Avoiding Route Breakage in Ad Hoc Networks using Link Prediction

Erik Weiss*, Kamil Kurowski*, Sven Hischke+, Bangnan Xu+
* : Communication Networks, Aachen University of Technology, Kopernikusstr.16, D-52074 Aachen, Germany
+: T-Systems, Technologiezentrum, Am Kavalleriesand 3, D-64295 Darmstadt, Germany

Ph: +49 241 8028575, Fax: +49 241 8022242, Email: erik.weiss@comnets.rwth-aachen.de

Abstract

This paper presents a new concept to improve Ad Hoc Routing Protocols exploiting IEEE 802.11a Link Adaptation information. The IEEE 802.11a Link Adaptation information is used to predict the link stability and link lifetime. After introducing the IEEE 802.11a MAC Layer information and its transmission modes, the paper reviews some insights of the IEEE 802.11a Link Adaptation behaviour. Based on the Link Layer Information, new route maintenance Protocols ERRA (Early Route ReArrangement) and ERU (Early Route Update) are proposed to improve the active route maintenance in Ad Hoc Networks.

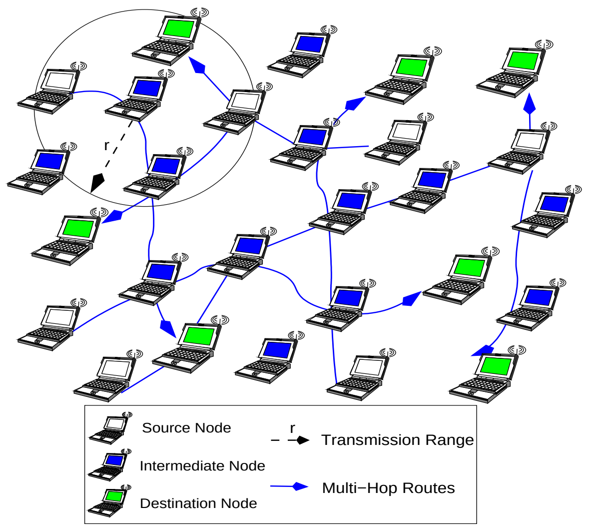

Figure 1.1: Wireless Ad Hoc Network

1 Introduction

Internet access is becoming increasingly important. Furthermore, the trend is towards the wireless world, providing public access to the Internet via wireless devices at high data rates. *Wireless Local Area Networks* (WLAN) like IEEE 802.11a work at the 5 GHz band, supporting transmission rates up to 54 Mbit/s. Due to the high attenuation at 5 GHz the coverage is limited. To extend the coverage, multi-hop routes have to be established. Being wireless enables the user to be mobile, therefore the network has to deal with the mobility, and all the effects introduced by a dynamic changing network topology.

High throughput and limited transmission range makes WLAN systems reasonable for areas with a high population density and users with the need for high data rates. Such places are called Hotspots like airports or fairs. Figure 1.1 shows the future idea of the Mobile Internet. Due to the limited transmission range the needed density of Access Point/Router has to be very high. The deployment of such a high number of Access Points would be economically inefficient.
This could be reduced by either increasing the transmit power or enabling intermediate terminals to forward the data to users outside the AR range. Increasing the transmit power burdens the batteries of the mobile node and increases exposure of operators to radio waves along with their yet undetermined health risk.

State of the Art

The solution is to expand the fixed infrastructure using multi-hop connections. To handle the mobility and fast topology changing on the network, Ad hoc Routing Protocols have been developed.
Routing protocols are divided in two groups, the proactive and reactive protocols. The reactive protocols request a route when needed. Whereas proactive protocols permanently maintaining routes to all network members. Thus, proactive approaches can use the route when requested, therefore minimizing the packet delay. Reactive protocols avoid to maintain unneeded routes, but with a higher route discovery and packet delay. Furthermore, hybrid approaches have been developed.
However, when a built route breaks, all routing approaches try to recover the connection. Most approaches inform the source node and it starts a completely new discovery process, thereby a large number of messages are exchanged, and the network is flooded. Some routing protocols try to keep the route discovery locally around the breakage, hence the flooding is limited. Nevertheless all approaches only react in a proper manner when the link is already broken. This leads to a high number of lost packets as well as increased route rediscovering and packet delay. This

paper presents a new approach. Our proposal does not wait until the link breaks. Based on Link Adaptation information we predict the link state. We already start to rearrange the route before the link breaks. Lower layer, especial the Link Adaptation provides information that allows predicting the link conditions. We present two new route rearrangement protocols based on the prediction, the Early Route Rearrangement (ERRA) and the Early Route Update (ERU). The presented approaches prevent unnecessary signalling, avoid packet lost and minimize packet delays. Therefore, our approach uses the Ad Hoc Network capacity more efficient than existing protocols. We structured the paper as followed: first we start with a brief overview about the IEEE 802.11.To explain the fundamentals for the prediction we focus on IEEE 802.11a Link Adaptation behaviour in section 2. Within the third section we present how the Link Adaptation supports the necessary prediction information. Finally, new signalling procedures (ERRA) and (ERU) are presented to use the prediction effectively. The last section concludes our consideration and describes further enhancements.

2 IEEE 802.11a Medium Access Layer Information

The IEEE 802.11a Medium Access Control (MAC) layer is mainly the same as the MAC layer of 802.11b and the legacy 802.11. The main difference to 802.11a are the transmission modes [1][2]. 802.11a can chose between eight coding schemes, so called ”*PhyModes*” (cf. table 2.1). IEEE 802.11 uses a distributed MAC protocol, the *Distributed Coordination Function* (DCF). The 802.11 standard also defines a *Point Coordination Function* (PCF) but no vendor has implemented it. The DCF is based on *carrier sense multiple access with collision avoidance* (CSMA/CA). For mobile nodes (MN) it is not possible to monitor the air interface while sending. Hence, the DCF uses backoff and request-/ clear to send (RTS/CTS) mechanisms to avoid collisions. Details of the IEEE 802.11a MAC protocol are shown in [3].

2.1 IEEE 802.11a Transmission modes

IEEE 802.11a has eight different transmission modes. The standard itself does not specify any rules for selecting the PHY mode. Figure 2.1 shows the physical layer PDU. The first four bits within the preamble refer to the PHY mode with which the data is coded. Figure 2.2 shows the Packet Error Rate (PER) versus C/I (Carrier to Interference). Higher transmission modes are capable to deliver higher data rates, but nevertheless, they also need a remarkable higher C/I. In table 2.1 the available modes are listed together with the maximum data rate and the bits per OFDM symbol. Due to the dependence between C/I and useable trans-

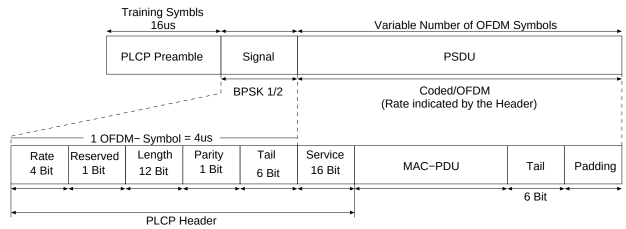

Figure 2.1 Physical PDU frame format of IEEE 802.11a ODFM PHY

mission mode, IEEE 802.11a allows to change the transmission mode when the channel quality is decreasing. Decreasing the channel quality could have several reasons. In Figure 2.3 the maximum reachable data throughput is shown for applying the 802.11 MAC with the DCF and the use of RTS/CTS. It could be seen that from the raw data rate of 54 Mbit/s at the transmission mode of QAM-64, 802.11a can support around 36 Mbit/s.

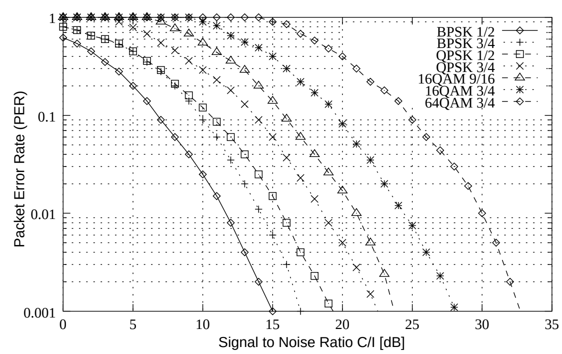

Figure 2.2: Packet Error Rate versus C/I

The IEEE 802.11a system offers the opportunity to choose an appropriate PhyMode. For every connection and every data packet, the PhyMode is chosen separately, depending on the received C/I. The idea is to take the envelop of Figure 2.2 to always choose the PhyMode with the best balance between throughput and PER. Terminals in a real system cannot measure the C/I, since each terminal only receives ’Energy’.

Data rate (Mbit/s)	Modulation	Coding rate (R)	Data Bits per Symbol
6	BPSK	1/2	24
9	BPSK	3/4	36
12	QPSK	1/2	48
18	QPSK	3/4	72
24	16-QAM	1/2	96
36	16-QAM	3/4	144
48	64-QAM	2/3	192
54	64-QAM	3/4	216

Table 2.1: Mode Dependent Parameters

The terminal cannot determine between signal power and interference power. Two ways to estimate the signal-to-noise ratio exist. Terminals could either measure the interference power within breaks or they count the successful received and lost packets. The ratio between successful and lost packets, in combination with Figure 2.2 leads to the current C/I. At the Chair of Communication Networks a simulator was built to simulate IEEE 802.11a/e together with HiperLAN/2 for coexisting questions.

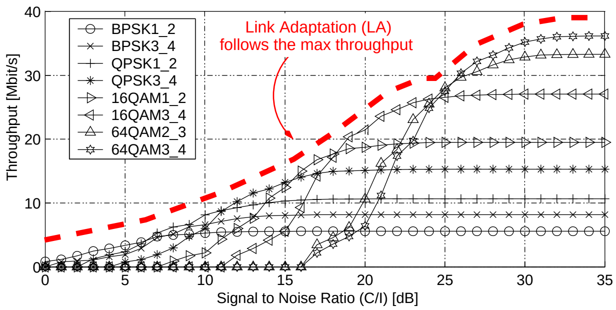

Figure 2.3: Max. Reachable Throughput per PhyMode

IEEE 802.11a uses a Link Adaptation based on successful/lost packets. To improve performance, an easy but valuable approach is used. After successfully receiving a reasonable number of packets, the PhyMode with higher date rate is used and after losing a certain number of packets a lower transmission mode is used. The main advantage of the implemented LA algorithm is the fast reaction to link changing. But some unnecessary changes may occur. Because even when the LA has found an optimum transmission mode it tries to improve it further, although this trying fails.

3 IEEE 802.11a Link Adaptation behaviour

This section shows the link adaptation behaviour of IEEE 802.11a. For the understanding of how routing protocols profit from LA information, it is important to understand how the LA works. As an example, we use a simple scenario: five stations are placed in a row, each in at a distance of 20 meters from the previous. The third station moves in the orthogonal direction between its initial point (40,0) and (40,150).

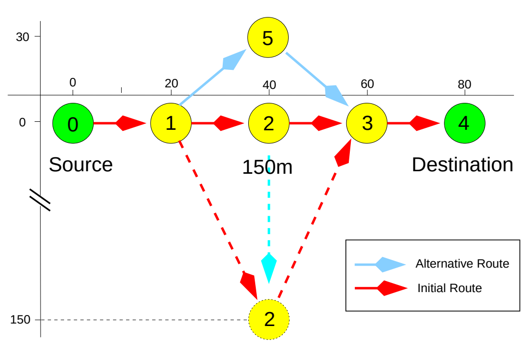

Figure 3.1: 5 Terminal, Number 2 Moves 150m and Returns

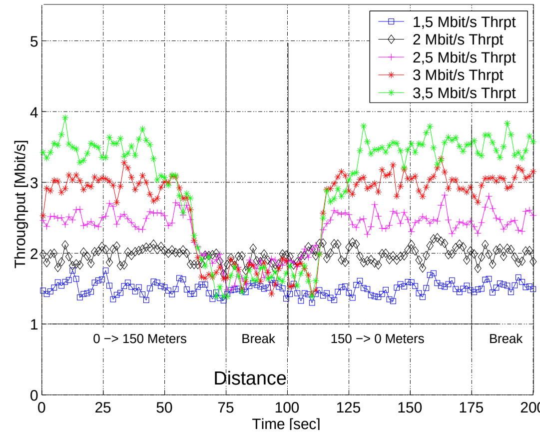

Figure 3.2: Resulting Throughput for 1.5 Mbit/s and 3.5 Mbit/s

Figure 3.1 shows the described scenario. The source node generates a traffic stream of constant bit-rate (CBR) traffic. The chosen packet length is 2000 bytes; and the RTS/CTS mechanism is used. At the beginning a route is established between the source and the destination. Node 2 moves 75 seconds with 2 m/s velocity, then stays there for 25 seconds and returns with the same speed. It makes another 25 second break, before starting again. Figure 3.2 shows the throughput at different load situations. The graphic shows five simulations, for each the traffic increases by 500 kbit/s starting with 1.5 Mbit/s up to 3.5 Mbit/s. With increasing the distance from node 2, the transmission to and from node 2 needs more bandwidth. Therefore, the channel capacity suffers from the increasing distance. A 2000 byte data packet sent with 64-QAM ½ needs 75 OFDM symbols (additionally some symbols for the preamble) for the data and transported with BPSK ½ the packet needs 667 OFDM symbols. Hence transmitting with BPSK ½ takes approx. 8.5 times longer, compared with 64-QAM ½. Therefore a load higher than 2.5 Mbit/s cannot be supported when the terminal distance is too large. Figure 3.3 shows the average transmission modes seen by node 2 for five different traffic offers.

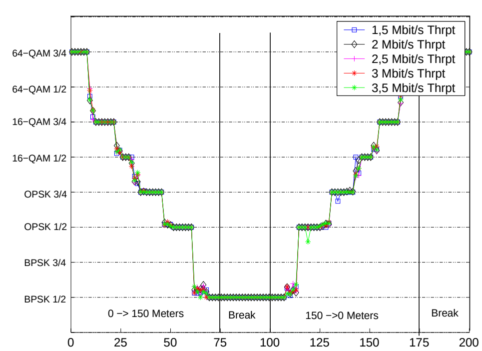

Figure 3.3: Transmission Modes for Node 2, Load from 1.5 – 3.5 Mbit/s

Due to the enlarged distance to node 2 the link adaptation adapts the transmission mode to maintain the connectivity. Figure 3.3 shows an important aspect of using the link adaptation as an indicator for the link state. Link State Routing (LSR) [6] would react to traffic streams with high load, due to the traffic descent, but LSR could not react to low traffic, since there is no indication. Using the Link Adaptation information, the ad hoc routing protocols can react to the link degradation even at a low traffic situation. Figure 3.3 shows that the PhyModes 64 QAM 1/2 and BPSK 3/4 are not used due to the similar C/I and power requirements to their neighbouring PhyMode.

4 Link-Layer-Information for the Ad Hoc Routing

Link Adaptation information could improve the performance of ad hoc routing mechanisms. The information about the chosen transmission modes in the past and in the present makes it capable to predict the near future of the link. In the described scenario (cf. Figure 3.1) the node returns after 150m. Even transmitting over such a large distance is very expensive. Furthermore if the node continuously departs the link will break. Before it breaks the LA adapts whenever necessary the transmission mode. Due to this behaviour we are able to predict the link breakage. Continuous decreasing of the transmission mode is a hint for the network layer that the link may break. This information triggers a mechanism to change and to adapt the route for the actual situation. Being able to predict a link breakage has a large benefit. The usual protocols can only react after the link is broken, while with LA information they can act before the breakage occurs.

We have discussed the LA behaviour so far. But the detailed LA functions are out of the scope of this paper. This issue will be discussed in another paper. We assume in this paper that the current, past and future states of the link connection are maintained and predicted by the LA. As the link may break/change, either the LA informs the routing protocol including the link characteristics, or the routing protocol does the prediction by monitoring the LA.

4.1 Proper Actions for upcoming Link Break

Assuming that the Link Adaptation delivers the necessary information about the link state characteristics, this information triggers appropriate actions, either to rescue the link and prevent the expensive route rediscovery or to guarantee a required link quality by finding a new route. Several proper actions are conceivable. Here we present two of them. The node that monitors the incoming and outgoing links knows if one of them, none of them or both are being adapted. This enables the node to distinguish three different cases.

1) The node recognizes that the outgoing link is adapting the transmission mode but the incoming link is stable. Thus, the next node on the route seems to move.
2) The node recognizes that the incoming as well as the outgoing links are adapting the transmission modes. Hence, the node itself seems to move.
3) The incoming link is adapting to changes but the outgoing link remains constant. Therefore, the previous node on the route seems to move.

In the scenario depicted in Figure 3.1 all three cases[i] could be found. Node 1 experiences case 1, node 2 experiences case 2 and node 4 experiences case 3. Due to the observed changes one node starts the route maintenance procedure.

Early Route ReArrangement (ERRA)

Ad Hoc On Demand Distance Vector (AODV)[4] routing already has a mechanism to repair a link break local around the interruption. AODV proposes to use this ‘Local Repair´ for the use with the multi-cast extension, MAODV [6]. We renamed the action because we are not repairing the link but we are rearranging the route. Therefore, we call this action the Early Route ReArrangement (ERRA); nevertheless the actions done are similar to the AODV Local Repairs steps. The upstream node decides to start the rearrangement procedure when it recognises that the link to its downstream neighbour will break. In Figure 4.1, node 5 is the upstream node, it increments the sequence number [4] for the destination and initiates a Rearrangement Request message (ERRA_REQ) to the destination. The time to life (TTL) for the ERRA_REQ is calculated in the same way as for the Local Repair.

$$\max(\mathrm{MIN\_REPAIR\_TTL},\ 0.5 * Nr_{hops}) + \mathrm{LOCAL\_ADD\_TTL}$$

Whereas, Nr_{hops} is the number of hops between source node and upstream neighbour of the interruption. Thus, update attempts should be invisible for the source node. To prevent that the still reachable node 10 replies with its current route entry, we propose the broadcasting of the EARR_REQ message with a higher transmission mode, than the current one. Subsequently node 10 is not able to receive the ERRA_REQ message. Each node with a valid route to the destination will reply with an ERRA_REP message**. This leads to an alternative route using a higher transmission mode** (cf. Figure 4.1). Depending on the number of hops for the alternative route and the state of the old route, the initiator node can determine whether to change or not. When the alternative route hop count is unequal to the old hop count, the initiator must send an ERRE_INFO to the source node containing the new hop

---

[i] Obviously the three cases can also occur when the neighbour terminals are moving, however the upcoming situation could be handled equally.

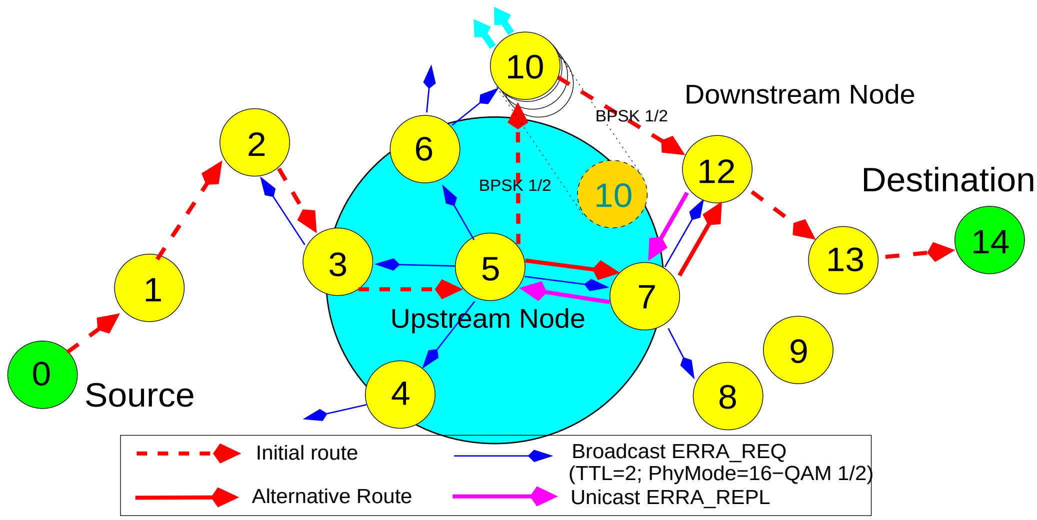

Figure 4.1: Early Route ReArrangement (ERRA)

count. Alternatively, to transmitting the ERRA_REQ with a certain PhyMode, node 5 knows that the reply from node 10 results from the old route, therefore node 5 discards the reply of node 10. Hence, node 12 or each other node that has a valid route to the destination, excepted node 10, replies with an ERRA_REP. The reply creates an alternative path and rearranges the route.
Using the ERRA approach avoids packet lost and transmission breaks. In addition, the number of control messages is minimized**.** Therefore, as well the ratio data to signalling packets increases compared to approaches without prediction. However, the biggest benefit of the link break prediction is not used. The still existing link could be used to communicate with the nodes behind the upcoming break**.** A large number of approaches using this advantage could be used, one of them is presented in the next section.

Early Route Update (ERU)

This section presents an approach that uses the still existing link across the upcoming link break. Thus, through the link layer information the nodes are able to figure out the upcoming link break but the upstream node can still communicate with the downstream node behind the expected interruption. Additionally we assume that the lower layers are permanently sensing the channel. Therefore, each node has an up-to-date list of its neighbourhood. Having this feature assists the routing. However, neighbour discovery messages are also useable.
Figure 4.2 shows the fundamental steps of the Early Route Update approach. Node 2 monitors its link to node 3. Node 2 notices that the transmission mode for this link is decreasing. Therefore, node 2 expects an interruption. Node 2 requests an route update and adds an ERU_REQ message to a regular data packet[ii] for the destination (node 7). The ERU_REQ message contains the neighbour table from node 2, a time to life (TTL) field set to 1, and a unique sequence number identifying the route. Node 3 also observes the changing link condition; it receives the piggyback information from node 2. Within its routing layer, it monitors as well the incoming and outgoing link. At this point, node 3 determines whether it should forward the information or if itself uses the information. This decision is based on the behaviour of its outgoing link. For example, in Figure 4.2 the link from node 3 to 4 is also decreasing the modes. Therefore, node 3 forwards the neighbour table to node 4 and increments the TTL field by one. Node 4 separates the neighbour table from the data packet and broadcasts the ERU_REQ message with the TTL set to two. If node 4 would also move rapidly it would increment the TTL and forward the ERU_REQ to the next downstream node (node 5). However, in Figure 4.2, finally, node 4 broadcasts the ERU_REQ message with a chosen TTL of two and containing the neighbour table from node 2. When a node receives the ERU_REQ message and knows one of the neighbour nodes or the initiator node (cf. Figure 4.2, node 8), it replies with an ERU_REP towards node 2 (cf. Figure 4.2, note 9 forwards the ERU_REP to node 2). This ERU_REP together with the broadcasted ERU_REQ creates the alternative path. When node 2 receives the ERU_REP from one of its neighbours, an alternative reverse path [4] is built. Depending on the alternative route hop count and the conditions for the old link, the node may confirm the reverse path by using it. Using the reverse path builds the forward path and rearranges the route. Note that the

[ii] To limit the overhead the message is piggybacked, hence this is only feasible when the data packet plus neighbour table size is smaller or equal to the max PDU size (IEEE 802.11 2304 bytes)

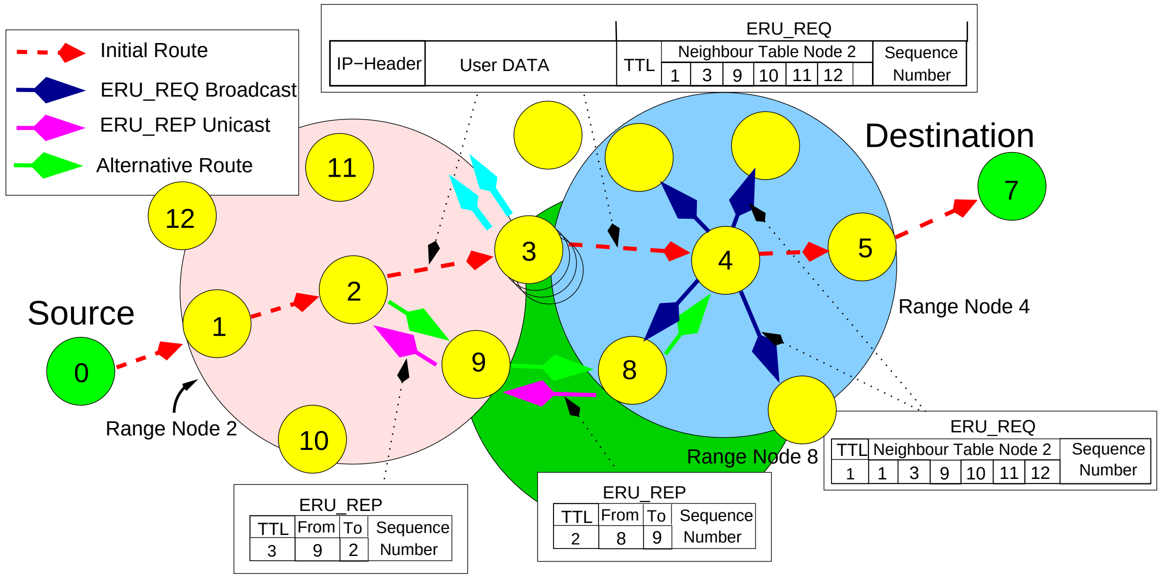

Figure 4.2: Early Route Update (ERU)

maximum hop count for the alternative route is the number of moving hops (here node 3) along the route, plus 2 (Figure 4.2; maximum hop count is four).When the hop count has changed the initiator node has to inform the source node using an ERU_INFO message containing the new hop count.

The ERU approach gives the opportunity to check the length of the upcoming link break, thereupon the TTL for the alternative route search is set accordingly. Due to sending the ERU_REQ attached to a normal packet, the signalling overhead is further minimized. In addition, the TTL calculation for the locally broadcasted ERU_REQ depends on the breakage size. Hence, only the absolute minimum numbers of broadcasts are initiated. This increases the routing protocol performance. The signalling packets are limited, route breakages are avoided, hence the delay is minimised. Finally routing protocols with link prediction and ERU uses the network capacity more efficient.

5 Conclusion

This paper shows the benefits given by recent systems, how ad hoc routing profits from methods to adapt the transmission modes. The Link Adaptation and their behaviour contain information, which are essential for the Network layer. We propose the information exchange between Link Adaptation and Network layer. Based on this information the Network layer is able to predict the link state and to initiates the proper action to prevent the link break or to keep the route optimized.

Previous sections explained two approaches for optimising and updating the route. The presented approaches limit the necessary signalling overhead to maintain a route to a minimum. Through avoiding link breaks the number of lost packets decreases and as well the packet delay is decreasing. Using the link prediction avoids unnecessary network flooding. Hence, network capacity could be used more efficient, with link prediction. Validating and improving the presented approaches via simulations, are our next steps.

6 Acknowledgement

This work was supported by T-Systems and the German research project IPonAir, funded by the German Ministry for Education and Research (BMBF). The authors would like to thank the members of the project for the valuable discussion.

References

[1] B. Walke, Mobile Radio Networks. New York, USA: Wiley & Sons Ltd., 1. ed., 1999.

[2] C. R. Nasser, B. Natarajan, Z. Wu, D. Wiegandt, S. A. Zekavat, and S. Shattil, Multi-carrier technologies for wireless communication. Boston, USA: Kluwer Academic, 1. ed., 2002

[3] IEEE LAN/WAN Standards Committee, “Part11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: High-speed Physical Layer in 5 Ghz Band,” IEEE Std 802.11a-1999, IEEE, New York, Nov. 1999.

[4] C. Perkins, E.M. Royer, S.R. Das. Ad Hoc On-Demand Distance Vector (AODV) Routing. IETF internet draft (draft-ietf-manet-aodv-13.txt), February 2003.

[5] Wireless Radio Access Protocol Simulator version 2. www.warp2.de. Chair of Communications Network

[6] C. E. Perkins. Ad Hoc Networks, USA, Addison-Wesley, December 2000.

[7] M. Engels. Wireless OFDM Systems, How to make them work? Boston, USA: Kluwer Academic, 2002