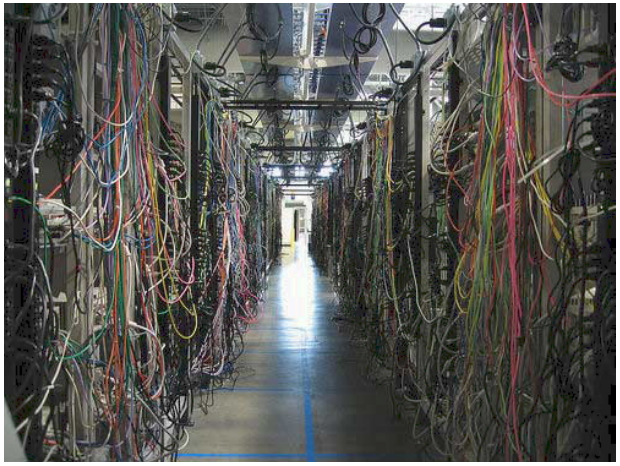
Theophilus Benson (tbenson@cs.wisc.edu)
Aditya Akella (akella@cs.wisc.edu)
David A Maltz (dmaltz@microsoft.com)

Unraveling the Complexity of Network Management

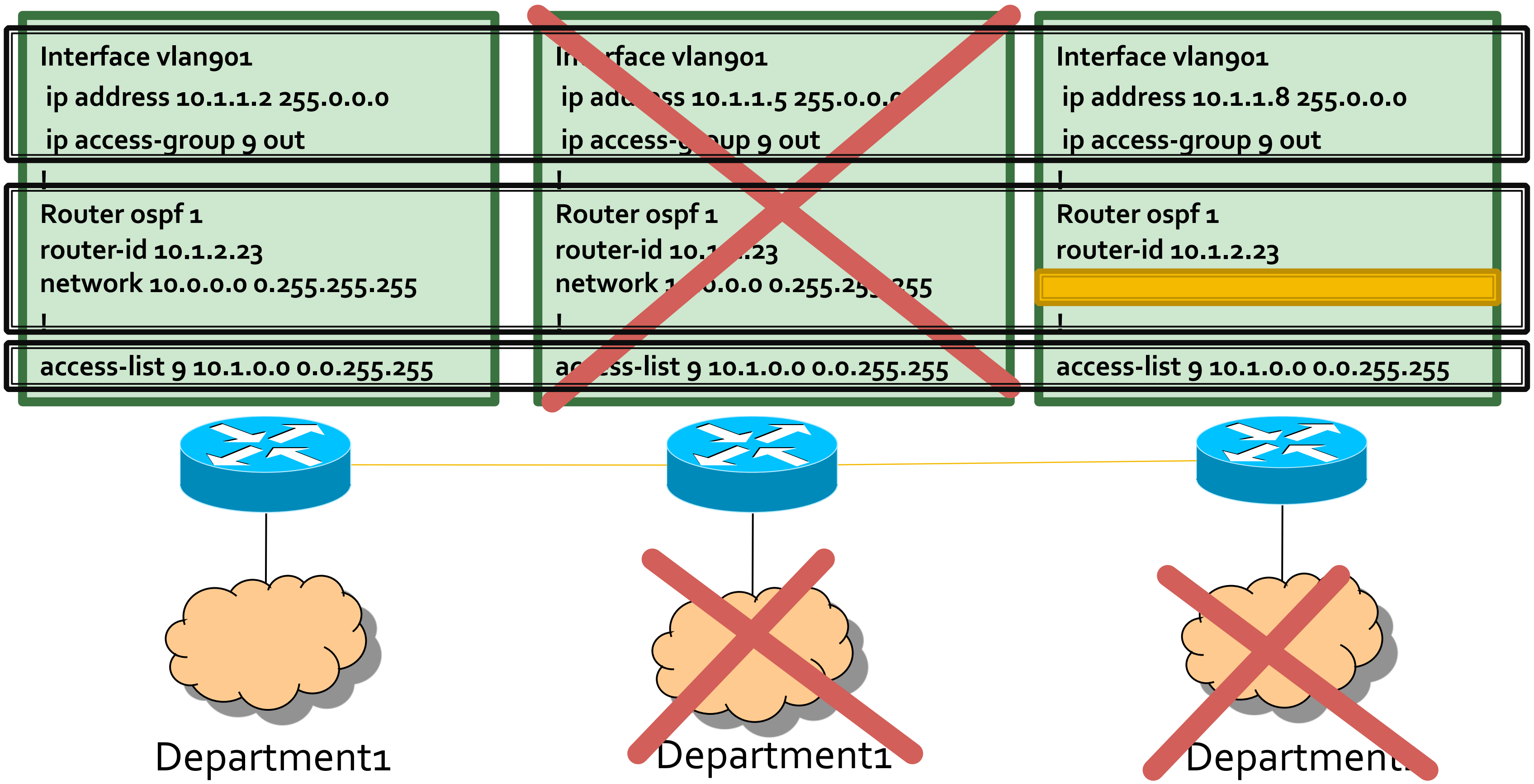
Enterprise Networks

- Intricate logical and physical topologies
- Diverse network devices
  - Operating on different layers
  - Requiring different command sets
- Operators constantly tweak network configurations
  - Implementation of new admin policies
  - Quick-fixes in response to crises
- Diverse goals
  - E.g. QOS, security, routing
- Complex configuration



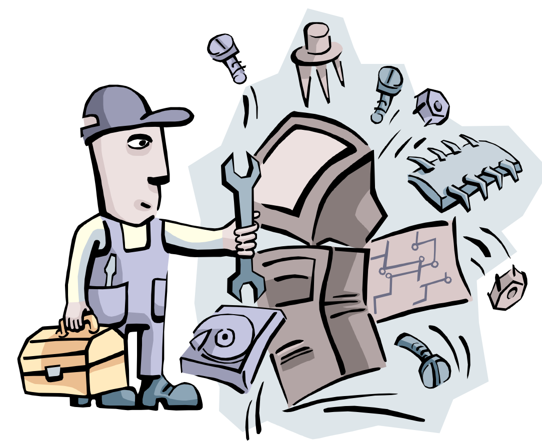
Example of a Configuration Change

- Adding a new department with hosts spread across 3 buildings



Complexity of Network Design

- Complexity leads to misconfiguration
- Can't measure complexity of network design
  - Other communities have benchmarks for complexity
- No complexity metric → can't understand difficulty of future changes
  - Quick fix now may **complicate** future changes
  - Hard to select from alternate configs
- Ability to predict difficulty of future changes is essential
  - Reduce management cost, operator error



Capturing Network Complexity

- Our metrics:
  - Succinctly describe design complexity
  - Can be automatically calculated from config files
  - Align with operator's mental models
    - **Predict** difficulty of **future changes**
- Empirical study of complexity of 7 networks
- Validated metrics through operator interviews
  - Questionnaire: tasks to quantify complexity
    - Network specific
    - Common to all operators
- Focus on layer 3

Networks Studied

- Complexity is unrelated to size or line count
  - Complex
  - Simple

Networks	Mean file size	Number of routers
Univ-1	2535	12
Univ-2	560	19
Univ-3	3060	24
Univ-4	1526	24
Enet-1	278	10
Enet-2	200	83
Enet-3	600	19

Two Types of Design Complexity

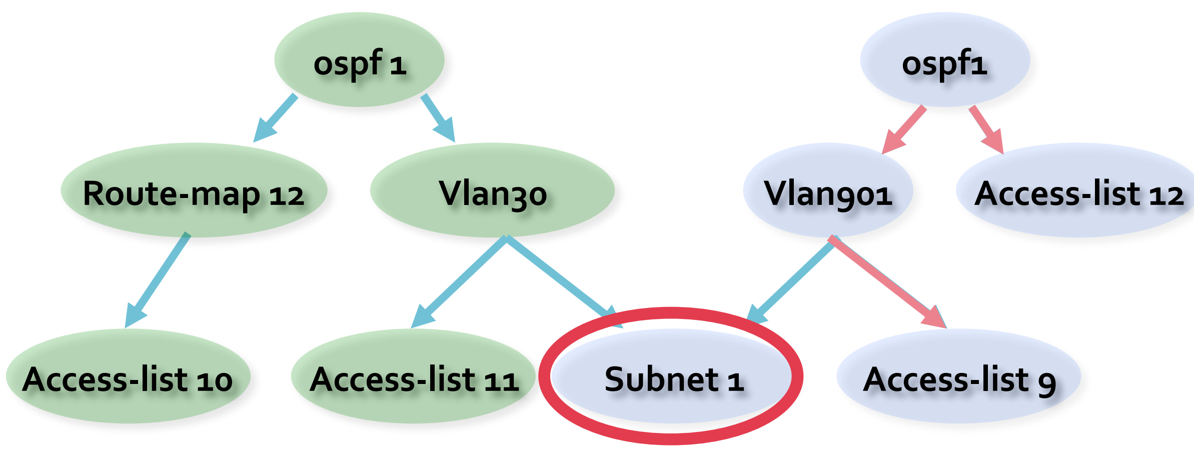
- Implementation complexity: difficulty of implementing policies
  - Referential dependence: the complexity behind configuring routers correctly
  - Roles: the complexity behind identifying roles for routers in implementing a network's policy (See paper for details)
- Inherent complexity: complexity of the policies themselves
  - Uniformity: complexity due to special cases in policies
  - Lower-bounds implementation complexity

Outline

- Referential complexity
- Inherent complexity
- Insights into complexity
- Related work and conclusion

Referential Dependency Metric: Example

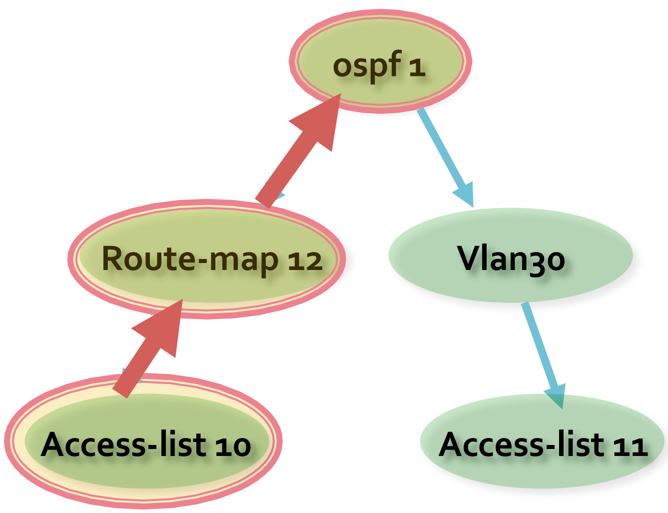
- Referential graph for shown config
  - Intra-file links, e.g., passive-interfaces, and access-group.
- Inter-file links
  - Global network symbols, e.g., subnet, and VLANs.




```
Interface Vlan901
  ip 128.2.1.23  255.255.255.252
  ip access-group 9 in
!
Router ospf 1
  router-id 128.1.2.133
  passive-interface default
  no passive-interface Vlan901
  no passive-interface Vlan900
  network 128.2.0.0 0.0.255.255
  distribute-list in 12
  redistribute connected subnets
!
access-list 9 permit  128.2.1.23  0.0.0.3 any
access-list 9 deny any
access-list 12 permit 128.2.0.0 0.0.255.255
```

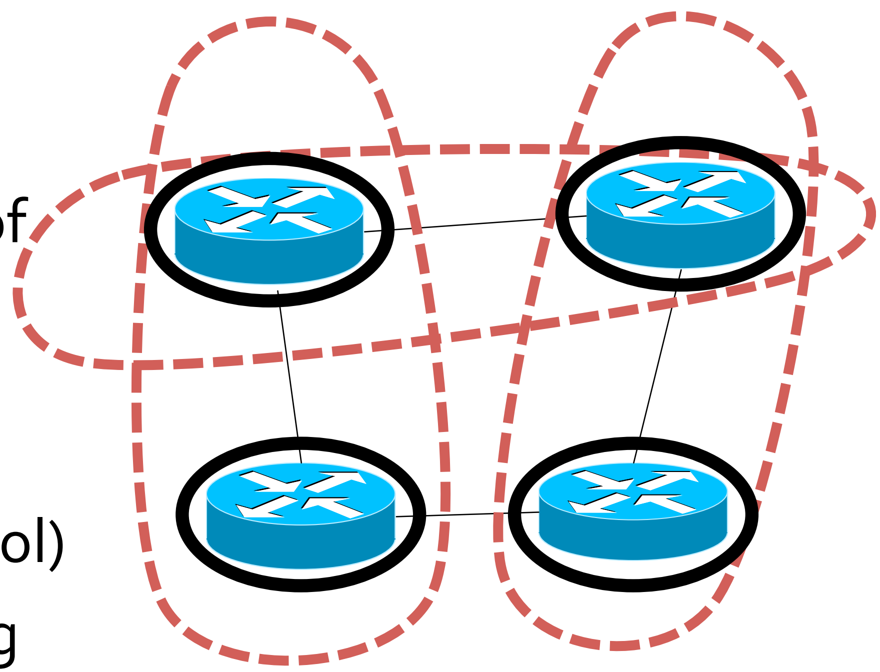
Referential Dependence Metric

- Operator's objective: short dependency chains in configuration
  - Few moving parts (few dependencies)
- Referential metric should capture:
  - Difficulty of setting up layer 3 functionality
  - Extent of dependencies



Referential Dependence Metric

- Metric: # ref links greater # links means higher complexity
  - Normalize by # devices
    - Holistic view of network
- Metric: # routing instances
  - Routing instance = partition of routing protocols into largest atomic domains of control
  - Routing instance = adjacent routing process (same protocol)
  - Difficulty of setting up routing



Empirical Study

- Complexity unrelated to network size
  - Complexity based on implementation details
  - Large network could be simple

Network (#routers)	Avg Ref links per router	#Routing instances
Univ-1 (12)	42	14
Univ-2 (19)	8	3
Univ-3 (24)	4	1
Univ-4 (24)	75	2
Enet-1 (10)	2	1
Enet-2 (83)	8	10
Enet-3 (19)	22	8

Metrics   Complexity

Task: Add a new subnet at a randomly chosen router

Network	Avg Ref links per router	#Routing instances
Univ-1 (12)	42	14
Univ-3 (24)	4	1
Enet-1 (10)	2	1

Num steps	#changes to routing
4-5	1-2
4	0
1	0

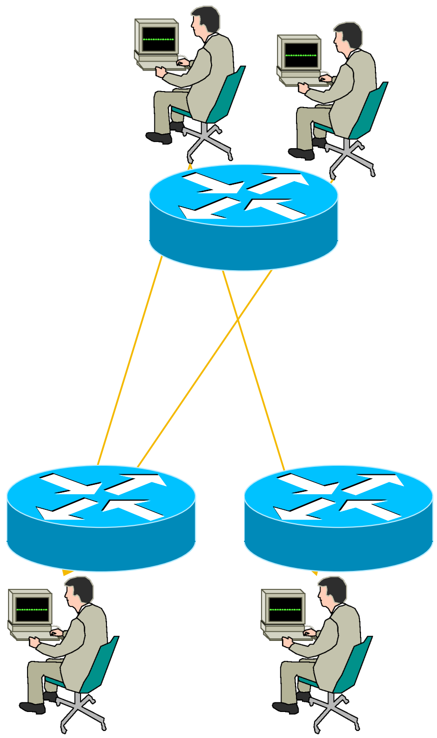
- Enet-1, Univ-3: simple routing design → redistribute entire IP space
- Univ-1: complex routing design → modify specific routing instances
  - Multiple routing instances add complexity
- Metric not absolute but higher means more complex

Inherent Complexity

- Policies determine a network's design and configuration complexity
  - Identical or similar policies
    - All-open or mostly-closed networks
    - Easy to configure
  - Subtle distinctions across groups of users:
    - Multiple roles, complex design, complex referential profile
    - Hard to configure: requires multiple special cases
- Challenges
  - Mining implemented policies
  - Quantifying similarities/consistency

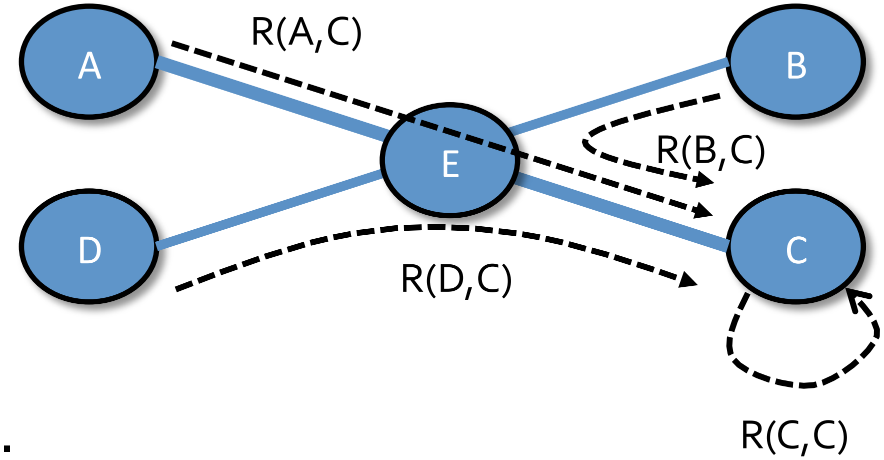
Capturing Network Policies With Reachability Sets

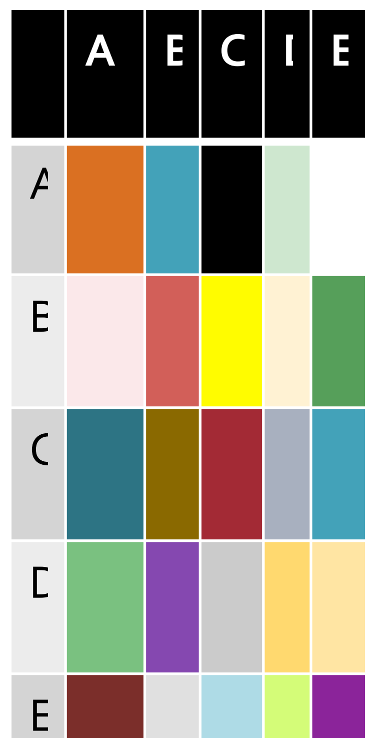
- Operator's goal = connectivity matrix between hosts
- Reachability set (Xie et al.) = set of packets allowed between 2 routers
  - One reachability set for each pair of routers (total of N^2 for a network with *N* routers)
- Reachability sets -> connectivity matrix between routers
  - Affected by data/control plane mechanisms
- Router level matrix
  - More efficient for computing set operations
  - No loss of information



Inherent Complexity: Uniformity Metric

- Variability in reachability sets between pairs of routers
- **Metric:** Uniformity
  - Entropy of reachability sets.
  - Simplest: *log(N)* → all routers should have same reachability to a destination C
  - Most complex: *log(N²)* → each router has a different reachability to a destination C





Empirical Results

Network	Entropy (diff from ideal)	
Univ-1	3.61	(0.03)
Univ-2	6.14	(1.62)
Univ-3	4.63	(0.05)
Univ-4	5.70	(1.12)
Enet-1	2.8	(0.0)
Enet-2	6.69	(0.22)
Enet-3	5.34	(1.09)

- Simple policies
  - Entropy close to ideal
- Univ-3 & Enet-1: simple policy
  - Filtering at higher levels
- Univ-1: **BUG!**
  - Router was not redistributing local subnet

Network (#routers)	Avg Ref links per router	#Routing instances
Univ-1 (12)	42	14

Our Foray into Complexity: Insights

- Implementation vs. inherent complexity
  - A few networks have simple configurations, but most are complex
  - Most of the networks studied have inherently simple policies
- Why is implementation complex?

Networks (#routers)	Ref links	Entropy (diff from ideal)
Univ-1 (12)	42	3.61 (0.03)
Univ-2 (19)	8	6.14 (1.62)
Univ-3 (24)	4	4.63 (0.05)
Univ-4 (24)	75	5.70 (1.12)
Enet-1 (10)	2	2.8 (0.0)
Enet-2 (83)	8	6.69 (0.22)
Enet-3 (19)	22	5.34 (1.09)

Our Foray into Complexity: Insights

- Network evolution
  - Univ-1: high referential link count due to dangling references (to interfaces)
  - Univ-2: caught in the midst of a major restructuring
- Optimizing for cost and scalability
  - Univ-1: simple policy, complex config
  - Cheaper to use OSPF on core routers and RIP on edge routers
    - Only RIP is not scalable
    - Only OSPF is too expensive

N/w (#rtrs)	Ref links per router	Entropy (ideal)
Univ-1 (12)	42	3.61 (3.58)
Univ-2 (19)	8	6.14 (4.52)

Related Work

- Reachability sets
  - Many studies on mining objectives/policies [e.g. Xie et al.] to check inconsistencies
- Measuring complexity
  - Protocol complexity [Ratnasamy et. al, Candea et al.]
  - Glue logic [Le et al.]: complexity of route redistribution in networks
- Informs clean slate
  - Inherent support for manageability [e.g., Ethane, 4D]

Conclusions

- Metrics that capture complexity of network design
  - Predict difficulty of making changes
- Empirical study of complexity
  - Evaluated commercial and public enterprises
- Results show:
  - Simple policies are often implemented in complex ways
  - Complexity introduced by non-technical factors
- Future work:
  - Apply to ISP Networks
  - Absolute vs. relative complexity