Course “Algorithmic Foundations of Sensor Networks”
Lecture 5: Energy-efficient and robust routing (multipath)

Sotiris Nikoletseas
Professor

Department of Computer Engineering and Informatics
University of Patras, Greece

Spring Semester 2020-2021

Overview

Multipath routing

A. Braided paths, GRAd, GRAB

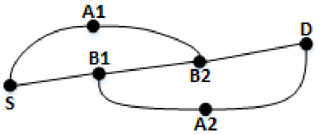
B. Probabilistic forwarding (PFR)

A. Multipath Routing (1/2)

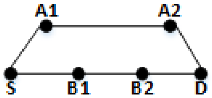
- multiple routes are used, towards increasing robustness
- the routes can be *disjoint or partially disjoint*
- *"Braided multipath routing"*:
  - There is *a primary path* that is used for routing.
  - Several *alternate paths* are maintained for use in case of a failure in the primary path.

A. Multipath Routing (2/2)

- *"Braided"* path: node disjointedness between alternate paths is not a strict requirement. Rather, for each node on the primary path, the method requires the existence of an alternate path from the source to the sink that *does not contain that node*, but which may otherwise overlap with the other nodes on the main path.
- Compared to disjoint paths, the braided path is more suitable for *isolated* failures, while disjoint paths are more appropriate for *pattern (geographically correlated) failures*.



primary path: S B1 B2 D
alternate paths: S A1 B2 D, S B1 A2 D



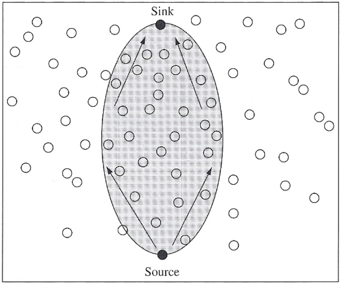
disjoint path

Gradient Cost Routing (GRAd)

- All nodes in the network maintain an estimated *cost to the sink* (such as the number of hops needed to reach it).
- When a packet is transmitted, it includes a field with the *cost "paid"* already in the current data propagation. Also, a *remaining value* field is maintained, acting as a TTL (time-to-live) field for that packet.
- Any receiver that receives this packet forwards the packet iff its own cost to the sink is *smaller* than the remaining value of the packet. Before forwarding, the "cost paid" field is increased by one and the remaining value field is decreased by one.
- GRAd actualy allows *multiple nodes to forward the same message*, so it essentially performs *a limited directed flooding* towards the sink and provides significant robustness but at the cost of a larger overhead.

Gradient Broadcast Routing (GRAB)

- It enhances GRAd by *incorporating a tunable energy-robustness trade-off through the use of credits*.
- Similarly to GRAd, GRAB maintains a cost field at each node. Additionally, the packets travel from a source to the sink with *a credit value* that is decreased at each step depending on the hop cost.
- *Credit-sharing mechanism*: *Earlier hops receive a larger share of the total credit in a packet*, while the later nodes receive a smaller share. Intermediate nodes with greater credit can spend a larger budget sending the packet to a larger set of forwarding neighbors. This way, paths *spread out initially* while eventually the *diverse paths converge to the sink* efficiently.



The GRAB Algorithm - Details (I)

- Each packet contains 3 fields:
  1. R_o: the credit assigned at the source node
  2. C_o: the cost-to-sink at the source node
  3. U: the budget already consumed from the source to the current hop

  (note: R_o, C_o never change in the packet, while U is increased in each hop).
- Each receiver i (with a cost-to-sink C_i) computes the following metric:

  $$\beta = 1 - \frac{R_{oi}}{R_o}$$

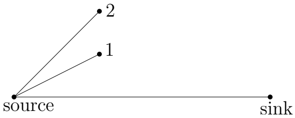
  and a threshold θ:

  $$\theta = \left(\frac{C_i}{C_o}\right)^2$$

  where $R_{oi} = U - (C_O - C_i)$
- *Note*: $C_o - C_i$ actually represents the optimal cost for travelling between source and node i, and R_{oi} determines how much credit has been already used. Nodes far away from the optimal direct source-sink line will have a higher U and thus a higher R_{oi} credit spent. Thus, β actually estimates how much credit is left for the packet.

The GRAB Algorithm - Details (II)

- The packet is forwarded iff $\beta > \theta$, i.e. when the credit left is sufficient to cover the remaining cost (captured by θ). The square exponent introduces a heavier weighting to θ, in the sense that the threshold is more likely to be exceeded in the early hops than in the later hops, as desired (for limiting the path spreading). An example:



It is $U_2 > U_1$ and $C_2 > C_1 \Rightarrow R_{o2} > R_{o1} \Rightarrow \beta_2 < \beta_1$ and $\theta_2 > \theta_1$

$\Rightarrow$ the node 1 is more likely to forward than 2.

- *The choice of the initial credit R_o actually provides a tunable parameter to increase robustness at the expense of energy consumption*, since broader paths are created (since $R_o \uparrow \Rightarrow \beta \uparrow$).

B. Probabilistic Multipath Forwarding (PFR)

We assume that each node (particle) has the following abilities:

i) It can estimate *the direction of a received transmission* (e.g. via the technology of direction-sensing antennae).

ii) It can estimate *the distance* from a nearby particle that did the transmission (e.g. via estimation of the attenuation of the received signal).

iii) It knows the *direction towards the sink S*. This can be implemented during a set-up phase, where the sink broadcasts information about itself to all particles.

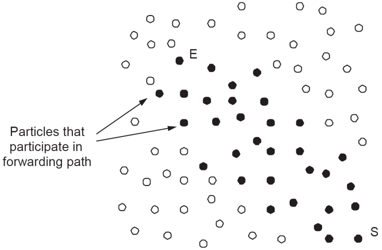
iv) All particles have a *common co-ordinates system*.

Note that *GPS information is not needed* for this protocol. Also, there is no need to know the global structure of the network.

Propagation Protocol Properties

- ***Correctness***. Protocol Π must guarantee that *data arrives to the sink* S, given that the network is operational.
- ***Robustness***. Π must guarantee that data arrives at enough points in a small interval around S, in cases when part of the network has become inoperative.
- ***Efficiency***. Π should have a *small ratio* of the number k of *activated* particles over the total number N of particles $r = \frac{k}{N}$. Thus r is an energy efficiency measure of Π.

PFR *probabilistically favors* redundant transmissions toward the sink within a *thin zone* of particles around the line connecting the particle sensing the event $\mathcal{E}$ and the sink.



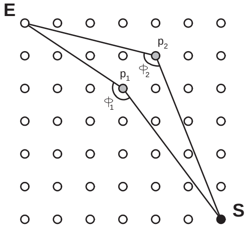
Thin Zone of particles

The Forwarding Probability

Data is propagated with a suitably chosen *probability* p, while it is not propagated with probability $1 - p$.

To *favor near-optimal transmissions* the following probability is used:

$$P_{fwd} = \frac{\phi}{\pi}$$



The two phases of PFR

Phase 1: The "Front" Creation Phase. Initially (by using a limited, in terms of rounds, flooding) a *sufficiently large "front" of particles* is built, to guarantee the survivability of the data propagation process. Each particle having received the data, *deterministically broadcasts* it toward the sink.

Phase 2: The Probabilistic Forwarding Phase.
Each particle *p* receiving info(ϵ), broadcasts it to all its neighbors with *probability* $\mathbf{P}_{fwd}$ (or it does not propagate any data with probability $1 - \mathbf{P}_{fwd}$) defined as follows:

$$\mathbf{P}_{fwd} = \begin{cases} 1 & \text{if } \phi \geq \phi_{threshold} = 134^o \\ \frac{\phi}{\pi} & \text{otherwise} \end{cases}$$

The Correctness of PFR

Lemma

PFR *always succeeds* in sending the information from E to S when the whole network is operational.

In the proof, geometry is used (i.e. we *cover* the network area by unit squares and show that there are always particles *"close enough" to the optimal line*, i.e. with $\phi > 134^o$, that deterministically broadcast).

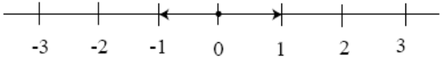
Stochastic Processes: basic definitions(1/2)

- Stochastic process: a random process evolving in time.
- Random walk on the line:

$$X_o = 0$$

$$X_i = \begin{cases} +1, \text{ with probability } \dfrac{1}{2} \\ -1, \text{ with probability } \dfrac{1}{2} \end{cases}$$

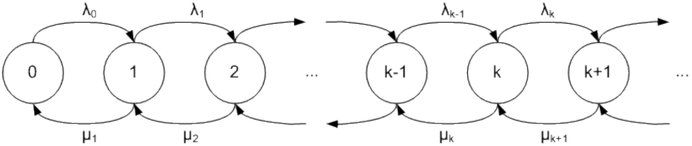
$$S_n = \sum_{i=1}^{n} X_i \text{ is the position of the random walk after time n.}$$



a. Random walk in one dimension.

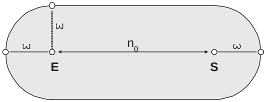
Stochastic Processes: basic definitions(2/2)

- Markov process: a stochastic process in which the future is independent of the past.
- Birth-death process: a Markov process with only two types of transitions: a birth (+1) or death (-1).



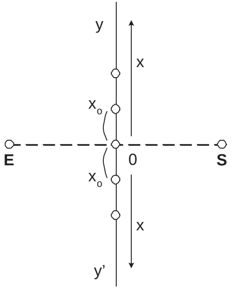
The Energy Efficiency of PFR

- We consider particles that are active but *as far as possible* from ES



The particles inside the L_Q Area

- We *approximate* w by the following random walk



By using stochastic dominance* by a continuous time "discouraged arrivals**" birth-death process, we prove:

Theorem

The energy efficiency of the PFR protocol is $\Theta\left(\left(\frac{n_0}{n}\right)^2\right)$ where $n_0 = |ES|$ and the total number of particles in the network is $N = n^2$. For $n_0 = o(n)$, this is $o(1)$.

* A random variable X stochastically dominates a random variable Y when for all values a: $P_r\{X \geq a\} \geq P_r\{Y \geq a\}$
example: X: the sum of two random dice
Y: the absolute value of how much the dice differ

** Discouraged arrivals: births have lower rate as the size of the population increases.

The Robustness of PFR

Particles *very near the ES line* are considered.

We study the case when *some* of these particles (at angles $> 134^o$) are *not operating*.

The probability that none of them transmits is *very small*.

It is shown:

Lemma

PFR manages to propagate the crucial data across lines parallel to ES, and of constant distance, with *fixed* nonzero probability.

Experimental Comparison of LTP and PFR

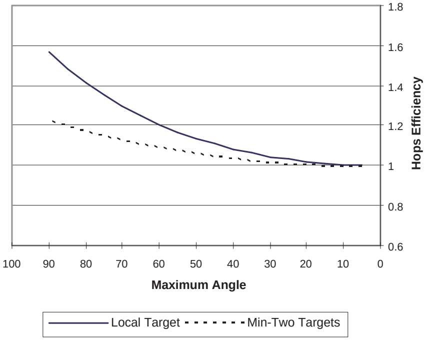
The simulation environment:

- C++
- 2D geometry data types of LEDA
- a variety of sensor fields in a $100\text{m} \times 100\text{m}$ square area:
  - we drop randomly $n \in [100, 3000]$ particles (i.e. $0.01 \leq d \leq 0.3$)
  - fixed radio range $R = 5$m
  - search angle $\alpha = 90^\circ$
- we repeated each experiment more than 5000 times to get good average results

LTP – The impact of angle α

- $\alpha \to 0 \Rightarrow C_h \to 1$ (assuming particles always exist)
- C_h initially decreases very fast, while having a limiting behavior for $\alpha \leq 40$

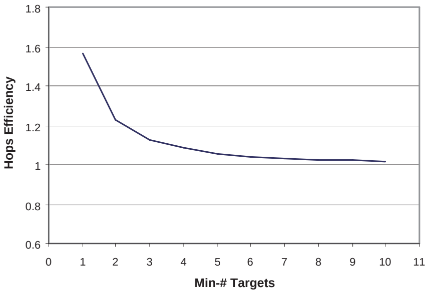
Ideal Hops Efficiency for angles $\alpha \in [5, 90]$



LTP – The impact of sampling several targets

- $\#$ targets $\uparrow \Rightarrow C_h \to 1$
- 4 targets $\Rightarrow$ already very close to optimal

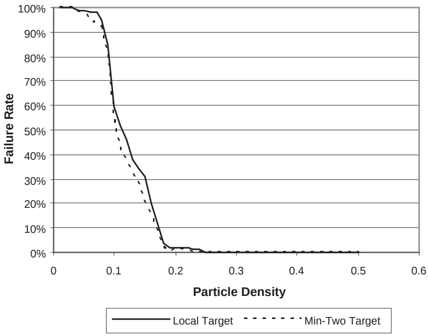
Ideal Hops Efficiency for different number of targets



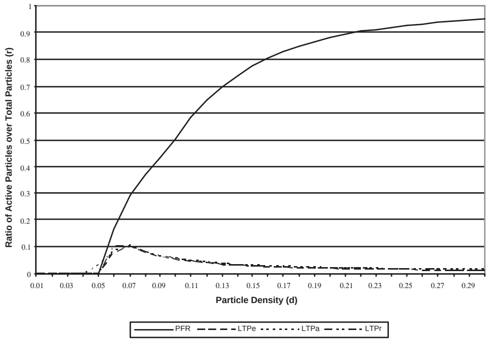
LTP – Failure rate

- for $d \leq 0.1$ both protocols almost always backtrack
- for $d \geq 0.2$ the failure rate drops very fast to 0.

Failure rate for density $d \in [0.01, 0.5]$ and $\alpha = 90$

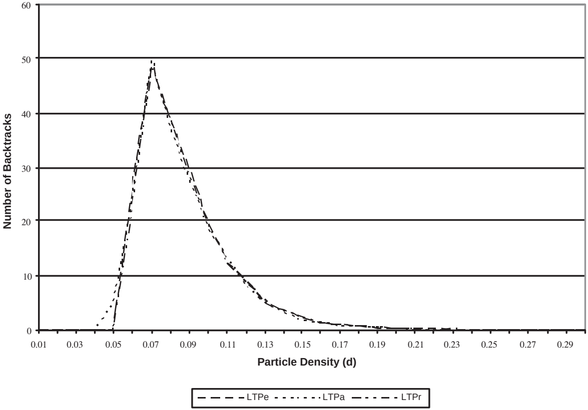


Ratio of activated particles over density



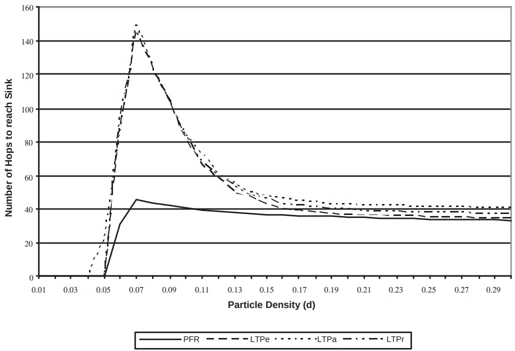
- PFR behaves very well ($r \leq 0.3$) for low densities ($d \leq 0.07$)
- PFR's energy dissipation increases with density
- LTP performs best in dense networks

Number of Backtracks



- For very low densities (i.e. $d \leq 0.12$), LTP backtracks a lot.
- As density increases, the number of backtracks of LTP reduces fast and almost reaches zero.

Average number of hops to the sink



- all protocols are near optimal (40 hops) even for low densities (≥ 0.17). PFR achieves this even for very low densities (≥ 0.07).
- LTP shows a pathological behavior for low densities (≤ 0.12) due to many backtracks.