LDPC Code Ensembles that Universally Achieve Capacity under Belief Propagation Decoding

A Simple Derivation

Anatoly Khina
Tel Aviv University

Joint work with:
Yair Yona, UCLA
Uri Erez, Tel Aviv University

ISIT 2015
Hong Kong
June 16, 2015

Requirements of a "Good Code"

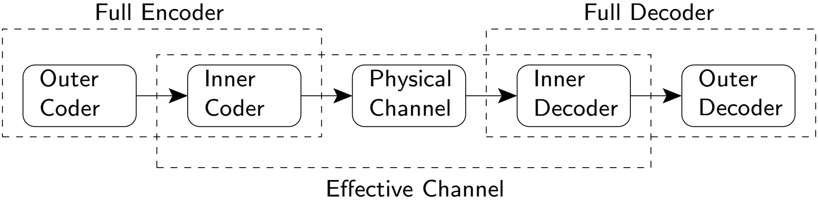
- Capacity achieving (universally?)
- Low computational complexity for encoding/decoding (linear)?
- Good performance in practice (BP?)

Known Code Constructions

Random codes [Shannon '48]

- Capacity achieving ✓
- Universal for **B**inary-Input **M**emoryless Output-**S**ymmetric (BMS) channels with same capacity ☺
- Exponential encoding/decoding complexity X
- Definitely impractical ☹

Known Code Constructions



Concatenated codes [Forney '61]

- Approach capacity ✓
- Can be extended to work universally over BMS channels
- Reduced (polynomial) complexity
- Works well in practice ✓

Known Code Constructions

Concatenated + Outer Expander Code [Barg–Zémor '04][Guruswami-Indyk '05]

- Approach capacity ✓
- Linear complexity ✓
- Can be extended to work universally over BMS channels
- Considered impractical X

Polar codes [Arıkan '09]

- Approach capacity ✓
- Low complexity: $\mathcal{O}(N \log N)$
- Practical? – Getting there...

Known Code Constructions

Recent (re-)entry: Convolutional/spatially-coupled LDPC codes [Felström–Zigangirov '99]

- Linear decoding complexity under BP decoding ✓
- Approach capacity over BMS channels [Kudekar et al. ISIT'12]
- Performance in practice? Good for long blocklengths
- Overall code performance under BP decoding

  $$=$$

  Shorter regular LDPC code performance under ML decoding
- Threshold saturation [Kudekar et al. ISIT'12]

Recapitulation

Several different goals

- Capacity achieving (for a given BMS)
- Universality
- Low complexity in practice (BP?)

Concatenated Convolutional/LDPC-BC: Overview

- LDPC ensemble (different from spatially-coupled)

Concatenated Convolutional/LDPC-BC: Overview

- LDPC ensemble (different from spatially-coupled)
- Achieves capacity under BP (linear complexity) ✓

Concatenated Convolutional/LDPC-BC: Overview

- LDPC ensemble (different from spatially-coupled)
- Achieves capacity under BP (linear complexity) ✓
- Simple “black box” analysis

Concatenated Convolutional/LDPC-BC: Overview

- LDPC ensemble (different from spatially-coupled)
- Achieves capacity under BP (linear complexity) ✓
- Simple “black box” analysis
- Follows classical code-concatenation approach

Concatenated Convolutional/LDPC-BC: Overview

- LDPC ensemble (different from spatially-coupled)
- Achieves capacity under BP (linear complexity) ✓
- Simple "black box" analysis
- Follows classical code-concatenation approach
- Builds upon the extremal properties of BEC and BSC:

Concatenated Convolutional/LDPC-BC: Overview

- LDPC ensemble (different from spatially-coupled)
- Achieves capacity under BP (linear complexity) ✓
- Simple "black box" analysis
- Follows classical code-concatenation approach
- Builds upon the extremal properties of BEC and BSC:
  - Inner code – should have good ML performance over BSC

    $\Downarrow$

    Convolutional code (BCJR = BP!)

Concatenated Convolutional/LDPC-BC: Overview

- LDPC ensemble (different from spatially-coupled)
- Achieves capacity under BP (linear complexity) ✓
- Simple "black box" analysis
- Follows classical code-concatenation approach
- Builds upon the extremal properties of BEC and BSC:
  - Inner code – should have good ML performance over BSC

    $\Downarrow$

    Convolutional code (BCJR = BP!)
  - Universality guarantee is immediate from **random ensemble + extremal properties of BSC** ✓

Concatenated Convolutional/LDPC-BC: Overview

- LDPC ensemble (different from spatially-coupled)
- Achieves capacity under BP (linear complexity) ✓
- Simple "black box" analysis
- Follows classical code-concatenation approach
- Builds upon the extremal properties of BEC and BSC:
  - Inner code – should have good ML performance over BSC

    $\Downarrow$

    Convolutional code (BCJR = BP!)
  - Universality guarantee is immediate from **random ensemble + extremal properties of BSC** ✓
  - High rate outer **LDPC** code – should be good for the BEC

Concatenated Convolutional/LDPC-BC: Overview

- LDPC ensemble (different from spatially-coupled)
- Achieves capacity under BP (linear complexity) ✓
- Simple "black box" analysis
- Follows classical code-concatenation approach
- Builds upon the extremal properties of BEC and BSC:
  - Inner code – should have good ML performance over BSC

    $\Downarrow$

    Convolutional code (BCJR = BP!)
  - Universality guarantee is immediate from **random ensemble + extremal properties of BSC** ✓
  - High rate outer **LDPC** code – should be good for the BEC
  - LDPC behavior over BEC guarantees suffice [Khandekar '02][Burshtein–Miller '02]

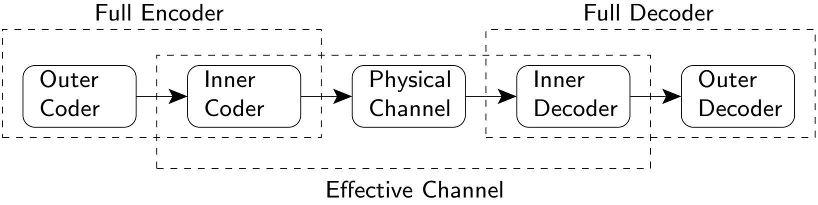
Concatenated Convolutional/LDPC-BC: Overview

- LDPC ensemble (different from spatially-coupled)
- Achieves capacity under BP (linear complexity) ✓
- Simple "black box" analysis
- Follows classical code-concatenation approach
- Builds upon the extremal properties of BEC and BSC:
  - Inner code – should have good ML performance over BSC

    $\Downarrow$

    Convolutional code (BCJR = BP!)
  - Universality guarantee is immediate from **random ensemble + extremal properties of BSC** ✓
  - High rate outer **LDPC** code – should be good for the BEC
  - LDPC behavior over BEC guarantees suffice [Khandekar '02][Burshtein–Miller '02]
  - **Regular LDPC / IRA** codes can be used

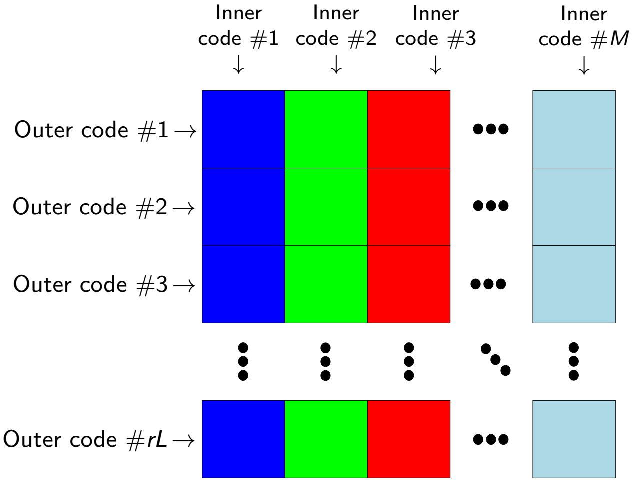
Encoder



Concatenated encoder

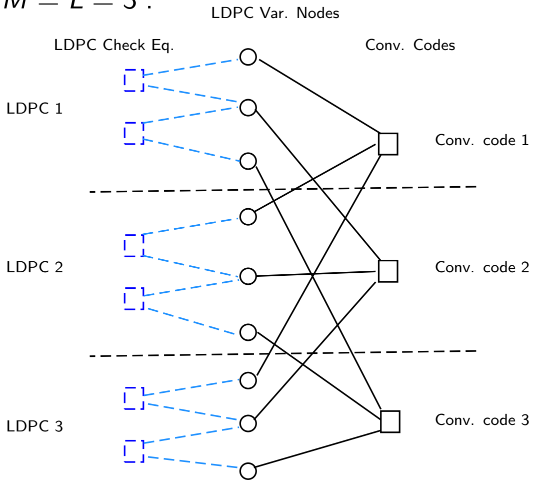
- Encode info. bits using an (outer) LDPC code of length M
- [Interleave LDPC coded bits]
- Encode LDPC coded bits using an (inner) zero-terminated convolutional code of length L

Interleaver



Decoder: BP Decoding over Overall Factor Graph

Example of $M = L = 3$:



- Factor graphs of LDPC codes
- Factor graphs of convolutional codes (state-space representation)

"Degraded" Decoder

Two-stage message passing decoder

- Apply BCJR decoding to the inner convolutional code
  $\Rightarrow$ Calculate LLRs of each input bit
- Apply slicer to get hard decisions (for sake of analysis)
- [De-interleave LLRs]
- Apply BP decoding of LDPC code over induced BSC channel
- [The induced channel is memoryless due to the interleaver]

(Full) BP decoding

BP decoder outperforms two-stage decoder (under tree assumption)

Concatenated Convolutional/LDPC-BC: Questions

- Capacity achieving?
- Universal?
- Practical?

Universality of (Capacity-Achieving) Block Codes

Exponential upper bound on block error probability [Gallager '68]

For a BMS c of capacity C:

$$P_b^{(c)} \leq e^{-N\, E_G(r)}$$

- Achievable by a random (block) code
- $E_G(r) > 0$ for $r < C$
- Upper bounds also the BER

Extremes of Error Exponents (EE) [Guillen i Fabregas et al. '13]

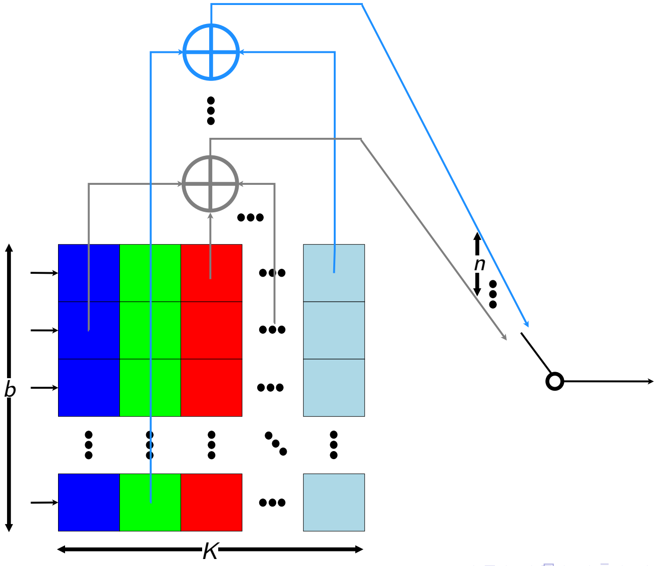
EE of BSC is worse than EE of any other BMS with same capacity:

$$E_G^{(c)}(r) \geq E_G^{\mathrm{BSC}}(r)$$

Conclusion for random block codes

Random codes designed for BSC(C) achieve better BER over all BMS(C)

Universality of Convolutional Codes



Universality of Convolutional Codes

Upper bound on BER of convolutional codes [Yudkin '65][Viterbi '67]

For a BMS c of capacity C and any $0 < \epsilon < 1$:

$$P_b \le \left(2^b - 1\right) \frac{2^{-K \frac{b}{r} E_{\text{VY}}(r,\epsilon)}}{\left[1 - 2^{-\epsilon \frac{b}{r} E_{\text{VY}}(r,\epsilon)}\right]^2} \triangleq P_b^{\text{UB}}$$

- Achievable by a random time-varying convolutional code
- $E_{\text{VY}}(r,\epsilon) > 0$ for $r < C(1-\epsilon)$
- Register length K plays the role of the blocklength

Crude lower bound on Viterbi–Yudkin EE

BSC VY EE is the worst and lower bounded by BSC block-code EE:

$$E_{\text{VY}}^{(c)}(r,\epsilon) \ge E_{\text{VY}}^{\text{BSC}}(r.\epsilon) \ge E_G^{\text{BSC}}\left(\frac{r}{1-\epsilon}\right) > 0$$

Conclusion for random (time-varying) convolutional codes

Random codes designed for BSC(C) achieve better BER over all BMS(C)

Performance of LDPC codes over BMS

LDPC over BEC

LDPC ensembles that approach capacity over the BEC under BP decoding are known [Luby et al. "97][Shokrollahi '01], . . .

LDPC performance over BMS [Khandekar '02]

BER of LDPC ensemble over BMS with Bhattacharyya param. B

$$\leq$$

BER of LDPC ensemble over BEC with erasure probability B

Conclusion

LDPC ensembles for BEC guarantee performance over BMS with same B

Analysis of Two-Stage HDD + Message-Passing Decoder

- Total rate: $R = C - \Delta$, for any $\Delta > 0$
- Convolutional code rate: $r \in (R, C)$
- Register length K is taken long enough (but fixed!), s.t.

$$0 < \underbrace{2\sqrt{P_b^{\text{Hard}}\left[1 - P_b^{\text{Hard}}\right]}}_{\text{Bhattacharyya of BSC}\left(P_b^{\text{Hard}}\right)} \triangleq B^{\text{Hard}} < \underbrace{\overbrace{1 - \frac{R}{r}}^{\epsilon_{\text{BEC}}} - \delta}_{\text{Threshold over BEC}}$$

- Use LDPC ensemble of
  - Rate $= R/r$
  - Threshold over BEC $> B^{\text{Hard}}$

Analysis of Two-Stage Message-Passing Decoder: Cont.

Using **regular** LDPC codes

- Regular LDPC ensemble:
  Variable- and check-nodes have degrees d_v and d_c
- Rate$=1 - d_v/d_c$
- Has a threshold T^{BEC} bounded away from zero
- High rate code $\rightarrow$ Approaches capacity

Conclusion:

- Take long enough register length K s.t. $B^{\text{Hard}} < T^{\text{BEC}}$
- Achieves performance at least as good as LDPC of blocklength length over BEC

Using IRA codes [Jin-Khandekar-McEliece ISIT'00]

- Performance guarantee via Bhattacharyya parameter holds for IRA
- Using IRA codes achieves $\Rightarrow$ **Systematic representation**
  Linear encoding

Conclusions

- Capacity-achieving with BP ✓
- Universal ✓
- Practical?
  - Who does the heavy lifting? Inner code or outer code...?
  - In practice: Use one long CC instead of multiple copies
  - Construction is actually a variation of serial turbo codes [Benedetto et al. '98]
    - Outer CC is replaced with LDPC code

Puncturing: Using Low-Rate LDPC Codes

- The presented scheme uses:
  - High-rate LDPC code (rate ≈ 1)
  - CC of rate $\approx C$

"Rate switch"

- Can start with low-rate LDPC code and puncture it
- Declare LDPC symbols in pre-determined positions as "erasures"
- Encode the rest using the convolutional coder
- Inner code $=$ CC $+$ "punctures": Rate ≈ 1
- Outer LDPC code of rate $\approx C$