Budapest University of Technology and Economics
Faculty of Electrical Engineering and Informatics
Department of Telecommunications and Media Informatics

New Methods for Efficient Data Transport in Future Networks

Summary of the Ph.D. Dissertation

Author:
Zoltán Móczár

Supervisor:
Dr. Sándor Molnár
Associate Professor

Budapest, Hungary
2015

1 Introduction

From the very beginning of the Internet, traffic congestion has been recognized as an undesirable phenomenon that must be avoided in order to maintain stable operation. Congestion occurs when the aggregate demand for a resource exceeds its capacity, which typically leads to significant performance degradation in communication networks. As a solution, the *Transmission Control Protocol (TCP)* [1] was introduced in 1981 and it defined a set of mechanisms to prevent such adverse events by adjusting the transmission rate based on various observations. Closed-loop congestion control performed by TCP has proven to be a successful approach, but several versions have been developed over the past decades to satisfy the ever-changing requirements of heterogeneous network environments [2]. However, in the recent years it became apparent that the continuous refinement of TCP cannot follow the incredibly fast evolution of technologies and applications, as well as the increasing user demands.

It is clear that emerging paradigms like cloud computing and software-defined networking, and what is more, the upcoming era of 5G mobile networks and Internet of Things will require much more efficient *transport methods governed by fundamentally different principles*. Taking into account these trends, it is natural to ask if congestion control is indispensable to ensure reliable communication. While the research community is urged to find the answer, only a few papers elaborate on this challenging issue. For example, the authors of [3] argue that it may not be necessary to keep the network uncongested to yield good performance, and a greedy transport protocol has the potential to outperform TCP. In 2007, a research plan [4] published by the organization of GENI (Global Environment for Network Innovations) recommended the omission of TCP's congestion control mechanism and suggested to use error correction techniques instead so as to cope with packet loss. The validity of this approach is supported by a recent study [5] claiming that congestion collapse does not happen in many cases even if no congestion control is applied at the network endpoints. However, to reveal whether the Internet can work efficiently without the key functionality of TCP, extensive research needs to be conducted.

2 Research Objectives

This dissertation addresses the challenging task of building an Internet architecture without congestion control while still ensuring reliable end-to-end communication. My research goal was to work out new methods for efficient data transport and to give an in-depth analysis of the proposed solutions.

My main contributions are the following:

- In the first part of the dissertation, I investigate a novel digital fountain based communication paradigm, which consists of a transport mechanism called *Digital Fountain based Communication Protocol (DFCP)* and an underlying network architecture where fairness is managed at the routers instead of endpoints. The design and operating principles are discussed in detail together with the potential benefits of the approach.
- The second part presents a comprehensive performance analysis of DFCP and TCP carried out in a multi-platform evaluation framework using packet-level simulations and testbed measurements. The experiments cover various network topologies and settings to reveal different properties such as goodput performance, buffer demand, flow transfer efficiency, fairness, network utilization and scalability.
- Since today's network traffic is highly dynamic, the third part is completely devoted to the characterization of the two transfer paradigms under rapidly changing conditions, with a special focus on the features of stability, convergence, responsiveness and saturation time.
- Finally, a bandwidth estimation method is proposed for mobile networks, which can estimate the available bandwidth by exploiting the user-generated downlink network traffic with a negligible additional load. The core of the solution is a busy period detection algorithm capable of providing reasonable accuracy in spite of quick and high variations often seen in mobile data networks.

I believe that the findings presented in this dissertation can greatly promote the research on alternative data transfer methods as they shed light on the important fact that communication without the need of controlling the congestion is possible and merits further investigation.

3 Methodology

The transport mechanism of DFCP has been implemented in the Linux kernel as in the case of TCP. To deeply investigate our proposal, measurements were conducted both in a *simulation framework* and in *real test networks*. As a *simulation tool*, the ns-2 packet-level simulator [6] was used with the Network Simulation Cradle (NSC) [7] extension modified in C++ to properly handle the kernel implementation of DFCP. Testbed measurements were performed in different *laboratory network configurations* and in a remote *network emulation environment called Emulab* [8]. In order to get sound results, these three platforms have been extensively cross-validated, which is unique in a sense regarding the literature of transport protocols. The available bandwidth estimation algorithm was evaluated on packet traces gathered from a 3G mobile network by using a *realistic traffic emulator* [9].

4 New Results

4.1 An Evaluation Framework for New Transport Protocols

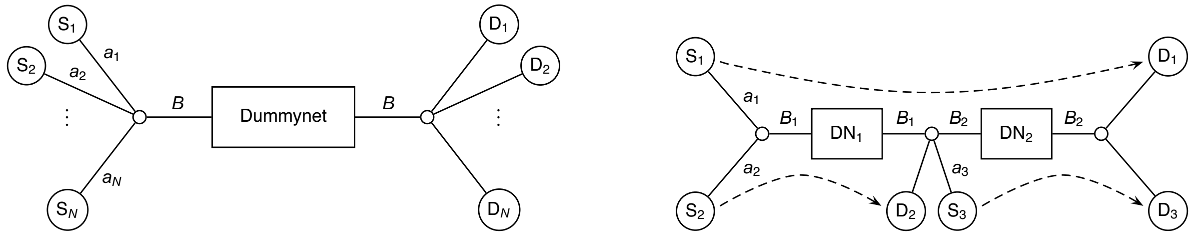
Although most researchers evaluating protocols and algorithms rely *solely on simulation or testbed measurement results*, in fact, these two methodologies are necessary to be applied together especially in the case of the investigation of fundamentally new paradigms. On the one hand, the main risk of using only simulations is that these environments are far from realistic in most cases, thus many real-world factors can easily be neglected [10, 11]. On the other hand, performing only testbed measurements can also lead to the loss of generality, because special hardware components can significantly affect the results. In addition, building a complex network testbed is a time-consuming process, and measurements are often very difficult to repeat [12, 13].

Thesis group 1. ***I have created a multi-platform evaluation framework, which enables the performance analysis of both the congestion control and digital fountain based transport mechanisms in a reliable and consistent way [C3, J1, J2].***

Thesis 1.1. *I have built a laboratory testbed to investigate the operation and properties of transport protocols. I have designed and constructed network topologies suitable for studying various protocol features. I have made possible the performance evaluation of data transfer methods in environments with different delay and loss characteristics.*

Thesis 1.2. *I have constructed a testbed in the Emulab network emulation environment maintained for research purposes in order to investigate complex network topologies. I have designed networks consisting of multiple nodes and links, which enable a more thorough understanding of protocol behavior under realistic conditions.*

Thesis 1.3. *I have created a simulation environment aimed at investigating measurement scenarios that cannot be performed in network testbeds. I have made possible the analysis of the novel transport protocol without congestion control in the ns-2 network simulator*



(a) Dumbbell topology

(b) Parking lot topology

Figure 1. Investigated network topologies

by modifying the NSC extension, which has only supported the simulation of TCP-based transport protocols.

The performance of data transfer mechanisms was evaluated on the *network topologies* of Figure 1 frequently used in the literature for investigating protocols and algorithms [14]. The dumbbell topology (Figure 1a) made it possible to study the operation of transport protocols and their features in the case of different network settings (bandwidth, delay, packet loss, buffer size, etc.). Moreover, other important properties such as fairness and scalability were also examined on the topology illustrated in Figure 1a for concurrent flows. The parking lot topology (Figure 1b) was used to analyze the behavior of protocols in multi-bottleneck networks.

The *integrated test environment* makes it possible to perform complex analysis by exploiting the benefits of each platform. Furthermore, it is able to produce reliable and consistent results, which is essential to draw right conclusions. Network Simulation Cradle (NSC) is an extension of the ns-2 simulator, which supports the simulation of network stacks of many operating systems (e.g. BSD and Linux) directly from the kernel code. NSC is capable of producing extremely accurate results, but it only enables the investigation of TCP versions and new TCP-like transport mechanisms. Since the proposed data transfer method is based on a fundamentally different principle compared to that of TCP, several protocol-specific modifications have been made to integrate the source code of DFCP into the framework. Figure 2 shows the main elements of the integrated simulation environment. The basic models of transport protocols are defined in ns-2 including all necessary parameters. The two simulator components (ns-2 and NSC) communicate through a common interface, provided by a C++ shared library. In case of an interaction, ns-2 invokes the related protocol-specific methods in NSC, which then call the proper kernel function. NSC can handle multiple copies of the global data used by the network stack making possible to run independent instances of protocol implementations within the same simulation scenario.

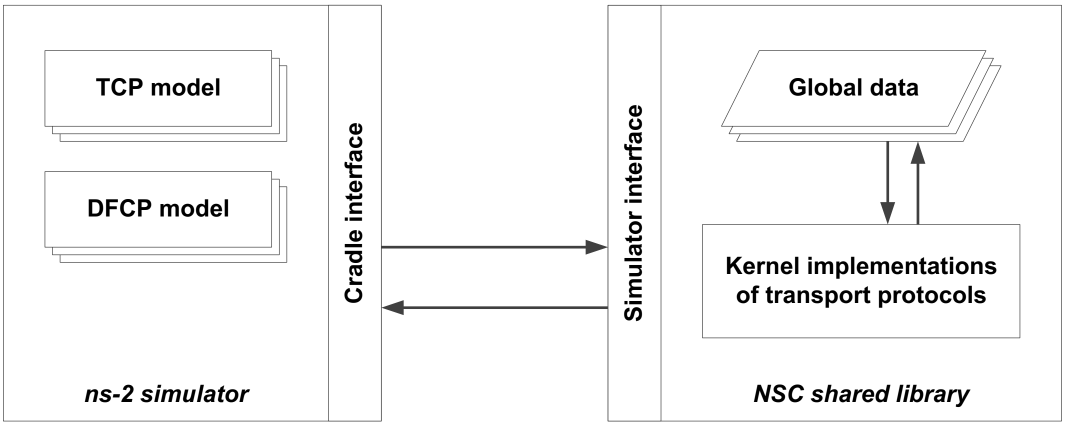


Figure 2. The DFCP-compatible integrated simulation framework

4.2 Analysis of Digital Fountain Based Communication

The key component of the data transfer paradigm presented in this dissertation is the DFCP protocol, which recovers lost packets by using *digital fountain codes* [15]. The main advantage of fountain coding is that it can generate an arbitrary length encoded stream from the original message, then decoding can be successfully performed if any part of this stream with size slightly larger than the original message is available. DFCP applies Raptor coding [16], which is the most efficient error correction technique today due to its linear encoding and decoding complexity.

The *network architecture built upon the transfer mechanism of DFCP* is shown in Figure 3. Let assume that the sender wants to transfer a message of length k to the receiver, and the data is transmitted at the maximum possible rate without applying congestion control. The use of Raptor codes enables the sender to generate a potentially infinite stream of encoded bytes by adding a redundancy of $\epsilon > 0$, and the original message can be recovered at the receiver with high probability if any subset of size $\lceil(1+\varepsilon)k\rceil$ encoded symbols arrive. Since in the case of real applications the value of ϵ is typically about a few percent, this approach provides efficient communication between the nodes while the full utilization of network resources is guaranteed. To avoid starvation of competing flows, equal bandwidth sharing has to be ensured by employing *fair schedulers* in the routers. Several fair queuing algorithms (e.g. DRR [17]) are already available and configurable in current network devices providing scalable way [18] for resource sharing.

The flow chart of the *coding and data transfer process* can be seen in Figure 4. The data bytes received from the application layer are organized into message blocks, then DFCP performs LDPC (Low-Density Parity-Check) encoding [19], also known as precoding, by adding redundant bytes to the original data. These encoded blocks provide the input for the next phase, which is called LT (Luby Transform) encoding [20]. Theoretically, the LT

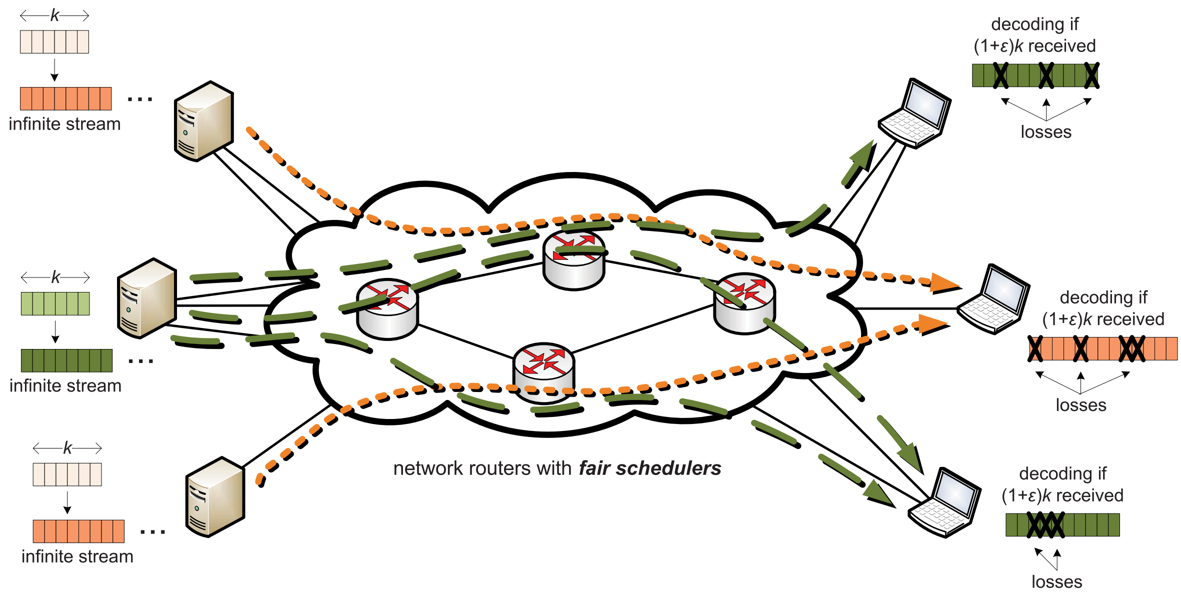


Figure 3. The network architecture built upon the transfer mechanism of DFCP

encoder can generate an infinite length encoded stream, but in practice, it can only be sent to the receiver in blocks. To this end, DFCP applies a *sliding window based flow control* mechanism intended to determine the maximum number of unacknowledged blocks in the network. The main goal is to prevent buffer overflows at the receiver side, but it works differently compared to the solution of TCP. In general, the size of the sliding window should be chosen such that it does not limit the transmission performance.

Thesis group 2. ***I have studied the operation and the fundamental properties of a novel transport protocol without congestion control called DFCP (Digital Fountain based Communication Protocol) in a multi-platform test environment [C3, C5, J1, J2].***

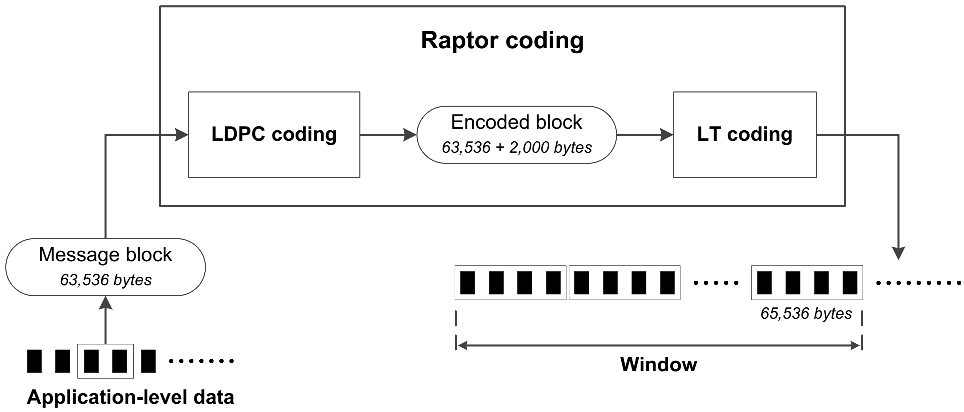


Figure 4. The flow chart of the coding and data transfer process

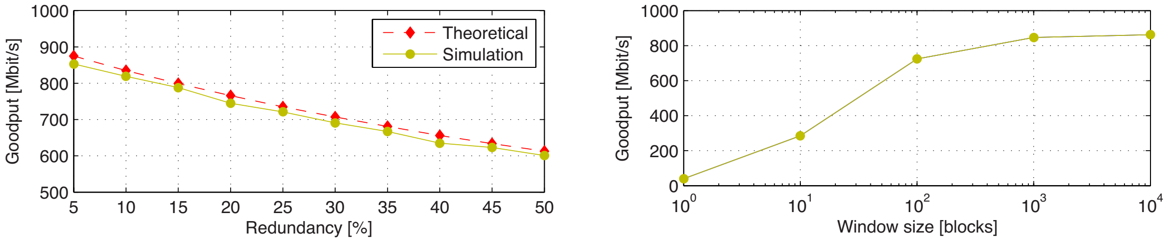
Thesis 2.1. *I have proved by extensive analysis performed on different network topologies that the prototype of DFCP can work efficiently in environments with a broad range of packet loss rates and round-trip times (0–50%, 0–500 ms). I have determined that the maximum achievable goodput is independent of the extent of packet loss and delay for values frequently occur in real networks (0–1%, 0–100 ms).*

Thesis 2.2. *I have investigated how the protocol-specific parameters like coding redundancy and window size affect the goodput performance, as well as the generated traffic characteristics. I have shown that increasing the amount of redundancy up to 50%, the transmission rate is decreased at most by 33%. Furthermore, I have verified that the use of a larger window leads to a more bursty traffic, hence in order to guarantee optimal operation, it must be adjusted to the lowest value where the flow control mechanism does not impose any limitation.*

To evaluate the performance of transport protocols, several well-known metrics can be found in the literature. One of the most widely used measures is *throughput*, which gives the amount of data transferred per second from source to destination. However, when comparing the efficiency of transport mechanisms based on different principles, it is better to investigate *goodput* instead of throughput, because goodput refers only to the useful data bytes excluding the protocol headers, the added redundancy and the coding overhead. The analytical calculation of the goodput can be obtained by the digital fountain based data transfer paradigm is feasible for the simple dumbbell topology (Figure 1a). Suppose that the bottleneck link with capacity c_B is fed by N senders connecting through the access links having capacities $c_1, c_2, \ldots, c_N$. Each sender transfers one flow simultaneously that results in N concurrent flows competing for the shared bottleneck capacity. Assuming that fair schedulers are used in the network routers, and the redundancy is denoted by ε, the goodput of flow i can be given as follows:

$$G_i = \begin{cases} \frac{c_B}{(1+\varepsilon_i)N} & \forall j : c_j \geq \frac{c_B}{N} \\ \frac{c_i}{1+\varepsilon_i} & c_i < \frac{c_B}{N} \\ \frac{c_B - \sum_{k=1}^{N} I_{\{c_k < \frac{c_B}{N}\}} c_k}{(1+\varepsilon_i) \sum_{k=1}^{N} I_{\{c_k \geq \frac{c_B}{N}\}}} & \exists j : c_j < \frac{c_B}{N} \text{ and } c_i \geq \frac{c_B}{N} \end{cases} .$$

The impact of DFCP parameters on the performance is illustrated in Figure 5 when redundancy and window size are adjusted to $\epsilon = 0.05$ and 1000 blocks, respectively (in



(a) Goodput in the function of redundancy

(b) Goodput in the function of window size

Figure 5. The impact of DFCP parameters on the performance

this case, $N = 1$ and $c_B = c_1 = 1000$ Mbps). We can observe that the decrease in goodput does not change linearly with increasing redundancy (Figure 5a), but the typical value is below 5%, which ensures high probability decoding in the case of recent Raptor codes. As the window size increases, the goodput gets higher until it achieves the maximum possible value (Figure 5b). However, the use of a larger window in DFCP leads to a more bursty traffic, hence it is practical to set the window size such that further increasing does not improve the goodput performance, or in other words, the flow control mechanism does not limit the transmission rate.

4.3 Performance Evaluation of Data Transfer Paradigms

In the current Internet, the Transmission Control Protocol (TCP) carries more than 80% of network traffic. TCP is a connection-oriented transport protocol that provides reliable data transfer in end-to-end communication. The most important feature of TCP is its congestion control mechanism, which is used to avoid congestion collapse [21] by properly adjusting the sending rate. The history of TCP dates back to 1981 when the official protocol specification was published by the IETF in RFC 793 [1]. Over the past three decades, a significant research effort has been devoted to TCP in order to meet the requirements of evolving communication networks. This process has resulted in countless TCP versions aimed to provide high performance in various environments [2].

In this dissertation, I compare DFCP to two widely used TCP versions, namely *TCP Cubic* and *TCP NewReno*. TCP Cubic [22], being an enhanced version of its predecessor, BIC TCP [23], serves as the default congestion control algorithm of Linux operating systems. BIC TCP was originally designed to solve the well-known RTT unfairness problem [24]. TCP Cubic simplifies the window control of BIC and it applies a cubic function in terms of the elapsed time from the last loss event, which provides good stability and scalability. TCP NewReno [25] is a newer variant of TCP Reno [26] intended to improve its performance when a burst of packets is lost. To achieve this goal, NewReno modifies

Reno's congestion control algorithm enabling faster recovery even from such undesirable events.

To deeply understand the operation of transport protocols, a comprehensive performance evaluation has to be carried out. The values of *QoS metrics* such as packet loss and delay can change in a wide range depending on the given environment, therefore a thorough analysis is required to reveal their impact on the performance. In the last decade, several researchers dealt with the issue of *optimal buffer sizing* as well, which has also great significance in the efficiency of transport protocols. Nowadays, router memories are considerably over-sized due to an outdated rule of thumb [27] leading to serious performance degradation in many cases as the consequence of increased end-to-end queuing delay [28]. At the same time, in all-optical networks only small buffers can be applied because of technological constraints [29], thus it is especially important for a data transfer mechanism to work under such conditions too.

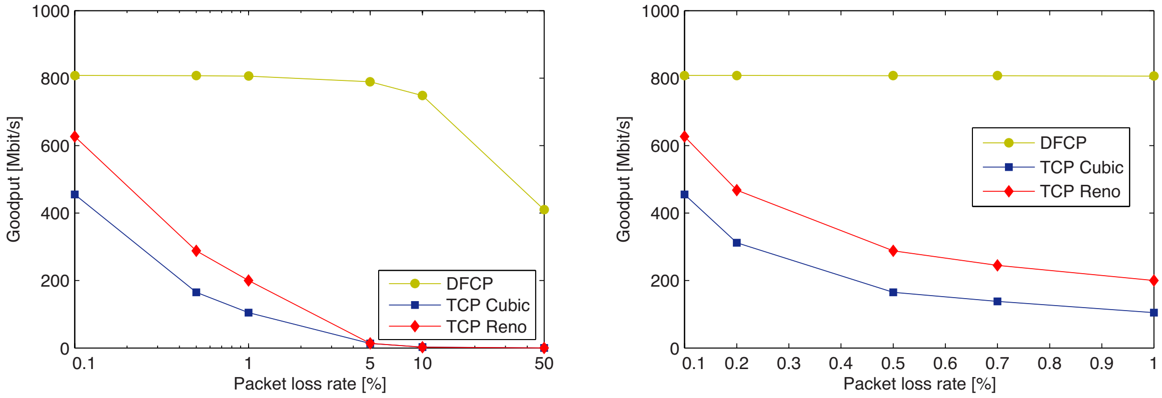
Most network operators choose over-provisioning as a way to satisfy resource demands even during the busy hours of the day [30], which is based on the assumption that quality issues can be readily addressed by allowing some excess capacity. However, this approach can often be very inefficient leading to degraded QoE, and for some environments, it is simply not a viable option due to various constraints. The *utilization*, provided by a transport protocol, is a key measure to support the proper design and sizing of networks. In this dissertation, network utilization is interpreted as the total mean utilization of all bottleneck resources (links and buffers).

Thesis group 3. ***I have revealed by a comprehensive analysis how DFCP performs in different network environments compared to TCP [C3, C4, C5, J1, J2].***

Thesis 3.1. *I have verified that DFCP provides faster data transmission than TCP even in the case of lossy and high-latency links (>50 ms).*

Thesis 3.2. *I have proved by simulations that, in contrast to TCP, the buffer size does not affect the performance of DFCP, thus it can work efficiently with small buffers (<100 packets) as well.*

Thesis 3.3. *I have shown that the transfer mechanism of DFCP keeps the queue length fairly stable, however, the congestion control algorithm of TCP causes significant fluctuation. While the extent of variation is typically a few percent for DFCP, regarding TCP the number of packets waiting in the queue can vary in the whole range determined by the buffer size.*



(a) A wide range of packet loss rates

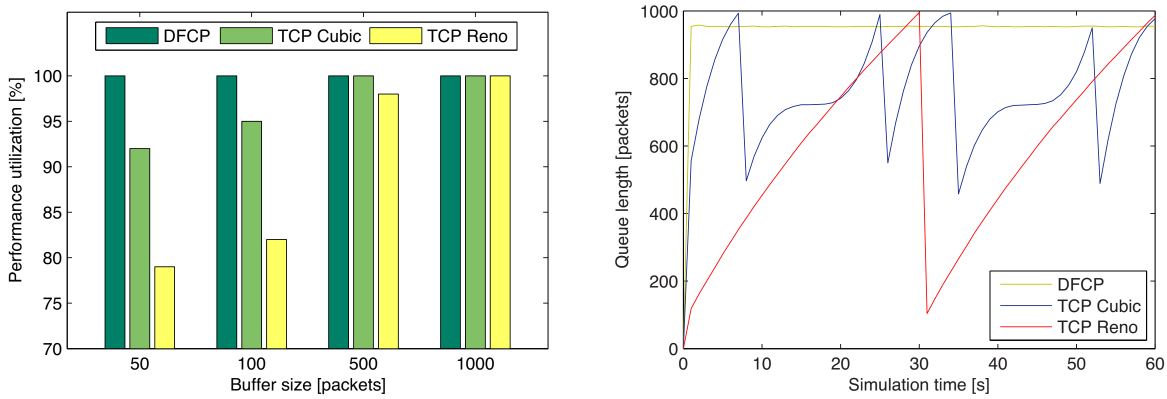
(b) A realistic parameter domain

Figure 6. The operation of DFCP and TCP in a lossy environment

Thesis 3.4. *I have determined by experiments carried out in complex networks that, with 90% probability, DFCP provides more than 80% network utilization while TCP works under the 80% level. I have also pointed out that, using tiny buffers (0.01 BDP), at least two times higher utilization can be obtained by DFCP compared to TCP.*

The operation of DFCP and TCP protocols in a lossy environment is shown in Figure 6. A great benefit of the digital fountain based transfer mechanism is the moderate sensitivity to packet loss both in the case of independent and bursty losses. Figure 6a illustrates the high resistance of DFCP over a wide range of loss rates with only a slight drop in performance, whereas TCP versions are almost unable to operate even if the ratio is a few percent. Moreover, Figure 6b shows that, assuming a realistic parameter domain, DFCP becomes insensitive to packet loss, hence the rate variation experienced in the case of TCP can be avoided leading to more predictable traffic characteristics.

Figure 7 demonstrates the buffer demand and usage of DFCP and TCP protocols for a link with a capacity of 1 Gbps and a delay of 10 ms. Figure 7a shows the performance utilization of the investigated transport protocols, which is interpreted as the ratio between the goodput can be obtained with a particular buffer size and the maximum goodput that can be achieved when the buffer size is set as high as to exclude it from the limiting factors. We can see that, with a buffer size of 1000 packets, each protocol is able to realize maximum performance utilization. However, by decreasing the buffer size the performance of TCP variants drops considerably. For example, with a small buffer of 50 packets, TCP Cubic and TCP Reno can work only at a reduced transfer rate, 92% and 79% of the ideal case, respectively. In contrast, DFCP can bring out the maximum performance not only for large buffers, but also for small ones, and thanks to this property the transport mechanism of DFCP is closely aligned to the concept of all-optical network-



(a) Buffer demand of protocols (b) Buffer occupancy in the function of time

Figure 7. Buffer demand and occupancy of DFCP and TCP

ing [29]. Figure 7b illustrates the utilization of the bottleneck buffer in the function of time. Considering the queue length dynamics, we can see that DFCP builds up the queue in a very short time and it works with a 95% mean buffer occupancy while TCP Cubic and Reno achieve only 75% and 58% of the available resource, respectively. It can also be observed that the queue length highly fluctuates in the case of TCP versions, but DFCP causes a moderate oscillation.

4.4 Dynamic Behavior of Transport Mechanisms

In the last decades, the characteristics of network traffic have changed considerably due to the evolving technologies and the diversity of applications. Today's Internet is a large-scale, highly dynamic network in which sudden variations are common due to topology and bandwidth changes. While a great portion of TCP evaluation studies deal with performance analysis solely in static environments, some researchers emphasize the importance of exploring how responsive a transport protocol is under rapidly changing conditions [31, 32]. In this dissertation, the *dynamic behavior of two fundamentally different paradigms* are investigated: TCP Cubic with FIFO queue management (*Congestion Control based Architecture, CCA*) and DFCP with DRR scheduling (*Digital Fountain based Architecture, DFA*). To this end, packet-level simulations were carried out focusing on the properties of *stability*, *convergence*, *responsiveness* and *saturation time*.

Network traffic is generated by heterogeneous applications that results in many concurrent flows traversing different network paths with multiple bottlenecks from source to destination. The transmission rates of these flows fluctuate rapidly since the currently used congestion control based transfer mechanism could not adapt to changing conditions as fast as needed. *Stability* is an important property from both traffic engineering and user experience point of views [33]. Rate variations often lead to the oscillation of queue length that can eventually cause buffer overflows. Such an undesirable behavior can result in the loss of synchronization among competing flows, periodic underutilization of link capacity and degraded quality of service. It is also crucial regarding efficiency how fast a flow can obtain its equilibrium rate or *converge* to the fair share in a dynamic environment.

Another key concern in the design of transport protocols is the ability to handle abrupt change of traffic conditions [33]. In a real network, competing flows governed by different transfer mechanisms often face with quick variations mainly originated from routing and bandwidth changes, or sudden congestion. *Responsiveness* is of high importance describing how fast and accurately a transport protocol can adapt to these environmental factors.

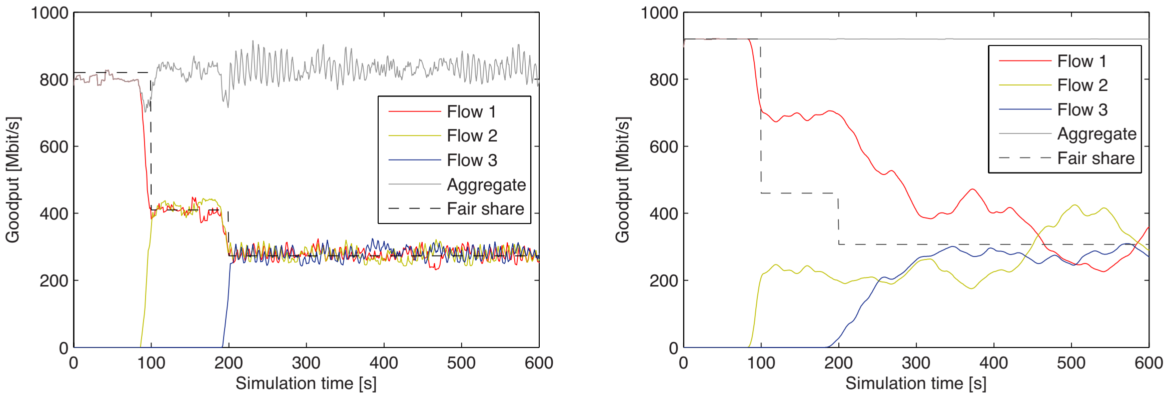
The operation of congestion control algorithms consists of two main transmission phases. In the initial phase, TCP gradually increases the sending rate until the bottleneck buffer is filled. Then, it is followed by an equilibrium state when the protocol achieves the maximum transmission rate and tries to keep it stable. The length of the transient phase highly determines the download efficiency of short-lived flows, therefore it can affect the quality of experience for many applications. In order to capture this behavior, a performance metric called *saturation time* is used [34], which can be given as the time elapsed from the starting of a flow until the first packet is dropped from the buffer. Queue saturation time is a good indicator of how fast a transport protocol can obtain its steady-state performance.

Thesis group 4. ***I have investigated the behavior of the digital fountain and the congestion control based data transfer mechanisms under dynamic network and traffic conditions [C1].***

Thesis 4.1. *I have verified by simulations that, while in the case of CCA the goodput of concurrent flows fluctuates less at small time scales, the operation of DFA is more stable in the long run.*

Thesis 4.2. *I have determined that DFA enables faster convergence (typically 1–2 sec) for the transmission rates of competing flows than CCA by an order of magnitude on average.*

Thesis 4.3. *I have shown by performance evaluation that DFA can guarantee much faster adaptation to the sudden change of the available bandwidth in comparison to CCA regarding both per-flow and aggregate traffic. The responsiveness becomes lower for CCA if smaller buffers (<1 BDP) are used, whereas it is independent of the buffer size for DFA.*



(a) Digital Fountain based Architecture

(b) Congestion Control based Architecture

Figure 8. Dynamics of concurrent flows started with different delays and their convergence to the fair share

Thesis 4.4. *I have proved that, in contrast to CCA, the queue saturation time provided by DFA is independent of the buffer size, the number of flows and the round-trip time.*

In Figure 8, the dynamics of three concurrent flows is illustrated when they were started with different delays, namely at 0, 100 and 200 sec. The goodput gives the current useful data transmission speed in one second resolution, and the curves were smoothed by using a 10 sec long moving window. We can observe that, for CCA, flows converge slowly to the fair share and then their goodput highly fluctuates around it. In the case of DFA, the convergence time is very low whereas the fluctuation around the fair share remains moderate. However, the transfer mechanism of DFA leads to a more bursty transmission than that of CCA, which is due to the impact of window size on the traffic characteristics.

4.5 Bandwidth Estimation in Mobile Networks

Bandwidth estimation has received considerable attention in the last decades due to its key role in many areas of networking [35] such as transport layer protocols, admission control, network management and multimedia streaming, just to mention a few. For example, transport protocols like TCP can use available bandwidth information to properly adjust the transmission rate, making possible the efficient utilization of network resources without causing congestion. Bandwidth estimation results also help network operators to identify the change of user demands by monitoring the network utilization and to plan capacity upgrades. *Estimating the available bandwidth* in mobile networks [36] is a serious challenge because of the continuously changing network conditions such as the location and motion speed of the mobile device, the number of active users in the current cell, or the signal strength.

Thesis group 5. ***I have designed a heuristic bandwidth estimation method for mobile networks and evaluated its operation on real traffic traces [C2].***

Thesis 5.1. *I have worked out a bandwidth estimation algorithm, which can estimate the instantaneous available bandwidth in mobile networks with negligible load by modeling the queue dynamics. I have verified the operability of the proposed scheme using real traffic traces.*

Thesis 5.2. *I have given a general rule for the proper setting of the threshold parameter, which highly affects the results provided by the algorithm. I have shown that, while the number of busy periods decreases as the threshold increases, the estimation becomes more accurate until a breakpoint is reached, and the threshold must be adjusted to this value in order to achieve optimal operation.*

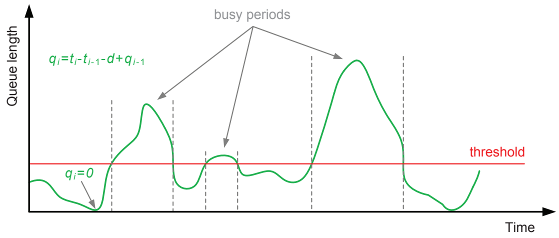
Thesis 5.3. *I have determined by measurements, conducted in a mobile network, that in the case of web traffic most busy periods (∼75%) are shorter than one second.*

Thesis 5.4. *I have verified that, in spite of the short busy periods, the proposed method is able to give accurate results.*

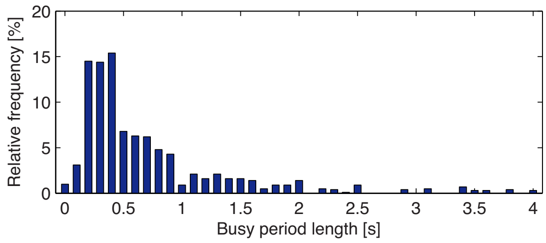
Bandwidth estimation algorithm

input : $trace, d, th, gap$
output: bw

```
m ← false; b ← false;
for i ← 1 to n − 1 do
    if t_{i+1} − t_i = d and m = false then
        q ← 0;
        m ← true;
    else if t_{i+1} − t_i > gap and m = true then
        m ← false;
        b ← false;
    else if m = true then
        q ← t_{i+1} − t_i − d + q;
        if q ≥ th and b = false then
            s ← t_i;
            b ← true;
        else if q < th and b = true then
            rates ← Add( amount of traffic in [s,t_i] / (t_i − s) );
            b ← false;
        end
    end
end
return bw ← Mean(rates);
```



(a) Queue dynamics modeling



(b) Busy period statistics

Figure 9. Operation of the bandwidth estimation algorithm

The core idea is that test packets are injected into the user-generated data traffic with a specified frequency, and by observing the arrival times of these packets at the mobile device, the dynamics of the bottleneck queue can be modeled (see Figure 9a). The algorithm takes different network conditions (e.g. jitter) into account and defines a positive threshold to determine those intervals when enqueued packets are serviced at the maximum rate. The pseudocode of the algorithm can be seen above where d, th, and gap denote the delay between the generation of test packets, the positive threshold used for busy period detection and the highest inter-arrival time can be accepted in the queue modeling phase, respectively. Furthermore, t_i is the arrival time of the i^{th} test packet captured at the mobile device and n gives the number of test packets in the traffic trace. The boolean variables m and b indicate if the queue modeling and busy period detection phases are active. Figure 9b shows that the vast majority of busy periods are shorter than one second in the case of web traffic, but long enough to estimate the current available bandwidth with high accuracy. The main advantage of the presented method is that, in contrast to many other solutions, *no knowledge is required about the network in advance*, as well as it exploits the user-generated network traffic *causing only a negligible additional network load*.

5 Application of the Results

The volume of Internet traffic has been growing exponentially each year thanks to the development of new technologies and the increasing user demands. The congestion control mechanism of TCP cannot keep up with today's continuously changing and heterogeneous network environments, resulting in poor performance in many cases. The digital fountain based data transfer paradigm presented in this dissertation is a promising approach, well-suited to a number of applications because of its beneficial properties. Since DFCP is insensitive to a wide range of packet loss and delay values observed in real networks, it is a good candidate for *wireless* and *lossy environments* where the packet loss ratio is non-negligible. The proposed scheme supports *multipath communication*, which makes it possible to achieve better network resiliency and load balancing. Moreover, the solution is closely aligned to the high utilization requirement of *data centers* and the concept of *all-optical networking* where only small buffers can be realized due to technological constraints.

References

[1] J. Postel, "Transmission Control Protocol", *RFC 793, IETF*, 1981.

[2] A. Afanasyev, N. Tilley, P. Reiher, L. Kleinrock, "Host-to-Host Congestion Control for TCP", *IEEE Communications Surveys and Tutorials*, vol. 12, no. 3, pp. 304–342, 2010.

[3] B. Raghavan, A. C. Snoeren, "Decongestion Control", *Proceedings of the 5th ACM Workshop on Hot Topics in Networks*, pp. 61–66, Irvine, CA, USA, 2006.

[4] D. Clark, S. Shenker, A. Falk, "GENI Research Plan (Version 4.5)", April 23, 2007.

[5] T. Bonald, M. Feuillet, A. Proutiere, "Is the 'Law of the Jungle' Sustainable for the Internet?", *Proceedings of the 28th IEEE Conference on Computer Communications*, pp. 28–36, Rio de Janeiro, Brazil, 2009.

[6] ns-2 Network Simulator, `http://www.isi.edu/nsnam/ns/`

[7] Network Simulation Cradle, `http://www.wand.net.nz/~stj2/nsc/`

[8] Emulab Network Emulation Testbed, `http://www.emulab.net/`

[9] S. Molnár, P. Megyesi, G. Szabó, "Multi-Functional Emulator for Traffic Analysis", *Proceedings of the 48th IEEE International Conference on Communications*, pp. 2397–2402, Budapest, Hungary, 2013.

[10] K. Pawlikowski, H.-D. Joshua Jeong, J.-S. Ruth Lee, "On Credibility of Simulation Studies of Telecommunication Networks", *IEEE Communications Magazine*, vol. 40, no. 1, pp. 132–139, 2002.

[11] M. Bateman, S. Bhatti, "TCP Testing: How Well Does ns2 Match Reality?", *Proceedings of the 24th IEEE International Conference on Advanced Information Networking and Applications*, pp. 276–284, Perth, Australia, 2010.

[12] S. Floyd, E. Kohler, "Tools for the Evaluation of Simulation and Testbed Scenarios", *Technical Report, IETF*, 2006.

[13] M. P. Fernandez, S. Wahle, T. Magedanz, "A New Approach to NGN Evaluation Integrating Simulation and Testbed Methodology", *Proceedings of the 11th International Conference on Networks*, pp. 22–27, Saint-Gilles, Réunion Island, France, 2012.

[14] D. X. Wei, P. Cao, S. H. Low, "Time for a TCP Benchmark Suite?", *Technical Report, California Institute of Technology*, pp. 1–6, 2005.

[15] D. J. C. MacKay, "Fountain Codes", *IEE Proceedings – Communications*, vol. 152, no. 6, pp. 1062–1068, 2005.

[16] A. Shokrollahi, "Raptor Codes", *IEEE Transactions on Information Theory*, vol. 52, no. 6, pp. 2551–2567, 2006.

[17] M. Shreedhar, G. Varghese, "Efficient Fair Queuing Using Deficit Round-Robin", *IEEE/ACM Transactions on Networking*, vol. 4, no. 3, pp. 375–385, 1996.

[18] A. Kortebi, L. Muscariello, S. Oueslati, J. Roberts, “On the Scalability of Fair Queuing”, *Proceedings of the 3rd ACM Workshop on Hot Topics in Networks*, pp. 1–6, San Diego, CA, USA, 2004.

[19] A. Shokrollahi, “LDPC Codes: An Introduction”, *Technical Report, Digital Fountain Inc.*, pp. 1–34, 2003.

[20] M. Luby, “LT Codes”, *Proceedings of the 43rd IEEE Symposium on Foundations of Computer Science*, pp. 271–280, Vancouver, BC, Canada, 2002.

[21] V. Jacobson, “Congestion Avoidance and Control”, *Proceedings of the 1988 ACM SIGCOMM Symposium*, pp. 314–329, Stanford, CA, USA, 1988.

[22] I. Rhee, L. Xu, “CUBIC: A New TCP-Friendly High-Speed TCP Variant”, *Proceedings of the 3rd International Workshop on Protocols for Fast Long-Distance Networks*, pp. 1–6, Lyon, France, 2005.

[23] L. Xu, K. Harfoush, I. Rhee, “Binary Increase Congestion Control (BIC) for Fast Long-Distance Networks”, *Proceedings of the 23rd IEEE International Conference on Computer Communications*, vol. 4, pp. 2514–2524, Hong Kong, China, 2004.

[24] T. V. Lakshman, U. Madhow, “The Performance of TCP/IP for Networks with High Bandwidth-Delay Products and Random Loss”, *IEEE/ACM Transactions on Networking*, vol. 5, no. 3, pp. 336–350, 1997.

[25] S. Floyd, T. Henderson, A. Gurtov, “The NewReno Modification to TCP’s Fast Recovery Algorithm”, *RFC 3782, IETF*, 2004.

[26] K. Fall, S. Floyd, “Simulation-based Comparisons of Tahoe, Reno, and SACK TCP”, *ACM SIGCOMM Computer Communication Review*, vol. 26, no. 3, pp. 5–21, 1996.

[27] G. Appenzeller, I. Keslassy, N. McKeown, “Sizing Router Buffers”, *ACM SIGCOMM Computer Communication Review*, vol. 34, no. 4, pp. 281–292, 2004.

[28] J. Gettys, K. Nichols, “Bufferbloat: Dark Buffers in the Internet”, *Communications of the ACM*, vol. 55, no. 1, pp. 57–65, 2012.

[29] H. Park, E. F. Burmeister, S. Bjorlin, J. E. Bowers, “40-Gb/s Optical Buffer Design and Simulation”, *Proceedings of the 4th International Conference on Numerical Simulation of Optoelectronic Devices*, pp. 19–20, Santa Barbara, CA, USA, 2004.

[30] Y. Huang, R. Guérin, “Does Over-Provisioning Become More or Less Efficient as Networks Grow Larger?”, *Proceedings of the 13th IEEE International Conference on Network Protocols*, pp. 225–235, Boston, MA, USA, 2005.

[31] D. Bansal, H. Balakrishnan, S. Floyd, S. Shenker, “Dynamic Behavior of Slowly-Responsive Congestion Control Algorithms”, *ACM SIGCOMM Computer Communication Review*, vol. 31, no. 4, pp. 263–274, 2001.

[32] Y. R. Yang, M. S. Kim, S. S. Lam, “Transient Behaviors of TCP-friendly Congestion Control Protocols”, *Proceedings of the 20th IEEE International Conference on Computer Communications*, vol. 3, pp. 1716–1725, Anchorage, AK, USA, 2001.

[33] S. Floyd, "Metrics for the Evaluation of Congestion Control Mechanisms", *RFC 5166, IETF*, 2008.

[34] S. Molnár, B. Sonkoly, T. A. Trinh, "A Comprehensive TCP Fairness Analysis in High Speed Networks", *Elsevier Computer Communications*, vol. 32, no. 13–14, pp. 1460–1484, 2009.

[35] C. D. Guerrero, M. A. Labrador, "On the Applicability of Available Bandwidth Estimation Techniques and Tools", *Elsevier Computer Communications*, vol. 33, no. 1, pp. 11–22, 2010.

[36] U. Hentschel, A. Schmidt, A. Polze, "Predictable Communication for Mobile Systems", *Proceedings of the 14th IEEE International Symposium on Object/Component/Service-Oriented Real-Time Distributed Computing*, pp. 24–28, Newport Beach, CA, USA, 2011.

Publications

Journal papers

[J1] S. Molnár, **Z. Móczár**, B. Sonkoly, "Living with Congestion: Digital Fountain based Communication Protocol", accepted to *Elsevier Computer Communications*, 2016.

[J2] **Z. Móczár**, S. Molnár, "Towards the Transport Protocols of Future Internet", *Infocommunications Journal*, vol. 6, no. 3, pp. 3–9, 2014.

[J3] G. Szűcs, **Z. Móczár**, "Iterative Text Clustering of Search Results", *International Journal of Computers*, vol. 7, no. 4, pp. 127–134, 2013.

Conference papers

[C1] **Z. Móczár**, S. Molnár, "On the Dynamic Behavior of Digital Fountain Based Communication", *Proceedings of the 58th IEEE Global Communications Conference, Exhibition and Industry Forum (GLOBECOM 2015)*, pp. 1–6, San Diego, CA, USA, 2015.

[C2] **Z. Móczár**, S. Molnár, "Bandwidth Estimation in Mobile Networks by Busy Period Detection", *Proceedings of the 25th IEEE Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC 2014)*, pp. 1354–1358, Washington D.C., USA, 2014.

[C3] **Z. Móczár**, S. Molnár, B. Sonkoly, "Multi-Platform Performance Evaluation of Digital Fountain Based Transport", *Proceedings of the Science and Information Conference 2014 (SAI 2014)*, pp. 690–697, London, UK, 2014.

[C4] S. Molnár, **Z. Móczár**, B. Sonkoly, "How to Transfer Flows Efficiently via the Internet?", *Proceedings of the 3rd IEEE International Conference on Computing, Networking and Communications (ICNC 2014)*, pp. 462–466, Honolulu, HI, USA, 2014.

[C5] S. Molnár, **Z. Móczár**, A. Temesváry, B. Sonkoly, Sz. Solymos, T. Csicsics, "Data Transfer Paradigms for Future Networks: Fountain Coding or Congestion Control?", *Proceedings of the 12th IFIP International Conference on Networking (Networking 2013)*, pp. 1–9, New York, NY, USA, 2013.

[C6] **Z. Móczár**, S. Molnár, "Comparative Traffic Analysis Study of Popular Applications", *Proceedings of the 17th EUNICE Workshop on Energy-Aware Communications (EUNICE 2011)*, pp. 124–133, Dresden, Germany, 2011.

[C7] S. Molnár, **Z. Móczár**, "Three-dimensional Characterization of Internet Flows", *Proceedings of the 46th IEEE International Conference on Communications (ICC 2011)*, pp. 1–6, Kyoto, Japan, 2011.

[C8] **Z. Móczár**, S. Molnár, “Characterization of BitTorrent Traffic in a Broadband Access Network”, *Proceedings of the 5th International Conference on Access Networks (AccessNets 2010)*, pp. 1–8, Budapest, Hungary, 2010.

Other scientific reports

[O1] **Z. Móczár**, “Modeling and Analysis of TCP Algorithms Based on Optimization Theory”, *Technical Report*, Budapest, Hungary, 2012.

[O2] S. Molnár, **Z. Móczár**, B. Sonkoly, Sz. Solymos, T. Csicsics, “Design and Performance Evaluation of the Digital Fountain based Communication Protocol”, *Technical Report*, Budapest, Hungary, 2012.