CARNIVORE: A Disruption-Tolerant System for Studying Wildlife

MATTHEW RUTISHAUSER, VLADISLAV PETKOV, JAY BOICE, KATIA OBRACZKA, PATRICK MANTEY, TERRIE M. WILLIAMS and CHRISTOPHER C. WILMERS

University of California Santa Cruz

Abstract—This paper presents CARNIVORE, a system for in-situ, yet unobtrusive monitoring of cryptic, difficult-tocatch/observe wildlife in their natural habitat. CARNIVORE consists of a network of mobile and static nodes that have sensing, processing, storage, and wireless communication capabilities. CARNIVORE's compact, low-power, mobile animalborne nodes are tasked with data collection (both sensing and communication), while static nodes' main task is to get collected data from the CARNIVORE network to the Internet. One of CARNIVORE's notable novel features is its robustness to intermittent node connectivity since, depending on the wildlife being studied, the network can be quite sparse and therefore disconnected frequently for arbitrarily long periods of time. To be able to support "disconnected operation", CARNIVORE uses an "opportunistic routing" approach taking advantage of every encounter between nodes (mobile-to-mobile and mobile-to-static) to propagate data.

In its current version, CARNIVORE's mobile node is outfitted in a collar and weighs 450 g. It is based on the MSP430 low-power processor, uses an IEEE 802.15.4 based transceiver which favors higher data rate and lower power consumption over range, and has 2GB of flash data storage. Field tests show that the "collar" can transmit data at 60 kbps within a range of 50-200 meters. It is equipped with a Li D-cell battery which provides the collar with a lifespan of 50-100 days depending on the data sampling rate and total number of transmissions. A typical deployment would collect and transmit 1GB of data compared to 450kB of transmitted data from comparable commercially available wildlife collars.

Regarding its sensing capabilities, the collar is currently equipped with a 3-axis accelerometer and GPS whose data can be used to infer animal behavior and energy consumption. Testing in both laboratory and free-range settings with domestic dogs show that a relatively simple algorithm can identify galloping and trotting. Recently collected data from first deployments on mountain lions (Puma concolor) near Santa Cruz, CA, USA, show that the system is a viable and useful tool for wildlife research.

I. INTRODUCTION

Known broadly as biotelemetry, remotely monitoring organisms has proved to be a powerful tool in understanding their physiology, behavior, and ecology [1]. Biologists have long recognized the need to study free-ranging animals in their natural environment. However, many species are cryptic and wide-ranging, and thus difficult to monitor directly or capture for repetitive physiological measures. To overcome these challenges biologists have long used VHF radio tracking [2] and archival data loggers on free-ranging animals [3]. New technologies have improved the effectiveness, efficiency, and ubiquity of biotelemetry. Increases in energy density of batteries and greater system miniaturization has allowed placement of VHF transmitters on the smallest mammals and large insects [4]. Researchers have also used the ARGOS satellite system for sensor data transmission, including highly accurate global positioning system (GPS) locations. In addition, VHF or UHF radio-modems are used to download data directly by the researcher. Unfortunately, ARGOS has very low data rate capabilities over a simplex data channel (1.5-7.2 kbits day⁻¹) [5]; radio-modems have yet to be automated, requiring the researcher to manually download data, and while their range is large (around 10 km) their data rates are low (around 9.6 kbps).

Advances in wireless communications, VLSI, and Micro-Electro-Mechanical Systems (MEMS) have enabled networks of low-cost, small form factor sensing devices which will bridge an important gap in the current biotelemetry state-ofthe-art. Due to their ability to sense, process, and communicate sensed data, sensor networks make sensed data readily available to scientists (and the community at large), in real-time (or quasi real-time) at low cost and with the required spatial and temporal resolution.

In this paper, we present the Carnivore Adaptive Research Network in Varied Outdoor Remote Environments (CAR-NIVORE), a sensor network system that specifically targets wildlife monitoring ¹. CARNIVORE was born out of an urgent need to gain deeper understanding of the interplay between predators, their ecosystem, and encroaching human populations. It is largely motivated by the ever increasing expansion of urban development into wildlife habitats and illustrated by an increasing number of interactions between wildlife and humans [7]. Predators also can exert heavy pressure on their prey species, sometimes reshaping their own ecosystem [8],[9]. The extent of pressure a predator puts on prey is directly linked to its energetic requirements for survival and reproduction. A firm understanding of their physiology and energy budget calls for high resolution behavioral and physiological data. This data can be difficult to collect for predators that are hard to capture and time consuming to monitor directly. Also, relatively rare but important events such as mating or consuming prey may be missed when animals are unobserved.

CARNIVORE's design was customized to fulfill the unique

This work has been partially supported by NSF grants 0963022, 0729707 and 0713994.

¹An earlier more condensed version of this work can be found in [6].

requirements imposed by wildlife monitoring applications including: energy efficiency, ability to operate with episodic connectivity, and reliability by being able to store data locally (when connectivity to the sink is unavailable). The resulting CARNIVORE monitoring network architecture consists of both mobile sensing– and fixed relaying nodes which provide sensed data to biologists wirelessly, eliminating the need to recapture the predators. The net effect is considerable reduction of the delay between data collection and data delivery, and increased effectiveness of data collection.

The CARNIVORE mobile, animal-borne, sensing nodes, or CSNs, are limited in weight, yet contain the required sensors (3-axis accelerometer and GPS), processing, storage, and communications capability. Each CSN must be capable of providing data that will allow biologists to monitor the physiology and behavior of the target species. Of particular interest is their hunting habits and energetic costs. In order to accurately track the animal's energy budget, its behavior can be categorized into activities such as walking, running, sleeping, hunting, feeding, and so on. Furthermore, the footfall frequency in any gait is obtained and can be used to calculate the expended energy. Acceleration data along three axes will be used to extrapolate behavior data such as activity and footfall pattern [10],[11]. After local, as well as centralized processing at the information sink(s), raw data will be turned into behavior and energetics data. Coupled with GPS position fixes and time stamps, we will put this data in perspective against other factors in the ecosystem such as human populations, habitat types, and other animals of the same- or different species.

Weight and power constraints have the biggest effect on design choices. With batteries as the single heaviest component, power is one of the system's most limited resources. Thus communication, processing, sensing, and data storage must all be optimized to minimize energy consumption and extend the operating life of each node. Furthermore, CSNs' storage capability should be carefully provisioned so that the system can withstand operation under episodic connectivity and still meet the specified data reliability requirements.

Coyotes (*Canis latrans*) were chosen as the first target species for developing the CARNIVORE network; however, the system is flexible enough to be used on a variety of species. The system is currently deployed on mountain lions (*Puma concolor*) in the Santa Cruz Mountains for first field testing. Here we present early results from data collected on mountain lions. We also present results of further testing and analysis of the accelerometer data, GPS, firmware, network protocol, and power consumption. We will outline the entire system, focusing on the more important components. The first fullyfunctional version of the CSN was developed by Petkov [12]. This first version of the collar allowed for substantial testing of the system, especially with respect to the accelerometer and real-time system (RTS) firmware.

The design of the CARNIVORE network allows for opportunistic data flow between CSNs and from CSNs to SRNs (Figure 1). CARNIVORE static relay nodes (SRNs) communicate with CSNs in range and also with other SRNs providing wider-range network connectivity and conveying



Fig. 1. Overview of the CARNIVORE network. A predator, such as a coyote, wears a collar containing a CSN while fixed basestations or SRNs act as data sinks. CSN-to-SRN wireless range is greater than coyote-to-coyote wireless range because the base stations employ high-gain collinear antennas. A SRN has been developed for capturing data from CSNs; however, the final SRN has yet to be implemented to deliver data via the internet.

sensed data to the information sink(s). Although the bridge between the lower and upper tiers of the network have yet to be implemented, we anticipate unlimited power supplies and long-range communication links for these nodes. Wireless links between CSNs and CSNs-to-SRNs utilize the 802.15.4 MAC layer and a CARNIVORE specific network protocol. The upper-tier links between SRNs have yet to be determined; however, 802.11, 900 MHz long-range links, or long-range ZigBee/802.15.4 are all possible choices.

II. HARDWARE

The CARNIVORE CSNs were designed from the ground up with the goal of maximizing battery life while meeting the application goals. Dictated by the CARNIVORE application requirements, the hardware specification for sensing and data storage of the CSN could not be met by existing solutions such as the Berkeley Mote platform [13],[14],[15]. Specifically, this platform was very early in its design when we began CARNIVORE and could not meet our requirements with respect to storage and low-power wireless. The components for the CARNIVORE platform were chosen to meet the sensor and long-lifespan requirements proposed by the biologists involved with the project (Figure 2). Components were chosen with low-power operation in mind to maximize collar lifespan and minimize weight through smaller batteries. The GPS provides location and velocity data while the accelerometer can provide data to monitor activity and behavior of the target animal. The MSP430 [16] provides very good performance with respect to code memory, peripheral modules, and low power operation. Individual modules can be turned off when not in use to



Fig. 2. Top level block-diagram of the CARNIVORE hardware. Black arrows indicate power connections. Thick, shaded arrows indicate control and data connections.

minimize power consumption. The Lassen iQ GPS receiver and MMA7260Q accelerometer also have good performance from both a sensor data perspective and power consumption.

The deployed CSN (Figure 3) also included off-the-shelf components used to guarantee tracking and recovery of the CSNs in the event of a total system failure for first field deployments. The timed drop-off was made by SirTrack [17] and causes the collar to fall off the animal at a specified date and time. The VHF beacon was produced by Telemetry Solutions [18] and was used to locate collars at long range (0.1 - 20 km). Both devices had separate power supplies and were fully independent of the CARNIVORE system.

A. Transceiver

A major change in the current version of the system was the removal of the ZigBee transceiver and protocol stack in favor of a CARNIVORE specific protocol. An early version of the CARNIVORE node [12] used the ETRX1 transceiver module with ZigBee protocol stack [19]. The ETRX1 utilized an Atmel Atmega 128 to implement the stack. The interface was unwieldy and the second microcontroller increased power consumption. By implementing a custom CARNIVORE network protocol and the 802.15.4 MAC layer on the MSP430, power consumption was reduced, the footprint of the radio reduced, and data transfer rate was increased by reducing the network overhead.

The CC2420 and associated balun circuit were taken from an ember application note for a ZigBee communication module [20]. This design allowed for single-ended operation and a 50 Ω impedance which allows for several different antennas. Schematic and layout specification from the application note were followed precisely.

A folded-F printed circuit board (PCB) antenna was used to minimize cost of the design [21]. Performance is comparable to surface-mount, chip antennas. If the PCB size must be



Fig. 3. Deployed CARNIVORE node. This collar was deployed on a mountain lion. The CARNIVORE electronics are above, Dcell battery and VHF beacon are lower-right, and a timed dropoff (SirTrack [17]) is lower-left. VHF antenna can be seen exiting the collar upper-right. The VHF beacon and drop-off use separate power supplies from the CARNIVORE platform. Components were assembled by Telemetry Solutions.

reduced for future designs, a chip-mount antenna can be used and easily incorporated.

B. Power supply

By using 3.6 V Li batteries with a very flat voltage profile, no power regulation is required as all components are compatible with this voltage. Lithium batteries at 3.6 V are available in D, C, AA and other sizes and so this design will be able to accommodate a variety of form factors and sizes of batteries for small and large animals. This allowed for a design without voltage regulators, reducing power consumption because regulators have efficiencies less than 100%. Dual MOSFETs were used to control power to individual components, allowing them to be turned off individually when not in use.

III. FIRMWARE

The firmware scheduler and framework [12] allowed for relatively easy modifications to the firmware even though these modifications were substantial during design iterations (Figure 4). Tasks are arranged in an array of function pointers,



Fig. 4. Top level block-diagram of the CARNIVORE firmware.

where each task is assigned a single element in the array. Tasks are started when an interrupts adds a task into the scheduler by inserting a function pointer into the high- or lowpriority task arrays. For example, when a frame is received, an interrupt is raised which inserts a function pointer into the high-priority array to begin the state machine which processes frames. Each state in the state machine is a function where the function pointer for the next state is inserted into the array. When the task is done, a null-pointer is inserted into the array so the task is no longer continued. Tasks in the low- and high-priority arrays are processed in a round-robin scheme. In each pass through the main loop, one function for each high priority task is called while only one low-priority function is called.The network and MAC sub systems will be discussed in Chapter IV.

In the early design stages, we made a difficult decision between completely custom firmware and TinyOS [22]. A flexible embedded operating system such as TinyOS provides a modular interface between software and hardware and takes on the burden of managing system resources and scheduling execution – all desirable attributes.

A flexible OS comes with a price however. For example, cpu cycles and memory need to be allocated to inter-process messages and operating system state variables. Each OS function must come at the expense of complexity (and thus increased power consumption). With the CARNIVORE CSNs, simplicity was chosen over flexibility to allow minimal power use and meet a design goal of 100Hz accelerometer sampling. The system functions entirely around interrupt-based cues allowing it to meet its real-time requirements. A simple scheduler exists for those tasks that are too large to put inside an interrupt service routine and adds almost no overhead to the system.

A. Task timers

Functions can be called at specified times in the future using the timer subsystem. This was needed for network and MAC protocols. For example the 802.15.4 MAC protocol uses a random, exponential back-off scheme. Thus, when the channel is busy, the MAC layer must attempt to send the frame at the specified time in the future. Three separate task timers were implemented: a fine scale task timer for the MAC layer, a task timer for the pseudo-random interval between neighbor discovery beacons, and a shared task timer for the election and file transfer timeouts. The last task timer could be shared because these do not occur at the same time.

To initialize a task timer, a function pointer is pointed at the function to be called, the required counter value is stored in a capture-compare register, and the interrupt is enabled. When the system clock advances the counter to the required value, the function pointer is dereferenced and the specified function is called.

B. Data storage

During initial debugging of the firmware, a FAT file system on the SD card was valuable for testing sensor data acquisition. However, troubleshooting file system errors became difficult to debug and the FAT file system was replaced by a system of FIFO queues on each CSN. Four queues are available so each data type (accelerometer and GPS) and data source (local or exotic) can be prioritized for forwarding through the network.

New data collected at the node or received from other nodes are enqueued at the tail of the appropriate queue. To allow for the multi-copy forward routing (Section IV-E), data sent to other CSNs can be dequeued from the middle of a queue. Only when data is sent to base station is data dequeued from the head pointer. If the head pointer catches up to the middle pointer, the middle pointer is moved along with head pointer. This allows for multiple copies to be forwarded through other CSNs to the base station while the originator of the data maintains a local copy for eventual download to the base station.

FIFO queues allowed for 512 byte data blocks to be the data segment routed through the network rather than entire files. The structure of these segments has a 3 byte header and a 509 byte payload (Figure 5). In the current implementation of the SRN the FAT file system remains in use. Data is saved in files which allow the microSD card to be easily accessed on a computer for parsing and analysis.

C. Receive buffer

We implemented a high-priority task that processes frames and a receive buffer for incoming bytes. Incoming frames raise an interrupt which buffers the bytes and begins the task of processing frames.

D. Accelerometer firmware

Timing information in the header for the accelerometer data allows for 1/1000th second accuracy for each accelerometer sample. Each accelerometer data segment contains 12 bytes of timing information and 110 3-axis accelerometer samples. The timing information in the header refers to the first accelerometer sample in the data segment. The 12-bit accelerometer data is packed in half-bytes to fully utilize the memory space



Fig. 5. Collar data segment. These segments are stored in the FIFO buffers of the microSD card. Numbers in parentheses are bytes.

and data payload. At a user defined interval, an interrupt triggers the capture of an accelerometer sample. Sampling rates of over 100Hz were achieved while still meeting all timing requirements. Higher sampling rates translate into higher energy expenditure not only because the accelerometer is active for a larger percentage of time but also because more data is generated and needs to be written to the SD card and eventually transmitted. The sampling frequency is a tunable parameter that can be set depending on the species being monitored, the capability of the system, and the requirements of the monitoring application (e.g., the fidelity needed by the scientist). We chose 60Hz because it is large enough to capture the frequencies of walking, trotting, and galloping of our target species without aliasing effects.

E. GPS firmware

The GPS firmware allows for network timing by updating the nodes system time and maintaing an accurate real-time clock for sensor sampling. In addition, the GPS firmware ensures that the almanac is always current. The Lassen iQ [23] on a cold start must download the satellite almanac, which describes current satellite locations. This requires 15 minutes of continuous signal from one satellite. Also, the almanac will expire after 8-10 weeks and require a new download. If the status packet from the GPS module indicates that the almanac is needed, the GPS timeout is increased to 18 minutes to allow for the download. Time, latitude, longitude, altitude, and velocity are recorded for each location and take up 30 bytes of space. 16 such locations can fit into one collar data segment (Figure 5). The firmware is capable of logging one location per second, but the energy cost of doing this is prohibitive. Commercial tracking collars available from Telemetry Solutions typically log anywhere from 1 to 48 locations per day. Depending on the species being tracked, biologists may be able to settle for lower temporal resolution on the location data. We chose to do 72 locations per day,

 TABLE I

 CARNIVORE FRAME TYPES AND SIZE (INCLUDING 802.15.4 HEADER AND FOOTER).

Frame type	Size (Bytes)
Neighbor Discovery Beacon	14
Election Nomination	12
Election Accept	12
File Transfer Request Data	12
File Transfer Data	98
File Transfer End-Of-Data	12



Fig. 6. Network stack and associated data structure. The CARNI-VORE stack uses a neighbor table to mediate use of the wireless channel. The list is populated during neighbor discovery and updated by various layers. Received frames are processed in the MAC layer which then updates the neighbor list.

giving us slightly higher temporal resolution.

IV. NETWORK PROTOCOL

The CARNIVORE network can be considered to be a highly-disconnected network or a usually-disconnected network because predators wearing the CSNs are typically not within wireless range of each other. Timely or complete recovery of the data at a base station is not required; however, as much data as possible should be captured.

There are three tasks which set-up the inter-CSN or CSNto-SRN connections: neighbor discovery, election, and data transfer (Figure 6). Each of these utilizes the MAC layer to send and receive data. A single MAC layer task, parses frames rapidly and updates the state variables for each task. A neighbor table is maintained at each CSN that stores the neighbor ID and a ranking metric. The complete CARNIVORE protocol requires six different CARNIVORE packet types (Table I). When two or more nodes come together, they form a star shaped network where the central node is chosen to receive the data from all the other nodes (Figure 7). If present, a SRN is always chosen to receive data. The chosen receiver mediates the round-robin scheme and minimizes competition for the channel, giving each node a request for data in turn (See Sections IV-D and IV-E).



Fig. 7. Round robin star network. The receiver mediates the roundrobin data transfer, accepting data from the senders. Either sender or receiver can end the transfer.

A. Disruption-Tolerant Routing

The low density of collared coyotes, the speed at which they can travel, and home ranges of 10-300 km² necessitates a disruption-tolerant data routing approach. In contrast to traditional routing protocols in which connectivity between any two nodes is generally assumed, a disruption-tolerant routing protocol must employ the long-term storage capabilities of each node to cooperatively route messages toward their destination (in this case, the SRNs).

An early approach to routing in such networks, Epidemic Routing [24], functions by replicating all messages to all nodes in the hope that one or more of the copies will reach the destination. More recent projects such as ZebraNet [25] and DieselNet [26] have explored routing between zebras and city buses respectively. Research on Data MULEs [27] explores topologies in which sensors are static devices, and a mobile node (a MULE) provides connectivity to a destination node.

CARNIVORES presents a unique networking challenge due to some of the characteristics of the collars, in particular, the large amount of storage space available in comparison to their limited bandwidth. Each CSN produces data at a rate of 2.1 kbps, and can store 2GB (approximately 88 days worth) of data. However, since it can be transmitted at a maximum rate of 63 kbps with relatively large power use compared to base-line power use, care must be taken to use the available bandwidth efficiently.

Using the Qualnet [28] network simulator, we studied different data routing/forwarding algorithms. Results from this study (shown in Section V-C) comparing epidemic, controlled epidemic, single-copy forwarding and multi-copy forwarding shows that the latter delivers the best performance in terms of delivery ratio and bandwidth usage. In our multi-copy forwarding implementation CSNs send messages to those CSNs with a more recent time-stamp from a sink. The source

coyote (the one who produces the data) keeps the messages, and will resend them again, though only directly to a base station. These messages are also marked in the buffer to be deleted first.

B. MAC layer

The current version of the CARNIVORE CSN utilizes a custom network protocol stack and implements the 802.15.4 MAC layer [29]. The MAC layer uses CSMA/CA (Carrier Sense Multiple Access with Collision Avoidance). A node wishing to transmit listens to the channel. If the channel is clear, the node transmits. If the channel is busy, the node waits a random time and listens again. Each time the channel is busy, the node waits an exponentially increasing and random amount of time up to the maximum number of back-offs.

C. Neighbor Discovery

The first step in the network protocol is to wake-up synchronously, announce yourself, and find your neighbors(Figure 8). The GPS time signal keeps all nodes synchronized. Each node sends out non-acknowledged beacons to the broadcast address with their node ID and a metric to be used in the election process. The beacons are not acknowledged to prevent an ACK swarm. Fifteen beacons are sent out at pseudo-random intervals to minimize collisions and guarantee a large amount of overlap when nodes are sending beacons. Each received beacon updates the neighbor list. If neighbors are found, this task puts itself to sleep and begins the election task.

D. Election of Data Recipient

In order to determine which node should receive data, a metric which correlates to likelihood of reaching a SRN is used. This type of routing is known as directed diffusion broadcast routing, where packets do not have a destination address and are simply forwarded along a direction or gradient most likely to result in delivery [30]. In the CARNIVORE network, the gradient is controlled by a saturating increasing counter that is reset to 1 whenever a node encounters an SRN. The node with the lowest metric has most recently visited a base station. And since nodes are on predators which likely have stereotypic behavior, this node should be the most likely to encounter a base station again.

Nodes choose the neighbor with the lowest metric to receive data (Figure 9). If their own metric is the lowest, they wait for a nomination. If they do not have the lowest metric, they send a nomination packet with an acknowledge request to the node with the lowest metric. The nominee must send a nomination acceptance packet back for a link to be established. In this way, a hidden node will not disrupt the formation of a network (Figure 9). An ignored nomination will cause the nominating node to time out. It will not attempt to initiate another link until the next network wake-up. Also, a node waiting for nominations but receiving none will also time out and must wait until the next network wake-up. A nominee becomes the receiver in the data transfer task. All nodes that



Fig. 8. Neighbor discovery protocol. This task initiates wireless communication at a specified interval, synchronized by the GPS time signal. A specified number of beacons are sent at pseudo-random intervals to prevent contention for the channel. The beacons do not request acknowledgments.

sent nominations and received acceptances become senders in the data transfer task. SRNs always have a metric of 0 and will therefore always win an election and act as receivers.

E. Data Transfer

Simulation of data forwarding in the CARNIVORE network (more extensively discussed in Section V-C) showed that a multi-copy-forward scheme performed the best with respect to delivery success and minimizing total transmissions but at the cost of buffer space. Since we are using 2 GB microSD cards, buffer space is not a problem and this strategy was chosen. In multi-copy-forward, a copy of data is stored locally on the generating node and a single copy is forwarded through the network. This part of the CARNIVORE network protocol, as well as data prioritization, is accomplished when a sending node chooses which data to send.

The receiver first checks if it has room for any more data. If yes, the receiver sends a data request and starts a short time-out. Upon receipt of the data, the receiver moves onto the next node and requests data. If a time-out occurs or an



Fig. 9. Hidden-station example. Radio range is shown by the shaded circles. Node B can communicate with A and C while A and C cannot communicate. After neighbor discovery the neighbor tables are filled as shown. A nominates B, B nominates C, and C nominates itself in the election. B ignores A's nomination and A times-out. C accepts B's nomination. B then sends data to C and A does not transmit any data during this network wake-up.

end-of-data packet is received, that neighbor is removed from the neighbor list. The receiver limits each node to sending a maximum number of data segments such that the round robin will end before the next network wake-up. The receiver terminates a link with a node by not sending a data request and letting that node time out.

The sender during data transfer sets a long time-out and waits for a data request from the receiver. This long timeout allows for one complete round-robin with the maximum number of nodes in the round-robin. Once a data request is received, the node picks a data type to send. If no data of any kind is available, the sender sends an end-of-data packet to terminate the transfer. If the node has data to send, it fragments the 512 byte data segment into 6 packets to accommodate the 128 byte maximum data size specified by the 802.15.4 standard. These packets are then sent with the *ACK request* bit set in their 802.15.4 frames, causing the receiver to send an acknowledgement automatically upon proper reception. If a transmission fails, the FIFO queue is restored and the transfer is ended.

V. EXPERIMENTS AND RESULTS

A. Power

Current consumption was measured for hardware components using a 1Ω current sense resistor and a Tektronix TDS3054C oscilloscope. Temporary changes were made to the firmware to enable or disable various components of

 TABLE II

 Deployed firmware settings and battery power.

Component	Value
1 Li D-Cell	19 Ah
GPS sampling interval	20 min
Accelerometer sampling rate	60 Hz
Network wake-up interval	5 min
Estimated lifespan	100 days

TABLE III Measured power consumption and percent time active per component for the deployed system.

Component	Current drain (at 3.6V)	Percent time active
Transceiver GPS GPS SD card access μ C and accelerometer Accelerometer SD card access	23.0 mA 41.4 mA 22.5 mA 5.8 mA 25.0 mA	1.0% 3.8% <0.1% 100.0% 1.4%
AVERAGE	7.9 mA	

the system. Voltage across the resistor was measured and converted to current using Ohm's law (Tables III and V). In addition, the amount of time each module was active was measured with the oscilloscope or calculated from firmware settings. These values could then be used to calculate the expected lifetime of a CSN given a battery with a specified Ah rating using equation 1 (Table II):

$$L = \frac{A}{24 * \sum_{C}^{i=1} c(i) * p(i)}$$
(1)

where L is the CSN lifetime in days, C is the number of components, c(i) is a components current consumption in mAh, p(i) is a components percent time consuming current, and A is the mAh rating of the battery.

To confirm this method of estimating lifespan, we performed an accelerated power test. We modified the settings of a CSN and used 2 AA Li 1.5 V batteries to power the CSN (Table IV). This produced a much greater total power consumption (Table V) and allowed us to drain the batteries in a relatively short time period, confirming our estimation method. We predicted the CSN would last 3.3 days. From the GPS and accelerometer data logged by the CSN, We found the actual lifespan was 3.4 days.

B. Wireless Radio Link

We performed a variety of range tests in an open field with waist to head-high vegetation. By using specialized firmware,

TABLE IV Accelerated power test firmware settings and battery.

Component	Value
2 Li AA-cell	3 Ah
GPS sampling interval	60 s
Accelerometer sampling rate	60 Hz
Network wake-up interval	100 sec
Estimated lifespan	3.3 days
Achieved lifespan	3.4 days

TABLE V Measured power consumption and percent time active per component for the AA battery test.

Component	Current drain (at 3.6V)	Percent time active
Transceiver GPS GPS SD card access	23.0 mA 41.4 mA 22.5 mA	3.6% 75.0% <0.1%
μ C and accelerometer Accelerometer SD card access	5.8 mA 25.0 mA	100.0% 1.4%
AVERAGE	38.0 mA	



Fig. 10. CSN-to-CSN range test. This test was conducted across a field with waist-high vegetation. Both collars were elevated 2 m. Bars indicate one standard deviation.

we were able to record the success rate of frames sent between nodes. Figure 10 shows that CSN-to-CSN communication performs reasonably well through and over vegetation. In our first deployment on mountain lions, biologists will approach the animal and manually download data using a hand-held SRN. Thus maximum range is needed. We equipped an SRN with a 12dBi high-gain directional antenna and saw a much improved range for the CSN-to-SRN. An extended range of approximately 150 m proved adequate to approach a mountain lion and download data from its CSN.

Sensor data was transferred between collars less than 10m apart at 63 kbps. This figure does not include network overhead. This data rate is approximately 30 times the rate at which data is collected by a CSN sampling the accelerometer at 60 Hz. Thus a CSN need only spend $\frac{1}{30}$ of its time near a SRN to download all it's data.

C. Network Simulation

We considered four routing methods with varied degrees of message replication, and evaluated each in a network simulation. We assume that messages are buffered in a FIFO queue with older messages being transmitted first.

Epidemic: Starting with the head of the FIFO buffer, send all messages to all neighbors. Each coyote records the neighbors to which it sent a message so they are not retransmitted.



Fig. 11. CSN-to-SRN station range test. This test was conducted across a field of waist-high vegetation. The base station was equipped with a high-gain (12dBi) directional antenna. Both nodes were elevated 2 m. Bars indicate one standard deviation.

Controlled Epidemic: Similar to Epidemic, except that a coyote only sends messages to those coyotes who have more recently been in contact with a base station. The sending coyote keeps the messages, and will send them again to other coyotes if the opportunity arises.

Single Copy: Coyotes send messages only to coyotes with a more recent time-stamp from a sink, then delete the messages from their own buffer.

Multi-copy Forwarding: Coyotes send messages to those coyotes with a more recent time-stamp from a sink. The source predator (the one who produces the data) keeps the messages, and will resend them again, though only directly to a base station. These messages are also marked in the buffer to be deleted first.

Although studying the mobility of predators in their native habitat is one of the goals of the sensor network, we generated a simple model with which to evaluate our proposed routing protocols. We chose to simulate the network with a relatively social predator, the Coyote (Canis latrans) to allow for both CSN-CSN and CSN-SRN data transfer. A Qualnet [28] simulation model was run for seven days of real time with 16 collared coyotes and four randomly placed base stations in an area of 64 square kilometers. The simulation was run with 10 random seeds for den location, SRN location, and covote movement. The results were averaged over the 10 seeds. Each coyote is assigned a den location to which it returns every eight hours; there is an average of two coyotes assigned to each location. During the remaining time the covotes move randomly around their home within a maximum radius of 2 kilometers. Collared covotes therefore have a population density of 0.25 covotes km⁻². Assuming 25% of all coyotes in a study area were collared, a reasonable estimate of capture success, our total simulated population was 64 coyotes at 1 coyote km⁻². This is a typical population density for coyotes whose population densities range from .2–2.3 coyotes km⁻² [31].

Delivery rate, as a percentage of of data packets successfully delivered to a base station, varied widely between protocols (Table VI). The performance of Epidemic routing suffers since a large amount of bandwidth is wasted retransmitting packets that may have already been successfully delivered. Multicopy Forwarding notably performs better than the Single Copy approach, showing that nodes sometimes needlessly transmit data to neighboring coyotes instead of storing them until a base station is near.

TABLE VI Delivery rates

Routing Protocol	Delivery (%)
Epidemic	13.83
Controlled Epidemic	17.11
Single Copy	21.41
Multi-copy	23.71

Table VII shows the average amount of time between data production and delivery. Again, Multi-copy Forwarding shows the best performance, as the additional message copy enables coyotes to make direct deliveries to a base station and reduce the amount of time messages spend in transit. The Epidemic and Controlled Epidemic protocols both result in high delays because much of the available transmission time is consumed by duplicate messages.

TABLE VII Delivery delay

Routing Protocol	Delay (Hours)
Epidemic	15.62
Controlled Epidemic	12.73
Single Copy	11.38
Multi-copy	10.49

With respect to bandwidth consumed per coyote, Single Copy forwarding proved to be the better choice (Table VIII). It is important to note that during much of the simulation, coyotes are not within range of each other and therefore do not consume any bandwidth. Epidemic routing, as expected, consumes the most bandwidth, even though this does not correlate to the highest delivery rate. Notably, Multi-copy Forwarding consumes more bandwidth than the Single Copy strategy due to direct communication. While this results in a higher delivery ratio with lower delay, it would also result in a higher rate of energy consumption but less than Epidemic and Controlled Epidemic.

TABLE VIII BANDWIDTH CONSUMPTION

Routing Protocol	Bandwidth (Bps)
Epidemic	205.41
Controlled Epidemic	165.47
Single Copy	123.02
Multi-copy	139.36

D. Data Collection Trials with Domestic Dogs

In addition to accelerometer data collected with human trials, we used domestic dogs on a treadmill (Figure 12) and



Fig. 12. Pippin on the treadmill with Mary Zavanelli holding his leash and giving him encouragement.



Fig. 13. Stride frequency and velocity of Pippin. Circles indicate a walking gait and asterisks indicate a galloping gait. Lines indicate best-fit linear regression for each gait.

running next to an electric cart. We analyzed this data to verify that stride frequency observed in video recordings of these trials matched the frequencies found in accelerometer data. In addition, we confirmed that frequency is correlated to speed for different gaits as was shown by Heglund [32] (Figure 13). This shows that speed of the collared predator can be determined from the accelerometer record if gait and stride frequency can be identified.

E. AMDF Analysis

Much of the behavioral analysis of the recorded accelerometer data remains as future work, but we did some preliminary analysis using the average magnitude difference function (AMDF), which allows us to determine the gait and stride frequency of an animal from the accelerometer record.

$$AMDF(t) = \frac{1}{L} \sum_{L}^{i=1} |s(i) - s(i-t)|$$
(2)



(a) Pippin walking on the treadmill at 6mph



(b) Pippin galloping next to the cart at 12mph

Fig. 14. AMDF functions for accelerometer trials with pippin.

In Equation 2, t is the period in seconds, L is the window size of the data to be examined, and s(i) is a normalized sample [33]. Periodicity in the signal is identified by minimums in the AMDF function. We chose this function because it can be computed with integer calculations, primarily multiplications, and a minimum of division operations. Looking to the future where behavior may be identified in real time on CSN, the AMDF function is a good candidate for an MSP430 based embedded system with a hardware multiplier.

Using the AMDF function to analyze data from several trials of Pippin that were also video taped, we were able to confirm that stride frequency and gait can be determined from the accelerometer data (Figure 14). There are two obvious characteristics of the AMDF function that differentiate walking from galloping. The first is the shape of the axis identified by a plus sign. It's shape is very different between galloping and walking. Furthermore, the amplitude of the AMDF function is much larger when Pippin is galloping. By comparing the observed stride frequency from the video to the AMDF function, it appears that the first minimum shared by all 3 axes is the stride frequency of the gait.

To further examine the usefulness of the AMDF function,



Fig. 15. Raw accelerometer data of transition from running (left) to walking (right).

we analyzed accelerometer data from a human running with the collar (Figures 15, 16 and 17). As with Pippin amplitude differences between walking and running would allow for identification of these gaits. Furthermore, the pitch where all three axes are at a first minimum accurately shows $\frac{1}{2}$ the stride period for walking or running. The first minimum shows every footfall: right-left-right-left. The second minimum identifies every other footfall: right-right-right-right or left-left-left-left. These pitches match the observed stride frequency for running and walking of the researcher. Also note that running in sand shows an asymmetric foot-fall pattern, where the second minimum is more pronounced than the first.

The main goal for the accelerometer data is to provide the energy budget for the animal being monitored. The stride frequency of the animal largely determines the animal's energy expenditure. In order to track energy intake as well, more subtle behaviors such as consumption of nourishment and sleep must be identifiable. The AMDF analysis can be used towards computing stride frequency, but for identifying the more subtle behaviors a pattern matching approach will most likely be necessary.

F. Test Deployment

In order to test the deployed system, a network of three nodes was set-up. A domestic dog and a human carried CSNs and a single SRN was placed at Long Marine Lab, Santa Cruz, CA, USA (Figure 18). Data was successfully transferred between all CSNs to the SRN, both directly and indirectly through an intermediate node.

G. Deployment on Mountain Lions

In the fall of 2008, the CARNIVORE system was deployed on mountain lions (*Puma concolor*) in the Santa Cruz Mountains, CA, USA. To date, we have deployed 3 collars and have collected 15 days of accelerometer and GPS data from one collar. Because mountain lions range widely, a portable SRN was used to download data from the mountain lion. A



Fig. 16. AMDF of accelerometer data during walking.



Fig. 17. AMDF of accelerometer data during running.



Fig. 18. GPS data from the test deployment. The yellow track was logged by the GPS on a CSN carried by a researcher. The black track is the GPS data from the CSN carried by a domestic dog. A SRN was located at Long Marine Lab, Santa Cruz, CA, USA (A). The dog lived at B and his owner worked at A. Data was successfully transferred to the SRN via 1 and 2 hops.





Fig. 19. Subset of mountain lion GPS data. This data was collected on October 17. 2008 from a CSN on a mountain lion in the Santa Cruz Mountains, CA, USA. Arrow indicates an event with periodic accelerometer data. See Figure 20.

researcher tracked the animal with the VHF beacon and then downloaded the data. Early analysis of both GPS (Figure 19) and accelerometer data (Figure 20) indicate that both location and acceleration data for the animals is successfully being recorded. An important next step in the development of the system is to observe the mountain lions carrying CSNs in order to correlate the accelerometer record with specific behaviors.

VI. RELATED WORK

Current wildlife telemetry technology from companies such as Telemetry Solutions, Vectronic Aerospace, Lotek and ATS does not allow for remote recovery of high-bandwidth sensor data. They also do not support networking among collars. Remote download of GPS data is available but the radio links used would not accommodate the large amounts of data the CARNIVORE platform records. Cellular GSM technology is also used and has a higher data rate. But GSM modems require cellular infrastructure, the modems use more power, and still do not have a high enough data rate. These off-theshelf solutions only satisfy the requirements for downloading relatively tiny amounts of GPS or other data.

The only comparable system with inter-collar networking capability is ZebraNet [25]. It allowed for inter-node routing of data to a base station. Their collars where much larger than CARNIVORE collars, weighing 1151 g compared to 450 g

for the collar designed for mountain lions. Such a large collar would be unsuitable for many terrestrial predators and would not support high data rate sensors.

Compared to commercially available wildlife tracking collars that allow for remote download, the CARNIVORE platform can deliver a much greater quantity of data. A state-ofthe-art GPS tracking collar of similar weight to the CAR-NIVORE collar with remote download (Quantum 5000 -Telemetry Solutions [18]) can record and transmit 15,000 locations over its lifetime. At 30 bytes per location, this is 450 kB of data compared to 1 GB of data for the CARNIVORE platform.

VII. CONCLUSION

The early results from deployments on Mountain Lions indicate that the CARNIVORE is a viable research tool. Furthermore, system tests prior to deployment and this first field test show that CARNIVORE is a viable option to gather data in a highly-disconnected system. At under \$1000 per CSN, compared to \$2-4000 for commercial collars, the CAR-NIVORE network will prove to be a valuable and affordable tool for wildlife biologists to ask and answer interesting questions about cryptic predators.

As future work, there are several areas for improvement in future versions of the CARNIVORE platform. First, power







(b) AMDF.

Fig. 20. Accelerometer data and AMDF analysis of mountain lion data from October 17, 2008. Time is hours since midnight GMT. See Figure 19 for the location where the data was recorded.

consumption can be further reduced by putting the MSP430 into a low-power mode as much as possible. This would dramatically reduce the baseline power consumption to a fraction of current values. Second, the SRN node needs to be developed as a bridge between the CSNs and the internet to allow for automated collection of data to a central database. Third, simulation of the implemented CARNIVORE network protocol is needed to identify shortcomings and optimize network parameters. Third, more sophisticated algorithms need to be studied to enable behavior extraction from the raw data. And finally, we need to conduct observations of mountain lions carrying CSNs in order to create correlations between specific behaviors and accelerometer signals and provide sufficient data for testing and selecting behavior identifying algorithms.

ACKNOWLEDGMENTS

Funding for this work was provided by the Institution for Innovation in Environment Research: Science, Technology, Policy, Society (STEPS) and the Center for Information Technology Research in the Interest of Society (CITRIS). Pippin, our research dog, was an invaluable in assessing the performance of the collar. Pippin's training to run on a treadmill and next to the electric cart is owed entirely to Mary Zavanelli. Telemetry Solutions graciously helped in the final encapsulation and assembly of the collar. The work presented here was conducted under the Chancellor's Animal Research Committee (U.C. Santa Cruz IACUC) protocol Will05.02.

REFERENCES

- S. J. Cooke, S. G. Hinch, M. Wikelski, R. D. Andrews, L. J. Kuchel, T. G. Wolcott, and P. J. Butler, "Biotelemetry: a mechanistic approach to ecology," *Trends in Ecology and Evolution*, vol. 19, no. 6, pp. 334–343, June 2004.
- [2] J. S. Hammerslough and R. G. Bjorklund, "Radio tracking of prematurely dislodged nestling herons," *Jack-Pine Warbler*, vol. 46, no. 2, pp. 57–61, 1968.
- [3] G. L. Kooyman, "Techniques used in measuring diving capacities of weddell seals," *Polar Record*, vol. 12, pp. 391–394, 1965.
- [4] B. Naef-Daenzer, D. Fruh, M. Stalder, P. Wetli, and E. Weise, "Miniaturization (0.2 g) and evaluation of attachment techniques of telemetry transmitters," *J Exp Biol*, vol. 208, no. 21, pp. 4063–4068, 2005. [Online]. Available: http://jeb.biologists.org/cgi/content/abstract/ 208/21/4063
- [5] Basic Description of the Argos System, Service Argos, Inc., 2005. [Online]. Available: http://www.argosinc.com/documents/sysdesc.pdf
- [6] M. Rutishauser, V. V. Petkov, T. Williams, C. Wilmers, J. Boice, K. Obraczka, and P. Mantey, "Carnivore: A disruption-tolerant system for studying wildlife," in *Proceedings of 19th Internatonal Conference* on Computer Communications and Networks, August 2010.
- [7] R. M. Timm, R. O. Baker, J. R. Bennett, and C. C. Coolahan, "Coyote attacks: An increasing suburban problem," in *Transactions, North American Wildlife & Natural Resources Conf.*, vol. 69, 2004, pp. 47–57.
- [8] M. L. Pace, J. J. Cole, S. R. Carpenter, and J. F. Kitchell, "Trophic cascades revealed in diverse ecosystems," *Trends in Ecology & Evolution*, vol. 14, no. 12, pp. 483–488, 12 1999. [Online]. Available: http://www.sciencedirect.com/science/article/
- [9] J. A. Estes, K. Crooks, and R. Holt, "Ecological role of predators," in *Encyclopedia of Biodiversity*, S. A. Levin, Ed. San Diego, CA: Academic Press, 2001, pp. 857–878.
- [10] R. Kram and C. R. Taylor, "Energetics of running: a new perspective," *Nature*, vol. 346, pp. 265–267, July 1990.
- [11] R. M. Alexander and R. Ker, "Running is priced by the step," *Nature*, vol. 346, pp. 220–221, July 1990.
- [12] V. Petkov, "Design and implementation of the carnivores monitoring network," Master's thesis, University of California, Santa Cruz, March 2006.
- [13] S. Shukla, N. Bulusu, and S. Jha, "Cane-toad monitoring in kakadu national park using wireless sensor networks," in APAN, Cairns, Australia, July 2004. [Online]. Available: citeseer.ist.psu.edu/ shukla04canetoad.html
- [14] A. Mainwaring, D. Culler, J. Polastre, R. Szewczyk, and J. Anderson, "Wireless sensor networks for habitat monitoring," in WSNA '02: Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications. New York, NY, USA: ACM Press, 2002, pp. 88–97.
- [15] I. Crossbow Technology, "Crossbow technology, inc." http://www.xbow.com, Crossbow technology, Inc. [Online]. Available: http://www.xbow.com
- [16] MSP430x1xx Family User's Guide (SLAU049D), Texas Instruments, 2004.
- [17] S. W. T. Solutions, "Sirtrack wildlife tracking solutions," http://www.sirtrack.com/.
- [18] T. Solutions, "Gps collars and gps backpacks for wildlife tracking," http://www.telemetrysolutions.com/1280/wildlife-gps-collars.php.
- [19] ETRXI (ZigBee Ready) Module Product Manual, Revision 1.09 ed., Telegesis, 2005.
- [20] Ember, "Design of an ieee 802.15.4-compliant, embernetTM-ready or emberznetTM-ready communication module using the em2420 radio frequency transceiver," Ember Corporation, Application Note 0057, 2005.

- [21] A. Andersen, "2.4 ghz inverted f antenna," Chipcon Products from Texas Instruments," Application Note, 2007.
- [22] J. Hill, R. Szewczyk, A. Woo, S. Hollar, D. E. Culler, and K. S. J. Pister, "System architecture directions for networked sensors," in *Architectural Support for Programming Languages and Operating Systems*, 2000, pp. 93–104. [Online]. Available: citeseer.ist.psu.edu/382595.html
- [23] Lassen iQ GPS Receiver System Designer Reference Manual, Revision a ed., Trimble, February 2005.
- [24] A. Vahdat and D. Becker, "Epidemic routing for partially connected ad-hoc networks," 2000.
- [25] P. Juang, H. Oki, Y. Wang, M. Martonosi, L. Peh, and D. Rubenstein, "Energy-efficient computing for wildlife tracking: Design tradeoffs and early experiences with zebranet," in ASPLOS, San Jose, CA, October 2002.
- [26] J. Burgess, B. Gallagher, D. Jensen, and B. Levine, "Maxprop: Routing for vehicle-based disruption-tolerant networking," in *Proceedings of IEEE Infocom*, 2006.
- [27] R. Shah, S. Roy, S. Jain, and W. Brunette, "Data mules: Modeling a three-tier architecture for sparse sensor networks," in *First International Workshop on Sensor Network Protocols and Applications*, 2003, pp. 30– 41.

- [28] S. N. Technologies, "Qualnet," http://www.scalable-networks.com/. [Online]. Available: http://www.scalable-networks.com/
- [29] IEEE, "Ieee 802.15 wpan task group 4 (tg4)," http://www.ieee802.org/15/pub/TG4.html. [Online]. Available: http: //www.ieee802.org/15/pub/TG4.html
- [30] D. Estrin, R. Govindan, J. Heidemann, and S. Kumar, "Next century challenges: Scalable coordination in sensor networks," *Proceeds of the International Conference on Mobile Computing and Networking* (MOBICOM), pp. 263–270, 1999.
- [31] F. F. Knowlton, "Prelimary interpretations of coyote population mechanics with some management implications," *Journal of Wildlife Management*, vol. 36, no. 2, pp. 369–382, April 1972.
- [32] N. C. Heglund and C. R. Taylor, "Speed, stride frequency and energy cost per stride: How do they change with body size and gait?" *Journal* of Experimental of Biology, vol. 138, pp. 301–318, 1988.
- [33] K. Sayood, Introduction to Data Compression, 3rd ed. Morgan Kaufmann, December 2005.