Introduction to Turbo Coding and Turbo Detection

• Armed with soft-input soft-output decoding, powerful turbo code was born
  – We briefly discuss parallel-concatenated turbo coding

• Iterative or turbo principle is more general than just for channel coding: iterative decoding-detection, iterative timing recovery-detection, iterative equalisation, etc.
  – We briefly discuss serial-concatenated turbo detection
Turbo Code: Introduction

• Turbo encoder consists of two usually identical simple encoders
  – Interleaver makes input bits to encoder 2 look “different” to input of encoder 1
  – Puncturing and multiplexing achieves required combined rate

• For example, half rate recursive systematic convolutional code RSC CC(2, 1, 2)
  – Feedforward generator polynomial $G_{RSC}^f = [1 \ 0 \ 1]_2$
  – Feedback generator polynomial $G_{RSC}^r = [1 \ 1 \ 1]_2$
  – Single RSC $CC'(2, 1, 2)$ is not powerful, but with two parallel concatenated, ⇒ very powerful turbo code

• Two parallel concatenated decoders,
  – each accepts soft channel output and information from the other decoder as a priori LLRs
  – and outputs a posteriori LLRs
  – LLRs are properly interleaved or deinterleaved
Turbo Decoding: Introduction

- Two component decoders exchange extrinsic information – extrinsic information outputed by one component becomes *a priori* information for the other component, and they iterate a few times
  - It is this **turbo principle** that makes turbo code, which is obtained by two parallel concatenated weak components, very powerful
- Each component decoder may for example be implemented as a soft-output Viterbi algorithm
  - Recall **slide 112**, soft-input bit decisions to Viterbi algorithm are provided by soft demapper as LLRs, at each iteration, together with *a priori* LLRs provided by the other decoder,
  - it calculates *a posteriori* LLRs, which can be expressed as

\[
L_p(b_k) = L_e(b_k) + L_a(b_k)
\]

- Since *a priori* LLR came from the other decoder, it would be **redundant** to sent it back to the other component decoder, therefore, it is taken out from *a posteriori* LLR to yield

\[
L_e(b_k) = L_p(b_k) - L_a(b_k)
\]

- After interleaver/deinterleaver, extrinsic LLR \(L_e(b_k)\) is sent to the other component as *a priori* LLR for the next iteration
- At first iteration, since there is no *a priori* information, or \(b_k = 0\) and \(b_k = 1\) are equiprobable, all *a priori* LLRs are set to zero
Turbo Decoding-Detection: Motivation

- Consider transceiver structure we have studied so far

Each component operates separately/independently from others

1. For example, soft detector/demapper of slide 112 provides LLRs of soft bit decisions to channel decoder
2. Decoder (may iterates a few times itself if is turbo code) to provide final hard-bit decisions

- To attain high performance, channel code employed must be high power, such as powerful low density parity check (LDPC) codes or turbo codes
  - These powerful channel codes have high decoding complexity
  - Simple RSC CC(2, 1, 2) however could not offer high performance

- Alternative: use simple code, e.g. RSC CC(2, 1, 2), but let detector/demapper and channel decoder iterate a few times
  - This turbo principle can boost the achievable performance
  - Two components involved are serial-concatenated
Two-Stage Turbo System

- Transmitter: usual channel encoder and modulator/mapper, separated by an interleaver
  - e.g. simple RSC CC(2, 1, 2) of slide 206
  - Channel encoding introduces correlation
  - **Interleaver** makes coded bits to modulator/mapper more random

- Recall slide 111, for $M = 2^n$-ary constellation $\mathcal{X} = \{\bar{x}_1, \bar{x}_2, \ldots, \bar{x}_M\}$, mapper at transmitter maps every $n$ bits to a symbol: $\{u_0, u_1, \ldots, u_{n-1}\} \rightarrow x \in \mathcal{X}$
  - Look at $i$th bit $u_i$, and divide $\mathcal{X}$ into two subsets with $u_i = 0$ and $u_i = 1$, respectively
    $\mathcal{X} = \mathcal{X}_i^{(0)} \cup \mathcal{X}_i^{(1)}$
  - Received signal sample is $y = g_0 x + \varepsilon$
    - $g_0$: CSI, $\varepsilon$: channel AWGN sample with power $N_0$

- Turbo receiver: SISO detector/demapper and SISO channel-decoder exchange extrinsic information a number of iterations
  - $L_{a,d}$, $L_{p,d}$, $L_{e,d}$: detector/demapper $a$ priori, $a$ posteriori, extrinsic LLRs
  - $L_{a,c}$, $L_{p,c}$, $L_{e,c}$: channel decoder $a$ priori, $a$ posteriori, extrinsic LLRs
    $\hat{x}$, $\hat{b}$
Iterative Detection-Decoding

- **SISO channel decoder**: e.g. soft-output Viterbi algorithm of slides 202 and 203
  - Each iteration, decoder accepts *a priori* LLRs $L_{a,c}(z_i)$ from detector/demapper, and computes *a posteriori* LLRs, expressed as: $L_{p,c}(z_i) = L_{e,c}(z_i) + L_{a,c}(z_i)$
  - At first iteration, there are no *a priori* information, all $L_{a,c}(z_i) = 0$
  - Extrinsic LLRs $L_{e,c}(z_i)$ of decoder after interleaver become *a priori* LLRs $L_{a,d}(u_i)$ to detector/demapper for the next iteration

- **Iterative demapper** accepts *a priori* LLRs $L_{a,d} = [L_{a,d}(u_0) \ L_{a,d}(u_1) \cdots L_{a,d}(u_{n-1})]^T$ from channel decoder at each iteration, and computes *a posteriori* LLRs

  $$L_{p,d}(u_i) = -\frac{1}{N_0} \left( \min_{x \in \mathcal{X}_i(0)} \left\{ |y - g_0 x|^2 \frac{2}{N_0} s_x^T L_{a,d} \right\} - \min_{x \in \mathcal{X}_i(1)} \left\{ -|y - g_0 x|^2 \frac{2}{N_0} s_x^T L_{a,d} \right\} \right)$$

  - where $s_x = [1 - 2u_{x_0} \ 1 - 2u_{x_1} \cdots 1 - 2u_{x_{n-1}}]^T$ and $\{u_{x_0}, u_{x_1}, \cdots, u_{x_{n-1}}\} \rightarrow x$
  - At first iteration, there are no *a priori* information, all $L_{a,d}(u_i) = 0$, and iterative demapper becomes non-iterative demapper of slide 112
  - Extrinsic LLRs $L_{e,d}(u_i) = L_{p,d}(u_i) - L_{a,d}(u_{n})$ after deinterleaver becomes *a priori* LLRs $L_{a,c}(z_i)$ to decoder for the next iteration

- After a few iterations between SISO channel decoder and SISO detector/demapper, the process converges, and decoder can output hard bit decisions $\{\hat{b}_i\}$
Two-Stage Turbo System: Performance

- A reference of soft demapper for iterative detection-decoding:

- Convergence performance analysis: extrinsic information transfer (EXIT) charts

- Typical performance of turbo detection-decoding
  - Dramatically boost achievable performance after a turbo cliff
  - Even with a powerful channel code, two-stage turbo system’s performance may still be far away from capacity line

- In order to achieve near-capacity, i.e. only a few dBs away from capacity line:
  - we need three-stage turbo system
Three-Stage Turbo System: Transmitter

- Transmitter consists of three serial-concatenated components: recursive systematic code (RSC) encoder, unitary rate code (URC) encoder, and modulator/mapper, separated by two interleavers.

- Low-complexity memory-1 URC has an IIR:
  - $G_{URC} = [1 \ 0]_2$, $G_{URC}^r = [1 \ 1]_2$
  - allows system to spread extrinsic information beneficially across iterative decoder components without increasing its delay
  - EXIT curve capable of reaching $(1.0, 1.0)$ point of perfect convergence to vanishingly low BER
  - Necessary for near-capacity performance operation and eliminating error floor

- With this three-stage structure, we do not need high-complexity powerful channel code in order to attain near-capacity performance:
  - Half rate RSC $CC(2, 1, 2)$ may be sufficient
  - Feedforward generator polynomial $G_{RSC} = [1 \ 0 \ 1]_2$
  - Feedback generator polynomial $G_{RSC}^r = [1 \ 1 \ 1]_2$
Three-Stage Turbo Detection-Decoding

- Three-stage iterative receiver structure, where $E(\cdot)$: extrinsic LLR, $A(\cdot)$: a priori LLR

- Inner decoder: soft detector/demapper and URC decoder exchange extrinsic information $I_{\text{in}}$ times, and operations are as described in slides 209 and 210

- Outer decoder loop: RSC decoder and composite inner decoder of detector/demapper and URC decoder iterate or exchange extrinsic information $I_{\text{out}}$ times
  - Each outer iteration, extrinsic LLRs $E(u_1)$ of URC decoder after deinterleaver become a priori LLRs $A(z_1)$ to RSC decoder
  - RSC decoder produces a posteriori LLRs, subtracts a priori LLRs $A(z_1)$ from them to yield extrinsic LLRs $E(z_1)$
  - Extrinsic LLRs $E(z_1)$ of RSC decoder after interleaver become a priori LLRs $A(u_1)$ to inner decoder, and inner detection-decoding procedure restarts

- At end of iterative procedure, hard bit decisions $\{\hat{b}\}$ are produced
An Illustration Example

- $4 \times 4$ MIMO with 4-QAM, transmitted signal power normalised to unity, and hence $\text{SNR} = \frac{1}{N_0}$
- Open tunnel exists between EXIT curves of inner and outer decoders at $\text{SNR} = -2.5 \text{ dB}$
- Performance is only 2.2 dB away from capacity
Summary

• We have briefly introduced turbo codes
  – Two parallel concatenated simple component decoders exchange extrinsic information a few times to produce “turbo” effort

• Turbo principle can be applied to serial concatenated simple components, and we have briefly introduced turbo detection-decoding
  – Two-stage serial concatenated turbo detection-decoding
  – Three-stage serial concatenated turbo detection-decoding for near capacity operation with low delay