2013 IEEE/RSJ International Conference on
Intelligent Robots and Systems (IROS)
November 3-7, 2013. Tokyo, Japan

Building Variable Resolution Occupancy Maps Assuming Unknown But Bounded Sensor Errors

Marco Langerwisch and Bernardo Wagner

***Abstract*— A widely used technique for constructing two dimensional maps employing range sensors is occupancy grid mapping assuming normal distributed sensor errors. An alternative to the grid map model with its fixed grid cell size are variable resolution grid maps, e.g. quadtrees. In this paper, the authors propose an approach for building occupancy quadtree maps assuming unknown but bounded sensor errors. Therefore, they examine possible types of sensor uncertainty when using laser rangefinders. They show that the majority of possible types of sensor errors can be covered much better by bounded error models than by probabilistic models. Hence, a novel inverse sensor model has been developed that incorporates measurement and pose uncertainty in a mathematical straightforward way using interval analysis. With this model, an approach for incrementally building occupancy quadtree maps is proposed. A first real world experiment has shown the applicability of the approach. Moreover, the authors compare the map with its probabilistic grid map counterpart. The bounded error quadtree has proved to be conservative but more reliable than the conventional probabilistic grid map.**

I. INTRODUCTION

Map building is the approach of generating a map of the environment using a mobile robot platform with erroneous sensor data and exactly known poses of the robot. Metric maps are, in contrast to topological ones, spatial representations of the environment. Having each subset of the space attributed to a value describing whether the space is occupied, free, or something in between, the map is called occupancy map.

If the space is divided into equally distributed grid cells, the map model is well-known as *Occupancy Grid Map*. Grid maps have been introduced by Moravec and Elfes [1], and form a widely known map format used in up-to-date mobile robot applications when range measurements are employed. A major drawback of spatial representation via equally spaced grid cells are the memory requirements, which only depend on the size of the mapped environment and the grid cell width, and are independent of the mapped environment properties. Grid maps of the same size always reserve the same amount of space, even if the entire map is completely free, e.g.

Hierarchical data structures, e.g. applied in image processing, allow the representation of space using tree structures [2]. The so-called *Quadtrees* are able to represent two dimensional homogeneous areas with low, and areas of interest with high resolution. They have been extended by Kraetzschmar et al. to *Probabilistic Quadtrees* for the use in mobile robot mapping systems (*Variable Resolution Mapping*) [3]. The concept of variable resolution occupancy maps can be generalized to maps of arbitrary dimension and tree structure [4]. The authors have proven these maps to require much less amount of memory space in typical indoor scenarios compared to grid maps.

Common probabilistic approaches for incremental grid mapping assume the data of the ranging sensors affected with gaussian noise in longitudinal and angular dimensions [5][6]. The map update is done by applying an inverse model of the sensor, and updating the occupancy probabilities of all affected grid cells using Bayes filter. One shortcoming is the lack of incorporating pose uncertainty into the sensor model [7], when mapping becomes the problem of building a map with known but uncertain robot poses. Moreover, most range sensors do not exhibit gaussian noise properties in longitudinal and angular dimensions [8][9].

An alternative approach of modelling sensor noise is assuming unknown but bounded sensor errors [10]. We will show that the majority of possible sources of sensor errors can be covered much better by bounded error models than by probabilistic models. If the errors can be assumed to be bounded in fixed intervals, mathematical methods of interval analysis can be applied [11]. For example, pose uncertainty can be incorporated in a straightforward manner. Since interval computations calculate with *outer boundaries* instead of *probabilities*, the results are, in general, more conservative.

Unknown but bounded sensor errors, and especially interval analysis, have been successfully applied to robotics. For example, [12] and [13] present approaches to localize mobile robots assuming bounded error models. In [14], Jaulin developed an offline SLAM approach for constructing a binary occupancy map based on subpavings. To the best of our knowledge, incremental construction of occupancy maps with range sensors assuming bounded errors has not been presented before. Therefore, we will investigate the applicability of interval analysis to occupancy mapping. Moreover, we will show a natural way of using so called *Subpavings* for mapping, which will directly lead to the use of *Quadtrees* in combination with interval methods. Maps built with bounded error assumptions will, in general, be more conservative, i.e. less precise, but more reliable in obstacle mapping than their probabilistic equivalents.

In this paper, we will present an approach for incrementally building a variable resolution occupancy map using a bounded error sensor model, which describes the noise

The authors are with the Real Time Systems Group (RTS), Institute for Systems Engineering, Leibniz Universität Hannover, D-30167 Hannover, Germany. {langerwisch,wagner}@rts.uni-hannover.de

of a range sensor better than a gaussian model. Therefore, we will propose an inverse error model for 2D laser range finders. The approach will be capable of accommodating pose uncertainty in a mathematical straightforward way when calculating with known but uncertain robot poses. A first real world experiment will show its applicability, and the calculated map will be compared to its occupancy grid counterpart.

The paper is structured as follows. The next section will introduce the basics of interval analysis. Sections III and IV will develop the sensor and map models used in the mapping algorithm (Section V). Experimental results will be presented in Section VI. The paper ends with a conclusion.

II. INTERVAL ANALYSIS

In this section, the basics of interval analysis will be introduced. Furthermore, *Subpavings* and the algorithm SIVIA will be presented. All notions and algorithms in this section are taken from [11].

A. Basic Notions

A closed and connected subset of $\mathbb{R}$ is an interval

$$[x] = [x^-, x^+] = \{x \in \mathbb{R} | x^- \leq x \leq x^+\}. \quad (1)$$

The width of an interval is $w([x]) = x^+ - x^-$. A scalar $a \in \mathbb{R}$ can be seen as point interval $[a] = [a, a]$. The set of all intervals is denoted as $\mathbb{IR}$. A box $[\mathbf{x}] \in \mathbb{IR}^n$ is the cartesian product of n intervals: $[\mathbf{x}] = [x_1] \times [x_2] \times \ldots \times [x_n]$. The width of this box is $w([\mathbf{x}]) = \max_{1 \leq i \leq n} w([x_i])$.

As the image of an interval by a function is not necessarily again an interval, the notion of inclusion function has been introduced. The interval function $[\mathbf{f}] : \mathbb{IR}^n \to \mathbb{IR}^m$ for a real-valued function $\mathbf{f} : \mathbb{R}^n \to \mathbb{R}^m$ is an inclusion function for $\mathbf{f}$ if $\mathbf{f}([\mathbf{x}]) \subset [\mathbf{f}]([\mathbf{x}]), \forall [\mathbf{x}] \in \mathbb{IR}^n$. Basic elementary arithmetic operations like $+$, $-$, $*$ and $\div$, functions like $\sin$, $\tan$ or $\exp$, and common operations on sets like $\subset$, $\cap$ or $\cup$ are easily extended to the interval domain. The interval hull operator $\sqcup$ calculates the convex hull of all its operands.

The notion of inclusion test is used to test, if the set of an interval box $[\mathbf{x}]$ satisfies a given property. The inclusion test $t([\mathbf{x}])$ results in an interval boolean which can take on the values 0, 1, or $[0, 1]$, where $[0, 1]$ means undetermined.

B. Regular Subpaving

A *Subpaving* is a set of non-overlapping interval boxes, defined as

$$\mathbb{X} = \{[\mathbf{x_1}], [\mathbf{x_2}], \ldots\}. \quad (2)$$

$\mathbb{X}_i \subset \mathbb{R}^n$ is the ith interval box $[\mathbf{x}_i]$ of $\mathbb{X}$.

A subpaving is an easy representable subset of a set of interest. To make it more easier to manipulate with a computer, a subpaving can be regular. A *Regular Subpaving* can be constructed by recursive bisections and selections of the set. In general, the dimension with the larger interval width will be chosen for bisection. A regular subpaving can easily be represented as a binary tree, where each node has exactly two child nodes. The left and right child nodes of a box $[\mathbf{x}]$ can be accessed via $L([\mathbf{x}])$ and $R([\mathbf{x}])$, respectively.

C. SIVIA

Set Inverter Via Interval Analysis (SIVIA) is an algorithm to calculate the inverse solution set of an inclusion function, i.e. $\mathbb{X} = [\mathbf{f}]^{-1}(\mathbb{Y})$. The solution set consists of an inner ($\underline{\mathbb{X}}$) and an outer ($\overline{\mathbb{X}}$) regular subpaving, such that $\underline{\mathbb{X}} \subset \mathbb{X} \subset \overline{\mathbb{X}}$.

In this paper, a more generic version of SIVIA will be used, based on an inclusion test instead of an inclusion function. It is stated in Alg. 1. t is the inclusion test, $[\mathbf{x}]$ is the initial search box, ϵ is the lower threshold for the box width, and $\underline{\mathbb{X}}$ and $\overline{\mathbb{X}}$ are the resulting inner and outer regular subpavings, initialized as $\emptyset$.

Algorithm 1 SIVIA

procedure SIVIA(in: $t, [\mathbf{x}], \epsilon$, inout: $\underline{\mathbb{X}}, \overline{\mathbb{X}}$)
  if $[t]([\mathbf{x}]) = 0$ **then**
    return
  end if
  if $[t]([\mathbf{x}]) = 1$ **then**
    $\underline{\mathbb{X}} := \underline{\mathbb{X}} \cup [\mathbf{x}]$
    $\overline{\mathbb{X}} := \overline{\mathbb{X}} \cup [\mathbf{x}]$
    return
  end if
  if $w([\mathbf{x}]) < \epsilon$ **then**
    $\overline{\mathbb{X}} := \overline{\mathbb{X}} \cup [\mathbf{x}]$
    return
  end if
  SIVIA($t, L([\mathbf{x}]), \epsilon, \underline{\mathbb{X}}, \overline{\mathbb{X}}$)
  SIVIA($t, R([\mathbf{x}]), \epsilon, \underline{\mathbb{X}}, \overline{\mathbb{X}}$)
end procedure

III. INVERSE SENSOR MODEL

In this section, we will develop an inverse sensor model for the SICK LMS200 laser rangefinder, which is a sensor widely used in mobile robotics and very typical for modern range sensors. The inverse sensor model can in turn be applied to map the environment of a mobile service robot. An analysis of possible error sources and types of the employed laser rangefinder is presented in the first subsection, followed by the mathematical derivation of the sensor model.

A. Types of Sensor Errors

Laser rangefinders are optical measurement devices to measure the range to an object in the environment. A laser beam is emitted and, if it hits an object, diffusely reflected. The returning echo is detected, and a range value can be calculated based on the speed of light and the measured time between emission and detection. In its most common configuration, the SICK LMS200 emits 181 laser beams in equidistant steps of 1°, resulting in a full scan of 181 measurement values.

Sensor noise and measurement errors have a variety of sources. The exact behaviour of the SICK laser beam has been examined by several authors. Skrzypczyński [9] distinguishes between quantitative errors like an offset, changing reflection characteristics depending on the object surface, signal noise and beam spreading, and qualitative errors. Ye and Borenstein [8] discovered that there is an offset between measured and real object range. The offset depends on the

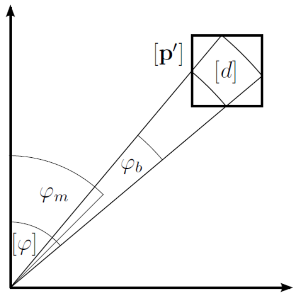


Fig. 1. Measurement box $[\mathbf{p}']$ of a single laser range measurement. Measured distance $[d]$ and measurement angle $[\varphi]$ are afflicted by sensor errors.

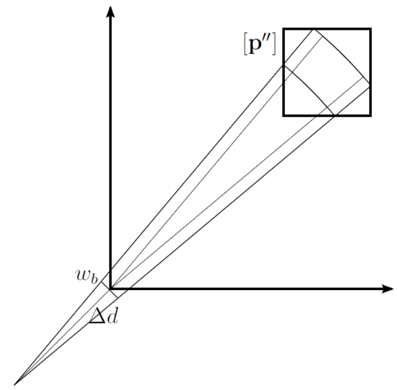


Fig. 2. Measurement box $[\mathbf{p}'']$ of a single laser range measurement with considered beam width.

sensor's state (temperature) and changes over runtime. They also detected a variance in the measured range depending on different surface materials, colors and impact angles of the laser beam. An exact relation could not be found. Sanz-Cortiella et al. [15] discovered an additional sensor noise approximately gaussian distributed. In contrast, they detected no probabilistic distribution in the intensity profile due to beam spreading, but rather fixed boundaries of the beam.

Comparing these results to existing approaches of modelling the sensor beam of laser rangefinders, a probabilistic model turns out to be not suitable for most types of sensor errors. Consequently, we can state fixed bounds for the following error types: the dynamic offset in the range measurements, the variation in the measurements due to surface properties, and the spreading of the laser beam. Additionally, we can also assume fixed bounds for the sensor noise, even if it is gaussian distributed. This leads to our novel approach of developing a sensor model that assumes all possible sources of noise to be absolutely bounded. Calculating with a bounded error model instead of a probabilistic one promises the outcome to be more conservative but more guaranteed, which means that the result will be more reliable than with probabilistic approaches regarding the mapping of obstacles.

Having a look on the second class of error sources, qualitative errors like specular reflections or so-called mixed pixels, it is obvious that they can not be modelled as absolutely bounded. They have to be treated as "outliers" during the mapping process.

B. Developing the Bounded Sensor Model

In addition to the afore mentioned sources of sensor noise and errors, the pose of the robot, i.e. the sensor, will be uncertain and modelled as a three dimensional interval box (two spatial and one angular dimension). Taking into account all uncertainties and bounded errors, we have developed a sensor model for typical laser rangefinders as follows.

As commercially available laser rangefinders do not allow to access the signal level of the detection electronics, sensor noise and errors have to be modelled via directly accessible values. Range offset, differing range measurements due to surface properties and signal noise will be modelled by an error in the distance measurement d. The error afflicted distance will be the interval $[d]$, having $w([d])$ as the maximum assumed error bound. The laser beam is assumed to spread in width with an angle of φ_b. Together with the angular position of the scanning mirror φ_m, the beam angle results in $[\varphi] = [\varphi_m - \frac{\varphi_b}{2}, \varphi_m + \frac{\varphi_b}{2}]$. Now, an outer boundary of the possible locations of the real object reflection can be calculated in the sensor coordinate frame:

$$[\mathbf{p}'] = \begin{pmatrix} [d] \cdot \cos([\varphi]) \\ [d] \cdot \sin([\varphi]) \end{pmatrix}. \tag{3}$$

This box and its further developments will be referred to as measurement box. It is depicted in Fig. 1.

Moreover, the width of the laser beam has to be considered. Therefore, we displace the origin of the beam virtually to behind. The distance $[d]$ is extended by

$$\Delta d = \cot(\frac{\varphi_b}{2}) \cdot \frac{w_b}{2}. \tag{4}$$

The new measurement box considering the beam width now calculates to

$$[\mathbf{p}''] = \begin{pmatrix} ([d] + \Delta d) \cdot \cos([\varphi]) - \Delta d \cdot \cos(\varphi_m) \\ ([d] + \Delta d) \cdot \sin([\varphi]) - \Delta d \cdot \sin(\varphi_m) \end{pmatrix}. \tag{5}$$

It is depicted in Fig. 2.

Furthermore, the displacement of the sensor on the robot platform (x_s, y_s, φ_s) has to be considered to obtain the measurement box in the robot coordinate frame. As the robot pose is known but uncertain, it is available as interval box $[\mathbf{x}] = ([x], [y], [\theta])^T$. Hence, the measurement box in the global coordinate frame is defined in (6).

IV. MAP MODELS

In this section, we will present the applied map models. First, the necessity of using a measure for the occupancy

$$[\mathbf{p}] = \begin{pmatrix} [x] + ([d] + \Delta d) \cdot \cos([\theta] + [\varphi] + \varphi_s) - \Delta d \cdot \cos([\theta] + \varphi_m + \varphi_s) + x_s \cdot \cos([\theta]) - y_s \cdot \sin([\theta]) \\ [y] + ([d] + \Delta d) \cdot \sin([\theta] + [\varphi] + \varphi_s) - \Delta d \cdot \sin([\theta] + \varphi_m + \varphi_s) + x_s \cdot \sin([\theta]) + y_s \cdot \cos([\theta]) \end{pmatrix} \tag{6}$$

certainty will be explained. When using mathematical techniques based on interval analysis, it suggests itself to use a map model based on subpavings. Therefore, *Occupancy Paving Maps* will be developed. Moreover, we will show as an outcome that *Occupancy Quadtree Maps* with interval methods for 2D mapping can be used.

A. Using Occupancy

In Sec. III-A we found out that most types of sensor errors can be modelled bounded. If all sensor errors could be covered by bounded modelling without any violation of these bounds, it would suffice to assign each map cell a binary value which takes on one of two states, *free* or *occupied*. However, it has been found that qualitative errors (”outliers”) and rare violations of the assigned error bounds cannot be completely avoided. Therefore, each map cell needs some kind of occupancy measure to fuse guaranteed and outlier measurements.

We propose to use a k-class occupancy state, where each cell can take over one of k occupancy classes. Class 1 corresponds to the empty state, class k is the occupied state, and $\lfloor \frac{k}{2} \rfloor$ is the unknown state. When a grid cell is affected by a scan update, its occupancy class is incremented or decremented.

When using hierachical tree structures as spatial representation, map cells are represented by leaf nodes. Inner nodes of the tree structures can be assigned the maximum occupancy class of its child nodes. As benefit, we obtain a variable resolution representation if we limit the depth level when traversing the tree. Hereby, a coarse path planning, e.g., could calculate on a coarse map, while a fine path planning could use the same tree structured map with a maximum detail level.

B. Occupancy Paving Map

The notion of a regular subpaving has been introduced in Sec. II-B. It is a widely used spatial representation when calculating with interval analysis. We propose to use it as a model for map representation. Moreover, we extend it to have each grid cell an occupancy class assigned as described above. When each subset of the space is annotated with an occupancy class explicitly, a *Regular Paving* can be used [16]. Thus, we obtain a map model, the *k-Class Occupancy Paving Map (k-OPM)*, easily created and modified by interval methods and well suited to represent spatial occupancy structures.

C. Occupancy Quadtree Map

Subpavings and quadtrees both are spatial structures constructed by recursive partitioning of the space, and both can be represented as trees. Obviously, they share the same properties, disregarding that subpavings are constructed by bisections, and quadtrees by quarterings. Since quadtrees are already common used in robotics for 2D mapping and path planning purposes, it is straightforward to modify existing interval methods to work on quadtrees instead of subpavings.

In addition, we extend the quadtree structure the same way as above, to have each grid cell assigned an occupancy class, resulting in a *k-Class Occupancy Quadtree Map (k-OQM)*. Properties of the resulting quadtree are well specified in [3] and [4]. Please note that an OQM can be transformed to a 2D OPM without loss of information, and vice versa.

V. MAPPING ALGORITHM

Using the techniques for modelling sensor errors and the map presented above, a novel algorithm for incremental map building can be developed. First, the method for calculating the occupied map cells of a single laser beam is derived from the inverse sensor model. Then, the calculation of the empty map cells of a single beam is presented. The section closes with the complete algorithm for updating the cells of a variable resolution map incrementally.

A. Calculating Occupied Cells

Because calculations with unknown but bounded error modelling are more conservative in their results, the outer hull of a scan point box as described in Sec. III-B is used for computing the occupied box set of a full 180° laser range scan. It includes all map cells that overlap with the scan point boxes. Therefore, the union

$$\mathbb{X}_{\mathbf{O}} = \bigcup_{i=1}^{n} [\mathbf{p}_i] \tag{7}$$

of all n scan point boxes is calculated with (6), and the union set is regularized to a regular subpaving, or quadtree respectively. The algorithm BUILDSP to build a regular subpaving from a set of boxes is stated in [17]. It can easily be extended to the construction of a quadtree. The resulting set is $\mathbb{X}_{\mathbf{OCC}}$.

B. Calculating Empty Cells

The computation of the empty map cells is not as straightforward as computing the occupied set. Calculating the empty box hull of a single laser beam could, depending on its angle and measured distance, result in a box covering most of the map space. Because the goal is to get a map estimate conservative in a sense of occupied cells instead of free cells, a different approach based on the SIVIA algorithm has been developed. It is based on the inclusion test whether a box $[\mathbf{x}_t] = ([x_t], [y_t])^T$ is included in the sensor beam between emitter and distance $[d] = [d^-, d^+]$. Therefore, in (8), (6) has been solved for $[\mathbf{d}_t] = ([d_x], [d_y])^T$. With (8), an inclusion

$$[\mathbf{d}_t] = \begin{pmatrix} \frac{[x_t]-[x]+\Delta d\cdot\cos([\theta]+\varphi_m+\varphi_s)-x_s\cdot\cos([\theta])+y_s\cdot\sin([\theta])}{\cos([\theta]+[\varphi]+\varphi_s)} - \Delta d \\ \frac{[y_t]-[y]+\Delta d\cdot\sin([\theta]+\varphi_m+\varphi_s)-x_s\cdot\sin([\theta])-y_s\cdot\cos([\theta])}{\sin([\theta]+[\varphi]+\varphi_s)} - \Delta d \end{pmatrix} \tag{8}$$

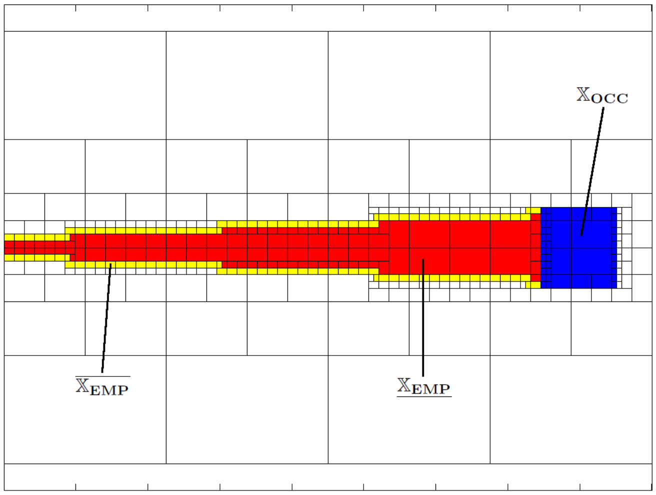


Fig. 3. Single sensor beam as a subpaving. Blue cells belong to the occupied cells set, yellow cells to the outer empty cells set, and red cells to the inner and outer empty cells set.

test testing whether a box has a non empty intersection with the empty space of a beam can be defined:

$$t_\cap([\mathbf{x}_t]) = \begin{cases} 1 & if\ [d_x] \cap [d_y] \cap [0, d^-] \neq \emptyset, \\ 0 & otherwise \end{cases}. \tag{9}$$

SIVIA (see Alg. 1) can be easily be simplified to compute the outer set in combination with $t_\cap$. If both inner and outer subpavings are necessary, the four corner points $[\mathbf{x}_t]_\ulcorner$, $[\mathbf{x}_t]_\urcorner$, $[\mathbf{x}_t]_\llcorner$, $[\mathbf{x}_t]_\lrcorner$ of the test box $[\mathbf{x}_t]$ have to be tested as well. The resulting inclusion test $t_{\mathbf{EMP}}$ is formulated as follows:

$$t_{\mathbf{EMP}}([\mathbf{x}_t]) = \\ \bigsqcup(t_\cap([\mathbf{x}_t]), t_\cap([\mathbf{x}_t]_\ulcorner), t_\cap([\mathbf{x}_t]_\urcorner), t_\cap([\mathbf{x}_t]_\llcorner), t_\cap([\mathbf{x}_t]_\lrcorner)). \tag{10}$$

To compute the empty inner and outer set of a full laser scan, the union of all n calls to SIVIA has to be computed as

$$\underline{\mathbb{X}_{\mathbf{EMP}}} = \bigcup_{i=1}^{n} \underline{\mathbb{X}}_i \tag{11}$$

and

$$\overline{\mathbb{X}_{\mathbf{EMP}}} = \bigcup_{i=1}^{n} \overline{\mathbb{X}}_i. \tag{12}$$

Fig. 3 visualizes a calculated regular subpaving of a single sensor beam, including occupied, inner and outer empty cells.

C. Map Update

With the $\mathbb{X}_{\mathbf{OCC}}$ and $\mathbb{X}_{\mathbf{EMP}}$ sets, a spatial map can be build incrementally. After calculating the $\mathbb{X}_{\mathbf{OCC}}$ and $\mathbb{X}_{\mathbf{EMP}}$ sets of a single laser scan, all affected map cells have to be updated. As we want our map to be more conservative, all cells that are part of both the $\mathbb{X}_{\mathbf{OCC}}$ and $\mathbb{X}_{\mathbf{EMP}}$ sets are removed from the $\mathbb{X}_{\mathbf{EMP}}$ set in each time step. Afterwards, the occupancy class of all map cells belonging to the $\mathbb{X}_{\mathbf{OCC}}$ set is increased, and the occupancy class of all cells belonging to the $\mathbb{X}_{\mathbf{EMP}}$ set is decreased. Moreover, if all children of a map node are leaf nodes and are member of the same occupancy class after update, they are removed, and the parent node gets a leaf node. Vice versa, if a cell of the $\mathbb{X}_{\mathbf{OCC}}$ or $\mathbb{X}_{\mathbf{EMP}}$ sets is smaller than the corresponding map cell, the map cell node is partitioned recursively to reach the same accuracy. The full algorithm is stated in Alg. 2. The choice whether to use the inner or outer $\mathbb{X}_{\mathbf{EMP}}$ set depends on the chosen error bounds and ϵ, as the inner set might be empty with relatively small error bounds and large minimum cell width.

Algorithm 2 UBBMAP

```
procedure UBBMAP(out: 𝕄)
    Initialize 𝕄 as root node with occupancy class ⌊k/2⌋
    while Sensor data available do
        Compute X_OCC and X_EMP
        X_EMP := X_EMP \ X_OCC
        INCOCCUPANCY(X_OCC)
        DECOCCUPANCY(X_EMP)
    end while
end procedure
```

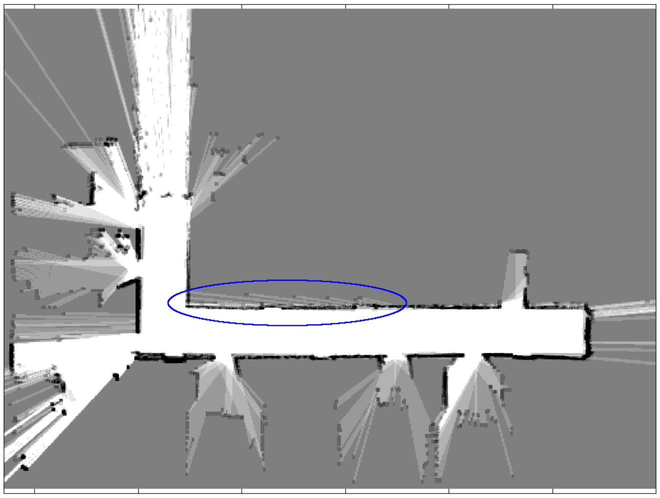
VI. EXPERIMENTAL RESULTS

To show the applicability of the presented approach for building a variable resolution map assuming unknown but bounded sensor errors, we have conducted a first real world experiment. A map of a typical indoor environment has been constructed with the presented map and sensor error models. Moreover, a probabilistic grid map has been built using the same sensor data. We will examine and compare both maps built with the different approaches.

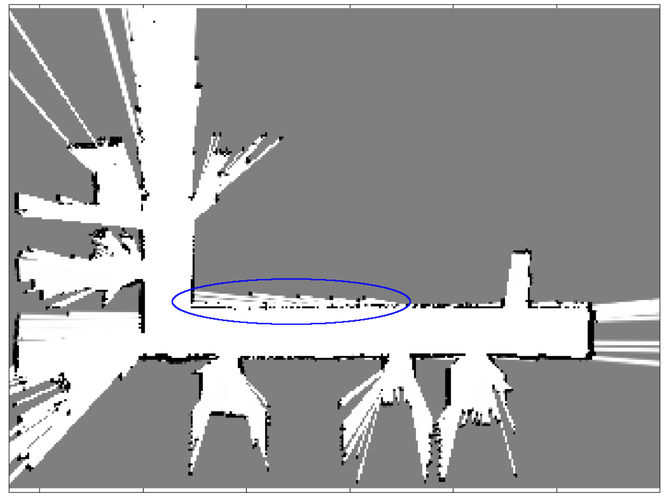
The experiments have been done using real sensor data in MATLAB. For the interval mathematics, the INTLAB toolbox [18] has been used. To build variable resolution hierarchical structures, the SCS toolbox [19] has been applied and extended. The sensor data was taken by a mobile robot platform equipped with a SICK LMS200 laser rangefinder, traversing along an office floor. During traversing, the current position of the robot was constantly computed by a Monte Carlo Localization (MCL) algorithm (200 particles, single iteration, reduced 3D laser rangefinder data [20]). Localization uncertainty has not been taken into account here. The following error bounds of the laser rangefinder were assumed: distance error $\pm 35mm$, opening angle of the beam $0.25°$, beam diameter at emitter $12mm$. They were taken from the specification provided by the manufacturer.

A. Map Building

A 10-class occupancy quadtree map has been built with the aquired sensor data. The minimum cell width threshold was $\epsilon = 100mm$. The resulting map is depicted in Fig. 4a. As can be seen, the environment has been well mapped. The floor structures are clearly identifiable. As described previously, for visualization or further processing, the map can be instantly output at a different resolution. Fig. 5



(a) Occupancy quadtree map with bounded errors



(b) Occupancy grid map with normal distributed errors

Fig. 4. Calculated occupancy maps of a real indoor environment. The occupancy scale of the cells ranges from white (free) to black (occupied).

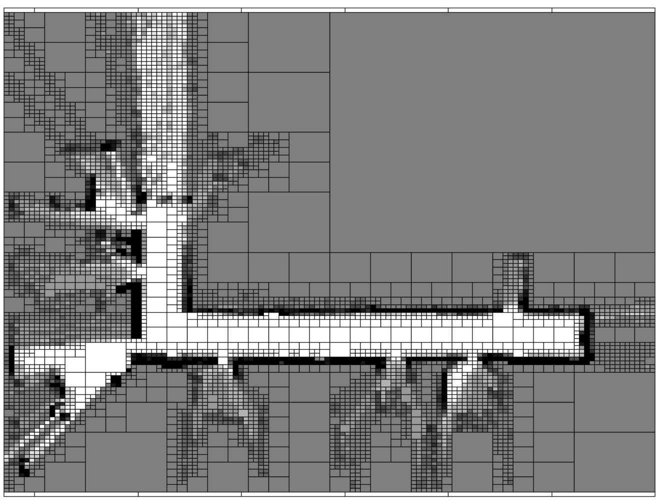


Fig. 5. The same occupancy quadtree map as in Fig. 4a, visualized with a reduced maximum grid cell width of $400mm$, and grid cell structure.

shows the same map as in Fig. 4a with a minimum cell width threshold of $400mm$. For clarity purposes, the grid structure of the quadtree has been made visible here. A map at this detail level would suffice for a coarse path planning algorithm, e.g.

B. Comparison with Occupancy Grid Map

For comparison, an occupancy grid map has been built on the same data as used in the previous experiment, applying an approach similar to those presented e.g. in [5] or [7]. Here, the errors were modelled as two normal distributions, one in longitudinal and one in angular dimension. The variances were chosen according to the bounded error parameters with $\sigma_d = 35mm$ and $\sigma_\varphi = 0.125°$. The beam diameter at the emitter could not be considered with the simple probabilistic error modelling. The cell width was constantly set to $\epsilon = 100mm$. Fig. 4b visualizes the resulting grid map.

To examine the differences, an exemplary section has been chosen and marked with a blue circle in both maps of Fig. 4. In the marked area, the grid map lacks some wall structures which have not been mapped consequently. In contrast, they are properly mapped in the quadtree map. This is due to the probabilistic model. There is an overconfidence in the center of the beam, where the normal distribution takes its highest probability values concerning both free and occupied space. When errors in the sensor model or localization errors occur, possibly occupied cells are updated with a low probability, which means they are designated as empty. On the contrary, the bounded error quadtree map is updated in equal increments along the whole beam, both in empty and occupied directions. In general, the quadtree map is more conservative in occupancy, possibly resulting in more falsely detected occupied cells compared to the grid map. The grid map could be made more conservative by increasing the variances of the normal distributions, neglecting the relation to the assumed error bounds. In fact, there is no direct relation between absolute error bounds and the variance of the normal distributions, which is a major drawback in probabilistic error models. Future work will examine this issue in detail.

The occupancy grid map consists of 73080 grid cells, while the occupancy quadtree map takes 39256 leaf nodes. Please note that computation time has not been considered yet. In its basic implementations, the grid mapping algorithm takes square, and the bounded error mapping exponential execution time. Here, the calculation of the full map took 60 minutes (grid mapping) and 1077 minutes (bounded error mapping) respectively using unoptimized MATLAB implementations.

VII. CONCLUSION

The paper presented an approach for variable resolution mapping of two dimensional indoor environments assuming unknown but bounded sensor errors. Therefore, map models and a novel inverse sensor model have been developed. We have shown that assuming absolutely bounded errors is more suitable for most possible error types of a laser rangefinder compared to probabilistic models. A first experiment has proven that the approach is capable of properly mapping a real world environment. Moreover, the built map has been compared with a fixed-size occupancy grid map employing a probabilistic sensor error model.

The experiment indicates that it is possible to obtain correct maps of variable grid size when calculating with bounded errors and interval methods. The comparison with a probabilistic computed grid map has shown that the bounded error map has typical properties for interval computations: they are more conservative, i.e. less precise, but more reliable in a sense of mapped obstacles. Moreover, the novel approach is capable of dealing with pose uncertainty in a mathematical straightforward way.

Future work will concentrate on the examination of the influence of different sensor error types and position uncertainty on the mapping result. The execution time of the approach will also be an issue. Furthermore, we will investigate the relation between absolute error bounds and the variances in the probabilistic sensor model.

REFERENCES

[1] H. Moravec and A. E. Elfes, “High resolution maps from wide angle sonar,” in *Proc. IEEE International Conference on Robotics and Automation*, March 1985, pp. 116 – 121.

[2] H. Samet, “The quadtree and related hierarchical data structures,” *ACM Comput. Surv.*, vol. 16, no. 2, pp. 187–260, Jun. 1984. [Online]. Available: http://doi.acm.org/10.1145/356924.356930

[3] G. K. Kraetzschmar, G. Pagès Gassull, and K. Uhl, “Probabilistic quadtrees for variable-resolution mapping of large environments,” in *Proc. 5th IFAC/EURON Symposium on Intelligent Autonomous Vehicles*, M. I. Ribeiro and J. Santos Victor, Eds. Lisbon, Portugal: Elsevier Science, July 2004. [Online]. Available: http://citeseer.ist.psu.edu/kraetzschmar04probabilistic.html

[4] E. Einhorn, C. Schroter, and H.-M. Gross, “Finding the adequate resolution for grid mapping - cell sizes locally adapting on-the-fly,” in *Proc. IEEE International Conference on Robotics and Automation*, may 2011, pp. 1843–1848.

[5] A. Elfes, “Using occupancy grids for mobile robot perception and navigation,” *Computer*, vol. 22, pp. 46–57, Jun. 1989.

[6] C. Cocaud and A. Jnifene, “Environment mapping using probabilistic quadtree for the guidance and control of autonomous mobile robots,” in *Proc. International Conference on Autonomous and Intelligent Systems (AIS)*, june 2010, pp. 1–6.

[7] S. Thrun, *Robotic mapping: a survey*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2003, pp. 1–35.

[8] C. Ye and J. Borenstein, “Characterization of a 2d laser scanner for mobile robot obstacle negotiation,” in *Proc. IEEE International Conference on Robotics and Automation*, vol. 3, 2002, pp. 2512 – 2518.

[9] P. Skrzypczyński, “How to recognize and remove qualitative errors in time-of-flight laser range measurements,” in *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems*, sept. 2008, pp. 2958–2963.

[10] M. Milanese and G. Belforte, “Estimation theory and uncertainty intervals evaluation in presence of unknown but bounded errors: Linear families of models and estimators,” *IEEE Transactions on Automatic Control*, vol. 27, no. 2, pp. 408–414, April 1982.

[11] L. Jaulin, M. Kieffer, O. Didrit, and E. Walter, *Applied Interval Analysis*. London: Springer, 2001.

[12] E. Seignez, M. Kieffer, A. Lambert, E. Walter, and T. Maurin, “Real-time bounded-error state estimation for vehicle tracking,” *Int. J. Rob. Res.*, vol. 28, no. 1, pp. 34–48, 2009.

[13] A. Gning and P. Bonnifait, “Constraints propagation techniques on intervals for a guaranteed localization using redundant data.” *Automatica*, vol. 42, no. 7, pp. 1167–1175, 2006. [Online]. Available: http://dblp.uni-trier.de/db/journals/automatica/automatica42.html

[14] L. Jaulin, “Range-only slam with occupancy maps: A set-membership approach,” *IEEE Transactions on Robotics*, vol. 27, no. 5, pp. 1004–1010, oct. 2011.

[15] R. Sanz-Cortiella, J. Llorens-Calveras, J. R. Rosell-Polo, E. Gregorio-Lopez, and J. Palacin-Roca, “Characterisation of the LMS200 laser beam under the influence of blockage surfaces. influence on 3d scanning of tree orchards,” *Sensors*, vol. 11, no. 3, pp. 2751–2772, 2011.

[16] J. Harlow, R. Sainudiin, and W. Tucker, “Mapped regular pavings,” *Reliable Computing*, vol. 16, pp. 252–282, 2012.

[17] E. Seignez and A. Lambert, “Complexity study of guaranteed state estimation for real time to robot localization,” *Journal of Automation, Mobile Robotics & Intelligent Systems*, vol. 5, no. 2, pp. 12–27, 2011.

[18] S. Rump, “INTLAB - INTerval LABoratory,” in *Developments in Reliable Computing*, T. Csendes, Ed. Dordrecht: Kluwer Academic Publishers, 1999, pp. 77–104. [Online]. Available: http://www.ti3.tu-harburg.de/rump/

[19] S. Tornil-Sin, V. Puig, and T. Escobet, “Set computations with subpavings in MATLAB: The SCS toolbox,” in *Proc. IEEE International Symposium on Computer-Aided Control System Design*, 2010, pp. 1403–1408.

[20] O. Wulf, K. Arras, H. Christensen, and B. Wagner, “2D mapping of cluttered indoor environments by means of 3D perception,” in *Proc. IEEE International Conference on Robotics and Automation ICRA 2004*, vol. 4, 2004, pp. 4204–4209 Vol.4.