NANDITA DUKKIPATI

High Performance Networking Group
Computer Systems Laboratory
353 Serra Mall, Room 356
Stanford, CA 94305-9030

Phone: (650) 387-5160
Fax: (650) 725-6949
nanditad@stanford.edu
http://yuba.stanford.edu/~nanditad

RESEARCH INTERESTS

The focus of my research is to make networks adaptable, evolvable, self-managed, and cost-efficient as well as improve their performance. To that end, my research interests lie in several areas of Networking including congestion control, routing, protocol design, router/switch architectures in both wired as well as wireless networks. I want to build and experiment with networking systems while making use of theoretical analysis where applicable. I am particularly enthusiastic in collaborating with disciplines outside of Networking.

EDUCATION

Stanford University — Stanford, CA
Ph.D. in Electrical Engineering — Expected June 2007
Thesis: Rate Control Protocol (RCP): Congestion Control to Make Flows Complete Quickly

Indian Institute of Science — Bangalore, India
M.S. in Electrical Engineering — August 2000
Thesis: Optimal Resource Allocation in Packet Networks

Birla Institute of Technology and Science — Pilani, India
B.E.(Hons) in Electronics and Instrumentation — June 1998
Thesis: PC-based Fiber Optic Sensing System to measure angular displacement

Ph.D. DISSERTATION

Title: “Rate Control Protocol (RCP): Congestion Control to Make Flows Complete Quickly”
Advisor: Professor Nick McKeown
Thesis Committee: Balaji Prabhakar, Mendel Rosenblum, Scott Shenker

Users typically want their flows to complete as quickly as possible. This makes Flow Completion Time (FCT) an important - arguably the most important - performance metric for the user. Yet research on congestion control focuses entirely on maximizing link throughput, utilization and fairness, which matter more to the operator than the user. My thesis is about a new congestion control algorithm - Rate Control Protocol (RCP) - designed for fast download times (i.e. aka user response times, or flow-completion times). Whereas other modifications/replacements to TCP (e.g. STCP, Fast TCP, XCP) are designed to work for specialized applications that use long-lived flows (scientific applications and supercomputer centers), RCP is designed for the typical flows of typical users in the Internet today.

I showed that with typical Internet flow sizes, existing (TCP Sack) and newly proposed (XCP) congestion control algorithms make flows last much longer than necessary - often by one or two orders of magnitude. In contrast, RCP makes flows finish close to the minimum possible, leading to a perceptible improvement for web users, distributed computing, and distributed file-systems. I have also addressed several questions for an RCP network - stability of an RCP network, coping with sudden network changes such as flash-crowds (the main weakness under RCP), RCP's router buffer-size requirements, proportional bandwidth-sharing with RCP and implementation of RCP in routers and end-hosts.

RESEARCH EXPERIENCE

THESIS RESEARCH

2002 – Present *Research Assistant, Computer Systems Laboratory, Stanford University*

Minimizing Flow-completion Times in a Network (Jan 2002 – July 2003): How would we design congestion control to minimize download times? Unfortunately, it is intractable to find an optimal solution for a general network under arbitrary flow arrivals and departures. I found that under a more constrained problem of allotting a single rate to all flows, there exists a rate that minimizes flow-completion times for heavy-tailed flow sizes (which they are in the Internet). This finding became the basis of RCP congestion control algorithm.

Rate Control Protocol (RCP) Congestion Control (Jan 2003 – Present)**:** I designed a congestion control algorithm (RCP) which heuristically updates the rate it offers to all flows passing through a link and achieves flow-completion times close to an ideal processor-sharing system. The algorithm requires no per-flow state/queue and decides the offered rate based only on the queue-size, aggregate traffic and average round-trip time of flows, making it simple and practical.

Network Stability under RCP (Sept 2003 – Dec 2004): A desirable characteristic of any congestion control mechanism is that it be stable even under sudden unexpected changes in traffic load. Using control-theory I proved that a linearized model of RCP congestion control is stable for a broad range of network and traffic conditions. In collaboration with control-theorists RCP stability was later extended to the more realistic non-linear system as well as for traffic conditions involving dynamic flow arrivals/departures.

Buffer-size requirements under RCP (Jun 2006 - Present): Buffer-sizing in Internet routers has recently received attention because of its huge implications on cost and complexity of router design. Along with a colleague, I analyzed the amount of buffering needed by RCP routers for a bounded packet loss probability. Analysis and simulations indicate that buffers have to scale with link-capacities, however a fraction of the bandwidth delay product (often as little as 5%) is sufficient to maintain flow-completion times that are at most 20% larger than FCTs under very large buffers.

Implementing RCP Congestion Control in Linux (Jun 2006 - Present): Implemented congestion control mechanisms of the RCP end-host in Linux 2.6.16.

OTHER RESEARCH

Sept 06 – Present *Research Assistant, Stanford University (collaboration with Cisco Systems, San Jose, CA)*

Making a case for explicit network feedback for Congestion Control: There is no current widespread deployment of explicit feedback mechanisms. Our goal is to create a compelling demonstration on how user-experience will be perceptibly better (examples include downloading High Definition video, web-browsing, file-system applications, on-line games) if the network explicitly participated in congestion control and ultimately making a case on why router/switch vendors should be interested in explicit feedback schemes.

Summer 2005 *Research Assistant, Computer Systems Laboratory, Stanford University*

Typical versus Worst Case Design in Networking: Observed that Networking research has a strong inclination toward designing systems for the worst-case scenarios. While designing for the worst case gives strong guarantees on the system, it often comes with a high cost and a sacrifice for the typical case performance. Along with colleagues, I demonstrated (through examples) that it is possible to design systems for the typical-case to reap the enormous benefits in performance, cost and complexity, without overly hurting the worst-case.

Summer 2004 *Research Intern, Fraser Research, Princeton, NJ*
Designing congestion control for zero packet-loss and fast flow-completion times: Packet loss is detrimental to many applications, video-on-demand, online gaming and data-center applications being the most notable of them. Although RCP congestion control finishes flows quickly, there are no guarantees of zero loss. I designed RCP-AC (RCP with Acceleration Control) – a tunable congestion control algorithm that can be adapted to achieve zero packet loss at one end of the spectrum and short flow-completion times at the other extreme. RCP-AC allows an operator to choose the operating point, and control packet loss, while still allowing flows to finish much more quickly than in conventional algorithms.

Summer 2002 *Research Intern, Bell Labs, Lucent Technologies, Holmdel, NJ*
Prefix Allocation to Minimize Routing Table Sizes: As the Internet grows rapidly one of the problems facing the backbone routers today is the increasing size of the routing table. Solved the problem of allocating prefixes to the network nodes in order to minimize the maximum routing table size in the network.

1998 - 2000 *Research Assistant, Indian Institute of Science, Bangalore, India*
Optimal Resource Allocation in Packet Networks: A key issue in delivering Quality-of-Service guarantees is resource allocation at the network nodes to satisfy the end-to-end delay requirements. Devised and implemented algorithms to determine the optimal capacity allocation satisfying end-to-end delay requirements in a network of Generalized Processor Sharing (GPS) schedulers. Demonstrated that these algorithms outperform previously proposed algorithms.

Aug-Dec 1997 *Undergraduate Researcher, Birla Institute of Technology and Science, Pilani, India*
PC-based Fiber Optic sensing system: Designed a PC based fiber optic sensing system that measures the angular displacement and Numerical Aperture of the fiber.

WORK EXPERIENCE

Aug 2000 - 01 *Research Associate, Himachal Futuristic Communications Limited, Bangalore, India*
Network Topology Design Tool: Designed and implemented computationally efficient heuristic algorithms for a Network Topology Design Tool. The tool helps ISPs in deciding the placement of routers in the network, topology connecting the routers and the optimal flow paths, all in a cost-effective way.

Jan–Jun 1998 *Intern, Honeywell India Software Operations, Bangalore, India*
Developed a client-server package in C++ that lets a mobile user access, monitor and control a Building Management System using a handheld PC. The server lets the remote client to access various sites of the corporate without having the need to connect to the individual sites. The client software provides a user-friendly interface.

TEACHING EXPERIENCE

Winter 2006 *Teaching Assistant, Electrical Engineering Department, Stanford University*
EE384X, Packet Switch Architectures-I (Graduate-level Networking class)**:** Teaching Assistant to Prof. Balaji Prabhakar and Prof. Nick McKeown. Prepared and gave lectures for material not covered in class. Formulated questions for homework sets, quizzes and midterm/final examinations, graded exams and interacted with students during weekly office hours.

Spring 2006 *Teaching Assistant, Electrical Engineering Department, Stanford University*
EE384Y, Packet Switch Architectures-II (Graduate-level Networking class)**:** Teaching Assistant to Prof. Balaji Prabhakar and Prof. Nick McKeown. Designed and graded homework sets, quizzes and examinations as well as guided students during the weekly office hours.

PUBLICATIONS

PAPERS

- N. Dukkipati, N. McKeown, A. G. Fraser, "RCP-AC: Congestion Control to make flows complete quickly in any environment," *High-Speed Networking Workshop: The Terabits Challenge (in conjunction with IEEE INFOCOM 2006)*, Barcelona, Spain, Jun 2006.

- N. Dukkipati, N. McKeown, "Why Flow-Completion Time is the Right Metric for Congestion Control," *SIGCOMM Computer Communications Review*, Vol. 36, No. 1, Jan. 2006.

- N. Dukkipati, Y. Ganjali, R. Zhang-Shen, "Typical versus Worst Case Design in Networking," *Fourth Workshop on Hot Topics in Networks (HotNets-IV)*, College Park, Nov 2005.

- N. Dukkipati, M. Kobayashi, R. Zhang-Shen, N. McKeown, "Processor Sharing Flows in the Internet," *Thirteenth International Workshop on Quality of Service (IWQoS 2005),* Passau, Germany, Jun 2005.

- A. Panagakis, N. Dukkipati, I. Stavrakakis, J. Kuri, "Optimal Call Admission Control on a Single Link with GPS Scheduler*," IEEE/ACM Transactions on Networking, vol. 12, No. 5*, Oct 2004.

- N. Dukkipati, J. Kuri, H.S. Jamadagni, "Optimal Call Admission Control in Generalized Processor Sharing (GPS) Schedulers," *Proceedings of IEEE INFOCOM*, Anchorage, April 2001.

- N. Dukkipati, J. Kuri, H.S. Jamadagni, "Optimal Resource Allocation in Packet Networks that use Rate-Based Schedulers," *IEEE Conference in Communications, Controls and Signal Processing (CCSP 2000),* Bangalore, July 2000.

PENDING PAPERS, INTERNET DRAFT

- A. Lakshmikantha, N. Dukkipati, R. Srikant, N. McKeown, C.L. Beck, "Performance Analysis of the Rate Control Protocol," Under preparation to *ACM Computer Communications Review*.

- H. Balakrishnan, N. Dukkipati, N. McKeown, C. Tomlin, "Stability Analysis of Explicit Congestion Control Protocols," Under submission to *IEEE Communications Letters*.

- N. Dukkipati, G. Gibb, N. McKeown, "Implementation and Experiments with the Rate Control Protocol (RCP)," Under preparation.

- N. Dukkipati, N. McKeown, F. Baker, "Implementing RCP in the IPv6 Hop-by-Hop Options Header," *Internet Draft* in preparation.

- N. Dukkipati, N. McKeown, "Rate Control Protocol (RCP): Congestion Control to Make Flows Complete Quickly," Under preparation to the *IEEE/ACM Transactions on Networking*.

TECHNICAL REPORTS

- A. Lakshmikantha, N. Dukkipati, R. Srikant, N. McKeown, C.L. Beck, "Performance Analysis of the Rate Control Protocol," *UIUC Technical Report*, December 2006.

- N. Dukkipati, N. McKeown, "Why Flow-Completion Time is the Right Metric for Congestion Control," *Stanford High Performance Networking Group Technical Report TR05-HPNG-112102*, November 2005.

- H. Balakrishnan, N. Dukkipati, N. McKeown, C. Tomlin, "Stability Analysis of Explicit Congestion Control Protocols," *Stanford University Department of Aeronautics and Astronautics Report: SUDAAR 776,* September 2005.

- N. Dukkipati, N. McKeown, "Processor Sharing Flows in the Internet," *Stanford High Performance Networking Group Technical Report TR04-HPNG-061604*, June 2004.

- N. Dukkipati, "Optimal Resource Allocation in Packet Networks," *Masters Thesis, Indian Institute of Science*, September 2000.

PATENTS

P. Chaporkar, N. Dukkipati, J. Kuri, A. Kumar, "Method for Fast Cost-effective Internet Topology Design," United States Patent Application Number 10/614,683, July 2003.

TALKS

CONFERENCES, ACADEMIA, RESEARCH LABS

- (*at several venues*) "RCP: Congestion Control to Make Flows Complete Quickly"
    - Microsoft Research, Redmond, Sept 2006
    - (*Invited by Craig Partridge*) End2End Meeting, MIT, Cambridge, July 2006
    - (*Invited by Prof. Steven Low*) Caltech Lunch Bunch Seminar, Nov 2005
    - Berkeley Systems Lunch, Oct 2005
    - (*Invited by Prof. Robert Shorten*) Hamilton Institute Workshop on Congestion, Sept 2005
    - (*Invited by Prof. Scott Shenker*) International Computer Science Institute (ICSI), July 2005
    - International Workshop on Quality of Service (IWQoS), Passau, Germany, June 2005
    - (*Invited by Prof. Jim Kurose*) University of Massachusetts at Amherst, April 2005
    - MIT CSAIL, April 2005

- "RCP: Prototyping, Demonstrating and Solving Deployment Questions," *Stanford, CA*, Sept 2006

- "RCP-AC: Congestion Control to Make Flows Complete Quickly in any Environment," *High Speed Networking Workshop: The Terabits Challenge, Infocom 2006, Barcelona, Spain*, April 2006

- "Designing Congestion Control for Fast Flow-completion Times," *100x100 Clean Slate Retreat, Houston*, Dec 2005

- (*Invited to be on a panel on Structure and Protocol Design*) "Using Simple Structures to Achieve Complex Properties," *100x100 Clean Slate Retreat, Houston*, Dec 2005

- "Typical versus Worst-case Design in Networking," *Fourth Workshop on Hot Topics in Networks, HotNets-IV*, Nov 2005

- "Congestion control in 100x100: Why TCP is a poor choice and how to redesign it from scratch," *100x100 Clean Slate Retreat, Pittsburgh*, Dec 2004

- "Optimal Resource Allocation in Packet Networks," *IEEE INFOCOM 2001, Anchorage, Alaska, USA*, April 2001

INDUSTRY

- "A Wish List for the next Congestion Control Protocol and Why TCP may not get us there," *Cisco-Stanford Meeting, Cisco Systems, San Jose*, Oct 2006
- "Making a case for Explicit Feedback for Congestion Control," *Cisco-Stanford Meeting, Cisco Systems, San Jose*, Nov 2006
- (*At several venues*) "RCP: Congestion Control to Make Flows Complete Quickly"
  - (*Invited by John Wakerly*) *Cisco Distinguished Engineers/Fellows Meet, Cisco Systems, San Jose*, July 2005
  - (*Invited*) Cisco Network Architecture Geeks (NAG) Conference, Cisco Systems, San Jose, July 2006
  - (*Invited*) Boeing Aerospace Corporation, June 2006
  - (*Inivited by Jean Bolot, Sprint ATL CTO*) Sprint Advanced Technology Labs, Burlingame, CA, USA, Oct 2005
  - (*Invited by Flavio Bonomi, DE Cisco Systems*) Cisco Systems, San Jose, July 2005

PROFESSIONAL ACTIVITIES

Ad-hoc Reviewer for IEEE/ACM Transactions on Networking, IEEE Communications Letters, ACM SIGCOMM Computer Communications Review, ACM SIGCOMM 2006, 2002, ACM/IEEE ANCS 2005, IEEE INFOCOM 2003.

REFERENCES

Prof. Nick McKeown (Advisor)
Associate Prof. of Elec. Eng. and Comp. Sci.
353 Serra Mall, Gates 340
Stanford University
Stanford CA 94305-9030, USA
Phone: (650) 725-3641
nickm@stanford.edu

Prof. Balaji Prabhakar (Associate Advisor)
Associate Prof. of Elec. Eng. and Comp. Sci
350 Serra Mall, Packard 269
Stanford University
Stanford CA 94305-9510, USA
Phone: (650) 723-6579
balaji@stanford.edu

Prof. Scott Shenker (Thesis committee member)
Professor of Computer Science, UC Berkeley
465 Soda Hall
Berkeley, CA 94720-1776, USA
Phone: (510) 643-3043
shenker@icsi.berkeley.edu

Dr. Alexander Fraser (Mentor)
President and Founder, Fraser Research
182 Nassau Street, Suite 301
Princeton, NJ 08542, USA
Phone: (609) 497-7337
agf@fraserresearch.org

Dr. Flavio Bonomi (Research Mentor and Collaborator)
Distinguished Engineer
Cisco Systems
170 West Tasman Dr.
San Jose, CA 95134, USA
Phone: (408) 526-4085
flavio@cisco.com

Dr. David Clark (Research Mentor)
Senior Research Scientist
MIT Comp. Sci. and Artificial Intelligence Lab. (CSAIL)
The Stata Center, 32 Vassar Street, 32-G816
Cambridge, MA 02139, USA
Phone: (617) 253-6003
ddc@csail.mit.edu

Statement of Research for Nandita Dukkipati

nanditad@stanford.edu

The Internet has been a remarkable success, yet in many ways it is woefully inadequate in meeting the needs of applications and users. Its performance is unpredictable, it is frequently unavailable when we need it, and it does not easily support new technologies such as wireless or evolve with changing needs of users and applications. There are applications we would not think of running on it, such as a remote health monitoring and diagnostic service which requires location awareness, ubiquitous and seamless connectivity, and high availability while being able to support a large number of devices. The focus of my research is to make networks more adaptable, evolvable, self-managed, and cost-efficient, as well as improve their performance so as to make these applications feasible. While my primary goal is to innovate and build practical solutions, I also enjoy the approach of using theoretical tools wherever applicable.

My current work is on designing a novel congestion control algorithm for the Internet that is appreciably better than the widely used TCP. I describe this work below followed by my ideas for future research. A lot of my work has immensely benefited through collaboration with my colleagues.

Current and Past Research

My research focus has been in the area of resource allocation in networks. For my Ph.D. dissertation, I designed a new congestion control mechanism - Rate Control Protocol (RCP) - for fast download times (aka user response times, or flow-completion times), that is also significantly more efficient, stable, adaptive, and robust compared to the widely used TCP.

Congestion control solves the problem of sharing network resources in a distributed, cooperative, and fair fashion. It is unique to Networking and integral for the Internet to function well. Lack of mechanisms to share bandwidth effectively under over-loaded conditions in the early Internet led to several network meltdowns, known as congestion collapses. It spurred a large body of work on how to prevent a congestion collapse, nearly all of which universally focusing on metrics such as throughput, bottleneck utilization, fairness, packet drops/delay, and network stability. While these metrics are interesting – particularly for the network operator – they are not very interesting to a network user. When users download a web page, transfer a file, send/read email, or involve the network in almost any interaction, they want their transaction to complete in the shortest time; and therefore, they want the shortest possible flow completion time (FCT). Today, most transactions are of this type and it seems likely that a significant amount of traffic will be of this type in the future. That is the problem I wanted to explore: *how do we design congestion control algorithms and mechanisms that minimize flow-completion times?* Unfortunately, it is intractable to minimize flow-completion times. So instead congestion control algorithms are focussed on an easier to achieve goal of efficiently using bottleneck links. But, I believe that instead of being deterred by the complexity of the problem, we should find algorithms that come close to minimizing FCTs, even if they are heuristic.

I discovered that with typical Internet flow sizes, existing (TCP Sack) and newly proposed (XCP) congestion control algorithms make flows last much longer than necessary - often by one or two orders of magnitude. RCP makes flows finish close to the minimum possible, leading to a perceptible improvement for web users, distributed computing, and distributed file-systems. It achieves this by explicitly emulating Processor Sharing at each router. In RCP, a router assigns a single rate, R(t), to all flows that pass through it. I designed RCP to be simple and practical - it is an adaptive algorithm that updates the rate assigned to the flows, to approximate processor sharing in the presence of feedback delay, without any knowledge of the number of ongoing flows. The flow rate, R(t), is picked by the routers based on very little information (the current queue occupancy and the aggregate input traffic rate), requires no per-flow state or queue, and has very few per-packet computations.

Congestion control must ensure the network's health and stability even under abnormal and extreme traffic patterns. Using techniques in control-theory, I showed that RCP is provably stable under severe network overloads, irrespective of round-trip times, link-capacities, and the number of flows. Not only is it stable, it also recovers to normal behavior within a small number of round-trip times even when confronted with sudden and unanticipated network changes.

Determining the correct size for router buffers has piqued a lot of interest recently because of its profound implications on router architecture and power consumption, flow performance, and network stability. It has been demonstrated experimentally and shown theoretically that TCP congestion control works well even under very small buffers. I demonstrated that RCP does not need huge buffers to deliver its promise of short FCTs. Along with a colleague, I showed that even with buffers as small as 5% of the bandwidth-delay product, flow-completion times are at most 20% longer than with very large buffers for a wide range of network and traffic characteristics.

One of the promises of RCP is the simplicity and practicality of its implementation in routers and end-hosts. Along with colleagues at Stanford, I demonstrated that RCP's implementation is practical in both the end-hosts as well as the routers.

Some newer applications in the Internet, such as High Definition video streaming, need extremely low packet-loss to operate well. I extended RCP to RCP-AC (Rate Control Protocol with Acceleration Control) which can be tuned to achieve low packet-loss and extremely well-behaved queues even under sudden and large traffic changes at one end of the spectrum and short flow-completion times at the other end. RCP-AC is designed with optimization in mind, and is adaptable to changing traffic patterns and expectations of applications using it.

I spent a significant amount of time in my Ph.D. to explore the impact RCP can have on Internet congestion control. Feedback from practitioners in networking industries pointed out that RCP congestion control for short download times is not only a good idea for the Internet, but is also useful in niche networks with atypical network and traffic characteristics such as satellite and data-center networks.

In the past, I had also worked on mechanisms to deliver hard QoS guarantees. A key issue in delivering Quality-of-Service guarantees is resource allocation at the network nodes to satisfy the end-to-end delay requirements. I devised and implemented algorithms to determine the optimal capacity allocation satisfying end-to-end delay requirements in a network of Generalized Processor Sharing (GPS) schedulers. I demonstrated that these algorithms outperform previously proposed algorithms.

Future Research

Over the years many applications have proliferated over the Internet ranging from short web downloads to high bandwidth high-definition video streaming. However, there is a much richer set of applications we would like to run in the future including those to monitor health remotely, manage, store, and share vast amounts of personal data online, discover local services from mobile clients and vehicles, and facilitate rich interaction among people spanning continents. While on one hand these represent scaling up from where we are to a much better Internet, more than half of the world does not even have basic web access and the technical challenges to overcome are of a different nature including making the infrastructure cost-effective, power-efficient, and tolerant to intermittent disruptions. As it stands, our current infrastructure cannot support either of these two extremes. My future research in broad terms will be on designing protocols and mechanisms to improve our networks' performance, make them more scalable, evolvable and adaptable while lowering costs, to get us from where we are to where we would like to be. I have listed below a cross-section of the problems I am inspired to work on in the near term:

Designing protocols for different network requirements

Networks with requirements very different from those of main-stream Internet are proliferating such as storage area networks and networks in data-centers, enterprises, homes, automobiles, and planes. They all have very different needs of speed, reliability, operating costs, and network conditions (such as very high or low round-trip times, link-speeds, unusual traffic patterns, and different wireless technologies). Yet the same mechanisms and protocols designed for the long-haul Internet are being retrofitted to work in these environments. As a consequence they are inefficient at best and simply do not work at worst.

As an example: Packet loss due to temporary network congestion is only a minor annoyance in the Internet and up until now there was no motivation to achieve a loss-free network. After all, not only does TCP retransmit lost packets but loss is its only indication of congestion. However, networks where even occasional congestive losses is no longer an option are proliferating. Examples include edge networks used for High Definition video-on-demand transfers where current practice is to vastly over-provision links so as to achieve a loss-free environment. This is not only an expensive solution but one that does not scale as video transfers become more wide-spread. Similarly, performance of distributed applications in data-centers takes a big hit due to packet losses and so a loss-free network is highly desirable. Clearly, TCP congestion control is not a good answer for either of these networks. I want to explore routing and congestion control mechanisms to fulfill the needs of these newer applications on networks with very different requirements.

Leveraging different infrastructures to provide efficient and low-cost connectivity for mobile users

Connectivity is fundamental. We have invested in abundant connectivity and there is more being put in place. We have multiple broadband, cellular (3G, 4G), Mesh networks, Satellite, WiMax, and Wi-Fi systems. There are many open access points available and more coming up, and while the coverage is not complete, it gives a sense of what is possible. Recent developments in software radios allow a single device to be reconfigured on the fly to access all of these networks. We have these multiple parallel infrastructures in place, but what users eventually care about is effective connectivity that also satisfies their needs. For example, it is inconceivable today for mobile users to seamlessly switch between blasting packets all the way to cell towers and using open Wi-Fi and mesh networks where available. The Internet does not support this and is completely unaware of the ephemeral mobile nodes.

I am interested in developing protocols that would leverage these different infrastructures to provide seamless connectivity that would also meet users' needs, especially for mobile users. Needs can be of different forms: lowest cost, best reliability, highest speed possible or lowest latency. Protocols should be able to adapt quickly to changing network conditions. How do we design mechanisms and protocols that would be native to this dynamic view of the infrastructure? One of the challenges would be to come up with adaptive protocols that anticipate discontinuous connectivity and packet loss and achieve efficient resource usage in such an environment.

Scaling networks to support Peer-to-Peer traffic

Peer-to-Peer (P2P) file-sharing has often been criticized because of its association with illegal distribution of copyrighted content. I am intrigued by P2P networks because of the possibilities they open in providing low cost communications without the need of a huge managed infrastructure. As more users download video and other high bandwidth content, the architecture of today's infrastructure with large data-centers located near cheap electricity with thousands of PCs, disk drives, uninterruptible power supplies, and lots of fiber can get prohibitively expensive. P2P changes the economics of the game and not surprisingly many commercial content providers (including Hollywood, BBC, Linux distributions) are increasingly providing their content via P2P networks.

P2P applications hog bandwidth and it is estimated that they account for 60 to 80% of traffic on consumer ISP networks. Service providers complain that networks are frequently clogged with P2P traffic, and yet, not all parts of the network are equally congested at the same instant. I want to explore solutions that will scale networks to support P2P traffic and treat it as a first-class application as opposed to unwanted traffic. For example, I want to investigate mechanisms that combine routing, traffic engineering and congestion control to diffuse P2P traffic in real-time to make use of paths with available bandwidth, deflect it away from congested paths, and use more of local bandwidth.

Integrating evolution into network protocols and mechanisms to adapt to unanticipated changes

The Internet's protocols and mechanisms are resilient in the sense that they continue to function even under unexpected changes in operating conditions. However, they do not necessarily function efficiently. While we are generally good at making a specific protocol work efficiently for some specific range of operating conditions, we do not quite know how to make them continue working efficiently under unanticipated changes. This is especially important in networks because unlike many traditional systems they do not have the luxury of being replaced very frequently.

The question I am interested in is: how do we integrate evolution in network controls including congestion control, Active Queue Management schemes, routing protocols, and traffic engineering, such that they are adaptable, tunable and can be optimized to be efficient under changing (and often unanticipated) conditions and constraints. Changes can be of different forms: traffic patterns, user behavior and expectations, and radically different applications. I want to explore solutions by drawing techniques from different disciplines including Machine learning, Statistics, Algorithms, Optimization Theory, and Distributed Systems.

The above problems present a flavor of the research I am inspired to work on. They all share one or more of the following themes at their core: creating solutions to lower costs; making networking systems more efficient, adaptable, and evolvable. I enjoy innovating solutions that appeal to theorists and make practical sense at the same time. In the past, I had a fulfilling experience collaborating with disciplines outside of Networking and I will actively pursue the same for my future research. In addition, I am immensely interested in collaborating with practitioners in Industry to both learn of realistic problems as well as to find a way for my research to have an impact in the real world.

Statement of Teaching for Nandita Dukkipati

nanditad@stanford.edu

I was fortunate to have had several outstanding teachers - their classes so engaging that I recall being excited after each lecture, eager to learn more, and looking forward to the next lecture. Learning was a pleasure. With passing years I forgot the textbooks but not the lectures. This is the kind of experience I want my students to have. My teaching objectives are rooted in my excitement for Networking research. Apart from getting across the subject material effectively, my goal is also to share my enthusiasm for the field of Networking and its related disciplines. Through my experience as a Teaching Assistant at Stanford as well as a student, I have identified the following as key to achieving my goals:

1) *Design intriguing lectures and motivate the content by choosing apt examples drawn from real systems:* An engaging Networking class is more than a collection of isolated facts. Rather, it is a coherently intertwined story weaved together from fundamental concepts and appropriate examples drawn from real/prototype systems of both Industry and Academia. Not only does it make attending classes a unique and worthwhile experience but perhaps more importantly, it helps students realize why learning the subject at hand is a good investment of their time. As a Teaching Assistant for Packet Switch Architectures, I applied this principle when teaching review sessions. For example, when teaching Probability, I took the time to prepare material on the applications of probability in Networking, how it not only enhances our understanding of existing systems but can also guide us when designing new ones. It appealed to the students.

   Design problems in Networking and Systems rarely have obvious right and wrong answers, with economics, technology, and non-obvious engineering constraints often influencing the choices made. A crucial part of my teaching will be to foster critical thinking of and draw lessons from historical and existing networking systems.
2) *Explicitly bring out the interdisciplinary flavor of Networking:* Networking is a blend of several disciplines. In my teaching, I will make a deliberate effort to explicitly bring out the interdisciplinary flavor - on how theoretical results in Probability, Control Systems, Graph Theory, Algorithms, and Optimization techniques are used to analyze existing systems and lay the foundations for new designs, how concepts in Operating Systems, Distributed Systems, and Hardware Design help us build practical systems, and how principles in Economics align the incentives for a multiplayer system. Fortunately, I have had good exposure to several of these disciplines as a student and I immensely enjoy the process of synthesizing concepts from disparate disciplines and applying/teaching them in the context of networks.

   I also believe that hands-on assignments/projects are of paramount importance in learning networking concepts. Using tools such as Virtual Machines (eg. User Mode Linux, VMWare), Stanford's Virtual Network System (VNS) and simulators such as *ns2*, I will create educative experimental projects for the students.
3) *Preparation is the key:* I believe to be an outstanding teacher comes through with a lot of preparation as well as trial and error, especially in the formative years of a class. As a teaching assistant at Stanford, I taught review sessions for which I spent a good amount of time preparing notes from several textbooks and coming up with realistic examples to drive home the point, teaser questions to bring out subtler points and increase interactivity in the class. The results were rewarding - the students loved the review sessions, the attendance was nearly 100%, students with backgrounds from diverse disciplines were appreciative of Networking, and the learning experience was fun for students and me alike.

The most important element in teaching is enthusiasm for the subject at hand. Much of my motivation for teaching comes from my enjoyment in what I do as a researcher. I have had a rewarding experience as a Teaching Assistant at Stanford. I had an opportunity to give many talks in industry and academia. Over time, I honed my presentation skills and became proficient on how to convey seemingly complex ideas in a simple way to a wide range of audience. I am prepared to teach a broad array of Networking classes – a basic Networking class for undergraduates, a more advanced graduate level class, as well as a seminar and project class that discusses the most recent advances along with an in-depth coverage of a few selected topics. Apart from that, I can also teach basic classes in Electrical Engineering and Computer Science such as Signals and Systems, Introduction to Probability, and Introductory Algorithms.