



Inventory Tracking with Mobile RFID Readers



Michael Lin, Tim Bolbrock, Qinghua Li, Hosam Rowaihy, Guohong Cao and Thomas La Porta

A common use of passive RFID tags is warehouse inventory tracking. Generally, fixed readers are attached to entrances and tags are scanned as items enter and leave. In certain situations, e.g. military or disaster-relief scenarios, this infrastructure may be unavailable, infeasible, or undeployed, and a more flexible solution is required. We present a real-time inventory tracking system using passive RFID tags and mobile, wireless networked RFID readers. The goal is low latency in response to queries. Towards this goal, we considered algorithms for the coordination of reader movement and for data caching. We consider both the *centralized* case, where all readers are in range of a base station, as well as the *multihop* case where there is no base station and communications are ad hoc.

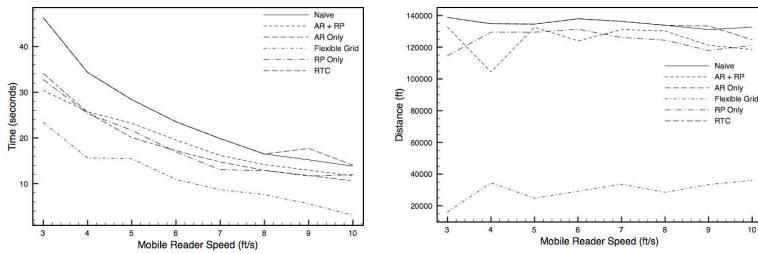
Query Process

1. A query is received: "How many shoes?"
2. -Centralized: all readers receive the query and all know each others' locations. The best reader is picked to retrieve the data.
-Multihop: only some readers receive the query and they do not know all other readers' locations. Best "known" reader begins moving towards data. If it comes within range of a closer reader, pass the query.
3. A reader reaches the destination and retrieves data: "27 shoes."
4. Data is returned. In multihop, the result may again be passed between readers to reach the query source.

Movement Algorithms

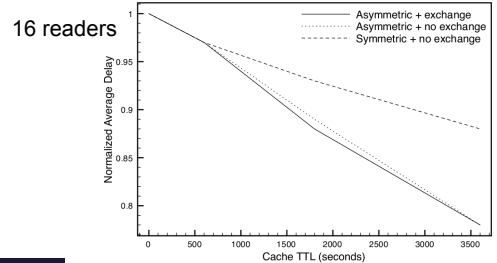
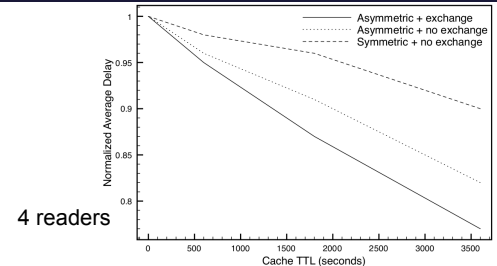
- Consider skewed query destination locations, i.e. a sale on hats leads to most queries being for the hats section.
 - One impetus for algorithms: keep a reader near this *hotspot*.
 - For multihop, algorithms that maintain even spacing help ensure uniform coverage of the area for connectivity and query passing.
1. **Naive**: Simply pick the reader closest to the data.
 2. **Area of Responsibility (AR)**: If the reader traveled short distances for recent queries, it is probably in a hotspot. An AR is a circle centered on the reader with a radius which is a moving average of distance traveled per query. Reader only accepts queries for destinations within the AR.
 3. **Flexible Grid**: Similar to AR, but with areas centered on fixed points on a grid rather than on the moving readers.
 4. **Rest Point (RP)**: Readers, after answering queries, return to within a radius of their RP, the average location of recent queries.
- Centralized case: the benefit of keeping readers near a hotspot is negated by overhead and non-optimal choices. Algorithms performed similarly.
 - Multihop: the advantage of Flexible Grid is great due to even spacing.

Simulation Data – Movement Algorithms



Caching

- Inclusion of caching is simple – whenever a reader receives any query for data in its cache, immediately return cached result.
- Active Caching**: Readers cache all data passed, even unrequested. For multihop case, where readers must travel both to and from data, add **asymmetric caching**: return via different path to pass near different data.
- Cache Exchange**: Readers exchange cache data with each other when within radio range, preventing cache misses due to network partitions.
- Analysis and simulations were performed primarily for the flexible grid case. Connectivity obviously helps caching and keeping the readers near a fixed point helps data locality.
- Improvement from caching is better in the case of hotspots, as the increased temporal and spatial locality increases hit rate.
- Simulations show significant improvement with asymmetric caching. They show significant improvement from cache exchange in 4-reader simulations but only minor improvement with 16 readers. With the flexible grid maintaining even spacing, the 16-reader case has high connectivity.



Demo Implementation

- Simple hardware platform with microcontroller and associated API provide a serial interface for hardware control. "Gumstix" small Linux computer runs algorithms and caching in Python, communicates over serial ports with microcontroller & Alien ALR-9932 RFID Reader, and communicates with other readers and query sources over 802.11.
- Test runs were performed to demonstrate feasibility and basic operation with naive and flexible grid algorithms using four readers in a ~8' x 8' space with tags located on an 11x11 grid. For multihop, artificially limit communications range.

