Cross-Layer Design: A Survey and the Road Ahead

Vineet Srivastava, Institute for Infocomm Research and National University of Singapore
Mehul Motani, National University of Singapore

ABSTRACT

Of late, there has been an avalanche of cross-layer design proposals for wireless networks. A number of researchers have looked at specific aspects of network performance and, approaching cross-layer design via their interpretation of what it implies, have presented several cross-layer design proposals. These proposals involve different layers of the protocol stack, and address both cellular and ad hoc networks. There has also been work relating to the implementation of cross-layer interactions. It is high time that these various individual efforts be put into perspective and a more holistic view be taken. In this article, we take a step in that direction by presenting a survey of the literature in the area of cross-layer design, and by taking stock of the ongoing work. We suggest a definition for cross-layer design, discuss the basic types of cross-layer design with examples drawn from the literature, and categorize the initial proposals on how cross-layer interactions may be implemented. We then highlight some open challenges and new opportunities for cross-layer design. Designers presenting cross-layer design proposals can start addressing these as they move ahead.

INTRODUCTION

As wireless communications and networking fast occupy center stage in research and development activity in the area of communication networks, the suitability of one of the foundations of networking, the layered protocol architecture, is coming under close scrutiny from the research community. It is repeatedly argued that although layered architectures have served well for wired networks, they are not suitable for wireless networks. To illustrate this point, researchers usually present what they call a cross-layer design proposal. Thus, there have been a large number of cross-layer design proposals in the literature recently. Generally speaking, cross-layer design refers to protocol design done by actively exploiting the dependence between protocol layers to obtain performance gains. This is unlike layering, where the protocols at the different layers are designed independently.

If one looks at the literature in the area of cross-layer design, several observations can be made. First, there are several interpretations of cross-layer design. This is probably because the cross-layer design effort has been made rather independently by researchers from different backgrounds, who work on different layers of the stack. Second, while there are many cross-layer design proposals in the literature, including those that build on top of other cross-layer design proposals, some more fundamental issues (coexistence of different cross-layer design proposals, when cross-layer design proposals should be invoked, what roles the layers should play, etc.) are not addressed directly. Third, the synergy between the performance viewpoint and implementation concerns is weak; most proposals focus and elaborate on the performance gains from cross-layer design, although there are some ideas on how cross-layer interactions may be implemented. Finally, the wireless medium allows richer modalities of communication than wired networks. For example, nodes can make use of the inherent broadcast nature of the wireless medium and cooperate with each other. Employing modalities like node cooperation in protocol design also calls for cross-layer design. Such cross-layer design ideas are only beginning to be discussed in the literature.

The motivation for this article came from the aforementioned state of the literature. In this article we have two aims. The first is to present a brief survey of the literature. In doing so, we take stock of ongoing work in the area of cross-layer design, put that work in perspective, and consolidate the existing results and insights. Our second aim is to identify the road ahead. We do so by raising some questions and issues related to cross-layer design that, in our opinion, are not getting sufficient attention in the literature today, and researchers may want to address as they move forward.

We start by suggesting a definition for cross-layer design. Although our definition is simple and arguably obvious, it serves to unify the different interpretations of cross-layer design in the literature. After presenting the definition, we identify some basic types of cross-layer designs and present relevant examples from the literature. This serves four purposes: first, it clarifies

 0163-6804/05/$20.00 © 2005 IEEE

and illustrates our definition of cross-layer design; second, it creates a taxonomy for classifying existing cross-layer design proposals; third, it highlights the different interpretations of cross-layer design in the literature and shows how they can be seen in a more unified way; and finally, it provides a framework for evaluating the implementation concerns raised by different kinds of cross-layer design proposals. As expected, the different kinds of cross-layer design proposals raise different implementation concerns. After creating the taxonomy of the cross-layer design proposals, we similarly categorize and discuss the initial proposals for implementing cross-layer interactions and highlight briefly for which kind of cross-layer design proposals the different implementation methods are suitable. This completes the survey part of the article.

Moving on to the second aim of the article, which is to point out the road ahead, we spell out the open challenges in cross-layer design. We highlight some issues we believe, if addressed, will make the current activity more complete and holistic. We also raise some ideas that could be new opportunities for cross-layer design. Researchers making cross-layer design proposals may want to address these issues as they move forward.

Recently, authors in [1] presented what they called a "cautionary" perspective on cross-layer design. They highlighted the importance of architecture and discussed the architectural problems that cross-layer design, if done without care, can create; they also warned designers about the possibility of inadvertent performance losses due to interaction between conflicting cross-layer design proposals — hence the term "cautionary." Our work in this article complements the work in [1]. Specifically, the definition and taxonomy of cross-layer design proposals allows a clearer appreciation of the cautionary perspective in [1]. Additionally, we raise some open challenges and new opportunities for cross-layer design, as mentioned above.

Understanding Cross-Layer Design

A Definition of Cross-Layer Design

A layered architecture, like the seven-layer open systems interconnect (OSI) model [2, p. 20], divides the overall networking task into layers and defines a hierarchy of services to be provided by the individual layers. The services at the layers are realized by designing protocols for the different layers. The architecture forbids direct communication between nonadjacent layers; communication between adjacent layers is limited to procedure calls and responses.

In the framework of a reference layered architecture, the designer has two choices at the time of protocol design. Protocols can be designed by respecting the rules of the reference architecture. In a layered architecture, this would mean designing protocols such that a higher-layer protocol only makes use of the services at the lower layers and is not concerned about the details of how the service is being provided. Following the architecture also implies that protocols would not need any interfaces not present in the reference architecture.

Alternatively, protocols can be designed by violating the reference architecture, for example, by allowing direct communication between protocols at nonadjacent layers or sharing variables between layers. Such violation of a layered architecture is cross-layer design with respect to the reference architecture.

Definition 1: Protocol design by the violation of a reference layered communication architecture is cross-layer design with respect to the particular layered architecture.

Comment 1: Examples of violation of a layered architecture include creating new interfaces between layers, redefining the layer boundaries, designing protocol at a layer based on the details of how another layer is designed, joint tuning of parameters across layers, and so on.

Comment 2: Violation of a layered architecture involves giving up the luxury of designing protocols at different layers independently. Protocols so designed impose some conditions on the processing at other layer(s).

Comment 3: Cross-layer design is defined as a protocol design methodology. However, a protocol designed with this methodology is also termed cross-layer design.

For exposition, consider a hypothetical three-layer model with the layers denoted L_1, L_2, and L_3 — L_1 is the lowest layer, L_3 the highest. Note that in such an architecture, there is no interface between L_3 and L_1. One could, however, design an L_3 protocol that needs L_1 to pass a parameter to L_3 at runtime. This calls for a new interface, and hence violates the architecture. Alternatively, one could view L_2 and L_1 as a single layer, and design a joint protocol for this "super layer." Or one could design the protocol at L_3, keeping in mind the processing being done at L_1, again giving up the luxury of designing the protocols at the different layers independently. All these are examples of cross-layer design with respect to the three-layer architecture in question.

Architecture violations, like those introduced by cross-layer design, clearly undermine the significance of the architecture since the architecture no longer represents the actual system. If many architecture violations accumulate over time, the original architecture can completely lose its meaning. Architecture violations can have a detrimental impact on system longevity, as has been argued for the case of cross-layer design in [1].

If many architecture violations accumulate over time, the original architecture can completely lose its meaning. Architecture violations can have a detrimental impact on system longevity, as has been argued for the case of cross-layer design.

General Motivation for Cross-layer Design

Why does the presence of wireless links in the network motivate designers to violate the layered architectures? There are three main reasons: the unique problems created by wireless links, the possibility of opportunistic communication on wireless links, and the new modalities of communication offered by the wireless medium.

On the pessimistic side, wireless links create several new problems for protocol design that cannot be handled well in the framework of the layered architectures. The classic case of a TCP

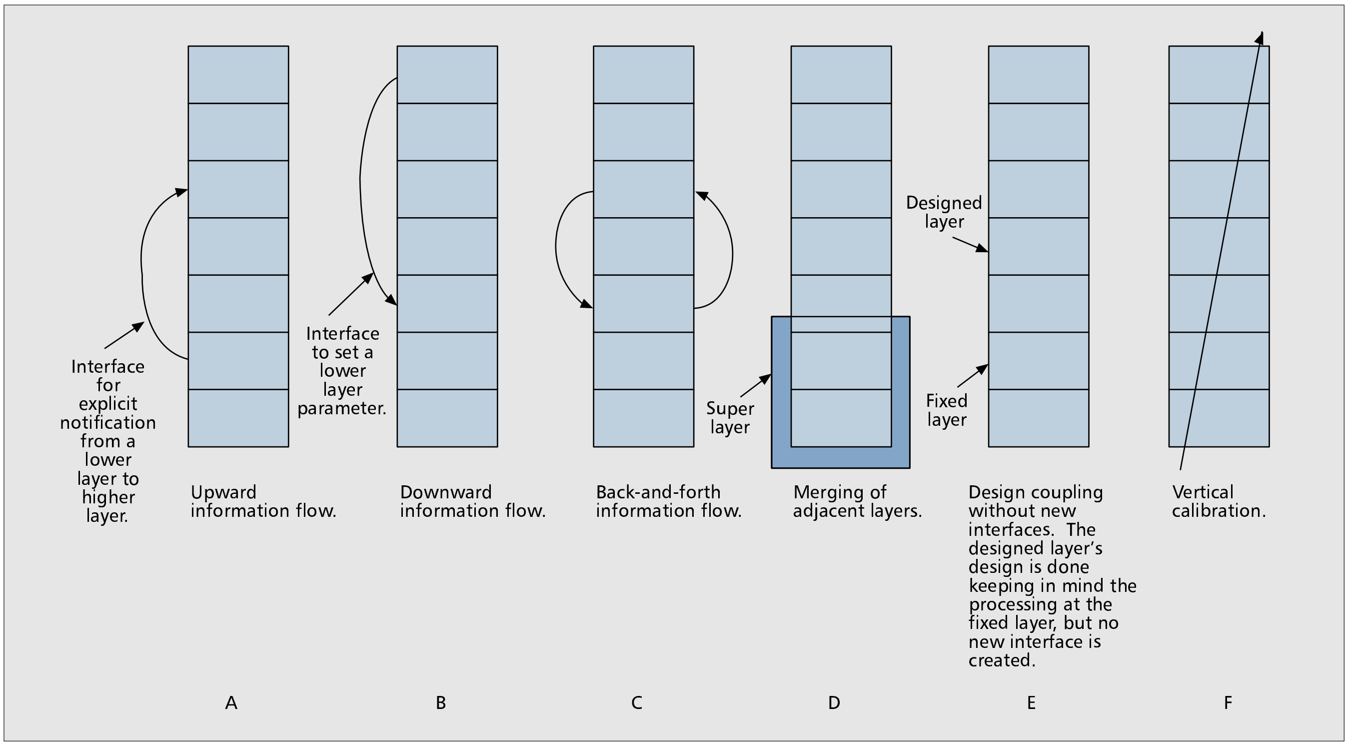

Figure 1. *Illustrating the different kinds of cross-layer design proposals. The rectangular boxes represent the protocol layers.*

sender mistaking a packet error on a wireless link to be an indicator of network congestion is an example [3]. On the optimistic side, wireless networks offer several avenues for opportunistic communication that cannot be exploited sufficiently in a strictly layered design. For instance, the time-varying link quality allows opportunistic usage of the channel [4, references therein], whereby the transmission parameters can be dynamically adjusted according to the variations in the channel quality, just to name one example. Additionally, the wireless medium offers some new modalities of communication the layered architectures do not accommodate. For instance, the physical layer can be made capable of receiving multiple packets [5] at the same time. The nodes can also make use of the broadcast nature of the channel and cooperate with one another in involved ways. Making use of such "novel" modes of communication in protocol design also requires violating the layered architectures.

A Snapshot of Cross-Layer Design Proposals

As mentioned above, there are many cross-layer design proposals in the literature. The authors in [6] present a survey of several cross-layer design proposals from the literature based on the layers that are coupled. Here, we are more interested in *how* the layers are coupled, in other words, what kind of architecture violation has taken place in a particular cross-layer design.

We note that the layered architecture can be violated in the following basic ways:

- Creation of new interfaces (Figs. 1a–c)
- Merging of adjacent layers (Fig. 1d)
- Design coupling without new interfaces (Fig. 1e)
- Vertical calibration across layers (Fig. 1f)

We find that most cross-layer design proposals in the literature fit into one of these basic categories. We shall now discuss the aforementioned four categories in more detail and point out some relevant examples. A few points are worth mentioning here. First, the examples we point out are meant to be representative, not exhaustive. Second, the architectural violations we identify can be combined to yield more complex cross-layer designs. Finally, the reference layered architecture we assume is a five-layer model, with the application layer, transport layer, network layer, link layer (comprising the data link control [DLC] and medium access control [MAC] sublayers) [2, p. 24], and physical layer; we assume that all the layers perform their generally understood functionalities.

Creation of New Interfaces

Several cross-layer designs require creation of new interfaces between the layers. The new interfaces are used for information sharing between the layers at runtime. The architecture violation here is obviously the creation of a new interface not available in the layered architecture. We further divide this category into three subcategories depending on the direction of information flow along the new interfaces:

- Upward: From lower layer(s) to a higher layer
- Downward: From higher layer(s) to a lower layer
- Back and forth: Iterative flow between two layers

In the literature one can find examples of all three subcategories. We discuss these now.

Upward Information Flow — A higher-layer protocol that requires some information from the lower layer(s) at runtime results in the creation of a new interface from the lower layer(s) to the higher layer, as shown in Fig. 1a. For instance, if the end-to-end TCP path contains a wireless link, errors on the wireless link can trick the TCP sender into making erroneous inferences about the congestion in the network, and as a result the performance deteriorates. Creating interfaces from the lower layers to the transport layer to enable explicit notifications alleviates such situations. For example, the explicit congestion notification (ECN) from the router to the transport layer at the TCP sender can explicitly tell the TCP sender if there is congestion in the network to enable it to differentiate between errors on the wireless link and network congestion [3].

Examples of similar upward information flow are also seen in the literature at the MAC layer (link layer in general) in form of channel-adaptive modulation or link adaptation schemes [4, references therein]. The idea is to adapt the parameters of the transmission (e.g., power, modulation, code rate) in response to the channel condition, which is made known to the MAC layer (link layer) by an interface from the physical layer.

It is interesting to compare and contrast cross-layer design proposals that rely on upward flow of information to what can be called self-adaptation loops at a layer. By a self-adaptation loop, we mean an adaptive higher-layer protocol that responds to events which, within the constraints of the layered architecture, are directly observable at the layer itself. Hence, self-adaptation loops do not require new interfaces to be created from the lower layer(s) to the higher layer, and cannot be classified as cross-layer designs. For example, consider the auto-rate fallback mechanism for rate selection in wireless devices with multirate physical layers. The idea is that if some number of packets sent at a particular rate are successfully delivered, the data rate is increased, whereas if a packet failure is experienced, the data rate drops. In this case the MAC layer rate selection mechanism responds to the acknowledgments, which are directly observable at the MAC layer. Hence, auto-rate fallback is not a cross-layer design.

Downward Information Flow — Some cross-layer design proposals rely on setting parameters on the lower layer of the stack at runtime using a direct interface from some higher layer, as illustrated in Fig. 1b. As an example, applications can inform the link layer about their delay requirements, and the link layer can then treat packets from delay-sensitive applications with priority [7].

A good way to look at the upward and downward information flow is to treat them as notifications and hints, respectively, as proposed in [8]. Upward information flow serves the purpose of notifying the higher layers about the underlying network conditions; downward information flow is meant to provide hints to the lower layers about how the application data should be processed.

Back and Forth Information Flow — Two layers, performing different tasks, can collaborate with each other at runtime. Often, this manifests in an iterative loop between the two layers, with information flowing back and forth between them as highlighted in Fig. 1c. Clearly, the architecture violation here is the two complimentary new interfaces.

As an example, we refer to the network-assisted diversity multiple access (NDMA) proposal [9, references therein], whereby the physical (PHY) and MAC layers collaborate in collision resolution in the uplink of a wireless LAN system. Basically, with improvements in the signal processing at the PHY, it becomes capable of recovering packets from collisions. Thus, upon detecting a collision the base station first estimates the number of users that have collided, and then requests a suitable number of retransmissions from the set of colliding users. Then PHY signal processing lets the base station separate the signals from all the colliding users.

A similar back and forth information flow between layers is seen in proposals performing joint scheduling and power control in wireless ad hoc networks; see [10] as an example.

MERGING OF ADJACENT LAYERS

Another way to do cross-layer design is to design two or more adjacent layers together such that the service provided by the new *superlayer* is the union of the services provided by the constituent layers. This does not require any new interfaces to be created in the stack. Architecturally speaking, the superlayer can be interfaced with the rest of the stack using the interfaces that already exist in the original architecture.

Although we have not come across any cross-layer design proposal that explicitly creates a superlayer, it is interesting to note that the collaborative design between the PHY and MAC layers (discussed earlier with the NDMA idea) tends to blur the boundary between these two adjacent layers.

Although we have not come across any cross-layer design proposal that explicitly creates a superlayer, it is interesting to note that the collaborative design between the PHY and MAC layers tends to blur the boundary between these two adjacent layers.

DESIGN COUPLING WITHOUT NEW INTERFACES

Another category of cross-layer design involves coupling two or more layers at design time without creating any extra interfaces for information sharing at runtime. We illustrate this in Fig. 1e. While no new interfaces are created, the architectural cost here is that it may not be possible to replace one layer without making corresponding changes to another layer.

For instance, [5] considers the design of a MAC layer for the uplink of a wireless LAN when the PHY is capable of providing multipacket reception capability. Multipacket reception capability implies that the PHY is capable of receiving more than one packet at the same time. Notice that this capability at the physical layer considerably changes the role of the MAC layer; thus, it needs to be redesigned.

VERTICAL CALIBRATION ACROSS LAYERS

The final category in which cross-layer design proposals in the literature fit is what we call vertical calibration across layers. As the name suggests, this refers to adjusting parameters that

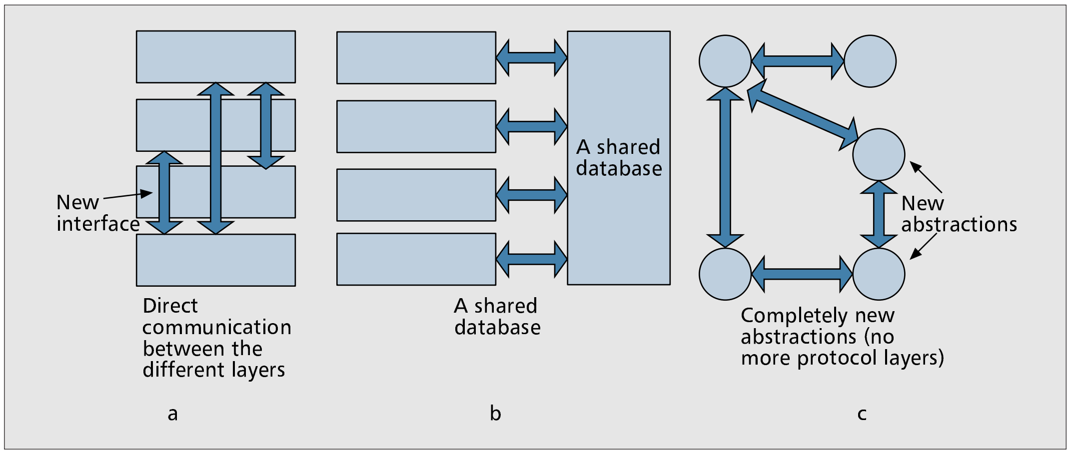

Figure 2. *Proposals for architectural blueprints for wireless communications.*

span across layers, as illustrated in Fig. 1f. The motivation is easy to understand. Basically, the performance seen at the level of the application is a function of the parameters at all the layers below it. Hence, it is conceivable that joint tuning can help to achieve better performance than individual settings of parameters — as would happen had the protocols been designed independently — can achieve.

As an example, [11] presents an example of vertical calibration where the delay requirement dictates the persistence of link-layer automatic repeat request (ARQ), which in turn becomes an input for deciding the rate selection through a channel-adaptive modulation scheme.

Vertical calibration can be done in a static manner, which means setting parameters across the layers at design time with the optimization of some metric in mind. It can also be done dynamically at runtime, which emulates a flexible protocol stack that responds to variations in the channel, traffic, and overall network conditions. Static vertical calibration does not create significant consideration for implementations since the parameters can be adjusted once at design time and left untouched thereafter. Dynamic vertical calibration, on the other hand, requires mechanisms to retrieve and update the values of the parameters being optimized from the different layers. This may incur significant cost in terms of overheads, and also impose strict requirements on the parameter retrieval and update process to make sure that the knowledge of state of the stack is current and accurate.

Proposals for Implementing Cross-Layer Interactions

Alongside the cross-layer design proposals discussed earlier, initial proposals on how cross-layer interactions can be implemented are also being made in the literature. These can be put into three categories:

- Direct communication between layers (Fig. 1a)
- A shared database across the layers (Fig. 1b)
- Completely new abstractions (Fig. 1c)

Direct Communication between Layers

A straightforward way to allow runtime information sharing between layers is to allow them to communicate with each other, as depicted schematically in Fig. 1a. Note that this is applicable when there has to be runtime information sharing between layers (e.g., in cross-layer designs that rely on new interfaces or in dynamic vertical calibrations). Practically speaking, direct communication between the layers means making the variables at one layer visible to the other layers at runtime. By contrast, under a strictly layered architecture, every layer manages its own variables, and its variables are of no concern to other layers.

There are many ways in which the layers can communicate with one another. For instance, protocol headers may be used to allow flow of information between layers. Alternatively, extra interlayer information could be treated as internal packets. The work in [12] presents a comparative study of several such proposals and goes on to present another such proposal, cross-layer signaling shortcuts (CLASS). CLASS allows any two layers to communicate directly with one another.

These proposals are appealing where just a few cross-layer information exchanges are to be implemented in systems that were originally designed in conformance with layered architectures. In that case, one can conceivably "punch" a few holes in the stack while still keeping it tractable. However, in general, when variables and internal states from different layers are to be shared as prescribed by such proposals, a number of implementation issues relating to managing shared memory spaces between layers may need to be resolved.

A Shared Database across Layers

The other class of proposals propose a common database that can be accessed by all layers, as illustrated in Fig. 1b (e.g., [6]). In one sense, the common database is like a new layer, providing the service of storage/retrieval of information to all the layers.

The shared database approach is particularly well suited to vertical calibrations across layers. An optimization program can interface with the different layers at once through the shared database. Similarly, new interfaces between the layers can also be realized through the shared database. The main challenge here is the design of the interactions between the different layers and the shared database.

COMPLETELY NEW ABSTRACTIONS

The third set of proposals present completely new abstractions, which we depict schematically in Fig. 1c. Consider, for example, the proposal in [13], which presents a new way to organize the protocols: in heaps, not in stacks as done by layering.

Such novel organizations of protocols are appealing as they allow rich interactions between the building blocks of the protocols. Hence, potentially they offer great flexibility, both during design as well as at runtime. However, they change the very way protocols have been organized, and hence may require completely new system-level implementations.

LOOKING FORWARD: OPEN CHALLENGES

Earlier we looked at the ongoing work in the area of cross-layer design. In doing so, we came face to face with several different interpretations of cross-layer design, looked at some representative cross-layer design proposals, and saw some initial ideas on how cross-layer interactions may be implemented. Having taken stock of the ongoing work, we now raise and discuss open challenges in cross-layer design. Broadly speaking, we include two kinds of issues in the open challenges: questions about cross-layer design that, in our opinion, are important but not getting sufficient attention in the literature; and some questions regarding the fundamental nature of the wireless medium, questions whose answers will influence how communication architectures for wireless networks should look like, and hence are pertinent to the cross-layer design effort. We note that some of these issues have been raised elsewhere in the literature. Our purpose here is to consolidate the different issues and discuss their significance with respect to the cross-layer design activity.

The following are some open challenges for designers proposing cross-layer design ideas:

- How do the different cross-layer design proposals coexist with one another?
- Will a given cross-layer design idea possibly stifle innovation in the future?
- What are the cross-layer designs that will have the most significant impact on network performance, and hence should be most closely focused on?
- Has a given design proposal been made with a thorough knowledge of the effect of the interactions between the parameters at different layers on network performance?
- Under which network and environmental condition would a particular cross-layer design proposal be invoked?
- Can the mechanisms/interfaces used to share information between the layers be standardized?
- What should the role of the physical layer in wireless networks be?
- Is the conventional view of the network, a collection of point-to-point links, appropriate for wireless networks?

We now look at some of these issues in greater detail.

IMPORTANT CROSS-LAYER COUPLINGS

While there are a number of cross-layer design proposals in the literature today, it is not clear which are the most important. To identify these, a thorough cost-benefit analysis of the different cross-layer design proposals in terms of implementation complexity vs. performance improvement is needed. Generally speaking, one can make the following inferences from the literature today: cross-layer design is needed between the network and MAC layers for ad hoc networks since the functionalities of the two layers interact [14]; explicit notifications by new interfaces to the transport layer improve end-to-end performance [3]; making use of channel knowledge at the MAC layer allows opportunistic usage of the channel and improves performance [14, references therein]; and energy, delay, and security related issues need to be handled across the layers in a holistic manner. It is time to move ahead from these general insights to specific holistic solutions. This requires comparative quantitative study of the different cross-layer design proposals, and is an open challenge for the community. Also relevant in this context is the question of coexistence of cross-layer design proposals, discussed next.

While there are a number of cross-layer design proposals in the literature today, it is not clear which are the most important ones. To identify these, a thorough cost-benefit analysis of the different cross-layer design proposals in terms of the implementation complexity versus the performance improvement is needed.

COEXISTENCE OF CROSS-LAYER DESIGN PROPOSALS

An important question to be answered is how different cross-layer design proposals can coexist with one another. To clarify by example, say the MAC layer in a stack responds to the variation in the channel by adjusting the data rate. The question is, will additionally adjusting the frame length at the link layer help further? How will an overriding control from, say, the transport layer, trying to control the link layer parameters, interact with these adaptation loops?

The question of coexistence of cross-layer design ideas is pertinent when it comes to determining whether some cross-layer design proposals can stifle further innovation. Let us say the physical and link layers are optimized for a certain performance metric in a cross-layer design scheme. If this scheme is deployed first, can other schemes that also rely on some (other) cross-layer couplings, or those that assume no coupling between the link and physical layers be deployed too at a later time?

Apart from presenting new cross-layer design proposals, designers need to start establishing which other cross-layer design interaction may or may not be employed together with their proposal. This has also been stressed in [1].

WHEN TO INVOKE A PARTICULAR CROSS-LAYER DESIGN

The network conditions in a wireless network are usually time-varying. In such a situation, one of the stated motivations behind cross-layer

These new modalities that the wireless medium offers cannot be accommodated into the layered architectures and hence inevitably require some degree of architecture violation, in other words cross-layer design. They represent the new opportunities for the cross-layer design effort.

design is to achieve the network equivalent of impedance matching [3]. The idea is to make the protocol stack responsive to variations in the underlying network conditions so that an optimal operating point is always maintained.

The pursuit of achieving such optimal operation throws up two complementary challenges. First, designers need to establish the network conditions under which the proposed cross-layer designs would result in performance improvements. Reference [1] presents an example to illustrate how a cross-layer design involving an iterative optimization of throughput and power leads to a loss in performance under a certain pathological network condition. The example in [1] underscores the need for designers to establish the network conditions under which their design proposals should and should not be used. Second, efficient mechanisms to make a timely and accurate assessment of the state of the network need to be built into the stack, and the corresponding overheads must be taken into account. This is also related to the question of interfaces between the modules, discussed next.

STANDARDIZATION OF INTERFACES

The one thing layering achieved was to present standardized boundaries and interfaces between modules of the system, the protocol layers. Now that the layered architecture is being violated in different ways, finding the new reference architecture becomes a challenge. What should the boundaries be between modules? Should we stick to traditional layer boundaries, as in Figs. 1a and b, and determine the new interfaces from there, or should we look at completely new boundaries, as in Fig. 1c? Or a combination? What should the interfaces between modules look like?

Addressing this challenge requires greater synergy between the performance viewpoint and implementation concerns than is seen in the literature today. Basically, the organization of the modules (layers or otherwise) and the interfaces between them determine how efficiently information can be shared between them, at what kinds of overheads and delays. This, in turn, determines how effective cross-layer design proposals that rely on sharing dynamic information between the modules can be. Hence, proposers of cross-layer design relying on back-and-forth information flow between layers or dynamic vertical calibrations need to start considering the impact of delays in the retrieval/updating of information on protocol performance. They also need to quantify the overheads associated with their cross-layer design proposals.

THE ROLE OF THE PHYSICAL LAYER

In wired networks the role of the physical layer has been rather small: sending and receiving packets when required to do so from the higher layers. As seen earlier, advances in signal processing at the physical layer can allow it to play a bigger role in wireless networks. This begs the question of how much of a role the physical layer should play. This is relevant to the cross-layer design effort because first, layered architectures like the OSI reference model do not allow much of a role for the physical layer besides providing a bit pipe, and second, enhancements in the physical layer will have to be balanced by corresponding changes to the higher layers. Hence, figuring out the role to be played by the physical layer is an important question. Cross-layer designs relying on advanced signal processing at the physical layer can be an interesting research ground for the future.

THE RIGHT COMMUNICATION MODEL

The last open challenge relates to the communication model assumed. Wired networks, by their very nature, are essentially a collection of well defined point-to-point communication links. The same cannot be said about wireless networks because the wireless medium is inherently broadcast, and there is no clearcut concept of a communication link in wireless networks. This gives rise to a fundamental question of whether it still makes sense to "create" links in a wireless network. Some recent work has made use of the inherent broadcast nature of the wireless medium to come up with innovative communication schemes for wireless networks. For example, exploiting the broadcast nature of the wireless medium can allow transmission schemes that rely on cooperation between the communication nodes (e.g., [15]).

These new modalities the wireless medium offers cannot be accommodated in layered architectures and hence inevitably require some degree of architecture violation, in other words, cross-layer design. They represent new opportunities for the cross-layer design effort. Hence, we believe that while proposing cross-layer design proposals to address the issues raised by wireless links, the designers should also keep an eye out for new opportunities created by wireless communication networks.

CONCLUSIONS

This article has taken stock of the current activity in the area of cross-layer design. After suggesting a definition, we bring the different interpretations of cross-layer design together and survey the ongoing work by creating a taxonomy of some representative cross-layer design proposals. We also look at the initial ideas for implementing cross-layer interactions. Next, we move on to highlight some open challenges in this area and discuss issues that, in our opinion, will make the ongoing cross-layer design work more holistic and complete. We also point out some new modalities of communication in wireless networks into which designers can tap. All in all, we have consolidated several scattered results and ideas in this area into a logical structure. By doing so, we have both summarized the current state of knowledge in this area as well as created a platform over which new research can be built.

REFERENCES

[1] V. Kawadia and P. R. Kumar, "A Cautionary Perspective on Cross Layer Design," *IEEE Wireless Commun.*, vol. 12, no. 1, Feb. 2005, pp. 3–11.

[2] D. Bertsekas and R. Gallager, *Data Networks*, 2nd ed., Prentice Hall, 1992.

[3] S. Shakkottai, T. S. Rappaport, and P. C. Karlsson, "Cross-Layer Design for Wireless Networks," *IEEE Commun. Mag.*, vol. 41, no. 10, Oct. 2003, pp. 74–80.

[4] Z. Ji *et al.*, "Exploiting Medium Access Diversity in Rate Adaptive Wireless LANs," *Proc. ACM Annual Int'l. Symp. Mobile Comp. and Net.*, Philadelphia, PA, Oct. 2004.
[5] L. Tong, V. Naware, and P. Venkitasubramaniam, "Signal Processing in Random Access," *IEEE Sig. Proc.*, vol. 21, no. 5, Sept. 2004, pp. 29–39.
[6] V. T. Raisinghani and S. Iyer, "Cross-Layer Design Optimizations in Wireless Protocol Stacks," *Comp. Commun.*, vol. 27, 2004, pp. 720–24.
[7] G. Xylomenos and G. C. Polyzos, "Quality of Service Support over Multi-service Wireless Internet Links," *Comp. Net.*, vol. 37, no. 5, 2001, pp. 601–15.
[8] L. Larzon, U. Bodin, and O. Schelen, "Hints and Notifications," *Proc. IEEE Wireless Commun. and Net. Conf.*, Orlando, FL, Mar. 2002.
[9] G. Dimic, N. D. Sidiropoulos, and R. Zhang, "Medium Access Control — Physical Cross-Layer Design," *IEEE Sig. Proc.*, vol. 21, no. 5, Sept. 2004, pp. 40–50.
[10] T. Elbatt and A. Ephremides, "Joint Scheduling and Power Control for Wireless Ad Hoc Networks," *IEEE Trans. Wireless Commun.*, vol. 3, no. 1, Jan. 2004, pp. 74–85.
[11] Q. Liu, S. Zhou, and G. B. Giannakis, "Cross-Layer Combining of Adaptive Modulation and Coding with Truncated ARQ Over Wireless Links," *IEEE Trans. Wireless Commun.*, vol. 3, no. 5, Sept. 2004, pp. 1746–55.
[12] Q. Wang and M. A. Abu-Rgheff, "Cross-Layer Signalling for Next-Generation Wireless Systems," *Proc. IEEE Wireless Commun. and Net. Conf.*, New Orleans, LA, Mar. 2003.
[13] R. Braden, T. Faber, and M. Handley, "From Protocol Stack to Protocol Heap — Role-Based Architecture," *Proc. Hot Topics in Net.*, Princeton, NJ, Mar. 2002.
[14] C. Barrett *et al.*, "Characterizing the Interaction Between Routing and MAC Protocols in Ad-Hoc Networks," *Proc. ACM Annual Int'l. Symp. Mobile Ad Hoc Net. and Comp.*, Lausanne, Switzerland, June 2002.
[15] B. Zhao and M. C. Valenti, "Practical Relay Networks: A Generalization of Hybrid-ARQ," *IEEE JSAC*, vol. 23, no. 1, Jan. 2005, pp. 7–18.

Biographies

VINEET SRIVASTAVA (vineet@i2r.a-star.edu.sg) is a research engineer in the Digital Wireless Department at the Institute for Infocomm Research, Singapore. His research interest is in the interaction between theoretical and practical aspects of signal processing, wireless communications, and networking. He holds an M.Eng. degree from the National University of Singapore. His M.Eng. research focused on cross-layer design, and his thesis, *Cross-Layer Design for Communication Systems*, was praised by the examiners for its scholarly content and lucid presentation.

MEHUL MOTANI [M] (motani@nus.edu.sg) is currently an assistant professor in the Electrical and Computer Engineering Department at the National University of Singapore. He graduated with a Ph.D. from Cornell University, focusing on information theory and coding for CDMA systems. Prior to his Ph.D., he was a member of technical staff at Lockheed Martin, Syracuse, New York, for over four years. Recently he has been working on research problems that sit at the boundary of communications and networking, including the design and analysis of wireless ad hoc and sensor networks. He was awarded the Intel Foundation Fellowship for work related to his Ph.D. in 2000 and nominated for the Best Teacher award at the National University of Singapore in 2003. He is on the organizing committees for ISIT 2006 and 2007, and has served on the technical program committees of several other conferences. He participates actively in IEEE and is currently Secretary of the IEEE Information Theory Society Board of Governors.

We believe that while proposing cross-layer design proposals to address the issues raised by wireless links, the designers should also keep an eye out for new opportunities being created by wireless communication networks.