

Bridge-Mounted River Stage Sensors (BMRSS)

ANTON KRUGER¹, (Member, IEEE), WITOLD F. KRAJEWSKI², JAMES J. NIEMEIER³,
DANIEL L. CEYNAR³, AND RADOSLAW GOSKA³

¹Electrical and Computer Engineering Department, The University of Iowa, Iowa City, IA 52242 USA

²Civil and Environmental Engineering Department, The University of Iowa, Iowa City, IA 52242 USA

³IHR-Hydroscience and Engineering, The University of Iowa, Iowa City, IA 52242 USA

Corresponding author: A. Kruger (anton-kruger@uiowa.edu)

This work was supported by the Iowa Flood Center at the University of Iowa, Iowa City, IA, USA.

ABSTRACT We have developed a robust sensor for mounting on bridges over rivers and streams. These bridge-mounted river stage sensors (BMRSS) make periodic measurements of the distance from the sensor to the water level below. Properly interpreted, these measurements provide river-stage information, data of great importance to society and crucial to effective flood forecasting. The traditional approach to river stage measurement is the installation of pipes in rivers, digging stilling wells, and the construction of attendant brick-and-mortar infrastructure. The cost of this approach limits the deployment to larger rivers. In most instances, river-stage data from smaller tributaries are few, even though such data can greatly enhance the quality of flood-forecasting models' outputs. In contrast, BMRSS units are an order of magnitude less expensive and allow for widespread deployment. BMRSS units incorporate an ultrasonic distance-measuring module, a solar panel/battery/charge controller, and a GPS receiver. In recent years, the Internet access through commercial cellular networks has become ubiquitous, even in most rural areas. BMRSS units incorporate cell modems and transmit data through the Internet to servers at the Iowa Flood Center. Here, the data are ingested into relational databases and made available to flood forecasting models and information systems. We have deployed and operated more than 220 BMRSS units across Iowa, many for several years continuously.

INDEX TERMS Sensor systems and applications, cellular networks, hazards, instrumentation and measurement.

I. INTRODUCTION

River stage (height of water level above a locally defined reference elevation) information is of great importance to society. In conjunction with an empirically determined functional relationship, accurate river stage information allows one to estimate good streamflow discharge (i.e., the quantity of water that passes a location). These estimates are used by officials charged with water- and hazard management, infrastructure design, and environmental- and hydrologic considerations. In addition to streamflow, information on the river stage itself is of interest, since it can indicate whether flooding is imminent. In flood-prone areas, officials rely heavily on trends in river stage to determine the appropriate response, which may include mandatory evacuations. River stage, streamflow, geographical information, and rainfall estimates are used as inputs of flood forecasting models. In the United States, the United States Geological Survey (USGS) collects and publishes river stage and other data on larger rivers. However, flood forecasting models [1] can benefit from river stage measurements on smaller rivers and streams

upstream from major rivers, where traditional methods of making river stage measurements are cost-prohibitive.

In June 2008, the Midwest in the United States experienced devastating floods [2], [3]. Fig. 1 shows some of the flooding in Eastern Iowa. During the late spring, early summer of 2008, parts of the states of Indiana, Illinois, Iowa, and Wisconsin received copious amounts of rainfall. Iowa was particularly hard-hit and some rivers exceeded their 500-year levels [4]. Much of the downtown of Iowa's second largest City, Cedar Rapids, situated in Eastern Iowa, was flooded. At the time, damages caused by the 2008 floods and tornadoes were considered the sixth largest FEMA (United States' Federal Emergency Management Agency) disaster declaration based on estimated financial public assistance. A problem that arose during the event was the scarcity of river stage information. Large areas in Eastern Iowa depended on two USGS streamflow gauging stations. During the flooding event, one of the stations was destroyed. The devastation caused by the flood led to the creation of the (Iowa) state-supported Iowa Flood Center (IFC), located at the University of Iowa.



FIGURE 1. Widespread floods in Iowa in 2008 showed the need for improved sensing of river stage in Iowa, especially on smaller rivers and streams.

Given the paucity, yet importance of river stage information with respect to flood forecasting, IFC personnel have developed the concept of mounting many comparatively inexpensive river stage sensors on bridges that span smaller rivers and streams. The sensors can make frequent (by default every 15 min, but could be as frequent as every 5 min) stage measurements and make the data available on the Internet with little latency. Bridges are natural locations to deploy such sensors, since they provide sturdy mounting platforms as well as river access infrastructure. Further, high water or even overtopping of bridges is a major concern for officials from a safety perspective. Thus, sensing river stage collocated with bridges is of interest for road-safety as well as for flood forecasting. In addition, bridges provide a secure environment to deploy sensors (we discuss this in more detail below).

Three Electrical and Computer (ECE) undergraduate students at the University of Iowa designed, built, and tested an initial prototype device in 2009–2010 as their Capstone Senior Design project. Some of the authors mentored the students with respect to the design aspects of the electronics, while others mentored and assisted the students with respect to properly enclosing and mounting the device on a bridge across a stream in Iowa City. This stream has a USGS stream flow station approximately 30 m upstream from the foot-bridge used as a mounting platform. This station provided convenient comparison data for the students' prototype. The ultrasonic sensor used in this prototype had a limited sensing range and is better suited for indoors, robotics applications. Still, it provided useful proof-of-concept measurements. The prototype used a cellular modem to transmit data via the Internet to a server at the Iowa Flood Center.

The lessons learned from the prototype are: (a) the concept of making river stage measurement using ultrasonic techniques and transmitting the data via cell modems is sound, (b) the http protocol is reliable and efficient for data transmission, (c) temperature effects on measurements should be considered, and (d) to develop a successful unit that can be

deployed in the field and left to operate unattended for years will require a multidisciplinary team.

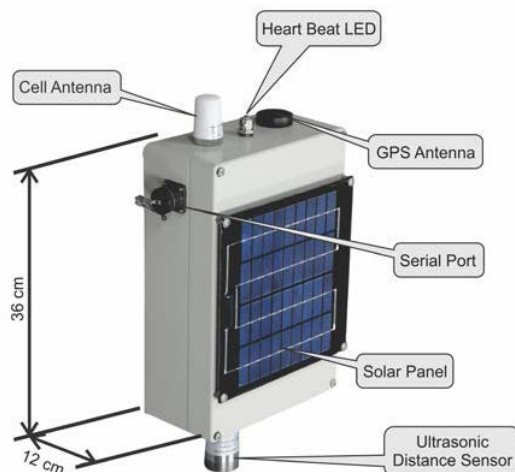


FIGURE 2. A Bridge-Mounted River Stage Sensor.

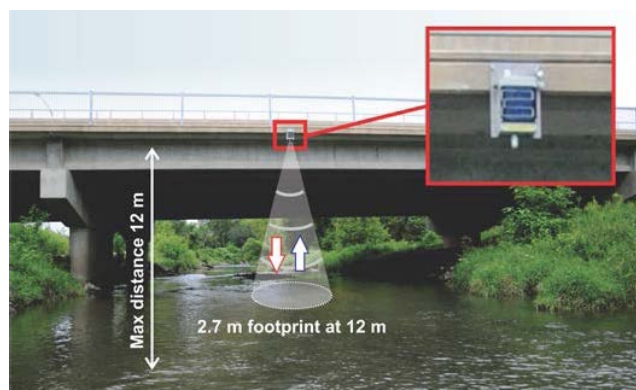


FIGURE 3. An installed BMRSS on a bridge across a river in Cedar Rapids, Iowa.

Following the success of the prototype, the Iowa Flood Center developed a self-contained, compact Bridge-Mounted River Stage Sensor (BMRSS) for deployment on smaller rivers and streams. Fig. 2 is an annotated photograph of a BMRSS that shows some of major components: an ultrasonic distance sensor, a solar panel, a GPS antenna, a heartbeat LED indicator, a cellular modem antenna, and a serial port. Fig. 3 shows an installed BMRSS. The physical construction of BMRSS units is robust so that they function unattended for many years. BMRSS' embedded software (i.e., firmware) is equally robust and accounts for dropped calls, remote server problems, bugs or peculiarities in cell modem firmware, and so on.

In recent years, the IFC has also developed the Iowa Flood Information System (IFIS). IFIS [5] ingests real-time data from several sources (including BMRSS units) and provides officials and the public with real-time flood monitoring. Fig. 4 provides an overview of the system. The first BMRSS was deployed in fall 2010 and there are currently 223 deployed

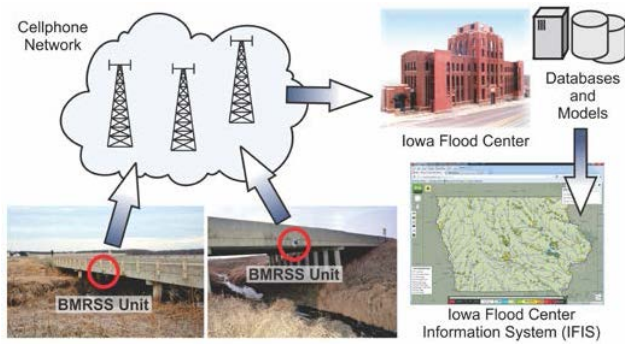


FIGURE 4. Overview of the BMRSS network. Sensors mounted on bridges over smaller streams and rivers make periodic measurements of water levels and transmit the data via cellular networks to Iowa Flood Center servers. The data are ingested into databases that are accessed by software systems such as flood forecast models or the Iowa Flood Center Information System.

across the state of Iowa, providing continuous river stage estimates of 136 rivers and streams.

II. USGS STREAMFLOW MEASUREMENTS

In the United States, the USGS is tasked with collecting and publishing river stage measurements and streamflow estimates. Since BMRSS river stage measurements can complement USGS measurements, a brief overview of USGS measurements is useful. There are approximately 7,000 stream gauge stations across the United States. USGS river stage data are typically recorded at 15- to 60-minute intervals. Data are relayed to USGS servers via satellite, telephone, and/or radio telemetry, and are generally available on the Internet within minutes of arrival. However, data are stored onsite, and then transmitted to USGS offices every 1 to 4 hours (determined by the data relay technique) so there is typically a 1-hour minimum latency. In some instances, the USGS may increase the data transmission rate, but this is not always feasible.

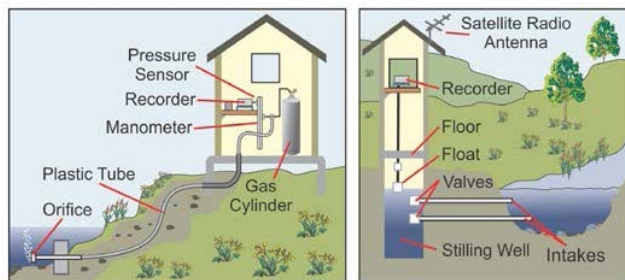


FIGURE 5. At some locations, the USGS uses a so-called bubbler system for measuring river stage, whereas at other locations, pipes connect the river to a stilling well where the river stage is measured.

The USGS streamflow measurements are made at stations that consist of pipes running from the center of the river and a stilling well, housed in a small room. Fig. 5 depicts a typical gaging station. The water depth at the stilling well provides the stage measurement [6]. The purpose of the stilling well

is to remove fluctuations that occur on a short time scale. In electrical engineering parlance, the stilling well is a low-pass filter. At some locations, the USGS uses a so-called bubbler system. A cylinder containing compressed air or nitrogen feeds an orifice that is submerged in the river. A manometer measures and records the gas pressure. The pressure in the tubing is a function of the water column above the orifice so that changes in the river stage produce changes in the gas pressure. From the gas pressure, one can infer the river stage with respect to a known datum (river stage).

In addition to the river stage measurements, the USGS develops and maintains a rating curve at each location. The rating curve functions as a lookup table that maps river stages to corresponding discharges. The USGS continually updates the rating curve by following a well-defined protocol that involves USGS crews making bathymetry- and water velocity measurements. USGS streamflow stations are expensive to build and maintain. Consequently, they are most often built on larger rivers, but even then, there are often only a few stations. For example, on the Iowa River, which is about 330 miles long, there are only 12 such gauges. Further, from a flood-forecasting perspective, river stage information on smaller upstream tributaries is often more valuable than river stage- and streamflow information on larger rivers. Finally, the number of USGS gauges has decreased in recent years.

TABLE 1. Summary of USGS and BMRSS streamflow measurements capabilities.

	USGS Streamflow	Bridge-Mounted Sensors
Cost	High	Low
Maintenance	Moderate	Very little
River Stage	Yes	Yes
Discharge	Yes	No
Deployment	Best on larger streams	Essentially anywhere there is access
Measurement Frequency	15–60 min	5, 15, 60 min
Normal Latency	1 hour	5 min
Accuracy	5%	1% of air distance

By contrast, the BMRSS units are primarily concerned with river stage, and in particular with the difference between river stage and bank overflow. That is, the headroom before flooding will occur. The aim is not to make streamflow estimates, though in principle one could, albeit at perhaps reduced accuracy compared with USGS streamflow estimates. Rather, a major design consideration was to keep the cost down without comprising performance from a flood forecasting perspective. One can then deploy many BMRSS units—and very importantly—one can deploy them on tributaries to larger rivers. This provides better information to flood forecasting models. Another consideration is that a BMRSS can provide high-frequency river stage estimates with low latency (5 min). Table 1 summarizes USGS streamflow gauges and BMRSS capabilities.

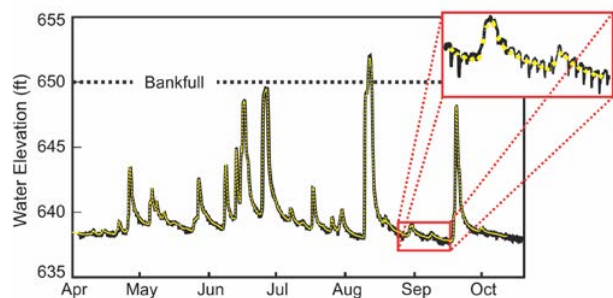


FIGURE 6. Sample river stage hydrograph data obtained using a BMRSS (black), overlaid on collocated USGS stream gauge data (yellow). The inset shows small diurnal variations in the BMRSS data. These are related to the effects of air temperature on the ultrasonic distance measurement module (see text).

III. SAMPLE DATA

Even though the focus of this manuscript is the engineering aspects of the BMRSS system, here we present some river stage data collected by a BMRSS unit. Fig. 6 is a time series of river stage, and is exemplary of such plots, called hydrographs. Hydrographs are plots that show a hydrologic variable over time. They are commonly used for visualizing stream discharge, but are also used for river stage. The shape of a river stage hydrograph depends on factors such as the river bathymetry, and how water entered the channel. As an example, a large rainfall event on already-saturated soil, or in an urban setting where there is little infiltration into the soil, can result in significant and quick runoff of surface water into a stream. An electrical engineering analogy could be that the influx of water is analogous to a signal pulse at the input of an electrical circuit. Depending on the size of the stream, this can lead to rapid rise in river stage. The corresponding hydrographs show a sharp increase and then a decay that is reminiscent of an exponential ($1 - e^{-t/\tau}$) function.

An alternative scenario is that the river stage fluctuates not because of input from surface runoff from rain, but from the so-called base flow. Base flow [7] is the combination of water that flows upwards from the deeper subsurface into the river channel, and delayed flow through the shallow subsurface into the channel.

Hydrographs of river stage where base flow is the major source of water normally do not show rapid changes. Riparian vegetation that access subsurface water can modulate base flow levels. Because the vegetation activity (water uptake, transpiration) has a diurnal cycle, the water the vegetation removes has a diurnal cycle, and so does the river stage. It is possible to measure these diurnal cycles. However, one has to be careful to discern between actual diurnal river stage changes and diurnal instrument changes, resulting from diurnal temperature changes that affect the instrumentation. Later in the manuscript, we explore the effect of temperature on a BMRSS unit in detail.

IV. BMRSS HARDWARE DESCRIPTION

Fig. 7 shows a block diagram of the BMRSS electronics, and Fig. 8 shows annotated photographs of a unit.

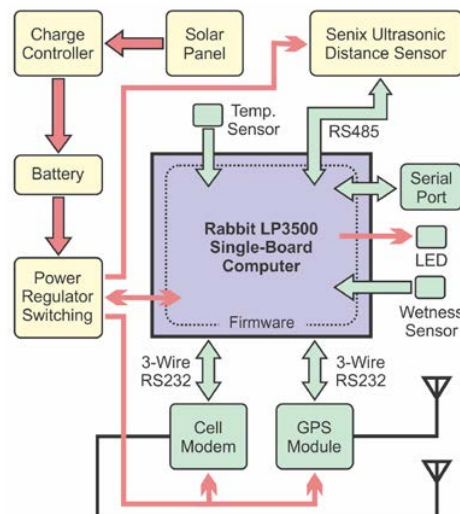


FIGURE 7. Block diagram of the major BMRSS electronic components.

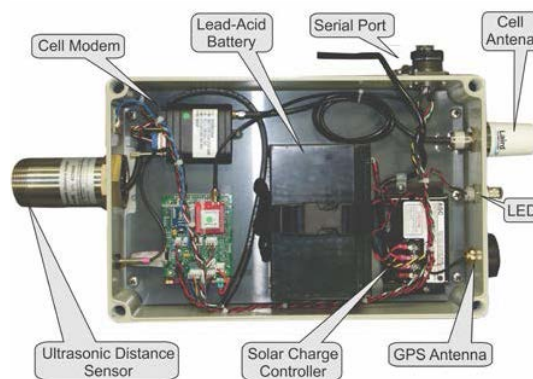


FIGURE 8. BMRSS Internal view showing major components.

As these figures show, a BMRSS incorporates an ultrasonic distance measurement module, a cell modem and antenna, GPS module and antenna, battery and charge controller, serial port connector, and electronics. A small solar panel is mounted on the enclosure. In addition to providing power, the solar panel provides some shading for the enclosure.

A. ULTRASONIC DISTANCE SENSOR

BMRSS units incorporate ultrasonic distance sensor from the Senix Company (www.senix.com). Most deployments use the ToughSonic 50 model, designed for operation up to 15.2 m (50 ft). In some instances, we deployed the ToughSonic 30 model, designed for operation up to about 9.1 m (30 ft). Fig. 9 is a photograph of a Senix ToughSonic 30 module. The modules are available either RS-232 or RS-485 serial data interfaces. We use modules with RS-485 interfaces. The RS-485 is a very mature, multi-drop, serial networking standard, and at the time when we designed the first BMRSS units, we contemplated operating selected units in a networked fashion. This could be useful on a large river. Consequently, we picked this model number. In reality, we



FIGURE 9. Photograph of a Senix ToughSonic 30 ultrasonic distance module used in some of the BMRSS units. It has a range of up to 9.1 m (30"). In most locations, we use the ToughSonic 50, which has a 15.2 m range (50").

do not operate BMRSS units in a wired network fashion, and we do not need RS-485. Rather, an RS-232 interface would suffice. However, the SBC we use has support for RS-485. The communication protocol is Modbus.

TABLE 2. Summary of the senix ultrasonic distance measurement modules used in the BMRSS units.

Input Power	10-30 VDC. Power consumption at 12 V is 40 mA.
Environmental	Temperature -40 °C to 70°C Humidity 0-100%
Repeatability	Nominal 0.2% of range at constant temperature
Housing	Stainless steel, potted and sealed, liquid tight
Protection	IP68, NEMA-4x, NEMA-6P
Serial interface	Modbus RS-485
Dead band	TSPC-15S-485: 25.4 cm, TSPC-21S-485: 30.5 cm
SenixVIEW enhanced capabilities	Adjustment: Over 60 functionally organized sensor parameters. Viewing: Large and small displays, chart, filtered & unfiltered. Analysis: Measure statistics; data logging with replay and Excel export.

Table 2 summarizes some of the modules' characteristics. The table also lists the SenixVIEW software. This software allows one to configure many parameter of the module. We set the data collection and filtering as follows. When instructed to measure, the module makes 15 measurements, each 0.25 s long. The measurements are grouped in five sets of three. For each of the sets one can take the average, the longest, or shortest distance.

For our application, it makes sense to take the longest distance. This is because there may be objects floating down the river or a bird flying through the measurement cone, and such events will lead to shorter distances. On the other hand, we know, a-priory, the maximum distance from the sensor to the bottom of the river, so it is easy to recognize false long-distance readings. After this step, one is left with 5 distances. These are then averaged. All of this processing is done by the Senix module, which simply outputs the final answer. Returning to the averaging, the SenixVIEW software calls the averaging "boxcar", but it is not clear exactly how the averaging is performed. We suspect that the module simply sums the 5 distances and divides the results by 5.

B. ENCLOSURE AND MOUNTING PLATES

The BMRSS enclosure is a sturdy die-cast aluminum box with dimensions indicated in Fig. 8. The holes for the

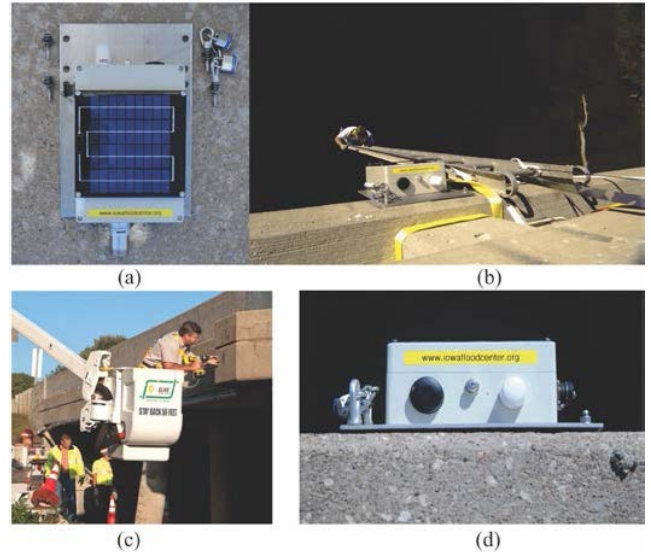


FIGURE 10. (a) BMRSS mounted on stainless steel mounting plate before attaching to a bridge. (b), (c) Attaching a BMRSS to a bridge. (d) A BMRSS after installation on a bridge.

ultrasonic distance sensors, cell- and GPS antennas, serial port, heartbeat LED and so on, are precision-machined using Computer Numerical Control (CNC) techniques. The electronics are mounted on a solid, CNC-machined baseplate. Many of the BMRSS units are assembled by students at the University of Iowa. Workers are well-trained and we have clear assembly protocols in place. For example, key fasteners are tightened using a torque-wrench to ensure that they are not too loose and compromise sealing, or too tight, which can lead to micro fractures which can also compromise sealing. Student helpers are trained and supervised by a mechanical engineer and an electrical engineer.

Siting and mounting BMRSS units are major undertakings. Siting is often a compromise between optimal placement from a flood-forecasting/hydrologic perspective, and access. Deployment requires obtaining permission from bridge owners and coordinating with various authorities. Additionally, the safety of personnel who deploy the units is of utmost importance. Units are mounted to a bridge with the aid of a ~6 mm thick stainless steel mounting plate. To accommodate different bridge structures, there are long (45×32 cm) and short (37×32 cm) mounting plates. The plates weigh about 8 kg. In some instances, a particular bridge requires that we adapt the standard mounting procedure.

C. ELECTRONICS AND SINGLE BOARD COMPUTER

Rather than developing tightly-integrated, custom electronics, our design follows a modular approach, and our preference is to use commercially-available modules for the major subsystems. Consequently, a printed circuit (PCB) "glue" board contains some electronics, but mostly connectors where all the other modules plug into or connect to. Earlier units used a commercial charge controller. In later models,

TABLE 3. The major BMRSS electronics components.

SBC with analog/digital converters, several serial ports	Manages data collection and transmission	Commercial
Solar charge regulator	Power conversion and ensure battery is not over-charged	In-house and commercial
GPS module	Provide accurate time as well as location information	Commercial
Cell modem	Internet connectivity using Several telephone companies	Commercial
Distance measurement module	Makes ultrasonic distance measurement	Commercial
Water intrusion sensor	Detects if water entered enclosure	In-house
Desiccant box	Absorbs moisture trapped in enclosure when it is sealed	Commercial
“Glue” board	Contain glue electronics and connectors that connect various sub systems	In-house
Cell-phone antenna	Multiband	Commercial
Cable harnesses	Connects the various components	Commercial and in-house

the charge controller is integrated with the glue board. Below we describe charge control issues in some detail.

We use a commercial low-power, single-board computer (SBC) that manages the data collection and transmission. The SBC is the Rabbit LP3500 model from Digi International. The SBC manufacturer provides a C language compiler and an extensive library for firmware development. The SBC has military temperature ratings. An attractive feature of the SBC is that it supports up to six serial communication ports that one can configure as RS-232 or RS-485. Communication with the cell modem and GPS module is via RS-232, whereas the ultrasonic distance measurement sensor uses RS-485. Another RS-232 port allows one to communicate with the BMRSS firmware menu to, for example, set the BMRSS identification.

We carefully evaluated all of the subsystems and components with respect to power consumption and temperature operating range. Additionally, we test each individual low-power SBC for compliance with the manufacturer’s power consumption specifications.

D. TIME KEEPING—REAL-TIME CLOCK AND GPS

Each BMRSS incorporates a temperature-compensated real-time clock (RTC) to provide accurate time. Still, BMRSS electronics are subject to large temperature variations and temperatures inside the enclosures can exceed 50° and dip below −30°, and the average BMRSS RTC drifts by about three minutes per year. Whereas this may seem small, our goal is to have BMRSS units run indefinitely without maintenance. Some of the units have been in operation for about 5 years, so that a 3 min per year drift would result in a 15 min discrepancy in 5 years. Thus, BMRSS units contain a GPS module that is used to discipline the RTC once every 5 days. All data are time-stamped with UTC time.

Conversion to local time and accounting for daylight savings time is done at the IFC data ingest server. The GPS also provides location information, which is embedded in the data packets.

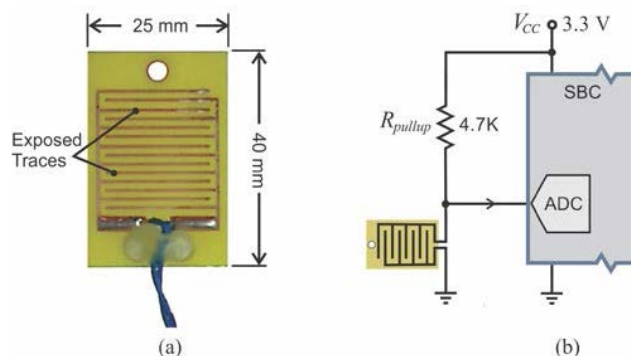


FIGURE 11. (a) The wetness sensor is a PCB with exposed traces. (b) Integration into the BMRSS electronics.

E. WETNESS SENSOR

Each BMRSS incorporates a wetness sensor to monitor the conditions inside the enclosure. The wetness information is part of the unit health information that is transmitted to the ingest server. The wetness sensor is a 25×40 mm PCB with two interleaved traces as shown in Fig. 11(a). The traces are exposed and separated by approximately ~1 mm. One of the traces is pulled to 3.3 V via a 4.7K resistor, and the other trace is connected to ground. The pullup resistor and sensor form a resistive voltage divider. Ambient moisture bridges the traces and lowers the sensor resistance and the voltage at the junction. The voltage at the junction is measured using the SBC’s internal ADC (see Fig. 11(b)).

The wetness sensor is installed in the bottom of the BMRSS enclosure. The enclosure cover, serial port, ultrasonic distance sensor, and heartbeat LED have air- and watertight seals. The enclosures are sealed in a laboratory environment with relative humidity between 30% and 65%. We do not specifically monitor the humidity level before sealing the enclosures. Non-condensing water vapor inside the sealed enclosure will lower the sensed voltage a few tenths of a volt. Condensation from the water vapor trapped when the enclosure is sealed will lower the sensed voltage by up to 1.5 V. If there is a breach of any of the seals and water enters the enclosure, the water collects at the bottom of the enclosure, and the sensed voltage drops to 0.1–0.9 V. As we describe in Section VII, BMRSS units undergo a water immersion test before they are deployed. During this test, we monitor the wetness sensors every 5 min. Once in the field, the wetness sensor voltage is measured and transmitted along other system health information during every transmission.

The wetness sensor, even though simple and inexpensive, has proven very valuable in detecting problems with enclosure sealing both before and after deployment.

F. TEMPERATURE SENSORS

The glue electronics board incorporates an Analog Devices' TMP36 analog temperature sensor. The sensor has typical accuracies of ± 1 °C at $+25$ °C and ± 2 °C over -40 °C to $+125$ °C. The sensor consumes less than 50 μ A quiescent current, and provides a voltage output that is linearly proportional to the Celsius temperature. The voltage is measured using the SBC's internal ADC. We choose the sensor because of its low power consumption and because of the SBC's ample number of ADC channels. Earlier BMRSS units had an additional temperature sensor mounted to the enclosure wall. However, both sensors provided very similar temperature reading, so the wall-mounted sensors are not present in later BMRSS units.

The Senix ultrasonic distance module has its own temperature sensor that it uses for temperature compensation. The BMRSS firmware polls the Senix sensor to obtain this temperature measurement and this is transmitted as part of the system health information. The temperature readings from the Senix and BMRSS' TMP36 are different but correlated. The reason why they are different is because the temperature sensors are in somewhat different thermal environments. The Senix module is encased in electronics potting compound, whereas the TMP36 sensor is mounted on a PCB board and is in contact with the air inside of the BMRSS enclosure.

Additionally, the Senix module has an option to compensate for self-heating. We discovered this during laboratory-based distance measurements that appeared to be wrong and subsequent consultation with the manufacturer. According to the manufacturer, in most applications, Senix modules are powered continuously. The modules' self-heating can increase the Senix internal temperature by up to 9 °C. The manufacturer recommends that the module should run for 30 min before use. This is to allow the internal temperature to stabilize. Since we power the Senix module (nominally) 15 s every 15 min, self-heating is not problem for the Senix module in our application. Consequently, we turn off self-heating compensation for the Senix using some undocumented Senix commands, provided by the manufacturer.

Fig. 12(a) is a representative time-series of temperature inside a BMRSS enclosure. As expected, there are diurnal and seasonal cycles. At some sites, the temperature variations exhibit pronounced up- and down jumps. The explanation for this lies in how shade from nearby trees, buildings, and even the bridges they are mounted on, move during over the course of the day. Fig. 12(b) shows the daily average temperature, which was computed as follows. For a particular day the average temperature across all deployed units were averaged. The plot shows that around day number 188 (July 6–7) the daily average temperature inside BMRSS enclosures was the highest. It also shows that the enclosure temperature is about 3 °C higher than the temperature reported by the ultrasonic distance sensor.

Fig. 13 shows the average daily temperature along with the record maximum- and minimum temperatures recorded

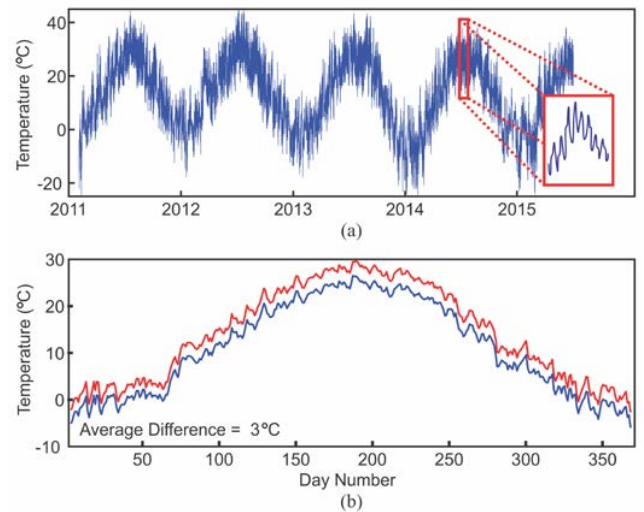


FIGURE 12. (a) Enclosure internal temperature time series for a typical BMRSS unit. The inset shows diurnal temperature variations. (b) Daily average internal enclosure temperature across all deployed BMRSS units over a 4.5-year period. The red trace corresponds to the temperature by the TMP36 sensor (see text) and the blue trace corresponds to what the Senix module reports.

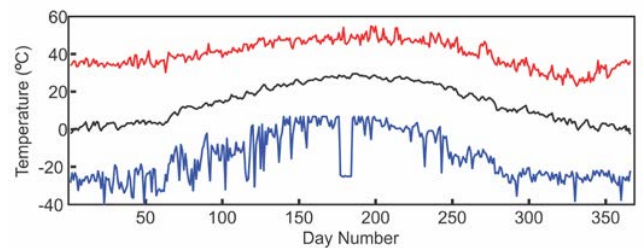


FIGURE 13. Daily average (black) internal enclosure temperature across all deployed BMRSS units. Also shown are record maximum- and minimum (red and blue respectively) internal enclosure temperature recorded by any BMRSS unit on a particular day of the year.

by any sensor inside any of the units (internal enclosure sensor). This shows that the record maximum enclosure temperature was about 55 °C around day 197 (July 15–16). The lowest internal temperature recorded was -40 °C on day 23 (January 23–24).

V. POWER MANAGEMENT

An important design requirement was that a BMRSS could operate unattended for several years. This is to reduce the cost associated with field maintenance. In addition, fewer field visits reduce the risk of injury to personnel. To achieve this goal we carefully considered the power requirements. Our overall strategy has three parts.

First, rather than utilizing idle- or sleep modes of components such as GPS modules and cell modems, we completely turn off units that are not used.

Second, all components were carefully selected for minimum power consumption, and very importantly, we check individual units to ensure that they meet

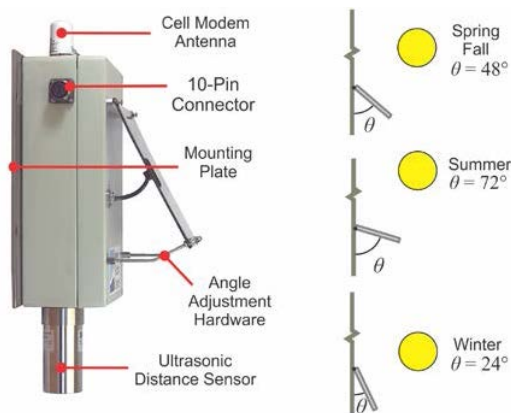


FIGURE 14. To increase the solar insolation, BMRSS panels are tilted. The optimum tilt angle in the center of Iowa varies between 24° and 72°. The default tilt angle is 25°, which works well for most locations in Iowa. The panel mounting hardware allows for adjusting the angle if needed.

manufactures’ specification. As an example, the SBC was selected for low power consumption. During the testing phase, we discovered that some BMRSS units exhibited higher than expected power consumption. We traced this back to SBCs that exceeded the manufacturer’s current consumption specifications. We replaced these SBCs, and instituted a test procedure for each SBC.

Third, we carefully considered and measured the power consumption of *connected* components. For example, the GPS module consumes 44 mA when active and 10 μA when in its standby mode. The GPS module communicates with the SBC via one of the SBC’s serial ports. We discovered that when connected to a SBC serial port, the standby current consumption jumps from 10 μA to 700 μA. We identified this as to how (high, low, or tri-state) and to what voltage level the GPS and SBC place their serial port lines when in idle mode.

Table 4 summarizes the power consumption of the various parts. The power source is a solar panel rated at 6 W maximum power output. At some locations, we use a 12-W panel, and the serial port connector has pins that allow for supplemental external power. A BMRSS unit incorporates a sealed, 12 V, 12 A/h lead-acid battery, and a battery charge regulator to maintain the battery state of charge (SOC). The battery is rated to operate over a temperature range −15° to 50° and has a nominal lifetime of 5 years. As expected [8], we see a reduction in the lead-acid battery capacity during winter. Some batteries of units that have been operating for close to 5 years exhibit signs of reduced capacity. To increase the insolation/irradiance of the solar panel, the BMRSS is normally mounted south-facing on a bridge. In Iowa, rivers generally flow from northwest to southeast or from north to south- or south-west. Consequently, most bridges allow for straightforward south-facing mounts. The panel mounting hardware allows for tilting the panel. At Ames, Iowa, near to the center of the state, the optimum tilt angle varies between about 24° in winter and 72° in summer. The default BMRSS tilt angle is 25, which works well for the majority of sites.

TABLE 4. The major BMRSS electronics components.

The major BMRSS electronics components		
Solar panel	6 W (default), 12 W as needed	>95% of locations use 6-W panels
Sealed lead-acid battery	12 V, 12 Ah nominal	
Ultrasonic distance sensor	Active: 40 mA	Normally turned off. At measurement time makes 15, 0.5 s samples.
GPS module	Active: 44 mA, Standby: 0.7 mA	Normally turned off. Once every 5 days turns on for ~10 min.
Modem R (CDMA)	Idle: ~40–120 mA, Active: 150–450 mA	Normally turned off. Turns on 1 min before scheduled transmission every 5 (default), 10, or 15 min. Average transmission duration is 15 s.
Modem E (GPRS)	Idle: 20 mA, Active: 850 mA	Normally turned off. Turns on 1 min before scheduled transmission every 5 (default), 10, or 15 min. Average transmission duration 15 s.
Single-board computer and glue electronics board	Sleep: 0.6–1.3 mA, Active: 18 mA	Idles for 15 s, then wakes up for 1 s, checks serial port for activity, blinks health beat LED for 120 ms and goes to sleep.
Commercial charge controller	6–10 mA Continuous	Not clear what charge strategy is, unlikely that it is Maximum Power Tracking.
IFC charge controller	85 μA	Maximum Power Tracking.
Heart beat LED	5 mA	Blinks 120 ms every 15 s.

If power problems arise, we can adjust the tilt angle. In about 10 sites where exposure to the sun is poor, we used a mounting adapter and replaced the 6-W solar panel with a 12-W panel.

On two sites the combination of the existing adapter plates and the tilt studs are such that the panel is tilted and rotated to the south for better sun exposure. On one site, where the sensor is mounted on the north-west side of a box culvert, we used an additional south facing-solar panel and rodent resistant wire to supplement the power.

The purpose of the charge controller is to manage the flow of fluctuating energy from the solar panel to the battery, by stepping up/down the voltage- and current levels from the panel. The charge controller also guards against overcharging of the battery, as this can drastically reduce its life or lead to the buildup of gasses and potential explosion [9]. Most commercially available solar charge regulators are designed for larger panel arrays, and there are few options for low power (few Watts) panels.

Our initial designs used a commercial charge regulator, but its quiescent current at 10 mA was too high, and our current design uses a custom-designed charge regulator. This charge regulator implements maximum power point tracking (MPPT) [10] and uses an LT3652 IC. We optimized the LT3652-based design for use with the 6-W panels and the capacity and lead-acid chemistry of the BMRSS battery. The float voltage (the voltage where the regulator halts charging) is 14.4 V. This charge controller has a quiescent current drain of 85 μA, and it has been instrumental in allowing for the placement of BMRSS on East/West facing bridges, and

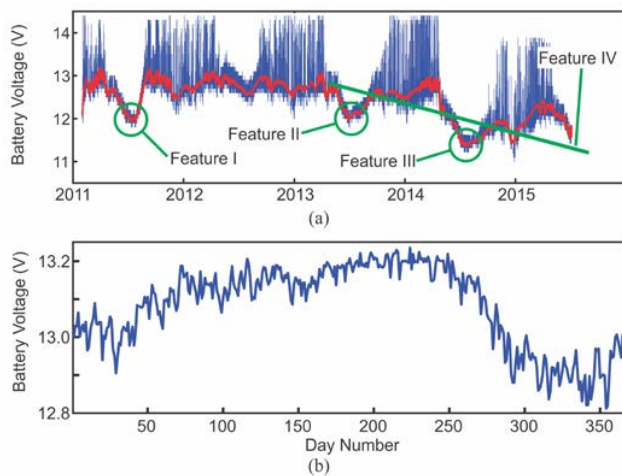


FIGURE 15. (a) Time-series of battery voltage for a representative BMRSS unit. See the text for an explanation of the various features indicated. (b) Average daily battery voltage across all BMRSS units.

bridges with dense riparian zones that typically receive only a few hours of direct sunlight each day.

Fig. 15 shows typical time series for a BMRSS battery voltage, as well as a summary of the daily battery state for the network of BMRSS units. Not unexpectedly, the battery shows diurnal and seasonal variations. With respect to the diurnal variations, during the day the batteries are being charged by the solar panels. Additionally, with warmer temperatures during the day, the battery chemistry is such that battery voltages are higher. With respect to seasonal variations, the default solar panel angle (see discussion above) is optimal for winter. Still, during winter the battery voltages are on average lower than in summer. Again, this is due to the battery chemistry.

There are several other noteworthy features visible in Fig. 15 (a). The first is Feature I. Here the battery voltage dropped and we were concerned that the battery may become depleted. During a field visit, we tilted the solar panel to increase the solar insolation. This clearly had the desired effect and the battery voltage recovered. However, two years later, Feature II appears. Note that the dip is in the middle of summer, when battery voltages are typically at their maximum. Similarly, the pronounced dip a year later (Feature III) also occurred in the middle of summer. The explanation for these dips is that at this location, there is significant vegetation on the bank of the river and in midsummer, this casts shadows on the panel during some parts of the day. The decline in battery capacity over time is evident in this plot (Feature IV). As explained in above, the batteries have a design lifetime of 5 years, and loss of capacity is to be expected.

VI. DATA COMMUNICATION

A. OVERVIEW

Data transmission uses, generally speaking, the same methodology we used and refined in other environmental

networks, see for example, [11]. A BMRSS incorporates a cell modem that, along with a data plan, provides Internet connectivity. The BMRSS SBC communicates with the modems through a three-wire RS232 interface [12]. A modem has two states. In the command state, the SBC communicates with the controller embedded in the modem. The SBC can send configuration commands, query for received signal strength indicator (RSSI), and so on. No data are transmitted.

The second state is when the modem is in the on-line or pass-thru state where it transmits data to a remote receiver. The modem has an embedded Packet Assembler/Disassembler or PAD. The PAD assembles data as Internet Protocol (IP) [13] packets before transmission. The PAD also converts incoming IP packets. The PAD provides both TCP and UDP [13] functionality that allows one to use well-known protocols such as ftp, telnet, and http for communication, or one could develop a custom protocol that uses the underlying TCP/IP mechanism. Additionally, newer modems' firmware incorporates ftp clients. Thus, a BMRSS could "ftp the data" to remote servers.

After careful consideration we decided to use the hypertext transfer protocol (http) [14]. Protocols such as ftp and telnet [13] were designed for interacting with a human operator through a text terminal interface. It is certainly possible to automate ftp and telnet data transfers, but it is difficult. By contrast, interacting with a web server via http protocol is simple.

One advantage of using http is that it is well-documented and straightforward to use. Further, webserver technology is highly-developed and scales very well (i.e., can handle thousands of concurrent sessions), and there are excellent open-source web servers available. We use Apache, and found it robust, fast, and able to handle many concurrent sessions. There are mature and feature-rich server-side processing engines available (we use PHP) that work well with Apache and a relational database (we use Postgress). We transfer data packets uncompressed and in plaintext. This works well, since the data packets are small.

Cell coverage is rapidly improving, but not one single telephone company provides reliable coverage at every location in Iowa. Consequently, we use two different providers, and our current installed base has an approximate 50/50 split between the providers. The overall operation for the providers is similar, but each has its own quirks. One provider uses the General Radio Packet Service (GPRS) of the Global System for Mobile Communications (GSM) standard [15], [16], whereas the provider uses CDMA technology [17], [18]. This dictates that we use two types of incompatible cell modems. The modems use serial (RS232) interfaces and an AT-command set [19]. Whereas many of the core AT commands are similar between modems, commands related to data transmission are different. Thus, the firmware for a BMRSS must accommodate both modems.

B. MODEM PROVISIONING

After purchase, a cell modem must be provisioned. Provisioning is the term used by the telephone companies for the process of configuring the cell modem and data plan and accounting so that the modem can access the Internet via their network. Provisioning for the first BMRSS units was complicated and quite time-consuming. These early units incorporated GSM/GPRS modems, which uses Subscriber Identity Module (SIM) cards. In principle, a user should be able to move a SIM between GSM devices without loss of service or the need to reconfigure the device. However, our first provider tied data plans to modems' International Mobile Equipment Identity (IMEI) number. (IMEI numbers uniquely identify wireless devices.) Additionally, the provider required that each modem be programmed with a unique username and associated password. To fully provision a cell modem would involve the following steps

1. Purchase cell modem and obtain IMEI and serial numbers from manufacturer.
2. Provide the GSM/GPRS service provider with the IMEI number and the cell modem serial number.
3. The GSM/GPRS service provider would then provide a phone number and a SIM card. Early on, the provider supplied, for each set of phone number, SIM card, IMEI number and modem serial number, a unique username/password pair.
4. Establish serial port connection with the cell modem.
5. Upgrade the modem's firmware if needed.
6. Program the phone number, user name, and phone number into the cell modem.
7. Program other settings, including those required by the provider, into the modem.
8. Insert the SIM card into the cell modem with corresponding IMEI and modem serial numbers.
9. Cycle power.
10. The modem should then "log on" to the GSM network and then the GPRS network. This process is called registration or registering.
11. Check proper registration to the GSM network.
12. Check proper registration to the GPRS network
13. Access a remote server on the Internet to check Internet access.
14. Save the settings in non-volatile memory.

We present provisioning as a methodical sequence, but in practice each of the steps listed has potential complications. For example, with respect to Step 2, our first modem vendor, even after multiple requests, provided only printed sheets containing the information. The vendor went out of business and the new vendor supplied the information in electronic form.

With respect to Step 4, the factory default serial interface settings (baud rate, flow control, number of bits, etc.) for some batches of cell modems were different (and not documented) than most of the other batches. To establish connection with the modem required a search of all the possible combinations. For example, try 8 data bits, no parity bits,

1 stop bit, 9600 bits per second baud rate, and no flow control. That is: 8N1 @ 9600 bps, no flow control. If this does not work, try 8N1 @ 14400 bits per second, no flow control, if that does not work, try another combination, and continue the process until one can communicate with the cell modem.

```

1      AT+IPR=19200
2      AT+IFC=0,0;$IFC=1;$AREG=1;$HOSTIF=2
3      AT+CGDCONT=1,"IP","isp.cingular"
4      AT%CGPCO=1,"ISP@CINGULARGPRS.COM,CINGULAR1"
5      AT&W

```

FIGURE 16. Snippet of showing some of the AT commands required to properly configure an earlier GSM modem.

With respect to Step 7, Fig. 16 is a snippet of commands that one would send to the modem for this step. The first line sets the modem DTE baud rate. The second line configures the modem to attempt to automatically register to the GSM/GPRS networks. The third line has the information needed to register to the GSM network and the 4th line has the information needed to register on the GPRS network. The 5th line saves the settings to the modem's non-volatile memory.

The telecommunications (cell phone) industry is dynamic and characterized by fierce competition, mergers and acquisitions, and constant technological advances. Referring to lines 3 and 4 in Fig. 16, this modem accesses the Cingular Wireless network. Through acquisition and rebranding, the company now operates under the AT&T banner. For each cell modem/data plan, several undocumented aspects are small but critical for reliable operation. These range from bugs in modem firmware, modem documentation discrepancies, and modem flow control and baud rate problems. To ease provisioning we developed simple yet very useful software tools. One such tool is a Visual Basic® application that runs on a PC/Windows® platform. This application sniffs every COM port on the PC and tries the all the plausible computer-modem DTE settings. Once it finds the modem, it programs it with the desired BMRSS settings. As another example, the terminal emulator program we use allows one write Visual Basic for Applications (VBA) scripts and associate scripts with, say function keys, on a PC. Fig. 17 shows part of a script used for programming settings into cell modems.

Fortunately, provisioning has become dramatically simpler over time. Cell phone providers have relaxed requirements such as a unique username and password for every modem. Currently, cell modem vendors perform most of the provisioning before shipping the modems to us. Still, the cell modem manufacturers are continually improving their hardware and provide firmware updates for existing modems, and new cell modem firmware sometimes introduces backward incompatibility problems. Thus, the BMRSS firmware may have to adapt to multiple versions of the same manufacturer's modem.

C. OTHER DATA COMMUNICATION CONSIDERATIONS

Other problems relate to the data plan and associated services. For one provider, the latency for the successful connection


```

:
' GPRS network registration
' With AT$AREG=1, performs auto GPRS registration.
  crt.Screen.Send "AT$AREG=1" &vbCr
  crt.Screen.WaitForString "OK"
' Configure as client. The ATD command will start
' a TCP PAD session. AT$ACTIVE=1 is the default,
' but just to be sure we set it again.
  crt.Screen.Send "AT$ACTIVE=1" &vbCr
  crt.Screen.WaitForString "OK"
' If AT$ACTIVE=1 (the default), then AT$HOSTIF=2
' establishes a TCP PAD session when ATDT is issued.
  crt.Screen.Send "AT$HOSTIF=2" &vbCr
  crt.Screen.WaitForString "OK"
:

```

FIGURE 17. Snippet of a VBA script to configure a cell modem.

via the Internet to the IFC ingest server is quite long, but once connected, data transfer is very reliable. For the other provider, the connection latency is much shorter, data transfer is fairly reliable, but obtaining responses from the IFC data ingest server is quite unreliable. A small percentage of the packets transmitted through both providers' networks are corrupted. By examining the corrupted packets it is clear that one provider bundles short messages from several other users and transmit a bundle as one block. For example, we have found advertisements for various products and Short Message Service (SMS) text messages urging support of a particular political candidate, embedded in our data packets.

Further, the performance of the network connection is variable. A BMRSS unit's modem may register on the service provider's network and transfer a data packet within 30 s at one time, and then a few hours later it may take more than a minute simply to register to the network, and this behavior may not be related to the signal strength. In some cases, the connection is "jerky". That is, data transmission and responses from the modem are in bursts of a few characters, with significant delay between bursts. A few such bursts may then be followed by continuous, smooth data communications. It is not possible for us to make definitive statements as to what cause this behavior. Regardless, to provide reliable communication between the BMRSS units in the field and the IFC data ingest servers, requires defensive programming with respect to the BMRSS firmware. That is, anticipate as best as one can the various things that can go wrong, and then provide well-defined behavior for such problems.

The following and Fig. 18 illustrate the point. When a GSM/GPRS modem has been properly provisioned (see above) it should automatically register first to the GSM and then to GPRS network when powered up. One approach could be to turn the modem on, wait some time and then start the data transmission. However, if the modem connection is erratic for some data transmission attempt, and the irregularity occurs during the registration process, the data transmission may start before registration is complete. This could cause problems at the server and could also cause serious problems at the BMRSS side—if the modem is not

```

int checkRegistration(int network)
{
  // Check if modem is registered on the networks.
  // "network" = GSM or GPRS.
  char c, str[10];
  int i;
  unsigned int start_cnt;

  // The following returns "+CREG: n,s" or
  // "+CGREG: n,s" where s is the status:
  // 0 = not registered and not searching
  // 1 = registered on home network
  // 2 = not registered but searching
  // 3 = registration denied
  // 4 = unknown
  // 5 = registered, roaming

  switch(network){
    case GSM:
      putsf1("AT+CREG?\r");
      if (waitfor("+CREG:",2) != 1) goto L10;
      break;

    case GPRS:
      putsf1("AT+CGREG?\r");
      if (waitfor("+CGREG:",2) != 1) goto L10;
      break;
  }
  if (waitfor(",",1) != 2) goto L10;
  start_cnt = timer_cnt;
  memset(str, '\0', 10);
  i = 0;

  while (i < 10) {
    if (rx_counter1 > 0) {
      c = getchar1();
      str[i++] = c;
      if (c == '\r') break;
    }
    if (diffcount(timer_cnt, start_cnt) > 1*TPS)
      goto L10; // 1-s timeout
  }
  if (c != '\r') goto L10;
  return (atoi(str));
L10:
  return(-1); // Timed-out.
}

```

FIGURE 18. This C function for checking registration to the cell phone networks illustrates the defensive programming we use in the BMRSS firmware.

properly registered, and it receives data, the behavior is not well-defined. The modem may interpret the data as nonsense AT modem configuration commands and generate many error messages that the SBC on the BMRSS unit will then see.

The checkRegistration C function in Fig. 18 illustrates how the BMRSS firmware handles registration. Another function (not shown here) will issue the AT command that will initiate the registration process. The checkRegistration function will then be called within a loop, checking several times for registration to the GSM network. To check, the firmware writes the string AT+CREG?\r to the serial port connected to the modem. The \r is the ASCII carriage return character (ASCII 0x0D). The modem should return the string +CREG: n,s\r where n should be 2 and s is a number that indicates the status of the connection. Note the call:

```
waitfor("+CREG:",2) ...
```

The waitfor function waits for 2 s for the first part of the required response, namely +CREG:. If it

does not see this string in 2 s, waitfor returns 0 and checkRegistration returns -1 to the caller. Otherwise, checkRegistration waits up to another 2 s for a comma. The way the code is written, it skips over the n in the +CREG: n, s\r response. Next, checkRegistration reads in the s in the +CREG: n, s\r response, and copies character to a variable str. There is a 1-s grace period between characters. Also, the code checks that the string is terminated with the \r character. The string str is then converted to an integer number using the atoi C function and returns that to the caller. The caller can examine the status code, which could be 0, 1, 2, 3, 4, 5, or -1 if it timed out. If the checkRegistration function returns 1 or 5, then the caller would call it again, but this time checking for GPRS registration. Assuming the modem is registered on both GSM and GPRS, the caller can initiate the data transfer. Otherwise, the caller can handle the failure to register in a graceful and defined manner.

```

POST /ingest.php HTTP/1.1
Authorization: RIVER_SENSOR
Host: s-iihr53.iihr.uiowa.edu
Pv: 1.2
Fv: 2.3A
User-Agent: A
Sid: SRS0999
Loc: N041 39.342998, W091 32.336597
Sn: $2866C86D010000D5*
H: 20.9,13.0,22,3.3, 0
Cb: 0,D,9999
Ph: 353530000064823

Ns: 1
Content-Length: 27

20110627_213006,15.67,22.7
    
```

FIGURE 19. A sample data packet, illustrating the general structure of the data packets.

D. DATA PACKET STRUCTURE

Fig. 19 shows a sample data packet, illustrating the general structure of the data packets. A data packet has two parts, namely, the header and the payload. The header contains metadata and system health information.

E. HEADER AND SYSTEM HEALTH

The first line uses the http POST method to invoke a PHP script on the server. The Authorization, User-Agent, Content-Length, and Host fields are standard http header fields. Even though we don't use the Host field at the server, it is mandatory under HTTP/1.1. We have added a number of other fields that we describe below. Following the header fields are a blank line and then the actual data (payload).

The Authorization field identifies the message to the server. The next two fields (Pv, Fv) are for tracking the version number of the hardware and the software of the device that sent the message. We use the User-Agent field to track if the packet is from manual field (M) testing, or normal

automatic data collection (A). Sid is an alphanumeric tag that identifies the BMRSS. Loc holds the GPS coordinates acquired from the embedded GPS module. Sn is the unique identifier for the BMRSS.

The H field provides system health statistics for the BMRSS such as its battery voltage, RSSI, and a reading from the wetness sensor inside the BMRSS. Cb refers to some calibration data, namely whether the modem is on the upstream or downstream side of the bridge and the compass heading giving the direction the BMRSS is facing. Ph refers to the phone number of the cell modem. Ns is a field used to inform the server how many records are being sent with the transmission.

F. PAYLOAD

Content-Length informs the server about the size of the payload. The payload contains the time and date of the measurement, the distance down to the water's surface and the temperature inside the enclosure. Normally, the payload will be a single line, corresponding to the current measurements. However, as we described above, if a sensor cannot successfully transmit, it will save the data for later transmission. In that case, Content-Length reflects the length of the multiple measurements in the payload. The ingest script updates the database accordingly. Fields in the payload are comma-delimited to ease parsing in the PHP ingest script on the server.

G. DATA INGEST

At the server, a PHP ingest script (ingest.php) performs two tasks, namely (a) saves a replicate of the data packets on disk in a hierarchical structure, and (b) populates a relational database with the packet information.

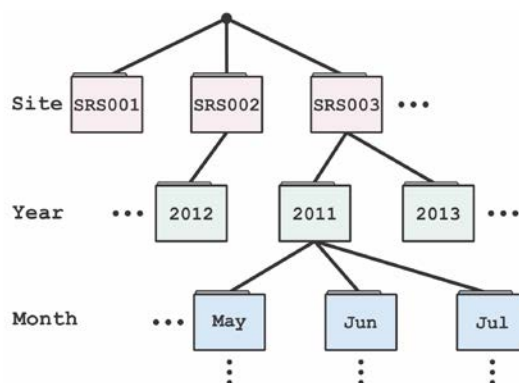


FIGURE 20. Directory tree structure for data backup at the ingest server.

First, consider the replicates on disk. Fig. 20 shows the structure. PHP provides a get_headers() function that returns an associative array populated with the headers (Host, Pv, ...) we describe above. The ingest script uses the Authorization key/value pair to validate the source of the packet. The ingest script uses the site id, year, month and date, and a number that increments for each file

and resets at midnight to generate a directory path and file name. The ingest script processes the set of key/value pairs and writes the packet information to disk. The resulting text file is essentially an exact replica of the original packet.

By design, the script performs no error checking when it creates the text files. By examining the original text files, we have discovered numerous bugs, features, and quirks that appear when transmitting data using the cellular technology. One example is unsolicited text messages (SMS) to our cell modems. This text would appear embedded in some payloads. The solution was to ask our carrier to disable SMS service to the account.

The second task the ingest script does it to populate the tables of a relational database. We use the Postgress database engine. A detailed description of the database is outside the scope of this manuscript. Finally, the ingest script echoes a text string OK, followed by an integer that is transmitted back to the BMRSS unit. The OK string confirms to the BMRSS unit that its information was received and successfully added to the database. The integer is a value that the sensor uses to set its data collection rate.

The sensors' firmware evolves as users request more features. For example, end users requested changes related to GPS coordinates. Another addition initiated by a user request is the Cb header field described above. Given that the sensors are deployed across the state of Iowa, it is a significant and time-consuming undertaking to upgrade the firmware. Thus, there may be several firmware versions in operation at any given time. The ingest script uses the P_V, and F_V header lines (see above) to differentiate between the packet- and firmware versions and respond accordingly.

VII. OPERATIONAL

A. TESTING

We test BMRSS units exhaustively before deployment. In addition to tests we perform during construction (i.e., power consumption of critical components, visual inspection, etc.) a BMRSS is tested as a unit. Once its enclosure is sealed, a BMRSS is considered complete and is not opened again unless it fails one of the environmental tests described below. However, BMRSS units have serial ports that allow one to connect them to a computer for configuration, if needed.

The BMRSS unit is placed in a tank for submersion testing. The tank is filled with water to the point where only the tip of the cell antenna is showing. This allows the system to still transmit to the ingest server. The system is set to transmit every 5 min during the test. The internal wetness sensor data field in the database turns red if water infiltrates the enclosure. The sensor can indicate light condensation to standing water in the box. The database administrator page is monitored closely for the indication of leaks. A BMRSS unit remains submerged in the tank for 72 hours. About 3.5% of the units fail the immersion test. We traced this back to micro-fractures in the enclosure or irregularities in the box lid.

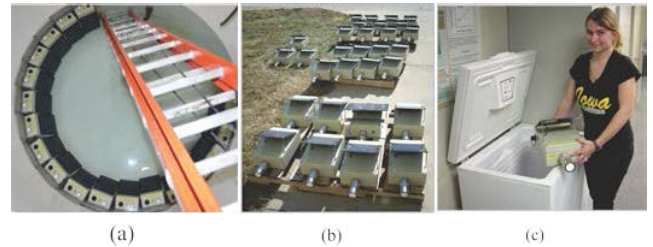


FIGURE 21. BMRSS units are thoroughly tested before deployment on bridges. Left: units are immersed in water (with small portion of cell modem antenna above water) for 72 hours to test for water leakage. Middle: solar power and associated charge controller test. Right: the units are placed in a freezer and cooled to -20°C .

Following the immersion test, the BMRSS units are placed outside to check the solar panel-based battery charging, and allow a BMRSS unit to go through diurnal temperature cycles. A large subset of BMRSS units undergo a freezer test to evaluate performance down to -20°C .

B. DEPLOYMENT

Placement is determined by requests from local communities. This requires interaction with local and state authorities (bridge owners) for permission to install the devices. Deployment is expensive, as it requires a minimum of two people from a safety standpoint. In some cases, deployment requires extended two-, three-day long trips.

VIII. RECEIVED SIGNAL STRENGTH INDICATOR (RSSI) DATA

Using the proper AT command (see Section VI) one can interrogate the cell modems to obtain a Received Signal Strength Indication or RSSI. On consumer cell phones, this will determine the number of “bars” on the phone display. Using information provided by the modem manufacturer, one can convert the RSSI number to signal strength in dBm. We use RSSI to refer to either the actual RSSI number or the corresponding value in dBm.

Good RSSI information is important for efficient management of the BMRSS network. If the signal strength is marginal at a particular site during fair weather, it may lose communication during a heavy precipitation event, since the signal strength drops during heavy downpours. Considering the purpose of the BMRSS network, namely flood forecasting, this is the worst time to lose communication. We have noticed a marked increase in the sensitivities of the modem receivers since the first units were deployed. Currently, for one modem manufacturer, the minimum signal strength required is -105 dBm, whereas for the other it is -115 dBm.

By analyzing the RSSI data, we infer that the values reported by the modems are quantized to 5-bits, and it is not clear to us how frequently the RSSI values are updated. For example, one may query a modem once per second, but it does not mean that it will actually make an RSSI measurement once per second. Rather, the modem may be

making measurements once per minute, and simply report previously-stored values. Still, the RSSI data provide useful and interesting information.

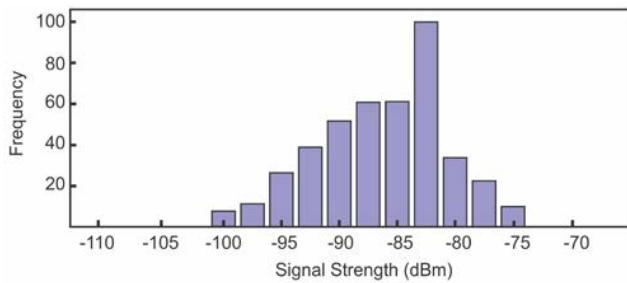


FIGURE 22. Typical distribution of received signal strength at a BMRSS site.

Fig. 22 shows a normalized histogram of the signal strength at a typical BMRSS site. This particular modem has a sensitivity of -105 dBm. On the time scales that the BMRSS network operates, RSSI is affected by water vapor in the atmosphere, rain and snow, adjustments made by the cell provider, and vegetation growth. In fact, in previous work we have used cell phone RSSI to infer diurnal patterns in corn [20].

Fig. 23 shows some plots of the RSSI information. The general appearance of Fig. 24 is representative of such plots. In general, the 15-minutes RSSI data is noisy. However, there are clear seasonal trends visible at most sites, and there is 15 dB or more difference between the midsummer RSSI and midwinter RSSI. These seasonal variations are almost certainly the result of variation of water content in the atmosphere.

The RSSI at some sites have erratic behavior. For example, in Fig. 23 (b) the RSSI undergoes step changes with shifts of 10 dB or more in a day. We do not have an adequate explanation for this behavior.

Because the 15-min data are so noisy, it is very difficult to discern diurnal cycles in the RSSI data in the time-domain for this site. However, there is indeed a diurnal cycle, as is evident from a power spectral density plot of the RSSI data in Fig. 23 (c). Whereas the time series in (b) appears quite different from that in (a), both their spectra show a clear peak between 10 and 15 μHz . This local maximum occurs at 11.6 μHz , which corresponds to 1 day. (There are $24 \times 60 \times 60 = 86,400$ seconds per day and $1/86400 = 11.56 \mu\text{Hz}$).

A. USER INTERFACE

The BMRSS firmware incorporates a command-line interface that allows a person to interact with the unit. This interface is accessed via the serial port as follows. First, connect a serial cable between a computer and the BMRSS unit. Next, on the computer, open a terminal emulator program such as HyperTerminal, PUTTY, or SecureCRT. Assuming the baud rate is correctly set, the user will see a text message every 15 seconds as the BMRSS unit wakes up and blinks the heartbeat LED. The unit provides a window during

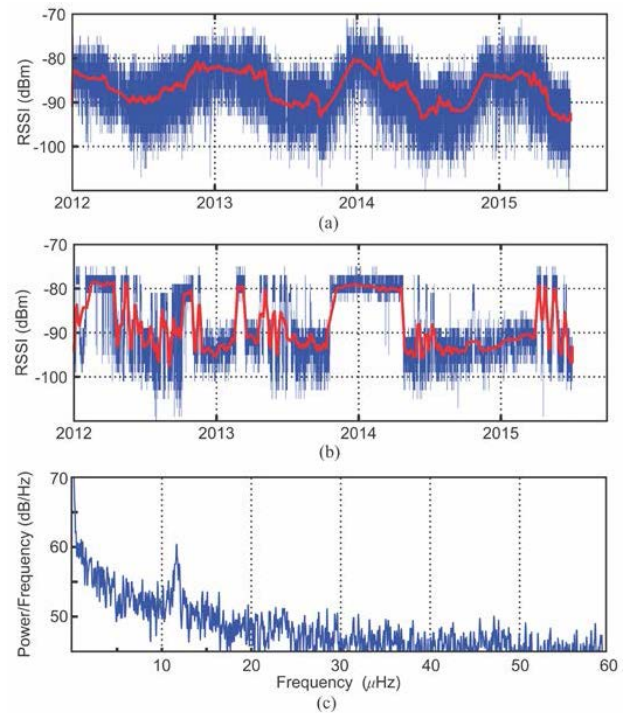


FIGURE 23. (a) The 15-min received wireless signal strength data for a representative site, shown in blue, show large variation. The red line is the daily average signal strength, which shows seasonal variation. (b) At some sites the received signal strength exhibit large step changes. (c) Power spectral density of data in (a).

```
? - This command
Data - Manipulates data:
      Data SHOW|CLEAR
Date - Display system date and time
Dist - Display Distance Measurement
Exit - Exit shell
Info - Shows serial number, Site-Id, etc.
Cell - Shows Cellular GSM GPRS Networks Registration
Modem ON - Turn modem on
Modem OFF - Turn modem off
GPS ON - Turn GPS on
GPS OFF - Turn GPS off
GPS GET - Get GPS output, attempt to get Lat/Lon & set time
Reboot - Reboot the system
Packet - Sends a test packet (modem must be ON)
Rssi - Measure and display RSSI (modem must be ON)
Site - Sets/displays Site-Id
Set U/D - Sets/display Up/Down Stream
Set ORIENT - Sets/display Orientation Degrees
Set DATUM - Sets/display Distance to Streambed
Set LAT - Sets/display LATITUDE e.g. N041 39.348278
Set LON - Sets/display LONGITUDE e.g. W091 32.336177
Reset count - Set reset counter to 0
```

FIGURE 24. BMRSS command-line interface after entering the help command, which is a question mark character.

which the user can enter the command-line interface or shell. This is done by typing an exclamation “!” at the terminal. When the unit sees the exclamation, it displays the list of possible commands that are shown in Fig. 24. A BMRSS unit enters the shell automatically after a power cycle

Fig. 25 shows the response when the user executes the Reboot command. The shell has an auto-logout feature that exits the shell and starts the automated data collection and transmission after 5 min of inactivity at the serial port. Thus, if a user forgets to start the data collection process, the shell will automatically remedy this.

```

RESET HARD
Stored Samples = 0
IFC Wireless Stage Datalogger. Copyright (c) 2010.
Pv: 1.2
Fv: 2.2E
User-Agent: A
SID: SRS0056
LOC: N041 39.328182,W091 32.291362
SN: 13A2004052A8F3
CB: 0.N.9999
Ph: 012002000373979
Resets: Hard=1 Soft=0 WD=0
Modem is OFF
Supply voltage = 12.72 V
Temp is = 26.88 C
Board Temp = 27.86 C
Wetness is = 3.30 V
Number of resets = 1
Type ! to enter shell

```

FIGURE 25. Sample BMRSS health information displayed after a reset.

B. VANDALISM AND RELATED

We lost only a few BMRSS units during five years of operation. We experienced only two instances of clear vandalism, which is remarkable. One reason for this is that Iowa has a low crime rate. Another reason is related to where the BMRSS are deployed—on the side of public bridges. Even though the units have heartbeat LEDs, the units are small and somewhat hidden from view. Second, in rural areas there are few pedestrians so that a vandal would have to park on or close to a bridge and then lean over the railing to get access to the unit, which may be out of arm's length reach. Further, an individual parked on a bridge and leaning over railings will draw attention from farmers and law enforcement.



FIGURE 26. Fig. 26. Photographs of a BMRSS unit damaged by vandals.

Fig. 26 shows a damaged BMRSS unit. We surmise that someone took a metal rod and punched out the GPS antenna, stuffed a rag into the orifice, and set it on fire. However, the unit kept functioning, but the wetness sensor, now exposed to the outside air, indicated a problem. The BMRSS network operators noticed the change in the wetness sensor data and a subsequent site visit showed that the unit was vandalized.

One BMRSS unit was accidentally damaged (see Fig. 27) by workers as they were doing stream bank restoration.

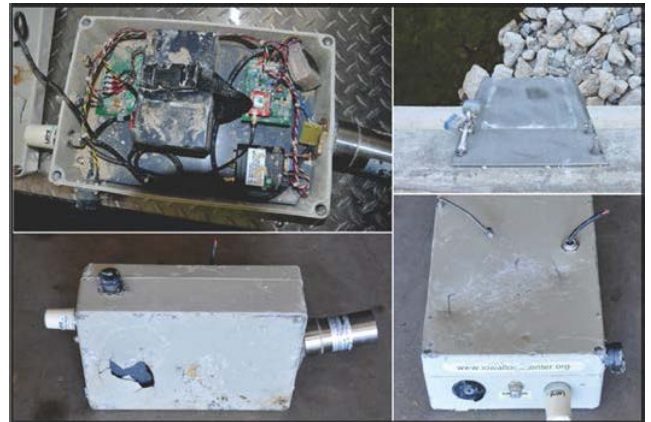


FIGURE 27. Unit accidentally damaged during stream bank restoration. Note in upper right how falling rock broke the unit from the mounting plate.

Large chunks of broken up concrete were scattered over the side of the bridge by a conveyor truck. As the chunks fell into the stream, one piece hit the unit, breaking the mount bolts and the unit fell into the stream. The sensor was crushed by other pieces of concrete that dented the lid and cracking the battery, which stopped the system. Another unit was damaged when runoff from a heavy rain event swept an upstream wooden bridge along and the bridge debris impacted the unit. The impact destroyed the solar panel, but the unit kept functioning. We replaced the solar panel and mounting bracket on-site.

IX. MEASUREMENT PERFORMANCE

A. TEMPERATURE EFFECTS

Examination of the time series data from deployed BMRSS units indicate small diurnal cycles in river stage. One potential source of such variation is diurnal base flow changes [7] and other processes. However, the variation is primarily related to changes in ambient air temperature. The speed of sound in air is a function of humidity and temperature with the latter the most important for the Senix ultrasonic distance measurement module used in the BMRSS units. The Senix module incorporates a temperature sensor. One can configure the Senix firmware as follows:

1. Perform no temperature compensation. Use a default and fixed value for the speed of sound.
2. Use the temperature reported by the Senix' internal temperature sensor and calculate the speed of sound. Use this value for distance measurement.
3. In addition (b), compensate for self-heating. If the Senix module is continuously powered, the module's electronics generate heat that can result in up to a 9 °C degree increase in the temperature that the embedded temperature sensor reports.

The actual temperature compensation algorithm employed in the Senix is not clear, but the following is a likely procedure. The module estimates the distance d by measuring the

time-of-flight (t_{TOF}) and then reports

$$d = \frac{1}{2}v \cdot t_{TOF}$$

where v is the speed of sound in air. The speed of sound in air is [21]

$$v = 331\sqrt{T_{air}/273} \approx 20\sqrt{T_{air}} \text{ m/s}$$

where T_{air} is the air temperature in Kelvin. Thus, if the Senix module uses the temperature T_S , obtained from its embedded temperature sensor, it will report a distance

$$d_S = 10\sqrt{T_S} t_{TOF}.$$

In the BMRSS application, the Senix-reported value, T_S , is generally different from the air temperature for the following reasons. The BMRSS unit is often exposed to direct sunlight, and mounted on a bridge with a large thermal capacity. Additionally, the electronics inside the enclosure generate some heat. The differences between the air temperature and what the sensors in the ultrasonic module and the BMRSS units report can be as high as 10 °C. This results in erroneous distance measurements. We see that when $T_S > T_{air}$, sensor will overestimate the distance, and if $T_S < T_{air}$ the sensor will underestimate the distance.

In order to quantify the effect of temperature on the distance measurements, we mounted a BMRSS unit on a concrete bridge/culvert structure across a dry creek. The structure has a solid concrete base and the creek was dry for the duration of the experiment. Using the Senix module we measured the distance every 5 min for 4 days, and we recorded the Senix sensor's temperature sensor values. We configured the Senix to perform temperature compensation, but without adjusting for self-heating (see above). We measured the air temperature independently using a thermometer 50 m away and shielded from the sun.

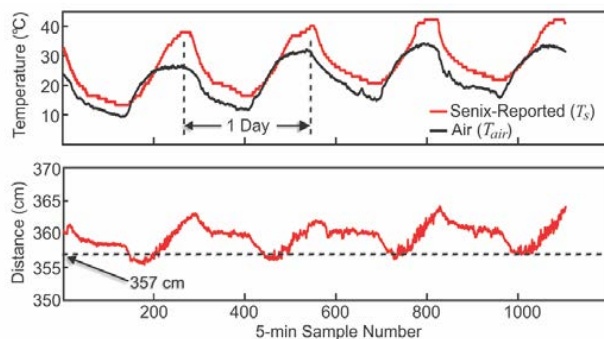


FIGURE 28. The top panel shows the Senix-reported (T_S) temperature and an independent air temperature measurement (T_{air}). The bottom panel shows the Senix-reported distance and the broken line indicates the true distance.

Fig. 28 (top panel) shows the Senix module temperature T_S and the air temperature T_{air} , and in the bottom panel is the reported distance to target d (the base of the culvert). The true distance is also indicated. For most of the time $T_S > T_{air}$

so that the Senix sensor overestimates the distance, which is evident from Fig. 28. This is consistent with the discussion above.

We explored the possibility of improving on the Senix' estimate of the distance. The first step is to calculate the t_{TOF} the Senix module used:

$$d_S = 10\sqrt{T_S} t_{TOF}$$

$$t_{TOF} = \frac{d_S}{10\sqrt{T_S}}$$

Then use T_{air} to calculate the speed of sound and then determine a better estimate, d' , of the distance:

$$d' = \frac{1}{2}v \cdot t_{TOF} = \frac{1}{2} \left(20\sqrt{T_{air}} \right) t_{TOF} = \frac{10\sqrt{T_{air}}}{1} \frac{d_S}{10\sqrt{T_S}}$$

$$= d_S \sqrt{T_{air}/T_S}$$

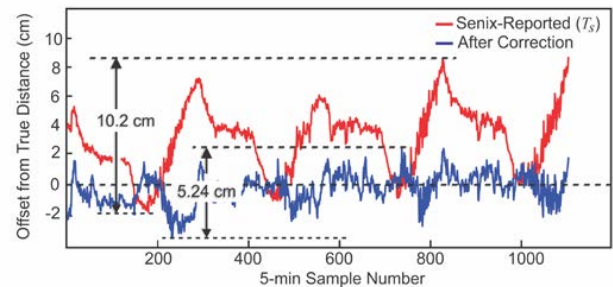


FIGURE 29. Plots of (Measured/Estimate Distance – True Distance) for the Senix-reported, and after correcting using an independent air temperature measurement.

Fig. 29 shows the effect of this correction by plotting the Senix measurement error ($d_S - d_{True}$), and the error after applying the temperature correction, namely ($d' - d_{True}$). Clearly, the correction improves the estimate, reducing the absolute variation from 10.2 cm to 5.24 cm, and also removes the bias.

In retrospect, the placement of the temperature sensor used to measure the air temperature was not ideal. We placed it in an open field adjacent to the stream and culvert about 50 m from the BMRSS unit. Whereas this temperature measurement provides a better estimate of the air temperature the ultrasonic pulse the Senix sensor generates travels through, the air temperature in the culvert is different (probably higher). We plan on exploring methods for performing better temperature compensation.

B. INTERNAL PRESSURE

BMRSS units are sealed and the internal temperature changes translate to significant changes in pressure inside the enclosure. The combined gas law provides a relationship between pressure, temperature and volume:

$$\frac{P_1 V_1}{T_1} = \frac{P_2 V_2}{T_2}$$

where P_1 and P_2 are pressures, V_1 and V_2 are volumes and T_1 and T_2 are temperatures in Kelvin. Here $V_1 = V_2$ so that

$$P_2 = P_1 \frac{T_2}{T_1}$$

Now assume a BMRSS enclosure is sealed at laboratory temperature and pressure, which we will take as 20 °C (\approx 293K) and 101.325 kPa (\approx 100 kPa) respectively. At 55 °C the internal pressure of the sealed BMRSS becomes

$$P_2 = P_1 \frac{55 + 273}{20 + 273} = \frac{328}{293} = 1.12P_1$$

This shows that the pressure inside a BMRSS enclosure can increase to 12% above the pressure at which the enclosure was sealed, nominally atmospheric pressure.

The dimensions of the cover and back of the enclosure 35.6 cm \times 24 cm so the areas of each these surfaces are 0.0854 m². The 12% pressure increase at 55 °C corresponds to \approx 12 kPa and the force pushing against the lid is

$$F = P \times A = 12 \times 0.0855 \approx 1.03 \text{ kN.}$$

This converts to about 230 pounds. A similar calculation for -40° shows that the pressure inside the enclosure can drop to about 80% of the pressure at which it was sealed. This translates to a force of 1.71 kN, which converts to 384 pounds. We confirmed the pressure calculations by measuring the internal pressure over a range of temperatures.

Clearly, there are large temperature-induced forces that stress the enclosure. From our experience with other deployments of environmental sensors and associated electronics in sealed enclosures, we know that repeated cycles of pressure changes can compromise seals and lead to micro fractures in enclosures. In fact, we have experienced instances with seemingly sturdy plastic enclosures, designed and marketed for environmental sensors, where the pressure changes have caused the bolts that fasten the enclosure lid, as well as the lid itself, to break. Consequently, the enclosure we use for the BMRSS units are high quality and have 0.25-inch (6.35 mm) thick walls. Additionally, units undergo stress tests before they are deployed. Apart from vandalism and floating debris impacting BMRSS units, we have had no failures of the enclosures.

X. DISCUSSION

A. PROCESSOR/SBC

BMRSS activities are controlled by a commercial processor board, namely the LP3500 from Digi International. The LP3500 has many resources and at first blush seems somewhat expensive, and some may consider using this board overkill. The task at hand appears simple enough: wake up, make a measurement by interrogating a commercial distance sensor, turn on a cell modem, transmit the data to a remote server, turn off the distance sensor and modem, go to sleep, and repeat. We have used the LP3500 SBC in other projects, so we considered it for the use in this project. The LP3500 SBC has 512K flash memory for program code, 512K SRAM

for program variables and data, 6 serial ports including one RS485, Real-Time Clock with battery backup, 10 8-bit timers and one 10-bit timer, digital outputs (8 can sink 200 mA), one 1-A relay, Watchdog/Power supervisor, and 8 ADC channels. In other projects, our departure point was an 8-bit AVR or PIC microcontroller and we considered going this route versus using the LP3500 SBC. One consideration was cost—an in-house AVR/PIC system with the required resources could cost less than 25% of the cost for an LP3500 SBC. We also considered other options such as commercial rather than in-house AVR/PIC boards or PC104-style SBCs.

After careful consideration, we decided to use the LP3500 for the following reasons. The “LP” in LP3500 refers to “Low Power”, and in its power-save mode can consume less than approximately 100 μ A. In practice it is very difficult to achieve this 100 μ A target. For example, with devices such as a GPS module or cell modem connected to one of the LP3500’s serial ports, currents can flow from the LP3500 to the device even when the LP3500 is in its deep power saving mode and the GPS/cell module is turned off. The explanation for this comes from considering how the transmit- and receive lines of the LP3500 and the connected devices idle—lines pulled high, lines pulled low, or lines in high-Z or tristate mode. These are other subtleties that result in the LP3500’s actual low power consumption closer to 1 mA and in sleep mode. This is 10 times the LP3500’s nominal 100 μ A minimum power consumption. Still, this is low and within our power budget.

Another consideration is the many serial RS232/RS485 ports available on the LP3500. This allows us to use RS232/RS485 rather than SPI or other such interfaces to interact with the cell modem, GPS module, and the Senix distance sensor. Additionally, a BMRSS unit has a serial port interface for users to interact with using a text-based menu. Finally, during development a serial port is invaluable for testing. The current BMRSS units use five of the LP3500’s six serial ports.

A very attractive feature of the LP3500 is that the vendor supplies a mature C compiler with extensive libraries. Programming the LP3500 is through the compiler GUI and thus straightforward. The company provides many code examples. Further, there is an active bulletin board-style user’s community.

With regard to cost, in reality, the SBC and indeed the electronics including the GPS module and cell modem, constitute but a small part of the overall cost. The enclosure and baseplate (including associated milling of holes) and mounting plate, bolts, battery, and solar panel are significantly more expensive items. Additionally, the personnel time for firmware development and testing are very expensive. Installations on bridges require road trips involving multiple people, which further add to the cost. Given this, the cost savings of an AVR/PIC board versus the LP3500 SBC is of little consequence for our application, but the resources that the LP3500 provides, significantly ease and reduce time for development, which ultimately is more cost-effective.

B. DATA TRANSMISSION

Most of the BMRSS hardware and firmware are quite stable and requires only occasionally small changes. The exceptions are the cell modems and associated issues such as data plans, different standards, different modem vendors, and so on. As the cell phone companies move to newer networks and standards (such as LTE), they want to sunset legacy networks (GSM) and especially data services (GPRS). Some of these changes will require that we install new modems in the BMRSS units. This is very undesirable since it would involve removing installed BMRSS units from bridges to perform the upgrade. Additionally, new modems will almost certainly require changes to the BMRSS firmware. Mergers and acquisitions of the companies complicate things further. It is often difficult to find qualified staff who can answer specific technical questions.

On the positive side, the monthly cost for data transmission for BMRSS units is currently about 20% from what it was five years ago. Cell phone coverage is good and improving. Further, cell modem vendors will now provision the modems on our behalf, which was not the case several years ago. The cell phone companies are also more flexible and have arrangements for pooling data allotments from several phones. This allows BMRSS network operators to increase data transmission frequency at some locations and offset that with reduced data transmission at other locations.

In hindsight, our decision to use the http protocol for data transfer was the correct choice. This allows us to use plaintext data packets that are easy to construct at the BMRSS, use server-side tools such as PHP, and browse incoming data packets with a web browser.

C. GPS

Since the BMRSS units are ultimately mounted on stationary bridges, location information is of secondary importance. The primary purpose of incorporating a GPS module into a BMRSS unit is for accurate time stamping of the data. Still, since the GPS location data are embedded in every data packet, and the raw packets are stored on disk as they arrive and then ingested into the IFC databases, location data are sometimes useful in resolving the true source of the data when unplanned events happen. Possible examples could be operator error when naming sensors when they are deployed (i.e., duplicating names), or software bugs when data are ingested into the relational data base tables at the servers.

D. SYSTEM HEALTH AND METADATA

The purpose of deploying a BMRSS is to obtain a river stage measurement. In practice, the river stage data are a small fraction of the overall data packet that a unit will transmit to an IFC server. The bulk of a data packet consists of system health (battery voltage, internal temperature), and metadata (GPS coordinates, type of distance sensor, calibration data). Our experiences with the BMRSS network and similar operations show that this additional information is vital for high-quality data and efficient maintenance of the network.

Even though the objective is completely unattended operation of the BMRSS network, this is not feasible in practice. Unforeseen events outside of our control interfere with the operation. Consider the following example. As part of periodic bridge maintenance, a road crew sprayed the side of a bridge (with a mounted BMRSS unit) with a translucent coating. This did not affect the distance measurement, but greatly reduced the efficiency of the solar panel and the average battery voltage started to decline. Without remedy, the unit would eventually die. However, a BMRSS unit embeds the battery voltage in every transmission. By monitoring the battery voltage at the ingest server (using automated software tools) we quickly identified a problem at this BMRSS location.

Another problem we have identified via system health data are riparian vegetation growth that affects RSSI or cast shadows on the solar panel and affect battery voltage, or both. Being able to track changes in battery voltage, RSSI, wetness sensor voltage, allows the BMRSS network operators to identify trends that may lead to failures. Site visits are expensive and time-consuming, especially when it requires substantial travel time. With solid system health data available, the network operators can consolidate maintenance trips and improve the efficiency of the overall network operation.

XI. SUMMARY AND FUTURE CHANGES

We have described the design, and operation of a device—a Bridge Mounted River Stage Sensor or *BMRSS*—that is mounted on bridges, and makes automated stream- and river stage measurements. A BMRSS unit incorporates a battery, solar panel and charge controller, GPS module, cell modem, and a single-board computer. In operation, a BMRSS wakes periodically, measures its distance from the water surface and transmits this information via the Internet to servers at the Iowa Flood Center.

We had two majors design objectives. First, a BMRSS should be inexpensive enough so that it is feasible to deploy many units, even on smaller rivers and streams. Such deployment will provide new and important information that will enhance the output from flood forecasting models. Second, a BMRSS must be robust and operate for several years unattended in what is a quite harsh environment, namely on bridges, and exposed to the elements.

We believe we have achieved our design objectives. The Iowa Flood Center has deployed and operated more than 220 BMRSS units across the state of Iowa, and some BMRSS units have operated continuously for more than five years. The units provide stage data with high temporal resolution on rivers and streams that were previously ungauged. We were able to achieve the objectives by bringing together a multidisciplinary team consisting of electrical-, mechanical-, and civil engineers.

Our approach is modular integration. That is, integrate modules such as a cell modem, GPS modem, and distance sensor using a glue board and interconnects, rather than integrating on the chip-level and producing a single board.

This modular integration is ideal for the volume/number of BMRSS units we expect to produce.

The Iowa Flood Center and its servers are housed at the University of Iowa. The network- and server administrators are diligent with respect to securing the Iowa Flood Center resources, but the data packets are currently in plaintext. Some administrators have expressed concerns regarding this and we are exploring alternative solutions.

REFERENCES

- [1] K. J. Beven, *Rainfall-Runoff Modeling: The Primer*, 2nd ed. Hoboken, NJ, USA: Wiley, 2012.
- [2] C. Mutel, Ed., *A Watershed Year-Anatomy of the Iowa Floods of 2008*. Iowa City, IA, USA: Univ. of Iowa Press, 2010.
- [3] J. A. Smith, M. L. Baeck, G. Villarini, D. Wright, and W. F. Krajewski, "Extreme flood response: The June 2008 flooding in Iowa," *J. Hydrometeorol.*, vol. 14, no. 6, pp. 1810–1825, Nov. 2013, doi:10.1175/JHM-D-12-0191.1.
- [4] R. C. Buchmiller and D. A. Eash, "Floods of May and June 2008 in Iowa," U.S. Dept. Interior U.S. Geol. Survey, Menlo Park, CA, USA, Open-File Rep. 2010-1096, 2010.
- [5] I. Demir and W. F. Krajewski, "Towards an integrated flood information system: Centralized data access, analysis, and visualization," *Environ. Model. Softw.*, vol. 50, no. 203, pp. 77–84, Dec. 2013, doi:10.1016/j.envsoft.08.009.
- [6] T. J. Buchanan and W. P. Somers, *Discharge Measurements at Gaging Stations*. Reston, VA, USA: U.S. Geological Survey Techniques of Water-Resources Investigations, ch. A8, 1996. [Online]. Available: <http://pubs.usgs.gov/twri/twri3a8/>
- [7] Z. Gribovszki, J. Szilágyi, and P. Kalicz, "Diurnal fluctuations in shallow groundwater levels and streamflow rates and their interpretation—A review," *J. Hydrol.*, vol. 385, nos. 1–4, pp. 371–383, May 2010, doi:10.1016/j.jhydrol.02.001.
- [8] D. Diemand, "Automotive batteries at low temperatures," USA Cold Regions Res. Eng. Lab., Hanover, NH, USA, Tech. Rep. 4–91, May 1991.
- [9] T. Reddy, *Linden's Handbook of Batteries*, 4th ed. New York, NY, USA: McGraw-Hill, 2010.
- [10] T. Efram, J. W. Kimball, P. T. Krein, P. L. Chapman, and P. Midya, "Dynamic maximum power point tracking of photovoltaic arrays using ripple correlation control," *IEEE Trans. Power Electron.*, vol. 21, no. 5, pp. 1282–1291, Sep. 2006.
- [11] H. F. Rezaei, P. R. Haugen, A. Kruger, and J. J. Niemeier, "Environmental sensor network data packaging and transmission via cellular networks," in *Proc. IEEE Int. Conf. Electro/Inf. Technol. (EIT)*, May 2011, pp. 1–4, doi:10.1109/EIT.5978580.
- [12] G. Niemiowski, *Serial Port and Microcontrollers: Principles, Circuits, and Source Codes*. Charleston, SC, USA: CreateSpace Independent Publishing Platform, 2013.
- [13] D. E. Comer, *Internetworking With TCP/IP*, 6th ed. Upper Saddle River, NJ, USA: Pearson, 2013.
- [14] D. Gourly, B. Totty, A. Aggarwal, and S. Reddy, *HTTP: The Definitive Guide*. Sebastopol, CA, USA: O'Reilly Media, 2002.
- [15] R. J. Bates, *GPRS: General Packet Radio Service*. New York, NY, USA: McGraw-Hill, 2001.
- [16] T. Halonen, J. Romero, and J. Melero, *GSM, GPRS and EDGE Performance: Evolution Towards 3G/UMTS*, 2nd ed. Hoboken, NJ, USA: Wiley, 2003.
- [17] S. C. Yang, *CDMA RF System Engineering*. Norwood, MA, USA: Artech House, 1998.
- [18] L. Harte, *Introduction to Code Division Multiple Access (CDMA): Network, Services, Technologies, and Operation*. Althos, 2004.
- [19] M. A. Banks, *The Modem Reference: The Complete Guide to PC Communications*, 4th ed. Medford Township, NJ, USA: Information Today, 2000.
- [20] K. P. Hunt, J. J. Niemeier, L. Cunha, and A. Kruger, "Using cellular network signal strength to monitor vegetation characteristics," *IEEE Geosci. Remote Sens. Lett.*, vol. 8, no. 2, pp. 346–349, Mar. 2011, doi:10.1109/LGRS.2073677.
- [21] OpenStax. (Jun. 21, 2012). *College Physics*, OpenStax. [Online]. Available: <http://cnx.org/content/col11406/latest>



ANTON KRUGER (M'86) received the Ph.D. degree from The University of Iowa in 1993. He is currently a Professor in electrical and computer engineering with The University of Iowa, where he is also a Research Engineer at IIHR-Hydroscience and Engineering, a unit of the College of Engineering at The University of Iowa.

His research interests include geo- and environmental sciences instrumentation development and embedded systems.



WITOLD F. KRAJEWSKI received the Ph.D. degree from the Warsaw University of Technology in 1980. He is currently the Rose & Joseph Summers Chair in water resources engineering and a Professor in civil and environmental engineering with The University of Iowa, where he is also a Research Engineer at IIHR-Hydroscience and Engineering, a unit of the College of Engineering at The University of Iowa.

He is the Director of the Iowa Flood Center, an entity funded by the State of Iowa in the United States, and housed at The University of Iowa. His research interests include rainfall estimation from weather radars and flood forecasting. He is a fellow of the American Meteorological Society and the American Geophysical Union.



JAMES J. NIEMEIER received the Ph.D. degree in electrical and computer engineering from The University of Iowa in 2010. He is currently a Research Scientist with IIHR-Hydroscience and Engineering, The University of Iowa.

He designs and develops instrumentation for the measurement of water quantity and quality variables in the natural environment.



DANIEL L. CEYNAR received the B.S. degree in mechanical engineering from The University of Iowa in 2002. He is currently a Project Engineer with IIHR-Hydroscience and Engineering, The University of Iowa.

His experience includes designing, deploying, and maintaining environmental monitoring networks including a mobile network of X-band polarimetric radars.



RADOSLAW GOSKA received the M.S. degree in mechanical engineering from the Warsaw University of Technology in 2002. He is currently a Design Engineer with IIHR-Hydroscience and Engineering, The University of Iowa.

He specializes in database and web applications and support data acquisition and modeling needs of the Iowa Flood Center, housed at The University of Iowa.

...