

DTN over Aerial Carriers

Ari Keränen
Helsinki University of Technology (TKK)
Dept. of Communications and Networking
Espoo, Finland
akeranen@comnet.tkk.fi

Jörg Ott
Helsinki University of Technology (TKK)
Dept. of Communications and Networking
Espoo, Finland
jo@comnet.tkk.fi

ABSTRACT

Delay-tolerant Networking has been discussed for mobile environments, both with predictable (e.g., between buses) and unpredictable opportunistic (e.g., between people) contact schedules. While some research projects have envisioned the larger scale use of DTNs to perform message forwarding based upon social interactions between humans, the geographic dimension of simulations has mostly been limited to rather local areas such as (parts of) cities or modestly sized areas covered by public transportation. In this paper, we explore how messages could be carried over a realistic large scale global network, between airports based upon scheduled flight connections. We investigate the interaction with different routing protocols, the impact of scheduling uncertainties, and the limiting factors by means of simulations and analysis. While we do not consider in depth the integration of aerial-carrier-based DTN message delivery into regional and local social networks, we provide some initial approximation on this subject and discuss some potential roles for message forwarding based upon mass transportation.

Categories and Subject Descriptors

C.2.2 [Computer Communication Networks]: Network Protocols

General Terms

Algorithms, Design, Performance

Keywords

Delay-tolerant Networking; DTN; Routing; Airport network

1. INTRODUCTION

Delay-tolerant networking can be used to enable communication in potentially sparse mobile ad-hoc networks formed between human users in a store-carry-and-forward manner. Various projects have explored exploiting these

Pocket-Switched Networks (PSN) [24] for message exchanges and content distribution, investigating a variety of different routing protocols, buffer management strategies, and application interactions.

For a meaningful evaluation, it is important that the mobility models used with analysis or simulations roughly reflect the real-world behavior of the expected user population. In support of this endeavor, on one hand, researchers have collected mobility and contact traces from diverse user groups in the real-world and analyzed their characteristics. On the other hand, sophisticated mobility models have been designed to synthetically generate motion patterns that increasingly match the characteristics of real world traces.

Most of these models focus on individual mobility while mobility following fixed schedules is mostly addressed in the context of space communications or sensor networks, with some exceptions discussed in section 2. Furthermore, even though some project presentations hint at long-distance communications via such DTNs [24], most investigations focus on rather local environments. It is surely sensible to exploit geographic proximity (and thus potentially shared contexts and interests) for interactions [12], searching [22, 11] and disseminating contents [16, 18], particularly since message delivery latencies may already become large in rather local settings. And, of course, long-distance communications could be carried out over infrastructure networks such as the Internet, eliminating the need for wide-area store-carry-and-forward networks.

Nevertheless, since the Internet is not yet ubiquitous [10] and people travel on scheduled transports (e.g., buses, trains, aircraft), it is worthwhile to investigate the impact of using large-scale scheduled transportation on the performance of mobile DTNs. In this paper, we take (a subset of) the global scheduled air traffic as an extreme example for high-latency long-distance scheduled transportation. The recent debate on the spreading of viruses and epidemics via international and intercontinental air traffic clearly suggests that airplanes (or airplane passengers) can make excellent vehicles to effectively spread messages and contents; but our considerations should be applicable to other transportation means as well.

Our contribution is threefold: we incorporate and analyze global air traffic scheduling data, present models for carrying messages over aerial carriers, and review the suitability and performance of DTN routing protocols as well as of the idea of aerial DTN at large. After a brief review of related work in section 2 with a focus on mobility models and traces, we present our integration of global air traffic data into our simulation environment in section 3. We introduce our sys-

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

CHANTS'09, September 25, 2009, Beijing, China.

Copyright 2009 ACM 978-1-60558-741-7/09/09 ...\$10.00.

tem model and simulation scenarios in section 4 where we also assess the relevance of different routing protocols and present a simple scheme specialized for scheduled transport. We present our findings in section 5 and discuss the role of large-scale scheduled transport in section 6 before concluding with a brief assessment of our findings and thoughts on future work in section 7.

2. RELATED WORK

Delay-tolerant Networking exploits *store-carry-and-forward* networking based upon messages (or *bundles*) of potentially arbitrary size (from a few bytes to gigabytes). Nodes may store messages for an extended period of time until a link to a (suitable) next hop becomes available, thus eliminating the need for an immediate end-to-end path. An available link is referred to as a *contact*, the link availability as *contact duration*, and the period between two contacts as *inter-contact time*. Contacts may be (virtually) permanent (e.g., in fixed networks), scheduled (i.e., known in advance as with scheduled buses), or opportunistic (e.g., for contacts between humans or animals).

A variety of DTN routing protocols [13, 14] have been developed (see, e.g., [32, 21] for overviews and [13, 23, 2] for a more formal treatment). Routing schemes differ with respect to the information they utilize to make routing decisions and whether they exchange routing information reactively (e.g., [19, 4, 1]), proactively (e.g., [7]), or simply make their decisions in isolation [28, 13]. Routing schemes can be further distinguished according to their *replication strategies*: single-copy schemes use just a single message copy whereas multi-copy protocols may create a limited [25] or unlimited [28, 19] number of copies. Finally, routing schemes may use erasure or network coding.

Messages are forwarded following the routing decisions and may be purged if their time-to-live (TTL) expires or due to a buffer overflow. Routing protocol performance may differ depending on the specific mobility scenario (where the number of nodes and the node density and contact frequency are usually quite important). For example, highly replicative routing (such as epidemic) easily floods and overwhelms the networks if the network is too well connected; protocols maintaining and exchanging per node [4] or link state [7] may not be viable for large-scale networks.

Routing protocols (and, ultimately, DTN applications) thus require evaluation in the context of their target mobile environments. To (better) understand the characteristics of human mobility in different contexts and the implications for communication opportunities between mobile users, numerous research projects have carried out collection of real-world traces for different user groups and settings. Such trace collections involve different sampling techniques—e.g., dedicated devices such as iMotes [5], Bluetooth monitoring software on mobile phones, or monitoring from WLAN access points—with different spatial and temporal resolution for determining, among other information, contact durations and inter-contact times. The settings range from regions of a city to students on a campus to conference participants to skaters in a city [26]. In addition to tracing individual users, some studies investigated unscheduled (taxis) and scheduled (buses) transportation means in cities to derive contact characteristics (e.g., [31]). The *Community Resource for Archiving Wireless Data At Dartmouth* [6] offers an archive from which many wireless traces of all kinds

are available. Our work complements previous activities by investigating a quite different scale, both in terms of mobile nodes and geographic coverage.

Such traces were also used to characterize contact durations and inter-contact times and to (in)validate assumptions about mobility models [5, 15], but they often represent fairly limited data sets. Therefore, synthetic mobility models capturing complex human behavior including scheduled long or short distance travel were developed. Mobility characteristics and mass motion of humans have been modeled in detail for city and traffic planning, using a modified simulator that was originally developed for epidemic studies [27]. In fact, routing of messages (particularly replication for the purpose of broadcasting) resembles the spreading of epidemics to some extent, [28] for which the impact of air traffic has been investigated in recent years (e.g., [3]).

Using aircraft (or other scheduled means for transportation) for carrying messages is somewhat similar to the ideas of *data mules* and *message ferries*, with the differences that, in contrast to data mules, our planes follow regular schedules and routing and, unlike message ferries, the motion/routing of the planes is not controlled to deliver messages.

Finally, RFC 1149 [29] defines how to carry IP packets over avian carriers, i.e., carrier pigeons, an approach basically also suitable for (if not demanding) DTNs—specifically as an implementation confirmed the expected high transmission latency and RTT.¹ However, these carrier pigeons so far do not follow complex schedules and the investigated deployments were fairly limited in scale.

3. MODELING AIR TRAFFIC

To experiment how well airplanes can be used for delivering messages between airports, we simulate global and local airline connections between the airports around the world. We run experiments using the ONE simulator [17] and observe how fast and reliably messages are transferred between the selected airports using scheduled flights between them.

For this purpose, we obtained the global flight schedules using the FlightStats Web Services API² for a total of 3879 airports worldwide shown in Figure 1 (every dot corresponds to a different airport). We retrieved all departure and arrival times for three consecutive days: Tue, 21 April (0:00 UTC) – Wed, 23 April 2009 (23:59 UTC). The dates were picked essentially randomly, but they were all taken from the middle of a week so that there would not be much difference in the schedules between different consecutive days.



Figure 1: Locations of the 3879 Airports

¹See <http://www.blog.linux.no/rfc1149/>

²<https://www.flightstats.com/developers/bin/view/Web+Services/>

Table 1: Number of entities used in simulations

Airports	Airplanes	Flights
100	6,598	55,460
300	12,968	128,150
700	17,329	184,490
3,879	21,358	248,469

During the three day period, 40 airports had no connection to any other airport, 70 had no arrivals, and 370 no departures. The distribution of airport degrees, i.e., concerning the number of peer airports to which direct connections exist as well as the number of flight connections, appears to be scale-free. When studying the global airport network in more depth, Guimerà et al. describe it as a small-world network in which the number of non-stop flight connections (node degree) and the number of shortest paths via a city (betweenness) follow scale-free distributions [9].³

For most of the simulations we used only the 100, 300, or 700 busiest airports (measured by the number of departures) and the first two days, simply to keep simulation state and time within reasonable bounds. When using 700 airports, this includes all airports with a node degree of 10 or more. From the flight schedule data, we selected the flights that depart from and arrive at one of the top airports and generated airport-airplane connectivity data: a plane is connected to the airport 1 hour before the departure time, it disconnects on departure, connects to the destination airport at arrival, and disconnects again 1 hour after the arrival time. During the connectivity period the airplane and airport can exchange messages. When an airplane is disconnected from the destination airport its message buffer is cleared so that airplanes carry messages only one hop between two airports. This also allowed us to “re-use” simulated airplanes so that we could simulate a large number of flights with a relatively small number of airplanes (see Table 1).

The connectivity data is fed into the ONE simulator as external connectivity events. The same external event interface was also used for flushing airplane message buffers before re-using them for different flights. The airports were placed in the simulation area according to the GPS coordinates of the FlightStats airport data, although the location data was used only for visualization and sanity checks since connections between nodes were created solely based on the flight schedule data. We did not track detailed flight routes, nor model in-flight communication between aircraft passing in nearby air corridors.

4. AERIAL MESSAGING SCENARIO

We are interested in the large-scale forwarding of messages carried by humans in their mobile devices when aboard airplanes. We assume a set of destination airports D reachable directly or indirectly from an origin A . Figure 2 shows a simple model for a single flight F from airport A to a next hop B from which flight connections head directly to a set of airports $C = \{C_1, C_2, \dots\}$. Some subset of airports U is not directly reachable from B so that further connections are required.

³Their investigation covered flight connections of 3883 cities for a period of one week in 2001, mapping combining several airports of a city into one element representing the city.

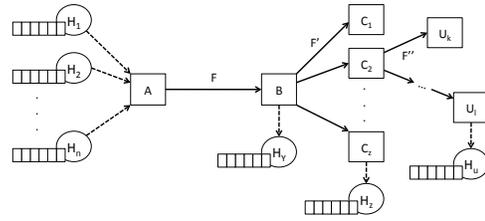


Figure 2: Passengers and messages on a flight

The flight F carries a set of passengers H , $n = |H|$. Every passenger $H_i \in H$ has a destination airport $D_i \in D$, chosen from D according to a distribution P_H , and carries a mobile device with m messages of mean size s . Each message has a destination represented by its airport D_j from which the message would be routed to its final destination local to the airport. The message destination airports are chosen based upon a distribution P_M . We assume that the messaging volume to a particular airport (=city) corresponds to the interest to people to travel to this airport and set $P_M = P_H$. For delivering a message to an airport, we assume that the passenger carrying the message has to have the airport chosen as her final destination.

When the passengers arrive at airport A and board the aircraft, they leave their mobile devices on. In an optimal case, each passenger has her electronic ticket or the boarding pass stored on the same mobile device so that the device actually knows the passenger’s routing and final destination. While the passengers are waiting, their mobile devices can thus exchange routing information and subsequently messages so that, ideally, every passenger P_i carries only messages for her final destination or for destinations “easily” reachable from there. If there are messages for destinations without passengers for the same destination (or more messages than capacity across all passengers heading there), we assume that other passengers with flight connections from the next hop B may carry these messages for one hop and then hand them over to more suitable passengers. This means that, for the purpose of replicating/moving messages between passengers and for determining the traffic volume, the indirectly reachable airports U_1, U_2, \dots can be subsumed by those C_1, C_2, \dots directly reachable from B .

Consider a single passenger H_i (out of n) carrying m messages. H_i heads for destination D_i with probability $P_H[X = D_i] = p(i)$. Each message M of H_i is addressed to a destination D_j with probability $P_M[X = D_j] = p(j)$ (recall that $P_M = P_H$). Thus, $m \times p(i)$ messages can be kept on the mobile device of the passenger, whereas $m \times (1 - p(i))$ messages need to be replicated to other passengers’ devices. Out of n passengers, $n \times p(i)$ will be heading for destination D_i , so that these passengers together need to transfer $N_i = n \times p(i) \times m \times (1 - p(i)) = n \times m \times (p(i) - p^2(i))$ messages to other nodes (and are able to receive as many messages).⁴ For all destinations, we thus obtain $N = n \times m \times \sum_{k=1}^{|D|} (p(i) - p^2(i))$.

For uniform distributions $P_M = P_H$, $N = n \times m \times (1 - \frac{1}{|D|})$

⁴We make a worst case estimation and assume that all messages not destined for the same airport as the device itself need to be transferred to other devices.

which quickly approaches the total number messages carried by all passengers as the number of destination airports grows. This also holds for a non-uniform distribution as shown in figure 3 as the diversity truly covers all the airports, but only 45% of the messages if only the top 100 airports are considered. Assuming 100 passengers on an airplane heading for a subset of the top 100 airports, each carrying 100 messages of 1 MB, requires them to exchange 4,500 messages with a total volume of 4.5 GB. Assuming a net data rate of 1 Mbit/s, 10 independent channels (or one channel at 10 Mbit/s) are required to complete the data exchange in one hour prior to departure (neglecting overhead and assuming perfect routing) which appears feasible. In the worst case, a single user’s device has to replace all 100 messages in its memory (thus sending and receiving 100 messages), requiring some three minutes activity at 10 Mbit/s, in addition to the coordination traffic to schedule the exchanges.

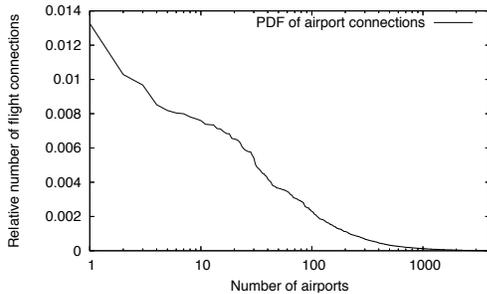


Figure 3: PDF of the number of airport connections

4.1 Simplified Model: Airports and Planes

As shown in the previous section, the total number of flights in our simulations is very large, and modeling individual passengers would—at this point—be beyond the capabilities of our simulation environment. Therefore, we make a simplifying assumption and do not consider individual passengers but rather model airplanes and airports only as depicted in figure 4.

The joint store-carry-and-forward capacity of all passengers aboard an aircraft is represented by a storage system (hard or solid state drive). An airport represents all passengers that board or leave any flight and thus the respective city as the message origin or destination. It also serves as a relay between connecting flights, with or without own storage capacity. Carriage of data aboard a plane is modeled by connecting the aircraft to the airport via a Gigabit Ethernet link (that may, e.g., come with the power supply for the aircraft). The airport has a hub-like structure and provides forwarding capacity representing passengers moving from one plane to another and (for practical purposes infinite) local storage capacity (S) representing passengers waiting for their connections. As noted above, for simulation purposes, an aircraft is assumed to be at an airport for one hour prior to departure and one hour after landing. This setup would allow an airplane to exchange some 400 GB of data at an airport, thus covering the capacity of a Boeing 747-400 or Airbus 340-600 with some 350 seats even if every passenger brought 1 GB of messaging data.⁵

⁵Note that, we do not consider memory capacity of modern

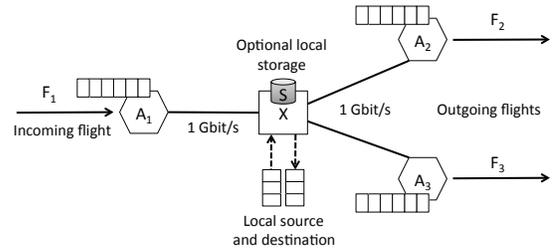


Figure 4: Simulation model: airports and planes

In our simulations, we focus on the scenarios in which the airport storage serves as a mediator for messages. Messages originate from and are destined to random airports (following a uniform distribution), using subsets of airports as indicated in section 3.

4.2 Routing Protocols

As outlined in section 2, numerous DTN routing protocols are available that could be leveraged for messaging over scheduled transportation means such as aircraft. The flight schedules are regular and known in advance, so that a routing system based upon a *Contacts Oracle* [13] could be created. Since the various airports are in regular interaction and schedules change only infrequently and predictively (e.g., once a year for some airlines such as Lufthansa), we can conceive a simple messaging protocol to distribute this airport link state information iteratively in advance and thus implement such an oracle in practice.

While DTLSR is deemed only suitable for small-scale deployments by its authors [7], the low frequency nature of updates to the schedule should make it possible to disseminate the complete information encapsulated in messages and then locally perform Dijkstra-based shortest path computations for the messages. For a global aerial network, we need to communicate state for about 250,000 links for three days. Each link can be represented by a (src,dst) pair, a (departure,arrival) time pair, a capacity metric, and a validity period (start, end). If we allow for 32 bits per value, this yields 28 bytes per link and thus we require some 16 MB data volume to describe all links for a week.⁶ This amount can be easily flooded several times per year to all airports, via aerial carriers or the traditional Internet without creating significant overhead. In our simulations, we assume the scheduling information to be distributed before the simulation starts. We refer to this algorithm as *Dijkstra*.

In contrast, history-based predictive algorithms such as PROPHET [19] or MaxProp [4] are unsuitable due to the state explosion in contact vectors (once nodes meet), the different timescales at which they meet, and the low contact

mobile devices to be an issue, with modern mobile phones featuring built-in memory of 8 or 16 GB and often allowing additional memory cards to be added—even if a significant fraction is used for map data, podcasts, music, photos, etc. Rather, it appears that the wireless link rate for forwarding data becomes the major constraint given the limited contact durations, so that too large storage capacity may not be fully usable as we also saw from the discussion of inter-passenger data exchange above.

⁶Our more fine-grained set of XML files retrieved from the web service was 281 MB in size for three days.

frequency. Also, the very nature of the system is that, while traveling, passengers are *not* likely to repeatedly encounter the same peers: a user heading out on a long-distance trip will change its context, unless traveling in a group.

Stateless algorithms such as *Epidemic* [28] and *Spray-and-Wait* [25] are more suitable. However, plain epidemic routing obviously faces the issue of an exploding number of messages. Since variants of epidemic routing may be quite useful for messaging in very sparse networks, we also include *native epidemic* routing into our comparison, but only for a limited number of airports and messages.

The major feature of epidemic routing is that it would automatically find the shortest and fastest path if only a single message would be in the system. This also would apply to our Dijkstra approach, but *only* if the presumed schedule is truly accurate and no flights are delayed or canceled. To determine the achievable performance, we use the notion of a *SuperEpidemic* routing system setup with three key features: 1) It uses infinite message buffers to avoid dropped messages and 2) assumes that messages are transferred instantly with no delay to eliminate queuing delays. 3) Once a message reaches a destination, an instant notification is generated and all message copies queued anywhere in the system are immediately discarded. While all three assumptions are unrealistic, they yield an optimized routing protocol able to find the shortest path that can serve as a reference.

The original Spray-and-wait has often shown good performance in various local environments. However, when limiting the number of messages it also limits forwarding (replication) as soon as only a single copy is left in a node—and thus may face the issue that this node may easily end up in the wrong city without further forwarding. Extensions guiding the direction of replication (e.g., based upon passenger routing information) would be preferable to random replication. Similarly, other routing protocols (such as the intention-based routing done by RAPID [1]) might be adapted to limit the amount of state information per peer and parameterized to consider flight schedule information in a suitable way, e.g., to minimize delivery latency or perform load balancing. We leave these ideas for further study and concentrate on the basic properties of aerial DTN routing in this paper.

4.3 Considering City-local Messaging

In addition to the aerial messaging between airports, we consider a single case for forwarding messages from Helsinki Vantaa airport to/from the city center—and, for simplicity, assume symmetry of delays and delivery ratios for other cities. Since Helsinki-Vantaa airport is 22 km away from the city center, we use the airport buses commuting between the airport and the railway station as means for message transportation rather than extending the city map. We assume a 30 min one-way delay and an interval of 20 min for the buses, thus adding a mean additional latency of 40 min.

Inside the city, we run the *Working Day Movement (WDM)* model [8] with 500 mobile nodes (pedestrians, cars, buses) plus one node representing the railway station bus stop. All nodes use Bluetooth at 2.1 Mbit/s net data rate and 10 m communication ranges. Each mobile node generates a message destined for the railway station node between 6:00am and 10:00pm on the first day. Similarly the railway station node generates a message for every mobile node during the first day. The message sizes are evenly distributed between

Table 2: Average performance metrics of simulations with 100 random messages

Airports	Delivery prob.	Delay	Hop count
300	99.2%	1,214 min	7.06
700	97.8%	1,347 min	8.17
3879	61.0%	1,900 min	11.33

500 KB and 1 MB. We run the simulations for a 48 hour period, after 24 hour warmup period for the movement model.

We are well aware that this simplified model does not adequately capture the true complexity, but it should provide us with a basic understanding of the order of magnitude of message delivery times. Further exploration remains subject to future work.

5. SIMULATION RESULTS

We performed simulation runs with the top 300, 700, and all 3,879 airports using both SuperEpidemic and Dijkstra. In the absence of any deviations from the schedules, they perform identically so that there is no need to distinguish between them further. We generated 100 messages sent randomly between airports at the beginning of the simulation and run each simulation for 48 hours simulation time. We run five iterations with different random seeds and calculate the mean for three performance metrics: delivery probability, delivery latency, and hopcount. Note that the hop count represents DTN hops which is twice the number of airline connections: one hop for the message transfer from the airport to the aircraft, another one from the aircraft to the airport. Thus, the mean number of flight connections a message takes is half the indicated hop count mean.

Table 2 shows the results for on-time flights. As we can see, the metrics are quite similar for the top 300 and 700 airports, most importantly, the message delivery ratio is—expectedly—close to 100% and non delivery is only due to simulation time ending before message is delivered (which can also be interpreted as message TTL expiry at the end of the simulation time). When considering all airports, the observed hopcount grows and messages are less likely to be delivered in time, because the less well connected airports may not have flights arriving in time for their messages to be delivered. Figure 5 shows delay distribution of one of the simulation runs with 300, 700, and 3879 airports. The chosen simulation run is the one with the median delivery probability, but other runs yielded similar results.

Figure 6 clearly shows that Dijkstra (=SuperEpidemic) performs much better than epidemic routing even in a case with very little load (a total of 100 messages between the top 700 airports). Epidemic creates so many message copies that not all transfers can complete between two planes F_1 and F_2 (via the airport) and a plane due to queuing delay if a F_1 comes in briefly before F_2 departs. In this scenario, epidemic routing created about 10,000 times more messages than Dijkstra and the latter has less than 10 message copies carried on the same flight.

If we consider that flights may deviate from their schedule due to delays (we do not care about early arrivals), the Dijkstra algorithm will use incorrect assumptions that should result in suboptimal routing. For our simulations, we as-

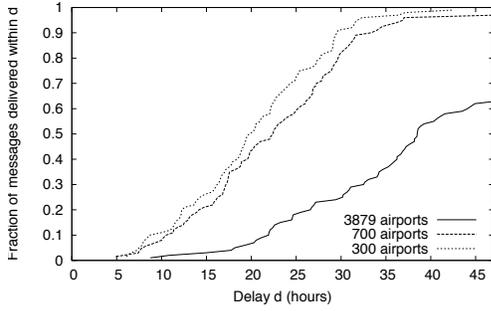


Figure 5: Delivery delay for 100 random messages

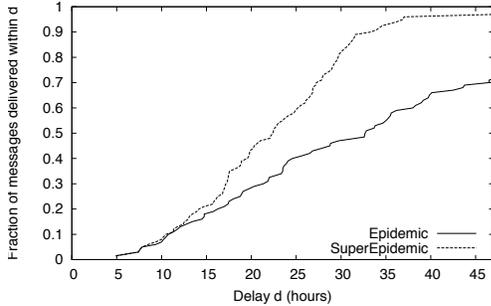


Figure 6: Epidemic vs. SuperEpidemic

sume that 22% of the flights are delayed⁷ and that 17% of all flights have low delays evenly distributed in 10–90 min, whereas 5% have longer extra delays, evenly distributed in 90–150 min. We compare the resulting CDF for message delays with schedule deviations against those without and plot the results in figure 7.

The resulting increase in delivery latency can be attributed to the delay itself and, to a lesser extent, to the missed flight connections. The overall impact is not so large as the message routing would need to be impacted and their connection be so tight that it would be missed to begin with. A closer investigation reveals that only 21% of the messages are actually impacted by the delayed flight schedules. And even if a message misses a connecting flight due to an incoming one being delayed, the routing system finds almost equally good alternative paths so that the message reaches the destination with only a small delay. Table 3 shows the delay impact for 1,000 messages sent: only 15.5% of the messages experience an additional delivery delay of more than one hour, just a single message with the worst impact almost 21 hours. Overall, the airport network seems to have sufficient redundant paths to cope with schedule inaccuracies, at least to a limited extent. More systematic exploration of delay as well as considering re-routing and airport “failures” (e.g., temporary shutdown due to weather conditions) is needed to characterize the robustness of an aerial DTN network.

As shown in table 2 and in more detail in figure 8 (recall that message hops = $2 \times$ airline hops), messages do not

⁷Based upon the domestic US figures for March 2009 as available from http://www.transtats.bts.gov/OT_Delay/OT_DelayCause1.asp

Table 3: Impact of schedule deviations (delays) on individual messages

Del. Delay	>1h	>2h	>3h	>4h	>5h	>20h
# Messages	155	91	58	34	29	1

take so many hops in most cases and, for well connected airports (which are preferred as further investigation revealed), alternative flights may be available in many cases.

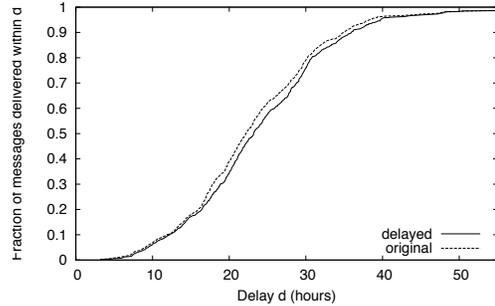


Figure 7: Impact of schedule deviations (delays) on message delivery latency

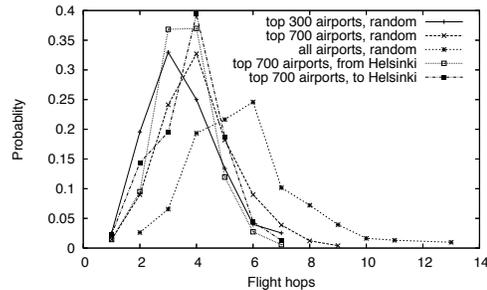


Figure 8: Distribution of airline hops

In figure 9, we show the distribution of airline hops a message takes and the mean delivery latency as a function of the number of hops for messaging between the top 300, top 700, and all airports as well as for the top 700 airports and Helsinki. From the delay for single hop trips (specifically for Helsinki from where not so many long-haul flights depart), we can conclude that a few hours of waiting at the airport for an appropriate connection contribute to the overall delay. Overall, with increasing number of hops, the mean per-hop delay appears to decrease (except for flights to and from Helsinki) since the complete view of the network allows choosing from many alternative paths and reduces the impact of waiting. More encompassing investigations with all airports and more traffic are needed to characterize this dependency further, also considering the load multiple messages sharing a flight connection cause.

The detailed message delivery statistics for on-time departures and arrivals between Helsinki and the other 699 airports of the top 700 are shown in figure 10. Over 90% of the

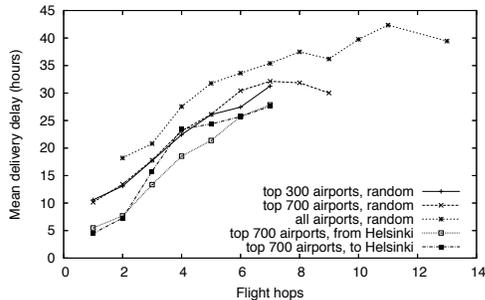


Figure 9: Mean delivery delay per airline hops

Table 4: Performance metrics of WDM scenario

Scenario	Delivery prob.	Average delay
to airport	99.2%	765 min
from airport	66.6%	980 min

world’s top 700 airport are reachable in a bit more than 24 hours. The asymmetry in delivery times stems from the fact that only few flights arrive in Helsinki during the night—which gives a clear limitation in messaging over scheduled means for transportation.

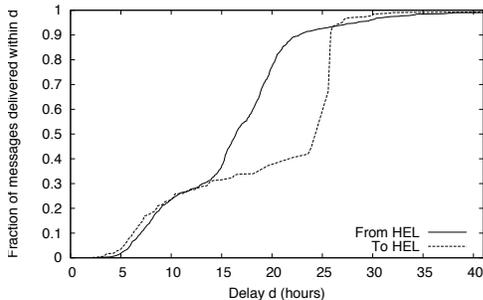


Figure 10: Delivery delay distribution for messages sent between Helsinki and the top 700 airports

Table 4 shows the results of WDM scenario for local transportation in downtown Helsinki, between individual users and the train station, the latter of which represents another scheduled connection to the airport. Adding the mean commuting time to/from the airport, we arrive at a mean delivery delay of 13 hours towards the airport and 17 hours towards an individual user. This accounts for 50–70% of the time it takes to reach any of the top 700 airports in the world. Assuming symmetric delivery performance, the local distribution will account for the same amount of time (if not more) required to carry the message around the world.

While most messages make it to the airport—simply because the train station is a fixed and central location—and also the delivery rate between airports approaches 100% after 40 hours, only two thirds of the messages are delivered in the opposite direction. Thus, we have—in the non-congested case—a relatively reliable and “fast” aerial backbone but a lossy and comparably slow access network.

6. DISCUSSION

Overall, our intention is to gain some initial understanding of large-scale DTN usage with predictable contact schedules for long-distance message carriage. Nevertheless, the above discussion begs the question how long distance aircraft-based DTN message delivery compares to the use of wireline infrastructure networks such as the Internet. Since loads of DVDs, CDs, and other mass storage media are shipped every year, a *parcel service* for (large) messages may turn out to be a suitable complement. For example, various on-line backup service providers offer receiving disk drives or DVDs to create the initial backup image based upon which subsequently incremental backups are carried out. While this happens primarily to circumvent local—relatively low speed or loaded—access links, (fiber-based) access networks offering data rates of 100 Mbit/s or more may not help since TCP’s steady state performance is limited as a function of the round-trip time (and packet loss)[20].

In this section, we want to perform a simple *gedankenexperiment*. For this purpose, we create an idealized scenario in which we only consider direct flights between different airports and assume aircraft-based (rather than passenger-based) data carriage using the parameters above. Such services might (in addition to regular airlines) be offered by today’s parcel delivery services. We also make the simplifying assumption that the data to be sent or received is already available inside a data center at the airport. We then compare data traffic using a single TCP connection between a sender and a receiver to the hypothetical message parcel service, assuming a packet loss rate of 0.1% for the wide area Internet and otherwise no path capacity constraints.

In table 5, we show the results for data transmission from Helsinki to five cities in different distances. We determine the message transmission delay conservatively by assuming a full one hour for uploading and downloading messages at the airport (even though FIFO could be applied to reduce the latency) and adding these two hours to the flight time. The indicated RTT is a rounded mean from a daytime series of ICMP PING measurements. The TCP throughput is calculated based upon [20] for a single TCP connection, the DTN throughput is determined by the indicated delay required to move 450 GB of data. The *break even* indicates at which data volume, the transfer using aerial DTN is faster than the corresponding TCP-based transmission. Under these circumstances, it is interesting to note that flying a data volume equivalent of a mini Blu-ray disc (15.6 GB dual layer) or three DVDs to four out of five destinations will be faster than transmitting the data over the network. Carrying the contents as messages bears the advantage over carrying physical disks [30] that the data does not require manual handling after arrival.

This idealistic example is surely flawed in many respects as it only holds for the atomic transmission of large data units over a single TCP connection (even though static bulk data could be striped across multiple), does not consider multi-path capable transport protocols or different congestion control regimes which could yield better results, does not take into account the limitations of physical devices (such as disk/flash memory read/write speed) and local airport network architectures to sustain this capacity, among others. Yet, the experiment indicates that routed store-carry-and-forward delivery may be a suitable complement even for well-connected environments under certain circum-

Table 5: Comparing traditional terrestrial and aerial DTN communication performance

Destination	Flight time	Delay	RTT	TCP throughput	DTN throughput	Break even
Stockholm	1 h	10800 s	8 ms	40.0 Mbit/s	333 Mbit/s	54 GB
Frankfurt	2 h	14400 s	40 ms	8.0 Mbit/s	250 Mbit/s	14 GB
Madrid	4 h	21600 s	60 ms	5.3 Mbit/s	167 Mbit/s	14 GB
New York	8 h	36000 s	120 ms	2.7 Mbit/s	100 Mbit/s	12 GB
Los Angeles	12 h	50400 s	190 ms	1.7 Mbit/s	71 Mbit/s	11 GB

stances. For example, beyond point-to-point communications, well-connected airports might also be used for content replication and distribution, e.g., for aerial DTN multicasting. And with longer stops or higher access rates, the feasible data volumes could be scaled up further.

7. CONCLUSION

In this paper, we looked at DTN-based communication using long-distance large-scale scheduled transportation means to carry messages (along with passengers). As expected, a simple Dijkstra-based routing protocol can be used to exploit the known schedules to deliver messages predictably and, since we employ single-copy message routing in this case, quite reliably to the destination—but, for scaling reasons, we only explored limited offered loads so far, thus avoiding congestion losses and aborted transfers.

Message delivery latencies become naturally significant for long-haul flights, but we also find particularly long delays for multi-hop messaging. In some cases, message paths get very long in terms of hop-counts (and the routing becomes quite surprising at a first glance). Inter-connecting co-located airports (e.g., in the same city) should help to some extent as this increases the number of available next hops; but this requires infrastructure and thus has only limited applicability to PSNs. Allowing for more delay and optimizing the Dijkstra calculation for the minimum hopcount could reduce the risk of missed connections. This becomes more relevant as soon as we introduce deviations from the schedules due to late departures and arrivals. Overall, however, we find that the airport network is fairly robust with respect to inaccurate scheduling information due to delays: the messages delivery rate does not change significantly and, while the delivery latency of some 20% of the messages increases, the increase is mostly moderate.

Combining individual messaging from a user in one city via flight connections to another one is doable, but comes at significant delay and message loss risk, as is often the case with opportunistic networking. For usually rather small point-to-point messages, this does not appear to be a viable alternative to utilizing Internet connectivity. For distributing and sharing content among (groups of) users, this opportunistic flight connectivity may be useful add-on. Especially when the intention is to exchange large data volumes, aerial DTN messaging might have potential to become a complement to Internet-based communications.

So far, we have only scratched the surface of DTN over scheduled carriers. Our above observations suggest further studies of the robustness properties and the impact of different delays and temporary airport failures. One route to enhance our Dijkstra algorithm to increase robustness may be to take into account a confidence in or an expected variation of the schedule to optimize the aerial routing. We will

also consider further routing protocols, such as a modified Spray-and-Wait and content dissemination protocols.

In addition, the scale-free properties of the airport network may allow applying theoretical results on complex networks to characterize the properties of the airport network and help with the design of suitable routing protocols. However, it needs to be considered that the global airport network was found to feature characteristics quite distinct from other complex networks such as the Internet, namely exhibiting less correlation of node degrees and betweenness than other (and random) complex networks [9] and that additional parameters (e.g., flight times, pauses during the night) may need to be taken into account.

For messaging between mobile users (who are possibly in different cities), numerous more fundamental issues need to be explored. In our artificial simulation scenario, we have kept local routing and aerial routing separate. While such an approach may be applicable when messages are forwarded using aircraft infrastructure, this gets more complicated as soon as we move back to passengers as message carriers. They may need to distinguish between their “modes of operation” to apply the appropriate routing scheme. This may still be doable if we assume valid tickets or boarding passes stored on the mobile device as discussed above. However, using other means of transportation may not be so easy to identify, e.g., if train tickets are bought immediately before or after boarding, so that the routing is not known ahead of time.⁸ Hence, a smoother integration of long-distance and local routing protocols deserves further investigation; such an approach might exploit partial knowledge of travel schedules for some users. Finally, since using PSNs for long-distance messaging still appears somewhat far-fetched at this point, understanding how *known* scheduled transportation can impact local opportunistic communication is an interesting direction for the near term.

8. REFERENCES

- [1] Aruna Balasubramanian, Brian Levine, and Arun Venkataramani. DTN Routing as a Resource Allocation Problem. In *Proceedings of ACM SIGCOMM*, August 2007.
- [2] Vincent Borrel, Mostafa Ammar, and Ellen Zegura. Understanding the Wireless and Mobile Network Space: A Routing-centered Classification. In *Proc. of the ACM MobiCom CHANTS workshop*, 2007.
- [3] Dirk Brockmann, Lars Hufnagel, and Theo Geisel. Dynamics of Modern Epidemics. In *SARS: A Case Study in Emerging Infections*, edited by A. McLean

⁸We also assume that the target city of the message destination is known, which may no longer hold with mobile users (particularly considering the message delivery latencies).

- and R. May and J. Pattison and R. Weiss. Oxford University Press, 2005.
- [4] J. Burgess, B. Gallagher, D. Jensen, and B. N. Levine. MaxProp: Routing for Vehicle-Based Disruption-Tolerant Networks. In *Proceedings of IEEE Infocom*, April 2006.
 - [5] Augustin Chaintreau, Pan Hui, Jon Crowcroft, Christophe Diot, Richard Gass, and James Scott. Impact of Human Mobility on the Design of Opportunistic Forwarding Algorithms. In *Proc. IEEE INFOCOM*, April 2006.
 - [6] Homepage of the Community Resource for Archiving Wireless Data At Dartmouth (CRAWDAD). <http://crawdad.cs.dartmouth.edu/>, 2007.
 - [7] Michael Demmer and Kevin Fall. DTLSR: Delay Tolerant Routing for Developing Regions. In *ACM SIGCOMM Workshop on Networked Systems in Developing Regions (NSDR)*, August 2007.
 - [8] Frans Ekman, Ari Keränen, Jouni Karvo, and Jörg Ott. Working day movement model. In *MobilityModels '08: Proceeding of the 1st ACM SIGMOBILE Workshop on Mobility models*, pages 33–40, 2008.
 - [9] R. Guimerà, S. Mossa, A. Turtschi, and L.A.N. Amaral. The world-wide air transportation network: Anomalous centrality, community structure, and cities' global roles. *Proc. National Academy of Sciences, USA*, 102(22):7794–9, May 2005.
 - [10] S. Guo, M.H. Falaki, E.A. Oliver, S. Ur Rahman, A. Seth, M.A. Zaharia, U. Ismail, and S. Keshav. Design and Implementation of the KioskNet System. In *International Conference on Information Technologies and Development*, 2007.
 - [11] Pan Hui, Jérémie Leguay, Jon Crowcroft, James Scott, Timur Friedman, and Vania Conan. Osmosis in Pocket Switched Networks. In *ChinaCom*, 2006.
 - [12] Md. Tarikul Islam, Anssi Turkulainen, Teemu Kärkkäinen, Mikko Pitkänen, and Jörg Ott. Practical Voice Communications in Challenged Networks. In *Proceedings of the ExtremeCom workshop*, 2009.
 - [13] Sushant Jain, Kevin Fall, and Rabin Patra. Routing in a Delay Tolerant Network. In *Proceedings of the ACM SIGCOMM*, 2004.
 - [14] Evan P. C. Jones, Lily Li, and Paul A. S. Ward. Practical routing in delay-tolerant networks. In *WDTN'05: Proceeding of the 2005 ACM SIGCOMM Workshop on Delay-tolerant Networking*, pages 237–243, New York, NY, USA, 2005. ACM Press.
 - [15] Thomas Karagiannis, Jean-Yves Le Boudec, and Milan Vojnovic. Power law and exponential decay of inter contact times between mobile devices. In Evangelos Kranakis, Jennifer C. Hou, and Ram Ramanathan, editors, *Proc. ACM MobiCom*, pages 183–194, September 2007.
 - [16] Gunnar Karlsson, Vincent Lenders, and Martin May. Delay-tolerant Broadcasting. In *ACM SIGCOMM Workshop on Challenged Networks (CHANTS)*, 2006.
 - [17] Ari Keränen, Jörg Ott, and Teemu Kärkkäinen. The ONE Simulator for DTN Protocol Evaluation. In *SIMUTools '09: Proceedings of the 2nd International Conference on Simulation Tools and Techniques*, New York, NY, USA, 2009. ICST.
 - [18] Jérémie Leguay, Anders Lindgren, James Scott, Timur Friedman, and Jon Crowcroft. Opportunistic content distribution in an urban setting. In *CHANTS '06: Proceedings of the 2006 SIGCOMM workshop on Challenged networks*, pages 205–212, New York, NY, USA, 2006. ACM Press.
 - [19] Anders Lindgren, Avri Doria, and Olov Schelen. Probabilistic routing in intermittently connected networks. In *The First International Workshop on Service Assurance with Partial and Intermittent Resources (SAPIR)*, 2004.
 - [20] Jitendra Padhye, Victor Firoiu, Don Towsley, and Jim Kurose. Modeling TCP Throughput: A Simple Model and its Empirical Validation. In *Proceedings of ACM SIGCOMM*, 1999.
 - [21] L. Pelusi, Andrea Passarella, and Marco Conti. Opportunistic networking: Data forwarding in disconnected mobile ad hoc networks. *IEEE Comm. Magazine*, November 2006.
 - [22] Mikko Pitkänen, Teemu Kärkkäinen, Janico Greifenberg, and Jörg Ott. Searching for Content in Mobile DTNs. In *Proceedings of the 7th Annual IEEE International Conference on Pervasive Computing and Communications (PerCom)*, 3 2009.
 - [23] Ram Ramanathan, Prithwish Basu, and Rajesh Krishnan. Towards a Formalism for Routing in Challenged Networks. In *Proc. of the ACM MobiCom CHANTS workshop*, 2007.
 - [24] James Scott, Pan Hui, Jon Crowcroft, and Christophe Diot. Haggle: A Networking Architecture Designed Around Mobile Users. In *Proceedings of IFIP WONS*, Les Ménuires, France, January 2006.
 - [25] Thrasyvoulos Spyropoulos, Konstantinos Psounis, and Cauligi S. Raghavendra. Spray and Wait: An Efficient Routing Scheme for Intermittently Connected Mobile Networks. In *Proc. ACM SIGCOMM Workshop on Delay-Tolerant Networking (WDTN)*, 2005.
 - [26] Pierre-Ugo Tournoux, Jérémie Leguay, Farid Benbadis, Vania Conan, Marcelo Dias de Amorim, and John Whitbeck. The Accordion Phenomenon: Analysis, Characterization, and Impact on DTN Routing. In *Proceedings of IEEE INFOCOM*, 2009.
 - [27] Homepage of the TranSims Transportation Analysis Simulation System. <http://www.ccs.lanl.gov/transims/>, 2007.
 - [28] A. Vahdat and D. Becker. Epidemic routing for partially connected ad hoc networks. Technical Report CS-200006, Duke University, April 2000.
 - [29] D. Waitzman. Standard for the transmission of IP datagrams on avian carriers. RFC 1149, 1 April 1990.
 - [30] Randolph Y. Wang, Sumeet Sobti, Nitin Garg, Elisha Ziskind, Junwen Lai, and Arvind Krishnamurthy. Turning the postal system into a generic digital communication mechanism. In *Proceedings of ACM SIGCOMM*, 2004.
 - [31] Xiaolan Zhang, Jim Kurose, Brian Neil Levine, Don Towsley, and Honggang Zhang. Study of a Bus-based Disruption-Tolerant Network: Mobility Modeling and Impact on Routing. In *Proc. of ACM MobiCom*, 2007.
 - [32] Z. Zhang. Routing in intermittently connected mobile ad hoc networks and delay tolerant networks: Overview and challenges. *IEEE Communications Surveys and Tutorials*, 8(4):24–37, January 2006.