MIMORPH: A General-Purpose Experimentation Platform for sub-6 GHz and mmWave Frequency Bands

Jesus O. Lacruz, Rafael Ruiz and Joerg Widmer
IMDEA Networks Institute, Madrid, Spain

Paper excerpted from “A Real-Time Experimentation Platform for sub-6 GHz and Millimeter-Wave MIMO Systems,” from *Proceedings of the 19th International Conference on Mobile Systems, Applications, and Services (MobiSys ’21)*, with permission. https://dl.acm.org/doi/10.1145/3458864.3466868 © ACM 2021

With the rapid increase in performance and complexity of wireless networks over the past years, it has become challenging to build experimentation platforms that can meet such performance requirements but at the same time are comparatively easy to use and flexible. The lack of suitable platforms inspired us to build MIMORPH, a single experimentation platform that supports massive MIMO sub-6 GHz systems, ultra-high bandwidth Millimeter Wave (mmWave) MIMO, as well as mixed sub-6 GHz and mmWave configurations. It can be operated in a *closed-loop* manner and is intended for WLAN, 5G-NR and future 6G research. MIMORPH is built on top of standard components such as a state-of-the-art RFSoC FPGA system and its implementation is made freely available to the research community.

INTRODUCTION

New applications such as connected vehicles, virtual/augmented reality, remote healthcare, and industry automation impose different sets of stringent requirements on physical layer design. Current specifications of WLAN and mobile networks have bandwidth requirements ranging from tens of MHz for sub-6 GHz to several GHz for mmWave frequencies, support MIMO with a large number spatial streams, digital and analog beamforming, and high-order modulation and coding schemes. This makes it challenging for researchers in academia and industry to keep pace with such dynamic requirements. There have been substantial efforts over the past years to design experimentation systems that address a specific part of the design space, such as bespoke sub-6 GHz narrow-band massive MIMO systems or mmWave SISO systems. For sub-6 GHz frequencies, USRP software defined radios are widely used thanks to their flexible design, extended online community and ease of use [1,2]. While USRP-based mmWave systems have also been proposed [3], their usefulness is limited due to the low bandwidth they support. Several existing mmWave testbed designs are based on more powerful hardware to overcome this limitation [4-6]. For example, the mm-FLEX platform [4] meets the multi-GHz bandwidth requirements of current mmWave standards.

Currently, no single testbed platform supports both sub-6 GHz and mmWave frequencies while providing the very large bandwidth and high-order MIMO required for current and future wireless networking research (e.g., IEEE 802.11ac/ax/be/ad/ay, 5G-NR and 6G).

Given the high complexity of supporting several bespoke platforms, we take the lessons learned from the mm-FLEX design to build a single high-performance system that fulfills these requirements. Our MIMORPH experimentation system is built on the state-of-the-art Xilinx RFSoC platform ZCU111 which integrates FPGA logic, multiple Giga-sampling AD/DA converters and ARM processors. Only a few external components such as amplifiers and antennas or mmWave RF front ends are needed to complete the system, making it more accessible, affordable and easy to deploy than prior solutions [4,5].

The MIMORPH architecture follows a “memory-based” design as shown in Fig. 1, i.e., the I/Q samples are generated offline, go from memory to the DACs, are transmitted, and then at the receiver side are sampled by the ADCs and again stored in memory. This way, experimenting with different waveforms and signal processing algorithms requires no implementation changes. We use standard interfaces (AXI) in the data-paths for a flexible and modular design which allows components to be implemented in hardware or software. If needed, hardware accelerators can be integrated in the AXI chain for time critical functions. For example, for mmWave experimentation we integrate hardware accelerators for packet detection, synchronization, channel estimation capabilities as well as fast antenna reconfiguration.

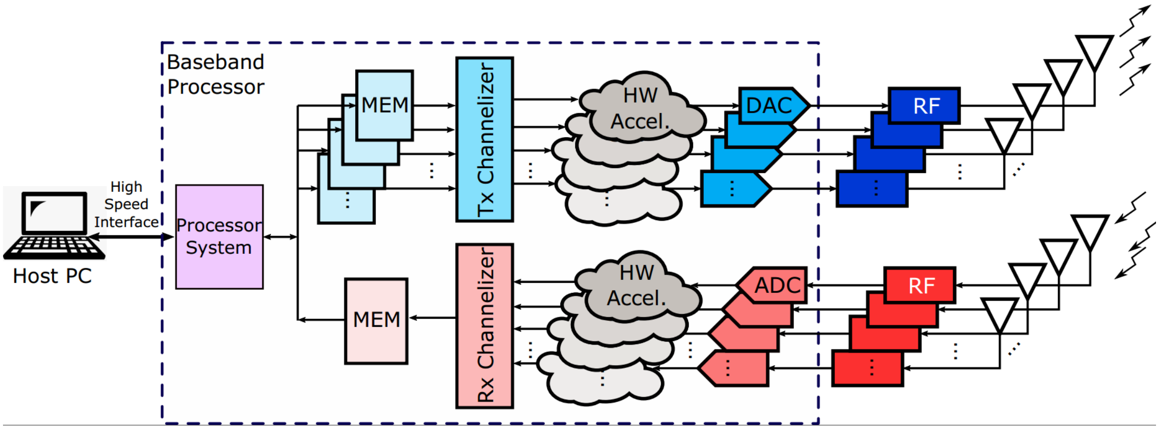

Fig. 1: MIMORPH top-level block diagram

MEMORY-BASED DESIGN

Addressing the diverse requirements of such a multi-band platform is challenging. We build MIMORPH with a “fixed” structure for memories, controllers, and buses together with configurable blocks that can *morph* the functionality of the system according to the requirements in terms of bandwidth and number of data paths. This avoids the cumbersome process of a full redesign for each configuration.

In its simplest form, we use loopback memories[1] for the transmitter datapath to store the frames that will be transmitted. These memories are implemented using the block RAM on the FPGA logic. At the receiver we use the on-board DDR memory to store the received samples. Note that while we use a separate loopback memory per channel at the transmitter,

---

[1] FIFO queues with feedback logic that act as circular buffers

the DDR is shared between the different received streams. Despite the different bandwidth configurations and number of spatial streams, the memory partitioning and mapping as well as the connections with the processor are fixed, which simplifies modifying the functionality while keeping the read/write part of the design unaltered.

One important consideration for a multi-band system with such a structure is the datapath width. While common FPGA clock frequencies allow processing of a single I/Q sample per clock cycle for narrowband systems, ultra wideband mmWave channels require processing multiple samples in parallel in a single clock cycle, called Super Sampling Rate (SSR). This allows processing I/Q samples at Giga sampling rates with FPGAs having clock frequencies of only hundreds of MHz. To interface the read/write structure with the AD/DA converters, we use channelizers (Fig. 1) that allow the system to *handle different SSR factors in a seamless manner*. Furthermore, the channelizers allow separation of the read/write blocks and the possible hardware accelerators used in the design.

At the transmitter side, the channelizer works as follows. The loopback memory is designed for the highest SSR factor of 16, i.e., the largest bandwidth configuration. Then, the channelizer distributes the samples from the memory to the DACs in order to match the desired bandwidth of the transmitted signal. Depending on the bandwidth requirements per DAC, the number of *active* loopback memories changes. This way, changing from one configuration to another only requires modifying the channelizer's internal configuration and how it distributes the samples, while keeping the memories and their connections unaltered. On the receiver side the process is analogous. The channelizer groups the samples from the ADCs in order to match the maximum SSR factor (16) of the DDR controller for maximum DDR write efficiency. Since the DDR is a shared resource, the samples are grouped in channels and then fed to the DDR in a *sequential* manner.

By selecting the channelizer and AD/DA converter configurations, it is possible to set up a 4x4 mmWave MIMO system with up to 2 GHz of bandwidth per stream, 8x8 sub-6 GHz streams or *mixed* multiband configurations, depending on the user's need. It is not mandatory to implement hardware accelerators in the FPGA logic in order to use MIMORPH in its simplest form. The tight synchronization between different streams makes MIMORPH ideal not only for MIMO communication purposes but also for environmental sensing where timing is crucial to accurately estimate the channel. Furthermore, since receiver and transmitter chains are fully independent, it is possible to use the system in a full-duplex manner for radar-like sensing.

One important aspect of a memory-based design is to not exceed the write speed limits of the single shared DDR memory. For example, for 4x4 MIMO with 2 GHz of bandwidth per stream and 16-bit samples, the raw throughput is 450 Gbps, far exceeding the DDR write speed of ~150 Gbps. In [8], we address this with two different approaches. i) We add a state machine for *inter-frame spacing* at the transmitter which can reduce the net communication throughput while preserving the signal quality. ii) We add processing blocks that *reduce the bit resolution* of the received I/Q samples and then concatenate them in a way that is compatible with the read/write structures. This approach preserves data throughput, but the loss in signal quality may impact the bit error rate (BER). This configuration is also useful to experiment with low-resolution signal processing algorithms which are an important topic in mmWave research. Both approaches can be combined to trade off signal quality and throughput. Fig. 2 shows a throughput / BER comparison for a IEEE 802.11ay 4x4 MIMO mmWave system for 16-bit and 5-bit samples. The full set of results can be found on [8].

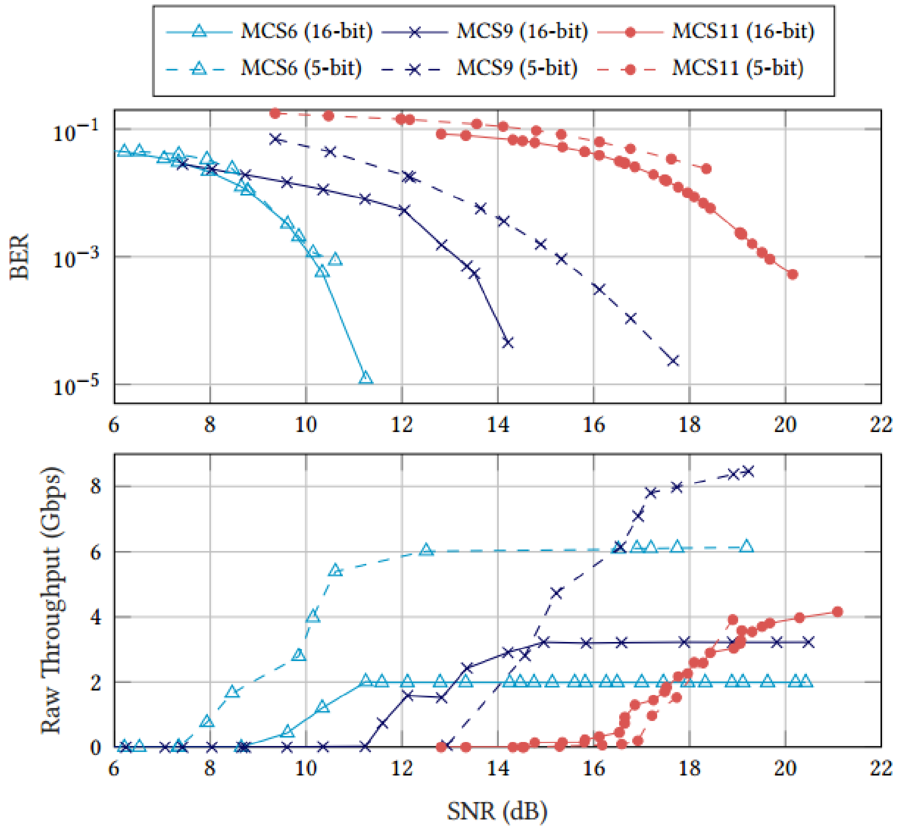

Fig.2: BER / throughput comparison for 16-bit and 5-bit samples for different modulation schemes.

For easy configuration and use of the testbed, we develop MATLAB functions to start the system, and send and receive I/Q samples for different configurations. Those functions as well as the different firmware projects are available as open-source in [7] and further details about the architecture of MIMORPH can be found in [8].

MULTI-BAND SETUP

One important advantage of MIMORPH is the multiband capability, which even allows the simultaneous operation of different streams at different frequency bands and with different bandwidths. Using the features of the AD/DA converters available in the RFSoC, it is possible to generate RF signals covering the whole sub-6 GHz frequency band, and for mmWave, our testbed can be used with different front-ends (e.g., 24-28 GHz for 5G-NR or 60 GHz for IEEE 9-2.11ad/ay).

We showcase this by setting up the system with 4 spatial streams at 2.4 GHz with 160 MHz of bandwidth each for IEEE 802.11ax frames and 2 streams at 60 GHz with 1.76 GHz of bandwidth each for IEEE 802.11ay frames. We perform over-the-air experiments in an indoor scenario, capturing and saving the samples in memory and then using MATLAB functions to process preamble, header and data fields. Fig. 3 shows a photo of a MIMORPH multiband node used for the experiments as well as the throughput results for the IEEE 802.11ax streams. We omitted the results of the mmWave streams since those are similar to the ones shown in Fig. 2. The full set of results can be found in [8]. Given that both mobile network and WLAN standards support simultaneous operation at different frequencies, we believe such configurations to be highly useful for multiband experimentation.

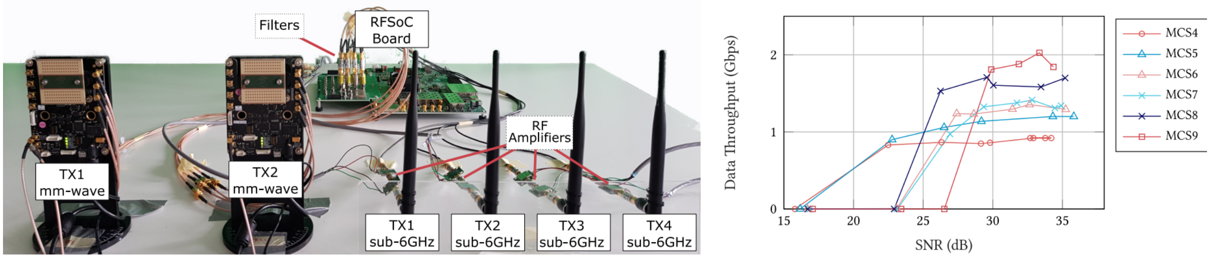

Fig. 3: MIMORPH multiband node (left) and 4x4 MIMO IEEE 802.11ax throughput results (right).

CLOSED LOOP OPERATION

In addition to the memory-based system, MIMORPH supports the implementation of real-time functionality as needed. To showcase this, we briefly introduce how to add hardware accelerators for real-time IEEE 802.11ay MIMO beam tracking. Specifically, we implement *in-packet* training [9], which allows to align both communicating nodes using a single packet, by appending training fields with complementary Golay sequences to test multiple transmit and receive beam patterns. Specifically, i) node *A* sends a packet with N_A+N_B training fields. *A* sweeps through N_A transmit beam patterns while the N_A training fields are being sent. ii) Node *B* detects the packets from *A* and estimates the channel in real-time for each training field. iii) *B* then sweeps through its own beam patterns for the remaining N_B training fields of the received packet and estimates the channel. iv) *B* then collects the channel estimates to determine the best transmit-receive beam pattern pair for *A* and *B*. v) *B* selects the best receive beam patterns and notifies *A* by means of a feedback frame of the best transmit beam patterns. vi) *A* receives the feedback packet from *B* and changes the transmit beam patterns for the subsequent packets.

Implementing this functionality requires hardware accelerators for real-time packet detection, synchronization, channel estimation, antenna reconfiguration, and selection of the best beam pattern combination for all streams. The beam pattern changes in both transmitter and receiver nodes are implemented using GPIO pulses [4]. Besides, we implement SPI controllers to alternatively change from transmitter to receiver mode and vice versa on each antenna, as required for a real-time closed-loop system. To collect the channel measurements, we implemented packet detection, synchronization (boundary detection) and channel estimation accelerator blocks on the FPGA logic. Details and code for the block architectures are in [7,8]. For simplicity, we send a 6-bit feedback frame per stream using 6 complementary Golay sequences, which allows us to reuse the channel estimation block to decode the message, rather than having to implement full packet decoding.

We validate the real-time training with two fully independent MIMORPH nodes, with one of them subject to lateral movement as shown in Fig. 4. Furthermore, we show on the top-right the estimated angle of departure (AoD) and arrival (AoA). In the bottom-right graph we show the received power for each of the possible beam patterns in light colours and the power of the chosen beam pattern pairs in dark blue. As can be seen, our implementation *always* selects the correct combination of beam patterns.

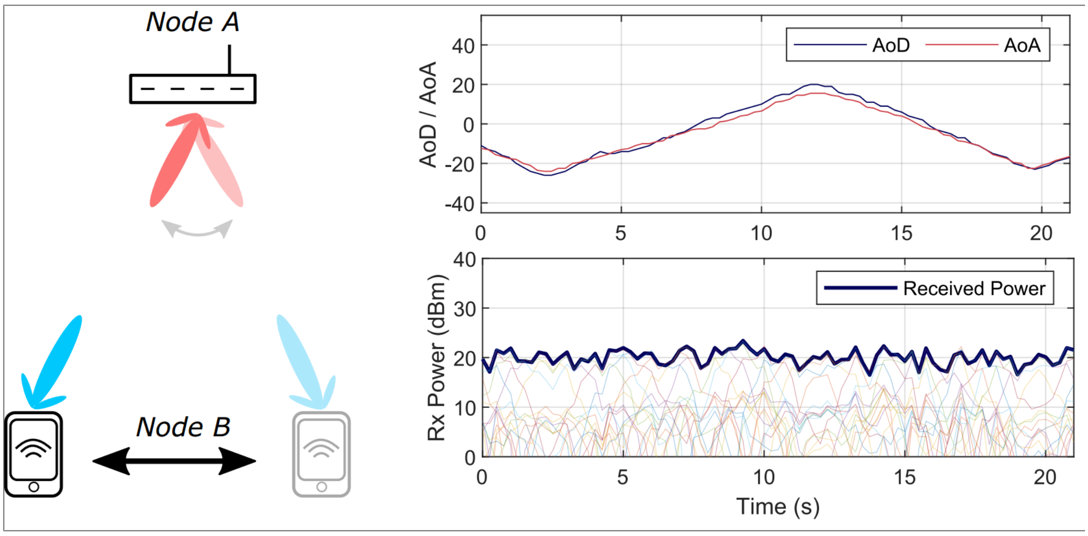

Fig. 4: Real-time mmWave MIMO beam tracking implementation.

CONCLUDING REMARKS

With MIMORPH, we extend high performance wireless experimentation to multiple spatial streams and multiband operation. We hope our platform will serve researchers in academia and industry for a wide variety of applications. To this end, we provide it as an open-source project for the research community [7] to use and extend its functionality as needed. We continue to actively extend the platform and our next steps on the roadmap include hardware accelerators for 5G/6G to develop a standard compliant system that supports real-time packet encoding and decoding, in addition to memory-based operation.

ACKNOWLEDGEMENTS

This research work was sponsored in part by the European Union's Horizon 2020 research and innovation programme under Grant No. 871249 (LOCUS), and in part by the Spanish Ministry of Science and Innovation (MICIU) grant RTI2018-094313-B-I00 (PinPoint5G+) and the Region of Madrid through TAPIR-CM (S2018/TCS-4496).

REFERENCES

[1] S. Malkowsky *et al.*, "The World's First Real-Time Testbed for Massive MIMO: Design, Implementation, and Validation," in *IEEE Access*, vol. 5, pp. 9073-9088, 2017, doi: 10.1109/ACCESS.2017.2705561.

[2] I. Gomez *et al.*, 2016. SrsLTE: an open-source platform for LTE evolution and experimentation. In *Proceedings of* ACM *WiNTECH '16*. New York, NY, USA

[3] A. Moreno, J. Lacruz, and J. Widmer. 2020. Open Source RFNoC Based Testbed for Millimeter-Wave Experimentation Using USRP Software Defined Radios. In Proceedings IEEE ISCAS-2020

[4] J. Lacruz *et al*., 2020. Mm-FLEX: an open platform for millimeter-wave mobile full-bandwidth experimentation. In Proceedings ACM MobiSys '20, New York, NY, USA

[5] S. Kumar, *et al*. 2019. X60: A programmable testbed for wideband 60 GHz WLANs with phased arrays. Computer Communications 133 (2019), 77–88.

[6] Renjie Zhao, *et-al*. 2020. M-Cube: a millimeter-wave massive MIMO software radio. *Proceedings of ACM MobiCom '20*, New York, NY, USA.

[7] R. Ruiz, J. Lacruz and J. Widmer. MIMORPH: A real-time platform for sub-6 GHz and mmWave MIMO systems. https://softwarewng.networks.imdea.org/mimorph/

[8] J. Lacruz, R. Ruiz and J. Widmer. 2021. A real-time experimentation platform for sub-6 GHz and millimeter-wave MIMO systems. In *Proceedings of MobiSys '21*, New York, NY, USA.

[9] Y. Ghasempour, *et-al*. IEEE 802.11ay: Next - Generation 60 GHz Communication for 100 Gb/s Wi-Fi. IEEE Communications Magazine, 55(12):186–192, 2017

BIOGRAPHIES

Jesus O. Lacruz is a Research Engineer at IMDEA Networks Institute, Spain. He received his PhD in Electronic Engineering from Universidad Politécnica de Valencia, Spain. His research interests lie in the design and implementation of fast signal processing algorithms for digital communication systems in FPGA devices.

Rafael Ruiz received his MSc in Electronic Engineering from Universidad Politécnica de Madrid. He is a Research Engineer at IMDEA Networks Institute. His main research interests are focused on digital embedded system design and the implementation of accelerators based on FPGA devices.

Joerg Widmer is the Research Professor and Research Director of IMDEA Networks in Madrid, Spain. His research focuses on wireless networks, in particular millimeter-wave communications. He is an IEEE Fellow and Distinguished Member of the ACM, and was awarded an ERC consolidator grant, the Friedrich Wilhelm Bessel Award of the Alexander von Humboldt Foundation, a Mercator Fellowship of the German Research Foundation, and a Spanish Ramon y Cajal grant.