Information Bang for the Energy Buck: Towards Energy- and Mobility-Aware Tracking

Philipp Sommer[1,3], Jiajun Liu[1,4], Kun Zhao[1], Branislav Kusy[1], Raja Jurdak[1], Adam McKeown[2], David Westcott[2]
[1]CSIRO, Data61, Brisbane, QLD, Australia
[2]CSIRO, Land & Water, Cairns, QLD, Australia
[3]ABB Corporate Research, Baden-Daettwil, Switzerland
[4]Renmin University of China, Beijing, China
philipp.sommer@ch.abb.com, jiajunliu@ruc.edu.cn, firstname.lastname@csiro.au

Abstract

Context-aware services rely critically on accurate and energy-efficient location tracking. While GPS receivers offer high accuracy positioning, energy harvesting and storage constraints of battery-powered devices necessitate duty-cycling of GPS to prolong the system lifetime. Furthermore, real-world dynamics dictate that the GPS sampling strategy adapts in real-time to achieve optimal positioning performance. We propose an information-based approach to solve the problem of online adaptive GPS sampling. We estimate the current positioning error through dead-reckoning and schedule a new GPS sample when the error exceeds a given threshold. The threshold adapts based on the current energy and movement trends to balance the expected information gain from a new GPS sample with its cost for longer-term tracking performance. We evaluate our approach on empirical traces from wild flying foxes and compare it to strategies that sample GPS using fixed and adaptive duty cycles. Our analysis shows that our information-based GPS sampling strategy reduces the tracking error over the fixed and adaptive duty cycle strategies by up to 96% and 64% respectively, and approaches the performance of the optimal offline sampling strategy.

Categories and Subject Descriptors

C.2.1 [**Computer-Communication Networks**]: Wireless Communication

Keywords

Tracking, Energy Awareness, Harvesting, Scheduler

1 Introduction

Position tracking has become an essential building block for many applications, ranging from location-based services for mobile phones to wildlife tracking. Tracking can either be user-initiated, where users request location from the device, or autonomous, where the tracking devices collect user location information based on events of interest. Advanced tracking systems have used additional sensor inputs for autonomous detection of traffic conditions [24] or potholes [10] in a city, tracking and evaluation of health of outpatients [26], or sharing cycling experiences [9].

Long-term location tracking, however, remains a challenge, with GPS requiring high power consumption to continuously deliver accurate positions. High energy consumption can be often traded off for lower localization accuracy, for example, by triangulating the locations of nearby Wi-Fi access points or cellular base stations. Energy improvements can be substantial [16] when continuously tracking the user context and selecting the most energy-efficient localization algorithm. Advanced techniques can use accelerometers to improve energy efficiency of location tracking [18, 25] and magnetometers to improve trajectory tracking [17] by a factor of three or more.

Nearly all of the existing energy-efficient tracking algorithms focus on minimizing energy consumption subject to an application-specific position error bound. Because of the predictability of their energy budget (for instance, mobile phones are recharged every day), most methods track position within given accuracy constraints and aim to minimize energy consumption in the process. The accuracy constraints then implicitly define the energy budget. For instance, modern smartphones enable users to select their location technology based on available energy; a user can switch off GPS and rely only on Wi-Fi or cell-based location in urban environments if the battery power indicator is low.

In this paper, we are interested in autonomous long-term tracking that relies on energy harvesting. As opposed to having a fixed daily energy budget, we investigate algorithms that adapt to the changing nature of harvested energy. Our focus is to maximize the tracking accuracy given this dynamic energy budget, which is the reverse problem of most existing energy-efficient tracking algorithms. Adaptive tracking highlights the question of how to schedule GPS samples in a given future period on the basis of the available energy. Historical movement patterns, such as typical daily trip times,

EWSN'16, February 15–17, 2016, Graz, Austria.

can be used to forecast movement in the near future and thus guide the choice of parameters for a given GPS sampling strategy. While historical patterns can be used for strategic sampling forecasts, it is important to re-evaluate the GPS sampling strategy periodically in case the current behavior of the tracked object deviates from historical data.

Building on these concepts, we propose energy- and mobility-aware tracking that minimizes tracking errors within the available energy budget. Our strategy uses low-power inertial sensors to detect object motion and schedules GPS sampling within the object motion periods. Specifically, we use an estimate of the available energy and a forecast of the remaining movement duration until the next recharge event to determine suitable parameters for GPS sampling. We use historical movement statistics coupled with instantaneous estimates of displacement from the most likely destination to forecast movement duration. More specifically, we predict the remaining movement distance for a tracking interval on the basis of three estimates: (1) population-level movement distance; (2) individual-based movement distance during the previous interval; and (3) current distance from the most likely destination. Each of these estimates proves to be more accurate in a specific timeframe within the tracking interval.

Both the available energy and the movement forecast are updated in real-time on the mobile node and the GPS sampling strategy is continuously adapted as new estimates become available. We propose an information-based GPS sampling strategy that considers expected movement duration, energy availability, and expected information gain of the next GPS sample. The ratio of the forecast movement duration and energy budget determines an adaptive threshold for the expected tracking error. Tracking error is estimated through dead reckoning and, once it exceeds the adaptive threshold, triggers a new GPS sample. This threshold balances the expected information gain from a new GPS sample with its cost for longer-term tracking performance.

We retrospectively analyze high-frequency GPS traces from wild flying foxes to evaluate our energy- and mobility-aware tracking strategy against online tracking strategies that sample GPS periodically during motion periods with fixed and adaptive duty cycles, as well as an optimal offline tracking strategy. Our results confirm that information-based tracking that relies on energy- and mobility-awareness delivers significant reduction in tracking error compared to static and adaptive GPS sampling and performs close to the optimum especially for longer movement durations.

2 Long-Term Mobility Tracking

Long-term tracking of small mobile entities is a challenging problem with high relevance in ecology, agriculture, and logistics. The key constraint here is energy. The very need to track mobile entities long-term implies that their location is unknown and not readily accessible, which limits opportunities for manually recharging the battery of tracking devices. An alternative approach is to support energy harvesting on tracking devices, such as through solar panels, to replenish energy supplies in situ. With energy harvesting, the available energy budget is subject to the amount of energy that has been harvested in the recent past. Furthermore, decisions to acquire position samples have to be made autonomously and in real-time by the tracking device on the basis of both available energy and likely movement activity as communicating with the device for setting parameters manually might not be feasible.

2.1 Accuracy- vs. Energy-bound Tracking

Most recent work has focused on accuracy-bound tracking. Given an uncertainty or accuracy bound on location, the goal is to minimise energy consumption while operating within this bound to maximise lifetime. The accuracy bound de-facto defines the minimum required energy budget of the position tracking, given a typical motion pattern of the tracked object. If the available energy resources are smaller than the tracking budget, the device will run out of energy and the accuracy-bound tracking will fail to deliver the required performance.

With energy harvesting, a well-designed system can operate near-perpetually as long as it tailors its consumption to its available energy budget. Given this implied longevity for trackers that use energy-harvesting, the focus becomes on how to best use the available energy to maximise the location accuracy within a specific period of time, *i.e.*, the time period between two energy-harvesting events. We refer to this problem as energy-bound tracking. Solving this problem naturally depends on the sampling frequency of GPS and other supporting sensors, which is defined by the underlying movement dynamics of the tracked object. Objects that move rarely can be optimally tracked through a sparse scheduling strategy, while determining the best schedule for tracking highly mobile objects with high accuracy is a challenging problem. For this reason, our work here not only considers the energy availability through harvesting sources, but also the mobility of the tracked objects.

2.2 Limitations of Static GPS Sampling

Current commercial motion trackers use a combination of motion-triggering and time-based duty cycling of GPS for energy management. In particular, motion sensors are used to detect the start and end of motion events according to a given movement threshold and GPS samples are only taken during the motion events. Time-based approaches set a fixed duty cycle for GPS. A simple combination of the two approaches uses motion sensors to detect motion and periodically duty cycles GPS within the movement period to reduce energy consumption. We refer to this approach as static motion-based tracking, which effectively fixes the GPS sampling strategy for the duration of a typical motion event. This approach does not work well when the mobility dynamics or the available energy vary day-to-day. A static scheduler might miss key parts of a trajectory if it underestimates the length of the trajectory or the energy availability. Autonomous long-term tracking applications, therefore, need to adapt GPS sampling in order to reconstruct the original trajectory as accurately and efficiently as possible.

To illustrate the energy and motion duration dynamics, we present data from the flying fox tracking application in Figure 1. We show an example of two trips made by the same animal that vary in duration. We also note large dif-

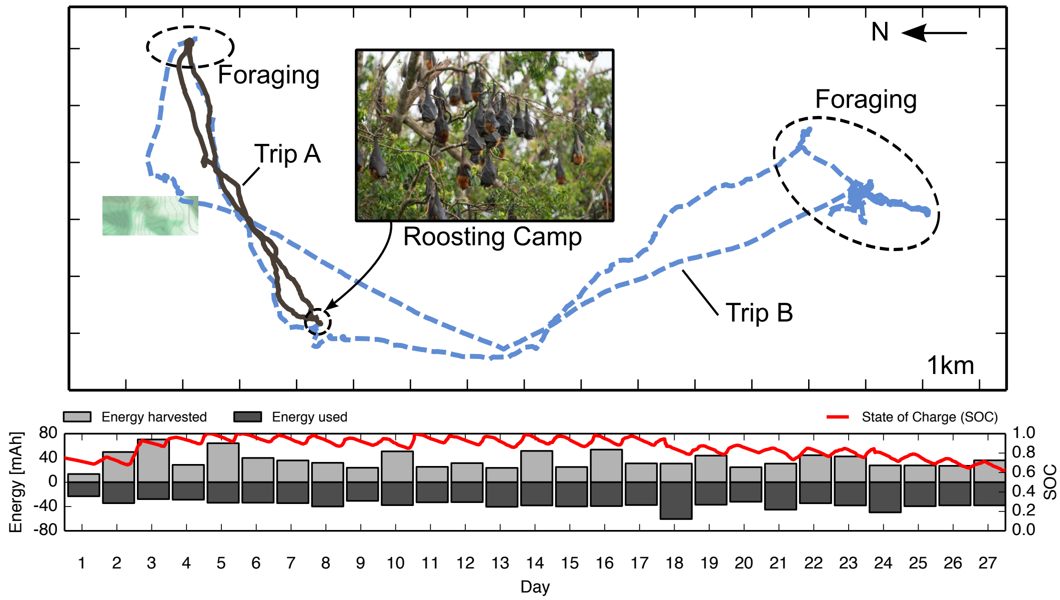


Figure 1: Two GPS trajectories of a spectacled flying fox (Pteropus conspicillatus) tagged with a mobile tracking device (top). Both example trajectories start and finish at the same roosting camp, but feature different motion patterns and durations. Dashed circles indicate the animal's foraging areas. Daily harvested solar energy, consumed energy of the tracking device, and battery state-of-charge (SOC) for the same animal during a period of four weeks (bottom).

ferences in harvested energy for an individual animal on a day-to-day basis over a 4-week period of empirical data, illustrating the likelihood for differences in energy intake and movement duration among individuals and over time for the same individual.

2.3 Motivating Application

Our approach can be applied to tracking any mobile objects that have regular motion patterns and periodic energy harvesting intervals, such as tracking commuting times of people to work, tracking people during leisure activities such as biking, roller-blading, or jogging, and tracking domestic, farm, or wild animals. We focus our evaluation on a specific application of tracking flying foxes in their habitat across Australia. Flying foxes, also known as fruit bats, play an important ecological role as dispersers of pollen and seed in wet tropical areas of Africa and Asia. They come into conflict with humans when their roost sites are located in urban areas, because of their raiding of fruit crops and because they are vectors for a range of emerging infectious diseases which have serious consequences for human health, e.g. Hendra virus in Australia, Nipah in Asia and Ebola in Africa. Flying foxes are highly mobile, with individuals able to fly hundreds of kilometers in a night's foraging. Key to understanding and managing these animals is to understand how they utilize landscapes and how they interact with other disease hosts. This requires a fine-grained understanding of their movement. However, their weight (600 g to 1 kg) limits the weight and size of tracking devices that can be placed on them [31, 23], imposing a tight energy budget on sensing activities, such as GPS sampling.

Behavioral patterns. Flying foxes are nocturnal animals and during the daylight rest in large groups at sites called *camps* or *roosts*. During the day they generally only move within the camp (within a radius of about 100-200 meters) and for most of the time they sleep at a single location. In contrast to the family of microbats, which are common in Europe and the US, flying foxes do not roost in caves, but hang upside down from the top branches of large trees exposed to direct sunlight. At dusk, they leave the camp and fly out to forage at sites that are usually tens of kilometers away. During the night they may change location several times before returning to the camp before sunrise. The long flying journeys between the camp and the foraging area are called commutes. Figure 1 shows the GPS positions collected by the collar of a single animal while it is flying from the roosting camp towards two separate foraging areas on different nights. From an ecological perspective, the movements between camps and foraging sites determine how pollen, seeds and disease spread through the landscape, and the forage locations determine the characteristics of the interactions with disease hosts (other animals, humans). Therefore, it is important that the tracking data captures the movement of the tagged animal during the whole time interval of interest.

The discussion in this section has shown that the mobility dynamics and available daily energy can vary widely among animals or even for the same animal. Thus, a key requirement for maximising tracking accuracy is for the tracker to optimally invest its available energy based on the expected movement trends to guide the selection of optimal GPS sampling parameters.

3 Energy and Mobility Estimation

In this section, we introduce our framework for location tracking of mobile nodes that adapts to changing energy availability and motion patterns. At its core is the *Task*

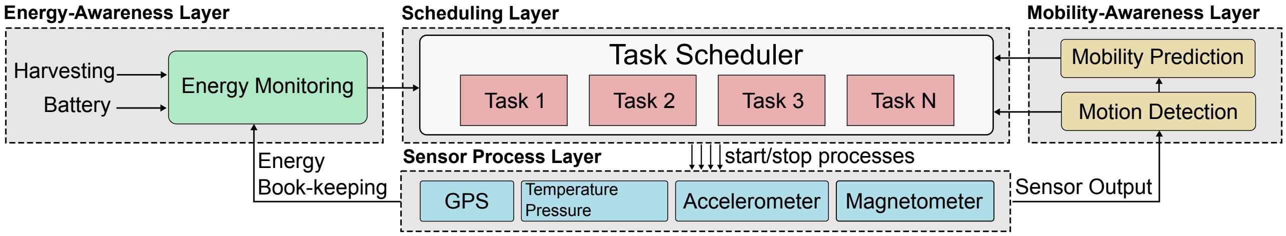


Figure 2: Building blocks of the energy- and mobility-aware scheduling framework.

Scheduler component, which uses inputs from the *Energy-Awareness* and *Mobility-Awareness* layers to schedule sampling of different hardware components in the *Sensor Process* layer, such as the GPS receiver or inertial sensors. The building blocks of the framework are depicted in Figure 2.

3.1 Task Scheduler

We assume that physical or remote access to the mobile node may not be feasible for prolonged time periods and develop mechanisms to support autonomous operation and reconfigurability. Specifically, the scheduler allows for adaptive execution of sensing tasks based on the input from energy and mobility layers. Sensing tasks are associated with a specific process type, *e.g.*, GPS or inertial sensor process, and can be executed or terminated by the scheduler at any time. This design principle allows us to adapt operating conditions of sensing tasks in real-time. For example, the user can configure high-frequency GPS sampling during night time when the animals are flying to foraging areas, while hourly GPS fixes can be acquired during day time when animals are within the roost or if available energy does not support high-frequency tracking.

3.2 Energy Awareness

Precise book-keeping of both energy consumption and energy harvesting is key to maintaining a realistic estimate of the available energy. The *Energy Monitoring* component keeps track of the available energy stored in the batteries, the so called *State of Charge (SOC)*, the energy consumed by the sensor node E_{used}, and the energy harvested by the solar panels $E_{\text{harvested}}$.

Energy Consumption and Harvesting. We keep track of the energy consumption during runtime using a software-based book-keeping mechanism. The instantaneous overall consumption of a sensor node can be broken down into the power consumption of its individual hardware components i, which includes the microcontroller, sensors, radio transceiver, and flash storage chip. The total instantaneous power consumption $P(t)$ of the sensor node can be calculated as the sum of the different components:

$$P(t) = \sum_i P_i(t) = \sum_i V_i(t) \cdot I_i(t) \tag{1}$$

where $P_i(t)$ is the instantaneous power consumption of component i, which depends on the voltage $V_i(t)$ and current consumption $I_i(t)$ of its operating state. We assume that the software is aware of the active operating state of a component and that its current consumption is known. Consequently, we can derive the energy consumed by the node E_{used}, within a specific time interval.

The hardware platform is able to measure the solar panel voltage $V_{\text{solar}}(t)$ and the charge current $I_{\text{solar}}(t)$ into the battery while solar energy harvesting is in progress, which allows to estimate the harvested energy $E_{\text{harvested}}$ per interval.

Separation of Harvesting and Motion Periods. The behavioural patterns of flying foxes imply separation of periods when the energy is used and when the energy is harvested. Interestingly, such a scenario is common for mobile devices like cellular phones or personal health monitors: the devices are typically charged at night and used during the day. We note that the scheduling framework presented in this paper can operate regardless of this separation.

State of Charge Estimation. Accurate estimation of the SOC of a battery is a non-trivial problem, since the battery voltage V_{battery} depends on the instantaneous current draw and remains relatively constant over a large fraction of the SOC range. Consequently, measuring the instantaneous battery voltage alone can only provide a rough estimate of the state of charge. A large body of literature exists on different methods for estimating the SOC of batteries in embedded devices. Approaches range from dedicated hardware chips for energy estimation [7, 13] to software metering approaches that utilize direct battery voltage measurements [2]. In this paper, we use a conflation based method to estimate the SOC, which combines battery voltage measurements with the book-keeping of energy harvesting and energy consumption. Using this approach, software book-keeping can estimate the energy outputs within 10% [28]. Combined with the measurement of solar charge current, we can accurately estimate SOC of the mobile device in software. We periodically update the available energy (SOC) at the end of the interval $[t, t+\tau)$ as follows:

$$SOC(t+\tau) = SOC(t) + E_{\text{harvested}}(t+\tau) - E_{\text{used}}(t+\tau) \tag{2}$$

where $E_{\text{harvested}}(t+\tau)$ denotes the harvested energy during the interval $[t, t+\tau)$ and $E_{\text{used}}(t+\tau)$ is the total energy consumed by different system components.

3.3 Mobility Awareness

It is critical to capture the underlying mobility dynamics of moving objects to predict their future movement accurately. As these dynamics are largely dependent on the type of tracked object, we focus on our motivating application of long-term tracking of flying foxes.

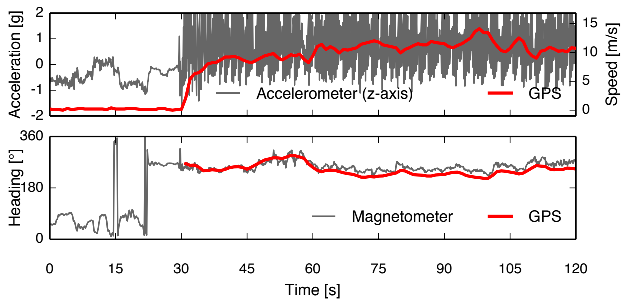


Figure 3: Timeline of z-axis accelerometer and GPS speed (top), and magnetometer heading and GPS speed (bottom) as the animal starts flying at t=30 s.

Algorithm 1 Mobility Prediction

Require: $d_{\text{camp}}(t)$ ▷ Current distance to camp
Require: $d_{\text{prev\_max}}$ ▷ Previous max distance from camp
Require: Δt_{motion} ▷ Estimate for motion duration
Require: $d_{\text{total}}(d)$ ▷ Estimate of total flight distance
Require: $t_{\text{start}}, t_{\text{end}}$ ▷ Observation interval
  procedure PREDICT_REMAINING_DISTANCE(t)
    if $d_{\text{camp}}(t) < d_{\text{prev\_max}}$ **then**
      $\alpha \leftarrow \frac{t - t_{\text{start}}}{t_{\text{end}} - t_{\text{start}}}$ ▷ Weight factor
      $d_{\text{remaining}}(t) \leftarrow (1-\alpha) \cdot \Delta t_{\text{motion}} \cdot v_{\text{avg}} + \alpha \cdot d_{\text{camp}}(t)$
    else
      $d_{\text{remaining}}(t) = d_{\text{total}}(d_{\text{camp}}(t))$
    end if
  end procedure

3.3.1 Motion Detection

In order to facilitate activity detection based on low-power sensor inputs, many mobile tracking platforms feature inertial sensors. For example, the Camazotz hardware platform (see Section 5.1) is equipped with a combined 3-axis accelerometer and 3-axis magnetometer in a single sensor chip. The accelerometer provides samples of the acceleration along its three axes at a sampling frequency between 1 Hz and several kHz. The magnetometer measures the strength of the magnetic field at a data rate between 0.75 Hz and 220 Hz along three axes, which can detect heading changes.

Conditional Task Scheduling. In addition to continuous sampling of acceleration and magnetic field, the accelerometer further provides a programmable interrupt for detection of motion events, *e.g.*, when the animal changes from roosting to flying. This functionality can unburden the microcontroller from having to continuously read samples from the accelerometer's output buffer so that it can remain in sleep mode for most of the time. This interrupt will then set the corresponding status flag in the scheduler, which can trigger start or termination of tasks when motion has been detected (see Figure 2).

Classification of Activity. We show a real-world example for activity detection using inertial sensors on a flying fox tagged with a Camazotz node in Figure 3. GPS data is used to provide ground truth on the animal motion state, motion duration, and heading. In our application scenario, continuous sampling of accelerometer at 10 Hz provides enough information to detect animal motion with high accuracy. Specifically, our motion classifier accumulates acceleration changes along the z-axis in each 5 second window. The animal is classified as flying in the time window, if the average accumulated acceleration change is above a threshold. We use a population-level threshold, the value of which we learn based on a high-frequency dataset with both GPS and accelerometer data that we collected in the initial phase of the animal tracking project.

Validation. We perform 5-fold cross-validation for 20 repetitions on the GPS and accelerator samples that span 137 minutes. In the overall dataset, 47% GPS samples are labeled as the animal being flying, and 53% samples as stationary. For each repetition, we randomly split the samples into 5 independent folds, and use one fold to train the threshold, which is later used in the testing phase on the rest of the data. The speed estimate provided by the GPS is used as the ground truth. If the GPS speed of the animal is greater than 1 meter per second, the animal is considered flying as the ground truth. In this evaluation, our classifier reaches 0.986 average precision and 0.999 average recall in 20 repetitions.

3.3.2 Prediction of Movement Duration

Optimal scheduling depends both on the available energy budget and the amount of energy that will be used for tracking movements. Therefore, it is important that the GPS process can estimate for how long the node will be moving within a certain time interval. Clearly, the accuracy of our prediction will directly influence the energy consumption of the tracking algorithm. If the actual total tracking time exceeds the predicted time, we risk running out of energy early and we sacrifice accuracy of tracking towards the end of the observation period. On the other hand, overestimating the total tracking time will result in conservative budgeting of energy resources and may lead to larger localization errors. The implementation of the mobility prediction module is specific to the designated application scenario and might take into account input from several sensors, last location from GPS, the current time/date information and historical data.

We developed a hybrid approach to predict animal mobility based on individual- and population-based models and real-time estimation of the remaining motion duration based on the remaining distance to the previous roosting camp (see Algorithm 1). We start by using an initial estimate for the total flight distance based on the previous night or an individual-based model. If we detect that the animal has exceeded the distance from camp observed in the previous night $d_{\text{prev\_max}}$, we switch to a population-based model to estimate the total flight distance based on historical trajectories.

Mobility models. We assume that we are given a set of historical trajectories that an object of interest traverses and a certain level of regularity of motion that allows us to extract simple motion features. Note that this motion regularity is shared by many other species including humans [30]. An estimate for the motion duration of an animal per night Δt_{motion} can be calculated by observing typical animal behavior in historical data (see Section 5). Tracking of new animals can be done using a population-level motion model which maps the current distance from the camp to the total flight distance $d_{\text{total}}(d_{\text{camp}}(t))$, while a more accurate individual-level model can be derived over time as more data becomes available.

Distance to home base. Tracked objects may exhibit large variations in motion patterns across the population, or across time (see Figure 1). Prediction of motion duration that is based on simple statistics only, such as the mean duration of motion, will therefore be inaccurate. We observe that flying foxes usually return to the same roosting camp after the nightly foraging (similarly to humans commuting to/from home and work). The current location of the animal can thus be used as a lower bound on the remaining flight distance. This provides us with a lower bound on the remaining motion time, which in turn determines the lower bound on energy required to capture the return trip to the camp. Clearly, this approach can only provide a good estimate after the animal has reached the maximum distance from the camp, which usually happens in the second half of the night. We therefore employ a linear weighting factor α which applies more weight to the individual- or population-based mobility prediction in the beginning of the observation period, while the distance to base becomes more important towards the end of the observation period.

4 GPS Sampling Strategies

The scheduling framework presented in the previous section is agnostic to the actual GPS sampling strategy, and it assumes that each strategy will tune its sampling parameters based on the given constraints and task configuration. In this section, we present three online and one offline strategy to configure sampling of the GPS receiver. The remainder of this section explains in detail how each of the strategies schedules GPS samples.

4.1 Static Motion-based GPS Tracking

The GPS process can be configured to acquire periodic single GPS samples based on motion events within a specific time window. For example, the GPS task can be configured for periodic execution (with period T_{sampling}) within the motion period Δt_{motion}. By specifying the corresponding flag in the task entry conditions we can configure the GPS task to only start when motion has been detected by the inertial sensor process. On each execution of the task a single GPS sample is acquired and the GPS receiver is set to sleep mode afterwards. The energy consumption in the time interval $\Delta t_{\text{interval}} = t_{\text{end}} - t_{\text{start}}$ is proportional to the GPS scheduling period and can be calculated as follows:

$$E_{\text{used}} = \Delta t_{\text{interval}} \cdot P_{\text{baseline}} + k \cdot T_{\text{hotstart}} \cdot P_{\text{tracking}} \quad (3)$$

P_{baseline} denotes the baseline power consumption of the system, P_{tracking} is the additional power used during tracking operation, k denotes the number of samples taken, and T_{hotstart} is the average duration of a GPS hotstart (see Section 5.4). The number of GPS samples that will be collected can be calculated as $k = \lfloor \frac{\Delta t_{\text{motion}}}{T_{\text{sampling}}} \rfloor$.

Performance of the static GPS scheduler critically depends on our initial estimates of the available energy resources and motion patterns of the tracked object, which determine the constant GPS sampling interval T_{sampling}. Specifically, the optimal GPS sampling interval can be calculated using the following inequality: $E_{\text{used}} \leq E_{\text{harvested}}$.

We can get estimates for both the harvested energy and motion duration by studying typical behavior of a population of animals. However, any individual deviations from the typical behavior, for example, due to weather or food availability changes, will result in a suboptimal GPS sampling strategy.

4.2 Adaptive Motion-based GPS Tracking

The adaptive motion-based approach schedules the maximum number of GPS samples given a certain energy budget and an estimate of motion duration. Therefore, it will select a GPS duty cycle that balances between energy consumption and number of GPS samples taken. In contrast to static motion-based tracking that sets a fixed $T_{sampling}$ for the whole time window, the adaptive approach continually updates the GPS sampling interval to meet the energy constraints. Formally, the energy used E_{used} should be lower than the energy budget E_{budget}. Therefore, the resulting upper bound for the number of GPS samples k can be calculated as follows:

$$k(t) \leq \frac{E_{\text{budget}} - \Delta t_{\text{interval}}(t) \cdot P_{\text{baseline}}}{T_{\text{hotstart}} \cdot P_{\text{tracking}}} \quad (4)$$

By distributing the k GPS samples uniformly over the predicted duration of motion Δt_{motion}(t), we obtain the GPS sampling interval T_{sampling}(t) and the corresponding duty cycle D(t) as follows:

$$T_{\text{sampling}}(t) = \frac{\Delta t_{\text{motion}}(t)}{k(t)} \qquad D(t) = \frac{k(t) \cdot T_{\text{hotstart}}}{\Delta t_{\text{motion}}(t)} \quad (5)$$

4.3 Information-based Scheduling of GPS

Periodic duty-cycling of GPS does not take into account the intrinsic characteristics of the trajectory, for example sharp turns, to identify key points that will result in small errors when used as input for interpolation. Sampling inertial sensors *while* the node is in motion can provide a rough estimate of the current heading while using only a small fraction of the power consumption of the GPS (see Figure 3). By employing dead-reckoning using the magnetometer heading, the current position can be extrapolated using the last known GPS position and speed v_{gps}. In each time step, we calculate the tracking errors $e[t_i]$ between dead-reckoning-based position estimates at time t_i and the corresponding interpolated positions on a straight line between the last known GPS position and the most recent position estimate (see Figure 4).

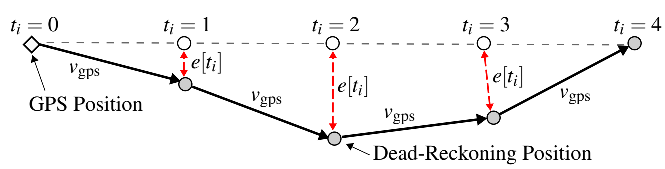


Figure 4: Dead-reckoning based on heading estimation using magnetometer data. The current position is extrapolated from the last known GPS position and speed v_{gps}.

If our estimation of the trajectory based on dead-reckoning exhibits small estimated tracking errors $e[t_i]$, there is little benefit in taking another GPS sample. On the other hand, a large estimated tracking error increases the value of taking another GPS sample to avoid incurring even larger errors in the future. A decision criterion is therefore needed to define the extent of error that warrants taking a new GPS sample. We calculate a threshold $R(t)$ for the expected tracking error based on the duty cycle $D(t)$ using the following equation:

$$R(t) = R_0 \cdot \left(\frac{1}{D(t)} - 1\right)^{\beta} \tag{6}$$

As soon as the estimated maximum tracking error $e_{max} = \max_i e[t_i]$ exceeds the threshold $R(t)$ another GPS sample is acquired, which will restart the dead-reckoning phase. The intuition is to set the GPS sampling policy based on the relationship between the expected movement duration and the energy budget. The ratio of movement duration and energy budget expressed in remaining GPS sample duration is effectively the inverse of $D(t)$ in Eq. 5. As a result, $R(t)$ will increase when the remaining energy budget decreases, which will yield fewer GPS samples and a more conservative sampling policy. On the other hand, the calculated threshold decreases if the remaining energy budget and the expected motion duration are similar (when the expected movement duration roughly matches the budgeted GPS samples), allowing for more aggressive GPS sampling. R_0 and β are tuning parameters for calibrating the scale of the threshold. In case of $\lim_{\beta \to 0}$, $R(t)$ converges to a fixed threshold R_0. The most suitable settings for R_0 and β depend on the characteristics of the trajectories and can be learned from historical data.

4.4 Optimal Offline Scheduling of GPS

As a performance benchmark, we consider an offline oracle-based optimal strategy that provides a lower bound on the tracking errors for a given energy budget by finding the optimal sampling points. The offline optimal strategy cannot be implemented on a mobile device since it requires knowledge of the complete trajectory. Let a sequence $P = \{1, 2, \ldots, N\}$ denote the points of the ground truth trajectory. Suppose we take GPS samples at the first and last points of the trajectory by default, and the energy budget only allows us to take at most m additional GPS samples to form an estimated trajectory. The optimal estimated trajectory is a subsequence of P, denoted by $S = \{1, i_1, \ldots, i_k, \ldots, i_{|S|-2 \leq m}, N\}$, which has the minimum worst-case tracking error $\varepsilon_{opt} = \operatorname{argmin} \varepsilon_{max}(S)$ (see Section 6). This problem is equivalent to the min-ε problem in curve approximation and can be solved by a graph-search approach in polynomial time [4]. The basic idea is to construct a directed graph $G(V, E)$ with the vertices set $V = P$ and edges set $E = \{(i, j) | d_{ij} \leq \varepsilon, i < j\}$ and use G to find the optimal estimated trajectory. Here d_{ij} is the tracking error between the estimated trajectory $\hat{ij}$ to the ground truth trajectory. A shortest path from vertex 1 to vertex N in G then represents the optimal estimated trajectory with tracking error ε.

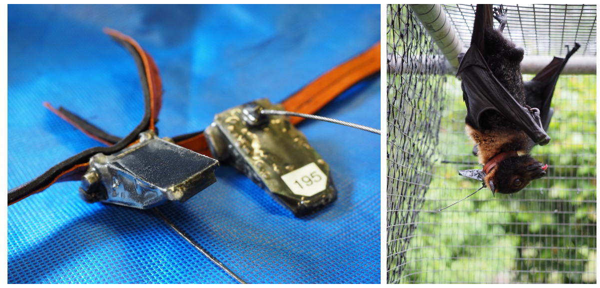


Figure 5: Camazotz collars (left) and a spectacled flying fox (Pteropus conspicillatus) with collar (right).

5 Experimental Setup

In this section, we evaluate the performance of our tracking framework based on experimental data gathered during a deployment on wild flying foxes.[1]

5.1 Hardware Platform

In this paper, we employ the Camazotz platform [15], a low-power sensor platform for GPS location tracking, which is optimized for small size and low weight (see Figure 5). The platform is based on the Texas Instruments CC430F5137 system-on-chip, which combines a microcontroller with a short-range radio transceiver in the 900 MHz band.

Energy Supply and Harvesting. Due to the stringent constraints in terms of weight and form factor, Camazotz employs a small single-cell Lithium-Ion battery with a capacity of 300 mAh and relies on energy harvesting to recharge the battery using a 34x22 mm^2 photovoltaic panel mounted on the outside of the node enclosure. The panel is rated at a maximum current of 40 mA in direct sunlight, however, we often observe a smaller cumulative daily charge between 10 and 50 mAh under typical conditions on animal collars, *e.g.*, on flying foxes roosting in trees.

Satellite-based Positioning (GPS). The Camazotz platform is equipped with the u-blox MAX-6 GPS module, which supports a backup mode (15 µA) where the GPS receiver is disabled, but satellite orbit information (almanac and ephemeris) are retained and the internal real-time clock is still operational. This allows to reduce the idle energy consumption while allowing GPS hotstarts.

Sensors and Communication. The Camazotz platform integrates several sensing modalities such as accelerometer/magnetometer, temperature/pressure sensor and a microphone. All on-board sensors can be put into a sleep state when not used, resulting in a very small baseline power consumption [15]. Camazotz nodes feature a flash storage chip for local data buffering. The short-range radio transceiver is used to offload data to gateway nodes installed at animal congregation areas (flying fox roosting camps) [29].

5.2 Empirical Mobility Traces

Our empirical data set is based on tracking data from 10 free-living flying foxes that were tagged with a Camazotz tracking device. We configured two separate GPS tasks on the tracking device to collect GPS samples to serve as ground

[1]Ethical approval for experiments with animal subjects has been granted.

Animal identifier		1	2	3	4	5	6	7	8	9	10
Number of trips		3	7	7	6	7	8	3	8	5	5
Motion duration/night [minutes]:	Minimum:	70.6	20.0	23.7	4.0	4.3	10.2	9.7	8.7	3.9	9.9
	Average:	82.6	36.5	45.1	7.7	13.5	21.6	23.0	15.0	31.1	13.9
	Maximum:	97.3	81.7	80.9	14.9	18.1	35.3	46.4	22.8	91.4	17.1
Average prediction error (individual) [minutes]		14.7	22.9	26.0	3.1	4.6	7.9	23.4	4.4	30.2	2.4
Average prediction error (population) [minutes]		57.4	15.5	20.5	18.8	12.9	7.4	17.0	11.4	22.3	12.5

Table 1: Actual and predicted trip time statistics for 10 animals and 59 trips in the flying foxes dataset where GPS position samples with high-temporal resolution from motion-triggered continuous tracking are available.

truth location data for our retrospective analysis of different scheduling mechanisms. During night time, sampling of the GPS receiver is triggered using a threshold of +/-2 g on the accelerometer and we acquire continuous GPS samples at a rate of 1 Hz until the speed as reported by the GPS becomes slower than 5 m/s. After that, the GPS is put into sleep mode until the next motion triggering event. As a fallback in case the battery SOC falls below roughly 10%, we only acquire a single GPS sample once every hour (static time-based GPS scheduling) regardless of any motion triggering. Position fixes are considered valid if the position accuracy reported by the GPS is below 50 meters. GPS samples are then stored into the external flash memory and transferred using the short-range wireless radio when within proximity of the base station located in bat roosting camps. During the observation period of several weeks, we select a dataset consisting of 59 trips containing more than 100,000 GPS data points with high temporal resolution, which allows reconstruction of the corresponding trajectories with only minor gaps (*e.g.*, due to hotstarts or temporary signal degradation). GPS heading information is used to emulate magnetometer heading data, as collecting raw samples from the inertial sensors is not feasible due to storage constraints on the mobile nodes. We further employ the empirical traces to learn the parameters used by the information-based tracking algorithm (see Equation 6) by using data from all other trajectories (leave-one-out).

5.3 Energy Consumption and Harvesting

We calculate the energy consumption E_{used} and the harvested solar energy $E_{\text{harvested}}$ for each 24-hour period starting at noon, as shown in Figure 6. Other than the energy consumption of the GPS receiver, we also account for the baseline consumption of the microcontroller, inertial sensors, flash storage chip, and radio transceiver during idle listening, contact logging and data offload to the base station. We report an average energy consumption of 33.0 mAh and an average harvested energy of 27.3 mAh daily.

It is important to note that continuous GPS tracking at 1 Hz during motion consumes more energy than what we can harvest from solar panels on an average day. Therefore, such an operation mode is not sustainable over a longer period of time and can only be executed for a few consecutive days. Once the battery SOC becomes too low for operating the GPS in continuous tracking mode anymore, we switch to a single sample every hour during the night as we always need to guarantee a minimum energy budget for radio beacons.

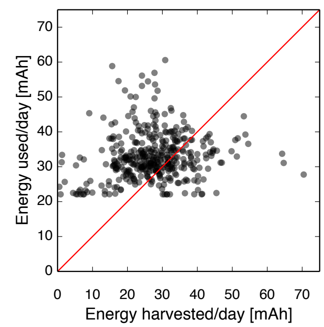


Figure 6: Daily energy consumption and harvested solar energy for the Camazotz nodes deployed on wild flying foxes.

In this case, a configurable hysteresis h_{SOC} on the minimum SOC value effectively disables the motion-triggered continuous GPS sampling task until the battery has recovered to a reasonable charge state.

5.4 Performance of GPS Tracking

GPS receivers need to lock onto the signals from several satellites to obtain an accurate position estimate, a so called GPS fix. Thereby, the time to the first fix (TTFF) after enabling the GPS, highly depends on the internal state of the receiver logic. If the receiver has no up-to-date satellite information and the current time, it needs to acquire this information first by listening to the data transmitted by the GPS satellites, which can take up to a minute or longer (coldstart). Subsequent starts of the GPS can then rely on satellite information locally stored in the receiver's memory and provide a first fix much faster (hotstart). During nightly foraging, GPS hotstarts are dominating (96 %) with an average TTFF of 5.9 seconds. We use this value to estimate the energy consumption associated with a GPS hotstart in our simulation framework (see Section 6).

5.5 Animal Motion Characteristics

We present a summary of motion statistics based on the high-resolution GPS dataset collected with ten tagged animals in Table 1. It can be observed that daily motion duration can vary greatly for individual animals and for the whole population. For each animal, we evaluate performance of trip duration prediction based on individual and population statistics. Specifically, we calculate the prediction error of a trip as the difference between the trip duration and the mean duration of all other trips in the dataset (constrained

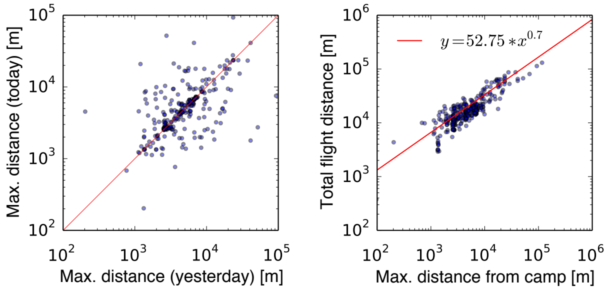


Figure 7: Maximum distance from the camp for two consecutive days (left), and maximum distance from the base camp and the total flight time (right).

to the trips of the same animal for individual-based prediction). Population-based estimation performs better for newly tagged animals for which we do not have many historical traces. However, individual estimates start working better if we have a large enough sample set.

Due to the relatively small sample size of the high resolution dataset, we also consider time periods where we do not have continuous high-resolution GPS trajectories available, but where missing parts of the trajectory can be interpolated based on hourly GPS positions. We then calculate the maximal distance from the camp and the total flight distance within that day (see Figure 7) for a total of 314 trips of 10 nodes. We can observe that today's and yesterday's maximum distance from the camp are weakly correlated (Pearson correlation = 0.35), confirming that yesterday's distance alone is not sufficient for predicting today's distance. Furthermore, the maximum distance from the camp is also related to the total flight distance on that day (Pearson correlation 0.86). We use this to build a model for the total flight distance $d_{\text{total}}(d_{\text{camp}}(t))$ based on the current distance from camp $d_{\text{camp}}(t)$ (see Algorithm 1).

6 Performance Evaluation

We use empirical GPS traces from 10 free living flying foxes to evaluate the performance of different tracking strategies under realistic energy constraints. We select three different power budgets for the operation of the GPS receiver and the inertial sensors. We evaluate our framework in three modes: *static motion-based*, *adaptive motion-based*, and *information-based*. In the static motion-based mode, a fixed duty cycle is selected at the beginning of the night to drive GPS sampling during motion periods that are detected by the inertial sensor. In the adaptive motion-based mode, our framework periodically adjusts the GPS sampling frequency during motion based on its most recent estimates of energy resources and remaining flight time. Information-based sampling uses an adaptive error threshold for GPS sampling decisions based on remaining flight time and on energy. We also compare the performance of our framework to an optimal *offline* schedule calculated with full access to the trajectory.

Metrics. We focus on two metrics when evaluating tracking strategies, namely the tracking error and the power consumption. Given a sampling strategy, the error metric is used to measure the accuracy of the obtained trajectories, and the power consumption characterizes its energy efficiency. The tracking error at the discrete time instant t_i is defined as the Euclidean distance between the ground truth location (x,y) and our estimated location $(\hat{x},\hat{y})$:

$$e[t_i] = \sqrt{(x[t_i]-\hat{x}[t_i])^2 + (y[t_i]-\hat{y}[t_i])^2}. \tag{7}$$

We define the average tracking error e_{avg} and the worst-case tracking error e_{max} as follows:

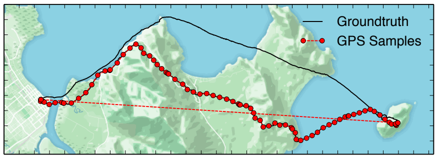
$$e_{avg} = \frac{1}{T}\sum_{i=0}^{T} e[t_i] \qquad e_{max} = \max_{0<i\leq T} e[t_i] \tag{8}$$

6.1 Comparison of GPS Tracking Strategies

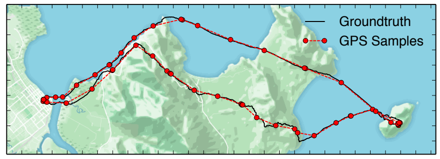
We first provide an example GPS trajectory with a detailed analysis of energy usage and motion prediction for three different sampling strategies. Figure 8 shows an example flight from our empirical dataset and demonstrates the execution of the static, adaptive, and information-based tracking strategies. All three maps show the original trajectory for reference. Clearly, the adaptive and information-based approaches capture the original trajectory well. In contrast, the static approach underestimates the flight time as this animal is flying longer than the population-based estimate would suggest, so the node depletes its energy resources early during the night and suffers high position errors from that point on. The information-based approach captures the tortuous sections of the trajectory better than the adaptive approach, while both perform similarly for relatively straight sections of the trajectory.

Figure 9 shows the energy consumption and predicted motion duration for the three approaches. In the static approach, the node starts with the population-based estimation of flight time and calculates a fixed GPS duty cycle accordingly. It then quickly reaches its energy budget just before midnight and stops sampling after that point. The remaining flight trajectory back to the roosting camp is interpolated, as shown by the dashed line in Figure 8. For the adaptive and information-based strategies, the node starts out with an *individual-based* historical estimate of movement duration, and as the time progresses, the weight of this estimate progressively decreases, while the weight of the current distance from base increases. By the end of the night, the node's travel time estimate converges to the actual flight time. Information-based tracking continuously adapts the error threshold for sampling, and thereby the sampling schedule, according to updated estimates of remaining flight time and energy budget. As a result, the sampled trajectory clearly shows non-uniform sampling patterns, where long straight segments are sampled infrequently and more tortuous segments are sampled with higher frequency. Information-based sampling captures the trajectory well while preserving more than 10% of the energy budget.

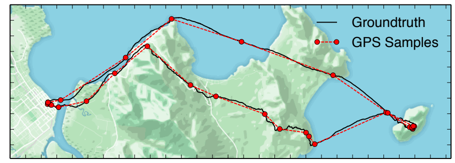
These three example traces demonstrate the benefits of adaptive motion-based and information-based scheduling strategies which take into account an updated estimate for the remaining flight time and available energy budget for tuning



(a) static motion-based tracking, population-based prediction of movement duration



(b) adaptive motion-based tracking, individual-based prediction of movement duration



(c) information-based tracking, individual-based prediction of movement duration

Figure 8: GPS sampling points for static motion-based (a), adaptive motion-based (b), and information-based tracking (c) on an example GPS trajectory of a flying fox. The unit length of the coordinate system is one kilometre. Map source: mapbox.com

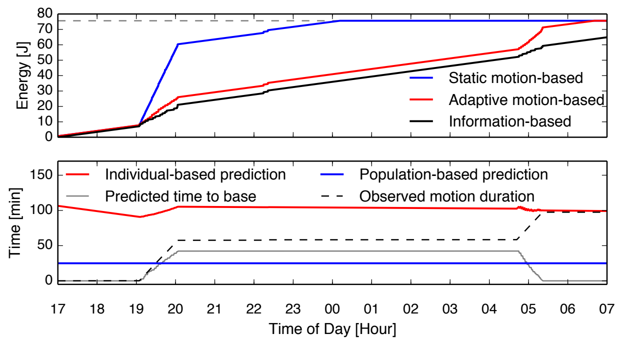


Figure 9: Energy consumption (top) and motion duration (bottom) for an example GPS trajectory of a flying fox.

tracking policy online. It is further apparent that there exists a tradeoff for both the adaptive and information-based strategies between the weight of the initial population-based prediction of flight time and the online motion prediction mechanism proposed in Section 6 for the flying foxes scenario.

6.2 Tracking Performance

We further study the tracking performance on the flying foxes dataset for all strategies given a fixed available energy budget and compare them to an optimal offline algorithm. The distributions of the maximum tracking errors for the different tracking algorithms, energy budgets and individual-/population-based motion prediction modes are shown in Figure 10. We can observe that adaptive approaches clearly outperform static scheduling in terms of the maximum tracking error. Interestingly, the static approach performs even worse with increasing energy budgets. This is due to the fact, that the initially calculated duty cycle is higher and the node will take samples more aggressively, which will cause the energy budget to be exceeded faster when the flight time is longer than predicted, causing high errors towards the end of the night. The benefits of adaptive scheduling become apparent when the motion period exceeds the average motion duration for the whole population, which is around 30 minutes. We observe that adaptive scheduling based on the previous night's motion statistics (*adaptive-prev-population*) only performs slightly worse than an individual-based estimate on multiple trajectories (*adaptive-individual*). Information-based sampling with individual-based prediction delivers consistently better accuracy than both adaptive and static approaches for longer movement durations of 60 minutes or more, while showing comparable accuracy performance for shorter trips. The key differentiator for information-based sampling compared to adaptive sampling is that it tailors the sampling frequency to the estimated information gain from taking the next sample, while considering the forecast movement duration and energy budget. In contrast, adaptive sampling uses periodic sampling of GPS during movement, while adapting the sampling period regularly as it updates its movement forecast.

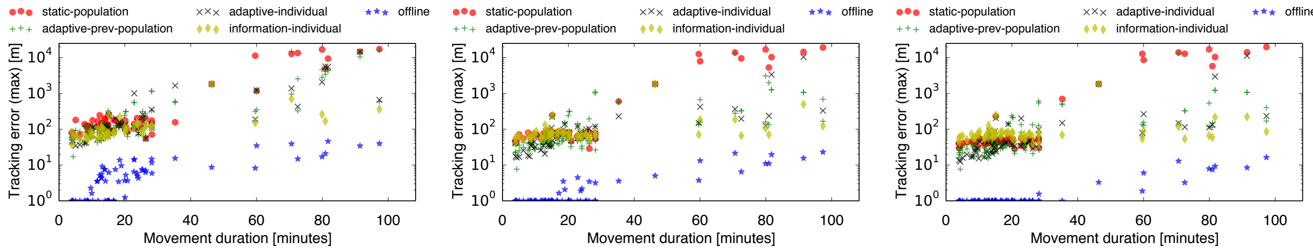
The results for our dataset of 59 empirical 14-hour traces from 10 wild flying foxes are summarized in Figure 11. While adaptive population-based sampling significantly improves accuracy over static sampling, it is slightly outperformed by adaptive individual-based sampling that considers individual histories in forecasting movement duration. Information-based sampling performs best of all online approaches with an accuracy improvement of 96%, 75% and 64% over the static, population-based, and individual sampling strategies respectively for a power budget of 1.5 mW. It delivers these improvements by capturing detailed trajectory features through more frequent sampling, and by reducing sampling rates when the trajectory can be well interpolated between samples.

We also note room for improvement given the tracking error of the optimal offline schedule through sophisticated models for flight destination prediction. As we collect more flight traces from wild flying foxes, this data will improve our prediction model, for example by matching the current trip to a set of previously visited locations or frequently traveled waypoints. This would allow a more accurate estimation of the flight destination and the remaining motion duration.

7 Related Work

Localisation of mobile devices have been studied extensively in the past motivated by applications in navigation, tracking and rescue. Location awareness is also a key requirement for emerging mobile applications such as location-based search, turn-by turn navigation, social media, or participatory sensing [10, 9, 24].

Wildlife tracking. GPS technology was first utilized on larger animals, but with the maturing of the technology the size of the units, and therefore the size of the species it can be used on is decreasing. However, certain wildlife tracking applications cannot rely on GPS due to the application scenario, *e.g.*, tracking badgers underground [8], and thus employ alternative localization techniques such as RFID tags. Lindgren et al. [20] equipped seals with short-range radio



(a) Budget: 1.5 mW

(b) Budget: 2.25 mW

(c) Budget: 3 mW

Figure 10: Distribution of the resulting maximum tracking error on the flying foxes dataset for different tracking strategies given an energy budget corresponding to an average power consumption of 1.5, 2.25 and 3 mW during the observation period.

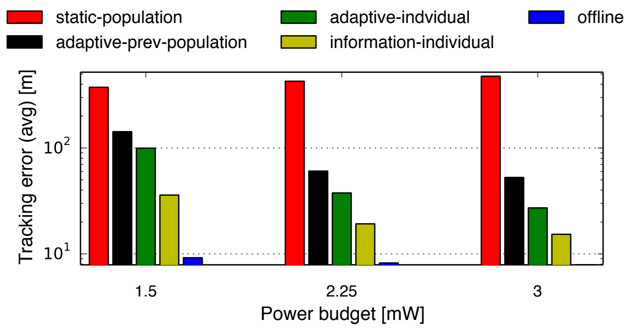


Figure 11: Average tracking errors for different strategies given a specific energy budget on the flying foxes dataset.

transceivers to collect contact logs between tagged animals instead of using GPS to determine the location of the animal. With our focus on tracking in remote areas with limited cellular network coverage, we rely on GPS only to determine the current position of tagged animals. Although the power consumption of GPS receivers has been continually improving, it remains a major part of the overall energy budget of mobile devices. Liu et al. [21] proposed to reduce the power consumption by offloading the GPS signal processing to the cloud. However, this requires to store and transfer a large amount of raw data, which creates another bottleneck for resource-constrained applications.

GPS Duty-Cycling. Continuous tracking applications exhibit a trade-off between accuracy of localization and the amount of energy spent to do so. Therefore, mechanisms to adapt the duty cycle of the GPS receiver and other localization modalities have been extensively studied in previous work [32, 9]. Paek et al. [25] use the location-time history of a user and a blacklist based on cell tower signal strength to decide when to activate the GPS and to avoid unavailability of GPS indoors. SensLoc [16] uses algorithms to detect the user context, such as frequently visited places, and movement detection to adapt the GPS duty cycle. Jurdak et al. [14] propose to use short-range radio contact-logging to bound the position uncertainty when duty-cycling the GPS. SmartDC [5] provides adaptive GPS duty-cycling using prediction mechanisms for regularities in user mobility. eNav [12] employs dead-reckoning using the acceleration in smartphones to enable the GPS only when the user reaches the next navigation waypoint.

Energy Management. Adaptive localization algorithms, such as EnLoc [6], EnTracked [17], and a-Loc [19], dynamically select the localization modality that provides the most energy-efficient position estimate within provided uncertainty bounds. Another approach to duty-cycling the GPS is to rely on low-power sensors such as accelerometer or compass to trigger GPS sampling only when the device is moving [16, 25] or when it changed its heading [17]. Jigsaw [22] manages battery life of mobile phones by balancing power consumption of sensing processes with quality requirements of the application and user context.

Unlike most adaptive localization algorithms, we focus on the inverse problem of determining the most accurate GPS sampling strategy given energy constraints. Consequently, instead of sampling GPS location when the uncertainty of the current location estimate increases above a certain limit, we constrain the GPS sampling subject to the available energy. Our approach will not strictly enforce that all location estimates are within the error bounds, but will strive to achieve best performance overall given the energy constraints.

Energy-aware process scheduling frameworks have been proposed to adjust the long-term system performance based on prediction of harvested energy [3]. While such frameworks consider long-term operation of solar energy harvesting systems over years at a guaranteed minimum performance level, the goal of this paper is to manage short-term energy usage under daily varying harvesting and mobility.

8 Discussion

This paper has presented information-based tracking that maximizes the tracking information return subject to an energy budget. Our strategy uses historical motion statistics and distance from the destination to forecast activity duration to strategically select the best sampling GPS policy. The algorithm then updates its forecast and adapts its acceptable error threshold in response to observed changes in distance from base and energy availability. We evaluated our approach on empirical traces from wild flying foxes and showed that our approach can significantly increase positioning accuracy for a given energy budget.

While our work was motivated by tracking flying foxes, our proposed energy- and mobility-aware scheduling framework is applicable to a broad range of other tracking applications. For instance, most living beings, including humans, exhibit very strong preferential return behaviour to one or two locations, typically home and work or study [11]. There are also strong time-of-day correlations with the most likely destination (going to work at 8 am or returning home at 5 pm [1]). As such, the concepts for scheduling of GPS by predicting motion duration on the basis of previous days, population-based statistics, or current distance from the most likely destination are clearly transferable to tracking people and indeed many other living species [27]. A related area is asset tracking in logistics. Unpowered ground assets often need to be tracked over long durations. Using historic motion patterns, whether for the asset group or individual assets, and energy availability to schedule position sampling can provide more accurate asset tracking.

An interesting direction for future work is to improve motion duration prediction algorithms for more accurate scheduling of GPS samples. Interestingly, information-based sampling can deliver better accuracy while preserving budgeted energy. This is likely a result of training the tuning parameters over a large number of trajectories, which can lead to more conservative sampling for some trajectories. We leave a deeper investigation of this aspect for future work. For flying fox tracking, it will be important to detect whether the animal is flying to a new location, or is on a typical foraging trip returning to the same roosting camp. In addition, temperature and solar exposure inputs can be used as proxies for weather conditions that may better predict the extent of movement and energy demand for a single forecast period. Finally, it will be important to evaluate the information-based tracking strategy for other application scenarios, such as for people or cattle. We expect our work to be an important step towards highly autonomous tracking with energy harvesting that delivers detailed trajectory data near-perpetually for a broad range of applications.

9 Acknowledgments

This work was supported by the Batmon Project in CSIRO's Sensor and Sensor Networks Transformation Capability Platform.

10 References

[1] A. Bamis and A. Savvides. Lightweight Extraction of Frequent Spatio-Temporal Activities from GPS Traces. In *Proc. RTSS*, 2010.

[2] B. Buchli, D. Aschwanden, and J. Beutel. Battery State-of-Charge Approximation for Energy Harvesting Embedded Systems. In *Proc. EWSN*, 2013.

[3] B. Buchli, F. Sutton, J. Beutel, and L. Thiele. Dynamic Power Management for Long-term Energy Neutral Operation of Solar Energy Harvesting Systems. In *Proc. SenSys*, 2014.

[4] W. S. Chan and F. Chin. Approximation of polygonal curves with minimum number of line segments or minimum error. *International Journal of Computational Geometry & Applications*, 6(01):59–77, 1996.

[5] Y. Chon, E. Talipov, H. Shin, and H. Cha. Smartdc: Mobility prediction-based adaptive duty cycling for everyday location monitoring. *IEEE Trans. Mobile Comput.*, 13(3):512–525, 2014.

[6] I. Constandache, S. Gaonkar, M. Sayler, R. R. Choudhury, and L. Cox. EnLoc: Energy-Efficient Localization for Mobile Phones. In *Proc. INFOCOM*, 2009.

[7] P. Dutta, M. Feldmeier, J. Paradiso, and D. Culler. Energy Metering for Free: Augmenting Switching Regulators for Real-Time Monitoring. In *Proc. IPSN*, 2008.

[8] V. Dyo, S. A. Ellwood, D. W. Macdonald, A. Markham, C. Mascolo, B. Pásztor, S. Scellato, N. Trigoni, R. Wohlers, and K. Yousef. Evolution and Sustainability of a Wildlife Monitoring Sensor Network. In *Proc. SenSys*, 2010.

[9] S. B. Eisenman, E. Miluzzo, N. D. Lane, R. A. Peterson, G.-S. Ahn, and A. T. Campbell. Bikenet: A mobile sensing system for cyclist experience mapping. *ACM Trans. Sen. Netw.*, 6(1):6:1–6:39, 2010.

[10] J. Eriksson, L. Girod, B. Hull, R. Newton, S. Madden, and H. Balakrishnan. The Pothole Patrol: Using a Mobile Sensor Network for Road Surface Monitoring. In *Proc. MobiSys*, 2008.

[11] M. C. Gonzalez, C. A. Hidalgo, and A.-L. Barabasi. Understanding individual human mobility patterns. *Nature*, 453(7196), 2008.

[12] S. Hu, L. Su, S. Li, S. Wang, C. Pan, S. Gu, M. T. Al Amin, H. Liu, S. Nath, R. R. Choudhury, and T. F. Abdelzaher. Experiences with eNav: A Low-power Vehicular Navigation System. In *Proc. UbiComp*, pages 433–444, 2015.

[13] X. Jiang, P. Dutta, D. Culler, and I. Stoica. Micro Power Meter for Energy Monitoring of Wireless Sensor Networks at Scale. In *Proc. IPSN*, 2007.

[14] R. Jurdak, P. Corke, D. Dharman, and G. Salagnac. Adaptive GPS Duty Cycling and Radio Ranging for Energy-Efficient Localization. In *Proc. SenSys*, 2010.

[15] R. Jurdak, P. Sommer, B. Kusy, N. Kottege, C. Crossman, A. Mckeown, and D. Westcott. Camazotz: Multimodal Activity-based GPS Sampling. In *Proc. IPSN*, 2013.

[16] D. H. Kim, Y. Kim, D. Estrin, and M. B. Srivastava. SensLoc: Sensing Everyday Places and Paths using Less Energy. In *Proc. SenSys*, 2010.

[17] M. B. Kjaergaard, S. Bhattacharya, H. Blunck, and P. Nurmi. Energy-efficient trajectory tracking for mobile devices. In *Proc. MobiSys*, 2011.

[18] M. B. Kjaergaard, J. Langdal, T. Godsk, and T. Toftkjaer. EnTracked: Energy-Efficient Robust Position Tracking for Mobile Devices. In *Proc. MobiSys*, 2009.

[19] K. Lin, A. Kansal, D. Lymberopoulos, and F. Zhao. Energy-Accuracy Trade-Off for Continuous Mobile Device Location. In *Proc. MobiSys*, 2010.

[20] A. Lindgren, C. Mascolo, M. Lonergan, and B. McConnell. Seal-2-Seal: A Delay-Tolerant Protocol for Contact Logging in Wildlife Monitoring Sensor Networks. In *Proc. MASS*, 2008.

[21] J. Liu, B. Priyantha, T. Hart, H. S. Ramos, A. A. F. Loureiro, and Q. Wang. Energy Efficient GPS Sensing with Cloud Offloading. In *Proc. SenSys*, 2012.

[22] H. Lu, J. Yang, Z. Liu, N. D. Lane, T. Choudhury, and A. T. Campbell. The Jigsaw Continuous Sensing Engine for Mobile Phone Applications. In *Proc. SenSys*, 2010.

[23] A. McKeown and D. Westcott. Assessing the accuracy of small satellite transmitters on free-living flying-foxes. *Australian Ecology*, 37, 2012.

[24] P. Mohan, V. N. Padmanabhan, and R. Ramjee. Nericell: Rich Monitoring of Road and Traffic Conditions using Mobile Smartphones. In *Proc. SenSys*, 2008.

[25] J. Paek, J. Kim, and R. Govindan. Energy-Efficient Rate-Adaptive GPS-based Positioning for Smartphones. In *Proc. MobiSys*, 2010.

[26] J. Ryder, B. Longstaff, S. Reddy, and D. Estrin. Ambulation: A Tool for Monitoring Mobility Patterns over Time Using Mobile Phones. In *Proc. CSE*, 2009.

[27] S. Scellato, M. Musolesi, C. Mascolo, V. Latora, and A. Campbell. NextPlace: A Spatio-temporal Prediction Framework for Pervasive Systems. In *Proc. Pervasive*. 2011.

[28] P. Sommer, B. Kusy, and R. Jurdak. Energy Estimation for Long-Term Tracking Applications. In *Proc. ENSSys*, 2013.

[29] P. Sommer, B. Kusy, A. McKeown, and R. Jurdak. The Big Night Out: Experiences from Tracking Flying Foxes with Delay-Tolerant Wireless Networking. In *Proc. REALWSN*, 2013.

[30] C. Song, Z. Qu, N. Blumm, and A.-L. Barabási. Limits of predictability in human mobility. *Science*, 327(5968):1018–1021, 2010.

[31] D. Westcott et al. The spectacled flying-fox, pteropus conspicillatus. Technical report, June 2001.

[32] P. Zhang, C. M. Sadler, S. A. Lyon, and M. Martonosi. Hardware Design Experiences in ZebraNet. In *Proc. SenSys*, 2004.