Collaborative Downloading for Multi-homed Wireless Devices

Ganesh Ananthanarayanan, Venkata N. Padmanabhan, Chandramohan A. Thekkath, Lenin Ravindranath
Microsoft Research

Abstract

Mobile devices are increasingly equipped with multiple network interfaces: Wireless Local Area Network (WLAN) interfaces for local connectivity and Wireless Wide Area Network (WWAN) interfaces for wide-area connectivity. The WWAN typically provides much wider coverage but much lower speeds than the WLAN. To address this dichotomy, we consider collaborative downloading among mobile devices in close proximity. We demonstrate the potential benefits of such an approach and discuss the many challenges to realizing it in practice: incentivizing cooperation by adequately compensating nodes, effecting such cooperation via an efficient protocol, and facilitating it with a suitable user interface. We present our current thinking on these as we design a collaborative downloading system called *COMBINE*.

I. INTRODUCTION

Mobile devices such as laptops, smartphones, and PDAs are increasingly being equipped with multiple wireless network interfaces. These include one or more wireless LAN interfaces (e.g., 802.11, Bluetooth) and wireless WAN interfaces (e.g., GPRS, UMTS). This allows devices to have a choice of radios to use *separately*. Since the WLAN offers much higher speeds (a few to tens of Mbps) than the WWAN (tens to hundreds of Kbps), the conventional wisdom is to use the WLAN interface when in range of a WLAN (e.g., at a WiFi hotspot), but settle for the much lower speeds offered by the WWAN interface at other times. Despite the proliferation of WiFi hotspots, their coverage is still quite limited (e.g., outside the city center or on trains and buses). Furthermore, even in locations with WiFi coverage, policy issues may impede a user's ability to take advantage of it (e.g., the user may not be a subscriber of the hotspot service provider and may have to set up yet another billing relationship). Such a mismatch is less likely to arise on the large footprint WWAN because the user's own provider is often within reach.

We take the position that the range-speed dichotomy between WLANs and WWANs can be bridged by using both kinds of networks in *combination*. Specifically, nodes in close vicinity band together their WWAN links, with the high-speed WLAN serving as the glue, to boost the effective wide-area bandwidth available to the active nodes. While pooling together WWAN links has been considered in prior work (Section II), doing so among a set of collaborating but uncoordinated nodes raises several challenges that we believe have not been adequately addressed:

1) **Incentives:** Both monetary costs (e.g., WWAN charges) and energy costs should be accounted for as peers provide or seek help to or from each other.
2) **Collaboration protocol:** We need a speed- and energy-efficient protocol for nodes to discover each other and to share their bandwidth resources effectively.
3) **Security:** Collaborative downloading should not compromise node security or privacy.
4) **User interface:** While the system could operate autonomously for the most part, we need a simple and intuitive interface to enable users to exercise control when they so wish.

Here, we discuss these challenges and argue the pros and cons of various design alternatives in the context of *COMBINE*, a collaborative downloading system we are designing and prototyping. We believe that collaborative downloading is not only beneficial to users but is also likely to be of interest to WWAN service providers, since it allows them to boost the effective download speeds experienced by their users without requiring expensive infrastructural upgrades.

II. RELATED WORK

Prior work has explored the idea of aggregating bandwidth across multiple WWAN links. Some of it has focused on mechanisms for inverse multiplexing (i.e., striping packets) across multiple WWAN links to avoid problems such as TCP packet reordering (e.g., [11], Horde [8]). These systems assume that the multiple WWAN links are attached to the same device, so the issues of how to accomplish local communication and provide incentives for cooperation are not considered. Other systems have considered sharing WWAN links across devices. However, these have again side-stepped the issue of incentives and simplified the local communication problem by assuming that the same user owns all devices (e.g., MOPED [2]), the presence of a separate aggregation router (e.g., MARS [9]), or that the improved performance alone is sufficient incentive for cooperation (e.g., Handheld Routers [10], PRISM [6]). However, the mere promise of improved performance may not be a sufficient incentive for cooperation because of the ephemeral association between mobile nodes (see Section IV). While UCAN [7] considers secure crediting, it uses the WLAN to increase the reach of the WWAN rather than for bandwidth aggregation, and more importantly, it ignores a key resource, viz. energy. Finally, all of the above systems operate at the network or transport layer. While this offers the advantage of application independence, it also requires infrastructure proxies to split and splice flows, which may be a deployment hurdle.

III. ESTIMATING COST

We consider a setting where a requester (termed *initiator*) seeks to utilize the WWAN links of one or more *collaborators*. Since collaborator nodes contribute network and computational resources, we must provide incentives to offset their cost of using these resources. We envisage a dynamic market where nodes sell their (unused) WWAN bandwidth resources in return for compensation from the initiator. This implies that, at minimum, each collaborator must be able to calculate its cost and communicate it as its price to the initiator. In turn, the initiator must be able to compare the prices offered by multiple collaborators. It is important for the cost to be modeled appropriately, so that the offered price is high enough to adequately compensate the collaborator but not so high as to be unattractive to the initiator.

There are two principal costs that a collaborator incurs in helping the initiator. The first is the cost of transferring data on the WWAN link for which the WWAN service provider extracts a fee. Given a tariff structure, one can estimate the cost of WWAN usage for transferring a given amount of data, although some projection of anticipated future usage may be needed (say, based on past usage patterns) to account for tiered pricing.

The second cost that we must account for is the battery drain on the collaborator. We believe it is vital to account for battery drain, because it is a meager resource on the nodes and there may be significant opportunity cost in squandering it on helping others. Intuitively, as the battery energy remaining decreases, the more valuable it becomes and the cost of collaboration increases.

The (incremental) battery drain on the collaborator corresponds to transferring data from the source over the WWAN and relaying it on to the initiator over the WLAN. Using experimental data, each collaborating node can, *a priori*, calculate the amount of battery required for a given amount of data transfer. (Our measurements on an iMate Pocket PC equipped with WiFi and GPRS interfaces confirm that the battery drain is roughly linear in the amount of data transferred in a sustained burst.) Knowing the amount of battery it has at the beginning of the transfer and the amount of data to be downloaded on behalf of the initiator, a node can calculate the fraction of the battery that will remain at the end of the proposed collaboration.

Thus, our initial plan is to have the collaborator estimate a cost that is directly proportional to the tariff imposed by the service provider and inversely proportional to the estimated fraction of battery remaining at the end of the transfer. Note that this cost has the same units as that imposed by the service provider (typically monetary units). We believe that this simple model adequately captures our basic intuition about the cost of using a scarce resource.

IV. ACCOUNTING

For an incentive scheme based on the cost model from Section III to work, we need an accounting mechanism to keep track of payments made by initiators to the collaborators they recruit. There are several challenges that a practical accounting scheme must address:

1) *Ephemeral association:* Given the mobility of nodes and the ephemeral association between them, a collaborator who provides help cannot count on the initiator being available to return the favor at a different time and place. So the accounting mechanism should be able to store credits for future use; a real-time tit-for-tat scheme as in BitTorrent would not suffice.
2) *Cheat-Proof:* The accounting scheme should be able to limit the damage caused by an initiator who fails to provide the promised compensation to a collaborator (e.g., by using counterfeit money for payment) or a collaborator who fails to provide the promised help.
3) *Privacy:* The accounting scheme should not leak information on the identities of the collaborator and the initiator to each other, or to a third party.
4) *Efficiency:* The computational and communication overhead of accounting should be low.

There are various alternatives for designing the accounting system. Electronic funds transfer (EFT), which is widely supported by banks, is a possibility. However, EFT is cumbersome in practice (e.g., the collaborator would have to share its bank account information with the initiator). It also imposes a significant overhead on the collaborator (especially when fine-grained payments are made) in confirming the EFT with its bank online, to prevent cheating by the initiator.

An alternative would be to use digital cash (e.g., [1]), possibly adapted for efficient micropayments (e.g., [3]). The main advantage of such systems is their privacy, which is equivalent to that of paper cash. However, double spending is a serious risk, preventing which requires either the overhead of online communication with the bank that issued the cash or the provision of a secure hardware device (e.g., smartcard) at the clients to detect or prevent duplication.

In *COMBINE*, we are considering an alternative, credit-based scheme that leverages a central authority (e.g., a WWAN provider or the purveyor of *COMBINE* software) to certify a fixed number of public/private key pairs for each user. When a payment needs to be made, the initiator signs an IOU for the appropriate amount, also embedding a nonce provided by the collaborator. The collaborator can satisfy itself of the authenticity and the freshness of the IOU without needing to communicate with the central authority. Such communication is only needed once in a while to redeem the IOUs. This credit-based scheme offers the advantage of efficient and duplication-resistant micropayments without requiring online WAN communication. Furthermore, it allows us to use arbitrary virtual money (e.g., energy credits) instead of real money. However, compared to digital cash schemes, we trade off some privacy, in particular allowing the authority to link the IOUs to the payer. We believe that this may be acceptable, especially if we can effectively obfuscate the identity of the payee using a forwarding chain (Section VI).

The assumed identity infrastructure also enables the construction of a reputation system to flag collaborators who fail to provide the promised service in return for an IOU.

V. PROTOCOL DESIGN

A system that supports collaborative download in the setting we envisage requires attention to three components: a protocol for mobile devices to form groups, a scheme to distribute work amongst the group, and a mechanism for low-level data transport and connection management to fetch data from servers, which are oblivious of the collaboration.

A. Group Formation Protocol

Group formation is the process by which the initiator identifies a set of collaborators. This is clearly a critical first step in doing collaborative downloads, but to the best of our knowledge, prior work has largely ignored it.

Group formation in the absence of Wi-Fi access points is tricky because mobile devices tend to switch off their 802.11 cards or put them in a power saving mode to conserve battery. Thus, the protocol cannot depend on WLAN cards being switched on in anticipation of group formation.

Furthermore, group formation must work correctly in an environment where mobile devices move in and out of range: it must tolerate non-responsive initiators and collaborators, as well as multiple simultaneous initiators.

The *COMBINE* prototype implements a simple robust and *energy efficient* protocol by letting collaborators put their WLAN cards in a new power saving mode called the *Waiting* mode. Collaborators in this mode wake up periodically, like in the standard 802.11 Power Saving Mode [5] (although at a much lower frequency), and broadcast an *I-am-Alive* message. This message contains a bid reflecting the the price of using this node as a collaborator (see Section III) and the bandwidth it is willing to offer. The initiator collects all the bids and selects the list of nodes with prices less than a certain threshold to be its collaborators.

Even the *Waiting* mode might be wasteful of energy if there are very few collaborative downloads happening. We are exploring the use of low-power radios (e.g., Bluetooth) to serve as a signaling channel for group formation. Thus a nodes would switch on its 802.11 radio only after it has been being enlisted as a collaborator. Our experiments show that the battery life of a mobile device with its Bluetooth radio turned on continuously is comparable to its standby life.

B. Work Distribution

We view work distribution as a *policy* decision that relies on a lower level *mechanism* to actually effect the transfer the data. We are considering multiple policies for work distribution. A simple one, whose performance is illustrated in this paper, uses a work-queue.

The initiator determines the size of the file to be downloaded from the server, partitions it into fixed chunks, and appends them to a work-queue, which the collaborators query. Collaborators pick up more work when they are done with their current work item; thus we dynamically adjust to a collaborator's speed variations without requiring perfect knowledge of its future speed. In addition to varying link characteristics, a collaborator's speed could also vary due of increased local network activity. COMBINE continuously monitors the amount of local activity in a device and backs off when that increases beyond a threshold.

C. Data Transfer and Connection Management

One technique to fetch data from an oblivious server is to interpose a special-purpose proxy to multiplex traffic from collaborators over a single connection to the server, and vice versa. This approach has been explored in prior work; however, we believe that the need for a special-purpose proxy can be a significant detriment against adoption because of the need to engineer it to handle the potentially heavy data path workload of collaborative downloads. Our goal is to make collaboration transparent to servers and place minimal demands on the infrastructure, albeit at the cost of slightly increased complexity on the mobiles. Our strategy is to use an HTTP level solution, but this raises at least two challenges.

First, if the server does not support byte-range HTTP requests, it is difficult to implement a solution. Second, if access to server content requires session establishment or authentication by the client, it is more difficult to have an untrusted collaborator act on behalf of an initiator.

The current prototype assumes that servers have byte-range support and they don't need session level information. We are investigating a different data transport mechanism, which relaxes these assumptions by treating collaborators as multi-homed HTTP proxies.

D. *Performance*

We implemented our prototype in a community consisting of up to five members, each of which is a Pentium 4 laptop with 1GB of RAM, running Windows XP SP2. The laptops use a PCMCIA Sierra Atlantic GPRS card as their WWAN NIC and a D-Link WG132A USB 2.0 802.11b dongle as their WWAN NIC. We measured throughput by transferring an 8MB file from a server on the Internet to one of the laptops, using HTTP byte-range requests to fetch parts of the file [4]. Figure 1(a) shows the measured speed up as well as the ideal speed up. Measured speed up includes the overhead of forming a group, while the ideal speed up ignores this cost.

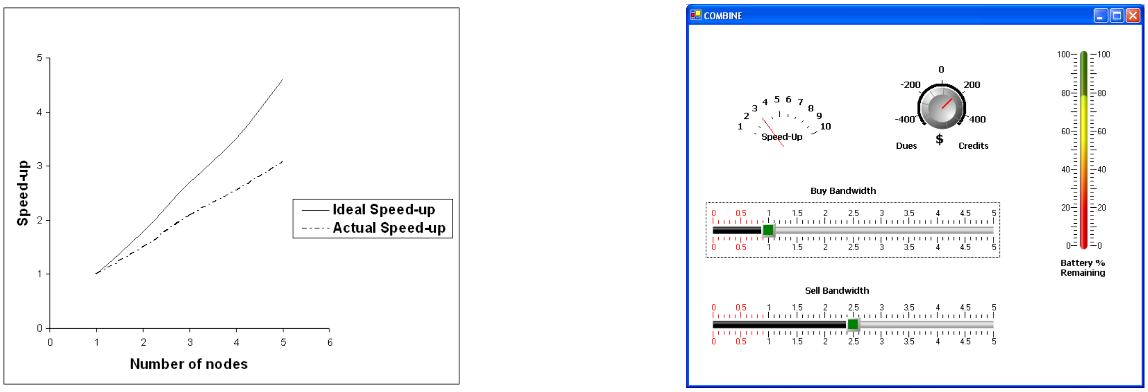

Fig. 1. **(a) Speedup versus collaboration group size, and (b) a mockup of the *COMBINE* user interface.**

VI. Security

While collaborative downloading offers a significant performance benefit, it also raises several security issues. We briefly discuss these and our approach to addressing them in *COMBINE*.

First, there is the issue of *privacy*, with regard to leaking information on a user's activity to collaborators and/or the accounting system. For example, a collaborator might learn the URL accessed by the initiator. However, the initiator can still effectively remain anonymous, since the certificate presented with an IOU in Section IV need not identfy the user it was issued to. Furthermore, by issuing multiple certificates to each user, any of which can be used in an IOU, we make it harder to track the initiator across multiple collaborative download sessions.

The central authority where client's redeem their IOUs can, of course, learn the identity of the issuer of each IOU. However, to prevent it from also identifying the payee (and thereby learning that the payer and the payee were likely in each other's vicinity), we are investigating the use of a random forwarding chain over the WAN, wherein the IOU is eventually redeemed by a node that has nothing to do with its original issuer.

Second, there is the issue of *confidentiality*. For example, the initiator might be downloading a music file that is only available to subscribers. While *COMBINE* by itself does not include any mechanism to maintain confidentiality, existing end-to-end encryption, if any, would help (e.g., SSL would shut out the collaborators just as it shuts out any other web proxy).

Finally, there is the issue of ensuring the *authenticity* of the content downloaded through the collaborators. For example, a collaborator could cheat by returning bogus content instead of expending WWAN bandwidth to download the real content, but still collect a payment from the initiator. Addressing this problem requires a combination of certification of content blocks by the source (as systems such as BitTorrent would need) and a reputation system to blacklist cheaters.

VII. USER INTERFACE

It is desirable for *COMBINE* to operate autonomously, without requiring user input on an ongoing basis. However, since users may expend valuable resources and/or accrue monetary charges, we would still want to give them the *option* of exercising control. We present here our initial ideas on the design of a simple user interface for *COMBINE*.

Figure 1(b) shows a mock-up of the *COMBINE* UI. It comprises elements that inform the user and those that allow the user to exercise control. In the former category are "dials" that show users (a) how much credit/dues they have accrued, (b) the speedup obtained by using *COMBINE*, and (c) the amount of resources expended (or remaining), in particular battery power. Users can view the information presented to drive how they control the operation of *COMBINE*.

To enable user control, the UI includes two sliders, one each to control the aggressiveness of selling and buying resources, respectively. The slider settings are translated to numerical factors, K_s and K_b, to control the buying and setting of resources, respectively. These factors are initially set to a neutral value (say 1) to ensure reasonable operation by default. The K_s factor is used to scale down or up the price computed in Section III, to make a collaborator more or less willing sell its resources. At the extreme, K_s is set to infinity, which means that the seller is unwilling to sell for any price. The factor K_b operates analogously, with the base price that the buyer (viz., the initiator) is willing to pay pegged to the cost it would incur (in terms of bandwidth and battery) were it to do the download by itself. At the extreme, K_b is set to zero, which means that the initiator is unwilling to pay any price, effectively opting out of collaborative downloading.

We could also overlay demand/bid information from the neighborhood on the sliders so that the user knows how aggressive they would need to be to make a deal. This could be the basis of a game for (idle) users looking to earn some extra money by parlaying their unused resources.

VIII. CONCLUSION

Collaborative downloading offers the potential of significant performance gains by utilizing WLAN and WWAN links in combination. However, as we have discussed in the context of *COMBINE*, a number of challenges need to be addressed in realizing this potential in practice.

REFERENCES

[1] ECash. *http//www.ecash.com*.

[2] C. Carter and R. Kravets. User Devices Cooperating to Support Resource Aggregation. In *IEEE WMCSA*, June 2002.

[3] S. Glassman et al. The Millicent Protocol for Inexpensive Electronic Commerce. In *Fourth WWW Conference*, Dec. 1995.

[4] Hypertext transfer protocol – HTTP/1.1: RFC 2616. http://www.w3.org/Protocols/rfc2616/rfc261 6.html.

[5] IEEE Std 802.11: Wireless LAN Medium Access Control and Physical Layer Specifications. IEEE Computer Society LAN MAN Standards Committee., Aug. 1999.

[6] K.-H. Kim and K. G. Shin. Improving TCP Performance over Wireless Networks with Collaborative Multi-homed Mobile Hosts. In *ACM/Usenix MobiSys*, June 2005.

[7] H. Luo et al. UCAN: A Unified Cellular and Ad-hoc Network Architecture. In *ACM MobiCom*, Sept. 2003.

[8] A. Qureshi et al. Horde: Separating Network Striping Policy from Mechanism. In *ACM/Usenix MobiSys*, June 2005.

[9] P. Rodriguez et al. MAR: A Commuter Router Infrastructure for the Mobile Internet. In *ACM/Usenix MobiSys*, June 2004.

[10] P. Sharma, S.-J. Lee, J. Brassil, and K. G. Shin. Handheld Routers: Intelligent Bandwidth Aggregation for Mobile Collaborative Communities. In *IEEE BROADNETS*, Oct. 2004.

[11] A. C. Snoeren. Adaptive Inverse Multiplexing for Wide-Area Wireless Networks. In *IEEE Global Internet*, Dec. 1999.