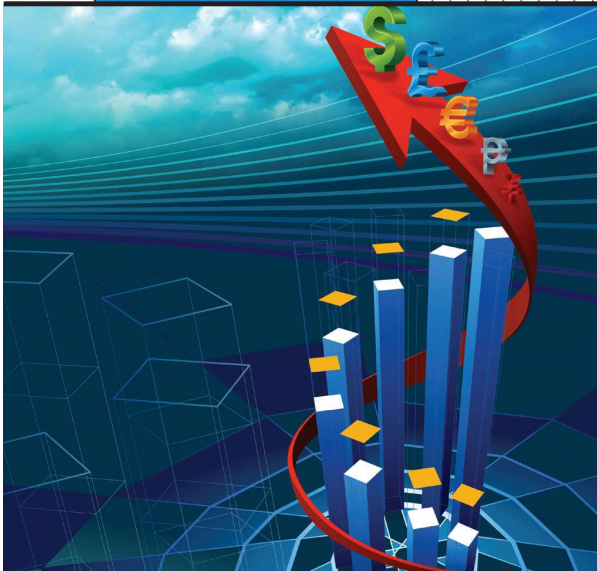




CLOUD COMPUTING FOR MOBILE USERS: CAN OFFLOADING COMPUTATION SAVE ENERGY?

Karthik Kumar and Yung-Hsiang Lu, *Purdue University*

The cloud heralds a new era of computing where application services are provided through the Internet. Cloud computing can enhance the computing capability of mobile systems, but is it the ultimate solution for extending such systems' battery lifetimes?

Cloud computing¹ is a new paradigm in which computing resources such as processing, memory, and storage are not physically present at the user's location. Instead, a service provider owns and manages these resources, and users access them via the Internet. For example, Amazon Web Services lets users store personal data via its Simple Storage Service (S3) and perform computations on stored data using the Elastic Compute Cloud (EC2).

This type of computing provides many advantages for businesses—including low initial capital investment, shorter start-up time for new services, lower maintenance and operation costs, higher utilization through virtualization, and easier disaster recovery—that make cloud computing an attractive option. Reports suggest that there are several benefits in shifting computing from the desktop to the cloud.^{1,2} What about cloud computing for mobile users? The primary constraints for mobile computing are limited energy and wireless bandwidth. Cloud computing can provide energy savings as a service to mobile users, though it also poses some unique challenges.

SAVING ENERGY FOR MOBILE SYSTEMS

Mobile systems, such as smart phones, have become the primary computing platform for many users. Various studies have identified longer battery lifetime as the most desired feature of such systems. A 2005 study of users in 15 countries³ found longer battery life to be more important than all other features, including cameras or storage. A survey last year by ChangeWave Research⁴ revealed short battery life to be the most disliked characteristic of Apple's iPhone 3GS, while a 2009 Nokia poll showed that battery life was the top concern of music phone users.

Many applications are too computation intensive to perform on a mobile system. If a mobile user wants to use such applications, the computation must be performed in the cloud. Other applications such as image retrieval, voice recognition, gaming, and navigation can run on a mobile system. However, they consume significant amounts of energy. Can offloading these applications to the cloud save energy and extend battery lifetimes for mobile users?

Low-power design has been an active research topic for many years. In IEEE Xplore, searching "low" and "power" in the document title produces more than 5,000 results. There are four basic approaches to saving energy and extending battery lifetime in mobile devices:

- *Adopt a new generation of semiconductor technology.* As transistors become smaller, each transistor consumes less power. Unfortunately, as transistors become smaller, more transistors are needed to provide more functionalities and better performance; as a result, power consumption actually increases.

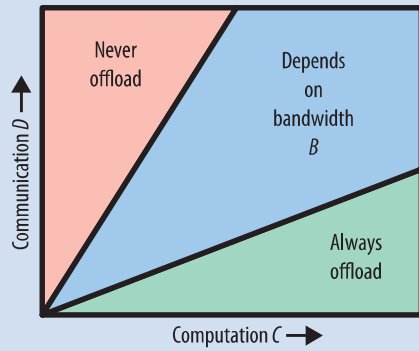


Figure 1. Offloading is beneficial when large amounts of computation C are needed with relatively small amounts of communication D .

- *Avoid wasting energy.* Whole systems or individual components may enter standby or sleep modes to save power.
- *Execute programs slowly.* When a processor's clock speed doubles, the power consumption nearly octuples. If the clock speed is reduced by half, the execution time doubles, but only one quarter of the energy is consumed.
- *Eliminate computation all together.* The mobile system does not perform the computation; instead, computation is performed somewhere else, thereby extending the mobile system's battery lifetime.

We focus on the last approach for energy conservation.

Offloading computation to save energy

Sending computation to another machine is not a new idea. The currently popular client-server computing model enables mobile users to launch Web browsers, search the Internet, and shop online. What distinguishes cloud computing from the existing model is the adoption of virtualization. Instead of service providers managing programs running on servers, virtualization allows cloud vendors to run arbitrary applications from different customers on virtual machines.

Cloud vendors thus provide computing cycles, and users can use these cycles to reduce the amounts of computation on mobile systems and save energy. Thus, cloud computing can save energy for mobile users through *computation offloading*.⁵ Virtualization, a fundamental feature in cloud computing, lets applications from different customers run on different virtual machines, thereby providing separation and protection.

Energy analysis for computation offloading

Various cost/benefit studies focus on whether to offload computation to a server.^{6,7} The following example provides

a simple analysis for this decision.

Suppose the computation requires C instructions. Let S and M be the speeds, in instructions per second, of the cloud server and the mobile system, respectively. The same task thus takes C/S seconds on the server and C/M seconds on the mobile system. If the server and mobile system exchange D bytes of data and B is the network bandwidth, it takes D/B seconds to transmit and receive data. The mobile system consumes, in watts, P_c for computing, P_i while being idle, and P_{tr} for sending and receiving data. (Transmission power is generally higher than reception power, but for the purpose of this analysis, they are identical.)

If the mobile system performs the computation, the energy consumption is $P_c \times (C/M)$. If the server performs the computation, the energy consumption is $[P_i \times (C/S)] + [P_{tr} \times (D/B)]$. The amount of energy saved is

$$P_c \times \frac{C}{M} - P_i \times \frac{C}{S} - P_{tr} \times \frac{D}{B}. \quad (1)$$

Suppose the server is F times faster—that is, $S = F \times M$. We can rewrite the formula as

$$\frac{C}{M} \times \left(P_c - \frac{P_i}{F} \right) - P_{tr} \times \frac{D}{B}. \quad (2)$$

Energy is saved when this formula produces a positive number. The formula is positive if D/B is sufficiently small compared with C/M and F is sufficiently large. The values of M , P_i , P_c , and P_{tr} are parameters specific to the mobile system. For example, an HP iPAQ PDA with a 400-MHz ($M = 400$) Intel XScale processor has the following values: $P_c \approx 0.9$ W, $P_i \approx 0.3$ W, and $P_{tr} \approx 1.3$ W.

If we use a four-core server, with a clock speed of 3.2 GHz, the server speedup F may be given by $(S/M) \approx [(3.2 \times 1,024 \times 4 \times X)/400]$, where X is the speedup due to additional memory, more aggressive pipelining, and so forth. If we assume $X = 5$, we obtain the value of $F \approx 160$.

The value of F can increase even more with cloud computing if the application is parallelizable, since we can offload computation to multiple servers. If we assume that $F = 160$, Equation 2 becomes

$$\frac{C}{400} \times \left(0.9 - \frac{0.3}{160} \right) - 1.3 \times \frac{D}{B} \approx (0.00225 \times C) - 1.3 \times \frac{D}{B}. \quad (3)$$

For offloading to break even, we equate Equation 3 to zero and obtain

$$B_o \approx 577.77 \times \frac{D}{C}, \quad (4)$$

where B_o is the minimum bandwidth required for offloading to save energy, determined by the ratio of (D/C) . If (D/C) is low, then offloading can save energy. Thus, as Figure 1 shows, offloading is beneficial when large amounts of computation C are needed with relatively small amounts of communication D .

Sample applications benefiting from offloading

Two sample applications illustrate the benefits of offloading: a chess game and image retrieval.

Chess is one of the world's most popular games. A chessboard has $8 \times 8 = 64$ positions. Each player controls 16 pieces at the beginning of the game. Chess is Markovian, meaning that the game is fully expressed by the current state. Each piece may be in one of the 64 possible locations and needs 6 bits to represent the location. (This is an overestimate: Some pieces have restrictions—for example, a bishop can move to only half of the board, that is, 32 possible locations). To represent a chess game's current state, it is sufficient to state that $6 \text{ bits} \times 32 \text{ pieces} = 192 \text{ bits} = 24 \text{ bytes}$; this is smaller than the size of a typical wireless packet.

The amount of computation for chess is very large; Claude Shannon and Victor Allis estimated the complexity of chess to exceed the number of atoms in the universe. Chess can be parallelized,⁸ making the value of F in Equation 2 very large. Since the amount of computation C is extremely large, and D is very small, chess provides an example where offloading is beneficial for most wireless networks.

An image retrieval application retrieves images similar in content to a query from an image collection. The program accomplishes this by comparing numerical representations of the images, called *features*. The features for the image collection can be computed in advance; for a query, the program computes its features during retrieval and compares these with the image collection. Since most of the computation is done in advance, less computation is performed online, and the value of C is small. D is large since considerable data must be sent. As a result, even if the values of F become ∞ , D/B might still be too large when compared to C/M in Equation 2. Thus, offloading saves energy only if B is very large—that is, at high bandwidths.

The “Mobile Image Processing” sidebar has more detail on the advantages of mobile devices offloading image retrieval to the cloud.

Making computation offloading more attractive

Analysis indicates that the energy saved by computation offloading depends on the wireless bandwidth B , the amount of computation to be performed C , and the amount of data to be transmitted D . Existing studies thus focus on determining whether to offload computation by predicting the relationships among these three factors.

However, there is a fundamental assumption underlying this analysis with the client-server model: Because the server does not already contain the data, all the data must be sent to the service provider. The client must offload the program and data to the server. For example, typically a newly discovered server for computation offloading does not already contain a mobile user's personal image collec-

➔ MOBILE IMAGE PROCESSING

Mobile devices such as cell phones and PDAs are becoming increasingly popular. Most of these devices are equipped with cameras and have several gigabytes of flash storage capable of storing thousands of images. With such large image collections, two functionalities become important: accessing specific sets of images from the collection, and transmitting the images over a wireless network to other devices and servers for storage.

For accessing a specific set of images, content-based image retrieval (CBIR) can be a better alternative than manually browsing through all of them. For example, a user might want to view all images containing a specific person or captured at a specific location. Mobile image retrieval allows the user to obtain the relevant pictures by comparing images and eliminating the irrelevant matches on the mobile system.

Several studies propose performing CBIR on mobile devices.¹⁻⁴ Because these mobile devices are battery powered, energy conservation is important.²⁻⁴ It is energy efficient to partition CBIR between the mobile device and server depending on the wireless bandwidth.³ As the bandwidth increases, offloading image retrieval saves more energy.

Most of the energy consumption for offloaded applications is due to transmission. For image retrieval, transmitting the images over a wireless network consumes significant amounts of energy. The images may be preprocessed on the mobile device before transmission⁵ to reduce the transmission energy. This reduction in transmission energy is achieved by reducing the file sizes. However, the amount of energy saved depends on the wireless bandwidth and the image contents.

Preprocessing the images saves energy if the reduction in transmission energy compensates for the energy spent due to preprocessing. If the wireless bandwidth is high, the value of the former reduces. Moreover, different images may have different values of the latter based on their contents. Hence preprocessing must be adaptive based on the wireless bandwidth and the image contents. Wireless transmission energy is the most significant bottleneck to energy savings in mobile cloud computing, and such techniques will become increasingly significant as it becomes more popular.

References

1. J. Yang et al., “A Fast Image Retrieval System Using Index Lookup Table on Mobile Device,” *Proc. 19th Int'l Conf. Pattern Recognition (ICPR 08)*, IEEE Press, 2008, pp. 265-271.
2. C. Zhu et al., “iScope: Personalized Multimodality Image Search for Mobile Devices,” *Proc. 7th Int'l Conf. Mobile Systems, Applications, and Services (Mobisys 09)*, ACM Press, 2009, pp. 277-290.
3. Y.-J. Hong, K. Kumar, and Y.-H. Lu, “Energy-Efficient Content-Based Image Retrieval for Mobile Systems,” *Proc. IEEE Int'l Symp. Circuits and Systems (ISCAS 09)*, IEEE Press, 2009, pp. 1673-1676.
4. D. Chen et al., “Tree Histogram Coding for Mobile Image Matching,” *Proc. 2009 Data Compression Conf.*, IEEE CS Press, 2009, pp. 143-152.
5. Y. Nimmagadda, K. Kumar, and Y.-H. Lu, “Energy-Efficient Image Compression in Mobile Devices for Wireless Transmission,” *Proc. IEEE Int'l Conf. Multimedia and Expo (ICME 09)*, IEEE Press, 2009, pp. 1278-1281.

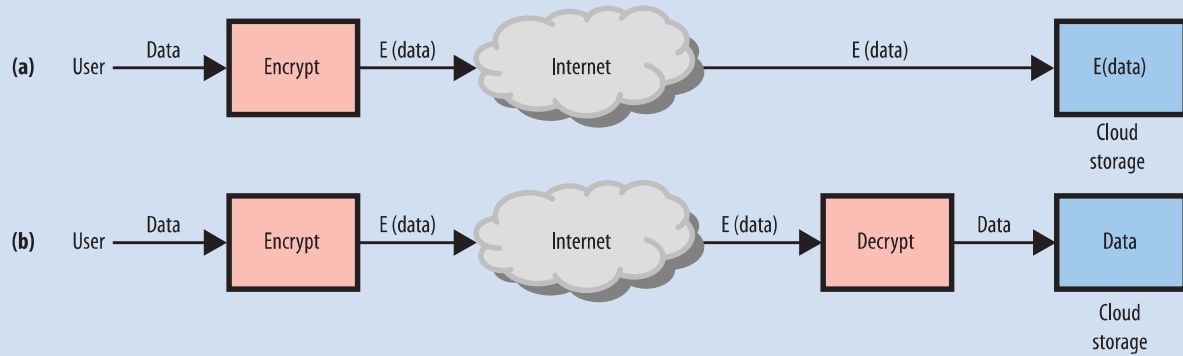


Figure 2. Two encryption scenarios for cloud computing. (a) Data remain encrypted at the cloud storage site, preventing unauthorized access through the Internet; the cloud vendor cannot access the data either. (b) Data are decrypted by the cloud vendor to enable necessary operations on the data.

tion. However, cloud computing changes that assumption: The cloud stores data and performs computation on it. For example, services like Google's Picasa and Amazon S3 can store data, and Amazon EC2 can be used to perform computation on the data stored using S3.

This results in a significant change in the value of D for most applications. There is no longer a need to send the data over the wireless network; it suffices to send a pointer to the data. Also, the value of F is elastic: Large numbers of processors can be obtained on the cloud. This increases the energy savings in Equation 2: A very small D and very large F imply that energy can always be saved.

CHALLENGES AND POSSIBLE SOLUTIONS

Does this make cloud computing the “ultimate” solution to the energy problem for mobile devices? Not quite. While cloud computing has tremendous potential to save energy, designers must consider several issues including privacy and security, reliability, and handling real-time data.

Privacy and security

In cloud computing, Web applications and data replace traditional stand-alone programs, which are no longer stored in users' computers. Shifting all data and computing resources to the cloud can have implications for privacy and security. Because the data is stored and managed in the cloud, security and privacy settings depend on the IT management the cloud provides.

A bug or security loophole in the cloud might result in a breach of privacy. For example, in March 2009, a bug in Google caused documents to be shared without the owners' knowledge,⁹ while a July 2009 breach in Twitter allowed a hacker to obtain confidential documents.¹⁰

Cloud service providers typically work with many third-party vendors, and there is no guarantee as to how these vendors safeguard data. For example, a phishing attack in 2007 duped a staff member for salesforce.com into reveal-

ing a password;¹⁵ the attacker then used the password to access confidential data.

Another potential privacy violation is the “tracking” of individuals through location-based navigation data offloaded to the cloud. Moreover, data stored at one location may not be secure at another due to different access rules. For example, the EU has stricter privacy protection than the US, where laws such as the Patriot Act give government agencies and other entities more power to access personal information. Since the storage locations are typically unknown to the user, determining what laws apply to safeguarding data can be difficult. This may raise serious security concerns for a company that stores its trade secrets in the cloud, or a user who uploads a patentable idea from a mobile system to the cloud.

Clearly, some types of data cannot be stored in the cloud without considering these privacy and security implications. One possible solution is to encrypt data before storage. However, encryption alone will not solve the problem. Figure 2 shows two encryption scenarios. In Figure 2a, the data remain encrypted at the cloud storage site. This can prevent unauthorized access even when the storage is breached in the cloud; the cloud vendor cannot access the data either. In Figure 2b, the cloud vendor decrypts data to perform operations on that data. For example, in the case of a document, the cloud vendor must know which words are used to check spelling; for a spreadsheet, the cloud application must know the numbers for calculation. This is the general case for cloud services.

Another possible privacy and security solution is to use a technique called steganography, as described in the “Data Outsourcing and Privacy” sidebar. Steganographic techniques can be used to transform the data so that operations can be performed without exposing them.

Performing encryption or steganographic techniques before sending data to the cloud requires some additional processing C_p on the mobile system and consumes additional energy. Equation 2 thus can be rewritten as

→ DATA OUTSOURCING AND PRIVACY

Data outsourcing refers to sending data outside users' immediate computing environment. This is an increasingly important concern for cloud computing. As government agencies shift to the cloud (for example, www.apps.gov and www.data.gov), information like healthcare, criminal, and tax records move to the cloud. This makes security and privacy a critical concern.

Multimedia content like images and videos have significant redundancy. This makes it possible to hide data in multimedia using steganography.¹ Figure A shows an example of steganography: images (a) and (c) look identical, but image (c) contains image (b) hidden in it. Applying appropriate transformations to image (c) can obtain image (b). Steganographic techniques can be used to transform the data before storage so that operations can still be performed on the data.

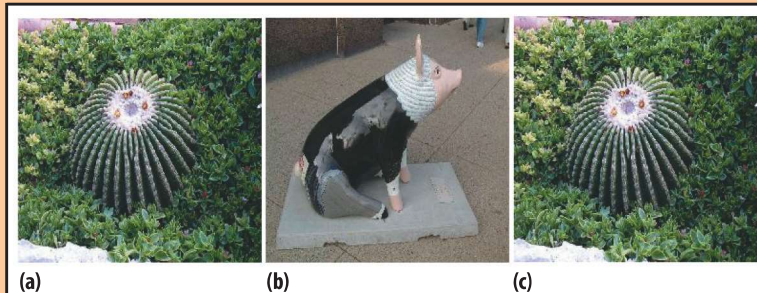


Figure A. An example of steganography: images (a) and (c) look identical, but image (c) contains image (b) hidden in it. Applying appropriate transformations to image (c) can obtain image (b).

For text documents and spreadsheets, however, there is little redundancy. This makes it difficult to transform the data and still perform operations. For example, it is hard to spell check a text document that has alphabets mapped onto symbols. If the mapping is one-to-one and onward, alphabet frequency can be easily used to determine the words. Similarly, if the elements in a spreadsheet are transformed, performing meaningful mathematical operations on them is difficult. Performing computation on encrypted or steganographic data still remains an open research problem.

Developing a hybrid cloud is an interim solution to the data outsourcing problem. A hybrid cloud is "a cloud computing environment in which an organization provides and manages some resources in-house and has others provided externally" (searchcloudcomputing.com). The claim is that this is more feasible currently because large enterprises already have the substantial investments in their infrastructure required to provide resources in-house. In addition, many organizations would prefer to keep sensitive data under their own control to ensure privacy and security.

Reference

1. F.A.P. Petitcolas, R.J. Anderson, and M.G. Kuhn, "Information Hiding—A Survey," *Proc. IEEE, Special Issue on Protection of Multimedia Content*, vol. 87, no. 7, 1999, pp. 1062-1078.

$$\frac{C}{M} \times \left(P_c - \frac{P_i}{F} \right) - P_{tr} \times \frac{D}{B} - P_c \times \frac{C_p}{M}, \quad (5)$$

where $P_c \times (C_p/M)$ is the additional energy required to protect privacy and security. If this value is significant, cloud computing might not save energy for the mobile user. Instead, it may be more energy efficient to perform operations on the mobile system, since $C_p = 0$ when the device performs the computation.

Reliability

Another potential concern with mobile cloud computing is reliability.¹¹ A mobile user performing computation in the cloud depends on the wireless network and cloud service. Dependence on the wireless network implies that cloud computing may not even be possible, let alone energy efficient, when connectivity is limited. This is typical in regions like national parks; a user may thus not be able to organize, retrieve, or identify any images captured in the park. Mobile cloud computing is also difficult in locations such as the basement of a building, interior of a tunnel, or subway. In these cases, where the value of B in Equation 2 can become very small or even zero, cloud computing does not save energy.

Dependence on the cloud for important computations could lead to problems during service outages. Several service providers such as Google, Amazon, and T-Mobile have experienced such outages, as Table 1 shows. These can significantly reduce the value of F in Equation 2.

Data storage presents another reliability problem. In October 2009, both T-Mobile's and Microsoft's mobile Sidekick service crashed, and all customers lost their data and contacts. One option in such scenarios involves an independent backup of data with an alternate service provider, which might increase the value of D in Equation 2.

Real-time data

Some applications—including chess, searching newly captured images for content-based image retrieval, mobile surveillance, and context-aware navigation—have real-time data. In such scenarios, D in Equation 2 is no longer a pointer to the data; it refers to the actual data. For applications like chess, the value of D is small and hence offloading can still save energy. When the value of D is large, offloading may not save energy. In such cases, performing the computation on the mobile system may be more energy efficient. A possible solution is partitioning computation between the mobile system and the cloud to reduce energy consumption. Such a solution may include partially

Table 1. Service provider outages.^{11,12}

Outage date	Vendor	Service
July 2008	Amazon	S3 outage
Feb 2008	Amazon	S3 outage
August 2008	Google	Gmail service outage
October 2009	T-Mobile/Microsoft	Sidekick loses users' data

processing the real-time data on the mobile system. If the processed data are smaller in size, sending the processed data to the server reduces the wireless transmission energy.

Our analysis suggests that cloud computing can potentially save energy for mobile users. However, not all applications are energy efficient when migrated to the cloud. Mobile cloud computing services would be significantly different from cloud services for desktops because they must offer energy savings. The services should consider the energy overhead for privacy, security, reliability, and data communication before offloading. 

Acknowledgments

We thank Jibang Liu for the steganography images used in this article.

References

1. M. Creeger, "CTO Roundtable: Cloud Computing," *ACM Queue*, June 2009, pp. 1-2.
2. Google Tech Talk, "Away with Applications: The Death of the Desktop," 4 May 2007; <http://video.google.com/videoplay?docid=-6856727143023456694>.
3. CNN.com, "Battery Life Concerns Mobile Users," 23 Sept. 2005; www.cnn.com/2005/TECH/ptech/09/22/phone.study.
4. J. Paczkowski, "iPhone Owners Would Like to Replace Battery," *All Things Digital*, 21 Aug. 2009; <http://digitaldaily.allthingsd.com/20090821/iphone-owners-would-like-to-replace-battery-att>.
5. K. Yang, S. Ou, and H.H. Chen, "On Effective Offloading Services for Resource-Constrained Mobile Devices Running Heavier Mobile Internet Applications," *IEEE Comm. Magazine*, vol. 46, no. 1, 2008, pp. 56-63.
6. C. Wang and Z. Li, "Parametric Analysis for Adaptive Computation Offloading," *ACM SIGPLAN Notices*, vol. 39, no. 6, 2004, pp. 119-130.
7. R. Wolski et al., "Using Bandwidth Data to Make Computation Offloading Decisions," *Proc. IEEE Int'l Symp. Parallel and Distributed Processing (IPDPS 08)*, 2008, pp. 1-8.
8. M. Newborn, "A Parallel Search Chess Program," *Proc. ACM Ann. Conf. Range of Computing: Mid-80's Perspective*, ACM Press, 1985, pp. 272-277.
9. J. Kincaid, "Google Privacy Blunder Shares Your Docs without Permission," *TechCrunch*, 7 Mar. 2009; <http://techcrunch.com/2009/03/07/huge-google-privacy-blunder-shares-your-docs-without-permission>.
10. R. McMillan, "Hacker: I Broke into Twitter," *PCWorld.com*, 1 May 2009; www.pcworld.com/businesscenter/article/164182/hacker_i_broke_into_twitter.html.
11. M. Armbrust et al., "Above the Clouds: A Berkeley View of Cloud Computing," tech. report UCB/EECS-2009-28, EECS Dept., Univ. of California, Berkeley, 2009.
12. T-Mobile Forums, "A Message from Our Chief Operations Officer, Jim Alling," 6 Oct. 2009; <http://forums.t-mobile.com/t5/Previous-Sidekick-Models/A-Message-From-Our-Chief-Operations-Officer-Jim-Alling/m-p/200661>.
13. R. McMillan, "Salesforce.com Warns Customers of Phishing Scam," *PCWorld*, 6 Nov. 2007; http://www.pcworld.com/businesscenter/article/139353/salesforcecom_warns_customers_of_phishing_scam.html.

Karthik Kumar is a PhD candidate in the School of Electrical and Computer Engineering at Purdue University. His research interests include energy conservation for mobile systems and resource allocation for scalable computing. Kumar is a student member of IEEE. Contact him at kumar25@purdue.edu.

Yung-Hsiang Lu is an associate professor in the School of Electrical and Computer Engineering at Purdue University. His research focuses on energy efficiency of computing systems. Lu received a PhD in electrical engineering from Stanford University. He is a senior member of IEEE. Contact him at yunghu@purdue.edu.

build your career IN COMPUTING

www.computer.org/buildyourcareer