SCoLE: Scalable Cooperative Latency Estimation

Michal Szymaniak Guillaume Pierre Maarten van Steen

Vrije Universiteit Amsterdam

{michal,gpierre,steen}@cs.vu.nl

Problem

- Context:
  - Large wide-area network
    - e.g.: Internet
  - Distributed system with M nodes
    - M is **very large**, say O(million)
    - e.g., peer-to-peer file-sharing platform

- How to estimate latencies between arbitrary nodes?
  - Quite easy, as long as M is very small..
  - ..but much harder, once M becomes large

Solution: Network Positioning

- GNP -- Global Network Positioning
  - by T.S. Eugene Ng and Hui Zhang (CMU)
  - Model the Internet as N-dimensional geometric space
  - For each node H, calculate its position P(H) in the space
  - **For any 2 nodes A and B**:
    - latency(A,B) ~ distance(P(A),P(B))
    - ~ == estimate with

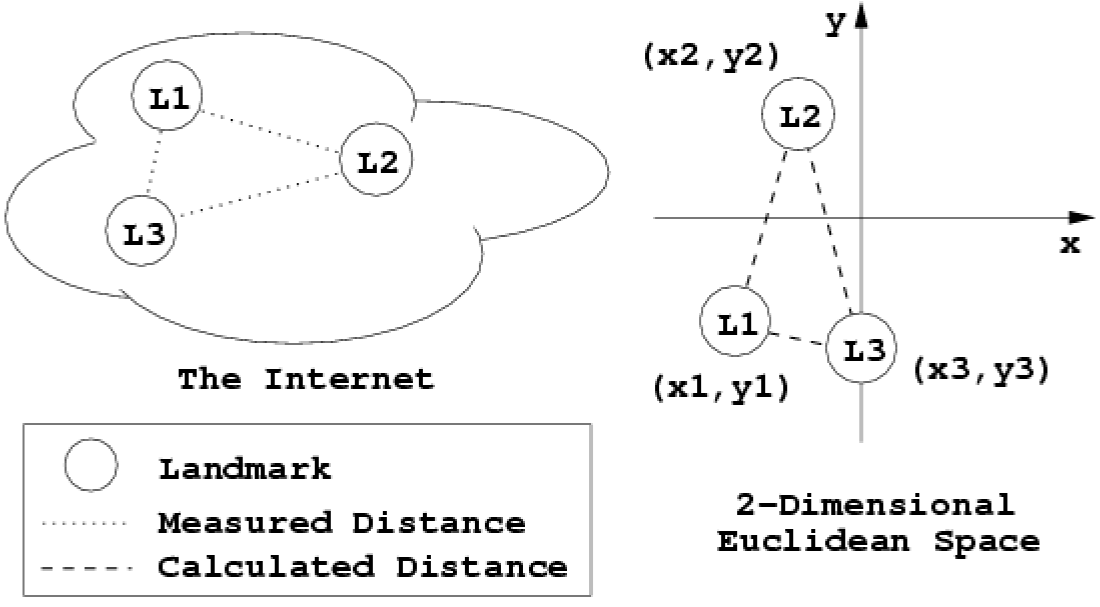
- Main benefit:
  - In a system with M nodes, GNP reduces the number of necessary measurements:
    - all-to-all : $O(M^2)$
    - GNP : $O(M)$

Talk Agenda

- GNP
  - Details
  - Performance
  - Limitations
- SCoLE
  - Personalized GNP
  - Architecture
  - Deployment
  - Prototype
- Conclusion

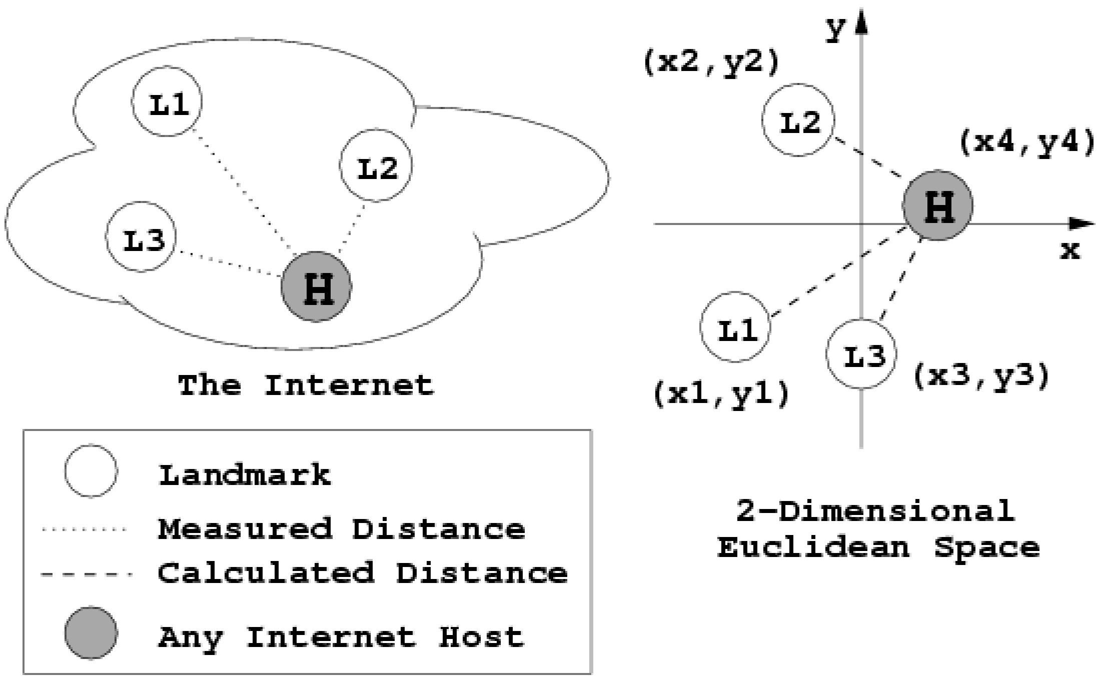
GNP: Space Construction

- N-dimensional space defined by N+1 reference nodes:
  - Select N+1 reference nodes, called landmarks: Li, 1 <= i <= N+1
  - Measure the latency between each pair of landmarks
  - Assign landmark positions P(Li) such that:
    - For any i,j: distance(P(Li),P(Lj)) ~ latency(Li,Lj)
    - In practice: minimize the global distance-vs-latency error



GNP: Node Positioning

- Node H positioning:
  - Measure the latencies between H and each landmark Li
  - Assign P(H) such that:
    - For any i: distance(P(H),P(Li)) ~ latency(H,Li)
    - Again, apply global error minimization

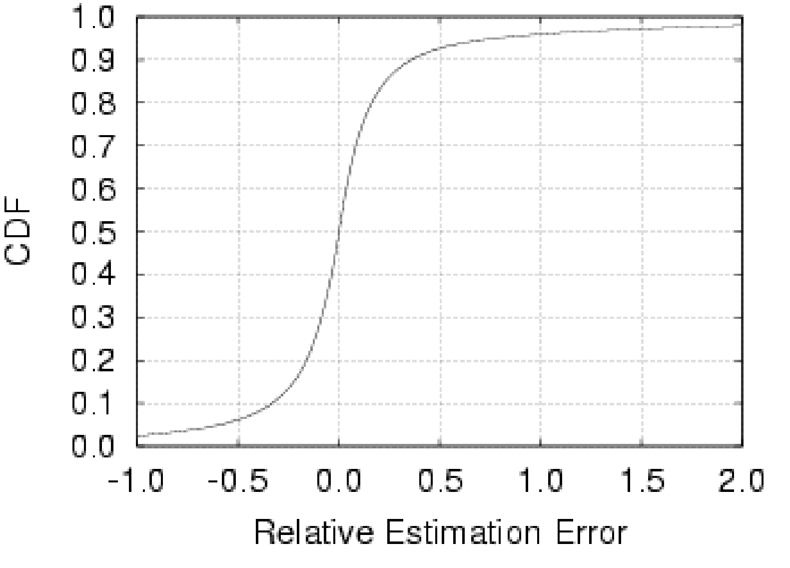


GNP: Cost/Performance

- Cost:
  - N is a small **constant,** we use N=6
  - In terms of measurements performed:
    - Space construction : **O(1)** (21 for N=6, clique of 7 landmarks)
    - Single node positioning : **O(1)** (7 for N=6, 1 per landmark)
    - Total for M nodes : **O(M)** (21 + 7 * M)
    - Single latency estimation : **0** (once the positioning is done)

- Performance:
  - For **90%** of latency estimations:

    $^2/_3$ real < estimated < $^3/_2$ real



GNP: Limitations

- GNP uses global landmarks
  - All the nodes must agree on which landmarks they use
    - Global negotiation + global knowledge = limited scalability
  - The same landmarks seldom suit all the nodes
    - Lack of flexibility

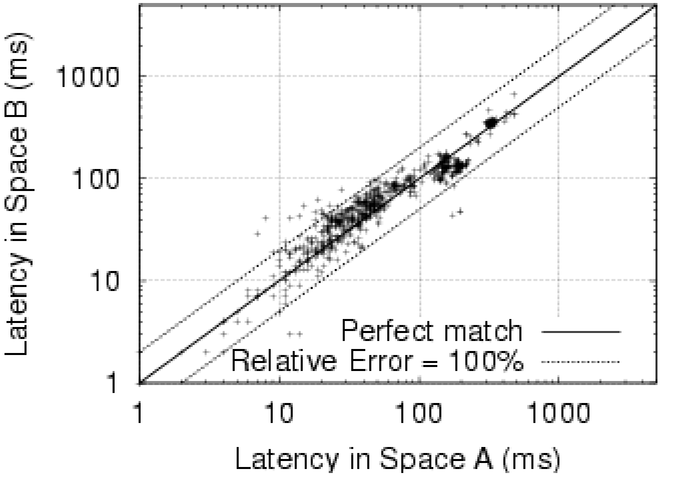
- Both problems can be removed..
  - ..if only we let nodes choose their landmarks.

  - But how can we calculate global positions then?
  - We can’t. But we do not need them, either.

  - Hint: we only care about latencies.

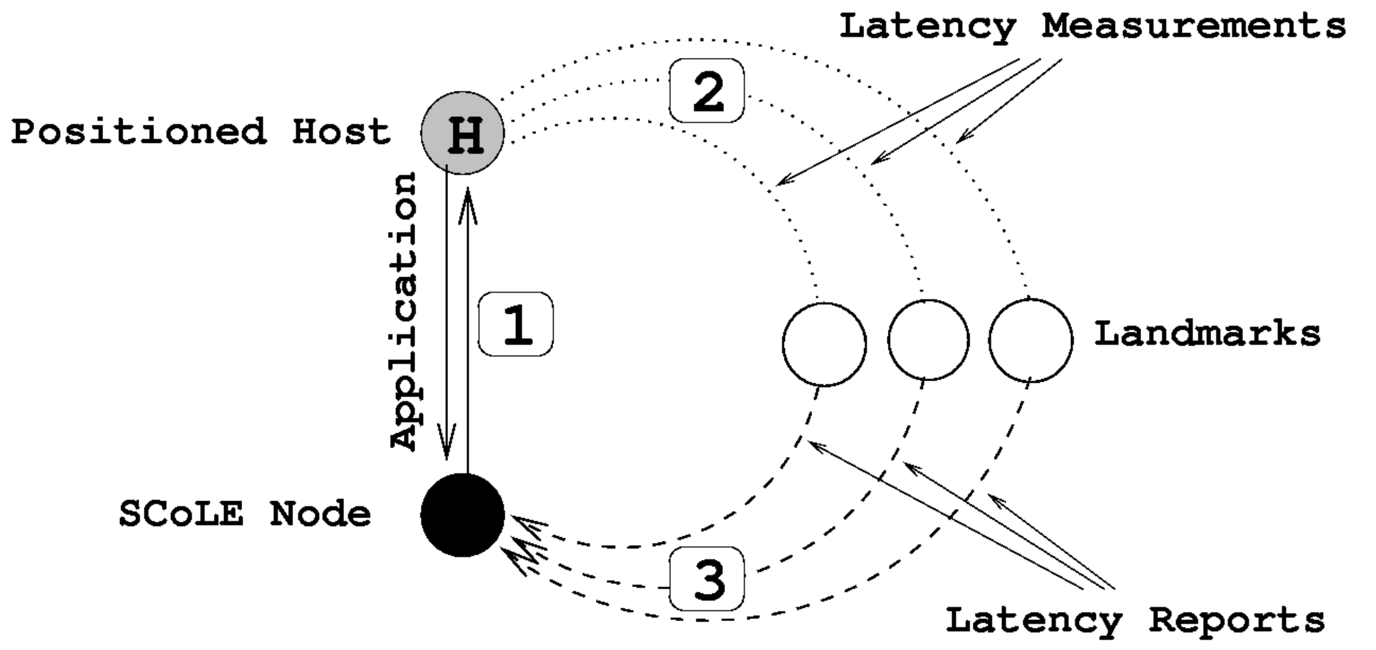
SCoLE: Personalized GNP

- SCoLE – Every node runs its own GNP
  - select your landmarks, position any nodes you want
- Properties:
  - no global negotiation nor knowledge
  - estimation adjustable on a per-node basis
  - positions calculated by different nodes may be different
  - but: latency estimates globally correlated



SCoLE: Architecture

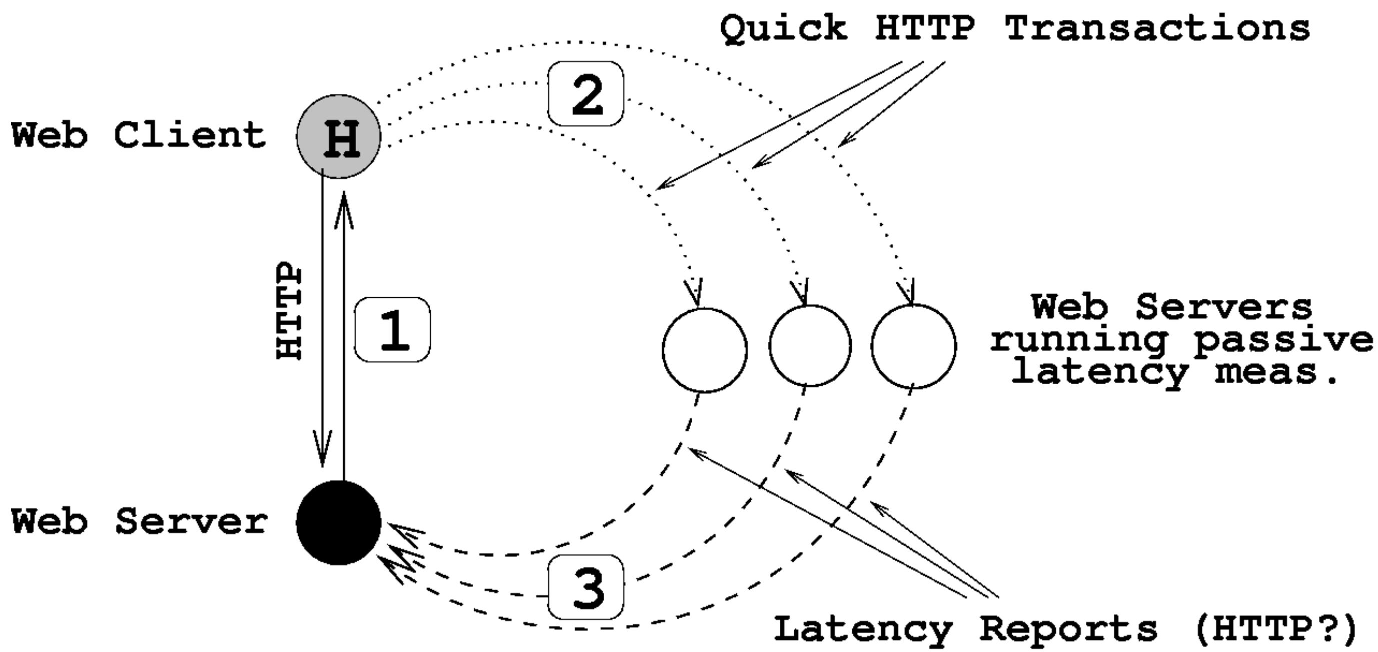
- SCoLE Instance:



- Watch out:
  - landmarks must be distributed (important for estimation accuracy)
  - landmarks measure latencies to each other (space construction)

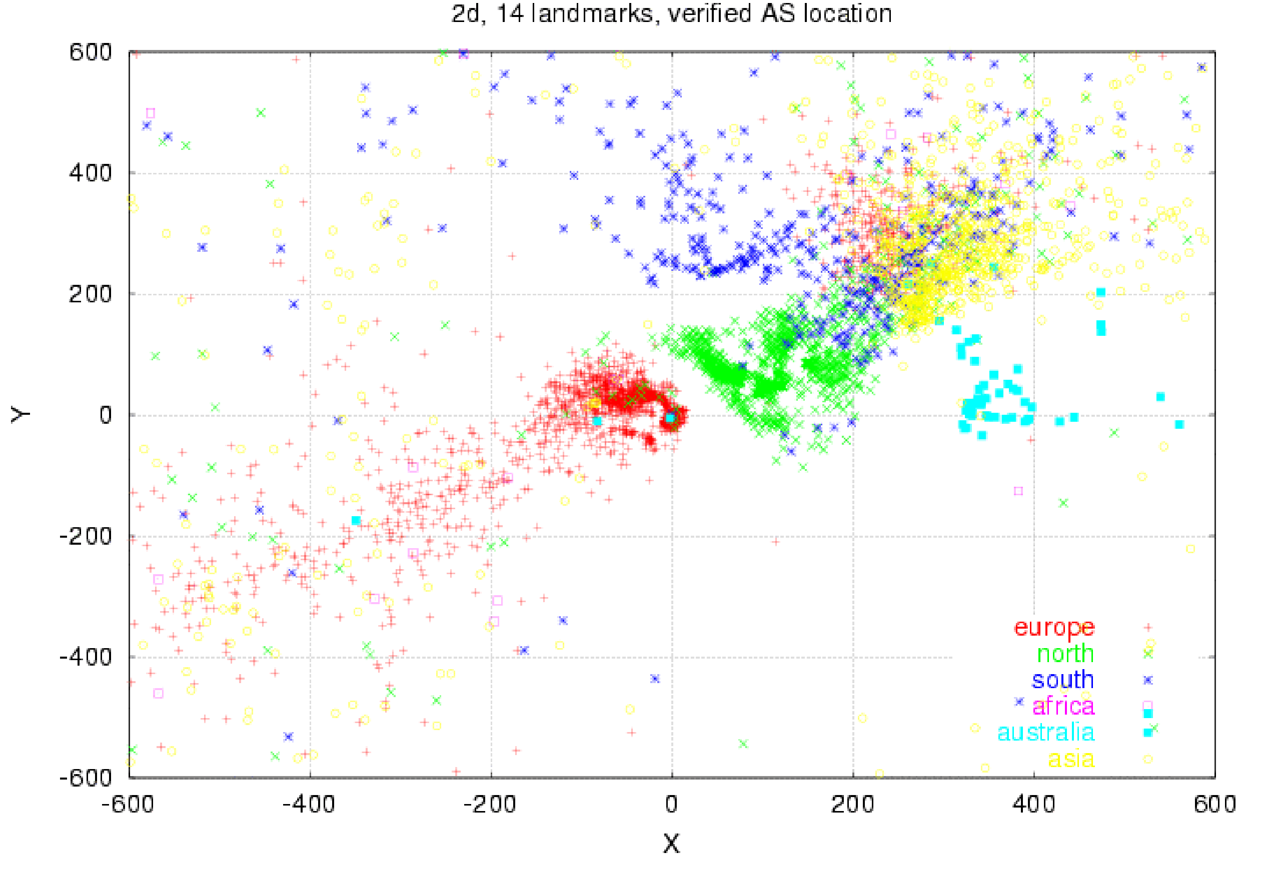
SCoLE: Deployment

- Example system:
  - CDN supporting latency-based redirection of Web clients



SCoLE: Prototype

- Deployed on the VU Website / PlanetLab nodes
- Clients positioned in 2D space:



Conclusion

- Network positioning:
  - Allows for scalable latency estimation
  - Is cheap in terms of number of measurements
  - Offers reasonable accuracy

- Can be personalized:
  - Each node runs its own GNP instance
  - Each instance can be adjusted to the owner's needs
  - Latency estimates are correlated across instances

Questions?