Optimal Policies for Distributed Data Aggregation in Wireless Sensor Networks

Hussein Abouzeid

Department of Electrical Computer and Systems Engineering
Rensselaer Polytechnic Institute
abouzeid@ecse.rpi.edu

Joint work with Zhenzhen Ye and Jing Ai (yez2@rpi.edu,aij)

May 10, 2007
Motivation: To Send or not to send

- A fundamental **trade-off** arises in data aggregation
  - **Send** immediately: No aggregation gain, ie energy loss due to redundant data transmission; but possibly lower delay hence lower distortion
  - **Wait** for more samples/packets to arrive: Higher degree of aggregation (DOA) means energy savings, but also higher delay & distortion.

- Decision making: a node should decide *optimal* instants to send so as to balance aggregation gain vs. delay.
Related Work

- Accuracy-driven data aggregation e.g. [Boulis et al. 2003].
  - Nodes decide transmission depending on an accuracy threshold.
- Timing control in tree-based aggregation
  - Fixed transmission schedule at a node once an aggregation tree is constructed, fixed and bounded wait time e.g., Directed Diffusion [Intanagonwiwat et al, 2000], TAG [Madden et al. 2002], SPIN [Heinzelman et al. 1999] and Cascading timeout [Solis et al. 2003].
  - Quality-driven, adjustable transmission schedule (by the sink node) [Hu et al. 2005].
- Distributed control of DOA [He et al., 2004]
  - FIX scheme
    - Fixed wait time for all nodes.
  - On Demand (OD) scheme
    - Each node locally adjusts its DOA, based on the delay on the MAC layer. Stops aggregating whenever MAC queue empty.
    - The control loop aims to minimize the MAC layer delay while energy saving is only an ancillary benefit.
A Sequential Decision Problem

- The random arrival of samples at a node can be viewed as a point process, called natural process.
- The availability of the multi-access channel for transmission is another random process (assuming a random access MAC protocol), defining the decision epochs.
- The state of a node is defined as the number of samples aggregated at the node, including locally generated samples.
- A decision epoch is the instant that the node has at least one sample and the channel is available for transmission.
- At each decision epoch, the node should choose a suitable action, i.e., to continue to wait for more aggregation \((a = 0)\) or stop current aggregation operation and send out the sample immediately \((a = 1)\).
A Sequential Decision Problem (Cont’d)

Random Sample Arrival (Natural Process)

Random Available Transmission Epochs

$\delta W_1 \delta W_2 \delta W_3$

$X_1 X_2 X_3$

$s=X_1$

$a=0$

$s=X_1+X_2$

$a=0$

$s=X_1+X_2+X_3$

$a=0$

$s=s_n$

$a=1$

$s=\Delta$

$s=1$

Decision Horizon

An assumption in modelling the decision process (Assumption 2.1)

- Given the state $s_n \in S'$ at the $n$th decision epoch, if $a = 0$, then the random time interval $\delta W_{n+1}$ to the next decision epoch and the random increment $X_{n+1}$ of the node's state are independent of the history of state transitions and the $n$th transition instant $t_n$. 

$\delta W_n$
A semi-Markov Decision Process Model

- SMDP described by the 4-tuple \( \{ S, A, Q^a_{ij}(\tau), R \} \).
- State space \( S = S' \cup \{ \Delta \} \), where \( S' = \{1, 2, \ldots\} \) and \( \Delta \) is an (artificial) absorbing state.
- Action set \( A = \{0, 1\} \), with \( A_s = \{0, 1\} \), \( \forall s \in S' \) and \( A_s = \{0\} \) for \( s = \Delta \).
- State transition distributions \( Q^a_{ij}(\tau) \), the distribution from state \( i \) to \( j \) given the action at state \( i \) is \( a \);
- Instant aggregation rewards \( \{ r(s, a) \} \), where \( r(s, a) = g(s) \) iff \( a = 1 \) and \( s \in S' \); \( g(s) \) is the aggregation gain achieved by aggregating \( s \) samples when stopping.
With the SMDP model, the objective of the decision problem is to find a policy $\pi^*$ composed of decision rules $d_n$ at decision epochs $n = 1, 2, \ldots$, to maximize the expected reward of aggregation.

To incorporate the impact of aggregation delay penalty in decisions, the expected total discounted reward optimality criterion with a discount factor $\alpha > 0$ is used; The optimal expected reward given initial state $s$

$$v^*(s) = \sup_{\pi} \left\{ E_s^\pi \left[ \sum_{n=0}^{\infty} e^{-\alpha t_n} r(s_n, d_{n+1}^\pi(s_n)) \right] \right\}$$
The Optimal Solution

- Under Assumption 2.2 (bounded expected reward under any policy and zero gain for infinite wait),
  - Optimality Equations:
    \[ v(s) = \max \{ g(s) + v(\Delta), \sum_{j \geq s} q_{sj}^0(\alpha)v(j) \}, \forall s \in S' \]
    and \( v(\Delta) = v(\Delta) \) for \( s = \Delta \), where \( q_{sj}^a(\alpha) \) is the Laplace-Stieltjes transform of \( Q_{sj}^a(\tau) \)
  - Can show by standard methods that a stationary decision policy exists, & the Optimal Decision Rule \( d \) is given by:
    \[ d(s) = \arg \max_{a \in A_s} \{ g(s), \sum_{j \geq s} q_{sj}^0(\alpha)v^*(j) \}, \forall s \in S' \]
    and \( d(\Delta) = 0 \).

- Challenges/Questions:
  1. Relies on the computation of \( v^* \) which might be computationally expensive for sensors.
  2. If certain conditions hold, are there simpler policies that are also optimal? specifically ones that do not require solving for \( v^* \)?
  3. Without structured policies, any approximate solutions and algorithms available for \( v^* \) and \( d \)?
The action is monotone in state space:

\[
d(s) = \begin{cases} 
0 \text{(wait)} & s < s^* \\
1 \text{(transmit)} & s \geq s^* 
\end{cases},
\]

where \( s^* \) is called a control limit.

The search for an optimal policy is reduced to find \( s^* \).

Attractive for implementation in energy/computation limited sensor networks.
Theorem 1

If \( g(s) \geq \sum_{j \geq s} q_{sj}^0(\alpha)g(j) \) for all \( i \geq s, \; i, s \in S' \) once it holds for certain \( s \), then a control-limit policy with control limit

\[
s^* = \min \{ s \geq 1 : g(s) \geq \sum_{j \geq s} q_{sj}^0(\alpha)g(j) \} \tag{2}
\]

is optimal...

- Implication: if it’s better to stop at current stage than just continuing one more stage and then stop, it’s optimal to stop now - One-Stage-Lookahead

- Difficult to check.
Corollary

Suppose \( g(i+1) - g(i) \geq 0 \) is non-increasing with state \( i \) for all \( i \in S' \) and if the following inequality holds for all states \( i \geq s, \ i, s \in S' \) once it is satisfied at certain \( s \),

\[
\sum_{j \geq k} Q_{ij}^0(\tau) \geq \sum_{j \geq k} Q_{i+1,j+1}^0(\tau), \ \forall k \geq i, \ \forall \tau \geq 0. \tag{3}
\]

Then, there exists an optimal control-limit policy.

Roughly, in words, a control limit policy is optimal when:

- the aggregation gain is concavely or linearly increasing with the number of collected samples; and,
- with a smaller number of collected samples at the node, it is more likely to receive any specific number of samples or more, than that with a larger number of samples already collected, by the next decision epoch.
Further assume that the inter-arrival time of consecutive decision epochs and the increment of the states are independent of the current state of the node; and

A linear aggregation gain setting $g(s) = s - 1$,

$$s^* = \left\lceil \frac{E[Xe^{-\alpha \delta W}]}{1 - E[e^{-\alpha \delta W}]} + 1 \right\rceil$$

(4)

Comparison to existing aggregation policies in [He et al. 2004]

- $s^*$ in (4) not fixed DOA threshold as in the FIX scheme
- In the extreme case, $\alpha \to \infty$ (very high delay penalty) $s^* \to 1$, (4) is reduced a policy similar to the On-demand (OD) scheme.
In case that the optimal policies of special structures do not exist, we have to look for approximate solutions of the optimal equations.

A finite-state approximation model: Considering the truncated state space $S_N = S'_N \cup \{\Delta\}$, $S'_N = \{1, 2, \ldots, N\}$ and setting $v_N(s) = 0$, $\forall s > N$, the optimality equations become

$$
    v_N(s) = \max \{g(s) + v_N(\Delta), \sum_{j \geq s} q^0_{sj}(\alpha) v_N(j)\}
$$

for $s \in S'_N$ and $v_N(\Delta) = v_N(\Delta)$.

**Theorem 2**

$$
    \lim_{N \to \infty} v^*_N(s) = v^*(s), \forall s \in S'.
$$
On-line Algorithms for the Finite-State Approximation

**ARTDP**

- $q^0_{sj}(\alpha)$ are unknown in practice, we should either obtain the estimated values of $q^0_{sj}(\alpha)$ from actual aggregation operations or use an alternate “model-free” method.


- An asynchronous value iteration scheme for MDP.
- Merges the model building procedure into value iteration, suitable for on-line implementation.
- We modify it for the SMDP model with a truncated state-space.
- Decision rule:
  
  $$d^*_N(s) = \arg \max_{a \in \{0,1\}} \{ g(s), \sum_{N \geq j \geq s} \hat{q}^0_{sj}(\alpha) \nu^*_N(j) \} \text{ for } s \in S'_N \text{ and } d^*_N(s) = 1 \text{ for } s > N.$$
**Algorithm I: ARTDP**

1. Set $k = 0$
2. Initialize counts $\omega(i, j)$, $\eta(i)$ and $\hat{q}_{ij}^0(\alpha)$ for all $i, j \in S'_N$
3. **Repeat**
   4. Randomly choose $s_k \in S'_N$;
   5. **While** ($s_k \neq \Delta$) {
      6. Update $v_{k+1}(s_k) = \max \{g(s_k), \sum_{N \geq j \geq s_k} \hat{q}_{sj}^0(\alpha)v_k(j)\}$;
      7. Rate $r_{s_k}(0) = \sum_{N \geq j \geq s_k} \hat{q}_{sj}^0(\alpha)v_k(j)$ and $r_{s_k}(1) = g(s_k)$;
      8. Randomly choose action $a \in \{0, 1\}$ according to
         $P_r(a) = \frac{e^{r_{s_k}(a)/T}}{e^{r_{s_k}(0)/T} + e^{r_{s_k}(1)/T}}$
      9. **if** $a = 1$, $s_{k+1} = \Delta$;
      10. **else** observe actual state transition $(s_{k+1}, \delta W_{k+1})$
          $\eta(s_k) +=$;
      11. **if** $s_{k+1} \leq N$,
          Update $\omega(s_k, s_{k+1}) = \omega(s_k, s_{k+1}) + e^{-\alpha\delta W_{k+1}}$;
      12. Re-normalize $\hat{q}_{sj}^0(\alpha) = \frac{\omega(s_k, j)}{\eta(s_k)}$, $\forall N \geq j \geq s_k$;
      13. **else** $a = 1$, $s_{k+1} = \Delta$;
      14. $k +=$.
   } }

- line 6: reward update with current estimated system model;
- line 7-9: randomized action selection to avoid the overestimation of rewards.
RTQ

- In a “model-free” method, we avoid to estimate $q_{sj}^0(\alpha)$.
- Algorithm II: Real-time Q-learning (RTQ) [Barto et al. 1995]
  - Does not take advantage of the semi-Markov model.
  - Relies on stochastic approximation for asymptotic convergence to the desired Q-function. In our case, the optimal Q-function is $Q_N^*(s, 1) = g(s)$, $Q_N^*(s, 0) = \sum_{j \geq s} q_{sj}^0(\alpha) v_N^*(j)$, $\forall s \in S_N'$, $Q_N^*(s, a) = 0$, $\forall s > N$, $a \in \{0, 1\}$ and $Q_N^*(\Delta, 0) = 0$.
  - A lower computation cost in each iteration than ARTDP but converges more slowly.
- Decision rule:
  
  $$d_N^*(s) = \arg \max_{a \in \{0, 1\}} \{ Q_N^*(s, a) \} \tag{6}$$

  for $s \in S_N'$ and $d_N^*(s) = 1$ for $s > N$. 

On-line Algorithms for the Finite-State Approximation
Algorithm II: RTQ

1 \hspace{1em} \textbf{Set} \ k = 0
2 \hspace{1em} \text{Initialize Q-value } Q_k(s, a) \text{ for } s \in S'_N, a \in \{0, 1\}
\hspace{2em} \text{and set } Q_k(s, a) = 0, \forall s > N, a \in \{0, 1\}
3 \text{ Repeat } \{ 
4 \hspace{1em} \text{Randomly choose } s_k \in S'_N; 
5 \hspace{1em} \textbf{While} (s_k \neq \Delta) \{ 
6 \hspace{2em} \text{Rate } r_{s_k}(0) = Q_k(s_k, 0) \text{ and } r_{s_k}(1) = Q_k(s_k, 1); 
7 \hspace{2em} \text{Randomly choose action } a \in \{0, 1\} \text{ according to} 
\hspace{3em} P_r(a) = \frac{e^{r_{s_k}(a)/T}}{e^{r_{s_k}(0)/T} + e^{r_{s_k}(1)/T}}; 
8 \hspace{2em} \textbf{if} a = 1, s_{k+1} = \Delta, 
9 \hspace{2em} \text{Update } Q_{k+1}(s_k, 1) = (1 - \alpha_k)Q_k(s_k, 1) + \alpha_k g(s_k); 
10 \hspace{2em} \textbf{else} \text{ observe actual state transition } (s_{k+1}, \delta W_{k+1}), 
11 \hspace{2em} \text{Update } Q_{k+1}(s_k, 0) = (1 - \alpha_k)Q_k(s_k, 0) + 
12 \hspace{3em} \alpha_k [e^{-\alpha \delta W_{k+1}} \max_{b \in \{0, 1\}} Q_k(s_{k+1}, b)] 
13 \hspace{2em} \textbf{if } s_{k+1} > N, a = 1, s_{k+1} = \Delta; 
14 \hspace{2em} k +=. \} \}

- line 7-8: randomized action selection (i.e., exploration);
- line 9-13: Q-value update according to actual state transition.
Performance Evaluation

1. Compare the schemes using a synthetic tunable traffic model
   ▶ Easier to isolate causes and effects; e.g. effect of state dependency

2. Compare the schemes using a distributed data aggregation simulation
   ▶ more closely resembles a real network
1. Schemes in Comparison and a tunable Traffic Model

- **Schemes in Comparison**
  - Control-limit policies: CNTRL (Theorem 1) and EXPL (eqn. (4))
  - Learning schemes: ARTDP and RTQ
  - LP: an off-line LP solution for the optimal reward is included as a performance reference, which uses the learning system model with a sufficient large number of iterations.

- **Traffic Model**
  - Inter-arrival time of decision epochs - Exponential with the mean
    \[
    \delta W_s = \delta W_0 e^{-A(s-1)} + \delta W_{min}
    \]
    where constant \( \delta W_{min} > 0 \).
  - Random sample arrival - Poisson with the rate
    \[
    \lambda_s = \lambda_0 e^{-B(s-1)}
    \]
    where \( A \geq 0 \) and \( B \geq 0 \) control the degree of state-dependency.
The Effect of State-dependency

- $N = 40$, state space truncation effect is negligible;
- Upper plot: low state-dependency, all policies converges to the optimal value of reward;
- Lower plot: high state-dependency, EXPL is sub-optimal since its optimality condition is not satisfied;
- ARTDP converges faster than RTQ - benefit of learning the system model.
Consider low state-dependency case in which EXPL is close to optimal;

Upper plot: when $N=10$, state space truncation effect is significant, calculated values (i.e., LP solution) is lower than the actual (measured) values;

Bottom plot: when $N=20$, much less state space truncation effect, LP solution is close to actual (measured) values;
2. Application Scenario and Parameters

- **Problem Context:** Distributed data aggregation
  - Each sensor estimates information of the whole sensing field through local data exchange and aggregation. Fully distributed, robust and flexible.

- 25 sensor nodes in a 2D square sensing field to track the maximum value of an underlying slow time-varying phenomenon.

- Omnidirectional antenna transmission range $r_0 = 10$ meters; the inter-node communication data rate is 38.4 kbps. Original sample size is 16 bits.

- Energy consumption model (MICA2-like): 686 nJ/bit for radio transmission, 480 nJ/bit for reception, 549 nJ/bit for processing and 343 nJ/bit for sensing.

- Delay discount factor $\alpha = 8$; the degree of finite-state approximation $N = 10$; nominal aggregation gain $g(s) = s - 1$. 
ARTDP and RTQ achieve the highest reward values; all proposed schemes outperform OD and FIX schemes;

reward for FIX with DOA=3 decreases when sampling rate increases, due to heavier congestion in the network;
CNTRL has the lower delay than ARTDP, RTQ, EXPL and OD, due to its smaller degree of aggregation (DOA);

delay in FIX with DOA=3 increases fast (in logarithm scale) with the sampling rate, due to congestion;
OD has highest energy cost since aggregation is only opportunistic.

EXPL has the lower energy cost than ARTDP, RTQ and CNTRL, due to its higher DOA;
The proposed schemes (as well as OD) can adapt the DOA with different sampling rates. No universal DOA.

DOA: CNTRL $<$ RTQ $\leq$ ARTDP $<$ EXPL (can explain energy-delay tradeoff in last two figures: a higher DOA, a higher energy saving but a longer delay);
Conclusion

► Provided a stochastic decision framework to study energy-delay tradeoff in distributed data aggregation.

► Formulated the problem of balancing the aggregation gain and delay as a sequential decision problem, under certain assumption, becomes a SMDP.

► Provided practically attractive control-limit policies and on-line learning algorithms and investigated their performance under a tunable traffic model and a practical distributed data aggregation scenario; the proposed schemes outperformed the existing schemes.
Thanks. Questions, comments,...