F.L. Lewis
Moncrief-O’Donnell Endowed Chair
Head, Controls & Sensors Group

Automation & Robotics Research Institute (ARRI)
The University of Texas at Arlington


Talk available online at http://ARRI.uta.edu/acs

F.L. Lewis
Moncrief-O’Donnell Endowed Chair
Head, Controls & Sensors Group

Automation & Robotics Research Institute (ARRI)
The University of Texas at Arlington
Wireless Sensor Networks

Data Acquisition Network

- Machine Monitoring
- Animal Monitoring
- Ship Monitoring

BST (Base Station Controller, Preprocessing)

Medical Monitoring

Wireless Data Collection Networks

BSC (Base Station Controller, Preprocessing)

Data Distribution Network

- Roving Human monitor
- PDA
- Online monitoring
- Printer
- transmitter
- Server

Wireless (Wi-Fi 802.11 2.4GHz, BlueTooth, Cellular Network, - CDMA, GSM)

Wireland (Ethernet WLAN, Optical)

Any where, any time to access

- Notebook
- Cellular Phone
- PC

Medical Monitoring

Wireless Sensor
Applications

- Wide area monitoring for personnel / vehicles
- Secure area intrusion monitoring and denial
- Environmental monitoring
  - animal habitats
  - migration
  - forest fires
  - natural disasters
- Subsea monitoring
- Environmental toxin detection
- Building monitoring
- Urban area environmental monitoring
  - sensors on buildings
  - sensors in taxis or buses
- Vehicle traffic monitoring & control
  - sensors on roadways and traffic lights
  - sensors on vehicles
- Remote site power substation monitoring
- Remote site patient medical monitoring
- Smart home
- Inventory management

Metrics / QoS

- Latency (delay)
- Energy efficiency
- Accuracy
- Fault-tolerance
- Scalability
- Security
WSN Issues

- **Limited**
  - Range
  - Power
  - Processing power / memory
  - Cost
- **Large number of nodes**
- **Prone to failures**
- **Easy to be compromised**
- **Changing topology**
- **Lack of global ID**
- **Long-term reliability**

**Disseminate**
- Sensor data
- Information
- User Interest

**Sensor Management Protocol (SMP)**
- Attribute-based naming
- Location-based addressing
- Data-centric routing
- User broadcast interest

---

Figure courtesy Akyildiz, Su, et al. 2002
**Deploy**

- Self-Organize
- Communication
- Localization
- Form clusters

Random vs. Structured topology

---

**Operate**

1. Program Missions
2. Accomplish Missions

**Monitoring**

- continuous
- event-based query

---

**Reconfigure**

- Post-deployment
- Redeployment of new nodes
- Fault recovery

---

**Mobility**

- user / observer
- sensors
- phenomena / target

---

**Changing Topology**

- mobile nodes
- event occurrence
- mobile target / phenomena
- changing user queries/interests

---

- node failure
- deploy additional nodes

---

- Agent interactions
- Bayesian, Uncertainty, Dempster-Shafer, and Utility Theory
- FSM
- Requirements
- Applications
- Event/ Context
- Algorithms and Techniques
- Change/Modify
- Sensor states
- Actions
- Sensors
Network layer

Routing
- Minimum energy
- Minimum hop
- Max. min power available

Interest dissemination
Responsive action
Event detection

Routing

Data fusion

SPIN protocol

Sensor Protocols for Information via Negotiation

Akyildiniz, Su, et al. 2002
Hierarchical Routing Allows Multicast – Efficient Routing

- **source node**
- **destination**
- **group leader**

1. Source to leader
2. Leader to destination

Standard peer-to-peer routing

Multicast routing

Taken from Chen et al. (2000)

Election of cluster heads
- Event-based
- Application-based
- LEACH
Research Topics

**Deploy**
- Self-organization
- Comms. wakeup
- Localization

**Decision-making & control**
- Programmable WSN
- Program missions quickly
- Task scheduling
- Dynamic resource assignment

**Sensing**
- Event detection
- Interpret data
- Responsive action
- User broadcast interest
- Respond to queries

**Energy conservation**
- Fault tolerance
  - node failure
  - link failure
- Security

**Communications**
- Dynamically reconfigurable
- Event-based routing

**Cooperation**
- Dynamic Clustering
  - Data Transmission
    - Event-based
    - Data aggregation
    - Sensor Data fusion
    - Information fusion
    - Decision fusion

**Fault**
- node failure
- link failure

**Use Mobility to**
- Localize nodes
- Maintain connectivity
- Optimize comms.
- Optimize sensor coverage
- Reduce measurement uncertainty

**Meet QoS requirements**
- Communications
- Sensing
- High priority data

**Scalability - NP complexity**
- Distributed local algorithms vs. global

**Lack of testbeds**
OSI- Open Systems Interconnection Protocol Stack
Cross-layer design

e.g. Integrate navigation, communication, congestion control, and sensing
IEEE 1451 Standard for Smart Sensor Networks

Concept of Smart Sensor
contains functions in addition to those needed for accurate presentation of the measurand
Smart Transducer Interface Module (STIM)

- XDCR → ADC
- XDCR → DAC
- XDCR → Dig. I/O
- XDCR → ?

Transducer Electronic Data Sheet (TEDS)

Address logic

Network Capable Application Processor (NCAP)

1451.1 Object Model

1451.2 Interface

Transducer Independent Interface (TII)

Network (NETWORK)
Failure of two nodes causes loss of sensor coverage

**Reliability Theory**

Survivor Function = prob. that a unit is still functioning at time $t$

$s(t) = 1 - \text{cdf}$

Reliability block diagram of square grid

$s_{block} = 1 - (1 - s_a)(1 - s_b s_c)$

$s_{net} = (s_{block})^N$

Reliability block diagram of hex grid

$s_{block} = 1 - (1 - s)(1 - s^3)$

$s_{net} = (s_{block})^{N/2}$
Finding node lifetime pdf

Node lifetime

\[ T_{\text{node}} = \frac{E_{\text{init}} - E_{\text{threshold}}}{\sum_i w_i P_i} \]

- Power consumed in mode \( i \)
- Fraction of time spent in mode \( i \)

Assume only 2 modes, then binomial pdf

\[ P\{w_1 = x\} = c^x p^x (1 - p)^{T-x} \]

- Pr node is idle
- Pr node is active (defined by net protocol)

\( T = \text{nr. of time units} \)

Results

Fig. 4. Probability Density Function of the lifetime of a node. (a) Theoretical pdf, (b) Actual pdf.

Fig. 5. Probability Density Function of the network lifetime employing the square-grid placement scheme. (a) Theoretical pdf, (b) Actual pdf.
Energy Conserving Sensor Coverage

Choi and S. Das, Mar 2005

Sample time #1

Grey = area not covered

Selected Sensors

Extra nodes selected for connectivity

Entire area covered in 2 sample times
latency (delay) = 2

Formal algorithms for specifying QoS
% coverage of sensors
max latency
Math Basis

Circular sensing region $SR_i$ of radius $r$

Select min. nr. $K$ of sensors s.t.

$$DSC \subseteq Q \cap \bigcup_{i=1}^{k} SR_i$$

and entire region is covered within desired latency $T$

$$\sum_{i=1}^{N} t_i \leq T$$

Assume:

- sensors are uniformly distributed
- location info not available

1. Find Required Number of Sensors for DSC

$DSC =$ probability of coverage of point $(x,y)$

Find probability that a point $(x,y)$ is not covered by randomly selected sensor $P_{q} q(x,y)$

Then, min. number of sensors needed to cover DSC is

$$k = \frac{\log(1 - DSC)}{\log \left( \frac{a^2 + 4ar}{a^2 + 4ar + \pi^2} \right)}$$

Algorithm `ConstructRS(k, |V|)` Begin

1. $\delta \leftarrow \left\lceil \frac{|V| - 1}{b} \right\rceil$; /* the number of reporting rounds in $C$ */
2. Allocate a bit array $A[\delta]$ and initialize all the entries with zero;
3. $i \leftarrow RAND[1, \delta]$;
4. $A[i] \leftarrow 1$;
5. return $A[\delta]$; /* reporting sequence $RS_{x_i}$ */

End-Algorithm

Fig. 5. Algorithm for Constructing Reporting Sequence ($RS_{x_i}$)
2. Add Extra Routing Nodes for Comm. Connectivity

Test probable connectivity of \( k \) sensors in \( k-1 \) steps, adding nodes when needed

\[
P_{S_i} = \text{radio range of node } i / \Lambda
\]

\[
P_{r_0} = \sum_{j=1}^{k-1} \binom{k-1}{j} (P_{s_j})^j (1 - P_{s_j})^{k-1-j}
\]

\[
P_{r_1} = P_{r_0} \times \left( 1 - \left( 1 - \sum_{j=1}^{k-2} \binom{k-2}{j} (P_{s_j})^j (1 - P_{s_j})^{k-2-j} \right)^2 \right)
\]

\[
P_{r_2} = P_{r_1} \times \left( 1 - \left( 1 - \sum_{j=1}^{k-3} \binom{k-3}{j} (P_{s_j})^j (1 - P_{s_j})^{k-3-j} \right)^2 \right)
\]

\[
P_{r_{k-2}} = P_{r_{k-3}} \times \left( 1 - \left( 1 - \sum_{j=1}^{k-4} \binom{k-4}{j} (P_{s_j})^j (1 - P_{s_j})^{k-4-j} \right)^2 \right)
\]

\[
P_{r_{k-1}} = P_{r_{k-2}} \times (1 - P_{s_k})^{k-1}
\]

Fig. 7. Probabilistic Model for Estimating The Connectivity of Selected \( k \) Sensors

3. Construct Data Gathering Tree (DGT)

For routing and sensor scheduling

Data sink sends flood message
Each sensor keeps a forwarding record with best upstream candidate
Sensors broadcast join request setup msgs.

Localized sensor scheduling algorithm

![Algorithm](image)
Distributed Greedy Algorithm for Connected Sensor Cover

Energy conserving sensor coverage
Find minimum connected sensor cover (MCSC)

Def. MCSC
1. Monitored area contained in Union of node sensor regions
2. Induced communication graph is connected via multihop

Problem of finding MCSC is NP-hard [Garey and Johnson 1991]

Graph = (nodes, edges)

Communication radius $R_c$
Sensing radius $R_s$

Assume $R_c \geq 2R_s$

<table>
<thead>
<tr>
<th>Induced comm. graph $G_c = (V, E_{R_c})$</th>
<th>Induced sensing graph $G_s = (V, E_{R_s})$</th>
</tr>
</thead>
<tbody>
<tr>
<td>edge $i,j$ exists if $d(s_i, s_j) &lt; R_c$</td>
<td>edge $i,j$ exists if $d(s_i, s_j) &lt; 2R_s$</td>
</tr>
</tbody>
</table>

Def. Independent Set
A subset of vertices such that no two vertices has an edge in G.

Def. MIS
An IS that is not contained in any other IS

Finding MIS for a general graph is NP-hard

Ghosh and S. Das, June 2005

Fig. 1. (a) Set of black vertices forms a maximum IS; set of white vertices forms a MIS but not a maximum. (b) Induced sensing graph of 10 nodes. The set of 5 darkened nodes forms a MIS.
Suboptimal MCSC using greedy approach

Phase 1 – Find Maximal Independent Set (MIS)

Use greedy approach looking only at 1-hop nearest neighbors

Def. Eligible next node given node $s_i$
1. $s_j$ not yet included in the connected MIS
2. $s_j$ a one-hop neighbor of $s_i$
3. sensing circle of $s_j$ does not overlap any selected sensing circles

**Algorithm 1** Phase 1: Distributed greedy algorithm to find a connected MIS.

```plaintext
1: Initialization:
2: $\Gamma \leftarrow \phi$;
3: Choose the first node $s_0$ and include it in $\Gamma$; $s_b \leftarrow s_0$;
4: Steps at each $s_b$:
5: $N_{sb}(R_c - 2R_s) \leftarrow \phi$;
6: for all $s_k \in N_{sb}(R_c)$ do
7:    if $2R_s \leq d(s_b, s_k) \leq R_c$ then
8:       $N_{sb}(R_c - 2R_s) \leftarrow N_{sb}(R_c - 2R_s) \cup s_k$;
9:    end if
10: end for
11: if $N_{sb}(R_c - 2R_s) \neq \phi$ then
12:    Find that $s_k \in N_{sb}(R_c - 2R_s) \setminus N_{sb}(R_c - 2R_s) \cap \bigcup_{s_j \in \Gamma, s_j \neq s_b} N_{sj}(2R_s)$, such that $d(s_b, s_k)$ is minimum;
13:    $\Gamma \leftarrow \Gamma \cup s_k$;
14: else if $N_{sb}(R_c - 2R_s) = \phi$ then
15:    $s_b \leftarrow s_q$, such that $d(s_b, s_q) = \max \{d(s_b, s_i), \forall s_i \in N_{sb}(R_c)\}$;
16: end if
17: $s_k$ becomes the next $s_b$ to execute the same steps 5 – 16.
```
Phase 2- Select extra nodes to get full sensor coverage

Result of Phase I MIS

Fig. 3. (a) The induced sensing graph and the total coverage (shaded area) achieved by the four nodes \((s_0, s_1, s_2, s_3)\), selected in phase 1. (b) Best Eligibility Criteria for a set of nodes deployed randomly.

Construct Voronoi Diagram for nodes selected in Phase 1

Voronoi Diagram divides the plane into convex polygons whose edges are equidistant from two nodes

Fig. 2. Voronoi diagram for a set of randomly deployed points in 2-D. \(V_k(p_j)\) denotes a Voronoi vertex.

Algorithm 2- Choose best 1-hop neighbor that maximally covers holes in its polygon

Voronoi structure allows efficient formal algorithm for doing this
**Math Analysis**

Let $N$ nodes be uniformly randomly distributed over area $A$. Density is $\rho = N / A$. Then number of nodes in Phase I MIS is bounded by

$$\frac{N}{5\rho \pi R_s^2} \leq \zeta \leq \frac{N}{\rho \pi R_s^2}$$

Time complexity of first algorithm is $O(\zeta N)$

Time complexity of second algorithm is $O(\zeta \log \zeta)$

---

**Fig. 5.** Simulation: Dark circles represent sensing ranges of nodes belonging to connected MIS: (a) $N = 150$, $R = 15m$, $A_Q = 10,000m^2$. (b) Variation of connected MIS cardinality with sensing radius $R_s$. 
1. (n,t) Threshold cryptography via polynomials

Random secret polynomial \( f(x) = a_0 + a_1x + a_{t-1}x^{t-1} \) where secret key is \( D = f(0) \)

\( f(x) \) can be reconstructed from \( t \) points from the set \( \{f(1), f(2), \ldots, f(n)\} \), with \( n \) = number of nodes

Select masking polynomial \( h(x) \) and securely predeploy personal secrets \( h(i) \) on each node \( i \).

The Sink broadcasts \( w(x) = f(x)g(x) + h(x) \)

Where revocation polynomial is \( g(x) = (x-r_1)(x-r_2)\ldots(x-r_w) \)

with the set of compromised nodes \( \{r_1, \ldots, r_w\} \) which has been broadcast to all nodes

Then each node \( i \) can evaluate its personal key \( f(i) = \frac{w(i) - h(i)}{g(i)} \)

Compromised nodes have \( g(i) = 0 \) and cannot find personal key

Now, \( t \) nodes can collaborate to exchange personal keys \( f(i) \) and so compute \( f(x) \), and hence find the secret group key \( D = f(0) \)

Since \( h(i) \) is securely predeployed and \( f(x) \) is random, the scheme can be shown to be unconditionally secure
2. Enhancement to avoid disclosing personal key \( f(i) \)

Select a random concealing polynomial \( L(x) \) of degree \( t-1 \) and securely predeploy concealing secret \( L(i) \) on each node \( i \)

Instead of exchanging personal keys \( f(i) \), the \( t \) collaborating nodes exchange the concealed personal key \( s(i) = f(i) + L(i) \)

Since \( s(x) \) has degree \( t-1 \), it can be reconstructed using \( t \) concealed personal keys from \( t \) nodes

Then, the group secret is \( D = s(0) \)

**Theorem 1.** Assume that the local exchange of concealed secrets is secure. Then the scheme is unconditionally secure, and has \( t \)-revocation capability.

The concealing key allows improved defense against compromised nodes, who now do not know the personal keys \( f(i) \)

This also allows self-healing strategies, i.e. in the presence of lost broadcast messages from the Sink
Multi-Layer Approach to Defense Against Compromised Nodes

Y. Liu and S. Das

Cross-Layer Design

Figure 1: Multiple Layer Defense against Compromised Nodes
Distributed Energy-Efficient Self-Organization

LEACH selects cluster heads based on energy available
  Selects randomly and does not give evenly spaced heads
  Requires global info – total number of nodes, and total energy available in all nodes

Expellant Self-Organization (ESO)

Based on cluster radius $R_c$, energy available, and number of neighbors

Energy threshold
$$E_{th} = \gamma_1 \max_{i \in N} E(i) + (1 - \gamma_1) \, \text{mean} \, E(i)$$

$E(i)$= energy of node $i$
$N$= set of neighbors

Number of neighbors threshold
$$n_{th} = \gamma_1 \max_{i \in N} n(i) + (1 - \gamma_1) \, \text{mean} \, n(i)$$

$n(i)$= number of neighbors of node $i$

Fig. 3. Flow chart of a node in ESO.
Use fuzzy logic to select node for next hop transmission using:
1. Distance of next node (NN) to destination- should be small
2. Remaining battery capacity of next node- should be large
3. Mobility of next node- should be small

**Rules:** If \((NN \text{ is near dest.})\) and \((NN \text{ has large remaining energy})\) and \((NN \text{ is stationary})\)

THEN \((NN \text{ is a strong candidate})\)

The required information is periodically locally broadcast via beacons.

When a node wishes to transmit, it sends a ROUTE NOTIFICATION,
and the receiving nodes send a REPLY packet.

If a node fails, the previous node broadcasts a ROUTE DELETION packet.
Cross-Layer Routing in Sensor Networks

Luo, Yonghe Liu, Sajal Das, 2006

Graph (nodes, edges) \( G = (V,E) \)

Find Data Gathering Tree

**Transmission cost** \( t(e) = w(e) \cdot c(e), \) \( e = \) an edge

- \( w(e) = \) amount of data
- \( c(e) = \) transmission cost – congestion, distance, latency, energy used

**Fusion cost** \( f(e) \)

Minimize the total cost

\[
\sum_{e \in \text{route}} [f(e) + t(e)]
\]

**Fusion Cost** for \( L \) bits = \( 2L x 5 \) nano Joules

the resulting data has \( L(1+\eta) \) bits

Data correlation model \( \eta = 1 - e^{-\alpha r} \)

- \( r = \) node separation, good for field measurements

Fonda, Zawodniok, S. Jagannathan, ISIC Munich 06

Minimize the cost with link cost factors given as

\[
\frac{\text{remaining energy at next node}}{(\text{delay to reach next node}) \times (\text{dist. from next node to destination})}
\]

*Fijkstra’s Algorithm can be used*
Transmission Costs

Transmission cost per hop \( \propto d^\gamma, \ 2 \leq \gamma \leq 4 \)

Total energy per packet using N hops

\[
E = \alpha N + \beta \sum_{hops} d_i^\gamma
\]

- Setup cost
- Trans. cost
For data representing the same property:

General distance measure

\[ d(P_1, P_2) = f \{ E_1 \{ C(L) \} \} \]

\[ f(.) \text{ an increasing fn.} \]
\[ C(.) \text{ a convex fn.} \]

J-divergence

\[ J(P_1, P_2) = E_1 \{ (L - 1) \log L \} \]

Matsusita distance

\[ d(P_1, P_2) = \{ E_1 \{ (\sqrt{L} - 1)^2 \} \}^{1/2} \]

Bhattacharyya’s distance

\[ B(P_1, P_2) = - \log(1 - d^2) \]
Distance is not symmetric

For multiple sensors, use matrix

\[
D = \begin{bmatrix}
  d_{11} & d_{12} & \cdots & d_{1m} \\
  d_{21} & d_{22} & \cdots & \vdots \\
  \vdots & \vdots & \ddots & \vdots \\
  d_{m1} & \cdots & d_{mm}
\end{bmatrix}
\]

Draw a digraph having edge \((i,j)\) if 

\[d_{ij} \leq \text{threshold}\]

Luo, Lin, Scherp 1988
Information Fusion & Sensor Selection in WSN

Select the best set of sensors that meet the goals of the applications with guaranteed QoS
Middleware for customized services and resource assignment

Use Bayesian Networks

Expected utility given evidence

\[ E\ U(\{S_n\}) = \sum_{i} P(G_i(\{S_n\})|\{S_n\}) U(G_i(\{S_n\})) \]

\( G_i(\{S_n\}) \) is the goal state reached as a result of the selected sensor set \( \{S_n\} \)

Utility of sensors can be written \( U(G_i(\{S_n\})) = u_n \)

Where utility factor for sensor \( s_n \) is

\[ u_n = \alpha D_n + (1 - \alpha) \frac{1}{\text{cost}(s_n)} \]

and \( D_n \) is a measure of the definitiveness or accuracy of sensor \( s_n \)
Security Threat Example

Utility is maximized by using sensor set \{RFID2, Video3\}
This gives 70% threshold

Table 1: Utility numbers for sensors in the example scenario

<table>
<thead>
<tr>
<th>Sensor</th>
<th>(D_i)</th>
<th>({Cost_i})</th>
<th>({u_i})</th>
</tr>
</thead>
<tbody>
<tr>
<td>RFID1</td>
<td>4</td>
<td>1</td>
<td>3.4</td>
</tr>
<tr>
<td>RFID2</td>
<td>19.85</td>
<td>2</td>
<td>15.98</td>
</tr>
<tr>
<td>Video1</td>
<td>2.5</td>
<td>3</td>
<td>2.066667</td>
</tr>
<tr>
<td>Video2</td>
<td>1.45</td>
<td>4</td>
<td>1.21</td>
</tr>
<tr>
<td>Video3</td>
<td>9.3</td>
<td>5</td>
<td>7.48</td>
</tr>
<tr>
<td>Biometric</td>
<td>3.95</td>
<td>6</td>
<td>3.193333</td>
</tr>
</tbody>
</table>
Adaptive Sampling with Mobile Sensor Nodes

Dan Popa, Sreenath, Mysorewala, F.L. Lewis
ICCA Budapest 2005

Mobile node dynamics

\[ x_{k+1} = x_k + h(x_k, u_k) + w_k \quad E[w_k w_k^T] = Q_k \]

Mobile node position measurement

\[ y_k = f(x_k) + \xi_k \quad E[\xi_k \xi_k^T] = R_k \]

Distributed field measurement model

\[ z_k = g(x_k, a_k) + \nu_k \]

Sum of Gaussian model

\[ z_k = a_o + a_1 g_1(x_k) + ... + a_m g_m(x_k) \quad \text{(RBF neural network)} \]

A. Estimation of Field Without Localization Uncertainty

\[ A_{k+1} = A_k \]

\[ z_k = G_k A_k + \nu_k, \quad G_k = \begin{pmatrix} 1 & \ldots & g_1(X_k) & \ldots & g_m(X_k) \end{pmatrix} \quad E[\nu_k \nu_k^T] = R \]

\[ A_o \sim (\bar{A}_o, P_o), P_o = P_{A_0} \]

\[ P_{k+1}^{-1} = P_k^{-1} + G_{k+1}^T R^{-1} G_{k+1}, \]

\[ A_{k+1} = A_k + P_{k+1} G_{k+1}^T R^{-1} (Z_{k+1} - G_{k+1} A_k) \]

B. Estimation of Field With Localization Uncertainty

\[
\begin{bmatrix}
X_{k+1} \\
A_{k+1}
\end{bmatrix} =
\begin{bmatrix}
X_k \\
A_k
\end{bmatrix} +
\begin{bmatrix}
I_3 \\
0
\end{bmatrix}
U_k +
\begin{bmatrix}
w_k \\
0
\end{bmatrix} =
\begin{bmatrix}
X_k \\
A_k
\end{bmatrix} + BU_k + \theta_k,
\]

\[
\begin{bmatrix}
Y_k \\
Z_k
\end{bmatrix} =
\begin{bmatrix}
1 & X_k^T
\end{bmatrix}
A_k +
\begin{bmatrix}
\xi_k \\
\nu_k
\end{bmatrix} =
\begin{bmatrix}
I_3 & 0
\end{bmatrix}
\begin{bmatrix}
X_k \\
A_k
\end{bmatrix} + \lambda_k,
\]

Select next sample point to minimize covariance

\[
P_{k+1}^{-1} = P_k^{-1} + Q,
\begin{bmatrix}
X_{k+1} \\
A_{k+1}
\end{bmatrix} =
\begin{bmatrix}
\hat{X}_{k+1} \\
\hat{A}_{k+1}
\end{bmatrix} + BU_k
\]

\[
P_{k+1} = ((P_{k+1}^{-1})^{-1} + G_{k+1}^T R^{-1} G_{k+1})^{-1}
\]

\[
G_k =
\begin{bmatrix}
I_3 & 0
\end{bmatrix}
\begin{bmatrix}
X_{k+1} \\
A_{k+1}
\end{bmatrix} =
\begin{bmatrix}
X_k \\
A_k
\end{bmatrix} + BU_k
\]

\[
\begin{bmatrix}
\hat{X}_{k+1} \\
\hat{A}_{k+1}
\end{bmatrix} =
\begin{bmatrix}
X_{k+1} \\
A_{k+1}
\end{bmatrix} + P_{k+1} G_{k+1}^T R^{-1} (Y_{k+1} - G_{k+1} \begin{bmatrix}
X_{k+1} \\
A_{k+1}
\end{bmatrix}).
\]
Implementation at ARRI’s Distributed Intelligence & Autonomy Lab (DIAL)

Mobile sensors
Built at DIAL Lab
By Dan Popa

Measured Field is a color map. Mobile robots have color sensors.

Raster Scan

Adaptive Sampling

Dan Popa
Greedy Adaptive Sampling Algorithm

Select next sample point to minimize covariance only among neighboring cells.
Cross-Layer Navigation Using Potential Fields

\[ m_i \ddot{r}_i + v_i \dot{r}_i = F_i \]  \quad \text{Mobile node eqs. of motion}

\[ F = -\nabla U(r) \]  \quad \text{attractive forces to the goals, repulsive forces among the robots and obstacles}

\[ F_{\text{restore}}(i, j) = u_{ij} (r_j - r_i) \]  \quad \text{Restoring force to avoid getting out of communication range}

\[ F_c = -\nabla C \]  \quad \text{Link communication capacity with internode distance } d

\[ C = W \log_2 \left( 1 + \frac{K P_i}{WN_o d^\alpha} \right) \]

Information potential

\[ F_{\text{inf}} = -\frac{\partial(||P_k(r)||)}{\partial r} \quad P_k(r) \text{ is the adaptive sampling error covariance calculated via the EKF}

Energy cons.

\[ v_i(t) = v_o (1 + k_v E_i(t)), \quad E_i(t) = \int_0^t F_i(\tau) \dot{r}_i(\tau) d\tau \]  \quad \text{Conserve energy by making damping increase with motion energy expended}

\[ \hat{M}_{k+1} = (\hat{X}_{k+1}^--X_k)^T W (\hat{X}_{k+1}^--X_k) \]  \quad \text{Work to go to next predicted state for adaptive sampling}
Initial configuration
Node 20 at (0,0) is a sink

Final configuration after
(7,8) is selected as a target point
Nodes 3, 12, 14 go to (7,8) to sense information
Other nodes move to maintain comm. links
Dynamic Localization of Mobile WSN

Node position

\[ X_i = [x_i, y_i]^T \]

Estimator for position

\[
\begin{bmatrix}
\dot{X}_i \\
\ddot{X}_i
\end{bmatrix} =
\begin{bmatrix}
O_2 & I_2 \\
O_2 & O_2
\end{bmatrix}
\begin{bmatrix}
X_i \\
\ddot{X}_i
\end{bmatrix} +
\begin{bmatrix}
O_2 \\
I_2
\end{bmatrix}
\begin{bmatrix}
f_i^x \\
f_i^y
\end{bmatrix}
\]

1. Relative Localization

Potential fn.

\[ V_{ugs} = \sum_{i=1}^{N} \sum_{j=1, i \neq j}^{N} \frac{1}{2} K_{ij} (r_{ij} - \bar{r}_{ij})^2 \]

Theorem. Let virtual force be given by

\[ \bar{f}_i = -\sum_{j=1}^{N} K_{ij} (r_{ij} - \bar{r}_{ij}) \frac{(X_i - X_j)}{\|X_i - X_j\|} - K_v \ddot{X}_i \]

Then the position estimates reach steady-state values that provide optimal estimates of the actual relative localization of the nodes in the sense that \( \|e\| \) is minimized.

Proof:

\[ L = V_{ugs} + \sum_{i=1}^{N} \frac{1}{2} \dot{X}_i^T \dot{X}_i \]

2. Absolute Localization

\( m \) nodes with GPS

\[ X_{ip}^a; p = 1, 2, \ldots, m \]

abs. loc. pot. fn.

\[ V_{avo} = \frac{1}{2} \sum_{p=1}^{m} \sum_{j=1}^{N} K_{ip,j} (r_{ip,j}^a - \bar{r}_{ip,j}^a)^2 + \frac{1}{2} \sum_{p=1}^{m} K_{ip} \|e_{ip}^a\|^2 \]

with \( e_{ip}^a = [(x_{ip}^a - \bar{x}_{ip}^a)^2 + (y_{ip}^a - \bar{y}_{ip}^a)^2]^{1/2} \)

Theorem. Let virtual force be given by

\[ f_{ip} = \sum_{j=1}^{N} K_{ip,j} (r_{ip,j} - \bar{r}_{ip,j}) \frac{(X_{ip}^a - X_j)}{\|X_{ip}^a - X_j\|} - K_v \ddot{X}_{ip} \]

\[ f_{ip}^a = -K_{ip}^a (X_{ip}^a - \bar{X}_{ip}^a) - \sum_{j=1}^{N} K_{ip,j}^a (r_{ip,j}^a - \bar{r}_{ip,j}^a) \frac{(X_{ip}^a - X_j)}{\|X_{ip}^a - X_j\|} - K_v \ddot{X}_{ip} \]

nodes with no GPS

nodes with GPS

Proof:

\[ V_p = \frac{1}{2} \sum_{p=1}^{m} K_{ip} \|e_{ip}^a\|^2 + \frac{1}{2} \sum_{p=1}^{m} \sum_{j=1}^{N} K_{ip,j} (r_{ip,j}^a - \bar{r}_{ip,j}^a)^2 \]

\[ + \frac{1}{2} \sum_{p=m+1}^{N} \sum_{j=1}^{N} K_{ip,j} (r_{ip,j} - \bar{r}_{ip,j})^2 \]
Range-Free Localization of Mobile WSN

1. Localization of Stationary Nodes

\[ x_{k+1}^i = A_k^i x_k^i + B_k^i u_k^i + G_k^i w_k^i \]
\[ z_k^i = H_k^i x_k^i + v_k^i \]

\[ A_k^i = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad B_k^i = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}, \quad G_k^i = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad H_k^i = \begin{bmatrix} 1 & 0 \end{bmatrix} \]

uncertainty in comm. range

\[ R_k = \sigma_{\text{Bot}} x \quad \sigma_{\text{Bot}} = \begin{bmatrix} \sigma_{\text{Bot}} x & 0 \\ 0 & \sigma_{\text{Bot}} y \end{bmatrix}, \quad \sigma_{\text{Bot}} x = \frac{\text{Range}_{\text{Bot}} x}{\sigma_{\text{const}}}, \quad \sigma_{\text{Bot}} y = \frac{\text{Range}_{\text{Bot}} y}{\sigma_{\text{const}}} \]

Algorithm 1: Static sensor node localization algorithm

1. At each discrete time instant,
2. if robot broadcast received by sensor
3. then
4. Update sensor state and uncertainty estimates using KF
5. else
6. Propagate estimates using time updates
7. end if

The first reading localizes the node to a projection on the robot’s path
2. Simultaneous Localization of Mobile Robot & Stationary Nodes

\[ \dot{X} = a(X, u, t) + G(t)w \]

\[ a(X, t) = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} v_1 \cos \alpha \cos \phi \\ v_1 \cos \alpha \sin \phi \\ \frac{v_1}{L} \sin \alpha \end{bmatrix}, \quad G(t) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \]

GPS update when available

\[ Z^*_{k \text{gps}} = h^\text{gps}[X(t_k), k] + v^\text{gps}_k \]

Update from UGS position when available

\[ Z^\text{ugs}_{k} = h^\text{ugs}[X(t_k), k] + v^\text{ugs}_k \]

\[ h^\text{gps}[X(t_k), k] = \begin{bmatrix} x \\ y \end{bmatrix}, \quad h^\text{ugs}[X(t_k), k] = \begin{bmatrix} x \\ y \end{bmatrix} \]

\[ R^\text{ugs}_k = \begin{bmatrix} P^2 + \sigma_i^2 \\ 0 \end{bmatrix}, \sigma_i = \begin{bmatrix} \sigma_{x_i} \\ 0 \\ \sigma_{y_i} \end{bmatrix} \]

\[ \sigma_{x_i} = \frac{\text{Range}_x^i}{\sigma_{\text{const}}}, \sigma_{y_i} = \frac{\text{Range}_y^i}{\sigma_{\text{const}}} \]

Includes uncertainty in position and in comm. range

**Algorithm 2:** Mobile robot localization algorithm.

1. Navigate robot along desired path.
2. Broadcast location information at discrete intervals.
3. if broadcast from GPS received
4. Update robot state and uncertainty estimates using measurement Eq. (20).
5. end if
6. if broadcast from sensor received
7. Update robot state and uncertainty estimates using measurement Eq. (21).
8. end if
3. Adaptive Localization

Mobile robot moves to localize the un-localized sensors

Problem- how does it know where to go to localize nodes with unknown positions?

Network communication connectivity is exploited

Initiation of the navigation request "NAV-REQ" packet from the robot

Badly localized sensors reply back with a localization request "LOC-REQ" packet. Already localized adjacent receiving nodes add their location and forward the request.

Algorithm 3 : Adaptive localization algorithm.
1 Broadcast Navigation request, NAV-REQ, packet.
2 Wait to receive Localization request, LOC-REQ, packets.
3 for all LOC-REQ with the same friendly neighbor
4 Combine uncertainty scalars of the requesting sensors.
5 end for
6 Pick friendly neighbor with maximum combined uncertainty scalar of the requesting sensors.
7 if multiple maxima arise
8 Among the maxima, pick the most localized friendly neighbor.
9 end if
10 Navigate around the picked friendly neighbor executing the simultaneous localization algorithm, on the sensors and on the mobile robot.
11 Repeat Steps 1-10 as required.
\[ s(t) = \sum_j w(t - jT_f - c_jT_c - \delta d_{\lfloor j/Ns \rfloor}) \]

where \( w(t) \) is the basic pulse of duration approx. 1ns, often a wavelet or a Gaussian monocycle, and \( T_f \) is the frame or pulse repetition time. In a multi-node environment, catastrophic collisions are avoided by using a pseudorandom sequence \( c_j \) to shift pulses within the frame to different compartments, and the compartment size is \( T_c \) sec. Data is transmitted using digital pulse position modulation (PPM), where if the data bit is 0 the pulse is not shifted, and if the data bit is 1 the pulse is shifted by \( d \). The same data bit is transmitted \( Ns \) times, allowing for very reliable communications with low probability of error.

**Precise time of flight measurement is possible.**

Use UWB for all three:
- Communications
- Node Relative positioning
- Target localization
Multi-Static Radar Target Localization

Intersection of two ellipses with semimajor and semiminor axes

\[
a = \frac{(d + d_2)}{2}, \quad s = \frac{d_{12}}{2}, \quad b = \sqrt{a^2 - s^2}
\]

Simultaneous solution of two quadratic equations, one for each ellipse

\[
X^T A X = 1
\]
\[
X^T B X = 1
\]

gives position of target.
ARRI Distributed Intelligence & Autonomy Lab (DIAL)

Small mobile Sensor-Dan Popa

Unattended Ground Sensors

Testbed containing MICA2 network (circle), Cricket network (triangle), Sentry robots, Garcia Robots & ARRI-bots
Supervisory Control of Mobile Wireless Sensor Networks

- **Discrete event Supervisory Controller - US Patent**
  - Fast programming of multiple missions
  - Real-time event response
  - Dynamic assignment of shared resources

**Performance Measures**

- Deadlock avoidance policy
- Program Missions - Selection of matrices
- Node Deployment & Failure - Modify $F_c$
- Select Resources - Priority modification of $F_c$

**NEXT TASK LOGIC:**
\[
\tau = F_v^r, F_v^r + F_v^r + P + P_{v_d}^r
\]
\[
v_y = S_x^r
\]

**RESOURCE RESET LOGIC:**
\[
r_y = S_x^r
\]

**MISSION COMPLETE LOGIC:**
\[
y = S_x^r
\]

**Wireless Sensor Net**
- Sensor readings events
- Tasks performed
- Resources available
- Missions completed

**Matrix DE Controller**

**Program DEC For WSN Applications**

**PC**
- High Level Controller
  - Rule Based Real Time Controller
  - Resource release:
  \[
  v_y = S_x^r 
  \]
  - Mission result:
  \[
  y = S_x^r 
  \]

**Medium Level Tasks Controllers**
- Robot 1
  - Task 1
- Robot 2
  - Task 1
- Robot 3
  - Task 1
- Wireless sensors

**Wireless Network with Internet connection**
- Finite state machine for each agent
- UC-TDMA MAC protocol

**LabVIEW User Interface**

**Supervisor control level**
- Agent control level
- Network control level
- Agents
User Interface, Monitoring, & Decision Assistance

Wireless Access over the Internet

LabVIEW Real-time Signaling & Processing

CBM Database and real time Monitoring

PDA access Failure Data from anytime and anywhere
Xbow wireless sensor boards

- Temperature, ambient light, acoustic sensors, accelerometer, and magnetometer, (can get GPS)
- Each node is endowed with a microcontroller, programmable with a C-based operating system
- Cricket motes have ultrasound rangefinders

Environmental Monitoring & Secure Area Denial
WSN for Machinery Monitoring- Diagnostics & Prognostics

Microstrain, Inc., Wireless Sensors

http://www.microstrain.com/index.cfm

RFID node
The Battery Consumption Equation

\[
\text{AmpHrs/Hr} = N_{s-tx} \left[ I_{tx} \left( T_{s-tx} + T_{tx} \right) + I_{txm} T_{tx} \right] + N_{rx-tx} \left[ I_{tx} \left( T_{rx-tx} + T_{tx} \right) + I_{txm} T_{tx} \right] \\
+ N_{s-rx} \left[ I_{rx} \left( T_{s-rx} + T_{rx} \right) \right] + N_{tx-rx} \left[ I_{rx} \left( T_{tx-rx} + T_{rx} \right) \right] \\
+ N_{rx-s} \left[ I_{r} \left( T_{rx-s} + T_{s} \right) \right] + N_{tx-s} \left[ I_{r} \left( T_{tx-s} + T_{s} \right) \right] + I_{rx/tx} T_{\text{turn-on}}
\]

- Number of times per hour, radio switches to transmit mode from sleep/receive mode
- Time taken by radio to switch to Transmit mode from sleep/receive mode
- Actual time for which radio transmit, each time it is in transmit mode
- Actual time for which radio receives, each time it is in receive mode
- Time taken by radio to switch to receive mode from sleep/transmit mode

\[ I_{tx} > I_{rx} > I_{s} \]
Sleep Schedule Calculations for Energy Conservation

Given, sweep rates for all node, number of data points from each node, frequency at which each node transmits (every \( r \) hours), the sleep durations for all the nodes in network is given by:

\[
T_p = \frac{U}{\text{diag}[S_r]}
\]

\[
S_d = \left[ T_p \times N_s^T \right] - \text{diag} \left( N_s^T \times T_p \right)
\]

\[
S_d = \left[ 3600 \times R_k^T \right] - \text{diag} \left( N_s^T \times T_p \right)
\]

Sleep duration for any given node is Total time – its own transmission time

Time taken by each node in transmitting its data

Updating rate is actually – approximate sleep duration for that particular node

If Updating rate is 1 hr for some node which transmits for 2 sec in each slot \( \Rightarrow \) the node will tx 2Sec, then sleep for 3600-2sec, and then again repeats..
CBM Network Developed and Implemented On ARRI Air Conditioning Machinery Room

Ankit Tiwari

FSM Running at Each Sensor Node

Path to Decision & Display

UC-TDMA Protocol Running at Base Station
OSI Layers Addressed

- Application
- Presentation
- Session
- Transport
- Network
- Data Link
- Physical

UC-TDMA MAC Protocol

- Application GUIs in LabVIEW

Provides all the services required by Application layer
Network Configuration Wizard

Install and Configure the Network in 1 hour

Loads with Default Values for Parameters

On Clicking, Current/default settings for that node appears in the next screen

Useful for making minor changes to node parameters

Try to Eliminate Node Naming Issue
DSP - Data to Information

Discrete Event - triggers
Advise, Decision Assistance, Alarm

LabVIEW GUIs Developed

Multiple Time Signal Display

Decision-Making
Diagnosis & Prognosis
Alarm Functions

Analysis and FFT
Wireless Sensor Nets for BCW Monitoring

- MEMS sensors for biochemical species including anthrax, nerve gases, NOx, organophosphorus
- Wireless Sensor Networks for remote site biochemical monitoring

Molecular Recognition - Rudkevich
Enzyme-Based Detection - Bob Gracy

Supported sensors for NOx based on colloidal films, mem branes, materials

Structured chemically-active nanosphere thin film - Rajeshwar

Interdigitated finger FET - Kolesar

3x3 IGEFET sensor microarray - Kolesar

DSP and C&C User Interface for wireless networks - Lewis
References


