A scalable architecture for distributed transmit beamforming with commodity radios: design and proof of concept

F. Quitin, M. M. U. Rahman, R. Mudumbai and U. Madhow

Abstract

We describe a fully-wireless prototype of distributed transmit beamforming on a software-defined radio platform. Distributed beamforming is a cooperative transmission technique that can achieve orders of magnitude increases in range or energy efficiency of wireless communication systems. However, this technique requires precise synchronization of the radio frequency signal from each transmitter. The significance of our prototype is in demonstrating that this requirement can be satisfied using digital signal processing methods on commodity hardware with low-quality oscillators. Our synchronization approach scales to large numbers of transmitters: each transmitter runs independent algorithms based on periodically transmitted feedback packets from the receiver. A key simplification is the decoupling of the algorithms for frequency locking and beamsteering at each transmitter, even though both processes use the same feedback packets. Frequency locking employs an Extended Kalman filter to track the local oscillator offset between a transmitter and the receiver, using frequency offset measurements based on the feedback packet *waveform*, while the phase adjustments for beamsteering are determined using a one-bit feedback algorithm based on the feedback packet *payload*. Our prototype demonstrates that distributed transmit beamforming can be incorporated into wireless networks without requiring hardware innovations, and provides open-source building blocks for future research and development.

F. Quitin and U. Madhow are with the Electrical and Computer Engineering Department, University of California, Santa Barbara (UCSB) ({fquitin, madhow}@ece.ucsb.edu)

M. Rahman and R. Mudumbai are with the Electrical and Computer Engineering Department, University of Iowa (mahboob-rahman@uiowa.edu, rmudumbai@engineering.uiowa.edu)

I. INTRODUCTION

Distributed transmit beamforming refers to a cooperative transmission technique for wireless networks under which several nodes calibrate their transmissions in such a way that their individual transmitted signals add up coherently at an intended receiver. In effect, under this scheme, the transmit nodes act as a "virtual antenna array" and direct a beam towards the receiver, enhancing communication range and/or power efficiency [1–3].

The key challenge in realizing the large potential gains from beamforming is in precisely synchronizing the RF signals from each transmitter so that they are aligned in phase at the intended receiver. Each transmitter obtains its RF carrier signal from a separate local oscillator, and even when two oscillators are set to the same nominal frequency, because of manufacturing tolerances and temperature variations, they would in general have a non-zero frequency offset with respect to each other. In addition, all oscillators undergo random phase and frequency drifts over time. Finally, unlike a traditional phased array, a virtual array of collaborating wireless nodes does not have a regular and precisely known geometry. Thus, even if the destination's location were known precisely (which is not the case in practice), it is not possible to determine the phases that the transmitters in the virtual array must employ in order to direct energy towards the destination. Indeed, explicitly computing beamforming weights requires knowledge of the geometry of the transmitters and receiver to within a small fraction of the carrier wavelength, and standard localization techniques such as GPS fall far short of the accuracy necessary to overcome this geometric uncertainty. Thus, distributed beamforming requires a sophisticated synchronization architecture that accounts for all of the preceding uncertainties. The prototype described in this paper illustrates a robust and scalable feedback-based approach to this problem, and successfully demonstrates that distributed transmit beamforming is indeed feasible even with off-the-shelf hardware and low-quality oscillators.

Contributions: We present an architecture for distributed transmit beamforming that is suitable for packet wireless networks, and demonstrate it using a software-defined radio testbed. The objectives of our prototyping effort are two-fold: (a) a proof-of-concept implementation to show that distributed beamforming can be implemented on commodity hardware with minimal overhead using signaling similar to that in existing packet wireless networks, and (b) reusable open-source building blocks for further development, which we hope will lead to eventual incorporation of distributed beamforming in real-world WiFi, Zigbee and cellular wireless networks. Our main contributions are summarized as follows.

1) **Scalable synchronization architecture.** We propose a decentralized, feedback-based synchronization architecture in which each transmitter adapts its frequency and phase independently, based on

feedback from the receiver. In principle, this allows the system to scale to an indefinitely large number of cooperating transmitters. At each transmitter, the problems of frequency locking and beamsteering are decoupled into two parallel algorithms. From the point of view of implementation, a key contribution is the use of the same all-digital feedback packets for both parallel algorithms: the single bit of feedback used for phase adaptation for beamsteering is embedded in the payload of a GMSK modulated packet, and the modulated waveform in the packet is used to extract information for frequency locking.

2) **Extended Kalman filter for frequency locking.** Each transmitter employs an extended Kalman filter (EKF) to lock its frequency to that of the receiver. The EKF is driven by phase and frequency offset measurements made from the modulated waveforms in the feedback packets, but does not make use of the payload of the feedback packet (which carries information to be used for beamsteering). The EKF provides a robust mechanism for using wrapped phase offset measurements to track the unwrapped phase offset in the state evolution. We provide an approximate analysis of the EKF. This yields design guidelines for the required feedback rate which are validated through experiments.
3) **1-bit feedback based beamsteering.** The receiver broadcasts one bit indicating the change in its received signal strength in each feedback packet. Each transmitter uses this to make phase corrections (on top of the frequency/phase corrections for frequency locking based on the EKF) for beamsteering, using the randomized ascent algorithm first proposed in [4].

Related work: Many information-theoretic analyses, ranging from three decades back [5, 6] to the present [7–9], rely on the concept of distributed transmit beamforming, without addressing the synchronization requirements involved in realizing it. Over the last decade, however, there have been serious attempts to tackle these difficult synchronization problems, and a menu of synchronization techniques have been studied, including closed loop methods with explicit channel feedback for each transmitter [10], one-bit aggregate feedback [4, 11, 12], implicit feedback using reciprocity [2], round-trip synchronization [13, 14], and two-way synchronization [15, 16]. These different synchronization techniques represent different sets of tradeoffs between overheads of coordination, channel feedback and complexity (see the survey article [3], and a discussion of more recent work in [17]). Distributed beamforming has also been recently studied in the context of "coordinated multipoint (CoMP)" capabilities for 4G-LTE cellular systems, where multiple base stations act as a distributed antenna array [18].

In addition to these theoretical investigations, this technique has also been demonstrated in several experimental prototypes [11, 19–21]. The 1-bit feedback algorithm, presented in [11, 12], has proved to

be especially popular for experimental investigations because of its simplicity and scalability.

The results reported in this paper represents a significant advance over previous work in a number of important respects. Early experimental prototypes for distributed beamforming [11, 19, 20] all used wired side-channels for distributing common reference clock signals and/or channel feedback messages to the transmitters. Similarly the CoMP studies for 4G cellular systems assume that the cooperating base stations are all connected by dedicated backhaul links with high bandwidths and low latencies, as well as uninterrupted GPS satellite connections. These assumptions substantially simplify the synchronization process, and therefore these early prototypes do not fully resolve the question of the feasibility of distributed beamforming in practical ad-hoc wireless networks. Furthermore these prototypes all use custom-designed hardware which is not compatible with existing standards and devices.

An important contribution of the prototype described here, relative to prior work, is to show that all-wireless distributed beamforming can be achieved with commodity hardware with modest overhead, without requiring external (e.g., GPS-based) synchronization or wired side channels. The results in this paper significantly improve upon our prior efforts [21, 22] towards this goal. The early prototype in [21] uses analog signaling, and signal processing techniques based on analog circuits such as Costas loops, which are less efficient and require a complex calibration process (specifically, manual estimation of the latency of each node every time it is powered on). The present prototype is an evolution of an earlier implementation reported in our recent conference paper [22]: while this also employed digital feedback messages, it employed separate training signals for frequency locking and beamsteering. The present prototype does not require manual calibration as in [21], and improves upon [22] by using the same digital feedback packets (with standard modulation formats) for both purposes. A recent live demo of the prototype was presented at [23]. We go beyond these conference publications by providing detailed description and analysis of our architecture, including an analysis of the EKF which yields design guidelines for setting the feedback rate.

As with prior prototypes, we demonstrate coherent addition of a carrier tone at the receiver, which requires frequency locking and phase alignment. For distributed beamforming with modulated waveforms, an additional requirement is baseband timing alignment (or more generally, channel dispersion compensation). While the latter is not addressed here, frequency synchronization and narrowband phase alignment as implemented here could be used as building blocks for multicarrier techniques for distributed beamforming over wideband dispersive channels.

Outline: The remainder of the paper is organized as follows. Section II presents the general architecture of our distributed transmit beamforming setup, and discusses the design implications of fundamental limits on

frequency/phase synchronization. Section III focuses on frequency synchronization. The extended Kalman filtering framework is presented, together with approximate modeling and performance analysis which yield design guidelines. Finally, Section IV describes implementation of our architecture on software-defined radios, and presents experimental results from our testbed.

II. Architecture for distributed beamforming

We first outline a system model motivating our implementation. The underlying approach is quite general, and is broadly applicable to WiFi, Zigbee and other packet wireless networks.

A. System model

Consider a distributed array with N nodes which seek to collaboratively transmit a common complex baseband message signal $m(t)$ to a receiver. Let the RF signal transmitted by a transmit node be denoted by $x(t)$ and θ the phase gain of the channel from the transmitter to the receiver (note that θ is different for each transmitter). In order to achieve beamforming, each transmit node needs to synthesize and transmit an RF signal $x(t) = \Re\left(m(t)\exp(j2\pi f_c t - j\theta)\right)$.

However, each transmitter has a local oscillator (LO) that derives its RF signal from its own separate crystal reference which in general have small but non-zero frequency offsets with each other. Let α denote the fractional offset of the reference signal of the transmit node from some standard universal reference. These offsets are usually specified in parts per million (ppm); thus, a 10 ppm offset corresponds to $\alpha = 10e-6$. The resulting frequency offset of the LO signal of the transmit node at the nominal frequency f_c is $\Delta f \equiv \alpha f_c$. Thus the node's LO signal can be written as $e(t) = \exp\left(j(2\pi(f_c + \Delta f)t + \Delta\phi)\right.$, where $\Delta\phi$ is the (unknown) phase offset. The problem of beamforming is to synthesize the signal $x(t)$ from $e(t)$.

Figures 1 shows a schematic representation of our system. As shown in the figure, the receiver regularly broadcasts feedback messages; the transmitter nodes use these feedback messages to estimate and correct for the offsets $\Delta f,\ \Delta\phi$ and to construct the desired beamforming signal $x(t)$.

Figure 2 shows the the time-slotting model for the transmit nodes. Every time a feedback message is received from the receiver, the transmit node will use the feedback message to make an estimation of the frequency and phase offset Δf and $\Delta\phi$. These estimations will be used by the transmit nodes to predict and compensate for the LO offset until the next feedback message is received.

The feedback messages can be transmitted over the same frequency band as the beamforming signal using a medium access control mechanism enabling time sharing, or we can employ frequency division

multiplexing, with the feedback and beamforming signals sent over different frequency bands. While our architecture applies to both scenarios, we employ frequency division multiplexing in our prototype. Thus, the beamforming nodes employ a reference signal at one frequency to synthesize a synchronized RF signal at a different frequency. To do this, we assume that both carrier frequencies are synthesized from the same crystal oscillator, hence there is a known multiplicative relationship between them which also applies to the frequency offsets between two nodes that we wish to estimate and correct for. Specifically, the offset Δf_2 at a frequency $f_{c,2}$ can be obtained using the measured offset Δf_1 at another frequency $f_{c,1}$ as $\Delta f_2 = \alpha f_{c,2} \equiv \Delta f_1 \frac{f_{c,2}}{f_{c,1}}$.

B. Two synchronization sub-processes

A key feature of our implementation is that the beamforming process is decoupled into two sub-processes that run independently and concurrently. Roughly speaking, the first sub-process compensates for the frequency offset Δf, and the second sub-process compensates for the unknown phases $\Delta\phi$ and θ.

1) **Frequency locking.** In this sub-process, each transmitter locks its oscillator on to a shared reference signal, which in our case is the set of feedback packets from the receiver. The purpose of this sub-process is to ensure that the transmitters all have RF signals with the same frequency and a fixed (but unknown) phase relationship with each other. The LO frequency offsets that can occur in typical software-defined radios can range up to several kHz, making the frequency synchronization of the transmit nodes especially challenging.
2) **Beamsteering.** This sub-process adjusts the phase relationship between the transmitters in such a way that their transmitted signals add up coherently at the intended receiver. In our case, the feedback packets from the receiver contains the one bit of SNR feedback in its payload that is used to steer the beam using the 1-bit algorithm.

The main motivation for this decoupled design is simplicity: the 1-bit algorithm is easy to implement and has low overhead. While it can be modified to provide both frequency and phase synchronization [19], it cannot handle the significant frequency drifts that we encounter in our prototype, especially given the large latencies in the feedback channel. The frequency locking process estimates and eliminates these frequency offsets and allows the simple 1-bit algorithm to achieve and maintain coherence.

Our implementation of the two synchronization sub-processes are shown in the block diagram of a transmit node in Figure 3. A key feature of our design is that the same set of periodically transmitted feedback packets from the receiver (depicted in Figure 4) are used for both synchronization sub-processes:

the payload of the feedback packet contains the one bit of SNR information to drive the beamforming algorithm, and the frequency and phase offset of the transmitter's oscillator with respect to the receiver is implicitly contained within the preamble and header symbols of the packet.

C. Fundamental limits of frequency and phase estimation

The performance limits and convergence properties of the 1-bit beamforming algorithm (assuming ideal frequency synchronization) have been studied in detail [12]. We therefore focus here on understanding the frequency synchronization sub-process, which estimates frequency and phase using training signals transmitted in short periodic bursts of duration T_{est}, and then smooths these estimates using an EKF. The time between these bursts is denoted by T_{slot}. In this section, we discuss what insights the Cramer-Rao Lower Bound (CRLB) for one-shot frequency/phase estimation provides regarding the desirable regime of operation for the frequency synchronization sub-process. These insights are then verified by simulations and experiments quantifying EKF performance.

Consider the process of obtaining one-shot frequency and phase estimates using a noise-corrupted reference signal received by a transmitter over the training epoch of duration T_{est} in one time-slot. Let $a(t) = A\exp(j\phi(t)) + n(t), t \in [0, T_{est}]$ which is the complex baseband waveform corresponding to one feedback packet upon demodulation using the LO signal of the transmit node. The post-integration SNR of this signal is defined as $\mathrm{SNR} \equiv \frac{A^2 T_{est}}{2N_0}$, where N_0 is the power spectral density of the white noise process $n(t)$. The CRLBs for this one-shot phase and frequency estimation process are well-known in the literature [24, 25]: if ϕ_{err} and f_{err} respectively denote the one-shot phase and frequency estimation errors, we have

$$\begin{aligned} \sigma_\phi^2 \doteq E\Big[\phi_{err}^2\Big] &\geq \frac{2}{\mathrm{SNR}} \\ \sigma_f^2 \doteq E\Big[f_{err}^2\Big] &\geq \frac{3}{2\pi^2 T_{est}^2 \mathrm{SNR}} \end{aligned} \tag{1}$$

Consider now the phase error that results when transmitters use one-shot frequency and phase estimates from the training interval to predict and correct for the frequency and phase offsets of their oscillators over the subsequent time slot. The variance of the resulting error $\phi(t) - \hat{\phi}(t)$ between the predicted phase offset $\hat{\phi}(t)$ and actual phase offset $\phi(t)$ of the transmitter with the reference signal grows with time and its value at the end of the time-slot can be written as

$$\begin{aligned} E\Big[\big(\phi(t) - \hat{\phi}(t)\big)^2\Big]_{t=T_{slot}} &= \sigma_\phi^2 + T_{slot}^2 (2\pi\sigma_f)^2 \\ &\geq \frac{2}{SNR}\left(1 + \frac{3}{\eta^2}\right). \end{aligned} \tag{2}$$

When the duty cycle of the estimation process is small i.e. $\eta \equiv \frac{T_{est}}{T_{slot}} \ll 1$, then the second term in (2) dominates; in this setting, *one-shot frequency estimates are highly unreliable* as compared to the phase estimate.

Now consider an alternative approach to the frequency estimation problem. Instead of doing one-shot frequency estimates, we can also estimate frequency by using two one-shot phase estimates in two successive training epochs T_{slot} seconds apart. In other words, we consider the frequency estimate $\tilde{f} \doteq \frac{\hat{\phi}(T_{slot})-\hat{\phi}(0)}{2\pi T_{slot}}$. This estimate has the variance

$$\mathrm{var}(\tilde{f}) = \frac{2\sigma_\phi^2}{(2\pi T_{slot})^2} \geq \frac{1}{\pi^2 \mathrm{SNR} T_{slot}^2}, \tag{3}$$

and this variance can be significantly smaller than the one-shot frequency variance σ_f^2 in (1). This suggests that we might be better off dispensing with one-shot frequency estimates altogether, and rely on averaging phase estimates over multiple time slots to get good frequency estimates. Indeed, this approach, implemented using a Kalman filter, is what is employed in [26]. However, using phase estimates alone for both phase and frequency tracking requires access to *unwrapped* phase estimates. This in turn requires that the frequency error in our estimate is small enough that 2π ambiguities in phase do not appear over the slot duration T_{slot} between successive training bursts. For the low-quality oscillators in our software-defined radios, the frequency drift is severe enough that satisfying this assumption would require excessive overhead.

In order to circumvent the preceding phase unwrapping problem, we employ crude one-shot frequency estimates to complement the phase estimates. These one-shot frequency estimates need only be good enough to avoid phase unwrapping errors over a single time-slot; in other words, we want $\sigma_f T_{slot}$ that is not too much larger than unity. Plugging this into (1), we obtain the following rule of thumb.

CRLB-based rule of thumb: $\frac{T_{slot}}{T_{est}} \approx k\sqrt{SNR}$, where $k = \sqrt{\frac{2}{3}}\pi$. Interestingly, this requirement only applies to the ratio $\frac{T_{slot}}{T_{est}}$ or equivalently to the duty-cycle of the training signal, not individually to T_{slot} or T_{est}. Note that this requirement is only meant to provide very rough guidance. More detailed design insights are obtained via numerical simulations and experiments in Section III-B.

III. Frequency synchronization

The frequency synchronization sub-process is divided into three stages, as shown in Figure 3. In the first stage, the transmit node, upon receipt of each feedback packet, makes a measurement of its LO frequency and (wrapped) phase offset relative to the receiver using a blind estimation algorithm. In the second stage, the EKF uses these LO frequency and wrapped LO phase offset measurements to keep

track of the unwrapped LO phase offset. In the third stage, the transmit node compensates for the LO offset based on the latest LO frequency and phase offset values as predicted by the EKF.

The blind estimation algorithm used in the first stage depends on the modulation format used for the feedback message. For most classical modulation formats (PSK, QAM, GMSK etc.), these algorithm transform the feedback message into a pilot tone, whose frequency can easily be estimated with classical frequency estimation theory. After compensating the feedback message for the LO frequency offset, the LO phase offset can easily be measured by correlating the feedback message with the (known) message header. This will yield a *wrapped* measurement for the LO phase offset. The estimation algorithm used for our implementation applies to GMSK feedback messages [27, 28], and is described in detail in Appendix A.

A. Extended Kalman filter state-space model

We use the following discrete state-space model for the LO offsets of each transmit node relative to the receiver.

$$\mathbf{x}_{k+1} = \mathbf{F}\mathbf{x}_k + \mathbf{w}_k \tag{4}$$

where $\mathbf{x}_k = \left[\phi_k, \omega_k\right]^T$ is the LO phase and angular frequency offset of the transmit node with respect to the receive node at time-slot k (where $\omega_k = 2\pi\Delta f_k$). The state update matrix $\mathbf{F}$ is defined by

$$\mathbf{F} = \begin{bmatrix} 1 & T_{slot} \\ 0 & 1 \end{bmatrix}$$

and T_{slot} is the period of the feedback messages. Note that if aperiodic feedback messages are considered, T_{slot} is not fixed and the state update matrix $\mathbf{F}$ is allowed to be time-varying. The process noise vector $\mathbf{w}_k \sim \mathcal{N}\left(0, \mathbf{Q}\left(T_{slot}\right)\right)$ is the noise that causes the LO phase and frequency offset to deviate from their nominal value.

We use the following measurement model for the blind LO offset estimation algorithm which provides the inputs for the EKF.

$$\mathbf{z}_k = \mathbf{h}\left(\mathbf{x}_k\right) + \mathbf{v}_k \tag{5}$$

where

$$\mathbf{h}\left(\mathbf{x}_k\right) = \begin{bmatrix} \cos\left(\phi_k\right) \\ \sin\left(\phi_k\right) \\ \omega_k \end{bmatrix}$$

and $\mathbf{v}_k \sim \mathcal{N}(0, \mathbf{R})$ is the additive white Gaussian measurement noise. Note that (5) defines a *non-linear* measurement model reflecting the fact that the blind estimation algorithm yields only an estimate of the *wrapped* phase offset.

The equations that determine the EKF evolution are split into two stages: an update phase and a prediction phase. The update phase corrects the current state estimate given the last measurement $\mathbf{z}_k$, and is mathematically defined as

$$\mathbf{y}_k = \mathbf{z}_k - \mathbf{h}\left(\mathbf{x}_{k|k-1}\right) \tag{6a}$$

$$\mathbf{S}_k = \mathbf{H}_k \mathbf{P}_{k|k-1} \mathbf{H}_k^T + \mathbf{R} \tag{6b}$$

$$\mathbf{K}_k = \mathbf{P}_{k|k-1} \mathbf{H}_k^T \mathbf{S}_k^{-1} \tag{6c}$$

$$\mathbf{x}_{k|k} = \mathbf{x}_{k|k-1} + \mathbf{K}_k \mathbf{y}_k \tag{6d}$$

$$\mathbf{P}_{k|k} = \left(\mathbf{I}_2 - \mathbf{K}_k \mathbf{H}_k\right) \mathbf{P}_{k|k-1} \tag{6e}$$

The matrix $\mathbf{H}_k$ is the Jacobian of the function $\mathbf{h}$:

$$\mathbf{H}_k = \left.\frac{\partial \boldsymbol{h}}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}_{k|k-1}}$$

The prediction phase gives an estimation of the future state $\mathbf{x}_{k+1|k}$ to be used in the update phase of next EKF cycle:

$$\mathbf{x}_{k+1|k} = \mathbf{F} \mathbf{x}_{k|k} \tag{7a}$$

$$\mathbf{P}_{k+1|k} = \mathbf{F} \mathbf{P}_{k|k} \mathbf{F}^T + \mathbf{Q} \tag{7b}$$

For each EKF cycle, the values contained in the vector $\mathbf{x}_{k|k}$ give a filtered estimate for the unwrapped LO phase offset and LO angular frequency offset. The output of the EKF in Figure 3 will be the filtered unwrapped LO phase offset $\widetilde{\phi}_k$ and LO frequency offset $\widetilde{\omega}_k$.

The interplay between LO phase offset and LO frequency offset in equations (6)-(7) can be intuitively understood by considering the elements of $\mathbf{y}_k$. It is first important to observe that the phase terms of $\mathbf{y}_k$ (the first two elements of $\mathbf{y}_k$) cannot exceed 2, whereas the frequency term of $\mathbf{y}_k$ (the third element of $\mathbf{y}_k$) can be arbitrarily large. In the early cycles of the EKF, the differences between the estimated and measured LO frequency offsets are often large. As a results, the phase terms of $\mathbf{y}_k$ will be negligible compared to the frequency term of $\mathbf{y}_k$, and the LO frequency offset will be the main driving element of the EKF. Once the estimated LO frequency offset approaches its measured values, the phase terms of $\mathbf{y}_k$ will no longer be negligible compared to the frequency term of $\mathbf{y}_k$. In this regime, the previously

predicted LO frequency offset is used to determine the number of 2π-phase wraps that has occurred between the previous cycle and the current one. The current LO phase and frequency measurement are then used to adjust the previously predicted LO phase and frequency offset.

B. EKF convergence

It was argued in Section II-C that in order for our estimation framework to successfully handle phase unwrapping ambiguities, $\sigma_f T_{slot}$ should not be too large. The value of $\sigma_f T_{slot}$ depends on T_{slot}, T_{est} and the SNR of the feedback link. In this section, we will investigate how these parameters affect the performances of the EKF, both with simulations and experiments.

Our simulations are based on a model for the process noise covariance matrix $\mathbf{Q}$ borrowed from [26, 29]. The state-space noise covariance matrix is defined by

$$\mathbf{Q}(T_s) = \omega_c^2 q_1^2 \begin{bmatrix} T_s & 0 \\ 0 & 0 \end{bmatrix} + \omega_c^2 q_2^2 \begin{bmatrix} \frac{T_s^3}{3} & \frac{T_s^2}{2} \\ \frac{T_s^2}{2} & T_s \end{bmatrix} \tag{8}$$

where ω_c is the carrier frequency and T_s is the sample period. The parameters q_1^2 and q_2^2 are the process noise parameters that correspond to white frequency noise and random walk frequency noise, respectively. For a class of oscillators, these two parameters can be obtained by using the Allan variance.

The Allan variance is a tool to characterize the frequency stability of an oscillator, under the presence of various noise sources. It is mathematically defined as

$$\sigma_y^2(\tau) = \frac{1}{2\tau^2} \left\langle \left(\phi(t+2\tau) - 2\phi(t+\tau) + \phi(t)\right)^2 \right\rangle_t \tag{9}$$

where $\phi(t)$ is the LO phase offset at time instant t with respect to some absolute reference. By applying equation (9) to the state-space model (4) and the noise model (8), it is shown in [29] that the following theoretical model can be obtained for the Allan variance:

$$\sigma_y^2(\tau) = \frac{q_1^2}{\tau} + \frac{q_2^2 \tau}{3} \tag{10}$$

The Allan variance can also be measured experimentally be sending a pilot tone with a transmitter, and by recording the received pilot tone (which will contain a certain LO clock offset). By entering the unwrapped phase of the received pilot tone in (9) for various values of τ, it is possible to obtain an experimental curve for the Allan variance. By fitting experimental Allan variance measurements to the theoretical model (10), it is possible to obtain values for q_1^2 and q_2^2. For the software-defined radios used in the setup described in this paper, the obtained parameters are $q_1^2 = 8.47 \times 10^{-22}$ and $q_1^2 = 5.51 \times 10^{-18}$.

Figure 5 shows the simulated average phase error at the output of the EKF for various values of T_{slot} and T_{est}. Figure 5 also shows the CRLB-based rule of thumb deduced in Section II-C. It can be observed from the simulation results that there is a steep transition between the area where the EKF works perfectly, with phase errors close to zero, and the area where the EKF diverges, with phase errors that become arbitrarily large. It can also be observed that, for various SNR levels of the feedback link, the convergence/divergence transition area of the EKF depends on $\frac{T_{slot}}{T_{est}}$. For SNR levels above 20 dB, it can be seen that the convergence/divergence transition area of the EKF matches fairly well with the CRLB-based rule of thumb. Note that the rule of thumb does not represent a hard threshold, but rather a transition zone between the EKF convergence and EKF divergence area.

The experimental setup (described in Section IV) with one transmit node was also used to determine the convergence/divergence of the EKF. Note that in order to obtain a measurement of the LO frequency offset, the GMSK feedback packet is transformed in a pilot tone with a non-linear transformation, as explained in Appendix A. The theory developed before for pilot tone thus also applies to our setup, which uses GMSK feedback packets. Figure 6 shows the phase of the signal received by the receive node when only one of the transmit node is transmitting (when the transmit node is synchronizing on the receive node). In this figure, T_{slot} is varied while T_{est} is fixed to 5.1 ms. It can be seen that for high values of T_{slot} (feedback rates below 20 Hz), synchronization is not achieved and the frequency of the transmitted signal is not synchronized with the receiver, hence the phase variations. For feedback rates of 20 Hz and above, the received signal has a fairly constant phase indicating EKF convergence.

Figure 7 shows the convergence/divergence of the EKF for various values of T_{slot} and T_{est}, both experimentally and with simulations. The CRLB of $\sigma_f T_{est}$ is again plotted for comparison. The LO phase offset error can not be determined experimentally since the real LO phase offset cannot be measured; it can only be observed if the EKF of the transmit node is synchronized on the receive node or not (similarly to what can be observed in Figure 6). The circles represents values of $\{T_{slot}, T_{est}\}$ for which the transmit node was able to synchronize to the receive node, the crosses represent values of $\{T_{slot}, T_{est}\}$ for which the transmit node could not synchronize to the receive node. It can be seen that the experimental results match well with the simulations, and that both match quite well with the CRLB-based rule of thumb.

C. Performance of the EKF

The error on the filtered LO phase and frequency estimate $\widetilde{\phi}_k$ and $\widetilde{\omega}_k$ can be recovered from the state error covariance matrix $\mathbf{P}_{k|k}$. The diagonal elements of this matrix correspond to the LO phase estimation error $\sigma^2_{\widetilde{\phi}}$ and the LO frequency estimation error $\sigma^2_{\widetilde{\omega}}$. For a stable system, this matrix converges to a stable

value, which can be computed recursively with equations (6b), (6c), (6e) and (7b). For an EKF, this value cannot be computed independently of the state, since the Jacobian $\mathbf{H}_k$ depend on the actual state $\mathbf{x}_{k|k}$. For that reason, an equivalent linear Kalman filter is defined with the state-space model from (4) and the following measurement model:

$$\mathbf{z}_k = \mathbf{H}\mathbf{x}_k + \mathbf{v}'_k \tag{11}$$

where $\mathbf{H} = \mathbf{I}_2$ is an identity matrix and $\mathbf{v}'_k \sim \mathcal{N}(0, \mathbf{R}')$ is the additive white Gaussian measurement noise. For this linear Kalman filter, the measurement matrix $\mathbf{H}_k = \mathbf{I}_2$ is independent of the state, and $\mathbf{P}_{k|k}$ can be computed independently of the state. If the EKF is in the convergence area determined in Section III-B, it can be assumed that the number of 2π-phase wraps is measured correctly, and the performance of the EKF is expected to be identical to the performance of the linear Kalman filter.

The measurement noise matrix corresponding to our setup is $\mathbf{R} = \begin{bmatrix} 0.05 \cdot \pi/180 & 0 \\ 0 & 1.5 \cdot 2\pi \end{bmatrix}$. By entering this and the process noise parameters in equations (6b), (6c), (6e), (7b) and (8), it can be computed that the state error covariance matrix converges to a stable value after a few cycles. The error on the frequency estimate at the output of the Kalman filter $\sigma^2_{\widetilde{\omega}}$ is given by element $(2,2)$ of $\mathbf{P}_{k|k}$.

The frequency offset correction block in Figure 3 will use the latest phase and frequency estimates until new ones are obtained from the EKF. When computing the error on the filtered LO phase and frequency estimates $\sigma_{\widetilde{\phi}}$ and $\sigma_{\widetilde{\omega}}$, it can be determined that the LO phase estimation error is very low. It is mainly the LO frequency estimation error that will deviate the phase of the beamformed signal until the next state estimate is obtained from the EKF. The standard deviation of the maximum phase error then corresponds to $T_{slot}\sigma_{\widetilde{\omega}}$. The standard deviation of the maximum phase error as a function of the feedback rate $1/T_{slot}$ is shown in Figure 8. In [2] it was determined that as long as the phase error stays below $15°$, the beamforming gain stayed above 95% of the maximum beamforming gain. The $15°$ treshold is shown in Figure 8. For feedback rates above 15 Hz, the phase error shows that the beamforming gain can be maintained. The standard deviation of the maximum phase error was also measured experimentally with our experimental setup (described in Section IV). It can be seen that there is a reasonable match between the theoretical prediction and the experimental result.

For the remainder of the paper, T_{slot} was fixed to 50 ms (a feedback rate of 20 Hz) and T_{est} was fixed to 5.1 ms, so that the setup would work even for lower SNR values, while maintaining the phase error of the beamformed signal below $15°$.

IV. Implementation on software-defined radios and results

The previously described architecture was implemented on a software-defined radio testbed using USRP RF and baseband boards [30]. Our setup used a mix of USRP-2 and USRP-N200 baseband boards, and WBX 50-2200 MHz RF daughterboards; however our implementation is completely portable to any compatible SDR hardware. Our implementation is freely available for download online [31].

The estimation of the LO frequency and phase offset, using the GMSK feedback message, is explained in Appendix A. This operation also gives the precise time of arrival of the packet, and this measured value is then used for T_{slot} in the state-space matrix of the EKF.

In our setup, the feedback messages from the receiver to the beamforming nodes are sent at a frequency of 964 MHz and the beamforming signal itself is transmitted at a different frequency 892 MHz. Thus the EKF tracks the LO offset at the 964 MHz frequency, and translates this to the 892 MHz frequency using the method described in Section II, to compensate for the LO offsets in the beamforming signal.

Once frequency synchronization is achieved between the different transmit nodes, the absolute phase of the different nodes must be adjusted so that the different transmitted signal add up constructively at the receiver. The one-bit feedback algorithm is used to achieve this [11], and works as follows. At each time slot, all the transmitters add a random phase perturbation to their current phase. The receiver monitors the aggregate received signal strength (RSS) of the beamforming signal and periodically broadcasts a packet with one bit of feedback indicating increase or decrease of the RSS. The transmitters retain those perturbations that lead to RSS increases and discard the others.

The main challenge for the implementation of this algorithm on the SDR testbed is the large global round-trip latency of the system. The round-trip latencies measured with the USRPs ranged from 10 ms to 30 ms. The chosen feedback rate of 20 Hz (or 50 ms between packets) ensures that the transmitters have the time to add their new random phase perturbation before the receiver starts to measure the RSS. In our implementation the random phase perturbations are chosen to be either $+15^\circ$ or -15° which offers a good trade-off between convergence speed and residual signal fluctuations [11].

A. Experimental results

Figure 9 shows the received power at the receiver node when two nodes are cooperating to beamform towards the receiver. When the second node is turned on, it can be seen that it takes some time for the second transmit node to synchronize its frequency and to synchronize its phase. Once this is achieved, it can be seen that the received amplitude is the sum of amplitudes of the two individual transmitters.

Figure 10 zooms in on the synchronization part in Figure 9. It can be seen that at first, the second transmit node is still synchronizing its frequency: the total signal is the sum of two sinusoids of unequal frequency. Once frequency synchronization is attained, the one-bit feedback algorithm will adapt the phases of the two transmit nodes until the resulting signal is the sum of the two individual signals. The time of the frequency synchronization can be reduced by entering a more precise value for the initial the LO frequency offset in the EKF.

The algorithm described in this paper is scalable to larger network sizes. Figure 11 shows an example with three transmit nodes. When running the prototype, it could be observed that the received signal power sometimes drops. This can happen when the EKF convergence is momentarily lost due to a software or hardware lags in the setup. However, it could be observed that the system was able to recover quite quickly from such errors.

V. Conclusion

We have presented a scalable feedback-based synchronization architecture for distributed transmit beamforming, and have demonstrated that theoretical beamforming gains can be attained with commodity SDR hardware with moderate overhead. The EKF provides a robust frequency locking mechanism, using measurements of the frequency and wrapped phase LO offsets to track the unwrapped phase offset. The theoretical predictions from our analysis of the EKF match experiments, and indicate that further reductions in feedback rate can be obtained with better quality LOs. The decoupling of the one-bit feedback based phase adjustments for beamforming from EKF-based locking, while using the same set of feedback packets for both algorithms, greatly simplifies the implementation. We have shown that it is possible to use advanced modulation formats for the feedback, which opens the way for incorporating our synchronization architecture into prevalent wireless standards. The open-source implementation of our prototype is publicly available, and hopefully provides a good starting point for further experimental research on distributed MIMO.

An important next step is to extend our framework to transmission of modulated messages over wideband dispersive channels, which require baseband timing alignment as well as channel dispersion compensation. This involves design of signal processing architectures and algorithms, and their incorporation into suitably defined cross-layer protocols. Experimental investigation of the effect of mobility is another important topic.

APPENDIX A
LO FREQUENCY AND PHASE OFFSET ESTIMATION WITH A GMSK SIGNAL

We first describe the blind estimation of the LO frequency offset. The complex envelope of a GMSK signal can be written as

$$s(t) = \exp\left(j\frac{\pi}{2}\sum_i a_i q(t - iT)\right) \tag{12}$$

where a_i is the symbol stream, T is the symbol period and $q(t)$ is the phase pulse (which is a Gaussian phase pulse in the case of GMSK). Note that for an over-sampling rate of P, the symbol period is $T = PT_s$ where T_s is the sample period of the system.

If transmitted over a AWGN channel, the complex envelope of the received signal in the presence of LO frequency offset can be modeled as

$$r(kT_s) = \exp(j(\omega kT_s + \phi))\exp\left(j\frac{\pi}{2}\sum_i a_i q(t - ikT_s)\right) + v(kT_s) \tag{13}$$

where ω and ϕ are the LO frequency and phase offset, respectively, and $v(kT_s)$ is the noise term.

We define a non-linear transformation of the signal as follows:

$$z[k] = (-1)^k r^2(kPT_s) \tag{14}$$

It is shown in [27, 28] that $z[k]$ can be approximated as a discrete time sine wave with a frequency that is twice the LO frequency offset of the original signal:

$$z[k] \approx A\exp(j(2\omega kPT_s + 2\phi)) + w[k] \tag{15}$$

where $w[k]$ is the noise term.

The estimation of the LO frequency offset turns to the estimation of a sine wave's frequency, which can be solved by several means. In our implementation, the frequency of $z[k]$ is estimated by evaluating the frequency that maximizes the amplitude of the DFT of $z[k]$:

$$\hat{f}_0' = \frac{1}{NPT_s}\arg\max_{-N/2\le m\le N/2}\{|Z[m]|\} \tag{16}$$

where $Z[m]$ is the DFT of $z[k]$, and N is the number of points of $z[k]$ used for the frequency estimation. This frequency estimate is further refined by using an interpolated estimate for our frequency estimate:

$$\hat{f}_1' = \frac{1}{NPT_s}\left[\hat{m} + I^+\frac{|Z[\hat{m}+1]|}{|Z[\hat{m}+1]| + |Z[\hat{m}]|} - (1 - I^+)\frac{|Z[\hat{m}-1]|}{|Z[\hat{m}-1]| + |Z[\hat{m}]|}\right] \tag{17}$$

where $\hat{m}$ is the index obtained in (16), and I^+ is defined as

$$I^+ = \begin{cases} 1, & \text{if } |Z[\hat{m}+1]| > |Z[\hat{m}-1]| \\ 0, & \text{if } |Z[\hat{m}+1]| < |Z[\hat{m}-1]| \end{cases} \tag{18}$$

The LO phase offset is then estimated as follows. If $s[k]$ is the sampled GMSK message sent by the transmitter over an AWGN channel (which contains the message header $t[k]$ of length N), the receiver will receive the following sampled message:

$$r[k] = \exp\left(j\left(\omega k T_s + \phi\right)\right) s[k] + v[k] \tag{19}$$

where $v[k]$ is the noise term. The receiver first estimates the LO frequency offset as described above, which will yield the estimate of the LO frequency offset $\hat{f}_1^{'}$. The receiver can then compensate the received message for the LO frequency offset as follows:

$$r'[k] = \exp\left(-j\left(2\pi \hat{f}_1^{'} k T_s\right)\right) r[k] \tag{20a}$$

$$\approx \exp\left(j\phi\right) s[k] + v'[k] \tag{20b}$$

where $v'[k]$ is the noise term. The receiver then correlates the frequency-compensated received signal r' with the (known) message header $t[k]$:

$$\gamma[k] = \sum_{n=1}^{N} r'[k-n]\, t[n] \tag{21}$$

The amplitude of $\gamma[k]$ will be maximum when $r'[k]$ is aligned with the message header. The phase offset is then also given by the phase of $\gamma[k]$:

$$k_0 = \arg\max\{(|\gamma[k]|\} \tag{22a}$$

$$\hat{\phi} = \angle\{\gamma[k_0]\} \tag{22b}$$

ACKNOWLEDGEMENTS

François Quitin acknowledges the financial support of the Belgian-American Educational Foundation. This work was supported in part by the Institute for Collaborative Biotechnologies through grant W911NF-09-0001 from the U.S. Army Research Office. The content of the information does not necessarily reflect the position or the policy of the Government, and no official endorsement should be inferred. This work is also funded in part by the US National Science Foundation under grants EPS1101284, ECCS-1150801 and a grant from the Roy J Carver Charitable Trust.

REFERENCES

[1] H. Ochiai, P. Mitran, H. Poor, and V. Tarokh, "Collaborative beamforming for distributed wireless ad hoc sensor networks," *Signal Processing, IEEE Transactions on*, vol. 53, no. 11, pp. 4110 – 4124, Nov. 2005.

[2] R. Mudumbai, G. Barriac, and U. Madhow, "On the feasibility of distributed beamforming in wireless networks," *Wireless Communications, IEEE Transactions on*, vol. 6, no. 5, pp. 1754 –1763, May 2007.

[3] R. Mudumbai, D. Brown, U. Madhow, and H. Poor, "Distributed transmit beamforming: challenges and recent progress," *Communications Magazine, IEEE*, vol. 47, no. 2, pp. 102 –110, February 2009.

[4] R. Mudumbai, J. Hespanha, and U. Madhow, "Scalable feedback control for distributed beamforming in sensor networks," in *Proc. 2005 IEEE International Symposium on Information Theory (ISIT 2005)*, 2005.

[5] T. Cover and A. El Gamal, "Capacity theorems for the relay channel," *IEEE Transactions on Information Theory*, vol. IT-25, no. 5, pp. 572–584, 1979.

[6] A. El Gamal and T. Cover, "Multiple user information theory," *Proceedings of the IEEE*, vol. 68, no. 12, pp. 1466–1483, 1980.

[7] S. Borade, L. Zheng, and R. Gallager, "Amplify-and-forward in wireless relay networks: Rate, diversity, and network size," *Information Theory, IEEE Transactions on*, vol. 53, no. 10, pp. 3302 –3318, Oct. 2007.

[8] S. Jin, M. McKay, C. Zhong, and K.-K. Wong, "Ergodic capacity analysis of amplify-and-forward MIMO dual-hop systems," *Information Theory, IEEE Transactions on*, vol. 56, no. 5, pp. 2204 –2224, May 2010.

[9] S. Gharan, A. Bayesteh, and A. Khandani, "Asymptotic analysis of amplify and forward relaying in a parallel MIMO relay network," *Information Theory, IEEE Transactions on*, vol. 57, no. 4, pp. 2070 –2082, April 2011.

[10] Y.-S. Tu and G. Pottie, "Coherent cooperative transmission from multiple adjacent antennas to a distant stationary antenna through AWGN channels," in *Vehicular Technology Conference, 2002. VTC Spring 2002. IEEE 55th*, vol. 1, 2002, pp. 130 – 134 vol.1.

[11] R. Mudumbai, B. Wild, U. Madhow, and K. Ramch, "Distributed beamforming using 1 bit feedback: From concept to realization," in *in Allerton Conference on Communication, Control, and Computing*, 2006.

[12] R. Mudumbai, J. Hespanha, U. Madhow, and G. Barriac, "Distributed transmit beamforming using feedback control," *Information Theory, IEEE Transactions on*, vol. 56, no. 1, pp. 411 –426, Jan. 2010.

[13] I. Ozil and D. Brown, "Time-slotted round-trip carrier synchronization," in *Signals, Systems and Computers, 2007. ACSSC 2007. Conference Record of the Forty-First Asilomar Conference on*, Nov. 2007, pp. 1781 –1785.

[14] D.R. Brown III and H. Poor, "Time-slotted round-trip carrier synchronization for distributed beamforming," *Signal Processing, IEEE Transactions on*, vol. 56, no. 11, pp. 5630 –5643, Nov. 2008.

[15] R. Preuss and D.R. Brown III, "Retrodirective distributed transmit beamforming with two-way source synchronization," in *Proc. of the 44th Annual Conference on Information Sciences and Systems (CISS 2010)*, 2010.

[16] ——, "Two-way synchronization for coordinated multicell retrodirective downlink beamforming," *Signal Processing, IEEE Transactions on*, vol. 59(11), pp. 5415–5427, 2011.

[17] R. Mudumbai, U. Madhow, D. Brown, and P. Bidigare, "DSP-centric algorithms for distributed transmit beamforming," in *IEEE Asilomar Conference on Signals, Systems and Computers, 2011*, 2011.

[18] R. Irmer, H. Droste, P. Marsch, M. Grieger, G. Fettweis, S. Brueck, H.-P. Mayer, L. Thiele, and V. Jungnickel, "Coordinated multipoint: Concepts, performance and field trial results," *Communications Magazine, IEEE*, vol. 49, no. 2, pp. 102–111, 2011.

[19] S. Munkyo, M. Rodwell, and U. Madhow, "A feedback-based distributed phased array technique and its application to 60-GHz wireless sensor network," in *Microwave Symposium Digest, 2008 IEEE MTT-S International*, June 2008, pp. 683 –686.

[20] S. Sigg and M. Beigl, "Algorithms for closed-loop feedback based distributed adaptive beamforming in wireless sensor networks," in *Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP), 2009 5th International Conference on*, Dec. 2009, pp. 25 –30.

[21] M. M. U. Rahman, H. E. Baidoo-Williams, S. Dasgupta, and R. Mudumbai, "Fully wireless implementation of distributed beamforming on a software-defined radio platform," in *the 11th ACM/IEEE Conference on Information Processing in Sensor Networks (IPSN 2012)*, 2012.

[22] F. Quitin, M. M. U. Rahman, R. Mudumbai, and U. Madhow, "Distributed beamforming with software-defined radios: frequency synchronization and digital feedback," in *to be presented at Globecom 2012*, 2012.

[23] F. Quitin, U. Madhow, M. M. U. Rahman, and R. Mudumbai, "Demonstrating distributed transmit beamforming with software-defined radios," in *Proc. of the Thirteenth International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM 2012)*, 2012.

[24] D. Rife and R. Boorstyn, "Single tone parameter estimation from discrete-time observations," *Information Theory, IEEE Transactions on*, vol. 20, no. 5, pp. 591 – 598, Sep 1974.

[25] S. Kay, "A fast and accurate single frequency estimator," *Acoustics, Speech and Signal Processing, IEEE Transactions on*, vol. 37, no. 12, pp. 1987 –1990, Dec 1989.

[26] D.R. Brown III, P. Bidigare, and U. Madhow, "Receiver-coordinated distributed transmit beamforming with kinematic tracking," in *Proceedings of the 37th International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2012, to appear.

[27] M. Morelli and U. Mengali, "Feedforward carrier frequency estimation with MSK-type signals," *Communications Letters, IEEE*, vol. 2, no. 8, pp. 235 –237, Aug. 1998.

[28] H. Peng, J. Li, and L. Ge, "Non-data-aided carrier frequency offset estimation of GMSK signals in burst mode transmission," in *Acoustics, Speech, and Signal Processing, 2003. Proceedings. (ICASSP '03). 2003 IEEE International Conference on*, vol. 4, April 2003, pp. IV – 576–9 vol.4.

[29] C. Zucca and P. Tavella, "The clock model and its relationship with the Allan and related variances," *Ultrason., Ferroelect, Freq. Contr., IEEE Tranasctions on*, vol. 52, no.2, pp. 289–296, 2005.

[30] USRP products, http://www.ettus.com, 2012.

[31] http://www.ece.ucsb.edu/~fquitin/.

FIGURES

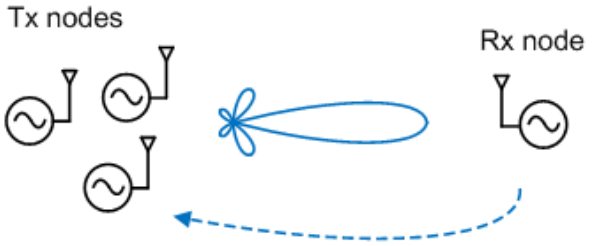

Fig. 1. Distributed beamforming general architecture

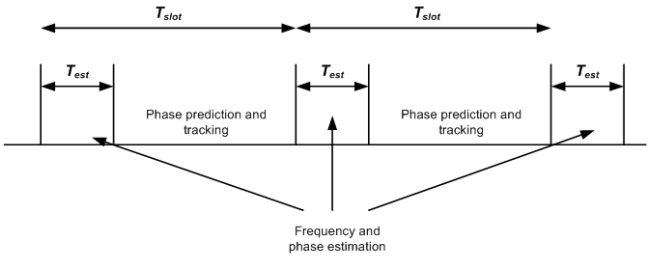

Fig. 2. Time-slotting model for frequency and phase estimation.

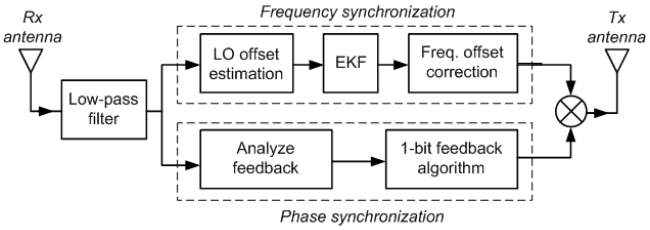

Fig. 3. Block diagram of a transmit node

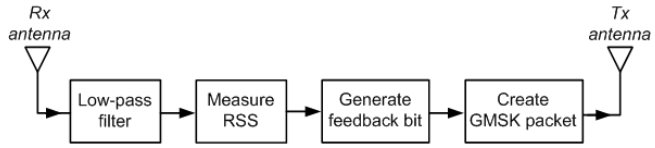

Fig. 4. Block diagram of a receive node

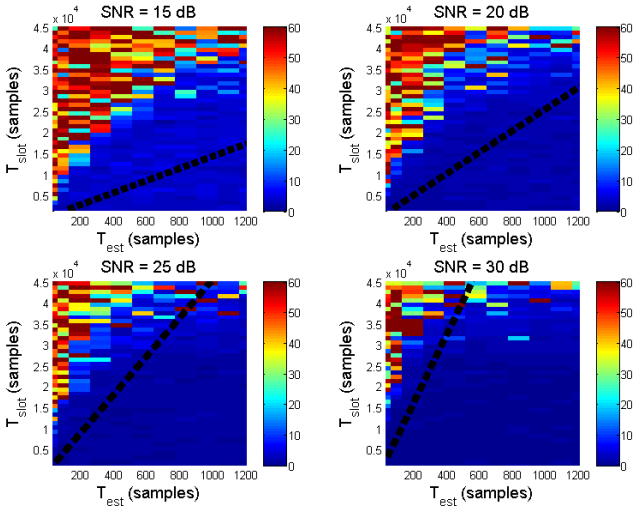

Fig. 5. Simulated average phase error of the EKF for various values of T_{est} and T_{slot}. The colors intensity represents the phase error (in $^\circ$), and the black line represents the CRLB.

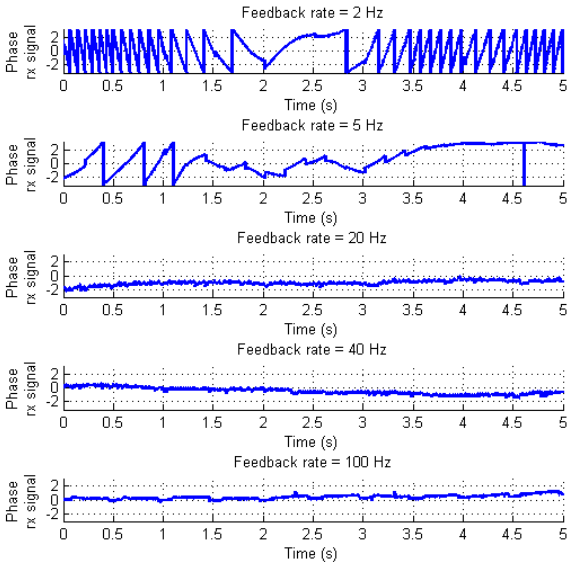

Fig. 6. Phase of the received signal as a function of feedback rate.

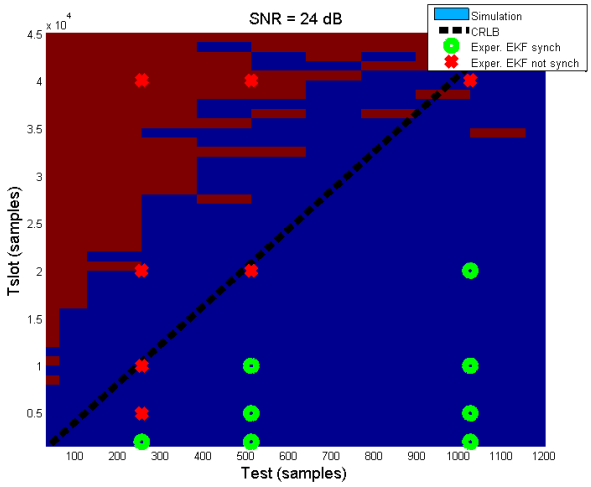

Fig. 7. EKF convergence/divergence as function of T_{est} and T_{slot}, for simulations and measurements. The blue/red area represents the $\{T_{slot}, T_{est}\}$-pairs for which the EKF converges/diverges with simulations, the circles/crosses represents the $\{T_{slot}, T_{est}\}$-pairs for which the EKF converges/diverges with the experimental setup, and the black line show the CRLB.

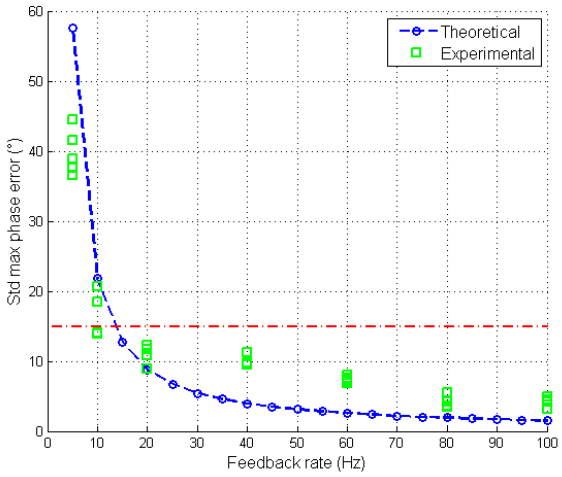

Fig. 8. Standard deviation of the maximum phase error as a function of $1/T_{slot}$, both theoretically and experimentally. The dash-dotted line represents the 15° treshold below which 95% of the maximum beamforming gain is achieved.

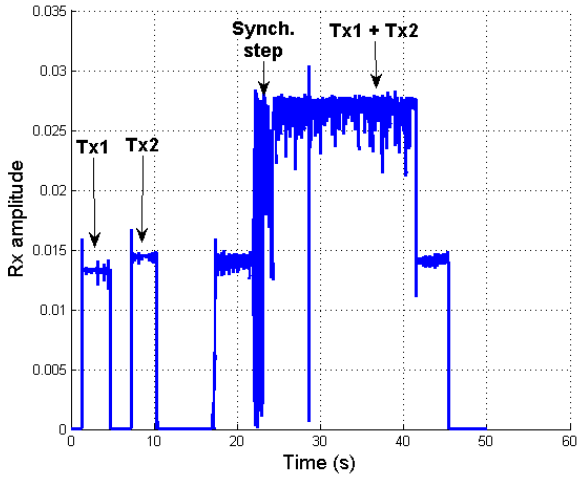

Fig. 9. Received power when two transmit nodes are beamforming. The first two bursts are the two individual nodes transmitting.

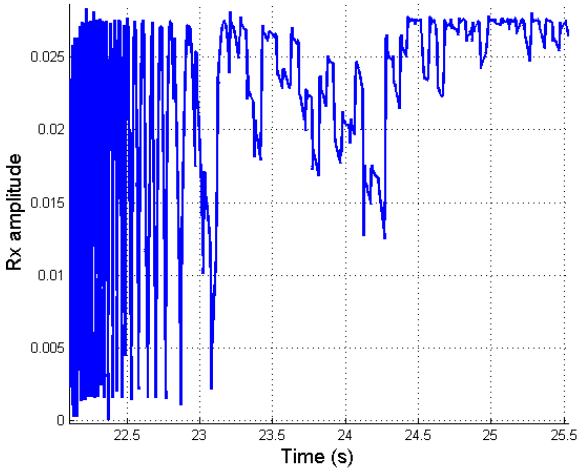

Fig. 10. Synchronization of the two transmit nodes. At first, the two nodes are synchronizing their frequency. Then, the phase of the two nodes is adapted so that their signals add up.

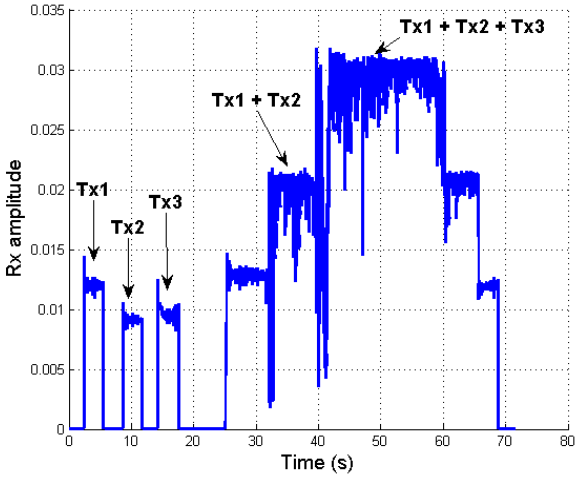

Fig. 11. Received power when three transmit nodes are beamforming.