

다중 이기종 센서를 보유한 Nano-Q+ 기반 센서네트워크에서 메타데이터 라우팅 테이블을 이용한 질의 최적화

황상원⁰, 남영광, 이광용, 마평수

연세대학교 소프트웨어공학연구소, ETRI 임베디드 S/W 연구단
oracle_rhapsody@hotmail.com⁰, yknam@yonsei.ac.kr, (kylee, pmah)@etri.re.kr

Query Optimization with Metadata Routing tables on Nano-Q+ Sensor Network with Heterogeneous Sensors

Sang-Won Hwang⁰, Young-Kwang Nam, Kwang-Yong Lee, Pyung-Soo Mah
Software Engineering Lab., Yonsei University, ETRI Embedded S/W Research Division

요 약

센서네트워크에는 한 가지 종류의 센서만을 사용하는 것이 아니라 여러 가지 종류의 센서를 하나의 네트워크에 연결하여 한 번에 여러 종류의 센서의 질의를 수행할 수 있어야 한다. 이 경우에 다양한 센서가 부착된 네트워크에서 한 가지 종류의 센서에 대한 질의를 수행할 경우 해당되는 센서를 포함하지 않는 센서에 대해서도 질의를 수행해야 한다.

본 논문에서는 여러 종류의 센서가 부착된 센서네트워크에서 라우팅 테이블을 이용하여 질의를 최적화하는 방법을 제안한다. 센서노드의 라우팅 테이블에는 그 센서가 질의명령어를 보내고 받아야 하는 센서노드에 대한정보를 포함한다. 자식노드에 명령어를 보낼 경우 하위노드에서 질의하는 센서의 종류가 없을 경우에도 명령어를 보내야 한다. 이 경우 하위 자식노드에 원하는 센서의 종류에 관한 정보를 포함하면 센서의 종류에 따라서 불필요한 명령어의 수행을 줄일 수 있도록 하였다. 이 방법은 ATmega128 CPU를 장착한 Nano-Q+ 시스템에서 구현되었다.

1. 서 론

센서네트워크에는 한가지 종류의 센서만을 사용하는 것이 아니라 각 센서노드마다 여러 종류의 센싱장치를 부착하여 네트워크를 구성하는 것이 가능하다. 예를 들어, 온도, 조도, 습도, 혹은 on/off 스위치를 부착한 서로 다른 센서노드들을 하나의 네트워크로 구성하여 때로는 온도값만을 구한다든지 혹은 스위치의 연결 상태를 검색 하든지 아니면 온도와 스위치값을 동시에 AND 조건을 이용하여 해당되는 센서노드의 위치 혹은 값을 구할 수 있는 이기종 센서네트워크 시스템을 구축하여 이기종 센서노드에 대한 질의를 처리할 수 있다.

센서네트워크에서 질의는 서버에서 보낸 질의가 단말 센서노드로부터의 센서데이터가 서버까지 전달되어 질의 엔진에서 처리되기 위해서는 센서노드에서 여러 단계의 명령어 처리과정을 거친다. 그 중에서 네트워크 형상에 관계없이 센서노드에서는 질의 송출 (query dissemination)과 질의 처리 (query processing)의 두 단계로 나눌 수 있다. 질의송출은 부모노드로부터 받은 질의 명령패킷을 라우팅테이블에 있는 정보를 이용하여 현재 노드의 자식노드에게 전달하는 과정과 질의결과를 부모노드에게 전달하는 것이고 질의처리는 질의에 대한 센

싱과 집합함수(aggregate function)에 대한 처리를 하는 과정이다.

센서 네트워크에서의 질의 최적화는 관계형 데이터베이스에서 조인 연산의 수를 줄이거나 혹은 연산 순서를 변경하여 테이블 접근 횟수를 줄이도록 하는 질의최적화는 달리 에너지의 손실이 제일 적게 일어나도록 하는 것이 중요한 고려사항 중의 하나이다[2]. 센서노드 자체에서는 메모리의 크기가 작고 CPU의 성능이 떨어지기 때문에 복잡한 최적화 프로그램을 적재하는 것이 불가능하며 적재한다하더라도 센서노드의 수명이 다하면 회수가 거의 불가능하기 때문에 가능한 노드의 수명을 연장하는 방법으로 최적화가 이루어진다.

센서노드에서는 노드 내에서의 연산 혹은 센싱에 의한 전력소모 보다는 노드간에 정보를 전달할 때 가장 많이 전력을 소모하는 것으로 알려져 있다[4]. 따라서 센서노드 질의 처리에서 사용되는 최적화 방법은 가장 많이 소모하는 질의송출 단계에서 명령어와 질의결과를 보내는 횟수를 줄이는 것이다.

본 논문에서는 다중 이기종 센서를 보유한 센서네트워크에서 라우팅테이블의 정보를 이용하여 질의송출 시에 보내고 받는 패킷의 횟수를 줄여 센서노드의 에너지를 절약하는 방법을 제안한다. 각 센서노드의 라우팅 테이블에는 현재 노드의 부모노드와 자식노드의 정보를 포함

함과 동시에 각 노드의 현재노드부터 단말까지의 경로에 있는 센서의 유무에 대한 정보를 포함하도록 하여 현재 질의가 자식노드에 계 질의를 보낼지 말지에 대한 결정을 할 수 있는 정보를 포함하도록 하면, 즉 센서 S_i를 포함하는 센서노드가 하위 자식노드에 없으면 질의 패킷을 보내지 않도록 하여 에너지를 줄일 수 있다.

본 논문의 구성은 다음과 같다. 2장에서는 질의최적화와 관련된 연구에 대하여 비교검토하고, 3장에서는 Nano-Q+ 시스템에서 구현된 질의 처리 시스템의 구조에 대해서 설명한다. 4장에서는 MRT(Metadata Routing Table)을 이용한 질의최적화 시스템에 대하여 설명하고 5장에서는 제안된 방법에 대하여 구현된 결과와 시뮬레이션에 의한 성능향상에 대하여 기술한다.

2. 관련연구

본 연구와 가장 비슷한 질의최적화 연구로서는 SRT(semantic routing tree)를 이용한 것이다[4]. 이 SRT 방법은 각 노드는 각 노드가 가지고 있는 값의 특성을 부모노드에 저장하여 질의가 주어질 때 자식노드에서 제공되는 값이 질의에서 원하는 의미의 값에서 벗어날 경우에는 자식노드에게 질의를 송출하지 않도록 하는 방법이다. 따라서 SRT에는 항상 자식노드에 대한 특성을 저장하고 있어야 한다. 이 방법은 좀 더 발전시킨 방법으로서는 DSRT(Dynamic Semantic Routing Tree)가 있다[1]. 이 방법은 일정 기간 동안의 값을 저장하고 계속해서 그 값을 갱신하여 항상 최신의 값을 비교하여 그 값의 범위에 들면 질의를 처리하도록 하는 방법이다.

[3]에서 제시한 방법은 다중 집성함수의 계산을 요하는 질의에서 계산 결과를 공유하는 방법을 통해서 통신 비용을 최소화하도록 하였다.

위에서 제시한 방법들은 모두 센서네트워크의 모든 노드들이 같은 종류의 센서만을 가지고 있는 상태에서 최적화를 다룬 방법이다.

3. Nano-Q+ 기반 질의 처리 시스템

3.1 Nano-Q+ Platform

Nano-Q+ 시스템은 확장이 가능하고 재구성이 가능한 시스템으로서 운영체제와 센서하드웨어로 구성되어 있다.

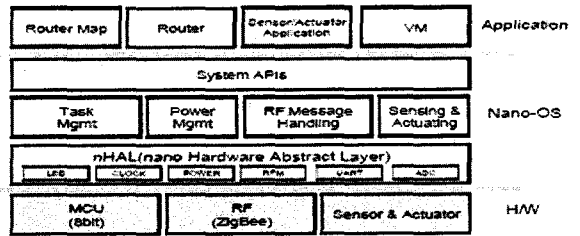


그림 1 Nano-OS 구조

Nano-Qplus 스택은 테스크 스케줄러, 나노-HAL, 전력관리기, RF 메시지 처리기와 같은 여러 핵심 모듈로 구성되어 있다. 그림[1]에 Nano-OS의 핵심 모듈의 구조가 나타나 있다. 센서하드웨어는 그림[2]와 같이 저비용, 저전력, 고도의 모듈화와 초점을 맞추어서 만들어진 것으로서 메인모듈(Main block), 기본블록, 센서모듈, Actuator모듈로 구성되어 있다. 메인모듈은 ATmega128 마이크로컨트롤러와 CC2420 IEEE 802.15.4 RF Transceiver를 가진다. 기본모듈은 RS-232 시리얼 인터페이스와 병렬 I/O, 그리고 전원장치로 구성되어 있다. 센서모듈은 조도, 습도, 온도와 Ultra를 측정하는 센서를 가질 수 있다. Actuator 모듈은 전기스위치와 기타 장치로 구성된다. 센서노드는 두 개의 AA 배터리를 사용한다.

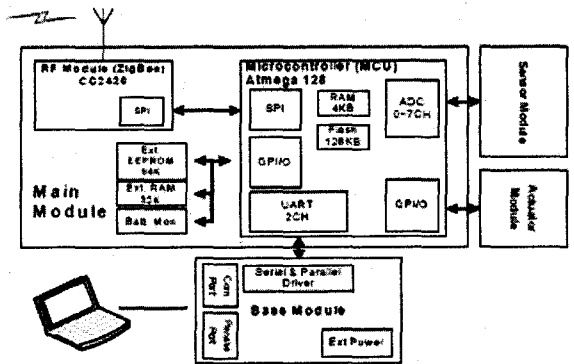


그림 2 센서노드 하드웨어 구조

3.2 통신 스택

Nano-Q+ 플랫폼의 확장성을 위해서 네트워크 스택은 Nano-OS위에 리눅스 기반 C 코드로 구현하였다. 그림 [3]에 계층모델에 기반한 네트워크 구조가 나타나 있다. 추가로 라우팅 테이블과 ad-hoc 노드의 수를 적게하기

위해서 클러스터-헤더 기반의 스타-메쉬 네트워크 구조를 사용하였다. 제 각 모듈의 기능은 다음과 같다.

MAC : MAC 계층의 주요 기능은 충돌 방지를 위한 노드 접근을 제어하는 것이다. CSMA-CA에 기반한 IEEE 802.15.3의 일부분을 구현하였다. 센서통신을 위해서 에너지 절약과 조정기능에 중점을 두었다.

라우팅 : 센서네트워크에서의 라우팅은 인터넷 혹은 ad-hoc 무선 네트워크에서의 라우팅과 다르다. 예를 들어 센싱 데이터는 싱크노드에 의해서만 전달된다. 또한 작동명령은 싱크노드로부터 actuator 노드로 전달되도록 설계하여 네트워크를 운영하기 위한 메모리와 전력의 사용을 절감할 수 있도록 하였다.

소켓 API : 센서노드의 메모리가 매우 적기 때문에 TCP나 UDP와 같은 소켓 API를 운영체제의 일부로서 제공한다.

패킷 형태 : 계층적 모델에서, 다른 계층에 대해 독립적으로 구현되어 있다. 각 계층은 헤더 필드의 내용을 채운후에 하위계층으로 보낸다.

Rx/Tx 버퍼 : 제한된 네트워크 스택에는 입출력되는 패킷을 처리하기 위해서 다섯 개의 배열 포인터를 가진 독립적인 Rx/Tx 버퍼를 가지고 있다.

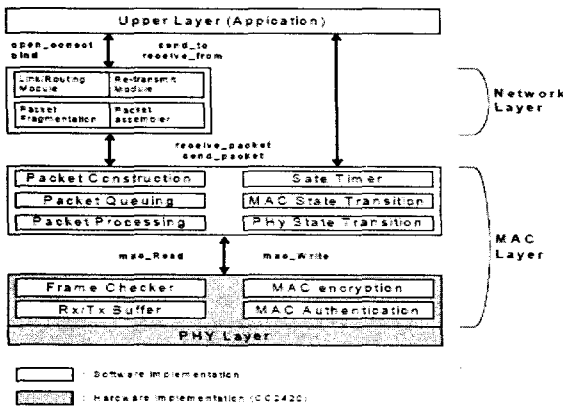


그림 3 통신 스택의 계층 구조

3.3 NanoDB (Nano-Qplus 질의처리 시스템)의 구조

Nano-Q+ 기반에서의 질의처리시스템을 NanoDB라 한다. NanoDB 시스템의 구조는 [그림 4]과 같다. PC 서버에서는 사용자를 위한 GUI, 질의파서, 명령어생성기, 명령어송출기, 결과변환기, 질의처리기등으로 구성된다. 사용자 GUI에서는 그래픽기반 GUI와 사용자가 직접 질의를 입력할 수 있는 텍스트기반 GUI가 제공된다. 그래픽

기반 GUI는 [그림 5]와 같다. 사용자는 원하는 질의조건만 입력하면 결과가 GUI 화면에 출력된다. 텍스트 기반 GUI에서는 질의를 XQuery로 입력하면 결과가 출력창에 출력된다. 텍스트기반 GUI의 실행예제는 [그림 6]과 같다. 질의파서는 질의를 파싱하며, 명령어 생성기 (command generator)는 파싱된 질의를 기반으로 센서노드에게 전달한 명령어 패킷(command packet)을 생성한다. 생성된 명령어 패킷은 센서노드가 인식할 수 있는 패킷으로 변환되어 각 질의에 적합한 패킷으로 변환되어 명령어송출기 (command disseminator)에 의해 각 센서노드에 전달된다. 결과변환기는 센서노드에서 측정된 데이터 패킷을 질의처리에 적합한 XML 형태로 변환하는 역할을 한다.

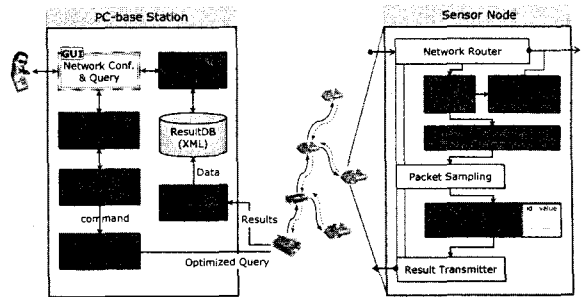


그림 4 NanoDB의 구조

서버로부터 명령어 패킷을 전달받은 센서노드는 각 노드가 가지고 있는 라우팅 테이블을 이용하여 부모로부터 받은 명령어를 자신의 자식노드에 맞게 재생성하여 명령을 보낸다. 명령어가 단말노드까지 전달되면 단말노드는 센싱한 데이터를 부모노드로 보내는데 이때에서는 라우팅 테이블에 있는 부모노드의 정보를 이용하여 센싱결과를 보낸다. 내부센서노드는 하위 자식노드로부터 전달받은 질의결과를 해석하여 외장 플래쉬 메모리에 저장하고 모든 자식노드로부터 결과를 받았는지를 확인하고 모두 받아서 보내고 난 후에 sleeping 모드로 전환된다.

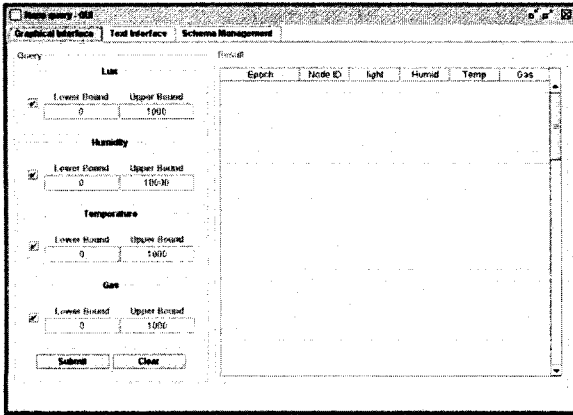


그림 5 그래픽기반 GUI

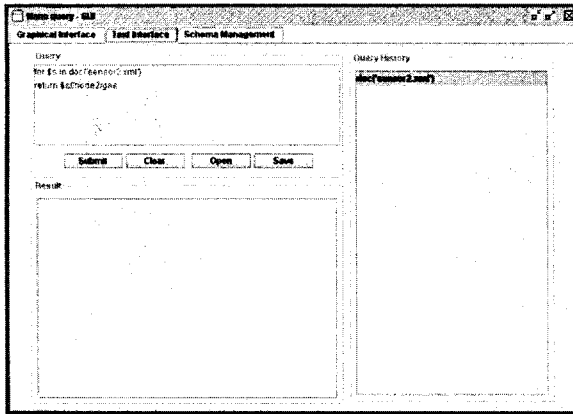


그림 6 텍스트 기반 GUI

4. MRT 기반 질의 최적화

4.1 MRT의 구조

앞 장에서 언급한 바와 같이 각 센서노드에는 PC서버에서 보내온 명령어패킷을 부모노드로부터 받아서 자식노드에게 전달할 패킷형태로 재생성하여 전달한다. 명령어 패킷에는 센싱할 데이터의 종류(이하 메타데이터라 칭함)와 상한값, 하한값, 혹은 노드의 식별자에 대한 정보가 포함될 수 있다. 그래픽 기반 혹은 텍스트 기반 GUI에서 사용자가 보내는 질의는 모든 종류의 메타데이터에 대한 값을 원하지 않을 수 있다. 이 때 원하는 메타데이터가 하위 자식노드에 존재하지 않는다면 하위자식노드에 대한 식별자를 포함하지 않도록 하여 명령어패킷의

송수신횟수를 줄일 수 있다. 어떤 센서노드 S_i 에 대하여 C_i^k 를 S_i 의 k 번째 자식노드라 하고, PC_i^k 는 노드 C_i^k 에서 단말 노드까지의 경로라 하자. 메타데이터를 $M_j = \{M_1, M_2, \dots, M_j\}$ 는 현재의 질의 시스템에서 사용될 수 있는 센서의 종류라 하고, $C_i^k M_j$ 를 노드 S_i 에서 S_i 의 PC_i^k 의 경로에 메타데이터 M_j 의 값의 유무를 알려주는 값 $C_i^k M_j$ 는 다음과 같이 정의된다.

$$C_i^k M_j = 1 \text{ if any sensor nodes in } PC_i^k \text{ have } M_j.$$

$$0 \text{ if any sensor nodes in } PC_i^k \text{ have no } M_j.$$

그러므로 M_j 는 현재 센서노드 S_i 가 가지고 있는 메타데이터의 종류를 의미하고 $C_i^k M_j$ 는 현노드 S_i 의 자식노드 C_i^k 에서부터 단말노드까지의 노드 중에서 메타데이터 M_j 를 포함하는 지의 여부를 결정한다. 다시 말하면, 자식노드에서 더 이상 M_j 를 포함하지 않는다면 자식노드 C_i^k 에 대해서는 더 이상 명령어 패킷을 보내지 않도록 하여 명령어를 보내고 받는 횟수를 줄이도록 한다.

Si (current node id)		Pi (Parent node id)	
M_1	M_2	...	M_j
C_i^1	C_i^2	...	C_i^k
$C_i^1 M_1$	$C_i^1 M_2$	$C_i^1 M_j$	$C_i^2 M_1$
$C_i^2 M_1$	$C_i^2 M_2$	$C_i^2 M_j$	$C_i^k M_1$
$C_i^k M_1$	$C_i^k M_2$	$C_i^k M_j$	$C_i^k M_j$

그림 7 센서노드 S_i 에서의 라우팅 테이블의 구조

만약 현재의 질의 $Q(M)$ 에 대해 S_i 의 자식노드 C_i^k 의 메타데이터 값 $C_i^k M_j = 0$ 이면 S_i 노드에서 자식노드 C_i^k 부터 M_j 에 대한 데이터를 얻을 수 없다는 것을 의미하기 때문에 명령어 패킷을 송출하지 않는다.

4.2 노드 추가 시 MRT의 값의 수정

앞에서 본 각 센서노드의 MRT는 센서네트워크에 노드가 하나씩 추가될 때마다 표의 값을 수정해야 한다. 즉 단말노드 S_i 가 네트워크 N 에서 사용가능한 메타데이터의 개수가 j 라고 가정한다. 즉 센서네트워크 N 이 가지는 메타데이터의 집합은 $M = \{M_1, M_2, \dots, M_j\}$ 이다. 만약, j 개 외에 메타데이터를 더 추가하려고 할 경우에는 메타데이터관리를 통하여 PC버서에서 각 노드에 메타데이터를 추가하는 명령을 실행하여 추가할 수 있다. $M_i = \{M_i^1, M_i^2, \dots, M_i^j\}$ 를 노드 S_i 가 가지고 있는 메타데이터의 집합이라고 가정한다. 따라서 센서노드 S_i 를 추가할 경우 M_j 는 M 의 부분집합이어야 한다. 노드 S_i 의 부모노드 P_i 의 자식노드중의 하나가 S_i 이므로 S_i 가 가지고 있는 메

타데이터의 집합 $M_i = \{M_i^1, M_i^2, \dots, M_i^j\}$ 중에서 M_i^j 의 값이 1인 M_i 에 대해 P_i 의 자식노드 S_i 의 M_i^j 값을 1로 모두 수정한다. 위와 같은 방법으로 하여 M_i 와 S_i 의 부모노드의 M_i 가 일치할 때까지 부모노드를 방문하면서 수정한다.

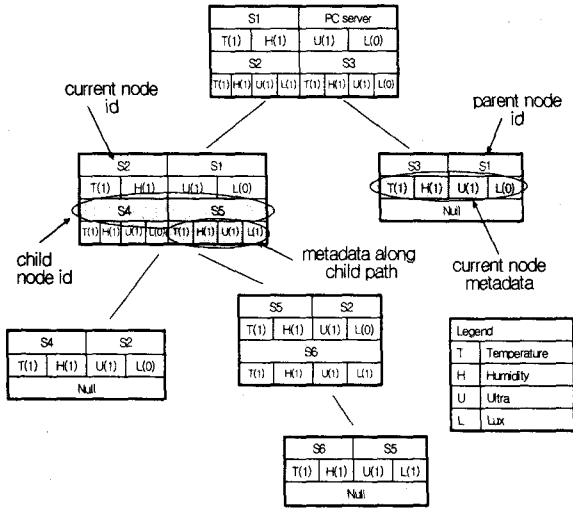


그림 8 네 개의 메타데이터를 가진 MRT 구조의 예

4.3 노드 삭제시 MRT 값의 수정

센서네트워크에서 임의의 노드를 삭제할 경우, 현재 노드의 부모 노드가 가지고 있는 메타데이터 값의 수정이 필요하다. 노드 추가시와는 반대로 부모 노드를 방문하면서 삭제된 노드가 가지고 있던 메타데이터의 값을 자식노드의 값으로 변경해야 한다. 수정을 위한 알고리즘은 다음과 같다.

```

Let P(Si) be the parent of Si;
While P(Si) is not null do
  for i = 1 to k
    Modify the value of of Sik in P(Si) by Mi =
    union of meta values of Cij;
    Si = P(Si);
end
    
```

5. 성능평가

MRT를 이용한 성능향상 정도를 계산하기 위하여 다음과 같이 가정한다. 어떠한 트리 $G=(V, E)$ 로서 V 는 꼭지점의 집합이고 E 는 edge의 집합이라고 하자. 트리 G 의 깊이(depth)를 d 라하고, h 는 나무의 높이, k 를 차수

(degree)라 한다. 이 경우 모든 노드가 질의에 참여한다고 가정하면 $|E| = |V|-1 = \frac{k^{h+1}-1}{k-1}$ 이기 때문에 총 통신회수는 $2 * |E| = 2 * \frac{k^{h+1}-1}{k-1}$ 가 됨을 알 수 있다. 만약 어떠한 메타데이터 M_i 가 깊이가 d 까지만 존재하는 노드의 개수의 비율을 $x\%$ 라 하면 이 때의 통신회수는 $2 * \left[\left(\frac{k^{h-d+1}-1}{k-1} * \frac{k^d * x}{100} \right) + \frac{k^d-1}{k-1} \right]$ 가 된다.

6. 결론 및 향후 연구과제

본 논문에서는 Nano-Q+ OS를 이요한 센서네트워크에서 질의 검색 시스템을 구현함과 동시에 MRT를 이용한 질의 최적화 방법을 제안하였다. 향후 연구과제로서는 Nano-Q+ 시스템을 위한 시뮬레이터를 이용하여 MRT를 이용한 에너지의 소비량과 MRT를 유지하기 위한 연산비용을 고려하여 실제로 어느 정도의 에너지가 절약되는 지를 알아보는 것이다.

참고문헌

[1] J. Zhu, Dynamic Semantic Routing in Sensor Network, Dept. of CS, RIT, NY, USA.
 [2] J. Gehrke, S. Madden, Query Processing in Sensor Networks, Pervasive computing, 2004, Jan-Mar
 [3] J. Shneidman, P. Pietzuch, M. Welsh, et. al, A Cost-Space Approach to Distributed Query Optimization in Stream Based Overlays
 [4] Samuel R. Madden, Michael J. Franklin, Joseph M. Hellerstein, and Wei Hong. Tinydb: an acquisitional query processing system for sensor networks. ACM Trans. Database Syst., 30(1):122 - 173, 2005.
 [5] Yong Yao and Johannes Gehrke. The cougar approach to in-network query processing in sensor networks. SIGMOD Rec., 31(3):9 - 18, 2002.
 [6] A. Agarwal and T.D.C. Little, Attribute Based Routing in Sensor Networks, Boston University, Boston, Massachusetts, MCL Technical Report No. 06-01-2006
 [7] N. Trigoni, et al., Multi-query Optimization for Sensor Networks