Matryoshka: Single RF Chain Multi-user Transmission through WiFi-in-WiFi Signal Emulation using COTS Hardware

Piotr Gawłowicz
School of Electrical Engineering
and Computer Science
TU Berlin, Germany
gawlowicz@tkn.tu-berlin.de

Anatolij Zubow
School of Electrical Engineering
and Computer Science
TU Berlin, Germany
zubow@tkn.tu-berlin.de

Falko Dressler
School of Electrical Engineering
and Computer Science
TU Berlin, Germany
dressler@tkn.tu-berlin.de

ABSTRACT

We see a trend toward serving Internet of Things (IoT) devices using the IEEE 802.11 protocol to offer cost-effective solutions. A WiFi AP serving simultaneously both broadband and IoT applications suffers from performance degradation as the slow IoT devices operating on a low modulation and coding throttle down the high-speed broadband devices. In this paper, we present Matryoshka, an approach that exploits the signal emulation technique developed in the context of cross-technology communication (CTC) to create multi-user transmissions from a SISO WiFi AP in the downlink. With such emulated multi-user transmissions, which is a form of hierarchical modulation, it is possible to simultaneously serve a high-speed 802.11 station together with a slow-speed station, which is more efficient than serving them one after another in the time domain. Our approach is a software solution and works with commodity WiFi hardware (COTS). Experimental results from our prototype show the practical feasibility.

ACM Reference Format:
Piotr Gawłowicz, Anatolij Zubow, and Falko Dressler. 2023. Matryoshka: Single RF Chain Multi-user Transmission through WiFi-in-WiFi Signal Emulation using COTS Hardware. In *The 17th ACM Workshop on Wireless Network Testbeds, Experimental evaluation & Characterization 2023 (ACM WiNTECH' 23), October 6, 2023, Madrid, Spain.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3615453.3616508

ACM WiNTECH' 23, October 6, 2023, Madrid, Spain

ACM ISBN 979-8-4007-0340-9/23/10…$15.00
https://doi.org/10.1145/3615453.3616508

1 INTRODUCTION

Today, IEEE 802.11 Wi-Fi holds a dominant position in providing wireless broadband Internet access [5]. In addition, we see a constant growth in the number of connected devices forming the Internet of Things (IoT) idea. Such IoT nodes are often constrained devices (e.g., battery-powered) that communicate wirelessly at very low data rates with small packet sizes and having reduced functionality, i.e., low order Modulation Coding Scheme (MCS). At the same time, there is a trend towards serving such IoT devices using the standard Wi-Fi protocol in order to offer cost-effective solutions. However, serving both broadband applications like video streaming and IoT exclusively with Wi-Fi may cause problems as the slow IoT devices (optimized for power consumption and not data rate) might throttle down high-speed broadband Wi-Fi devices like TVs, laptops, or smartphones as they share the same radio spectrum.

In this paper, we propose Matryoshka,[1] an approach that exploits the signal emulation technique developed in the context of cross-technology communication (CTC) [3] to create multi-user Wi-Fi transmissions in the downlink (DL) (Fig. 1). This allows a Wi-Fi AP equipped with even only a single antenna and RF chain, i.e., a Single Input Single Output (SISO) system, to simultaneously serve a high-speed broadband station together with a slow-speed IoT Wi-Fi station which is more efficient than serving them one after another in the time domain. Such WiFi-in-WiFi transmission is a form of hierarchical modulation (HM) (see., e.g., [2]). In

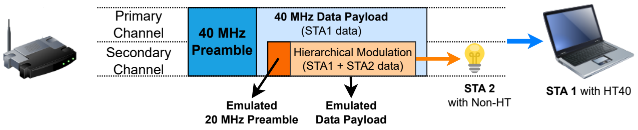


Figure 1: Matryoshka uses CTC signal emulation technique enabling a SISO Wi-Fi AP to create multi-user transmissions (Wi-Fi-in-Wi-Fi frame) in the DL.

[1] Matryoshka dolls are a set of wooden dolls of decreasing size placed one inside another.

that sense, Matryoshka is utilizing CTC to create a virtual multi-user transmission on top of a SISO system. Note, the difference to classical multi-user transmission which requires a transmitter equipped with multiple antennas and RF chains, i.e., Multiple Input, Multiple Output (MIMO) system. Our approach is a pure software solution and works with COTS IEEE 802.11n/ac SISO hardware. Experiments with a real prototype reveal that Matryoshka is able to emulate a valid 20 MHz Wi-Fi frame with PHY preamble and header carrying IoT data inside a 40 MHz frame whereas the former uses a very low MCS (i.e., BPSK) which is sufficient to serve low-data rate IoT devices. In contrast, the outer Wi-Fi frame serving a broadband user can be transmitted with high MCS (i.e., 64-QAM) and thus a high data rate.

Contributions: We propose Matryoshka, a generic software-only solution enabling multi-user multiplexing technique for the DL using commodity 802.11n/ac Wi-Fi SISO systems. The technique is based on the HM concept and is achieved utilizing the QAM/OFDM-based signal emulation technique proposed in the context of cross-technology communication [8]. Specifically, Matryoshka does not require hardware modification on the Wi-Fi AP side, instead, it designs a DL Wi-Fi frame payload so that a single waveform can be decoded correctly by two Wi-Fi stations using distinct channel bandwidth and MCS configurations. Moreover, it is fully transparent to slow IoT Wi-Fi stations, where not even a software update is needed. For the high-speed broadband Wi-Fi stations, some additional post-processing of the frame's payload needs to be done which, however, can be fully implemented in software. Note, that our approach is fully compatible with the coding and modulation modules in COTS Wi-Fi devices and can be implemented as an add-on module on top of the current 802.11n/ac standards. To our best knowledge, the proposed technique is the first that can enable multi-user transmission in 802.11n/ac SISO systems without making changes to the physical layer. Our theoretical results show that compared to the traditional SISO system where the DL stations are served in a time-sharing manner, our approach can provide up to a 3× increase in the total network throughput. Moreover, to show its feasibility, we implemented a prototype using commodity Wi-Fi hardware (i.e., Atheros-based NICs). The results reveal the feasibility of our approach under real conditions as a software-only solution.

2 BACKGROUND

2.1 Downlink Scheduling in 802.11

A Wi-Fi Access Point (AP) serving multiple stations in the DL has to distribute the available radio resources among them. The most important scheduling algorithms are: (i) Round-Robin (RR) and (ii) airtime scheduling. With RR scheduling, the AP sends one packet at a time to each station in turn (assuming that there are always DL frames for each station). Here neither the MCS nor the packet length is taken into account. This leads to the situation that a station served on a low MCS will block the channel for a long time, even though its packet might be small. In case all frames have the same payload size with RR scheduling every station is served by the AP with the same data rate on the MAC layer $\hat{R}_i$ which is determined by the PHY data rate of the slowest station, i.e., $\hat{R}_i = (\sum R_i^{-1})^{-1}$ where R_i is the PHY rate of the station i. With airtime scheduling, the AP schedules channel resources based on the channel occupation time of the users. Each user is assigned equal time to occupy the channel, ensuring fairness in channel usage. Hence, the AP can send more bits to a fast station. The data rate on the MAC layer $\hat{R}_i$ of a user i is computed as $\hat{R}_i = R_i \times N^{-1}$ where N is the total number of stations to be served in the DL.

2.2 802.11 Physical Layer

2.2.1 Waveform. IEEE 802.11n uses Orthogonal Frequency Division Multiplexing (OFDM) as the physical layer. OFDM divides the available spectrum bandwidth B into many small and partially overlapping frequency bands called subcarriers. The subcarrier frequencies are selected in such a way that they are orthogonal to one another, i.e., signals on subcarriers do not interfere. In practice, OFDM is efficiently implemented using Fast Fourier Transform (FFT). In an OFDM system with FFT size N, each subcarrier has the same width of B/N Hz. Each subcarrier can be modulated independently (e.g., QAM). After modulation, the sender performs an inverse FFT to convert the frequency domain representation into the time domain which is sent over the air interface. The time needed to transmit these N samples is usually called the FFT period, which is equal to N/B sec. On the receiver side, the OFDM signal is converted back into the frequency domain using FFT, and each subcarrier is demodulated. In 802.11n, the 20 MHz channel consists of 64 subcarriers with 312.5 KHz spacing, however, only 56 of these 64 are used for communication, occupying the bandwidth of 17.5 MHz. The remaining eight subcarriers (i.e., three and four guards at both bandwidth edges and one DC component in the middle) are null-subcarriers that do not carry any signal. Moreover, four of those 56 subcarriers, so-called pilots, are used for channel state estimation and tracking. They are loaded with pseudo-random pilot symbols and their inviolability is crucial for demodulation of Wi-Fi signal.

2.2.2 Frame Detection. Wi-Fi transmits data as self-contained asynchronous frames which can be independently detected and decoded thanks to the prepended preamble and PLCP header (i.e., control data), respectively. A legacy preamble which is common among all MCS is used for easier frame detection. For transmissions with higher bandwidth than 20 MHz, the preamble is copied across all secondary channels. The preamble begins with the so-called Legacy Short Training Field (L-STF), which uses 12 out of the available 52 subcarriers to repeat the same sequence of constellation points 10 times (the remaining 40 subcarriers are not used, i.e., filled with *empty (zero) symbols*). Due to this self-repeating nature, a simple correlator is used to detect a Wi-Fi transmission. Once a preamble is declared, a receiver can use the following Legacy Long Training Field (L-LTF), which repeats the same known sequence twice across all 52 subcarriers, to calculate the exact sample offset and achieve sample synchronization to decode the rest of the preamble.

2.3 Cross-Technology Communication

Cross-Technology Communication (CTC) enables direct over-the-air communications across heterogeneous (incompatible) wireless technologies, which removes the need for multi-radio gateways and therefore avoids their drawbacks (e.g., hardware cost, deployment complexity, or increasing wireless traffic). Therefore, signal emulation techniques are required, which were first introduced in a pioneering CTC scheme called WeBee [14], which enabled a Wi-Fi device to transmit (i.e., emulate) a ZigBee signal by proper selection of its frame payload bits. TwinBee [4], LongBee [15], and WIDE [10] further improve the quality of signal emulation and hence the reliability of WeBee. Such signal emulation also enabled CTC between Wi-Fi and Bluetooth [12], Wi-Fi and LTE [8]. Since these schemes rely on the OFDM modulator of 802.11n Wi-Fi, they cannot perfectly emulate foreign waveform during the OFDM cyclic prefix (CP), which constitutes 10-20% of each symbol time depending on whether short or long CP is used. But also the old 802.11b Wi-Fi standard can be used to emulate signals. In our work [9], we showed that for the case of emulation of LoRa waveform using the CCK-based modulator from 802.11b.

2.4 Hierarchical Modulation for 802.11

Hierarchical Modulation (HM) [16], often referred to as *overlayed constellations*, is an efficient technique for multimedia broadcast in wireless networks. It exploits the broadcast nature of the wireless channel and allows a single transmission to reach different users with various qualities. Therefore, it embeds different data streams, e.g., a high and a low priority stream, into the same transmission but with different MCS. High-quality users will receive both streams and decode high-quality application content, e.g., image or video data. Meanwhile, low-quality users can receive only the high-priority data stream and enjoy some acceptable video quality.

Existing HM-based schemes are based on specially designed hardware. Recently, Chen *et al.* proposed a solution named SoftHM [2] that is based on 802.11 hardware. It is based on QAM-based signal emulation (cf. Section 2.3) where the transmitter is carefully selecting the payload bits of a Wi-Fi packet to be broadcasted with a single rate (e.g., 64-QAM), while diverse receivers can decode and extract different amount of information (with different reception rate / MCS, e.g., QPSK or 64-QAM) from it based on their reception qualities. For example, in the case of 64-QAM that uses 6 bits to encode a constellation point, we can dedicate 1 out-of 6 bits to a user who experiences a low SNR and can only detect whether the received constellation point is in-phase or out-of-phase (i.e., BPSK modulation), whereas the remaining 5 bits carry data of a high SNR user.

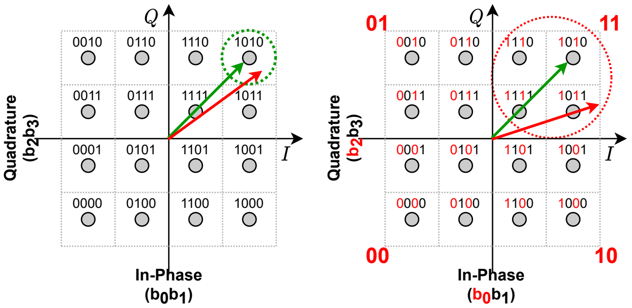


Figure 2: Illustration of hierarchical modulation.

An example of HM is shown in Figure 2. A Wi-Fi device transmits a modulated OFDM signal with different constellation symbols on each subcarrier (constellation point 1010 in this example). A receiver with a high SNR can distinguish among all the 16 symbols with minimal error. However, a receiver with a low SNR can only identify the quadrant of the transmitted constellation symbol and can decode only the two most significant bits of the transmitted symbol. Therefore, in this example, we can deliver only the two most significant bits of a 16-QAM symbol to the low SNR user (as it is easier to decode the quadrant) and the two least significant bits of the symbol to the user experiencing a high SNR. As illustrated in Figure 2, the received symbols at the lower SNR user have a noise sphere of larger radius implying a larger error probability compared to that of a high SNR user in Fig. 2(a) with a smaller noise sphere.

3 THE MATRYOSHKA APPROACH

3.1 In a Nutshell

Matryoshka uses a single Wi-Fi AP equipped with only a single antenna (SISO) which is serving the DL high-speed broadband data users as well as slow-speed IoT devices. The two users can be served in parallel by utilizing HM which is in our case fully emulated. Matryoshka exploits the QAM/OFDM-based signal emulation technique to create an emulated multi-user Wi-Fi transmission in the DL.

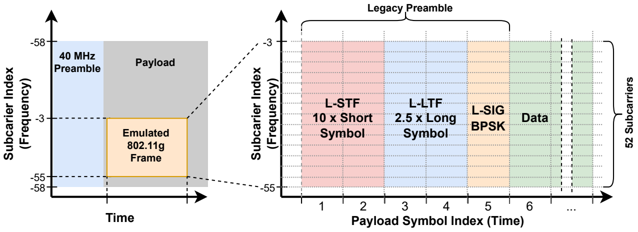


Figure 3: Matryoshka emulates an inner 20 MHz Wi-Fi frame inside an outer 40 MHz Wi-Fi frame.

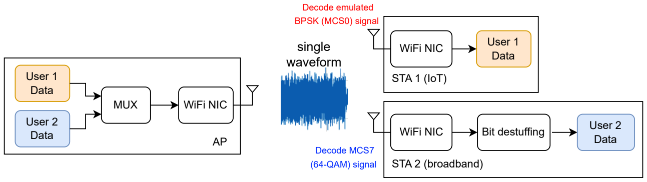


Figure 4: A high-level illustration of Matryoshka architecture with two users being served by the AP in parallel using emulated HM.

Specifically, we create a WiFi-in-WiFi transmission with the slow IoT device (which we refer to as User 1) being served by the inner 20 MHz Wi-Fi frame using a low MCS and the outer 40 MHz frame delivered at high MCS to a fast Wi-Fi broadband STA (User 2) (cf. Figure 3). Note that the inner 20 MHz Wi-Fi is a valid Wi-Fi frame with a fully emulated preamble, PHY header, and payload. For the receiver of the outer 40 MHz frame some additional post-processing, which however can be done fully in software, is needed. Specifically, some of the payload bits need to be removed, as they were injected into the User 2 data payload to modify the original signal waveform and make it emulate a signal carrying User 1 data. The simultaneous DL transmission of both broadband and IoT data has the advantage that valuable channel airtime is no longer blocked by slow IoT transmissions (i.e., inner frame) as they can be transmitted piggyback with broadband data (i.e., outer frame) with only a slight reduction in data rate for the latter due to HM (cf. Figure 4). Moreover, Wi-Fi channel access overhead (i.e., DIFS and back-off) is reduced as it can be fully avoided for the piggybacked IoT frames. In contrast to SoftHM, no changes to the 802.11 protocol are needed. Finally, it is a software-only solution, and unmodified inexpensive SISO COTS hardware can be used for Wi-Fi devices, AP and STAs.

3.2 Emulated Hierarchical Modulation

Matryoshka uses a form of HM where we are interleaving User 2 (broadband) data with extra bits on proper positions so that the produced waveform looks like (emulates) BPSK waveform that carries User 1 (IoT) data. Figure 5 illustrates the details of that data multiplexing process for the case when 64-QAM modulation is used. As User 1 uses BPSK modulation, one bit of data determines only whether a 64-QAM symbol is +1 or -1, (i.e., whether the most significant bit is 0 or 1). The other bits can be freely loaded with User 2 data. However, this process is not trivial as we have to take into account the additional steps involved in the transmission path, i.e., scrambling, interleaving, and block convolution code. Therefore, as shown in Figure 5, in step 2, we compute for User 1 the *Reverse Path* (i.e., we undo operations of the Wi-Fi TX Chain from step 1) in order to find out which bits need to be fixed in User 2 payload in order to emulate BPSK (stage *vi*). In step 3, the *Forward Path* is computed for User 2 and its stream of payload bits is stuffed at the appropriate positions in order to make sure they match the bits fixed by User 1. This is possible as the convolution encoder used in 802.11n Wi-Fi can be represented as a finite state machine (FSM), where the one input bit activates the transition between states and two output bits are generated during the transition [8]. When using 64-QAM, we can observe that all 64 possible states of the Wi-Fi encoder can be classified into only four groups, generating the same output bits when fed with the same input bit. Another important observation is that in each state group, we can arbitrarily set one of the two output bits by switching the input bit between 0 and 1. For instance, when the encoder is in the state from group D, we can put bit 1 to its input to set the next output bit at the position 0 to 0 or put bit 0 as input to set it to 1. Similarly, we can set the output bit at position 1. However, we cannot set both output bits at the same time.

Table 1: Example of Matryoshka encoder

	Ex.1	Ex.2	Ex.3
Required CE Output b_j	x$\underline{1}$	$\underline{0}$x	xx
Encoder State Group	C	A	x
Input Bit	1	1	User 2 Data Bits
Encoder Output	0$\underline{1}$	$\underline{0}$0	xx

The Matryoshka TX exploits the above observation to determine the input bit, knowing the current state of the convolution encoder and required output b_j in the next step. This allows us to multiplex User 2 data bits with extra bits on proper positions so that the output created on the *Forward Path* has the bits fixed for User 1 as computed from the *Reverse Path* (see stage *vi* in Figure 5). In Table 1, we show three examples. The BPSK signal emulation is not perfect. This is because we are using all 64-QAM constellation points to emulate the BPSK, i.e., some with low power (closer to the coordinate origin), and some with high power. As all points are equally probable, the average power of the emulated BPSK symbol is ≈3.7 dB lower than the power of a real BPSK

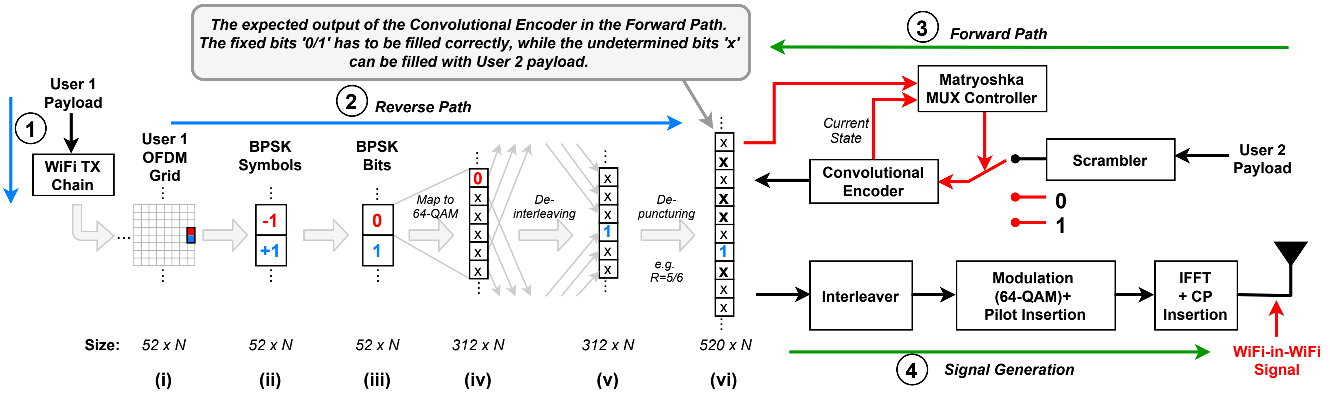

The expected output of the Convolutional Encoder in the Forward Path. The fixed bits '0/1' has to be filled correctly, while the undetermined bits 'x' can be filled with User 2 payload.
3 Forward Path
User 1 Payload
1
WiFi TX Chain
2 Reverse Path
User 1 OFDM Grid
BPSK Symbols
BPSK Bits
Map to 64-QAM
De-interleaving
De-puncturing
e.g. R=5/6
-1
+1
0
1
Current State
Matryoshka MUX Controller
Convolutional Encoder
Scrambler
User 2 Payload
0
1
Interleaver
Modulation (64-QAM)+ Pilot Insertion
IFFT + CP Insertion
WiFi-in-WiFi Signal
Size: 52 x N 52 x N 52 x N 312 x N 312 x N 520 x N
(i) (ii) (iii) (iv) (v) (vi)
4 Signal Generation

Figure 5: Internal structure of Matryoshka Data Multiplexer.

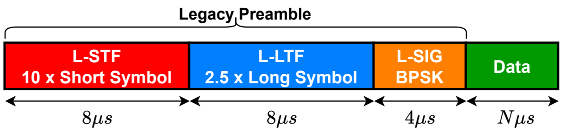


Figure 6: Legacy Wi-Fi (802.11g) frame structure.

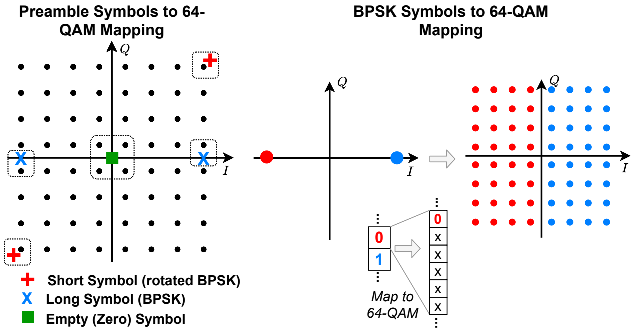

Preamble Symbols to 64-QAM Mapping
BPSK Symbols to 64-QAM Mapping
Q
I
0
1
Map to 64-QAM
0
x
x
x
x
x
Short Symbol (rotated BPSK)
Long Symbol (BPSK)
Empty (Zero) Symbol

Figure 7: Mapping of preamble symbols to 64-QAM constellation.

symbol. We believe that we can use the average loss in power as a measure of the SNR loss for User 1. Finally, in step 4, we use the multiplexed data payload to generate a Wi-Fi signal.

3.3 Preamble Emulation

Our emulated multi-user transmission scheme requires the accurate emulation of a valid 802.11g 20 MHz preamble (cf. Figure 6) for the inner Wi-Fi frame to enable proper frame detection for the IoT user. This is doable as the 52 OFDM subcarriers in 20 MHz 802.11g, which are used to modulate the preamble, have the same positions as some of the 108 subcarriers of the outer 40 MHz using 802.11n (cf. Figure 8). Specifically, the 802.11g preamble of the inner frame with its short and long symbols is emulated by selecting the closest points from the 64-QAM constellation.

Since a lot of bits have to be fixed (i.e., 4/5 out of 6 bits carried on 52 out of 108 subcarriers) in 64-QAM symbols to generate valid L-STF and L-LTF signals, our multiplexing approach cannot be used as it cannot cope with two adjacent fixed bits. We encoded the required bit sequence as a soft-bit sequence that we put into the decoder in the following way. The fixed bits are set with the maximal confidence (value -1.0 for bit 0 and value +1.0 for bit 1), while the remaining bits are set with zero confidence (value 0). This way, we force the used Viterbi decoder to use the fixed bits as anchor points while the remaining bits as degrees-of-freedom (DoF).

The last part of the legacy 802.11g preamble is the legacy signal (L-SIG) (cf. Figure 6) which consists of 24 bits that contain rate, length, and parity information. The L-SIG field is transmitted using BPSK modulation with rate 1/2 binary convolutional coding (BCC). This is a dynamic part as we need to signal the proper MCS (e.g., BPSK) used for the inner Wi-Fi frame to have proper decoding at the receiver side. Figure 1 shows the placement of the inner frame within the outer frame: we have 40 MHz preamble then three OFDM symbols unmodified to allow decoding of the data header, then emulation of the 20 MHz preamble.

3.4 Pilot Sub-Carriers Issue

The pilot subcarriers in 802.11n OFDM transmit a known data sequence (BPSK modulated pseudo-random binary sequence) and are used to correct the residue frequency offset (CFO). In Figure 8, we see that 3 out of 4 pilots of the 20 MHz 802.11g (non-HT) frame overlap in the frequency domain with pilots of the 40 MHz 802.11n (HT40) transmissions. The pilots are set by hardware and hence are uncontrollable. We can emulate only one pilot (± 7) correctly, as the respective subcarrier in 40 MHz frame is used for data. This may cause issues at a receiver node, as by not matching the pilot sequence, it will compute a wrong phase error value and apply a wrong correction, eventually destroying the received signal. However, we found out that we can minimize the error by selecting the index of the OFDM symbol from which the 802.11g frame emulation starts. In our particular case, the

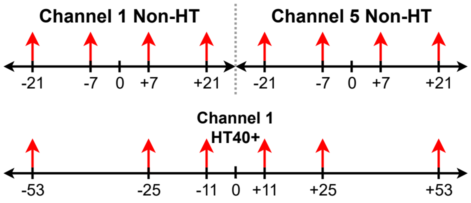


Figure 8: Pilot subcarriers in 20 and 40 MHz channels.

inner non-HT frame has to be sent with an offset of three OFDM symbols after the end of the HT40 preamble to have the best match between the pilot sequences. The remaining pilot (-7) is emulated. Our experiments prove that the error is negligible, and the approach is viable as the inner frame can be received successfully.

3.5 Post-Processing on Broadband Station

While the operation of Matryoshka is fully transparent (i.e., no extra processing needed) to the IoT station receiving the inner 20 MHz frame, some additional post-processing is required for decoding of the outer 40 MHz frame. Here the MAC frame payload received by the Wi-Fi NIC is post-processed in software in order to remove the stuffed bits used for emulation of the inner frame as they do not carry proper data. The positions of the stuffed bits are always the same (i.e., the most significant bits) and can be computed by taking into account the steps involved in the multiplexing process (cf. Figure 5). Note that our solution is very suitable for the envisioned IoT use-case as no post-processing is needed for decoding the inner frame destined for the IoT device which is a desired solution as those devices are of low-speed, low complexity, and battery-powered whereas the additional complexity of the post-processing happens only inside the fast stations which are not resource constrained.

3.6 802.11 MAC layer Issues

The 802.11 Automatic repeat request (ARQ) mechanism requires, in its basic version, the receiver of a unicast frame to send an acknowledgment (ACK) frame after a fixed inter-frame space (i.e., SIFS). This would create a problem, as the ACK for the shorter inner frame might be sent by STA before AP is done with the transmission of the longer outer frame. Moreover, even if the inner and outer frames have the same duration their respective ACKs are sent on two different but partially overlapping channels, hence they would collide at the AP. Matryoshka offers teh following solutions: First, the disabling of the MAC layer ARQ mechanism and relying on reliability provided by higher layers. Second, the usage of deferred block acknowledgments as defined in 802.11n, which are sent contention-based. Third, a mix of both, i.e., IoT users with no ACK policy and broadband users with block ACK.

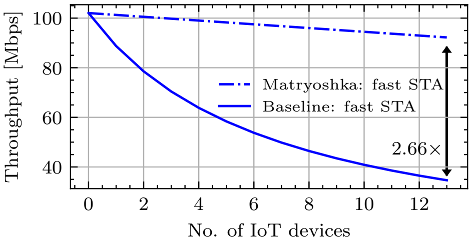


Figure 9: Throughput of the fast user for a different number of IoT devices.

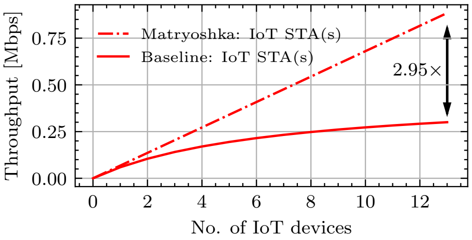


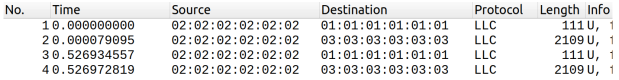
Figure 10: Total throughput of the IoT users for different numbers of IoT devices.

4 ANALYTICAL EVALUATION

We consider a single Wi-Fi BSS consisting of a Wi-Fi AP serving a single fast STA together with N IoT STAs in the downlink. The fast STA is served using 802.11n (40 MHz, long guard interval) using MCS 7 with MPDU aggregation of 10 and single MPDU of size 1500 Bytes resulting in a frame air time of 1176 μs. In contrast, the $N = 1 \ldots 13$ IoT STAs have to be served in DL using 802.11g (20 MHz) using MCS 0 (BPSK) and having a very small MAC payload of 10 Bytes, i.e., frame air time of 76 μ. All DL transmissions are layer-2 broadcasts, i.e., no 802.11 ACK frames and therefore retransmissions. For the AP we assume a scheduler targeting packet-level fairness.[2] Two approaches are compared. First, in the baseline, the single fast and the N IoT STAs need to be served in a round-robin (RR) manner. That means for each transmitted frame, there is a channel access delay before transmission. Second, with Matryoshka the IoT frames are piggybacked onto the fast user's frame using hierarchical modulation (HM) so that only a single channel access delay for the fast user is required. However, due to HM, the data rate of the fast user is slightly reduced (i.e., only 5 out of 6 bits used to carry the fast user's data) resulting in longer transmissions.

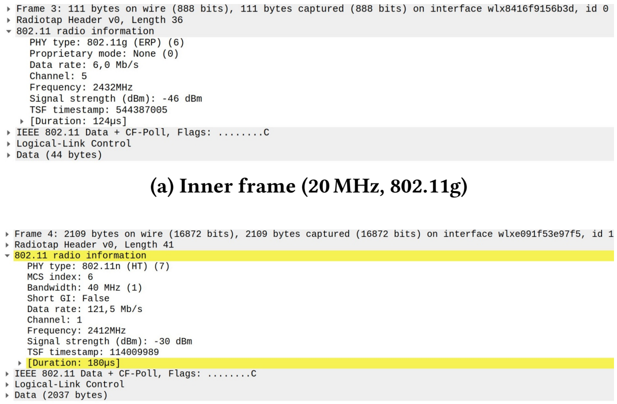
The throughput results for the fast and IoT STA(s) are shown in Figures 9 and 10, respectively. We see that Matryoshka is able to clearly outperform the baseline, as the overhead of channel access for the IoT STAs can be fully omitted as long as all the IoT frames can be served within the large outer frame of the fast STA. In the selected configuration, up

[2] Note, due to space constraints we left out the results for time fairness which give similar results.



```
No.  Time          Source             Destination        Protocol  Length Info
   1 0.000000000   02:02:02:02:02:02  01:01:01:01:01:01  LLC          111 U, f
   2 0.000079095   02:02:02:02:02:02  03:03:03:03:03:03  LLC         2109 U, f
   3 0.526934557   02:02:02:02:02:02  01:01:01:01:01:01  LLC          111 U, f
   4 0.526972819   02:02:02:02:02:02  03:03:03:03:03:03  LLC         2109 U, f
```

Figure 11: Screenshot from Wireshark showing simultaneous reception of two frames.



```
▸ Frame 3: 111 bytes on wire (888 bits), 111 bytes captured (888 bits) on interface wlx8416f9156b3d, id 0
▸ Radiotap Header v0, Length 36
▾ 802.11 radio information
    PHY type: 802.11g (ERP) (6)
    Proprietary mode: None (0)
    Data rate: 6,0 Mb/s
    Channel: 5
    Frequency: 2432MHz
    Signal strength (dBm): -46 dBm
    TSF timestamp: 544387005
  ▸ [Duration: 124µs]
▸ IEEE 802.11 Data + CF-Poll, Flags: ........C
▸ Logical-Link Control
▸ Data (44 bytes)
```

(a) Inner frame (20 MHz, 802.11g)

```
▸ Frame 4: 2109 bytes on wire (16872 bits), 2109 bytes captured (16872 bits) on interface wlxe091f53e97f5, id 1
▸ Radiotap Header v0, Length 41
▾ 802.11 radio information
    PHY type: 802.11n (HT) (7)
    MCS index: 6
    Bandwidth: 40 MHz (1)
    Short GI: False
    Data rate: 121,5 Mb/s
    Channel: 1
    Frequency: 2412MHz
    Signal strength (dBm): -30 dBm
    TSF timestamp: 114009989
  ▸ [Duration: 180µs]
▸ IEEE 802.11 Data + CF-Poll, Flags: ........C
▸ Logical-Link Control
▸ Data (2037 bytes)
```

(b) Outer frame (40 MHz, 802.11n)

Figure 12: Wireshark screenshot of the two frames.

to 13 IoT frames are piggybacked onto one fast user’s frame. The rate reduction due to the hierarchical modulation for the fast user is around 10% which is negligible due to the saving of $N \times (\text{DIFS} + \text{CWmin}/2)$ where N is the number of IoT devices. The best performance is achieved with 13 IoT devices served in the DL, as it is the maximum, which fits into the outer frame. Here, the throughput for the fast STA is 2.66× faster as compared to the baseline. The improvement is also visible for IoT devices, which see an increase in throughput by a factor of 2.95×.

Takeaway message. Matryoshka outperforms classical Wi-Fi in a mixed broadband/IoT scenario due to the gain from HM and the saved overhead in the channel access for the IoT transmissions. So, thanks to the HM we can trade the SNR of the IoT users to increase the total throughput which is possible in case the IoT users experience good channel conditions that they cannot take advantage of because of being restricted to low MCS.

5 PROTOTYPE IMPLEMENTATION

The prototype of Matryoshka was implemented using COTS Wi-Fi hardware. On the Wi-Fi side for both AP and stations, we used Atheros AR928x (802.11n) NIC in SISO mode. All required software modules (QAM emulation, post-processing) were prototypically implemented in Matlab/Python.

In order to show the feasibility of our approach, we performed over-the-air measurements. Therefore, the AP was transmitting frames according to the Matryoshka approach. On the receiver side, we used a device equipped with two Wi-Fi interfaces in order to emulate the reception by two stations. Here, one interface was configured on channel 1 HT40+ (40 MHz, 802.11n) and the other on channel 5 Non-HT (40 MHz, 802.11g). Figures 11 and 12 show screenshots from Wireshark of the two captured packets. As can be seen, there are two frames (11n and 11g) received nearly at the same time, i.e., the time offset is around 38 μs. In addition, we estimated the packet reception rate which was 96.6% and 84.0% for the emulated 20 MHz and the outer 40 MHz, respectively. This confirms the functionality of our approach.

6 DISCUSSION

Matryoshka has other advantages beyond those already mentioned. For example, a user operating with 40 MHz can also decode the content of the inner frame. This is possible as she can reconstruct its content from the punctured bits of her payload. This possibility is beneficial for different use cases. For example, normal Wi-Fi management frames like beacons which need to be transmitted by AP periodically could be sent as inner frames. This would save a lot of airtime which would be otherwise wasted by the beacon frames as they have to be transmitted on the lowest MCS0. Moreover, this also reduces the risk of having delayed beacon transmissions.

Matryoshka is limited to DL SISO transmissions, where the gain from HM is highest. In future work, we aim to explore the possibility of extending Matryoshka towards newer Wi-Fi versions (802.11ac/ax/be). We expect that higher modulation orders will improve signal emulation quality. Specifically, with more QAM constellation points, we can closely approximate even complex signals (e.g., VHT preambles). However, the more advanced coding schemes may efficiently prevent the implementation of the WiFi-in-WiFi signal emulation. Specifically, it might be not possible to multiplex long sequences of data bits at proper positions. Moreover, we want to explore the usage of MIMO transmissions for WiFi-in-WiFi signal emulation. Note that StarLego [1] exploits MIMO Wi-Fi transmissions to emulate custom signals, however, so far its authors could not create a multi-user Wi-Fi transmission.

7 RELATED WORK

Our work was inspired by the progress in CTC using the signal emulation technique as a way to create an emulated waveform of a target wireless system. In general, an advanced wideband wireless technology (e.g., Wi-Fi) has enough degrees of freedom in its signal modulation to emulate the complete waveform of a simpler (i.e., narrowband) technology (e.g., ZigBee or Bluetooth) [3]. The emulated signal follows the standard of simple technology, and hence it can be directly demodulated without a need for hardware or software modifications. In our specific case, we utilize the so-called

QAM emulation technique to emulate a WiFi-in-WiFi transmission, where a high-order MCS transmission contains its own data stream, but also emulates a lower-order transmission (e.g., BPSK). In other words, we emulate signals for multiple MCS layers within a single 802.11n transmission.

Extensive prior work was done in realizing multiple transmissions using devices equipped with only a single RF chain, including Non-Orthogonal Multiple Access (NOMA) [6, 13] and HM [7, 11]. The SoftHM [2] approach emulates HM in software so that different data with different MCS can be sent within a single Wi-Fi transmission. Matryoshka can be seen as an extension of SoftHM by supporting different channel bandwidth configurations for high and low-priority data streams. In SoftHM both need to be of the same bandwidth, i.e., 20 MHz, whereas with our approach the low-priority data stream uses a larger bandwidth, e.g., 20 Mhz. Moreover, our approach is fully compliant with 802.11 and can be used with unmodified Wi-Fi hardware as the signal field of the high-priority data carrying the rate information of the payload (used MCS) is emulated as well. In SoftHM changes are required to configure the signal field. The SIMBA [7] approach uses so-called overlayed constellations, which is a form of HM to create multi-stream multi-user DL transmission via a single RF chain in 60 GHz Wi-Fi (802.11ad). This allows grouped users at different locations to share the same transmit beam from the AP. In contrast to our approach, SIMBA requires changes to be made to Wi-Fi hardware.

8 CONCLUSION

This paper introduces Matryoshka, the first system that uses signal emulation technique to emulate multi-user transmission to serve multiple client stations simultaneously in the DL from a Wi-Fi AP equipped with only a single RF chain. It is a full software solution and works with COTS 802.11n/ac hardware. While our current implementation of Matryoshka is limited to the two-user transmission case, we believe that it can be extended to support more users by utilizing larger bandwidth. With 80 MHz and 160 MHz channel bandwidth as defined in 802.11ac, it should be possible to serve even more users at the same time on different 20 MHz sub-channels (effectively emulating OFDMA transmission with legacy Wi-Fi versions). Extensive evaluation of our Matryoshka prototype through real-world experiments is planned for future work.

ACKNOWLEDGEMENTS

This work was supported by the Federal Ministry of Education and Research (BMBF, Germany) within the 6G Research and Innovation Cluster 6G-RIC under Grant 16KISK020K as well as by the German Research Foundation (DFG) within the project ML4WiFi under grant DR 639/28-1.

REFERENCES

[1] Ruirong Chen and Wei Gao. 2021. StarLego: Enabling Custom Physical-Layer Wireless over Commodity Devices. In *ACM HotMobile 2021*. ACM, Austin, TX, 80–85. https://doi.org/10.1145/3376897.3377852

[2] Weiwei Chen, Yunhuai Liu, and Tian He. 2021. SoftHM: A Software-Based Hierarchical Modulation Design for Wireless System. *IEEE/ACM Transactions on Networking* 29, 1 (Feb. 2021), 452–464. https://doi.org/10.1109/tnet.2020.3040006

[3] Ying Chen, Ming Li, Pengpeng Chen, and Shixiong Xia. 2019. Survey of cross-technology communication for IoT heterogeneous devices. *IET Communications* 13, 12 (July 2019), 1709–1720. https://doi.org/10.1049/iet-com.2018.6069

[4] Yongrui Chen, Zhijun Li, and Tian He. 2018. TwinBee: Reliable Physical-Layer Cross-Technology Communication with Symbol-Level Coding. In *IEEE INFOCOM 2018*. IEEE, Honolulu, HI, 153–161. https://doi.org/10.1109/INFOCOM.2018.8485816

[5] Cisco. 2018. *Visual Networking Index: Forecast and Trends, 2017–2022*. Press Release. Cisco, Inc.

[6] Linglong Dai, Bichai Wang, Yifei Yuan, Shuangfeng Han, I. Chih-lin, and Zhaocheng Wang. 2015. Non-orthogonal multiple access for 5G: solutions, challenges, opportunities, and future research trends. *IEEE Communications Magazine* 53, 9 (Sept. 2015), 74–81. https://doi.org/10.1109/mcom.2015.7263349

[7] Keerthi Priya Dasala, Josep Miquel Jornet, and Edward W. Knightly. 2020. SIMBA: Single RF Chain Multi-User Beamforming in 60 GHz WLANs. In *IEEE INFOCOM 2020*. IEEE, Virtual Conference, 1499–1508. https://doi.org/10.1109/infocom41043.2020.9155441

[8] Piotr Gawłowicz, Anatolij Zubow, Suzan Bayhan, and Adam Wolisz. 2020. Punched Cards over the Air: Cross-Technology Communication Between LTE-U/LAA and WiFi. In *IEEE WoWMoM 2020*. IEEE, Virtual Conference, 297–306. https://doi.org/10.1109/WoWMoM49955.2020.00058

[9] Piotr Gawłowicz, Anatolij Zubow, and Falko Dressler. 2022. Wi-Lo: Emulation of LoRa using Commodity 802.11b WiFi Devices. In *IEEE ICC 2022*. IEEE, Seoul, South Korea, 4414–4419. https://doi.org/10.1109/ICC45855.2022.9838667

[10] Xiuzhen Guo, Yuan He, Jia Zhang, and Haotian Jiang. 2019. WIDE: Physical-level CTC via Digital Emulation. In *ACM/IEEE IPSN 2019*. ACM, Montreal, Canada, 49–60. https://doi.org/10.1145/3302506.3310388

[11] Szymon Jakubczak and Dina Katabi. 2011. A Cross-Layer Design for Scalable Mobile Video. In *ACM MobiCom 2011*. ACM, Las Vegas, NV. https://doi.org/10.1145/2030613.2030646

[12] Wenchao Jiang, Zhimeng Yin, Ruofeng Liu, Zhijun Li, Song Min Kim, and Tian He. 2017. BlueBee: A 10,000x Faster Cross-Technology Communication via PHY Emulation. In *ACM SenSys 2017*. ACM, Delft, Netherlands, 1–13. https://doi.org/10.1145/3131672.3131678

[13] Evgeny Khorov, Aleksey Kureev, Ilya Levitsky, and Ian F. Akyildiz. 2020. Prototyping and Experimental Study of Non-Orthogonal Multiple Access in Wi-Fi Networks. *IEEE Network* 34, 4 (July 2020), 210–217. https://doi.org/10.1109/mnet.011.1900498

[14] Zhijun Li and Tian He. 2017. WEBee: Physical-Layer Cross-Technology Communication via Emulation. In *ACM MobiCom 2017*. ACM, Snowbird, UT, 2–14. https://doi.org/10.1145/3117811.3117816

[15] Zhijun Li and Tian He. 2018. LongBee: Enabling Long-Range Cross-Technology Communication. In *IEEE INFOCOM 2018*. IEEE, Honolulu, HI, 162–170. https://doi.org/10.1109/INFOCOM.2018.8485938

[16] Alexander Schertz and Chris Weck. 2003. *Hierarchical modulation-the transmission of two independent DVB-T multiplexes on a single frequency*. EBU Technical Review. Institut für Rundfunktechnik.