Traps and Pitfalls of Using Contact Traces in Performance Studies of Opportunistic Networks

Nikodin Ristanovic, George Theordorakopoulos and Jean-Yves Le Boudec.

Introduction

- Background Research
- The Experiment
  - Design goals
  - Architecture
- Contact trace results
- Meta-data results
- Conclusions

Background Research

- Validation of simulation model results
- Contact-based opportunistic communication
- Effect of backbones in opportunistic communication

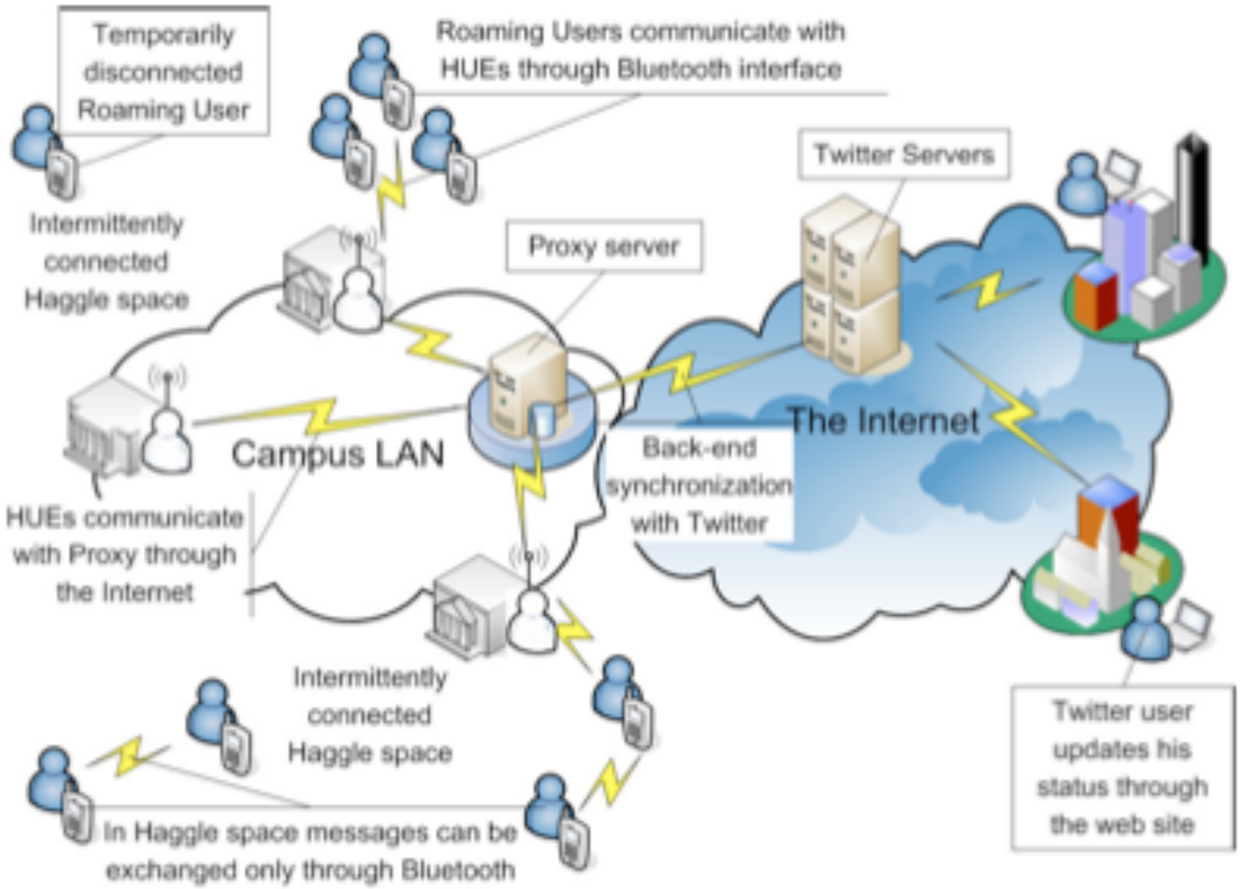
The Experiment - design goals

- Collect application data to extract performance metrics
- Collect contact traces to be used in event simulations
- Compare results with contact-based simulation values

The Experiment - architecture

- 50 Roaming users (RU)
- 10 Home user equivalents (HUE)
- 1 Proxy
- Opportunistic twitter application
- Haggle pub/sub
- 3 RU Caching strategies
- Context-based HUE Caching strategy

The Experiment - architecture



Contact Trace Results

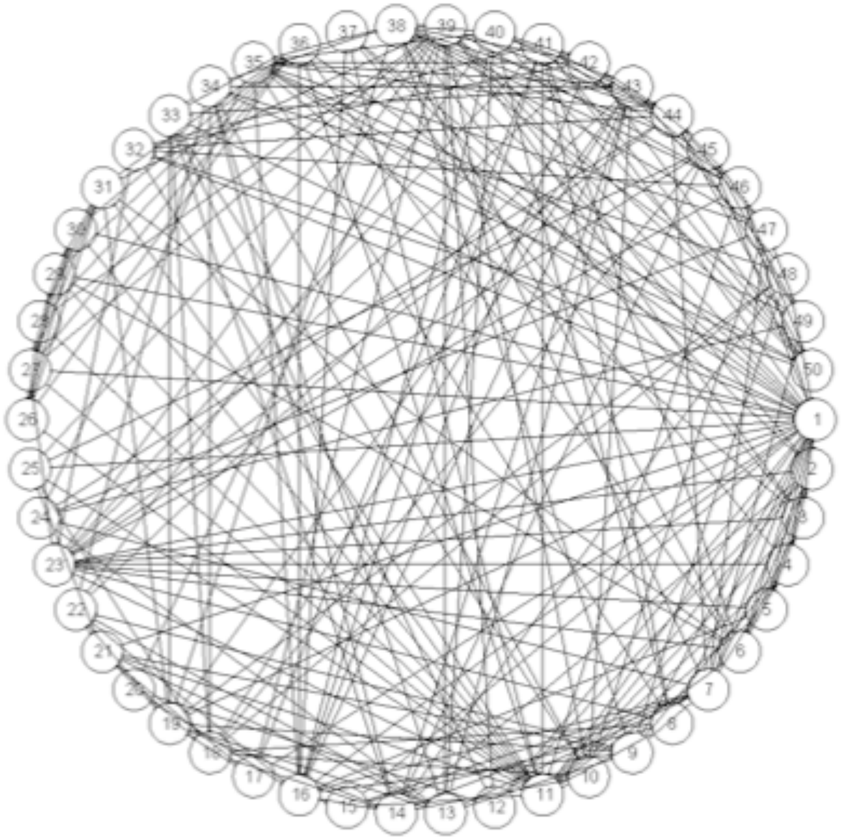


Fig. 2. "Twitter following relationships" between internal users. An edge between two users means that one of the users follows the other one.

Contact trace Results

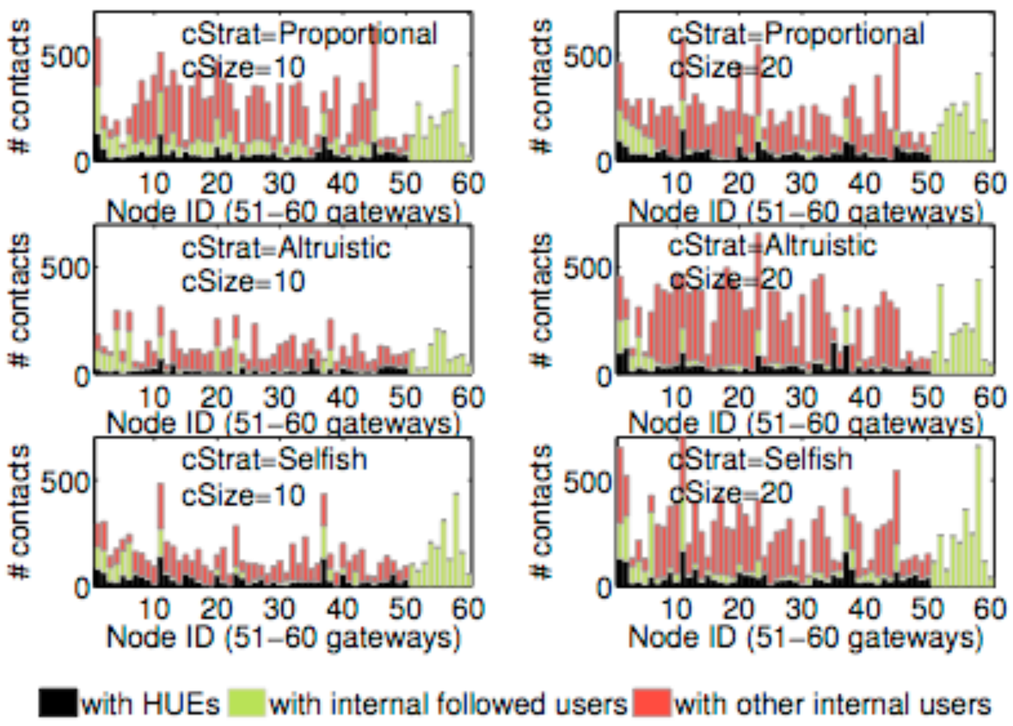


Fig. 3. Total number of contacts experienced by internal users ($j = 1, .., 50$) and HUEs ($j = 51, .., 60$) during 2-day evaluations of the 6 combinations of caching strategy $cStrat$ and cache size $cSize$.

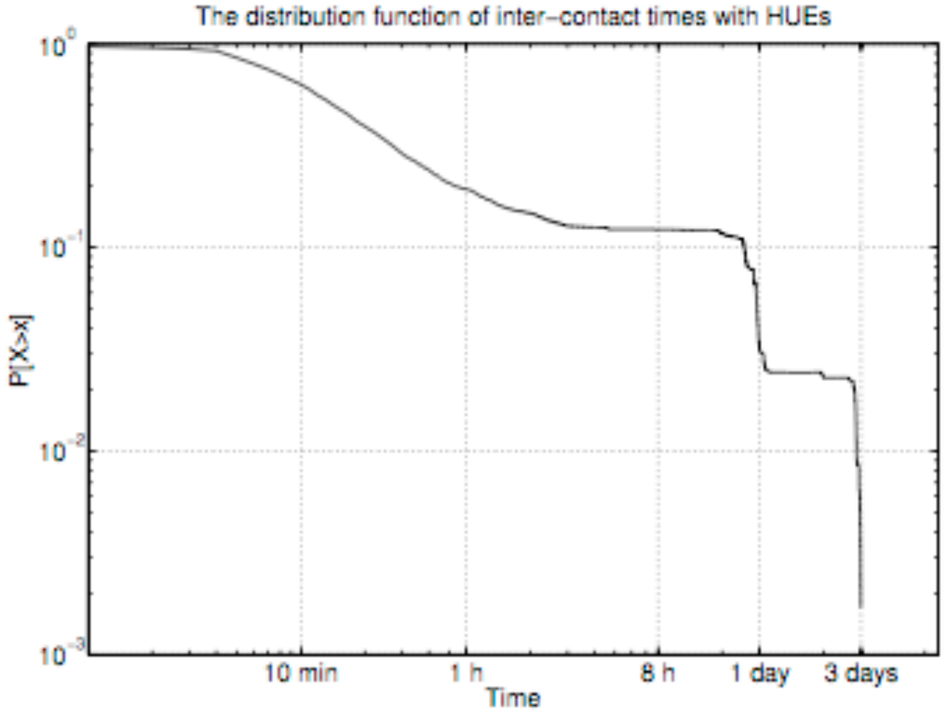


Fig. 4. The distribution function of inter-contact times with HUEs obtained for the whole duration of the experiment.

Meta-data Results

$$R_j^{\mathcal{A}} = \frac{M_{\mathcal{A}\to j}}{M_{\mathcal{A}\cap\mathcal{F}_j\to}}.$$

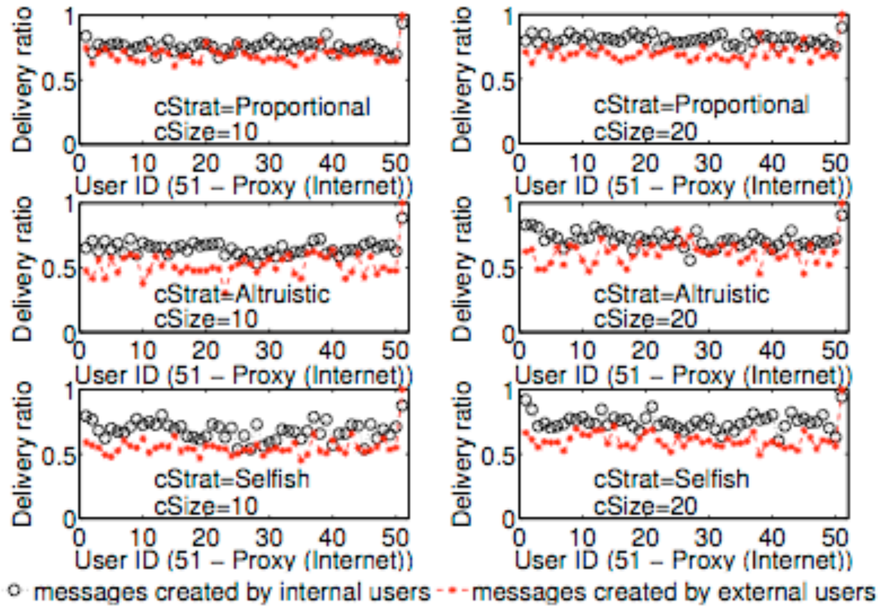
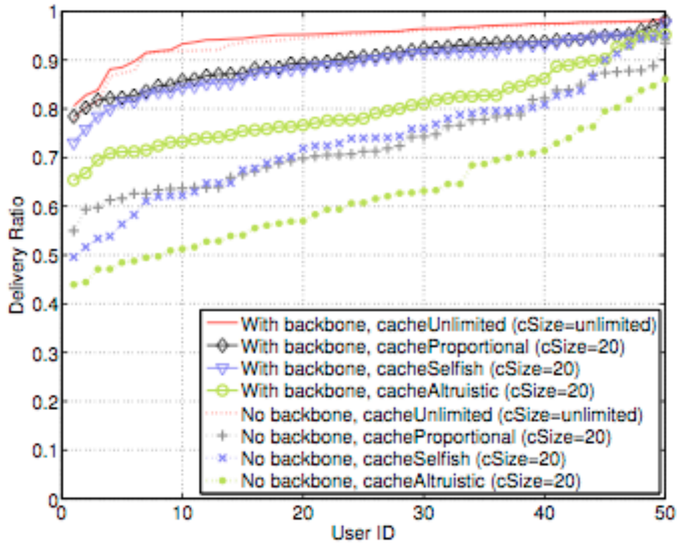
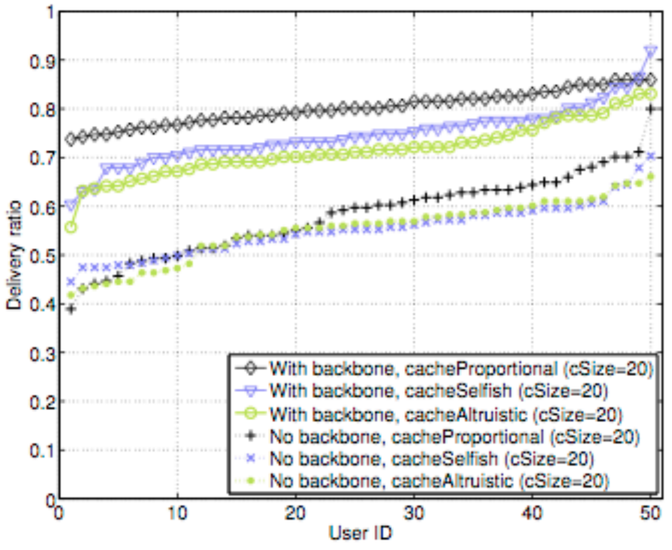


Fig. 5. Internal and external delivery ratios, $R_j^{\mathcal{N}}$ and $R_j^{\mathcal{X}}$, seen by the internal users ($j = 1, ..., 50$) and by the proxy ($j = 51$). Every combination of caching strategy ($cStrat$) and cache size ($cSize$) was evaluated during 2 days.



(a) Delivery ratios - simulation



(b) Delivery ratios - experiment

Fig. 6. Delivery ratios obtained from the simulations and from the experiment for different caching strategies. The full lines correspond to the system with the backbone (HUEs, proxy), while the dotted lines describe the system without the backbone.

Meta-data Results

$$D_j^{\mathcal{A}} = \frac{\sum_{m \in \mathcal{M}_{\mathcal{A} \to j}} D_j^m}{M_{\mathcal{A} \to j}}$$

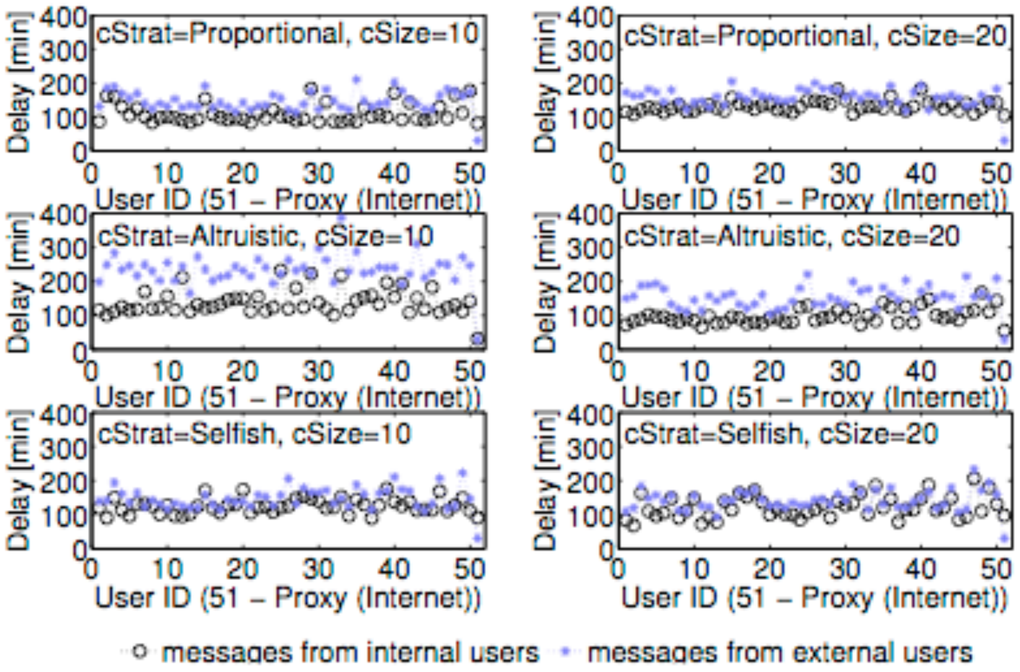


Fig. 7. The average age of received message observed by internal users. Every combination of caching strategy ($cStrat$) and cache size ($cSize$) was evaluated during 2 working days.

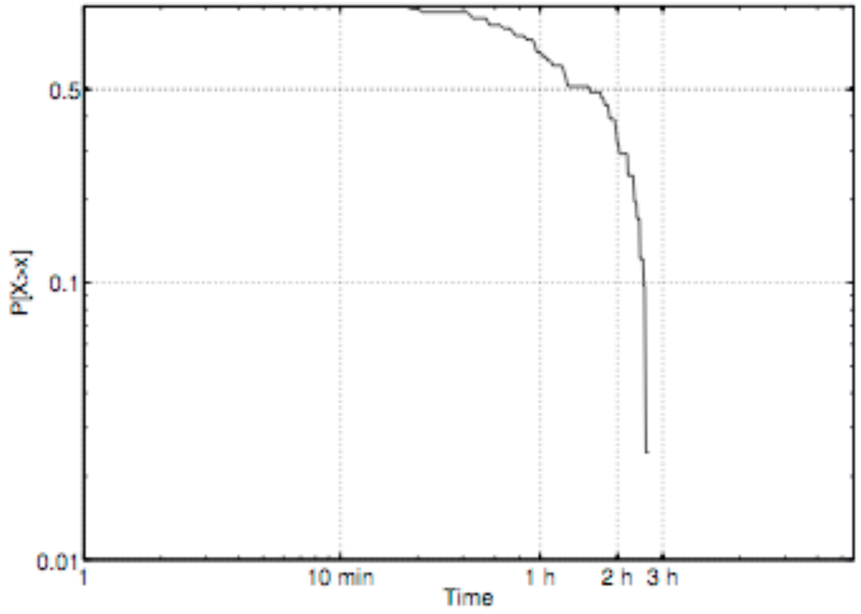
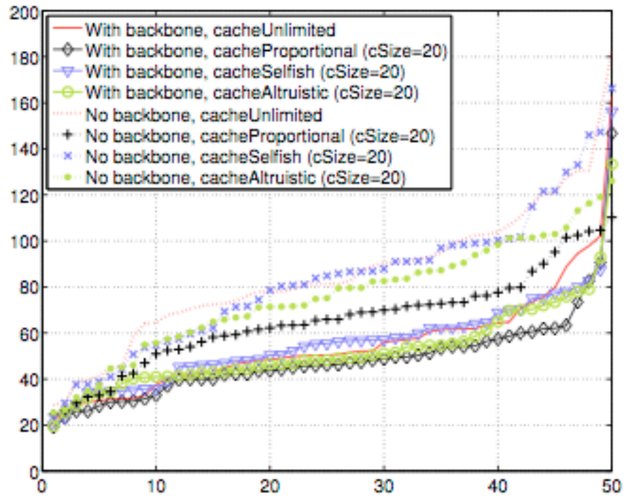
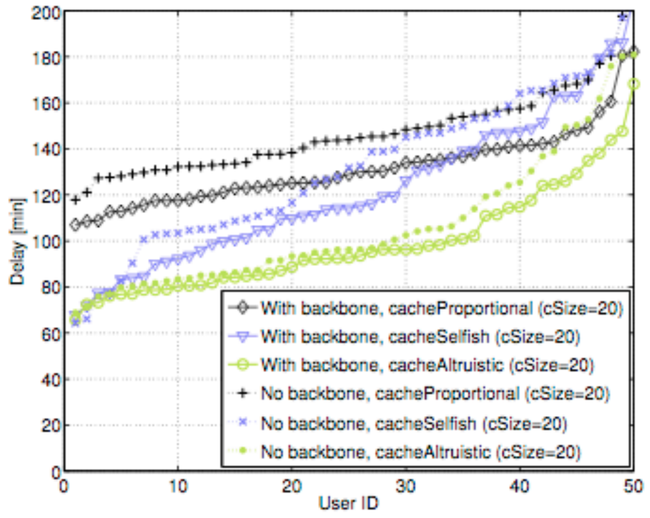


Fig. 9. The distribution function of observed delays for the tweets whose reception led to the creation of @replies by the recipients.

Meta-data Results



(a) Delays - simulation



(b) Delays - experiment

Fig. 8. Delays obtained from the simulations and from the experiment for different caching strategies. The full lines correspond to the system with the backbone, while the dotted lines describe the system without the backbone. The case with unlimited caches is also simulated.

$$d_i(\lambda) = \frac{\sum_{j \in \mathcal{N} \setminus \{i\} \cup \{I\}} d_{ij}(\lambda)}{N},$$

$$R_i = \frac{1}{1 + k d_i(\lambda)}, \lambda = 0.95,$$

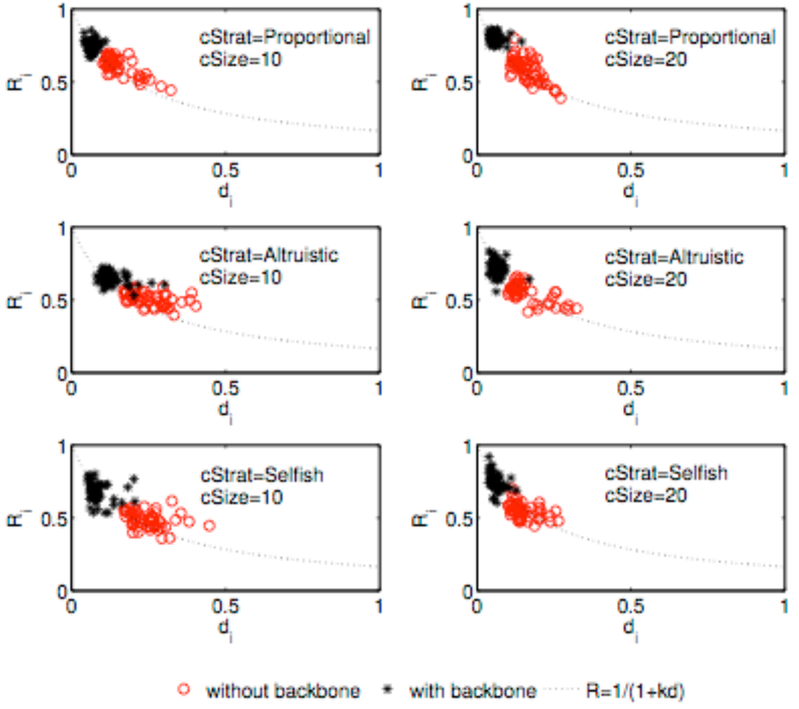


Fig. 10. Dependency between delivery ratio R_i and closeness centrality d_i.

Conclusions

- Technological limitations need to be simulated
- Limiting cache size can aid architecture development
- Statistical predictions can aid system performance

Thanks

- Any Questions?