Experiments with Autonomous Mobile Radios for Wireless Tethering in Tunnels

Kevin L. Moore*, Manoja D. Weiss**, John P. Steele, Ken Anderson, Jesse Hulbert, Christer Karlsson, Eric Larson, Chris Meehan, and Alejandro Weinstein
Division of Engineering, Colorado School of Mines
Golden, CO 80403

ABSTRACT

Tunnels are a challenging environment for radio communications. In this paper we consider the use of autonomous mobile radio nodes (AMRs) to provide wireless tethering between a base station and a leader in a tunnel exploration scenario. Using a realistic, experimentally-derived underground radio signal propagation model and a tethering algorithm for AMR motion control based on a consensus variable protocol, we present experimental results involving a tele-operated leader with one or two followers. Using radio signal strength measurements, the followers autonomously space themselves so as to achieve equal radio distance between each entity in the chain from the base to the leader. Results show the feasibility of our ideas.

Keywords: Autonomous radio relays, tunnels, wireless tethering.

1. INTRODUCTION

Consider the scenario depicted in Figure 1. Shown is a two-dimensional plan of an underground mine. Notionally a cave-in has blocked access to a portion of the mine equipped with a sensor network that provides monitoring and communications for personnel who may be trapped in the back part of the mine. Because mine environments in an emergency can be very dangerous, it is preferable to attempt to clear the rubble via tele-operation and inspect the tunnels remotely before human rescue teams enter the mine. However, the nature of radio propagation in underground tunnels is such that typical commercial tele-operation systems will not be able to provide non-line-of-sight (NLOS) communication with the rubble-removal equipment from the surface. Although it is possible for the equipment to pull a communication tether or to deploy a "daisy-chain" of fixed-location radio relays, the scenario depicted in Figure 1 suggests another approach: the use of autonomous mobile radio nodes (AMRs) that act as relays for data to-and-from the remote equipment and eventually to-and-from the underground sensor network. The advantage of using autonomous relays is that the AMRs can automatically adjust to find the optimal AMR-to-AMR spacing for the specific situation at hand, without the need for *a priori* calculation of the spacing, where "optimal" means providing for the best possible radio communication between all the units.

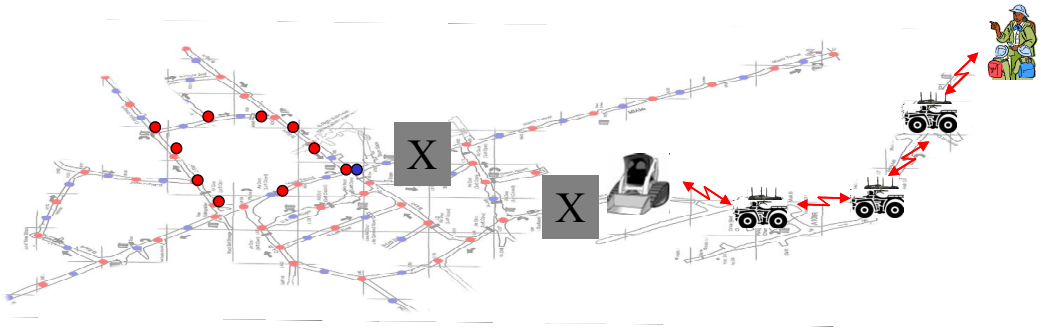


Figure 1: Autonomous mobile relay node scenario.

* Corresponding Author: kmoore@mines.edu
** Now with Ball Aerospace

Unmanned Systems Technology XII, edited by Grant R. Gerhart, Douglas W. Gage, Charles M. Shoemaker,
Proc. of SPIE Vol. 7692, 76920R · © 2010 SPIE · CCC code: 0277-786X/10/$18 · doi: 10.1117/12.849328

Downloaded From: http://proceedings.spiedigitallibrary.org/ on 11/07/2016 Terms of Use: http://spiedigitallibrary.org/ss/termsofuse.aspx

In this paper we details the results of a series of experiments aimed at the scenario depicted in Figure 1. We do not discuss the generic issues related to underground robotics, but refer the interested reader to the recent survey[1]. Likewise, we do not provide extensive details of the RF models and control algorithms used, but focus instead on our implementations. The interested reader is referred to Weiss, et al. [2] and the references therein for technical information about radio propagation models and control algorithms. We also refer the interested reader to Moore, et al. [3] for more information about the generic UGV auto pilot architecture we have developed and that was used for all the autonomous vehicles in the experiments.

The paper is organized as follows. We begin with a description of one of the AMR hardware platforms that we developed. Then we discuss the RF environments in tunnels, including an experimental characterization obtained in our school's experimental mine. Next we describe results from three different radio tethering experiments, each demonstrating one or more of the components of the scenario shown in Figure 1. These include an:

(1) Outdoor radio tethering experiment with one autonomous follower that uses wall-following-based steering control (this experiment uses the autonomous vehicle platform discussed in[3]);

(2) Underground person-in-the-loop radio tethering experiment with one and two autonomous (human) followers;

(3) Underground person-in-the-loop radio tethering experiment with one semi-autonomous follower (human steering the AMR described in this section, with autonomous velocity control based on radio signal strength).

In the experiments described "radio tethering" means the scenario depicted in Figure 2. As shown, there is a leader (possibly static or mobile) and a base station. The goal is for the AMR to autonomously move so as to achieve equal radio signal strength (RSS) between itself and the base and itself and the leader. If there is more than one follower, then the goal of each AMR is to equalize the signal strength between itself and the radios behind and in front of the AMR. The meaning of "person-in-the-loop" is explained in the appropriate sections below.

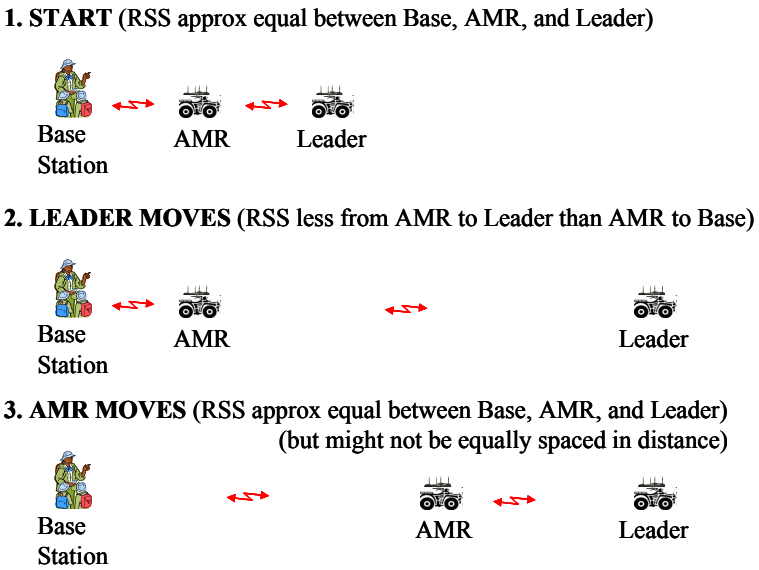


Figure 2: Radio tethering experimental scenario.

2. SYSTEM COMPONENTS

The system components developed in this project included two different AMRs (one based on a "mini-Baja" vehicle and the other on an "EZ-Go" golf cart), a tele-operated Bobcat vehicle, and a mesh radio infrastructure. Additionally various operator interfaces were used. In this section we detail these system components, with the exception of the mini-Baja vehicle, which was described in[3].

Downloaded From: http://proceedings.spiedigitallibrary.org/ on 11/07/2016 Terms of Use: http://spiedigitallibrary.org/ss/termsofuse.aspx

2.1 Golf Cart-Based AMR

Figure 3 shows the EZ-Go golf cart that was modified to enable drive-by-wire and drive-by-computer operation, thus enabling its use as an AMR. Modifications included actuation, sensor additions, on-board computing, and operator interfaces, each described below.

Figure 3: EZ-Go AMR in the Colorado School of Mines' Edgar Mine.

2.1.1 Vehicle Actuation

The vehicle actuation requires special consideration due to the requirement that manual operation needs to be preserved. The golf cart's steering and brake are entirely mechanical, whereas the throttle is an electromechanical system which uses an inductive sensor to sense the position of the throttle pedal. A Roboteq motor control board drives the steering and brake actuators to a position set point using a PID control algorithm.

The steering is actuated using a geared-down servomotor mounted near the steering wheel. The servomotor's rotation is mechanically linked to the steering shaft via a timing belt and two timing pulleys. To allow for manual operation without back-driving the servomotor, the timing pulley on the steering shaft is allowed to freely spin on the shaft. A separate, splined assembly fixed to the steering shaft is pinned to the free-spinning timing pulley, mechanically linking the pulley rotation to the steering shaft. By sliding the splined assembly axially, the mechanical connection between the steering shaft and timing pulley can be engaged or disengaged allowing the operator to change between manual and electronic steering control.

The brake is more seamlessly integrated, using a guided rod and linear actuator to push directly on the brake arm. At any time, the operator can manually push on the brake, and the rod will simply move in the guide. Conversely when the linear actuator extends, the rod moves until it makes mechanical contact with the brake arm and thus push in the brake.

The throttle is actuated by simply placing a circuit in parallel with the throttle sensor which mimics the same functionality. Applying a pulse width modulated signal to the circuit changes the voltage at the sensor line used by the golf cart speed controller. Because the circuit is in parallel with the sensor, manual and electronic throttle operation are available at all times.

2.1.2 Sensors

There are a total of 6 propreoceptive sensors on the AMR: a Hall Effect sensor on each wheel to measure odometry and calculate vehicle speed, a string potentiometer on the steering arm to measure the Ackerman steering angle, and a 5 degree of freedom Inertial Measuring Unit (IMU) to measure vehicle orientation. Combined, these sensors provide steering angle, velocity, and yaw, all necessary to compute Ackerman vehicle dynamics.

For environmental sensing there are currently14 sonar ranging sensors that symmetrically surround the AMR and measure the distance to the wall or obstacle, as shown in Figure 4. These sensors have a range between 15 cm and 6 meters.

Downloaded From: http://proceedings.spiedigitallibrary.org/ on 11/07/2016 Terms of Use: http://spiedigitallibrary.org/ss/termsofuse.aspx

Sonar sensors are ideal in mining environments because the diffuse mine walls scatter sound waves, allowing the sensor to easily detect the wall. However, these sensors have some drawbacks because they have a slow update rate and are prone to interference from neighboring sensors. To help alleviate the slow update rate, the 14 sensors are broken up into 4 triggering groups; each triggering group has between 2 and 4 sonars, all triggered simultaneously. These triggering groups cycle through every 55 milliseconds, making the total update rate 5 Hz. To avoid interference, each sensor is oriented 90 degrees apart from other sensors in the group. Measurements from these sensors are used to build an environment model of mine walls.

One problem with the sonar sensors is they are only accurate to +/- 2 inches, which causes problems when trying to build a good environment model. In the future, we may add a scanning laser sensor, which should dramatically improve accuracy. In addition, we may also add a stereovision system, which would allow us to build a three-dimensional environmental model, further improving accuracy, and therefore, navigation.

Another problem with the sonar sensors is they have trouble getting range measurements at steep angles (from 45deg to parallel with the wall). Thus, we added 4 Sharp Infrared long-range sensors are oriented towards the 4 corners of the vehicle at 45 degree angles. These sensors serve as 'backups' to the 4 sonars pointed in the same direction, in the event that the sonars have difficulty reading the mine walls at steep angles.

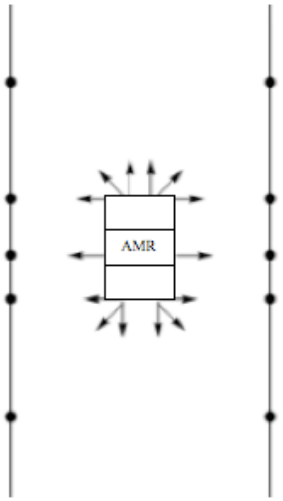


Figure 4: AMR sonar positions.

2.1.3 On-Board Computing and Software Infrastructure

Low-level controller hardware and algorithms: The computing structure consists of three hierarchical computer layers as shown in Figure 5. The lowest level (far right) is the servo controller. This controller is a RoboteQ AX3500, and sets the steering and brake actuators to commanded values using a PID loop. The middle computer layer is a digital signal processing Microchip PIC processor and handles all robotic functions such as analog to digital conversion, servo set point calculation, command processing, and speed control. This 'robotic' controller sends commands down to the servo controller and receives commands from the 'autonomous' controller via RS232 communication. The autonomous controller (far left) runs higher-level navigation routines.

Autonomy controller and algorithms**:** Currently, the autonomous controller is implemented on a laptop (or netbook) and coded in the Python programming language. The autonomous controller communicates with the robotic controller via a USB to serial adapter. A communication packet structure was created to standardize communication and detect communication errors. The autonomous controller sends requests for sensor data, system status, or motion actions to which the robotic controller responds appropriately with data or vehicle motion. When enabled, the autonomous controller requests sensor data, processes it, and determines the appropriate motion commands to send to the robotic controller.

Steering is controlled by a fuzzy controller, which compares current, sonar sensor readings with simulated sonar sensor responses for different ideal situations. Each situation has an associated steering angle with the intent of keeping the AMR in the center of the mine drift. Multiple situations were simulated on the computer, each generating a vector of sonar responses, V_{SIM}. These simulated sensor responses are normalized and organized into a matrix as row vectors, M_{SIM}. During a sensor update, sensor data is stored in a column vector, V_{SENS}, and normalized. The sensor responses are then compared to the simulated situations by multiplying M_{SIM} and V_{SENS}, resulting in a column vector, V_C, where each element is the dot product of simulated sensor responses and the actual response. The higher the value of each element, the more strongly the current sensor response relates to the corresponding situation. These values are used to get a weighted average of all steering angles for each situation, which is performed by taking the dot product of V_C and the vector of steering angles, V_{SA}. In short, at each control update (~16Hz) the steering angle is obtained using:

$$\text{Steering Angle} = \left\langle M_{SIM} \cdot V_{SENS}, V_{SA} \right\rangle$$

The resulting steering angle is then limited to ± 40˚, which is slightly less then the mechanical limits of the golf cart.

Downloaded From: http://proceedings.spiedigitallibrary.org/ on 11/07/2016 Terms of Use: http://spiedigitallibrary.org/ss/termsofuse.aspx

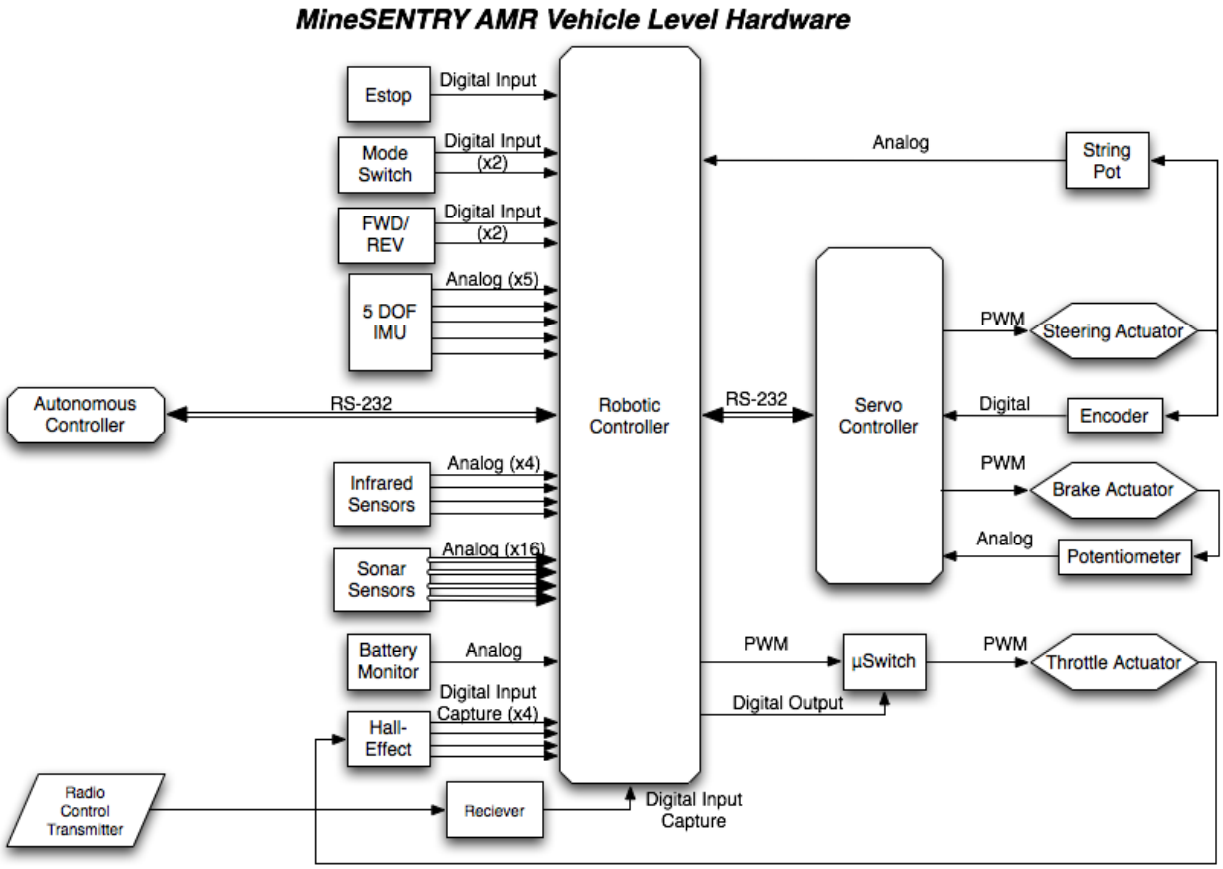


Figure 5: Computer architecture.

Due to scheduling constraints for the first experimental test at Edgar Mine, the fuzzy controller was not designed to navigate past side passages which intersected the main passage. However, detection of side passages was integrated by adding simulated situation vectors which included side passages. The results for the situation comparison in V_C allowed logic to be added that switched the steering control to a proportional controller that followed the wall opposing the passage. When the element in V_C corresponding to a side-passage situation passes a certain threshold, logic would switch the steering control over to the proportional controller. After the AMR passes the side passage, logic gives steering control back to the fuzzy controller.

During radio tethering experiments, a separate program monitors RSS signal strength and intermittently sends distance values to the autonomous controller program. The autonomous controller uses a set speed and the fuzzy controller to drive down the mine drift the specified distance. Once the AMR reaches the target distance it stops and notifies the other program. The AMR maintains its position for approximately 30 seconds and collects data before the next target distances is sent.

2.1.4 Operator Interfaces

The AMR has three user-selectable modes: manual, remote, and autonomous. In manual mode, the brake and steering actuators are disengaged, which allows a human operator to drive the golf cart without computer interference. In remote mode, the robotic controller takes over actuation, allowing a human operator to control the golf cart wirelessly using a standard hobby-grade transmitter and receiver pair. Finally, in autonomous mode, the autonomous controller takes over all actuation and automatically guides the AMR through the mine.

Downloaded From: http://proceedings.spiedigitallibrary.org/ on 11/07/2016 Terms of Use: http://spiedigitallibrary.org/ss/termsofuse.aspx

In autonomous modes, the operator operates the AMR through a small graphical interface written in Python (shown in Figure 6). Currently the interface is simple, utilizing buttons or keyboard commands to allow the operator to turn the fuzzy controller on or off, turn on data logging, or add comments to the log files. A set of slider bars is also available which can be configured for PID tuning, speed, steering, or brake control. Future development will include indicators for system status and greater control of logging or system setup. The program may be started via command line or by double clicking on the Python file, and debugging information is displayed either in the command line window or a secondary window if opened by double clicking on the Python file.

Figure 6: AMR interface.

2.2 Tele-operated Bobcat

Though we did not use the Bobcat in the experiments describe model Bobcat front end loaders have the added option of a remote ol system uses a proprietary frequency hopping spread spectrum wireless transmission protocol. This communication protocol is incompatible with our Ethernet/Wi-Fi network, and could not be used for remote control within an underground mine environment. In order to make the Bobcat remote control system compatible with an Ethernet/Wi-Fi network two Ethernet enabled microprocessors were used to trick the remote control system into thinking it is receiving signals from its own electronics. One microprocessor reads information from the user, in the form of joysticks and switches. This microprocessor then communicates with the other microprocessor via an Ethernet/ Wi-Fi channel. The second microprocessor then interprets the data and relays information in the form of simulated signals to the original Bobcat remote control system. With an Ethernet/Wi-Fi network, adding cameras and other Ethernet enabled devices is very easy since off the shelf components exist for most applications. Within our system we have cameras on every vehicle and will have multiple cameras on the Bobcat because it is tele-operated. Figure 7 shows the resulting architecture for the tele-operated Bobcat. Figure 8 shows the Bobcat tele-operation user interface hardware and an example of the video feedback available to the operator from the Bobcat.

2.3 Communication Infrastructure

To wirelessly transmit high bandwidth data, like video, a high frequency signal is required. Since real time video is necessary for tele-operation of the Bobcat front end loader, we need a high frequency signal to carry the information. Because of this high frequency signal requirement, and because of the many off the shelf components already available,

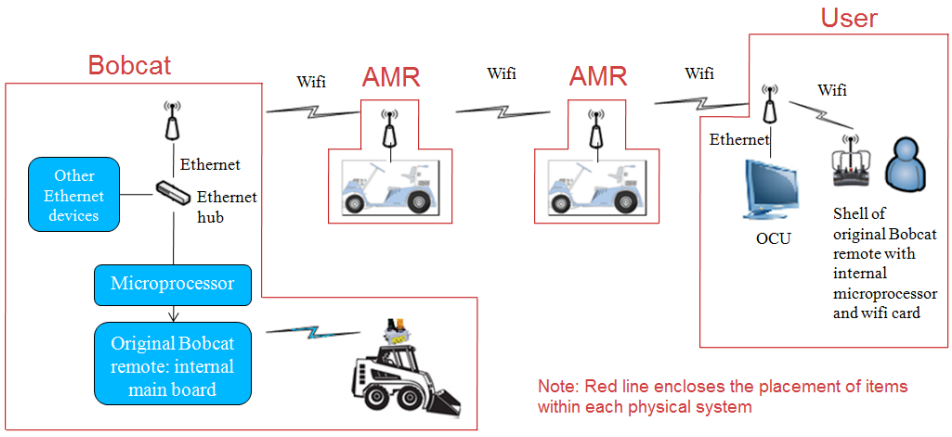


Figure 7: Bobcat tele-operation layout.

Downloaded From: http://proceedings.spiedigitallibrary.org/ on 11/07/2016 Terms of Use: http://spiedigitallibrary.org/ss/termsofuse.aspx

Figure 8: Bobcat tele-operation interface and video feedback.

we chose to use a standard 802.11 b/g Wi-Fi network. The company Rajant provides Wi-Fi equipment with the added ability to interconnect in an ad-hoc fashion and adapt to changing environments. Rajant Wi-Fi nodes connect to as many other Rajant Wi-Fi nodes that are within range, providing a highly redundant and reliable Wi-Fi network. The more nodes, the more resilient the meshed network becomes: if one node loses connection the data can be relayed through the other nodes to get the information across. Additionally, software provided by the vendor allows the ability to obtain and use the radio signal strength between connected units. We used a Rajant ME2 Breadcrumb radios on each vehicle in our system as well as at the base station. Because these are commercial units, we do not discuss this part of the system in any more detail.

3. THE RADIO ENVIRONMENT IN TUNNELS

3.1 Overview of Experiments

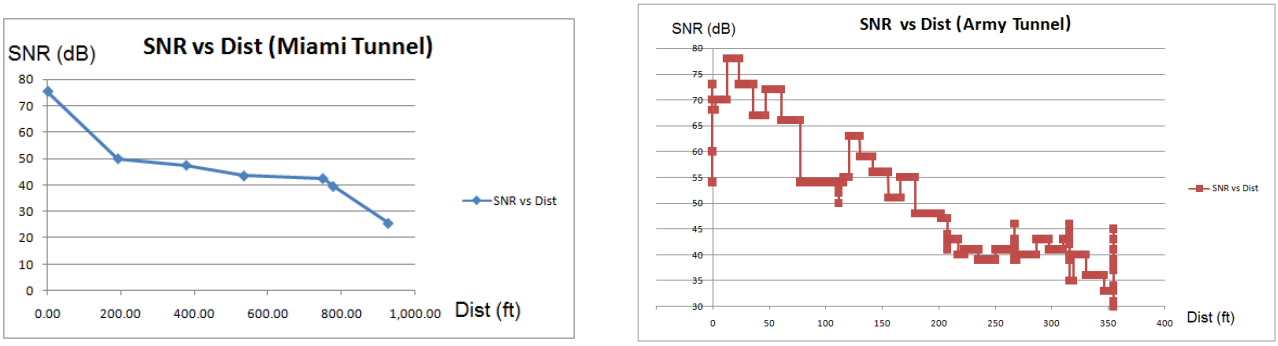
The propagation of wireless signals within an underground mine or tunnel environment is dependent upon many parameters. The cross sectional shape of the mine, the composition of the surrounding rock, the curvature of tunnels, the position of antennas, and the frequency, among many others. Within our project we use 802.11b/g Wi-Fi for the necessary high bandwidth communication. This standard Wi-Fi protocol uses a frequency of 2.4 GHz, which has a small wavelength in comparison to the dimension of the mine. Because the wavelength is small, there is minimal diffraction of waves around corners. Though there is a strong waveguide effect in many mine tunnels, there is also large bending loss around corners. What this means is, for straight tunnels, the loss is less than in free space because of waveguiding (reflections from the walls, if you will for these frequencies of interest). However, at a bend, non-specular scattering due to the surface roughness of the walls and the low reflection coefficient of the walls result in high losses. Thus, getting wireless signals around corners is dependent upon reflected waves. However, the reflection of waves is a very inefficient process because most of the wave energy is absorbed into the walls of the mine. What this means for transmitting wireless information in an underground mine, is that LOS placement of equipment is the best means of transmitting and relaying high frequency signals.

We performed several experiments within the Edgar experimental mine to quantify the distance verses signal strength for a few tunnels within the Edgar mine (note: these are in addition to previous work conducted by some of us on this problem; see [2] and the references therein for additional information on this topic). We took tests of line-of-sight (LOS) and non-LOS environments within in the Miami and Army tunnel sections of the mine (we refer to these tunnel sections by their name for delineation, but do not show them on a map due to space limitation; please contact the authors for more information). The experiments were performed using two Rajant ME2 BreadCrumb Wi-Fi nodes, at transmitter power of 34 dBm, and omni-directional TX and RX antennas. In the experiments the first node was placed at 0ft and the second node was placed at several locations at varying distances from the first node. SNR, which has a direct correlation to the signal strength in a noiseless environment of the mine, was recorded at several distance points along the tunnels.

Downloaded From: http://proceedings.spiedigitallibrary.org/ on 11/07/2016 Terms of Use: http://spiedigitallibrary.org/ss/termsofuse.aspx

3.2 Experimental Results

The first tests we performed were using the straight sections of the Miami tunnel and Army tunnel. The cross sectional shape of the Miami tunnel has a rough diameter of about 8 ft, while the Army tunnel has a diameter of about 15ft. The SNR versus line-of-sight (LOS) distance results are shown in Figure 9. From the results, it is obvious that signals are not attenuated as quickly within the Miami tunnel, this could be for many reasons, but most likely is due to a waveguide effect.



(a) **(b)**

Figure 9: SNR versus LOS distance (a) Miami tunnel; (b) Army tunnel.

The next tests considered a 90° turn at 750 ft within the Miami tunnel. Figure 10 shows the SNR versus non-line-of-sight (NLOS) distance as we moved into the turn at varying distances. From the results, it is easy to see that signal strength is attenuated very quickly when a turn around a corner is taken within a mine. From both the LOS test and the non-LOS, the non-LOS signal drops at a rate of about twenty times quicker than the LOS signal. Thus the experiments confirm our assumptions about transmitting signals in tunnel environments, that LOS placement of equipment is the optimum solution.

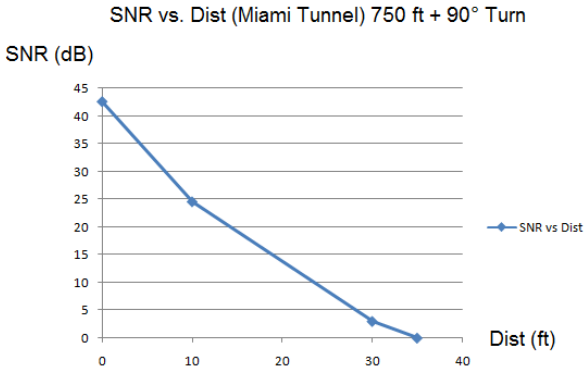


Figure 10: SNR versus NLOS distance (Miami tunnel).

4. RADIO TETHERING EXPERIMENTS

As noted in the introduction, we have completed three sets of radio tethering experiments. We discuss each of these separately below:

4.1 Outdoor Radio Tethering with One Autonomous Follower

As a first step, and as a way to validate our algorithms, we prepared an experiment with one manned leader and one autonomous follower. The objective of this experiment is to equalize the RSSs between the follower and the leader, and between the follower and the base station, while following a wall at the left side of the follower. The experiment was located outside the Brown Hall, on the Colorado School of Mines campus. Figure 11 shows an aerial view of the location. As can be seen, the first half of the path is a semicircle of an approximate radius of 9 meters, follow by a straight ramp about 20 meters long. The base station was placed in the interior of the building. As a follower we used an electric Minibaja vehicle with drive-by-wire capabilities shown in Figure 12 and described in [3]. The algorithms described in [2] were implemented in a single board computer based on a Pentium-M processor, and were written in Python, using a multithreaded architecture. Two SONARs (one in the left side and one in the front) were added to the vehicle to measure the distance to the wall. As a leader we used a manned golf cart. The leader and the follower, as well as the base station, were equipped with a Rajant Radio. The experiment starts with the follower at the start of the path,

Downloaded From: http://proceedings.spiedigitallibrary.org/ on 11/07/2016 Terms of Use: http://spiedigitallibrary.org/ss/termsofuse.aspx

and the leader somewhere in front of the follower. When the autonomous mode in the follower is enabled, it moves forward until the RSSs are equal, while following the wall. Then the leader move forward and stops, and the follower follows it, keeping the RSSs equal. This sequence is repeated until the follower finishes the path. The problem was simplified by assuming that the Minibaja only moves forward. There are two variables we want to control: the distance between the Minibaja and the wall, and the difference between the RSSs. Since the Minibaja is moving at fairly low speeds (smaller than 1 meter per second), we can decouple the two variables, and control the distance to the wall through the steering, and the RSSs differences with the vehicle speed. The algorithmic basis for this control was given in [2]. Here we detail the implementation of the algorithm.

Figure 11: Aerial view of experiment.

Figure 12: Electric Minibaja.

4.1.1 Wall-Following Algorithm

The distance to the left wall is measured with a SONAR mounted in the left side of the Minibaja. In order to be able to negotiate the corner at the end of the circular arc of the path, we also added a SONAR in front of the Minibaja. The control objective is to keep a constant distance between the vehicle and the wall. To this end, we implemented a simple proportional controller that steer the car to the right if we are to close to the wall, and to the left if we are too far. Some extra logic switches between the proportional controller and the ad-hoc behavior that negotiate the corner. The following is the function used for the steering controller:

```
1: def steering_control(self):
2:     max_s = 15.0
3:     K = 1
4:     error = Setpoint - self.sonar.get_sonar()
5:     front_sonar = self.sonar_front.get_sonar()
6:     if front_sonar > 180:
7:         u = K * error
8:         if u > max_s:
9:            u = max_s
10:        if u < -max_s:
11:           u = -max_s
12:    elif 170 < front_sonar <= 180:
13:        u = 20.0
14:    elif 160 < front_sonar <= 170:
15:        u = 25.0
16:    else:
17:        u = 30.0
18:    return u # This is the
steering angle
```

Lines 4 and 5 read the SONARs and compute the error signal. If the distance between the front SONAR is bigger than 180 inches, we execute the proportional controller. Lines 8 to 11 check if the output of the controller is beyond the steering range of ±15 degrees used to follow the left wall. If the front SONAR measurement is smaller than 180 inches, a right turn is executed.

4.1.2 Tethering Algorithm

To equalize the RSSs difference, we query the Rajant radio mounted in the Minibaja to get the RSS from the base station, and the RSS from the leader. The difference between these measurements is used as the error signal for a proportional controller that commands the speed of the vehicle. The following is the code used for the RSS controller is shown below. Lines 3 to 5 query the Rajant radio to get the RSSs. Line 7 computes the speed. Lines 8 to 11 check that the speed is in the range 0 to 1 meters per second:

Downloaded From: http://proceedings.spiedigitallibrary.org/ on 11/07/2016 Terms of Use: http://spiedigitallibrary.org/ss/termsofuse.aspx

```
1: def follow_the_leader(self):
2:     """Follow the leader using rss."""
3:     rss = self.rss_server.get_rss()
4:     leader = float(rss['leader'])
5:     base_station = float(rss['base_station'])
6:     k = 0.5
7:     u = k * (base_station - leader)
8:     if u < 0:
9:         u = 0
10:    elif u > 1:
11:        u = 1.0
12:    return u # This is the vehicle speed
```

4.1.3 Experimental Results

Figures 13 and 14 show the signals logged during one of the experimental trials. Figure 13 shows the left SONAR signal and the corresponding steering angle. In this case, the setpoint was 75 inches. Initially, the vehicle is too close to the wall, and thus, the controller steers the car to the right. When the distance is greater than the setpoint, the controller steers the car to the left, keeping a constant distance with respect the left wall. Figure 14 shows the RSSs from the base station and from the leader. Initially, the RSS signal from the base station is greater than the RSS from the leader (we are too close to the base station), thus, the controller makes the Minibaja to move forward, making the RSSs approximately equal. Then the leader moves forward, which causes a decrease of its RSS. The controller responds by moving the Minibaja forward. Notice that by the end of the experiment, the RSS from the leader is higher than the RSS from the base station. However, since we are only allowing positive speeds, the controller can not compensate this for this difference (this is because in this experiment we did not have the ability to drive the vehicle backwards).

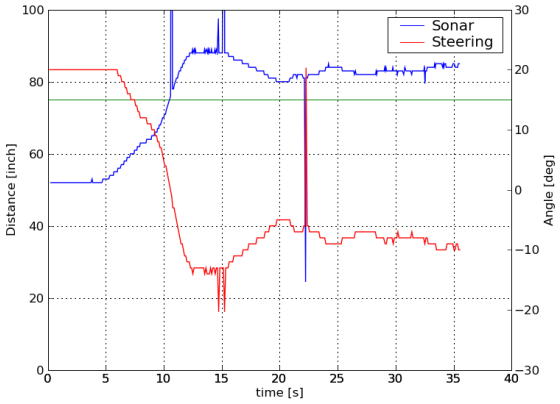


Figure 13: Steering controller results.

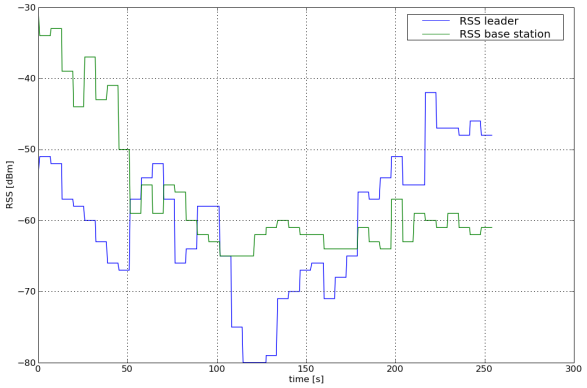


Figure 14: Experimental result for the RSS controller.

4.2 Underground Human-in-the-Loop Radio Tethering Experiment with One and Two Autonomous Followers

We performed two experiments simulating how a robot would navigate and position themselves for optimal relaying of data. In these experiments "human-in-the-loop" means that a human moved in response to commands from the AMR interface. Specifically, a laptop was associated with a radio and observed the RSS between the unit in front of and behind the radio. Computations were made (similar to the previous section) and then directions were displayed to the human, directing them to either "MoveForward," "MoveBackwards," or "Stop." In the first experiment there was one leader node far away from the base station and the AMR (a human in this case) moves to position itself at an equal radio signal strength away from both the leader and base station, acting as a relay for data from the leader to the base station. Figure 15 shows that the system indeed did command the AMR (human) to move to the correct distance so that the SNR between the AMR and the base is essentially the same as between the AMR and the leader.

The same type of human-in-the-loop experiment was performed using two AMRs (humans). The results are shown in Figure 16, this time with two humans acting as AMRs, trying to equalize the signal strength between each node. Unfortunately the battery died on one laptop leaving AMR 3 not finished with the experiment and not at equal radio signal strength. However, up to that point it can be seen that the radio signal strength (SNR) from both the leader and follower become equal as the AMRs navigate by the signal strength measurements they receive.

Downloaded From: http://proceedings.spiedigitallibrary.org/ on 11/07/2016 Terms of Use: http://spiedigitallibrary.org/ss/termsofuse.aspx

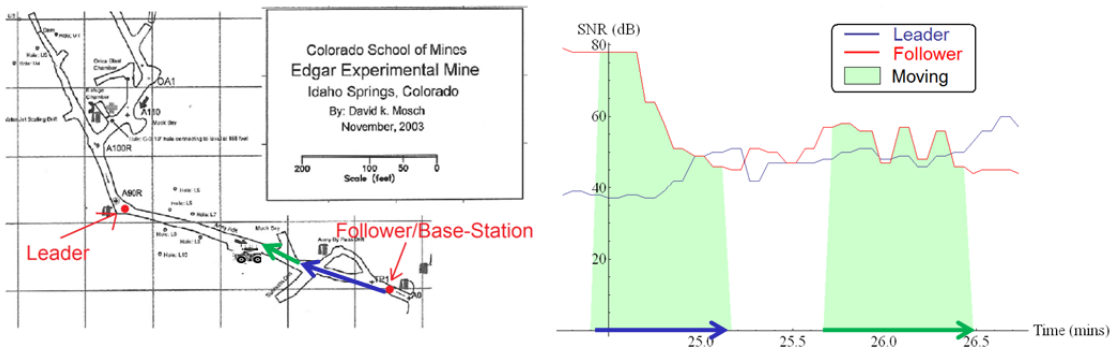


Figure 15: Human-in-the-loop experiment (one AMR).

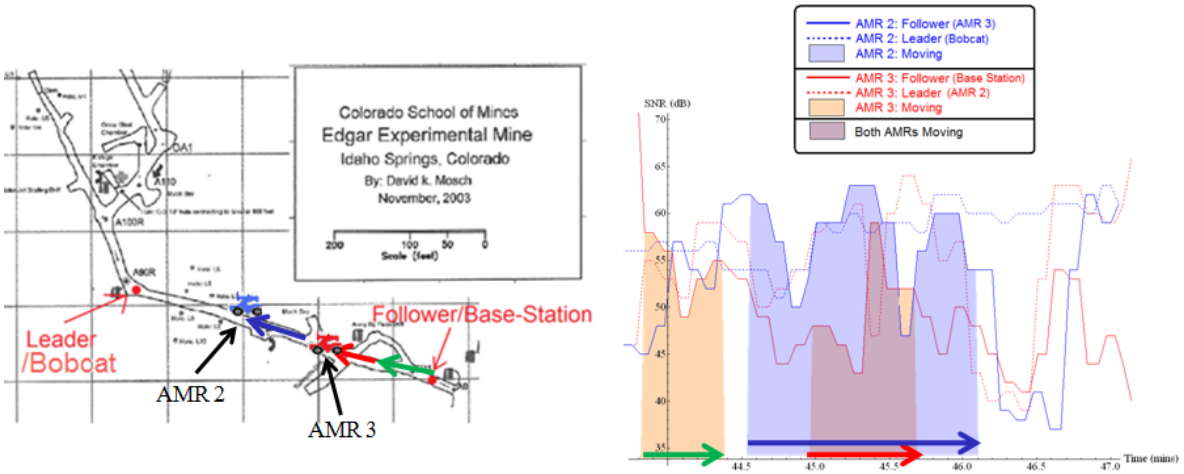


Figure 16: Human-in-the-loop experiment (two AMRs).

4.3 Underground Human-in-the-Loop Radio Tethering Experiment with One Semi-Autonomous Follower

The final experiment reported here used a single semi-autonomous follower. Here "semi-autonomous" means that a human performed steering, as wall-following in the mine was not functional, due to damage sustained by several of the key sonar sensors during transport to the test site. As in the outdoor experiment described above, forward motion of the AMR was commanded by the RSS control algorithm and this command was executed autonomously. The only role of the human was to steer during AMR movements. The AMR navigated from the base station to the optimal position, guided by only the signal strengths to the leader and base station (follower) nodes. The AMR would gather radio signal data for 30 seconds, and then move to an estimated position. It would then repeat the gathering of data for 30 seconds and reposition again if necessary. The AMR repeated the gathering of data and moving five times during the experiment, until it ultimately reached its goal of equal signal strength between the leader and base station (follower). Figure 17 shows the resulting signal strength plots and distance versus time. Figure 18 shows the corresponding locations in the mine. Clearly the algorithm was effective in achieving a balanced RSS between the AMR and the base and between the AMR and the leader.

5. CONCLUSIONS

We have described a series of successful experiments aimed at demonstrating AMRs for radio tethering in underground tunnels. Additional work remains to demonstrate a complete autonomous AMR. Specifically, it remains to develop control logic to recognize side passages and properly account for them so as to maintain correct center-of-the-tunnel navigation.

Downloaded From: http://proceedings.spiedigitallibrary.org/ on 11/07/2016 Terms of Use: http://spiedigitallibrary.org/ss/termsofuse.aspx

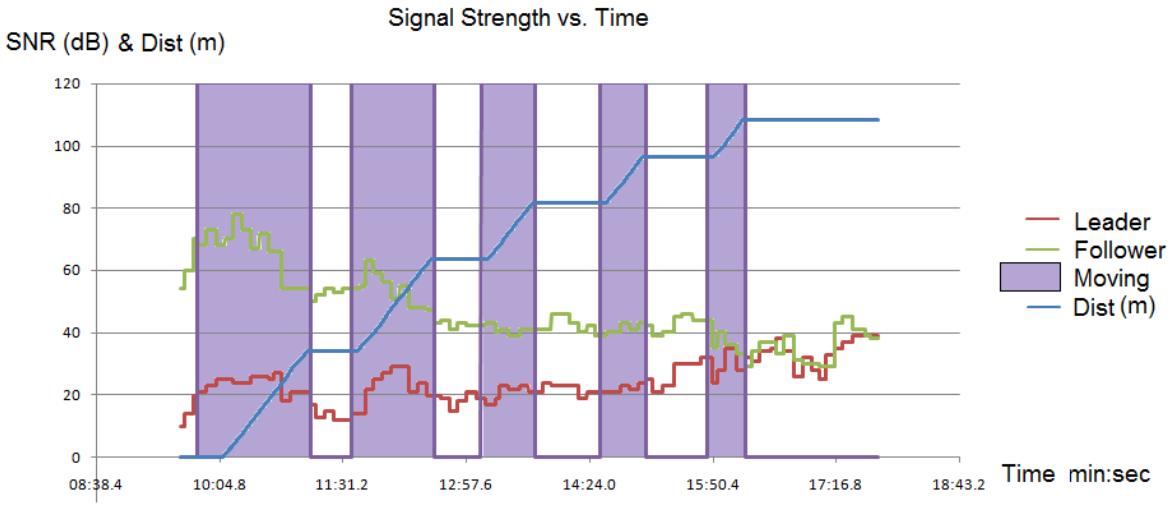


Figure 17: SNR and distance versus time for the semi-autonomous AMR experiment.

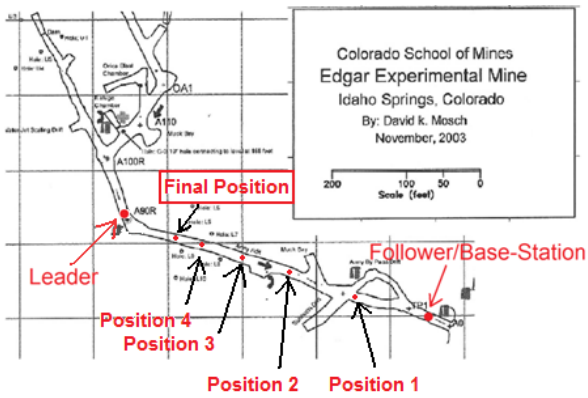


Figure 18: AMR start/stop positions.

ACKNOWLEDGEMENTS

The authors gratefully acknowledge the support of the National Institutes of Health under grant number 1R01OH009612-01.

REFERENCES

[1] Morris, A., Fergyson, D., Omohunndro, Z., Bradley, D., Silver, D., Baker, C., Thayer, S., Wittaker, C., and Wittaker, W., "Recent Developments in Subterranean Robotics," *Journal of Field Robotics,* vol. 23, no. 1, pp. 35-57, 2006.

[2] Moore, K. and Weiss, M., "Autonomous Mobile Radios for Enhancing Wireless Communications though Wireless Tethering in Tunnel-Like Environments," *Proceedings of 2009 IFAC Workshop on Networked Robotics,* pp. 31-36, Golden, CO, Oct. 2009.

[3] Moore, K., Whitehorn, M., Weinstein, A., Xia, J., "Towards a UGV Autopilot," *Proceedings of SPIE Security and Defense Symposium*, Orlando FL, April 2009.

Downloaded From: http://proceedings.spiedigitallibrary.org/ on 11/07/2016 Terms of Use: http://spiedigitallibrary.org/ss/termsofuse.aspx

PROCEEDINGS OF SPIE

Unmanned Systems Technology XII

Grant R. Gerhart
Douglas W. Gage
Charles M. Shoemaker
Editors

6–9 April 2010
Orlando, Florida, United States

Sponsored and Published by
SPIE

Volume 7692

Proceedings of SPIE, 0277-786X, v. 7692

SPIE is an international society advancing an interdisciplinary approach to the science and application of light.

Downloaded From: http://proceedings.spiedigitallibrary.org/ on 11/07/2016 Terms of Use: http://spiedigitallibrary.org/ss/termsofuse.aspx

The papers included in this volume were part of the technical conference cited on the cover and title page. Papers were selected and subject to review by the editors and conference program committee. Some conference presentations may not be available for publication. The papers published in these proceedings reflect the work and thoughts of the authors and are published herein as submitted. The publisher is not responsible for the validity of the information or for any outcomes resulting from reliance thereon.

Please use the following format to cite material from this book:
Author(s), "Title of Paper," in *Unmanned Systems Technology XII*, edited by Grant R. Gerhart, Douglas W. Gage, Charles M. Shoemaker, Proceedings of SPIE Vol. 7692 (SPIE, Bellingham, WA, 2010) Article CID Number.

ISSN 0277-786X
ISBN 9780819481566

Published by
SPIE
P.O. Box 10, Bellingham, Washington 98227-0010 USA
Telephone +1 360 676 3290 (Pacific Time) · Fax +1 360 647 1445
SPIE.org

Copyright © 2010, Society of Photo-Optical Instrumentation Engineers

Copying of material in this book for internal or personal use, or for the internal or personal use of specific clients, beyond the fair use provisions granted by the U.S. Copyright Law is authorized by SPIE subject to payment of copying fees. The Transactional Reporting Service base fee for this volume is $18.00 per article (or portion thereof), which should be paid directly to the Copyright Clearance Center (CCC), 222 Rosewood Drive, Danvers, MA 01923. Payment may also be made electronically through CCC Online at copyright.com. Other copying for republication, resale, advertising or promotion, or any form of systematic or multiple reproduction of any material in this book is prohibited except with permission in writing from the publisher. The CCC fee code is 0277-786X/10/$18.00.

Printed in the United States of America.

Publication of record for individual papers is online in the SPIE Digital Library.

SPIEDigitalLibrary.org

Paper Numbering: Proceedings of SPIE follow an e-First publication model, with papers published first online and then in print and on CD-ROM. Papers are published as they are submitted and meet publication criteria. A unique, consistent, permanent citation identifier (CID) number is assigned to each article at the time of the first publication. Utilization of CIDs allows articles to be fully citable as soon they are published online, and connects the same identifier to all online, print, and electronic versions of the publication. SPIE uses a six-digit CID article numbering system in which:

- The first four digits correspond to the SPIE volume number.
- The last two digits indicate publication order within the volume using a Base 36 numbering system employing both numerals and letters. These two-number sets start with 00, 01, 02, 03, 04, 05, 06, 07, 08, 09, 0A, 0B … 0Z, followed by 10-1Z, 20-2Z, etc.

The CID number appears on each page of the manuscript. The complete citation is used on the first page, and an abbreviated version on subsequent pages. Numbers in the index correspond to the last two digits of the six-digit CID number.

Downloaded From: http://proceedings.spiedigitallibrary.org/ on 11/07/2016 Terms of Use: http://spiedigitallibrary.org/ss/termsofuse.aspx