Superposition Coding Strategies:
Design and Performance Evaluation
These slides summarize two papers:

- **Part 1 (A two-user superposition-coded system prototype):**

- **Part 2 (Coding gain from practical superposition codes):**
A Two-User SC System Prototype

The Coding Gains from Practical Superposition Codes
A Two-User SC System Prototype [Vanka12a]
What is Superposition Coding?

- BS sends information to *two* users N (near) and F (far)
  ↔ Communicating over a **Broadcast Channel** (BC)
- BS has full CSI: *Gaussian* BC [Cover06]¹
- BS has no CSI: *Fading* BC [Zhang09]²
- Capacity achieved by **Superposition Coding** (SC) and **Successive Decoding** (SD)

---

The Team Effort

The Team: Sundaram Vanka, Sunil Srinivasa, Peter Vizi, Zhenhua Gong, Kostas Stamatiou

Contributions

1. Superposition coding techniques that work for small to medium-sized packets (100 – 500 bytes).
2. Designed the complete SC physical layer.
3. Developed the reference Matlab model and provided extensive assistance in C code integration, testing and debugging.
4. Proposed practical approaches to leverage the coding gain from superposition-based multiuser channel codes.
5. Designed efficient experiments that measure performance gains from SC.
SC with Finite Blocklength Channel Codes

- IT result **existential**, not constructive
- Need to understand how SC works with well-known codes
- Identify key practical issues that arise in its implementation

**Definition (Code Library)**

A collection of $M < \infty$ encoder-decoder function pairs with spectral efficiencies (aka "rates") $r_1 < r_2 \cdots < r_M$

**Definition: Packet Error Rate (PER)**

The probability of codeword decoding error

**Definition ($\epsilon$–Feasible on a Link)**

A code with rate $r$ is $\epsilon$–feasible on a link if the PER of a codeword encoded at $r$ is no greater than $\epsilon$
Achievable Rates with a Code Library

- Need \((\gamma_n, \gamma_f)\) to specify BC
- Set \(\gamma_n\) s.t. \(r_n = r_M\) is feasible
- Set \(\gamma_f\) s.t. \(r_f = r_K < r_M\) is feasible
- For \(m = 1, 2, \ldots, M\)
  
  \[
  \max_{\{r_1, \ldots, r_M\}} r_f
  \]
  
  s.t. \((r_f, r_k)\) is jointly \(\epsilon\)-feasible

- Transmission scheme decides joint feasibility!
Convexify solution set \( \{(r_k, r_i^*(k)) : k \in [M] \} \) to get the rate-region boundary.

Solution requires finding desired Tx power split for SC

- \( \alpha_k \): N’s share for rate \( r_k \)
- \( \bar{\alpha}_k \triangleq 1 - \alpha_k \): F’s share for rate \( r_i^*(k) \)
The BICM Code Library

- Pairs powerful binary codes with well-known modulation techniques [Caire98]³
- Combines the advantages of signal space coding with well-known binary codes
- Flexible and easy to implement
- Coding technique in DSL, Wi-Fi, WiMAX...

In our library:
- Modulations: BPSK, QPSK, 16-QAM
- Channel codes:
  - Standard const. length 7 rate-1/2 convolutional code with generator matrix [133,171]
  - Rates 2/3, 3/4, 5/6 punctured versions of mother code

Can approximate PER as a function of SNR: PER at N:

- Pick $\gamma_n \gg \gamma_f$ so N can almost certainly decode F's packet
- If F's signal is *perfectly cancelled* at N, N decodes its packet from the matched filter outputs

$$Y_n(m) = \alpha X_n(m) + W_n(m), \ m \in [L]$$

- For this p2p case [Caire98]4

$$\text{PER}_n \lesssim NWQ \left( D_n \sqrt{\frac{C_n \gamma_n}{2}} \right), \ \gamma_n \to \infty.$$ 

- $N$: payload size, $W$: # free distance error events $D_n$: constellation min. distance, $C_n$: free distance of N's conv. code

---

F decodes its packet from

\[ Y_f(m) = \tilde{\alpha} X_f(m) + \alpha X_n(m) + W_n(m), \ m \in [L] \]

- Discrete interference
  \[ \implies \text{Symbol Clusters} \]
- ML demodulation for F’s symbols: Find the nearest cluster
- Expression similar to N, but "Constellation Min. Distance" = Closest cluster separation
Problem: Bound on $\text{PER}_f$ too loose at practical PERs ($\approx 0.1$), esp. for small intercluster separations

Numerically find the rate region

$\gamma_n = 18 \text{ dB}, \gamma_f = 8 \text{ dB}, \epsilon = 0.1$

<table>
<thead>
<tr>
<th>Point</th>
<th>$\alpha$</th>
<th>$(r_n, r_f)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>1</td>
<td>(3.33,0)</td>
</tr>
<tr>
<td>B</td>
<td>0.21</td>
<td>(2,1)</td>
</tr>
<tr>
<td>C</td>
<td>0.13</td>
<td>(1.5,1.33)</td>
</tr>
<tr>
<td>D</td>
<td>0.06</td>
<td>(1,1.5)</td>
</tr>
<tr>
<td>E</td>
<td>0</td>
<td>(1.67,0)</td>
</tr>
</tbody>
</table>
Towards an SC-BICM Prototype on a USRP Platform

- Flexible
  - Multi-Protocol
  - Multi-Band
- Board has FPGA, DAC/ADC, RF Frontends
- USB 2.0 Interface with Linux PC
- Software-based DSP on GNURadio
  - Open Source
  - In-built USRP drivers
Frame Structure

- TS1: Packet acquisition, timing and frequency sync. Duration $T_s = 48\, \mu s$
- TS2: Channel estimation. Duration $T_{ch} = 34\, \mu s$
Emulating a Gaussian BC

Step 1: Only BS→N active

- Increase BS power $P$, measure PER for highest rate
- Find $P_n = \text{Smallest } P \text{ s.t. highest rate is feasible}$
- Note down largest backoff $\beta_k$ from $P_n$ for rate $r_k$ to be feasible $k \in [M]$
- $\beta_k = \text{initial guess for } \alpha_k$
- Special case: PER curves (PER for all $P, r_k$)
Step 2: Only BS→F active

- Fix target rate $r_K$ for F
- With BS power = $P_n$, choose largest attenuation $a_f$ s.t. $r_K$ is feasible
- Power control granularity 0.5 dB, attenuator granularity 1 dB
- **SNR** = \( \frac{\text{Preamble power}}{\text{Noise power}} \)
- Digitally measured for fixed amplifier gain setting
- Worst-case implementation loss \( \approx 3.5 \) dB (16-QAM, rate-5/6 10% PER)
The Rate Region Experiment

Initialize $r_{prev} = r_K$
For $k = 1, \ldots, M$:

Step 1: $\alpha_k = \beta_k; r_f(k) = r_{prev}$

Step 2: Measure PER for $r_k, r_f(k)$

Step 3:

- N not feasible: Increase $\alpha_k$, go to 2)
- N but not F: $r_f(k) =$ Next lowest library rate, go to 2)
- Both N & F: $k^{th}$ solution found. $r_{prev} = r_f(k)$
The Choice of F

\[ P_n = -43 \text{ dBm}, \ a_f = 9 \text{ dB}. \]

F’s single user rate: BPSK-3/4

\[ P_n = -43 \text{ dBm}, \ a_f = 5 \text{ dB}. \]

F’s single user rate: QPSK-5/6

\[ \gamma_N = 18 \text{ dB}, \ \gamma_F = 5 \text{ dB} \]

\[ \gamma_N = 18 \text{ dB}, \ \gamma_F = 10 \text{ dB} \]

\[ \text{Near user rate} \quad \text{Far user rate} \]

Rate pairs achievable by SC
Rate region boundary for SC
Rate region boundary for TD

F is "too close" : Not enough disparity. "Too far": no codes to support rate

Sweet spot appears to be between QPSK-5/6 and BPSK-3/4
Interference from N’s Symbols at F

High SIR regime: $\alpha \ll 1 - \alpha$

- Fix F’s rate (in this case to BPSK-1/2)
- Compare equal-power Gaussian, BPSK, QPSK and 16-QAM interferers
- BPSK > QPSK/16QAM > Gaussian

SIR = 6 dB
Interference from N’s Symbols at F

Low SIR regime: $\alpha \gg 1 - \alpha$

Far user: BPSK−1/2, $\alpha = 0.8$ (SINR = −6 dB)

- BPSK interferer
- QPSK interferer
- 16QAM interferer
- Gaussian interferer

Now Gaussian > 16QAM > QPSK > BPSK

Situation reversed!

SIR = -6 dB
Why does this happen?

High SIR regime

- Small $\alpha$: min. distance determined by cluster separation
- For a given interference power BPSK perturbs all parent points to the max. extent
- Denser interferer constellations place fewer points on the edges
Why does this happen?

Low SIR regime

- Large $\alpha$: min. distance determined by cluster density
- For a given interference power BPSK causes the least dense clusters!
- Denser interferer constellations make the problem worse

Conclusion: Must be careful in using the Gaussian approximation in SC systems
The Coding Gain from Practical Superposition Codes [Vanka12b]
Orthogonal Coding on the BC

Min. link SNR independent of $u$!

Min. link SNR independent of $u$!
Constraining $(ur_n, \bar{ur}_f)$ to be feasible with SC

$$\gamma_f^*(\gamma_n; ur_n, ur_f) = \frac{\gamma_n e(\bar{ur}_f)}{\gamma_n - e(ur_n)(1 + e(\bar{ur}_f))}.$$ 

- Packets encoded **exactly** at $(ur_n, \bar{ur}_f)$
- For each $u$, require $\alpha > e(ur_n)/\gamma_n$ with SC
- Coding gain increases with $\gamma_n$ $\Leftrightarrow$ pair F with high-SNR N!
Performance Gain in the Finite Blocklength Regime

- Non-zero decoding error probability or Packet Error Rate (PER) $\epsilon$
- At PER $= \epsilon$, typical packet req. $\frac{1}{1-\epsilon}$ to reach F
- Easy to measure the Reliability Gain $\text{RG} = \frac{1-\epsilon_{SC}}{1-\epsilon_{TD}}$
SC with Finite Blocklength Channel Codes

- IT result **existential**, not constructive
- Need to understand how SC works with well-known codes
- Identify key practical issues that arise in its implementation

**Definition (Code library)**

A collection of $M < \infty$ encoder-decoder function pairs with spectral efficiencies (aka "rates") $r_1 < r_2 \cdots < r_M$

**Definition: Packet Error Rate (PER)**

The probability of codeword decoding error

**Definition ($\epsilon$–feasible on a link)**

A code with rate $r$ is $\epsilon$–feasible on a link if the PER of a codeword encoded at $r$ is no greater than $\epsilon$
Important special case: N close to BS, F at cell-edge.

- $r_n = r_M$, $\bar{u}r_f$ is small (can set to $r_1$)
- Set $ur_M = r_k$, so that
  
  $u_k = r_k/r_M$, $r_f = r_1/\bar{u}_k$, $k \in \{1, \ldots, M\}$
- If library has codes $r_a < r_f < r_b$, time-share between $r_a$ and $r_b$

Compare SC using $(r_k, r_1)$ with TD using $(r_M, r_1/u_k)$, for $k = 1, \ldots, M$. 
Setting up the BC

$P$: BS power, $\alpha$: N’s share

$$\gamma_n \propto \alpha P \triangleq P_n$$
$$\gamma_f \propto \bar{\alpha} P \triangleq P_f$$

Rate $r$ is reliable $\leftrightarrow$ $\text{PER} \lesssim 0.1$

For $k = 1, \ldots, M$:

**SC Step:**

**Step 1:** Set $P_n = 0$ & $\uparrow P_f$ s.t. $r_1$ is reliable

**Step 2:** $\uparrow P_n$ s.t. $r_k$ is reliable

**Step 3:** Keeping $P_n/P_f$ constant $\uparrow P_f$ s.t. $r_1$ is reliable

**TD Step:** Find $\text{PER}_f$ at BS power $P_n + P_f$ and rate $r_1/u_k$
Experimental Results

- $\bar{\nu} r_f = 0.5$ [bps/Hz], SC always uses BPSK-1/2
- 16QAM-5/6 always feasible at N with full power
- SC adjusts N’s power and code to provide the same rate as TD

<table>
<thead>
<tr>
<th>SC</th>
<th>TD</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\gamma_f$ (dB)</td>
<td>SIR (dB)</td>
</tr>
<tr>
<td>8.8</td>
<td>1</td>
</tr>
<tr>
<td>7.4</td>
<td>1.95</td>
</tr>
<tr>
<td>5.5</td>
<td>5</td>
</tr>
<tr>
<td>4.3</td>
<td>5</td>
</tr>
<tr>
<td>2.7</td>
<td>6</td>
</tr>
<tr>
<td>2.6</td>
<td>7.5</td>
</tr>
</tbody>
</table>
Conclusions

- Experimentally demonstrated a practical approach to exploit superposition codes
- Specific decoding strategies such as demod-and-decode can render the Gaussian approximation for inter-user interference inaccurate
- Signal superposition opens up new possibilities for link-layer scheduling policies [Vizi11]\(^5\)

---