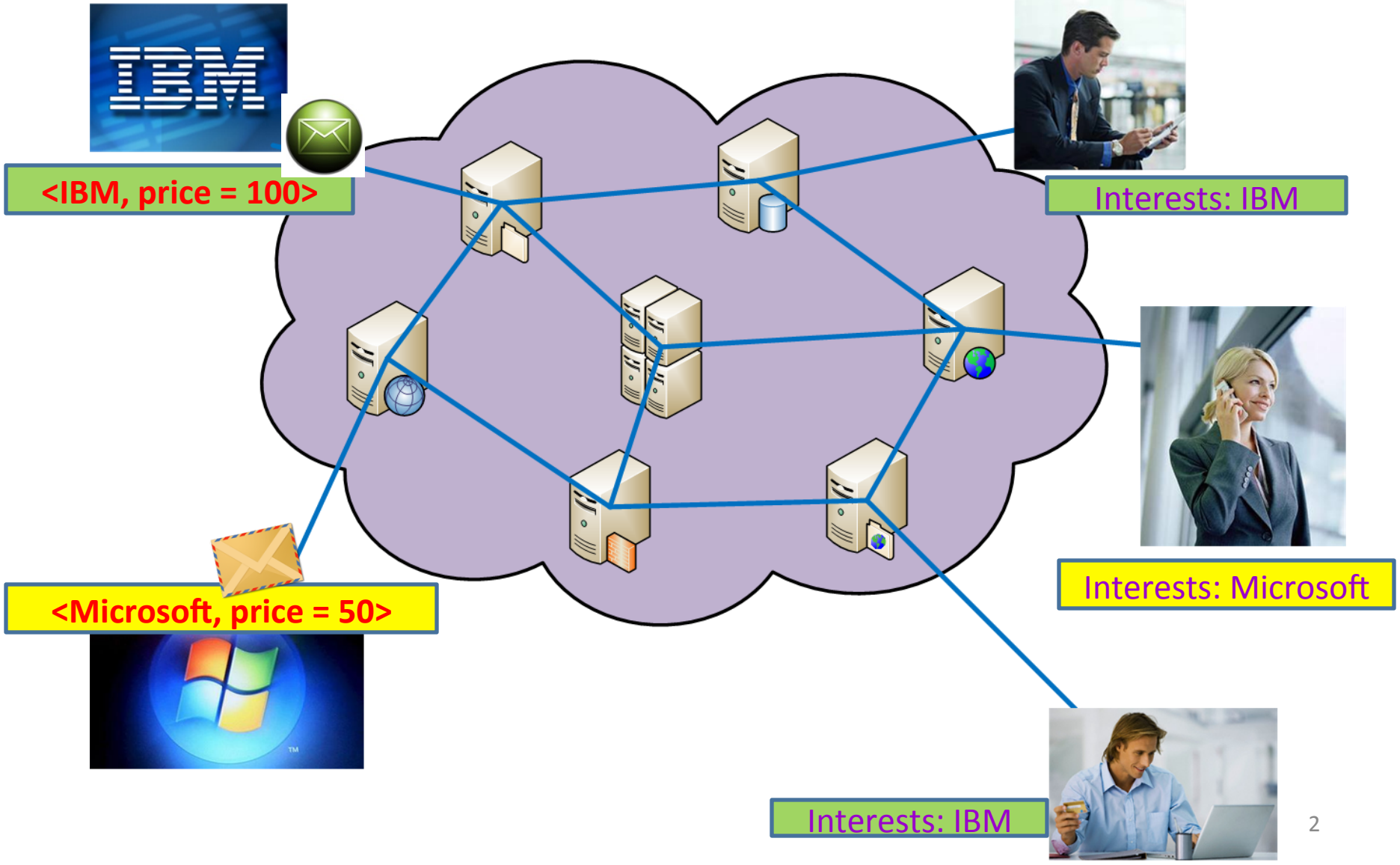
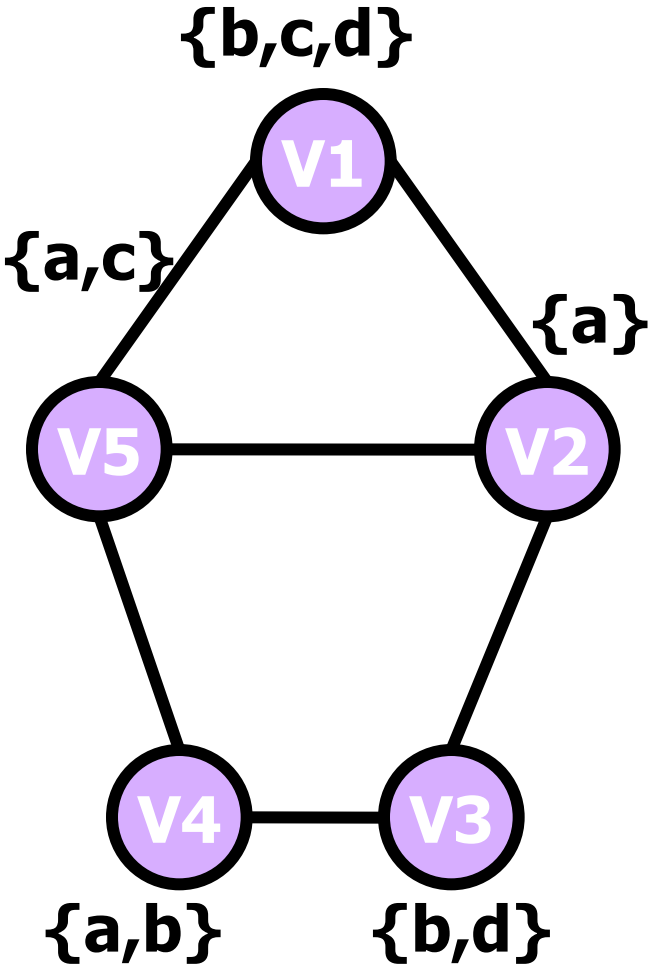
ElastO: Efficient Maintenance of Scalable Overlays for Topic-based Pub/Sub under Churn [Submission to Middleware '14]

Chen Chen
Joint work with Roman Vitenberg and Hans-Arno Jacobsen

Publish/Subscribe (pub/sub)

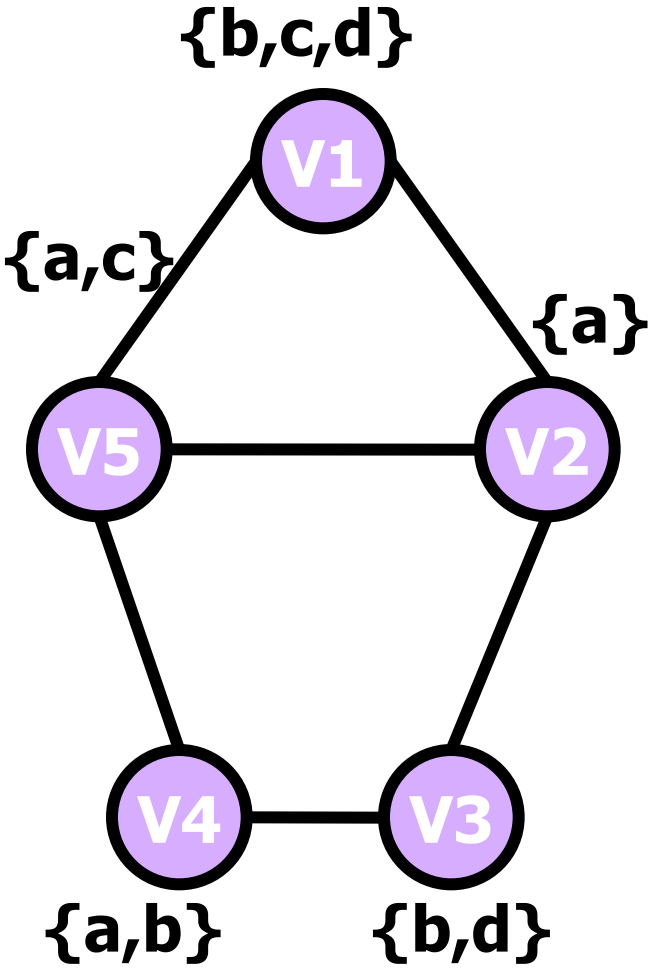


Topic-connected overlay (TCO)

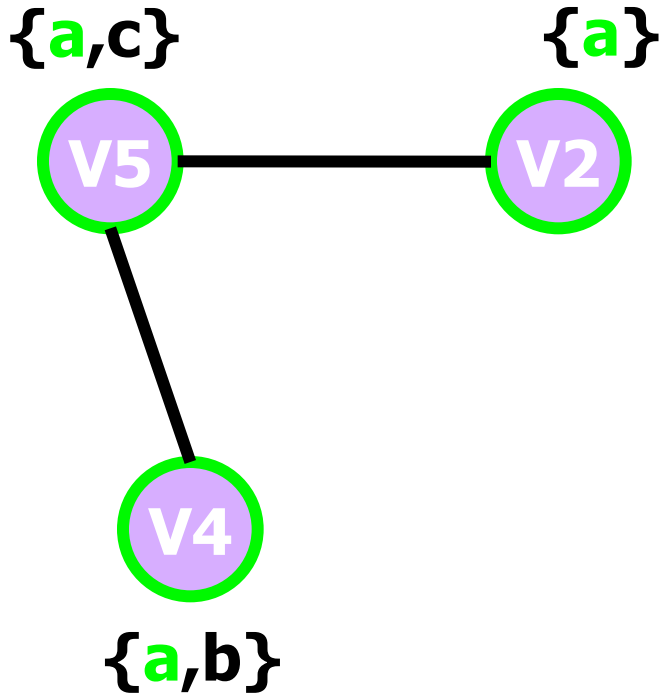


An overlay G

Topic-connected overlay (TCO)

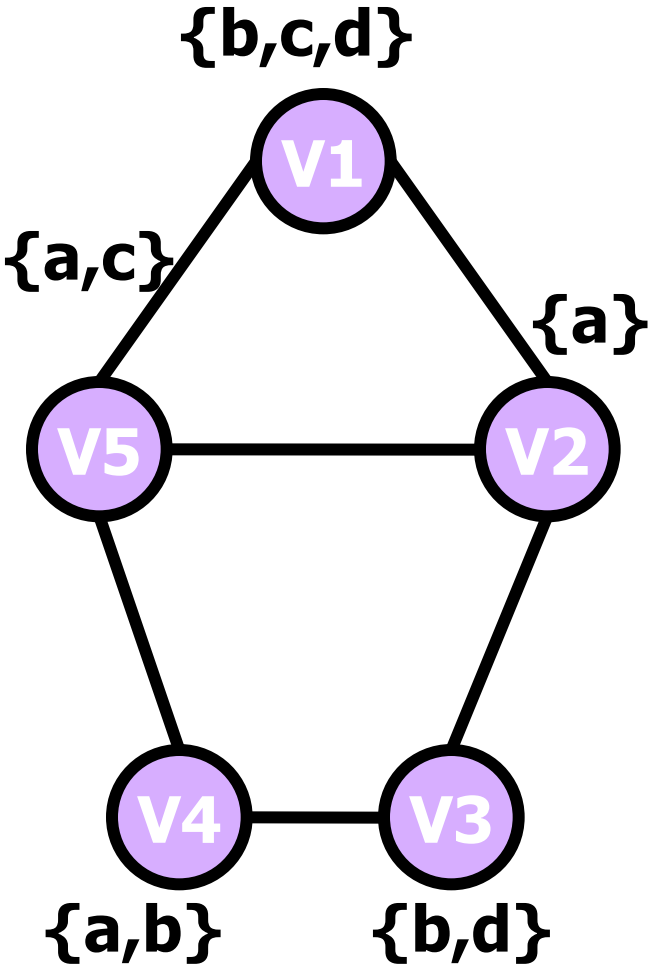


An overlay G

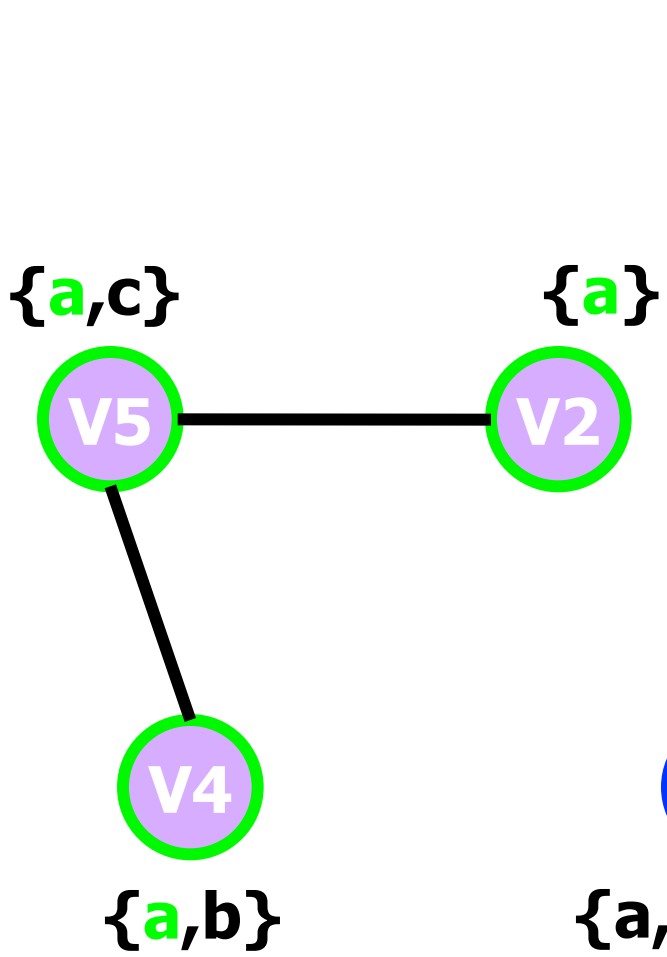


Ga is topic-connected
with one *TC-component*

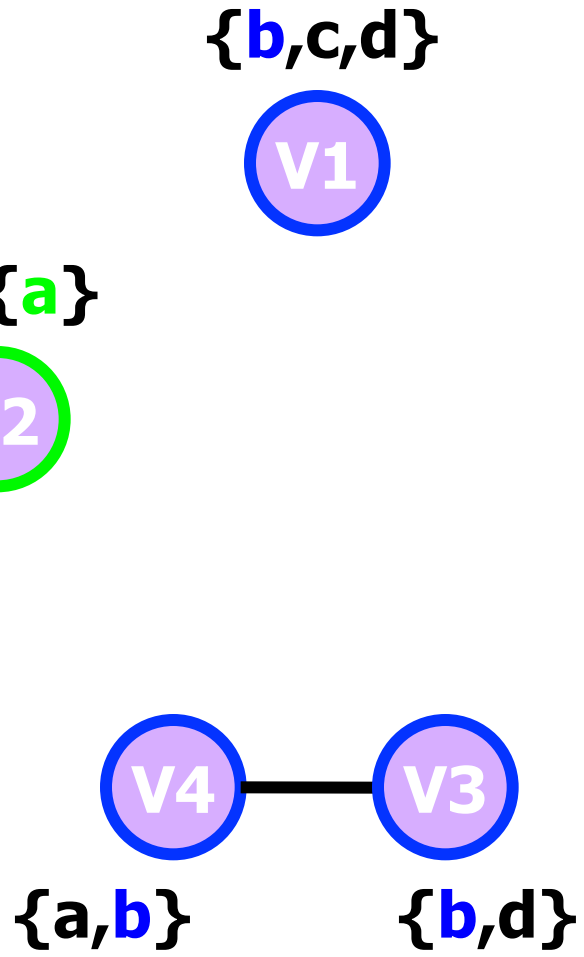
Topic-connected overlay (TCO)



An overlay G



Ga is topic-connected with one *TC-component*



Gb is NOT topic-connected with two *TC-components*

Approaches to build pub/sub TCO

		Knowledge	Churn Handling	Time	Average degree	Maximum degree
Centralized algorithms	Low-ODA	**Global**	✘	**Slow**	$O(\rho \ln \lvert V \rvert \lvert T \rvert)$	$O(\lvert V \rvert / \rho \ln \lvert V \rvert \lvert T \rvert)$
	GM				$O(\ln \lvert V \rvert \lvert T \rvert)$	$\Theta(\lvert V \rvert)$
	MinMax-ODA				$\Theta(\lvert V \rvert)$	$O(\ln \lvert V \rvert \lvert T \rvert)$
	DC				$O(p \ln \lvert V \rvert \lvert T \rvert)$	$\Theta(\lvert V \rvert)$
	DCBR-M				$\Theta(\lvert V \rvert)$	$O(\eta + \ln \lvert V \rvert \lvert T \rvert)$
	GM2				$O(U + \ln \lvert V \rvert \lvert T \rvert)$	$\Theta(\lvert V \rvert)$
Decentralized protocols	SpiderCast, PolderCast, StAN, etc.	**Local/Global**	✓	**Fast**	**Unknown**	**Unknown**

Our hybrid solution: ElastO

		Knowledge	Churn Handling	Time	Average degree	Maximum degree
Hybrid	**ElastO**	**Local**	✓	**Fast**	$\approx O(\rho \ln \lvert V\rvert \lvert T\rvert)$	$\approx O(\lvert V\rvert/\rho \ln \lvert V\rvert \lvert T\rvert)$
Centralized algorithms	Low-ODA	**Global**	✗	**Slow**	$O(\rho \ln \lvert V\rvert \lvert T\rvert)$	$O(\lvert V\rvert/\rho \ln \lvert V\rvert \lvert T\rvert)$
	GM				$O(\ln \lvert V\rvert \lvert T\rvert)$	$\Theta(\lvert V\rvert)$
	MinMax-ODA				$\Theta(\lvert V\rvert)$	$O(\ln \lvert V\rvert \lvert T\rvert)$
	DC				$O(p \ln \lvert V\rvert \lvert T\rvert)$	$\Theta(\lvert V\rvert)$
	DCBR-M				$\Theta(\lvert V\rvert)$	$O(\eta + \ln \lvert V\rvert \lvert T\rvert)$
	GM2				$O(U + \ln \lvert V\rvert \lvert T\rvert)$	$\Theta(\lvert V\rvert)$
Decentralized protocols	SpiderCast, PolderCast, StAN, etc.	**Local/Global**	✓	**Fast**	**Unknown**	**Unknown**

Requirements

A. **Overlay quality**
   - TCO, low node degrees, small diameters
   - Close to centralized algorithms

B. **Responsiveness**
   - Comparable to decentralized protocols

C. **Scalability of the decentralized solution**
   - No centralized control
   - Partial and local view
   - Churn handling only impacts a small portion of nodes

D. **Fairness and load balancing**
   - Computation, communication, storage

E. **Reliability** against concurrent churn events

Our contributions - ElastO

- Complete architecture and protocol design
- Shadow-based strategies

  TCO recovery under churn
- Primary-backup mechanism

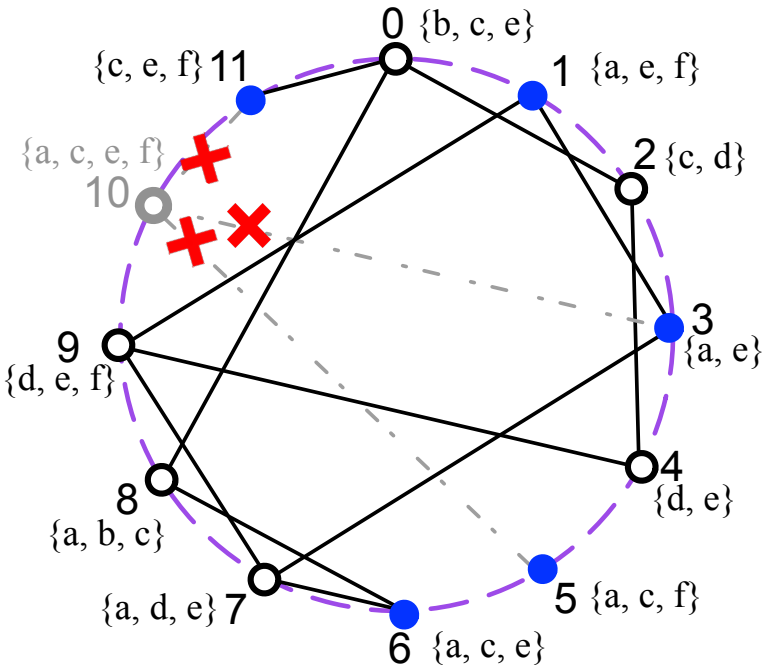
  support shadow sets in decentralized systems
- Gossip-based peer sampling service with unique local view selection algorithms

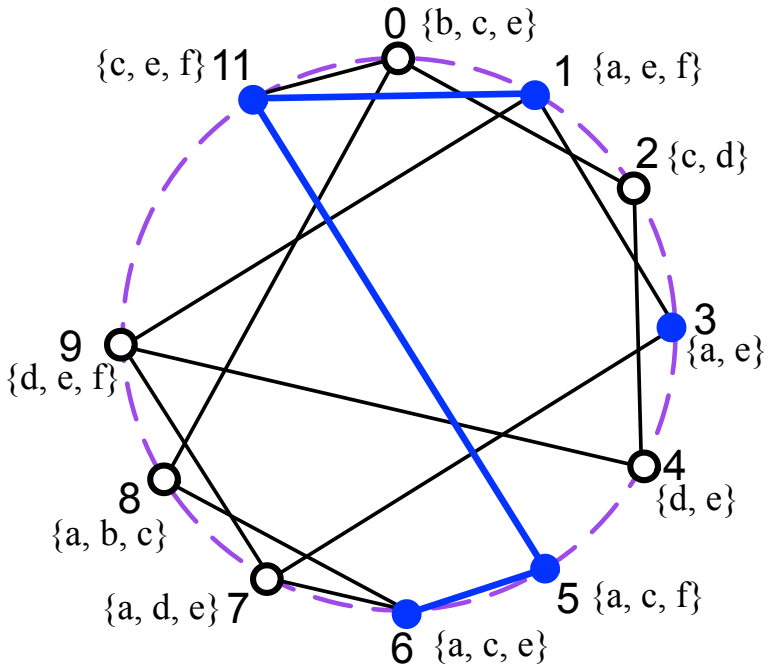
  build and deploy backup sets for all nodes
- Comprehensive evaluation

Shadow-based strategies

Upon each churn event

- Select a *proper* subset of nodes, namely the **shadow set**, for TCO recovery

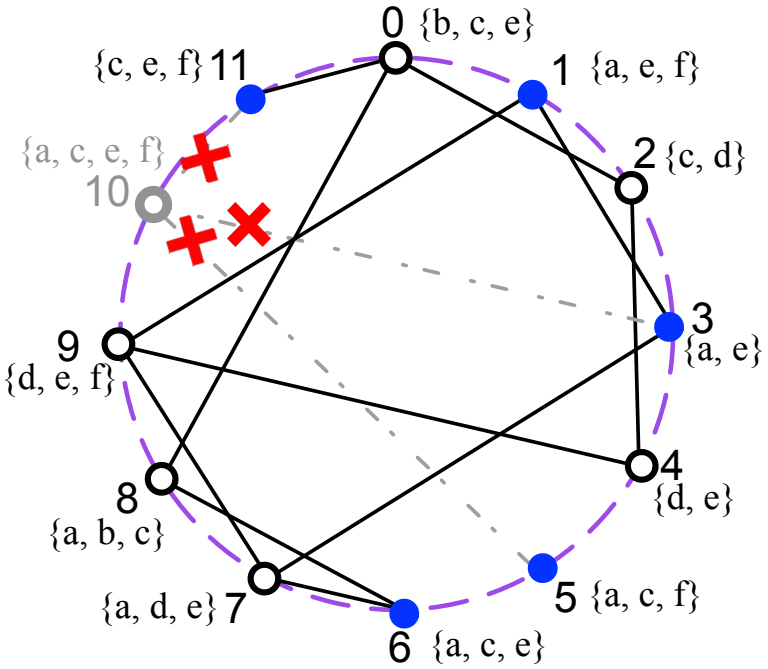


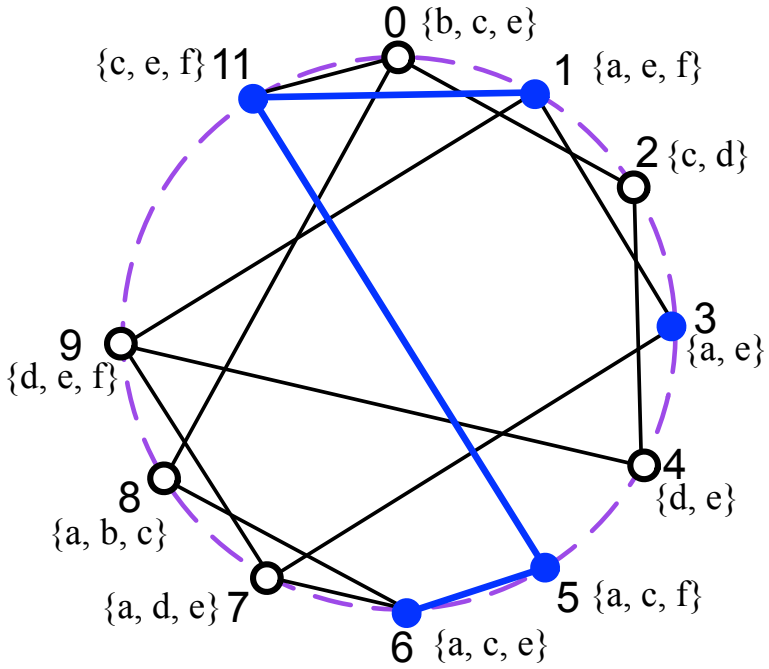


Pre-compute the shadow set
primary-backup mechanism

- ElastO needs to remember
  1) TCO neighbors, $v_{10}.N = \{v_3, v_5, v_{11}\}$
  2) backup set, $B(v_{10}) = \{v_1, v_6\}$

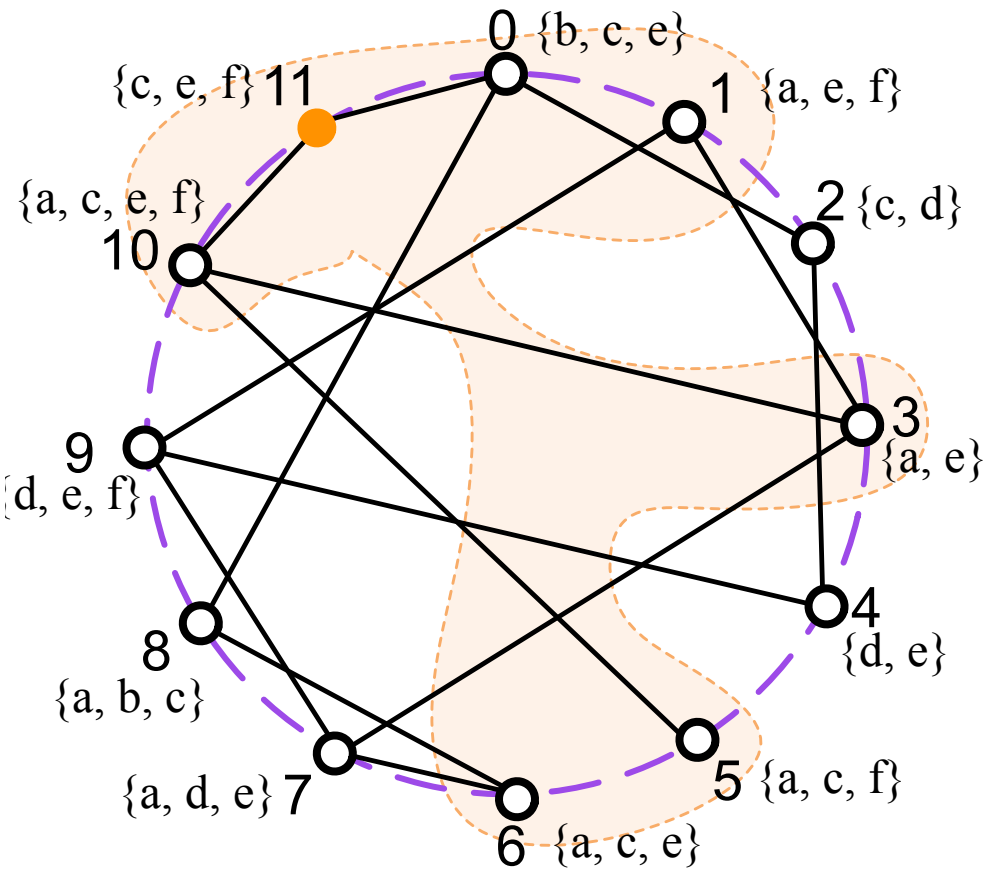
  in preparation of v10's leaving





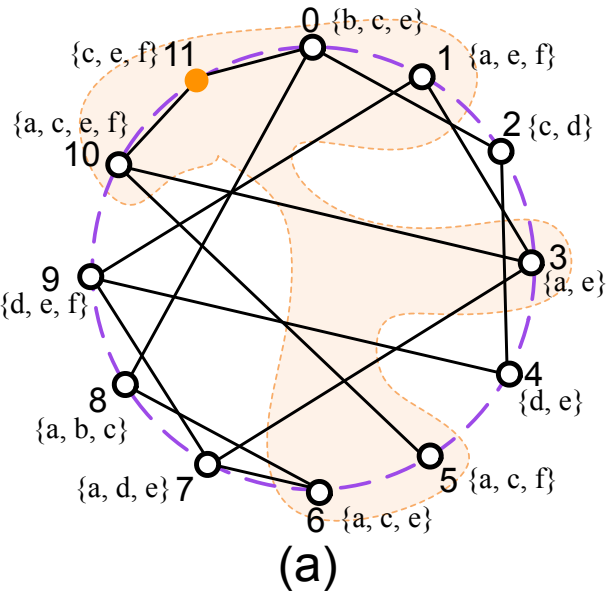
Local view of each node

$.self$	v_{11}
$.\mathscr{N}$	$\{v_0, v_{10}\}$
$.\mathscr{D}$	$\emptyset$
$.succ[1]$	v_0
$.pred[1]$	v_{10}
$.pneighb[1]$	$\{v_3, v_5, v_{11}\}$
$.backup[1]$	$\{v_1, v_6\}$



Example: initially stable

$.self$	v_{11}
$.\mathscr{N}$	$\{v_0, v_{10}\}$
$.\mathscr{D}$	$\emptyset$
$.succ[1]$	v_0
$.pred[1]$	v_{10}
$.pneighb[1]$	$\{v_3, v_5, v_{11}\}$
$.backup[1]$	$\{v_1, v_6\}$

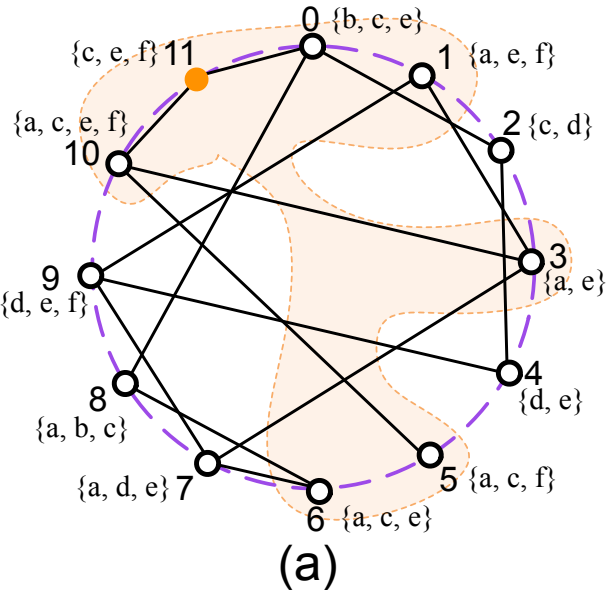


(a)

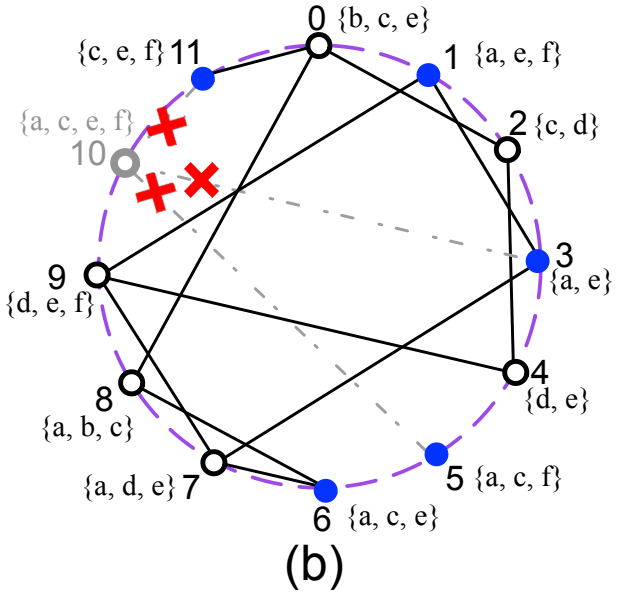
Example: node departure

$.self$	v_{11}
$.\mathscr{N}$	$\{v_0, v_{10}\}$
$.\mathscr{D}$	$\emptyset$
$.succ[1]$	v_0
$.pred[1]$	v_{10}
$.pneighb[1]$	$\{v_3, v_5, v_{11}\}$
$.backup[1]$	$\{v_1, v_6\}$

$.self$	v_{11}
$.\mathscr{N}$	$\{v_0, v_{10}\}$
$.\mathscr{D}$	$\{v_{10}\}$
$.succ[1]$	v_0
$.pred[1]$	v_{10}
$.pneighb[1]$	$\{v_3, v_5, v_{11}\}$
$.backup[1]$	$\{v_1, v_6\}$



(a)



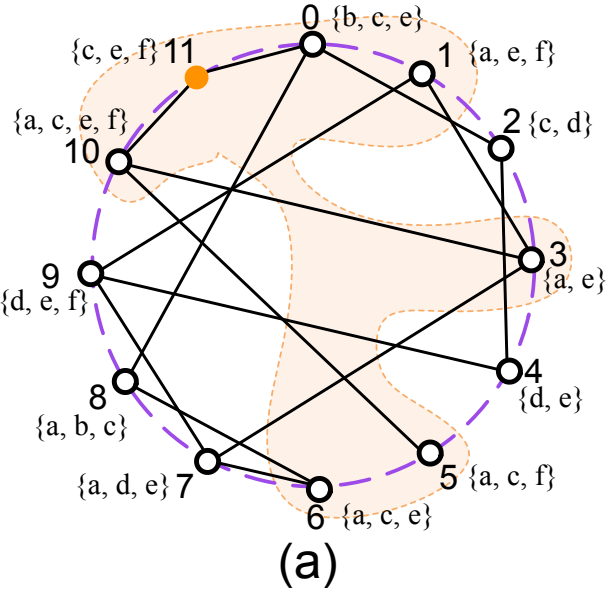
(b)

Example: node departure handled

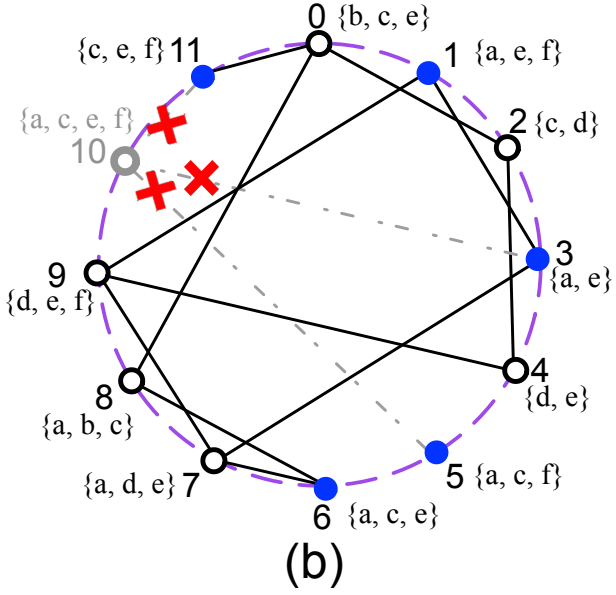
$.self$	v_{11}
$.\mathcal{N}$	$\{v_0, v_{10}\}$
$.\mathcal{D}$	$\emptyset$
$.succ[1]$	v_0
$.pred[1]$	v_{10}
$.pneighb[1]$	$\{v_3, v_5, v_{11}\}$
$.backup[1]$	$\{v_1, v_6\}$

$.self$	v_{11}
$.\mathcal{N}$	$\{v_0, v_{10}\}$
$.\mathcal{D}$	$\{v_{10}\}$
$.succ[1]$	v_0
$.pred[1]$	v_{10}
$.pneighb[1]$	$\{v_3, v_5, v_{11}\}$
$.backup[1]$	$\{v_1, v_6\}$

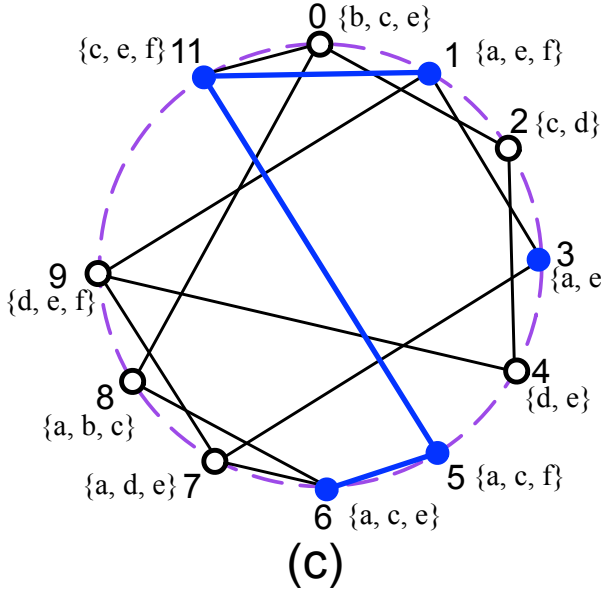
$.self$	v_{11}
$.\mathcal{N}$	$\{v_0, v_1, v_5\}$
$.\mathcal{D}$	$\{\}$
$.succ[1]$	v_0
$.pred[1]$	v_9
$.pneighb[1]$	$\{v_1, v_4, v_7\}$
$.backup[1]$	$\{v_5, v_7\}$



(a)



(b)



(c)

Evaluation: experimental setup

- Real-world pub/sub workloads

- Synthetic pub/sub workloads

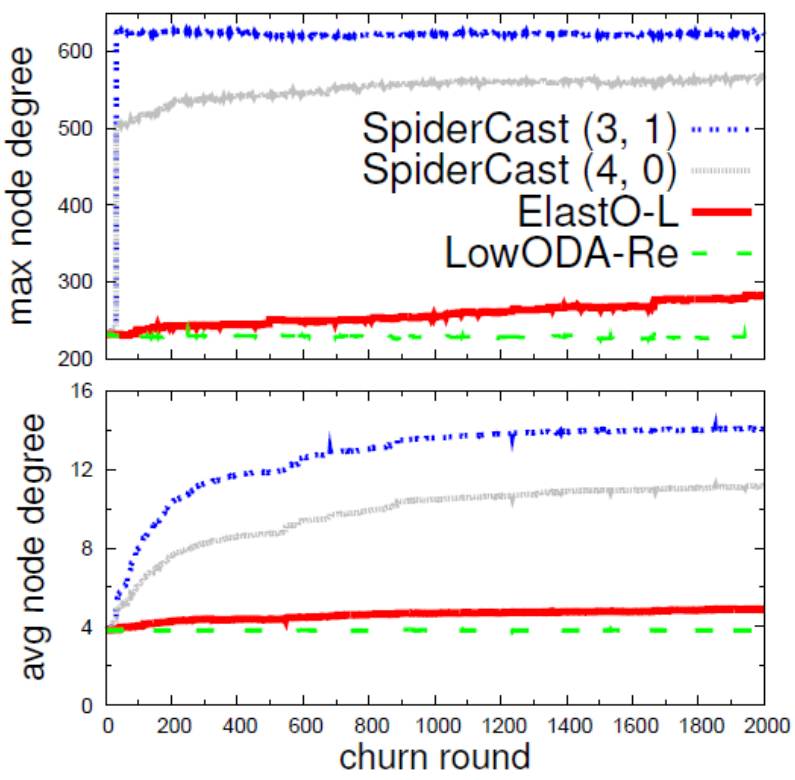
  Expo, Zipf, Unif

- Churn traces 

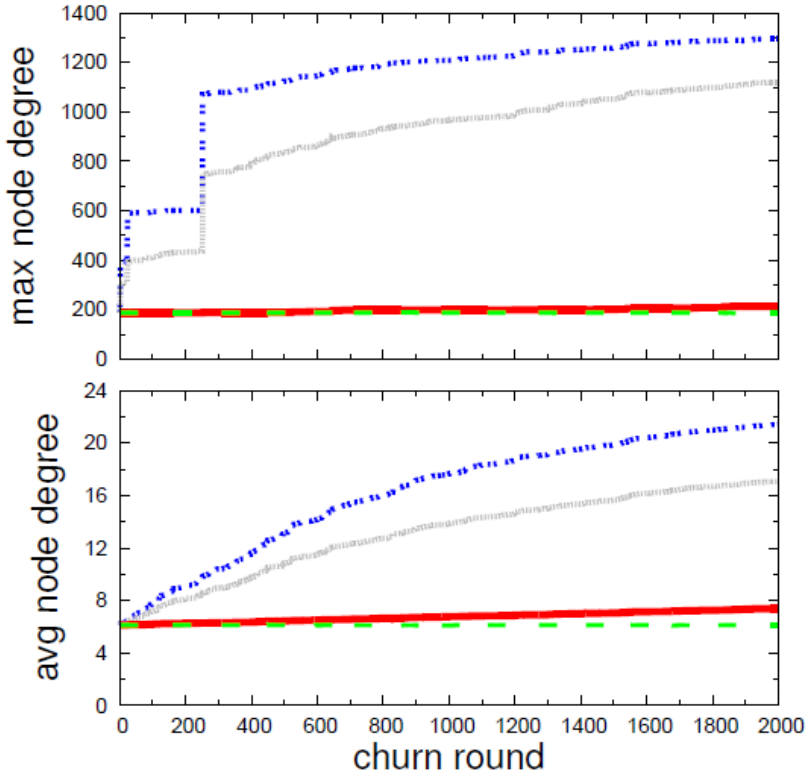
Evaluation: algorithms and protocols

ElastO	**Our proposed system**
ElastO-L	Local view at each node
ElastO-G	Global view at each node
LowODA	**Low Max and Avg Degree Overlay Design Algorithm**
LowODA-Inc	Incrementally repair TCO regarding existing links
LowODA-Re	Reconstruct TCO from scratch regardless of existing links
SpiderCast	**A peer-to-peer protocol to build TCO**
SpiderCast(Kg,Kr)	Two neighbor selection heuristics: greedy and random

Evaluation: node degrees under churn

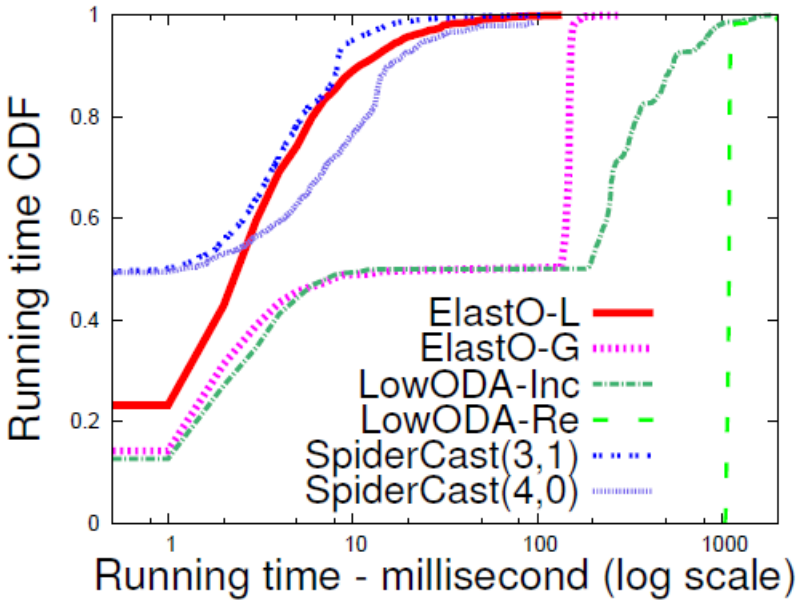


(a) FB 1K

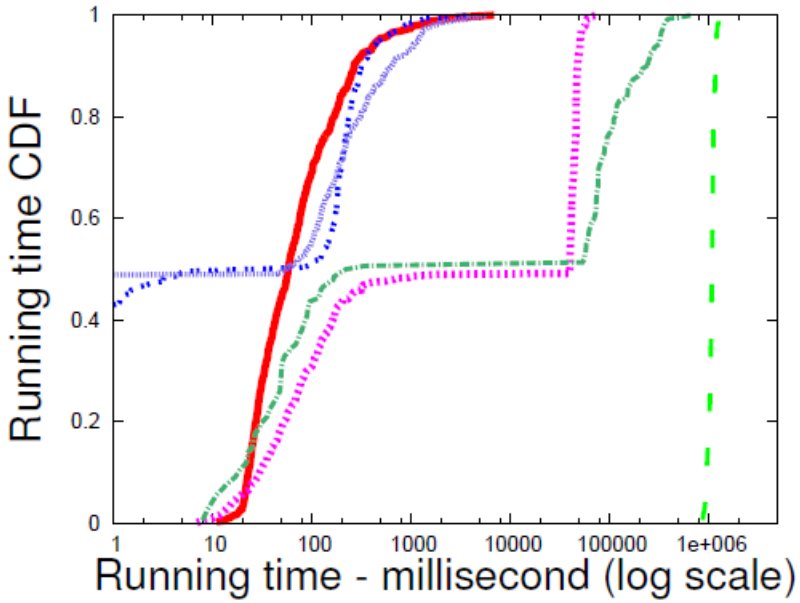


(b) FB 10K

Evaluation: runtime cost



(c) TW 1K



(d) TW 10K

Conclusion

		Knowledge	Churn Handling	Time	Average degree	Maximum degree
Hybrid	**ElastO**	**Local**	✓	**Fast**	$\approx O(\rho \ln \lvert V \rvert \lvert T \rvert)$	$\approx O(\lvert V \rvert / \rho \ln \lvert V \rvert \lvert T \rvert)$
Centralized algorithms	Low-ODA	**Global**	✘	**Slow**	$O(\rho \ln \lvert V \rvert \lvert T \rvert)$	$O(\lvert V \rvert / \rho \ln \lvert V \rvert \lvert T \rvert)$
	GM				$O(\ln \lvert V \rvert \lvert T \rvert)$	$\Theta(\lvert V \rvert)$
	MinMax-ODA				$\Theta(\lvert V \rvert)$	$O(\ln \lvert V \rvert \lvert T \rvert)$
	DC				$O(p \ln \lvert V \rvert \lvert T \rvert)$	$\Theta(\lvert V \rvert)$
	DCBR-M				$\Theta(\lvert V \rvert)$	$O(\eta + \ln \lvert V \rvert \lvert T \rvert)$
	GM2				$O(U + \ln \lvert V \rvert \lvert T \rvert)$	$\Theta(\lvert V \rvert)$
Decentralized protocols	SpiderCast, PolderCast, etc.	**Local/Global**	✓	**Fast**	**Unknown**	**Unknown**

Current research in IBM

Distributed pub/sub for federated messaging and IoT

- Scalability: 100 Million users/topics/subscriptions
- Quality of Service (QoS) for MQTT
  - 0: at most once
  - 1: at least once
  - 2: exactly once
- Topics and subscriptions
  - hierarchy and wildcards
  - e.g., sensors/+/temperature/+
- Overlay topology and routing protocols
- Membership management