Parametric Probabilistic Sensor Network Routing

C.Barrett, S.Eidenbenz, L.Kroc, M.Marathe, J.Smith

NNSA

Los Alamos

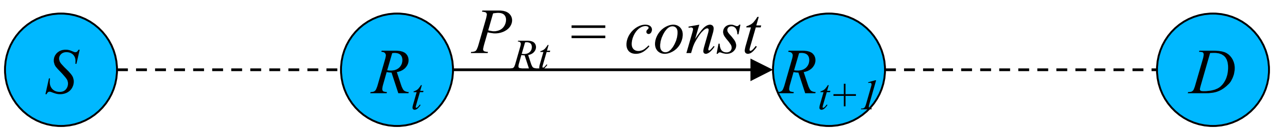
Introduction

- Sensor network specifics
  - Frequent topology changes
  - Short infrequent data sends
  - Limited energy supply

- Routing protocol design goals
  - Reliability, Robustness to misinformation, Energy efficient

Gossip Based Approach

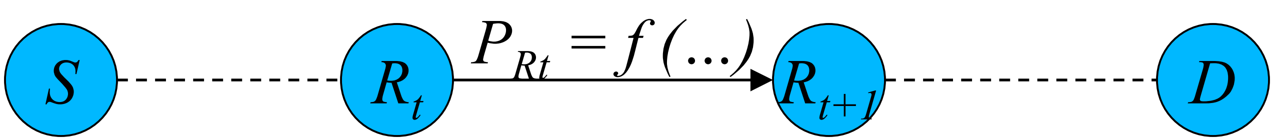
- Each node forwards received packet with fixed probability to ALL of its neighbors
  - multiple copies on the network at the same time
- Consecutive duplicates are dropped



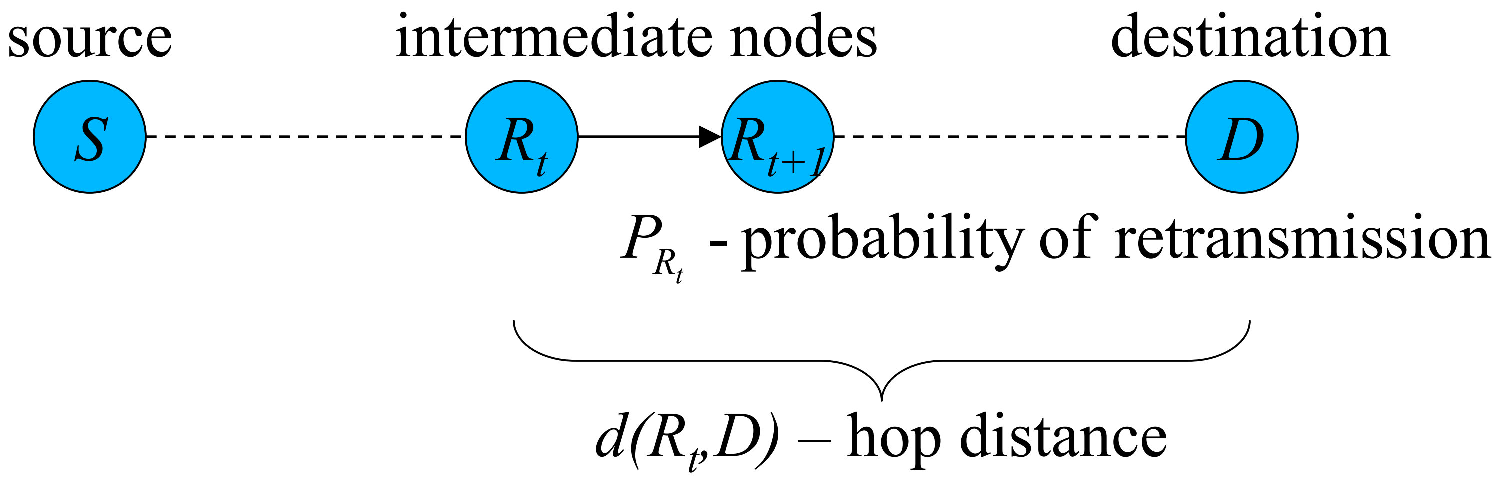
Parametric Probabilistic Sensor Network Routing Protocols

Adaptive gossiping, uses information to decide whether to forward or not

- information used may be of various kind
  - Distances, packet age, time, node degree
- reduces load, increases reliability

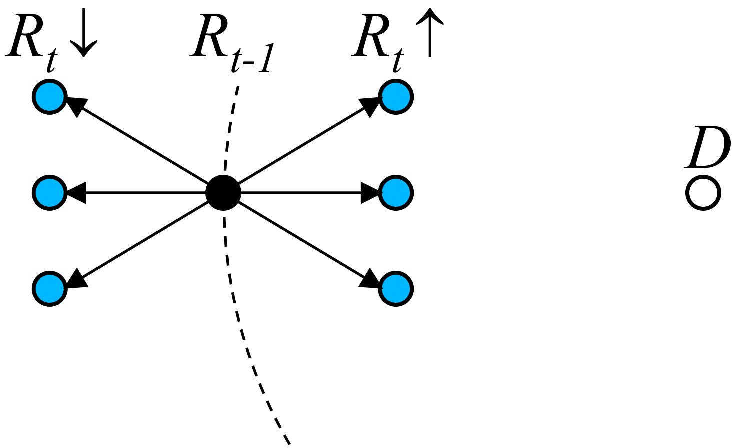


Used Notation



Destination Attractor

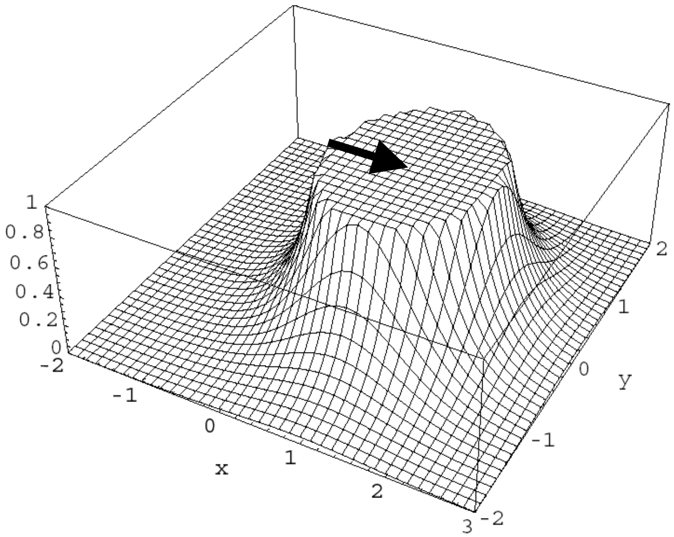
$$P_{R_t} = \begin{cases} (1+k)P_{R_{t-1}} & \text{closer to } D \\ (1-k)P_{R_{t-1}} & \text{further from } D \\ P_{R_{t-1}} & \text{otherwise} \end{cases}$$



Destination Attractor

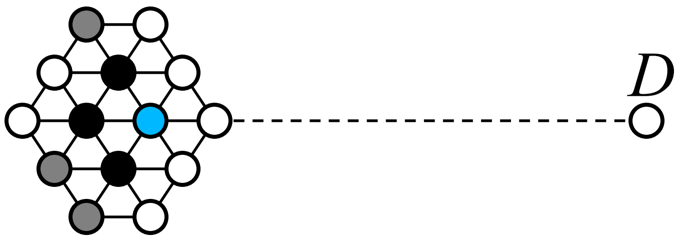
$$P_{R_t} = \{1 + k[d(R_{t-1}, D) - d(R_t, D)]\} P_{R_t}$$

$$\ldots \approx \boxed{\exp\{k[d(S, D) - d(R_t, D)]\}}$$



Directed Transmission

- Nodes almost as close to *D* as the best node forward packets with higher probability

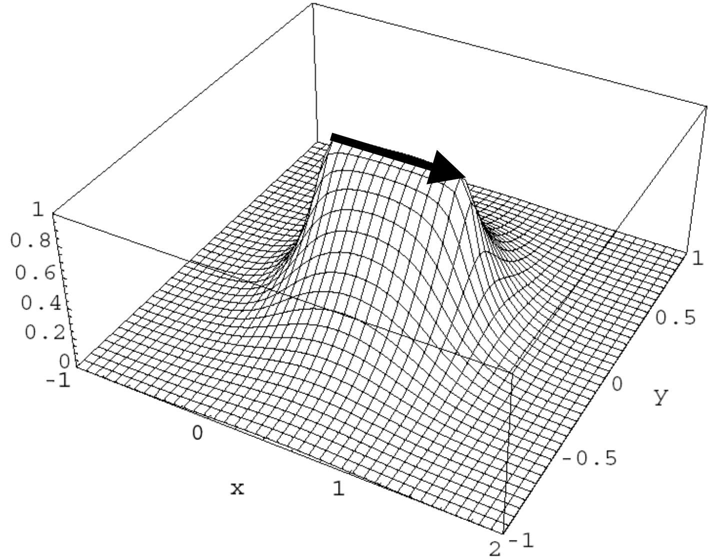


$$P_{R_t} = \exp\left\{ k \left[\min_{R'_t} d(R'_t, D) - d(R_t, D) \right] \right\}$$

Directed Transmission

$$\min_{R'_t} d(R'_t, D) \approx d(S, D) - t$$

$$P_{R_t} = \boxed{\exp\{k[d(S,D) - d(R_t, D) - t]\}}$$



Methods Used in Comparison

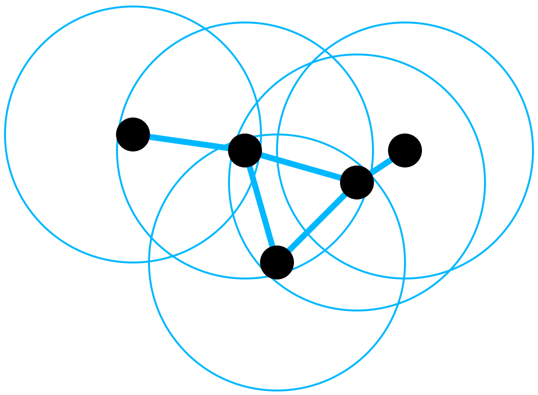
- Single packet copy
  - Wanderer
  - Shortest Path
  - Short Path
- Multiple packet copies
  - Gossiping
  - Flooding
  - Destination Attractor
  - Directed Transmission

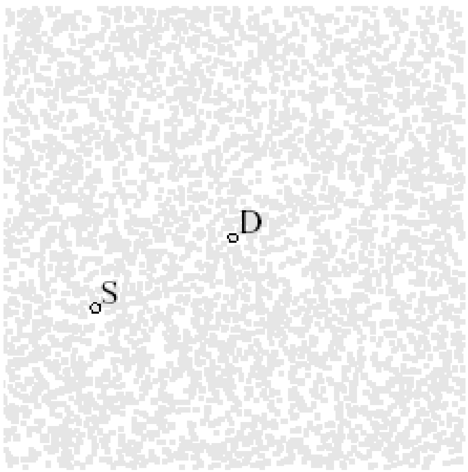
Estimating Global Information: Noise Model Used

- Precise information *i* computed off-line
- Noised value chosen uniformly at random from interval $[(1-q)i, (1+q)i]$
- Noise level *q* is a parameter
  - values used: 0%, 3%, 10%, 30%, 100%, 300%

Experimental Setup

- Unit disc graph of 5000 nodes in a square field, average degree ~ 6.7
- Single *D* in the middle of the field
- *S* chosen randomly for each run
- Only one data packet sent at a time with no competing traffic
- Only Routing layer considered
- 1000 runs made for each parameter/noise/method

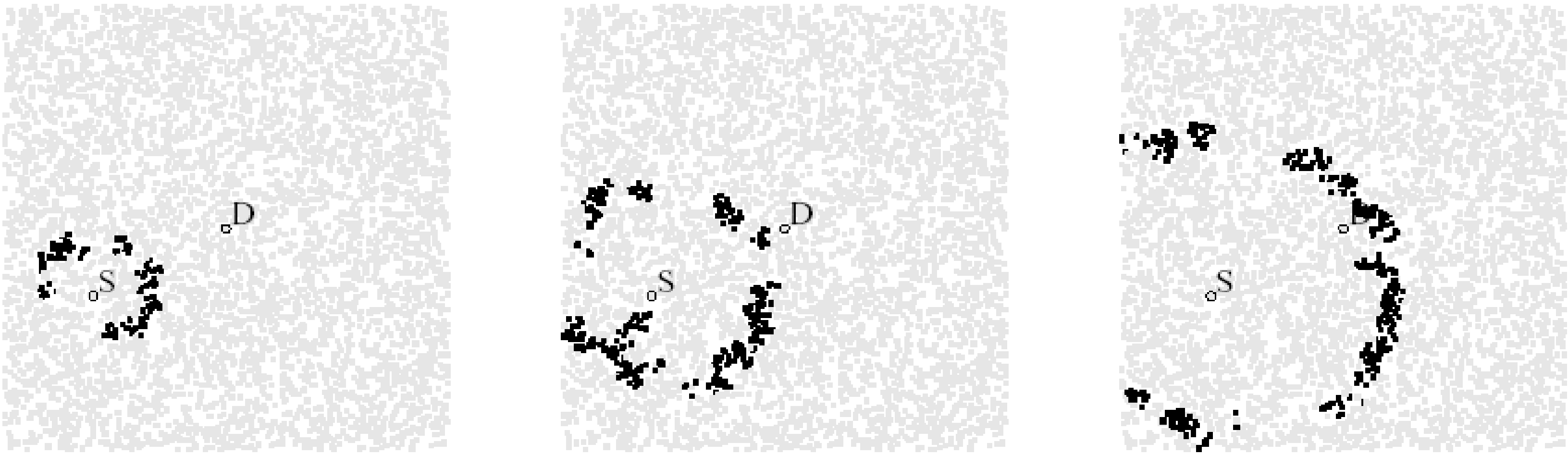




Measures of Performance

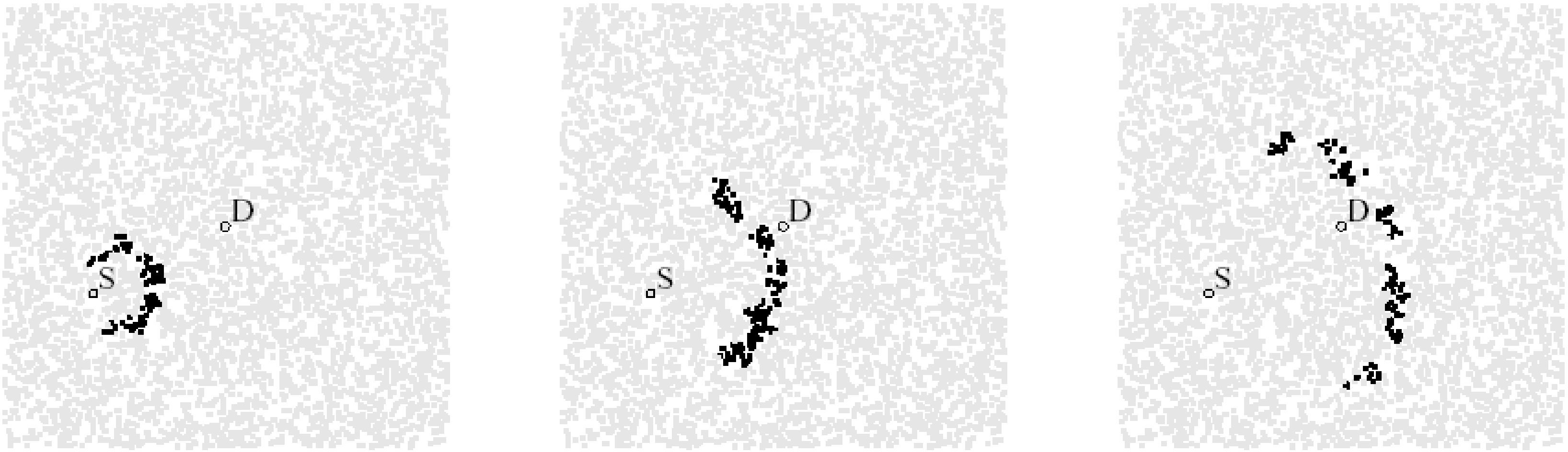
- **Load**
  - Number of times any sensor transmits a packet (energy consumed)
- **Fraction Delivered**
  - Fraction of runs where *D* receives the data
- **Lag**
  - Number of time steps to deliver at least one copy of the packet to *D*
- **Sensitivity to Noise**
  - Effect of misinformation to the above measures

Sample Run: Gossiping



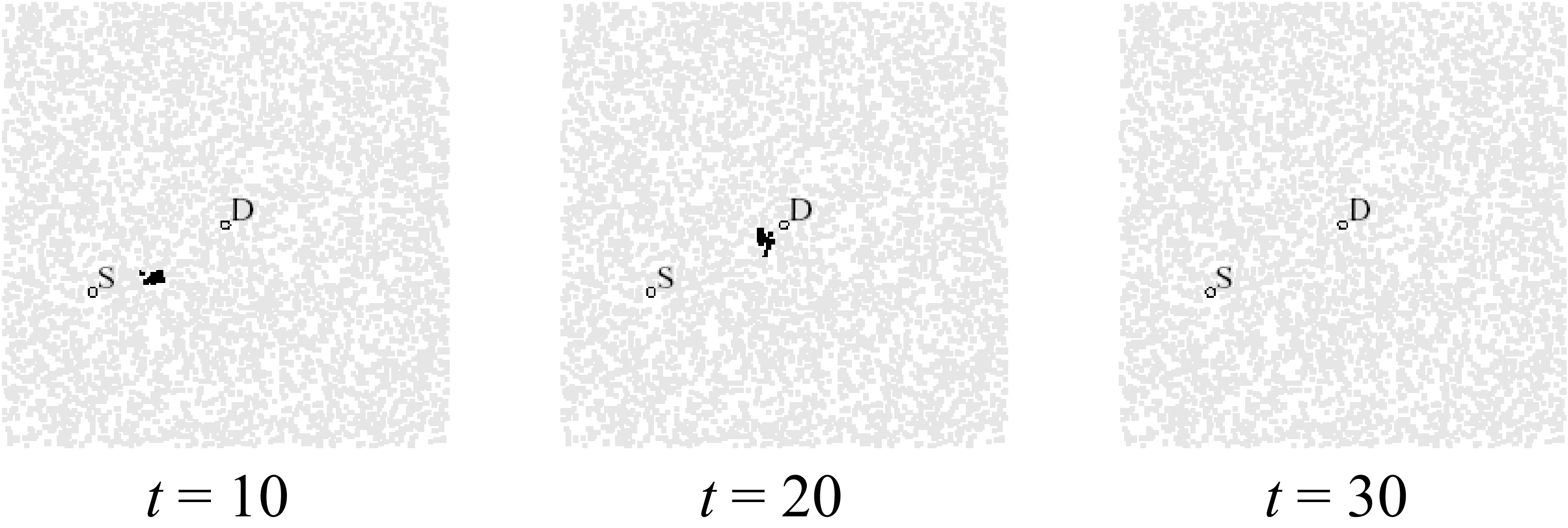
$t = 10$ $t = 20$ $t = 30$

Sample Run: Destination Attractor

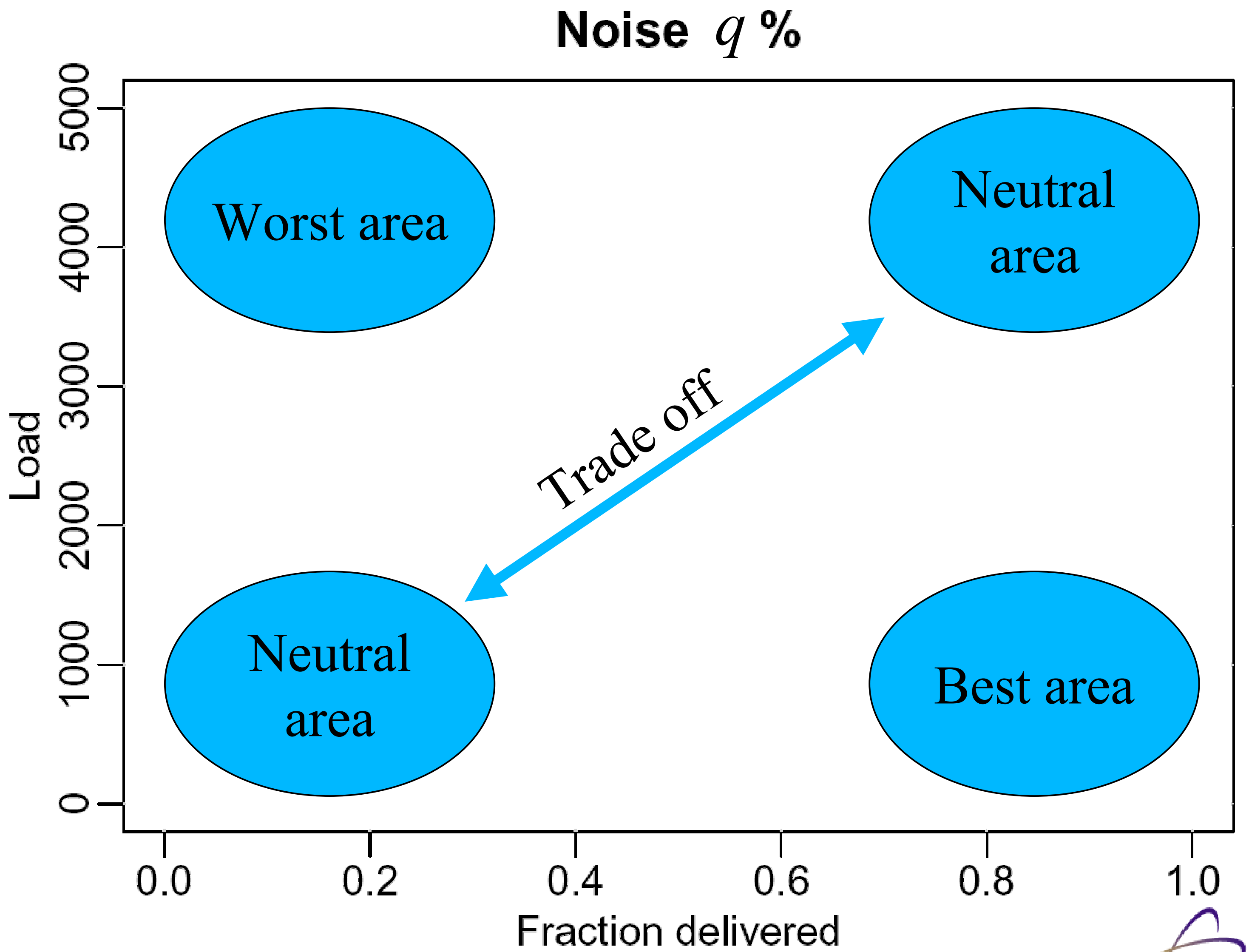


$t = 10$ $t = 20$ $t = 30$

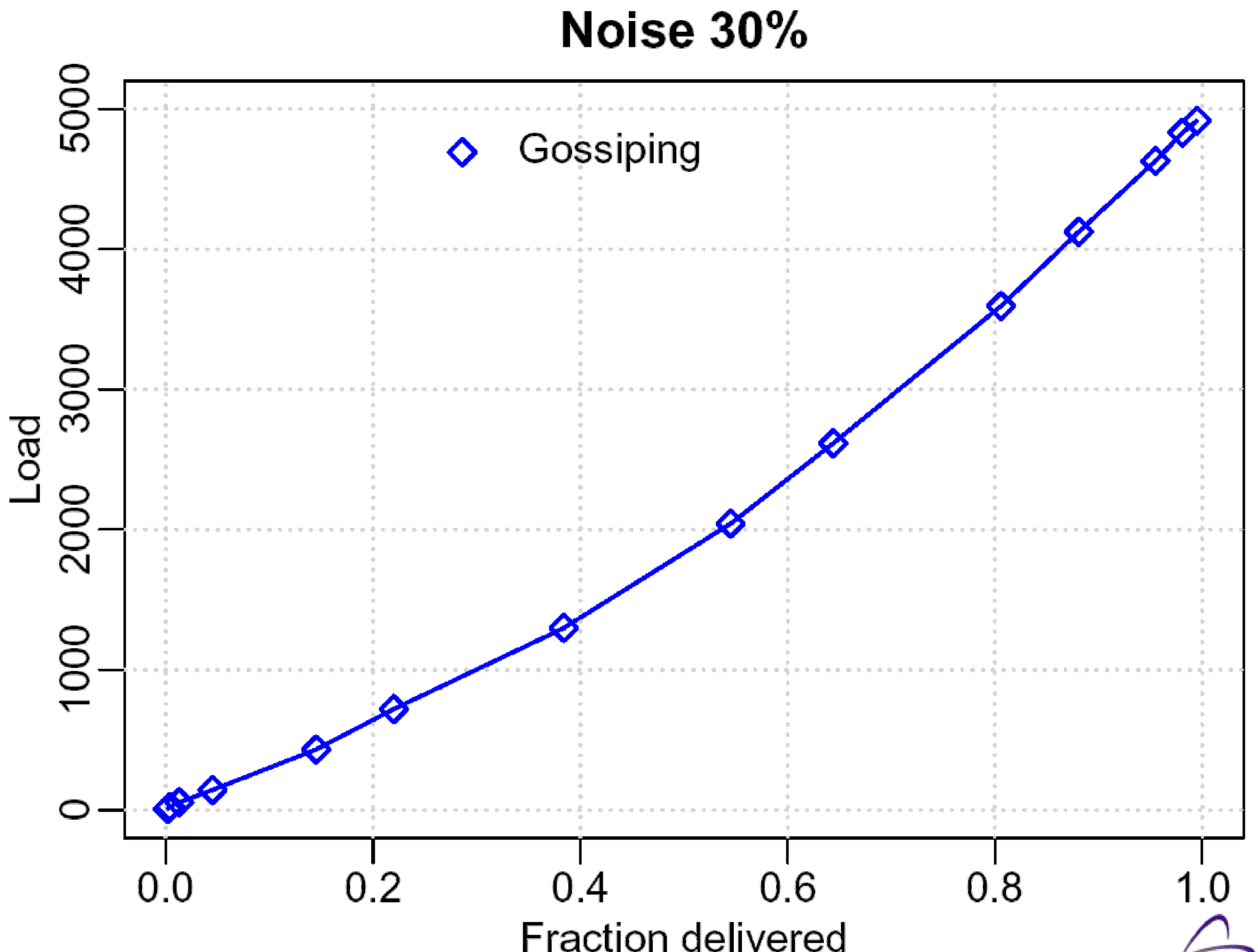
Sample Run: Directed Transmission



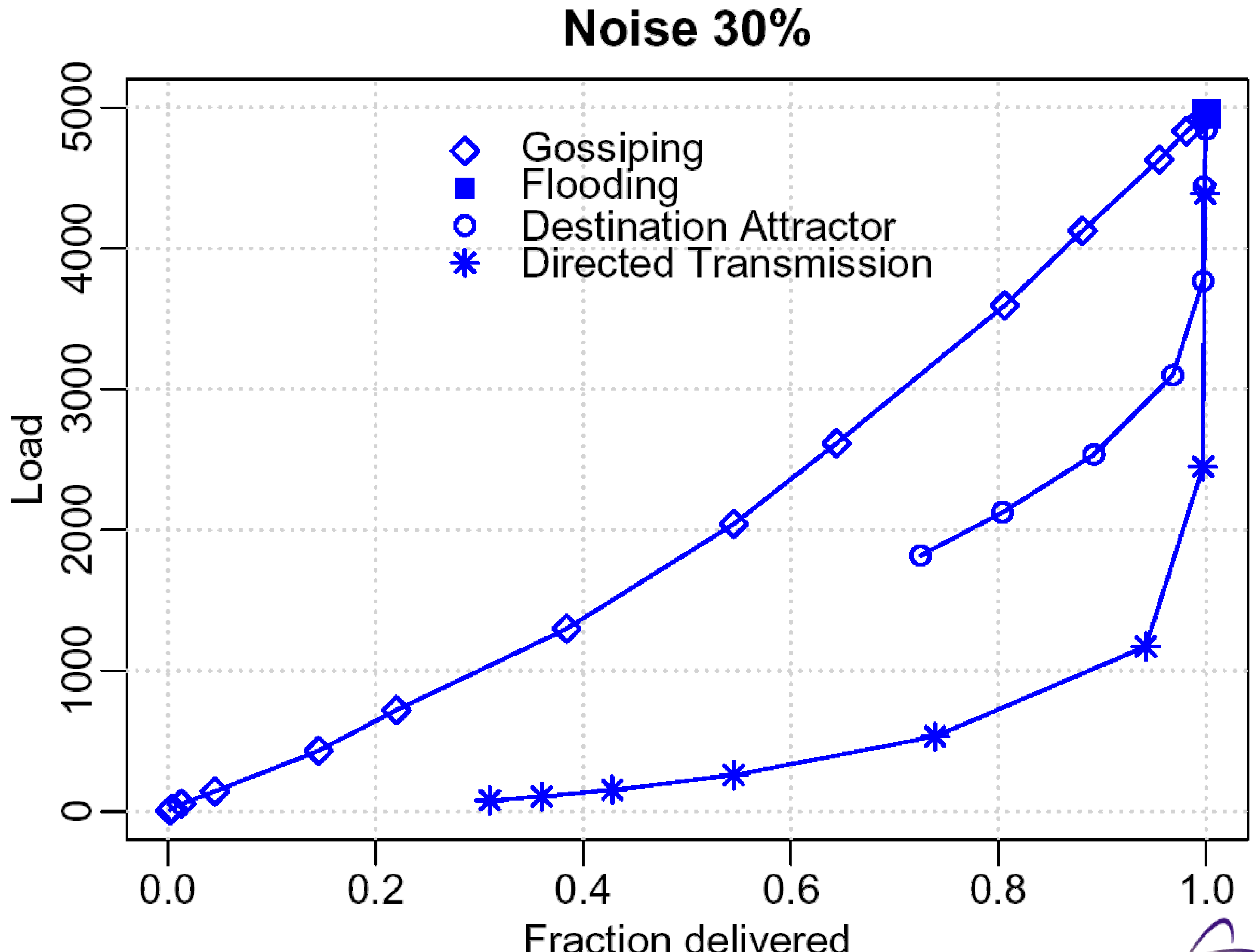
Comparison Results



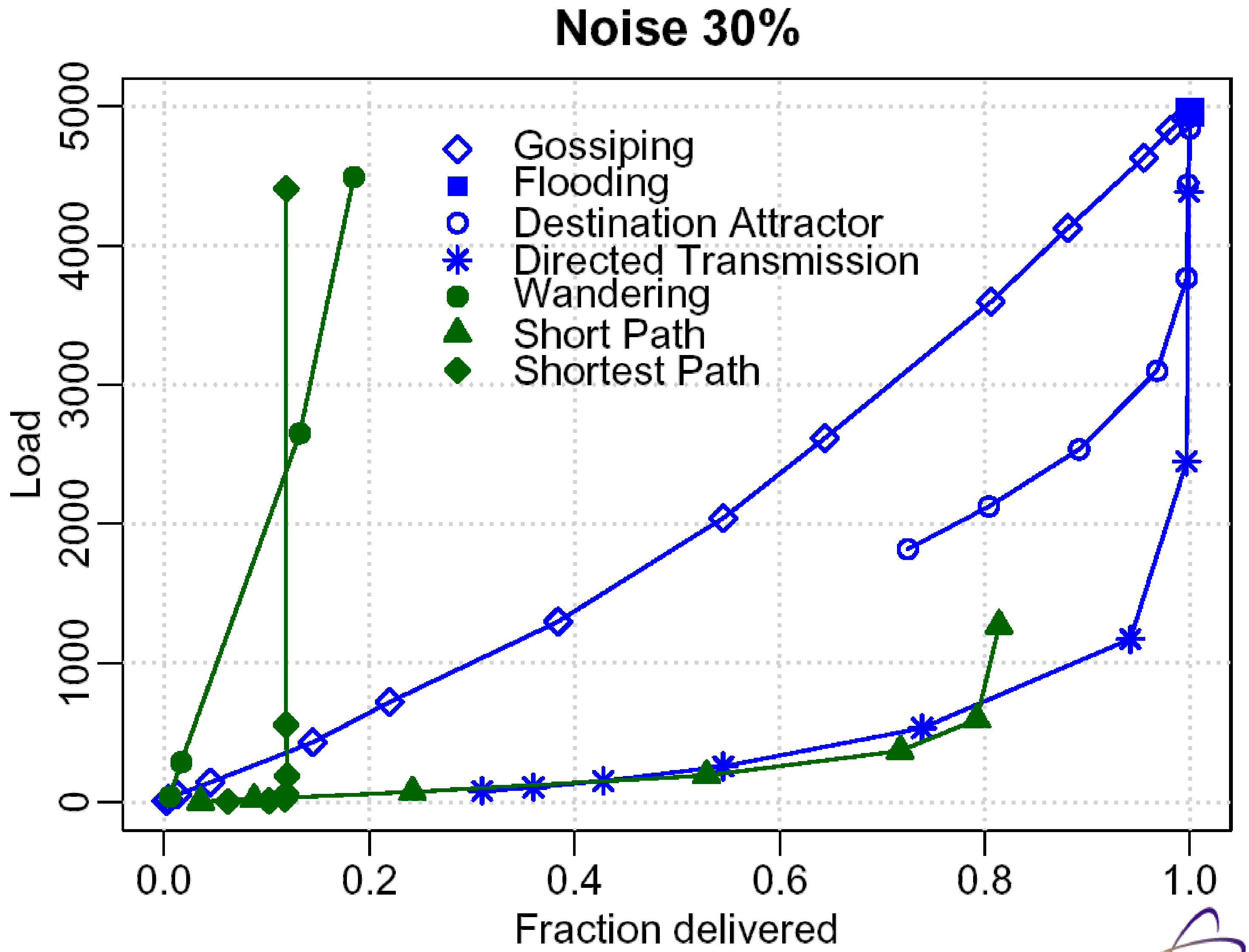
Comparison Results: Load



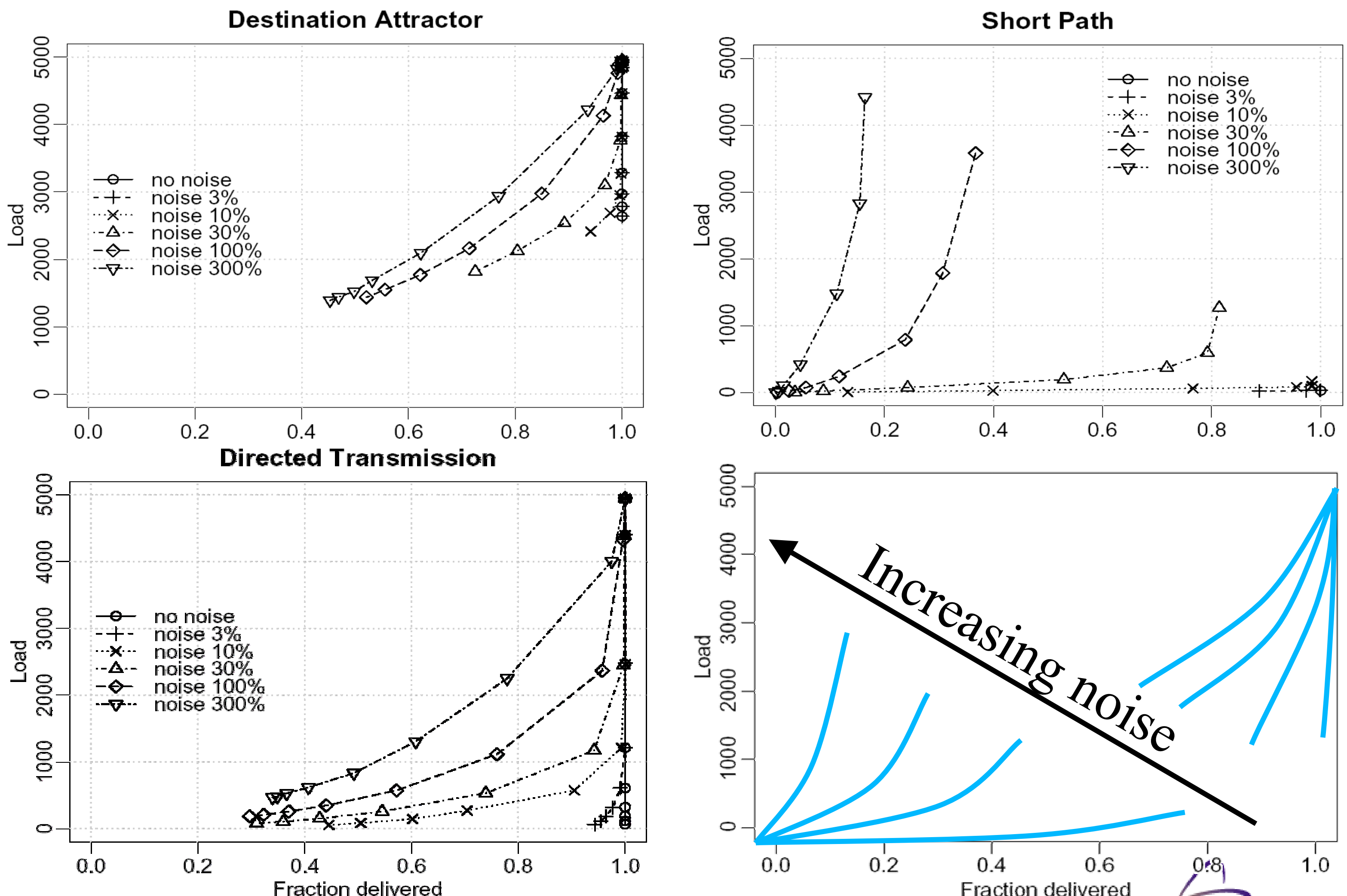
Comparison Results: Load



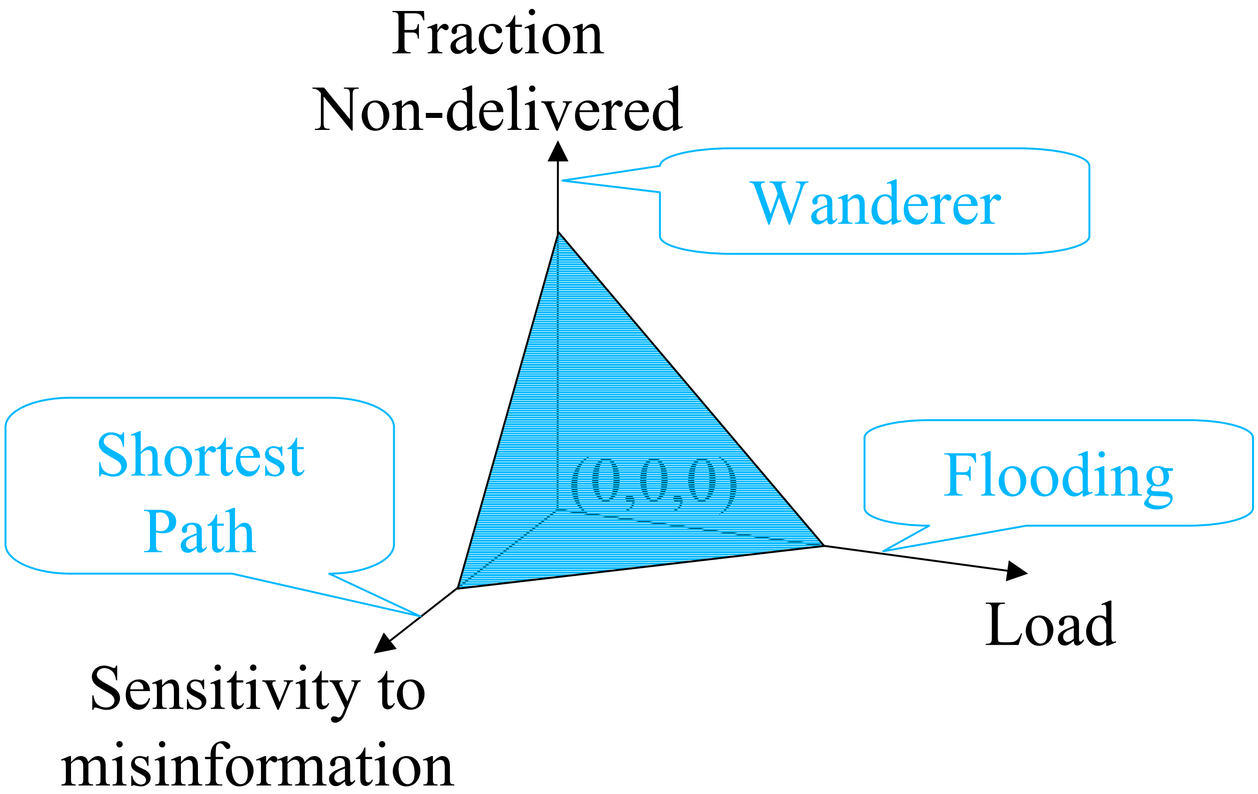
Comparison Results: Load



Comparison Results: Sensitivity to Noise



Trade-offs in Design Goals



Future Research Directions

- Different network topologies
  - Easier to analyze: grids
  - More realistic: realistic urban setting

- Other retransmission probability functions
  - Inverse polynomial
- More realistic simulation setup
  - Information propagation model
  - Inclusion of other protocol layers
  - Comparison to advanced protocols

Outline

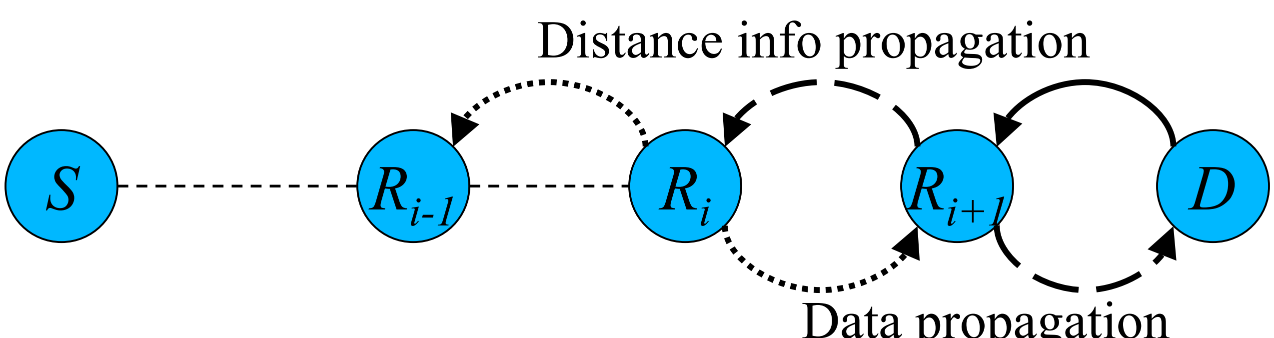
- Introduction
- Parametric Probabilistic Sensor Network Protocols family
- Experimental comparison
- Trade-offs in design goals
- Future research directions

Estimating Global Information

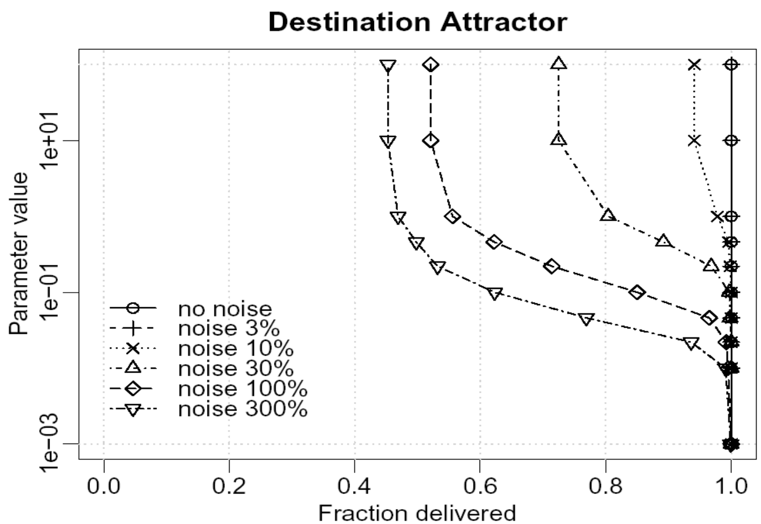
- Global information used: $d(S,D)$, $d(S,R_i)$, $d(S,R_{i-1})$, $d(R_i,D)$, $d(R_{i-1},D)$
- Proposed way of calculating it: use omni-directional transmission to back-propagate
- In simulations: precise information computed off-line, random perturbations added

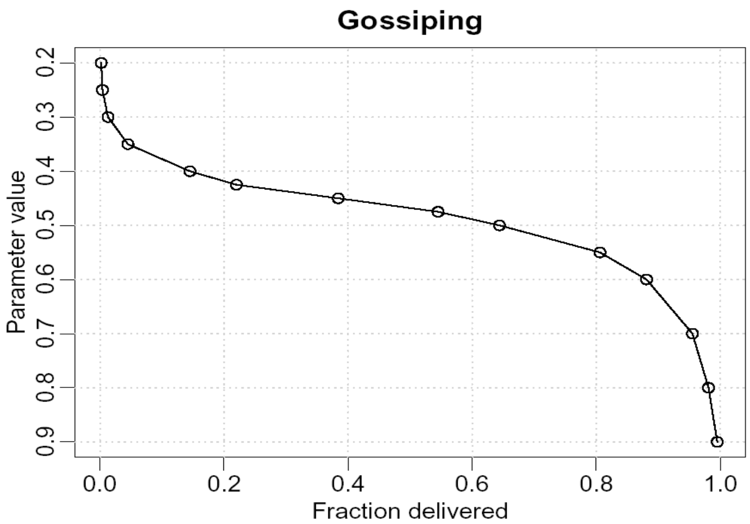
Estimating Global Information: Back-propagation

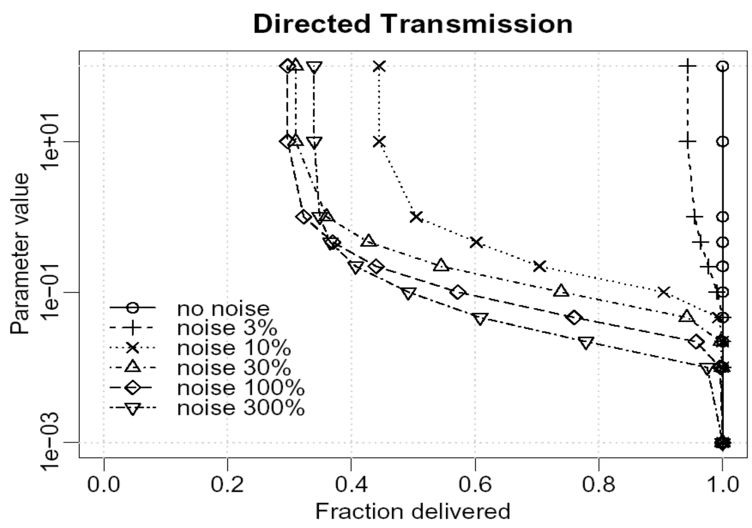
- *D* notifies its neighbors by sending ACK packet
- Node distance update: $d(R_i,D)$=min $d(R_j,D)$, R_j is neighbor of R_i, only information not older than *T* is used
- *R* at distance *d* get to know it in ~ *d* time steps

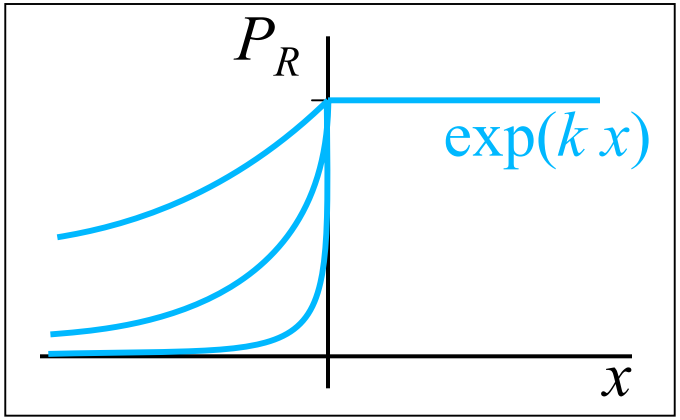


Parameter Dependence



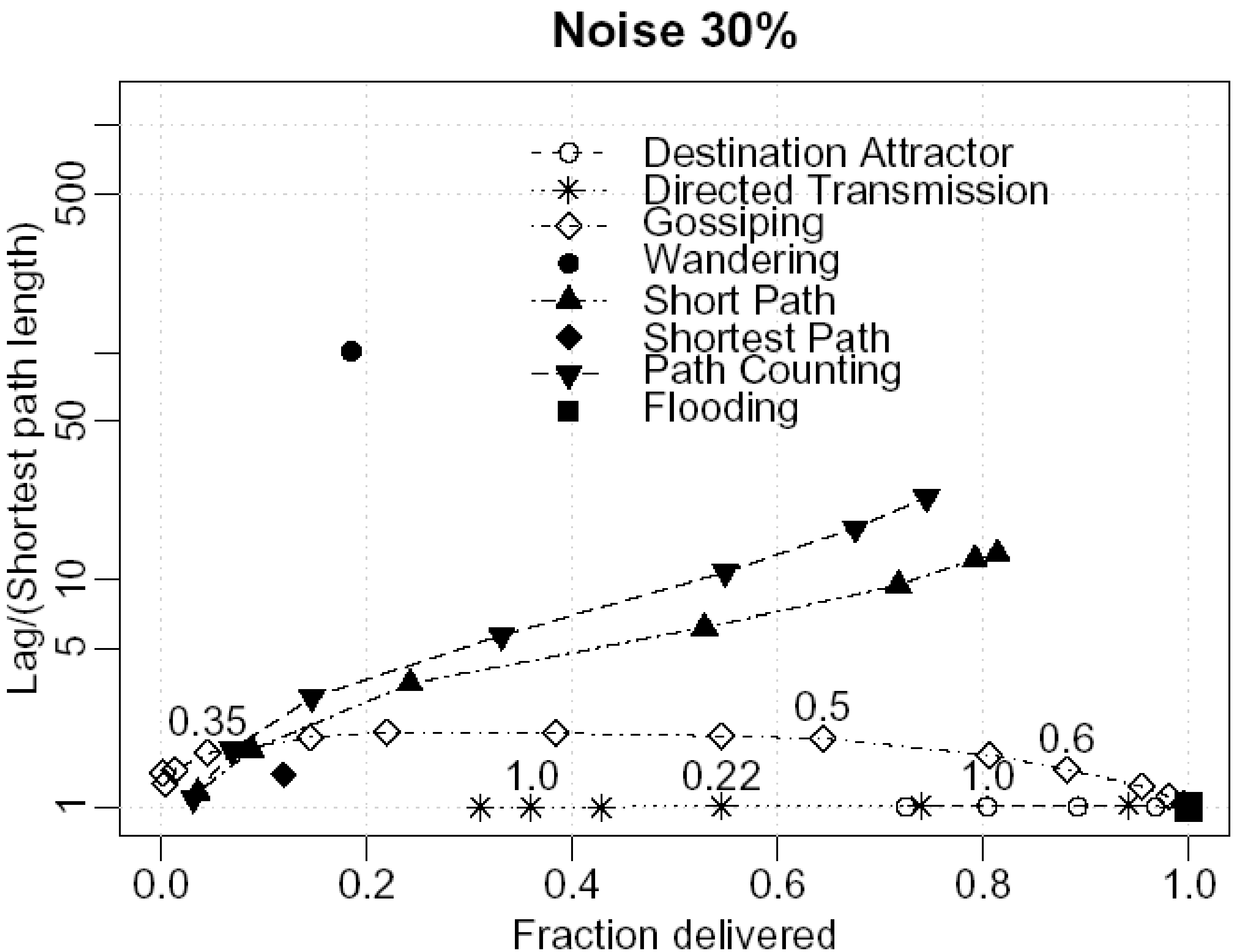






NNSA Los Alamos

Comparison Results: Lag



Parameter Dependence