ICELUS: Investigating Strategy Switching for Throughput Maximization to a Mobile Sink

Richard Figura*, Oliver Schmitz*, Tobias Hagemeier*, Matteo Ceriotti*,
Falk Brockmann*, Margarita Mulero-Pázmány**, Pedro José Marrón*
*University of Duisburg-Essen, Germany, **Doñana Biological Station, CSIC, Spain

***Abstract*—Wireless sensor networks offer a pragmatic solution for monitoring in a variety of scenarios. For efficient and practical data gathering, especially in large-scale systems deployed in inaccessible areas, unmanned vehicles are becoming a compelling solution. The added infrastructure flexibility comes at the cost of limited contact time between the mobile entity and the stationary devices. The channel fading caused by mobility further decreases the data yield. We address this challenge by analysing the relevant classes of data transfer schemes and identifying adaptation conditions that enable the selection of the best fitting strategy. The result of this analysis, ICELUS, provides an integrated protocol that exploits the available communication resources.**

I. INTRODUCTION

Sensor networks are being deployed in a number of scenarios, where environmental monitoring can increase the knowledge of physical processes. These factors will promote the density, scale and heterogeneity of networks of wireless sensors. In such settings, the use of mobile devices to gather data becomes a compelling solution, further promoted by the advancements in the field of unmanned ground and aerial vehicles (UGVs/UAVs) and the increasing public awareness.

In the EU-Project PLANET [1], we could experiment with such a vision. In this project, biologists were interested in monitoring the behaviour of wild horses in a natural reserve. Each horse was tagged with a collar, able to monitor the position and the context, e.g., standing, grazing, running, and providing the collected information over the radio. Given the scale of the reserve, the exclusive use of fixed data gathering infrastructures would have been cumbersome and expensive. In contrast, UAVs offered the flexibility to periodically visit the different groups of horses anywhere in the reserve.

Such scenario brings challenges and opportunities, rising from the mobility of the data gathering station and the foreseen network density. Horses group together and create a highly clustered, mostly stationary network. A mobile sink provides the flexibility required to efficiently gather data by opportunistically establishing communication links with identified data sources. Unfortunately, these links manifest unpredictable quality changes and limited bandwidth, caused by fading and short availability. Physical obstacles and their shadowing can also result in short-term hindering of communication quality. Considering the cost of each UAV flight and the corresponding environmental impact on the natural reserve, an efficient data collection scheme was of primary importance. The expected density of sensor nodes allows to set up alternative communication paths to increase reliability and throughput of the volatile direct connection to the mobile device.

We acknowledge the approaches optimising data delivery to mobile or stationary sinks for different settings (Section II). In this work, we start with the characterisation of our reference scenario and the resulting possible communication setups (Section III). We classify and model the most appropriate data transfer schemes (Section IV). This analysis reveals a new optimisation space, where switching among different classes of data delivery protocols can increase the achievable throughput depending on the current conditions. We present our adaptation protocol, ICELUS, which identifies a set of transmission strategies, able to exploit the available communication resources to counteract the unreliability of the link between source and sink (Section V). We instantiate the protocol design in an implementation that we evaluate in real-world testbed scenarios (Section VI). The results show that ICELUS is capable of increasing the throughput in favourable settings, without incurring overheads when the direct connection is sufficiently reliable. We then conclude the paper (Section VII).

II. RELATED WORK AND BACKGROUND

Efficient data transfer is crucial for practical data collection in low-power networks with mobile sinks [7]. We now discuss the different facets of the problem in light of the related work.
Discovery: Source and sink first need to discover each other, before starting the actual data exchange. Improving the efficiency of that discovery can already increase the time available for data transfer. In 2BD [19], beacons transmitted at different transmission powers enable the early detection of a mobile sink. In [3], the communication parameters are adapted at runtime based on the communication quality and the amount of data to be sent. These solutions rely on the availability of the direct link between source and sink and they must coexist with the impact of fading caused by the sink movement. The available communication resources as well as their quality and variation can be monitored through a variety of estimators [5].

ICELUS bases on the ability to estimate the link quality towards surrounding resources as a mean to trigger a context switch and apply a dedicated data transfer strategy. While we propose a simple and integrated solution with a low memory footprint, different discovery mechanisms can be integrated.
Data Transfer: After identifying the candidate communication resources, the data is transferred to the sink. We can distin-

ISBN 978-3-901882-80-7

guish three different classes of routing protocols that match our scenario.

One option is to establish a direct link between source and mobile sink to perform a *direct transfer*. Approaches like [12] investigate the opportunities offered by single mobile elements whose trajectory can be planned according to application requirements. The speed can be adjusted to match the number of data sources and the amount of data to be gathered. The main limitation is the unreliability of the used link, depending on the speed and mobility pattern of the mobile element [4].

A second possibility is offered in dense networks. E.g., WSC-MAC [15] transmits messages twice and selects forwarding nodes to synchronously amplify the second transmission. A similar approach exploits constructive interference [8], where the whole network performs synchronous retransmission to increase reliability. MRL-CC [14] considers multiple candidate relays and selects them based on link qualities and forwarding delay. In [9], using the information about the physical location of the nodes, the choice of the forwarders is computed to minimise the involved energy consumption. All these works perform relaying *on a single channel*.

The use of *multiple channels* [11], as third option, allows concurrent interference free transmissions in the same neighbourhood on different channels. Throughput can then be increased at the cost of coordinating the channel switches. While many approaches have been proposed for stationary or 802.11 networks [21], their application to the case of a mobile sink with low-power embedded systems has not been much investigated.

This work analyses these three classes of communication strategies as well as their potential benefits and trade-offs in our target scenario. ICELUS exploits this analysis to alternate between classes based on the observed system conditions. The provided reasoning also allows to apply the approach to alternative protocols and scenarios.

Adaptive Protocol Selection: One main contribution of this work is the study of the benefits rising from the combination of alternative routing schemes and the ability to select the most suitable one for the current conditions. In [23], a reconfiguration framework is introduced to enable the coexistence of applications with different requirements in a fixed network infrastructure. The work proposes a solution to dynamically choose the operational network protocols based on specified triggering conditions and application requirements.

Instead, we consider one application with a single goal: maximise the throughput to a mobile sink. We investigate a dynamic setting where the system properties change with continuity and a decision on when to perform a reconfiguration is challenging. Therefore, our work 1) analyses the communication schemes relevant for the target scenario and application, 2) identifies the adaptation conditions and 3) introduces an integrated protocol that exploits such analysis.

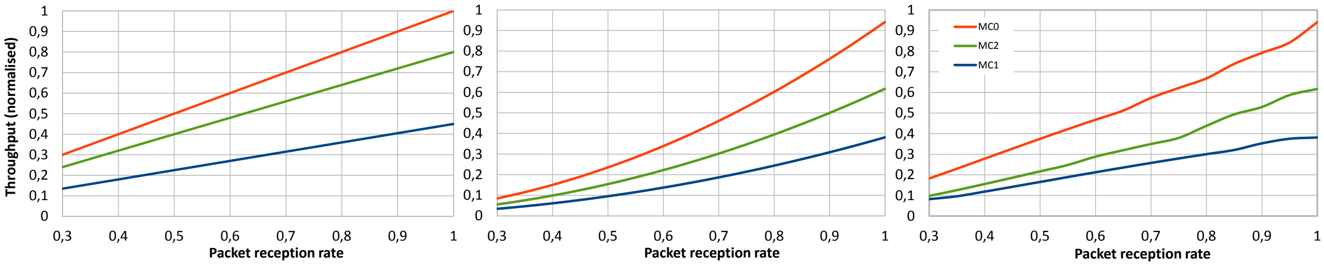
III. SCENARIO AND COMMUNICATION SETUPS

In PLANET [1], our goal was to monitor wild horses in a natural reserve. This scenario presents characteristics that we can exploit in the design of a fitting solution. In particular, feral horses group in bands, composed by up to 20 individuals, in average between 4 and 8 members [18]. They generally remain close to each other in within 15 meters; The exact distance vary according to predatory risk, intragroup harassment and resources distribution [13]. Considering that each horse carries a monitoring and communicating device, this results in a highly clustered network where nodes have short, reliable links with several neighbours. Moreover, wild horses are typically feeding or resting for around 80 % of the day [18], moving at a speed that is less than 5 mi/h [10]. Therefore, the network on the ground presents a relatively stable topology in comparison to the volatile links towards a mobile sink.

In our system, we consider nodes to take GPS samples every 20 minutes and to provide this data to a cluster head, a node responsible for providing all collected data to the mobile sink. All nodes check the presence of data collectors every second. By utilizing solar panels on our devices [16], we guarantee the required lifetime of one year, as also confirmed by similar systems with a comparable configuration [17].

In this scenario, our goal is to maximise the throughput between the cluster head and a mobile sink, carried by a UAV. While a UAV provides the required flexibility regarding coverage, its movement causes unpredictable quality changes, mostly due to medium-scale fading. In fact, small-scale fading, noise and interference Furthermore, we have to consider small communication times, in order of a few seconds, since the UAV is moving fast and it has to maintain a given distance to the group of animals, to minimise its impact on the horses behaviour. However, the horses tend to group together and high reliable links are available between neighbouring nodes. Thus, it is likely that other devices in the group offer the possibility for a relayed delivery able to overcome medium-scale fading or antennas orientation issues. We refer to the either direct or relayed communication as the “last mile” to the sink, given that it spans at most 2 hops. Since the nodes on the ground cluster in dense groups with short links, additional hops would neither help to counteract medium-scale fading problems nor contribute significantly to bring data closer to the sink.

For this scenario, we can identify three different possible communication setups. In the most desirable case, the source delivers data to the sink *directly* (we refer to this class of transfer strategies as MC_0). Once the link is detected, the source can fully use the available bandwidth to transfer the messages until the mobile sink leaves the communication range. In the case where only one additional node can serve as an intermediary between the source and the sink, *single-channel relaying* protocols (MC_1) can be employed. If more valid alternatives are available, multiple concurrent streams can be activated along different radio frequencies with *multi-channel relaying* (MC_2). We consider paths of maximum 2-hops length and, consequently, restrict the selection to two candidate relaying nodes and channels. Any further relay or channel would not increase the number of possible concurrent transmissions in a scenario with one data source.



(a) Best-Effort Strategies. (b) Single-Ack Reliable Strategies. (c) Adaptive-Ack Reliable Strategies.

Fig. 1. Theoretical throughput of alternative transfer strategies , assuming $p_{or} = 0.9$, $t_o = t_d$, symmetric links, and $m = 16$. The x axis reports the PRR of the last hop, e.g., p_{os}, p_{rs} or $p^*_{rs} = \min(p_{r_1s}, p_{r_2s})$. The throughput is normalised with respect to MC_0 and $p_{os} = 1$.

IV. Modelling Transfer Strategies

In this section, we model the classes of protocols suitable for the identified communication scenarios. We provide an analytic model of the best-effort versions of these classes, and then introduce reliability with a Selective Repeat ARQ technique with a single or an adaptive number of acknowledgements.

A. Model Parameters

We provide an analysis of the achievable throughput for each class of communication strategy previously identified, based on their analytic description. While the provided models are a generalisation of the specific protocols pertaining to each class, they serve as a reference to study their expected behaviour. Given

- $p_{xy} \in [0, 1]$ and $x, y \in \{(o)rigin, (r)elay, (s)ink\}$, Packet Reception Rate (PRR) from x to y
- v, application data per message (in bytes)
- t_d, time for sending a message/ack (in seconds)
- t_o, time for a channel switch (in seconds)
- m, number of messages in a burst

we describe the throughput for each given class of strategies, T_{Class}. The models ignore processing times and details of the strategies realisation, providing an upper bound to what achievable by a concrete implementation. Figure 1 depicts the resulting theoretical throughput achievable under simplifications applicable to our target scenario. In particular, we assume a p_{ro} equal to at least 0.9 given the short distance between devices and approximate t_o with t_d. Despite the generality of the models, for the sake of simplicity, we solve the equations considering symmetric links and the same link properties of the relays. We then report the normalised throughput taking as reference the maximum achievable throughput, i.e., with a perfect link between the origin and the sink with no losses and no acknowledgement scheme.

B. Best-Effort Strategies

Without a feedback scheme ensuring the delivery of messages, data flow uniquely from the source to the sink. The throughput of the direct connection to the sink can be modelled as

$$T_{MC_0} = (p_{os} \cdot v)/t_d, \tag{1}$$

where the delivered data along a link with PRR p_{os} is divided by the time needed to send the data. With one relay, the links unreliability and the additional time necessary for the transmission penalise the throughput

$$T_{MC_1} = (p_{or} \cdot p_{rs} \cdot v)/(2 \cdot t_d). \tag{2}$$

With two relays (r_1 and r_2), two bursts of data can be sent in parallel using different radio channels, which almost doubles the throughput compared to MC_1 (Figure 2, ignoring the exchange of acknowledgements). Furthermore, the model for MC_2 considers the additional time required to switch between channels for each burst of m messages. The achievable throughput is then

$$T_{MC_2} = \frac{(p_{or_1} \cdot p_{r_1s} + p_{or_2} \cdot p_{r_2s}) \cdot v \cdot m}{2 \cdot (m \cdot t_d + 2 \cdot t_o)}. \tag{3}$$

Figure 1(a) depicts the throughput of the different classes of strategies as described by the introduced models. The impact of the additional link used in class MC_1 results in less then half of the throughput achieved by MC_0. The difference is further amplified when unreliable links are used, as caused by the sum of effects on both the traversed links. The MC_2 class tries to counteract the drawbacks of a serial relaying by transmitting two bursts of data in parallel. However, relayed communication suffers from the impact of unreliable links, in addition to the cost of the channel switch time.

C. Single-Ack Reliable Strategies

In reality, reliability is often required to ensure that data have been correctly delivered to the sink. Therefore, a feedback scheme from the sink towards the source needs to acknowledge the successful reception.

To account for this additional feedback, we have to consider, when computing the throughput for the delivery of a burst of m messages, the time necessary to send such a burst as well as receive a single acknowledgement. The resulting time is $t_d \cdot (m + 1)$. Second, we need to address the probability of

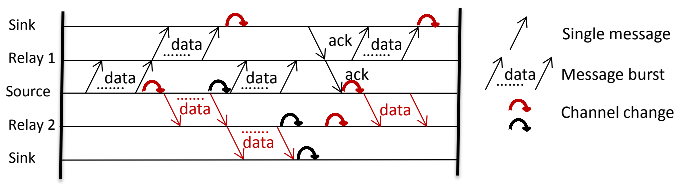


Fig. 2. Representation of the MC_2 strategy where two relays transfer concurrently multiple data streams on different radio channels.

losing the single acknowledgement, p_{so}, and consider the potential need for retransmitting the whole burst of messages. In case of a successful acknowledgement, just missing messages have to be repeated. Therefore, we introduce two additional multipliers, the first one provides the new time interval in its denominator and the second one considers the probability of losing the returning acknowledgement. The resulting model for the direct transfer to the sink becomes then

$$T^*_{MC_0} = T_{MC_0} \cdot \frac{t_d \cdot m}{t_d \cdot (m+1)} \cdot p_{so}. \tag{4}$$

Similar changes apply to the model of the reliable transfer through one relay. However, we need to account for the relaying of the acknowledgement, which makes it more likely to lose the acknowledgement over two links with PRR p_{sr} and p_{ro}. This results in

$$T^*_{MC_1} = T_{MC_1} \cdot \frac{2 \cdot t_d \cdot m}{2 \cdot t_d \cdot (m+1)} \cdot p_{sr} \cdot p_{ro}. \tag{5}$$

The same argument holds for the use of two relays that handle concurrent data streams by switching channels. In this case, in addition to the parallel transmission along different channels of m packets ($m/2$ for each channel), one additional message is required to acknowledge the delivery (sent over one of the chosen relays, as shown in Figure 2). The model becomes then

$$T^*_{MC_2} = T_{MC_2} \cdot \frac{2 \cdot (m \cdot t_d + 2 \cdot t_o)}{2 \cdot ((m+2) \cdot t_d + 2 \cdot t_o)} \cdot p_{sr} \cdot p_{ro}. \tag{6}$$

Figure 1(b) presents the comparison of the updated models accounting for reliability. The use of acknowledgements has a clear impact on the achieved throughput. However, this already indicates that in case of a poor direct connection to the sink alternative routes can become preferable, depending on the presence of relays with good link qualities. It is also possible to notice the decreased improvement provided by the use of the second relay over the single relay strategies, in comparison to the best-effort case. Nonetheless, it offers a substantial increase in the achievable throughput, preserving its effectiveness.

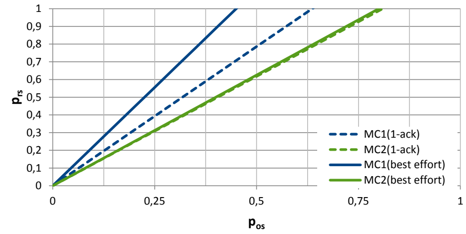
D. Adaptive-Ack Reliable Strategies

The previously modelled classes of transfer strategies can be optimised by adjusting the number of acknowledgements, according to the experienced link quality. In fact, each additionally sent acknowledgement can increase the throughput, avoiding retransmitting received bursts that failed to be acknowledged. However, additional acknowledgements also require valuable bandwidth, otherwise available for transmitting data. Therefore, the resulting models consider a longer transmission time for sending a number of additional acknowledgements and a decreased probability of repeating a complete burst of messages, when losing all the acknowledgements.

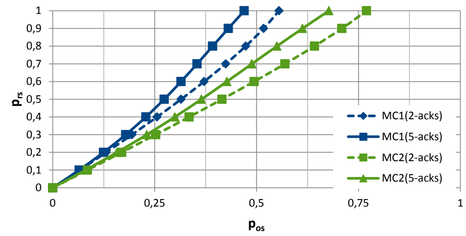
The resulting models are (a: number of acknowledgements)

$$T^{**}_{MC_0} = T_{MC_0} \cdot \frac{t_d \cdot m}{t_d \cdot (m+a)} \cdot (1 - (1 - p_{so})^a), \tag{7}$$

$$T^{**}_{MC_1} = T_{MC_1} \cdot \frac{2 \cdot t_d \cdot m}{2 \cdot t_d \cdot (m+a)} \cdot (1 - (1 - p_{sr})^a) \cdot (1 - (1 - p_{ro})^a), \tag{8}$$



(a) Best-Effort and Single-Ack Strategies.



(b) Multiple-Ack Strategies.

Fig. 3. Link conditions for which a relayed communication performs equally to the direct communication between origin and sink, assuming p_{or} of 0.9, m of 16, symmetric links, same characteristics at each relay and $t_d = t_o$.

$$T^{**}_{MC_2} = T_{MC_2} \cdot \frac{m \cdot t_d + 2 \cdot t_o}{(m + 2 \cdot a) \cdot t_d + 2 \cdot t_o} \cdot (1 - (1 - p_{sr})^a) \cdot (1 - (1 - p_{ro})^a). \tag{9}$$

Based on further empirical measurements, we identified an optimal number of acknowledgements. Figure 1(c) shows an increased throughput, compared to the single-ack scheme.

E. Switching Between Strategies

Based on the introduced models, we analyse the conditions under which a relayed connection can outperform the direct communication. To simplify the discussion, we apply the following approximations: 1) $t_o \approx t_d$, 2) $p^*_{rs} = \min(p_{r_1s}, p_{r_2s})$, 3) $p^+_{xy} = \sqrt{p_{xy} \cdot p_{yx}}$. Furthermore, in compliance with our assumptions, we set $p_{or} \geq 0.9$ and $p_{ro} \geq 0.9$, for each involved relay.

Best Effort Strategies: The following two equations show when it is beneficial to switch from a best-effort direct transmission to either MC_1 (Equation 10) or MC_2 (Equation 11). In either case, the relationship between p_{rs} and p_{os} is linear. As shown in Figure 3(a) (solid lines), a relayed best-effort strategy can be beneficial if the direct link PRR, p_{os}, is below 0.4 in the case of MC_1 or 0.8 in the case of MC_2.

$$T_{MC_1} \geq T_{MC_0} \Leftrightarrow p_{rs} \geq 2 \cdot p_{os}/p_{or} \Rightarrow p_{rs} \gtrsim 2,22 \cdot p_{os} \tag{10}$$

$$T_{MC_2} \geq T_{MC_0} \Rightarrow p^*_{rs} \geq \frac{(m+2) \cdot p_{os}}{m \cdot p^*_{or}} \Rightarrow p^*_{rs} \gtrsim 1,25 \cdot p_{os} \tag{11}$$

Single-Ack Reliable Strategies: Similarly, it is possible to estimate the conditions for which a strategy switch is beneficial in the case of a reliable communication with a single acknowledgement for MC_1 (Equation 12) and MC_2 (Equation 13). As depicted in Figure 3(a), the single relay strategy manifest advantages earlier in comparison to the best-effort strategies;

instead, the 2-relays strategy shows almost the same behaviour. As soon as the direct communication quality drops below 0.6, a switch to a single relay strategy can provide benefits.

$$T^*_{MC_0} = T^*_{MC_1} \Rightarrow p^+_{rs} \geq \sqrt{2}\frac{p^+_{os}}{p^+_{or}} \\ \Rightarrow p_{rs} \gtrsim 1.57 \cdot p^+_{os} \tag{12}$$

$$T^*_{MC_0} = T^*_{MC_2} \Rightarrow p^{2+*}_{rs} \geq \frac{p^{2+}_{os}}{p^{2+*}_{or}} \cdot \frac{(m+4)}{(m+1)} \\ \Rightarrow p^{+*}_{rs} \gtrsim 1.21 \cdot p^+_{os} \tag{13}$$

Multiple-Acks Reliable Strategies: In the case of multiple acknowledgements, the formula is more complex and depends on the number of acknowledgements employed, as expressed by Equations 14 and 15 for MC_1 and MC_2 respectively. Figure 3(b) shows representative solutions of the equations for $a \in \{2, 5\}$. Even if the actual threshold varies, a relayed communication can be beneficial when p_{os} drops below $0.5 - 0.7$.

$$T^{**}_{MC_0} = T^{**}_{MC_1} \Leftrightarrow p_{rs} \cdot (1-(1-p_{sr})^a) \\ \geq \frac{2 \cdot p_{os} \cdot (1-(1-p_{so})^a)}{(p_{or} \cdot (1-(1-p_{ro})^a)} \tag{14}$$

$$T^{**}_{MC_0} = T^{**}_{MC_2} \Rightarrow p^*_{rs} \cdot (1-(1-p_{sr})^a) \\ \geq \frac{p_{os}}{p^*_{or}} \cdot \frac{m+2 \cdot a+2}{m+a} \cdot \frac{(1-(1-p_{so})^a)}{(1-(1-p_{ro})^a)} \tag{15}$$

V. ICELUS: Composing Transfer Strategies

The provided analysis demonstrates that different classes of transfer strategies have unique characteristics and their benefits depend on the specific system conditions. To highlight this, we ran an experiment in a testbed with a mobile sink carried by a model train, a stationary data source and a stationary node available for relaying information (detailed description of this testbed is deferred to Section VI). This setup should simulate our target scenario in PLANET. Instances of the classes MC_0, MC_1 and MC_2 are referred to C_0, C_1 and C_2 respectively in the following. As shown in Figure 4, C_0 (for a direct transfer) can provide very high throughput but only if the link between source and sink is reliable. When such a link is not available anymore, the loss in throughput is significant. On the other side, C_1 (for a relayed transfer) has a lower throughput but a more stable performance across the experiment. This already suggests to switch between the different classes of transfer strategies, selecting the one achieving the highest throughput in the current conditions. $C_{0,1}$ (for a simple adaptive scheme), as introduced next in this section, can closely follow the performance of the best strategy, exploiting the actual system properties. This confirms the possible benefits of switching between the different transfer strategies.

In this section, we introduce ICELUS, an adaptive scheme that exploits the available network resources in the neighbourhood to increase the reliability of the “last mile” to the sink. The main idea is to identify, among the alternative classes of transfer strategy, the one best fitting the current system state. More concretely, we assume a single mobile *sink*, a single *source* and several *listening nodes* as potential *relays*.

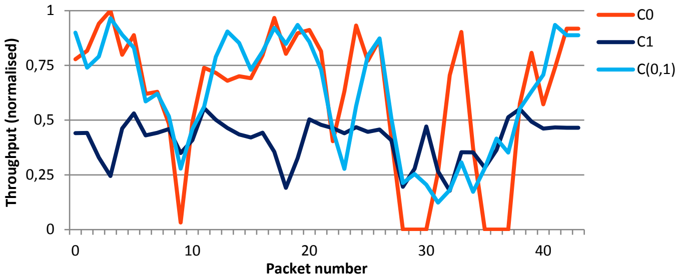


Fig. 4. Throughput achieved by two transfer strategies and an adaptive scheme able to switch between them.

ICELUS selects among different strategies to choose which listening nodes, if any, take the role of relays to maximise the throughput to the mobile sink.

A. ICELUS *in a Nutshell*

The main contribution of ICELUS is the composition of different transfer strategies and the conditions triggering the adaptation of the running strategy. In order to precisely understand, predict and control the behaviour of the solution once deployed, we decided to target an integrated protocol. In principle, we could have used representative protocol implementations for each class of transfer strategies. However, given that the experiments were performed in a sensitive area with limited access and thus limited debugging time, we favoured a single integrated solution in order to avoid debugging the possible detrimental interactions between existing protocol implementations not originally designed to coexist in the same system. As already identified in the literature [22], this forced coexistence of independent protocols can severely hinder the observed performance.

Like traditional protocols for data gathering with mobile sinks, ICELUS distinguishes between different phases: i) A *discovery phase*, in which the sink informs the surrounding nodes about its availability and preliminary link quality information is collected; ii) A *data transmission phase*, where a specific transmission strategy is used to deliver data to the sink; iii) A *candidature phase*, where alternative strategies are evaluated. ICELUS restricts its scope to the direct neighbourhood of the sink and the source and selects at most 2 nodes as relays, as justified in Section III. In this set of alternatives, depending on the observed communication quality, e.g., PRR or RSSI, the source selects one among three different possible transfer strategies: i) C_0, where the direct link between source and destination is used for data delivery; ii) C_1, in which one channel and one relay node is selected to forward data; iii) C_2, which extends C_1 by using two radio frequencies and selecting two nodes that collaboratively relay information, to allow two concurrent data streams. In addition to the selection taken by the source, the sink can also trigger a strategy switch, e.g., by observing an increase in the quality of the direct connection.

1) Discovery Phase: Before starting the data transfer, the sink continuously broadcasts discovery messages during its flight. These messages deactivate duty cycling techniques for the nodes that are relevant for the data transfer. In fact, ICELUS considers as relays only nodes that are neighbours of both sink and source. For this reason, the listening nodes that hear the discovery messages from the sink further broadcast the message themselves. The source can then hear the discovery messages coming from both the sink, if a direct link exists, and the nodes that could possibly act as relays, along with the information about the observed link qualities. By using such topology information, the source can decide on the transfer strategy to use and start the transmission phase. During transfer, the information about the network topology is refined and the choice of the best strategy is reconsidered.

2) Transmission Phase: Once the source has discovered all available communication resources, it decides how to transfer data to the mobile sink in the transmission phase. It will always prefer the C_0 strategy, in case of a reliable direct link to the sink or if no relay with a good link exists. Otherwise, it decides for a relayed connection as C_1 or C_2, utilizing the most promising relay in range. The sink permanently monitors the resulting throughput and can enforce a switch to another strategy through a candidature phase, as described later.

C_0 Strategy: Direct Transmission. In C_0, the source transfers a burst of m messages, acknowledged by the sink. If the throughput drops below a certain threshold, the sink enforces a candidature phase by raising a flag in the acknowledgements.

C_1 Strategy: Single-Channel Relaying. In C_1, the source sends a burst of m messages on a single channel to one relay. The relay buffers the complete burst and then takes its turn in sending the burst to the sink. In case the relay has received a burst just partially, it still sends m messages to the sink and uses the additional time for retransmitting the available information to the sink. The sink replies with a number of acknowledgement messages sent to the relay, which then forwards those to the source. The sink continuously monitors the connection from the source by overhearing the messages sent to the relays. If the direct communication reaches a given quality threshold, the sink enforces a strategy switch to C_0 by rising a flag in the acknowledgements.

C_2 Strategy: Multi-Channel Relaying. In dense networks, it is very likely that several nodes are available for relaying information to the mobile sink. In our target scenario, we consider a maximum of two relays as justified in Section III. If the source decides on two relays, it becomes possible to further increase the achievable throughput by handling two different data streams in parallel, each on a different radio channel. In practice, the procedure starts as in C_1, with a burst $m/2$ messages delivered to the first relay on the default radio channel. This burst piggybacks the id of the second relay and the chosen strategy, to inform the second relay about the strategy switch. As soon as the messages are delivered to the first relay, the source switches to the second channel and transmits a second burst of data of $m/2$ messages to the second relay. During this transfer, the first relay can safely transmit its burst to the sink on the default channel. The sink, at the end of this first burst, will then switch to the second radio channel to receive the burst of data from the second relay. As soon as also this delivery is completed, the sink can report acknowledgements for both data transfers through the relay with the best link quality. For further optimization, the source starts sending a third burst to the first relay, while the second relay is transferring the second burst to the sink. This third burst will already be available at the subsequent iteration of the process. In our implementation, the transmission of this third burst is explicitly triggered by a single message, sent by the source. The complete process is depicted in Figure 2. By using two relays concurrently, the C_2 strategy allows to almost double the throughput of the C_1 strategy. However, the sink can not monitor the direct connection to the source continuously. Therefore, the direct connection is explicitly measured periodically, in a regularly enforced candidature phase.

3) Candidature Phase: ICELUS allows switching between the different strategies during data transfer to account for possible (and likely) changes in the communication link qualities. Choosing the best transfer strategy requires awareness of the link qualities to the relays and the sink. In ICELUS, we avoid the continuous overhead of reporting link qualities. The sink, in fact, can periodically request a new candidature phase or trigger it only if the sink experiences a degradation in the observed throughput for the currently used strategy.

When the sink decides to evaluate alternative relaying paths, it raises a flag for a candidature phase in the acknowledgement sent towards the source. All listening nodes that overhear such message can measure the current link quality from the sink, whose information often correlates with the one of the link towards the sink [6]. In addition, the listening nodes have information about the quality from the source achieved by overhearing the data messages previously transmitted. If the measured qualities are above a given threshold, the node forwards this information to the source. The source waits then for a given time for the candidature information, before making a decision about the subsequent strategy. If no reliable link is available, C_0 is used as a default strategy.

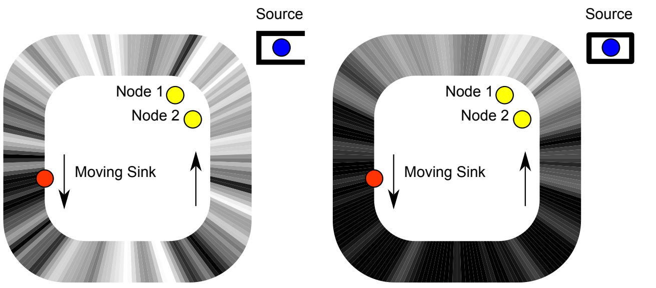
Finally, if no data is received in a given time, a candidature phase is triggered explicitly by the sink to recover possibly broken link.

VI. EVALUATION

After describing the protocol design, we evaluate its implementation in testbed and real-world scenarios.

A. Evaluation Setup

We realised a prototype implementation of ICELUS in TinyOS. The implementation considers a data message structure with an array of 4 x 16-bit integer values. When evaluating the protocol performance, we report average (∅) and standard deviation (σ) values out of at least 10 complete transmission processes of at least 2000 data bursts, with each burst composed of 16 messages (8 messages for each stream traversing one of the two relays in the C_2 strategy), sent at a frequency of 25ms. A data burst is acknowledged by the sink either through a static number of acknowledgements, fixed prior an experiment, or through an adaptive acknowledgement scheme. We first evaluate the performance of each single



(a) ABRUPT: Scenario. (b) GRADUAL: Scenario.

Fig. 5. Representation of the two testbed scenarios and the corresponding RSSI maps for the link connecting the data source to the mobile sink (high RSSI is in white, low RSSI in black).

strategy (direct C_0, single-channel relayed C_1 and multi-channel relayed C_2). We then compare the performance of those with the combination of C_0 and C_1 or C_2; We refer to these combinations with $C_{0,1}$ or $C_{0,2}$.

To experiment with a precise control of the mobile sink and enable reproducible experiments, we used a testbed based on a model railway [20]. A train carrying the sink moves in circles with a diameter of around 150 cm at a constant speed of 19 cm/s. As depicted in Figure 5, we reproduced two different scenarios, i.e., ABRUPT and GRADUAL, each involving one data source and two potential relays, with all the devices using transmission power level 3. The closed distance between the source and the candidate relays and their fixed positions result in high quality, stable links, like the ones available in the target scenario inside the clustered network of animals. In ABRUPT (Figure 5(a)), the communication range of the source was attenuated using an open metal box. The use of an open metal box resulted in abrupt link RSSI changes depending on the different positions of the mobile sink. This is mainly due to the strong impact of reflected signals, coming through the open side of the metal box. ABRUPT may present worse conditions than our target scenario; however, it allows us to evaluate ICELUS in a more challenging setup and better understand its behaviour. In GRADUAL (Figure 5(b)), instead, the metal box was closed, thus producing more gradual link quality changes since the sink is mainly receiving the attenuated signal from the source directly; Other effects like reflection have a less significant influence. Each experiment was performed with 10 different initial positions of the sink.

B. Single Strategy Optimisation

We first analyse the benefits of using an adaptive number of acknowledgements as modelled in Section IV-D.

In our testing scenario, each of the single strategies (C_0, C_1 and C_2) are compared with each other by either using one single static-ack or the adaptive-ack scheme. Table I shows, for the GRADUAL scenario, how each singular strategy can benefit from an adaptive-ack optimisation. This gain is indeed quite limited, in particular in comparison to the benefits of adapting full transfer strategies, as we discuss next. Moreover, the same experiment executed in the ABRUPT scenario reports a decrease in performance of, at least, 15 %. In fact, in the case of highly dynamic scenarios, link conditions change too

	Single-ack		**Adaptive-ack**	
(values/s)	∅	σ	∅	σ
C_0	744.3	33.0	813.2	34.2
C_1	406.2	24.8	438.9	25.7
C_2	215.4	14.2	218.3	12.2

TABLE I
THROUGHPUT OF THE ISOLATED STRATEGIES C_0, C_1 AND C_2 WITH SINGLE-ACK AND ADAPTIVE-ACK SCHEMES.

	Single Strategy			**Composite Strategy**	
(values/s)	∅	σ		∅	σ
C_0	500.0	62.4	-	-	-
C_1	399.9	1.6	$C_{0,1}$	600.0	27.1
C_2	488.8	4.1	$C_{0,2}$	501.0	62.6

TABLE II
COMPARISON OF THROUGHPUT BETWEEN INDIVIDUAL TRANSMISSION STRATEGIES AND THEIR COMPOSITION IN THE ABRUPT SCENARIO.

quickly for fine-grained optimisation, as the one we suggested in the adaptive-ack scheme. Furthermore, in ICELUS the sink monitors only the PRR along the incoming link from the source. Even if incoming and outgoing PRRs are correlating, the differences can cause a sub-optimal choice of the number of acknowledgements. Considering the limited gain and the potential performance decrease, the following experiments employ the more robust single-ack scheme.

C. Composition of Strategies

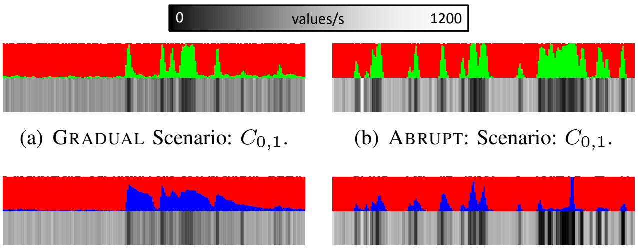
To evaluate how a combination of different strategies can improve the overall throughput, we configured ICELUS to use C_0 as initial strategy. If the communication quality decreases below a PRR of 0.6, the sink requests a strategy change to a relayed strategy (C_1 or C_2), as discussed in Section IV. When using a relayed strategy, the sink periodically monitors C_0 and switches back to a direct transfer if beneficial.

Table II shows the results for each isolated strategy (C_0, C_1, C_2) as well as for a combination of those ($C_{0,1}$, $C_{0,2}$) for the ABRUPT scenario. As expected, the single strategies C_0 and C_2 provide almost the same performance, with a higher throughput than C_1. For a combination of strategies, $C_{0,1}$ performs better than both C_0 and C_1, confirming our analysis. Selecting the most appropriate transfer strategy allows to exploit the benefits of the best fitting solution for the specific system conditions. Unexpected, however, is the bad performance of $C_{0,2}$, which does not provide any improvements over C_2 or C_0. To analyse this behaviour, we performed experiments to test the impact of the changes in the link quality on the resulting performance of $C_{0,1}$ and $C_{0,2}$.

The experiment results are summarised in Table III. Due to the different scenarios, the absolute numbers can not be compared with each other directly. However, the measured throughput in the GRADUAL scenario is lower for either strategy. Here, the algorithm adapts more efficiently because there is more time to identify and exploit a change in the

	GRADUAL		ABRUPT		**Outdoor**	
(values/s)	∅	σ	∅	σ	∅	σ
C_0	306.5	12.7	500.0	62.4	495.9	165.1
$C_{0,1}$	533.1	18.2	600.0	27.1	613.9	144.0
$C_{0,2}$	328.1	10.1	501.0	62.6	685.3	114.6

TABLE III
COMPARISON OF THROUGHPUT FOR THE DIFFERENT STRATEGY COMPOSITIONS IN THE GRADUAL, ABRUPT AND OUTDOOR SCENARIOS.



(a) GRADUAL Scenario: $C_{0,1}$. (b) ABRUPT: Scenario: $C_{0,1}$.

(c) GRADUAL: Scenario: $C_{0,2}$. (d) ABRUPT: Scenario: $C_{0,2}$.

Fig. 6. Average throughput and strategy maps for the experiments in the GRADUAL and ABRUPT scenarios. The bottom part reports the throughput; The upper part indicates the probability of ICELUS to choose the C_0 strategy (red) and C_1 (green) or C_2 (blue). A higher number of strategy switches caused by drastic link quality changes in the ABRUPT scenario results in higher overhead and worse throughput.

link quality. In such case, both the $C_{0,1}$ and $C_{0,2}$ strategies perform better than C_0. However, the performance of $C_{0,1}$ is significantly better than the performance of $C_{0,2}$, which is affected by the overhead of the explicit link monitoring phase.

Figure 6 provides a detailed picture of how a scenario affects the behaviour of the strategies. Each diagram reports the average throughput and strategy probability. As can be seen, more abrupt quality changes are causing more and shorter strategy switches. This incurs a higher switching overhead and a sub-optimal strategy utilisation. This overhead is higher in $C_{0,2}$ because the relays might not be available together in all cases and the algorithm requires a regular and explicit initialisation phase, making $C_{0,2}$ slower for adaptation.

D. Outdoor Scenario

We also executed a set of experiments in a national reserve in the context of the PLANET project. A source and four additional nodes were deployed at fixed positions in a distance of around 1.5 m, representative of the scenario depicted in Section III. The sink was attached to a UAV [2], whose trajectory was controlled remotely and maintained at a height of 3 m. The speed was fixed to 5 m/s along a circular trajectory with a diameter of about 15 m, all nodes where sending at transmission power level 30. Differently from before, these measurements are subject to the impact of physical objects (e.g. bushes) blocking the line of sight between the source and the sink. Moreover, the fast movement of a flying sink exposes different communication properties than a slow ground vehicle like the previously employed model train. Finally, the trajectory of the mobile sink was controlled manually, which introduces deviations in each repetition of the experiments. This fact effectively reproduces the effect of imprecise location information, as typically experienced in real deployments.

Table III shows that an improvement of the throughput is still possible. In this setting, $C_{0,2}$ is able to further increase the throughput. In fact, the communication from different relays is likely to be obstructed or hindered at different times, therefore benefiting from concurrent data streams. This adds to the fact that the direct connection is blocked for a long period. These experiments performed in a real deployment provide two interesting results compared to the testbed. First, the gain of using relays is lower, with a higher deviation. This effect is caused by the flying UAV that is reachable most of the time; this also justifies the high throughput of C_0. The terrain is also quite inhomogeneous, which creates highly variable links and thus causes a high standard deviation. Second, $C_{0,2}$ performs significantly better and has a lower standard deviation than in the testbed, in comparison to the other strategies. Indeed, the use of multiple channels makes the connection more robust against position changes of the mobile sink.

VII. CONCLUSION

This work focuses on maximising the "last mile" throughput to a mobile sink. We analyse different classes of transfer strategies and propose an adaptation scheme able to switch among them. The experimentation in real-world scenarios demonstrates that the resulting integrated solution, ICELUS, can exploit available communication resources to sustain a higher throughput at a negligible overhead. We now plan to further investigate the space of adaptation possibilities, e.g., with dynamic thresholds for the selection of relays.

REFERENCES

[1] "Planet project," 2013, http://www.planet-ict.eu/ (access: 01.03.2015). [Online]. Available: http://www.planet-ict.eu/
[2] "Parrot AR Drone 2.0," 2015, http://ardrone2.parrot.com/ (access: 30.11.2015). [Online]. Available: http://ardrone2.parrot.com/
[3] G. Anastasi *et al.*, "An adaptive data-transfer protocol for sensor networks with data mules," in *WoWMoM*, 2007.
[4] ——, "Motes sensor networks in dynamic scenarios: an experimental study for pervasive applications in urban environments," *JUCI*, 2007.
[5] N. Baccour *et al.*, "Radio link quality estimation in wireless sensor networks: A survey," *TOSN*, 2012.
[6] A. Cerpa *et al.*, "Statistical model of lossy links in wireless sensor networks," *IPSN*, 2005.
[7] M. Di Francesco *et al.*, "Data collection in wireless sensor networks with mobile elements: A survey," *TOSN*, 2011.
[8] F. Ferrari *et al.*, "Low-power wireless bus," in *SenSys*, 2012.
[9] D.-t. Ho *et al.*, "Heuristic algorithm and cooperative relay for energy efficient data collection with a UAV and WSN," *ComManTel*, 2013.
[10] T. Huang *et al.*, "Real-time horse gait synthesis," *JVCA*, 2013.
[11] O. D. Incel, "Survey paper: A survey on multi-channel communication in wireless sensor networks," *Computer Networks*, 2011.
[12] A. Kansal *et al.*, "Intelligent fluid infrastructure for embedded networks," in *MobiSys*, 2004.
[13] K. Krueger, "Social ecology of horses," in *Ecology of Social Evolution*. Springer, 2008.
[14] X. Liang *et al.*, "A novel cooperative communication protocol for QoS provisioning in wireless sensor networks," *TRIDENTCOM*, 2009.
[15] B. Mainaud *et al.*, "Cooperative communication for wireless sensors network: a mac protocol solution," *Wireless Days*, 2008.
[16] J. Mann *et al.*, "Prospeckz-5 a wireless sensor platform for tracking and monitoring of wild horses," in *DSD*, 2014.
[17] G. P. Picco *et al.*, "Geo-referenced proximity detection of wildlife with wildscope: Design and characterization," in *IPSN*, 2015.
[18] J. I. Ransom and B. S. Cade, *Quantifying Equid Behavior–A Research Ethogram for Free-Roaming Feral Horses*. General Books, 2011.
[19] F. Restuccia *et al.*, "Analysis and optimization of a protocol for mobile element discovery in sensor networks," *TMC*, 2014.
[20] H. Smeets *et al.*, "Trainsense: a novel infrastructure to support mobility in wireless sensor networks," in *EWSN*, 2013.
[21] J. So and N. H. Vaidya, "Multi-channel mac for ad hoc networks: Handling multi-channel hidden terminals using a single transceiver," in *MobiHoc*, 2004.
[22] M. Stolikj *et al.*, "Improving the performance of trickle-based data dissemination in low-power networks," in *EWSN*, 2015.
[23] M. Szczodrak *et al.*, "Dynamic reconfiguration of wireless sensor networks to support heterogeneous applications," in *DCOSS*, 2013.