2014 IEEE International Conference on Robotics & Automation (ICRA)
Hong Kong Convention and Exhibition Center
May 31 - June 7, 2014. Hong Kong, China

Belief Propagation Based Localization and Mapping Using Sparsely Sampled GNSS SNR Measurements

Andrew T. Irish, Jason T. Isaacs, François Quitin, João P. Hespanha, and Upamanyu Madhow

Abstract— **A novel approach is proposed to achieve simultaneous localization and mapping (SLAM) based on the signal-to-noise ratio (SNR) of global navigation satellite system (GNSS) signals. It is assumed that the environment is unknown and that the receiver location measurements (provided by a GNSS receiver) are noisy. The 3D environment map is decomposed into a grid of binary-state cells (occupancy grid) and the receiver locations are approximated by sets of particles. Using a large number of sparsely sampled GNSS SNR measurements and receiver/satellite coordinates (all available from off-the-shelf GNSS receivers), likelihoods of blockage are associated with every receiver-to-satellite beam. The posterior distribution of the map and poses is shown to represent a factor graph, on which Loopy Belief Propagation is used to efficiently estimate the probabilities of each cell being occupied or empty, along with the probability of the particles for each receiver location. Experimental results demonstrate our algorithm's ability to coarsely map (in three dimensions) a corner of a university campus, while also correcting for uncertainties in the location of the GNSS receiver.**

I. Introduction

For decades, global navigation satellite systems (GNSS) have been widely used for geolocation purposes. Several satellite systems have already been successfully deployed, (e.g., America's GPS and Russia's GLONASS), with others to follow soon (Europe's Galileo and China's BeiDou). The end result is that an ever growing constellation of satellites is broadcasting signals so that earth-based GNSS receivers can determine their geolocation. GNSS receivers, such as the iPhone 5S smartphone which supports both GPS and GLONASS, can passively log information derived from different satellite signals. This information, while principally containing the estimated latitude/longitude coordinates of the receiver (plus an uncertainty value), also contains the azimuth, elevation, identifier, and signal strength of each satellite in view.

In this paper, we take the view that GNSS devices, which are nearly ubiquitous in modern society, are effectively passive environment sensors. When a mobile GNSS receiver traverses an area, the line-of-sight (LOS) to some of the satellites will frequently be blocked by obstacles, such as buildings, trees, etc. This will result in non-line-of-sight (NLOS) signals, characterized by a lower signal-to-noise ratios (SNR) than LOS signals. Conversely, this observation can provide geographical information about obstacles: If the receiver and satellite coordinates are known, whenever a GNSS signal is blocked it is possible to determine the direction of an obstacle. An example illustration can be seen in Figure 1, where at a particular receiver location the paths to some of the satellites are LOS and some NLOS. By fusing information from multiple receiver locations and multiple satellites, it becomes possible to estimate the locations of obstacles. This simple observation is used in this work to build 3D maps of unknown environments in a Bayesian manner. The uncertainty in the GNSS receiver's location, which can become high in dense urban environments, is accounted for by jointly estimating the environment map and the receiver's location.

Contributions: We present a solution to the simultaneous localization and mapping (SLAM) problem using sparsely sampled GNSS data as the only source of measurements. By using a physically motivated nonlinear sensor model, discretizing the environment, and representing uncertainty in GNSS receiver positions as particles we show that the joint posterior probability distribution of the map and receiver positions can be represented by a factor graph. A reduced complexity Loopy Belief Propagation algorithm can then be used to efficiently estimate the probabilities of each cell being occupied or empty, along with the probability of the particles for each receiver location. Finally, we validate our SLAM algorithm on a real-world dataset collected using an off-the-shelf GNSS device.

Related work: Using known 3D maps to improve the positioning accuracy of GNSS devices in cluttered urban environments has been an active research area. Shadow matching (SM) is a technique where NLOS signals are matched with corresponding points of reception in the "shadows" of signal-blocking buildings, thereby constraining the space of possible receiver locations [1], [2]. While this technique has been used to achieve significant localization improvement, especially in urban canyons and heavily cluttered environments, up-to-date 3D maps of urban environments are not always available and are expensive to acquire. Surprisingly, despite the massive amount of location data being gathered by mobile GNSS devices (such as smartphones), the converse problem of building 3D maps by using GNSS signal strength has attracted little attention, with only a handful of papers published on the subject. In [3], [4], GNSS-derived information is used to learn shadows of buildings with respect to different

A.T. Irish, J.T. Isaacs, J.P. Hespanha and U. Madhow are with the Electrical and Computer Engineering Department, University of California, Santa Barbara (UCSB) ({andrewirish, jtisaacs, hespanha, madhow}@ece.ucsb.edu), F. Quitin is with the School of Electrical and Electronic Engineering, Nanyang Technological University, (fquitin@ntu.edu.sg)

This work was supported by the Institute for Collaborative Biotechnologies through grant W911NF-09-0001 from the U.S. Army Research Office. The content of the information does not necessarily reflect the position or the policy of the Government, and no official endorsement should be inferred.

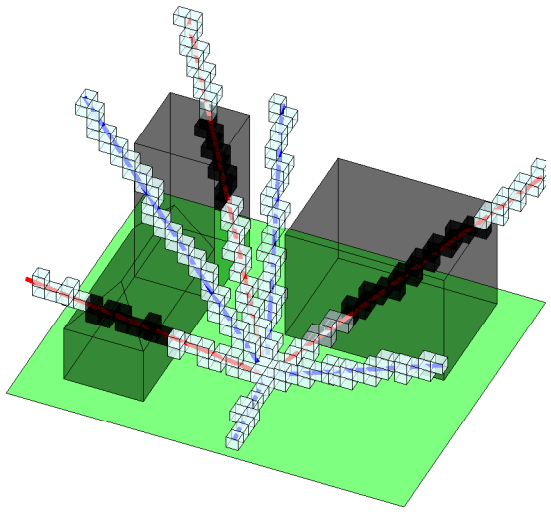


Fig. 1. Example SNR measurement scenario with occupancy grid illustration. Blue/red lines represent LOS/NLOS signal paths to satellites, with NLOS signals characterized by statistically lower SNRs. Light/dark grid cells approximate empty/occupied space.

satellite configurations, after which ray-tracing methods are used to construct environment maps. The presented methods, however, rely on non-probabilistic heuristics resulting in "hard" maps with no notion of uncertainty. As we will show, since open areas are much easier (and better) identified than occupied ones, we believe that capturing such uncertainty in the form of a probabilistic map (and by using a probabilistic approach in general) yields more informative results.

Accordingly, it should be noted that probabilistic techniques are now standard in the mapping and localization literature [5]. For robotic applications, some form of obstacle detection is used, with sensors (for example lidar [6], [7]) mounted on robots and taking obstacle-range readings. As in our problem, it is typical that neither the environment nor the agent's poses are perfectly known, so that the full SLAM problem (see [8] for an introduction) must be considered. A major difference, however, is that our LOS/NLOS sensing scheme only gives likelihoods of whether particular satellite signals are blocked or not, yielding no direct information on obstacle ranges. The assignment of environmental features to sensor readings (i.e., the data association problem [5]) is therefore highly non-trivial in our scenario. To alleviate this problem, we devise a physically-motivated nonlinear sensor model, and represent the environment as an occupancy grid [9] en lieu of sets of "landmarks". We then formulate the posterior distribution of the latent variables (map and poses) as a factor graph, much as in GraphSLAM [10]. However, in our scenario measurement rays observe many (tens to hundreds of) grid cells, thus intertwining widely separated portions of the map and forming large cliques in the graph. Techniques such as GraphSLAM that rely on linearity assumptions and variable elimination are thus impractical if not inapplicable. Instead, we employ a low complexity version of Loopy Belief Propagation [11] to efficiently estimate the marginal distributions of the variables.

II. Belief propagation based SLAM algorithm

Our algorithm relies on ray tracing from sets of hypothetical receiver locations (particles) towards given satellites. Probabilistic "beams" of rays – sets of parallel rays emanating from the same particle set and heading to the same satellite – are then assigned likelihoods of being LOS or NLOS, depending on the measured satellite SNR values. Finally, these beams are stitched together to form a soft probabilistic occupancy map using Loopy Belief Propagation, which concurrently re-weights position particles, yielding revised location estimates as well.

A. Mathematical formulation

We model both the environment and receiver positions non-parametrically: 1) The map is represented as a 3D grid of cells, $m = \{m_i\}_{i=1}^{L}$, with $m_i \in \{0,1\}$ denoting empty and occupied space; 2) The space of possible GNSS receiver trajectories $x = \{x_t\}_{t=1}^{T}$ is represented using sets of particles, so that individual positions are $x_t \in \{x_t^k\}_{k=1}^{K}$. The SLAM problem is then formulated as estimating the marginal distributions of each latent variable m_i and x_t.

1) SNR measurement model: To arrive at the SLAM solution, data gathered and reported by GNSS receivers is used. The first type of information is the satellite SNR measurements, which are noisy and consist of T vector SNR readings, $z = \{z_t\}_{t=1}^{T}$, where $z_t = [z_{t,1}, \ldots, z_{t,N_t}]$, and N_t is the number of satellites in view for the tth data sample. Together with individual SNR readings, the receivers also provide satellite elevations and azimuths $[\theta_{t,n}, \phi_{t,n}]$, which we consider noiseless. Under the assumption of a static world (where the map m does not change over time), the SNR measurements can be modeled as conditionally independent given the map and poses, yielding the following factorization

$$p(z|m,x) = \prod_{t,n} p(z_{t,n}|m,x_t). \tag{1}$$

In reality, the SNR of a given GNSS signal depends on additional factors, including environmental parameters and satellite elevation. To that end, useful statistical models exist for the narrowband Land to Mobile Satellite (LMS) channels of interest, such as those presented in [12], [13]. However, to simplify the computation of messages in our LBP-based inference algorithm, we use the following sensor model

$$p(z_{t,n}|m,x_t^k) = \begin{cases} f_{LOS}(z_{t,n}), & m_i = 0 \quad \forall i \in \mathscr{M}(t,n,k) \\ f_{NLOS}(z_{t,n}), & \text{otherwise} \end{cases} \tag{2}$$

where $\mathscr{M}(t,n,k)$ contains the indices of the cells intersected by the ray starting at particle x_t^k, in the direction of satellite n at time t. In other words, an SNR reading is LOS-distributed if all cells intersected by its associated receiver-satellite ray are empty; otherwise, it is NLOS-distributed. As shown in Figure 2, the SNR under LOS and NLOS hypotheses is modeled using Rician and log-normal distributions, both being common approaches for modeling such wireless channels. Defining the LOS density on the decibel scale, we have after

a change of variables

$$f_{LOS}(r_{dB}) = \frac{\ln 10}{20} 10^{r_{dB}/20} f_{Rice}\left(10^{r_{dB}/20}\right),$$

where

$$f_{Rice}(r) = \frac{2(K_R+1)r}{\widehat{\Omega}} \exp\left(-K_R - \frac{(K_R+1)r^2}{\widehat{\Omega}}\right) \times I_0\left(2\sqrt{\frac{K_R(K_R+1)}{\widehat{\Omega}}}r\right), \quad r \geq 0$$

is the Rician fading density [14], $I_0(\cdot)$ is the 0th order modified Bessel function of the first kind, $\widehat{\Omega}$ is the estimated total channel power, and K_R is the Rician "K factor" (ratio of LOS to diffuse power). As for the NLOS log-normal fading model, in decibels it is simply described by a normal density with mean μ and variance σ^2.

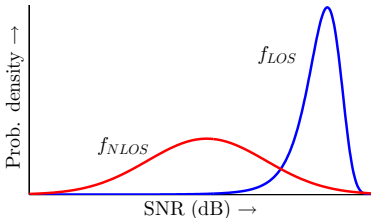


Fig. 2. The LOS/NLOS satellite channels were modeled according to Rician/log-Normal distributions.

2) Positioning error model and particle sampling: The second type of information used are the receiver position estimates (GNSS fixes), which are noisy and modeled as independent Gaussian random variables

$$y_t = x_t + e_t, \quad e_t \sim \mathcal{N}(0, C_t). \tag{3}$$

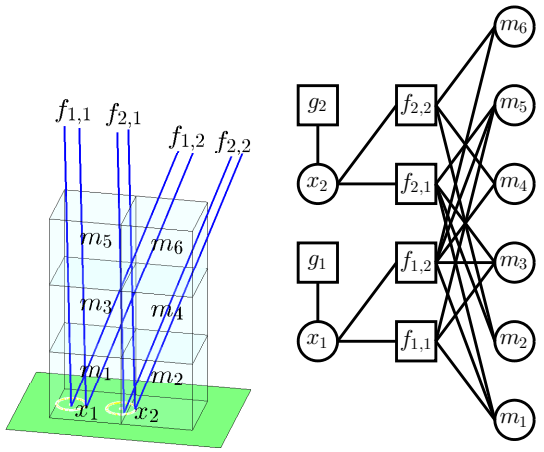
As in Chapter 7 from [15], we estimate the error covariance matrix C_t using the formula for HDOP scaled by the uncertainty reported by the receiver, so that the position fixes do not depend directly on the state of the map. Given the position uncertainties, the particles $\{x_t^k\}$ are then sampled according to $\mathcal{N}(y_t, C_t)$ and assigned equal weights. Note that, since (3) assumes the pose errors are uncorrelated, while in reality consecutive GNSS fixes are known to be correlated, it is important that the input data is sparsely sampled in time.

3) The map posterior: Assuming no a-priori information on the map and poses (such as information on building locations or a motion model governing x), and making use of (1),(3), the posterior distribution of the latent variables given the measurements factorizes as follows

$$\begin{aligned} p(m,x|y,z) &\propto p(y,z|m,x) = p(z|m,x)p(y|m,x,z) \\ &= \prod_{t,n} p(z_{t,n}|m,x_t) \cdot \prod_t p(y_t|x_t). \end{aligned} \tag{4}$$

B. Solving the SLAM problem

The overarching idea of this paper is that the factorization (4) represents a factor graph on which the Loopy Belief Propagation algorithm can be used to perform the SLAM computation.



(a) Measurement scenario (b) Factor graph

Fig. 3. Simplified measurement scenario with $T = N_1 = N_2 = K = 2$ and its corresponding factor graph, with circles/squares representing variable/factor nodes.

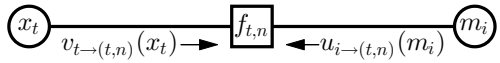
1) The factor graph: A factor graph $G = (\{X, F\}, E)$, with variable nodes X, factor nodes F, and edges E, describes the factorization of a global function into a product of local functions (factors). It is a bipartite graph where an edge between a factor node and a variable node appears if and only if the variable is an argument of the factor. Referring to Figure 3 as an example, in our graph there are two classes of variables and factors, $X = \{x_t\} \bigcup \{m_i\}$ and $F = \{f_{t,n}\} \bigcup \{g_t\}$. Each "SNR measurement" factor node $f_{t,n}$ corresponds to $p(z_{t,n}|m,x_t)$, and each "receiver position" factor g_t is associated with $p(y_t|x_t)$.

While each g_t is singly connected to x_t, the edges between SNR measurements and map cells in the factor graph are only known after ray tracing. The indices of the $\{m_i\}$ adjacent to $f_{t,n}$ are $\mathcal{M}(t,n) = \bigcup_k \mathcal{M}(t,n,k)$. Physically speaking, $\mathcal{M}(t,n)$ describes the set of cells sensed by the "beam" of parallel rays emanating from hypothetical receiver positions $\{x_t^1, \ldots, x_t^K\}$ in the direction of satellite n.

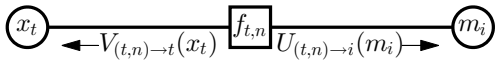
2) Low complexity LBP message passing: As can be seen in Figure 3, the factor graph representing the posterior contains cycles. An efficient algorithm used for approximate inference on such graphs, which has also been used for SLAM [16], is the Loopy Belief Propagation (LBP) (or Sum-Product) algorithm [11]. Essentially, LBP is a message passing (MP) algorithm whereby messages are exchanged locally along edges of the factor graph until convergence (assumed but not guaranteed in loopy graphs). In our implementation, we use synchronous MP, as shown in Figure 4.

At the variable nodes, computing the outgoing messages is simple. The first step in arriving at them is to compute the variables' beliefs, which are also used to determine convergence (see Section II-B.3). For nodes m_i and x_t, on their respective domains ($m_i \in \{0,1\}$ and $x_t \in \{x_t^1, \ldots, x_t^K\}$) the beliefs are given by

$$b_i(m_i) \propto \prod_{(t,n)\in\mathcal{F}(i)} U_{(t,n)\to i}(m_i), \qquad b_t(x_t) \propto \prod_{n=1}^{N_t} V_{(t,n)\to t}(x_t),$$



(a) Step 1: Send all messages from variables to factors



(b) Step 2: Send all messages from factors to variables

Fig. 4. Illustration of one iteration of synchronous MP. Note that, because g_t is singly connected and all particles are initialized with the same weights, g_t continually sends uninformative "flat" messages to x_t, with the effect that g_t can be ignored in the MP framework.

where $U_{(t,n)\to i}$, $V_{(t,n)\to t}$ are incoming messages from $f_{t,n}$, $\mathscr{F}(i) = \{(t,n) : i \in \mathscr{M}(t,n)\}$ indexes the $\{f_{t,n}\}$ neighboring m_i, and the beliefs are normalized to sum to one. These nodes' outgoing messages to factor node $f_{t,n}$ can then be written as

$$u_{i\to(t,n)}(m_i) \propto \frac{b_i(m_i)}{U_{(t,n)\to i}(m_i)}, \qquad v_{t\to(t,n)}(x_t) \propto \frac{b_t(x_t)}{V_{(t,n)\to t}(x_t)},$$

which are also normalized to sum to one after computation.

In the other direction, computing the factor-to-variable messages is more complicated. For example, naively applying (6) in [11] to the calculation of the messages from $f_{t,n}$ to x_t, and evaluating at $x_t = x_t^k$, yields

$$V_{(t,n)\to t}(x_t^k) = \sum_m p(z_{t,n}|m, x_t^k) \prod_{j\in\mathscr{M}(t,n)} u_{j\to(t,n)}(m_j),$$

which involves the summation of $2^{|\mathscr{M}(t,n)|}$ terms. Since $|\mathscr{M}(t,n)|$ (the degree of $f_{t,n}$) counts the number of cells intersected by SNR measurement (t,n), which in our setup is often in the hundreds, directly evaluating the above expression is unfeasible. However, upon substitution of the binary sensor model (2), due to the normalization of the incoming messages the above expression simplifies to

$$V_{(t,n)\to t}(x_t^k) = \gamma_{t,n}^k f_{LOS}(z_{t,n}) + \left(1 - \gamma_{t,n}^k\right) f_{NLOS}(z_{t,n}),$$

where

$$\gamma_{t,n}^k = \prod_{j\in\mathscr{M}(t,n,k)} u_{j\to(t,n)}(0),$$

which is of *linear complexity* in $|\mathscr{M}(t,n,k)|$. Likewise, for the message from $f_{t,n}$ to m_i we initially have the (even more) complicated expression

$$U_{(t,n)\to i}(m_i) = \sum_{m\setminus m_i} \sum_k p(z_{t,n}|m, x_t^k) \times v_{t\to(t,n)}(x_t^k) \prod_{j\in\mathscr{M}(t,n)\setminus i} u_{j\to(t,n)}(m_j)$$

However, again by consequence of the binary sensor model (2), the above expression reduces to

$$U_{(t,n)\to i}(m_i) = \alpha_{t,n,i} + \beta_{t,n,i}(m_i)$$

with

$$\alpha_{t,n,i} = \sum_{k\notin\mathscr{K}(t,n,i)} v_{t\to(t,n)}(x_t^k) \times \left\{\gamma_{t,n}^k f_{LOS}(z_{t,n}) + \left(1 - \gamma_{t,n}^k\right) f_{NLOS}(z_{t,n})\right\},$$

$$\beta_{t,n,i}(0) = \sum_{k\in\mathscr{K}(t,n,i)} v_{t\to(t,n)}(x_t^k) \times \left\{\gamma_{t,n,i}^k f_{LOS}(z_{t,n}) + \left(1 - \gamma_{t,n,i}^k\right) f_{NLOS}(z_{t,n})\right\},$$

$$\beta_{t,n,i}(1) = f_{NLOS}(z_{t,n}) \sum_{k\in\mathscr{K}(t,n,i)} v_{t\to(t,n)}(x_t^k),$$

where

$$\gamma_{t,n,i}^k = \gamma_{t,n}^k / u_{i\to(t,n)}(0),$$

and $\mathscr{K}(t,n,i) = \{k : i \in \mathscr{M}(t,n,k)\}$ maintains a list of which particles among $\{x_t^1, \ldots, x_t^K\}$ observe cell m_i when looking at the nth satellite.

3) LBP convergence: In this paper, convergence of LBP is declared when the mean of all variables' belief residuals falls below a predefined threshold $\bar{R}_{th}$. To limit oscillations and help ensure that LBP converges, as is common in practice we apply message damping with damping factor $\rho \in [0,1)$ (see [17] for a detailed discussion on the subject). The belief residuals of m_i and x_t are defined via the L_1 norm, as in [18] (4.1):

$$R_i \triangleq \sum_{m_i=0}^{1} \left|b_i(m_i) - b_i'(m_i)\right|, \quad R_t \triangleq \sum_{k=1}^{K} \left|b_t(x_t^k) - b_t'(x_t^k)\right|$$

where b_i', b_t' are the beliefs from the previous iteration. Upon convergence, the approximate SLAM solution is simply taken to be the beliefs of all the latent variables, i.e., the marginal posteriors of the map and poses are estimated as $p(m_i|y,z) \approx b_i(m_i)$ and $p(x_t|y,z) \approx b_t(x_t)$.

III. Experiments and Results

To verify the efficacy of our proposed algorithm, we set out to map the eastern corner of the University of California, Santa Barbara campus (see Figure 5). A Samsung Galaxy Tablet 2.0 running the Android operating system was used as a GPS/GLONASS collecting device. Over over the course of several days we gathered a total of 34 test datasets, where the duration of each test ranged from 3 to 23 minutes and measurement data was logged at 1 Hz. During a typical test SNR readings in the range of $7-48$ dB were recorded and approximately 13 satellites were visible at any one time.

To summarize the parameters used, a grid size of 5 m was selected, and the map height was set to 30 m. For the Rician channel hypothesized under LOS conditions, $\widehat{\Omega}$ for each SNR reading was to the maximum of all linear SNR readings $10^{z_{t,n}/10}$ from the same satellite during the same time window. The Rician K factor was set constant to $K_R = 2$, for simplicity, indicating moderate fading conditions. Guided roughly by the results in [13], [19], for the log-normal (NLOS) distribution we set μ to 18 dB below the reference power level $\widehat{\Omega}$, and $\sigma = 10$ dB, allowing for a large variability in shadowing conditions. When visibility to a particular satellite was temporarily lost in the middle of an observation window (presumably most often caused by total occlusion), the satellite coordinates were interpolated and LOS/NLOS likelihoods of $0.1/0.9$ were assigned. Because the received signal strength from low elevation satellites is known to vary more widely [19] (thus producing more

Fig. 5. Google Maps aerial view of the eastern portion of the University of California, Santa Barbara campus. Two typical receiver paths as logged by the GNSS collecting device are shown in yellow (shorter test run) and magenta (longer test run).

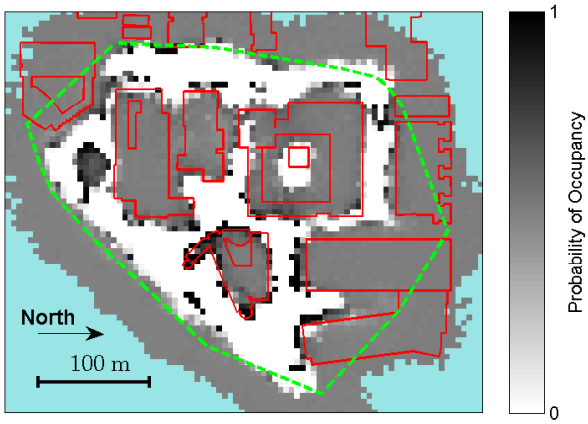
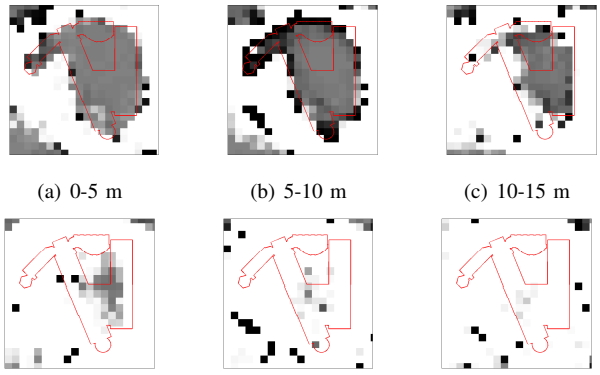


Fig. 6. Horizontal slice (5-10 m above ground level) of generated map covering the eastern portion of the University of California, Santa Barbara campus. The measurement region borders are marked in green, and building contours obtained from OSM are shown in red.



(a) 0-5 m (b) 5-10 m (c) 10-15 m

(d) 15-20 m (e) 20-25 m (f) 25-30 m

Fig. 7. Horizontal slices of the generated map around Kohn Hall at UCSB.

"noisy" SNR readings), measurements from satellites with elevations below 10° were discarded, similar to [3].

In all, the 34 datasets comprised of 5.0×10^3 position samples and 6.5×10^4 SNR measurements interacting with 4.2×10^4 grid cells. To mitigate the effects of spatially correlated geolocation errors, the 5.0×10^3 position samples represent a factor of 5 downsampling of the original datasets. Although in our experiment data was essentially thrown away by downsampling, in a real-world application sparse sampling would allow for potential energy savings at the GNSS receiver. To represent the possible receiver positions for each time sample, we used $K = 30$ quasi-random particles (drawn from a Sobol sequence [20] which was then transformed to a set of Gaussian samples). Compared to drawing independent samples, low discrepancy sampling more uniformly separated the particles, allowing us to use smaller K. For the LBP-based inference step, $\rho = 0.4$ damping and a convergence threshold of $\bar{R}_{th} = 10^{-3}$ were used. With these settings LBP terminated after 60 iterations, taking 13 minutes on a 64 bit PC with a 3.20 GHz Intel Core i7 processor and 32 GB of RAM running MATLAB R2013b. During runtime, total system memory usage never exceeded 6 GB.

A portion of the resulting map – that is, the unprocessed LBP output – can be seen in Figure 6. The dark areas are those where the probability of occupancy is close to one, the white areas represent likely open-space, and the light blue regions are unobserved. The green border encloses the feasible mapping region (borders the convex hull of all position samples). Data from OpenStreetMap was used to provide building contours, shown in red. Although the map includes errors, it closely resembles the aerial view in Figure 5. Note that several dark spots can be seen outside of buildings (e.g., the dark spot on the south side of the southernmost building); those often correspond to trees, as can be seen from the aerial view. One interesting aspect of our mapping problem can be clearly noticed, however: open areas are much better mapped than occupied ones. The reason is that, while a LOS signal is observed if *all* cells it passes through are empty, a NLOS reading only informs that *some* cell(s) occluded it. In our LBP-based algorithm, when a NLOS signal penetrates a large occupied region, each intersected cell in the region will have a tendency to "blame" (via message passing) other cells for the occlusion. The end result is that "gray zones" appear around large buildings, while open space is more clearly marked in white. We suspect that further processing of the generated map (using, e.g., thresholding) may improve the map quality, but such methods are beyond the scope of this paper.

To assess the 3D mapping capability of our algorithm, the first 6 layers of the resulting map are shown in Figure 7 for the area around Kohn Hall, which is approximately 10 meters in height. As can be seen in Figure 7(a) and 7(b), the outline of the building is well approximated by the map. Additionally, the $15-20$ meter horizontal layer of the map and above, shown in Figure 7(d)–7(f), correctly represent open space. The lingering gray area in Figure 7(c) can be explained by the fact that no LOS satellite signals passed through that portion of the map. Note, however, that any

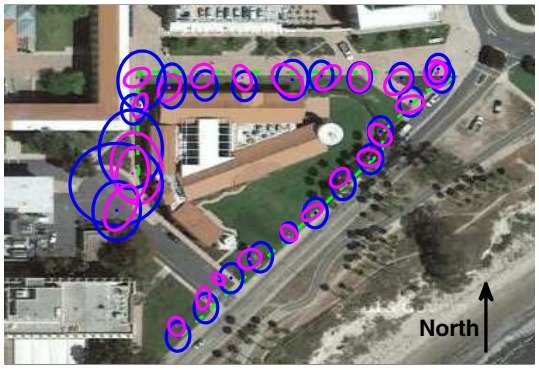


Fig. 8. Positioning improvement around Kohn Hall. The true receiver path is marked with the dashed green line, with blue/magenta ellipses representing positioning uncertainty before/after LBP.

ground-based sensor platform would likely suffer from this problem as well.

Finally, an example of the localization improvement can be seen in Figure 8, which shows a portion of a dataset isolated for analysis. While positioning uncertainty is lower after LBP in general, of particular interest are the points along the west side, north-west corner, and north side of the building. The proposed algorithm assigns low weights to particles that are in the building, and more highly weights those on the sidewalk near the true path, reducing uncertainty and improving overall geolocation accuracy. Without this position correction, the resulting map would have underestimated the occupancy of the cells on the northern wall of the building. Separately, we note that receiver positioning on the south side of the building on the sidewalk is more or less unimproved by our algorithm. This is to be expected because the positioning uncertainty there was lower to begin with (since fewer signal-occluding buildings are nearby).

IV. Conclusion

A novel approach to SLAM based on sparsely sampled GNSS SNR readings was presented and validated on a real-world dataset. The environment is represented as an occupancy grid and the receiver positions are approximated by sets of particles. In essence, our approach relies on ray tracing to satellites from potential receiver positions (particles), and assigning sets of parallel rays likelihoods of being LOS/NLOS depending on the corresponding SNR. Afterwards, these rays are stitched together to form a soft probabilistic map using Bayesian Belief Propagation, which concurrently refines position estimates as well.

This work is one of the first steps in a broader effort on GNSS SNR based mapping and localization improvement. Several research opportunities are: employing asynchronous BP algorithms to perform real-time localization and mapping; developing more advanced LOS/NLOS sensor models with learned parameters; experimenting with much larger crowd-sourced datasets, and implementing the proposed approach in a cloud computing framework; and modifying the algorithm to support advanced motion models for more densely sampled GNSS data.

Acknowledgement

The authors would also like to acknowledge Adam Ehrlich for developing the Android application used to log GPS data.

References

[1] M. Obst, S. Bauer, P. Reisdorf, and G. Wanielik, "Multipath detection with 3D digital maps for robust multi-constellation GNSS/INS vehicle localization in urban areas," in *Proc. of the Intelligent Vehicles Symposium.*, 2012, pp. 184–190.

[2] L. Wang, P. D. Groves, and M. K. Ziebart, "Shadow matching: Improving smartphone GNSS positioning in urban environments," in *Proc. of the China Satellite Navigation Conference.*, 2013, pp. 613–621.

[3] K. Kim, J. Summet, T. Starner, D. Ashbrook, M. Kapade, and I. Essa, "Localization and 3D reconstruction of urban scenes using GPS," in *Proc. of the IEEE International Symposium on Wearable Computers.*, 2008, pp. 11–14.

[4] A. Weissman, B. Ben-Moshe, H. Levi, and R. Yozevitch, "2.5D mapping using GNSS signal analysis," in *Proc. of the Workshop on Positioning Navigation and Communication*, 2013, pp. 1–6.

[5] S. Thrun, "Robotic mapping: A survey," *Exploring Artificial Intelligence in the New Millennium*, vol. 1, pp. 1–35, 2003.

[6] S. Thrun, W. Burgard, and D. Fox, "A real-time algorithm for mobile robot mapping with applications to multi-robot and 3D mapping," in *Proc. of the International Conference on Robotics and Automation.*, vol. 1, 2000, pp. 321–328.

[7] A. Howard, D. F. Wolf, and G. S. Sukhatme, "Towards 3D mapping in large urban environments," in *Proc. of the International Conference on Intelligent Robots and Systems.*, vol. 1, 2004, pp. 419–424.

[8] S. Thrun, "Simultaneous localization and mapping," in *Robotics and cognitive approaches to spatial mapping*. Springer, 2008, pp. 13–41.

[9] A. Elfes, "Using occupancy grids for mobile robot perception and navigation," *Computer*, vol. 22, no. 6, pp. 46–57, 1989.

[10] S. Thrun and M. Montemerlo, "The GraphSLAM algorithm with applications to large-scale mapping of urban structures," *International Journal on Robotics Research*, vol. 25, no. 5/6, pp. 403–430, 2005.

[11] F. Kschischang, B. Frey, and H.-A. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Trans. on Information Theory*, vol. 47, no. 2, pp. 498–519, 2001.

[12] A. Abdi, W. C. Lau, M.-S. Alouini, and M. Kaveh, "A new simple model for land mobile satellite channels: first-and second-order statistics," *IEEE Trans. on Wireless Communications*, vol. 2, no. 3, pp. 519–528, 2003.

[13] C. Loo, "A statistical model for a land mobile satellite link," *IEEE Trans. on Vehicular Technology*, vol. 34, no. 3, pp. 122–127, 1985.

[14] A. Abdi, C. Tepedelenlioglu, M. Kaveh, and G. Giannakis, "On the estimation of the K parameter for the Rice fading distribution," *IEEE Communications Letters*, vol. 5, no. 3, pp. 92–94, 2001.

[15] E. D. Kaplan and C. J. Hegarty, *Understanding GPS: Principles and Applications*, 2nd ed.

[16] A. Ranganathan, M. Kaess, and F. Dellaert, "Loopy SAM," in *Proc. of the International Joint Conference on Artificial Intelligence.*, 2007, pp. 6–12.

[17] K. P. Murphy, Y. Weiss, and M. I. Jordan, "Loopy belief propagation for approximate inference: An empirical study," in *Proc. of the Fifteenth Conference on Uncertainty in Artificial Intelligence*, 1999, pp. 467–475.

[18] J. E. Gonzalez, Y. Low, C. Guestrin, and D. O'Hallaron, "Distributed parallel inference on large factor graphs," in *Proc. of the Conference on Uncertainty in Artificial Intelligence*, 2009, pp. 203–212.

[19] F. Perez-Fontan, M. Vazquez-Castro, S. Buonomo, J. P. Poiares-Baptista, and B. Arbesser-Rastburg, "S-band LMS propagation channel behaviour for different environments, degrees of shadowing and elevation angles," *IEEE Trans. on Broadcasting*, vol. 44, no. 1, pp. 40–76, 1998.

[20] H. Niederreiter, "Low-discrepancy and low-dispersion sequences," *Journal of number theory*, vol. 30, no. 1, pp. 51–70, 1988.