Computing (2014) 96:569–585
DOI 10.1007/s00607-013-0334-0

Energy efficiency on location based applications in mobile cloud computing: a survey

Lian Wang · Yong Cui · Ivan Stojmenovic · Xiao Ma · Jian Song

Received: 31 October 2012 / Accepted: 16 May 2013 / Published online: 29 May 2013

Abstract The constrained battery power of mobile devices poses a serious impact on user experience. As an increasingly prevalent type of applications in mobile cloud environments, location-based applications (LBAs) present some inherent limitations concerning energy. For example, the Global Positioning System based positioning mechanism is well-known for its extremely power-hungry attribute. Due to the severity of the issue, considerable researches have focused on energy-efficient locating sensing mechanism in the last a few years. In this paper, we provide a comprehensive survey of recent work on low-power design of LBAs. An overview of LBAs and different locating sensing technologies used today are introduced. Methods for energy saving with existing locating technologies are investigated. Reductions of location updating queries and simplifications of trajectory data are also mentioned. Moreover, we discuss cloud-based schemes in detail which try to develop new energy-efficient locating

L. Wang · Y. Cui (✉) · X. Ma · J. Song
Department of Computer Science, Tsinghua University, Beijing, People's Republic of China
e-mail: cuiyong@tsinghua.edu.cn

L. Wang
e-mail: wanglian12@mails.tsinghua.edu.cn

X. Ma
e-mail: max11@mails.tsinghua.edu.cn

J. Song
e-mail: song-j12@mails.tsinghua.edu.cn

I. Stojmenovic
School of Information Technology and Engineering, University of Ottawa, Ottawa, Canada
e-mail: stojmenovic@gmail.com

I. Stojmenovic
School of Software, Tsinghua University, Beijing, People's Republic of China

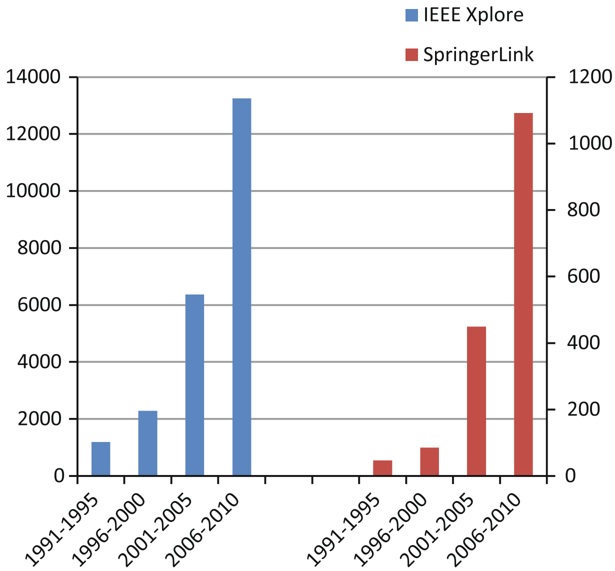

Fig. 1 With mobile and energy in title or abstract

technologies by leveraging the cloud capabilities of storage, computation and sharing. Finally, we conclude the survey and discuss the future research directions.

Keywords Mobile · Cloud computing · Location based service · Energy-efficiency

Mathematics Subject Classification 68-02

1 Introduction

Mobile cloud computing leverages powerful computing and storage resources in the cloud to provide abundant services in mobile environment conveniently and ubiquitously. The features of MCC include no need of up-front investment, lower operating cost, highly scalable and easy access, etc. However, with the characteristics of user mobility and the wireless access pattern, many obstacles such as mobility management, quality of service (QoS) guarantee, energy management, security and privacy issues are brought to MCC. One of the most critical issues among them is the energy efficiency of mobile devices. Since the battery manufacturing industry moves forward slowly (battery capacity grows by only 5 % annually [36]), but the demand for computing and storage capability is rapidly increasing, providing better user experience with constrained battery power supply has become urgent in recent years. Plenty of research has been proposed during the course of the last 5 years as shown in Fig. 1.

As one of the most typical services in MCC, location-based services (LBS) which make use of the geographical position of mobile device, have the advantages of both user mobility and cloud resources in MCC. These services gain user's current position via locating, and provide various location-related services.

The locating technologies used today mainly include GPS, WiFi, and GSM. These technologies can vary widely in energy consumption and localization accuracy. Experiments have shown that, when power consumption is translated to net battery life of a phone, GPS is able to run continuously for only 9 h, while WiFi and GSM can be sustained for 40 and 60 h, respectively. At the same time, the individual corresponding localization accuracy of the three is about 10, 40, and 400 m [10], respectively. Recently, most LBAs prefer GPS for its accuracy although it is also perceived as extremely power-hungry. What is worse, the lack of sensor control makes energy consumption more inefficient [23]. Additionally, many LBAs require continuous localization on reasonably long-time scales. Therefore, energy-efficient locating sensing methods must be adopted to obtain accurate position information while expending minimal energy.

To the best of our knowledge, our previous work is the first survey that takes the energy efficiency issue of locating sensing in mobile cloud environment into account in detail [28], while earlier works in the literature usually mitigate the problem by performing different optimizations without considering the additional characteristics brought in by remote computing schemes and user behavior patterns. We extend [28] and pay more attention on cloud-based technologies in this paper. We believe our survey could provide a better understanding of the design of energy-efficient locating sensing in MCC, and pave the way for further research in this area.

The rest of the paper is organized as follows. Section 2 introduces LBAs and locating technologies. Section 3 presents standalone optimizations, mainly including dynamic tracking, management of multiple LBAs and trajectory simplification. Cloud-based schemes are proposed in Sect. 4, including history-based mapping, computation offloading and sharing among mobile devices. Finally, to address future trends, we summarize and conclude the survey in Sect. 5.

2 LBAs and locating technologies

LBA is one of the most typical applications of MCC. It gains user's current position and provide various user position related services (e.g. social network, health care, mobile commerce, transportation and entertainment). Besides, many of these LBAs need continuous position updates, such as My Experience [16], Real Time Traffic, health care applications that visualize daily patterns and habits of patients [37].

The energy consumption involved with location sensing is extremely tremendous. Therefore, energy saving involved with mobile location sensing in MCC is a vital issue that cannot be ignored. Before moving on to the energy saving schemes, we will briefly introduce the locating principles of the main locating technologies used today.

GPS locating is presented in detail in [17]. In GPS locating, the GPS satellites broadcast data to GPS receivers at rate of 50 bps and a full data packet transmission costs 30 s. To calculate the location, a receiver needs three pieces of information: a precise time T, the ephemeris which means a set of visible space vehicles (SVs, i.e. GNSS satellites) and their locations at time T, the pseudoranges which mean the distances from the receiver to each SV at time T. The ephemeris is decoded from data packet by the mobile device every 30 min in general GPS and the pseudoranges are

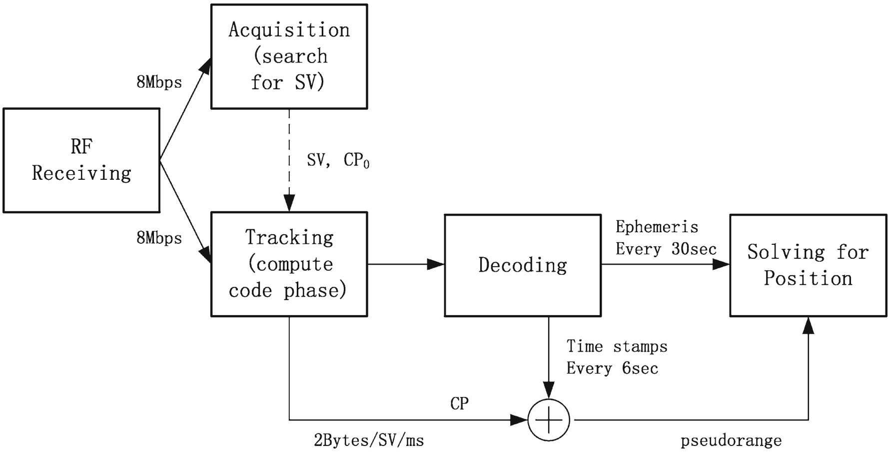

Fig. 2 A simplified illustration of GPS processing stages

calculated using doppler frequency and code phase (CP). There are several stages in the locating process of a typical GPS receiver as shown in Fig. 2. When it first starts up it needs to detect the visible SVs and search through 30+ frequency bins multiplying 8,000+ code phase possibilities for each SV. This stage is called *acquisition*. After acquisition stage, the GPS receiver enters the *tracking* stage which just decodes the time stamp T and adjusts the doppler frequency and code phase of a SV. With time stamps, the ephemeris, doppler frequency and code phase of SVs, the GPS receiver can calculate the location using least-square minimization. A GPS receiver takes minutes to get its first location denoted time to first fix (TTFF) which is rather expensive. The *tracking* is relatively inexpensive, but it also needs continuous running as the GPS receiver will return to *Acquisition* stage after 30 s non-tracking.

Bahl et al. first proposed to use WiFi for locating in [5]. They presented a in-building location system called RADAR which is deployed in a very limited area. Place Lab [8] and Skyhook [2] tried to build a large scale location system using WiFi which leads to the popular use of WiFi locating now. There is a training process before locating with WiFi called *war driving*. In war driving, they use software on WiFi and GPS equipped mobile computers to collect traces of WiFi beacons while driving or walking through a neighborhood and then build a map. Each point in the map consists of a latitude–longitude coordinate measured with GPS and a unique *fingerprint* which contains a set of APs seen and their associated signal strengths in that position. The map is stored in the cloud and needs to be updated for locating. When locating, a mobile user will get a *fingerprint* which is unique to the user's position and send it to the server. The server gets the user's location by searching the closest match to that fingerprint in the map. It can works well in AP dense areas (13–20 m) and may be inaccuracy in more suburban neighborhoods especially rural areas (40 m or more).

Place Lab also researched the use of GSM for locating [22] in a similar way to WiFi but with just 100–150 m accuracy to minimize mapping effort and save more energy.

3 Standalone optimizations

Intuitively, leveraging large intervals between contiguous position updates may minimize the power consumption. The next challenge involves maintaining position accuracy, which is the motivation of the most general solution called dynamic tracking. Additional methods used before or after locating are also proposed to include the management of multiple LBAs and simplification of the trajectories for data transmission. We call these methods *standalone optimizations* which completely depend on the resources on the mobile devices.

3.1 Dynamic tracking

Dynamic tracking powers GPS on and off dynamically to minimize the frequency of GPS sampling. There are mainly two methods used in dynamic tracking: dynamic prediction (DP) and dynamic selection (DS). *Dynamic prediction* dynamically predicts positions and estimates the uncertainty with the aid of less power intensive sensors such as compass and accelerometer after a GPS sampling. GPS will be powered on when the estimated uncertainty in position exceeds an accuracy threshold. *Dynamic selection* dynamically selects among the existing locating technologies of GPS, WiFi and GSM according to the coverage of locating technologies and accuracy requirements of applications.

Dynamic tracking is based on the fact that the power consumption of sensors on a phone differs much. Continuous GPS sampling (once a second) consumes about 400 mW power considering both periodical re-acquisition and tracking on a AT&T Tilt (HTC TyTN II) phone, and the power consumption of WiFi is similar when it scans every second or two [25]. Bluetooth is as power expensive as WiFi as measured in [25]. GSM is much more energy efficient especially when the radio is off. The energy consumed on radio can be removed from GSM locating as the radio is mainly for data communication, then power consumption of GSM locating is less than 5 mW [38]. Accelerometers and compasses also have low power consumption. For example ADXL 330 accelerometers use about 0.6 mW when continuously sampling and the MicroMag3 compass uses about 1.5 mW in continuous sampling [38]. The power consumptions of the locating technologies and mobile sensors are also the basic compositions of the energy models of these schemes.

Leonhardi et al. [24] first studied time-based and distance-based tracking about 10 years ago. Several researches, [15,43] focusing on both energy efficiency and GPS positioning, have formally proposed dynamic tracking techniques. Farrell et al. [15] take into account a constant positioning delay and target speed, while You et al. [43] in take into account a constant positioning accuracy and delay, target speed and acceleration which detects if the target is moving or not. They assume that the parameters mentioned are constant, deeming their methods to be less inefficient and unreliable. Dynamic tracking is further developed in [19,20,25,33]. EnTracked [20], RAPS [33] and $EnTracked_T$ [19] share a similar system structure, while differing from each other in some technologic details. EnTracked and $EnTracked_T$ represent the most typical instances of dynamic prediction while RAPS and a-Loc [25] pay more attention on dynamic selection.

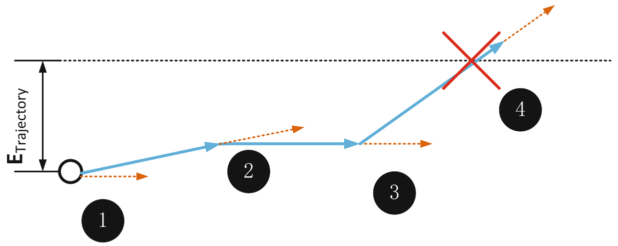

Fig. 3 Illustration of heading-aware strategy

EnTracked [20] can be divided into several steps. First, it uses GPS to obtain a position as a start point and powers it off. Second, it detects the user states (i.e. whether the device is moving or not) using an accelerometer. If the device is not moving, EnTracked waits for a movement. When it is moving, EnTracked estimates the user's speed using the speed and accuracy provided by the GPS module. An error limit (accuracy threshold) is previously given to EnTracked. EnTracked proposes an power model to estimate the energy consumption of a real phone. It then calculates when to power GPS and UMTS radio (for data transmission) on and off based on the parameters (speed, error limit and so on) using the power model to achieve energy savings.

However, this method has several limitations. First, the accelerometer would not be powered off when EnTracked is running. The power used by the accelerometer may be higher than the power consumed in occasionally waking up for a simple position update and to calculate a new sleeping period in some scenarios. Second, it can only detect one sudden move with the device using the raw data from accelerometer. The movement detection can be misled by handset activities and is also deemed to be suitable only for pedestrians with a speed less than 10 m/s.

$EnTracked_T$ [19] extends EnTracked system in several aspects. It proposes the idea of *trajectory tracking* which refers to a sequence of continuous positions corresponding to *position tracking* in EnTracked which focuses on a current position. Firstly, $EnTracked_T$ adopts a heading-aware strategy as illustrated in Fig. 3. It get a precise position using GPS as point 1 and powers GPS off. Then a compass is employed to detect the heading direction and turning. Point 2 and 3 are turning points detected by the compass. The position can be estimated using the speed and directions between two GPS locations. $EnTracked_T$ calculates the accumulated distance traveled orthogonally to the initial heading given by the compass, and compares this with the prescribed trajectory error threshold. Heading deviations will increase the orthogonal distance beyond the threshold and force the GPS position to be updated at point 4. We can see that the trajectory error threshold $E_{Trajectory}$ is much smaller than the distance traveled by the user from point 1 to 4 in Fig. 3 and intervals in GPS usage can be much larger than those in EnTracked. Secondly, $EnTracked_T$ uses adaptive duty cycling strategies for the accelerometer and compass sensors, which make the system more efficient. Thirdly, $EnTracked_T$ uses a speed threshold based strategy together with an accelerometer-based strategy for movement detection. This strategy enables the system to handle different transportation modes, e.g., walking, running, biking and commuting by a car. Fourthly, it explores algorithms of a simplified motion trajectory to reduce data size and communication costs caused by sending motion information.

The error percentage of the $EnTracked_T$ system is relatively high when the requested error threshold is small. Although $EnTracked_T$ claims to have joint *trajectory tracking* and *position tracking*, it seems to work better for trajectory-based applications.

RAPS [33] is based on the observation that GPS is generally less accurate in urban areas. A significant portion of the energy savings of RAPS come from avoiding GPS activation when it is likely to be unavailable. At this time RAPS will use a list of celltower-RSS (the received signal strength) to get the location instead. It records the current celltower ID and RSS information and associates with the success or failure of GPS to help determine if GPS is available. RAPS developed a history-based speed calculation method. It introduces the concept of activity ratio, which is the fraction of time that the user is in motion between two position updates. It uses an accelerometer to detect movement while measuring the activity ratio at the same time. It then uses this activity ratio along with the history of speed information to estimate the current speed of the user. RAPS duty-cycles the accelerometer carefully, using a duty-cycling parameter deduced empirically. Additionally, RAPS also proposed to share location information among mobile users nearby which will be introduced in later section.

RAPS has limitations as well. First, RAPS is mainly designed for pedestrians in urban areas. Second, the user space-time history and the celltower-RSS list must be populated for RAPS to work efficiently and may provide more service for locating with the aid of cloud storage and computation which will be discussed later. Third, its velocity estimation based on activity ratio can be misled by handset activities not related to human motions.

RAPS used both *dynamic prediction* and *dynamic selection*. It uses celltower for locating only when GPS is unavailable just as SenseLess [6] does. a-Loc[25] gives deeper insight into *dynamic selection* which takes both coverage of different locating technologies and different accuracy requirements of applications into consideration. It's based on the observation that the availability and accuracy of locating technologies is variable under different circumstances (eg. indoor or outdoor) and other locating technologies besides GPS may be able to meet the accuracy requirements of some applications. a-Loc then will dynamically select the most energy efficient locating technology according to the current availability and accuracy of locating technologies, their power consumption and the accuracy requirement.

Dynamic prediction and dynamic selection are to reduce the usage of GPS with the aid of less power intensive sensors and can be combined to achieve better energy saving. Considering individual components, more works are related. For example, EEMSS [41], LBAs [44] SenseLess [6] employ the idea of using low power sensors (i.e. accelerometer) to detect user state and context, while triggering activation of high power sensors (i.e. GPS) only if necessary. EnLoc [10] proposes a simple linear predictor, and so on.

3.2 Multiple LBAs management

As more than one location-based application may run on a single smartphone simultaneously, the asynchronous invocations of GPS from different LBAs unnecessarily lead to a higher energy cost. LBAs [44] presents a design principle called Sensing Piggybacking (SP) to overcome this limitation. It proposes a middleware to manage multiple LBAs to avoid unnecessary GPS invoking events.

Applications mainly request and register location sensing in two ways. The first one is One-time Registration, which statically registers a location listener and periodically notifies the listener of location updates based on the specified parameters such as time interval and distance interval. The other type of registration is Multi-time Registration which explicitly registers/unregisters GPS requests to enable hardware sleeping. LBAs focuses on Multi-time Registration, as mobile platforms such as Android have already employed mechanisms to synchronize the location sensing actions for One-time Registration scenarios.

SP listens to the sensing requests of LBAs and forces the incoming registration request to synchronize with existing location-sensing registrations. LBAs uses a triple (G1, T1, D1) to describe the location sensing requirement of the joining LBA, where G1 is the granularity of sensing [e.g.,fine (or GPS) and coarse (or Net)] or the accuracy requirement level, T1 is the minimum time interval and D1 is the minimum distance interval for location updating. It uses (Gf, T2, D2) to denote the existing most fine-grained GPS registration, where T2 and D2 are the minimum sensing intervals. Similarly, it uses (Gc, T3, D3) to denote the existing most fine-grained Net registration. The incoming triple is compared with the existing registration, and SP determines whether to register a new request or simply use the current one according to the granularity and interval requirements. It can re-use the existing sensing registrations and thus eliminate some location-sensing invocations.

Since more than one LBA may be running on one smartphone at the same time, a middleware of multiple LBA management is essential for energy-efficient sensing. This middleware should be redesigned when incorporated with other energy-efficient mechanisms, just the same as SP used with other principles in LBAs.

3.3 Trajectory simplification

Besides locating, the wireless transmission of location data can be optimized too [11,42]. Based on the observation that most services will enforce a more verbose data format for sending trajectory data (e.g. for reasons of cross-platform utilization and web-service compliance), a scheme named trajectory simplification has been proposed as a means to reduce communication overhead by reducing sensing data transferred during location updating. It is usually used for applications which need mobile terminal's trajectory information instead of a single position.

To utilize data redundancy characteristics in trajectory updating, we can enforce a considerably lower amount of data to be sent per time stamped position which results in higher energy savings. The basic idea of trajectory simplification is to replace the obtained positions with a smaller subset, the one which is minimal in size while still reflecting the overall motion information.

As early as in 2000, the academic community has already made several efforts to reduce the amount of data for location updating [24]. However, they mainly focus on reducing data transmissions by decreasing the number of location updates, which may

Table 1 Standalone optimizations for location sensing

Paper	Sensors	Schemes	Accuracy	Power saving[a]
[10] EnLoc	GPS, WiFi, GSM, compass, accelerometer	DS DP	5–400 m	variable
[20] EnTracked	GPS and accelerometer	DP	Variable	Variable
[44] LBAs	GPS, GSM, accelerometer	DS DP	Variable	About 40 %
[19] $\text{EnTracked}_\text{T}$	GPS, compass, accelerometer	DP simplify	Variable	Variable
[6] SenseLess	GPS, WiFi, accelerometer	DS DP	Variable	About 60 %
[33] RAPS	GPS, cell-tower, Bluetooth, accelerometer	DS DP	Avg 80 m	About 70 %
[25] a-Loc	GPS, WiFi, Bluetooth, cell-tower	DS	Variable	About 45 %
[7] MLSA	GPS and accelerometer	DP simplify	Variable	Variable

[a] Compared to simple GPS locating

influence the accuracy of location sensing. These efforts they have made indicate that a ideal simplification method should first provide lossless compression compared with original information.

In EnTracked, trajectory simplification is viewed as a special case of line simplification (which has been thoroughly discussed in the computational geometry community). In $EnTracked_T$, to consider the trade-off between computation cost and simplification benefits, it designed several algorithms and made comparisons. The power consumptions of different algorithms are measured to choose the suitable one and it may be relevant to different applications or mobile systems.

In 2012, Yi Yin et al. [7] have recently proposed an Inter-Frame Coding (IFC) algorithm that exploits spatio-temporal localities on the data communication layer to compress trajectory based locating sensing data. The main innovation of the scheme is that the compressed data can be directly used for spatio-temporal operations without any decompression process, which means the computation overhead on the server side can be easily eliminated. There are two different data points in the algorithm: I (Index) frames which indicate the index data points of a trajectory, and O (Offset) frames which represent the offsets of the subsequent data points that correspond to the I frames. One I frame is usually associated with n O frames, the larger n is, the lower the compression ratio will be. The authors' evaluation results indicate that this approach can achieve up to 50 % compression ratio.

Generally speaking, the simplification scheme in the future should first provide complete information for servers to recover the correct user trajectory. Secondly, it should be self-adaptive with minimal requirements for extra knowledge and other sensors. Thirdly, it should also be lightweight and portable for current mobile devices.

We summarized several typical standalone methods of energy-efficient locating sensing in (Table 1). DS is for dynamic selection among alternative location-sensing mechanisms of GPS, WiFi and GSM in dynamic tracing which DP is for dynamic prediction with less power-intensive sensors such as compass and accelerometer. According to different levels of accuracy requirements, different schemes, locating intervals and/or locating technologies are selected, and thus different levels of power saving are achieved.

4 Cloud-based schemes

Beyond the standalone optimizations mentioned above, many novel schemes have been proposed to utilize the powerful storage and computing resources in mobile cloud computing (MCC) to largely reduce power consumption on the mobile devices. There are mainly two classes of MCC which are both considered here. The first is to access computing resources (services, hardware and systems software) provided by remote centralized data centers [12] without participation of other mobile devices [3]. The second is to combine a collection of smartphones to share computing resources [30] which is also referred as crowd computing [31].

We classify the cloud-based schemes into three classes: (1) history-based mapping which is based on the storage of historic trajectory information; (2) computation offloading which offloads the computation intensive part in locating to the cloud and (3) sharing among mobile devices which share the precise position information of other mobile phones nearby.

4.1 History-based mapping

In recent years, electronic maps such as Google Map [1] have been greatly developed and contain very detailed geography information in global range. With the construction of electronic maps, large amount of location and path information has been collected and stored in the cloud. By making use of these information properly we can get locating schemes which are more energy efficient. Intuitively, people's mobility is often spatio-temporal consistent which means most people would go to the same place at around the similar time of a day or a week. This indicates that we can locate in a much restricted area with better efficiency.

The motivation of history based mapping is to store plenty of historical position or trajectory information associated with both accurate GPS fix and other labels which can be cheaply accessed. It's based on the rich storage and computation resources in the cloud. The idea of storage and mapping is not new. In fact, the currently used locating technologies of WiFi and GSM both work in this way by storing the position of access points (APs) and cell towers as introduced earlier. It has been extended later to leverage more effective information such as road map and user mobility.

Road map is introduced into general locating for better accuracy with low energy consumption. It is first used for vehicle tracking [21,27] to improve accuracy with frequently-sampled GPS. VTrack [39] combined road map matching with WiFi locating to achieve both accuracy and energy efficiency. VTrack is still aimed for traffic tracking and converts radio fingerprints to (lat, lon) coordinates for matching. They extended VTrack and proposed CTrack [38] later to match a sequence of GSM tower observations in historical based database for general locating. CTrack divides a geographic area of interest into uniform square grid cells of a fixed size and associate them with cell tower fingerprints (a list of nearby GSM towers and their signal strengths). It also utilizes various sensors such as compasses and accelerometers on mobile phones to improve the mapping accuracy. CompAcc [9] even eliminated usage of WiFi to

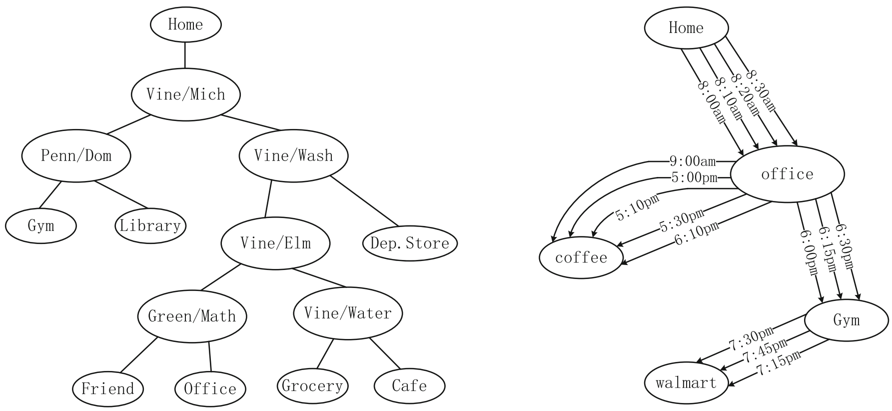

Fig. 4 Personal mobility profile: **a** a spatial logical mobility tree (LMT), **b** a spatio-temporal LMT

avoid expensive war-driving based on the road map. CompAcc uses GPS to get a start path signature and downloads a local digital road map extracted from Google Maps. It then generates directional trails using compasses and accelerometers and matches path signatures with trails on the map. What's more, semantic place learning technologies [4, 18] can help recognize logical places (like McDonalds) instead of physical coordinates (latitude/longitude) and may extend mapping from road level to micro-mobility level in doors.

These methods map the position based on geographic information. The database can be built and updated using war driving without participating of users. However the historical information may be of huge size and rather redundant for a single user. Some researchers proposed to record historical places or paths traveled of individuals, thus data size will be smaller, and response to querying of the same person will be quicker.

EnLoc [10] explored how to make use of the spatio-temporal consistency in user mobility. When exploiting habitual mobility, EnLoc uses the logical mobility tree (LMT) to record the person's actual mobility paths showed in Fig. 4. The vertices of the LMT are also referred to as uncertainty points. The basic idea is to sample the activity at a few uncertainty points, and EnLoc predicts the rest.

The scheme mentioned above highly relies on, as well as is limited to the spatio-temporal consistency in user mobility. It cannot handle users' deviation from habits. So EnLoc further exploits mobility of large populations as a potential indicator of the individual's mobility.

EnLoc hypothesizes that a "probability map" can be generated for a given area from the statistical behavior of large populations. Then an individual's mobility in that area can be predicted. For example, considering a person approaching a traffic intersection of street A: since the person has never visited this street, it is difficult to predict how he/she will behave at the imminent intersection. However, if most people are used to taking a left turn to Street B, the person's movement can be inferred accordingly.

EnLoc has several limitations and leaves room for improvement. EnLoc didn't take speed changes or stop actions into account while moving along a predicted path. It's hard to generate probability maps for many places outside school.

CAPS [34] presents a Cell-ID Aided Positioning System based on the consistency of traveled routes and consistent cell-ID transition points. It is designed for highly mobile users who travel long distances in a predictable fashion. CAPS stores the history of cell-ID sequences on the traveled paths along with the associated GPS position sequences when training, and then senses the cell-ID sequences to estimate the current position using a cell-ID sequence matching technique when tracking. According to the observation, for mobile users with consistent routes, the cell-ID transition point for each user can often uniquely represent the current user position.

CAPS consists of three core components—sequence learning, sequence matching and selection, and position estimation. CAPS opportunistically learns and builds the history of a user's route for future usage. Using a small memory footprint, CAPS maintains the user's past routes and triggers GPS, if necessary. For each cell-ID in a sequence, CAPS maintains a list of tuples following $<position, timestamp>$, where position represents a GPS reading, and timestamp is the time at which that reading was taken. It uses modified Smith–Waterman algorithm for cell-ID sequence matching.

History-based mapping tries to save energy by leveraging the known geography information or building a travel map of a person. WiFi is already in use provided by Skyhook and Google and more efficient mapping technology with good accuracy may be put into practice in the near future too.

4.2 Computation offloading

Offloading is a technology to distribute tasks on mobile phones for remote execution [29]. Tasks can be partitioned and partly offloaded or migrated fully as a whole. It helps mobile devices augment the computational and storage capabilities by leveraging the resources in the cloud. Computation offloading is aimed to offload the computation intensive part of the tasks to the cloud and is considered as one of the important methods to achieve energy saving for mobile terminals. The processing energy cost by CPU is significant and energy can be saved if the overhead of transmission is less than computation.

Some efforts [14,26,35] have been made to partition the GPS receiver and offload the signal processing of decoding and computing to a cloud server.

A-GPS [14,40] was proposed to improve GPS, which inspired the idea of GPS partition. A-GPS makes use of a server equipped with a reference GPS receiver, base stations and a mobile switching center which connects to base stations. The server obtains the cell level location of a mobile user from the base station used by the user via the mobile switching center. It gets the signals from GPS satellites seen by the base station the user uses at the same time. Then it predicts the approximate doppler frequency and code phase of satellites and sends them to the user. Therefore the search space of the actual doppler frequency and code phase of satellites can be much reduced thus TTFF is greatly reduced. The user's GPS receiver then directly sends code phases back to the server for position calculation, or return pseudoranges after some additional signal processing to reduce transmission, or even completes the location fix itself. The main idea of A-GPS is to reduce the uncertainty of the doppler frequency and code phase of satellites, and provide the ephemeris by the server. A-GPS can provide larger scale of coverage.

A-GPS is mainly for improving location experience, and the offloading of power intensive part is not fully considered to reduce energy consumption, thus leaves room for further energy saving.

Based on the works above, a Low Energy Assisted Positioning (LEAP) [35] is proposed to partition the signal process and offload time stamps decoding and least-square location calculation to a cloud server. LEAP mainly adopts a known GPS technique coarse-time navigation (CTN)[40], also called instant GPS, to trade off the local GPS signal processing cost and communication cost. CTN can get the location using code phases of the SVs along with a reference to a known object without decoding. LEAP uses the cell tower that the phone locks to as the reference. LEAP transmits a *LEAP tuple* including two bytes for each code phase, 8 bytes for time stamp, and 8 bytes for cell tower ID to the cloud server for location calculation. The ephemeris is obtained from the Ultra-Rapid Orbits available on the International GNSS Service (IGS) website on the server. LEAP didn't use signals from GPS satellites seen by the base station to reduce TTFF, but provides a fast *reacquisition* mechanism which simplifies the frequency and code phase searching based on previous tracking results to get the code phases. With CTN and fast *reacquisition*, LEAP can aggressively duty-cycle GPS. After *acquisition* or *reacquisition*, LEAP just need to conduct updating of code phases without decoding or location calculation, which further saves energy.

Though LEAP is promising in energy saving, there remain some challenges or limitations. Firstly the real-time clock of the devices may not be synchronized with the satellites. Secondly, a nearby reference location may be hard to get. Thirdly, the signal quality may degrade temporarily due to aggressive duty cycling. What's more, the *acquisition* with much decoding and searching and the code phase calculation on device still consume much power and can be further optimized by offloading. Liu et al. [26] extended LEAP and proposed a cloud-offloaded GPS (CO-GPS) later to eliminate those problems. CO-GPS transmits raw GPS signals to the cloud and conducts the whole signal processing in the server. CO-GPS is also based on LEAP, but it has removed the nearby reference position requirement of the CTN technology and uses an Earth elevation database to generate landmarks for locating. It uses clock radio [32] in different parts of the world to achieve better clock synchronization. Regarding transmission cost, they show that 2 ms of raw data is enough to get a position and 10 ms of raw data (40 kB) can achieve less than 35m location accuracy. They designed and built a GPS sensor platform called CLEO to evaluate the accuracy and efficiency of the solution and presented that they can achieve three orders of magnitude lower energy consumption per location tagging in comparison with 30 s of processing on standalone GPS [26].

By leveraging the rich resources in the cloud, CO-GPS can save energy of signal processing on the mobile devices and reduce the power consumption of the GPS receiver greatly. An important issue for CO-GPS is the trade-off between energy and accuracy. If the accuracy requirement is high, large amount of raw data may be needed to collect and transmit to the server thus power consumed on storage and transmission would grow.

4.3 Sharing among mobile devices

In public places, a mobile phone may get its position information directly from other devices nearby which have already acquired a precise position through GPS. Then the locating process can be skipped and much energy can be saved.

Dhondge et al. [13] proposed an Energy-Efficient Cooperative and Opportunistic Positioning System (ECOPS) to share location information among devices. The main idea of ECOPS is to build a cooperative positioning system among mobile devices with heterogeneous positioning methods by establish an virtual ad-hoc network using WiFi. The mobile devices involved are divided into two classes: Position Broadcasters (PB) and Position Receivers (PR). A GPS-equipped device becomes a PB if it has enough battery life and up-to-date location information, otherwise it's a PR. ECOPS defined a valid time of the location information to tradeoff between location accuracy and energy consumption. PBs ought to start the WiFi hotspot mode and provide the latest location information for PRs. PRs will collect as many GPS coordinates and the corresponding Received Signal Strength Indicator (RSSI) values as possible. To obtain accuracy, a PR needs to find at least three PBs within a threshold distance and uses trilateration to calculate its position. A PR will use the PB's position as its approximate position if it finds only one PB and uses the intersection of the two PBs' if it finds two.

ECOPS can work well for both indoor and outdoor scenes by using RSSI. However it needs to obtain enough devices to achieve good accuracy and obvious energy-saving effect which may not be guaranteed in many environments. Another concern is that PBs may not be willing to spend additional power for broadcasting. Security issues also arise along with position information sharing. The deployment of ECOPS will remain difficult if these problems can not be solved.

Cloud-based schemes are exploring new locating technologies with the aid of outer resources while *standalone* schemes mainly rely on the existing locating technologies. We summarized the cloud-based technologies in Table 2.

5 Conclusion

In this paper, we present an in-depth survey of energy-efficient locating sensing schemes within the environment of MCC.

To reduce power consumption and ensure locating accuracy at the same time, many approaches have tried to reduce the frequency of GPS sampling and dynamically select among the alternative location-sensing mechanisms with the aid of less power-intensive sensors such as compass and accelerometer. Beyond the action of locating, some application-related methods used before or after locating are also proposed in recent years, such as management of multiple LBAs to reduce overall location querying and trajectory simplification to reduce transmission. These methods attempt to make a better use of the existing locating technologies of GPS, WiFi and GSM and completely depend on the resources on mobile platforms, which is called standalone schemes.

By leveraging outer resources, researchers also put forward many ideas to get new energy-efficient locating technologies. Inspired by the training and mapping mode of WiFi locating, history-based mapping schemes have been explored to use the storage

Table 2 Cloud-based schemes

Paper	Sensors	Scheme	Accuracy	Coverage
[8] WiFi	WiFi	War driving and matching	13–40 m	Dense urban areas
[22] GSM	GSM	War driving and matching	100–200 m	Outdoor
[39] VTrack	WiFi	War driving and matching, road map	–	Urban areas
[38] CTrack	GSM	War driving and matching, road maps	40 m in median	Outdoor
[9] CompAcc	Compasses, accelerometers, GPS	Road maps	11 m	Walkers
[10] EnLoc	GPS	Individual trajectory map	5–10 m	Outdoor
[34] CAPS	GSM	Individual trajectory map	75 m in median	Outdoor
[35] LEAP	A-GPS	Offloading	10–40 m	Outdoor
[26] CO-GPS	CO-GPS	Offloading	Variable	Outdoor
[13] ECOPS	WiFi, GPS	Sharing	5–30 m	Device dense areas

of geographical path maps or trajectory maps of individuals. Cloud offloading is also proposed to achieve energy efficiency by offloading the computation intensive part of GPS to the cloud. Location information sharing among mobile devices is also considered. These cloud-based technologies have different coverage and accuracy and can be combined with standalone schemes.

Though large amount of mechanisms have been proposed to provide energy efficient locating-based services for mobile users, some issues are still not well addressed. The construction of power model and the management of various sensors are to be improved. Characteristics of the locating requirements (e.g. accuracy, frequency, what information is needed, lasting time) of existing and potential location-based applications are still to be analyzed. The characteristics of practical location-based applications have hardly been taken into consideration up till now. In addition to that, the accuracy and power consumption of different locating technologies may vary as the environment changes, which still need further study. The potentiality of MCC on energy efficient LBS is to be further explored.

To explore one ideal locating technology to adapt for all applications or users in all circumstances is unrealistic, but different energy efficiency technologies can be implemented and selectively triggered under appropriate circumstances to achieve energy-efficient pervasive locating 1 day. We believe there are still tremendous opportunities for researchers to make great contributions in this field and push forward the development of the industry.

Acknowledgments This work is supported by NSFC (no. 61120106008, 60911130511), National Major Basic Research Program of China (no. 2009CB320501, 2009CB320503). The work of Ivan Stojmenovic was also supported by the Government of China for the Tsinghua 1000 Plan Distinguished Professor (2012–2015) position and by NSERC Discovery grant.

References

1. Google maps api. http://code.google.com/apis/maps/. Accessed Nov 2011
2. Skyhook wireless. http://www.skyhookwireless.com/. Accessed Oct 2012
3. Armbrust M, Fox A, Griffith R, Joseph A, Katz R, Konwinski A, Lee G, Patterson D, Rabkin A, Stoica I et al (2010) A view of cloud computing. Commun ACM 53(4):50–58
4. Azizyan M, Constandache I, Roy Choudhury R (2009) Surroundsense: mobile phone localization via ambience fingerprinting. In: Proceedings of the 15th annual international conference on mobile computing and networking. ACM, New York, pp 261–272
5. Bahl P, Padmanabhan V (2000) Radar: an in-building rf-based user location and tracking system. In: INFOCOM 2000. Proceedings of nineteenth annual joint conference of the IEEE computer and communications societies, vol. 2. IEEE, New York, pp 775–784
6. Ben Abdesslem F, Phillips A, Henderson T (2009) Less is more: energy-efficient mobile sensing with senseless. In: Proceedings of the 1st ACM workshop on networking, systems, and applications for mobile handhelds. ACM, New York, pp 61–62
7. Chang Y, Lin C, Chen L (2012) A two-layer approach for energy efficiency in mobile location sensing applications. NETWORKING 2012, pp 304–315
8. Cheng Y, Chawathe Y, LaMarca A, Krumm J (2005) Accuracy characterization for metropolitan-scale wi-fi localization. In: Proceedings of the 3rd international conference on mobile systems, applications, and services. ACM, New York, pp 233–245
9. Constandache I, Choudhury R, Rhee I (2010) Towards mobile phone localization without war-driving. In: INFOCOM, 2010 Proceedings. IEEE, New York, pp 1–9
10. Constandache I, Gaonkar S, Sayler M, Choudhury R, Cox L (2009) Enloc: energy-efficient localization for mobile phones. In: INFOCOM 2009. IEEE, New York, pp 2716–2720
11. Cui Y, Ma X, Wang H, Stojmenovic I, Liu J (2013) A survey of energy efficient wireless transmission and modeling in mobile cloud computing. Mobile Netw Appl 18(1):148–155
12. Cui Y, Wang H, Cheng X, Li D, Ylä-Jääski A (2013) Dynamic scheduling for wireless data center networks. IEEE Trans Parallel Distrib Syst. doi:10.1109/TPDS.2013.52012.12
13. Dhondge K, Park H, Choi B, Song S (2012) Energy-efficient cooperative opportunistic positioning for heterogeneous mobile devices. In: 2012 21st international conference on computer communications and networks (ICCCN). IEEE, New York, pp 1–6
14. Djuknic G, Richton R (2001) Geolocation and assisted GPS. Computer 34(2):123–125
15. Farrell T, Cheng R, Rothermel K (2007) Energy-efficient monitoring of mobile objects with uncertainty-aware tolerances. In: 11th international database engineering and applications symposium, 2007. IDEAS 2007. IEEE, New York, pp 129–140
16. Froehlich J, Chen M, Consolvo S, Harrison B, Landay J (2007) Myexperience: a system for in situ tracing and capturing of user feedback on mobile phones. In: Proceedings of the 5th international conference on mobile systems, applications and services. ACM, New York, pp 57–70
17. Kaplan E, Hegarty C (2006) Understanding GPS: principles and applications. Artech House Publishers, London
18. Kim D, Kim Y, Estrin D, Srivastava M (2010) Sensloc: sensing everyday places and paths using less energy. In: Proceedings of the 8th ACM conference on embedded networked sensor systems. ACM, New York, pp 43–56
19. Kjærgaard M, Bhattacharya S, Blunck H, Nurmi P (2011) Energy-efficient trajectory tracking for mobile devices. In: Proceedings of the 9th international conference on mobile systems, applications, and services. ACM, New York, pp 307–320
20. Kjærgaard M, Langdal J, Godsk T, Toftkjær T (2009) Entracked: energy-efficient robust position tracking for mobile devices. In: Proceedings of the 7th international conference on mobile systems, applications, and services. ACM, New York, pp 221–234
21. Krumm J, Letchner J, Horvitz E (2007) Map matching with travel time constraints. In: SAE World Congress
22. LaMarca A, Chawathe Y, Consolvo S, Hightower J, Smith I, Scott J, Sohn T, Howard J, Hughes J, Potter F et al (2005) Place lab: device positioning using radio beacons in the wild. Pervasive Comput, 301–306

23. Lane N, Miluzzo E, Lu H, Peebles D, Choudhury T, Campbell A (2010) A survey of mobile phone sensing. In: Communications magazine, vol 48, no 9. IEEE, New York, pp 140–150
24. Leonhardi A, Rothermel K (2001) A comparison of protocols for updating location information. Cluster Comput 4(4):355–367
25. Lin K, Kansal A, Lymberopoulos D, Zhao F (2010) Energy-accuracy trade-off for continuous mobile device location. In: Proceedings of the 8th international conference on mobile systems, applications, and services. ACM, New York, pp 285–298
26. Liu J, Priyantha B, Hart T, Ramos H, Loureiro A, Wang Q (2012) Energy efficient GPS sensing with cloud offloading. In: Proceedings of the 10th ACM conference on embedded networked sensor systems. ACM, New York
27. Lou Y, Zhang C, Zheng Y, Xie X, Wang W, Huang Y (2009) Map-matching for low-sampling-rate gps trajectories. In: Proceedings of the 17th ACM SIGSPATIAL international conference on advances in geographic information systems. ACM, New York, pp 352–361
28. Ma X, Cui Y, Stojmenovic I (2012) Energy efficiency on location based applications in mobile cloud computing: a survey. Procedia Comput Sci 10:577–584
29. Ma X, Cui Y, Wang L, Stojmenovic I (2012) Energy optimizations for mobile terminals via computation offloading. In: 2012 2nd IEEE international conference on parallel distributed and grid computing (PDGC). IEEE, New York, pp 236–241
30. Marinelli E (2009) Hyrax: cloud computing on mobile devices using mapreduce. Tech. rep, DTIC Document
31. Murray D, Yoneki E, Crowcroft J, Hand, S (2010) The case for crowd computing. In: Proceedings of the second ACM SIGCOMM workshop on networking, systems, and applications on mobile handhelds. ACM, New York, pp 39–44
32. Nelson G, Lombardi M, Okayama D (2005) Nist time and frequency radio stations: Wwv, wwvh, and wwvb. NIST Spec Publ 250(67):161
33. Paek J, Kim J, Govindan R (2010) Energy-efficient rate-adaptive gps-based positioning for smartphones. In: Proceedings of the 8th international conference on mobile systems, applications, and services. ACM, New York, pp 299–314
34. Paek J, Kim K, Singh J, Govindan R (2011) Energy-efficient positioning for smartphones using cell-id sequence matching. In: Proceedings of the 9th international conference on mobile systems, applications, and services. ACM, New York, pp 293–306
35. Ramos H, Zhang T, Liu J, Priyantha N, Kansal A (2011) Leap: a low energy assisted GPS for trajectory-based services. ACM Ubicomp 2011, pp 335–344
36. Robinson S (2009) Cellphone energy gap: desperately seeking solutions. Strategy analytics tech, rep
37. Ryder J, Longstaff B, Reddy S, Estrin D (2009) Ambulation: a tool for monitoring mobility patterns over time using mobile phones. In: International conference on computational science and engineering, 2009. CSE’09, vol 4. IEEE, New York, pp 927–931
38. Thiagarajan A, Ravindranath L, Balakrishnan H, Madden S, Girod L, et al. (2011) Accurate, low-energy trajectory mapping for mobile devices. In: Proceedings of 8th USENIX symosium on networked systems design and implementation (NSDI 2011). ACM Press, New York
39. Thiagarajan A, Ravindranath L, LaCurts K, Madden S, Balakrishnan H, Toledo S, Eriksson J (2009) Vtrack: accurate, energy-aware road traffic delay estimation using mobile phones. In: Proceedings of the 7th ACM conference on embedded networked sensor systems. ACM, New York, pp 85–98
40. Van Diggelen F (2009) A-GPS: assisted GPS, GNSS, and SBAS. Artech House Publishers, London
41. Wang Y, Lin J, Annavaram M, Jacobson Q, Hong J, Krishnamachari B, Sadeh N (2009) A framework of energy efficient mobile sensing for automatic user state recognition. In: Proceedings of the 7th international conference on mobile systems, applications, and services. ACM, New York, pp 179–192
42. Xiao Y, Cui Y, Savolainen P, Siekkinen M, Wang A, Yang L, Ylä-Jääski A, Tarkoma S (2013) Modeling energy consumption of data transmission over wi-fi. IEEE Trans Mobile Comput. doi:10.1109/TMC.2013.51
43. You C, Huang P, Chu H, Chen Y, Chiang J, Lau S (2008) Impact of sensor-enhanced mobility prediction on the design of energy-efficient localization. Ad Hoc Netw 6(8):1221–1237
44. Zhuang Z, Kim K, Singh J (2010) Improving energy efficiency of location sensing on smartphones. In: Proceedings of the 8th international conference on mobile systems, applications, and services. ACM, New York, pp 315–330