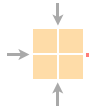
ETH zürich



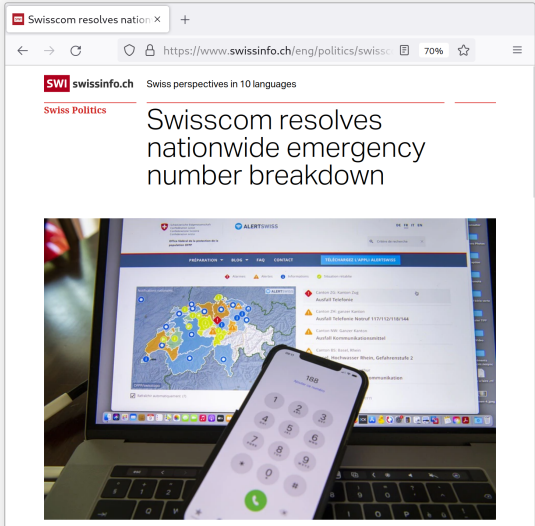
Networked Systems
ETH Zürich — seit 2015

On the Complexity of Network-Wide Configuration Synthesis

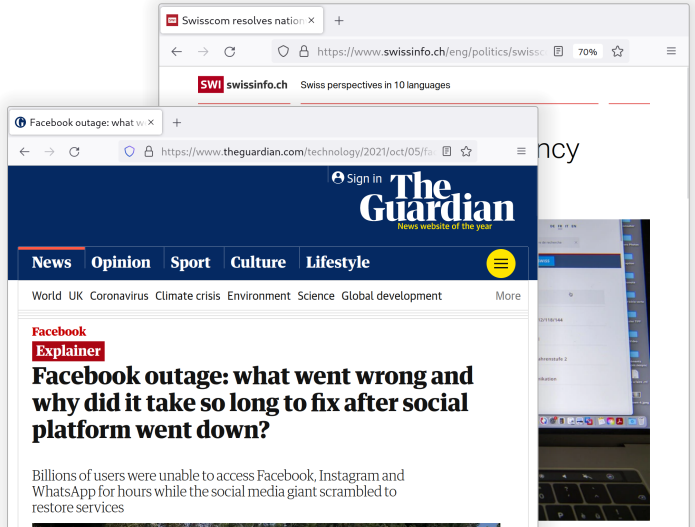
Tibor Schneider, Roland Schmid, Laurent Vanbever

IEEE ICNP 2022, November 2, 2022

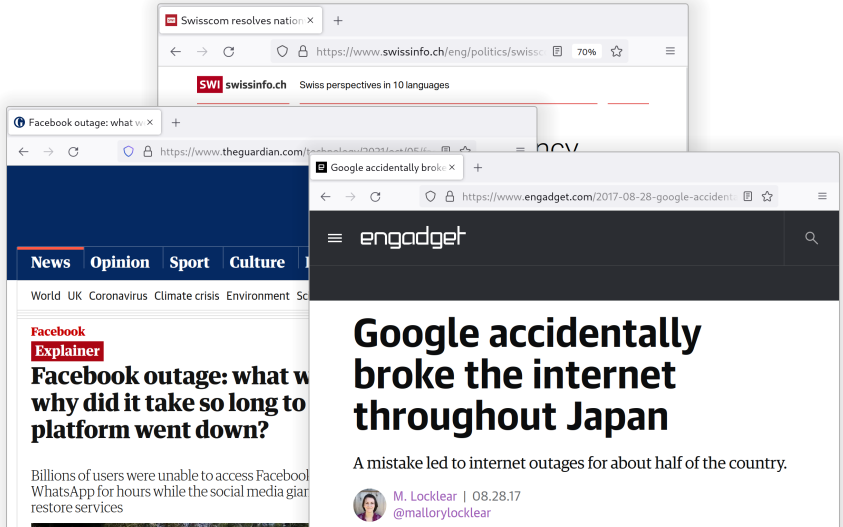
Network outages are very common.



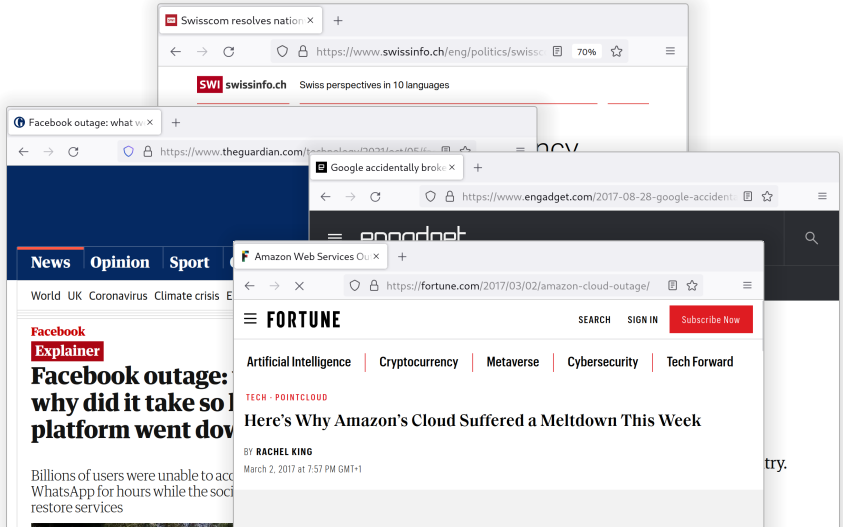
Network outages are very common.



Network outages are very common.



Network outages are very common.



Network verification can help preventing such mistakes.

Idea *Verify configuration **before** deployment.*

Network verification can help preventing such mistakes.

Idea *Verify configuration **before** deployment.*

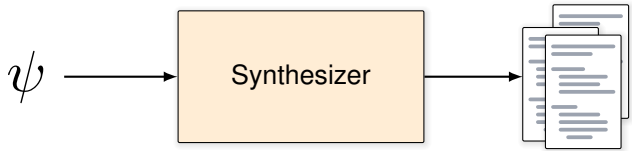
Problem *You still need to **find** a valid configuration.*

Network verification can help preventing such mistakes.

Idea *Verify configuration **before** deployment.*

Problem *You still need to **find** a valid configuration.*

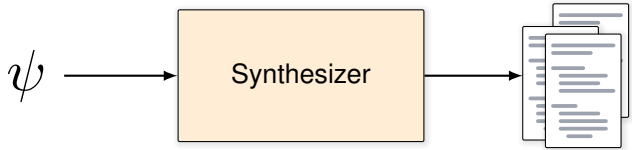
Solution ***Configuration Synthesis***


ψ
Synthesizer

Reachability

Access control

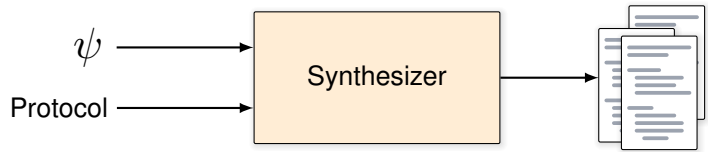
Traffic optimization



Reachability

Access control

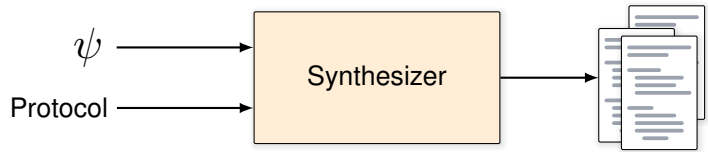
Traffic optimization



Reachability

Access control

Traffic optimization



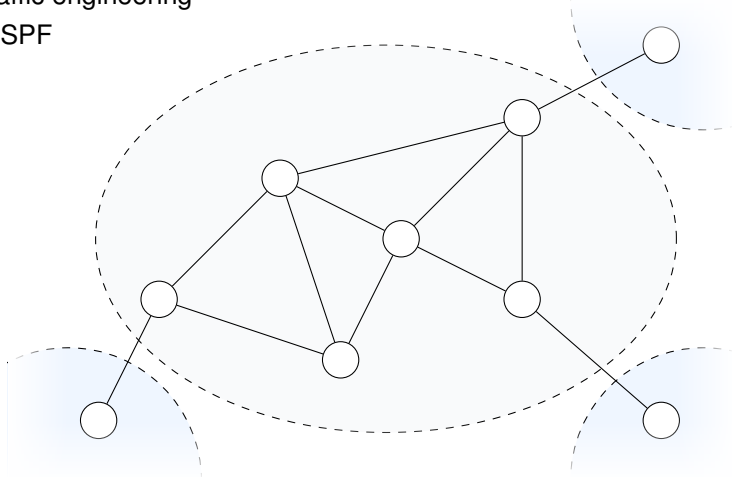
OSPF

OSPF + BGP

Static Routes

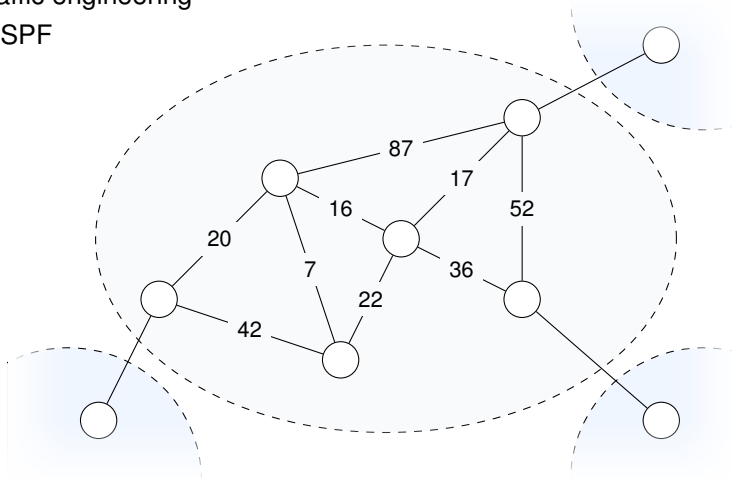
Specification: traffic engineering

Protocol: OSPF



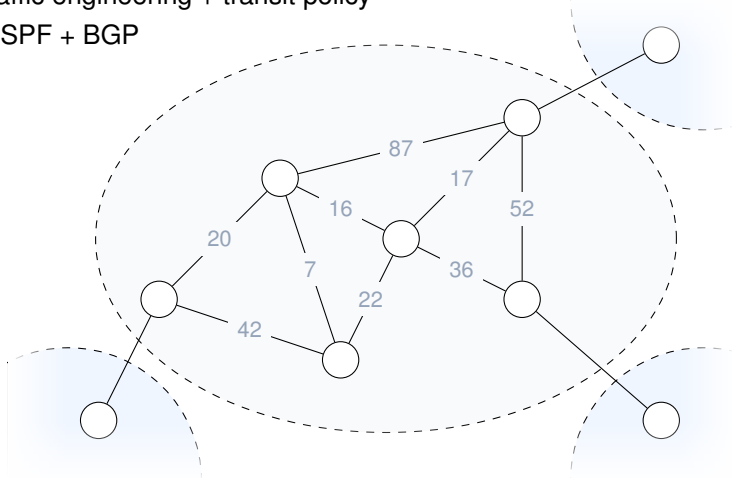
Specification: traffic engineering

Protocol: OSPF



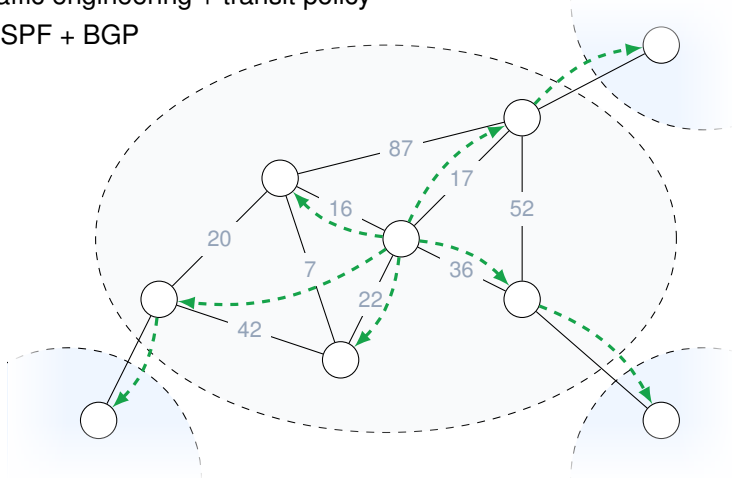
Specification: traffic engineering + transit policy

Protocol: OSPF + BGP



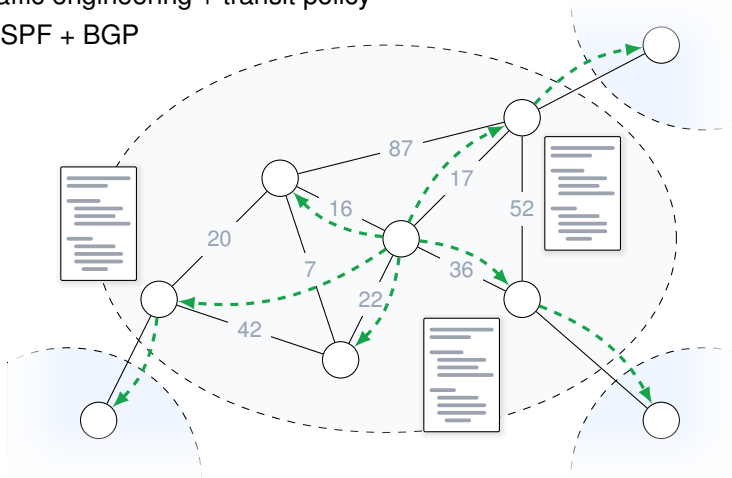
Specification: traffic engineering + transit policy

Protocol: OSPF + BGP



Specification: traffic engineering + transit policy

Protocol: OSPF + BGP



Configuration synthesizers already exist.

SyNet CAV'17

A. El-Hassany et al. "Network-wide configuration synthesis". *CAV*. 2017

Configuration synthesizers already exist.

SyNet CAV'17

A. El-Hassany et al. "Network-wide configuration synthesis". *CAV*. 2017

Propane/AT SIGCOMM'16, PLDI'17

R. Beckett et al. "Don't mind the gap: Bridging network-wide objectives and device-level configurations". *SIGCOMM*. 2016

R. Beckett et al. "Network Configuration Synthesis with Abstract Topologies". *PLDI* (2017)

Configuration synthesizers already exist.

SyNet CAV'17

A. El-Hassany et al. "Network-wide configuration synthesis". *CAV*. 2017

Propane/AT SIGCOMM'16, PLDI'17

R. Beckett et al. "Don't mind the gap: Bridging network-wide objectives and device-level configurations". *SIGCOMM*. 2016

R. Beckett et al. "Network Configuration Synthesis with Abstract Topologies". *PLDI* (2017)

Zeppelin SIGMETRIC'18

K. Subramanian et al. "Synthesis of fault-tolerant distributed router configurations". *SIGMETRICS*. 2018

Why do some systems scale better than others?

SyNet CAV'17

A. El-Hassany et al. "Network-wide configuration synthesis". *CAV*. 2017

Propane/AT SIGCOMM'16, PLDI'17

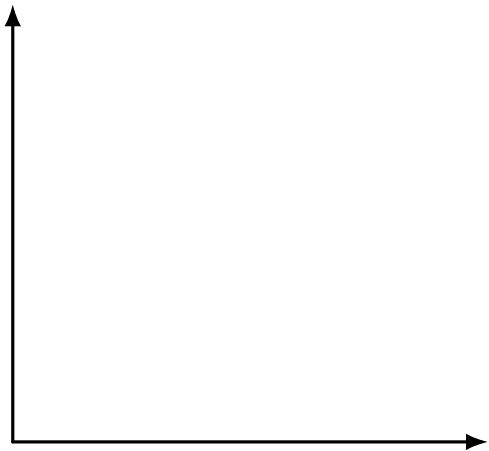
R. Beckett et al. "Don't mind the gap: Bridging network-wide objectives and device-level configurations". *SIGCOMM*. 2016

R. Beckett et al. "Network Configuration Synthesis with Abstract Topologies". *PLDI* (2017)

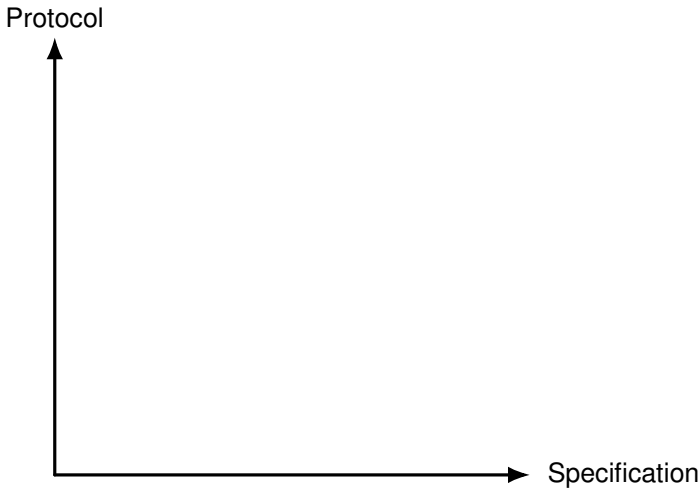
Zeppelin SIGMETRIC'18

K. Subramanian et al. "Synthesis of fault-tolerant distributed router configurations". *SIGMETRICS*. 2018

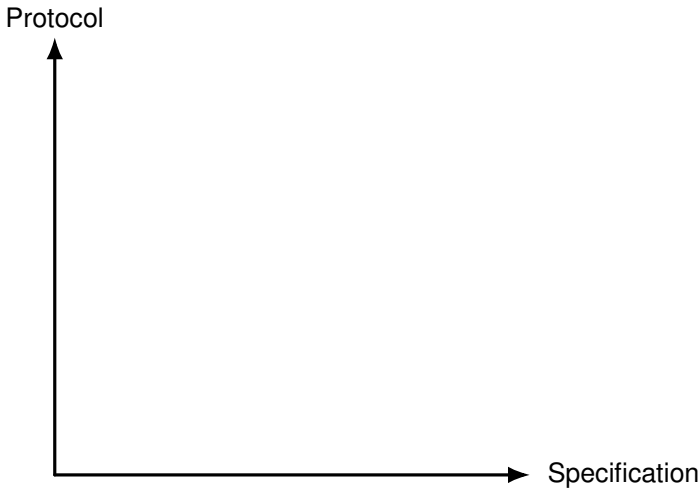
Scalability depends on both the specification and the protocol.



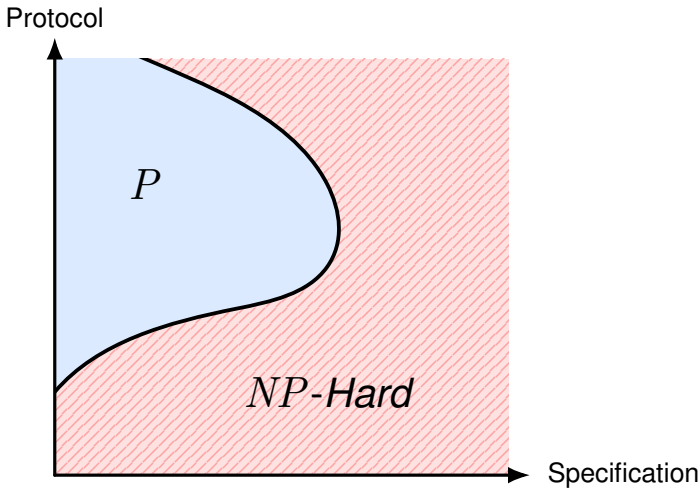
Scalability depends on both the specification and the protocol.



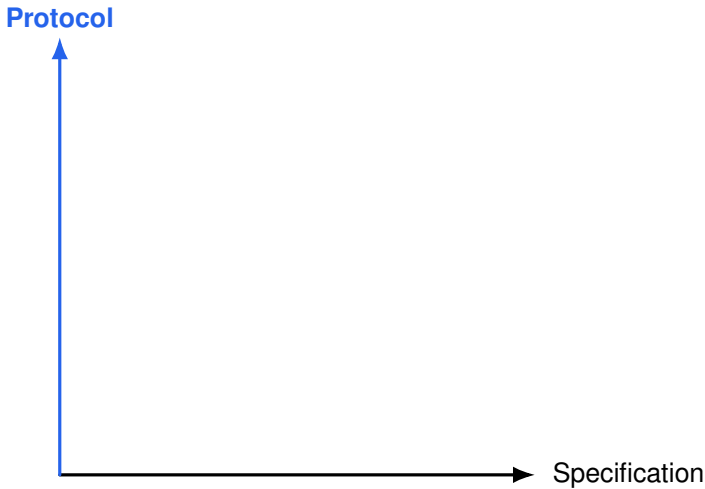
Scalability depends on both the specification and the protocol.



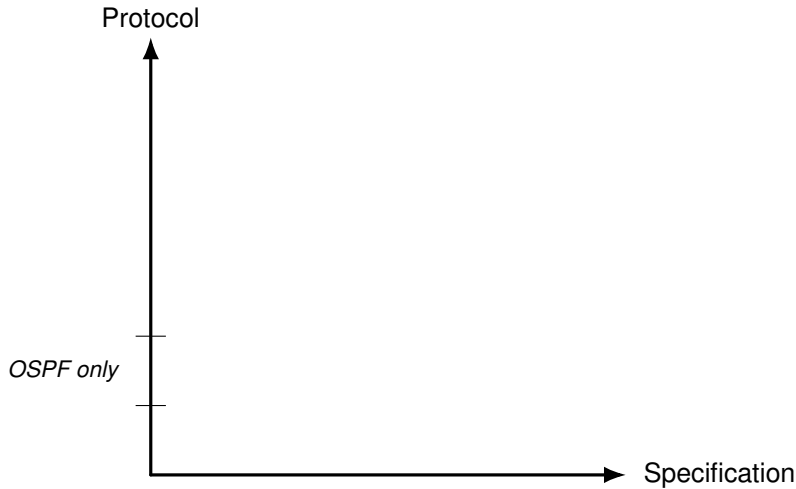
We explore the computational complexity of configuration synthesis.



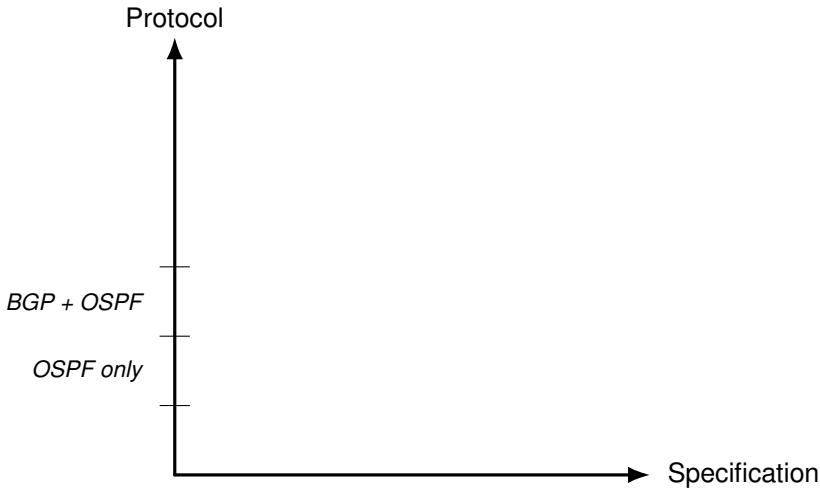
How do we define the protocol axis?



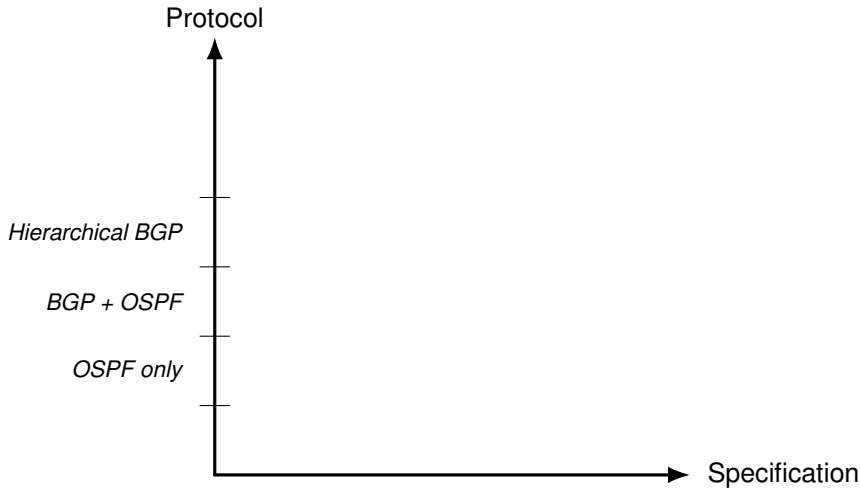
How do we define the protocol axis?



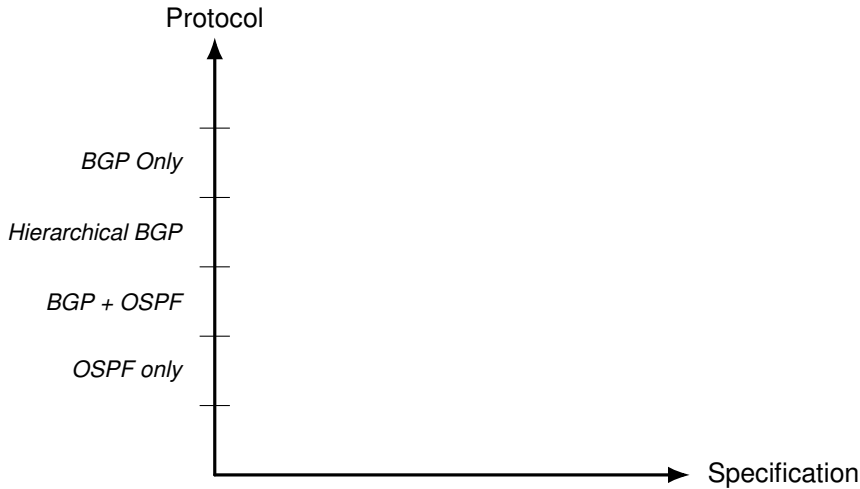
How do we define the protocol axis?



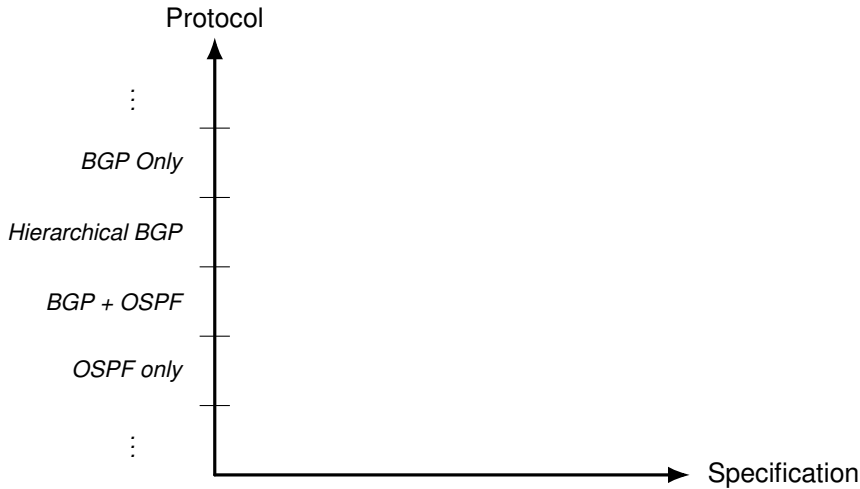
How do we define the protocol axis?



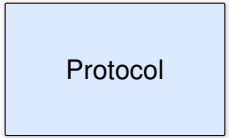
How do we define the protocol axis?



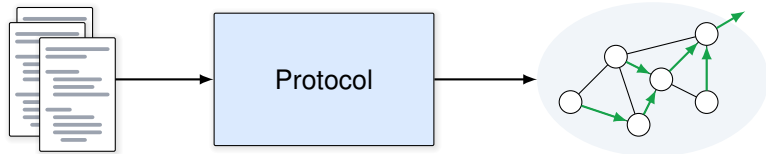
What about other (future) protocols?



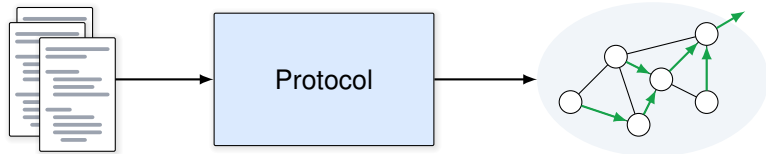
We need a system for characterizing protocols.



We need a system for characterizing protocols.

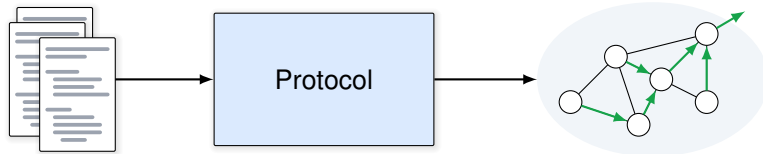


We need a system for characterizing protocols.



Min-Hop

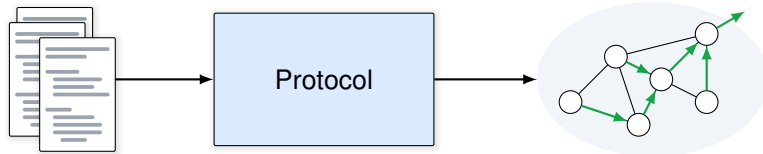
We need a system for characterizing protocols.



Min-Hop

OSPF

We need a system for characterizing protocols.

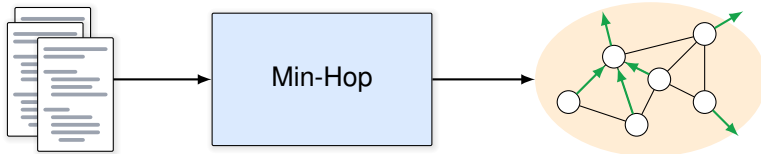


Min-Hop

OSPF

BGP

Protocols can represent different forwarding states.

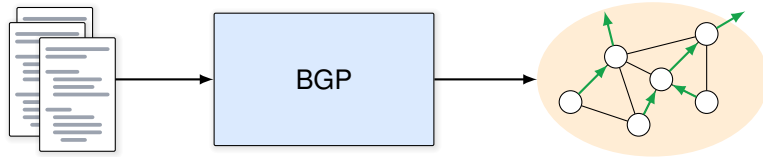


Min-Hop *Single forwarding state.*

OSPF

BGP

Protocols can represent different forwarding states.

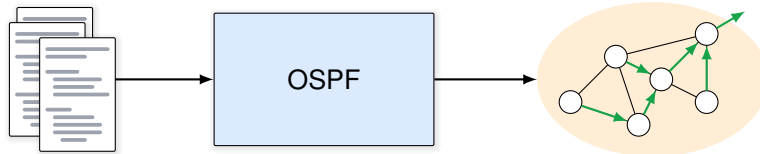


Min-Hop *Single forwarding state.*

OSPF

BGP *Arbitrary forwarding states.*

The Expressivity measures the number of forwarding states.



Min-Hop *Single forwarding state.*

OSPF *Suboptimality of shortest paths.*

BGP *Arbitrary forwarding states.*

We capture this expressivity of protocols by defining properties.

Linearity *As expressive as Shortest-Path routing*

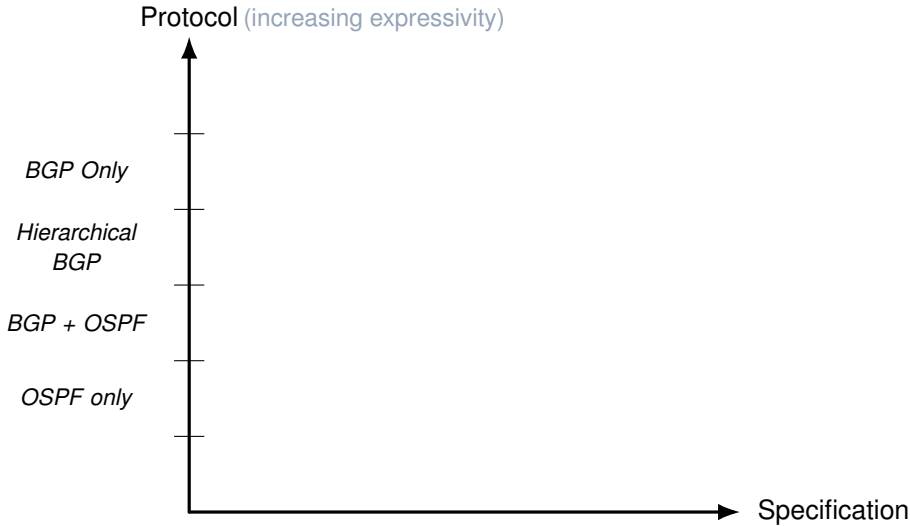
We capture this expressivity of protocols by defining properties.

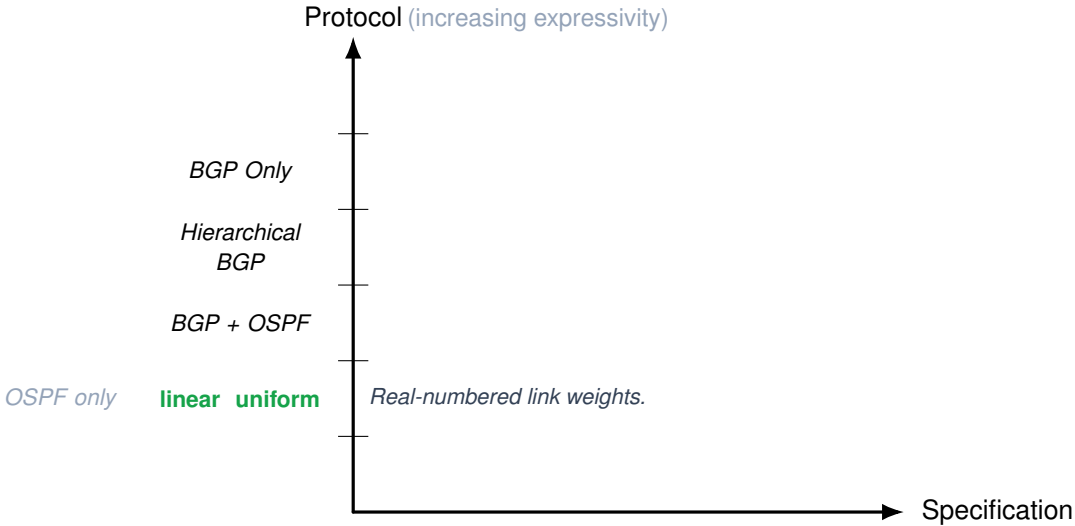
Linearity *As expressive as Shortest-Path routing*

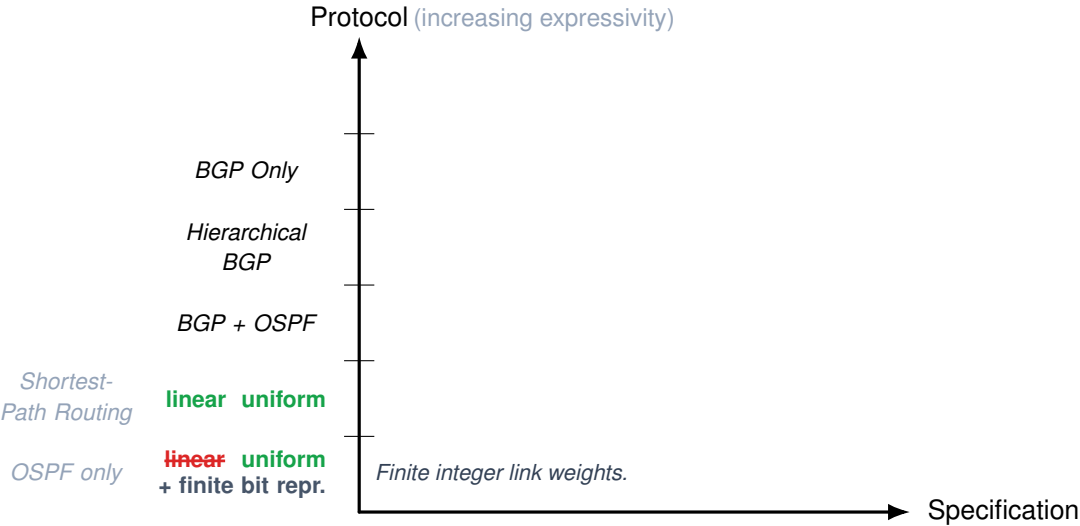
Uniformity *All destination prefixes are treated equally.*

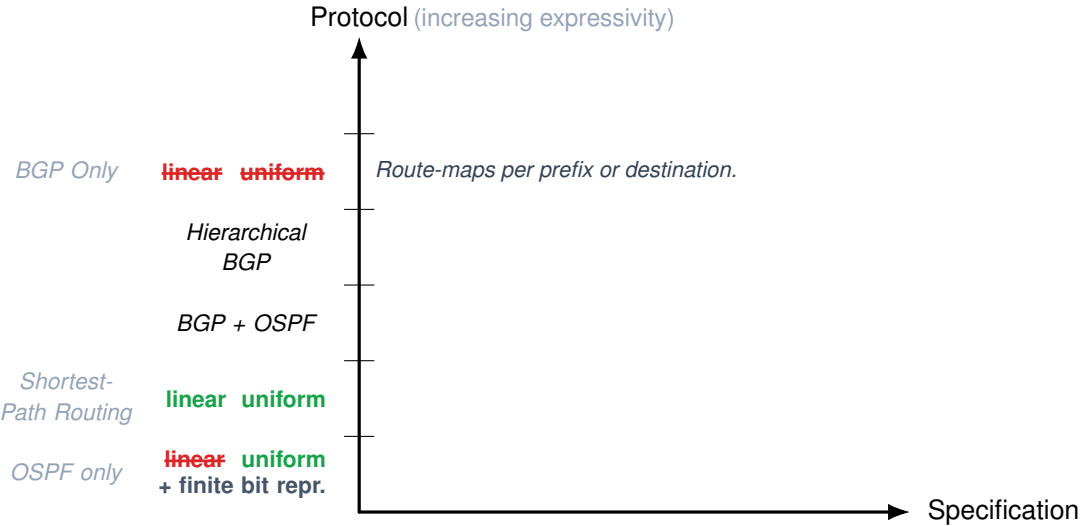
We capture this expressivity of protocols by defining properties.

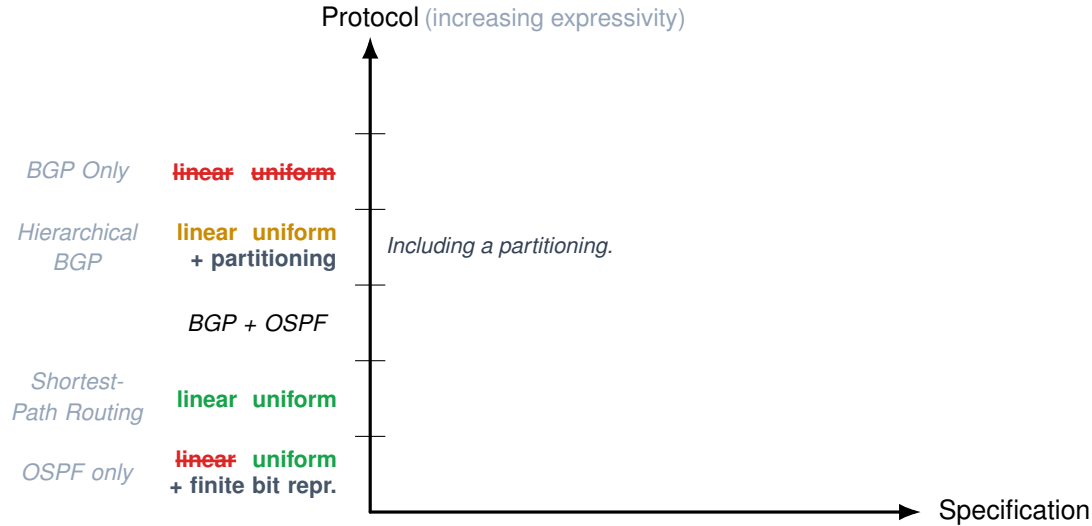
Linearity	*As expressive as Shortest-Path routing*
Uniformity	*All destination prefixes are treated equally.*
Filtering	*Specific links can be disabled.*

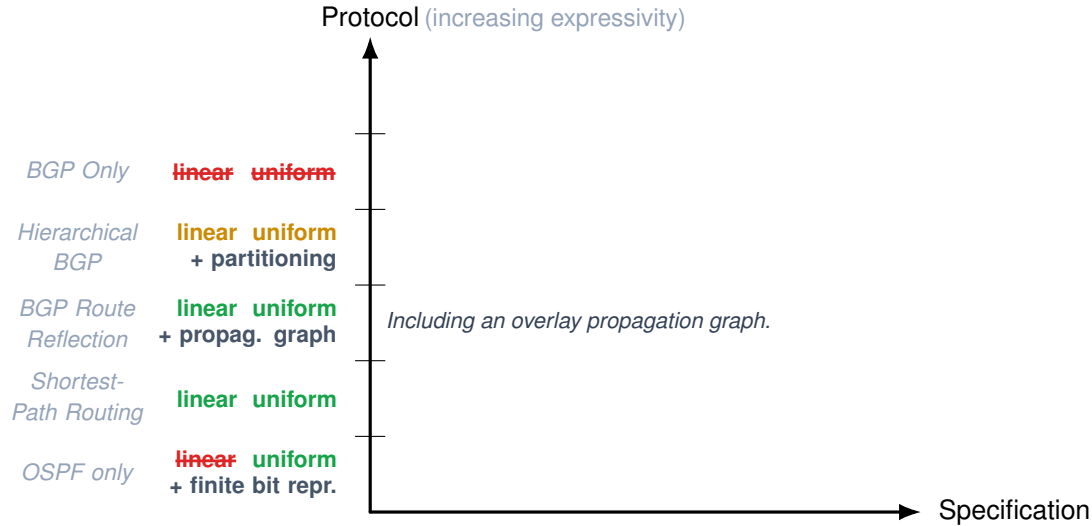

Protocol (increasing expressivity)
BGP Only
Hierarchical BGP
BGP + OSPF
OSPF only
Specification

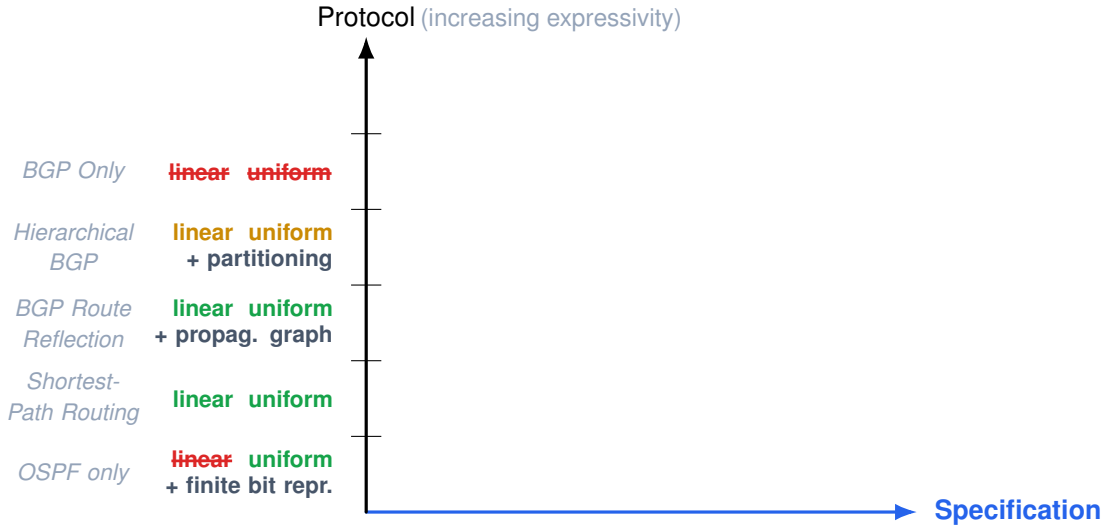

Protocol (increasing expressivity)
BGP Only
Hierarchical BGP
BGP + OSPF
OSPF only
linear uniform
Real-numbered link weights.
Specification


Protocol (increasing expressivity)
BGP Only
Hierarchical BGP
BGP + OSPF
Shortest-Path Routing
linear uniform
OSPF only
~~linear~~ uniform + finite bit repr.
Finite integer link weights.
Specification

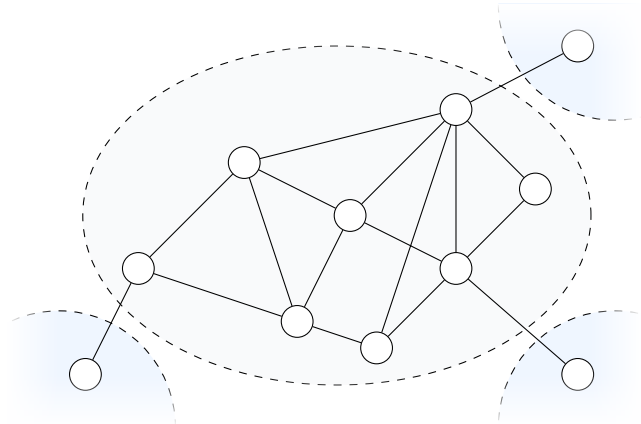

Protocol (increasing expressivity)
BGP Only
linear uniform
Route-maps per prefix or destination.
Hierarchical BGP
BGP + OSPF
Shortest-Path Routing
linear uniform
OSPF only
linear uniform + finite bit repr.
Specification


Protocol (increasing expressivity)
BGP Only
linear uniform
Hierarchical BGP
linear uniform
+ partitioning
Including a partitioning.
BGP + OSPF
Shortest-Path Routing
linear uniform
OSPF only
linear uniform
+ finite bit repr.
Specification


Protocol (increasing expressivity)
BGP Only
linear uniform
Hierarchical BGP
linear uniform
+ partitioning
BGP Route Reflection
linear uniform
+ propag. graph
Including an overlay propagation graph.
Shortest-Path Routing
linear uniform
OSPF only
linear uniform
+ finite bit repr.
Specification

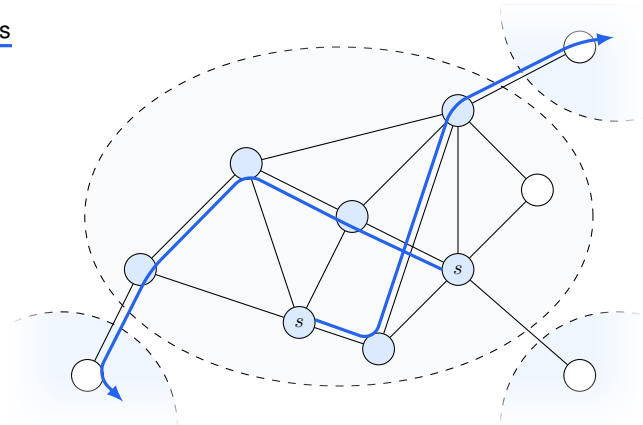

Protocol (increasing expressivity)
BGP Only
linear uniform
Hierarchical BGP
linear uniform
+ partitioning
BGP Route Reflection
linear uniform
+ propag. graph
Shortest-Path Routing
linear uniform
OSPF only
linear uniform
+ finite bit repr.
Specification

We focus on *fundamental* forwarding properties.



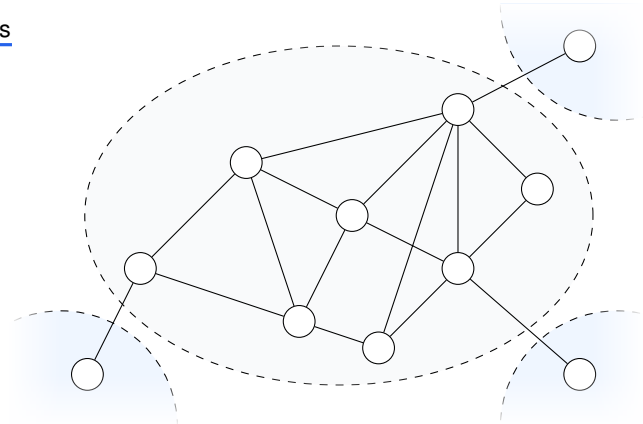
We focus on *fundamental* forwarding properties.

Paths

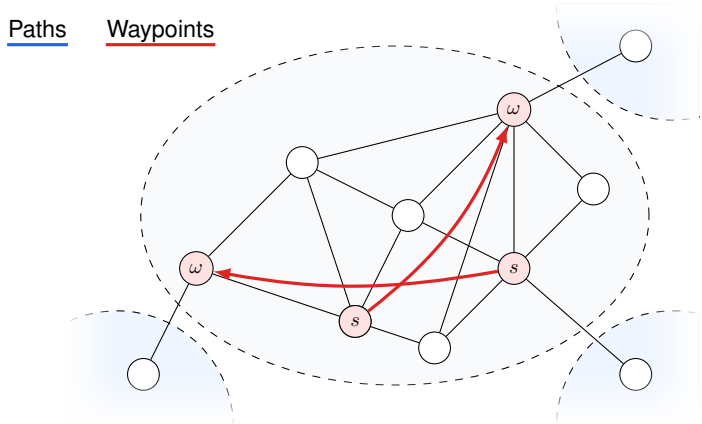


We focus on *fundamental* forwarding properties.

Paths

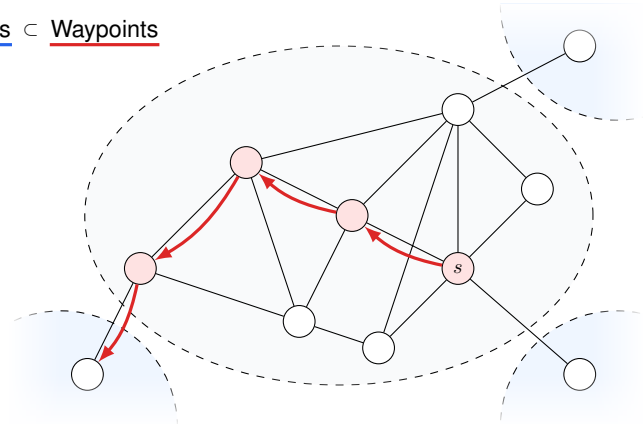


We focus on *fundamental* forwarding properties.

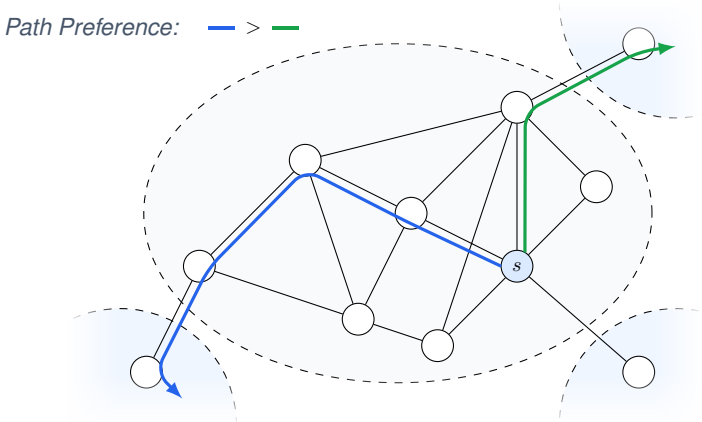


We focus on *fundamental* forwarding properties.

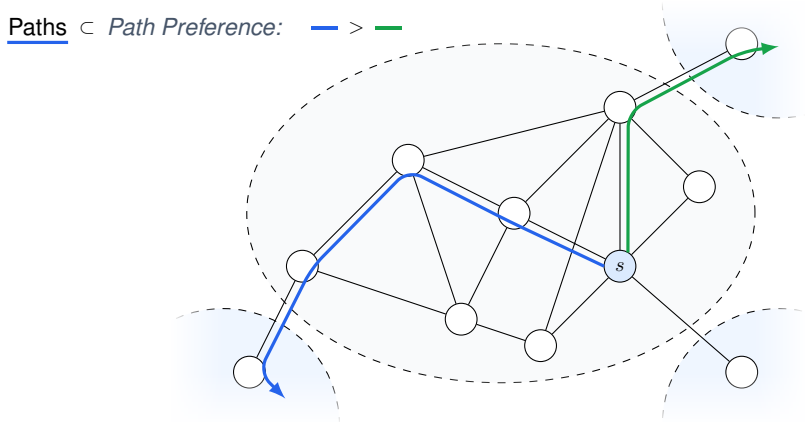
Paths ⊂ Waypoints

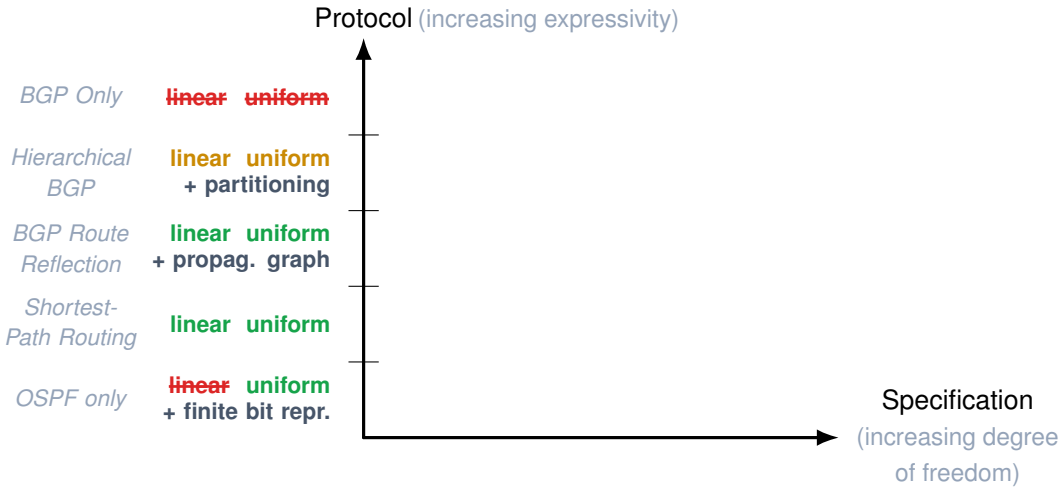


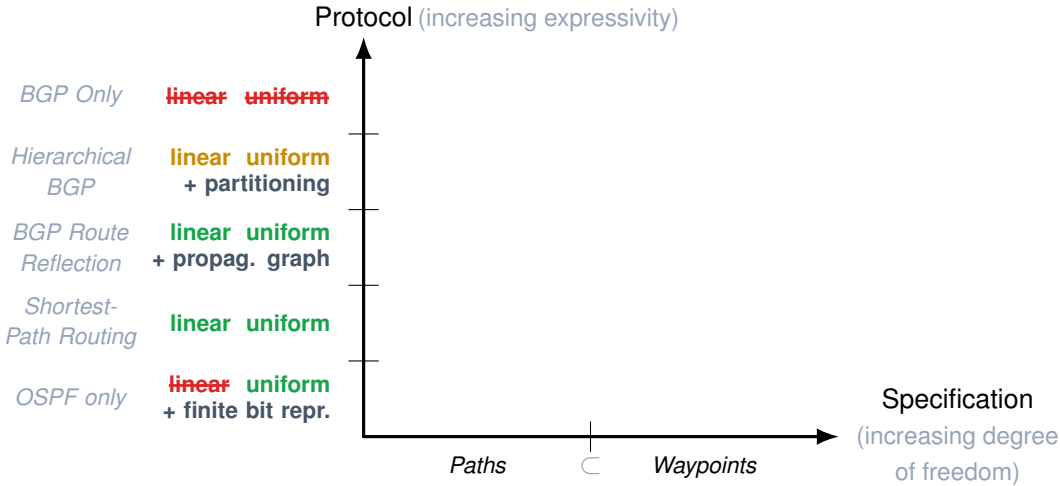
Paths and Waypoint properties are the basis for other properties.

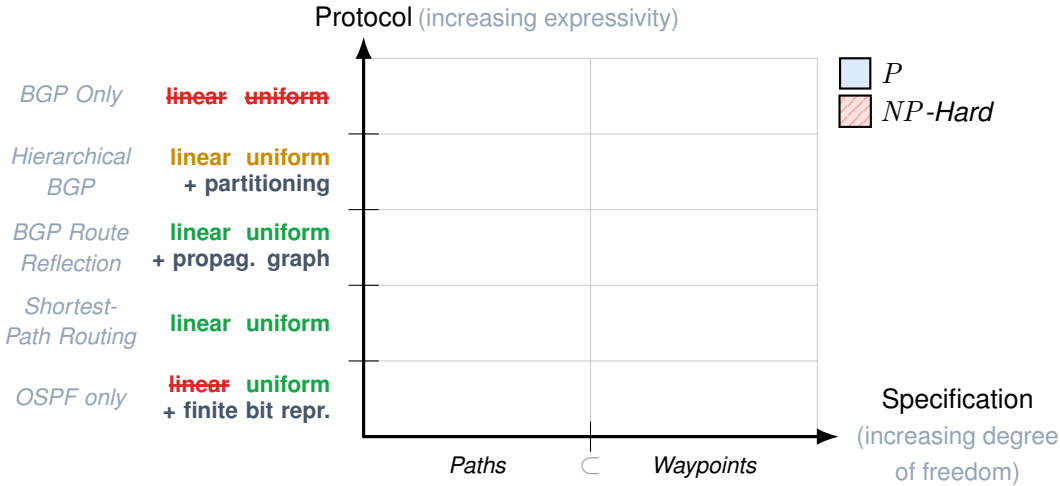


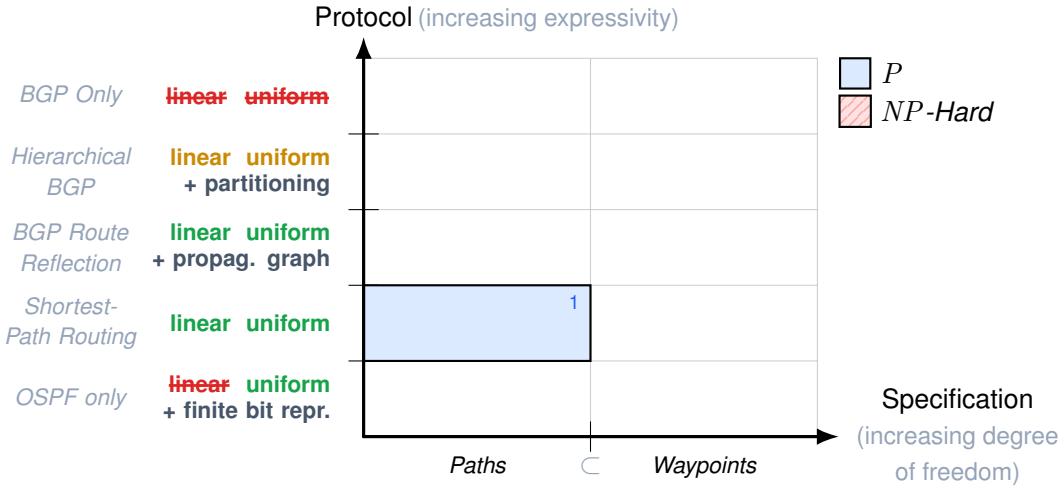
Paths and Waypoint properties are the basis for other properties.



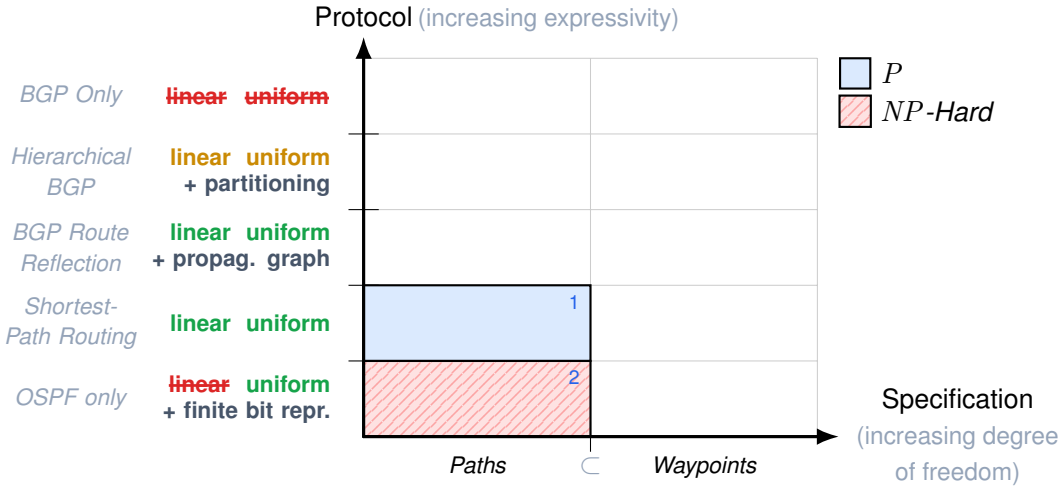

Protocol (increasing expressivity)
BGP Only
linear uniform
Hierarchical BGP
linear uniform
+ partitioning
BGP Route Reflection
linear uniform
+ propag. graph
Shortest-Path Routing
linear uniform
OSPF only
linear uniform
+ finite bit repr.
Specification
(increasing degree of freedom)


Protocol (increasing expressivity)
BGP Only
linear uniform
Hierarchical BGP
linear uniform
+ partitioning
BGP Route Reflection
linear uniform
+ propag. graph
Shortest-Path Routing
linear uniform
OSPF only
linear uniform
+ finite bit repr.
Specification
(increasing degree of freedom)
Paths
⊂
Waypoints


Protocol (increasing expressivity)
BGP Only
linear uniform
Hierarchical BGP
linear uniform + partitioning
BGP Route Reflection
linear uniform + propag. graph
Shortest-Path Routing
linear uniform
OSPF only
linear uniform + finite bit repr.
P
NP-Hard
Specification (increasing degree of freedom)
Paths
Waypoints

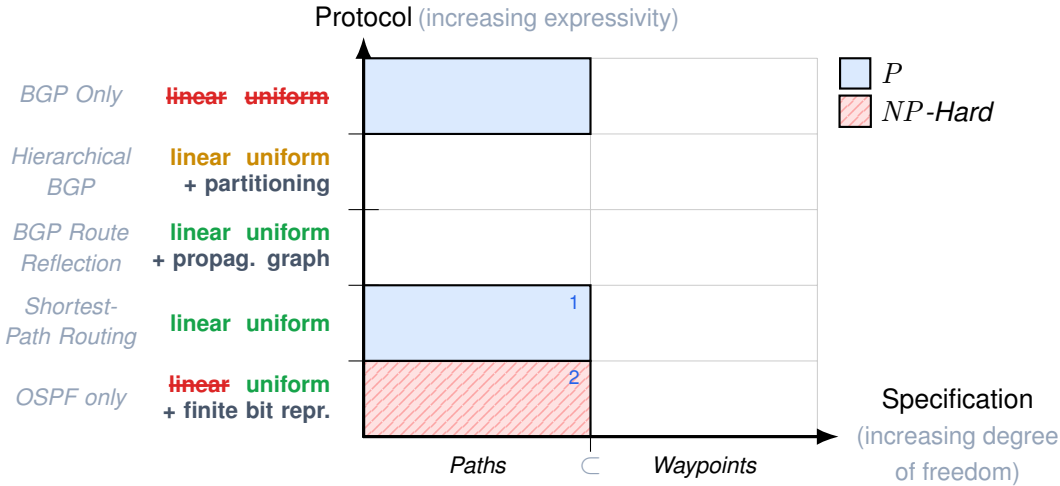


[1] W. Ben-Ameur et al. "Internet routing and related topology issues". *SIDMA* (2003)



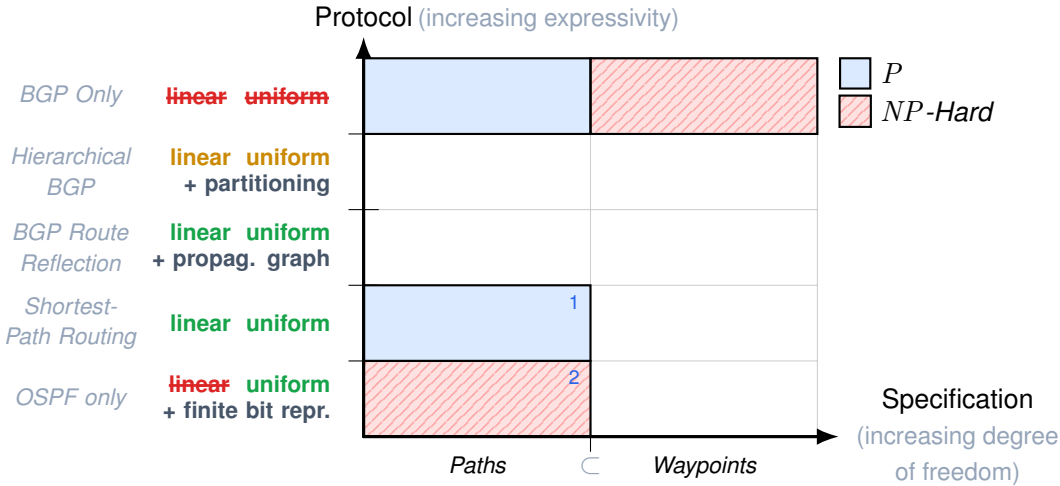
[1] W. Ben-Ameur et al. "Internet routing and related topology issues". *SIDMA* (2003)

[2] A. Bley. "Inapproximability results for the inverse shortest paths problem with integer lengths and unique shortest paths". *Networks* (2007)



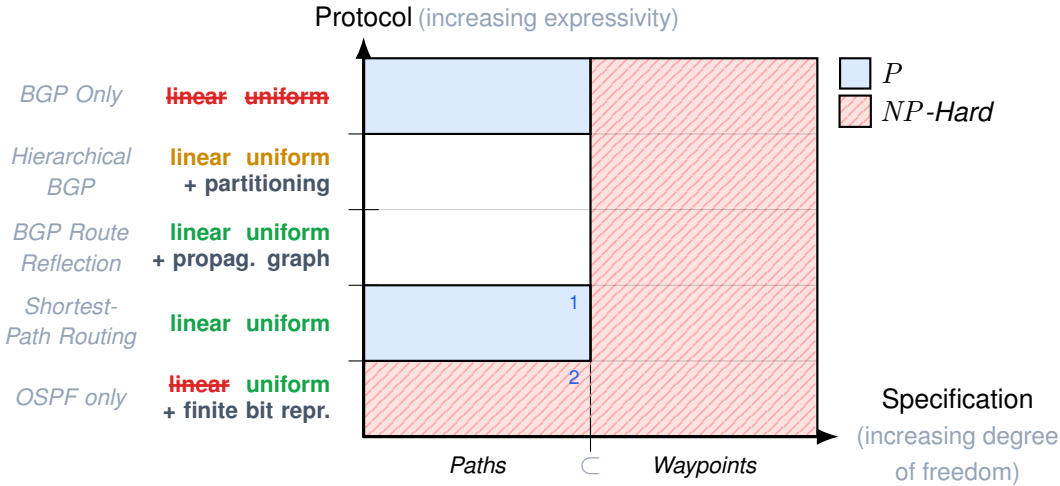
[1] W. Ben-Ameur et al. "Internet routing and related topology issues". *SIDMA* (2003)

[2] A. Bley. "Inapproximability results for the inverse shortest paths problem with integer lengths and unique shortest paths". *Networks* (2007)



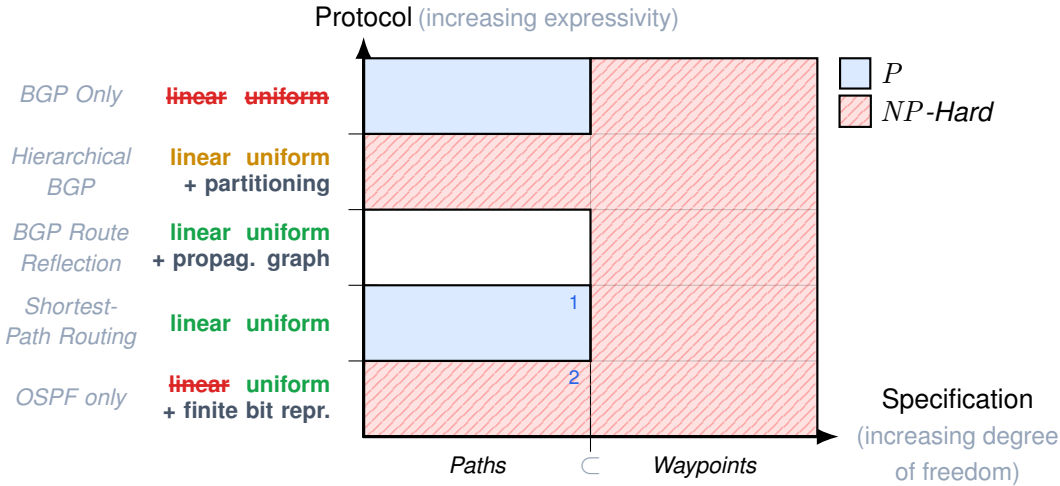
[1] W. Ben-Ameur et al. "Internet routing and related topology issues". *SIDMA* (2003)

[2] A. Bley. "Inapproximability results for the inverse shortest paths problem with integer lengths and unique shortest paths". *Networks* (2007)



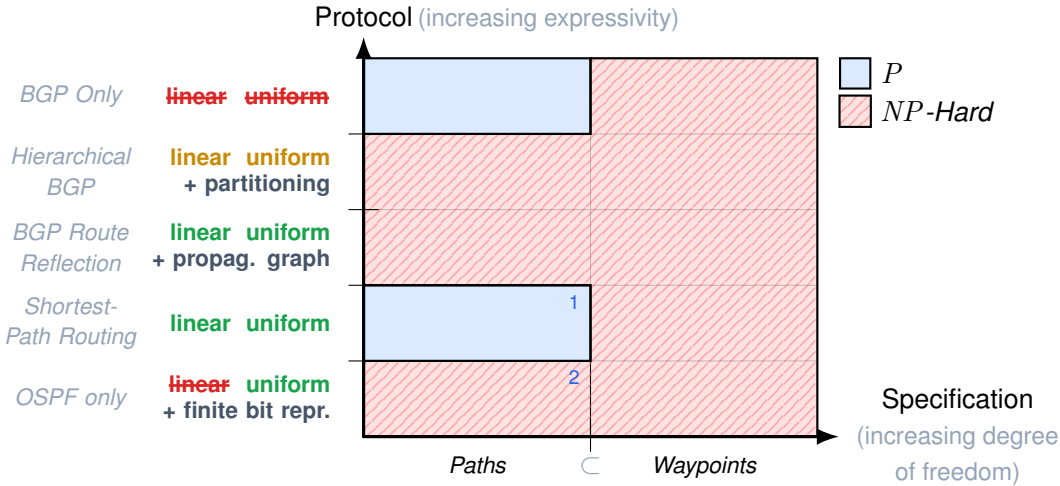
[1] W. Ben-Ameur et al. "Internet routing and related topology issues". *SIDMA* (2003)

[2] A. Bley. "Inapproximability results for the inverse shortest paths problem with integer lengths and unique shortest paths". *Networks* (2007)



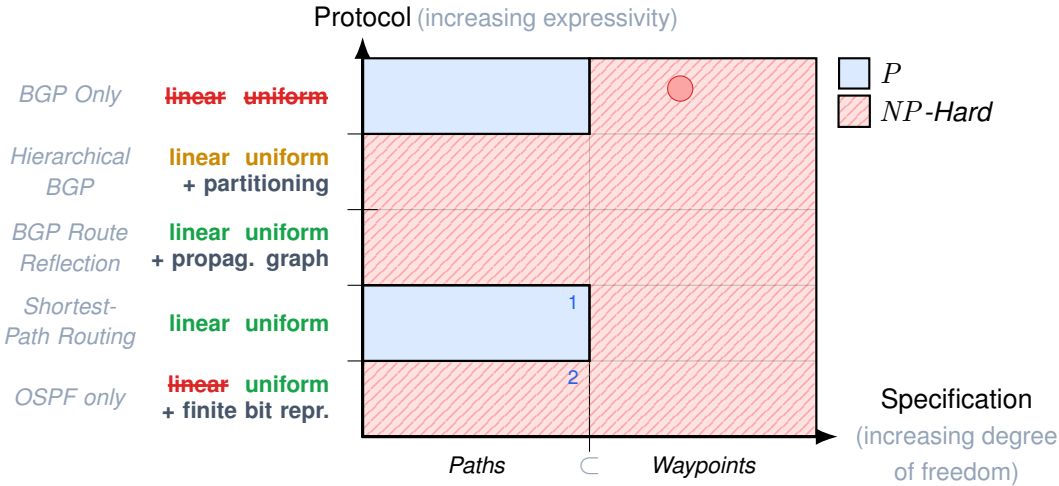
[1] W. Ben-Ameur et al. "Internet routing and related topology issues". *SIDMA* (2003)

[2] A. Bley. "Inapproximability results for the inverse shortest paths problem with integer lengths and unique shortest paths". *Networks* (2007)



[1] W. Ben-Ameur et al. "Internet routing and related topology issues". *SIDMA* (2003)

[2] A. Bley. "Inapproximability results for the inverse shortest paths problem with integer lengths and unique shortest paths". *Networks* (2007)



[1] W. Ben-Ameur et al. "Internet routing and related topology issues". *SIDMA* (2003)

[2] A. Bley. "Inapproximability results for the inverse shortest paths problem with integer lengths and unique shortest paths". *Networks* (2007)

What does this mean?

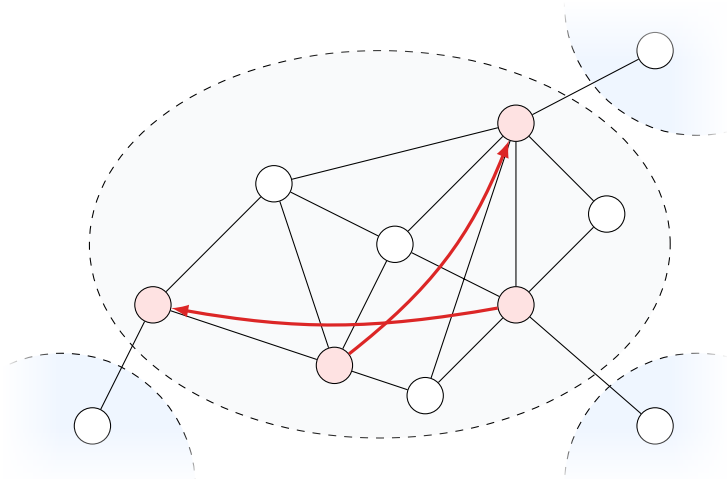
There **cannot** exist an efficient algorithm to solve **every** problem unless $P = NP$.

What does this mean?

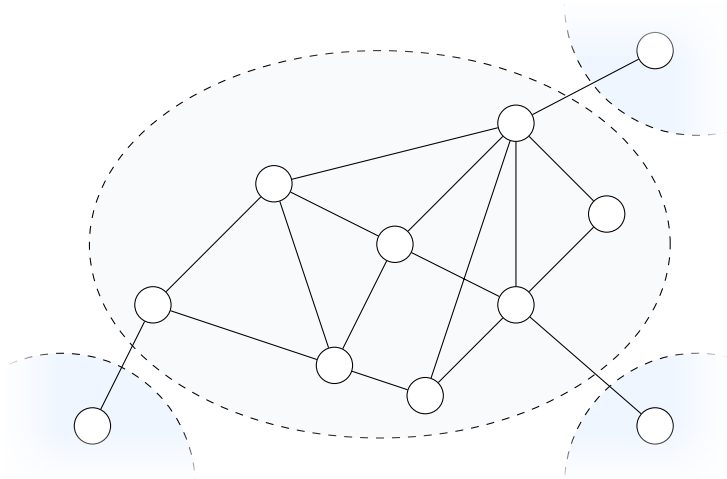
There **cannot** exist an efficient algorithm to solve **every** problem unless $P = NP$.

But, there **might** exist an efficient algorithm to solve **some** problems.

Waypoint properties are usually structured.



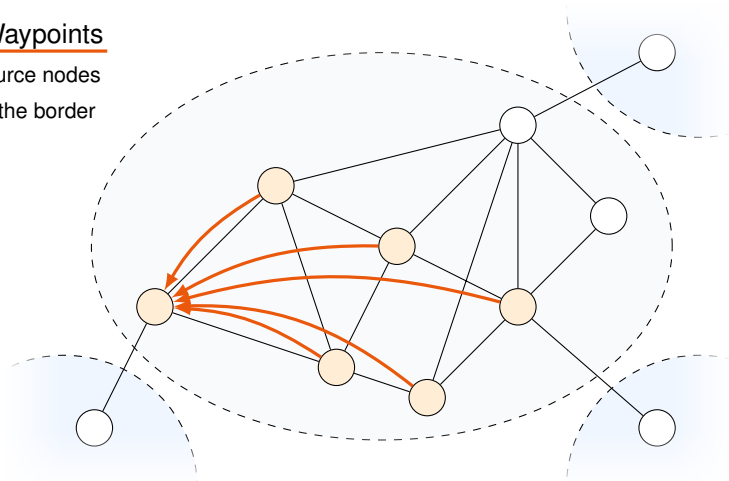
Waypoint properties are usually structured.



Waypoint properties are usually structured.

Connected Waypoints

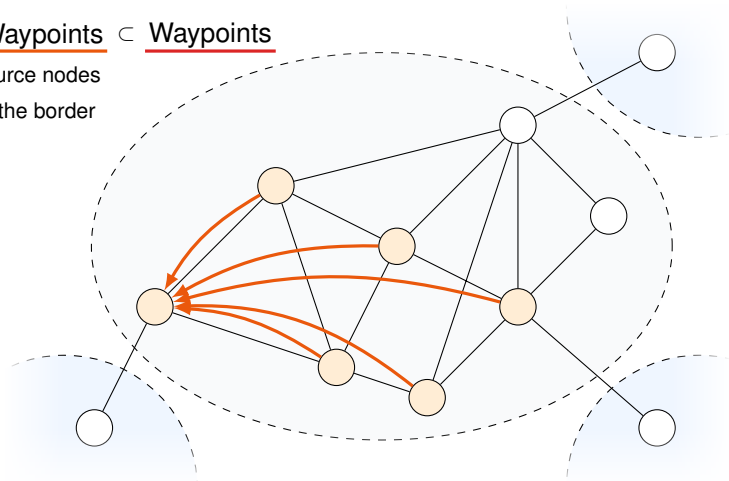
- Region of source nodes
- Waypoint on the border

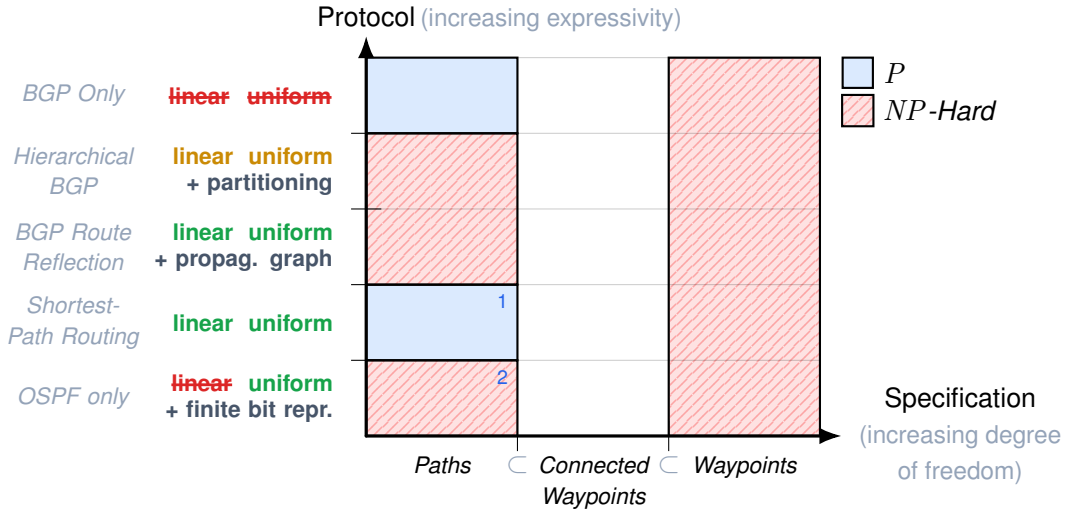


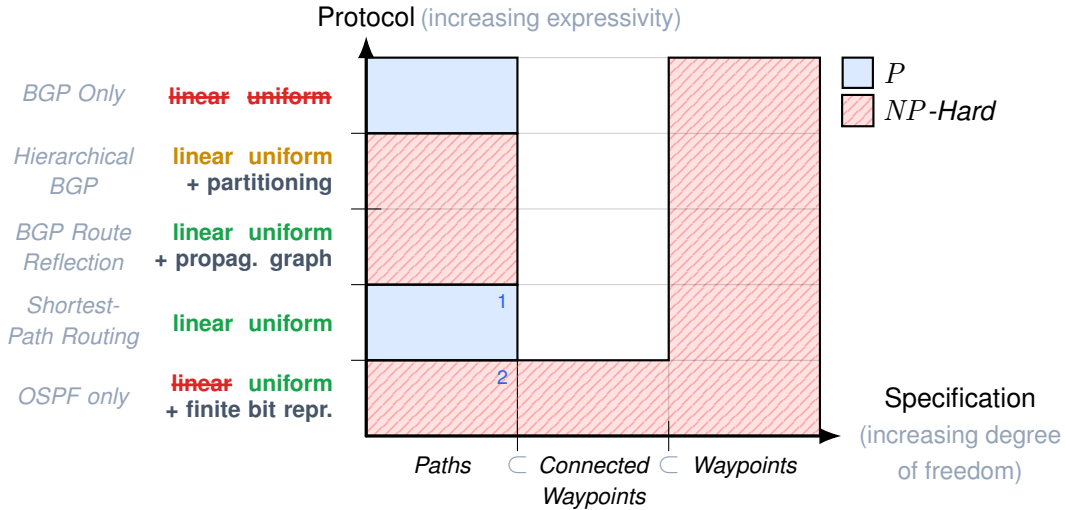
Waypoint properties are usually structured.

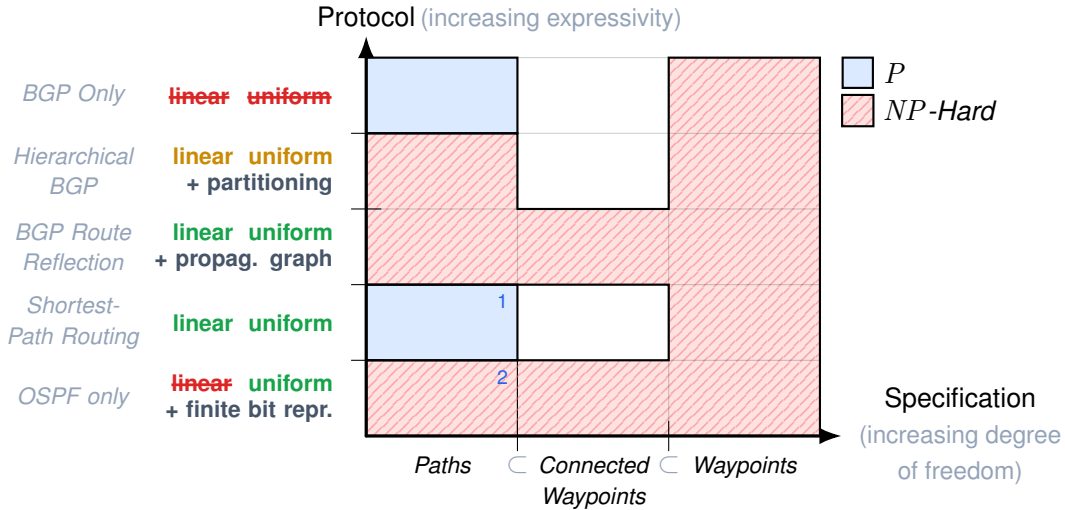
Paths ⊂ Connected Waypoints ⊂ Waypoints

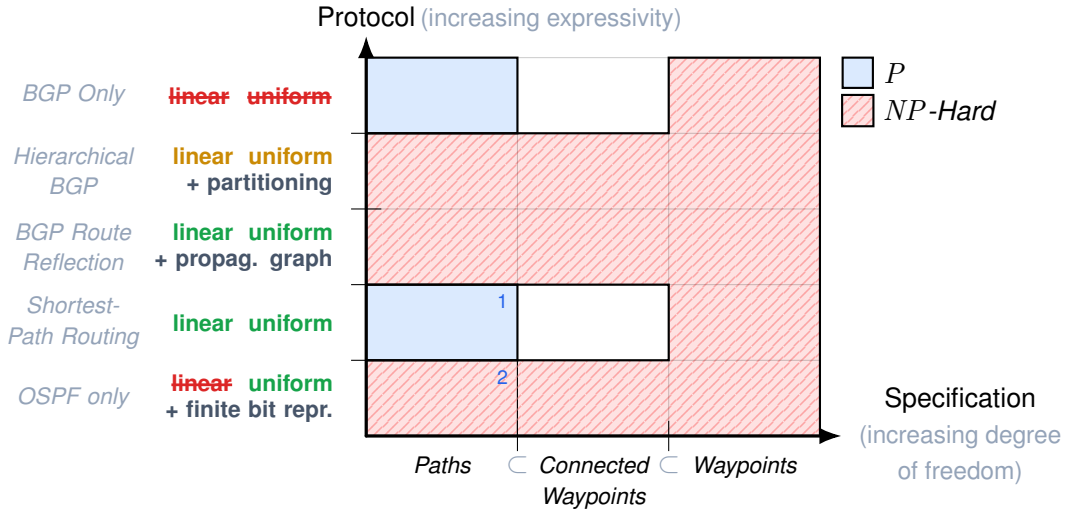
- Region of source nodes
- Waypoint on the border

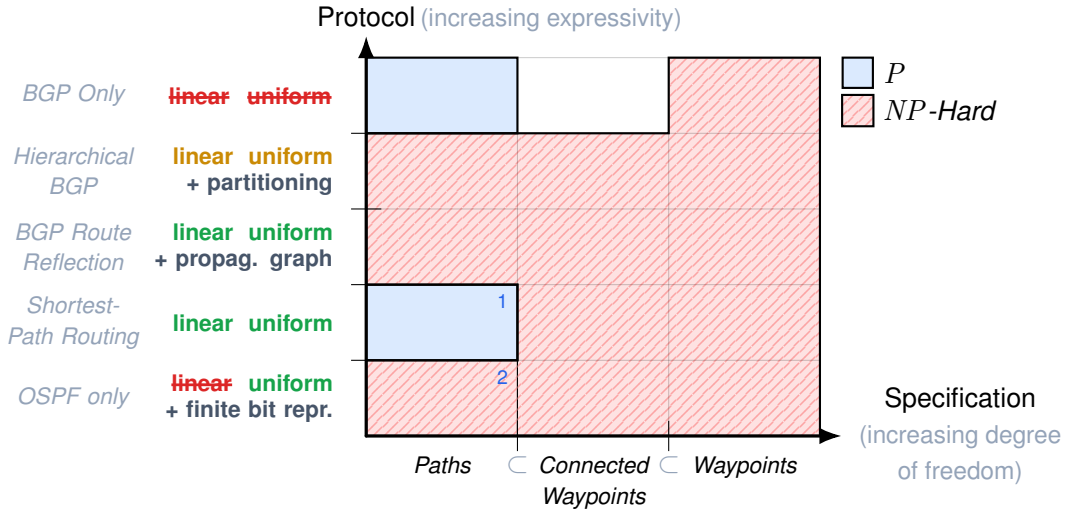


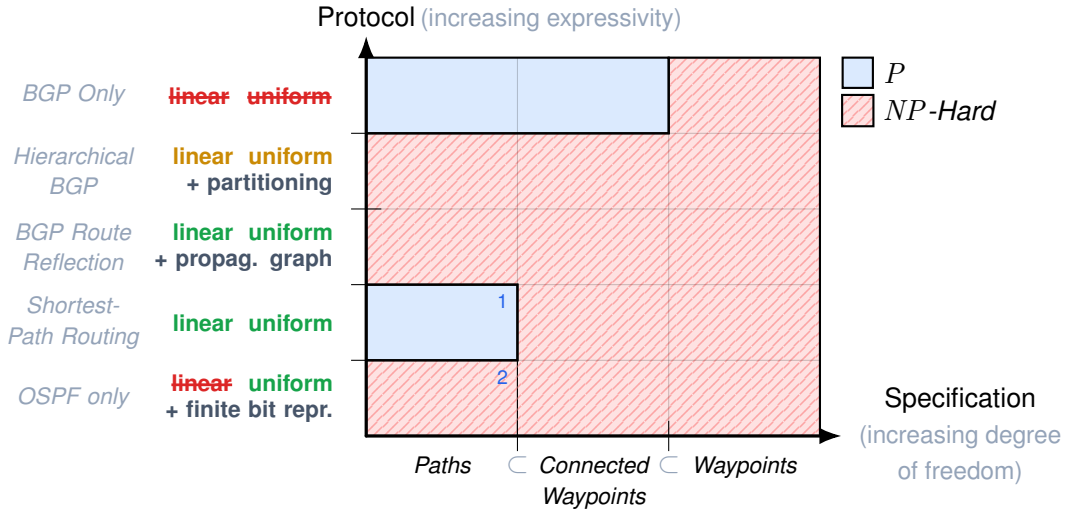

Protocol (increasing expressivity)
BGP Only
linear uniform
Hierarchical BGP
linear uniform
+ partitioning
BGP Route Reflection
linear uniform
+ propag. graph
Shortest-Path Routing
linear uniform
OSPF only
linear uniform
+ finite bit repr.
1
2
P
NP-Hard
Specification
(increasing degree of freedom)
Paths
⊂ Connected Waypoints
⊂ Waypoints


Protocol (increasing expressivity)
BGP Only
linear uniform
Hierarchical BGP
linear uniform + partitioning
BGP Route Reflection
linear uniform + propag. graph
Shortest-Path Routing
linear uniform
OSPF only
linear uniform + finite bit repr.
1
2
P
NP-Hard
Specification (increasing degree of freedom)
Paths
Connected Waypoints
Waypoints

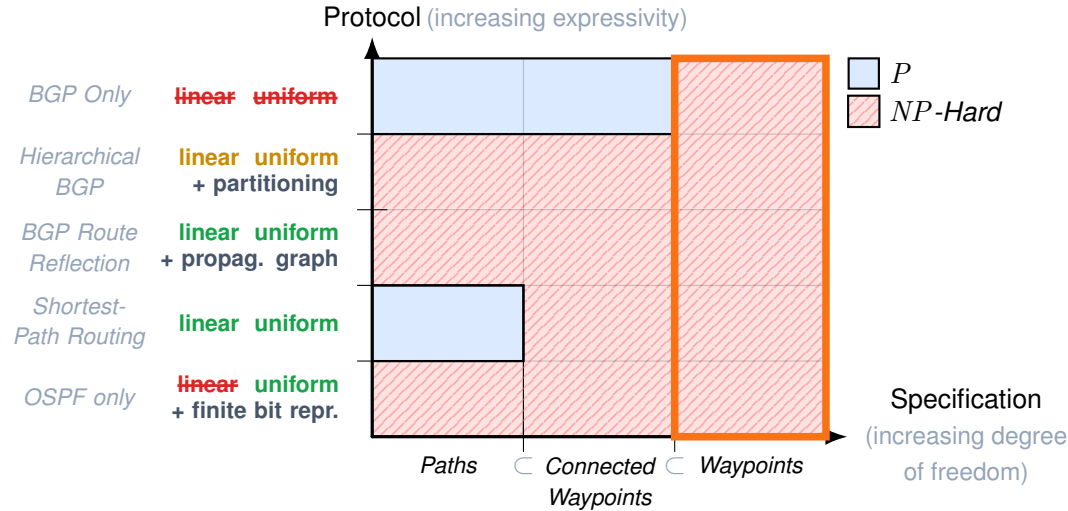

Protocol (increasing expressivity)
BGP Only
linear uniform
Hierarchical BGP
linear uniform
+ partitioning
BGP Route Reflection
linear uniform
+ propag. graph
Shortest-Path Routing
linear uniform
OSPF only
linear uniform
+ finite bit repr.
1
2
P
NP-Hard
Specification
(increasing degree of freedom)
Paths
Connected Waypoints
Waypoints


Protocol (increasing expressivity)
BGP Only
linear uniform
Hierarchical BGP
linear uniform
+ partitioning
BGP Route Reflection
linear uniform
+ propag. graph
Shortest-Path Routing
linear uniform
OSPF only
linear uniform
+ finite bit repr.
1
2
P
NP-Hard
Specification
(increasing degree of freedom)
Paths
Connected Waypoints
Waypoints

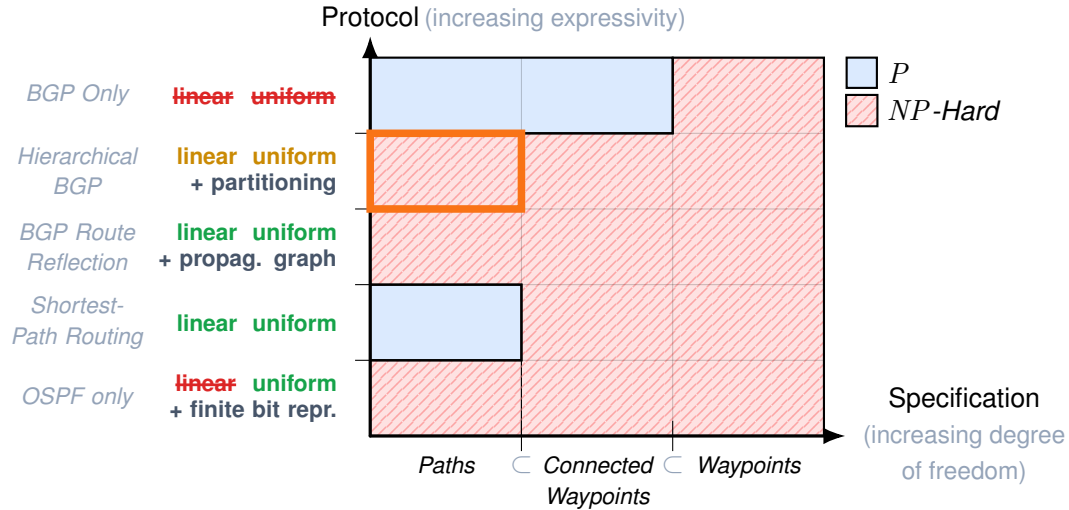

Protocol (increasing expressivity)
BGP Only
linear uniform
Hierarchical BGP
linear uniform
+ partitioning
BGP Route Reflection
linear uniform
+ propag. graph
Shortest-Path Routing
linear uniform
OSPF only
linear uniform
+ finite bit repr.
1
2
P
NP-Hard
Specification
(increasing degree of freedom)
Paths
Connected Waypoints
Waypoints


Protocol (increasing expressivity)
BGP Only
linear uniform
Hierarchical BGP
linear uniform
+ partitioning
BGP Route Reflection
linear uniform
+ propag. graph
Shortest-Path Routing
linear uniform
OSPF only
linear uniform
+ finite bit repr.
1
2
P
NP-Hard
Specification
(increasing degree of freedom)
Paths
Connected Waypoints
Waypoints

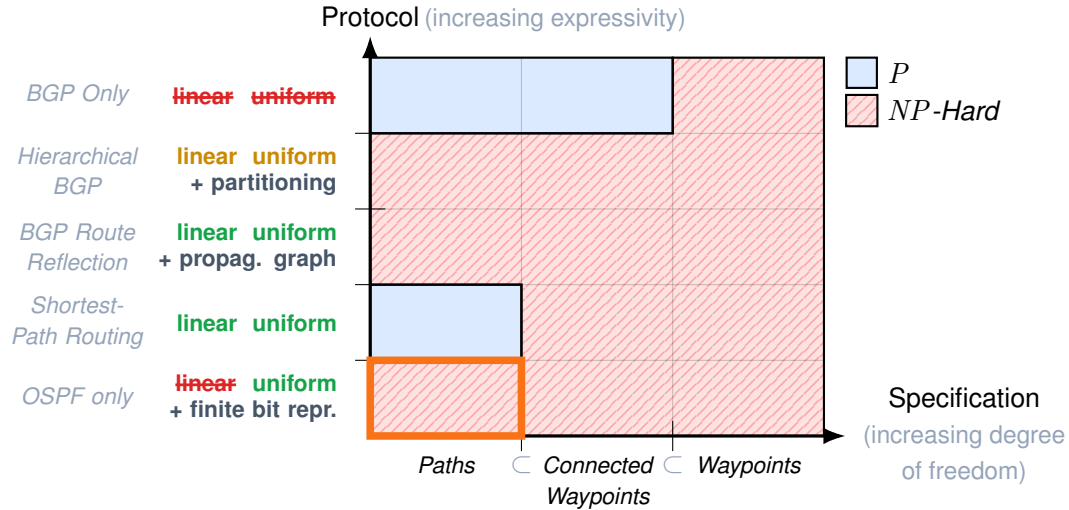
The synthesis problem under arbitrary waypoints is hard.



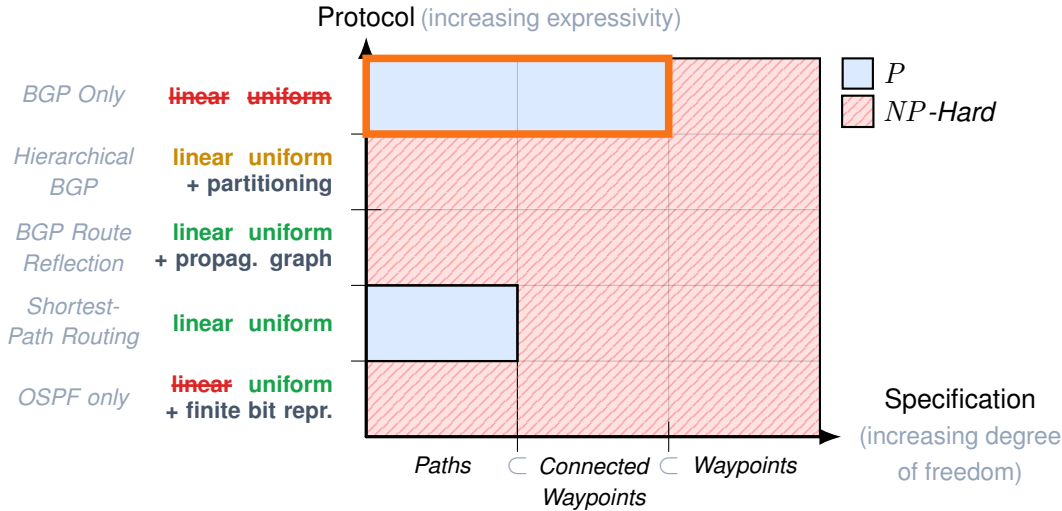
The problem in an NP-Hard region may still be solvable efficiently.



The problem in an NP-Hard region may still be solvable efficiently.



Consider non-linear and non-uniform protocols.



Conclusion

We analyze the computational complexity for configuration synthesis.

Conclusion

We analyze the computational complexity for configuration synthesis.

Our complexity results can **generalize** to other protocols and specification.

Conclusion

We analyze the computational complexity for configuration synthesis.

Our complexity results can **generalize** to other protocols and specification.

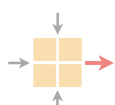
Our results can guide future synthesis systems to achieve better scalability.

Conclusion

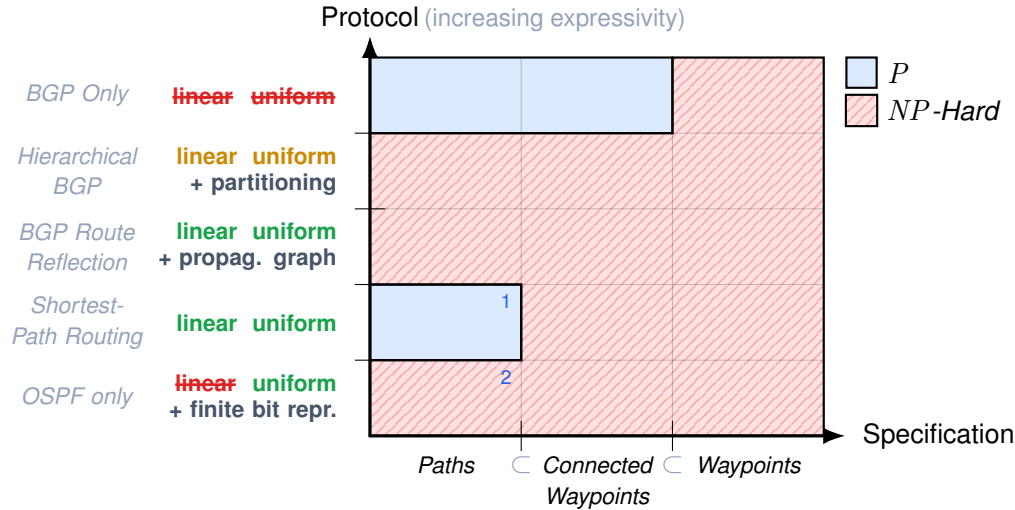
We analyze the computational complexity for configuration synthesis.

Our complexity results can **generalize** to other protocols and specification.

Our results can guide future synthesis systems to achieve better scalability.



nsg.ee.ethz.ch
sctibor@ethz.ch



[1] W. Ben-Ameur et al. "Internet routing and related topology issues". *SIDMA* (2003)

[2] A. Bley. "Inapproximability results for the inverse shortest paths problem with integer lengths and unique shortest paths". *Networks* (2007)