Multiple Mobile Target Tracking in Wireless Sensor Networks

Charly Lersteau  Marc Sevaux  André Rossi

February 17, 2014
1 Introduction

2 Discretization

3 Scheduling and routing

4 Examples

5 Conclusion
1 Introduction

2 Discretization

3 Scheduling and routing

4 Examples

5 Conclusion
Initial data

- A set of *m static sensors* $I = \{1, \ldots, m\}$.
- Position of sensors: $(x_i, y_i) \in \mathbb{R}^2$, $i \in I$. 
Initial data

- A set of $m$ static sensors $I = \{1, \ldots, m\}$.
- Position of sensors: $(x_i, y_i) \in \mathbb{R}^2$, $i \in I$.
- A set of $n$ moving targets $J = \{1, \ldots, n\}$.
- Trajectory of targets: $T_j(t) \in \mathbb{R}^2$, $j \in J$. 
Initial data

- Sensing radius of sensors: \( R_i^S \).
- Energy consumption for sensing: \( e_i^S \).
Initial data

- Sensing radius of sensors: $R_i^S$.
- Energy consumption for sensing: $e_i^S$.
- Communication radius of sensors: $R_i^C$.
- Energy consumption for communication: $e_i^T$ and $e_i^R$. 
Initial data

- A base station receiving data.
Initial data

- A base station receiving data.
- A communication graph (to build).
## Problem

### Missions

- **Monitoring** the targets.
- **Reporting** the sensing data.

### Method

- **Schedule** active and sleep states of sensors.
- **Route** the sensing data to a base station.

### Objectives

- Minimize total energy consumption.
- Maximize network lifetime.

### Constraints

- Every sensor has a limited battery lifetime $E_i$.
- Every target must be covered by at least one sensor.
Problem

Missions

- **Monitoring** the targets.
- **Reporting** the sensing data.

Method

- **Schedule** *active* and *sleep* states of sensors.
- **Route** the sensing data to a *base station*.
Problem

Missions

- **Monitoring** the targets.
- **Reporting** the sensing data.

Method

- **Schedule** active and *sleep* states of sensors.
- **Route** the sensing data to a *base station*.

Objectives

- Minimize total energy consumption.
- Maximize network lifetime.
Problem

Missions

- **Monitoring** the targets.
- **Reporting** the sensing data.

Method

- **Schedule** *active* and *sleep* states of sensors.
- **Route** the sensing data to a *base station*.

Objectives

- Minimize total energy consumption.
- Maximize network lifetime.

Constraints

- Every sensor has a limited battery lifetime $E_i$.
- Every target must be covered by at least one sensor.
Two-step method

Two steps

- Discretization.
- Scheduling and routing.
Two-step method

Two steps
- Discretization.
- Scheduling and routing.

Discretization
- Goal: Transform the problem to make it solvable as a LP.
- Split the time into a set of time windows using geometric intersections.
Two-step method

Two steps
- Discretization.
- Scheduling and routing.

Discretization
- Goal: Transform the problem to make it solvable as a LP.
- Split the time into a set of time windows using geometric intersections.

Scheduling and routing
- Solve the LP
- Then, in each time window:
  - Decide which sensors are in active state to cover the targets.
  - Decide which sensors send their data to which sensors.
<table>
<thead>
<tr>
<th></th>
<th>Introduction</th>
<th>Discretization</th>
<th>Scheduling and routing</th>
<th>Examples</th>
<th>Conclusion</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Introduction</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Discretization</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Scheduling and routing</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Examples</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Conclusion</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Computing time windows

The monitored area can be seen as a planar graph [BCSZ04, SP01].

Figure: An example with $n = 3$ sensors.
Computing time windows

To each face is associated a set of covering sensors. Example: face 2 is covered by \( \{s_1, s_2\} \).

Figure: A planar graph example with \( n = 3 \) sensors, resulting in 6 faces.
Computing time windows

For each target, compute the geometric intersections between sensing area of sensors and its trajectory.

Figure: Temporal discretization of target 1
Computing time windows

For each target, compute the geometric intersections between sensing area of sensors and its trajectory.

Figure: Temporal discretization of target 2
Computing time windows

Finally, merge all computed times in the same set $T$.

Figure: Temporal discretization of all targets
Computing time windows

- Compute intersections for each target.
- Suppose sensing area of a sensor is a circle of radius $R$ and center $\gamma$, then we solve:
  \[(T_j x(t) - \gamma_x)^2 + (T_k y(t) - \gamma_y)^2 = R^2.\]
  \[\implies \text{Result is a sequence of time values ("ticks").}\]
- Finally, merge all ticks in a set $T$.
  \[\implies \text{Result is a set of } p \text{ time windows.}\]
Computing sets

Computing $S^k(j)$

For each time window $k \in \{1, \ldots, p\}$ and each target $j \in \{1, \ldots, n\}$:

Compute $S^k(j)$: set of candidate sensors covering the target $j$ during the time window $k$. 
Computing sets

**Computing** $S^k(j)$

For each time window $k \in \{1, \ldots, p\}$ and each target $j \in \{1, \ldots, n\}$:
Compute $S^k(j)$: set of candidate sensors covering the target $j$ during the time window $k$.

**Computing** $T^k(i)$

For each time window $k \in \{1, \ldots, p\}$ and each sensor $i \in \{1, \ldots, m\}$:
Compute $T^k(i)$: set of targets that can be covered by the sensor $i$ during the time window $k$. 
Computing sets

Computing $N^T(i)$

For each sensor $i \in \{1, \ldots, n\}$:
Compute $N^T(i)$: set of sensors in the communication radius of sensor $i$. 
Computing sets

**Computing $N^T(i)$**

For each sensor $i \in \{1, \ldots, n\}$:
Compute $N^T(i)$: set of sensors in the communication radius of sensor $i$.

**Computing $N^R(i)$**

For each sensor $i \in \{1, \ldots, n\}$:
Compute $N^R(i)$: set of sensors that have the sensor $i$ in their communication radius.
1 Introduction

2 Discretization

3 Scheduling and routing

4 Examples

5 Conclusion
Problem data

- $I$ : Set of sensors $\{1, \ldots, m\}$.
- $J$ : Set of targets $\{1, \ldots, n\}$.
- $K$ : Set of time windows $\{1, \ldots, p\}$.
Problem data

- $I$: Set of sensors $\{1, \ldots, m\}$.
- $J$: Set of targets $\{1, \ldots, n\}$.
- $K$: Set of time windows $\{1, \ldots, p\}$.
- $S^k(j)$: Set of sensors covering target $j$ during time window $k$.
- $T^k(i)$: Set of targets covered by sensor $i$ during time window $k$. 
Problem data

- $I$: Set of sensors $\{1, \ldots, m\}$.
- $J$: Set of targets $\{1, \ldots, n\}$.
- $K$: Set of time windows $\{1, \ldots, p\}$.
- $S^k(j)$: Set of sensors covering target $j$ during time window $k$.
- $T^k(i)$: Set of targets covered by sensor $i$ during time window $k$.
- $N^T(i)$: set of sensors in the communication radius of sensor $i$.
- $N^R(i)$: set of sensors that have the sensor $i$ in their communication radius.
Problem data

- $I$: Set of sensors $\{1, \ldots, m\}$.
- $J$: Set of targets $\{1, \ldots, n\}$.
- $K$: Set of time windows $\{1, \ldots, p\}$.
- $S^k(j)$: Set of sensors covering target $j$ during time window $k$.
- $T^k(i)$: Set of targets covered by sensor $i$ during time window $k$.
- $N^T(i)$: set of sensors in the communication radius of sensor $i$.
- $N^R(i)$: set of sensors that have the sensor $i$ in their communication radius.
- $e_i^S$: Energy consumption for sensing.
- $e_i^T$ and $e_i^R$: Energy consumption for transmission/reception.
- $E_i$: Battery lifetime of sensor $i$.
- $\Delta^k$: Duration of time window $k$ ($\Delta^k = T_{k+1} - T_k$).
Problem data

Decision

- $d_{ij}^k \geq 0$: duration of active state for sensor $i$ during the time window $k$ to watch target $j$.
- $f_{ij}^k \geq 0$: amount of data transmitted from sensor $i$ to $j$ during the time window $k$.  

Remark

LP model inspired from a model involving static targets [LCL+09].
Problem data

Decision
- \( d_{ij}^k \geq 0 \): duration of active state for sensor \( i \) during the time window \( k \) to watch target \( j \).
- \( f_{ij}^k \geq 0 \): amount of data transmitted from sensor \( i \) to \( j \) during the time window \( k \).

Additional variables (for max-lifetime)
- \( y^k \in \{0, 1\} \)
- \( \delta^k \geq 0 \)

Remark
- LP model inspired from a model involving static targets \([LCL+09]\).
Minimize energy consumption

\[
\min E = \sum_{i \in I} \sum_{k \in K} \left( \sum_{j \in T^k(i)} e^S_{ij} d^k_{ij} + \sum_{j \in N^T(i)} e^T_{ij} f^k_{ij} + \sum_{j \in N^R(i)} e^R_{ij} f^k_{ji} \right)
\]  

(1)

(6)
Minimize energy consumption

\[
\min E = \sum_{i \in I} \sum_{k \in K} \left( \sum_{j \in T^k(i)} e^S_{ij} d^k_{ij} + \sum_{j \in N^T(i)} e^T_{ij} f^k_{ij} + \sum_{j \in N^R(i)} e^R_{ij} f^k_{ji} \right) \quad (1)
\]

s.t. \[
\sum_{k \in K} \left( \sum_{j \in T^k(i)} e^S_{ij} d^k_{ij} + \sum_{j \in N^T(i)} e^T_{ij} f^k_{ij} + \sum_{j \in N^R(i)} e^R_{ij} f^k_{ji} \right) \leq E_i, \forall i \in I \quad (2)
\]

\[
\sum_{j \in T^k(i)} d^k_{ij} + \sum_{j \in N^R(i)} f^k_{ji} = \sum_{j \in N^T(i)} f^k_{ij}, \quad \forall k \in K, i \in I \quad (3)
\]
Minimize energy consumption

\[
\min E = \sum_{i \in I} \sum_{k \in K} \left( \sum_{j \in T^k(i)} e_i^S d_{ij}^k + \sum_{j \in N_T(i)} e_i^T f_{ij}^k + \sum_{j \in N_R(i)} e_i^R f_{ji}^k \right)
\]

s.t. \[
\sum_{k \in K} \left( \sum_{j \in T^k(i)} e_i^S d_{ij}^k + \sum_{j \in N_T(i)} e_i^T f_{ij}^k + \sum_{j \in N_R(i)} e_i^R f_{ji}^k \right) \leq E_i , \forall i \in I
\]

\[
\sum_{j \in T^k(i)} d_{ij}^k + \sum_{j \in N_R(i)} f_{ji}^k = \sum_{j \in N_T(i)} f_{ij}^k , \quad \forall k \in K, i \in I
\]

\[
\sum_{i \in S^k(j)} d_{ij}^k = \Delta^k , \quad \forall k \in K, j \in J
\]
Minimize energy consumption

\[
\begin{align*}
\text{min } E &= \sum_{i \in I} \sum_{k \in K} \left( \sum_{j \in T_k(i)} e_i^S d_{ij}^k + \sum_{j \in N^T(i)} e_i^T f_{ij}^k + \sum_{j \in N^R(i)} e_i^R f_{ji}^k \right) \\
\text{s.t. } &\sum_{k \in K} \left( \sum_{j \in T_k(i)} e_i^S d_{ij}^k + \sum_{j \in N^T(i)} e_i^T f_{ij}^k + \sum_{j \in N^R(i)} e_i^R f_{ji}^k \right) \leq E_i, \forall i \in I \\
\sum_{j \in T_k(i)} d_{ij}^k + \sum_{j \in N^R(i)} f_{ji}^k &= \sum_{j \in N^T(i)} f_{ij}^k, \quad \forall k \in K, i \in I \\
\sum_{i \in S_k(j)} d_{ij}^k &= \Delta^k, \quad \forall k \in K, j \in J \\
d_{ij}^k &\geq 0, \quad \forall k \in K, i \in I, j \in T_k(i) \\
f_{ij}^k &\geq 0, \quad \forall k \in K, i \in I, j \in N^T(i)
\end{align*}
\]
Maximize lifetime

\[
\max L = \sum_{k \in K} \Delta^k y^k + \sum_{k \in K} \delta^k
\]

(7)
Maximize lifetime

\[
\max L = \sum_{k \in K} \Delta^k y^k + \sum_{k \in K} \delta^k
\]

\[
\text{s.t. } \sum_{k \in K} \left( \sum_{j \in T^k(i)} e_i^S d_{ij}^k + \sum_{j \in N^T(i)} e_i^T f_{ij}^k + \sum_{j \in N^R(i)} e_i^R f_{ji}^k \right) \leq E_i, \forall i \in I
\]

\[
\sum_{j \in T^k(i)} d_{ij}^k + \sum_{j \in N^R(i)} f_{ji}^k = \sum_{j \in N^T(i)} f_{ij}^k, \quad \forall k \in K, i \in I
\]
Maximize lifetime

\[
\text{max } L = \sum_{k \in K} \Delta^k y^k + \sum_{k \in K} \delta^k \quad (7)
\]

\[
\text{s.t. } \sum_{k \in K} \left( \sum_{j \in T^k(i)} e^S_i d^k_{ij} + \sum_{j \in N^T(i)} e^T_i f^k_{ij} + \sum_{j \in N^R(i)} e^R_i f^k_{ji} \right) \leq E_i, \forall i \in I \quad (8)
\]

\[
\sum_{j \in T^k(i)} d^k_{ij} + \sum_{j \in N^R(i)} f^k_{ji} = \sum_{j \in N^T(i)} f^k_{ij}, \quad \forall k \in K, i \in I \quad (9)
\]

\[
\sum_{i \in s^k(j)} d^k_{ij} = \Delta^k y^k + \delta^k, \quad \forall k \in K, j \in J \quad (10)
\]
Maximize lifetime

\[ \text{max } L = \sum_{k \in K} \Delta^k y^k + \sum_{k \in K} \delta^k \] (7)

s.t. \[ \sum_{k \in K} \left( \sum_{j \in T^k(i)} e_i^S d_{ij}^k + \sum_{j \in N^T(i)} e_i^T f_{ij}^k + \sum_{j \in N^R(i)} e_i^R f_{ij}^k \right) \leq E_i, \forall i \in I \] (8)

\[ \sum_{j \in T^k(i)} d_{ij}^k + \sum_{j \in N^R(i)} f_{ji}^k = \sum_{j \in N^T(i)} f_{ij}^k, \forall k \in K, i \in I \] (9)

\[ \sum_{i \in S^k(j)} d_{ij}^k = \Delta^k y^k + \delta^k, \forall k \in K, j \in J \] (10)

\[ \delta^k \leq \Delta^k \left( y^{k-1} - y^k \right), \forall k \in K \ (y^0 = 1) \] (11)

\[ y^{k-1} \geq y^k, \forall k \in K \ \setminus \ \{1\} \] (12)
Maximize lifetime

\[
\max L = \sum_{k \in K} \Delta^k y^k + \sum_{k \in K} \delta^k
\]  

\[
\text{s.t. } \sum_{k \in K} \left( \sum_{j \in T^k(i)} e_i^S d^k_{ij} + \sum_{j \in N^T(i)} e_i^T f^k_{ij} + \sum_{j \in N^R(i)} e_i^R f^k_{ji} \right) \leq E_i, \forall i \in I
\]  

\[
\sum_{j \in T^k(i)} d^k_{ij} + \sum_{j \in N^R(i)} f^k_{ji} = \sum_{j \in N^T(i)} f^k_{ij}, \quad \forall k \in K, i \in I
\]  

\[
\sum_{i \in S^k(j)} d^k_{ij} = \Delta^k y^k + \delta^k, \quad \forall k \in K, j \in J
\]  

\[
\delta^k \leq \Delta^k (y^{k-1} - y^k), \quad \forall k \in K (y^0 = 1)
\]  

\[
y^{k-1} \geq y^k, \quad \forall k \in K \setminus \{1\}
\]  

\[
y^k \in \{0, 1\}, \quad \delta^k \geq 0, \quad \forall k \in K
\]  

\[
d^k_{ij} \geq 0, \quad \forall k \in K, i \in I, j \in T^k(i)
\]  

\[
f^k_{ij} \geq 0, \quad \forall k \in K, i \in I, j \in N^T(i)
\]
Result of LP

Result of LPs

- Durations: \( D^k = \begin{pmatrix} d_{1,1}^k & d_{1,2}^k & \cdots & d_{1,n}^k \\ d_{2,1}^k & d_{2,2}^k & \cdots & d_{2,n}^k \\ \vdots & \vdots & \ddots & \vdots \\ d_{m,1}^k & d_{m,2}^k & \cdots & d_{m,n}^k \end{pmatrix} \)

- Data flows: \( F^k = \begin{pmatrix} f_{1,1}^k & f_{1,2}^k & \cdots & f_{1,m}^k \\ f_{2,1}^k & f_{2,2}^k & \cdots & f_{2,m}^k \\ \vdots & \vdots & \ddots & \vdots \\ f_{m,1}^k & f_{m,2}^k & \cdots & f_{m,m}^k \end{pmatrix} \)
Bipartite graph and matching problems

- Each matrix $D^k$ can be expressed as a bipartite graph [LCL+09], connecting sensors to targets, where each $d_{i,j}^k$ corresponds to an edge.

- Solve successive perfect matching problems to decompose $D^k$ as a sum of cover matrices [LCL+09].

\[
D^k = c_1 \begin{pmatrix} 0 & 1 & \cdots & 0 \\ 0 & 0 & \cdots & 1 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 0 & \cdots & 0 \end{pmatrix} + c_2 \begin{pmatrix} 0 & 1 & \cdots & 0 \\ 1 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{pmatrix} + \cdots + c_N \begin{pmatrix} 0 & 0 & \cdots & 1 \\ 1 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 1 & \cdots & 0 \end{pmatrix}.
\]
Routing

Iterative flow filling

- Successively fill the edges as the data comes.
- When an edge is saturated, use others.
### Remarks

#### Advantages
- *Linear* models (continuous for energy minimization).
- Easily extensible to adjustable sensing ranges, flow limitations, Q-coverage \([LCL^{+}11]\)...

#### Drawbacks
- Many variables.
- A sensor *i* watching several targets consumes several times \(e_i^S\) units of energy (not suitable in some situations).
1 Introduction

2 Discretization

3 Scheduling and routing

4 Examples

5 Conclusion
Example (min energy)

- $m = 3$ sensors.
- $n = 2$ targets moving during 10 units of time.
- $E_i = 40$, $\forall i \in I$
- $e_i^S = e_i^T = e_i^R = 1$, $\forall i \in I$
- Optimal objective value: 70.5542 units of energy
Example (min energy)
Example (min energy)
Example (min energy)
Example (min energy)
Example (min energy)
Example (min energy)
Example (min energy)
Example (min energy)
Example (min energy)
Example (min energy)
Example (min energy)

\[ t_1 \quad s_1 \quad t_2 \quad s_2 \quad t_3 \quad s_3 \]
Example (max lifetime)

- \( m = 3 \) sensors.
- \( n = 2 \) targets moving during 10 units of time.
- \( E_i = 32 \), \( \forall i \in I \)
- \( e_i^S = e_i^T = e_i^R = 1 \), \( \forall i \in I \)
- Optimal objective value : 8 units of time
Example (max lifetime)
Example (max lifetime)
Example (max lifetime)
Example (max lifetime)
Example (max lifetime)
Example (max lifetime)
Example (max lifetime)

\[ t_1 \quad s_1 \]

\[ t_2 \quad s_2 \]

\[ t_3 \quad s_3 \]
Example (max lifetime)
Example (max lifetime)
Example (max lifetime)
Example (max lifetime)
1 Introduction

2 Discretization

3 Scheduling and routing

4 Examples

5 Conclusion
Conclusion

- Exact optimal solution for basic multi-target tracking.
- Linear models proposed.
- Two objectives handled:
  - Minimize energy consumption.
  - Maximize network lifetime.
- Base problem for multiple variants.

Further research

- Coping with large problems (exponential number of variables).
- Handle uncertainty (speed or trajectory not known).
- Evaluation of performance of target tracking protocols in WSN.
