Proceedings of the 2001 IEEE/RSJ
International Conference on Intelligent Robots and Systems
Maui, Hawaii, USA, Oct. 29 - Nov. 03, 2001

Probabilistic Methods for Robotic Landmine Search

Yangang Zhang Mark Schervish Ercan U. Acar Howie Choset
Carnegie Mellon University
Pittsburgh, PA15213
yazhang,mark,eua,choset+@andrew.cmu.edu

Abstract

One way to improve the efficiency of mine search, compared with a complete coverage algorithm, is to direct the search based on the spatial distribution of the minefield. The key for the success of this probabilistic approach is to efficiently extract the spatial distribution of the minefield during the process of the search. In our research, we assume that a minefield follows a regular pattern, which belongs to a family of known patterns. A Bayesian approach algorithm to the pattern extraction is developed to extract the underlying pattern of the minefield. The algorithm performs well in its ability to catch the "actual" pattern in the situation where placement and detector errors exist. And the algorithm is efficient, therefore, online implement of the algorithm on a mobile robot is possible. Compared to the likelihood approach, the advantage of using a Bayesian approach is that this approach provides information about the uncertainty of the extracted "actual" pattern.

1 Introduction

The problem of detecting surface-laid mines and minefields is of great interest to civilians and military alike. Using autonomous agents, e.g. robots to detect mines offers promising, because it decreases the danger and cost involved in manual mine detection. In robotic search, a robot or multiple robots equipped with mine-detectors cover the minefield and detect the possible locations of mines.

In demining, to ensure that all mines are found, a robot must pass a mine-detecting sensor over all points in the region that might conceal a mine. To do this, the robot must traverse a carefully planned path through the target region. Conventional path planners are inadequate for demining because they only produce paths between two points and pay no attention to the intervening area. Coverage path planning, as its name suggests, especifically emphasizes the space swept out by the robot's detector. Integrating the robot's footprint (detector range) along the coverage path yields an area identical to that of the target region. Exhaustive Coverage algorithm[1], [2] is a coverage-path planning technique where the robot explicitly passes over all points in the minefield at least once. Exhaustive coverage is the best strategy when the robot has unlimited time and a perfect mine detector.

However, in many situations time or power limitation may not permit covering a target environment completely. Probabilistic planning technology can significantly extend the capabilities of current sensors in such demining applications. If the planner has access to a probabilistic map of mine locations, it can opportunistically guide the robot. For example, the planner might direct the robot to sweep first the area most likely to contain mines. After reaching a time limit without encountering a mine, the planner could then postulate that the area is mine-free and direct the robot to another area.

Our research addresses the problem of optimal search strategy determining location of mines and /or unexploded ordnance. We achieve this application in two steps. First, we want to construct the probability map of the mine locations in target field. Second, using this map, we plan an optimal path for the robot to locate mines.

Extracting the characteristic of dispersion pattern of the minefield helps to quickly build a probability map and to design a path for the robot searching. There are two types of typical dispersion patterns: scatter pattern and regular pattern. Scatter patterns are usually produced by submunitions released from airborne or projectile. Elliptical impact pattern with the higher density of impacts progressively towards the center is a common dispersion pattern. When mines are deployed by ground vehicles or human, it is possible that minefields follow some forms of regularity, because of the military doctrine, tactical efficiency and inherent limitations in the mine laying process[9]. The typical characteristics of regularity are collinearity and equal-spacing. That is the mines are laid sequentially roughly a constant distance apart in approximately parallel rows. Fig 1 shows such a pattern. In this paper, we focus on extracting the characteristics of a regular pattern and we will move to model the scatter pattern in our future work.

In this paper we discuss our pattern extraction method for regular spatial distribution of the minefield. A minefield follows a regular distribution i.e., the intended mine locations can be expressed as a function of a set of parameters, which characterize the underlying

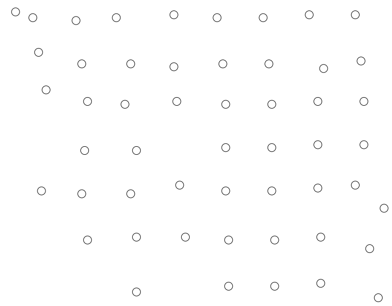

Fig. 1. Regular pattern with characteristics of collinearity and equal-spacing.

pattern. We focus on a particular minefield pattern described in Section 3. Fig. 2 shows a graph of this regular distribution. The pattern can be characterized by six parameters, which means that given these six parameters we could reproduce the minefield pattern. The key to this work is to extract the "true" pattern parameters of the spatial distribution during the process of the detection efficiently. The extracted pattern information can be used to design the optimal search strategy and provide real-time decision models for spatial orientation of robots. This work also serves as a beginning point to extend our methodology to extract patterns that belongs to a larger families of the patterns.

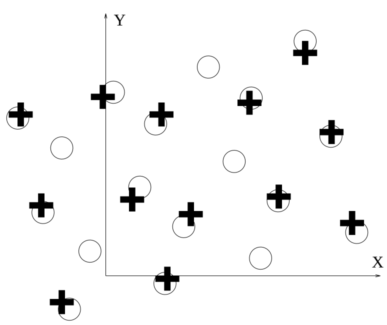

Fig. 2. A typical example of the regular pattern. "Circle" represents the intended mine position, which are the function of a set of characteristic parameters. "Cross" represents the detected mine position.

We have the following requirements to our method:

- **The method should be able to deal with uncertain information.** Presenting a strategy to determine the mine location is hard enough. The detector error further complicates the problem. Detector produces false negative, false positive and inaccurate reading about the location of a detected mine. Moreover, the model of the minefield pattern is generally inaccurate. Possible reasons for deviations of the model from the real world are deploying errors, mine explosion and model simplification. All of these random errors increase the difficulty to decode the underlying pattern.
- **The method should be computationally efficient.** It is important that the computation can be finished in real-time on the robot. The detector information must be used continuously to update the probability map of the mine locations. Moreover, real time path planning based on the updated map must be implemented onboard the robot.
- **The method should allow the integration of sensor readings from different detectors over time.** Integration of sensor reading over time could compensate noise and is necessary to resolve ambiguities about the minefield pattern.

2 Literature Review

This work has roots in classical motion planning for robots. Conventional path planning was developed to search a collision-free path to the goal from the initial position, considering the size and the shape of the robot[8]. This does not serve the application of locating for landmines. Coverage path planning on the other hand specifically emphasizes the space swept out by the robot's sensor. Integrating the robot's footprint (detector range) along the coverage path yields an area identical to that of the target region. An approach to coverage path planning problem makes use of the exact cellular decomposition method [2], [1]. to divide the target region into overlapping regions called cells such that adjacent cells share a common boundary. Complete coverage is achieved by covering each cell.

Complete coverage may be time consuming or not possible with robots that has limited power budget. Therefore, we consider a probabilistic planning that has access to a probabilistic map of mine locations. The planner can guide opportunistically the robot in the situation that time does not permit a complete coverage of a target environment. In [6], Gelenbe and Cao discuss strategies for directing robots to search for mines in a pre-selected area based on a priori spatial distribution. The prior information about the minefield is represented by a probability distribution of presence of mines. Simplified Infinite Horizon Optimization is designed to optimize the rate of finding mines. First, a stochastic process is introduced to describe robot positions. Then, the SIHO algorithm makes decisions based on optimizing the long term probability that a robot will be directed to locations proportionally to the probability that those locations contain mines. The SIHO algorithm computes the transition rates at each tth step so that the long run probability after tth step, that the robot visits a point in the search area

R matches closely the probability of finding a mine at that point. The advantage of the SIHO algorithm is that the decision is not only based on robot's perception of its immediate neighborhood, but also use of global information to match its perception of the coverage of the whole field.

However, in most of situation, a probabilistic map of mine locations is unknown . Therefore, constructing the probabilistic map by navigating the minefield is the first task. Modeling the characteristics of the spatial distribution of the minefield could help us building the probabilistic map efficiently. In statistical literatures related to the minefield detection, the characteristics of the spatial distribution of the minefield were modeled in both regular and scatter pattern minefields. The models were used to classify mines from other objects including other metal or plastic objects in the image obtained from Aerial Reconnaissance, which is another promising technology to detecting minefields and individual mines. In Aerial Reconnaissance, an aircraft flies over an area and the images of it are then analyzed to detect minefields.

Lake, Sadler and Casey [10] suggest a method to detect collinearity and regularity in regular pattern minefield. One advantage of their approach is that it effectively detects generic regularity in minefields without explicitly taking advantage of collinearity and equal-spacing. Therefore, the method can be used to detect generic regularity without prior knowledge of the possible patterns. A two-step procedure for detecting minefields is proposed whereby collinear points are first detected using a variant of a standard approach, the Hough transform and the period (spacing of the mines) then estimated using the modified Euclidean algorithm.

A different way of detecting approximate collinearity and regularity is to model the minefield as a point process. In doing so, Walsh and Raftery [11] model the approximate collinearity and equal space of mines in a more explicit way in their sequential placement model. The basic idea of their model is that mines are laid sequentially roughly a constant distance apart in approximately parallel. The distance between sequential mines, the mean distances between rows and the direction of the rows are parameters to be inferred. A Bayesian framework is used to obtain posterior probabilities of each point being a mine and a Metropolis-Hasting algorithm is designed to estimate the parameters. Six different types of proposal distributions are used, Add, Delete, Swap, Grow, Kill and Jump. At any given iteration of the Markov Chain Monte Carlo (MCMC) algorithm, one type of move is proposed. The first five moves are designed to explore current mode of the posterior locally. Jump move is intended to enable the chain to make large jumps from one posterior mode to another. Add move is to add a randomly selected point at the end of the mine row. Delete is to delete a randomly chosen end mine. Swap is proposed to swap a noise point for a randomly chosen end mine. In Grow step, three mines are proposed to form a new row. In kill step, a row with exactly 3 mines is selected and changed to noise points at random. Jump move proposes new model parameters and a new minefield based on these parameters.

A critical difference between the above two methods is that the method by Walsh and Raftery is based on the framework of an explicit statistical model. This leads directly to good estimation methods using established statistical principles. It also suggests ways of improving the method's performance in different situations, by modifying the model so as to approximate the situation considered more closely. Cressie and Lawson [3], [4] also fitted a Bayesian model via MCMC. In their case, it is a hierarchical point process model and it does not specifically model approximate linearity. The advantage of the explicit statistical model is important in our case, so we will follow the explicit method.

3 Problem Setup

Our goal is to determine the parameters for a regular spatial distribution. We focus on the particular minefield pattern described in Section 3. The pattern can be characterized by the parameter vector $\mathbf{\Delta} = (C_1, C_2, \nu, \theta, X_{00}, Y_{00})$, which means that given $\mathbf{\Delta}$ we can reproduce the minefield pattern $\boldsymbol{\mu}$. In the remainder of the paper, we will use the symbol $\mathbf{\Delta}$ to represent the pattern parameter vector. The key is to extract the unobserved underlying pattern $\boldsymbol{\mu}$ so that the probability map of the minefield can be constructed based on the underlying pattern and the optimal search strategy can be designed to guide the robot based on the probability map of the mine locations.

Our strategy to solve the pattern extraction problem is to implement a full coverage of a partial minefield region A as described in earlier work[1], [2]. The observed information is a set of detected mines at positions $\mathbf{y} = (\mathbf{y}_1, ..., \mathbf{y}_k)$, where $\mathbf{y}_i \in A, i = 1, ..., k$. A minefield pattern is simply a set of intended mines at positions $\boldsymbol{\mu} = (\boldsymbol{\mu}_1, ..., \boldsymbol{\mu}_k, ...)$. The intended mines are in some spatial relationship to each other, i.e. the minefield pattern $\boldsymbol{\mu}$ can be reproduced based on the pattern parameter vector. See Fig. 3 for a diagram of the data collection strategy. In the main parts of this section, we will concentrate on describing our methodology to solve the pattern parameter estimation problem using the information collected in the covered re-

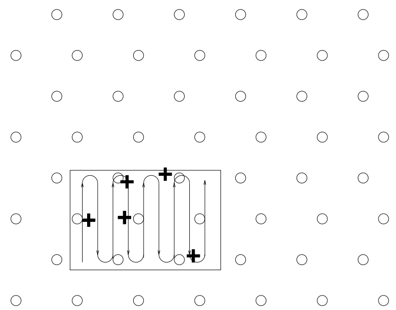

Fig. 3. Data collection strategy. The rectangle represents the cover region A. "Cross" are the found mines in the cover region.

gion.

A strong motivation to adopt a Bayesian approach is that the Maximum Likelihood Estimate (MLE) provides only a point estimate of the parameters and does not directly tell the uncertainty of the estimation. (Methods do exist for attaching uncertainties to MLE's.) A Bayesian approach calculates the posterior distribution $f_{\mathbf{\Delta}|\mathbf{Y}}(\delta|\mathbf{y}) = \frac{f_{\mathbf{Y}|\mathbf{\Delta}}(\mathbf{y}|\delta)f_{\mathbf{\Delta}}(\delta)}{f_{\mathbf{Y}}(\mathbf{y})}$ after observing the locations of some mines. Meanwhile, the observed mines $\mathbf{y}$ depend on the true minefield pattern $\mathbf{x}(\delta)$ through known conditional probability density $f_{\mathbf{Y}|\mathbf{\Delta}}(\mathbf{y}|\mathbf{x}(\delta))$, which is also called likelihood function. $f_{\mathbf{Y}|\mathbf{\Delta}}$ can be specified based on the noise model (see Section 3.2) and it is derived in Section 3.3. The posterior distribution is often impossible to compute in closed form, and even if it were possible, the density $f_{\mathbf{\Delta}|\mathbf{Y}}$ is typically impossible to recognize as anything familiar. Instead of trying to calculate the density, we will use Markov chain Monte Carlo (MCMC) to create a sample from the posterior distribution of the parameters.

3.1 Minefield Pattern

This initial research addresses a grid pattern with every other row shifted with respect to neighboring rows (Fig. 4). That pattern is selected from Field Manual 20-32 of the Department of the Army. Ultimately, this work will serve as one component of a system in which many possible patterns are considered. We use (I, J) to index the mines on the grid. The first found mine is indexed as $(0, 0)$ mine. Then, (I, J) represents the mine in the I^{th} column and J^{th} row with respect to the $(0, 0)$ mine as illustrated in Fig. 4. The scale and rotation of the pattern are described by six parameters; see Table 1 for the their descriptions.

3.2 Noise Model

Random error in location of detected mines is introduced from two main sources. First, the mine detector is imperfect; the detected mine position may be different from the real mine position. Second, error is introduced in the mine-laying process; the real

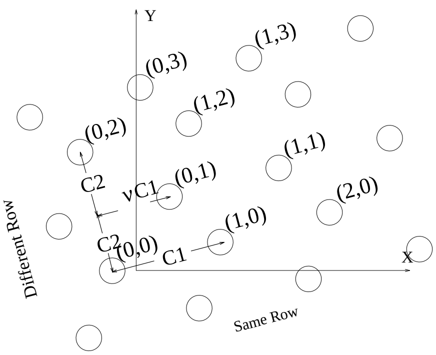

Fig. 4. The grid pattern with one row shifted with respect to the neighboring row

TABLE 1

THE PARAMETERS TO DESCRIBE THE SCALE AND THE ROTATION OF THE PATTERN.

C_1	distance between two mines on the same row.
C_2	distance between two rows.
ν	νC_1 is the shift of the odd rows with respect to the even rows.
θ	angle of the row direction with respect to the robot coordinate frame (usually w.r.t. horizon)
(X_{00}, Y_{00})	Location of the $(0, 0)$ intended mine with respect to the robot coordinate frame.

laid position is different from the position in which the mine is intended to be laid. A Gaussian white noise model is used to model the combination of errors from two sources. The detected mine position, (x, y), is bivariate normally distributed with mean (μ_x, μ_y) and fixed variance-covariance matrix $\sigma^2 I(1, 1)$. The density function is

$$f((x, y)|(\mu_x, \mu_y)) = \frac{1}{2\pi\sigma^2} e^{-\frac{(x-\mu_x)^2+(y-\mu_y)^2}{2\sigma^2}}, \quad (1)$$

where (μ_x, μ_y) is the intended mine position, which is the function of the index (I, J) and the pattern parameters $\mathbf{\Delta}$. The chance that the detected mine is located in any region decreases as the distance between the intended and detected locations increases. With 99.9% of chance, each mine will fall within a circle centered at the intended mine position with radius 3.2σ.

Typical mine detectors produce false negatives; the detector can be near a mine but not sense it. False positives are false alarms – the detector says there is a mine where there is none. Although false positives may delay demining, false negatives are catastrophic. For the sake of simplicity, the initial research assumes that the robot does not produce false positives, but false negatives. Gage [5] suggests a "cookie cutter" model for false negatives, which says that the detector responds with probability p_d if it comes within a fixed radius of the mine. Therefore, the probability of detection of a mine given that it is covered is

$$P(\text{detected}|\text{mine} \in \text{covered region}) = p_d.$$

Another possible reason that an intended mine is not detected is that the mine was accidentally not deployed or the mine exploded before the demining. Therefore, we modeled the probability of detection and mine absence together as p_d in

3.3 Likelihood Function

We assume that the detections of all of the mines in the covered region are independent of each other and of their intended locations $\mu_{(I,J)}$. We match the k found mines with the intended mines and analyze the matched (detected) intended mines and unmatched (undetected) ones separately. All of the possible combinations of the matching of the k found mines with the intended (I,J) mines should be considered. The likelihood function can be expressed as following:

$$f_{\mathbf{Y}|\mathbf{\Delta}}(\mathbf{y}_k \in A|\delta) = \sum_{\substack{in\,all\\ combinations}} \left\{ \prod_{\substack{matched\\ (I,J)\,mine}}^{k} P_m((x_k, y_k)|\mu_{(I,J)}) \prod_{\substack{unmatched\\ (I,J)\,mine}} P_{un}((I,J)) \right\}. \quad (2)$$

The calculation of P_m and P_{un} are shown in equations (4) and (5) and will be explained.

We do not have much information about the location of mines far away from the covered region. Therefore, matching only intended mines within the covered region and outside but near the covered region gives a good approximation to the likelihood function. Furthermore, the summation in (2) should sum all of the possible combinations of the matching of the k found mines with the intended (I,J) mines. But, in all of these matches, the "correct" match has significantly higher probability than the other matches. The "correct" match is defined as the match which has the highest probability among all of the matches. Therefore, we can approximate the likelihood function (2) as follows:

$$f_{\mathbf{Y}|\mathbf{\Delta}}(\mathbf{y}_k \in A|\delta) \approx \max_{\substack{in\,all\\ comb}} \left\{ \prod_{k} P_m((x_k, y_k)|\mu_{(I,J)}) \right\} \times \prod_{\substack{unmatched\\ (I,J)\,mine\\ near\,A}} P_{un}((I,J)) \quad (3)$$

In the next part, we will show how to calculate the P_m and P_{un} functions approximately.

- *Calculation of P_m.* P_m is the probability that the (I,J) mine, which is supposed to be positioned at $(\mu_{(I,J)x}, \mu_{(I,J)y})$, is detected at (x_k, y_k), which is modeled by the location error distribution f in Section 3.2. We also know that the mine is detected. Therefore

$$\begin{aligned} & P_m((x_k, y_k)|\mu_{(I,J)}) \\ = \; & p_d \cdot f((x_k, x_k)|(\mu_{(I,J)x}, \mu_{(I,J)y})) \quad (4) \\ = \; & \frac{p_d}{2\pi\sigma^2} \exp\left(-\frac{(x_k - \mu_{(I,J)x})^2 + (y_k - \mu_{(I,J)y})^2}{2\sigma^2} \right). \end{aligned}$$

- *Calculation of P_{un}.* P_{un} is the probability that the (I,J) mine is not found inside the robot covered region. Two possible reasons are (i) the mine exists inside the covered region but is not detected. (ii) the mine is outside the covered region. Therefore, P_{un} can be expressed in the following way:

$$\begin{aligned} P_{un}((I,J)) = \; & p_d \cdot [1 - P((\text{I,J}) \text{ mine} \notin A)] \\ & + P((\text{I,J}) \text{ mine} \notin A) \quad (5) \end{aligned}$$

where

$$P((\text{I,J}) \text{ mine} \notin A) \approx \begin{cases} \Phi(-\frac{h}{\sigma}) = \int_{-\infty}^{-h} \frac{1}{\sqrt{2\pi}} \exp(-\frac{h^2}{2\sigma^2}) & \text{if } (I,J) \in A \\ \Phi(\frac{h}{\sigma}) = \int_{-\infty}^{h} \frac{1}{\sqrt{2\pi}} \exp(-\frac{h^2}{2\sigma^2}) & \text{if } (I,J) \notin A \end{cases}$$

p represents the probability that a mine is not detected, h is the minimum distance between an intended (I,J) mine and the bound of a polygon A; here we approximate the cover region by a convex polygon for implementation reason.

4 Bayesian Approach Computation

A Bayesian approach calculates the posterior distribution of the parameters after observing the locations of some mines. In abstract terms, let $\mathbf{\Delta}$ be the parameter and let $\mathbf{y}$ stand for the observed data. We begin (before observing any data) with a prior distribution for the parameter $\mathbf{\Delta}$ with density $f_{\mathbf{\Delta}}(\delta)$ together with a statistical model that gives the probability of observing each possible value $\mathbf{y}$ of $\mathbf{Y}$ conditional on $\mathbf{\Delta} = \delta$ as in Section 3. The posterior density of $\mathbf{\Delta}$ after observing $\mathbf{y}$ is given by Bayes' theorem to be

$$f_{\mathbf{\Delta}|\mathbf{Y}}(\delta|\mathbf{y}) = \frac{f_{\mathbf{Y}|\mathbf{\Delta}}(\mathbf{y}|\delta) f_{\mathbf{\Delta}}(\delta)}{f_{\mathbf{Y}}(\mathbf{y})},$$

where $f_{\mathbf{Y}}(\mathbf{y}) = \int f_{\mathbf{Y}|\mathbf{\Delta}}(\mathbf{y}|\delta) f_{\mathbf{\Delta}}(\delta) d\delta$.

Instead of trying to calculate the density, we will instead draw a large sample of observations $\delta_1, \ldots, \delta_m$ from the distribution with density $f_{\mathbf{\Delta}|\mathbf{Y}}$. The method for doing this, Markov chain Monte Carlo (MCMC) [7], can be summarized as follows. Pick a starting value δ_0 from the possible values of $\mathbf{\Delta}$ and let $i = 0$.

Then we propose another value δ^*_{i+1} and compute the ratio

$$\frac{f_{\mathbf{Y}|\boldsymbol{\Delta}}(\mathbf{y}|\delta^*_{i+1})f_{\boldsymbol{\Delta}}(\delta^*_{i+1})}{f_{\mathbf{Y}|\boldsymbol{\Delta}}(\mathbf{y}|\delta_i)f_{\boldsymbol{\Delta}}(\delta_i)}.$$

If this ratio is sufficiently large, let $\delta_{i+1} = \delta^*_{i+1}$; otherwise, let $\delta_{i+1} = \delta_i$. Then increment i by one and repeat. The sequence of δ_i values forms a Markov chain whose distribution eventually becomes close to the stationary distribution of the Markov chain, which is the distribution $f_{\boldsymbol{\Delta}|\mathbf{Y}}$. After i becomes very large, we start extracting the values of our sample $\delta_1, \ldots, \delta_m$ from this Markov chain.

4.1 Mode Partitioning Algorithm

It is practically impossible to apply the above general MCMC algorithm directly. The algorithm is very inefficient because of the specialty of the likelihood function. First, the likelihood function has many local maxima. Second, the likelihood function could peak very quickly around the local maxima and the function is relatively flat when it is away from maxima. This means the MCMC algorithm could either easily be trapped in maxima or accept proposed parameters with near zero probability.

A mode partitioning algorithm is developed to solve these two problems. The goal of the mode direction algorithm is to partition the parameter space into subspaces. In each subspaces, only one local maximum exists, and the subspace only includes the area where the likelihood value is not too close to zero. Then, we can direct the MCMC algorithm from one subspace to another one. In the next part, we describe the details of the mode partition algorithm.

Let label the first found mine as $(0,0)$ mine. Then, the possible (I,J) label for the second found mine, the found mine nearest to the first mine, can be calculated in the following way. We assume that the intended locations of $(0,0)$ and (I,J) mines can lie anywhere within circles of radius $r = 3.2\sigma$ centered at the found mine locations. Also, we assume that the mines could not be closer together than some fixed distance:

$$C_1 > C_{1min} \qquad C_2 > C_{2min}$$

The possible minimum distance n between the intended locations of $(0,0)$ and (I,J) mines for a given ν can be calculated as:

$$n = \begin{cases} \sqrt{(ic_{1min})^2 + (jc_{2min})^2} & if \quad j \text{ is even.} \\ \sqrt{((i+\nu)c_{1min})^2 + (jc_{2min})^2} & if \quad j \text{ is odd.} \end{cases}$$

The valid (I,J) labels for the second mine should satisfy the following equation for some ν:

$$n < l + 2r,$$

where l is the distance between the first and second found mines.

Based on the intended mine location and the minimum distance restriction, the boundary of the parameter subspace when the second mine is labeled as a valid (I,J) can be calculated sequentially. However, due to space restriction, we are not going to give the details of these calculations. We are only going to give an outline of our approach.

- Given a valid (I,J) pair, find a bounded interval $[\nu_{min}((I,J)), \nu_{max}((I,J))]$, of valid ν values.
- Given (I,J) and ν, find a bounded interval $[\theta_{min}((I,J),\nu), \theta_{max}((I,J),\nu)]$ of valid θ values.
- Given (I,J), ν and θ, we find a region of valid $(x_{(0,0)}, y_{(0,0)})$ pairs.
- Finally, given (I,J), ν, θ and $(x_{(0,0)}, y_{(0,0)})$, we calculate bounds for c_1 and c_2 separately.

4.2 Priors

The prior is decomposed as following

$$\begin{aligned} f_{\boldsymbol{\Delta}}(\delta) &= f(c_1, c_2, \nu, \theta, x_{00}, y_{00}) \\ &= P(c_1)P(c_2)P(\nu)P(\theta)P(x_{00})P(y_{00}). \end{aligned}$$

This decomposition is chosen mostly for simplicity. Every probability is specified by prior intelligence.

4.3 Proposal Distribution for MCMC Algorithm

For all of the parameters $\boldsymbol{\Delta}$, two types of proposal distributions are used: Move and Jump. The subspaces can be indexed by (I,J), which represent the subspace in which the second found mine is indexed as (I,J) mine. Let define $\boldsymbol{\Delta}_i^{(I,J)}$, (i=1,...,6) as the i^{th} coordinate of the parameter vector $\boldsymbol{\Delta}$, which is in the (I,J) subspace. At any given iteration of the MCMC algorithm, either all of the parameters $\boldsymbol{\Delta}_i^{(I,J)}$, (i=1,...,6) are proposed to be moved within (I,J) subspace (individually, in random order, using Move proposal distribution). Or all of the parameters jump from (I,J) model to (I',J') model (using Jump proposal distribution). The Move proposal is designed to explore the current mode of the posterior locally. The Jump proposal is intended to enable the chain to make large jumps from one posterior mode to another. Which type of moves is proposed depends on a specified probability vector $p_{prob} = (p_{move}, p_{jump})$. The two types of proposals will be described below.

Move Within Subspace

The griddy Gibbs sampler for Gibbs sampler method is modified to design the proposal distribution of Move step. The method is proposed by Ritter and Tanner and the idea is to evaluate the density on a grid and use an approximate cumulative distribution function based on these grid values to generate variables with approximately the right conditional distribution. The problem to use this algorithm in its pure form is that it requires quite a fine grid and thus a very large number of posterior density evaluations to control the error in the approximation. The problem can be solved by embedding this algorithm in a Metropolis chain to ensure that the equilibrium distribution is exactly the posterior distribution even for a coarse grid. Our algorithm reaches high efficiency by balancing the acception rate and the computational complexity in every iteration. In higher-dimensional problems, the one-dimensional algorithm can be applied to each coordinate in turn as Gibbs sampler or along randomly chosen directions.

Let $[\Delta^{(I,J)}_{imin}, \Delta^{(I,J)}_{imax}]$ be the bound for the i^{th} coordinate of the parameter space, assuming that the values of the parameters except for the i^{th} coordinate, $\Delta^{(I,J)}_{-i} = (\Delta^{(I,J)}_1, \Delta^{(I,J)}_2, ..., \Delta^{(I,J)}_{i-1}, \Delta^{(I,J)}_{i+1}, ..., \Delta^{(I,J)}_n)$, are all known and the parameters are associated with the second mine in the (I,J) pair. Both $\Delta^{(I,J)}_{imin}, \Delta^{(I,J)}_{imax}$ are a function of $\Delta^{(I,J)}_{-i}$; the computation is similar as we did in section 4.

We propose a candidate value $\Delta^{*(I,J)}_i$ within the interval $[\Delta^{(I,J)}_{imin}, \Delta^{(I,J)}_{imax}]$ using a sliding lattice centered at the current location $\Delta^{(I,J)}_i$ of the chain.

- Define

$$h_m = \begin{cases} h_{\text{m-left}} = \frac{\Delta_i^{(I,J)} - \Delta_{imin}^{(I,J)}}{m} & \text{grid point value} < \Delta_i^{(I,J)} \\ h_{\text{m-right}} = \frac{\Delta_{imax}^{(I,J)} - \Delta_i^{(I,J)}}{m} & \text{grid point value} \geq \Delta_i^{(I,J)} \end{cases}$$

  Evaluate the conditional posterior density function $\pi(\Delta_i^{(I,J)} + jh_m|\Delta_{-i}^{(I,J)})$ on the grid $\Delta_i^{(I,J)} + jh_m$ for $-(m-1) \leq j \leq m-1$.

- Select a point $\Delta_i^{(I,J)} + jh_m$ from the grid points according to a distribution that is proportional to the density values $\pi(\Delta_i^{(I,J)} + jh_m|\Delta_{-i}^{(I,J)})$.

- Generate a candidate $\Delta_i^{*(I,J)}$ following uniform distribution with density function:

$$f(\Delta_i^{*(I,J)}|\Delta_i^{(I,J)}) = \begin{cases} \frac{2}{h_{\text{m-left}}+h_{\text{m-right}}} I_{[\Delta_i^{(I,J)} - \frac{h_{\text{m-left}}}{2}, \Delta_i^{(I,J)} + \frac{h_{\text{m-right}}}{2}]} & \text{if } k = 0 \\ \frac{1}{h_m} I_{[\Delta_i^{(I,J)} + kh_m - \frac{h_m}{2}, \Delta_i^{(I,J)} + kh_m + \frac{h_m}{2}]} & \text{if } k \neq 0 \end{cases}$$

Therefore, the proposal distribution is

$$q_{move}(\Delta_i^{(I,J)}, \Delta_i^{*(I,J)}|\Delta_{-i}^{(I,J)}) = \frac{\sum_{j=-(m-1)}^{(m-1)} \pi(\Delta_i^{(I,J)} + jh_m|\Delta_{-i}^{(I,J)}) f(\Delta_i^{*(I,J)}|\Delta_i^{(I,J)})}{\sum_{j=-(m-1)}^{(m-1)} \pi(\Delta_i^{(I,J)} + jh_m|\Delta_{-i}^{(I,J)})}$$

Jump Between Subspaces

Simulation efficiency is improved by incorporating the information from the Maximum Likelihood Estimation (MLE) into the proposal distribution. The local MLE can be obtained by applying a general numeric optimization algorithm in every (I,J) restricted subspace. We can get $\mathbf{\Delta}^{(I,J)}_{max}$, the MLE in (I,J) subspace, and $p^{(I,J)}_{max}$, the corresponding local maximum likelihood value. These values will be used in designing the Jump proposal. The Jump proposal is constructed in the following two steps.

- We determine the (I,J) subspace to which the MCMC algorithm will visit. The decision depends on a specified probability vector $P_{visit} = (p^{(I_1,J_1)}_{max}, p^{(I_2,J_2)}_{max}, ..., p^{(I_n,J_n)}_{max})$, where n is the number of valid (I,J) pair. We specify P_{visit}, the probability to visit (I_i, J_i) pair, to be proportional to the local maximum likelihood value in the (I,J) subspace.
- We propose a new parameter vector $\mathbf{\Delta}^{*(I,J)}$ within picked (I,J) subspace. The modified one-dimension grid Gibbs sampler algorithm, introduced in Section 4.3, is applied to each coordinate in turn to propose $\mathbf{\Delta}^{*(I,J)}$, pretending $\mathbf{\Delta}^{(I,J)}_{max}$ as the parameters accepted in the previous iteration. Therefore, the proposal distribution for the Jump step is:

$$\begin{aligned} & q_{jump}(\mathbf{\Delta}^{*(I,J)}|\mathbf{\Delta}^{(I,J)}_{max}) \\ = \ & q_{move}(\Delta^{(I,J)}_{1max}, \Delta_1^{*(I,J)}|\Delta^{(I,J)}_{2max}, ..., \Delta^{(I,J)}_{6max}) \\ & \times q_{move}(\Delta^{(I,J)}_{2max}, \Delta_2^{*(I,J)}|\Delta_1^{*(I,J)}, \Delta^{(I,J)}_{3max}, ..., \Delta^{(I,J)}_{6max}) \\ & \times \cdots \times q_{move}(\Delta^{(I,J)}_{6max}, \Delta_6^{*(I,J)}|\Delta_1^{*(I,J)}, ..., \Delta_5^{*(I,J)}) \end{aligned}$$

5 Results from Simulation Experiments

Our approach works well in terms of performance and efficiency. These are illustrated with some simulation results. A set of four "d" mines in a covered region is randomly generated based on the noise model from an intended minefield characterized by the parameters $\mathbf{\Delta}$. The mode direction algorithm guarantees that all local maxima are explored, therefore, a global maximum is guaranteed. Further, the mode direction algorithm directs the MCMC algorithm quickly out of the trap of local maxima, therefore the algorithm is very efficient and fast. In Fig 5, we show the sample of possible intended minefield simulated from our MCMC algorithm based on the posterior distribution of $\mathbf{\Delta}$. The uncertainty of the parameters depends on the amount of information observed. When we observe 11 mines instead of 4 mines by covering a larger region, in Fig [5. Bottom], we see that the parameter uncertainty is much smaller compared to Fig [5. Top]. The advantage to calculating uncertainty in the estimates using the Bayesian approach is now clear.

Because we use the maximum likelihood information in the "Jump Between Two Subspaces" part of the MCMC algorithm, and because of our design of the coordinatewise proposal distribution in the "Move Within One Subspace" part, the MCMC algorithm reaches a very high acceptance rate. That is, the MCMC algorithm is efficient. Online implementation of the algorithm on a mobile robot is clearly feasible.

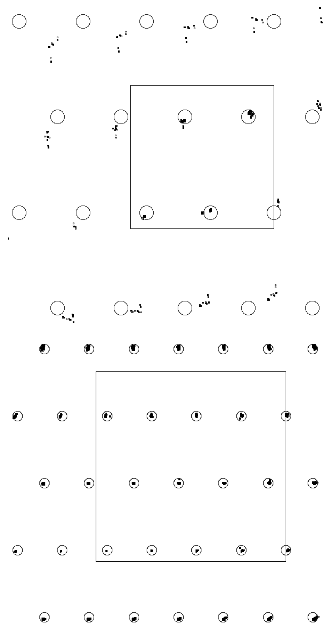

Fig.5 Sample of possible intended minefields simulated from MCMC algorithm. Top: 4 found mines, bottom: 11 found mines.

6 Conclusions

This paper discusses the strategy for directing autonomous agents to search for mines in a minefield which has a regular spatial distribution. The key to the strategy is to efficiently extract the pattern of the spatial distribution of the minefield at the beginning of the search process. Then the extracted probability distribution for the configuration of the minefield can be used to guide the search for more mines. We discuss pattern extraction algorithms assuming that the underlying pattern belongs to a family of known patterns, which can be characterized by a set of parameters.

A Bayesian approach is introduced to calculate the posterior distribution of the pattern parameters, which allows us to get not only a point estimate of the optimal parameters for constructing the underlying pattern configuration but also the uncertainty of this estimation.

The simulation evaluations are used to illustrate the performance and efficiency of the Bayesian approach algorithm. The mode direction algorithm guarantees that a global maximum is reached efficiently in the MCMC algorithm. Because we use the maximum likelihood information to design the proposal distributions, our MCMC algorithm reaches very high acceptance rate. Therefore, online implementation of our pattern search algorithms on a mobile robot is feasible.

Although this paper makes a significant contribution to the design of search strategies when the minefield follows some regular pattern, there remains much more work to be done. Current researches include: (i) to generalize the regular pattern to include the deployment error in the mine-laying process; (ii) to extend our methodologies to extract patterns, which belongs to many possible families; (iii) to incorporate the detector false positive into our noise model.

References

[1] E. Acar and H. Choset. Critical point sensing in unknown environments. In *Proc. of IEEE ICRA'00*, San Francisco, CA, 2000.

[2] H. Choset, E. Acar, A. Rizzi, and J. Luntz. Exact cellular decompositions in terms of critical points of morse functions. In *Proc. of IEEE ICRA'00*, San Francisco, CA, 2000.

[3] N. Cressie and A. B. Lawson. Models and inference for clustering of locations of mines and mine-like objects. In A. C. Dubey and R. L. Barnard, editors, *Detection and Remediation Technologies for Mines and Minelike Targets II*, volume 3079, pages 519–530, 1997.

[4] N. Cressie and A. B. Lawson. Bayesian hierarchical analysis of minefield data. In A. C. Dubey and R. L. Barnard, editors, *Detection and Remediation Technologies for Mines and Minelike Targets III*, volume 3392, pages 930–940, 1998.

[5] D. Gage. Randomized search strategies with imperfect sensors. volume Mobile Robots VIII, pages 270–279, Boston, 1993.

[6] E. Gelenbe and Y. Cao. Autonomous search for mines. *European Journal of Operational Research*, 108: 319–333, 1998.

[7] W. R. Gilks, S. Richardson, and D. J. Spiegelhalter. *Markov chain Monte Carlo in Practice*. Chapman & Hall, London, 1996.

[8] E. Hou and D. Zheng. Mobile robot path planning based on hierarchical hexagonal decomposition and artificial potential fields. *Journal of Robotic Systems*, pages 605–614, 1994.

[9] D. E. Lake and D. M. Keenan. Identifying minefields in clutter via collinearity and regularity detection. In A. C. Dubey, I. Cindrich, J. M. Ralston, K. Rigano, editors, *Detection and Remediation Technologies for Mines and Minelike Targets*, volume 2496, pages 519–530, 1995.

[10] D. E. Lake, B. Sadler, and S. Casey. Detecting regularity in minefields using collinearity and a modified euclidean algorithm. In A. C. Dubey and R. L. Barnard, editors, *Detection and Remediation Technologies for Mines and Minelike Targets II*, volume 3079, pages 500–518, 1997.

[11] D. Walsh and A. E. Raftery. *Detecting Mines in Minefields with Linear Characteristics*. Technical Report No. 359, Department of Statistics, University of Washington, 1999.