Distributed Estimation with Unreliable Communications

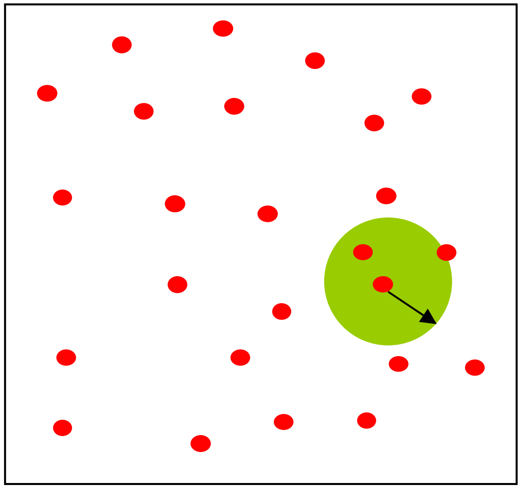
David Castañón and Venkatesh Saligrama

Boston University

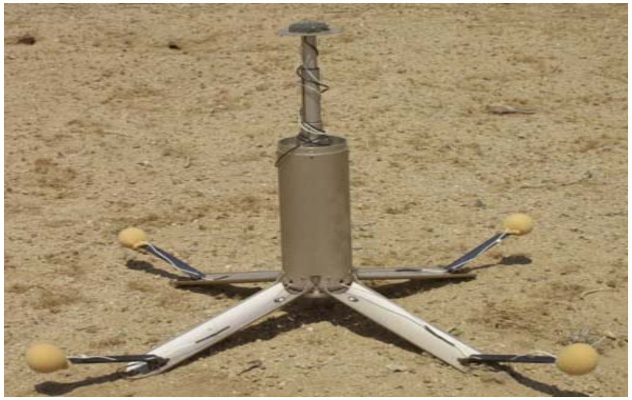
Sponsored by NSF

Sensor Networks

- Sensor Networks
  - Interconnected networks of sensing devices, typically wireless
  - Heterogeneous sensing, communications, processing capabilities per node
  - Spatially distributed to create observability, redundancy



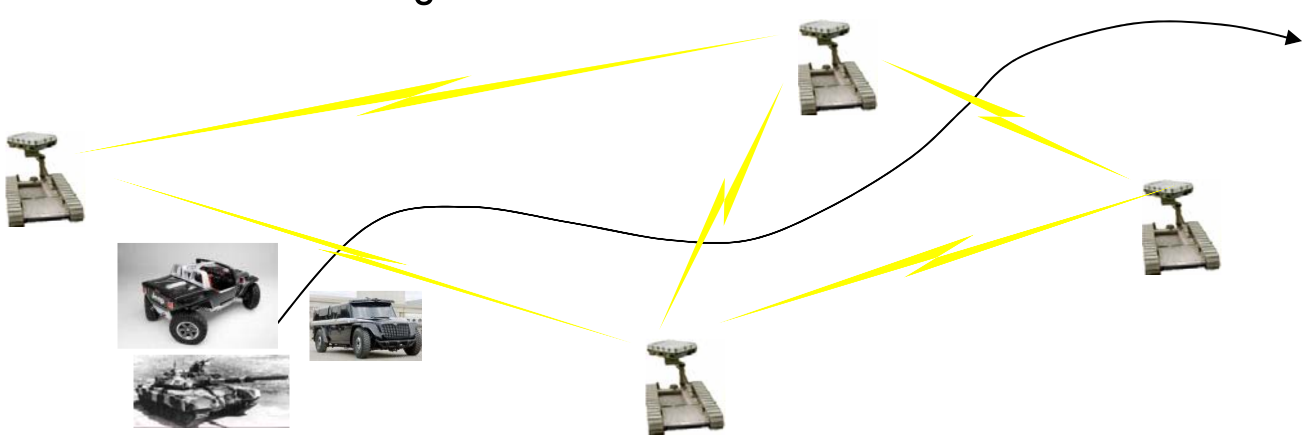
- Important Aspects
  - Power requirements for comms >> local computation



- Challenge: Efficient fusion of information in the presence of unreliable communications
  - Focus on tracking information

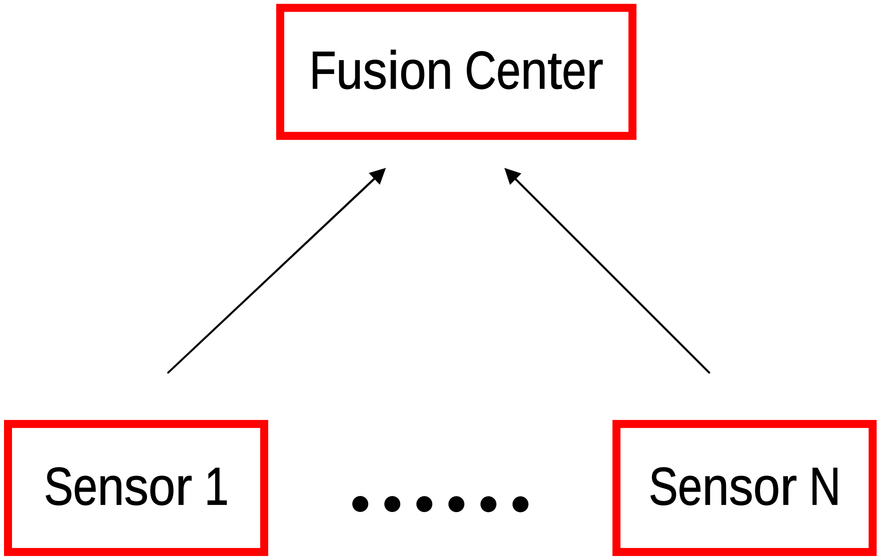
Motivating Application

- Tracking using distributed network of REDOWL sensors
  - IRobot body with sensor “head”
  - Acoustic arrays for source separation, direction finding, feature extraction, classification of sound sources
  - Local estimates of source bearing and features generated at a rate of 4 Hz
  - Active ladar and video camera mounted on pivoting head for generating video and ladar range data
  - Mobile
  - Networked using 802.11b



- Issues
  - Limited sensing range
  - Lack of single sensor observability
  - Packet losses
  - Comms rate smaller than measurement rate

Model



- N Sensors
- Linear Target Dynamics

$$x_{t+1} = Ax_t + w_t$$

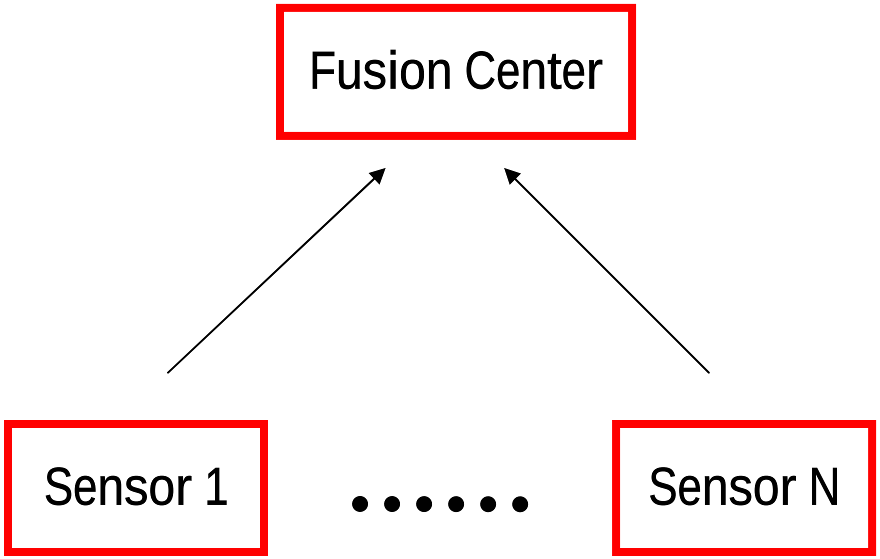
- Sensor k observations:
  - Idealized model: linear, Gaussian model

$$y_t^k = C^k x_t + v_t^k, k = 1, \ldots, n$$

  - Noise white, independent; x(0) ~ N(0,Σ_0)
  - Application: extend to nonlinearities including range-dependent detection
- Comms
  - Each sensor can transmit messages concerning its information
  - Packet losses ➔ comm. messages may not reach fusion center
    - Different statistical models for message losses
- Challenge:
  - Efficient fusion of information in the presence of unreliable communications

Previous Work

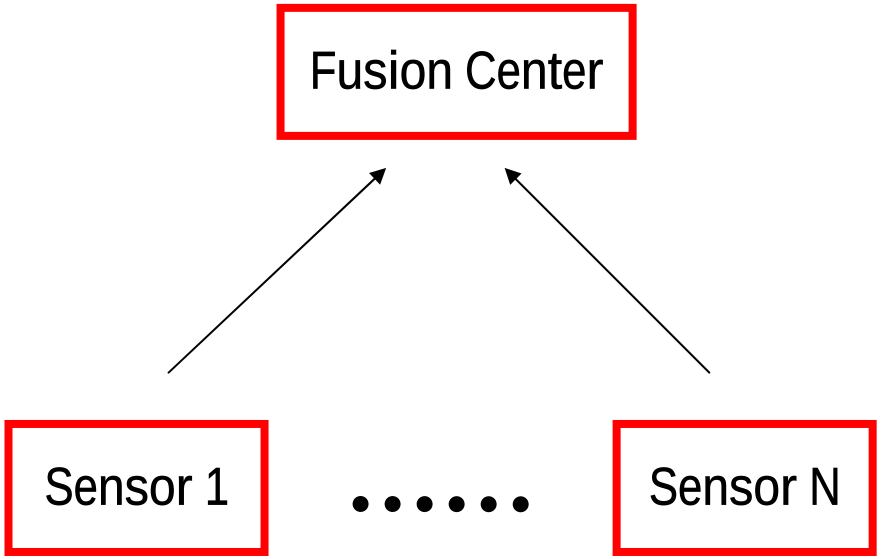
- Distributed Estimation (Speyer, Willsky, Bello, Castañón, Levy, Verghese, … ('79-'85)), Gupta, Murray et al (04, …)
  - Communicate optimal estimates of current state given local observations only
  - Fusion center performs dynamic processing to "decorrelate" and combine estimates
  - When communications rate is smaller than measurement rate, messages can include local compensators to replace fusion processing
- Main Result: recover centralized estimate as if all measurements were directly available at fusion center



- Limitations:
  - No message losses (or very limited)
  - Synchronous communications
  - Known "global" statistics
    - number of sensors
    - measurement error statistics
    - Local models

Previous Work – 2

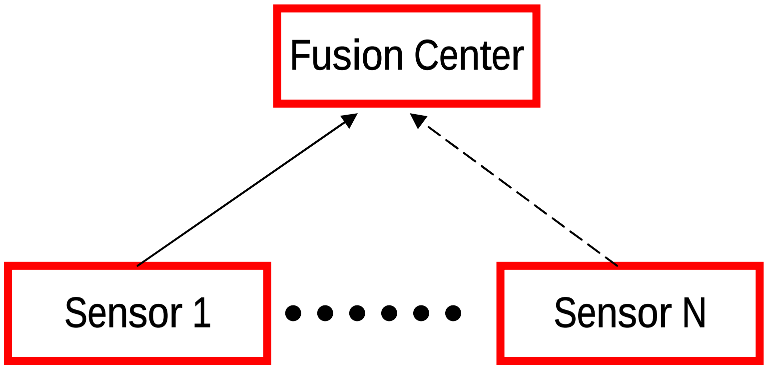
- Transmit measurements
  - Durrant-Whyte et al (90's, 00's)
    - Distributed robotics
    - Transmit weighted measurements
    - Information form estimation
    - No message losses
    - Extend to network topologies
  - Sinopoli et al ('04), others …
    - Sensors transmit measurements with Bernoulli message losses
    - Kalman Filter with random observation matrices
    - Conditions for existence of average steady state error covariance



- Limitations:
  - Message losses ➔ information loss
  - Synchronous model
  - Lack of local processing, multi-rate communications

Transmitting Estimates vs. Raw Data

- Example:
  - N Identical sensors
  - zero process noise
    - $x_t = Ax_{t-1}$
  - Packet loss: Bernoulli, prob. Of successful comms q = 1/N



- Raw data vs. Transmit Local MMSE Estimates
  - Result (asymptotically)
    - $MMSE_{localest} = N^{-1}\ MMSE_{raw} = N^{-1}\ MMSE_{single}$

$$E(X_t \mid (Y_1)^{\tau}, (Y_2)^{\sigma}, \ldots (Y_N)^{\gamma})$$

Data Length ~ NT
As though N sensors work

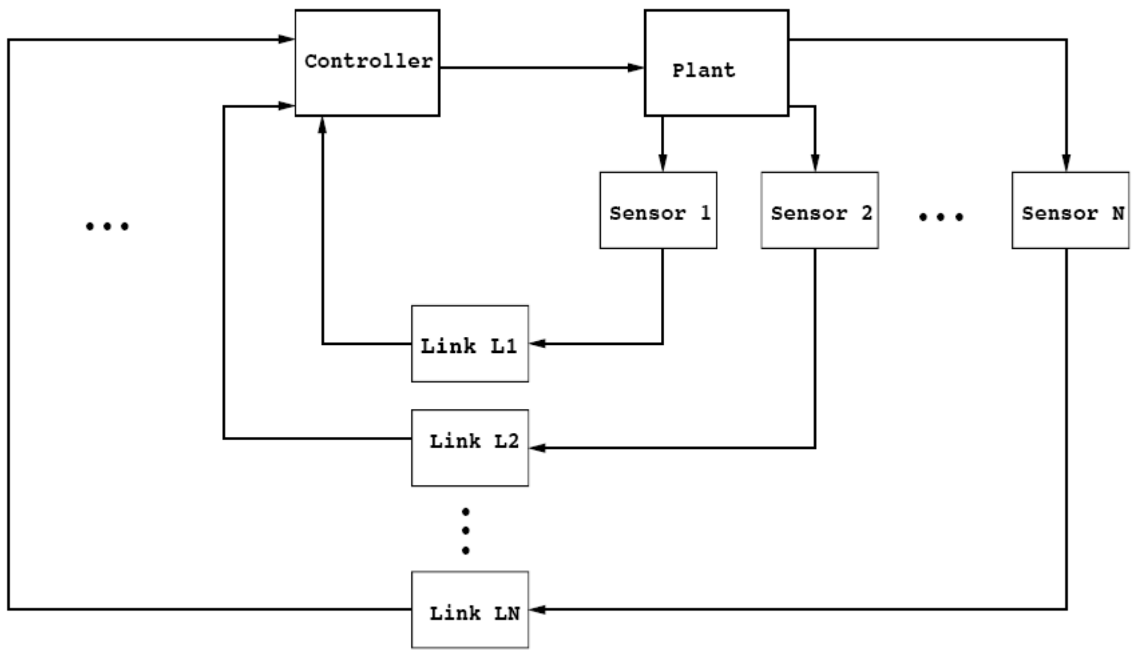
Only one sensor on average
Data Length ~ qTN = T
Same as data from one sensor

$$E(X_t \mid Y^t)$$

One reliable sensor

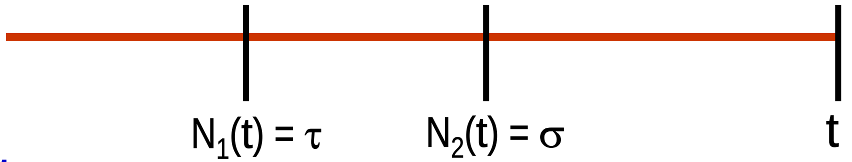
Can one exploit local processing?

- Raw measurements vs processed measurements
- Model: DistriubeGupta et al (04), others
  - Restrictions: At most 2 sensors
  - If 2 sensors, at least 1 sensor totally reliable, other sensor can drop measurements
  - Markov drop model (generalizes Bernoulli)



Objective: Anytime Optimality

- $N(t) = (N_1(t), N_2(t), N_3(t)) = (\tau,\sigma,\lambda, \dots \gamma)$
  - Last successful comms times@sensor k < t



- Best achievable centralized estimate
  - As though all data up to N(t) arrived from different sensors at fusion center

$$E(X_t \mid (Y_1)^\tau, (Y_2)^\sigma, \dots (Y_N)^\gamma)$$

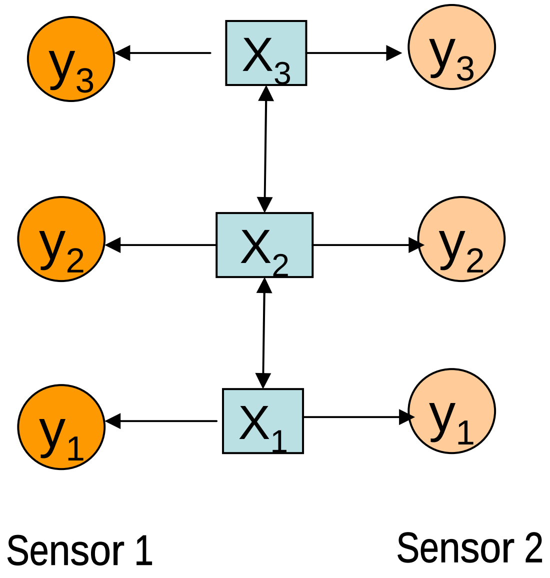
- Problem
  - Achieve anytime optimality under asynchronous messaging
  - Possible in zero process noise case
  - How about general case?
  - Channel noise, Coding/quantization??

Main Idea

- Conditional Independence (zero process noise)

$$y_1^t \longleftrightarrow x_t \longleftrightarrow y_2^t$$

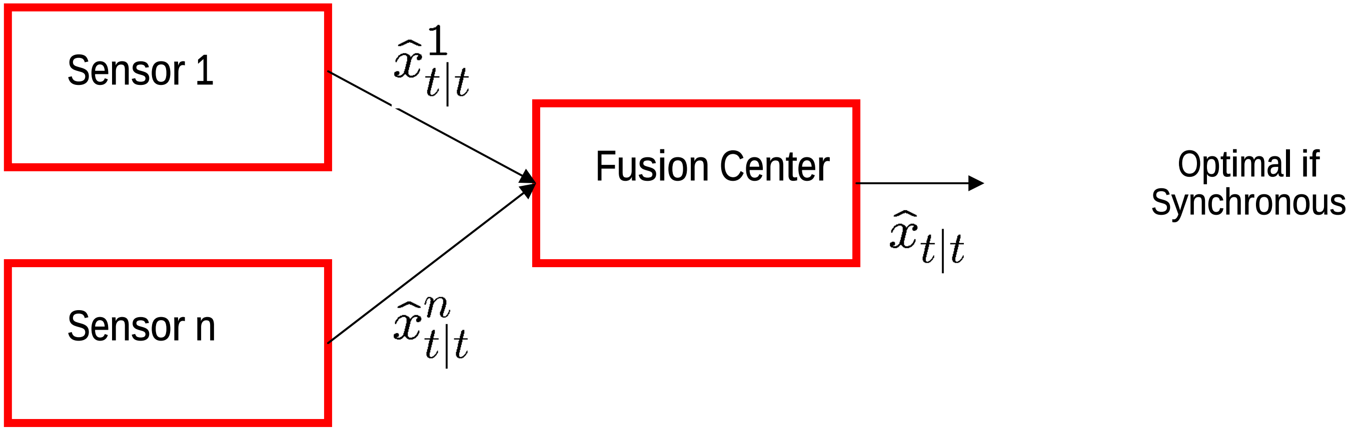
  - Global MMSE = Weighted sum of Local MMSEs
  - Single local estimate is summary of all the past data



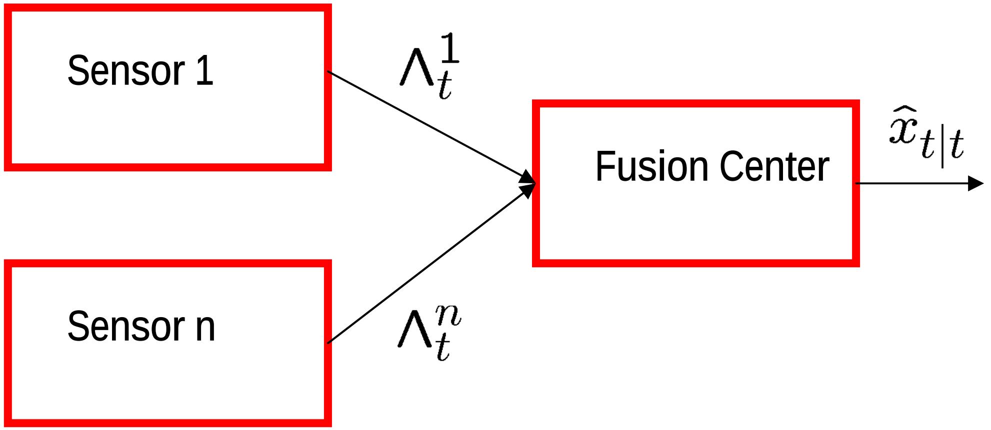
- General Case
  - Fusion of Correlated Info over space/time
    - No longer conditionally independent
  - Conventional Solution (Bar-Shalom 1981):
    - Fuse Local State Estimates
    - Sub-optimal
  - Must also handle asynchronous arrivals

Control of Encoder

- Synchronous distributed estimation paradigm: fixed encoder, optimize decoder



- Unreliable Communications: optimize encoder/decoder pair



General Case: Distributed Processing

- Distributed filter (Information Form)

$$(C^k)^T (R^k)^{-1} y_t^k$$

Current info = Past info + New info

$$\Sigma_{t|t}^{-1}\widehat{x}_{t|t} = \Sigma_{t|t-1}^{-1}\widehat{x}_{t|t-1} + \sum_{k=1}^{n} i_t^k$$

$$= \Sigma_{t|t-1}^{-1} A \Sigma_{t-1|t-1}(\Sigma_{t-1|t-1}^{-1}\widehat{x}_{t-1|t-1}) + \sum_{k=1}^{n} i_t^k$$

  - Linear System ➔ Superposition

- Leads to a simpler recursion for local statistics:

$$\Sigma_{t|t}^{-1}\widehat{x}_{t|t} = \Sigma_{t|t-1}^{-1}\widehat{x}_{t|t-1} + \sum_{k=1}^{n} i_t^k$$

$$= \sum_{k=1}^{n} \Lambda_t^k$$

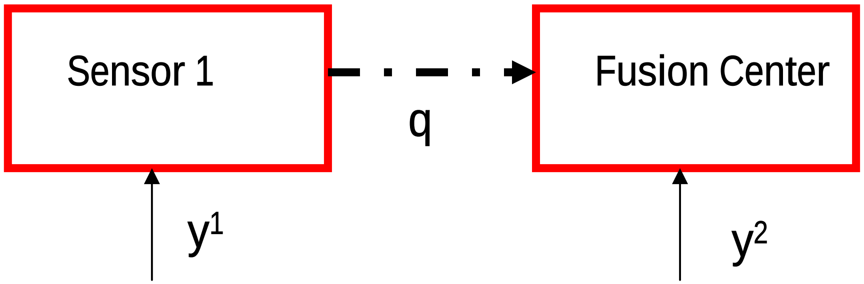
Fusion Rule

$$\Lambda_{t+1}^k = \Sigma_{t+1|t}^{-1} A \Sigma_{t|t} \Lambda_t^k + i_{t+1}^k$$

Observation: Computing local statistics depends on global covariance

Case 1: Single Remote Sensor

- Setup



  - Unreliable link between Sensor 1 and fusion center: Bernoulli model, success probability q

- Local Processing at both fusion center and sensor:

$$\Lambda_{t+1}^{k} = \Sigma_{t+1|t}^{-1} A \Sigma_{t|t} \Lambda_{t}^{k} + (C^{k})^{T} (R^{k})^{-1} y_{t}^{k}, \quad , k = 1, 2$$

- Messages: Sensor 1 sends at time t: Λ_{t}^{1}

Case 1: Single Remote Sensor Contd

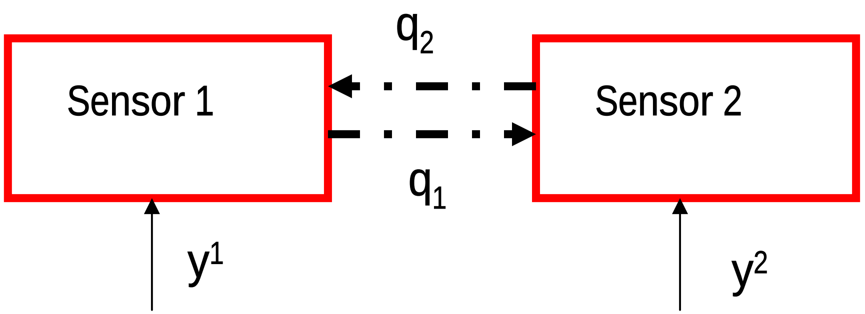
- **Fusion Center Processing**
  - If message received, generate new estimate as

$$\widehat{x}_{t|t} = \Sigma_{t|t}(\Lambda_t^1 + \Lambda_t^2)$$

  - If message is not received, generate new estimate
    - KF equations using only y^2_t using last available centralized estimate

- **Anytime Optimality:**
  - Quick proof: True for $t = \tau$ by previous results;
  - Extend by induction to $t > \tau$
- **Main observation:**
  - Whenever a message arrives at fusion center at t, it is to be fused with history of fusion center data up to time t

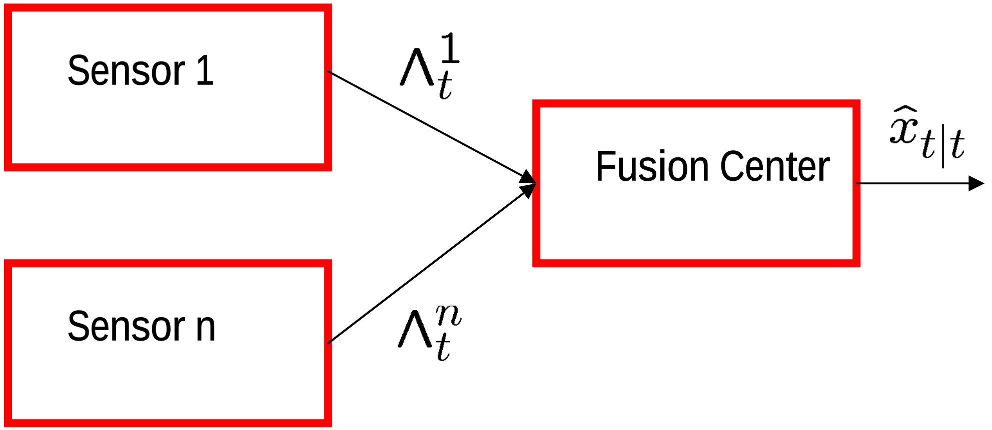
Case 2: Two communicating sensors



- Algorithm: duplicate previous algorithm for fusion center processing in each sensor
- Result:
  - Anytime Optimality Achievable
  - Each sensor has centralized estimate based on own information up to current time, plus other sensor's information up to last communication time

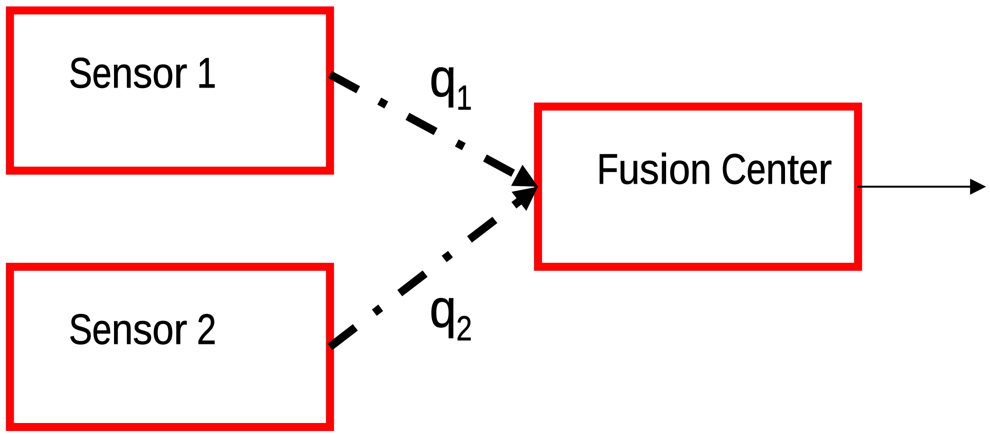
Case 3: N identical Sensors

- N identical sensors with collision
  - Either 0 get through or K gets through
- Result:
  - Process as if only K sensors get through (i.e. use covariances based on K sensors)
  - Decentralized Scheme MMSE = Centralized MMSE with K sensors
  - Local statistics do not have to be communicated at each time t
- Drawbacks
  - Not anytime optimal
    - ignores previous estimates
  - Global Statistical knowledge
  - Identical sensors restrictive

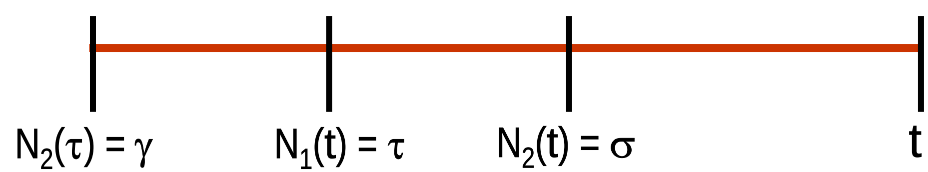


Case 4: More than 2 sensors?

- Simple case: 2 distinct sensors and a fusion center



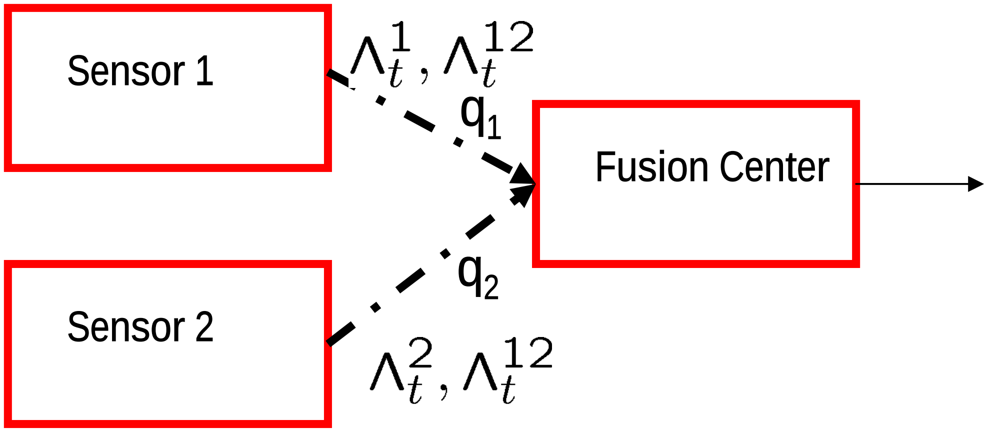
- Anytime Optimality ??
  - Requires feedback
    - Arrival times



  - Conflicting requirements:
    - For time τ sensor 1 → accounts for sensor 2 data available only up to γ
    - But for time σ → sensor 1 needs to assume data from Sensor 2 available up to τ

Anytime Optimality: Explosion in Complexity

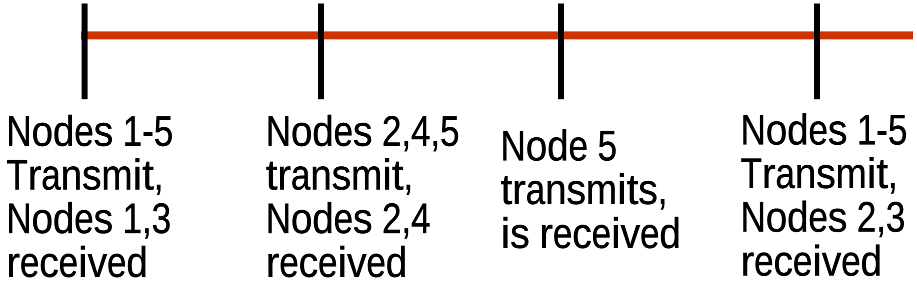
- Same architecture



- Assumption: successful communications are observed by all sensors
- Result: Can be recovered with $2^{n-1} - 1$ local statistics…
  - Tuned to different combinations of successful communications at a specific time by different sensors

Relaxed Optimality Concept

- Give up on anytime optimality: Sampled optimality
  - Optimality properties achieved when sensor k successfully communicates
- Assumption: Global feedback available on successful communications
- Protocol: Once nodes successfully communicate, they stop transmission until other nodes successfully communicate



- Relaxed optimality: Recover best estimate of x(t) after all nodes are received, given local data up to times of successful transmission
  - Less data than available if continued transmission

Relaxed Optimality

- Local Processing: Two statistics
  - One as before

$$\Lambda_{t+1}^k = \Sigma_{t+1|t}^{-1} A \Sigma_{t|t} \Lambda_t^k + (C^k)^T (R^k)^{-1} y_t^k$$

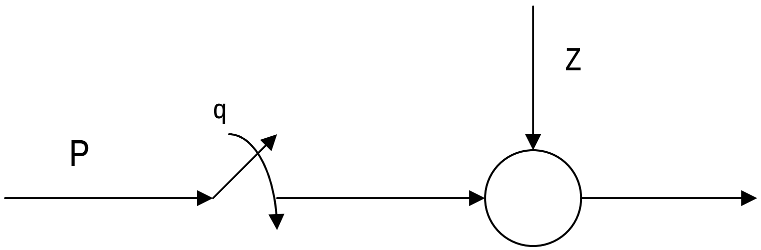
  - Second one as if sensors that have successfully transmitted in current round take no further measurements from time of transmission
    - Changes matrices in processing

- Result: Optimal estimates $E(X_t \mid (Y_1)^\tau, (Y_2)^\sigma, \ldots (Y_N)^\gamma)$ achieved after each round of successful transmissions by all sensors

Is feedback necessary for anytime optimality?

- Setup
  - Assume no feedback on successful communications
  - No statistics on when sensor drop out, i.e., do not transmit
  - Simplify to case where only one sensor at any time can transmit

- Model: AWGN channel + Erasures
  - Power constraint



- Problem
  - Find best worst-case (with respect to arrival times) mean square estimator

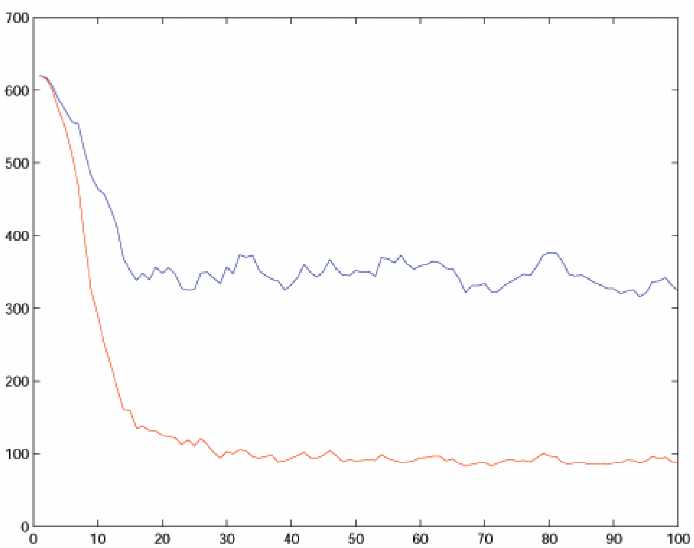
- Conjecture:
  - Optimal to encode local state estimate ~ conventional suboptimal case

Experiments

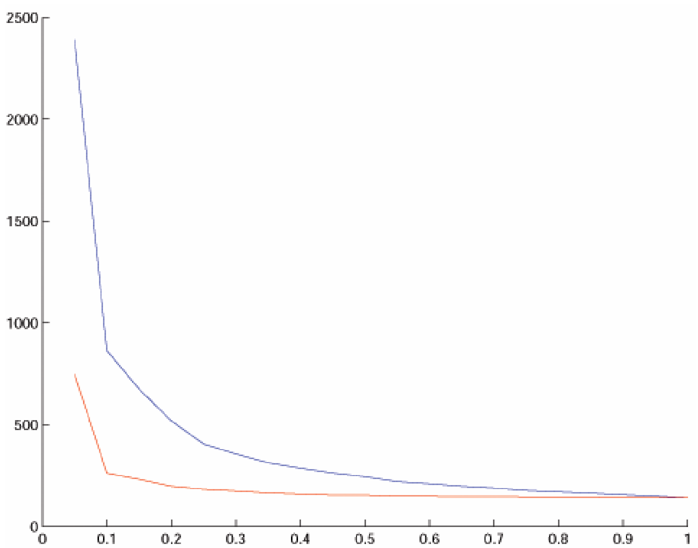
- Model: Tracking in 2 D, 4 states (position-velocity)
  - Linear position measurements
- Two sensors, one fusion center without measurements
- Two cases: unobservable and identical
  - Unobservable: each sensor has unobserved subspace
  - Identical: Model is observable under each sensor
- Two communications to fusion center
  - One always gets through
  - Bernoulli channels with identical failure rates

Unobservable subspaces

- Blue is error if messages were only measurements, Red is error with our local protocols



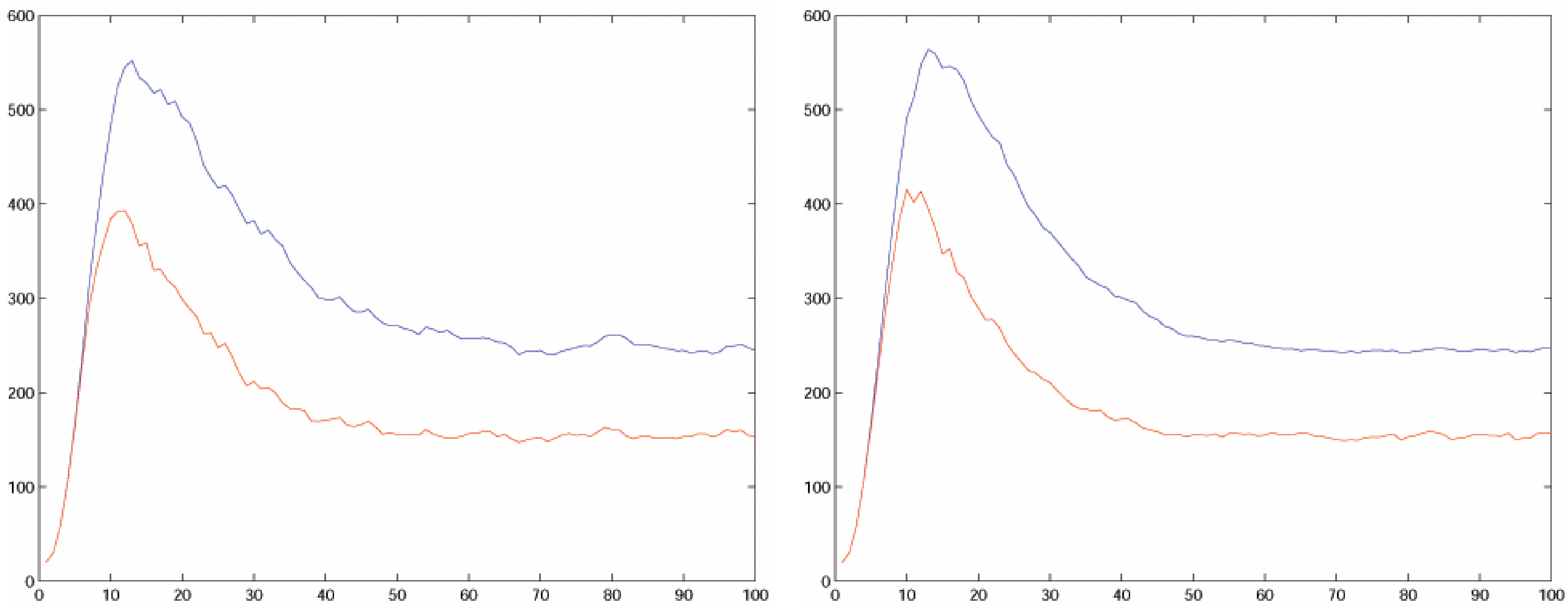
error covariance vs time
Bernoulli 0.5 channels



Steady State Expected Error Covariance vs Probability of Successful Comms

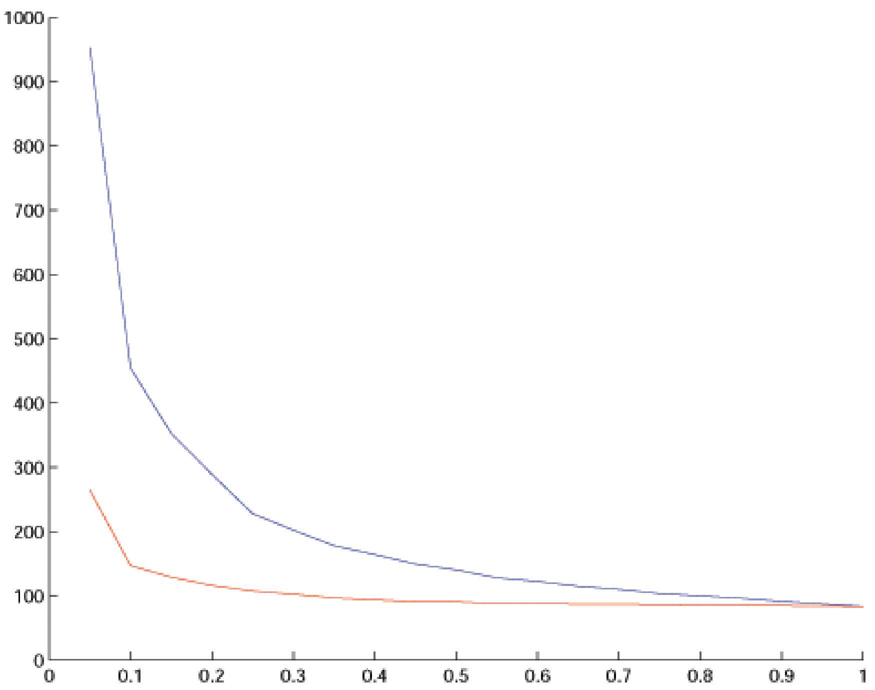
Unobservable subspaces

- Trace of average error covariance vs time for case with Bernoulli 0.5 channels vs one always channels
- Blue is error if messages were only measurements, Red is error with our local protocols

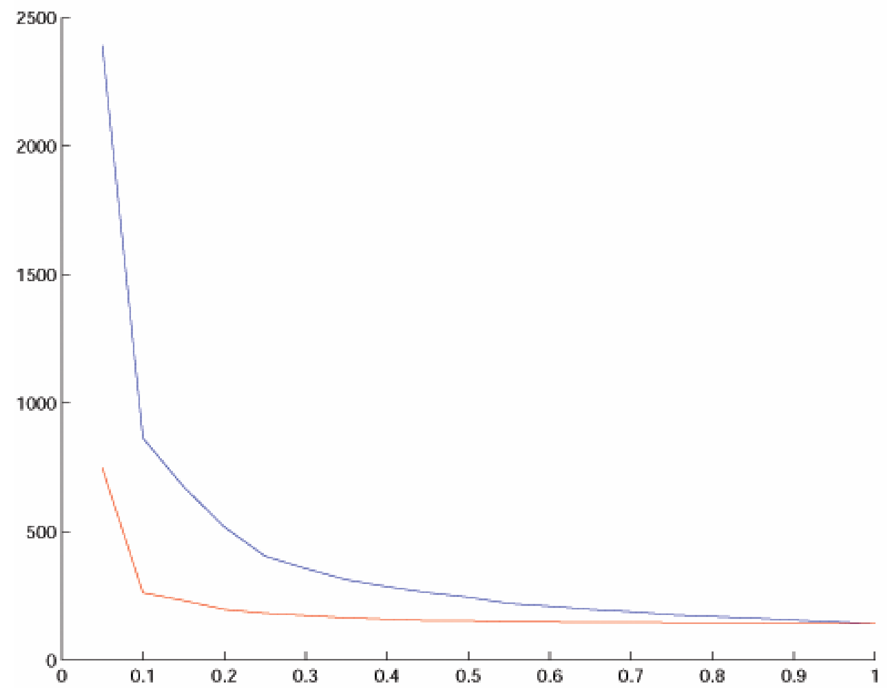


Observable vs Unobservable Local Estimates

Trace of Steady State Expected Error Covariance vs Probability of Successful Comms



Identical Sensors (Bernoulli)



Opposite Observability (Bernoulli)

Conclusions

- New problems created by asynchronous message losses
  - Need to observe message losses
  - Must incur additional computation, communication to compensate for lack of synchronicity
- Many current protocols do not scale for anytime optimality
  - Optimal, but with computation growing exponentially with sensors
  - Can recover scalability and optimality with coordinated transmission policies, but requires feedback to sensors
- Alternatives to be investigated involve different tradeoffs between local and central processing
  - Feedback architectures, tracklets, incremental processing…
  - Balance between local and global processing
- Future Work
  - Channel Noise, Coding, Quantization
    - Isolated encoder/decoder
    - ``Optimal Scaling'' for Large no. of sensors ….

Conclusions

- Recent work in distributed estimation on sensor networks revisits old topics
  - Local vs Fusion processing
  - Decentralized statistics
  - Distribution of centralized processing using local and central computation
- New problems created by asynchronous message losses
  - Need to observe message losses
  - Must incur additional computation, communication to compensate for lack of synchronicity
- Current protocols do not scale well with number of sensors
  - Optimal, but at a cost
- Alternatives to be investigated involve different tradeoffs between local and central processing
  - Feedback architectures, tracklets, incremental processing