I. Introduction

A. Motivation and Prior Work

THE problem of communicating with many receivers arises in many downlink scenarios such as communication from an access point to stations in WiFi or from a base station in cellular systems. The conventional approach is to set up orthogonal channels to each user by time/frequency/code-division multiplexing. Although this approach eliminates interference between transmissions, it does not in general achieve the highest possible transmission rates for a given packet error rate (or reliability) [1]. In fact, Superposition Coding (SC) [2] is a well-known non-orthogonal scheme that achieves the capacity on a scalar Gaussian broadcast channel.

We motivate the use of SC for the two-receiver case. Consider a cellular downlink with several active users. Given the user density in typical networks, it is always possible to pick two users N (the *near* user) and F (the *far* user), as shown in Fig. 1. The key observation here is that N being geographically closer to the base station (BS) has a stronger

of Padova, Italy (e-mail: kstamat@dei.unipd.it).

Digital Object Identifier 10.1109/TWC.2012.051512.111622

not vice versa). The idea behind SC is to optimally exploit this channel ordering.

A BS that uses two-receiver SC can transmit *superimposed* F and N packets (or more precisely, the far and near user codewords) in *both* F s and N s time slots (see Fig. 1). Thus both links enjoy the *combined* degrees of freedom available to N and F, while sharing the transmit power. For large blocklengths, it can be shown that it is possible to encode F s packets such that they can be decoded in the presence of interference from N s packets. Since N has a stronger link to the BS, N can replicate this step to regenerate and thereby cancel F s signal from its received signal. It can then decode its own packet. This is the well-known *Successive Decoding* (SD) or *Successive Interference Cancellation* (SIC) procedure [1].

We can extend this two-user scheme to any number of users. In fact, SC (combined with SD) achieves the capacity on a scalar Gaussian broadcast channel. This implies that *any* TD-achievable rate-pair (i.e., the pair of *spectral efficiencies* on a Gaussian channel) can also be achieved using SC, with the rate gain over TD increasing with the disparity in the user link qualities.

While information theory sufficiently motivates the use of SC, it is largely silent on practical issues such as finite block length codes, finite encoding and decoding complexity, hardware non-idealities (e.g., carrier frequency offset, phase noise) that one would encounter while designing such a system. This motivates the experimental study of SC.

For rapid prototyping and streamlining the design effort, we adopt a software-defined radio (SDR) [3] paradigm using the well-known open-source GNU Radio platform in conjunction with the Universal Software Radio Peripheral (USRP) hardware board that serves as an analog and RF front-end [4]. A well-known prototyping system [5], [6], it has been recently used in testbed design, including UT Austin's Hydra [7] and by Bell Labs and Microsoft Research [8].

B. Main Contributions

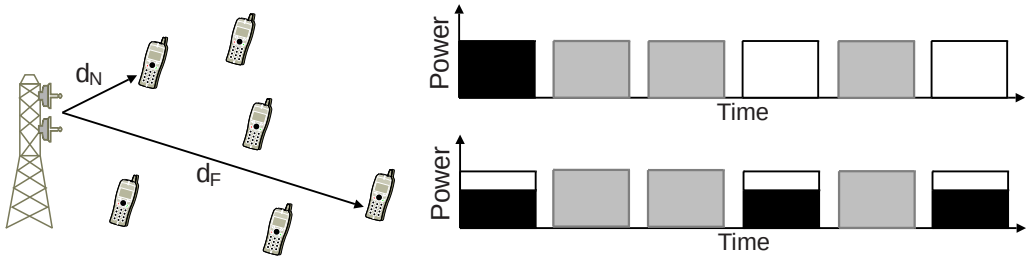


Fig. 1. Illustration of two-user SC. (Left) The users N and F picked are at distances d_N and d_F respectively with $d_N < d_F$. (Right) Typical transmission timelines with and without SC. The gray slots represent transmissions to other active users which can remain unchanged. With Time-Division (TD, top) N and F are served in different slots (black and white). With SC (bottom), BS transmits a linear combination of individually-coded user waveforms.

attempt at systematically designing and characterizing an SC physical layer that, along with its accompanying hardware, forms a functioning system.

3) We study the implications of using a finite constellation along with a demodulate-and-decode receiver architecture on the statistics of the interference-plus-noise term.

C. Paper Overview

The remainder of the paper is organized as follows. In Section II we briefly summarize how SC achieves capacity and discuss some implications of restricting the library of codes to a finite set of finite-blocklength codes. In Section III, while retaining the rate-centric approach, we propose a design technique for SC with such a finite code library and specialize this technique to a library comprised of a well-known family of codes designed using the Bit Interleaved Coded Modulation (BICM) technique [11], and predict the theoretically achievable rate region. In Section IV, we describe the system architecture that uses these BICM codes to implement SC. In Section V, we present an experimental setup that emulates a Gaussian BC and use it to experimentally determine the achievable spectral efficiency pairs for a two-receiver BC under a packet-error constraint. The resulting rate region is the finite-library analog of the information-theoretic rate region. We also discuss some practical issues that arise in the design of superposition-coded systems, including the validity of treating inter-user interference as Gaussian noise. In Section VI we conclude the paper and suggest possible avenues for future work.

II. Superposition Coding: From Theory to Practice

We will briefly summarize relevant results from [1], [2] on achieving the capacity of the (scalar) Gaussian Broadcast Channel (BC) using SC with SD. In addition to making this paper self-contained, this discussion identifies the key architectural building blocks of a superposition-coded system. A closer examination of the blocks allows us to identify some key practical issues in implementing this ideal scheme.

We use calligraphic fonts (e.g. $\mathcal{C}$) to represent sets and sans-serif fonts (e.g. $f(\cdot)$) to denote the encoding/decoding maps. Also, we use $[M]$ to represent $\{1, \cdots, M\}$ for $M \in \mathbb{Z}^+$, and occasionally use the short-hand Tx for a transmitter and Rx for a receiver.

A. Achieving the Capacity on the Gaussian BC

Consider a BS that wants to communicate with two receivers N and F. The broadcast nature of the wireless medium is captured by the broadcast channel model $X \to (Y_N, Y_F)$ where X denotes the channel input and Y_N and Y_F are the channel outputs at N and F.[1] Let $(X(n))$ be a sequence of channel inputs indexed by the channel use $n \in [L]$. Clearly $(X(n))$ must encode information relevant to each user. The capacity region of this channel is the closure of the set of all possible pairs of transmission rates at which the BS can reliably send two independent information streams, one each to N and F (allowing $L \to \infty$).

For a Gaussian BC, we have

$$Y_N(n) = h_N X(n) + Z_N(n); \quad Y_F(n) = h_F X(n) + Z_F(n) \quad (1)$$

where N (resp. F) has a complex channel gain h_N (resp. h_F) and $Z_u, u \in \{N, F\}$ denote the WGN processes. We assume the BS operates with an average power constraint P [W] and a (baseband) bandwidth W [Hz], and denote the noise power spectral density by N_0 [W/Hz]. From the above, the power constraint per channel use is P/W and $E[|Z(n)|^2] = N_0 W$. From the definition of N and F, $|h_N|^2 > |h_F|^2$.

One way for $(X(n))$ to encode information is to communicate with each user in turns by partitioning the total number of channel uses into time slots (as in TD). For a given n, $X(n)$ contains information pertaining to just one user. This is the well-studied point-to-point communication problem, for which good practical encoding and decoding schemes exist. However, for a BC it is known that TDM is suboptimal in general; the root cause lies in its inability to fully exploit the fact that $|h_N| > |h_F|$: N has a "stronger" channel to BS, and hence can always decode information that can be decoded at F. This makes the scenario ideal for the SC scheme which achieves every pair of transmission rates in the capacity region.

The key architectural elements of an SC system are:

1) A superposition encoder f that consists of
   a) Two point-to-point encoders $f_N : \{0,1\}^{2^{LR_N}} \to \mathcal{C}^L$ (which we call the near-encoder) and $f_F : \{0,1\}^{2^{LR_F}} \to \mathcal{C}^L$ (which we call the far-encoder), that map their respective inputs (the near- and far-messages) to complex-valued sequences $(X_N(n))$ and $(X_F(n))$, each of block

[1] In practical terms, $X(n)$ can be understood as a (coded) symbol stream from the BS, and the Y's as the corresponding noisy and/or distorted observations of this symbol stream at N and F.

reduce this computational overhead by reducing the search space of possible far-rates for a given near-rate. In fact, that formula to compute the PER for a given rate pair and near fraction would obviate the need for simulations.

Unfortunately, accurate formulas for the PER are difficult to obtain even for the point-to-point case, although there exist well-known upper bounds that are asymptotically tight in the high-reliability regime (PER $\to 0$) [12]. In the high-reliability regime, these upper bounds can be suitably modified, as we will show.

The key difference between SC-BICM and point-to-point BICM lies in the far-demodulator x_F^{-1} that estimates the reliabilities of the far-code bits from observations of the form

$$\boldsymbol{Y}_F(n) = \underbrace{\boldsymbol{h}_F \sqrt{1-\alpha}\boldsymbol{X}_F(n)}_{\text{Signal}} + \underbrace{\boldsymbol{h}_F \sqrt{\alpha}\boldsymbol{X}_N(n) + \boldsymbol{Z}_F(n)}_{\text{Perturbation}}, \quad n \in [L]. \tag{8}$$

Unlike in point-to-point BICM, the perturbation term is not Gaussian; its statistics depend on N's constellation, which we assume is known to the demodulator.[11] Whether or not x_N is known, the finiteness of the interference constellation raises an interesting question about the validity of the Interference-As-Gaussian-Noise (IAGN) model that assumes such Gaussianity. We investigate this question in greater detail in Section V-C3.

When N's constellation is known, it is useful to treat each composite symbol $X(n) = \sqrt{1-\alpha}X_F(n) + \sqrt{\alpha}X_N(n)$ as a member of a superconstellation with $2^{b_{x_N}+b_{x_F}}$ points (see also Fig. 7). Viewed from the demodulator, interference perturbs each original far-symbol (the *parent* point) to a randomly chosen *daughter* point. For each parent point, define the set of all possible daughter points to be its *potential daughter cluster* (cluster for short). The size and shape of this cluster depends on the interferer's constellation. Thus a maximum-likelihood demodulator interested only in the far-packet infers the most probable parent point of the observed (noisy) daughter point by identifying the most probable cluster to which an observation belongs. Identifying successively less probable clusters helps the demodulator refine its reliability estimate of each detected code bit of F. Analogous to the single-user case, the reliability of the k^{th} bit in the n^{th} symbol can be approximated using the max-log-MAP approximation:

$$L_n^{(k)} \approx N_0 \left(\min_{s \in \sqrt{1-\alpha}\mathcal{S}_{x_F}^{(k-)}} \; \min_{\sqrt{\alpha}\mathcal{S}_{x_N}} |Y_F(n) - h_F s|^2 - \min_{s \in \sqrt{1-\alpha}\mathcal{S}_{x_F}^{(k+)}} \; \min_{\sqrt{\alpha}\mathcal{S}_{x_F}} |Y_F(n) - h_F s|^2 \right) \tag{9}$$

where $\mathcal{S}_{x_F}^{(k-)} \subset \mathcal{S}_{x_F}$ and $\mathcal{S}_{x_F}^{(k+)} \subset \mathcal{S}_{x_F}$ comprise symbols whose k^{th} bits are 0 and 1, respectively.

In the high-reliability regime, the dominant error events in such a demodulator are events where a daughter point is incorrectly identified with a neighboring cluster. Analogously to the point-to-point case (when these clusters are just points), the probability of these error events is controlled by the effective cluster separation $d_{\text{eff}} \equiv d_{\text{eff}}(\alpha, x_F, x_N)$ given by

$$d_{\text{eff}} = \min_{\substack{p_1, p_2 \in \sqrt{1-\alpha}x_F \\ p_1 \neq p_2}} \; \min_{d_1, d_2 \in \sqrt{\alpha}x_N} |p_1 + d_1 - p_2 - d_2|. \tag{10}$$

Here the superscript (F,N) on the left hand side emphasizes that the parent points are drawn from x_F and the interferer is drawn from x_N. Using arguments similar to those in [12], [13], PER_F can be approximated as

$$\text{PER}_F \approx B_F w_{c_F} Q\left(d_{\text{eff}}^{(F,N)} \sqrt{\frac{d_{c_F}\gamma_F}{2}} \right) \tag{11}$$

in the high-reliability regime, where $Q(x)$, $x \geq 0$ is the Q-function. Assuming perfect cancellation at N,

$$\text{PER}_N \approx B_N w_{c_N} Q\left(d_{x_N} \sqrt{\frac{d_{c_N}\gamma_N}{2}} \right), \tag{12}$$

which can be plugged into (12) to approximate PER_N.

However, for PERs of practical interest (say PER 0.1) these Q-function bounds are too loose to predict the correct solutions to (6). In this regime we use these bounds as estimates that reduce the search space of the achievable far-rates, and refine them further via simulation. In the following subsection we illustrate these ideas with a design example. The code library chosen in the example is the same as the one used in our system design.

C. A Design Example

We use a BICM code library with a decoder structure explained in Section III-B2. These BICM codes were implemented in our testbed in a point-to-point setting.

The library consists of all possible pairings of convolutional codes $\{c^{(1)}, c^{(2)}, c^{(3)}, c^{(4)}\}$ with three constellation mappers.[12] There is no interleaving. Table I summarizes the details of these convolutional codes. Note that $c^{(2)}$, $c^{(3)}$ and $c^{(4)}$ are obtained by appropriately puncturing $c^{(1)}$, which is the standard rate-1/2 constraint length 7 convolutional code with the generator matrix $[133, 171]$. The constellations are $x \in \{\text{BPSK}, \text{QPSK}, \text{16QAM}\}$. These are all QAM constellation with even b_x, so $d_x = \sqrt{6/(2^{b_x}-1)}$ [10]. With this code library the available set of spectral efficiencies is the sequence $(r_1, r_2, \ldots, r_{12})$ obtained by ordering the elements of the set $\{1, 2, 4\} \times \{\frac{1}{2}, \frac{2}{3}, \frac{3}{4}, \frac{5}{6}\} = \{\frac{1}{2}, \frac{2}{3}, \frac{3}{4}, \frac{5}{6}, 1, \frac{4}{3}, \frac{3}{2}, \frac{5}{3}, 2, \frac{8}{3}, 3, \frac{10}{3}\}$ in ascending order.

We now solve (6) via Monte Carlo simulation (using the simplified procedure outlined in Section) for $= 0.1$ and $L = 1536$ with perfect receiver CSI and perfect interference cancellation. For our implementation this choice of L strikes a balance between code performance and implementation constraints. Besides illustrating the design procedure for a concrete example, the far-rates from this simulation provide upper bounds for a practical system (where neither of these conditions holds). The simulation procedure closely follows the experimental procedure described in Section V-B. The first step involves obtaining the single-user PER curves similar to those in Fig. 5 by simulating a link operating over a point-to-point Gaussian channel. As with the experiment, these results yield $\{\beta(l)\}$, $l \in [M]$ that are given to another Matlab program that simulates a Gaussian BC and numerically

[11] In practice sending this information entails a small overhead, which we neglect in this paper.

[12] Code performance can be optimized by suitably tailoring the interleaver. However its absence does not change the main message of this example (or indeed, that of the paper), which is to show substantial gains from SC even for finite blocklengths and constellations.

TABLE I
KEY PARAMETERS OF THE CONVOLUTIONAL CODES IN THE CODE LIBRARY.

Code	Rate ρ_c	Free Dist d_c	#Free Dist. Error Events β_c
$c^{(1)}$	1/2	10	11
$c^{(2)}$	2/3	6	1
$c^{(3)}$	3/4	5	8
$c^{(4)}$	5/6	4	14

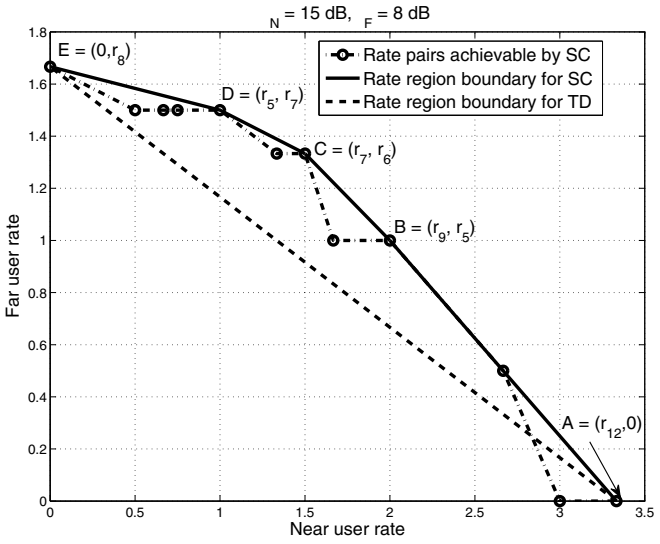


Fig. 3. Optimal rate pairs (solution to (6)) for N = 15 dB and F = 8 dB for the library of $M = 12$ codes described in Section III-C. The TD rates are obtained by time-sharing between the single-user operating points. The values of at corner points A-E are 1 0 2138 0 1259 0 0631 0 respectively.

solves (6) using the procedure described in Section V-B. Fig. 3 shows the simulated rate region for $\gamma_N = 15$ dB and $\gamma_F = 8$ dB. For these parameters, it takes about 10 hours to complete this procedure on Matlab version 7.13 running on a dual-core Linux workstation running at 2.4 GHz and with 2 GB RAM.

IV. IMPLEMENTING A SUPERPOSITION-CODED SYSTEM

A. The Platform

We implemented all physical layer processing steps at BS, N and F by suitably modifying an existing point-to-point wireless testbed. The testbed uses Orthogonal Frequency Division Multiplexing (OFDM), and its design parameters are similar to those in the IEEE 802.11a standard. The testbed runs on GNU Radio (revision 10923) on a Linux PC. GNURadio provides driver functions that interface the PC with the USRP board that functions as the analog frontend and the RF.

B. Packet Structure

Transmissions occur in *frames*. As in WiFi, each frame consists of a preamble followed by a header and a payload. The preamble assists the receiver in frame acquisition and channel estimation. The header encodes the BICM code type and α. The preamble and the header designs are left unchanged from the single-user case.[13] The only multiuser section in the frame is the payload. For lack of space, we will not provide more details here; they can be found in our technical report [16].

TABLE II
PARAMETERS USED IN THE EXPERIMENT

Center Frequency	903 MHz
Message Bandwidth	2 MHz
Modulation	16-tone OFDM (8 data, 4 pilot, 4 null)
CP Length	1μs

The block of $L = 1536$ coded symbols is transmitted over 192 OFDM symbols with 8 subcarriers for the payload in each OFDM symbol. We retain the 4 pilot subcarriers used by the testbed for frequency and phase tracking and 4 null subcarriers for spectral shaping. This brings the total number of subcarriers to $8 + 4 + 4 = 16$. The message bandwidth of 2 MHz is limited by the USRP, and the cyclic prefix is made commensurate with the (relatively) flat frequency response of the channel for this bandwidth. We summarize the system parameters in Table II.

C. Single-User Characterization

Creating experimental conditions that ensure a time-invariant wireless propagation loss can be complicated: although the indoor radio channel and the USRP boards have a reasonably flat response over a 2 MHz bandwidth, the propagation loss is quite sensitive to changes in the environment (e.g., those caused by motion). One approach would be to compensate for such changes by appropriate power control at the BS. Doing so would require SNR feedback from the users on a control channel; designing such feedback links would be worth the effort only if SC could provide substantial rate-gains in a *perfectly power-controlled* environment. We focus on the latter question in the paper.

While manual power adjustments work at smaller time scales lasting a few minutes [18], they are quite cumbersome for longer experiments (e.g., measuring the PER for all the rates in the library, or solving (6), see Section V-A).

To circumvent this problem, we emulate perfect power control by connecting the Tx and Rx with a coaxial cable (see Fig. 4), resulting in controllable experimental conditions and reproducible outcomes.[14] With this coaxial setup, we measure the point-to-point PER as a function of the transmit power P (which implicitly determines the SNR γ) for each rate index in the code library using parameters from Table II.

Starting from a value of P chosen such that the received power (at the antenna port, as measured by a spectrum analyzer) is at a fixed level above the theoretically predicted thermal noise floor, we change the transmit power in 1 dB

[13] Consequently, known techniques (i.e. the use of windowed correlators as in [15]) and low-rate header encoding (as in IEEE 802.11a) can be used.

[14] Indeed, the same practical considerations motivate the use of channel emulators.

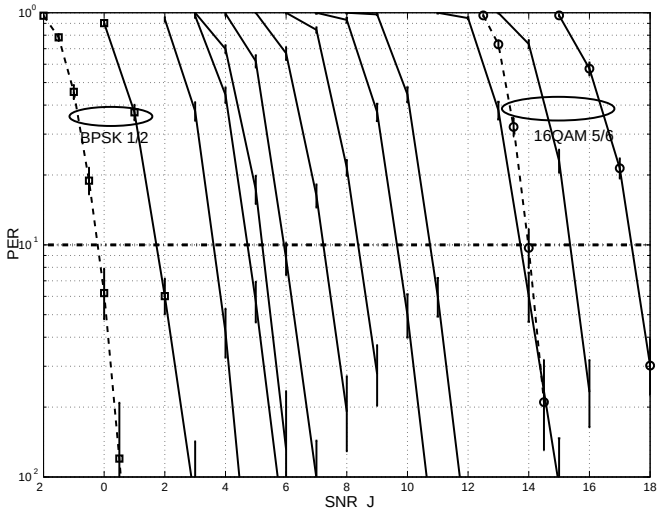


Fig. 4. The setup used to approximate a Gaussian BC. The USRP boards are connected via cables. A splitter is used to split the transmitted signal to the two receivers, while an attenuator is used to (virtually) create the presence of a far user.

Fig. 5. Single-user PER versus SNR, along with 95% confidence intervals for all the 12 rates in the library. The solid lines depict the experimental results; the ideal curves for BPSK-1/2 and 16QAM-5/6 are also shown (dashed lines). Observe that at 10% PER the implementation loss is in the range 2dB ≤ 3.5dB.

steps and measure the PER for each rate index.[15] For comparison we overlay the PER plots for a simulated Gaussian system with the same theoretical noise floor but without additional sources of noise and distortion in Fig. 5. For a rate index l, let $P_{min}(l;\ )$ $P_{min}(l)$ be the smallest experimentally obtained power level that makes a link feasible for rate index l. Denote the corresponding power level by $\tilde{P}_{min}(l)$ on the simulated Gaussian link.

Observe that the slopes of the ideal and experimental waterfall curves are similar up to 1% PER. At 10% PER, the combined non-ideality of the hardware and implementaion result in a maximum power loss of $10\log\frac{P_{min}(l)}{\tilde{P}_{min}(l)} \approx 3.5$ dB, i [12] (= 3 × 4) from the ideal results. Thus we take this "coaxial channel" to be a reasonable approximation of a Gaussian channel in our experiments.

Using these results we obtain the smallest feasible near fractions (l) for N as

$$(l) = \frac{P_{min}(l)}{P_{min}(12)}, \qquad l \quad [12]. \tag{13}$$

We use $(\ (l))$ as starting points to nd (l) in the rate region experiment in Section V-B.

D. Modi-cations to Implement an SC-BICM System

The SC encoder is implemented as two instances of a point-to-point BICM (sub-)encoder and a combiner. The encoded data is then transmitted via a standard OFDM modulator. The receiver is implemented as a single GNU Radio signal processing block, and contains a successive decoding block to decode the near user packet. More details on the transceiver operation including their block diagrams may be found in our technical report [16].

[15]This mimics a system with a 1 dB granularity in power control. Also note that the effective signal distortion seen at the receiver is usually higher than that due to thermal noise alone, due to additional sources of noise and distortion such as imperfect receive implementation and hardware imperfections. In our experiments an initial power level of 20 dB above the theoretical value was found to be adequate.

E. The Runtime Requirement

The GNU Radio runs on a general purpose computer. In our experiment, we use the desktop with two quad-core Intel Xeon CPU E5520@2.27 GHz with 8 GB RAM. The transmitter code takes approximately 2.1 ms for the transmitter to generate a single packet. For the far receiver, the decoding procedure takes approximately 8.7 ms. For the near receiver, it needs approximately 19.7 ms to decode its packet. The near user needs to do the decoding twice and encoding once so that the complexity is more than doubled compared to the far user. The time is measured by recording the timestamps before and after encoding(decoding) a packet. The near user needs to do the decoding twice and encoding once so that the complexity is more than doubled compared to the far user. In the experiment, the transmitter has an idle gap 100 ms between two packets. Hence the receiver has enough time to decode. More detailed discussion about implementation complexity is available in [16].

V. Experimental Results

A. Emulating a Gaussian BC

The time variation of the propagation loss is a bigger problem for the BC because of the presence of two Tx-Rx paths in the BC as opposed to just one in the point-to-point setting of Section IV-C. Moreover, checking the feasibility of candidate rate-pairs while solving (6) requires repetitive PER measurements. Thus we emulate a Gaussian BC using a combination of coaxial cables, a splitter, and an attenuator bank, as shown in Fig. 4. The USRPs shown communicate with three Linux PCs via USB 2.0. The PCs are con gured to run the appropriate Tx or Rx code in GNURadio.

The second step is to x the single-user SNRs $_N$ $_F$. In our experiment we x them implicitly by choosing a transmit power P_N and an attenuator setting $_F$.[16] In the single-user

[16]It is possible to obtain an estimate of $_N$ by measuring the signal and noise powers in the digital domain (see [16]).

mode, P_N is chosen to be P_{min} (12) (plus some additional loss due to the splitter) using the single user results from Section IV-C. The measured $_N$ was found to be 18 dB (see [16] for details on SNR measurement). With this value of P_N, we set F in the single-user mode to operate at a desired rate (e.g., QPSK, rate-5/6), and increase the attenuation in = 1 dB steps until the far link violates the PER constraint. is the receive SNR for the largest attenuation setting that supports a 10% (or lower) PER at F (for QPSK rate-5/6, this was found to be 10dB). With this approximation to the Gaussian BC, we are now ready to obtain the rate region for this system by solving (6) experimentally.

B. The Rate Region Experiment

Given P_N, a_F and a code library C with rate-indices l [M], we use the above experimental setup to solve (6) to find $(l, i_F(l))$ using the following procedure:

1) Initialize:
   a) $(l) = (l)$ where (l) is calculated from (13).
   b) $i_F(l) = \arg\max_l \{P_{min}(l) < (1 - (l))a_F P_N\}$.
2) Calculate the PER for $(l, i_F(l))$ for stream weights (l), $1 - (l)$.
3) If
   a) (feasible at N) AND (feasible at F): set $_l(l) = i_F(l)$, $k \quad l + 1$, go to Step 1.
   b) (feasible at N) AND (infeasible at F): $i_F(l) \quad i_F(l) - 1$, go to Step 2.
   c) (infeasible at N) AND (feasible at F): $(l) \quad (l) \times 10^{10}$, go to Step 2.
   d) (infeasible at N) AND (infeasible at F): $(l) \quad (l) \times 10^{10}$, $i_F(l) \quad i_F(l) - 1$, go to Step 2.

Here is the step size for the parameter (in this paper, we set = 1 dB).

C. Results

We use the procedure in Section V-B to study three interesting problems: (a) How does the measured rate region change with a_F (i.e., the far-link is made stronger or weaker)? (b) how much does imperfect interference cancellation at N affect the rate region?, and (c) what are good models to account for N's interference at F, and, in particular, how useful is the popular Interference-As-Gaussian-Noise (IAGN) model in predicting its impact? We discuss these problems in the following.

1) Changing the Strength of the Far-Link: To study this problem, we find the rate region for two possible far-link SNRs: $_F$ = 5 dB and $_F$ = 10 dB, which correspond to single-user rate indices K = 3 and K = 8 respectively. The near-user SNR for both cases is kept at $_N$ = 18 dB. These scenarios are emulated by using suitable attenuator values a_F = 9, 4 respectively[17]. The results are shown in Fig. 6. Here, we used a transmit power P_N = 43 dBm and step size = 1 dB. We clearly see dependence on the choice of a_F. With $_F$ = 5 dB, there is not enough disparity between the near- and far-links to fully benefit from superposition (F is "too close" to BS). This is in fact predicted by theory [1]. On the other hand, we see the effect of a finite code library when K = 3 (F is "too far" from BS): since its single-user rate is too small to begin with, interference from N's symbols rapidly degrades its link quality so as to make any far-rate infeasible.

Therefore, the far-user modulation and rate pair may be appropriately chosen based on the rate that the near user's traffic demands. For instance, when the near user's spectral efficiency is 1 (QPSK-1/2), choosing BPSK-3/4 for the far user provides a rate gain of about 28% over TD (as compared to a gain of about 2% over TD for QPSK-5/6), while when the near user's spectral efficiency is 3, it is preferable to choose QPSK-5/6 for the far user (over BPSK-3/4).

2) Impact of Imperfect Interference Cancellation at N: The deviation of (l) from its ideal value (l) is a measure of the residual far-user interference seen at N due to the imperfect cancellation of F's symbols (even when the far-packet is decoded correctly). The s are calculated from the single-user results using (13). The s are determined from the experiment. For K = 8, we find that when l = 6, (l) = 0.13 (l) = 0.1. At small near-rates the desired symbol stream (of near-symbols) has a much lower power than the interference symbol stream (of far-symbols). In this regime, even small regeneration errors manifest as large residual interference, necessitating an increase of (l) beyond (l). For example, 1 - (1) 90% of the transmit power is assigned to F. Even if only 10% of this power remains after cancellation, it is still about the same as the signal power. As (l) increases, so does the near-rate, thereby making the near-link susceptible to even smaller levels of residual interference. as well.

The root cause of this problem lies in the small (but non-zero) estimation error in the channel frequency response and a small error in compensating the carrier frequency and phase offsets. Although this level of inaccuracy may result in relatively small losses in a single-user system (as shown in Fig. 5), a multiuser system is much less tolerant to these errors as our results show. Despite this inaccuracy, we find that SC can still provide rate gains using reasonably well-designed single-user building blocks.

3) Modeling the Near-User Interference at the Far User: We study the performance of F's (maximum-likelihood) demodulator for the three choices of the interfering (N's) signal's constellations—BPSK, QPSK, 16QAM— and for two different interferer strengths = 0.2 and = 0.8. F's rate is maintained at BPSK-1/2. Indeed, as explained in Section III-B2, the performance of the demodulator is dictated by the superconstellation with $2^{b_N + b_F}$ points; its error probability critically depends on the effective minimum distance (10). Fig. 8 depicts the far user PERs versus the SINR at F which we define as

$$\text{SINR} \quad \frac{(1 -)}{1 + } \quad _F. \tag{14}$$

We observe the following:

- In the weak interference regime (for = 0.2, i.e., SIR 6 dB), it is seen that BPSK is the worst interferer. For small , the error probability arises primarily from the separation between the clusters (see Fig. 7 (left)).
- In the strong interference regime (e.g., = 0.8, i.e., SIR 6 dB), it is seen that 16QAM is the worst

[17] Note that a_F is not simply equal to $_N - _F$ owing to the two different boards and cables used, which had to be calibrated separately.

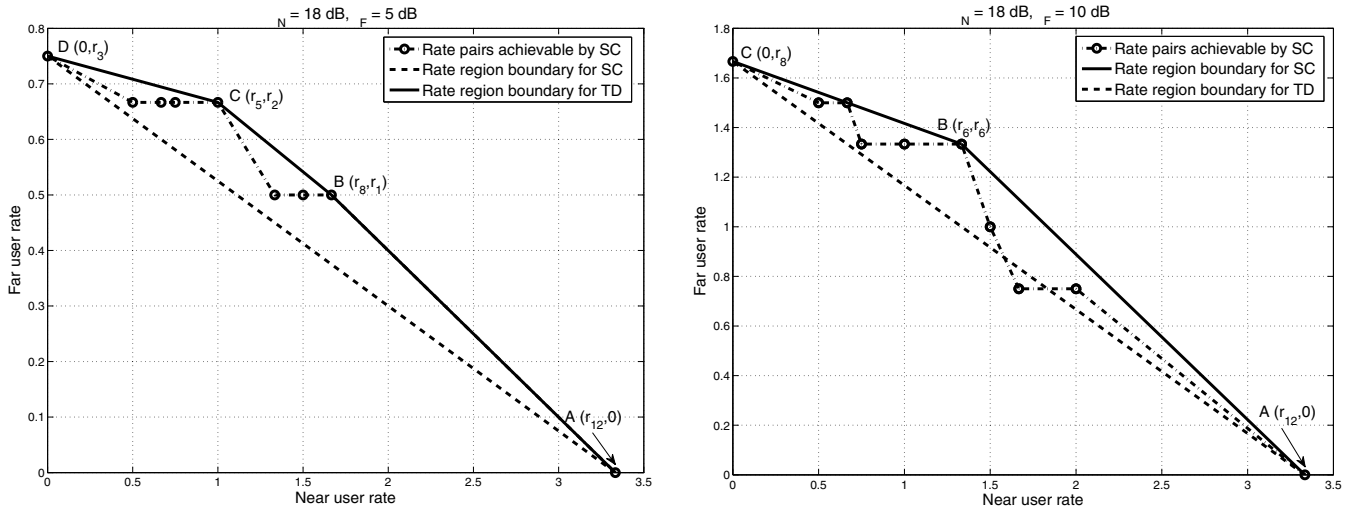


Fig. 6. Experimentally obtained rate region for the library of $M = 12$ codes using the setup shown in Fig. 4 for two different choices of F. (Left) The values of at corner points A-D are respectively 1.0 1.26 0.079 0. (Right) The values of at corner points A-C are respectively 1.0 1.26 0.

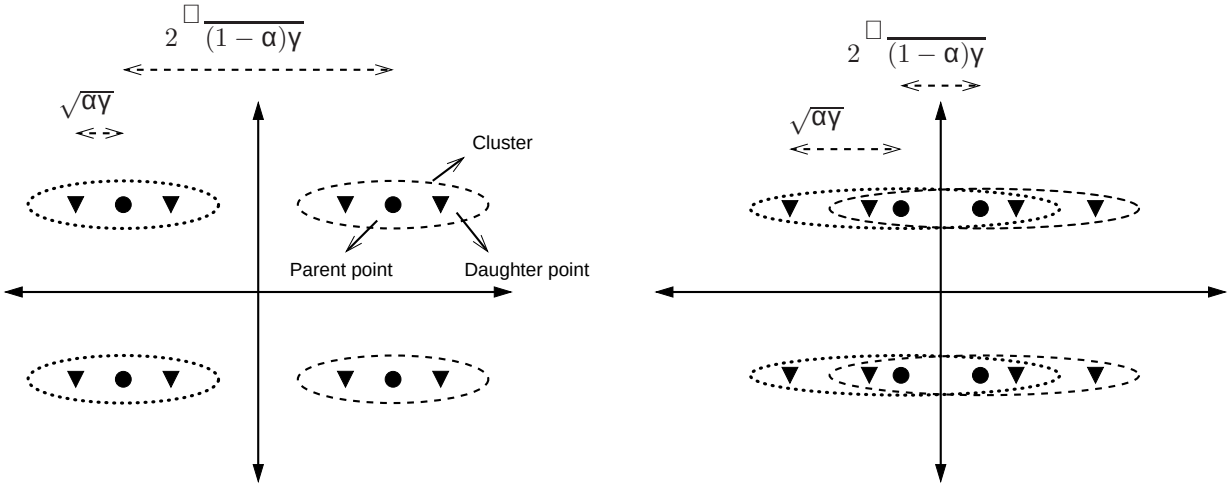


Fig. 7. Depiction of the superconstellation points in the weak (left) and strong (right) interference cases. Evidently, when is small, the PER depends on the cluster separation, while when is large, the PER is determined by the cluster density.

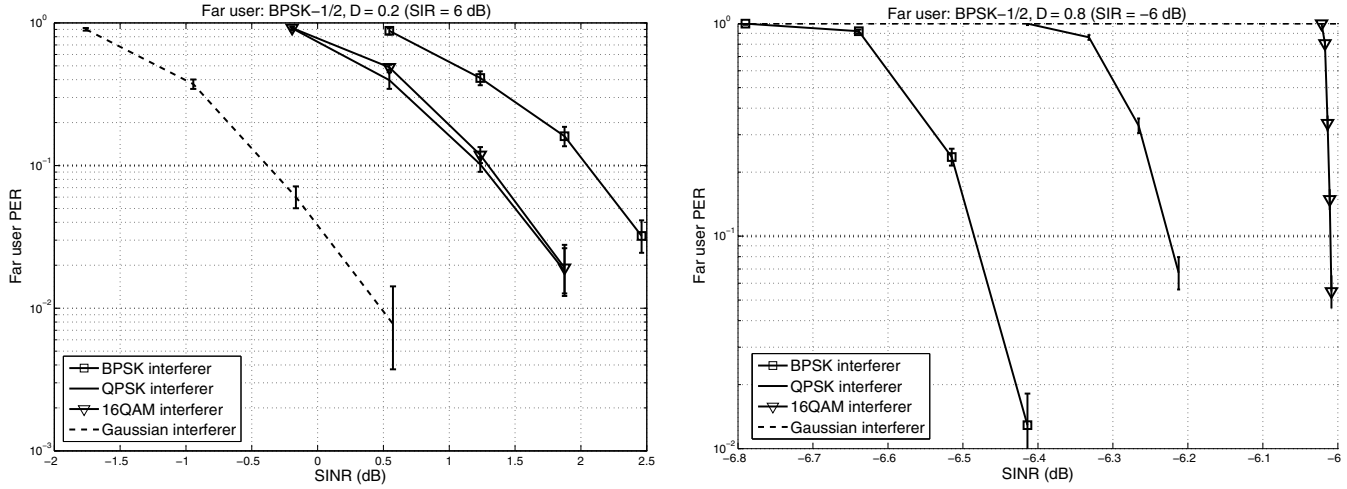
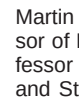


Fig. 8. Far user PER versus SINR for different near-user constellations in the weak (left) and strong (right) interference cases.



Martin Haenggi (S'95, M'99, SM'04) is a Professor of Electrical Engineering and a Concurrent Professor of Applied and Computational Mathematics and Statistics at the University of Notre Dame, Indiana, USA. He received the Dipl.-Ing. (M.Sc.) and Dr.sc.techn. (Ph.D.) degrees in electrical engineering from the Swiss Federal Institute of Technology in Zurich (ETH) in 1995 and 1999, respectively. After a postdoctoral year at the University of California in Berkeley, he joined the University of Notre Dame in 2001. In 2007-08, he spent a Sabbatical Year at the University of California at San Diego (UCSD). For both his M.Sc. and his Ph.D. theses, he was awarded the ETH medal, and he received a CAREER award from the U.S. National Science Foundation in 2005 and the 2010 IEEE Communications Society Best Tutorial Paper award.

He served as a member of the Editorial Board of the Elsevier Journal of Ad Hoc Networks from 2005-08, as a Guest Editor for the IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS in 2008-09, as an Associate Editor for the IEEE TRANSACTIONS ON MOBILE COMPUTING (TMC) from 2008-11 and for the ACM Transactions on Sensor Networks from 2009-11, and as a Distinguished Lecturer for the IEEE Circuits and Systems Society in 2005-06. He also served as a TPC Co-chair of the Communication Theory Symposium of the 2012 IEEE International Conference on Communications (ICC'12), and as a General Co-chair of the 2009 International Workshop on Spatial Stochastic Models for Wireless Networks and the 2012 DIMACS Workshop on Connectivity and Resilience of Large-Scale Networks. Presently he is a Steering Committee Member of TMC. He is a co-author of the monograph "Interference in Large Wireless Networks" (NOW Publishers, 2009) and the author of the textbook Stochastic Geometry for Wireless Networks (Cambridge University Press, 2012). His scientific interests include networking and wireless communications, with an emphasis on ad hoc, cognitive, cellular, sensor, and mesh networks.