Dynamic Network Selection for Robust Communications – Why Disruption Tolerance Matters

Sven Lahde and Lars Wolf
Institute of Operating Systems and Computer Networks
Technische Universität Braunschweig
D-38106 Braunschweig, Germany
[lahde|wolf]@ibr.cs.tu-bs.de

ABSTRACT

Today, mobile users can choose between a wide range of communication technologies when they what to access the Internet. However, they still have to select one of the available communication systems manually. Especially for mobile users, this leaves a large amount of the communication potential unused, since they have to deal with mobility and fluctuating availability of networks by themselves. The term Always Best Connected describes a user-centric network selection concept that is proposed for providing always the best communication performance at any time. However, existing approaches assume a continuous connectivity of the mobile device over time, which is obviously not practical. Disruptions and communication paths with low and varying performance are ubiquitous especially in rural areas. In this paper we present muXer, an architecture aiming at proactively handling disrupted paths and path bundling for increasing the efficiency of dynamic network selection. We discuss why even disrupted paths matter and present our work-in-progress on an arbitrating decision concept providing a modular and flexible basis for network selection strategies with highly heterogeneous decision criterions.

Categories and Subject Descriptors

C.2.1 [**Network Architecture and Design**]: Wireless communication

Keywords

Dynamic Network Selection, Arbitrating, Delay Tolerant Networks, Always Best Connected

1. MOTIVATION

Mobile Internet access enjoyed an increasing popularity during the past years. Mobile devices come with innovative operational concepts and user interfaces allowing for a convenient usage of common (Internet) applications despite of small displays and limited processing power. Moreover, they are equipped with an increasing number of network interfaces, thus featuring high potentials for communications. Actually, a large amount of this potential is wasted. Users have to choose from the available communication systems by themselves and in general they are only connected to one access network at a time, not making use of the existing heterogeneity. Besides, most users will be overstrained in selecting the best path for their communications, in particular when they are mobile and the situation changes constantly. From this context, the idea of Always Best Connected (ABC)[11] arose. It defines the concept of a user-centric network selection providing seamless connectivity and always the best communication performance at any time.

Past research work on ABC communications focuses on scenarios where mobile devices are continuously connected to at least one network.However, this assumption does not hold for real world scenarios. A city dweller, commuting to work by public transportation, will surely be able to choose between various network technologies and operators. A high density of solvent customers encourages operators to extend networks and provide high bandwidth and operational availability. But when leaving metropolitan areas, situations change considerably. Lower population density often means lower network coverage and lower stages of completion. Even GSM networks do not provide seamless outdoor connectivity over the whole country as coverage maps show [9]. GSM indoor coverage as well as EDGE, UMTS or even HSDPA availability is even lower. In the near future, this situation will not change fundamentally. Most upcoming technologies provide high bandwidths only in a limited area around the base station and not over wide areas.

These facts make clear, that the vision of being "always-on" is not practical in the current situation. Challenged communication paths with fluctuating characteristics, disruptions and probably long delays are common and the heterogeneity of networks will further increase. Thus, delay tolerant networking (DTN) aspects will play an important role for future Internet protocols [18].

Existing ABC approaches show at least three limitations relating to challenged communication scenarios: First, previous work chooses candidate networks only from the set of access networks being available at a time. But, various applications like e.g. e-mail or downloading a file in the background do not require an immediate data exchange. A user may accept a longer data transmission period (e.g. within 4 hours) if it is free of charge, instead of using a pricey GPRS link. Thus, even disrupted paths matter and could be a viable alternative in a network selection process.

MobiArch'09, June 22, 2009, Kraków, Poland.

Second, existing decision strategies target at selecting a single network either for the whole mobile device[24] or in an application-specific way[10]. But, both methods ignore that only a combined evaluation of application-specific requirements on the one side and application-spanning optimizations on the other side can really result in an efficient network selection strategy. Just this makes it possible to meet individual application requirements and minimize mutual interferences between concurrent applications.

As a third limitation, the potential of using multiple networks in parallel is left untapped. If the remaining capacities of specific communication links cannot meet performance requirements of running applications, path bundling, either for concurrent multipath transfer (CMT), or for separating up- and downlink on different access networks, is a reasonable option to increase the communication performance.

To our knowledge, none of the existing work combines these different aspects: an efficient network selection strategy taking multipath capabilities into account and dealing proactively with times of disconnections. The remaining part of the paper is organized as follows. After discussing why disruption tolerance matters, we would like to describe the requirements and challenges of rating disrupted paths in the decision process. Afterwards, in section 3 we will introduce our architecture for dynamic network selection, called muXer. Following, we present two major parts of this architecture: an arbitrating concept for decision making as well as the mechanisms for path management. Finally, the paper concludes with an outlook on our work in progress.

2. BRINGING ABC AND DTN TOGETHER

The DTN paradigm faces new challenges to network selection strategies since it does not match the notion of the common Internet Architecture assuming almost persistent end-to-end connections between the communication partners. The first task that needs to be addressed is the description of links with long delays to make them ratable. Users or applications experience disruptions as a very high delay. Obviously, rating the time a network may remain available (e.g. based on the received signal strength) is far easier than predicting the time a certain network will become accessible again. Further, it is difficult to estimate the achievable performance on that future link. Thus, context information is inevitably needed to make reasonable assumptions. In the following, we propose three possible measures for evaluating a path with respect to delay tolerance aspects:

Availability: To estimate the data delivery time over a disrupted path, we need to know when the path will be accessible again and how long the connection will last. The remaining time an active path may be available can be predicted based on the radio signal strength. But if the path is disrupted, no such information is available. Thus, context information about the locations and coverage areas of candidate access networks is needed. This information may be made available by network operators, extracted from databases like e.g. JiWire [13] or derived from own experiences in the past. Using such context sources in conjunction with information about the current location of the mobile device, its speed and perhaps its destination, it is possible to draw reasonable conclusions about the future availability of candidate access networks. This does not provide any guarantees about the actual availability as well as the perceived performance at that time. On the other hand, it may happen as well that further unknown candidate networks become available in between. Balancing all these aspects and analyzing their effects on the decision process and the resulting communication performance will be one of the main focuses in our further research activities.

Reliability: The estimation of a network's availability becomes imprecise if context information is not obtainable or detailed enough. A measure of the path's predictability can help in such situations. In everyday life, users tend to take the same routes (e.g. from home to work and back) at similar times. Even without location information, a mobile device may collect statistical data about when specific access networks are available. An interpretation of this data set allows for finding dependencies on the availability of networks (e.g. on average, network B appears 10 minutes after the coverage area of network A was left) or the availability cycles of an access network. In the latter case, communication paths may be rated whether their availability is almost durable, scheduled to fixed times (e.g. Mondays and Fridays at 9:30 am), periodic (e.g., every 10 hours) or completely opportunistic. [16] shows that the prediction of the upcoming bandwidth in such scenarios is possible at a quite high accuracy. Nevertheless, this gives only a sight on the first hop along the path to the destination. In a store-carry-forward or MANET networking environment, where a mobile device has no direct connection to an access point, it is reliant on intermediate nodes which also may suffer from disconnections. Thus, determining a measure for the chance of data packets to be delivered to the destination or the access point within time could be a further optimization.

Delay: The main interest of users and applications is not only the time data can be sent, but also the time it is delivered to the destination. If we only consider the first hop to be challenged, this delay will typically be dominated by the time it takes for an access network to become available. In other scenarios with multiple challenged hops, the end-to-end transmission delay will carry a higher weight. Measuring this delay at the sender requires feedback from protocol instances at other nodes along the path. For acknowledged data transmission, the perceived RTT will only provide a very coarse grained clue on the delay. Especially in mobile environments, the path to the destination may feature clearly different characteristics than the return path. In addition, the perceived performance will vary over time. Thus, efficient and reliable path estimation requires novel protocol mechanisms (e.g. based on routing information).

Although these metrics can help to predict the time data will be delivered or at least sent, they do not allow for estimating of the network's expected QoS characteristics. Getting this information without collaboration with either infrastructure components or other mobile users being connected to the specific networks and reporting performance measures is hardly possible. On the other hand, the implementation of cooperative mechanisms requires a high penetration of participating nodes to provide a sufficient large data set. Thus, their deployment relies on standardized and provider-independent mechanisms.

Besides, a further challenge is on the application side. Applications should be able to state their disruption tolerance. Obviously, sending e-mails is very disruption tolerant, since it operates asynchronously and e-mails can even be easily transferred hop-by-hop in a store-carry-forward environment. Even an interactive application like web browsing may tolerate certain delays that are mainly limited by the user's patience of waiting for a web site to appear. For live video streaming, short disruption might be tolerable, if the application uses a playout buffer that is able to bridge a certain time gap. In contrast, interactive real-time applications like e.g. VoIP may not be disruption tolerant at all, since long delays lead to a loss of interactivity between the users. As a measure, applications may define during which time period data needs to be delivered or acknowledged. In addition, they may state the time potential disruptions may take without affecting the operation or whether they can cope with opportunistic links.

Finally, a network selection strategy needs to balance the delay tolerance of an application with the characteristics of challenged communication paths to decide, whether data has to take an available network path or may also be delivered later. Of course this may take the risk, that predicted future connectivity of a network may not come true or boundary conditions while waiting for a network to appear. To aggravate the situation, not only the first hop might suffer from disruptions, but also intermediate ones towards the destination (e.g. when assuming a store-carry-forward network paradigm). The evaluation of dependencies and effects of such decisions is subject of ongoing work.

3. THE MUXER ARCHITECTURE

Based on the mentioned ideas and concepts we developed a modular and flexible architecture for efficient and dynamic network selection called muXer. It is made up of five main modules as depicted in figure 1.

The *Link Monitor* observes the status of possible communication paths and determines the current network characteristics. Especially parameters regarding the networks' availability, communication performance and security mechanisms are of interest for the decision process. In general, we can distinguish between three possible ways of obtaining measures: active probes, passive traffic monitoring and information provided by the network itself (e.g.[1]. The first option causes additional overhead and requires support either by the communication endpoint or e.g. dedicated reference server in the Internet [16]. In muXer, we focus on the last two data sources since they manage on local information of the mobile device, although the data accuracy may be slightly worse. Another disadvantage compared to periodic probes is the lack of information on path not being used. In this case, we have to rely on past observations for that network and link layer information. The whole process of network monitoring runs asynchronously and independently from the actual decision process. Thus, the measures for each candidate network are updated continuously. To allow for comparing network technologies with clearly heterogeneous characteristics with each other, the Link Monitor maps technology-specific parameters onto generic input parameters. It provides a universal interface to the decision engine and hides technology-specific characteristics of the underlying communication networks. In addition, it is also responsible for predicting the future trend of specific parameters e.g. based on previous measurements, context information and the mobility of the mobile device. This additional

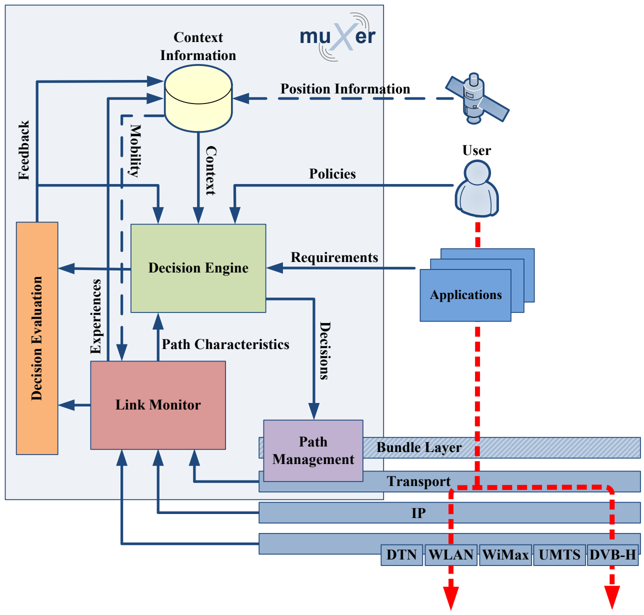

Figure 1: The muXer Architecture

data is stored in the *Context Information Storage* including e.g. information on services and service costs of specific network providers. In addition, location-specific data like the position of WLAN hotspots can improve the prediction accuracy. The most crucial part of the architecture is the *Decision Engine*, where the selection strategies are implemented. An application registers at the Decision Engine with its specific requirements to the underlying network. Together with user-defined policies on e.g. the maximum costs or the application priority, these requirements are mapped onto available or upcoming communication opportunities. Details on this process are described in section 4. It runs periodically and passes resulting decisions to the *Path Management*. This module is responsible for handling the different network connections and enforcing decisions being made. In the first step, we only control the first hop towards the Internet, since today's Internet architecture does not allow for explicitly selecting a path from source to destination (except for some specialized source routing mechanisms in e.g. MANETs). Future work will analyze possibilities of making path decisions at each individual DTN node along the path. Finally, the *Decision Evaluation* module introduces self-optimization capabilities, by verifying whether the decisions made map onto the expected system behavior. This is another asynchronous process that in case it discovers any optimization potentials or outdated context information is able to adapt decision parameters or to update context data.

In the following, we focus on two main aspects of the architecture: the strategy for network selection as well as the mobility management.

4. AN ARBITRATING CONCEPT FOR DECISION MAKING

Various approaches have been proposed to realize dynamic network selection in the context of ABC scenarios. Trivial strategies only use very simple decision criterions like signal strength([14, 4]). Class-based schemes ([3, 7]) categorize applications or services and allow for separating different user and application requirements, but only at a very coarse-grained level. Policy-based systems ([12, 22]) enable a fine-grained description of requirements and constraints, but in general they cannot deal well with vague information and the balancing of alternative solutions can be complex. These limitations overcome intelligent control techniques like fuzzy systems ([23]) or neural networks ([8]) that are fault-tolerant and work on imprecise data. But, they get very complex with increasing numbers of parameters and (as policies) strongly rely on the knowledge of experts that parameterized the system. Finally, there is work on analytic models like utility functions ([17, 5]) or multi-criteria decision analysis. But again, these techniques do not cope with imprecise information and "soft" decision criterions.

In summary, existing approaches commonly suffer from a degree of flexibility in combining highly heterogeneous characteristics and optimization goals. But this is a central requirement for an efficient decision strategy. It must provide a modular structure and has to be able to adapt to future communication systems and new optimization goals. Most existing work only focuses on either an application- or device-oriented network selection. However, both approaches either neglect cross-application interferences or individual requirements. To overcome these limitations, the overall decision process in muXer is split-up into two phases: an application-oriented rating of networks followed by an application-spanning optimization.

We base the muXer decision process on an arbitrating concept derived from the domain of mobile robot navigation [20] that is designed for scenarios requiring real-time responsiveness and dealing with uncertain information. The reason for this was the very high level of dynamics and flexibility. In contrast to any kind of expert system, the resulting decisions are clearly less predictable. Simultaneously, this is one of its main strengths. It allows for coping with unknown and unforeseen situations, finding unconventional solutions to the given decision problem. Since the concept does not make any restrictions to input or output parameters, even highly heterogeneous optimization goals can be considered collectively. It relies on the principle that various desired system behaviors are modeled as individual decision modules. Each of these modules corresponds to an optimization goal or specific communication characteristics. It rates independently and asynchronously for or against the suitability of candidate networks. The semantics of ratings are standardized. Votes have to be in an interval $[-M, M]$. $-M$ and M imply a strong support or decline of a network respectively, while 0 indicates a neutral position of the module. An arbiter implements a higher-level reasoning by casting and weighting all votes to get the overall rating. This avoids drawbacks of methods that jointly evaluate different goals. They average specific ratings which may lead to decisions being suboptimal to any of the individual goals.

During the first phase of the decision process, an application arbiter collects the votes of all application-specific optimization goals for each possible candidate networks. At this, candidate networks are not limited to common infrastructure networks like UMTS or WLAN hotspots, but may also cover specific networking paradigms as MANETs or DTNs. The Arbiter determines the local optimum from each application's point of view.

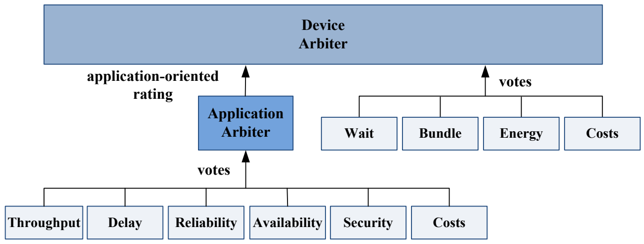

Figure 2: muXer Arbitrating Concept

In the second phase, those ratings are combined by analyzing, whether the joint requirements of a set of applications wanting to use a specific path comply with the network's total QoS offer. If that is not the case, it is necessary to assign a subset of these applications to a different network. Several optimization techniques might be used for this. In the first step, we rely on user defined application priorities. Further optimizations may try to maximize the overall voting of the selected configuration. In addition, global reasoning criterions for minimizing the overall monetary and energy costs of data transmission are considered. This way, it is assured that available networks are used to capacity without breaking the user-defined requirements.

Finally, the Decision Engine assigns to each application flow one of three possible actions:

- use ONE path
- use MULTIPLE paths
- WAIT for an upcoming transmission opportunity

While the first action is straightforward, the other ones require additional support by networking protocols beyond vertical handover mechanisms. For muXer, we implemented concurrent multipath transfer functionalities at the transport layer and support disruption-tolerant applications by using the Bundle Protocol[21]. Both aspects will be discussed later in section 5.

4.1 Decision Criterions

After discussing the general decision process, we now introduce the set of decision goals being considered for muXer as shown in figure 2.

The first two modules deal with typical QoS aspects: *throughput* and *delay*. Both are essential when trying to find the best network for an application. The throughput is characterized by the data rate that may be achieved over a specific path. The bandwidth provided by a communication system is just one aspect, since it only indicates the maximum possible data rate at the lower layers. It does not reflect performance limitations due to channel contention. Thus, we also want to consider the transport layer throughput. This provides a better indication on the experienced performance supposing that the bottleneck is located at the first hop. For estimating the delay, we measure round trip times of ongoing or past connections. Again, the same assumption as for the throughput estimation holds. In contrast to existing work, we do not want to make decisions purely on the current values of these measures, but also their progression in the past. From this, we can derive a future trend. Adding context information about, e.g., upcoming access networks, we can also predict future capabilities and times of disruptions. Thus, the system allows for making predictive decisions. Besides, we integrate measures on the *reliability* of a path into our decision concept. This aims at describing how stable a specific network is and how reliable data is transferred (in terms of packet loss or the chance to be delivered). Additionally, the likeliness of disconnections, upcoming connections and performance changes is used to minimize the number of handovers for applications with long-lasting and continuous data exchange. A further decision aspect is the *availability* of a network, as extensively discussed in section 2. Based on signal strength measurements as well as on location and context information, it is possible to determine the time to connection or disconnection of a path and thus to rate its suitability for an application. The remaining criterions cover the *security level* of a path as well as the *monetary costs*. For some applications, the user may define a minimum security level that should be met. In addition, users may not always demand for the cheapest service. When having an important video conference, they may be willing to spend more money to have an undisrupted and well performing conference, while running a download in background should be as cheap as possible.

All of these aspects relate to the first phase of the decision process. Further goals are defined for the application-spanning optimization. They vote for specific configurations that arise when combing the results of the first phase. On the one hand, *monetary cost* play an important role here as well, together with *energy consumption* arising by using multiple interfaces. This way, it may happen, that not the best, but second best path is selected for an application if it is only slightly worse in terms of the decision results, but allows for reducing the number of active interfaces or overall costs. Additional modules propose which applications should send data over *multiple paths* or have to *wait* with their transmission due to limited network resources. The constraints for the former module depend on ongoing evaluations of potentials and limitations of path bundling mechanisms. One main issue of CMT is reordering, that have to be dealt with in cases where ordered delivery is important. Our preliminary results show, that the main limitation is not the number of bundled paths in general, but the reordering efforts if paths differ clearly in their characteristics. An increasing delay-spread between the paths complicates the scheduling and requires higher efforts for buffering at the sender and/or receiver. Details on this tradeoff will be subject of a separate publication. Finally, the latter module has to reason about whether an upcoming communication opportunity is a viable option from the current point of view.

So far, we implemented first prototypes of different decision modules. Future work will include the evaluation of the interaction of different criterions and the benefit of different levels of context information on predicting upcoming changes of the communication environment. Finally, we plan to develop meaningful measures to analyze and compare the suitability of strategies for dynamic network selection.

5. PATH MANAGEMENT

The decision engine requires support by networking protocols to enforce decisions being made. These functionalities are bundled in the path management module that implements suitable interfaces and protocol optimizations. The main tasks of this module are:

- seamless vertical handovers (VHOs)
- concurrent multipath transfer (CMT) and
- handling of disruptions / long delays

For implementing VHOs, various approaches at different layers of the ISO/OSI protocol stack have been proposed. Most common mechanisms are mSCTP, MIP, HIP or IEEE 802.21. In addition, there has been a lot of discussion on which layer is best for mobility management tasks[19]. In the scenarios considered for muXer, protocol mechanisms must allow for separation and individual handling of application flows. In addition, when e.g. switching between different networks, communication characteristics may change clearly, affecting flow and congestion control mechanisms of transport protocols. For this reason, we decided to implement the muXer VHO mechanism at the transport layer [15]. It is based on mSCTP that provides multihoming support innately and thus is able to handle multiple IP addresses for one connection. Such a solution is independent of any additional mechanism and infrastructure components. Hence, it does not rely on support by specific network providers. In addition, transport layer approaches offer the possibility for additional optimizations of congestion and flow control mechanisms in the event of handovers, if information on the characteristics of the new path is available.

The CMT mechanisms are integrated into SCTP as well. Our preliminary results show that in various scenarios even with heterogeneous communication paths, bundling achieves almost the full capacity that is shared by the individual paths. However, as mentioned earlier, there are strong limitations on the difference of the paths' characteristics.

Finally, muXer aims at adopting delay tolerance mechanisms. On the one hand, it would be possible to hide disruptions of the first hop along the communication path, by freezing the state of the transport layer protocol until a path gets available again[2]. This is a viable solution that could easily be implemented into the muXer path management. Nevertheless, it is bound to the assumption that only the first hop is challenged. Instead, the Bundle Protocol (BP)[21] provides additional support for store-carry-forward networks by splitting data up into small bundles that are handled individually. For muXer, we focus on this second option, whereas the BP acts as session layer above the transport protocol. Thus, an individual path selection even at intermediate hops carrying the bundle would be possible and transport layer mechanisms remain unaffected by discontinuous connections. A further possibility would have been the concentration of handoff and multipath procedures at the bundle layer. However, this requires all applications to support the bundle protocol. The session layer would have to handle multiple sockets and especially with respect to CMT, it may not necessarily be able to possess enough information to schedule data along the paths as efficiently as a transport layer solution is able to do. Thus, we decided to pursue a split approach, while the close integration could be a promising option for future research.

6. CONCLUSIONS AND OUTLOOK

The muXer architecture aims at overcoming limitations of existing strategies for dynamic network selection, by considering disrupted and bundled paths as candidates for data transmission instead of focusing only on single path transfer along available networks. This way, communication capabilities can be utilized more efficiently resulting in a higher user satisfaction. The decision process is made up of two phases. At first, it determines application-specific optima, followed by an application-spanning downstream refinement. Thus, muXer tries to find a global optimum for all applications running on the mobile device. Besides, we introduced an arbitrating concept for decision making that is characterized by high flexibility and proposed a set of candidate decision criterions. The implementation of the muXer architecture is still work-in-progress. The next steps of our work will include a closer investigation of the decision process by analyzing the effects of different decision criterions on the performance of the selected paths. Moreover, we will integrate muXer and a DTN implementation[6] to evaluate mechanisms and measures for delay tolerant networking.

Acknowledgment

Sven Lahde would like to thank Deutsche Telekom Stiftung for their generous support of this research work. In addition, we would like to thank all participants of the Dagstuhl seminar 09071 on Delay and Disruption-Tolerant Networking (DTN) II for fruitful discussions and new impulses.

7. REFERENCES

[1] IEEE Standard 802.11k-2008 : Radio Resource Measurement of Wireless LANs, June 2008.

[2] M. Bechler, S. Jaap, and L. Wolf. An Optimized TCP for Internet Access of Vehicular Ad Hoc Networks. In *Proc. of Networking 2005*, Waterloo, Canada, May 2005.

[3] T. Bejaoui and N. Nasser. Handover and Class-based Call Admission Control Policy for 4G-Heterogeneous Mobile Networks. In *Proc. of AICCSA 2008*, pages 373–380, Apr. 2008.

[4] H. Bing, C. He, and L. Jiang. Performance Analysis of Vertical Handover in a UMTS-WLAN Integrated Network. In *Proc. of the 14th IEEE Personal, Indoor and Mobile Radio Communications (PIMRC 2003)*, volume 1, pages 187–191, Sept. 2003.

[5] L.-J. Chen, T. Sun, B. Chen, V. Rajendran, and M. Gerla. A Smart Decision Model for Vertical Handoff. In *Proc. of the 4th ANWIRE International Workshop on Wireless Internet and Reconfigurability*, Athens, Greece, 2004.

[6] M. Doering, S. Lahde, J. Morgenroth, and L. Wolf. IBR-DTN: An Efficient Implementation for Embedded Systems. In *Proc. of the Third ACM Workshop on Challenged Networks (CHANTS '08)*, pages 117–120, San Francisco, CA, USA, Sept. 2008.

[7] G. Fan and X. Lu. Novel Service Oriented Handoffs in Heterogeneous Wireless Networks. In *Proc. of the 2nd Int. Conference on Quality of Service in Heterogeneous Wired/Wireless Networks (QSHINE '05)*, Aug. 2005.

[8] L. Giupponi, R. Agustí, J. Pérez-Romero, and O. Sallent. A Fuzzy Neural JRRM in a Heterogeneous Scenario Supported by Prediction Strategies for Horizontal and Vertical Handovers. In *Proc. of the IEEE International Conference on Fuzzy Systems*, pages 655–662, 2006.

[9] GSM Association. GSM World – Coverage Maps. http://www.gsmworld.com/technology/roaming/gsminfo/. visited in Feburary 2009.

[10] X. Guan, R. Tang, S. Bai, and D. Yoon. Enhanced Application-Driven Vertical Handoff Decision Scheme for 4G Wireless Networks. In *Proc. of the Int. Conference on Wireless Communications, Networking and Mobile Computing (WiCom 2007)*, pages 1771–1774, 2007.

[11] E. Gustafsson and A. Jonsson. Always Best Connected. *IEEE Wireless Communications*, 10(1):49–55, Feb. 2003.

[12] P. A. A. Gutierrez and I. Miloucheva. Automated QoS policy adaptation for heterogeneous access network environments. In *Proc. of ICSNC 2007*, Aug. 2007.

[13] JiWire, Inc. Wi-Fi Advertising Network, Wi-Fi Finder and Hotspot Directory. http://www.jiwire.com. visited in March 2009.

[14] J.-M. Kim and J.-W. Jang. Low Latency Vertical Handover Using MIH L2-Trigger Algorithm in Mobile IP Networks. In *Parallel and Distributed Processing and Applications*, volume 4742 of *LNCS*, pages 707–718. Springer, 2007.

[15] S. Lahde, M. Doering, and L. Wolf. Dynamic Transport Layer Handover for Heterogeneous Communication Environments. *Computer Communications*, 30(17):3232 – 3238, 2007. Special Issue on Concurrent Multipath Transport.

[16] A. J. Nicholson and B. D. Noble. BreadCrumbs: forecasting mobile connectivity. In *Proc. of the 14th ACM international conference on Mobile computing and networking (MobiCom '08)*, pages 46–57, New York, NY, USA, Sept. 2008. ACM.

[17] O. Ormond, J. Murphy, and G.-M. Muntean. Utility-based Intelligent Network Selection in Beyond 3G Systems. In *Proc. of the IEEE International Conference on Communications 2006 (ICC '06)*, volume 4, pages 1831–1836, June 2006.

[18] J. Ott. Delay Tolerance and the Future Internet. In *Proc. of the 11th International Symposium on Wireless Personal Multimedia Communications*, Sept. 2008.

[19] M. Ratola. Which Layer for Mobility? – Comparing Mobile IPv6, HIP and SCTP. Seminar on Internetworking, Helsinki Univ. of Technology, 2004.

[20] J. K. Rosenblatt. *DAMN: A Distributed Architecture for Mobile Navigation.* Phd thesis, The Robotics Institute, CMU, Jan. 1997. CMU-RI-TR-97-01.

[21] K. L. Scott and S. Burleigh. Bundle Protocol Specification. Internet Engineering Task Force Request for Comments 2960, Nov. 2007.

[22] J. Strassner, D. Raymer, and S. Samudrala. Providing Seamless Mobility Using the FOCALE Autonomic Architecture. In *Proc. of NEW2AN 2007*, volume 4712 of *LNCS*, pages 330–341. Springer, 2007.

[23] A. Wilson, A. Lenaghan, and R. Malyan. Optimising Wireless Access Network Selection to Maintain QoS in Heterogeneous Wireless Environments. In *Proc. of the Wireless Personal Multimedia Communications 2005 (WPMC05)*, Aalborg, Denmark, Sept. 2005.

[24] A. H. Zahran, B. Liang, and A. Saleh. Signal threshold adaptation for vertical handoff in heterogeneous wireless networks. *Mobile Networks and Applications*, 11(4):625–640, Aug. 2006.