Enabling partial forwarding by decoding-based one and two-stage selective cooperation

Stefan Valentin*, Tobias Volkhausen*, Furuzan Atay Onat†, Halim Yanikomeroglu†, and Holger Karl*
*University of Paderborn, Germany, {stefan.valentin|volk|holger.karl}@upb.de
†Carleton University, Ottawa, Canada, {furuzan|halim}@sce.carleton.ca

***Abstract*— With Selection Decode-and-Forward (SDF) relaying, an accurate forwarding decision is essential to avoid error propagation and achieve full cooperation diversity. The methods in literature are either based on CRC or SNR thresholds, both resulting in poor end-to-end performance in practical wireless scenarios with non-block fading and channel coding. We propose to improve performance by accurate and frequent adaptation with *partial forwarding*. To achieve this in channel-coded systems, we introduce a *decoding-based* forwarding decision at the relay and propose (1) *Minimum Path Difference (MPD)* as a non-overhead and accurate decoding-based decision metric and (2) MPD-based SDF protocols with one and two decision stages. Our performance study validates that these protocols significantly improve SDF's end-to-end BER under realistic assumptions.**

I. INTRODUCTION

Cooperative relaying can provide substantial diversity gains by exploiting the broadcast nature of the wireless channel even when multiple antennas per node are not applicable. To organize the relaying process, cooperation protocols are required that achieve a special form of spatial and temporal transmit diversity, called user cooperation diversity [1]. In their seminal work [2], Laneman et al. show that so-called Selection Decode-and-Forward (SDF) protocols can reach full diversity (in order of cooperating nodes) if the relay perfectly prevents error propagation by forwarding only correct messages.

This ideal case, where the relay always makes a perfect forwarding decision, was assumed in many papers that followed. It implies (1) ideal channel coding, meaning that an outage (in terms of SNR falling below a threshold) can be directly translated in a decoding error at the relay, and (2) message-wise block-fading (i.e. constant channel state during a message's transmission). While these assumptions provided tremendous insight in the performance regions of SDF protocols, they are not sufficient for actually *achieving* these bounds in practice.

First, in practical systems, ideal channel codes cannot be assumed. Consequently, an outage does not directly map to a decoding error and SNR at best roughly approximates the Bit Error Rate (BER) of decoded messages [3]. Second, in realistic mobile scenarios, fading is a continuous time-correlated process where the channel state may change at any time [4]. Consequently, message-wise block fading cannot be assumed – instead the channel may vary several times per message; causing errors only in parts of a message.

So far, many papers assume that the relay uses either Cyclic Redundancy Check (CRC) [2] or an SNR-threshold [5, 6] to decide if there are any bit errors in a received message. However, both methods show considerable drawbacks with the above practical assumptions. While CRC-based decision performs excellent with message-wise block fading, it becomes ineffective if only parts of a message are corrupt. In such a case, a CRC-based SDF ineffectively drops the *complete* message even though *partial forwarding*, i.e. forwarding the correct parts of the message, would improve the end-to-end performance at the destination.

To decide which parts of a message can be forwarded, partial forwarding requires an independent observation for each small consecutive part of a message, called block. To improve performance, this frequent (block-wise) observation has to accurately assess a message's decoding errors without considerably decreasing the data rate. Such an efficient observation is not possible with SNR or CRC: CRC is efficient only when calculated over large blocks as a checksum adds overhead to each block and CRC's detection rate decreases for shorter blocks [7]. Measuring SNR comes at the cost of training symbols and is, therefore, only performed at the beginning of each message in many systems [8]. With this practical limitation, SNR can neither accurately characterize all parts of a message nor, as discussed above, decoding errors with practical Forward Error Correction (FEC) codes (cf. Sec. III of this paper).

To this end, we introduce a *decoding-based* forwarding decision at the relay enabled by an efficient observation of FEC decoding. In particular we, first, propose the Minimum Path Difference (MPD) as a new decoding-based metric for error characterization. MPD provides an accurate, continuous BER approximation similar to the A Posteriori Probability (APP) of a decoded bit in ideal MAP decoders [9]. Unlike APP, however, it relies on standard decoding functions of cellular and WLAN receivers and can thus be implemented without significantly increasing complexity. We describe MPD's integration into the widely-used Viterbi decoding algorithm and demonstrate its accuracy in Sec. III. MPD's main advantage is that its accurate observation requires no additional redundancy, training sequences or checksums. Hence, even calculating it for very small blocks does not reduce the available data rate.

As our second contribution, we propose two MPD-based cooperation protocols supporting partial forwarding (Sec. IV). The first protocol is a simple Threshold-based SDF (TSDF) approach using MPD to make forwarding decisions for each block. This enables partial forwarding of small blocks and,

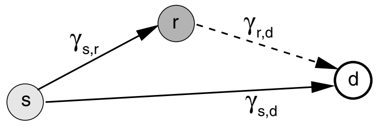

Fig. 1. Cooperation scenario with three single antenna nodes. Source s and relay r cooperate to reach destination d. The figure shows the *instantaneous* SNR values γ for the half-duplex channels used during phase 1 (solid line) and 2 (dashed line) of a cooperation cycle.

thus, allows frequent adaptation, which is especially beneficial when the channel changes at a fast rate (Sec. V). The second protocol, called Two-stage SDF (2SDF), combines CRC-based SDF and MPD-based TSDF in two decision stages. 2SDF benefits from both the accurate error detection of CRC and the partial forwarding capability of the first protocol and closely reaches the theoretical end-to-end performance bound even in practical scenarios.

II. SYSTEM MODEL AND BASIC SDF OPERATION

We consider the scenario in Fig. 1 where a single relay node r may employ SDF to cooperate with source node s for reaching the destination node d. For comparison, direct transmission from s to d is considered. Each node is equipped with a single omnidirectional antenna and may perform coding and decoding of messages. As in IEEE 802.11a/g transceivers, we assume the following coding procedure: First, an (inner) CRC code is employed and the resulting checksum is appended. Second, the resulting message is coded using an (outer) convolutional FEC code at rate R_c. Finally, the resulting coded message, called codeword c, is modulated and transmitted.

Each subsequent transmission cycle is split into two phases. In the first phase, s encodes its message m and broadcasts the resulting codeword c via channels (s,r) and (s,d) to r and d, respectively. With SDF, r now demodulates the signal received from channel (s,r) and obtains codeword $c_{s,r}$. After decoding, the possibly erroneous message $m_{s,r}$ is available at the relay. In phase 2 of SDF, r may either forward a newly encoded version of $m_{s,r}$ to d via channel (r,d), remain silent or transmit other, unrelated data. This binary forwarding decision depends on an estimate of the transmission errors in $m_{s,r}$. In this paper, we study several metrics to be used for this estimate. For example, as $m_{s,r}$ is CRC-coded, r is able to test the complete message for errors not corrected by FEC decoding. If the message is estimated to be correct, r encodes, modulates, and forwards $m_{s,r}$ in phase 2. In this case, d can receive and decode the resulting codeword $c_{r,d}$ to message $m_{r,d}$. Finally, d achieves diversity gain by combining this message with message $m_{s,d}$ received during phase 1.

To combine messages $m_{r,d}$ and $m_{s,d}$ at the destination, we assume the Maximum Ratio Combining (MRC) scheme as commonly used in diversity receivers [3]. Furthermore, for all nodes, we consider Physical layer (PHY) functions and parameters according to IEEE 802.11a WLAN transceivers [8]. At the transmitters, we exemplarily assume a standard CRC-32 code polynomial, convolutional FEC coding with generator polynomial $g_0 = 133_8, g_1 = 171_8$, code rate $R_c = 1/2$, and $4\,\mu$s BPSK modulation symbols. On the receiver side, coherent detection and soft-decision Viterbi decoding are assumed. During transmission, we assume a constant message flow and a Medium Access Control (MAC) scheme assuring orthogonal channels for the cooperation phases.

Algorithm 1: MPD-extended Viterbi decoding

```
Input: Codeword $c$ with $u$ code symbols: $c_1,\dots,c_u$;
       Codeword $\tilde{c}$ with $u$ code symbols: $\tilde{c}_1,\dots,\tilde{c}_u$
Output: Message $m$ with $u$ message symbols: $m_1,\dots,m_u$;
        Metric values $mpd$ per code symbol: $mpd_1,\dots,mpd_u$
  // Viterbi (1): Search minimum-weight path
1 $edge_{1,\dots,u}$ = findPath($c$);
  // Viterbi (2): Traceback over path
2 for $i = u,\dots,1$ do
3     $m_i = \text{messagesymbol}(edge_i)$;
      // MPD calculation adds line 4
4     $mpd_i = \text{diff}(\tilde{c}_i, \text{codesymbol}(edge_i))$;
5 end
6 return $m, mpd$
```

All studied channels reflect a Non-Line Of Sight (NLOS) situation with moving nodes. We consider frequency-flat, half-duplex, i.i.d. Rayleigh fading channels where small-scale fading is parameterized by a maximum Doppler shift according to velocity v. Instead of assuming block-fading per message, we model fading per modulation symbol using the common "Jakes-like" method with the "land mobile" autocorrelation function (Table 2.1 in [4]). This widely-used model is suitable for mobile indoor or urban scenarios with many stationary, uniformly distributed scatterers.

III. DECODING-BASED DECISION METRIC

First, we propose Minimum Path Difference (MPD) to approximate the BER *during* Viterbi decoding. Second, we show that this approximation is more accurate than SNR as measured in WLAN receivers and close to the ideal case.

A. MPD definition and Viterbi decoder integration

MPD approximates the BER by observing the trellis decoding process of a received codeword c and can be calculated based only on local redundancy information. Specifically, MPD expresses the minimum distance of c to the closest valid codeword in the trellis diagram. This expression includes the full decoding history and is calculated over all symbols of a codeword while traversing the path with the minimum weight in the trellis structure. Since this path is found and traced-back during normal Viterbi decoding, MPD can be calculated *during* the second step (traceback) of this standard algorithm.

Algorithm 1 shows a brief version of the standard Viterbi algorithm[1] extended by MPD calculation. The algorithm performs Viterbi decoding of a message m encoded at rate $R_c = k/n$ with n coded bits per k message bits. In total,

---

[1] For clarity, implementation details as quantization and pipelining are omitted. Details of Viterbi decoding are provided in standard literature [3].

message m consists of $u = l/k$ message symbols or l (uncoded) message bits and is decoded from codeword c which consists of u code symbols or l/R_c code bits. Viterbi decoding is performed in two steps: First, the algorithm searches for the trellis path which minimizes the accumulated weight (function findPath() in line 1). Second, for all u edges of this path, a traceback is performed (line 2–5) and one k-bit message symbol is returned per edge (function messagesymbol() in line 3). Finally, the decoded message m is returned.

MPD's calculation is integrated into the second step of Viterbi decoding. Here, the traceback iterates over the complete weight-minimizing path and MPD can be calculated per code symbol (line 4). To express the distance between a received code symbol to a symbol from the edge of the trellis, with hard-decision decoding simply the Hamming distance can be used. Here, MPD can be directly calculated over c. With soft-decision decoding, $\tilde{c}$ is required as an offset-free copy of codeword c including *normalized* soft-decision variables (aka soft-bits) $\in [-1,1]$. For these normalized soft-bits, the euclidean distance can be calculated as $\mathrm{diff}(a,b) = \sum_{j=0}^{n}(a_j - b_j)^2$ where a_j stands for one of n soft-bits in symbol a from the received codeword and b_j corresponds to one of n soft-bits in code symbol b from the trellis edge (as returned by function codesymbol() in line 4). After this distance is obtained for all symbols of the weight-minimizing path, Algorithm 1 returns vector mpd as the MPD for all symbols of codeword $\tilde{c}$.

For applying MPD in practical systems, the following observations are important:

- Integrating MPD into standard hard and soft-decision Viterbi decoding does not significantly increases calculation complexity. As only simple operations are added and no additional trellis iterations are required, an MPD-extended Viterbi algorithm still has complexity $\mathcal{O}(u)$.
- An MPD value can be obtained for each decoded symbol. Although this enables *symbol-wise* adaptation (e.g. one SDF decision per symbol), in practice "smoothed" MPD values averaged over blocks of N symbols would be used.

B. MPD accuracy study

We now study the accuracy of MPD's BER approximation for transmitting over channel (s,r) in a vehicular WLAN scenario (cf. Sec. II). We compare one average MPD value per message to the ideal case where SNR is *continuously* measured over each message symbol. In this unrealistic case, called *ideal* $\gamma_{s,r}$, *each* message symbol represents a training symbol and, thus, no data is transmitted. Further, we compare MPD to a realistic metric, where $\gamma_{s,r}$ is extracted only from the first 4 preamble symbols of a message. This corresponds to SNR measurements in IEEE 802.11a/g WLAN transceivers [8] and is called *realistic* $\gamma_{s,r}$.

We study the accuracy of these metrics by comparing obtained metric values to the number of errors *remaining* in a decoded message, i.e. equivalent to the actual BER of this message. An ideal metric would, thus, provide this BER by expressing the number of remaining bit errors as a function of metric values. We illustrate this expression as a scatter plot in Fig. 2. Per metric, the figure shows a single point for each of the 4200 transmitted 500 Byte messages. The resulting structure shows how accurately the studied metric approximates the number of errors. Realistic $\gamma_{s,r}$ shows a high standard deviation and can, thus, not accurately express a distinct number of bit errors (Fig. 2). No similarity compared to ideal $\gamma_{s,r}$ and no clear structure is shown. In consequence, realistic $\gamma_{s,r}$ is not a useful indicator of message quality. This situation changes completely for the MPD. Even with only a single MPD value per message, the points in the scatter plot fall in a very small region, i.e. low standard deviation, very similar to ideal SNR. Moreover, MPD and number of bit errors can be (almost) injectively mapped onto each other by virtue of monotonic increase.

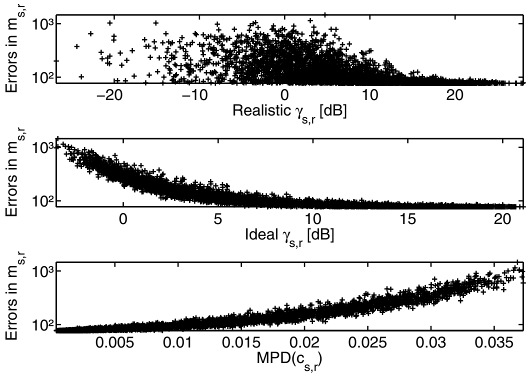

Fig. 2. Scatter plots for the metrics *realistic* $\gamma_{s,r}$, *ideal* $\gamma_{s,r}$, and *MPD* vs. the number of bit errors in message $m_{s,r}$. Each metric is observed during the same transmission via channel (s,r) at $v = 20$ m/s and mean SNR $\in [0,20]$ dB.

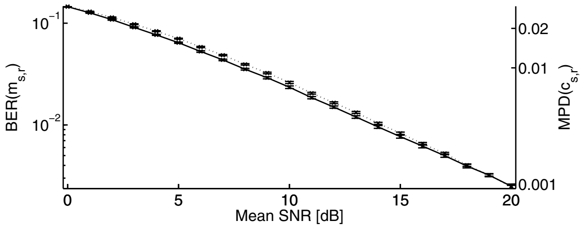

Fig. 3. Mean MPD of $c_{s,r}$ (dotted line) compared to mean BER of $m_{s,r}$ (solid line) for channel (s,r). Confidence intervals are shown for 95%.

This high accuracy of MPD results from the fact that each average MPD value consists of many samples continuously observed during the decoding process. In addition to this statistical benefit, MPD is calculated *during* FEC decoding and, thus, takes the actual decoding status into account. None of these benefits are achieved with realistic $\gamma_{s,r}$ as measured only over short training sequence at the beginning of a message. Due to these benefits, Fig. 3 shows a close qualitative match of BER and MPD averaged over all messages in the above scenario. This shows that MPD is an excellent estimator for the actual number of errors in a decoded message as well as for the BER of many messages. Consequently, MPD enables more accurate adaptation decisions than realistically measurable SNR.

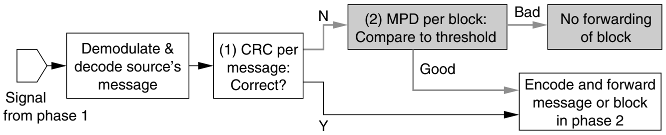

Fig. 4. Two-stage decision process of the 2SDF protocol performed at relay r between the two phases of each cooperation cycle. The shaded parts illustrate the proposed extensions to CRC-based SDF.

IV. DECODING-BASED SELECTION DF PROTOCOLS

We apply MPD in cooperation protocols to support partial forwarding in TSDF and propose Two-stage SDF (2SDF) combining the benefits of MPD and CRC-based decision.

A. *One-stage TSDF with partial forwarding*

A straightforward approach to profit from MPD's accurate BER approximation is to apply MPD instead of SNR in conventional TSDF protocols. Here, the relay makes only a single threshold-based forwarding decision per message based on the MPD averaged over all u message symbols.

Further, MPD can be averaged over small blocks of a message (Sec. III). For each of these blocks the relay may choose a block length $N \in [1,u]$, calculate the corresponding MPD value, and make an independent, MPD threshold-based forwarding decision. This allows *partial forwarding* of messages; beneficial in scenarios where fading does not affect an entire message or the relay may reconstruct parts of a completely faded message by FEC. According to its threshold, this cooperation protocol, called *MPD-based TSDF (block)*, drops erroneous blocks while still forwarding correct blocks.

Partial forwarding requires the relay to signal the dropped blocks to the destination. Therefor, we replace each dropped block by a short delimiter sequence. By selecting the block length N such that it is always larger than the delimiter, using delimiters does not increase the length of the message and, consequently, does not decrease the data rate.

B. *Two-stage SDF (2SDF)*

In practice, the threshold chosen for the above MPD-based TSDF protocol may be not optimal. This may lead to erroneous forwarding decisions, e.g. the relay drops even *correct* messages if the chosen threshold is too low. This erroneous dropping of correct messages is prevented by combining MPD's block-wise with CRC's message-wise decision in the so-called Two-stage SDF (2SDF) protocol.

2SDF can be employed in systems using CRC *and* FEC codes, e.g. IEEE 802.11a/g. With 2SDF the relay decides in two stages whether to forward the decoded message $m_{s,r}$ or not (Fig. 4). In the first stage the complete message is tested by CRC. If the message passes the CRC, it is considered to be correct and relayed completely. If the message fails the CRC, a standard (one-stage) SDF protocol would drop this message. This is not the case with 2SDF. Here, in a second stage, an MPD threshold-based decision is made for each message block. Hence, each block with an MPD sufficing the threshold is forwarded.

Combining CRC and MPD threshold-based decision in 2SDF provides the following benefits: In its first stage, 2SDF effectively prevents to erroneously drop *correct* messages since, here, CRC provides an accurate error detection for long messages and no threshold is involved. In its second stage, 2SDF employs MPD thresholds to enable partial forwarding of possibly erroneous messages (Sec. IV-A). By estimating the decoding quality block-wise, here, MPD enables a more fine-grained forwarding decision than CRC-based SDF. Instead of conservatively dropping the entire message if the CRC fails, with 2SDF only small erroneous blocks are dropped. Blocks considered to be correct are still forwarded, improving the end-to-end BER at the destination. For these reasons we propose to use block-wise MPD and 2SDF in systems with CRC and FEC coding. The performance of this combination, called *MPD-based 2SDF (block)*, is studied in the next section.

V. END-TO-END PERFORMANCE STUDY

We study the end-to-end BER (BER_{e2e}) performance of direct transmission and ideal and feasible one and two-stage cooperation protocols using several decision metrics. In particular, we consider the ideal cases *Genie-aided* SDF (hypothetical case where the relay knows and forwards only the correct bits) and *Ideal $\gamma_{s,r}$-based TSDF (symbol)* with unrealistic symbol-wise SNR decision to study an ideal metric with threshold-based decision. To these ideal cases standard message-wise CRC and SNR-based protocols are compared as implementable choices for WLAN systems, i.e. *Realistic $\gamma_{s,r}$-TSDF (msg.)* measured over the first 4 preamble symbols of each message (Sec. III-B) and *CRC-based SDF (msg.)* with a single CRC-checksum per message. These message-wise straightforward applications of SDF are compared to the feasible block-wise schemes proposed in Sec. IV. Here, one-stage *MPD-based TSDF (block)* and two-stage *MPD-based 2SDF (block)* are studied for 4 Byte blocks. We choose this block length to avoid a data rate decrease by signaling the dropped blocks to the destination (cf. Sec. IV-A). Further, we use a constant MPD threshold of $3.91 \cdot 10^{-4}$ while BER_{e2e}-optimal thresholds are chosen for SNR [6].

For these protocols, we show BER_{e2e} results *prior* to decoding to allow comparison to studies for uncoded cooperation systems [2, 5, 6]. We assume a cooperative WLAN with standard IEEE 802.11a parameters as in Sec. II. In this scenario, a constant stream of 500 Byte messages is CRC coded and, then, FEC coded at rate $R_c = 1/2$. Per message, this results in 8000 Bits transmitted at 6 MBit/s using BPSK modulation. All i.i.d. Rayleigh channels are parameterized with the same mean SNR and the simulations were performed at *symbol level*, i.e. a channel state change may affect at least one modulation symbol.

Two different fading scenarios are studied: First, all channels fade slowly, i.e. highly time-correlated fading, with Doppler shift $f_d = 17.34$ Hz. At the carrier frequency of 5.2 GHz this corresponds to a maximum velocity of $v = 1$ m/s, e.g. an indoor scenario. In the second scenario all channels vary faster according to $f_d = 350$ Hz, i.e. $v = 20$ m/s

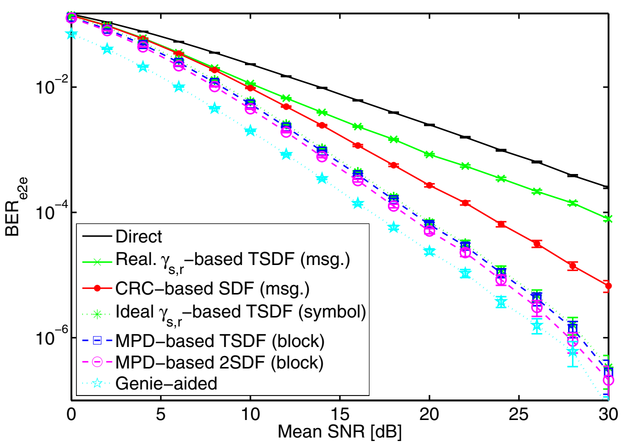

Fig. 5. End-to-end BER vs. mean SNR with direct transmission and different SDF metrics/protocols; i.i.d. Rayleigh fading channels, $v = 1\,\text{m/s}$, 10^5 transmitted messages per shown value, and 95% confidence intervals

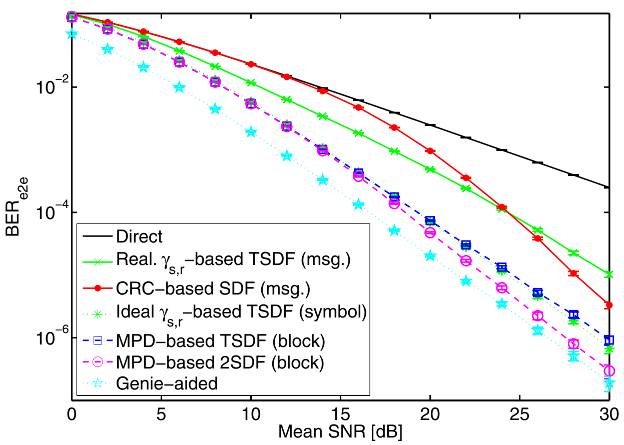

Fig. 6. End-to-end BER vs. mean SNR with direct transmission and different SDF metrics/protocols; i.i.d. Rayleigh fading channels, $v = 20\,\text{m/s}$, $6 \cdot 10^4$ transmitted messages per shown value, and 95% confidence intervals

corresponding to a vehicular scenario. Here the state of each channel decorrelates, i.e. a channel is "less stable" over time.

The BER_{e2e} results for the *indoor* channel scenario are shown in Fig. 5. Even with slowly fading channels, poor performance is reached if SDF adapts only once per message. A further effect results from the quality of the decision metric. As *Realistic* $\gamma_{s,r}$*-TSDF (msg.)* bases its decision only on a few symbols per message it achieves significantly lower performance than *CRC-based SDF (msg.)* with its ideal CRC-based decision. Comparing these results to the ideal cases or to feasible block-wise schemes clearly demonstrates that directly applying CRC or SNR-based SDF in WLANs is not the best design choice. Instead, partial forwarding with MPD significantly improves the performance. In this indoor scenario, simple *MPD-based TSDF (block)* performs as well as the ideal SNR case with symbol-wise decision. As only a suboptimal (constant) threshold is used, *MPD-based 2SDF (block)* can slightly improve this BER_{e2e} by avoiding erroneously dropping of correct messages.

Fig. 6 shows the BER_{e2e} results for the *vehicular* channel scenario. With this "faster" fading process the performance of *Realistic* $\gamma_{s,r}$*-TSDF (msg.)* has improved. Since here the channel state decorrelates over the complete message, an SNR sample measured over its preamble provides a more accurate estimate than with high time-correlation (cf. Fig. 5). For medium and low SNR, *CRC-based SDF (msg.)* suffers from CRC's conservativeness. In this scenario, where channel outages are shorter but more frequent, with CRC the *complete* message is dropped even if only a small part is affected. MPD, instead, forwards the correct parts which may decrease BER_{e2e} after combining at the destination. Therewith, *MPD-based TSDF (block)* reaches the performance of SDF with symbol-wise SNR, *MPD-based 2SDF (block)* improves this performance for high SNR. With high SNR, correct messages occur more frequently and are, thus, more likely to be dropped by a suboptimal MPD-threshold decision. As shown, this is effectively avoided by 2SDF's first-stage-CRC.

VI. CONCLUSION

Based on our BER performance study for indoor and vehicular WLAN scenarios we conclude that, in practice, the *message-wise* forwarding decision of CRC or SNR-based SDF cooperation protocols [2, 5] is not beneficial. Instead, *partial forwarding* significantly increases SDF's decision accuracy but relies on accurate and frequent channel estimation. In trellis-coded systems, this is provided by our *decoding-based* MPD metric for short term channel estimation which does not decrease data rate and adds only minor complexity. MPD may be interesting for many channel adaptation schemes. Employing it for partial forwarding in one or two-stage decoding-based SDF protocols closely reaches the performance of the ideal case even with practical channel and system assumptions.

As future work we suggest the optimization of MPD's threshold and partial forwarding's block length in order to minimize BER. Both requires an analytical framework which is the main target of our current study.

REFERENCES

[1] A. Sendonaris, E. Erkip, and B. Aazhang, "Increasing uplink capacity via user cooperation diversity," in *Proc. IEEE Int. Symp. on Inf. Theory (ISIT)*, Aug. 1998, p. 156.

[2] J. N. Laneman, G. W. Wornell, and D. N. C. Tse, "Cooperative diversity in wireless networks: Efficient protocols and outage behavior," *IEEE Trans. Inf. Theory*, vol. 50, no. 12, pp. 3062–3080, Dec. 2004.

[3] J. G. Proakis, *Digital Communications*, 4th ed. McGraw-Hill, 2000.

[4] M. K. Simon and M.-S. Alouini, *Digital Communications over Fading Channels*, 2nd ed. John Wiley & Sons, Inc., 2004.

[5] P. Herhold, E. Zimmermann, and G. Fettweis, "A simple cooperative extension to wireless relaying," in *Proc. Int. Zurich Sem. on Commun.*, 2004, pp. 36–39.

[6] F. A. Onat, A. Adinoyi, Y. Fan, H. Yanikomeroglu, J. S. Thompson, and I. D. Marsland, "Optimal threshold for SNR-based selective digital relaying in cooperative wireless networks," *IEEE Trans. Wireless Commun.*, submitted Apr. 2007, revised Sep. 2007, Nov. 2007.

[7] A. Willig, "Intermediate checksums for improving goodput over error-prone links," in *Proc. Vehicular Technology Conf. (VTC-Fall)*, Sept. 2004.

[8] B. O'Hara and A. Petrick, *IEEE 802.11 Handbook: A designers companion*. IEEE Press, 1999.

[9] E. C. Strinati, S. Simoens, and J. Boutros, "Error rate estimation based on soft output decoding and its application to turbo coding," in *Proc. IEEE Wireless Commun. and Netw. Conf. (WCNC)*, Mar. 2007.