Comparison of Incentive-based Cooperation Strategies for Hybrid Networks

Attila Weyland, Thomas Staub, and Torsten Braun

University of Bern
Institute of Computer Science and Applied Mathematics
Neubrückstrasse 10, 3012 Bern, Switzerland
`{weyland|staub|braun}@iam.unibe.ch`

Abstract. Today's public Wireless LANs are restricted to hotspots. With the current technology, providers can only target a small audience and in turn charge high prices for their service to generate revenue. Also, providers can not react appropriate to dynamic changes of the demand. With multi-hop cellular networks the coverage area can be increased and the installation costs and investment risks for the provider can be reduced. However, the individual customers play an important role in such networks and their participation must be encouraged. Therefore, we propose a cooperation and accounting scheme which introduces monetary rewards. We compare our scheme called CASHnet with the Nuglet scheme using simulations under the criteria of network liveliness as well as goodput, overhead and packet error rate.

1 Introduction

The current wireless network installations consists of a number of access points deployed in selected areas, where they are expected to serve a minimum amount of customers to bring revenue to the provider, e.g. at airports or railway stations. Potential customers outside the area covered by the access point can not be served. Besides the financial risk limiting the deployment of access points, location properties can also be restricting factors.

With multi-hop cellular networks, also called hybrid networks, the single-hop limit does not exist any more. Customers act as packet forwarders (like in mobile ad hoc networks) and a gateway offers the connection to the the Internet. This gives the provider a greater coverage area with more customers and reduces the network installation costs. Customers get connectivity outside hotspot areas and can reduce their energy consumption due to shorter next-hop distances. The advantages of mobile ad hoc networks come together with the disadvantages such as maintaining accurate route information, protecting customers from attacks as well as the need to encourage cooperation among customers for keeping the network alive.

Although individual customers may have a common interest in obtaining connectivity, customers tend to prioritize their self-generated packets over packets to be forwarded from other customers when energy is regarded as precious, limited good. Thus cooperation among selfish individuals (customers) can not be taken for granted, but can either be enforced or made attractive. We believe that in civilian applications without a single authority, enforcement is not attractive to individual customers.

Early work enforced cooperation by not allowing any non-cooperative participants [1] or by threat of punishment in case of non-cooperative behavior [2]. In [3] rewards have been introduced as incentive for cooperation in mobile ad hoc networks. The authors of [4] and [5] extended this notion to the multi-hop cellular network environment. They both heavily rely on centralized accounting and security mechanisms.

In a previous publication [6], we proposed a scheme called CASHnet (Cooperation and Accounting Strategy in Hybrid Networks), which allows selfishness, but at the same time makes cooperation a rewarding alternative. We took a highly decentralized approach for the accounting as well as for the security architecture. Accounting is done on the device and authentication is based on public key cryptography. We also allow cost sharing between sender and receiver located in different subnetworks. Each of them pays an amount related to their respective distance to the gateway. Distance related charges generate revenue at the location in the network where the expenses occur. Because all intermediate customers only participate if they get rewarded, longer distances with more intermediate customers on the path to the gateway imply more rewards and thus raise the cost for obtaining the service at a distant location. The alternative would be for the provider to install a new hotspot at the distant location, with all the financial risks implied or for the customer to have no service at all.

In this paper we compare our proposal with the Nuglet [1] scheme in terms of liveliness of the network and overall performance. We find that even with a single, centered Service Station in the network, CASHnet performs better than Nuglet. With an increasing number of Service Stations, the goodput in CASHnet is much higher compared to Nuglet. We also analyze the generated overhead, the packet drop reasons and discuss further improvements to CASHnet.

The rest of the paper is structured as follows. In section 2 we present and compare the Nuglet and the CASHnet schemes. Section 3 describes our evaluation process. In Section 4 we discuss the results we obtained. We conclude our paper and give an outlook in Section 5.

2 Cooperation Schemes

In the introduction we presented some of the available cooperation schemes in the literature and we motivated our approach. The following two sections describe the CASHnet and the Nuglet scheme, which will be compared through simulations later on. From the available cooperation schemes we chose the Nuglet approach because it is - like CASHnet - a decentralized approach with similar requirements and therefore easier to compare. In the description of the schemes we focus on the aspects important for this comparison. More detailed explanations can be found in the given references.

2.1 CASHnet

The CASHnet charging and rewarding mechanism works as follows: Every time a node (customer) wants to transmit a self-generated packet, it has to pay with *Traffic Credits*. The amount is related to the distance in hop counts to the gateway. Every time a node forwards a packet, it gets *Helper Credits*. Traffic Credits can be bought for real money or traded for Helper Credits at Service Stations. A Service Station is similar to a low-cost terminal for loading prepaid cards and has a secure, low-bandwidth connection to the provider, which is used for authentication and payment operations.

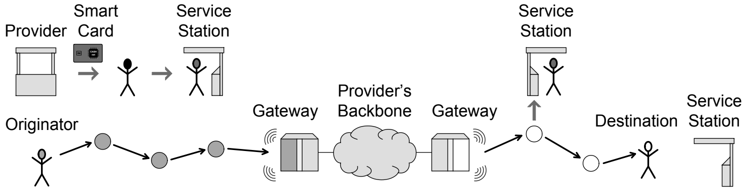


Fig. 1. CASHnet example scenario

Figure 1 displays an example scenario of the CASHnet scheme in operation. After preparation a node is ready to execute the different operation phases illustrated as flow graphs in Figure 2. Suppose a node called *Originator* wants to communicate with another node called *Destination* located in a different subnetwork. First, the *Originator* obtains a smart card from the provider (Preparation). Then it authenticates to all nodes along the path to the destination (Authentication Message Generation, Reception & Forwarding Phases). Now it can start to transmit self-generated packets (Packet Generation Phase). The *Originator* will only pay for the distance to the gateway (in hop counts) of its subnetwork and the *Destination* will pay for the distance to its corresponding gateway. An intermediate node gets rewarded (Rewarding Phase), after the forwarded packet reaches the next hop along the path toward the destination (Packet Reception Phase, Packet Forwarding Phase). All packets are digitally signed and verified upon reception to ensure non-repudiation, i.e. data integrity and data origin authentication. For a more detailed description of our scheme we refer to [6].

Preparation: Node N obtains a personal smart card from the provider X with an unique identifier ID_N, a public/private key pair K_N/KP_N, a certificate $Cert_X(ID_N, K_N)$ issued by provider X for N and the provider's public key K_X. It then performs the following steps:

- load Traffic Credits account TCA at provider's Service Station by paying with real money and/or by transferring from Helper Credits account HCA (as necessary) and
- update certificate $Cert_X(ID_N, K_N)$ (as necessary)

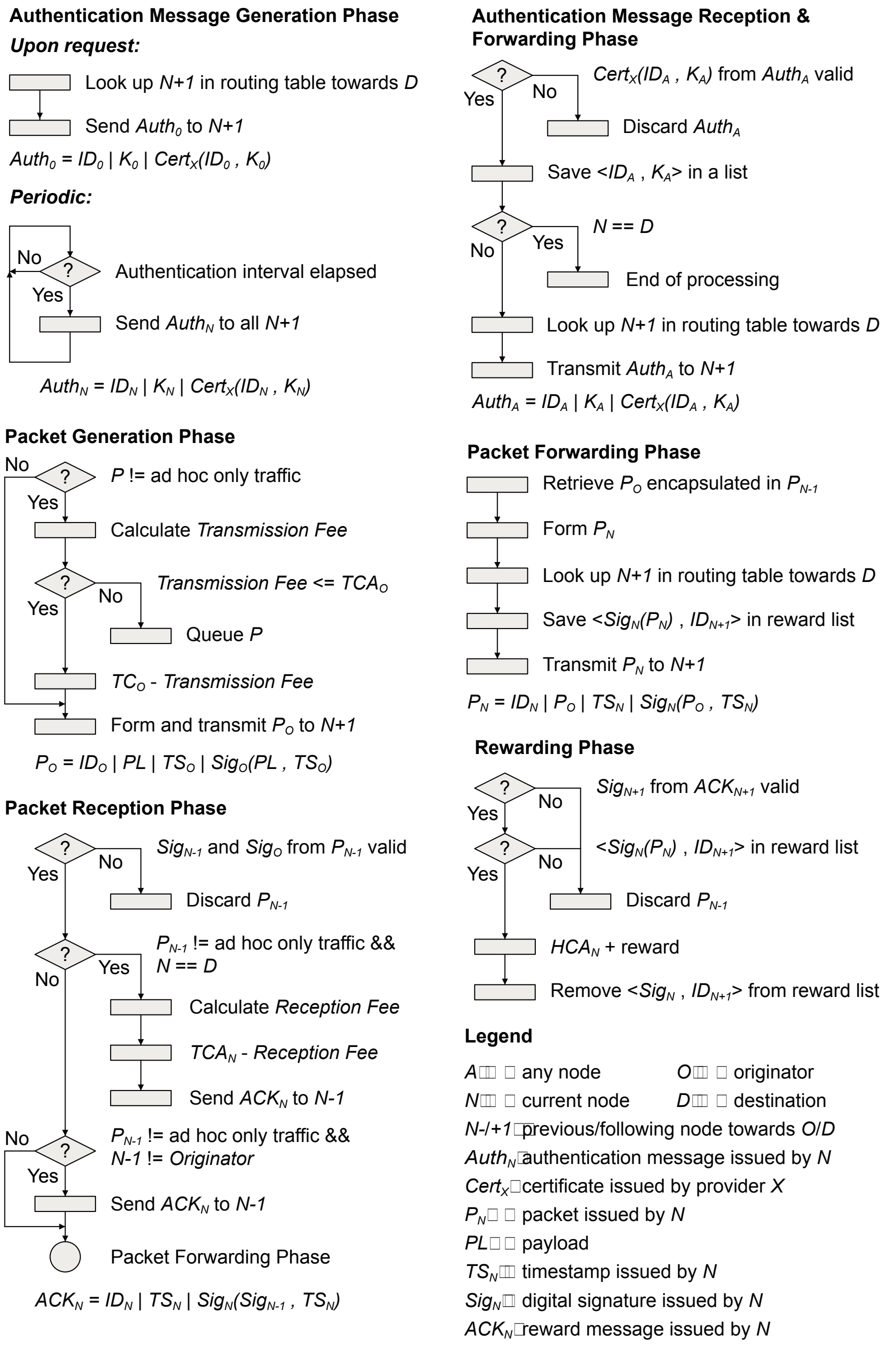


Fig. 2. CASHnet operation in detail

2.2 Nuglet

The Nuglet [1] cooperation scheme has the following main principle: Every time a node wants to transmit a self-generated packet, it has to pay with *Nuglets*. The amount corresponds to the estimated number of nodes between the sender and receiver (intermediate nodes). Every time a node forwards a packet it receives one *Nuglet*.

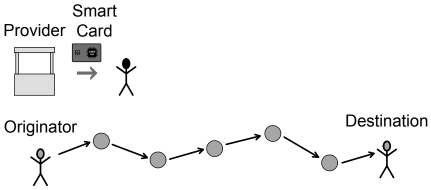


Fig. 3. Nuglet example scenario

Figure 3 shows an example scenario for the Nuglet scheme. As the Nuglet scheme was designed for ad hoc only networks, *Originator* and *Destination* reside in the same network. Also here, the node obtains a smart card. The authentication is done on a node-to-node session basis as the packet travels to the destination. The *Originator* will pay for the estimated number of intermediate nodes located on the path toward the destination. Each node stores the rewards for nodes, from which it has previously received forwarded packets. A synchronization protocol runs in a periodic interval to transfer all pending rewards to reachable nodes. A detailed description of the Nuglet scheme can be found in [1].

2.3 Comparison

Both schemes rely on tamper resistant hardware and public key cryptography. Although the two schemes were targeted at different networks, they both follow a decentralized design pattern. The two schemes charge for the transmission of self-generated packets. In CASHnet the cost is related to the hop count to the gateway, in Nuglet to the number of intermediate nodes to the destination. If a node has not enough virtual money (Traffic Credits or Nuglets), it is not allowed to transmit its own packets. Both schemes stimulate the cooperation among nodes (forwarding packets) through rewards. A node, which forwards a packet receives 1 or more Helper Credits or 1 Nuglet respectively. In the Nuglet scheme, a node can only earn its right for transmission, i.e. it must forward enough packets to be able to send its own packets. The CASHnet scheme additionally allows a node to buy its right for transmission, using additional infrastructure in the network (i.e. Service Stations). Another difference lies in the distribution of rewards. In the Nuglet scheme, each node collects rewards for nodes from which it has received a forwarded packet. In a periodic interval these pending counters are synchronized, in a way that all so far collected Nuglet are transmitted to the current reachable nodes. The remuneration in CASHnet happens immediately after a node receives a forwarded packet such that it sends an ACK message to the previous hop.

Because our current implementation of the two schemes does not yet include the cryptographic functionality, the security mechanisms are left out from the evaluation. In Nuglet each pair of communicating nodes establishes a symmetric key session to reduce the computational overhead. CASHnet only uses public key cryptography. The high mobility in ad hoc networks is a disadvantage for the session establishment in Nuglet, whereas the power constraints of mobile devices might be a minor disadvantage for the public key operations in CASHnet.

3 Simulation Scenarios

We evaluate both schemes through simulations where we measure the amount and frequency of starving event in the network, i.e. nodes that can not transmit, because they run out of virtual money (Traffic Credits or Nuglets) and use this as an indicator for the liveliness of the network. Also we give results on the overall packet delivery ratio as well as generated overhead and packet drop reasons. We adjusted our schemes' parameters to match the Nuglet scheme to provide a solid foundation for the comparative evaluations.

For the simulation we use ns-2 [7], where we implemented a version of the Nuglet and the CASHnet scheme including the charging and rewarding functionality and leaving out the security mechanisms. In particular, we used the wireless and mobility extensions [8] with an extended version of the AODV protocol called AODV+ [9], which adds Internet gateway discovery support.

Figure 4 shows our simulation scenarios. We only consider a single multi-hop cellular network to be compatible with the Nuglet schemes, which is targeted at mobile ad hoc networks. All nodes in the network send their packets to the gateway. The simulation scenario for Nuglet differs from the CASHnet scenario by removing the Service Stations and replacing the gateway by a normal mobile node.

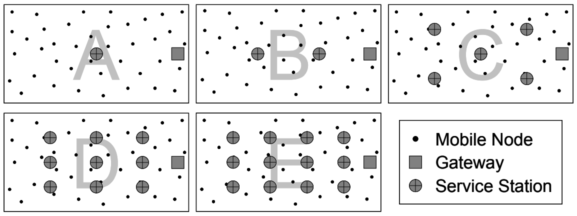


Fig. 4. Simulation Scenarios

Table 1 lists the parameters for the simulations. Except for the scheme specific properties, all parameters are identical. Within an area of 1500m x 800m we deploy 40 nodes. The nodes move according the random waypoint model using pre-generated movements files. We vary the number and the distribution of deployed Service Stations (none for Nuglet and 1, 2, 5, 9 and 12 for CASHnet) as shown in Fig. 4 as well as the packet generation interval at the CBR traffic sources (1, 2, 5 and 10 s). In total we investigate 24 (6 x 4) simulation scenarios and for each of the scenarios we conduct 20 simulation runs using 20 independent movement files.

Table 1. Simulation parameters

Parameter	Value	
	Nuglet	CASHnet
Space	1500 m x 800 m	
Number of nodes	40	
Transmission range	250 m	
Mobility model	random waypoint	
Speed	uniformly distributed between 1 and 10 m/s	
Pause time	uniformly distributed between 0 and 20 s	
Packet generation rate	0.1, 0.2, 0.5 or 1.0 pkt/s	
Routing	AODV	
Simulation time	900 s	
Initial virtual money account state	100 Nuglets	100 Traffic Credits
Initial real money account state	—	500
Nuglet synchronization interval	5 s	—
Traffic/Helper Credits exchange rate	—	1:1
Exchange thresh. at Service Stations	—	10 Helper Credits
Distance thresh. to Service Stations	—	50 m
Number of Service Stations	—	1, 2, 5, 9 or 12

In both schemes, the amount of initial virtual money is set to 100. To reflect the ability of a CASHnet node, to buy its right for transmission from available Service Stations, each node also has a real money account initially set to 500. Real money does not exist in the Nuglet scheme and is not equal to virtual money, as it must be exchanged first. Therefore, we believe the comparison to be fair in a sense that both schemes have the same initial situation according to their abilities. In the Nuglet scheme, a node needs to find other nodes and forward their packets to earn Nuglets. In the CASHnet scheme, a node needs to find a Service Station to exchange the Helper Credits and the real money against Traffic Credits. The exchange threshold defines the minimum amount of Helper Credits necessary before a node exchanges them into Traffic Credits at the Service Station. The distance threshold specifies the maximum distance between a node and a Service Stations to be able to exchange the Helper Credits. We measured the frequency of occurrence of starving events and the duration of the starvation. In addition we analyzed the overall goodput, the generated overhead and the reasons for dropped packets.

4 Simulation Results

First we investigate the starvation properties of both schemes. Then we discuss the overall protocol performance. Figure 5, 7, 8, 9 and 10 combine the mean results over the 20 simulation runs from all 24 scenarios. Each label marking on the x-axis consists of two lines. The first line indicates the number of Service Stations used (1, 2, 5, 9, 12 for CASHnet and 0 for Nuglet). The second line lists the packet generation interval (1, 2, 5, 10 s). The four packet generation intervals are separated by vertical lines.

4.1 Starvation

With starvation we describe the nodes inability to transmit self-generated packets due to lack of virtual money (Traffic Credits or Nuglets). Figure 5 contains the average starvation length for a node. Considering the total simulation time of 900 seconds, the two schemes perform poorly under high network load, with CASHnet being a little better. We find that CASHnet performs quite well under low network load, much in contrast to Nuglet.

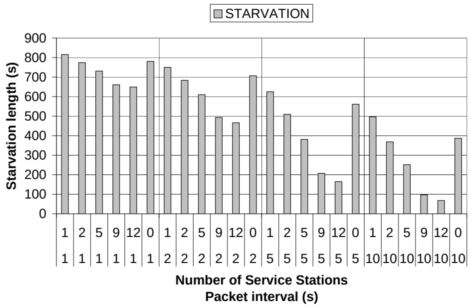
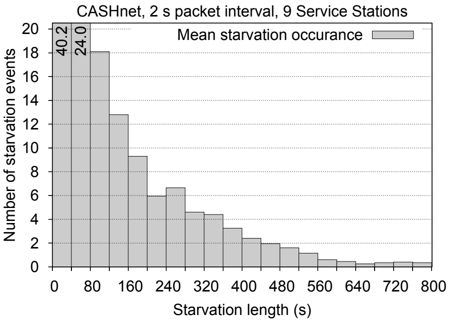


Fig. 5. Starvation length of a node in CASHnet & Nuglet

Both schemes charge for sending self-generated packets and reward for forwarding packets. In Nuglet, a node has only one source of income for virtual money (Nuglets): it has to forward packets from other nodes. In CASHnet, a node has two sources of income for virtual money (Traffic Credits): it can exchange the virtual money earned while forwarding packets from other nodes (Helper Credits) or pay with real money. We find that a self-perpetuating cycle of virtual money, which is assumed by the Nuglet scheme, is difficult to achieve. In such a cycle, each node always receives enough virtual money to be able to transmit self-generated packets. Under low network load (packet interval 10 s), a node in Nuglet starves in average for 43% of the simulation time, whereas in CASHnet the average starvation length is only 8% of the simulation time in the same scenario. This shows, that a node can not cover the cost of sending its own packets solely by forwarding packets from other nodes.

CASHnet performs worse than Nuglet in scenarios with 1 Service Station. In CASHnet, Traffic Credits can only be obtained at Service Stations, whereas in Nuglet only one virtual currency exists and is distributed directly to reachable nodes. When deploying 1 Service Station, the simulation area can not be covered sufficiently in a way that most of the nodes get enough opportunities to fill their Traffic Credits account.

To see the actual distribution of the starvation events, we categorized the events according to their lengths Figure 6(a) and 6(b) show the average distributions of 20 simulation runs for the scenario with a packet interval of 2 s and 9 Service Stations for CASHnet and Nuglet respectively. We see that the average number of nodes starving for the complete simulation time is rather low.



(a) CASHnet

(b) Nuglet

Fig. 6. Mean number of starvation events per duration category

4.2 Goodput, Overhead and Packet Drop Reasons

Figure 7 shows the goodput. We define goodput as the number of received packets divided by the number of sent packets. The goodput in CASHnet is worse or equal to Nuglet in scenarios with 1 and 2 Service Stations and better with 5, 9 and 12 Service Stations. CASHnet performs considerably better than Nuglet under high packet generation rate. CASHnet provides a 88% increase in goodput compared to Nuglet with a packet interval of 1 s and 12 Service Stations. Under low network load the improvement for CASHnet is lower. 39% increase in goodput with a packet interval of 10 s and 12 Service Stations. However, the goodput is very low for both schemes, for different reasons, which we analyze in the following paragraphs.

In Figure 8 the outcome of the sent packets is shown. We distinguish between received packets, packets dropped because of lack of virtual money and

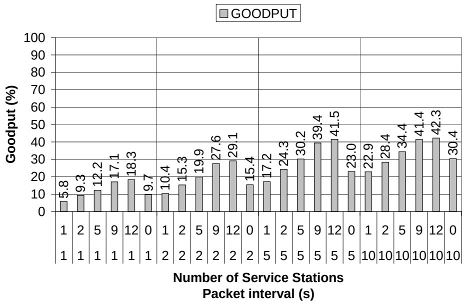


Fig. 7. Goodput for CASHnet & Nuglet

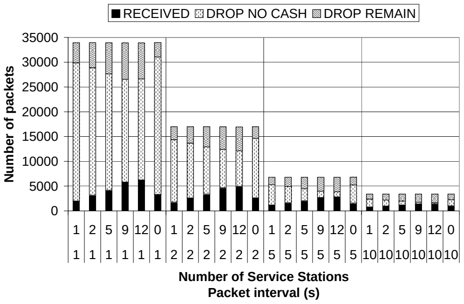


Fig. 8. Packet sent outcome for CASHnet & Nuglet

packets dropped for other reasons. The other packet drop reasons will be discussed afterwards. We see that while the number of Service Stations increases, the number of received packets follows as expected, but at the same time the number of packets dropped for other reasons increases too. While we can increase the initial virtual money account on the nodes to reduce the drops caused by lack of money, we have to consider the overhead introduced by the CASHnet scheme. In CASHnet we see, that increasing the number of Service Stations does not automatically decrease the number of packets dropped due to lack of Traffic Credits. This behavior can be observed when we compare the results for 9 and 12 Service Stations for a packet interval of 1 s.

Figure 9 illustrates the actual overhead introduced by both schemes. It shows the amount of packets sent and received as well as reward messages (CASHnet ACK/Nuglet SYNC). In CASHnet, the overhead is much higher than in Nuglet because every packet is rewarded immediately on a per-hop basis by an ACK packet, which increases the nodes Helper Credits account. In Nuglet, each node collects the rewards for its neighbors in personal accounts and transfers this virtual money periodically to all reachable nodes. A former neighbor node, that is not reachable at the time of the synchronization looses all its earned virtual money on that node. When comparing scenario B (2 Service Stations) with scenario C (5 Service Stations) from Figure 4, we see that increasing the number of Service Stations not automatically increases the overhead in CASHnet, and at the same time helps to increase the number of received packets.

The different reasons for dropped packets are displayed in Figure 10. We retrieved the following drop events from the trace files: lack of virtual money (NO CASH), no available route (NO ROUTE), routing loop (LOOP), MAC layer callback timer (CALLBACK) and delay in ARP (ARP). However, only NO CASH, NO ROUTE and CALLBACK events have considerable impact. LOOP and ARP events occur very rarely and their frequency does not change much between the different scenarios. In both schemes, the main reason for packet drops is the lack of virtual money. In Nuglet, it is difficult to generate enough traffic to build up a self-perpetuating cycle of virtual money. In CASHnet, a node has the possibility to buy its right for transmission. However, due to the

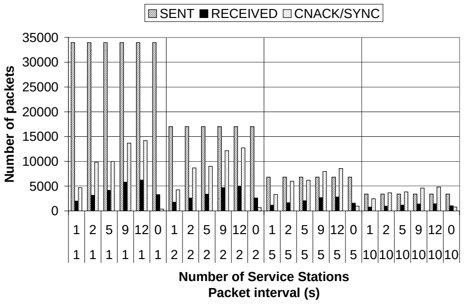


Fig. 9. Overhead for CASHnet & Nuglet

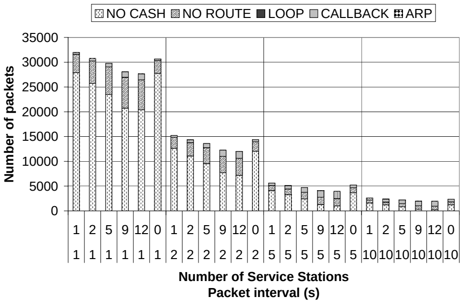


Fig. 10. Packet drop reasons for CASHnet & Nuglet

dependency on the fixed Service Stations as the only place for obtaining Traffic Credits, the positive effect of having two sources of income for virtual money is reduced. The second major packet drop reason is the unavailability of a route. In the simulation runs we used an extended version of AODV. When a route is not available in AODV, a route request is send and the packet will be retained until the route requested succeeds or times out. In CASHnet, we suspect the high protocol overhead to be reasons for the high number of unsuccessful route requests.

5 Summary and Outlook

We presented CASHnet, our cooperation and accounting strategy for hybrid networks, which uses a highly decentralized accounting and security architecture. It allows selfish nodes and supports cost sharing between sender and receivers located in different subnetworks. To put the performance of our scheme in context with other work in this area, we compared the CASHnet with the Nuglet scheme, which was also explained in this paper. We implemented both schemes in ns-2 and evaluated them through simulation runs. We monitored the network liveliness and the overall network performance.

As a result from the evaluation we see that the goal of the Nuglet scheme, that is a self-perpetuating cycle of virtual money, is difficult to achieve. We find that in CASHnet nodes already starve less than in Nuglet with only 2 Service Stations deployed. However, the high protocol overhead of CASHnet weakens the positive effect of an additional source of income for virtual money.

For CASHnet we see room for improvement in the granularity of the charging and rewarding mechanisms. This would help to reduce the overhead. In our current approach, we use Service Stations as low-bandwidth, low-cost terminals for buying and exchanging virtual money (similar to loading a prepaid card). We are currently investigating the possibility of changing the role of the Service Station. The simple combination of gateway and Service Station is more expensive and therefore might pose a risk for the provider. It also might not be possible to install gateways at certain locations. However, to keep the multi-hop cellular network alive, the nodes need possibilities (Service Stations) to fill their Traffic Credits account.

Using other mobility models with more realistic user behavior and adapting the deployment of Service Stations accordingly could also greatly improve our schemes performance. Additionally, the generic behavior of the customers themselves could be made more realistic, e.g. when running out of virtual money, the movement direction changes to the closest Service Station.

Further work will include more extensive simulation runs to determine the amount and location of Service Stations required for minimal packet loss as well as the implementation of a prototype of our CASHnet scheme using JavaCards. We will analyze our security mechanisms in terms of effectiveness against different attack types and resource consumption. Also, we will study possible extensions to our scheme and optimize the relation between charging and remuneration.

References

1. Buttyán, L., Hubaux, J.P.: Stimulating cooperation in self-organizing mobile ad hoc networks. ACM Mobile Networks & Applications **8** (2003) 579–592
2. Buchegger, S., Boudec, J.Y.L.: Performance analysis of the confidant protocol (cooperation of nodes - fairness in dynamic ad-hoc networks). In: Proceedings of 3rd ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc), Lausanne, Switzerland (2002) 226–236
3. Zhong, S., Chen, J., Yang, Y.R.: Sprite: A simple, cheat-proof, credit-based system for mobile ad-hoc networks. In: Proceedings of 22nd IEEE INFOCOM. Volume 3., San Francisco, CA, USA (2003) 1987–1997
4. Salem, N.B., Buttyán, L., Hubaux, J.P., Jakobsson, M.: A charging and rewarding scheme for packet forwarding in multi-hop cellular networks. In: Proceedings of 4th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc), Annapolis, MD, USA (2003) 13–24
5. Lamparter, B., Paul, K., Westhoff, D.: Charging support for ad hoc stub networks. Elsevier Journal of Computer Communications **26** (2003) 1504–1514
6. Weyland, A., Braun, T.: Cooperation and Accounting Strategy for Multi-hop Cellular Networks. In: Proceedings of 13th IEEE Workshop on Local and Metropolitan Area Networks (LANMAN), Mill Valley, CA, USA (2004) 193–198
7. Breslau, L., Estrin, D., Fall, K., Floyd, S., Heidemann, J., Helmy, A., Huang, P., McCanne, S., Varadhan, K., Xu, Y., Yu, H.: Advances in network simulation. IEEE Computer **33** (2000) 59–67
8. Broch, J., Maltz, D.A., Johnson, D.B., Hu, Y.C., Jetcheva, J.: A performance comparison of multi-hop wireless ad hoc network routing protocols. In: Proceedings of 4th ACM/IEEE International Conference on Mobile Computing and Networking (MOBICOM), Dallas, TX, USA (1998) 85–97
9. Hamidian, A.: A study of internet connectivity for mobile ad hoc networks in ns 2. Master's thesis, Lund University, Sweden (2003)