UCLA

Posters

Title

The Acquire Mechanism for Efficient Querying in Sensor Networks

Permalink

<https://escholarship.org/uc/item/0fs0f24f>

Authors

Narayanan Sadagopan
Bhaskar Krishnamachari
Hel

Publication Date

2003

THE ACQUIRE MECHANISM FOR EFFICIENT QUERYING IN SENSOR NETWORKS

Narayanan Sadagopan, Bhaskar Krishnamachari, Ahmed Helmy
Autonomous Network Research Group (ANRG) - <http://ceng.usc.edu/~bkrishna/anr/>

Introduction:

One Shot Queries for Replicated Data

- Example Query: Is there a location in the network where (Temperature > 80) AND (Wind Speed < 20 mph) AND (Humidity < 70 %)
- Example Query: Fetch the calls of the following birds: Nightingale, Blue Jay and Warbler.

ACQUIRE: An Efficient Mechanism (SNPA'03)

- Two Phases
 - Update Phase: An active node requests for information from all nodes within d hops of itself (d is the **look-ahead parameter**) and tries to resolve the query
 - Forward Phase: If the query is not completely resolved, the node forwards it to some other node (chosen randomly or intelligently) d hops away.
- Spans the entire space from random walk ($d=0$) to flooding ($d=D$, the network diameter)

Problem Description:

Is there an optimal value for the look-ahead parameter d in ACQUIRE, and what does it depend on?

Proposed Solution:

We derive analytical expressions for the average energy usage of ACQUIRE and prove the following:

- The Optimal value of d depends only on the querying rate and the data dynamics (c) and is independent of the query Size (M) and the total number of variables tracked by the network (N).
- For a given query rate, lower the data dynamics, the higher the optimal value of d . So flooding is best suited when sensor data rarely changes, while random walks are better suited for highly dynamic scenarios.
- ACQUIRE can significantly outperform other alternatives such as expanding ring search.

ACQUIRE: Basic Mechanism

Optimal Look-ahead (d^*) vs. the data dynamics (c)

Average Energy to Answer a Query vs. d