

RESEARCH STATEMENT

ARUN VENKATARAMANI

SEPTEMBER 2007

Advances in computer networks and the remarkable success of the Internet have dramatically changed how we live, work, and interact. Today, the Internet forms a critical infrastructure for business, health care, education, entertainment, and other networked services. Internet connectivity is widely perceived and purveyed as a utility service similar to power or transportation. The increasing number of mobile devices and the diversity of edge networks connected to the Internet further contribute to its growth and value.

Despite its success, the Internet today is not *dependable*. Users of Internet-based networked systems frequently experience service disruption or performance problems. Internet availability is about two orders of magnitude lower than that of traditional utility services. The Internet's core protocols, originally designed assuming a benign environment, are vulnerable to abuse and frequently exploited by attackers. Furthermore, these protocols perform poorly in wireless environments such as mesh, sensor, mobile ad hoc, or delay-tolerant networks, thereby falling short of the Internet's promise as a universal communication infrastructure.

Research goals *My research is in the design, analysis, and implementation of dependable networked systems.* A dependable networked system must be robust to 1) changing load, i.e., it should scale gracefully to handle massive load spikes and changing traffic patterns, 2) changing node behavior, i.e., it should tolerate failure, selfish behavior, and malicious behavior by nodes, and 3) changing network conditions, i.e., it should gracefully tolerate diverse and challenging network conditions. My work seeks to identify fundamental principles underlying the design of dependable *network protocols* as well as *distributed systems* built over these networks.

Research approach My research approach is founded on both building real systems and applying theoretical techniques to better understand these systems. Building systems enables my research to identify practical problems and deliver usable solutions, while the underlying theory enables my research to distill fundamental principles applicable beyond the specific system being investigated. My research also relies on simulation experiments, particularly those based on real-world traces, to further inform or validate the design. In addition to papers, some systems I helped create have seen real use, e.g., BitTyrant [NSDI 2007] has been downloaded over 500K times; RAPID [SIGCOMM 2007] and a Web search service based on it have been deployed on buses in Western Massachusetts; iPlane [OSDI 2006] provides a publicly queryable map of the Internet's routing topology refreshed daily and has attracted considerable interest from researchers and industry; my earlier work on TCP Nice [OSDI 2002] has been incorporated into IBM products.

A particular strength of my approach is its cross-disciplinary nature and an openness to drawing on ideas from other fields. My research in network protocols is informed by the theory and practice of distributed systems, as exemplified by ongoing research on *consensus routing* and *swarming*, and earlier work on TCP Nice. I also liberally explore ideas from optimization and control, queueing theory, game theory, algorithms, information theory, and statistics, often by creating successful collaborations with experts in these areas, to strengthen my research in networking.

The rest of this statement describes my main contributions and ongoing research towards 1) a swarming data transfer architecture robust to massive scale and selfish node behavior, 2) a network layer robust under diverse edge network environments, and 3) a secure network robust to byzantine node behavior; followed by an education statement. Detailed citations for all papers referred in this statement are available in my CV. This statement focuses on my research since joining the faculty at the University of Massachusetts.

Swarming Internet data transfer

A dependable network must perform data transfer in a scalable, fault-tolerant, and incentive-compatible manner, a trio of properties difficult to achieve today. Inspired by the tremendous success of swarming systems like BitTorrent—studies estimate that BitTorrent accounts for up to half of all Internet traffic today—we are pursuing the following research question: *can swarms form the basis of a universal data transfer architecture and if so, what is an appropriate architecture?* The term *universal* means that said architecture benefits practically all data transfer, and the term *swarm* means that a set of loosely interconnected nodes act in a selfish and highly decentralized manner and are always in a state of adaptation. Natural systems with swarm-like properties are known to be extremely robust both for an individual and for the system as a whole.

A swarming data transfer architecture will systematically 1) leverage multipoint-to-point connections, 2) account for selfish behavior at every step in data transfer, 3) enable fluid replication of data at any location where it is accessed with in-network support for caching and retrieval, 4) use self-certifying names for secure, location-independent data transfer, and 5) integrate existing infrastructure based on servers and managed content distribution systems. The envisioned architecture raises several questions: Is BitTorrent’s incentive strategy really robust to selfish peer behavior? Where are the performance and availability bottlenecks in BitTorrent today? Can decentralized peer systems benefit from more information about global network performance and availability? Can peers assemble such information in a decentralized manner? How does widespread use of multipoint-to-point connections impact congestion control? Can end-system based congestion control techniques enable better allocation of network resources? Our recent research contributions provide the foundation to answer some of these questions.

BitTyrant [NSDI 2007], developed with Mike Piatek, Tomas Isdal, Tom Anderson, and Arvind Krishnamurthy, debunked the popular belief that BitTorrent’s incentive strategy is robust to selfish peer behavior. The widespread use of BitTorrent makes it a valuable target of investigation for networking researchers. In this work, we modeled BitTorrent’s performance parameterized by real-world traces and discovered the presence of significant altruism, i.e., all peers regularly contribute resources to the system that do not directly improve their performance. Intrigued by this observation, we designed and implemented a modified BitTorrent client, BitTyrant, designed to benefit strategic peers. The key idea in BitTyrant is to carefully 1) select peers, and 2) select upload rates to those peers, so as to maximize return (= download rate) on investment (= per unit upload bandwidth). In comparison, we found that BitTorrent in practice effectively uses a random strategy for both choices.

We showed, as has been confirmed by several independent sources including the popular press, that typical users of BitTyrant see a three-fold decrease in download times. Furthermore, we showed that BitTyrant is not only selfish but also a superior BitTorrent client, i.e., peers individually benefit from using BitTyrant regardless of how many other peers are using the same strategy. In particular, when all peers use BitTyrant, global performance improves over BitTorrent. Since its release, BitTyrant has seen over half a million downloads, an encouraging finding for networking researchers grappling with the problem of evolvable system design. Our paper on this work received the best student paper award at NSDI 2007.

BitTorrent availability [INFOCOM 2007], a measurement study with Giovanni Neglia, Honggang Zhang, Don Towsley and others, investigated the impact of two recent developments in BitTorrent on its availability. The first is the use of *replicated trackers* for load balancing or fault-tolerance. We found that some torrents employ up to fifty trackers for the same torrent. The second is the prevalence of DHT-based tracking as an alternative to replicated trackers, perhaps making BitTorrent the largest known real system using DHTs. We found that both replicated and DHT trackers improve availability in complementary ways. Replicated trackers ensure low tracking latency, but are vulnerable to correlated failures. DHT trackers yield high availability even though a majority of the nodes are always down, but at a much higher latency. Our study suggests that tracker load and availability are serious concerns in BitTorrent today.

iPlane [OSDI 2006, IMC 2006], developed with Harsha Madhyastha, Tom Anderson, Arvind Krishnamurthy and others, is an *Internet information plane* to benefit distributed services by providing them sophisticated information about global network behavior. The Internet, by design, is opaque to its users today. However, many popular distributed services such as content distribution, peer-to-peer file swarming,

and VoIP can benefit from sophisticated information about network performance. iPlane addresses this need by providing a rich link-level annotated routing and performance map of the Internet that can be frequently refreshed (say, once an hour) with minimal overhead. The key insight in iPlane is a structural approach to predict the path between two arbitrary Internet nodes based on routes observed from a small number of vantage points. The term *structural* means that iPlane uses knowledge of router- and AS-level topology and properties of Internet routing protocols to infer unobserved paths and their characteristics. Our case studies indicate that iPlane yields significant performance benefit for popular overlay services. We currently support a publicly queriable service to infer Internet routes that has drawn significant attention from both researchers and industry. Our ongoing work includes 1) enabling iPlane to provide failure diagnosis and recovery services to benefit end-users as well as network operators, and 2) designing iPlane such that global network behavior can be assembled and disseminated relying only on unmanaged peers.

Multipath Nice [INFOCOM 2007], developed with Ravi Kokku and others, is a multipath routing and congestion control algorithm for background transfers that builds on my initial dissertation work on TCP Nice. Background transfers, i.e., transfers on which humans do not actively wait, dominate Internet traffic today. However, the best-effort Internet does not distinguish between background and foreground transfers, so the two interfere causing long wait-times that hurt human productivity. To address this problem, our system, Harp, is designed to provide background transport as an edge service with two benefits. First, Harp is incentive-compatible. Our previous experience with TCP Nice suggested that selfish users are unwilling to become “good network citizens” by slowing down their background transfers. Harp aligns the incentive for deployment with the goals of network customers by being located at gateways in enterprise or other shared access environments. Second, Harp leverages multiple paths to move background out of the way of foreground transfers to improve performance for both transfers.

This work also illustrates the intellectual merit of systems research grounded in theory. The multipath Nice controller was inspired by multipath congestion controllers recently shown to be stable under feedback delay by Kelly and Voice, and independently by Han et al. Our effort to build a practical system revealed a shortcoming of the fluid model assumed in the design of these controllers, namely, that the utilization is significantly below that predicted theoretically when the number of flows is small. Our work analytically quantifies the impact of the problem and the multipath Nice controller addresses it with the resulting benefits being applicable to both background transfers and regular Internet traffic today.

Internetworking diverse edge networks

A dependable network must be robust to diverse and challenging network conditions. Unlike well-connected core networks that exhibit predictable behavior, edge networks such as mesh, sensor, mobile ad hoc, or delay-tolerant networks (DTNs) inherently have a high degree of uncertainty in topology and network characteristics. Due to this uncertainty, traditional routing and transport protocols perform poorly or utterly break down. For example, DTNs may be always in a state of partition, but traditional routing protocols rely on the existence of a contemporaneous end-to-end path. Our work on DTN routing suggests that packet replication is fundamental to addressing uncertainty in network characteristics. Replication, opportunistic hop-by-hop transport, and decentralized utility-driven algorithms appear to yield significant benefits in many wireless environments such as mesh, sensor, and mobile ad hoc networks. Our research seeks to unify such observations into a *generalized network layer* that is robust to uncertainty across a variety of edge network environments and, in particular, gracefully degrades in performance from well-connected core networks all the way to delay-tolerant networks. Our preliminary contributions to this end are as follows.

RAPID [SIGCOMM 2007], developed with Aruna Balasubramanian and Brian Levine, is a novel delay-tolerant network (DTN) routing protocol. DTNs are inherently characterized by high uncertainty and limited feedback. The burden of finding even one route is so high that many proposed mechanisms for DTN routing have only an *incidental* effect on routing metrics of interest to applications such as delay-based metrics, cost constraints, or deadlines. This disconnect between application needs and routing protocols hinders deployment of DTN applications.

RAPID routing (resource allocation protocol for *intentional* DTN routing) is designed to explicitly optimize an administrator-specified routing metric. The key insight in RAPID is to treat routing as a resource allocation problem, unlike the traditional view of routing as a path computation problem. RAPID routes a packet by opportunistically replicating it until a copy reaches the destination. To address the resource management challenge introduced by replication, RAPID translates the routing metric into per-packet utilities that control packet replication. This utility-driven approach exhibits significant gains over existing approaches and is empirically observed to be close to optimal. We also showed fundamental hardness results for the DTN routing problem for any real algorithm that is either computationally bound or lacks future knowledge. We evaluated RAPID by deploying it over a vehicular network testbed consisting of 40 buses operating in a 150 sq. mile area in Western Massachusetts; to our knowledge, this is the first non-flooding routing protocol deployed in a real DTN.

Web search from a bus [CHANTS 2007], with Aruna Balasubramanian, Brian Levine and others, is an ongoing research effort that 1) demonstrates the benefit of RAPID’s utility-driven approach, and 2) addresses the challenges involved in developing applications over interconnected mesh network and DTN environments. Our system, Thedu, leverages open wireless access points to enable Web search for bus passengers. Thedu imposes a proxy in the wired network that prefetches responses to Web search queries and routes these responses prioritized by relevance scores to the appropriate bus. Using relevance as routing utilities addresses the resource management challenge introduced by aggressive prefetching, and improves user-perceived delay to receive a relevant response. The cross-disciplinary nature of this work done in collaboration with information retrieval experts resulted in contributions in that field as well: we developed a novel normalized ranking procedure to compare relevance of responses to different queries. These normalized scores are used as utilities by RAPID to determine how to allocate limited network resources in order to maximize user utility.

Multi-user data sharing sensor networks [SENSYS 2007], with Ming Li, Deepak Ganesan, and others, further illustrates the value of a utility-driven approach in wireless networks. This work addresses sensing challenges faced by the Center for Collaborative Adaptive Sensing of the Atmosphere (CASA) in serving the needs of multiple concurrent users such as meteorologists who provide weather forecasts, environmental scientists, automated tornado detection systems, and emergency response personnel. We refer to such environments as multi-user data sharing (MUDS), where different end-users with different performance requirements, priorities, or deadlines, operate on common data. The goal of the system to sense and transmit data in a manner so as to maximize overall utility across users. The key insight in our solution is to use a utility-driven progressive compression scheme that senses and transmits the most important blocks of data prioritized by normalized utilities from the wireless radar sensors to a central control system. Our ongoing work is further investigating the benefits of more network layer support for optimizing application utilities.

Secure network protocols

A dependable network must be robust to byzantine behavior on part of both end-hosts and routers. The Internet’s core protocols were originally designed assuming a benign environment, which is at the root of its many fundamental security problems. Our research seeks to develop network protocols from first principles that are secure against a fraction of byzantine faulty nodes. A first step towards this ambitious goal is our ongoing work on consensus routing,

Consensus routing [UMass TR-07-40, Web search keywords: *consensus routing*], developed with John P. John, Ethan Katz-Bassett, Tom Anderson, and Arvind Krishnamurthy, is a novel consistency-first approach to interdomain routing based on classical distributed systems algorithms. The key idea is simple: we observe that Internet routing today favors responsiveness over consistency, i.e., a router applies an update to its forwarding table before propagating it to other routers, including those that depend on the outcome of that update, frequently resulting in routing loops and blackholes. Instead, consensus routing recognizes consistency as a safety requirement and achieves it using two logically distinct modes of packet delivery: a *stable* mode, where routers periodically engage in a consensus protocol to agree upon consistent routes, a

transient mode, where routers failover to backup routes, detours, or deflections ensuring delivery with high probability. Our preliminary results suggest that consensus routing simplifies interdomain routing, cleanly accommodates several existing failover options, assumes no policy restrictions, and improves overall route availability compared to existing approaches.

Having reduced the routing problem to a well known distributed computation problem, the next step is to apply known techniques for byzantine fault-tolerant consensus to tolerate malicious behavior on part of ASes or selfish deviations from expected protocol behavior. We also seek to enable multipath routing and congestion control to tolerate malicious router behavior in the forwarding plane with quantifiable byzantine security guarantees, and make this approach work for both intradomain and interdomain routing.

Dependable distributed systems

My research on making networks dependable is a natural evolution of my earlier and continuing research investigating large-scale distributed systems.

Sandpiper [NSDI 2007], developed with Tim Wood, Prashant Shenoy and Mazin Yousif, is a data center system that uses *virtual machine migration* to eliminate hotspots. Sandpiper automates the task of monitoring and detecting hotspots, determining a new mapping of physical to virtual resources, and initiating the necessary migrations. Sandpiper uses simple auto-regressive predictors for hotspot detection, queuing theoretic models to infer resource needs, and multi-dimensional bin-packing algorithms to compute a new placement of virtual machines. Our prototype based on Xen and evaluation using standard data center benchmarks suggest that Sandpiper can eliminate hotspots in a few seconds and scales well to large data centers while successfully meeting SLA requirements. Our ongoing work includes making data centers more secure and manageable.

PRACTI replication [NSDI 2006], developed with Mike Dahlin and former colleagues at UT Austin, is a new approach to large-scale replication. PRACTI systems can cache any subset of data on any node (Partial Replication), provide a broad range of consistency guarantees (Arbitrary Consistency), and permit any node to exchange information with any other node and make progress (Topology Independence). PRACTI is the first system to be able to provide all three properties, thereby 1) enabling better trade-offs than existing mechanisms, and 2) enabling a universal replication architecture that subsumes a broad range of existing systems and reduces development costs for new ones. Our PRACTI prototype demonstrates these benefits as well as an order of magnitude improvement in performance for a wide range of mobile and distributed applications. The core ideas in PRACTI build upon my dissertation work on TCP Nice and *safe speculative replication* (under submission), an end-to-end architecture for aggressive prefetching with consistency constraints, based on TCP Nice. This body of work illustrates the intellectual merit of cross-pollinating ideas across networking and distributed systems leading to innovations in both areas. Our ongoing work includes making PRACTI systems secure against byzantine node behavior.

EDUCATION STATEMENT

Teaching and mentoring students is a most rewarding aspect of my job and I take pride in having this opportunity. In the classroom, I seek to spark in students my enthusiasm for networking. I encourage interactive participation from students, emphasize real-world application often by showing them little demos, and seek to make the classroom experience fun for all of us. My philosophy towards research colors both my classroom teaching and mentoring of students in the way I emphasize the balance of theory and practice. My philosophy to mentoring students is to set them free but ensure they do not drift. I view my role as a mentor as one who inspires students to create ideas rather than as one who supervises their implementation of their advisor's ideas. My interaction with my students reflects the work dynamics of professional equals. I work closely with them and meet them as often as needed in addition to scheduled weekly meetings. I also seek to tap the interest and potential of undergraduate students in research and regularly advise undergraduate honors theses or summer projects.

Teaching

I have taught or am teaching the following courses at UMass:

- CS691EE: Advanced Network Systems, Fall 2005. (Graduate seminar)
- CS677: Distributed Systems, Spring 2006. (Graduate lectures)
- CS791J: Game Theory and Computer Networks, Spring 2006. (Graduate seminar)
- CS453: Computer Networks, Fall 2006. (Undergraduate lectures)
- CS591G: Computer Networking Lab, Spring 2007. (Undergraduate/graduate)
- CS453: Computer Networks, Fall 2007. (Undergraduate lectures)
- CS591G: Computer Networking Lab, Fall 2007. (Undergraduate/graduate)

Curriculum development

My curriculum development contributions are described below. The material for all classes I have taught at UMass is available online at <http://www.cs.umass.edu/~arun>.

Distributed systems (CS677): I significantly revamped this core graduate class. This class used to be previously taught as an Operating Systems class. I introduced new material on distributed systems that is not otherwise covered in any graduate class at UMass. I am currently working towards adding more material on networking and re-modeling this class as a core graduate class on Networked and Distributed Systems. This class will fill the gap between the existing graduate class on Operating Systems and Advanced Computer Networking that focuses more on theoretical aspects of networking. I seek to expose students to concepts in networked and distributed systems with significant hands-on exposure through programming projects.

Game theory and computer networks (CS791J): With Don Towsley, this graduate seminar class was broadcast via the Internet with interactive participation from faculty and students at Federal University of Rio De Janeiro in Brazil, University of Rome and University of Palermo in Italy, and EPFL in Switzerland. The seminar covered many recent topics in game theory with applications to computer networking and was an immense success. Organizing the course also led to the development of some teaching tools to facilitate similar seminars planned for the near future.

Computer networks (CS453): I introduced a new set of programming projects in this introductory undergraduate class on computer networks. These projects are based on Fishnet, an educational software toolkit for networking classes originally developed by faculty and students at the University of Washington. At UMass, I introduced a top-down version of Fishnet to complement the textbook "Computer Networking: A Top-Down Approach", where students develop their own toy network protocol stack, one level at a time.

Computer networking lab (CS591G): This laboratory class for upper-level undergraduate and graduate students is designed to give more hands-on exposure to students who have completed an introductory class on computer networks. Students work with Cisco routers to learn about practical issues in network protocols. I am currently working towards adding several new assignments to give them a similar hands-on exposure to topics in wireless networks, sensor networks, and delay-tolerant networks.

Mentoring

Almost all of my research has resulted from advised research done by students and I take pride in their accomplishments. Last year, my advisee Aruna Balasubramanian passed her graduate portfolio exam with distinction, typically awarded to one outstanding graduate student in the department each year. I have also advised several undergraduate research projects, some of which have resulted in published papers. The list of students I advise at UMass Amherst and University of Washington is as follows:

PhD

Himanshu Agrawal (with Deepak Ganesan)

Aruna Balasubramanian (with Brian Levine)

John P. John (with Tom Anderson and Arvind Krishnamurthy at U. Washington)

Harsha Madhyastha (with Tom Anderson and Arvind Krishnamurthy at U. Washington)

Daniel Menasche (with Don Towsley)

Sookhyun Yang (with Don Towsley)

MS

Karthik Sivaraman

Synthesis project

Marc Maier (with David Jenson)

Undergraduate

Adrian Sud (Summer 2007, REU)

Eric Pomerleau (2005-06, Honors thesis)

John Danaher (2005-06, Special project, with Don Towsley)

Rob Hall (2005-06, Honors thesis, with Arnold Rosenberg)

Brook Arnold (2005-06, Honors thesis, with Jim Kurose)

Additionally, I have had the opportunity to collaborate closely with several other talented student co-authors (refer list of papers in my CV), talk at informal department seminars, organize and participate in reading groups, and serve on doctoral thesis committees.