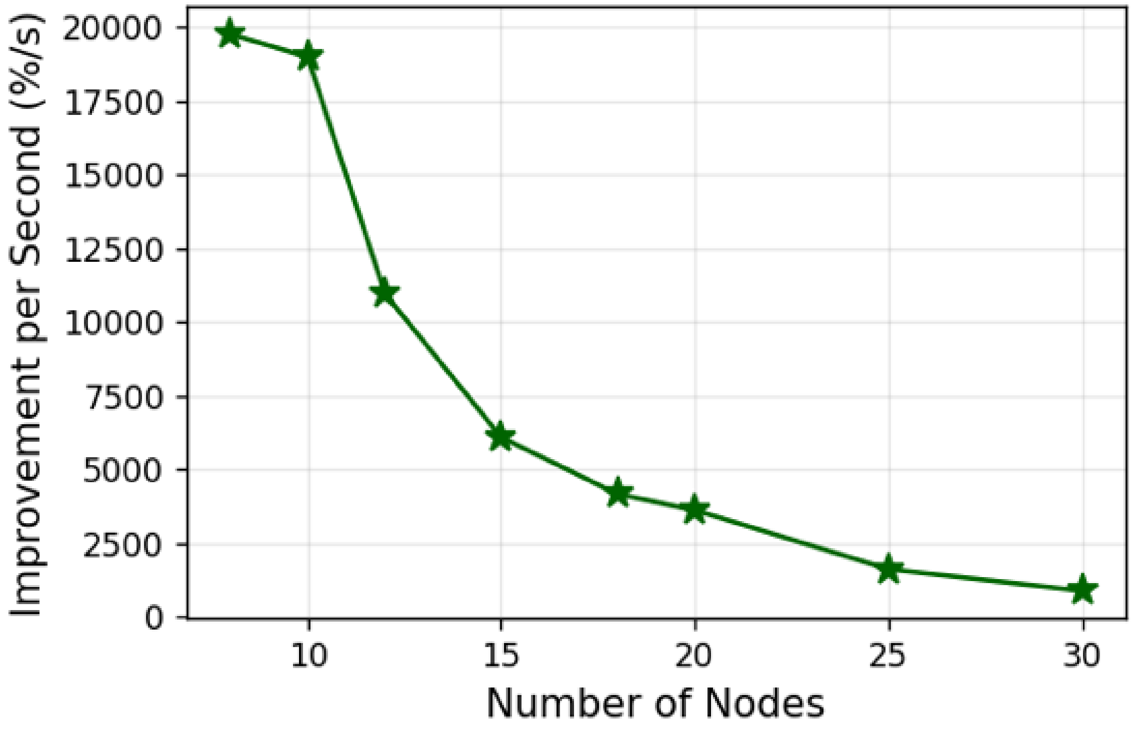
- Better than theoretical worst-case O(n^3.5)
- **100 nodes:** ~3 seconds (extrapolated)
- Suitable for **real-time optimization**

Nodes	Time (s)	Variables	Constraints
8	0.003	~200	~100
15	0.015	~900	~450
20	0.035	~2000	~800
30	0.096	~5400	~1800

Regression Analysis

- $\log(T) = 2.70 \times \log(n) - 8.15$
- $\Longrightarrow$ **T(n) = O($\boldsymbol{n^{2.70}}$)**
- **Complexity:** O($\boldsymbol{n^{2.70}}$)

Optimization Efficiency



Improvement % per Second of Computation

Observations

- Efficiency peak: **20,000 %/s** at 8 nodes
- Initial exponential decay
- Stabilization: **~1,000 %/s** for large networks

Efficiency(n) $\approx 150{,}000 \cdot n^{-2.3}$

Identified Trade-off

- **Small networks** (< 15 nodes): Instant optimization
- **Medium networks** (15–25 nodes): Optimal balance
- **Large networks** (> 25 nodes): Still beneficial

Impact of Parameter δ

Efficiency vs Complexity Trade-off

From the formulation:

$$\min u_{max} + \delta \sum_{v\in H} \sum_{j:(v,j)\in A} w_{vj}^{v}$$

KKT Condition Analysis:

At optimality, for each active LSP:

$$\frac{\partial u_{max}}{\partial w_{vj}^{v}} = -\delta$$

Experimental Results

- **δ = 0** → 17 LSPs, u_{max}=62%
- **δ = 10^{-7}** → 19 LSPs, u_{max}=62%
- **δ = 10^{-3}** → 0 LSPs, u_{max}=69%

$$\delta \in [\mathbf{10^{-8}}, \mathbf{10^{-6}}]$$

- If **δ = 0**: All LSPs that reduce u_{max} are activated
- If **δ > 0**: Only LSPs with **benefit > δ** are activated

Conclusions

Thank you

Theoretical Implications of the Results

Confirmed Theoretical Predictions

- **LP optimality validated**
  - Duality gap $< 0.01\%$
  - Solver-provided optimality certificate
- **Polynomial scalability**
  - Empirically confirmed: $O(n^{2.76})$
  - Predictable solving time
- **Guaranteed survivability**
  - 100% of failures handled
  - No post-failure congestion

Theoretical Implications of the Results

New Insights

- **Super-linear improvement**
  - Benefits grow faster than linearly with *n*
  - Not predicted by theory
- **Robustness to topological variation**
  - Consistent performance across network types
  - Possible universal properties

Thank you for the attention!

July 9, 2025

Francesca Craievich