Occupation Sensing, Estimation and Prediction

Christopher Clingerman
January 31, 2011
chcl@seas.upenn.edu
Outline

- Introduction
- Background Information
- Approaches to Occupancy Sensing
  - Sensor Belief Networks [1]
  - Sensor Utility Network (SUN) [3]
  - Fusion of Agent-Based Models and Sensor Data [2]
- Future Directions
Introduction

- Occupancy Sensing: Determine the number of individuals in a room, zone, building, etc.
  - Track both location and movement of occupants.

- Necessary information for efficient use of lighting and environmental conditioning, as well as security.
Occupancy sensing is typically performed with simple devices: Video cameras, IR motion sensors, optical tripwires, CO$_2$ detectors, etc.

Problems with sensor use:
- Noisy and/or inaccurate responses.
- Expense.
- Location optimization.
- Generic solutions for unique buildings.

Typically large amounts of data are produced, but only simple analysis is performed.
Sensor Belief Networks [1]


Goals:

- Use cheap sensors in a distributed network.
- Develop new, more sophisticated control algorithms.
- Combine sensor inputs through probabilistic inference to create more accurate occupancy data.
Sensor Belief Networks

- Belief Network: “A collection of conditional probability distributions associated with a directed graph.”

![Diagram of belief network](image)

Fig. 1. The probability model represented by this figure is $p(A,B,C,A,B,C,D,E) = p(D|C)p(E|Q)p(C|A,B)p(A)p(B)$. Variables $A$ and $B$ have no parents. The parents of $C$ are $A$ and $B$. The parent of $D$ is $C$, and the parent of $E$ is $C$ also. $C$ is the child of $A$ and $B$, and $D$ and $E$ are children of $C$. $D$ and $E$ have no children.
Sensor Belief Networks

- Construction of Occupancy Sensing Belief Network (for two rooms)

  “The total number of occupants in all rooms is just the sum of the numbers in each room.”

  “[T]he number of occupants persists over time.”

\[
\begin{align*}
N_1 & \rightarrow N_1 + N_2 \\
N(t-1) & \rightarrow N(t) \rightarrow N(t+1)
\end{align*}
\]
Sensor Belief Networks

Construction (cont.)

“Sensor measurements depend on the number of occupants.”

“A sensor may respond to occupancy in different ways depending on its status.”
Sensor Belief Networks

- **Temporal Dependencies**
  - Sensor status persistence (including malfunctioning sensors.)
  - Physical variable persistence (humidity, CO$_2$ levels, etc.)

- **Spatial Dependencies**
  - Overlapping sensor views.
  - Humidity and temperature.
  - HVAC properties.
Sensor Belief Networks

Fig. 6. Two slices of a belief network to model occupancy in a building, combining the effect of occupancy on observable variables, a model for each sensor, total occupancy as a function of occupancy in each space, and temporal dependence of occupancy from one time slice to the next.
Sensor Belief Networks

- Experimental Setup
  - Two private offices, each with three passive infrared (PIR) sensors and a telephone “off-the-hook” sensor.
    - PIR sensors four feet above the floor, on north (1), east (2), and south (3) walls.
    - Polled every second, returning discrete “1” and “0” information.
  - Collected data over two days.
Sensor Belief Networks

- Calibration
  - Occupancy Transition Matrix
    - Transitions in the sequence N(t-1), N(t), N(t+1) form a Markov chain.
    - Probabilities of “unoccupied→occupied” and “occupied → unoccupied” were provided by a human observer.
    - Probability of transitioning out of a state is equivalent to the inverse of the expected sojourn time.
      - Occupied state is estimated at 1024 seconds.
      - Unoccupied state is estimated at 474 seconds (ignoring unoccupied state durations over 5000 seconds.)
Sensor Belief Networks

- Calibration
  - If occupancy is a Markov chain, the distribution should be exponential.

Fig. 7. Duration of presence, both offices aggregated, data collected February 2 and 3, 2005. 78 sojourms of presence.

Fig. 8. Duration of absence, both offices aggregated, data collected February 2 and 3, 2005. Seventy-two sojourns of absence less than 5000 s long.
Sensor Belief Networks

- Calibration
  - PIR Sensitivity
    - Some sensors are more sensitive to occupancy than others.
    - Mean values calculated and used to describe PIR firing events as Poisson distributions.

<table>
<thead>
<tr>
<th>Room 1</th>
<th>Present</th>
<th>Absent</th>
</tr>
</thead>
<tbody>
<tr>
<td>PIR1</td>
<td>0.500</td>
<td>0.00163</td>
</tr>
<tr>
<td>PIR2</td>
<td>0.761</td>
<td>0.00279</td>
</tr>
<tr>
<td>PIR3</td>
<td>0.197</td>
<td>0.00159</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Room 2</th>
<th>Present</th>
<th>Absent</th>
</tr>
</thead>
<tbody>
<tr>
<td>PIR1</td>
<td>0.393</td>
<td>0.00234</td>
</tr>
<tr>
<td>PIR2</td>
<td>0.704</td>
<td>0.00190</td>
</tr>
<tr>
<td>PIR3</td>
<td>0.757</td>
<td>0.132</td>
</tr>
</tbody>
</table>
Data Validation and Comparison
- Two additional measures: Occupancy by human observers and digital video camera images.
  - Human observations have transcription errors.
  - Camera images are not available for the entire data collection period.
- PIR composite: Cumulative number of seconds in which all detectors were firing at the polling moment.
- PIR smoothed: PIR composite data recoded to show continuous occupancy within any given event.
## Sensor Belief Networks

Table 3  
Number of seconds each office (Rooms 205B and 203D) occupied and unoccupied over the 2-day monitoring period (February 02 and 03, 2005)

<table>
<thead>
<tr>
<th></th>
<th>205B 02</th>
<th>205B 03</th>
<th>203D 02</th>
<th>203D 03</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Time occupied</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Observed raw</td>
<td>17,493</td>
<td>16,197</td>
<td>33,821</td>
<td>11,494</td>
</tr>
<tr>
<td>Observed fixed</td>
<td>17,723</td>
<td>16,789</td>
<td>33,881</td>
<td>11,494</td>
</tr>
<tr>
<td>Image</td>
<td>15,085</td>
<td>16,766</td>
<td>25,829</td>
<td>11,543</td>
</tr>
<tr>
<td>PIR composite</td>
<td>13,011</td>
<td>14,050</td>
<td>26,712</td>
<td>9,823</td>
</tr>
<tr>
<td>PIR smoothed</td>
<td>17,827</td>
<td>16,979</td>
<td>34,125</td>
<td>11,538</td>
</tr>
<tr>
<td><strong>Time unoccupied</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Observed raw</td>
<td>29,308</td>
<td>30,604</td>
<td>12,980</td>
<td>35,307</td>
</tr>
<tr>
<td>Observed fixed</td>
<td>29,078</td>
<td>30,012</td>
<td>12,920</td>
<td>35,307</td>
</tr>
<tr>
<td>Image</td>
<td>31,716</td>
<td>30,035</td>
<td>20,972</td>
<td>35,258</td>
</tr>
<tr>
<td>PIR composite</td>
<td>33,790</td>
<td>32,751</td>
<td>20,089</td>
<td>36,978</td>
</tr>
<tr>
<td>PIR smoothed</td>
<td>28,974</td>
<td>29,822</td>
<td>12,676</td>
<td>35,263</td>
</tr>
</tbody>
</table>
Sensor Belief Networks

Fig. 9. Detail of Room 203D occupancy profile from morning of February 3rd, 2005.

Note: Belief network uses no post-processed PIR data.
Sensor Belief Networks

Note the faulty sensing of PIR3.

Fig. 10. PIR and belief network occupancy estimate for Room 205B on February 3rd, 2005.
Sensor Utility Network (SUN) [3]


Goals:
- Through sensory inputs and historical data, create a receding–horizon convex optimization problem.
- This solution creates efficient measures of occupancy with little cost.
Sensor Utility Network (SUN)

- **Definition:** An estimator making use of sensory input, history of a building’s energy use (utility), and the network design of the building.
- **Sensors:** CO₂, PIR, sound, video, badge counters, etc.
- **Utility:** Historical data of same building or similar, forecasts based on room schedules, walking speed in hallways, hard limits on room occupancy.
- **Network:** Building structure decomposed into zones and individual rooms, depending on primary goal of estimation and sensor distribution.
Sensor Utility Network (SUN)

- SUN Estimator
  - Combine sensory inputs to overcome weaknesses in individual sensors.
  - State process:
    \[
    \phi(t) = \begin{pmatrix} x(t) \\ r(t) \end{pmatrix}
    \]
    - where \(x(t)\) is a vector of occupancy of each zone and \(r(t)\) is a vector indicating the number of people moving from one zone to another during time-step \(t\)
  - Mass–balance constraint:
    \[
    x_i(t + 1) - x_i(t) = \sum_j r_{ji}(t) - \sum_l r_{il}(t)
    \]
To understand the role of prior information, look at the general nonlinear Bayesian model:

\[ \phi(t + 1) = f_t(\phi(t)) + W(t + 1) \]
\[ y(t) = h_t(\phi(t)) + V(t + 1) \]

- \( y(t) \) is the sequence of observations

Assume noise and \( \phi(0) \) are jointly Gaussian and mutually independent, and the noise is i.i.d. Then the MAP estimate is:

\[ \hat{\phi}(t) = \arg \max_\phi p(\phi \mid y(0), \ldots, y(t - 1)) \]
Sensor Utility Network (SUN)

Also,

\[
- \log(p(\phi_0, \ldots, \phi_T \mid y_0, \ldots, y_{T-1})) \\
\alpha \left[ \|\phi_0 - \bar{\phi}_0\|^2_{\Sigma_0^{-1}} + \sum_{t=0}^{T-1} \left\{ \|y(t) - h_t(\phi(t))\|^2_{\Sigma_{yt}^{-1}} \right\} \right. \\
+ \|\phi(t + 1) - f_t(\phi(t))\|^2_{\Sigma_{dt}^{-1}}
\]

\((\Sigma_{dt}, \Sigma_{yt})\) are the covariance matrices for \((W(t), V(t))\). \\
\(\Sigma_0\) is the covariance of the initial condition. \\
\(\bar{\phi}_0\) is the mean.
Sensor Utility Network (SUN)

- Utility and Penalty Functions

\( \mathcal{U}_x \): Occupancy utility function defined for each room or connections between rooms, such as corridors.

\( \mathcal{P}_0 \): Penalty function that models confidence in the initial estimate \( \bar{\phi}_0 \).

\( \mathcal{P}_y \): Penalty function based on sensor information

\( \mathcal{P}_d \): Penalty function based on temporal dynamics.

- Each is a function of \( \phi = (x,r) \)

- The convex program that defines the SUN estimator:

\[
\arg \min_{\phi_0, \ldots, \phi_T} \left\{ \mathcal{P}_0(\phi_0) + \sum_{t=0}^{T-1} [\mathcal{P}_y(\phi(t), y(t)) + \mathcal{P}_d(\phi(t+1), \phi(t)) - \mathcal{U}_x(\phi(t))] \right\}
\]
Sensor Utility Network (SUN)

- Construction of Functions

  - Penalty: The sensor penalty function is based on a linear observation model, while the temporal dynamics penalty function is based on the mass–balance constraint.

  - Utility: Considerations include prior knowledge of occupancy, traffic rates, patterns of behavior, and seasons or times of day.
Sensor Utility Network (SUN)

- Experimental Setup

Fig. 3. Sensor layout for people traffic estimation.
Sensor Utility Network (SUN)

Types of sensors:
- Video cameras and PIR detectors
  - Useful for capturing people counts and directional flows.
  - Can exhibit significant errors if configured incorrectly.
- CO$_2$ sensors
  - Gas concentration information is indicative of occupancy.
  - Problems include high variability and slow response time.
Sensor Utility Network (SUN)

- SUN Experimental Implementation:
  \[
  \sum_{t=0}^{T-1} \left\{ P_y(\phi(t), y(t)) + \| \phi(t+1) - A_t \phi(t) \|^2_{\Sigma_d^{-1}} - U_x(\phi(t)) \right\} + \| \phi_0 - \bar{\phi}_0 \|^2_{\Sigma_0^{-1}}
  \]

  - \( P_y \) and \( U_x \) are quadratic functions of \( \phi = (x,r) \).
  - Soft penalty functions for video cameras and PIR sensors were derived based on bias and variance estimates.
  - Occupancy utility was derived based on mean and variance of zone occupancy and smoothed over 16 days.
  - Partial information for the limited number of CO\(_2\) sensors was incorporated into the mass–balance constraint.
Sensor Utility Network (SUN)

- CO$_2$:

Fig. 6. Top panel shows CO$_2$ concentration with no occupancy, while the middle panel shows a typical profile on a weekday. Bottom panel shows the modeled occupancy (black line) and actual occupancy (red).
Comparison of SUN estimator and naïve estimator

- Naïve estimator based solely on simple counting:

\[ \hat{x}_i(t+1) - \hat{x}_i(t) = \sum_j \hat{r}_{ji}(t) - \sum_l \hat{r}_{il}(t), \]

- \( r(t) \) are estimates of people flow from video cameras and PIR detectors

- Ground truth determined by manual observation of video frames.
Fig. 7. Zone level occupancy estimates obtained from naive estimator and from the SUN estimator. Also shown are the zonal occupancy bounds used in SUN.
Sensor Utility Network (SUN)

Estimation error of naïve is about 70%. Estimation error of SUN is about 11%.

Fig. 8. Occupancy estimation at the building level using the naïve estimator and the SUN estimator.
Estimation error of naïve is about 30%.
Estimation error of SUN is about 21%.

Fig. 9. Similar plot as Figure 8 but for combined zones 7 and 8.
Fusion of Agents and Sensors [2]


Goals:
- Combine agent–based models that simulate occupant behavior with sensor data, since sensors alone have high levels of uncertainty.
- Examine agent–based models for normal, day-to-day building occupancy.
- Improve models in prior work simulating one person in one room to scale to multiple people and multiple rooms.
Fusion of Agents and Sensors

Problem: Agent-based models are largely complex and unsuitable for real-time fusion with sensor data.

Approach:
◦ Develop stochastic agent-based model that can scale to an arbitrary number of individuals and zones.
◦ Extract graphical occupancy models using Monte Carlo simulations of the agent-based model.
◦ Use the graphical model with some sensor data and the Linear Minimum Variance (LMV) estimator to compute occupancy in all zones.
The **Mixed Agent-based Rules Model** (MARM)
- Consider \( i=1, \ldots, m \) individuals, \( j=1, \ldots, n, n+1 \) nodes (\( n+1 \) indicating “outside”), and \( k=1, \ldots, 672 \) discrete time slices of 15 minutes (over the course of one week.)
- \( P_{i,j}(k) \) is the probability of agent \( i \) occupying node \( j \) at time \( k \).
  - **Nominal occupancy profile**: \( \{P_i(k), k = 1, \ldots, K\} \) is specified for every agent \( i \), where \( P_i(k) = [P_{i,1}(k), \ldots, P_{i,n+1}(k)]^T, K = 672. \)
  - Obtained by conducting a survey or from collecting sensor data.
Fusion of Agents and Sensors

- The *Mixed Agent–based Rules Model* (MARM)

  - Other considerations:
    - Occupants in hallways tend to leave quickly, whereas occupants in rooms tend to stay.
      - Transition probabilities $p_h$ and $p_r$.
    - Simulate long absences due to illness, vacations, and conferences.
      - Probability specified *a priori*, randomly initiated.
    - Access profiles for each agent determining to which rooms he/she has access. Also specifies occupancy limits.
Fusion of Agents and Sensors

Fig. 1. The process of constructing the agent-based MARM model and generating occupancy time-series using it.
Fusion of Agents and Sensors

- Preliminary Model Validation
  - Simulate the $m = n = 1$ case and compare MARM with prior work (the “Page model”) and raw sensor data.
  - Nominal occupancy profile from raw sensor data:

![Graph showing nominal occupancy profile](image)

Fig. 2. Nominal occupancy profile of the single occupant in the one-room building extracted from sensor data reported in [12].
Fusion of Agents and Sensors

- From Monte–Carlo simulations, the resulting time series for each model are used to estimate CDFs for first arrival time, total daily presence, length of continuous presence, etc.
Fusion of Agents and Sensors

- Covariance Graph Model
  - A more compact representation to use with real-time sensor data.
  - Defined by a mean $\mu$ and covariance matrix $\Sigma$, where $\Sigma$ defines a graph $G$ where a non-existent edge implies $\sigma_{i,j} = 0$.
  - Varies with time, so we have $K$ covariance graph models.
  - To identify a model:
    - First, compute sample covariance matrix through $N$ Monte-Carlo simulations for each time $k$ (to determine structure).
    - Second, use an iterative conditional fitting algorithm to determine the non-zero values of $\Sigma$. 
Fusion of Agents and Sensors

- Occupancy Estimation
  - The LMV (Linear Minimum Variance) estimator of $X$ in terms of $Y$ is:
    \[
    \hat{X}(k) = \mu_X + \Sigma_{xy} \Sigma_{yy}^{-1}(Y(k) - \mu_Y)
    \]
    where $\Sigma_{xy} = \text{cov}(X,Y)$ and $\Sigma_{yy} = \text{cov}(Y,Y)$.
  - In a linear sensing model, with a few basic assumptions, the LMV reduces to:
    \[
    \hat{X}(k) = \hat{X}(k) + \Sigma(k)C^T(C\Sigma(k)C^T + R(k))^{-1} \\
    \times (Y(k) - C\hat{X}(k))
    \]
    where $C$ is a matrix of \{0,1\}-values specifying sensor information for different nodes.
Fusion of Agents and Sensors

- Performance Evaluation (Simulation)

Fig. 4. The floor plan of the 3rd floor of MAE-B in the University of Florida campus, in which estimation was carried out.
Fusion of Agents and Sensors

- Performance Evaluation (Simulation)
  - Informal survey conducted to create nominal occupancy profiles, probability of beginning a period of long absence, and duration of long absences for each individual (45 in total.)
  - Monte–Carlo simulations used to generate one-week occupancy histories.
  - 672 covariance graph models identified from the agent-based model simulations.
  - Estimation of occupancy (using LMV) performed using 7 sensors with noise added on top of generated agent–based models.
Fusion of Agents and Sensors

(a) Estimated and true occupancy of the whole floor with extra uncertainty.

Note: Total occupancy of the whole floor cannot be inferred from sensors alone, since they do not cover all areas.

Mean error: 0.1, Standard deviation: 2.1
Fusion of Agents and Sensors

(b) Estimated and true occupancy of room 4 (no sensor) with extra uncertainty.

Room with no sensor, nominal occupancy of 1.

Mean error: 0.05, Standard deviation: 0.59
Fusion of Agents and Sensors

Room with one sensor and nominal occupancy of 7.

Mean error: 0.03, Standard deviation: 1.1
While these approaches have been verified experimentally, many have yet to be applied to real or large-scale environments. Results have not been determined for predictive applications. Each approach has the potential to be extended to more sophisticated sensor networks as well as helping to determine optimal sensor placement.
Bibliography

