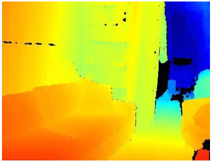
Bayesian Occupancy Grid Mapping via an Exact Inverse Sensor Model

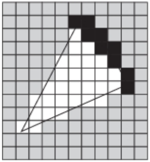
Evan Kaufman, Taeyoung Lee,
Zhuming Ai, and Ira S. Moskowitz

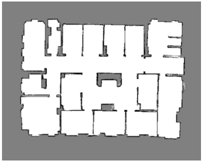
Mechanical and Aerospace Engineering
George Washington University

Introduction

- Robotic Mapping
  - Goal: generate a map representing surrounding regions
  - Crucial for simultaneous localization and mapping (SLAM) and autonomous exploration of uncertain environments
  - Popular mapping representations include ***occupancy grids***, octomaps, and feature-based maps
- Occupancy Grid Mapping
  - The environment may be decomposed into evenly spaced grid cells that are either ***occupied*** or ***free***
  - Probabilistic map: the goal is to obtain the ***probability*** of each grid cell being occupied



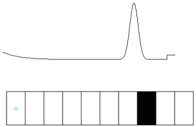




Introduction
Problem Definition

- The Map and the Robot
  - Map m is composed of n_m grid cells with known location and size
  - The i-th grid cell $\mathbf{m}_i$ is a ***static binary*** random variable, independent of other grid cells: $P(m) = P(\mathbf{m}_1, \mathbf{m}_2, \dots, \mathbf{m}_{n_m}) = \prod_{i=1}^{n_m} P(\mathbf{m}_i)$
  - ***Pose*** X_t is known, containing robot ***position*** and ***attitude***

- Depth Measurements
  - Each measurement origin and direction is known ***deterministically***
  - A measurement ***scan*** $Z_t = \{z_{t,1}, z_{t,2}, \dots, z_{t,n_z}\}$ contains n_z measurement ***rays*** (depths)
  - The ***forward sensor model*** is known from the sensor properties

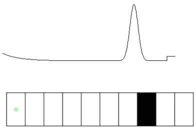


Beam Model for Range Finders

Introduction
Problem Definition

- The Map and the Robot
  - Map m is composed of n_m grid cells with known location and size
  - The i-th grid cell $\mathbf{m}_i$ is a ***static binary*** random variable, independent of other grid cells: $P(m) = P(\mathbf{m}_1, \mathbf{m}_2, \ldots, \mathbf{m}_{n_m}) = \prod_{i=1}^{n_m} P(\mathbf{m}_i)$
  - ***Pose*** X_t is known, containing robot ***position*** and ***attitude***

- Depth Measurements
  - Each measurement origin and direction is known ***deterministically***
  - A measurement ***scan*** $Z_t = \{z_{t,1}, z_{t,2}, \ldots, z_{t,n_z}\}$ contains n_z measurement ***rays*** (depths)
  - The ***forward sensor model*** is known from the sensor properties



Beam Model for Range Finders

Introduction
Problem Definition

- Bayesian Framework
  - Markov Assumption: latest a priori cell occupancy probabilities capture the information from all prior observations
  - Log-Odds Ratio: popular representation due to simple additive update structure and probability truncation avoidance, but requires the ***assumption***

$$P(Z_t|\mathbf{m}_i, X_{1:t}, Z_{1:t-1}) \approx P(Z_t|\mathbf{m}_i, X_t)$$

- Inverse Sensor Model

$$P(\mathbf{m}_i|z_{t,l}, X_{1:t}, Z_{1:t-1}) = \eta_{t,l} \sum_{m \in \mathcal{M}_i} p(z_{t,l}|m, X_t) P(m|X_{1:t-1}, Z_{1:t-1}).$$

  - Given n grid cells: $\mathcal{O}(2^n)$ is ***computationally intractable***, motivating a different solution

Introduction
Problem Definition

- Bayesian Framework
  - Markov Assumption: latest a priori cell occupancy probabilities capture the information from all prior observations
  - Log-Odds Ratio: popular representation due to simple additive update structure and probability truncation avoidance, but requires the ***assumption***

$$P(Z_t|\mathbf{m}_i, X_{1:t}, Z_{1:t-1}) \approx P(Z_t|\mathbf{m}_i, X_t)$$

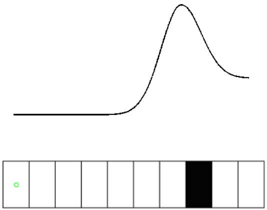
- Inverse Sensor Model

$$\begin{aligned}&P(\mathbf{m}_i|z_{t,l}, X_{1:t}, Z_{1:t-1})\\&= \eta_{t,l} \sum_{m\in\mathcal{M}_i} p(z_{t,l}|m, X_t) P(m|X_{1:t-1}, Z_{1:t-1}).\end{aligned}$$

  - Given n grid cells: $\mathcal{O}(2^n)$ is ***computationally intractable***, motivating a different solution

Introduction
Problem Definition

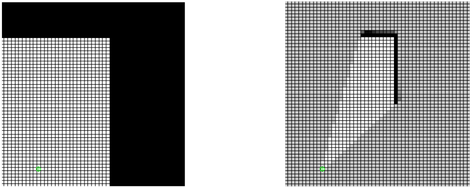
- Approximate a function for the inverse sensor model based on intuition
  - Simple to implement, but mathematically inaccurate



- Find a solution through learning
  - Simulate maps, poses, and measurements and use learning to obtain an inverse sensor model
  - Complicated, unclear how these parameters are chosen
- Goal: design a ***simple*** and ***accurate*** occupancy grid mapping method avoiding log-odds ratio assumptions

Single Measurement Ray

- Main idea: make use of occupancy grid mapping ***assumptions*** and extract ***patterns*** from probabilistic properties to find a ***computationally-efficient*** solution
  - Since the origin and direction of each measurement ray is known deterministically, the set of grid cells that the ray intersects is ***known through geometry***
  - A depth reading follows the forward sensor model, which ***only depends*** on the first occupied grid cell along the measurement ray



(a) True Grid (b) Probabilistic Map

Single Measurement Ray

- Reduced Map
  - Consider a reduced map of the l-th measurement ray, namely $r_l = \left\{\mathbf{r}_{l,1}, \mathbf{r}_{l,2}, \ldots, \mathbf{r}_{l,n_{r,l}}\right\}$, consisting of grid cells ***intersected*** by the measurement ray within the sensor limits, ***indexed by increasing distance***
  - Since reduced map r_l is composed of binary random variables, it has $2^{n_{r,l}}$ possible solutions
  - If $\mathbf{r}_{l,1}$ is occupied (a priori probability $P(\mathbf{r}_{l,1}|X_{1:t-1}, Z_{1:t-1})$), then each occupancy $\mathbf{r}_{l,2:n_{r,l}}$ does ***not*** affect forward sensor model $P(z_{t,l}|\mathbf{r}_{l,1}, X_t)$
  - Similarly, if $\mathbf{r}_{l,2}$ is occupied (a priori probability $P(\bar{\mathbf{r}}_{l,1}|X_{1:t-1}, Z_{1:t-1})P(\mathbf{r}_{l,2}|X_{1:t-1}, Z_{1:t-1})$), then forward sensor model $P(z_{t,l}|\mathbf{r}_{l,2}, X_t)$ is independent of grid cells of higher index
  - This concept is extended for all grid cells in r_l

Single Measurement Ray

- Define $P(\mathbf{r}_{l,k}|z_{t,l}, X_{1:t}, Z_{1:t-1}) = \eta_{t,l}\tilde{P}(\mathbf{r}_{l,k}|z_{t,l}, X_{1:t}, Z_{1:t-1})$ and its complement $P(\bar{\mathbf{r}}_{l,k}|z_{t,l}, X_{1:t}, Z_{1:t-1}) = 1 - P(\mathbf{r}_{l,k}|z_{t,l}, X_{1:t}, Z_{1:t-1}) = \eta_{t,l}\tilde{P}(\bar{\mathbf{r}}_{l,k}|z_{t,l}, X_{1:t}, Z_{1:t-1})$
- The k-th unnormalized probability is written generally as

$$
\begin{aligned}
&\tilde{P}(\mathbf{r}_{l,k}|z_{t,l}, X_{1:t}, Z_{1:t-1}) \\
&= P(\mathbf{r}_{l,k}|X_{1:t-1}, Z_{1:t-1}) \times \left[\sum_{i=1}^{k-1} \left\{ \prod_{j=0}^{i-1} P(\bar{\mathbf{r}}_{l,j}|X_{1:t-1}, Z_{1:t-1}) \right\} \right. \\
&\quad \left. \times\, p(z_{t,l}|\mathbf{r}_{l,i+}, X_t) P(\mathbf{r}_{l,i}|X_{1:t-1}, Z_{1:t-1}) \right] \\
&\quad + \left\{ \prod_{j=0}^{k-1} P(\bar{\mathbf{r}}_{l,j}|X_{1:t-1}, Z_{1:t-1}) \right\} \\
&\quad \times\, p(z_{t,l}|\mathbf{r}_{l,k+}, X_t) P(\mathbf{r}_{l,k}|X_{1:t-1}, Z_{1:t-1})
\end{aligned}
$$

Single Measurement Ray

- The unnormalized complement $\tilde{P}(\bar{\mathbf{r}}_{l,k}|z_{t,l}, X_{1:t}, Z_{1:t-1})$ takes a similar form
- Because $P(\mathbf{r}_{l,k}|z_{t,l}, X_{1:t}, Z_{1:t-1}) + P(\bar{\mathbf{r}}_{l,k}|z_{t,l}, X_{1:t}, Z_{1:t-1}) = 1$, the normalizer is

$$\eta_{t,l} = \Bigg[\sum_{i=1}^{n_{r,l}+1} \Bigg\{ \prod_{j=0}^{i-1} P(\bar{\mathbf{r}}_{l,j}|X_{1:t-1}, Z_{1:t-1}) \Bigg\} \times p(z_{t,l}|\mathbf{r}_{l,i+}, X_t) P(\mathbf{r}_{l,i}|X_{1:t-1}, Z_{1:t-1}) \Bigg]^{-1}$$

Single Measurement Ray

- Computational Efficiency
  - Several of the terms required for $\eta_{t,l}$ and $\tilde{P}(\mathbf{r}_{l,k}|z_{t,l}, X_{1:t}, Z_{1:t-1})$ for each $k = 1, 2, \ldots, n_{r,l}$ are repeated and need not be computed more than once
  - ***Computational order*** $\mathcal{O}(n_{r,l} + 1)$ for all grid cells in reduced map r: each grid cell once and the empty map case
  - Analytic solution is valid for any forward sensor model dependent on the closest occupied space
- Comparison to other approaches
  - It is commonly believed that the computational order is $\mathcal{O}(2^{n_{r,l}})$; the proposed method is $2^{n_{r,l}}\left(\frac{n_{r,l}}{n_{r,l}+1}\right)$ ***times faster*** for the ***same solution***
  - Proposed method completely avoids inaccuracies associated with heuristic solutions, learned methods, and log-odds ratio update assumptions

Single Measurement Ray

- Computational Efficiency
  - Several of the terms required for $\eta_{t,l}$ and $\tilde{P}(\mathbf{r}_{l,k}|z_{t,l}, X_{1:t}, Z_{1:t-1})$ for each $k = 1, 2, \ldots, n_{r,l}$ are repeated and need not be computed more than once
  - ***Computational order*** $\mathcal{O}(n_{r,l} + 1)$ for all grid cells in reduced map r: each grid cell once and the empty map case
  - Analytic solution is valid for any forward sensor model dependent on the closest occupied space
- Comparison to other approaches
  - It is commonly believed that the computational order is $\mathcal{O}(2^{n_{r,l}})$; the proposed method is $2^{n_{r,l}}\left(\frac{n_{r,l}}{n_{r,l}+1}\right)$ ***times faster*** for the ***same solution***
  - Proposed method completely avoids inaccuracies associated with heuristic solutions, learned methods, and log-odds ratio update assumptions

Full Measurement Scan

- Two Methods
  - ***Ray-By-Ray Inverse Sensor Model***: each measurement ray is evaluated individually for updating the map
  - ***Synergistic Scan Inverse Sensor Model***: all measurement rays are evaluated simultaneously

Ray-By-Ray Inverse Sensor Model

- Main idea: given a scan of measurement rays, update the map based on each measurement ray ***individually*** and ***sequentially***
- Conditional probability:

$$P(\mathbf{m}_i | X_{1:t}, Z_{1:t}) \\ = P((\ldots(((\mathbf{m}_i | X_{1:t}, Z_{1:t-1}) | z_{t,1}) | z_{t,2}) | \ldots) | z_{t,n_z})$$

- Benefit: the measurement rays maintain some dependency as they capture the same map
- Drawback: the best order for evaluating rays is not necessarily chosen

Synergistic Scan Inverse Sensor Model

- Main idea: consider every measurement ray inside a scan ***simultaneously*** to update all grid cells inside the FOV
- Probability derived from a Bayesian approach:

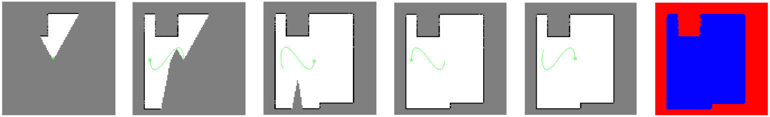
$$P(\mathbf{m}_i|X_{1:t}, Z_{1:t}) = \xi_i P(\mathbf{m}_i|X_{1:t-1}, Z_{1:t-1}) \times \prod_{l\in\mathcal{L}_i} \hat{P}(\mathbf{r}_{l,k}|z_{t,l}, X_{1:t}, Z_{1:t-1}),$$

$$\hat{P}(\mathbf{r}_{l,k}|z_{t,l}, X_{1:t}, Z_{1:t-1}) \triangleq \frac{\tilde{P}(\mathbf{r}_{l,k}|z_{t,l}, X_{1:t}, Z_{1:t-1})}{P(\mathbf{m}_i|X_{1:t-1}, Z_{1:t-1})}$$

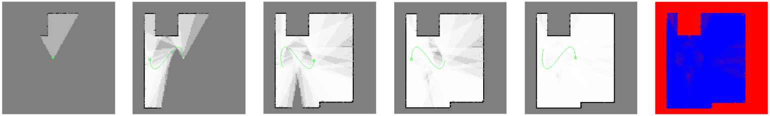
- Benefit: synergistic update provides a method where all rays are considered simultaneously
- Drawback: required assumption that the measurement rays are independent, though they capture the same map

Numerical Example

- The proposed occupancy grid mapping algorithm and a heuristic solution are compared in a simulated scenario where a robot moves around a closed room



(c) Exact Proposed Occupancy Grid Mapping



(d) Heuristic Occupancy Grid Mapping

- The robot trajectory and the same set of measurements are used with both approaches

Experimental Example

- Kinect is placed under a desk and receives a single measurement scan

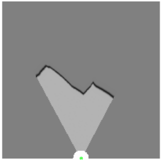
(a) Test Setup

(b) Camera View

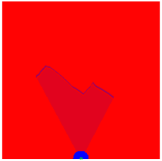
- Exact and approximate inverse sensor models are compared

(a) Exact Occupancy

(b) Exact Uncertainty



(c) Approx. Occupancy



(d) Approx. Uncertainty

Experimental Example

- Kinect is placed under a desk and receives a single measurement scan

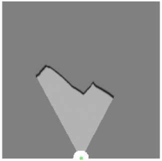
(a) Test Setup

(b) Camera View

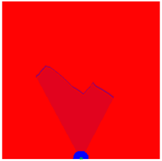
- Exact and approximate inverse sensor models are compared

(a) Exact Occupancy

(b) Exact Uncertainty



(c) Approx. Occupancy

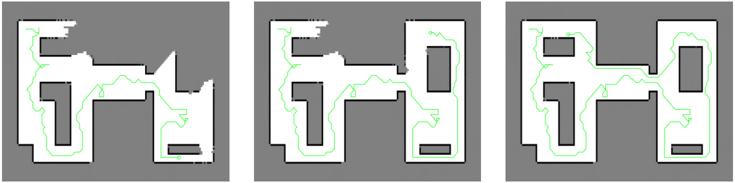


(d) Approx. Uncertainty

Numerical Example

- Similar concepts are extended to autonomous exploration
- The poses are chosen to maximize map information gain

(a) $t = 1$ sec (b) $t = 100$ sec (c) $t = 200$ sec



(d) $t = 300$ sec (e) $t = 400$ sec (f) $t = 500$ sec

Conclusions

- Proposed an occupancy grid mapping technique that uses the ***exact probabilistic solution***
- Computational cost is reduced ***substantially*** for ***real-time implementation*** using probabilisitic properties and exploiting mathematical patterns
- Assumptions required for log-odds ratio Bayesian updating are avoided
- The proposed technique is compared with a heuristic solution to show the improvement in algorithm performance

Current and Future Work

- Current Reseach
  - Using the normalizer from this research, a method to ***predict map information gain*** governs an ***autonomous exploration*** algorithm
  - Both the mapping and autonomous exploration algorithms are successfully tested with C++ ROS nodes
  - We are completing ground vehicle tests where a robot autonomously explores a room
- Future Work
  - Apply occupancy grid mapping and autonomous exploration in a ***3D*** setting with a ***flying robot***
  - Extend problem to ***multiple vehicles*** for exploring uncertain spaces