Segment-Based Packet Combining: How to Schedule a Dense Relayer Cluster? *

Andreas Willig, Holger Karl, Danil Kipnis
University of Canterbury, University of Paderborn, Technical University of Berlin

March 11, 2011

Abstract. The classical three-terminal relaying scenario can be generalized to an entire cluster of relayers, which in this paper is assumed to be densely deployed. We consider the combination of cluster-based relaying with a packet-combining technique where the user data of a packet is partitioned into contiguous segments that can be individually checked for correctness. In such a setup the question arises how to schedule the transmissions of the relayers to either minimize the average transmission costs until at least one (and then all) relayers have the full set of segments, or to maximize the per-segment diversity, i.e. the number of distinct relayers sending the same segment. We investigate different options for scheduling the relayer cluster and show that under certain assumptions, average-optimal and easily implementable schedules exist. We furthermore provide numerical evidence that the adoption of a segment-based approach gives performance benefits over the "classical" method which only considers correctness of whole packets.

Keywords: Packet combining, Relayer cluster, Scheduling

1. Introduction

Transmission schemes based on cooperative diversity [13], [9] promise to increase transmission reliability in wireless networks by using cooperating single-antenna nodes. Examples of cooperative schemes include relaying [3] and cooperative MIMO schemes [4]. An often-used ingredient to cooperative diversity schemes is that the receiver of a packet *combines* different copies obtained over spatially different channels. In classic relaying [3], one of these copies is obtained from the direct link between transmitter and receiver, a second copy is provided by a relay node, who captures the transmitter's packet and forwards its observations to the receiver, who combines both its observations.

It is well known that with one relayer, the maximum achievable diversity gain is two [10]. It is therefore promising to increase the number of relayers to achieve a higher diversity gain. We are using a **relayer cluster**, *all* of whose members forward the packet (unlike in

* This work was partially supported by the Germany ministry of Education and Research in the AVS-ZESAN project, contract 01BN0712D, and the MIKOA project, contract V3AVS009

opportunistic forwarding, where only one, properly determined member would forward, see e.g. [22]). To improve chances that all relayers indeed have a correct copy of the packet, the relayers themselves can adopt the combining approach. Specifically, when a relayer A forwards its observations of the source packet towards the destination, another relayer B can in turn try to extract information from A's packet (which is broadcast) and combine it with its own observations of the source packet. Relayer B can then send more complete information towards the destination.

The usefulness of a relayer cluster essentially depends on two factors:

– How well is the cluster able to capture all user data sent by the source?

– Assuming that after the source transmission not all relayers have the full user data, how can the operation of the relayer cluster be organized so that the cluster efficiently achieves a high diversity gain (from which the final destination would benefit)?

In this paper, we address these questions for a particular kind of packet-combining schemes (or type-III hybrid ARQ schemes [12]), which have low memory requirements at receiving nodes and thus are especially interesting for implementation in wireless sensor networks.Their key property is that the full user data is partitioned into smaller contiguous segments, the correctness of each of which can be checked separately (e.g. using a per-segment CRC). We refer to this as **segment-based packet combining**, examples for such schemes are given in [19], [6], [21]. With this, it is not necessary that a single relayer captures all segments from the source for the cluster to possess all user data; it is sufficient when each segment is captured by at least one relayer. To achieve a high diversity gain for the full user data, each segment should then be re-transmitted by as many distinct relayers as possible. Letting relayers overhear transmissions of other relayers is the key tool to achieve this.We assume throughout this paper that the relayer cluster is dense enough so that any relayer can successfully receive any other relayers transmission with very high probability.

The main contribution of this paper concerns the scheduling of the relayer cluster. Each relayer possesses some subset of all segments after the source has transmitted, we call this subset the relayers **source segments**. For segment-based combining schemes, we deviate somewhat from the usual definition of the diversity gain. Instead, we measure the diversity gain on the segment level and we wish to maximize the achievable per-segment diversity gain (more precisely: the number of *different* relayers by which one individual segment is transmitted) by choosing proper schedules for relayer transmissions. We consider two different classes of relayer scheduling problems, defined next.

In the class of **two-stage cooperation** schemes, the relay cluster operates in two different stages. In the **combining stage** the relayers exchange their segments among each other until one or more relayers (the "winning relayers") have all segments. One of the winning relayers then starts the **forwarding stage** by transmitting all segments towards the final destination. By the assumption of a dense cluster, all other relayers will pick up from the winning relayers transmission any segments that they are missing. As a result, their transmissions to the final destination then also include all segments. Our main focus is on the combining stage. Assuming that each relayer transmission in this stage has a cost $r(k)$ (e.g. representing the energy costs to transmit a packet with k segments), we seek a schedule for relayer transmissions which minimizes the average total costs. We formulate this problem as a stochastic sequencing problem and show that for a particular class of cost models (where the cost of sending a packet is an affine function of the number of segments included), the average-optimal transmission sequence has a particularly appealing form: the more segments a relayer has directly captured from the source, the earlier it transmits. We refer to this as a **monotonically decreasing** sequence.[1] Monotonically decreasing sequences lend themselves to easy implementation as it suffices to let the relayers use a MAC protocol which supports prioritization to ensure that a relayer with the largest number of segments captured from the source starts first.

In the class of **single-stage cooperation** schemes there is essentially only a single forwarding stage. The relayers try to obtain segments from other relayers' previous transmissions, which are directly intended for the destination. The main goal here is to find a schedule for the relayers that maximizes the per-segment diversity when each relayer is allowed to transmit only once. We show that also in this class, a monotonically decreasing schedule gives close to optimal results.

The paper is structured as follows. In the next Section 2, we give more details on the operation of the relayer cluster. In Section 3 we model and analyze a particular class of two-stage cooperation schemes. Following this, we analyze the single-stage case in Section 4. In Section 5 we compare segmented and unsegmented transmission with respect to their ability to let the relayer cluster capture the full user data. After looking at related work in Section 6, we end the paper in Section 7 with a discussion of our results. This paper includes results from a previous conference publication [20].

---

[1] We use the terms sequence and schedule interchangeably.

2. Problem Description and Considered Schemes

We consider a wireless system with one source node and a cluster with R relayers. The relayer cluster is densely deployed so that the channels between the relayers have negligible error rates.[2] All nodes transmit on a shared channel and their transmissions are separated in time.

The source node accepts a number of s user data bits from higher layers and partitions them into n segments of c bits each. A separate checksum is computed for each segment and appended to it. These extended segments are then concatenated and the source forms a packet by adding a header. The source broadcasts the packet once. A relayer, upon receiving the packet, first checks the header. If it is wrong, the relayer discards the packet. Otherwise, the relayer checks each segment in turn and buffers the correct segments. The next steps depend on the particular relayer cooperation scheme.

2.1. Two-stage Cooperation

We illustrate two-stage cooperation with an example. Suppose there are three relayers and the source has sent six segments, numbered from 1 to 6. After the source transmission, relayer A has segments $\{1, 2\}$, relayer B has segments $\{1, 3\}$, and relayer C has segments $\{1, 4, 5, 6\}$, we refer to these segment sets as the **source segments**. Suppose A starts the combining stage. After A's transmission, relayer B has segments $\{1, 2, 3\}$, and relayer C has segments $\{1, 2, 4, 5, 6\}$. In the next step, B transmits its captured source segments $\{1, 3\}$. During the combining stage B would not repeat segment 2 that it has received from A, since by the dense cluster assumption all other relayers have received segment 2 as well. However, B repeats segment 1, which it has directly received from the source – this simplifies the problem formulation, see below. As a result relayer A possesses segments $\{1, 2, 3\}$ and relayer C possesses $\{1, 2, 3, 4, 5, 6\}$ and thus can be considered the (single) winner of the combining stage. In the forwarding stage, relayer C would start transmitting all segments. By overhearing, nodes A and B receive from

[2] This assumption is clearly idealized, but practically error-free channels can be achieved with realistic setups. For example, the IEEE 802.15.4-compliant CC2420 transceivers operating in the 2.4 GHz band are widely used. With a transmit power of 0 dBm, a maximum distance of one or two meters between any two relayers, and in the absence of external interference practically error-free channels can be achieved. It is a subject of future research to determine the impact of non-vanishing error rates and only partial reachability among relayers on the choice of relayer transmission schedules. However, we believe that the insights gained here under those idealized assumptions provide a good starting point for these investigations.

C the segments $\{4, 5, 6\}$ that they both miss. Subsequent transmissions of A and B also include the full set of segments.

We consider two different cases for two-stage cooperation: in the **arbitrary collector** case the combining stage stops when *any* relayer has collected all segments or when all but one relayers have transmitted their source segments without success (i.e. no relayer having all segments). In the **designated collector** case, there is one prenegotiated member of the relayer cluster which collects segments from the other relayers and the combining stage stops as soon as the designated collector has all segments or all other relayers have sent their source segments. The designated collector case is the natural choice when the final destination of a source packet is actually part of the relayer cluster. Another reason to consider this case is that no additional agreement among relayers is needed when several of them have collected all segments and these have to agree which one starts the forwarding stage.

We first formulate the considered problem for the designated collector case. Suppose that relayer $i \in \{1, \dots, R\}$ has n_i source segments, i.e. segments it directly captured from the source. In the designated collector case, we assume that relayer R is the collector. A transmission schedule or a transmission sequence for the combining stage is specified by a permutation $(i_1, \dots, i_{R-1})$ of the numbers $(1, \dots, R-1)$; the designated collector itself does not transmit during the combining stage but performs the first transmission in the forwarding stage. Within the combining stage, given a schedule $(i_1, \dots, i_{R-1})$, first relayer i_1 sends its n_{i_1} source segments to the designated collector. If afterwards the designated collector possesses all segments of the data packet, it declares a **success** and the combining stage ends.[3] For example, the designated collector can send an appropriate acknowledgment packet. Otherwise, the next relayer i_2 will transmit its n_{i_2} source segments to the designated collector, and so on.[4] This is continued until either the designated collector declares success or all $R-1$ other relayers have transmitted their segments. We assume that we are given a cost function $r(\cdot)$ so that a cost of $r(k)$ is incurred when a relayer transmits a packet containing its k source segments during the combining stage.

[3] Clearly, when the designated collector already has all segments immediately after the source transmission no collection process is needed at all. We do not consider this trival case anymore.

[4] To keep our model tractable (specifically: to keep an additive expression for the total costs), we make the assumption that relayer 2 includes *all* its source segments into its packet, even those that have already been sent by relayer 1. In some settings such an assumption can be justified, for example when the relayers follow a fixed, TDMA-like schedule and relayers want to switch off their transceivers during the other relayers' slots.

When $J \in \{1, \ldots, R-1\}$ relayer transmissions are needed until either success is reached or all $R-1$ relayers have transmitted, the total combining costs are given by

$$\sum_{\nu=1}^{J} r(n_{i_\nu}) \tag{1}$$

We assume that the construction of a schedule might use knowledge of the numbers $n_1, \ldots, n_{R-1}$, but that no information is available about the specific sets of source segments that each relayer has. When the source's user data consists of n segments in total and relayer i has $n_i \leq n$ of these segments received from the source, we assume that each n_i-element subset of the set of all segments is equally likely to have been received by relayer i. The variable J above is thus a random variable, more precisely it is a stopping time. Under these assumptions we formulate the optimization problem as minimizing

$$V_{(i_1, i_2, \ldots, i_{R-1})}(n_1, \ldots, n_{R-1}) = E\left[\sum_{\nu=1}^{J} r(n_{i_\nu})\right] \tag{2}$$

over all possible permutations $(i_1, \ldots, i_{R-1})$ of $\{1, \ldots, R-1\}$, where the numbers $n_1, \ldots, n_{R-1}$ are given, and the expectation is taken over all possible allocations of n_1-element subsets of all segments to the first relayer, n_2-element subsets to the second relayer and so forth. The stopping time J depends on this precise allocation.[5]

The problem setting for the arbitrary collector case is similar: there is no pre-designated collector but the combining stage ends successfully when *any* of the relayers has collected all segments.

In this paper we focus on the designated collector case. The arbitrary collector case appears to be much more complicated and is a subject of future research. To keep the discussion simple, we do not consider coding schemes (although segment-based schemes can for example be coupled with erasure coding, see [19]).

[5] Readers familiar with finite-horizon sequencing or Markov-decision problems will notice that Equation 2 does not consider a terminal cost term, i.e. a cost that is incurred **after** the last transmission and which depends on the state in which the system ends up. In our particular setting the only sensible end state would reflect whether the designated collector has all segments (success) or not (failure). However, due to the assumption of error-free channels between relayers, it is true that if a particular schedule leads to a success at the collector, then any other schedule does so, too. Therefore the terminal costs cannot be influenced by the schedule and have thus not been included in our model.

2.2. Single-stage Cooperation

With single-stage cooperation, the goal is to maximize the average per-segment diversity under the assumption that each relayer has only a single opportunity to transmit a packet. However, relayers that transmit later are allowed to pick up segments that they do not yet have from transmissions of previous relayers and to transmit these as well. Each relayer sends a segment only once. We are again given only the numbers $n_1, \ldots, n_R$ of source segments that the individual relayers have and seek the transmission schedule maximizing the average per-segment diversity.

3. Two-stage Cooperation: The Designated Collector Case

In this section we consider the designated collector case. More specifically, given $n_1, n_2, \ldots, n_R$ as the numbers of source segments that relayers $1, \ldots, R$ have (with relayer R being the designated collector), we are interested in finding the permutation $(i_1, \ldots, i_{R-1})$ of $(1, \ldots, R-1)$ that minimizes the average costs until the end of the combining stage (compare Equation 2). The combining stage ends when either the designated collector possesses all data segments, i.e. **success** occurs, or when all $R-1$ relayers have transmitted their source segments, compare Equation 2. To avoid trivialities, we assume $n_R < n$ (where n is the total number of segments making up the user data) and $n_i \geq 1$ for $i = 1, \ldots, R-1$. We furthermore assume that when relayer i has k source segments, then each k-element subset of all n segments is equally likely. We first provide an analytical model and remark that for general cost models, it is necessary to consider all $(R-1)!$ possible permutations to identify the optimal one. However, for affine cost models where the transmission costs are of the form $r(k) = a + b \cdot k$ for k being the number of segments transmitted and $a \geq 0$, $b \geq 0$, we show that the optimal permutation is monotonically decreasing, i.e. the relayers having the most source segments transmit first.

3.1. Analysis for General Cost Models

Consider two generic relayers called A and B. Suppose that relayer A has $n_A \leq n$ distinct and randomly chosen segments and relayer B has $n_B \leq n$ distinct and randomly chosen segments. Let $X_0 = n_A$ denote the number of distinct segments at relayer A before relayer B transmits its observations over a perfect channel, and X_1 denotes the number of distinct segments in relayer A after the transmission. Then we define $T_{n_A,n_B,\delta}$ as the probability that relayer one has $n_A+\delta$ distinct segments after the other relayer's transmission of n_B segments. This can be expressed as:

$$T_{n_A,n_B,\delta} := \Pr\left[X_1 = n_A + \delta \middle| X_0 = n_A; n_B\right] = \frac{\binom{n_A}{n_B-\delta}\binom{n-n_A}{\delta}}{\binom{n}{n_B}} \qquad (3)$$

as can be seen from simple combinatorial arguments similar to those used in [19, Appendix B]. This equation is constrained to $n_A + \delta \leq n$. For $\delta > n_B$ we set $T_{n_A,n_B,\delta} = 0$.

Our model involves a state variable that tracks the transmissions seen by the designated collector. The state is represented as a probability distribution $p(\cdot)$ over $\{0, 1, \ldots, n\}$, where $p(m)$ represents the probability that the designated collector (relayer R) has m distinct segments. Our state space is given by the set of all feasible probability distributions. At the beginning of the combining process, the start state is chosen as

$$p_0(m) = \begin{cases} 1 & : \quad m = n_R \\ 0 & : \quad m \neq n_R \end{cases}$$

Suppose now that we are given a probability distribution $p(\cdot)$ from the state space and we choose action i, i.e. node $i < R$ transmits all its n_i randomly chosen segments. Being in state $p(\cdot)$, the posterior probability $\beta(p, i)$ that node R has all segments after node i has transmitted its n_i segments is given by:

$$\beta(p, i) = \sum_{\nu=n-n_i}^{n} p(\nu) \cdot T_{\nu,n_i,n-\nu}$$

If, after node i's transmission, the collector does not have all segments, the resulting state $U_i(p)$ becomes the posterior probability distribution $U_i(p)(\cdot)$, in which the posterior probability that the collector has m segments is given by:

$$U_i(p)(m) = \begin{cases} 0 & : \quad m = n \\ \sum_{\nu=0}^{m} \frac{p(\nu) \cdot T_{\nu,n_i,m-\nu}}{1-\beta(p,i)} & : \quad 0 \leq m < n \end{cases} \qquad (4)$$

This equation reflects the fact that for node R the probability of having m distinct segments afterwards is the sum over all ν to have ν segments before and receive $m-\nu$ new segments from relayer i, conditioned on the failure of node R to acquire all n segments after i's transmission. Please note that for $n_i > m$ Equation 4 becomes zero, which is consistent with our assumption of error-free channels between the relayers, as otherwise the collector would end up with at least $n_i > m$ segments and the event to have just m segments cannot occur. We extend this notation and write $U_{i,j}(p)$ for $U_i(U_j(p))$, we write $U_{i,j,k}(p)$ for $U_i(U_j(U_k(p)))$ and so forth. Furthermore, when the permutation $(i_1, \ldots, i_k, i_{k+1}, \ldots)$ is understood, we write $U'_k(p)$ to denote $U_{k,k-1,\ldots,1}(p)$.

For any given schedule $\pi = (i_1, i_2, \ldots, i_{R-1})$ and initial state $p(\cdot)$, the average total cost (Eqn. 2) can be expressed as:

$$\begin{aligned} V_\pi(p) = & \\ & r_{i_1} \\ + & \gamma(p, i_1) \cdot r_{i_2} \\ + & \gamma(p, i_1) \cdot \gamma(U_{i_1}(p), i_2) \cdot r_{i_3} \\ + & \gamma(p, i_1) \cdot \gamma(U_{i_1}(p), i_2) \cdot \gamma(U_{i_2,i_1}(p), i_3) \cdot r_{i_4} \\ + & \ldots \\ + & \gamma(p, i_1) \cdot \ldots \cdot \gamma(U_{i_{R-3},\ldots,i_2,i_1}(p), i_{R-2}) \cdot r_{i_{R-1}} \end{aligned} \tag{5}$$

where the abbreviation $\gamma(p, i) = 1 - \beta(p, i)$ has been used, representing the probability that i's transmission in collector state $p(\cdot)$ does not lead to a success at the collector. Furthermore, we have used the expression r_{i_k} to denote the cost value $r(n_{i_k})$. Intuitively, this expression represents the fact that the costs r_{i_1} are incurred unconditionally, since the first relayer transmits anyway. If its transmission fails (with probability $\gamma(p, i_1)$) then the cost r_{i_2} for the second relayer's i_2 transmission are incurred and so on.

Now consider two schedules $\pi_1 = (i_1, \ldots, i_{k-1}, i, j, i_{k+2}, \ldots, i_{R-1})$ and $\pi_2 = (i_1, \ldots, i_{k-1}, j, i, i_{k+2}, \ldots, i_{R-1})$ that differ only in places k and $k+1$. We want to compare the costs of both schedules. The average cost $V_{\pi_1}(p)$ of schedule π_1 when the start state is $p(\cdot)$ can be re-arranged as follows:

$$\begin{aligned} V_{\pi_1}(p) = & \\ & V_{(i_1,\ldots,i_{k-1})}(p) \\ + & \gamma(p, i_1) \cdot \ldots \cdot \gamma(U'_{k-2}(p), i_{k-1}) \\ & \cdot \big\{ r_i + \gamma(U'_{k-1}(p), i) \cdot \big(r_j + \gamma(U_i(U'_{k-1}(p)), j) \\ & \cdot \big[r_{k+2} + \gamma(U_j(U_i(U'_{k-1}(p))), i_{k+2}) \cdot (\ldots) \big] \big) \big\} \end{aligned}$$

Similarly, for schedule π_2 we get

$$\begin{aligned} V_{\pi_2}(p) = & \\ & V_{(i_1,\ldots,i_{k-1})}(p) \\ + & \gamma(p, i_1) \cdot \ldots \cdot \gamma(U'_{k-2}(p), i_{k-1}) \\ & \cdot \big\{ r_j + \gamma(U'_{k-1}(p), j) \cdot \big(r_i + \gamma(U_j(U'_{k-1}(p)), i) \\ & \cdot \big[r_{k+2} + \gamma(U_i(U_j(U'_{k-1}(p))), i_{k+2}) \cdot (\ldots) \big] \big) \big\} \end{aligned}$$

It is shown in Appendix B that $U_i(U_j(p)) = U_j(U_i(p))$ holds for any $p(\cdot)$, which implies that the expressions in square brackets $[\ldots]$ are identical for both schedules. Furthermore, all expressions *outside* the curly braces $\{\ldots\}$ depend only on the initial part $(i_1, \ldots, i_{k-1})$ of the two schedules and are therefore identical. It is furthermore shown in Appendix B that

$$(1 - \beta(p,i))(1 - \beta((U_i(p), j))) = (1 - \beta(p,j))(1 - \beta(U_j(p), i))$$

holds for arbitrary $p(\cdot)$. These facts together imply that $V_{\pi_1}(p) \leq V_{\pi_2}(p)$ if and only if

$$\frac{\beta(U'_{k-1}(p), i)}{r_i} \geq \frac{\beta(U'_{k-1}(p), j)}{r_j} \tag{6}$$

holds.

Unfortunately, this expression, while having some similarity to expressions occuring in stochastic scheduling and sequencing analyses (compare [2, Sec. 4.5]), depends not only on i and j, but through U'_{k-1} it depends on all previous actions $(i_1, \ldots, i_{k-1})$. Therefore, for general cost models, all $(R-1)!$ permutations of $(1, \ldots, R-1)$ have to be evaluated for their average combining costs (using Equation 5) in order to find the optimal permutation. For general cost models, the optimal permutation is not even uniquely determined. We hence restrict the class of cost models in the following.

3.2. The case of Affine Cost Models

In the following we show that for affine cost models $r(n) = a + b \cdot n$ with $a \geq 0$, $b \geq 0$ and $(a,b) \neq (0,0)$ the optimal schedule can be computed much more easily and in fact monotonically decreases in n_i. In such a model, a represents a fixed per-packet cost (e.g. for the fixed-length packet header and trailer) and b represents the per-segment cost. To start with, we expand Equation 6 for the two schedules $\pi_1 = (i_1, \ldots, i_{k-1}, i, j, i_{k+2}, \ldots, i_{R-1})$ and $\pi_2 = (i_1, \ldots, i_{k-1}, j, i, i_{k+2}, \ldots, i_{R-1})$ as above, and write $p(\cdot)$ instead of $U'_{k-1}(p)$. We find that $V_{\pi_1}(p) \leq V_{\pi_2}(p)$ if and only if:

$$\frac{\sum_{\nu=0}^{n} p(\nu) \cdot T_{\nu,n_i,n-\nu}}{r_i} \geq \frac{\sum_{\nu=0}^{n} p(\nu) \cdot T_{\nu,n_j,n-\nu}}{r_j} \tag{7}$$

holds. This relation is certainly fulfilled when we can show that for the given values n_i and n_j the relation

$$\frac{T_{\nu,n_i,n-\nu}}{r_i} \geq \frac{T_{\nu,n_j,n-\nu}}{r_j}$$

holds for all $\nu \in \{1, \ldots, n-1\}$ (the converse needs not be true). After simplification this is equivalent to showing that

$$\frac{\binom{n_i}{n-\nu}}{r_i} \geq \frac{\binom{n_j}{n-\nu}}{r_j} \tag{8}$$

holds for all ν. We note an important intermediate result: if the cost model $r(\cdot)$ is such that for all choices of n_i and n_j (and resulting costs $r_i = r(n_i), r_j = r(n_j)$) Inequality 8 is fulfilled for all $\nu \in \{1, \ldots, n-1\}$, then the optimal schedule becomes easy to compute: we start with the node i_1 that maximizes

$$\frac{\binom{n_i}{n-\nu}}{r_i} \tag{9}$$

over $i \in \{1, \ldots, R-1\}$, in the next step we select the node i_2 that maximizes 9 over $i \in \{1, \ldots, R-1\} \setminus \{i_1\}$ and so on.

We now specialize this to an affine cost model. Assume that $r(x) = a + b \cdot x$, where x is the number of segments a relayer transmits. We consider the case $n_i = n_j + \alpha$ for $\alpha \in \mathbb{N}, \alpha \geq 1$ and assume furthermore that $n_j \geq n - \nu$, since otherwise Inequality 8 is trivially satisfied (if $n_j < n - \nu$ then clearly the $n - v$ missing segments cannot be recovered from a transmission of just n_j segments, and both sides become zero). Under these additional assumptions, Inequality 8 is true if and only if:

$$\frac{\binom{n_j+\alpha}{n-\nu}}{\binom{n_j}{n-\nu}} \geq \frac{a + b \cdot n_j + b \cdot \alpha}{a + b \cdot n_j} = 1 + \frac{b \cdot \alpha}{a + b \cdot n_j} \tag{10}$$

If we can establish this inequality for $\alpha = 1$, then we can inductively carry it over to $\alpha > 1$ since it is well known that for $x \geq y$ the expression $\binom{x+\alpha}{y}$ (and thus the left-hand side of 10) grows superlinearly in α, whereas the right-hand side of 10 grows only linearly in α. Setting $\alpha = 1$ in Inequality 10 and simplifying, we need to check the inequality:

$$\frac{n_j + 1}{n_j + 1 - (n - \nu)} \geq 1 + \frac{b}{a + b \cdot n_j}$$

Since for given n_j the choice for ν minimizing the left-hand side is $\nu = n - 1$ (remember that we have restricted n_j and ν to not exceed $n - 1$) the left-hand side can be simplified to become:

$$\frac{n_j + 1}{n_j} = 1 + \frac{1}{n_j} \geq 1 + \frac{b}{a + b \cdot n_j}$$

However, this inequality is true for any n_j when $a \geq 0$, $b \geq 0$ and $(a, b) \neq (0, 0)$ holds. We thus have established that for the affine cost model and arbitrary $p(\cdot)$ we have $V_{\pi_1}(p) \leq V_{\pi_2}(p)$ whenever $n_i \geq$

n_j holds. By invoking the procedure established in Equation 9 we consequently get a monotonically decreasing schedule. It should be noted that we have successfully validated this analysis by extensive simulations, but do not include results to conserve space.

There are two different situations in which the relayer identified by the above rule (Equation 9) is not uniquely determined:

- When there exist $i \neq j$ with $n_i = n_j$ and $\beta(p, i) > 0$, then it does not matter whether i or j is chosen for transmission. Such a tie can be broken arbitrarily.
- There are scenarios in which $\beta(p, i) = 0$ for all i. For example with $n = R = 5$ and $n_1 = \ldots = n_5 = 1$, for the selection of the first relayer the rule given in Equation 9 yields zero for all i since $n_i = 1 < n - 1 = 4$. For affine cost models we can nonetheless apply the monotonically decreasing scheduling rule. Extensive simulations have confirmed that the generated schedules are still minimal.

3.3. Numerical Results

In this section we present some results for two different cost models. For an affine cost model, we compare the average cost achieved by the optimal (decrementing) schedule against the average costs incurred by the incrementing schedule to demonstrate that indeed the choice of schedule can make a significant difference for the total costs in the combining state. For example, with $n_1 = 3$, $n_2 = 8$, $n_3 = 4$, $n_4 = 6$ and $n_5 = 7$ (with the last node being the designated collector), the increasing sequence would be $3, 4, 6, 8$ and the decreasing sequence would be $8, 6, 4, 3$. It is argued in Section 7 why monotone sequences are particularly appealing for practical implementations.

We consider the cost model $r(x) = 1 + x$ which accounts for fixed packet overhead (headers etc.) as one cost unit plus one cost unit for each transmitted segment. This model is appropriate when the packet overhead has about the same size as a segment. We state some facts for this cost model:

- As has been shown in Section 3.1 and is confirmed by simulation results, the decreasing schedule always has minimal costs.
- The maximum ratio of the average costs obtained with an incrementing schedule to the decrementing (i.e. optimal) schedule, computed for each possible allocation from $\{1, \ldots, n-1\}^R$ has been computed for $n = 16$ and different values of R. The results are shown in Table I and indicate that the average and maximum

Table I. Statistics of cost ratios of incrementing schedules over decrementing (and optimal!) schedules for the cost model $r(x) = 1 + x$ and different numbers of relayers R, computed over all possible allocations from $\{1, \ldots, n-1\}^R$ for $n = 16$.

	Avg. cost ratio	**Max. cost ratio**
$R = 3$	1.039	1.345
$R = 4$	1.107	1.517
$R = 5$	1.186	1.629

Table II. Statistics of cost ratios of incrementing schedules over truly optimal schedules for the cost model $r(x) = (1 + x)^2$ and different numbers of relayers R, computed over all possible allocations from $\{1, \ldots, n-1\}^R$ for $n = 16$.

	Avg. cost ratio	**Max. cost ratio**	**Min. cost ratio**
$R = 3$	1.018	1.242	1
$R = 4$	1.044	1.293	1
$R = 5$	1.067	1.305	1

performance gain of the optimal schedule over the incrementing schedule increases with R. They also indicate that the performance gains can be substantial, confirming that indeed the result of scheduling can have significant cost impact.

Table III. Statistics of cost ratios of decrementing schedules over optimal schedules for the cost model $r(x) = (1 + x)^2$ and different numbers of relayers R, computed over all possible allocations from $\{1, \ldots, n-1\}^R$ for $n = 16$.

	Avg. cost ratio	**Max. cost ratio**	**Min. cost ratio**
$R = 3$	1.006	1.25	1
$R = 4$	1.021	1.462	1
$R = 5$	1.045	1.650	1

To illustrate that for other choices of cost models, a monotonically decreasing (or monotonically increasing) sequence needs no longer be optimal in terms of average costs, we consider the (somewhat artificial) cost model $r(x) = (1 + x)^2$. Some facts about this cost model are:

- Neither the monotonically decreasing nor the monotonically increasing schedule are necessarily optimal. As one example, consider $R = 5$ and the allocation 1,5,15,14,1 (i.e. $n_R = 1$). The decreasing schedule 15,14,5,1 has average costs of 471.478 (compare Equation 5), the increasing schedule 1,5,14,15 has average costs of 484.541. The truly optimal schedule (found after enumerating all schedules and evaluating each of it) is 15,1,5,14 and has average costs of 427.346.

- The statistics of the ratios of the average costs for the incrementing schedule to the truly optimal schedule are shown in Table II; similar results for the decrementing schedule can be found in Table III. The results indicate that on average (using the assumption that each allocation for given R is equiprobable) the monotone schedules have similar costs as the optimal schedules, with the decrementing schedule being slightly closer. In some instances the monotone schedules are optimal, in other instances their average costs are much higher than the optimal average costs.

Our results suggest that, though not necessarily optimal in all cases, the monotone schedules do (on average) a fairly good job in keeping the average combining costs low. This would need to be confirmed for a broader range of cost models, however.

3.4. Implementation of Monotonic Schedules

The results of this section (and also the findings reported in the next Section 4) indicate that monotonically decreasing schedules often very good, if not optimal, choices.

In a practical system, we could implement the monotonically decreasing schedule by means of medium access control protocols supporting priorities. Here, packets are associated with priorities and these protocols aim to achieve that packets of higher priority are transmitted before packets of lower priority. One example is the IEEE 802.11 EDCA [7] which supports stochastic prioritization, another example is the WiDom scheme [16] which supports deterministic priority enforcement in a dense cluster of nodes that has no hidden terminals, provided the priority values used by the nodes are unique. The WiDom protocol utilizes a priority field for arbitration, the value of this field controls the amount of backoff time that a contender listens for preceding transmissions before starting itself. This field has a configurable length. When we have n segments and R relayers, we could allocate $b = b_1 + b_2$ bits for the priority field, such that the most important $b_1 = \lceil \log_2 n \rceil$ bits encode the number of source segments a relay has, and the remaining $b_2 = \lceil \log_2 R \rceil$ bits encode a unique identification of a relayer within the cluster. Under this rule of assignment, the priorities are unique and the relayers having the largest number of source segments have the highest priority. It is a topic of possible future research to assess the effects of stochastic prioritization similar to EDCA on the achievable cost or per-segment diversity.

4. Single-stage Cooperation

In this section we consider single-stage cooperation, where each relayer only transmits once. Given numbers $n_1, \ldots, n_R$ so that n_i indicates the number of source segments for relayer i, we wish to identify a relayer schedule that maximizes the average per-segment diversity. Relayers that transmit later are allowed to collect segments that they do not yet have from transmissions of previous relayers (or the source) and to re-transmit them. The average per-segment diversity counts how often each segment is transmitted on average by the relayer cluster, under the assumption that a relayer transmits any segment at most once.

We first present the different scheduling schemes considered in this paper and then present a range of results.

4.1. Scheduling Schemes

In the first scheduling scheme, called **no combining**, the relayers do not cooperate at all, they completely ignore other relayers' observations and do not capture any missing segments from them. Instead, each relayer just transmits its source segments to the destination, i.e. those segments that it has received itself from the source. The order in which the relayers transmit is not important; a random order has been adopted.

In the second scheduling scheme, called **genie combining**, it is assumed that the relayers are interconnected through an additional and perfect side channel that allows them to exchange segments at no cost and without any transmission on the wireless medium. Immediately after the source transmission, the relayers use the side channel to exchange their segments, and afterwards each relayer has all segments that are present in the relayer cluster. Then, the relayers transmit all the segments they possess to the destination in random order. This scheme has been included to provide an upper bound for the average per-segment diversity.

The third scheduling scheme is called **decreasing schedule**. In this scheme the first relayer transmits its source segments. The second relayer transmits its source segments plus those new segments that it has picked up from the packet of the first relayer. The third relayer transmits its source segments plus those new segments that it has picked up from the packets of the first two relayers, and so on. The sequence in which relayers transmit is decided only once, immediately after the source has transmitted its packet: the relayer having the most source segments starts, followed by the relayer with the second-most source segments, followed by the relayer with the third-most source segments, and so on. Those relayers that initially do not have any source segment are appended in arbitrary order to the chosen sequence; due to the assumedly error-free channels between relayers they all transmit the same set of segments (the union of all segments transmitted by the other relayers).

The final scheduling scheme is called **dynamic schedule**. The relayer having the most source segments starts to transmit. Suppose it transmits the set $\mathcal{S}_1 \subset \mathcal{S} = \{1, 2, \ldots, n\}$. After the first relayer's transmission, we determine the set $\mathcal{S}_1' = \mathcal{S} \setminus \mathcal{S}_1$ of segments that have not yet been transmitted. The next relayer is chosen to have the maximum number of segments in $\mathcal{S}_1'$, this subset is denoted as $\mathcal{S}_1^* \subset \mathcal{S}_1'$. Next, the chosen relayer transmits the segment set $\mathcal{S}_2 = \mathcal{S}_1 \cup \mathcal{S}_1^*$ (since the second relayer repeats everything that the first one already has sent). Afterwards, we again determine the set of not yet transmitted segments as $\mathcal{S}_2' = \mathcal{S} \setminus \mathcal{S}_2$. The next relayer is chosen to have the maximum number of segments in $\mathcal{S}_2'$ and so on. Those relayers that cannot contribute any new segments transmit in some arbitrarily chosen order at the end. This scheme provides the optimum average per-segment diversity, since it always strives to transmit as many not-yet transmitted segments as soon as possible, in order to transmit them to as many relayers that have not yet transmitted as possible.

4.2. Evaluation Method

The different scheduling schemes have been evaluated by simulation. Since segment-based packet combining schemes are especially appealing for wireless sensor networks (due to their low memory requirements), we have chosen the major PHY parameters (rate, transmit power, etc.) to comply with the 2.4 GHz spread-spectrum PHY specified in the IEEE 802.15.4 standard. The relayers are arranged on a circle with a small radius of 0.5 m. With a transmit power of 0 dBm used by source and relayers, and for the channel models chosen for evaluation (see below), the channels between relayers exhibit very small packet loss rates and thus the relayer deployment complies to the dense-cluster assumption made throughout this paper. The source is located at a distance of D meters from the center of the relayer circle.

Between each pair of nodes there exists a channel that is stochastically independent of all other channels. We use two different channel models. The first model (called **no-fading** model) uses a distance-dependent path loss that follows a log-distance path loss model [17] with a path loss exponent of $\gamma = 2.5$, a reference distance of 1 m and a path loss at the reference distance of 50 dB [18], [14]. In the second channel model (called **block-Rayleigh** model), the log-distance model is augmented with multiplicative, flat, block Rayleigh-fading. That is, the received signal amplitude is not only attenuated by the path loss (same as for the no-fading model), but it is further attenuated by a random fading gain that follows a Rayleigh distribution with parameter $\sigma = (\pi/2)^{-1/2}$ (i.e. the average attenuation factor equals one) and a channel coherence time of 0.5 s. To simplify the statistical evaluation of the simulation results the instants where the block-fading process changes channel gains are aligned with the instants where the source generates new transmission requests, so that in effect each transmission request (and its accompanying relayer transmissions) is confronted with a new realization of the Rayleigh channel gains.

All nodes transmit on a shared channel and their transmissions are separated in time. To achieve this separation, the decreasing schedule scheme can use a MAC protocol supporting priorities (see Section 3.4), whereas the other three schemes are idealized schemes that have been included for comparison, so we assume that they agree on a schedule "magically". These assumptions (which are fairly common in sensor networks) allow one relayer to overhear transmissions of other relayers. The source periodically generates a user message of $s = 1024$ bits and formats a packet according to a segment-based scheme described in [19]. We use a fixed value of $c = 64$ bits for the segment size. The checksum fields have a width of $h = 16$ bits and packets have a header of $o = 100$ bits. Please note that our choices for c divide s evenly and there are no slack segments.

4.3. Results

In Figures 1 and 2 we show the average per-segment diversity for the different scheduling schemes for both the no-fading and the block-Rayleigh channel model. For each parameter value we have run simulations for

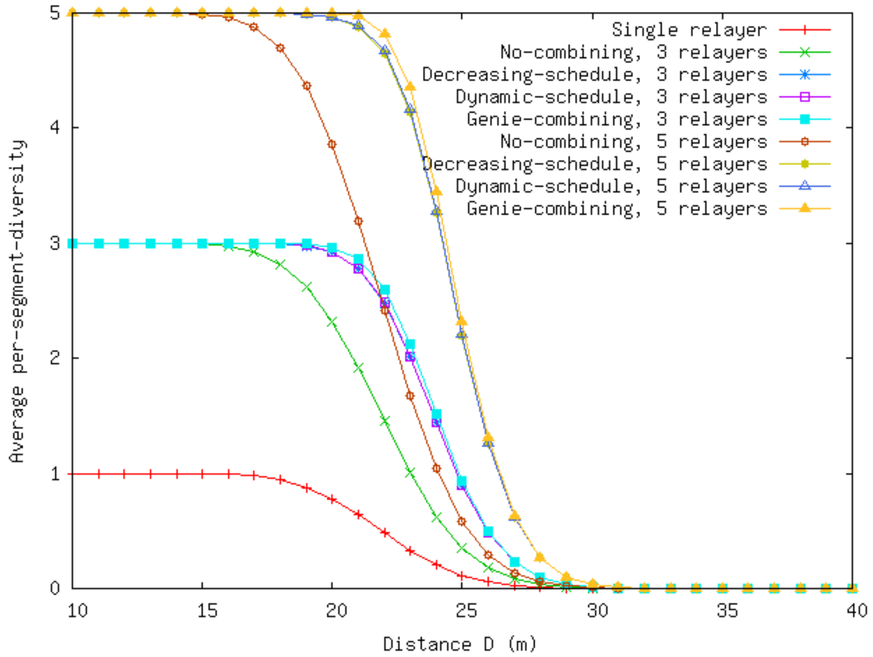

Figure 1. Average per-segment diversity of the different scheduling schemes, radius of relayer circle is 0.5 m, No-fading channel

400,000 simulated seconds, corresponding to 800,000 trials (the source generates new packets every 0.5 seconds). Since for both channel models each trial is independent, the maximum confidence interval halfwidth for the average per-segment diversity at a confidence level of 1 % is at most 0.0023. The confidence intervals are not shown in the figures.

For both channel models the performance difference between the no-combining scheme as compared to the other scheduling schemes is significant, thus confirming our expectation that combining in the cluster really pays off. The advantage of the other scheduling schemes over the no-combining scheme grows with the number of relayers, and this growth is more significant for the block-Rayleigh model.

For both channel models there is virtually no performance difference between the decreasing-schedule scheme (where a scheduling transmission is made only once at the beginning, using only the numbers n_i) and the dynamic-schedule scheme (decision after each relayer transmission, using knowledge of specific segment sets of each relayer). A likely explanation is that the dynamic-schedule scheme tends to choose the same actual sequence of relayers as the decreasing-schedule scheme would: both schemes select the same relayer as the first one, and since the relayer with the second-most source segments is likely also the relayer

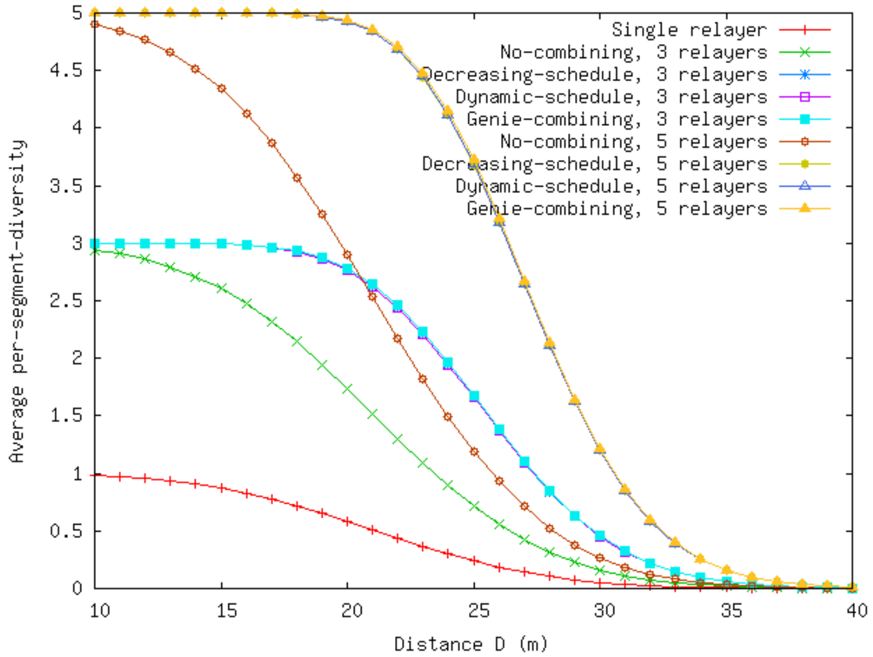

Figure 2. Average per-segment diversity of the different scheduling schemes, radius of relayer circle is 0.5 m, Block-Rayleigh channel

having the most source segments that the first relayer has not sent, the second relayer chosen by the dynamic-schedule scheme will often be the same as the one chosen by the decreasing-schedule scheme, and so on. This justifies to use the simpler decreasing schedule in practice.

For the block-Rayleigh model the decreasing- and dynamic-schedule schemes have almost exactly the same performance as the ideal genie-combining scheme, whereas for the no-fading channel model the genie-combining scheme has only slight advantages over the other two schemes. A possible explanation for the block-Rayleigh model rests on the fact that for the block-Rayleigh fading model the probability distributions for the number of source segments that the relayers have depends on the instantaneous realizations of the fading gains between the source and a relayer, and thus the per-relayer distributions likely differ. Those relayers with favorable channel conditions towards the source have likely more segments than those with poor conditions. The average fading gain is one and in the presence of several relayers it is therefore more likely that at least one relayer enjoys good channel conditions and has already all the segments that the cluster in total has. Choosing this relayer as the first one to transmit then gives (by the good channel qualities between relayers) the maximum achievable per-segment diversity.

5. Comparison of Segmented and Unsegmented Transmission

In the introduction we have raised two major questions about the usefulness of relayer clusters: the question how well the cluster is able to capture the full user data, and the question how to schedule the relayers in the cluster to achieve a high per-segment diversity gain. So far, we have concentrated on scheduling. In this section, we address the first question and specifically look at the probability that the cluster possesses the full user data; we call this the **possession probability**. More specifically, we will demonstrate analytically and by simulation that the segment-based scheme has a significantly better possession probability than the "classical" or unsegmented transmission. In the segmented case the cluster is said to possess the full user data when each of its n segments has been received by at least one relayer. In the unsegmented case the cluster possesses the data when at least one relayer has correctly received a packet with the full data from the source.[6]

We assess the possession probability for both channel models by simulation. In addition to this, for the no-fading model we also provide an analytical expression for the possession probability and use it to validate the simulation results. We first present the analytical expression.

5.1. Analytical Expression for Possession Probability

To keep the model tractable, we assume that (i) all relayers have the same distance to the source node, (ii) all channels between source and relayers are pairwise independent, and (iii) on each of these channels bits are erroneous with probability $e \in (0, 1)$. These assumptions are consistent with the no-fading channel model. The source broadcasts a packet with $n = \frac{s}{c}$ segments once. A relayer first checks the MAC and PHY header for errors (for simplicity we assume that one bit error is sufficient to destroy the header), and if the header is correct, each

[6] Please note that in our paper s is fixed, but the source packet in case of the segment-based schemes will be longer than for the unsegmented case, because of the additional checksums. In theory this provides an unfair advantage for the segment-based schemes, since they use more energy. However, our measurement results presented in [8] show that segment-based schemes also have significant advantages over unsegmented transmission when this energy difference is accounted for.

segment is checked individually. From this description, the probability that a relayer receives the header and k segments is given by:

$$r(k) = p_H \cdot \binom{n}{k} \cdot p_S^k \cdot (1 - p_S)^{n-k} \tag{11}$$

for $k \in \{0, 1, \ldots, n\}$, where $p_H = (1-e)^o$ is the probability that the header is correctly received, o is the header length in bits, and $p_S = (1-e)^{c+h}$ is the probability that a segment of size c bits plus the accompanying checksum of h bits is correctly received. With this, the probability distribution $\pi(k)$ that an individual relayer possesses k segments is then given by:

$$\pi(k) = \begin{cases} (1 - p_H) + r(0) & : \quad k = 0 \\ r(k) & : \quad k \in \{1, 2, \ldots, n\} \end{cases} \tag{12}$$

since the case $k = 0$ either occurs if the relayer has not received the header, or if it has received the header but failed to receive any segment. This distribution of possessing k segments is the same for all relayers and the relayers are independent. When a relayer possesses k segments, then each of the $\binom{n}{k}$ selections of k out of n segments is equiprobable.

Now we develop an expression for the possession probability of a cluster of R relayers. Fix one particular relayer, say, relayer R. The event that the cluster possesses the full information is equivalent to the event that the fixed relayer R possesses the full information after *all* other relayers $1, 2, \ldots, R-1$ have conveyed their segments over perfect channels to relayer R. These transmissions happen successively and each relayer (except relayer R) transmits exactly once. The order of transmissions does not matter. We can model the evolution of the number of segments in relayer R as a finite-state time-homogeneous Markov chain [15] as follows. Let $(X_i)_{i \geq 0}$ denote the (random) number of distinct segments that relayer R possesses after the i-th transmission of another relayer. All X_i have the common range $\{0, 1, \ldots, n\}$. The transition matrix $\mathbf{P}$ has the entries

$$p_{i,j} = \begin{cases} \sum_{k=j-i}^{n} \pi(k) \cdot T_{i,k,j-i} & : \quad n \geq j \geq i \\ 0 & : \quad \text{otherwise} \end{cases} \tag{13}$$

The initial state probability distribution for X_0 is just given by the $\pi(k)$ defined in Equation (12). The possession probability can then be expressed as:

$$\left[\lambda^{(0)} \cdot \mathbf{P}^{R-1}\right]_n = \left[\pi \cdot \mathbf{P}^{R-1}\right]_n \tag{14}$$

which is the n-th component of the vector obtained by multiplying the probability vector π with the matrix $\mathbf{P}^{R-1}$. The latter matrix can be regarded as the $R-1$-step transition matrix, corresponding to having all $R-1$ relayers transmitting their observations to relayer R.

5.2. Comparison of segmented and unsegmented transmission

We have obtained the possession probability by simulation and, for the no-fading model, also by numerical evaluation of the model represented by Equation 14. We have used the same simulation setup as in Section 4.2. The simulated time is 100,000 seconds, corresponding to 200,000 generated packets The maximum confidence interval halfwidth for the possession probability at a confidence level of 1 % is according to [1, p. 417] approximately 0.003. The confidence intervals are not shown. The bit error probability e in the analytical model has been derived from the distance and path loss parameters in the same way as for the simulation model.

We compare simulation results for the possession probability for different numbers of relayers and both channel models. In Figures 3 and 4 we show, for the no-fading and the block-Rayleigh channel models, respectively, the possession probability versus the distance D between the source and the relayer cluster. To avoid visual cluttering, we have restricted the figures to the case of two and five relayers. Furthermore, we have included numerical results (from evaluating Equation 14) into the figure for the no-fading model. The results show excellent agreement between simulation and numerical analysis and we thus have strengthened our trust that our simulation results are accurate.

For two or more relayers the segment-based scheme outperforms the unsegmented scheme in terms of the possession probability for both channel models. For the case of five relayers the difference is significant. For a small range of distances (22 m and more) and the no-fading channel model the unsegmented scheme with five relayers is even outperformed by the segment-based scheme with two relayers. The shape of the curves and the differences between the segment-based and the unsegmented scheme depend on the channel error model. For the no-fading channel the difference for the case of five relayers is larger than for the block-Rayleigh channel.

6. Related work

Segment-based packet-combining schemes (or, more generally, schemes where contiguous pieces of a packet are kept for combining purposes) have been considered for a while now, see for example [19], [6], [21],

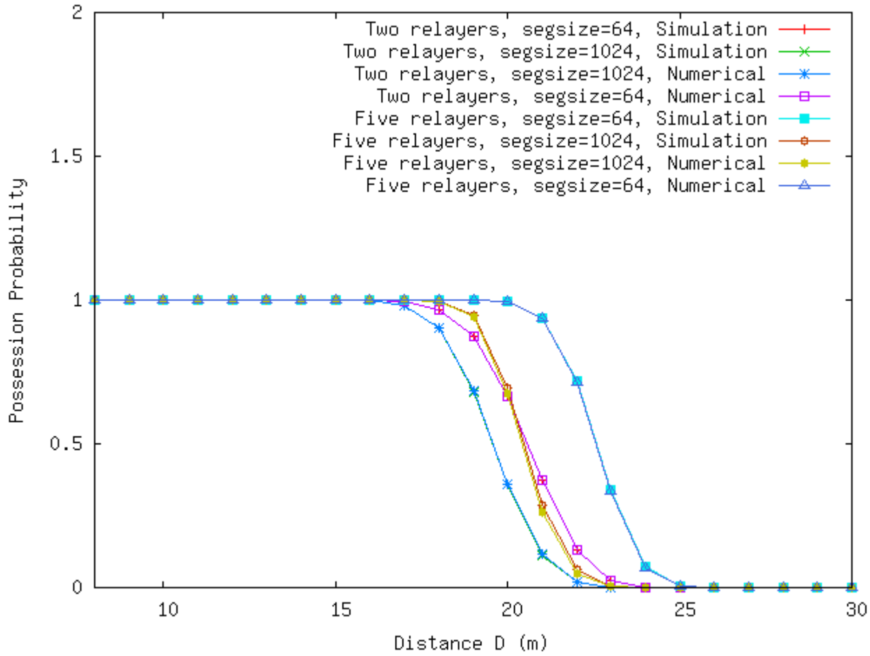

Figure 3. Possession probabilities (simulation) for one, two and five relayers, and the segmented and unsegmented schemes. The results for the no-fading model also include numerical results

[11]. In [6] a very similar approach has been designed, implemented and evaluated in the context of wireless sensor networks. They consider that the allowed frame size is in general smaller than the message size. The message is fragmented into small blocks, several of which can fit into a frame. The focus of the protocol is on efficiently streaming the blocks such that one frame can at the same time contain retransmissions of earlier failed blocks and new blocks from the same or the next message. They also suggest a cooperative version of the segment-based scheme (see also [5]), but have not followed this path. In [21] a segment-based scheme with feedback is considered, in which additionally the whole resulting packet is encoded using a convolutional code.

To the best of our knowledge, this paper is the first one that investigates the combination of segment-based packet-combining and the usage of a relayer cluster in more detail.

7. Conclusions

This paper presented two main results for dense relayer clusters. First, with two or more relayers the segment-based schemes provide a better

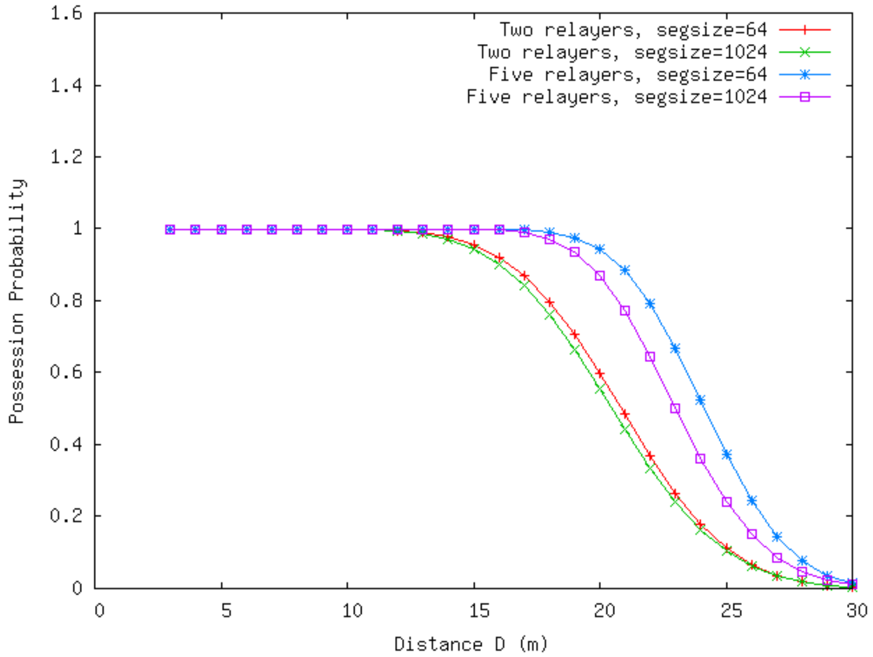

Figure 4. Possession probabilities (simulation) for one, two and five relayers, and the segmented and unsegmented schemes. Block-Rayleigh channel

probability for the cluster to possess the full information than the unsegmented scheme. Secondly, monotonically decreasing schedules are optimal for single-stage cooperation and two-stage cooperation with a designated collector and affine cost models. For another cost model they come on average close to the optimal performance. Furthermore, since in the single-stage cooperation case the results for the decreasing-schedule schemes are almost identical to the results of the (much more information-demanding) dynamic-schedule scheme, we can conclude that in the considered scenarios it is sufficient to determine the transmission schedule only once, immediately after the source transmission. No tracking of state is necessary in the relayer cluster.

There are further opportunities for future research. A very promising example is to consider the additional effect of coding schemes, it is for example demonstrated in [19] that by applying Luby-type erasure codes to segments, significant reliability gains can be achieved.

Appendix

A. Independence on Transmission Sequence for Two Transmissions

Suppose that we have three nodes with $n_1 < n$, $n_2 < n$ and $n_3 < n$ randomly chosen segments (out of a total of n segments), respectively, node i has n_i segments. The segment sets of all three nodes are independent of each other. The first node remains quiet. We look at two sequences of events: in the first sequence node two transmits its n_2 segments first and is followed by node three transmitting its n_3 segments. In the second sequence node three transmits first and node two follows. Let $X_0 = n_1$ be the number of different segments node one has in its cache before any transmission, and let $X_1 = n_1 + \delta$ with $n_1 + \delta \leq n$ be the number of different segments node one has after the other two nodes have transmitted their data. More precisely, let

$$T_{n_1,n_2,n_3,\delta} = \Pr\left[X_1 = n_1 + \delta \middle| X_0 = n_1; (n_2, n_3)\right] \quad (15)$$

be the conditional probability that node one receives δ additional segments when it has already n_1 segments and when first node two transmits, followed by node three. On the other hand, let

$$T_{n_1,n_3,n_2,\delta} = \Pr\left[X_1 = n_1 + \delta \middle| X_0 = n_1; (n_3, n_2)\right]$$

be the analogous quantity for the scenario in which first node three transmits, followed by node two. We assume that all involved channels are perfect. Our goal is to show that

$$T_{n_1,n_2,n_3,\delta} = T_{n_1,n_3,n_2,\delta} \quad (16)$$

holds. We first observe that from the law of total probability and assuming independence between the transmissions of n_2 and n_3 we can express $T_{n_1,n_2,n_3,\delta}$ as:

$$T_{n_1,n_2,n_3,\delta} = \sum_{\nu=\max\{0,\delta-n_3\}}^{\min\{\delta,n_2\}} T_{n_1,n_2,\nu} \cdot T_{n_1+\nu,n_3,\delta-\nu} \quad (17)$$

(where $T_{n_1,n_2,\nu}$ is defined according to Equation 3), where the summation is carried out over all possible increments after one step. We first consider the case where $\delta \leq \min\{n_2, n_3\}$ so that we get

$$\begin{aligned} & T_{n_1,n_2,n_3,\delta} \quad (18) \\ = & \sum_{\nu=0}^{\delta} T_{n_1,n_2,\nu} \cdot T_{n_1+\nu,n_3,\delta-\nu} \end{aligned}$$

$$= \sum_{\nu=0}^{\delta} \frac{\binom{n_1}{n_2-\nu}\binom{n-n_1}{\nu}}{\binom{n}{n_2}} \frac{\binom{n_1+\nu}{n_3-\delta+\nu}\binom{n-n_1-\nu}{\delta-\nu}}{\binom{n}{n_3}}$$

$$= \frac{1}{\binom{n}{n_2}\binom{n}{n_3}}$$

$$\left[\sum_{\nu=0}^{\delta} \binom{n_1}{n_2-\nu}\binom{n-n_1}{\nu}\binom{n_1+\nu}{n_3-\delta+\nu}\binom{n-n_1-\nu}{\delta-\nu}\right]$$

$$= \frac{\binom{n-n_1}{\delta}}{\binom{n}{n_2}\binom{n}{n_3}} \left[\sum_{\nu=0}^{\delta} \binom{n_1}{n_2-\nu}\binom{n_1+\nu}{n_3-(\delta-\nu)}\binom{\delta}{\nu}\right]$$

where in the last equation we have used the easily verifiable identity:

$$\binom{n-n_1}{\nu}\binom{n-n_1-\nu}{\delta-\nu} = \binom{n-n_1}{\delta}\binom{\delta}{\nu}$$

To show $T_{n_1,n_2,n_3,\delta} = T_{n_1,n_3,n_2,\delta}$ it therefore suffices to show that

$$0 = \sum_{\nu=0}^{\delta} \binom{\delta}{\nu}\binom{n_1}{n_2-\nu}\binom{n_1+\nu}{n_3-(\delta-\nu)} - \sum_{\nu=0}^{\delta} \binom{\delta}{\nu}\binom{n_1}{n_3-\nu}\binom{n_1+\nu}{n_2-(\delta-\nu)} \tag{19}$$

holds. To show this, we use the identity

$$\binom{n}{k-m} = \sum_{j=0}^{m} \binom{n+m-j}{k}\binom{m}{j}(-1)^j$$

which is easily seen by induction for $m \leq k$. If we use this identity in Equation 19 we need to consider the expression:

$$\left[\sum_{\nu=0}^{\delta} \binom{\delta}{\nu} \left(\sum_{j_1=0}^{\nu} \binom{n_1+\nu-j_1}{n_2}\binom{\nu}{j_1}(-1)^{j_1}\right) \cdot \left(\sum_{j_2=0}^{\delta-\nu} \binom{n_1+\delta-j_2}{n_3}\binom{\delta-\nu}{j_2}(-1)^{j_2}\right)\right] - \left[\sum_{\nu=0}^{\delta} \binom{\delta}{\nu} \left(\sum_{j_3=0}^{\nu} \binom{n_1+\nu-j_3}{n_3}\binom{\nu}{j_3}(-1)^{j_3}\right) \cdot \left(\sum_{j_4=0}^{\delta-\nu} \binom{n_1+\delta-j_4}{n_2}\binom{\delta-\nu}{j_4}(-1)^{j_4}\right)\right] \tag{20}$$

This expression can be rewritten as a sum of the form

$$\sum_{x,y=0}^{\delta} a_{x,y} \binom{n_1+x}{n_2}\binom{n_1+y}{n_3}$$

with to-be-determined coefficients $a_{x,y}$. We will show that $a_{x,y} = 0$ holds for all $x, y \in \{0, 1, \ldots, \delta\}$. From inspecting Equation 20 it is clear that for $x < y$ only the upper sum contributes to $a_{x,y}$, for $x > y$ only the lower sum and for $x = y$ both sums contribute to $a_{x,y}$. Because of symmetry, it suffices to consider the cases $x < y$ and $x = y$:

Case 1 : We assume $x = y$. In this case it can be seen that $j_1 = j_3 = 0$ and $j_2 = j_4 = \delta - x$ and we get

$$\begin{aligned} a_{x,x} &= \binom{n_1+x}{n_2}\binom{n_1+x}{n_3} \cdot \\ &\left[\binom{\delta}{x}\binom{x}{0}(-1)^0\binom{\delta-x}{\delta-x}(-1)^{\delta-x}\right. \\ &\left. - \left[\binom{\delta}{x}\binom{x}{0}(-1)^0\binom{\delta-x}{\delta-x}(-1)^{\delta-x}\right]\right] \\ &= 0 \end{aligned}$$

Case 2 : We assume $x < y$. In this case we get:

$$\begin{aligned} &a_{x,y} \\ &= \binom{n_1+x}{n_2}\binom{n_1+y}{n_3} \\ &\quad \left[\sum_{\nu=x}^{y}\binom{\delta}{\nu}\binom{\nu}{\nu-x}(-1)^{\nu-x}\binom{\delta-\nu}{\delta-y}(-1)^{\delta-y}\right] \\ &= \binom{n_1+x}{n_2}\binom{n_1+y}{n_3}(-1)^{\delta-x-y} \\ &\quad \left[\sum_{\nu=x}^{y}\binom{\delta}{\nu}\binom{\nu}{x}\binom{\delta-\nu}{\delta-y}(-1)^{\nu}\right] \\ &= \binom{n_1+x}{n_2}\binom{n_1+y}{n_3}(-1)^{\delta-x-y} \\ &\quad \frac{\delta!}{x!(\delta-y)!(y-x)!}\left[\sum_{\nu=x}^{y}\binom{y-x}{\nu-x}(-1)^{\nu}\right] \\ &= C \cdot \sum_{\nu=0}^{y-x}\binom{y-x}{\nu}(-1)^{\nu} = 0 \end{aligned}$$

We finally have to extend this proof to the cases where $n_2 < \delta$ or $n_3 < \delta$ holds (compare Equation 17). However, from inspecting Equation 18 it can be seen that it is not harmful to let ν run from 0 to δ, since $\binom{n}{k}$ becomes zero for $k < 0$.[7] We can therefore in all cases adopt Equation 18.

B. Designated Collector Case: $U_i \circ U_j = U_j \circ U_i$

To establish $U_i(U_j(p)) = U_j(U_i(p))$ for the designated collector case, we consider for given initial state $p(\cdot)$ the distribution of the number of segments in node R after node i's and node j's transmissions. After expanding (see Equation 4), we get:

$$\begin{aligned} U_i(U_j(p))(m) \;=\; & \frac{1}{(1-\beta(U_j(p),i))\cdot(1-\beta(p,j))} \\ & \cdot \sum_{\nu=0}^{m}\sum_{\mu=0}^{\nu} p(\mu)\cdot T_{\nu,n_i,m-\nu}\cdot T_{\mu,n_j,\nu-\mu} \end{aligned}$$

We consider the two factors of this expression separately. For the double sum term, one can establish after simple algebraic rearrangement the identity:

$$\begin{aligned} & \sum_{\nu=0}^{m}\sum_{\mu=0}^{\nu} p(\mu)\cdot T_{\nu,n_i,m-\nu}\cdot T_{\mu,n_j,\nu-\mu} \\ & = \sum_{\nu=0}^{m} p(\nu)\cdot T_{\nu,n_j,n_i,m-\nu} \\ & =: S(m,n_j,n_i) \end{aligned}$$

(see Equation 15 for the definition of $T_{\nu,n_j,n_i,m-\nu}$). An auxiliary result (Equation 16 in Appendix A) establishes that $S(m,n_j,n_i) = S(m,n_i,n_j)$ holds. For the other factor we consider $(1-\beta(U_j(p),i))\cdot(1-\beta(p,j))$. Expanding $\beta(U_j(p),i)$ we get:

$$\begin{aligned} & \beta(U_j(p),i) \\ & = \sum_{\nu=0}^{n-1} U_j(p)(\nu)\cdot T_{\nu,n_i,n-\nu} \\ & = \frac{1}{1-\beta(p,j)}\sum_{\nu=0}^{n-1}\sum_{\mu=0}^{\nu} p(\mu)\cdot T_{\nu,n_i,n-\nu}\cdot T_{\mu,n_j,\nu-\mu} \end{aligned}$$

[7] For $k < 0$ we have $\binom{n}{k} = \binom{n}{n-k} = \binom{n}{m}$ for some $m > n$, and the binomial coefficient $\binom{\alpha}{\beta}$ is defined to be zero for $\beta > \alpha$.

$$= \frac{1}{1-\beta(p,j)} \left(S(n,n_j,n_i) - \sum_{\mu=0}^{n} p(\mu)\cdot T_{n,n_i,0}\cdot T_{\mu,n_j,n-\mu} \right)$$
$$= \frac{1}{1-\beta(p,j)} \left(S(n,n_j,n_i) - \beta(p,j) \right)$$

where in the first equation the fact that the sum over ν runs only to $n-1$ is due to $U_j(p)(n) = 0$ after a failed transmission of j (compare Definition 4), and in the second-last line the fact $T_{n,n_i,0} = 1$ has been used. With this result we get:

$$\begin{aligned}&(1-\beta(U_j(p),i))\cdot(1-\beta(p,j))\\ &\quad = 1-\beta(p,j) - (S(n,n_j,n_i) - \beta(p,j))\\ &\quad = 1 - S(n,n_j,n_i)s\end{aligned}$$

which, by again referring to Equation 16 in Appendix A), is insensitive to exchanging i and j. This proves the claim.

References

1. Banks, J., J. S. Carson, B. L. Nelson, and D. M. Nicol: 2000, *Discrete-Event System Simulation.* Upper Saddle River, NJ: Prentice-Hall, third edition.
2. Bertsekas, D. P.: 2005, *Dynamic Programming and Optimal Control – Volume 1.* Belmont, Massachusetts: Athena Scientific, 3rd edition.
3. Cover, T. M. and A. A. E. Gamal: 1979, 'Capacity Theorems for the Relay Channel'. *IEEE Trans. Information Theory* **25**(5), 572–584.
4. Cui, S., A. J. Goldsmith, and A. Bahai: 2004, 'Energy-Efficiency of MIMO and Cooperative MIMO Techniques in Sensor Networks'. *IEEE Journal on Selected Areas in Communications* **22**(6), 1089–1098.
5. Dubois-Ferriere, H., D. Estrin, and M. Vetterli: 2005, 'Packet Combining in Sensor Networks'. In: *Proc. 3rd Intl. Conf. on Embedded Networked Sensor Systems (SenSys).* San Diego, CA.
6. Ganti, R. K., P. Jayachandran, H. Luo, and T. F. Abdelzaher: 2006, 'Datalink streaming in wireless sensor networks'. In: *Proceedings of the 4th international conference on Embedded networked sensor systems (ACM SenSys).* pp. 209–222.
7. IEEE Computer Society, sponsored by the LAN/MAN Standards Committee: 2007, 'IEEE Standard for Information technology – Telecommunications and Information Exchange between Systems – Local and Metropolitan Area Networks – Specific Requirements – Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications'.
8. Kipnis, D., T. Jacob, and A. Willig: 2010, 'Measured Performance of a Memory-Efficient Segment-Based Packet-Combining Scheme'. In: *Proc. IFIP Wireless Days 2010.* Venice, Italy.

9. Kramer, G., I. Maric, and R. D. Yates: 2006, 'Cooperative Communications'. *Foundations and Trends in Networking* **1**(3-4), 271–425.
10. Laneman, J. N., D. N. C. Tse, and G. W. Wornell: 2004, 'Cooperative Diversity in Wireless Networks: Efficient Protocols and Outage Behaviour'. *IEEE Trans. Information Theory* **50**(12), 3062–3080.
11. Lettieri, P. and M. Srivastava: 1998, 'Adaptive Frame Length Control for Improving Wireless Link Throughput, Range and Energy Efficiency'. In: *Proc. INFOCOM 1998*. San Francisco, CA, pp. 564–571.
12. Liu, H., H. Ma, M. E. Zarki, and S. Gupta: 1997, 'Error control schemes for networks: An overview'. *MONET – Mobile Networks and Applications* **2**(2), 167–182.
13. Liu, K. J. R., A. K. Sadek, W. Su, and A. Kwasinski: 2009, *Cooperative Communications and Networking*. Cambridge, UK: Cambridge University Press.
14. Molisch, A. F., K. Balakrishnan, C.-C. Chong, S. Emami, A. Fort, J. Karedal, J. Kunisch, H. Schantz, U. Schuster, and K. Siwiak: 2004, 'IEEE 802.15.4a channel model-final report'. Technical Report IEEE 802.15-04-0662-01-04a, IEEE P802.15 Working Group for Wireless Personal Area Networks (WPANs).
15. Norris, J. R.: 1997, *Markov Chains*. Cambridge, UK: Cambridge University Press.
16. Pereira, N., B. Andersson, and E. Tovar: 2007, 'WiDom: A Dominance Protocol for Wireless Medium Access'. *IEEE Trans. Industrial Informatics* **3**(2), 120–130.
17. Rappaport, T. S.: 2002, *Wireless Communications – Principles and Practice*. Upper Saddle River, NJ, USA: Prentice Hall.
18. Sohrabi, K., B. Manriquez, and G. J. Pottie: 1999, 'Near Ground Wideband Channel Measurement in 800-1000 MHz'. In: *Proc. IEEE Vehicular Technology Conference (VTC) '99*.
19. Willig, A.: 2009, 'Memory-Efficient Segment-Based Packet-Combining Schemes in Face of Deadlines'. *IEEE Trans. Industrial Informatics* **5**(3), 338–350.
20. Willig, A., D. Kipnis, and H. Karl: 2010, 'Segment-Based Packet Combining in a Cluster: To Combine or Not To Combine?'. In: *Proc. Sixth International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP)*. Brisbane, Australia.
21. Zhou, Y. and J. Wang: 2006, 'Optimum Subpacket Transmission for Hybrid ARQ Systems'. *IEEE Trans. Communications* **54**(5), 934–942.
22. Zorzi, M. and R. R. Rao: 2003, 'Geographic Random Forwarding (GeRaF) for ad hoc and sensor networks: multihop performance'. *IEEE Trans. Mobile Computing* **2**(4), 337–348.