Demonstrating distributed transmit beamforming with software-defined radios

François Quitin, Upamanyu Madhow
Electrical and Computer Engineering
University of California, Santa Barbara
{fquitin, madhow}@ece.ucsb.edu

Muhammad Mahboob Ur Rahman, Raghuraman Mudumbai
Electrical and Computer Engineering
University of Iowa
mahboob-rahman@uiowa.edu
rmudumbai@engineering.uiowa.edu

***Abstract*—We present a fully wireless implementation of distributed transmit beamforming using software-defined radios. Distributed beamforming is a cooperative transmission scheme whereby a number of nodes in a wireless network organize themselves into a *virtual antenna array* and focus their transmission in the direction of the intended receiver, potentially achieving order of magnitude improvements in energy efficiency. The main technical challenge in realizing these gains is in precisely synchronizing the radio frequency signals of the cooperating nodes. This idea has been studied extensively over the past decade, and several techniques and architectures for its practical implementation have been developed. In this work, we demonstrate our recent implementation of distributed beamforming on a standard, open-source, software-defined radio platform, where the low quality oscillators make synchronization particularly challenging. Our demonstration will consist of three cooperating transmitters sending signals that add up constructively at the receiver. Low-rate feedback packets broadcast from the receiver are employed for frequency and phase synchronization at each transmitter in completely distributed fashion.**

***Keywords*-Wireless systems, software-defined radios, distributed beamforming**

I. Introduction

Distributed transmit beamforming consists of a number of transmit nodes sending a common message such that their signals add up at the receiver. Under this scheme, the transmitters cooperatively organize themselves into a *virtual antenna array* and focus their transmission in the direction of the intended receiver [1–3]. This scheme allows small nodes with simple omni-directional antenna to cooperatively emulate a large highly directional antenna array. This potentially offers large increases in energy efficiency: a N-node beamforming array can achieve the same received signal strength (RSS) at the receiver with as little as $\frac{1}{N}$ of the *total transmit power* required by a single node transmitting individually. On the other hand, if each transmitter transmits at a fixed power, the received power scales as N^2, thus enabling large increases in attainable range.

Background. While early work on cooperative communication did not focus on synchronization issues, this changed in the last decade, and a number of synchronization techniques for distributed beamforming have now been developed. Different techniques for synchronization have been developed featuring different sets of tradeoffs between simplicity, overheads associated with coordination messages between the transmitters, and overheads associated with feedback from the receiver.; see [3] for a survey, and [4] for a discussion of more recent work. There have also been several recent experimental studies of distributed beamforming [3]; however, all previous experimental demonstrations of beamforming use wired side-channels to synchronize the clocks. Our implementation is the first ever all-wireless demonstration of distributed beamforming, and represents a significant advance in bringing this technique to real-world wireless networks.

Contributions. In this work, we present an architecture for distributed transmit beamforming, and demonstrate an all-wireless implementation on software-defined radios (SDRs) using the USRP-2 [5] platform. The prototype presented in this work is an evolution of the architecture presented in [6]. It features a novel Extended Kalman Filter (EKF) that estimates and corrects for frequency offsets between the RF signals at each transmitter, and a beamforming algorithm that steers the transmission of the virtual array to a receiver using a simple 1-bit feedback algorithm [3].

The RF oscillators that come standard with the USRP hardware are of relative low quality, and these oscillators can have frequency offsets as large as several kHz, with significant drifts over time. The EKF is powerful enough to effectively synchronize these inexpensive oscillators using a minimal amount of feedback: one 40-byte packet transmitted every 50 milliseconds. The same feedback packet also contains the 1 bit of feedback required by the 1-bit beamsteering algorithm, hence the whole process incurs very little network overhead.

Brief description of proposed demo. We propose a demonstration of the distributed beamforming prototype. In the demonstration, we will show three transmit nodes cooperatively transmitting to a receiver. Each of the transmit nodes can be turned on and off, and the one-bit feedback algorithm that takes care of phase alignment can be enabled or disabled independently at each of the transmit nodes. The receiver has a real-time display that shows the total received power. It can also be configured to display the signal waveform or the signal spectrum in a software oscilloscope or FFT window. It will be shown that when the distributed beamforming is enabled, the total received power is substantially greater than the sum of powers of the cooperating nodes transmitting individually. Specifically, if the receiver sees a signal power P from each of the 3 cooperating nodes individually, this power increases to almost $9 \times P$ with beamforming.

II. Distributed beamforming architecture

A. General architecture

The general architecture of the distributed beamforming prototype is given in Figure 1. Three transmitters want to co-operate to communicate, thereby creating a beam towards the receiver. Two types of synchronization between the transmit nodes must be achieved for the transmitted messages to add up coherently at the receiver: frequency synchronization and phase synchronization (see block diagram of the transmitter in Fig. 2). The receiver broadcasts a feedback message to the transmitters, that is used for both frequency and phase synchronization. Note that the beamformed message and the feedback message use two separate frequencies: the beamformed signal is at 892 MHz and the feedback signal is at 964 MHz.

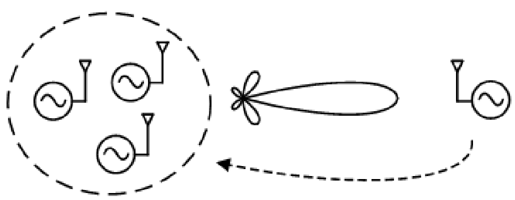

Figure 1. Distributed beamforming general architecture

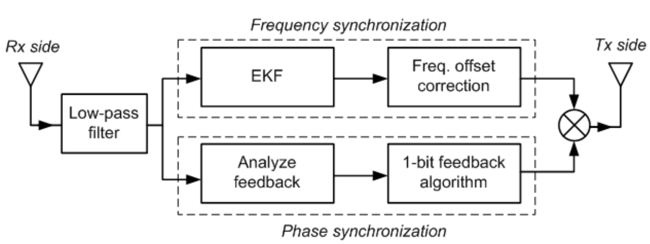

Figure 2. Block diagram of a transmitter node

B. EKF frequency synchronization

The EKF approach presented here aims at tracking the LO frequency and phase offset by using periodical LO frequency and phase offset measurements. These measurements are fed into an EKF, which then yields filtered estimates for the LO frequency and phase offset. These filtered estimates are then used by each transmit node to compensate for its LO offset. The LO frequency and phase measurements are obtained from the periodic feedback message that is sent by the receiver.

C. One-bit feedback phase synchronization

Phase synchronization is achieved using the one-bit feedback algorithm. This algorithm tries to align the phases of the transmitted messages at the receiver for coherent combining. The one-bit feedback algorithm works as follows: at each time slot, each transmit node adds a random phase perturbation to its current phase. The receiver estimates the received signal strength (RSS), and broadcasts a single bit of feedback to all the transmitters indicating whether the estimated RSS is higher or lower than in the previous time slot. If the RSS is higher, the transmit nodes keep their last phase perturbation, and go on to the next time slot. If the RSS is lower, the transmit nodes return their phase to the one of the previous time slot, before going on to the next time slot.

III. SDR distributed beamforming prototype

The nodes used in our experiments are USRP RF and baseband boards [5], a popular SDR platform. For this implementation, USRP-2 nodes were used with a WBX RF daughterboard. All the signal processing is performed in real-time on laptops using the GNU Radio framework [7].

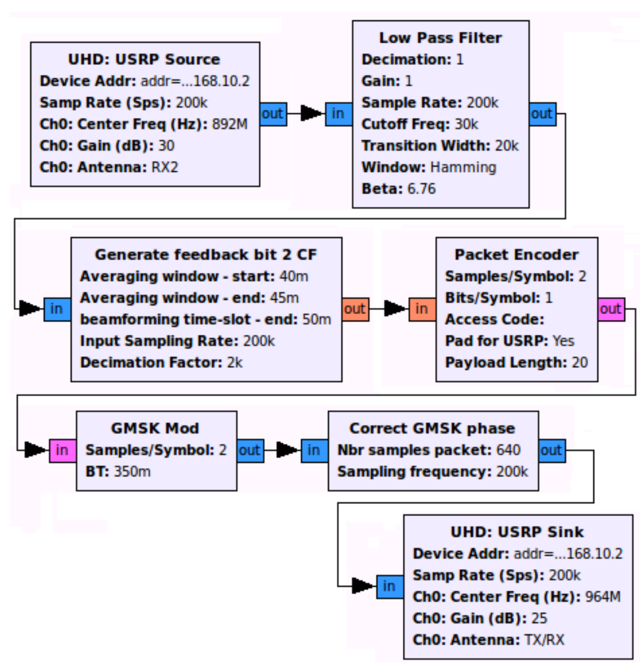

Figure 3. GNU Radio flowgraph of the receiver node.

The demonstration has three transmit nodes and one receive node. Each node is controlled by a laptop with the

GNU Radio software, where all the blocks of the nodes can be identified, as shown in Figure 3. Each of the transmit nodes can be turned on and off, and the one-bit feedback algorithm can be enabled or disabled in real-time. At the receiver, the real-time display shown in Figure 4 permits to view to total received power. During the demonstration,

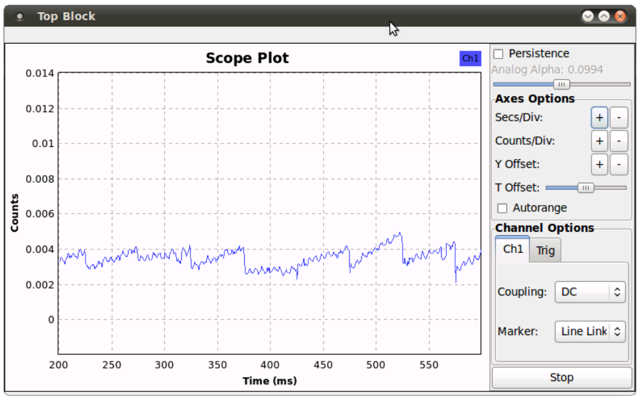

Figure 4. Receiver node real-time display. The curve shows the received amplitude as a function of time.

attendees will be able to run the distributed beamforming prototype, enabling and disabling nodes, and enabling and disabling the one-bit feedback algorithm for each node. The resulting received power will be displayed in real-time with the GNU Radio software.

It can be seen in Figure 5 that when nodes use the previously presented architecture, the amplitude of the received signal is close to the sum of the amplitudes of the received signals from the individual transmit nodes, which implies that the received power when the 3 nodes are transmitting together is close to 9 times greater than the power received from each individual transmitter.

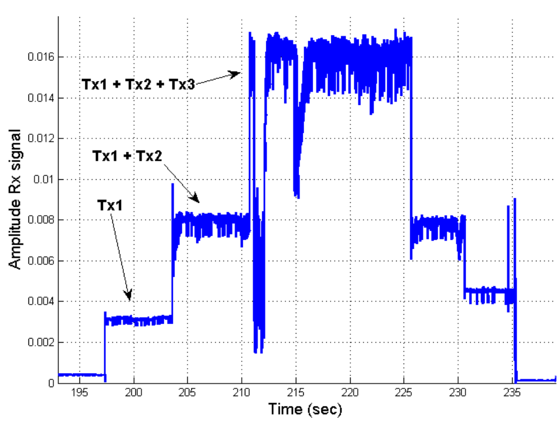

Figure 5. Received amplitude when the transmitters are turned on one by one, and then turned off one by one. The one-bit feedback algorithm is enabled at all times.

The system sometimes suffers from software or hardware latencies. In these cases, the filtered frequency estimate is corrupted or the feedback arrives too late for the transmitters to use it. With the presented architecture, the system is able to recover from such errors, as can be seen in Figure 5, around 215 s. However, since the EKF needs several cycles to recover from an erroneous measurement, it is important that the rate of software or hardware lags is low enough for the system to recover in between lags.

IV. Conclusion

This work demonstrates an implementation of a distributed transmit beamforming prototype on software-defined radios. Two problems of distributed beamforming are addressed: frequency synchronization and phase synchronization. Frequency synchronization is achieved by implementing an EKF at each of the transmit nodes. The feedback message is used to make a measurement of the LO frequency and phase offset, which are used as an input for the EKF. Phase synchronization is attained by using the one-bit feedback algorithm. The feedback message contains a single bit of information that is used by the transmit nodes to adapt their phase perturbation. An implementation of this architecture on software-defined radios shows that the expected gains are achieved: with three transmit nodes, the RSS is the sum of the RSS of the individual nodes, meaning and order-of-magnitude increase in received power.

Acknowledgment

François Quitin acknowledges the financial support of the Belgian-American Educational Foundation. This material is based upon work supported by the Institute for Collaborative Biotechnologies through grant W911NF-09-D-0001 from the U.S. Army Research Office, a grant from the Roy J. Carver Charitable Trust, and by the National Science Foundation under grants CNS-0832154 and EPS1101284.

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, 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.

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

[6] 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.

[7] GNU radio, http://gnuradio.org/redmine/projects/gnuradio/wiki, 2012.