

Autonomous Exploration and Mapping of Abandoned Mines

Sebastian Thrun², Scott Thayer¹, William Whittaker¹, Christopher Baker¹
Wolfram Burgard³, David Ferguson¹, Dirk Hähnel³, Michael Montemerlo², Aaron
Morris¹, Zachary Omohundro¹, Charlie Reverte¹, Warren Whittaker¹

¹ Robotics Institute, Carnegie Mellon University, Pittsburgh, PA

² Computer Science Department, Stanford University, Stanford, CA

³ Computer Science Department, University of Freiburg, Freiburg, Germany

Abstract

Abandoned mines pose significant threats to society, yet a large fraction of them lack accurate maps. This article discusses the software architecture of an autonomous robotic system designed to explore and map abandoned mines. We have built a robot capable of autonomously exploring abandoned mines. A new set of software tools is presented, enabling robots to acquire maps of unprecedented size and accuracy. On May 30, 2003, our robot “Groundhog” successfully explored and mapped a main corridor of the abandoned Mathies mine near Courtney, PA. The article also discusses some of the challenges that arise in the subterranean environments, and some of the difficulties of building truly autonomous robots.

1 Introduction

In recent years, the quest to find and explore new, unexplored terrain has led to the deployment of more and more sophisticated robotic systems, designed to traverse increasingly remote locations. Robotic systems have successfully explored volcanoes [5], searched meteorites in Antarctica [1, 44], traversed deserts [3], explored and mapped the sea bed [12], even explored other planets [26]. This article presents a robot system designed to explore spaces much closer to us: abandoned underground mines.

According to a recent survey [6], “tens of thousands, perhaps even hundreds of thousands, of abandoned mines exist today in the United States. Not even the U.S. Bureau of Mines knows the exact number, because federal recording of mining claims was not required until 1976.” Shockingly, we are unaware of the location of many mines—despite the fact that most mines were built just a few generations ago! A recent near-fatal accident in Somerset, PA, speaks to this end: When miners in their routine work accidentally breached a nearby abandoned mine, fifty million gallons of water poured upon them, cutting off nine miners and almost burying them alive. The cause of this accident was officially determined to be a lack of accurate mine maps; the breached and flooded mine had been suspected to be several hundred feet away [35].

Even if accurate mine maps exist, those are usually just idealized 2-D drawings. Little can be inferred from such sketches with regards to critical measures, such as the volume and the



Figure 1: Left: The Groundhog robot is a 1,500 pound custom-built vehicle equipped with onboard computing, laser range sensing, gas and sinkage sensors, and video recording equipment. Right: Testing the system inside the Bruceon Research Mine, a well-maintained mine accessible to research teams.

structural soundness of an abandoned mine. Accurate models of such abandoned mines would be of great relevance to a number of problems that directly affect the people who live or work near them. One is subsidence: structural shifts can cause collapse on the surface above. Ground water contamination is another problem of great importance, and knowing the location, volume, and condition of an abandoned mine can be highly informative in planning and performing interventions. Accurate volumetric maps are also of great commercial interest. Knowing the volume of the material already removed from a mine is of critical interest when assessing the profitability of re-mining a previously mined mine.

Abandoned mines are usually not accessible to people. Lack of structural soundness is one reason; another is the harshness of the environment (e.g., low oxygen levels, flooding) and the danger of explosion of methane, a gas that frequently accumulates when mines are no longer ventilated. This makes mine a superb target domain for autonomous robots. However, mapping a mine with a robotic vehicle is a challenge. The vehicle must be rugged enough to survive the harsh environmental conditions inside the mine. It must be able to perceive and negotiate major obstacles. Unfortunately, existing wireless communication techniques are generally unfit for mines.

This article reports experiments with a robotic system designed to autonomously explore and acquire 3-D maps of abandoned mines. The 1,500 pound vehicle, nicknamed “Groundhog” and shown in Figure 1; a detailed description of the hardware can be found in [2]. Groundhog is essentially built out of the front halves of two ATVs, endowing it with identical steering mechanisms on either end. While the exact configuration of the robot varied from experiment to experiment, in its final configuration Groundhog was essentially symmetrical, enabling it to retract without having to turn around. For acquiring 3-D maps, Groundhog is equipped with tiltable SICK laser range finders on either end. It also carries two mine-certified portable gas detectors to enable it to detect methane and other combustible gases. To navigate, Groundhog analyzes local 3-D scans with regards to traversability. A fast C-space planner determines whether the terrain ahead can be negotiated, and if so, identifies suitable paths. Those are then



Figure 2: On October 27, 2002, Groundhog was deployed into the Florence mine near Pittsburgh, PA. This experiment is a first in a series in which the robot navigates an environment inaccessible to people, but still under remote control.

executed via PD control, using fast 2-D scan matching to keep the vehicle localized. Failure to find a suitable path leads Groundhog to retract in reverse motion. Groundhog also acquires large-scale consistent maps of the voids it explores.

Groundhog's development began in the Fall of 2002. Approximately a dozen test runs were carried out in a well-maintained inactive coal mine accessible to people: the Bruceton Research Mine located near Pittsburgh, PA. Mining was discontinued in the early 1940s, but since that time the mine had been maintained in a state safe for people to enter. The mine features hallways several hundred meters long, putting to a test the vehicle's physical endurance and its ability to manage large amounts of data. However, this mine is technically not abandoned and therefore not subject to collapse and deterioration. On October 27, 2002, Groundhog descended for the first time into an abandoned mine inaccessible to people. This mine, the Florence Mine near Pittsburgh, PA, had been abandoned and flooded for many decades. Before the robot's entry, the mine was mostly drained, leaving behind acidic mud that miners refer to as "yellow boy." Figure 2 depicts the vehicle after descending approximately 30 meters into the mine, here operating on a tether and under remote control. On May 30, 2003, after a long series of test runs carried out in the Bruceton Research Mine, Groundhog finally entered an inaccessible abandoned mine in fully autonomous mode. The mine is known as the Mathies mine and is located in the same geographic area as the other mines. The core of this surface-accessible mine consists of two 1.5-kilometer long corridors which branch into numerous side corridors, and which are accessible at both ends. This was an important feature of this mine, as it provided natural ventilation and thereby reduced the chances of encountering combustible gases inside the mine. Figure 3 depicts both ends of the mine. A map of the mine, provided to us by the Mine Safety and Health Administration and the mine owner, is shown in Figure 4; apparently this is the most accurate map on record for this mine. To acquire a more accurate map of one of the main corridors, the robot was programmed to autonomously navigate through the corridor. 308 meters into the mine, the robot encountered a broken ceiling bar draping diagonally across its path. The robot made the correct decision to retract.

The data acquired on these runs has provided us with models of unprecedented detail and accuracy, of subterranean spaces that may forever remain off limits for people. This article



Figure 3: On May 30, 2003, Groundhog enters the Mathies mine near Courtney, PA. For the first time, Groundhog operates in fully autonomous mode, beyond the reach of our wireless communication link. It autonomously descended 308 meters into the mine before making the correct decision to turn around after sensing a non-negotiable ceiling beam draped diagonally across the robot's path. These photographs show the two entrances to the mile-long corridor system.

provides a comprehensive description of Groundhog's software architecture. It offers visualizations of 2-D and 3-D maps of some of the mines mapped by the vehicle. We discuss some of the shortcomings of the present system and lay out a roadmap to future research based on the challenges that remain.

2 Chassis and Electronics

Groundhog's chassis unites the front halves of two all terrain vehicles, allowing all four of Groundhog's wheels to be both driven and steered. The two Ackerman steering columns are linked in opposition, reducing Groundhog's outside turning radius to approximately 2.44m. A hydraulic cylinder drives the steering linkage, with potentiometer feedback providing closed-loop control of wheel angle. Two hydraulic motors coupled into the front and rear stock ATV differentials via 3:1 chain drives result in a constant 28.5 0.145 m/sec velocity. When in motion, Ground hog consumes upwards of 1kW, where processing and sensing only draw 25W and 75W respectively. Therefore, time spent sensing and processing has minimal impact on the operational range of the robot. The high power throughput combined with the low speed of the robot means that Groundhog has the torque necessary to overcome the railways, fallen timbers, and other rubble commonly found in abandoned mines. Equipped with six deep-cycle lead-acid batteries, and in later experiments with eight such batteries, Groundhog has a locomotive range greater than 3km.

Mine safety regulations require that all electronics either be intrinsically safe or be encased in an explosion proof enclosure. An intrinsically safe device may not, through capacitance or inductance, discharge enough energy into a spark to ignite an explosive atmosphere. Groundhog's enclosure is designed to prevent an interior explosion from transferring enough energy to the external environment to trigger an explosion of that environment. To satisfy this requirement, Groundhog was fitted with a 225kg steel enclosure. To compensate for this load and keep the ground clearance of the robot above 25cm, the suspension was re-mounted in a precompressed configuration.



Figure 4: The best existing map on record of the Mathies mine. It shows at the center two parallel vertical corridors, of which the robot entered the left corridor from the supply yard end shown towards the bottom. This map is obviously incorrect: the two corridors do not connect properly towards the top of this map.

The explosion-proof enclosure houses a 24 VDC, 90 Amp electric motor that drives the pump for the hydraulic system. The 300 MHz PC/104+ CPU and associated I/O electronics also occupy the enclosure along with the hydraulic manifold and its six solenoid actuators. All outgoing power lines are computer-controlled and individually fused on both the positive and negative terminals. In addition to being able to explicitly cut power to external devices Groundhog's CPU is equipped with a watchdog timer that automatically disables external power in the event of a computer failure.

Coal mine corridors average 1–2, tall and 3–6m wide. At 1m tall and 1.2m wide, Groundhog is able to operate in all but the shortest of coal mines with room to maneuver in all but the thinnest. The original dimensions of Groundhog were engineered to accommodate the constraints at the breach between the Quecreek and Saxman mines. This breach was 2m wide by 1.2m tall, 1.2m deep, with a 30cm step on either side. While Groundhog was denied the chance to explore this breach, the configuration and dimensions chosen have proven effective in the experiments reported here.

In groundhog's landmark exploration of the Matthies mine—which was our only experiment so far in which the robot was truly autonomous, the worst-case scenario was for Groundhog to traverse almost the whole 1.5km, then have to traverse an additional 1.5km back out of the mine, for a total of 3km, which was determined to be within the operational range of the robot.

Periodic messages were sent to the base of operations via a simple UDP broadcast message over an 802.11 wireless link, indicating that the robot was alive and working.

3 Simultaneous Localization and Mapping

The core of the Groundhog navigation system is comprised of a software package that solves the SLAM problem by acquiring 2-D maps. The SLAM problem—which is short for simultaneous localization and mapping [11]—arises when a vehicle attempts to build a map while simultaneously localizing itself relative to this map. On the surface, mobile robots can often utilize GPS to acquire absolute position information. Underground, we do not have the luxury of GPS localization.

At the lowest level of processing, Groundhog’s mapping system utilizes a real-time scan matching technique for registering consecutive scans [16, 19]. Scans are acquired using laser range finder pointed forward. As is common in the scan registration literature, our algorithm aligns scans by iteratively identifying nearby points in pairs of consecutive range scans. It then calculates the relative displacement and orientation of these scans by minimizing the quadratic distance of all pairs of points [7]. In our implementation, all calculations are carried out in real-time at 75 Hertz, the data rate of the SICK scanner (see [19]).

The scan matcher enables Groundhog to recover two quantities: locally consistent maps and an estimate of the robot’s own motion. Figure 5a shows a 2-D map obtained using our scan matching algorithm from a data set lacking any odometry information. It is well-understood that local scan matching is incapable of achieving global consistency [8, 16, 40]. This is because of the residual error in scan matching, which accumulates over time.

The limitation is apparent in the 2-D map of the Bruceton Research Mine shown in Figure 5a. This map is the result of applying local scan matching in a mine that is approximately 250 by 180 meters in size. While parts of this map are locally consistent, the map is globally inconsistent: Several of the hallways traversed more than once have falsely been mapped into parallel corridors. A consistent map is shown in Figure 5b and its creation will be described in turn.

Our approach addressing the SLAM problem is described in depth in [14], with a previous versions described in [42]. It is in many ways similar to a seminal paper by Lu and Milios [24] and research in [31, 43], in that essential map information is represented by relative constraints. In particular, every five meters of consecutive robot motion, the data gathered during this period is mapped into a local map, just as in [8]. Figure 6a shows such a local map, along with a range scan.

As common in the literature, let us denote the absolute location and orientation of the k -th map by $\xi_k = (x_k \ y_k \ \theta_k)^T$; here x and y are the Cartesian coordinates of the map and θ is its orientation. The set of coordinates for all local maps will be denoted $\mathcal{X} = \{\xi_1, \xi_2, \dots\}$. The graph of these local coordinates as obtained from the scan matcher is shown in Figure 6b. Clearly, if only we knew the correct coordinates \mathcal{X} of all local maps, it would be straightforward to paste them together into a single global mine map using, for example, occupancy grid mapping techniques [28]. However, the true values of ξ_k are not known.

What we do know, however, is the approximate relative displacement between consecutive

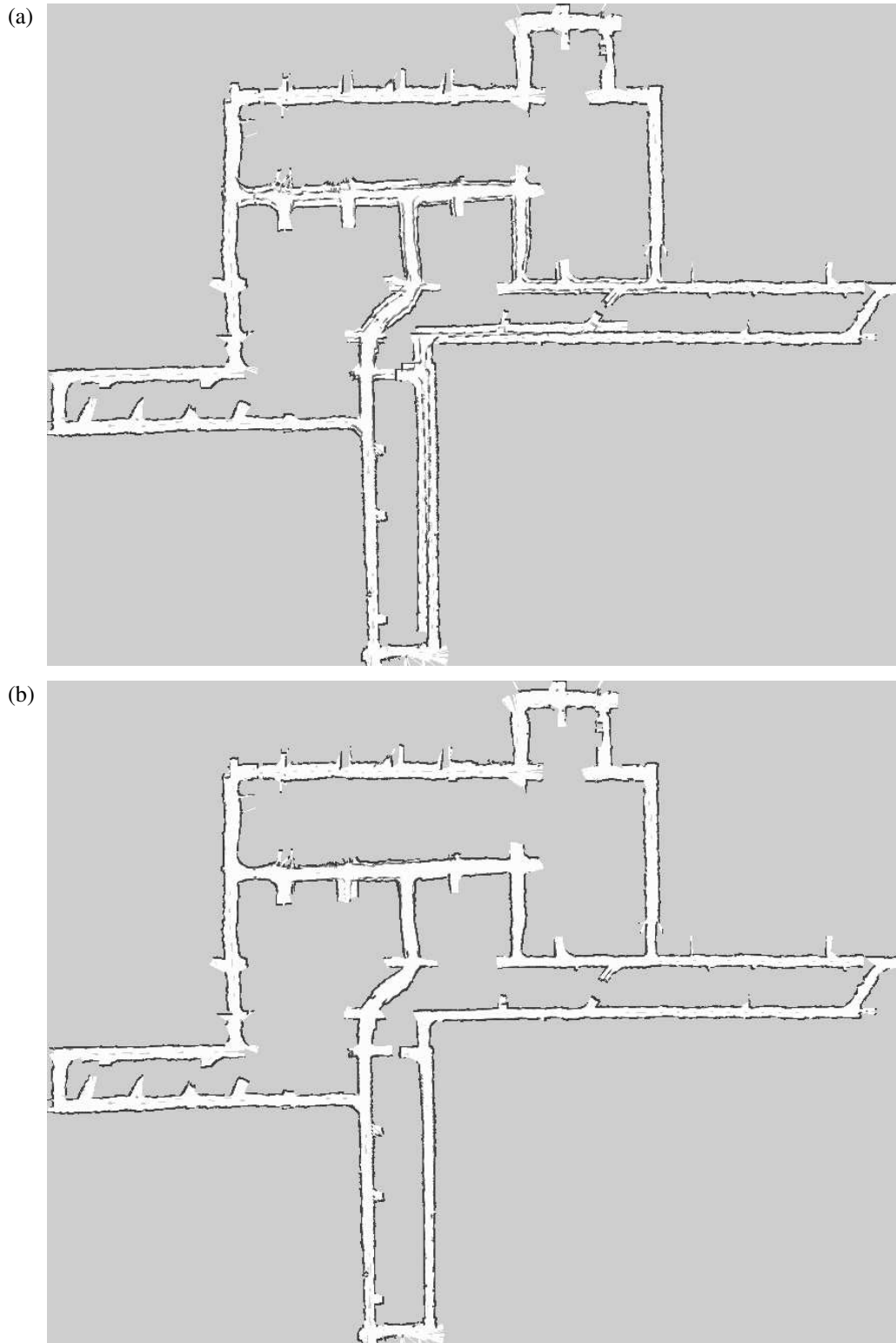


Figure 5: Map of the Bruceton Research mine, obtained via incremental scan matching (left) and using our lazy data association approach (right). The map is approximately 250 by 180 meters in size.

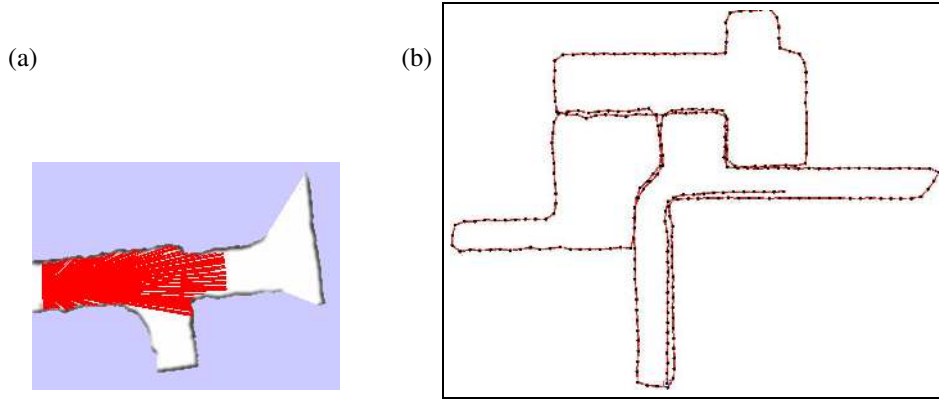


Figure 6: (a) Example of a local map and a single 2-D range scan. (b) The resulting Markov random field: Each node is the center of a local map, acquired when traversing the Bruceton Research Mine near Pittsburgh, PA.

maps ξ_k and ξ_{k-1} . This information is recovered from the scan matcher. It is of the form $\delta_{k,k-1} = (\Delta x_{k,k-1} \ \Delta y_{k,k-1} \ \Delta \theta_{k,k-1})^T$, where the individual delta values measure the relative displacement along the coordinate axes. The Δ -values map coordinates ξ_{k-1} into ξ_k via the obvious trigonometric function $\xi_k = f(\xi_{k-1}, \delta_{k,k-1})$. If scan matching was free of errors, this recursion would enable us to recover absolute information via the following recursion, assuming we know the initial coordinate ξ_0 . However, scan matching is not without errors. To account for those errors, our approach generalizes this recursion into a sequence of soft “rubber-band”-type constraints that assumes Gaussian error. More specifically, ξ_{k-1} and $\delta_{k,k-1}$ induce a Gaussian probability distribution over ξ_k with covariance Σ : $f(\xi_{k-1}, \delta_{k,k-1})$

$$\phi(\xi_k, \xi_{k-1}) = |\pi\Sigma|^{-\frac{1}{2}} \exp -\frac{1}{2}(\xi_k - f(\xi_{k-1}, \delta_{k,k-1}))^T \Sigma^{-1} (\xi_k - f(\xi_{k-1}, \delta_{k,k-1})) \quad (1)$$

Functions like ϕ are often called *potential* in the statistical literature. Potentials link together consecutive local maps in a soft way: The expectation of the pose ξ_k is equivalent to the result of the scan matcher, but the potential allows for deviations from this expectation. Such a rubber-band representation is known as Markov random fields (MRF) [36, 46]. It is reminiscent of information filters, as previously used for SLAM in [31, 43].

Recovering the global map is equivalent to finding the sequence of map coordinates \mathcal{X} that minimizes the product of potentials $\prod_k \phi(\xi_k, \xi_{k-1})$, or the sum of the logs $\sum_k \log \phi(\xi_k, \xi_{k-1})$.

The key advantage of the MRF representation is that it encompasses the residual uncertainty in local scan matching. This enables us to alter the shape of the map in accordance with global consistency constraints. Suppose we know that the k -th map overlaps with some map $j < k - 1$ acquired at an earlier point in time, and suppose we have a good estimate of the relative displacement between these maps. To incorporate this into the global map definition, we define a potential $\phi(\xi_k, \xi_j)$ between the coordinates of those maps ξ_k and ξ_j . This potential, or consistency constraint, is of the same form as the local constraints in (1), but usually with a tighter covariance Σ . By adding this potential $\phi(\xi_k, \xi_j)$ to the set of potentials, we softly enforce the known displacement between ξ_k and ξ_j . Thus, the language of potentials is rich enough to add additional non-local constraints that can improve the global consistency of the map.

For any fixed set of potentials $\Phi = \{\phi(\xi_k, \xi_j)\}$, which includes both the original potentials between consecutive maps and the new potentials, the resulting MRF is described through the

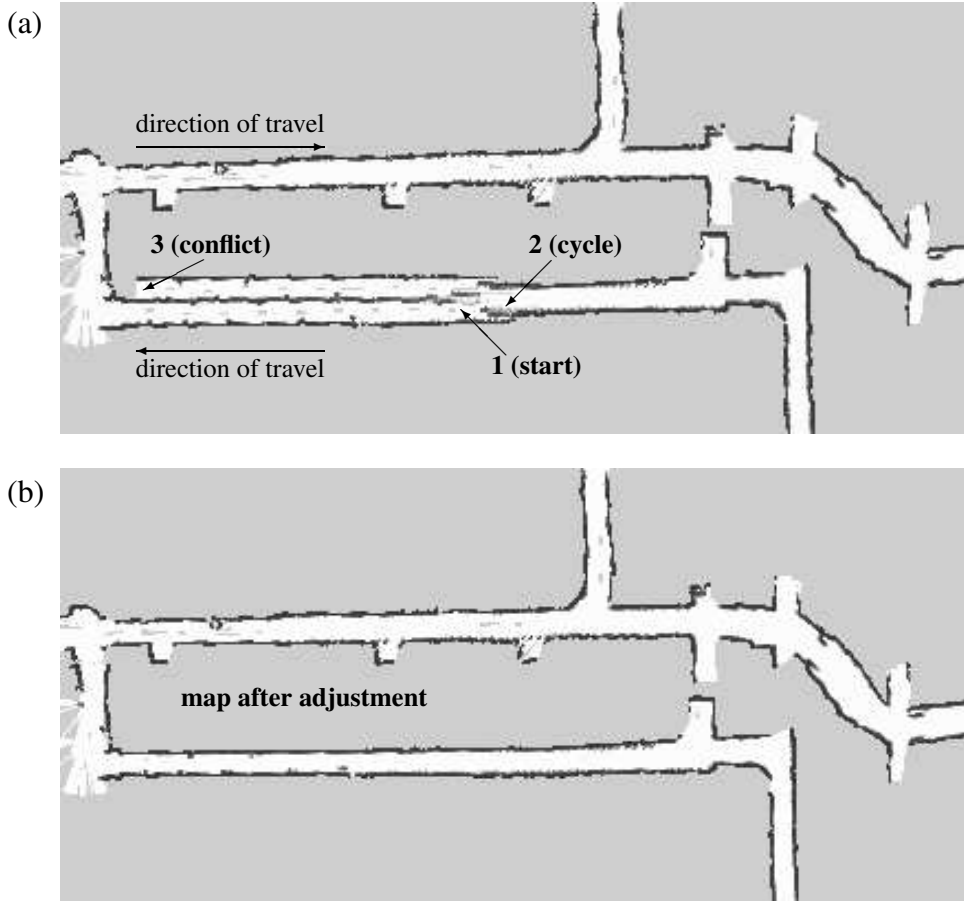


Figure 7: Example of our data association technique: When closing a large loop, the robot first erroneously assumes the existence of a second, parallel hallway. However, this model leads to a gross inconsistency as the robot encounters a corridor at a right angle. At this point, our approach recursively searches for improved data association decisions, arriving at the map shown on the bottom.

following function. This function can be thought of as a non-normalized probability over the joint global locations of all submaps:

$$p(\mathcal{X}) \propto \prod_{k,j} \exp -\frac{1}{2}(\xi_k - f(\xi_j, \delta_{k,j}))^T \Sigma^{-1} (\xi_k - f(\xi_j, \delta_{k,j})) \quad (2)$$

where $\mathcal{X} = \xi_1, \xi_2, \dots$ is the set of all map poses. The global map is now recovered by minimizing this expression over the the locations \mathcal{X} of all submaps. The negative log-likelihood $-\log p(\mathcal{X})$ is quadratic in the ξ and f -values, but the fact that f is non-linear makes it impossible to minimize this expression in closed form. The classical approach to minimizing such functions is to approximate f by its Taylor expansion. This turns $-\log p(\mathcal{X})$ into a quadratic function over the variables \mathcal{X} . Setting the first derivative to zero yields the desired minimum in closed form, as described in more detail in [17] for details. We also note that there are a number of alternative techniques for minimizing $-\log p(\mathcal{X})$, some of which exploit the sparse nature of the potentials [15, 30, 45]

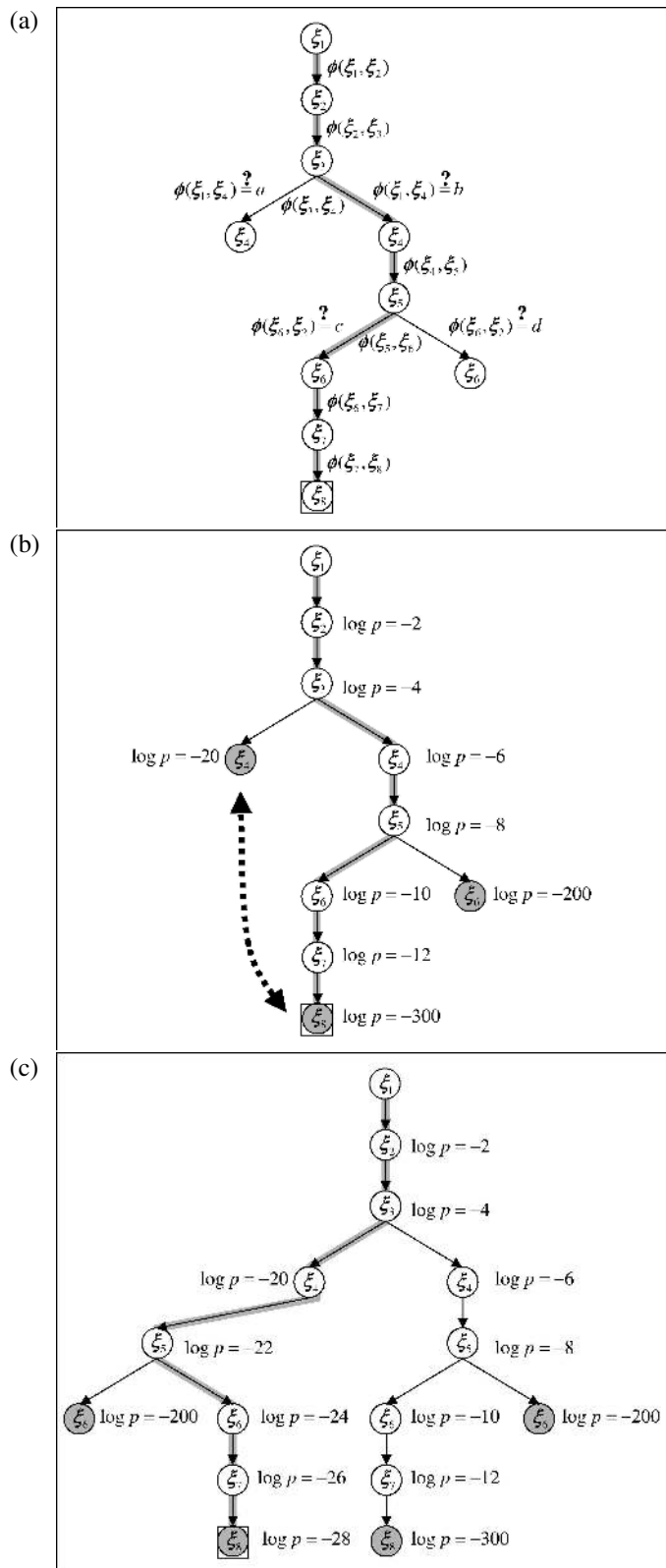


Figure 8: Searching the data association tree: (a) shows the tree and a path chosen by locally determining the most likely data association. (b) shows the associated log-likelihoods. The arrow indicates an opportunity to increase the log-likelihood by revising past data association decisions. (c) shows the result of the search for an improved data association, which provides a map of increased likelihood.

4 Data Association

The remaining question for building consistent maps is: Where do the consistency constraints come from? Clearly, the approach described thus far leads only to a consistent global map when the constraints $\phi(\xi_k, \xi_j)$, obtained after loop closure, are qualitatively correct.

Finding the “correct” consistency constraints is an instance of a more general problem known as the *data association problem* [4, 10]. The data association problem comes about when a robot has to decide whether two measurements, taken at different points in time, correspond to the same object. The scan matcher already addresses the data association problems when aligning scans. However, here the spatial error between consecutive scans is typically small, and simple heuristics such as nearest neighbor work well [7]. When closing loops, however, the error may be large, and nearest neighbor may be misleading. Such a situation is shown in Figure 7a, where a localization error induces a data association error which, in turn, leads to a broken map. The importance of this problem has been pointed out by a number of authors, who have proposed a flurry of techniques for handling them [8, 13, 18, 27, 38]. The importance of robust data association cannot be overemphasized. Mines often contain numerous loops. Mismapping even a single loop can have a devastating effect on the overall map, and as a result the vehicle may get lost in the mine and never return.

Our approach performs likelihood maximization through a lazy search of the data association tree. The data association tree is a tree of all discrete data association decisions that can be made when building a global map; its size is exponential on the length of the data sequence. An example tree is visualized in Figure 8a, which depicts the sequential data association process as new local maps are being acquired. For each new map, a decision is to be whether to introduce a consistency constraint, and what the value of this constraint may be. A consistency constraint is introduced if the map overlaps with sufficient probability with a previously acquired local map. Localizing this the new local map relative to this previous map, however, may yield more than just one possible alignment, and each such alignment may give rise to a different value for the relative displacement between the corresponding maps. In Figure 8a, such a decision is made for map ξ_4 , and then again for map ξ_6 . The map ξ_4 overlaps with map ξ_1 , and localizing ξ_4 relative to ξ_1 leads to two possible displacements, labeled a and b in this diagram. The constraint that maximizes the log-likelihood function happens to be b in this example, so the corresponding constraints $\phi(\xi_1, \xi_4) = b$ is added to the set of constraints. Similarly, ξ_6 overlaps with ξ_2 , and d appears to be the more likely value of the resulting constraint. The gray path in Figure 8a illustrates the resulting sequence of data association decisions and the affiliated potentials.

However, maximum likelihood data association is prone to errors, and sometimes such errors become only evident in retrospect. An example of this, taken from actual mine data, is depicted in Figure 7a: The misalignment happens when the cycle is first closed, at the location labeled “2” in Figure 7a. However, the inconsistency caused by this misalignment is not detected until the robot reaches the end of the corridor, labeled “3” in that figure. Figure 8b illustrates such a situation in the data association tree: For each node in the tree, it depicts the log-likelihood of the map (the sum of log-potentials and all log-probabilities obtained by matching maps). When adding the map ξ_8 , the log-likelihood takes a dip, indicating that the map is perceptually highly inconsistent. The key idea for recovering from our situation is to memorize

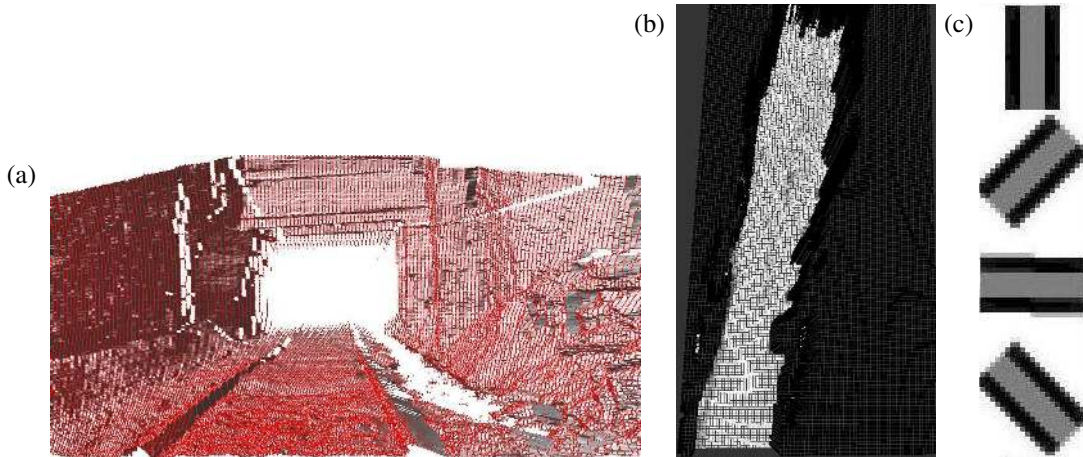


Figure 9: (a) A local 3-D model of the mine corridor, obtained by a scanning laser range finder. (b) The corresponding $2\frac{1}{2}$ -D terrain map extracted from this 3-D snapshot: the brighter a location, the easier it is to navigate. (c) Kernels for generating directional C-space maps from the $2\frac{1}{2}$ -D terrain map. The two black bars in each kernel correspond to the vehicle’s tires. Planning in these C-space maps ensures that the terrain under the tires is maximally navigable.

not only the log-likelihood along the chosen path, but also for the entire frontier of the tree. The frontier is the set of all leaf nodes of the tree at the present state of expansion: Frontier nodes are shaded gray in Figure 8b. If the log-likelihood of a node on the frontier is larger than the log-likelihood of presently chosen leaf in the tree—which happens to be the case for the left branch in Figure 8b—a revision of past data association decisions may potentially increase the overall log-likelihood, thereby improving the map. Our approach then simply starts expanding all nodes on the frontiers whose log-likelihood exceeds the log-likelihood of the chosen leaf. If a new leaf yields a higher likelihood, this leaf is chosen and the consistency constraints are modified accordingly. Figure 8c illustrates the potential outcome of this approach: in this example, a different sequence of data association decisions yields a better map. As described in detail in [17], adding and removing consistency constraints can be done efficiently, and calculating the resulting configuration \mathcal{X} does not require a full solution of the optimization problem.

Our approach is guaranteed to find the best data association sequence. Most of the time, it simply follows the locally best strategy; however, once in a while it is forced to backtrack. Figure 7 illustrates such a situation: Here the initial mine map is false, in that the robot erroneously assumes that the bottom area of the map consists of two parallel hallways. This decision, whose fallacy is not obvious at the time of loop closure, leads to a gross inconsistency later on, as indicated in Figure 7. Our approach then revises its data associations and yields the map shown in Figure 7b. This map is part of the larger map shown in Figure 5b.

5 Configuration Space Models

To make navigation decisions, the robot maps its sensor data into a configuration space representation [23], in which planning amounts to finding a trajectory for a point object. In indoor mobile robotics, it is common to navigate using 2-D maps [21], a strategy that has been reported to work even for active underground mines [25]. In abandoned mines, however, the robot needs richer information than the 2-D information used for acquiring large-scale maps.

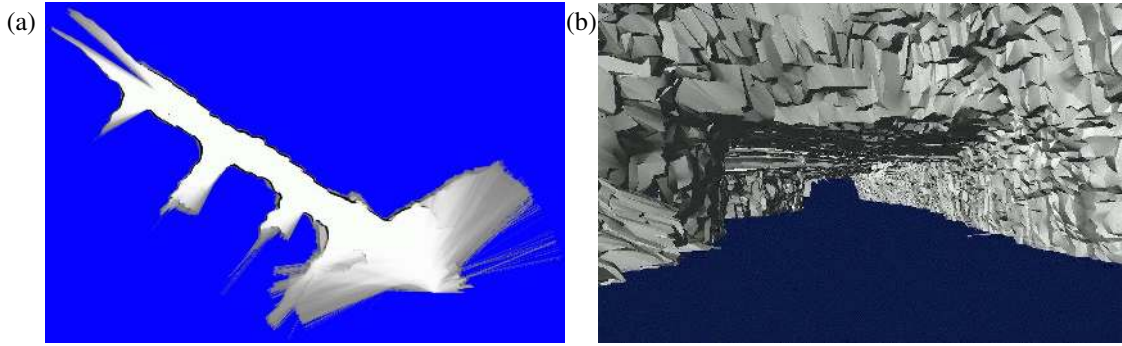


Figure 10: (a) A small 2-D map acquired by Groundhog in the Florence Mine near Burgettstown, PA. This remotely-controlled mission was aborted when the robot’s computer was flooded by water and mud in the mine. (b) View of a local 3-D map of the ceiling.

This is because holes and debris on the ground may easily create insurmountable obstacles. Other obstacles may reduce the freespace above the ground, such as low-hanging wires and partially collapsed roof structures. These challenges tend not to pose problems in active mines, which are typically kept free of debris. They are, however, paramount in abandoned mines. For exploring abandoned mines it is therefore imperative that the vehicle analyzes the full 3-D structure of what lies ahead.

Our solution to this problem is based on the growing literature on rough terrain navigation [20]. In periodic intervals, Groundhog employs its tilting mechanism to acquire 3-D range scans of the area ahead of the robot. The resulting 3-D scans are transformed into a 3-D point cloud, of the type shown in Figure 9a. The point cloud captures the ground surface, the ceiling and—most importantly—the free-space in between. Groundhog then transforms these point clouds into $2\frac{1}{2}$ -D terrain maps. The $2\frac{1}{2}$ -D map captures the traversability of the local area: the lower the value (cost) at an (x, y) position, the easier it is to navigate. Figure 9b shows an example terrain map. The gray-level in this map illustrates the degree at which the map is traversable: the brighter a 2-D location, the lower its terrain cost, and the better suited it is for navigation.

The terrain map is obtained by analyzing all measurements $\langle x, y, z \rangle$ in the 3-D scan (where z is the vertical dimension). For each rectangular surface region $\{x_{\min}; x_{\max}\} \times \{y_{\min}; y_{\max}\}$, it identifies the minimum z -value, denoted \underline{z} . It then searches for the largest z value in this region whose distance to \underline{z} does not exceed the vehicle’s height (plus a safety margin); this value will be called \bar{z} . The difference $\bar{z} - \underline{z}$ is the navigational coefficient: it loosely corresponds to the ruggedness of the terrain under the height of the robot. If no measurement is available for the target region $\{x_{\min}; x_{\max}\} \times \{y_{\min}; y_{\max}\}$, the region is marked as unknown. For safety reasons, multiple regions $\{x_{\min}; x_{\max}\} \times \{y_{\min}; y_{\max}\}$ overlap when building the terrain map. Features like railway lines represent sharp changes in height between two potentially flat surfaces (the rail and the adjacent floor) and without incorporating overlap between regions it is possible to produce terrain maps oblivious to these artifacts.

Finally, the $2\frac{1}{2}$ -D map is mapped into a configuration space representation that permits for efficient path planning and robot control. The configuration space, or C-space, is the three-dimensional space of poses that the vehicle can assume; it comprises the vehicle’s x - y location along with the vehicle’s orientation θ . Groundhog obtains its C-space maps by convolving the terrain map with oriented kernels that describe the robot’s footprint. Figure 9c shows some of



Figure 11: Camera images recorded while autonomously exploring the Mathies mine. Both image shows signs of degradation characteristic of abandoned mines, enabling mine safety personnel to assess the degree of deterioration.

these kernels: they consist of two rectangular region of high cost, which enclose a rectangular region with lower costs. The intuition behind this approach is quite straightforward: the robot is composed of two pairs of wheels on each side. Clearly, the ruggedness of the terrain matters the most under the wheels, since this is the place where the robot establishes ground contact. However, in between the wheels there is a good chance the robot may touch tall obstacles; hence the convolution kernel also incorporates the ruggedness of the terrain in between. This kernel has the nice property that it makes the robot avoid small obstacles, such as railroad tracks. Abandoned mines often possess an abundance of railroad tracks. While it is perfectly acceptable to navigate with a track between the wheels, traversing or riding these tracks causes unnecessary damage to the tires and increase the overall energy consumption.

The result of the convolution is a 3-D representation of the C-space, where two coordinates correspond to the robot's x - y location relative to its environment, the third to its orientation. Each point in the space measures the "costs" of assuming the corresponding coordinates with the robot. The C-space representation enables us to solve all planning and control problems by treating the robot as a point object.

6 Navigation

The remaining major software component pertains to the problem of navigation. The task here is to make control decisions so as to best explore and map an abandoned mine. Most of our expeditions involved a remotely controlled robot; hence all navigation decisions were made by the human operator. In our final experiment on May 30, 2003, in which Groundhog explored the Mathies Mine near Pittsburgh, PA, the robot made all navigation decisions by itself, and navigated truly autonomously. In this case, however, the exploration involved following essentially a straight corridor with a slight right bent, which is significantly simpler than the general exploration problem of exploring many different hallways. Literature on the latter is manifold [9, 22, 37, 39, 47].

Our first processing step pertained to finding a path through C-space. For that, the robot devises a goal function whose center is a location in the desired travel direction (e.g., 5 meters straight ahead in the mine). Because even roughly straight passages in mines can have several cross-cuts, it is important for the robot to be able to distinguish the current corridor from a side-hall. This can be difficult if only the current sensor information is taken into account, since the

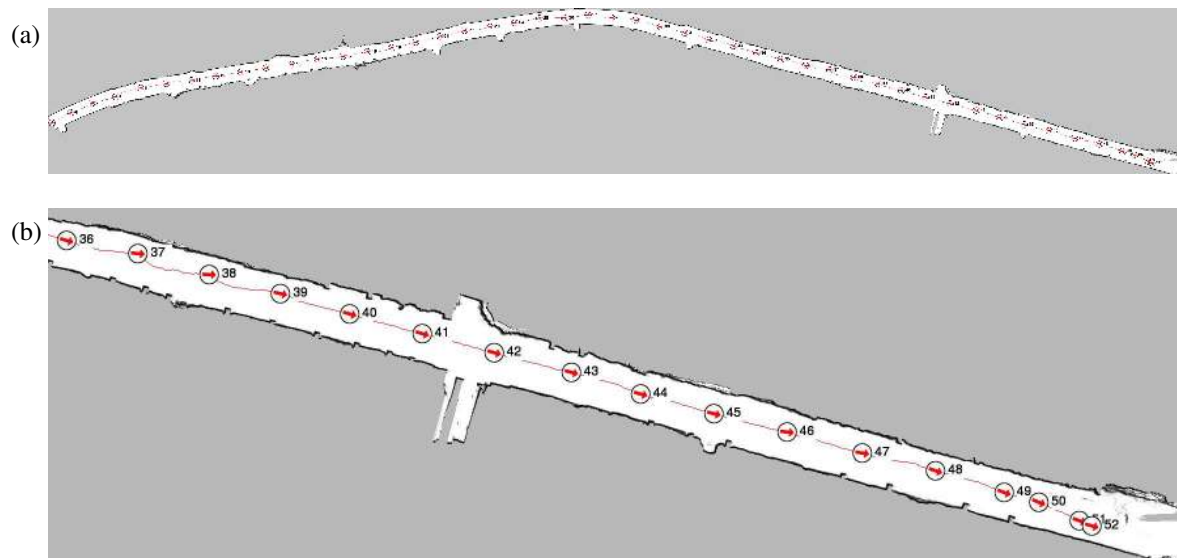


Figure 12: (a) 2-D map of the Mathies mine. This 308-meters long map has been acquired autonomously: A world’s first in successful autonomous mine exploration. (b) The final few meters of this map, with the vantage points at which the robot chose to acquire a 3-D scan. The protruding obstacle, which ultimately led the robot to back up, shows up as a small dot-like obstacle in the 2-D map.

robot may be angled relative to the current corridor (as a result of avoiding an obstacle, etc). So a number of previous robot positions are combined in a linear estimator to produce an estimate of the general direction of the mine corridor.

Once the goal region has been determined, control is generated by applying the A* algorithm [34] in C-space. Initially, the goal region is kept small; however, if planning fails to find a path below a certain cost threshold, the goal region is gradually increased. In this way, Groundhog favors trajectories that go through the center of the mine corridor; however if the center is not navigable, the robot is able to take local detours around possible obstacles. If no navigable path can be found to any of the goal points. The robot concludes that the corridor is unnavigable and initiates a high-level decision to reverse. On the reverse journey, the robot uses the exact same navigation routines, exploiting the fact that from a navigational standpoint of view, the robot is symmetric. However, to avoid getting stuck on its journey back, the safety diameter is reduced. The path found by A* is executed using a PD controller.

7 Results and Lessons Learned

Groundhog was tested in a number of experiments, some taking place in laboratory settings, others in actual mines. As described in the introduction to this article, Groundhog navigated and mapped three different mines, all with vastly different characteristics. The Bruceton research mine enabled us to perform large-scale experiments, testing the vehicle’s endurance and ability to acquire large mine maps with many cycles. However, our tests in this mine focused on the ability to acquire large-scale maps, not on autonomous navigation. The Florence mine enabled us, for the first time, to acquire a 3-D map of an environment truly inaccessible to people. The fact that it was partially flooded limited the operational range of the robot. So far, our only autonomous run was performed in the Mathies mine, where the robot operated partially outside

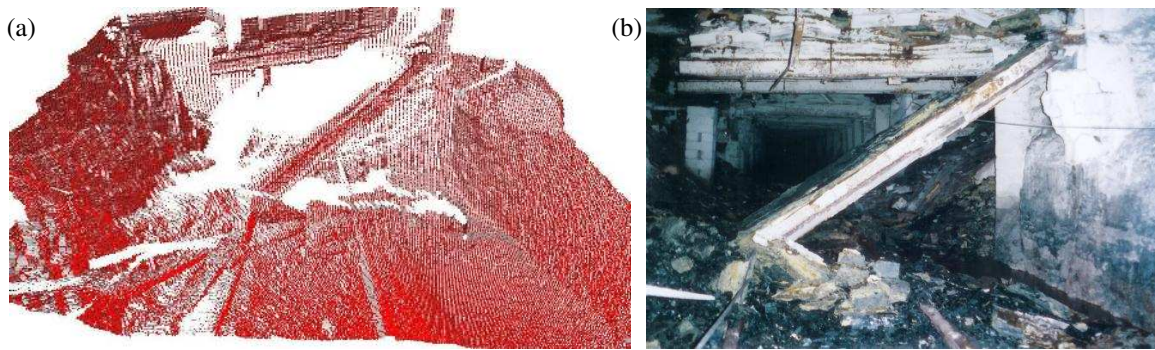


Figure 13: (a) Local 3-D map and (b) image of a broken ceiling bar that renders the corridor segment unnavigable. This obstacle was encountered 308 meters into the abandoned Mathies mine. It led Groundhog to the correct decision to retract.

the range of our wireless communication link.

Figure 10 depicts the map of the Florence mine (see Figure 2). Here the robot's configuration involved a forward-pointed laser for 2-D mapping, and an upwards pointed laser to map the ceiling structure; no sensor was available to map the ground. While this configuration is insufficient for autonomous navigation, it has the nice advantage that the 3-D map can be constructed easily from the 2-D map as the robot moves [41].

Groundhog entered the Mathies mine autonomously on May 30, 2003 at 10:55 AM, EST. Shortly thereafter, it lost radio contact with the ground station. One hour and 308m into the mine, the robot encountered a roof-fall, including a steel support beam that draped diagonally across the corridor and blocked further progress. The machine made the appropriate decision to retract and began to back out of the mine at approximately 12:00 PM, but encountered software difficulties starting at approximately 12:20 PM. After another 30 minutes, the system had not resolved its problems, and it was decided to try to intervene over the weak wireless link at 12:56 PM. Under the strain of teleoperation, the wireless link locked up shortly thereafter, stranding the robot an estimated 200 meters inside the mine at 1:04 PM. Subsequent efforts to re-establish the link failed, and at 3:30 PM, two mine safety inspectors received permission to suit up and proceed into the mine to try to manually reset Groundhog's wireless link. The link was successfully reestablished at 3:50 PM and the robot exited the mine under manual control at 4:02 PM.

Figure 11 depicts imagery acquired inside the Mathies mine. These images were recorded with a low-light camera, using the robot's active IR light source for illumination. The 2-D map of the Mathies mine is shown in Figure 12. This 308 meter-long map shows the obstruction on its right end. In 2-D, the obstruction appears to be small and navigable. In 3-D, however, it becomes apparent that the obstacle is not navigable. The robot's 3-D map of the situation is shown in Figure 13, along with the image.

The resulting 2-D map and the corresponding 3-D maps were found to provide an unprecedented glimpse into the interior of this quickly deteriorating environment. A subsequent debrief with members of mine safety and environmental protection agencies confirmed that the level of detail provided by these models opens up unprecedented opportunities to understand the situation inside an abandoned mine, and to target corrective actions.

8 Shortcomings and Future Challenges

We have described the software architecture of a deployed system for robotic mine mapping. The most important algorithmic innovations of our approach are new, lazy techniques for data association, and a fast technique for navigating rugged terrain. The system has been tested under extreme conditions, and generated accurate maps of abandoned mines that are inaccessible to people. Our research demonstrates that the autonomous acquisition of maps of abandoned mines is indeed feasible with autonomous robotic systems.

Our extensive experimentation with the Groundhog system suggests a number of opportunities for further research. Chief among them is to develop systems that can autonomously map entire mines, not just fractions thereof. Difficulties in this task arise from the fact that side-corridors were frequently closed before miners abandoned them, to stop the flow of gases from inactive into the active parts of a mine. Such closures pose insurmountable obstacles to our present system, but might be surmountable given appropriate means of environment modification. In a parallel effort, we have investigated the feasibility of building borehole-deployable robotic systems [29], which can be placed in deep mines from the surface. However, the small radii of conventional boreholes makes it difficult to lower a vehicle large enough to negotiate the rough ground terrain.

A second limitation of the present system is its inability to negotiate water and heavy mud. A good fraction of mines in the U.S. is flooded. This creates an opportunity to build submersible mine mapping robots, which would have the advantage of not being forced to the ground of a mine. Another possibility would be an amphibious vehicle for exploring partially flooded mines.

Finally, being able to communicate with a robot while inside a mine would have great operational benefits, both with regards to trouble shooting and for assisting the robot in its exploration decisions. At present, there are only low-bandwidth technologies for communicating directly through solid matter. Establishing a network of wireless repeater stations, as proposed in [32, 33], would be a viable extension to mine mapping robots, which could critically enhance the operational capabilities of future mine-exploring robots.

Despite these limitations, Groundhog's success in exploring and mapping abandoned mines opens a world of opportunities for subterranean robotic exploration. While the surface of the Planet has been mapped with great detail, most underground voids lack accurate maps, often to the detriment of the people who live or work nearby. This applies not just to man-made voids, such as mines. It equally applies to natural voids such as caves. For the first time in history, we now begin to have means to explore and map voids inaccessible to people.

Acknowledgments

We acknowledge the contributions of the students of the class 16-861 Mobile Robot Development at CMU who helped build Groundhog. We also acknowledge the assistance provided by Bruceton Research Mine (Paul Stefko), MSHA, and PA-DEP, and the various people of agencies, mining industry and Workhorse, LLC., who supported this work.

This research has been sponsored by DARPA's MARS Program (contracts N66001-01-C-6018 and NBCH1020014), which is gratefully acknowledged. We also gratefully acknowledge

financial support from the State of Pennsylvania.

References

- [1] D. Apostolopoulos, L. Pedersen, B. Shamah, K. Shillcutt, M.D. Wagner, and W.R. Whittaker. Robotic antarctic meteorite search: Outcomes. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 4174–4179, 2001.
- [2] C. Baker, Z. Omohundro, S. Thayer, W. Whittaker, M. Montemerlo, and S. Thrun. A case study in robotic mapping of abandoned mines. In *Proceedings of the International Conference on Field and Service Robotics*, Lake Yamanaka, Japan, 2003.
- [3] D. Bapna, E. Rollins, J. Murphy, M. Maimone, W.L. Whittaker, and D. Wettergreen. The Atacama desert trek: Outcomes. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, volume 1, pages 597–604, 1998.
- [4] Y. Bar-Shalom and X.-R. Li. *Estimation and Tracking: principles, Techniques, and Software*. YBS, Danvers, MA, 1998.
- [5] J. Bares and D. Wettergreen. Dante II: Technical description, results and lessons learned. *International Journal of Robotics Research*, 18(7):621–649, 1999.
- [6] J.J. Belwood and R.J. Waugh. Bats and mines: Abandoned does not always mean empty. *Bats*, 9(3), 1991.
- [7] P. Besl and N. McKay. A method for registration of 3d shapes. *Transactions on Pattern Analysis and Machine Intelligence*, 14(2):239–256, 1992.
- [8] M. Bosse, P. Newman, M. Soika, W. Feiten, J. Leonard, and S. Teller. An atlas framework for scalable mapping. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2003.
- [9] W. Burgard, D. Fox, M. Moors, R. Simmons, and S. Thrun. Collaborative multi-robot exploration. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, San Francisco, CA, 2000. IEEE.
- [10] J.C. Cox. A review of statistical data association techniques for motion correspondence. *International Journal of Computer Vision*, 10(1):53–66, 1993.
- [11] G. Dissanayake, H. Durrant-Whyte, and T. Bailey. A computationally efficient solution to the simultaneous localisation and map building (SLAM) problem. Working notes of ICRA'2000 Workshop W4: Mobile Robot Navigation and Mapping, April 2000.
- [12] H. Durrant-Whyte, S. Majumder, S. Thrun, M. de Battista, and S. Scheduling. A Bayesian algorithm for simultaneous localization and map building. In *Proceedings of the 10th International Symposium of Robotics Research (ISRR'01)*, Lorne, Australia, 2001.
- [13] A. Eliazar and R. Parr. DP-SLAM: Fast, robust simultaneous localization and mapping without predetermined landmarks. In *Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence (IJCAI)*, Acapulco, Mexico, 2003. IJCAI.
- [14] D. Ferguson, A. Morris, D. Hähnel, C. Baker, Z. Omohundro, C. Reverte, S. Thayer, W. Whittaker, W. Whittaker, W. Burgard, and S. Thrun. An autonomous robotic system for mapping abandoned mines. In S. Thrun, L. Saul, and B. Schölkopf, editors, *Proceedings of Conference on Neural Information Processing Systems (NIPS)*. MIT Press, 2003.
- [15] Anshul Gupta, George Karypis, and Vipin Kumar. Highly scalable parallel algorithms for sparse matrix factorization. *IEEE Transactions on Parallel and Distributed Systems*, 8(5):502–520, 1997.
- [16] J.-S. Gutmann and K. Konolige. Incremental mapping of large cyclic environments. In *Proceedings of the IEEE International Symposium on Computational Intelligence in Robotics and Automation (CIRA)*, 2000.

- [17] D. Hähnel, W. Burgard, B. Wegbreit, and S. Thrun. Towards lazy data association in SLAM. In *Proceedings of the 11th International Symposium of Robotics Research (ISRR'03)*, Sienna, Italy, 2003. Springer.
- [18] D. Hähnel, D. Fox, W. Burgard, and S. Thrun. A highly efficient FastSLAM algorithm for generating cyclic maps of large-scale environments from raw laser range measurements. In *Proceedings of the Conference on Intelligent Robots and Systems (IROS)*, 2003.
- [19] D. Hähnel, D. Schulz, and W. Burgard. Map building with mobile robots in populated environments. In *Proceedings of the Conference on Intelligent Robots and Systems (IROS)*, Lausanne, Switzerland, 2002.
- [20] K. Hashimoto and S. Yuta. Autonomous detection of untraversability of the path on rough terrain for the remote controlled mobile robots. In *Proceedings of the International Conference on Field and Service Robotics*, Lake Yamanaka, Japan, 2003.
- [21] D. Kortenkamp, R.P. Bonasso, and R. Murphy, editors. *AI-based Mobile Robots: Case studies of successful robot systems*, Cambridge, MA, 1998. MIT Press.
- [22] B. Kuipers and Y.-T. Byun. A robot exploration and mapping strategy based on a semantic hierarchy of spatial representations. *Journal of Robotics and Autonomous Systems*, 8:47–63, 1991.
- [23] J.-C. Latombe. *Robot Motion Planning*. Kluwer Academic Publishers, Boston, MA, 1991.
- [24] F. Lu and E. Milius. Globally consistent range scan alignment for environment mapping. *Autonomous Robots*, 4:333–349, 1997.
- [25] R. Madhavan, G. Dissanayake, H. Durrant-Whyte, J. Roberts, P. Corke, and J. Cunningham. Issues in autonomous navigation of underground vehicles. *Journal of Mineral Resources Engineering*, 8(3):313–324, 1999.
- [26] L. Matthies, E. Gat, R. Harrison, B. Wilcox, R. Volpe, and T. Litwin. Mars microover navigation: Performance evaluation and enhancement. *Autonomous Robots*, 2(4):291–311, 1995.
- [27] M. Montemerlo, S. Thrun, D. Koller, and B. Wegbreit. FastSLAM 2.0: An improved particle filtering algorithm for simultaneous localization and mapping that provably converges. In *Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence (IJCAI)*, Acapulco, Mexico, 2003. IJCAI.
- [28] H. P. Moravec. Sensor fusion in certainty grids for mobile robots. *AI Magazine*, 9(2):61–74, 1988.
- [29] A. Morris, D. Kurth, W. Whittaker, and S. Thayer. Case studies of a borehole deployable robot for limestone mine profiling and mapping. In *Proceedings of the International Conference on Field and Service Robotics*, Lake Yamanaka, Japan, 2003.
- [30] K.P. Murphy, Y. Weiss, and M.I. Jordan. Loopy belief propagation for approximate inference: An empirical study. In *Proceedings of the Conference on Uncertainty in AI (UAI)*, pages 467–475, 1999.
- [31] P. Newman. *On the Structure and Solution of the Simultaneous Localisation and Map Building Problem*. PhD thesis, Australian Centre for Field Robotics, University of Sydney, Sydney, Australia, 2000.
- [32] H.G. Nguyen, H.R. Everett, N. Manouk, and A. Verma. Autonomous mobile communication relays. In *Proc. of the SPIE Symposium on AeroSense/unmanned Ground Vehicle technology IV*, Orlando, FL, 2002.
- [33] H.G. Nguyen, N. Pezeshkian, M. Raymond, A. Gupta, and J.M. Spector. Autonomous communication relays for tactical robots. In *Proceedings of the 11th International Conference on Advanced Robotics (ICAR)*, Coimbra, Portugal, 2003.
- [34] N. J. Nilsson. *Principles of Artificial Intelligence*. Springer Publisher, Berlin, New York, 1982.
- [35] E. Pauley, T. Shumaker, and B. Cole. Preliminary report of investigation: Underground bituminous coal mine, non-injury mine inundation accident (entrapment), July 24, 2002, Quecreek, Pennsylvania, 2002. Black Wolf Coal Company, Inc. for the PA Bureau of Deep Mine Safety.
- [36] J. Pearl. *Probabilistic reasoning in intelligent systems: networks of plausible inference*. Morgan Kaufmann Publishers, San Mateo, CA, 1988.

- [37] R. Simmons, D. Apfelbaum, W. Burgard, M. Fox, D. an Moors, S. Thrun, and H. Younes. Coordination for multi-robot exploration and mapping. In *Proceedings of the AAAI National Conference on Artificial Intelligence*, Austin, TX, 2000. AAAI.
- [38] J.D. Tardós, J. Neira, P. Newman, and J. Leonard. Robust mapping and localization in indoor environments using sonar data. Technical Report TM 2001-04, MIT Marine Robotics Lab, 2001.
- [39] S. Thrun. Exploration and model building in mobile robot domains. In E. Ruspini, editor, *Proceedings of the IEEE International Conference on Neural Networks*, pages 175–180, San Francisco, CA, 1993. IEEE Neural Network Council.
- [40] S. Thrun. A probabilistic online mapping algorithm for teams of mobile robots. *International Journal of Robotics Research*, 20(5):335–363, 2001.
- [41] S. Thrun, W. Burgard, and D. Fox. A real-time algorithm for mobile robot mapping with applications to multi-robot and 3D mapping. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, San Francisco, CA, 2000. IEEE.
- [42] S. Thrun, D. Hähnel, D. Ferguson, M. Montemerlo, R. Triebel, W. Burgard, C. Baker, Z. Omohundro, S. Thayer, and W. Whittaker. A system for volumetric robotic mapping of abandoned mines. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2003.
- [43] S. Thrun, D. Koller, Z. Ghahramani, H. Durrant-Whyte, and A.Y. Ng. Simultaneous mapping and localization with sparse extended information filters. In J.-D. Boissonnat, J. Burdick, K. Goldberg, and S. Hutchinson, editors, *Proceedings of the Fifth International Workshop on Algorithmic Foundations of Robotics*, Nice, France, 2002.
- [44] C. Urmson, B. Shamah, J. Teza, M.D. Wagner, D. Apostolopoulos, and W.R. Whittaker. A sensor arm for robotic antarctic meteorite search. In *Proceedings of the 3rd International Conference on Field and Service Robotics*, Helsinki, Finland, 2001.
- [45] M. J. Wainwright. *Stochastic processes on graphs with cycles: geometric and variational approaches*. PhD thesis, Dept. of Electrical Engineering and Computer Science, MIT, Cambridge, MA, January 2002.
- [46] Y. Weiss and W.T. Freeman. Correctness of belief propagation in gaussian graphical models of arbitrary topology. *Neural Computation*, 13(10):2173–2200, 2001.
- [47] B. Yamauchi. A frontier-based approach for autonomous exploration. In *Proceedings of the IEEE International Symposium on Computational Intelligence in Robotics and Automation*, pages 146–151, Monterey, CA, 1997.