Causal analysis of network logs with layered protocols and topology knowledge

Satoru Kobayashi, Kazuki Otomo, Kensuke Fukuda

CNSM 2019, Halifax

Oct 23, 2019

Outline

- Background and research goal
- Approach
  - Introduction to causal analysis of network logs
  - Proposed method for using domain knowledge in causal analysis
- Evaluation
- Conclusion

Difficulty of leveraging system log in network management

- Huge dataset
  - Large scale and complicated systems
  - 150,000 lines / day in SINET 5
  - Automated analysis required
- Difficulty in automated analysis
  - Free-format and sparse data
  - Contextual information required for troubleshooting

Causal analysis in operational data

- Causal analysis: A popular approach for extracting contextual information
  - More reliable than correlation-based approach
- Problem:
  - Efficiency (large processing time)
  - No consideration of network knowledge

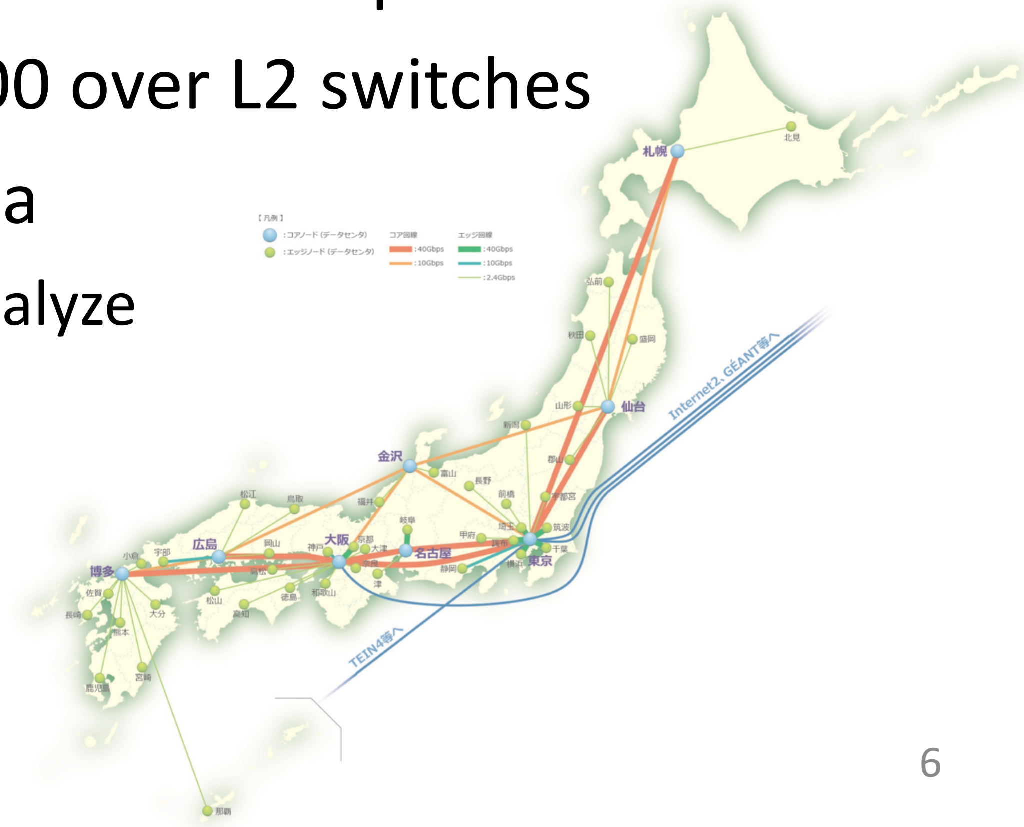
⇒ Causal analysis with network domain knowledge

Goal

- Provide contextual information for system management and troubleshooting from network system logs
  - Causal analysis + Network domain knowledge
  - Improve efficiency and reliability

Dataset

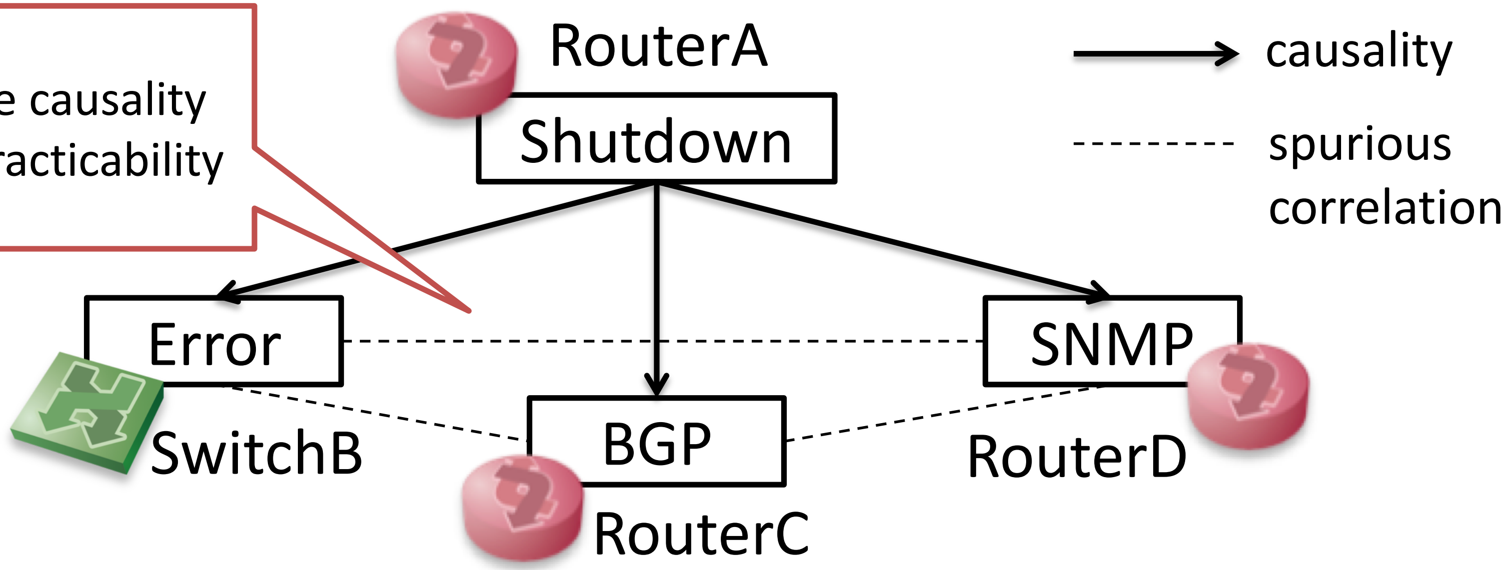
- SINET4
  - https://www.sinet.ad.jp/en/top-en
  - A nation-wide R&E network in Japan
  - 8 core routers and 100 over L2 switches
  - 15 months syslog data
    - 3.5 million lines to analyze



Causal analysis of network logs[1]

```
Oct 17 17:00:00 routerA System shutdown by root
Oct 17 17:00:05 switchB Error detected on eth0
Oct 17 17:00:15 routerC BGP state changed from Established to Idle
Oct 17 17:00:15 routerD SNMP trap sent to routerA
……
```

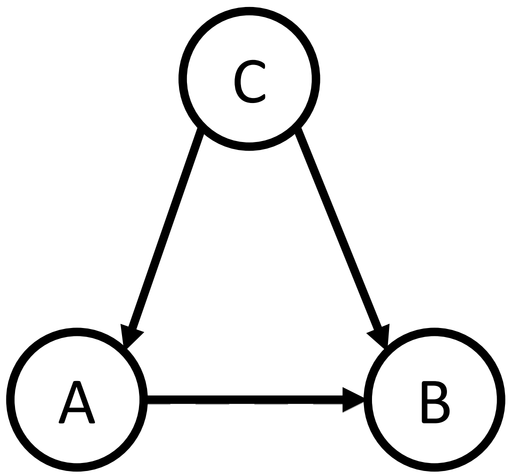
Remove false causality
-> Improve practicability



[1] S. Kobayashi et al. "Mining causality of network events in log data", IEEE TNSM, vol. 15, no.1, pp. 37–67, 2018.

Causal Inference

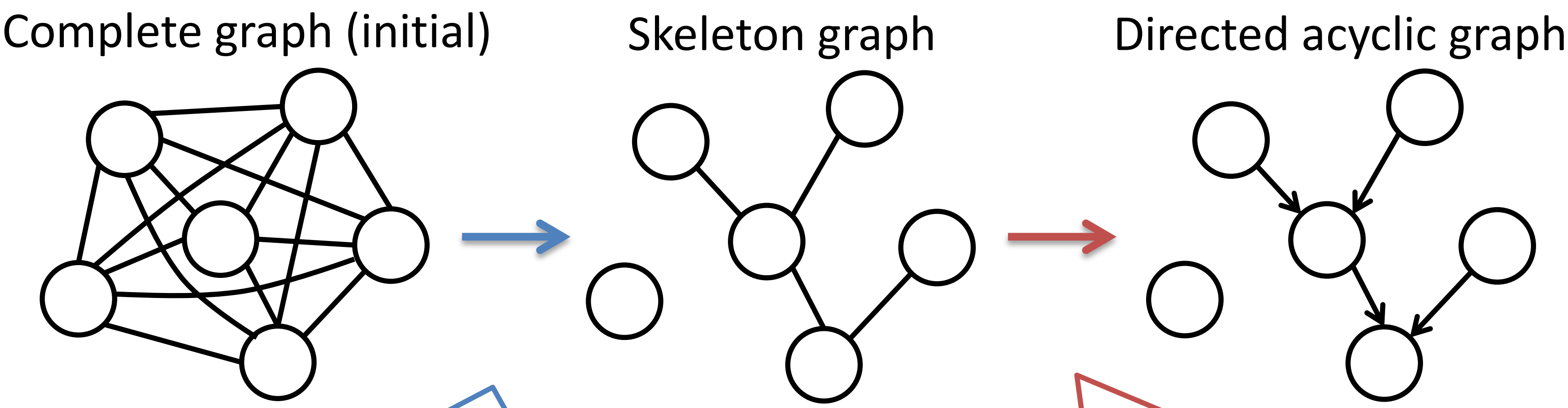
- Conditional Independence
  - A and B are independent if the effect of confounder C is excluded
  - A and B are conditionally independent given C
- PC algorithm [2]
  - Directed acyclic graph (DAG)
  - Explore conditional independence and remove false edges



$$P(A|C)P(B|C) = P(A, B|C)$$

[2] P. Spirtes, et al. “Causation, prediction, and search”. MIT press, 2000.

Flow of PC algorithm

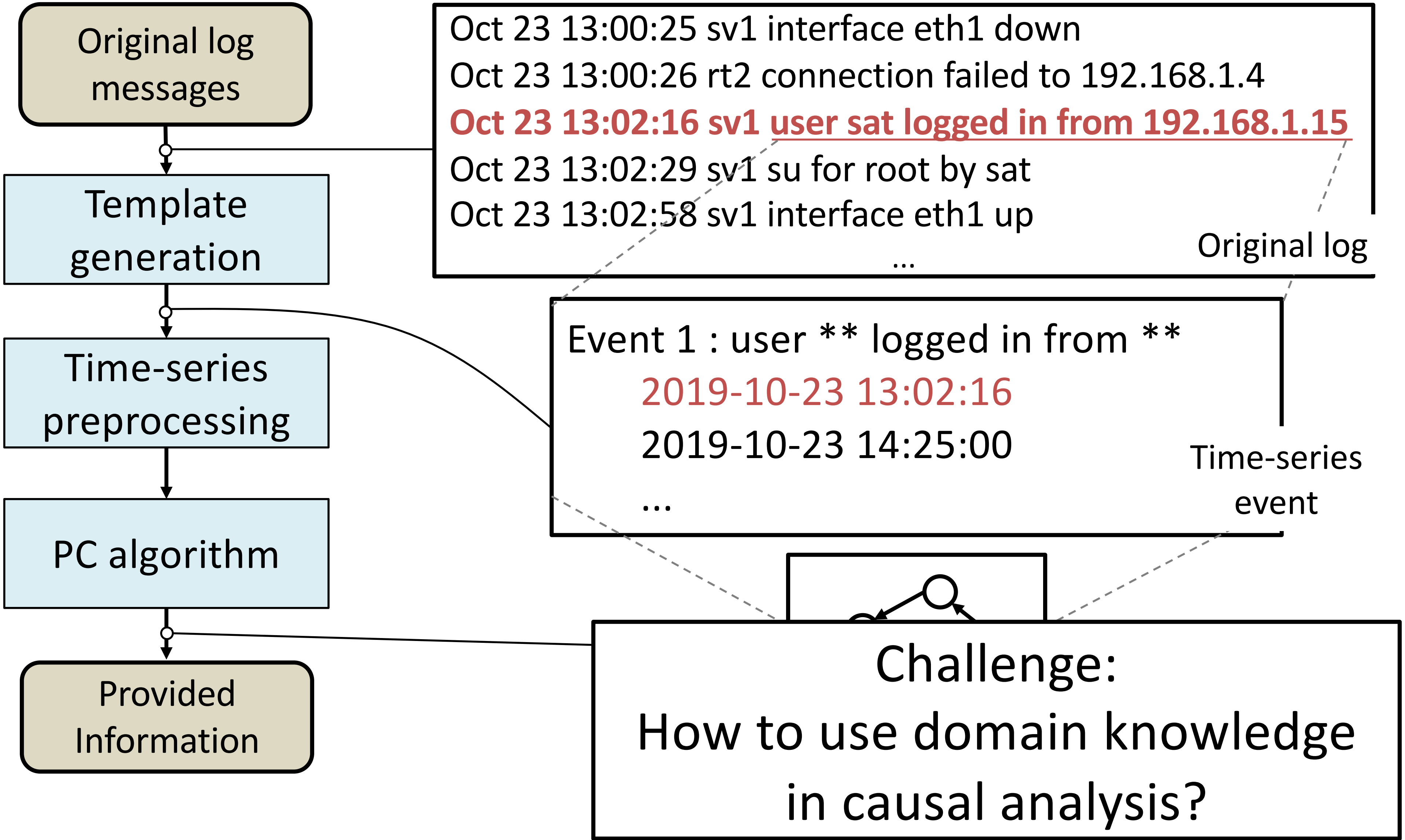


- Remove edges of conditional independence
- Statistical test for conditional independence
  - G2 test (for binary or multi-level data) [3]
  - Fisher-Z test (for continuous data) [3]

ps)

[3] R. E. Neapolitan. "Learning Bayesian Networks." Prentice Hall Upper Saddle River, 2004.
[4] T. Verma, et al. "An algorithm for deciding if a set of observed independencies has a causal explanation". In Proceedings of UAI'92, pp. 323–330, 1992.

Causal analysis with network logs [1]

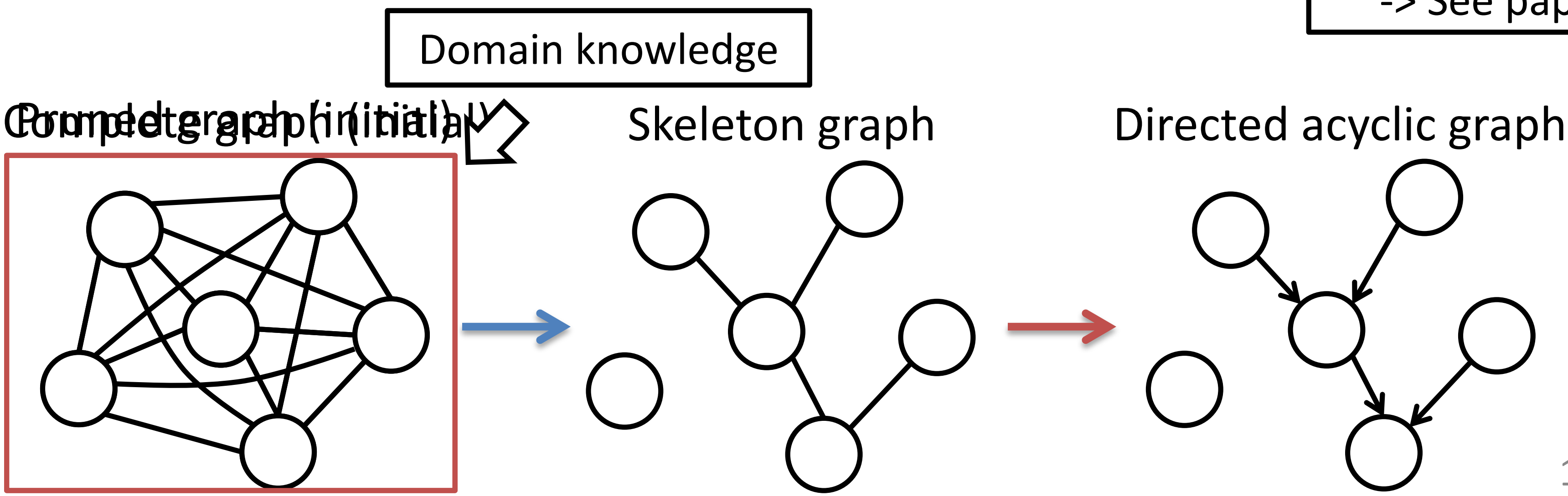


[1] S. Kobayashi et al. "Mining causality of network events in log data", IEEE TNSM, vol. 15, no.1, pp. 37–67, 2018.

Approach: Pruning initial graph

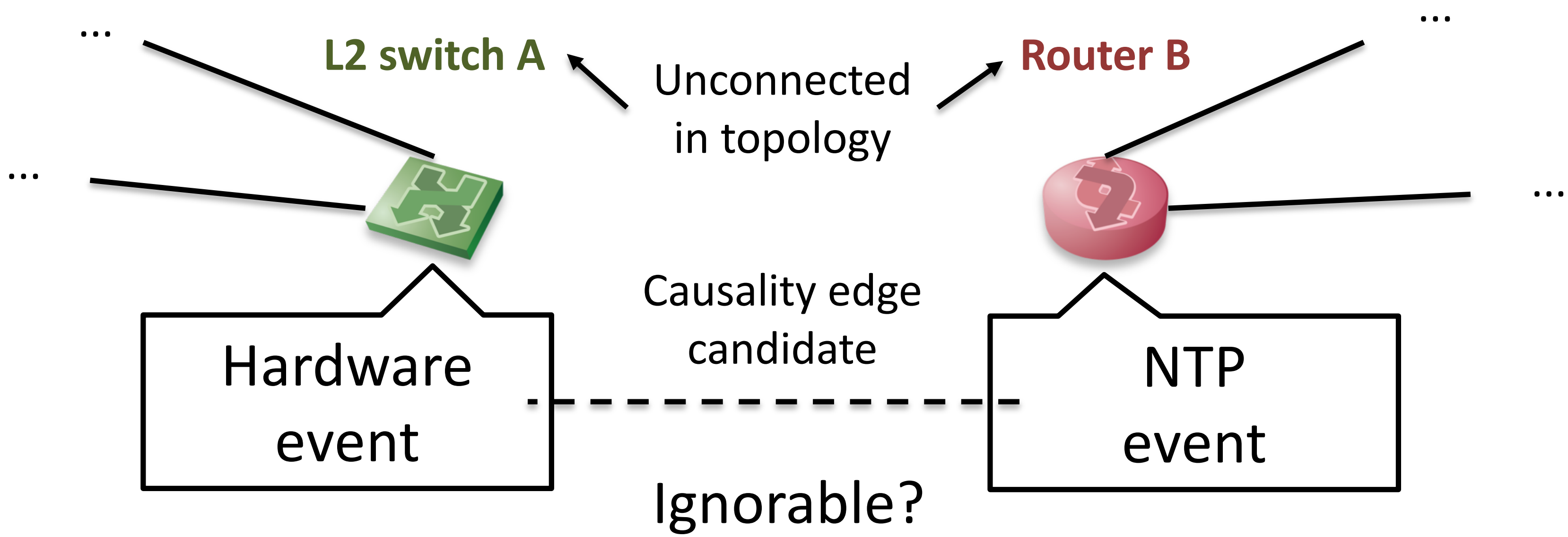
- PC algorithm usually starts with complete graph
  - Takes large processing time if network structure is large and complex
- Prune edges in initial graph of PC algorithm
  - Complete graph -> Pruned graph

Is it OK in theory?
-> See paper

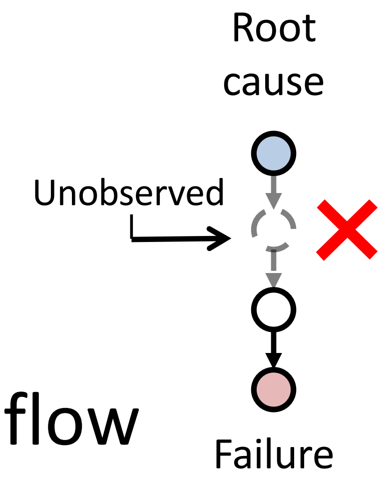


Pruning edges with domain knowledge

- Basic idea
  - Some edge candidates are clearly not causality
    - Compared with domain knowledge of operators
  - Ignore in calculating causality

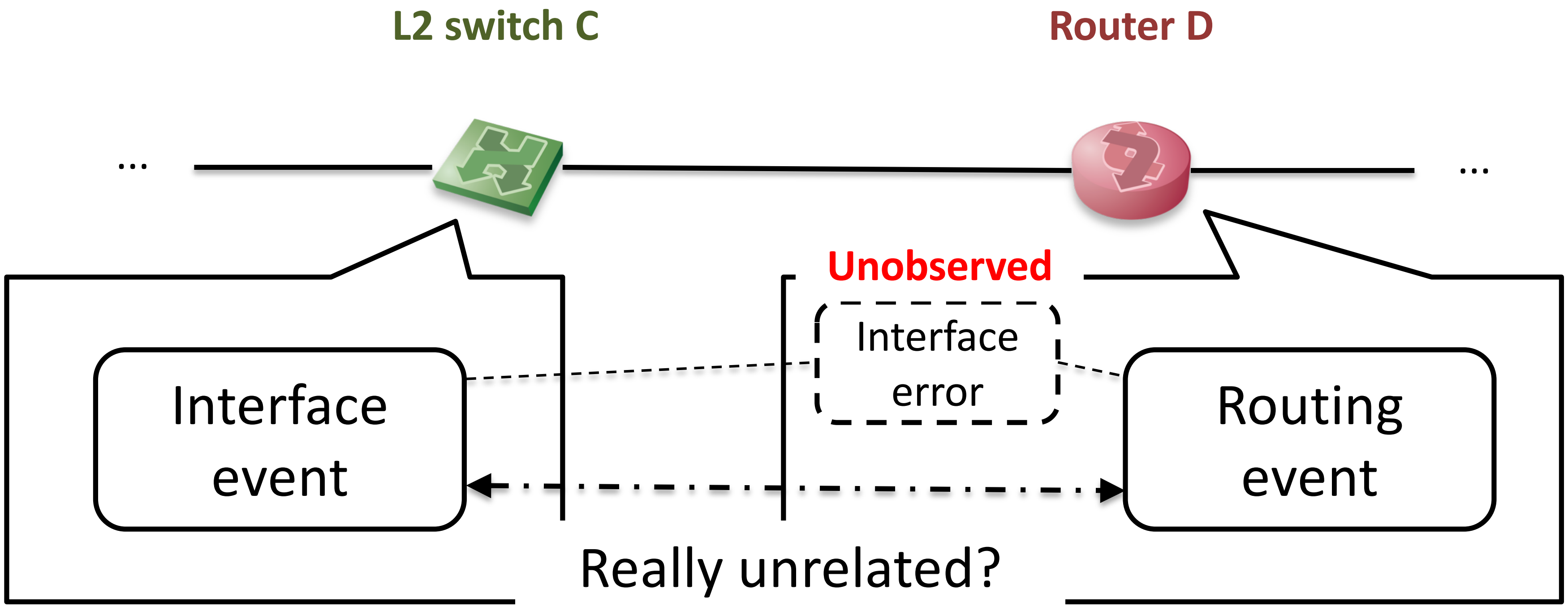


Difficulty in pruning



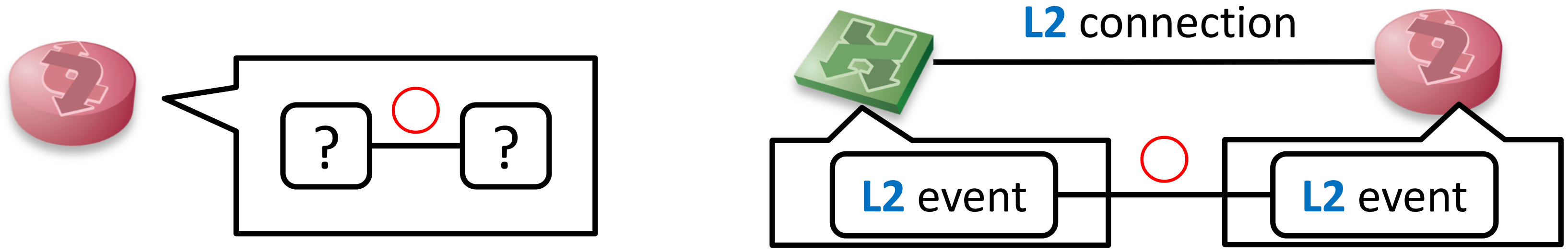
- **Unobserved events** mediate causality
  - Pruning mediated causality breaks causal flow

-> How to determine the criteria?

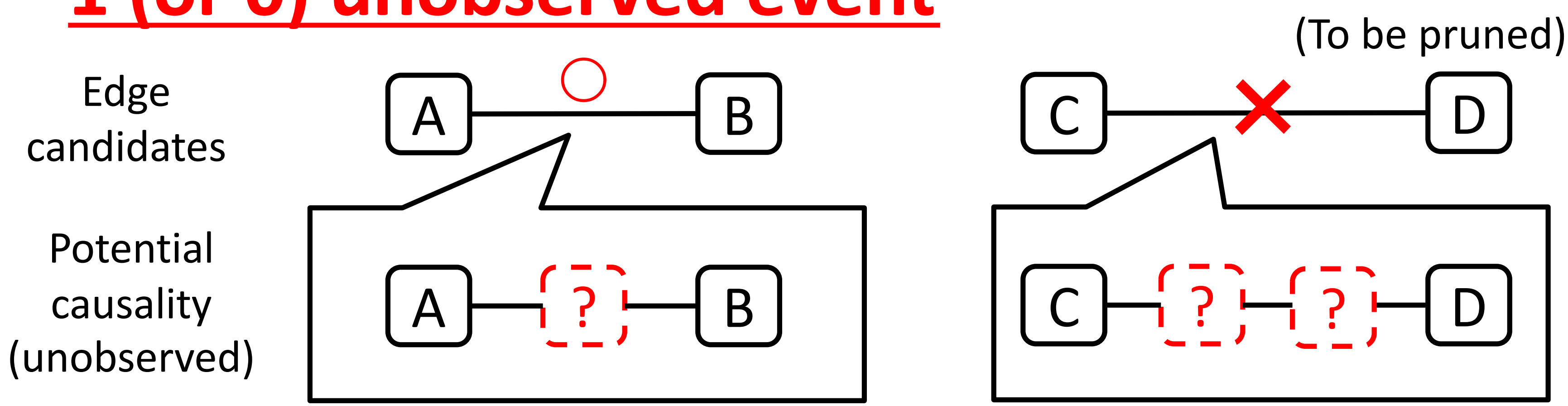


Proposed method: 2 criteria

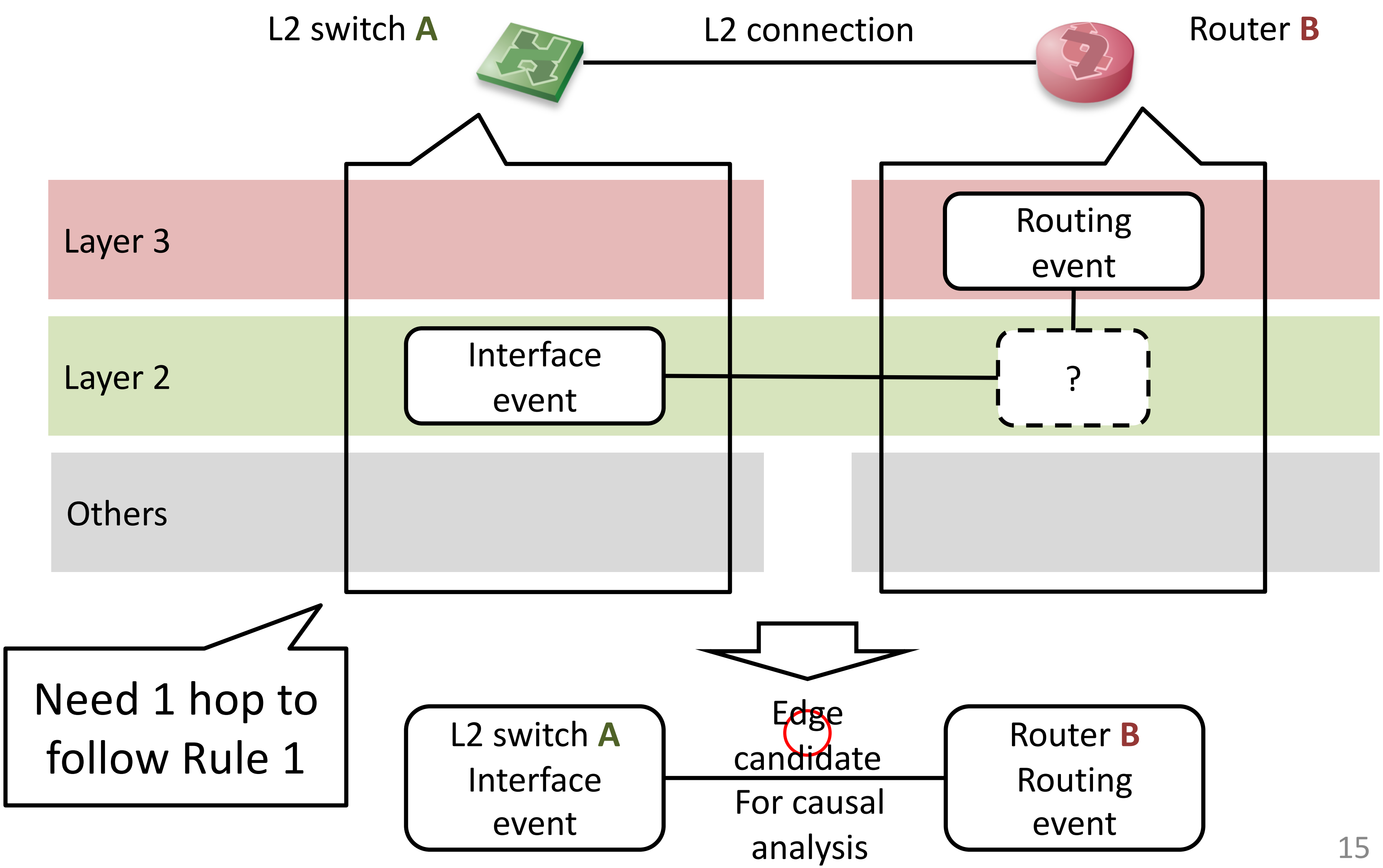
Rule 1. Events **in same device**, or **in same functional layer and in connected devices**



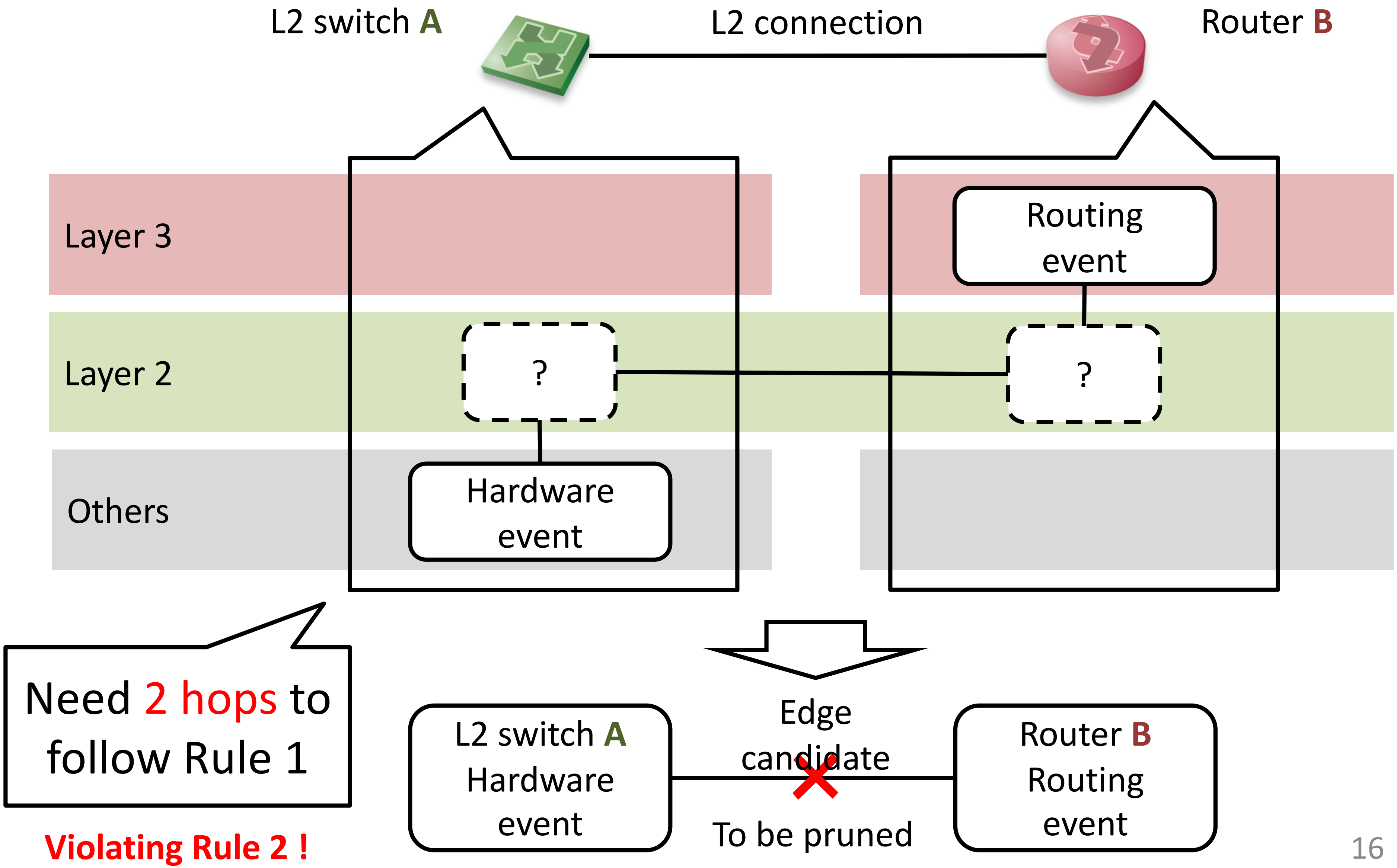
Rule 2. A causal edge can be mediated with **1 (or 0) unobserved event**



Example: Good causality candidate

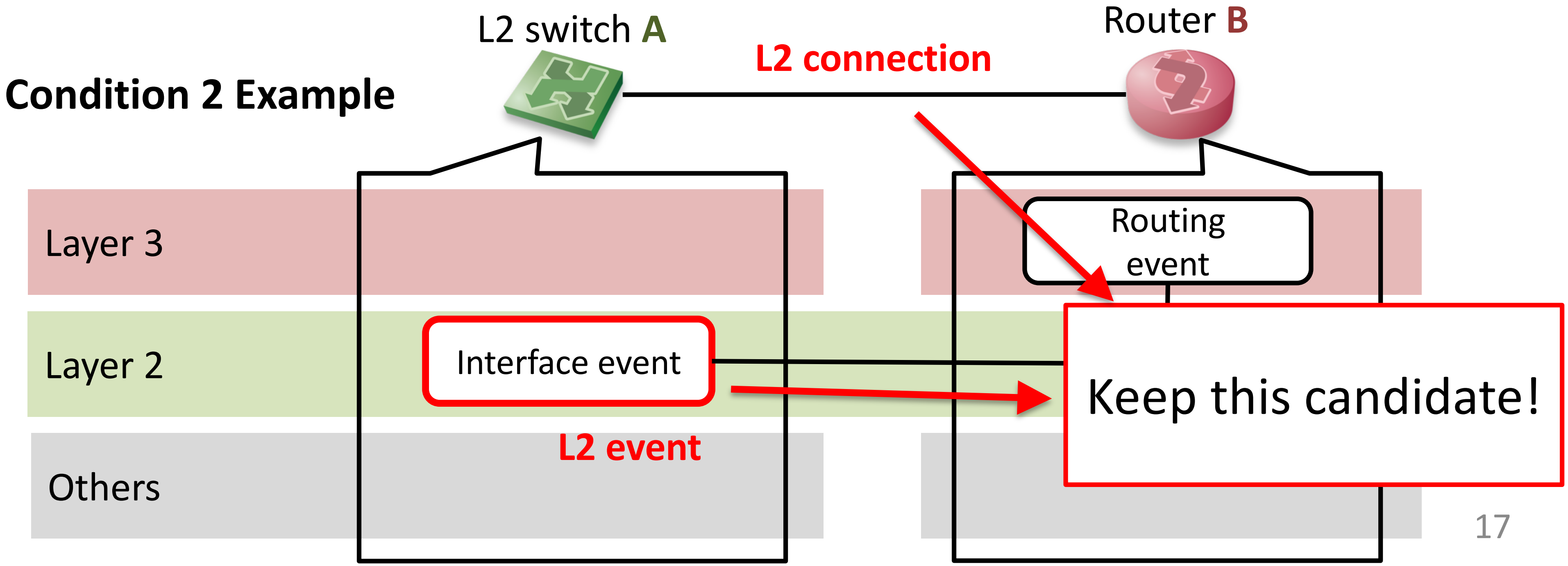


Example: Bad causality candidate

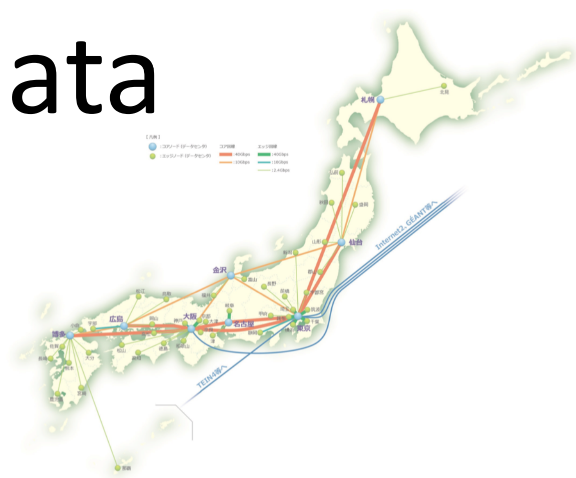


Algorithm to classify causality candidate

- Keep a causal edge if satisfying 1 or 2
  1. 2 events appear in same device
  2. At least 1 end node (event) is on a functional layer that connects the devices



Analysis in SINET4 data

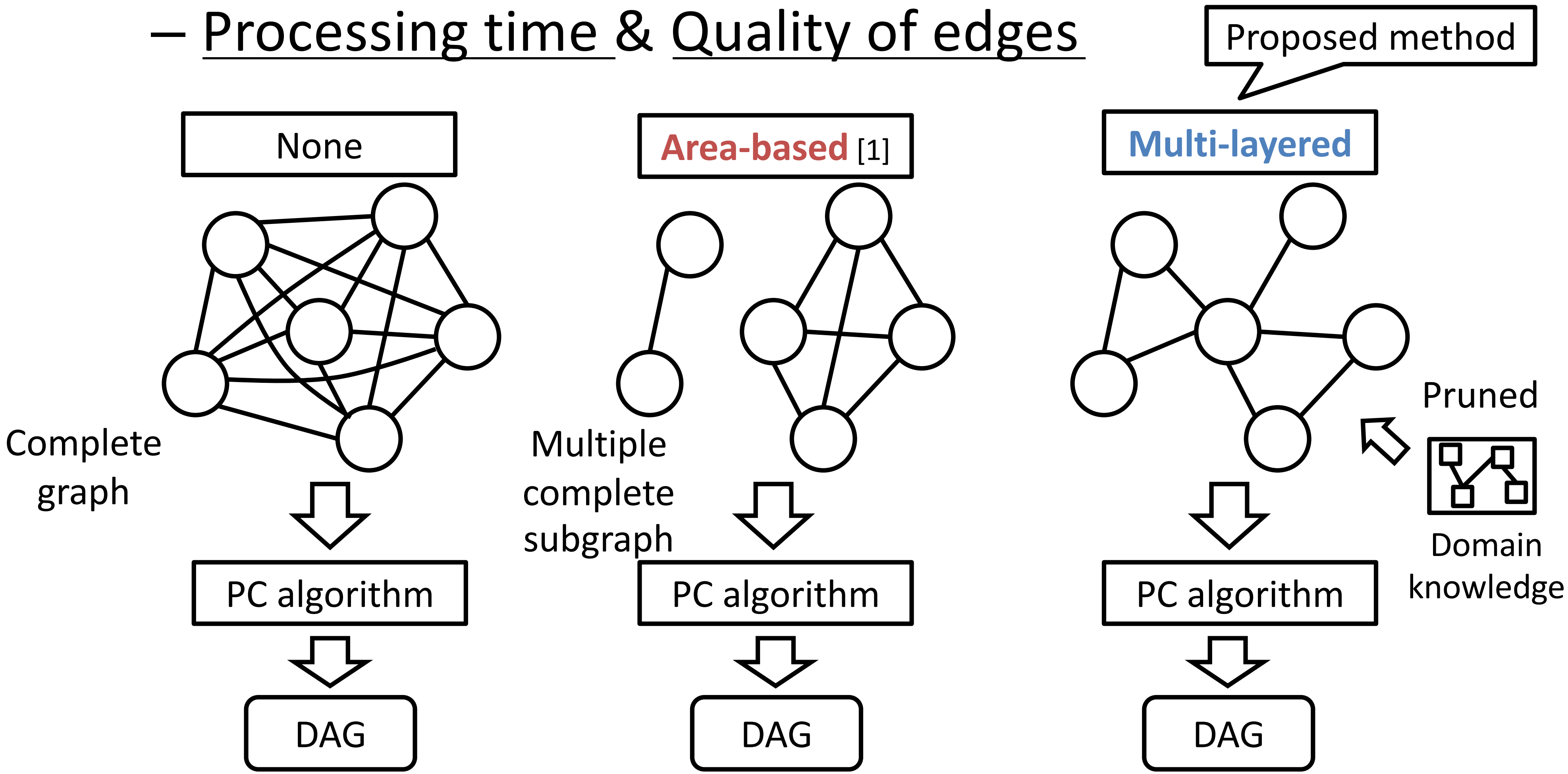


- Domain knowledge for pruning
  - Network topology (L2, L3)
  - Functional layer definition of events (L2, L3, others)
    - Manually labeled 9 classes for log templates
    - Layer definition for the classes ↓

Layer definition	Event group (label)
L3	Routing-EGP, Routing-IGP, VPN
L2	Interface, Network
Others	System, Service, Management, Monitor

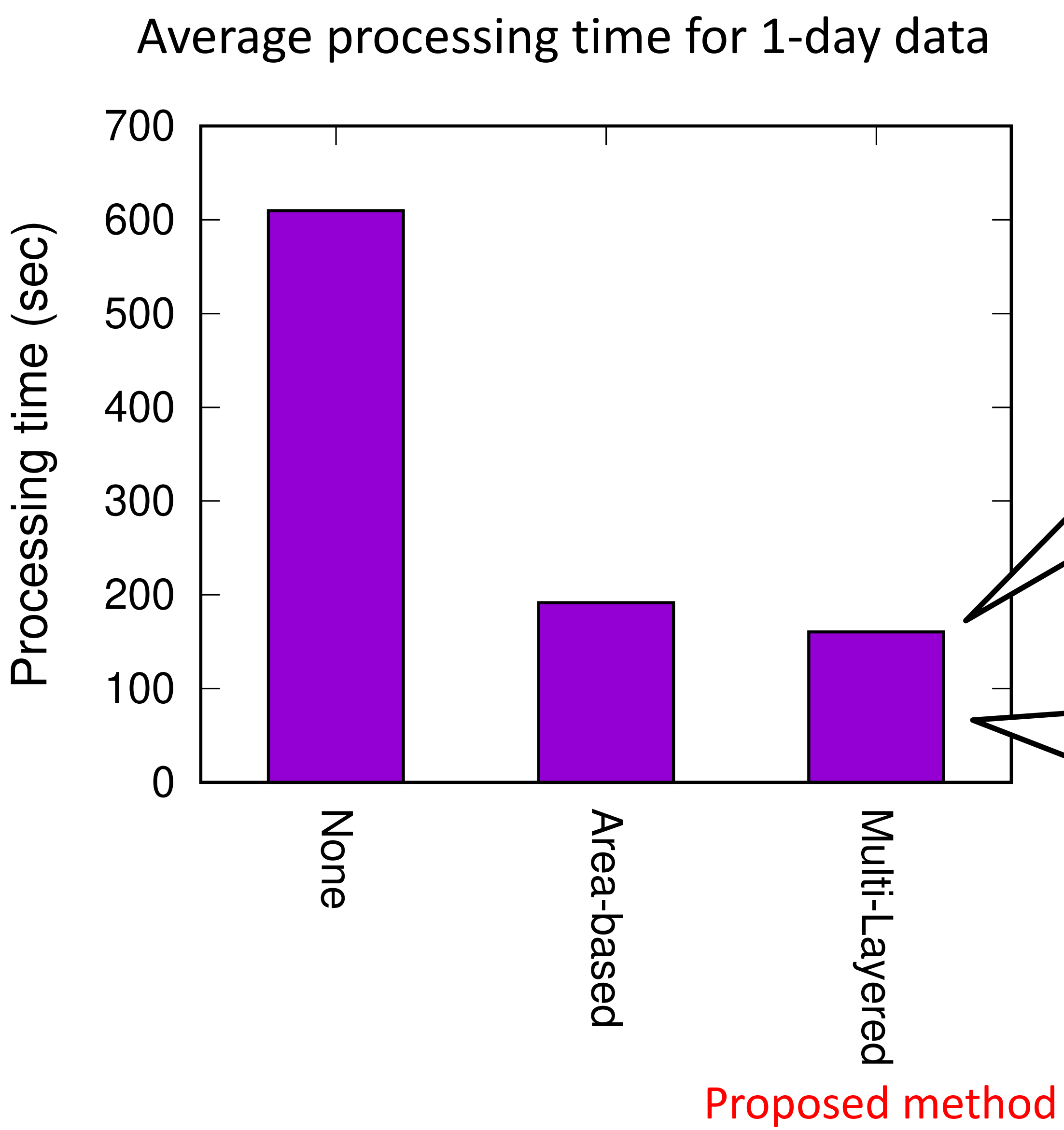
Evaluation

- Compare 3 methods (different initial graph)
  - Processing time & Quality of edges



[1] S. Kobayashi et al. "Mining causality of network events in log data", IEEE TNSM, vol. 15, no.1, pp. 37–67, 2018.

Processing time of PC algorithm



Decreased 74% compared with None

16% faster than existing method

Quality of causal edges

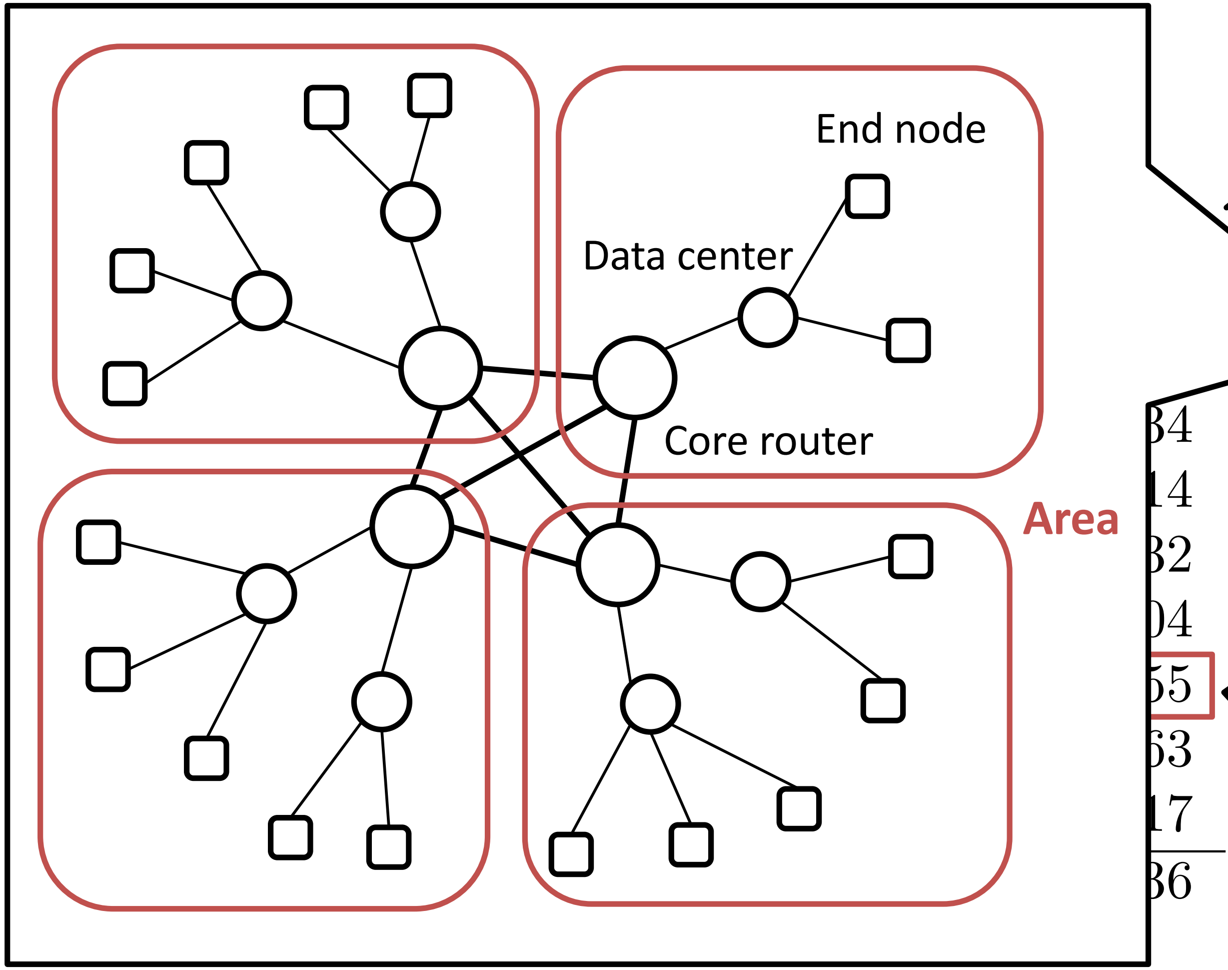
- Event classes of end nodes of detected edges

Type	#Nodes	#Ends of edges		
		None	Area	ML
System	49,005	24,577	23,033	22,662
Network	10,585	1,402	1,391	1,355
Interface	13,562	1,943	2,062	2,134
Service	7,697	742	435	314
Mgmt	81,628	29,379	27,911	26,332
Monitor	2,467	267	305	304
VPN	4,538	97	1,171	155
Rt-EGP	4,738	1,923	2,063	2,063
Rt-IGP	870	18	19	17
Total	175,090	60,348	58,390	55,336

Multi-Layered method:
Same distribution with None

Quality of causal edges

- Event classes of end nodes of detected edges



Multi-Layered method:
Same distribution with None

Area-based method
has a problem:
Relations among multiple areas are missed

⇩

Fails to detect
conditional independence

⇩

False Positive edges

Summary of evaluation

Pruning methods	Processing time	Quality of edges
None	× Takes 10 minutes / day	○ (Shown in previous paper [1])
Area-based method	○ Decrease 69%	× No consideration of area gaps
Multi-Layered method (proposed method)	◎ Decrease 74%	○ Similar distribution to None

[1] S. Kobayashi et al. "Mining causality of network events in log data", IEEE TNSM, vol. 15, no.1, pp. 37–67, 2018.

Discussion

- Parallel processing?
  – Available in PC algorithm [5]
- Available in other causal algorithms?
  – Depends on algorithms
  – Easily available in regression-based methods or constraint-based causal methods
- Available in any network?
  – Effective even in full-mesh-topology network

[5] T. Le, et al. “A fast PC algorithm for high dimensional causal discovery with multi-core PCs,” *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 13, no. 9, pp. 1–13, 2014.

Conclusion

- Causal inference approach with network domain knowledge for helping troubleshooting
- Pruning initial graph of PC algorithm
  – Considering unobserved events
- Improvement in terms of processing time and quality of edges
  – Decrease 74%, 16% faster than Area-based method
  – Solve area-gap problem in Area-based method
- https://github.com/cpflat/logdag