Fundamental bounds for power consumption at the physical layer: "waterslide curves" and the price of certainty

Anant Sahai
based on joint work with student
Pulkit Grover

Wireless Foundations
Department of Electrical Engineering and Computer Sciences
University of California at Berkeley

Support from NSF and Sumitomo Electric

LIDS Seminar: Apr 28th, 2008

Shannon tells us:

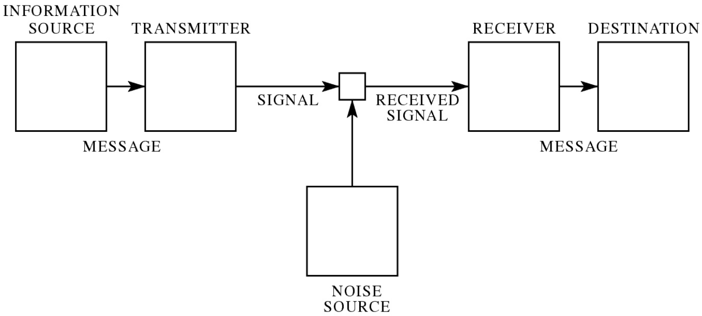


Fig. 1 — Schematic diagram of a general communication system.

Shannon tells us:

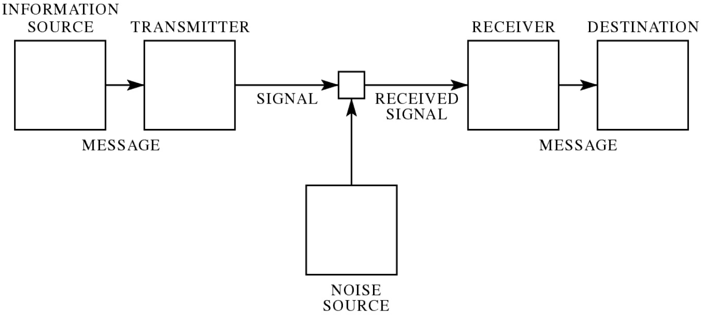


Fig. 1 — Schematic diagram of a general communication system.

- Delay: needed for laws of large numbers to apply

Shannon tells us:

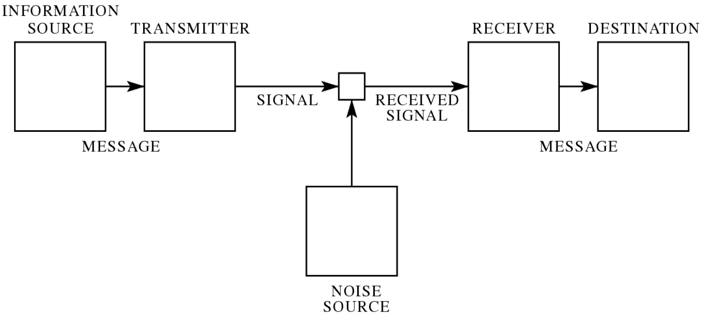
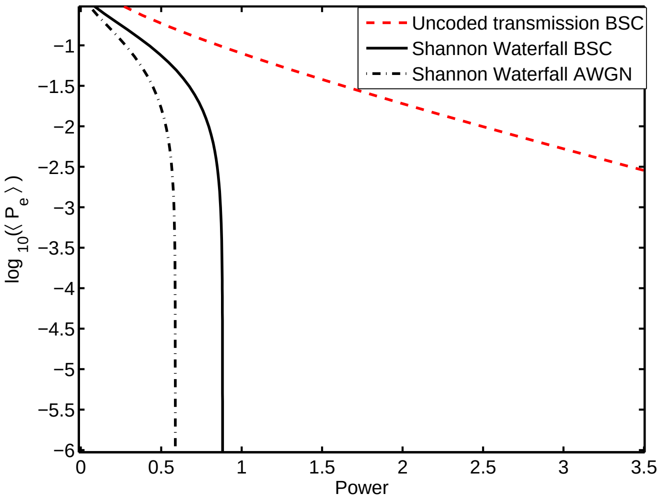


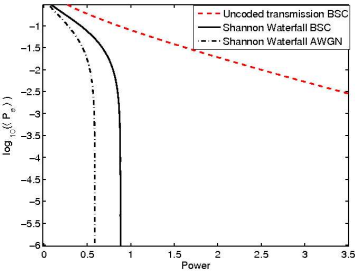
Fig. 1 — Schematic diagram of a general communication system.

- Delay: needed for laws of large numbers to apply
- Power: needed to apply the laws of large numbers

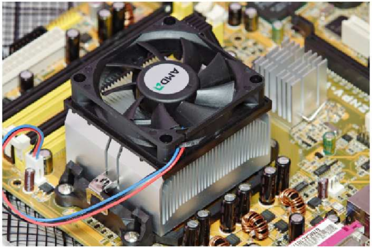
The promise of the waterfall curve



The new problem

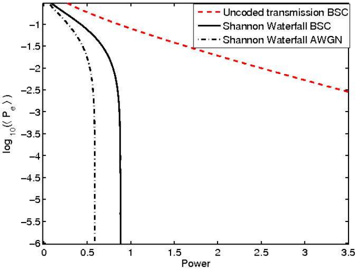


+

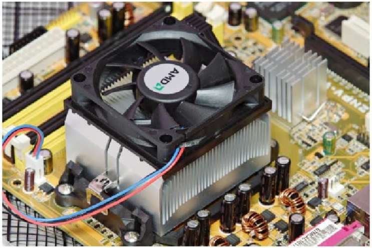


- Classical goal: arbitrarily low probability of error.
- Classical assumption: not delay sensitive at all.
- New twist: minimize **total power consumption**

The new problem

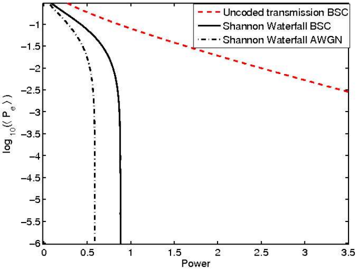


+

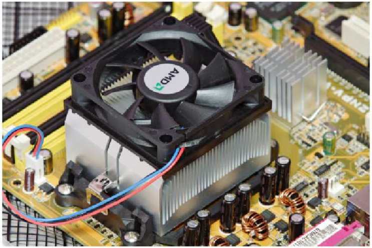


- Classical goal: arbitrarily low probability of error.
- Classical assumption: not delay sensitive at all.
- New twist: minimize **total power consumption**
- Important technology trends

The new problem

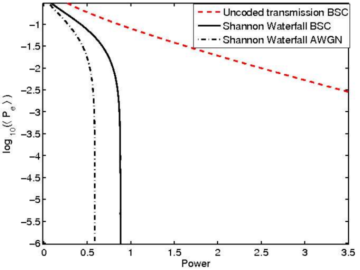


+

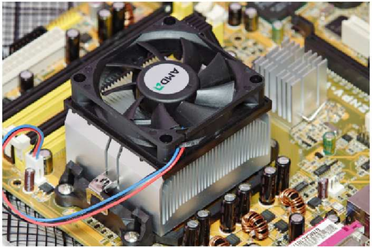


- Classical goal: arbitrarily low probability of error.
- Classical assumption: not delay sensitive at all.
- New twist: minimize **total power consumption**
- Important technology trends
  - "Moore's law" allows billions of transistors, and but only mildly reduces power-consumption per transistor.

The new problem



+



- Classical goal: arbitrarily low probability of error.
- Classical assumption: not delay sensitive at all.
- New twist: minimize **total power consumption**
- Important technology trends
  - "Moore's law" allows billions of transistors, and but only mildly reduces power-consumption per transistor.
  - New short-range applications: in-home networks, dense meshes, personal-area networks, UWB, between-chip communication, on-chip communication, etc.

Review of key prior work

- *Ephremides*, Wireless Communications Magazine, 2002
  - Power consumption crucial across networking layers
  - Many related papers.

Review of key prior work

- *Ephremides*, Wireless Communications Magazine, 2002
  - Power consumption crucial across networking layers
  - Many related papers.
- *Howard, Schlegel, and Iniewski*, EUARASIP Wireless Comm/Net, 2006
  - Empirical study
  - Found uncoded is often better

Review of key prior work

- *Ephremides*, Wireless Communications Magazine, 2002
  - Power consumption crucial across networking layers
  - Many related papers.
- *Howard, Schlegel, and Iniewski*, EUARASIP Wireless Comm/Net, 2006
  - Empirical study
  - Found uncoded is often better
- *Cui, Goldsmith, Bahai*, Journal of Wireless Comm, 2005
  - Semi-empirical with uncoded modulation
  - Found larger-constellations and higher rates better
- *Massad, Medard, and Zheng*, ISITA 2004
  - Information-theoretic with constant decoder power
  - Found better to use higher rates

Review of key prior work

- *Ephremides*, Wireless Communications Magazine, 2002
  - Power consumption crucial across networking layers
  - Many related papers.
- *Howard, Schlegel, and Iniewski*, EUARASIP Wireless Comm/Net, 2006
  - Empirical study
  - Found uncoded is often better
- *Cui, Goldsmith, Bahai*, Journal of Wireless Comm, 2005
  - Semi-empirical with uncoded modulation
  - Found larger-constellations and higher rates better
- *Massad, Medard, and Zheng*, ISITA 2004
  - Information-theoretic with constant decoder power
  - Found better to use higher rates
- *Bhardwaj and Chandrakasan*, Allerton 2007
  - Focus on receiver sampling cost in UWB
  - Found lower-rates and adaptive sampling is better

Outline

1. Motivation and introduction
2. Classical results revisited
3. A model for decoder power consumption
4. General lower bounds
5. Asymptotic behavior near capacity
6. Optimal choice of rate

What should capacity-achieving mean?

$$\min \xi_T P_T + \xi_C P_C + \xi_D P_D$$

- ξ_T: net path loss (e.g.$\approx$ 86dB for short-range)
- ξ_C, ξ_D choice of weights for encoder and decoder power.

What should capacity-achieving mean?

$$\min \xi_T P_T + \xi_C P_C + \xi_D P_D$$

- ξ_T: net path loss (e.g.$\approx$ 86dB for short-range)
- ξ_C, ξ_D choice of weights for encoder and decoder power.
- Assume $P_e \longrightarrow 0$
- What happens to optimizing P_T, P_C, P_D?

What should capacity-achieving mean?

$$\min \xi_T P_T + \xi_C P_C + \xi_D P_D$$

- ξ_T: net path loss (e.g.$\approx$ 86dB for short-range)
- ξ_C, ξ_D choice of weights for encoder and decoder power.
- Assume $P_e \longrightarrow 0$
- What happens to optimizing P_T, P_C, P_D?
  - P_T stays bounded: *weakly certainty achieving*

What should capacity-achieving mean?

$$\min \xi_T P_T + \xi_C P_C + \xi_D P_D$$

- ξ_T: net path loss (e.g.$\approx$ 86dB for short-range)
- ξ_C, ξ_D choice of weights for encoder and decoder power.
- Assume $P_e \rightarrow 0$
- What happens to optimizing P_T, P_C, P_D?
  - P_T stays bounded: *weakly certainty achieving*
  - P_T, P_C, P_D all stay bounded: *strongly certainty achieving*

What should capacity-achieving mean?

$$\min \xi_T P_T + \xi_C P_C + \xi_D P_D$$

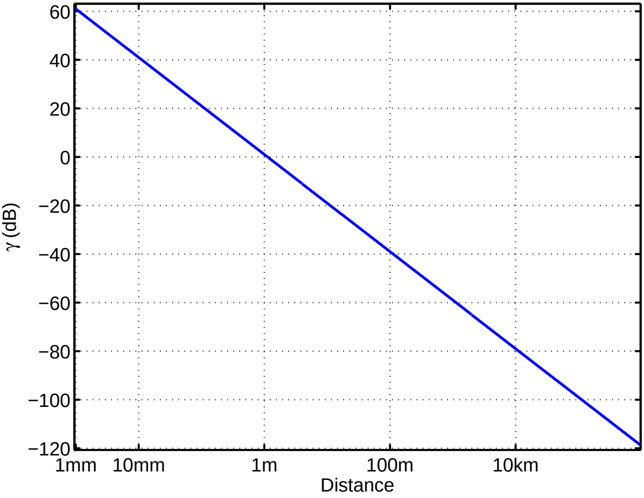
- ξ_T: net path loss (e.g.$\approx$ 86dB for short-range)
- ξ_C, ξ_D choice of weights for encoder and decoder power.
- Assume $P_e \to 0$
- What happens to optimizing P_T, P_C, P_D?
  - P_T stays bounded: *weakly certainty achieving*
  - P_T, P_C, P_D all stay bounded: *strongly certainty achieving*
- What happens as $\xi_C, \xi_D \to 0$?
  - Computation *asymptotically free*, but not actually free.

What should capacity-achieving mean?

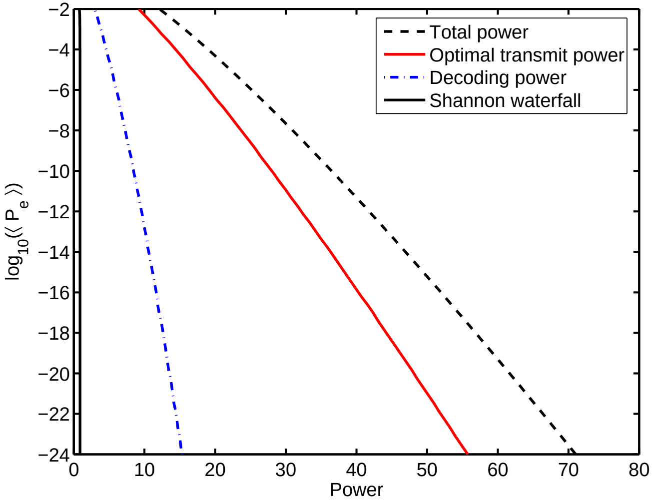
$$\min \xi_T P_T + \xi_C P_C + \xi_D P_D$$

- ξ_T: net path loss (e.g.$\approx$ 86dB for short-range)
- ξ_C, ξ_D choice of weights for encoder and decoder power.
- Assume $P_e \to 0$
- What happens to optimizing P_T, P_C, P_D?
  - P_T stays bounded: *weakly certainty achieving*
  - P_T, P_C, P_D all stay bounded: *strongly certainty achieving*
- What happens as $\xi_C, \xi_D \to 0$?
  - Computation *asymptotically free*, but not actually free.
  - $P_T \to C^{-1}(R)$: *capacity achieving*.

Decoding power vs communication range

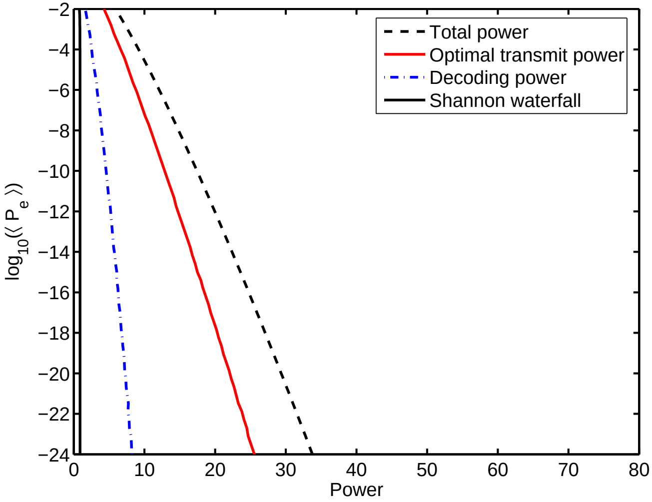


Dense linear codes with brute-force decoding



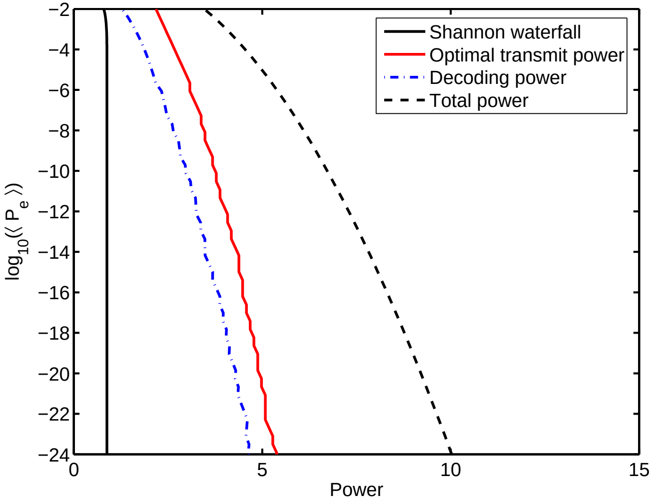
Decoding Power $nR2^{nR}$, Error Prob $2^{-E_{sp}(R,P)n}$

Convolutional codes with Viterbi decoding



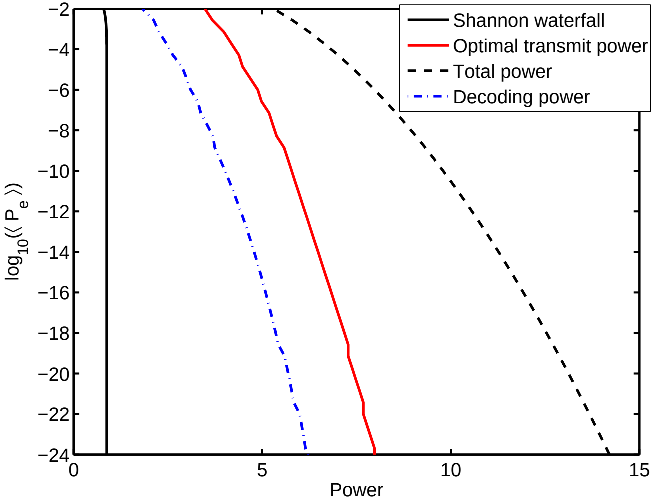
Decoding Power $L_c R 2^{L_c R}$, Error Prob $2^{-E_{conv}(R,P)L_c}$

Convolutional with "magical" sequential decoding



Decoding Power $L_c R$, Error Prob $2^{-E_{conv}(R,P)L_c}$

Dense linear codes with "magical" syndrome decoding



Decoding Power $(1-R)nR$, Error Prob $2^{-E_{sp}(R,P)n}$

Outline

1. Motivation and introduction
2. Classical results revisited
3. A model for decoder power consumption
4. General lower bounds
5. Asymptotic behavior near capacity
6. Optimal choice of rate

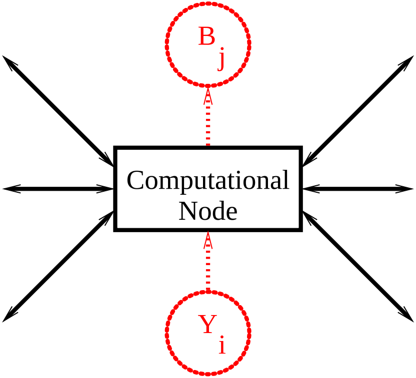
A new hope: iterative decoding

- Make assumptions about the decoder rather than the code.

- Rich enough to capture LDPC, RA, Turbo, etc. codes.

A new hope: iterative decoding

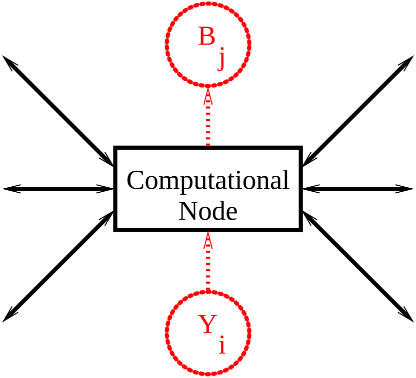
- Make assumptions about the decoder rather than the code.



- Rich enough to capture LDPC, RA, Turbo, etc. codes.

A new hope: iterative decoding

- Make assumptions about the decoder rather than the code.

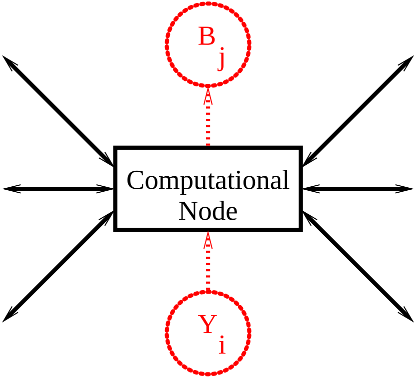


  - Each consumes E_{node} energy per iteration and can send *arbitrary* messages to its $\alpha + 1$ neighbors.

- Rich enough to capture LDPC, RA, Turbo, etc. codes.

A new hope: iterative decoding

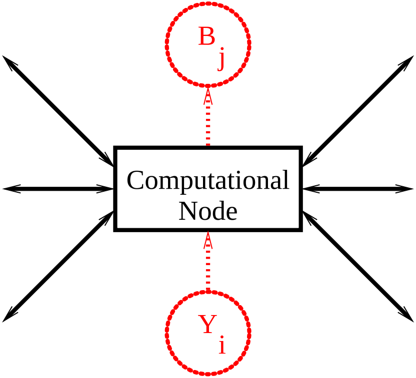
- Make assumptions about the decoder rather than the code.



  - Each consumes E_{node} energy per iteration and can send *arbitrary* messages to its $\alpha + 1$ neighbors.
  - Run for a fixed number of iterations i.
- Rich enough to capture LDPC, RA, Turbo, etc. codes.

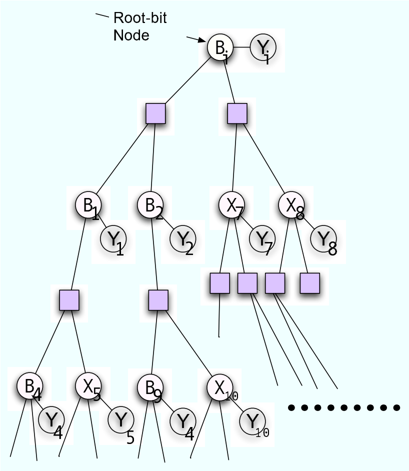
A new hope: iterative decoding

- Make assumptions about the decoder rather than the code.



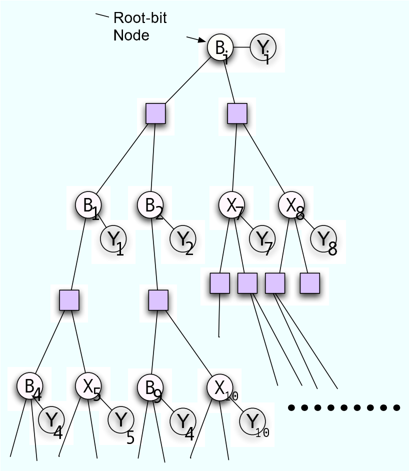
  - Each consumes E_{node} energy per iteration and can send *arbitrary* messages to its $\alpha + 1$ neighbors.
  - Run for a fixed number of iterations i.
- Rich enough to capture LDPC, RA, Turbo, etc. codes.
- Power-consumption $\geq iE_{node}$ per received sample.

How to lower-bound the number of iterations?



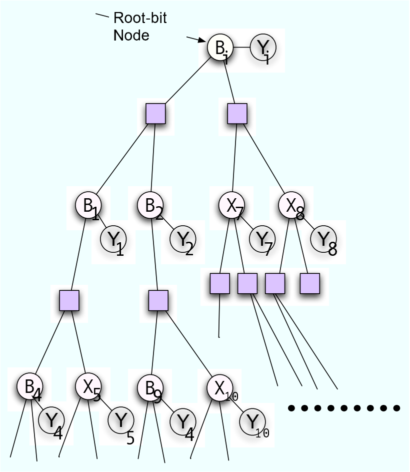
- Key concept: *decoding neighborhood*

How to lower-bound the number of iterations?



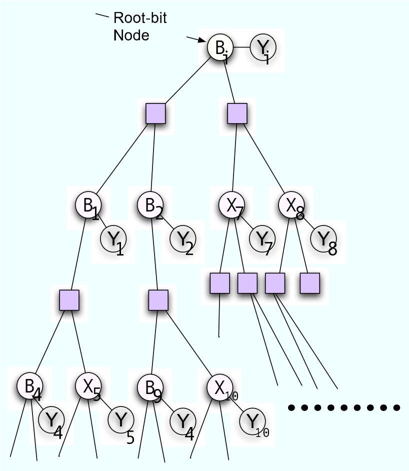
- Key concept: *decoding neighborhood*
- Decoding neighborhood size $n \leq 1 + (\alpha + 1)\alpha^{i-1} \approx \alpha^i$.

How to lower-bound the number of iterations?



- Key concept: *decoding neighborhood*
- Decoding neighborhood size $n \leq 1 + (\alpha + 1)\alpha^{i-1} \approx \alpha^i$.
- Need to lower-bound average probability of bit error in terms of n.

How to lower-bound the number of iterations?



- Key concept: *decoding neighborhood*
- Decoding neighborhood size $n \leq 1 + (\alpha + 1)\alpha^{i-1} \approx \alpha^i$.
- Need to lower-bound average probability of bit error in terms of n.
- Key insight: n is playing a role analogous to delay.

Outline

1. Motivation and introduction
2. Classical results revisited
3. A model for decoder power consumption
4. General lower bounds
5. Asymptotic behavior near capacity
6. Optimal choice of rate

A **local** "sphere-packing" bound for the AWGN

Decoding neighborhood size $n \le 1 + (\alpha + 1)\alpha^{i-1} \approx \alpha^i$.

$\langle P_e \rangle \ge \sup_{\sigma_G^2 > \sigma_P^2 \mu(n):\ C(G) < R}$

$$\frac{h_b^{-1}(\delta(G))}{2} \exp\left(-nD(\sigma_G^2 || \sigma_P^2) - \frac{1}{2}\phi(n, h_b^{-1}(\delta(G)))\left(\frac{\sigma_G^2}{\sigma_P^2} - 1\right)\right)$$

- $C(G) = \frac{1}{2}\log_2(1 + \frac{P_T}{\sigma_G^2})$, $\delta(G)$: $1 - \frac{C(G)}{R}$

A **local** "sphere-packing" bound for the AWGN

Decoding neighborhood size $n \leq 1 + (\alpha + 1)\alpha^{i-1} \approx \alpha^i$.

$\langle P_e \rangle \geq \sup_{\sigma_G^2 > \sigma_P^2 \mu(n):\ C(G) < R}$

$$\frac{h_b^{-1}(\delta(G))}{2} \exp\left(-nD(\sigma_G^2 || \sigma_P^2) - \frac{1}{2}\phi(n, h_b^{-1}(\delta(G)))\left(\frac{\sigma_G^2}{\sigma_P^2} - 1\right)\right)$$

- $C(G) = \frac{1}{2}\log_2(1 + \frac{P_T}{\sigma_G^2})$, $\delta(G)$: $1 - \frac{C(G)}{R}$
- $\mu(n) = \frac{1}{2}(1 + \frac{1}{T(n)+1} + \frac{4T(n)+2}{nT(n)(1+T(n))})$
- $T(n) = -W_L(-\exp(-1)(1/4)^{1/n})$
- $W_L(x)$ solves $x = W_L(x)\exp(W_L(x))$
- $\phi(n, y) = -n(W_L\left(-\exp(-1)(\frac{y}{2})^{\frac{2}{n}}\right) + 1)$

A **local** "sphere-packing" bound for the AWGN

Decoding neighborhood size $n \leq 1 + (\alpha + 1)\alpha^{i-1} \approx \alpha^i$.

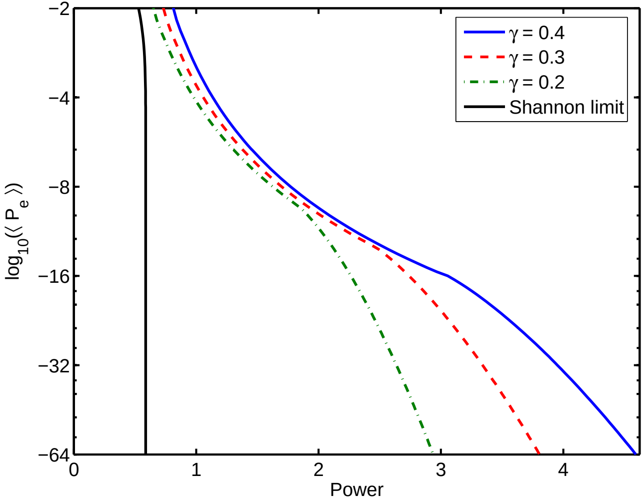
$\langle P_e \rangle \geq \sup_{\sigma_G^2 > \sigma_P^2 \mu(n):\ C(G) < R}$

$$\frac{h_b^{-1}(\delta(G))}{2} \exp\left(-nD(\sigma_G^2 || \sigma_P^2) - \frac{1}{2}\phi(n, h_b^{-1}(\delta(G)))\left(\frac{\sigma_G^2}{\sigma_P^2} - 1\right)\right)$$

- $C(G) = \frac{1}{2}\log_2(1 + \frac{P_T}{\sigma_G^2})$, $\delta(G)$: $1 - \frac{C(G)}{R}$
- $\mu(n) = \frac{1}{2}(1 + \frac{1}{T(n)+1} + \frac{4T(n)+2}{nT(n)(1+T(n))})$
- $T(n) = -W_L(-\exp(-1)(1/4)^{1/n})$
- $W_L(x)$ solves $x = W_L(x)\exp(W_L(x))$
- $\phi(n, y) = -n(W_L\left(-\exp(-1)(\frac{y}{2})^{\frac{2}{n}}\right) + 1)$

Double-exponential potential return on investments in decoding power!

Waterslide curves for general AWGN case



A **local** "sphere-packing" bound for the BSC

$$\langle P_e \rangle \geq \sup_{C^{-1}(R) < g \leq \frac{1}{2}} \frac{h_b^{-1}(\delta(g))}{2} 2^{-nD(g||p)} \left(\frac{p(1-g)}{g(1-p)} \right)^{\epsilon\sqrt{n}}$$

- $h_b(p)$: Binary entropy function
- $\delta(g)$: $1 - \frac{C(g)}{R}$
- $D(g||p)$: KL Divergence
- ϵ: $\sqrt{\frac{1}{K(g)} \log(\frac{2}{h_b^{-1}(\delta(G))})}$
- $K(g)$: $\inf_{0<\eta<1-g} \frac{D(g+\eta||g)}{\eta^2}$.

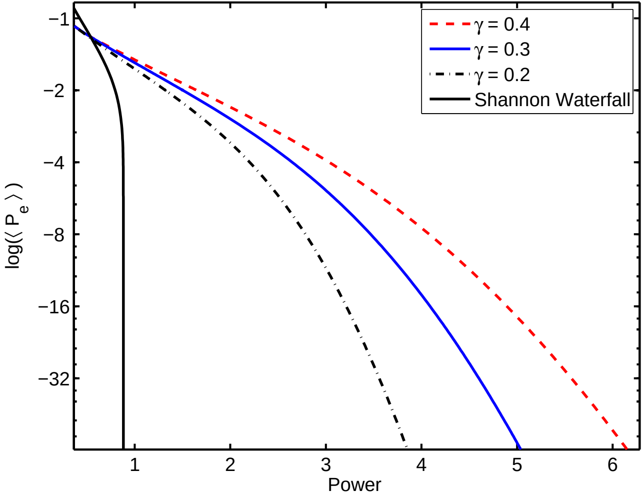
A **local** "sphere-packing" bound for the BSC

$$\langle P_e \rangle \geq \sup_{C^{-1}(R) < g \leq \frac{1}{2}} \frac{h_b^{-1}(\delta(g))}{2} 2^{-nD(g||p)} \left(\frac{p(1-g)}{g(1-p)} \right)^{\epsilon\sqrt{n}}$$

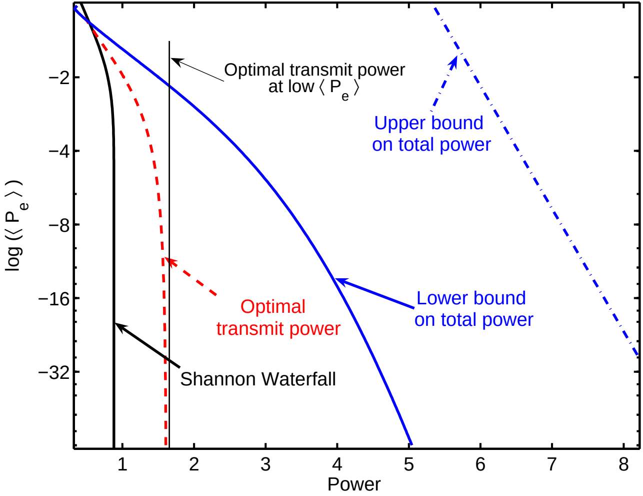
- $h_b(p)$: Binary entropy function
- $\delta(g)$: $1 - \frac{C(g)}{R}$
- $D(g||p)$: KL Divergence
- ϵ: $\sqrt{\frac{1}{K(g)} \log(\frac{2}{h_b^{-1}(\delta(G))})}$
- $K(g)$: $\inf_{0<\eta<1-g} \frac{D(g+\eta||g)}{\eta^2}$.

Double-exponential **actual returns** observed by *Lentmaier, et al.* 2005 for regular LDPC codes with iterative decoding!

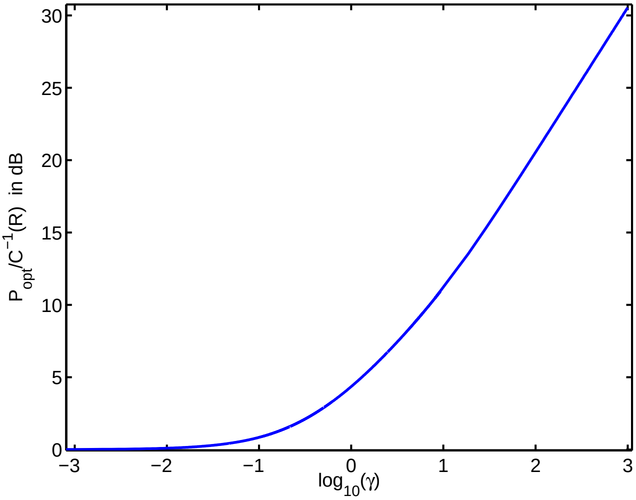
Detailed look at BPSK case



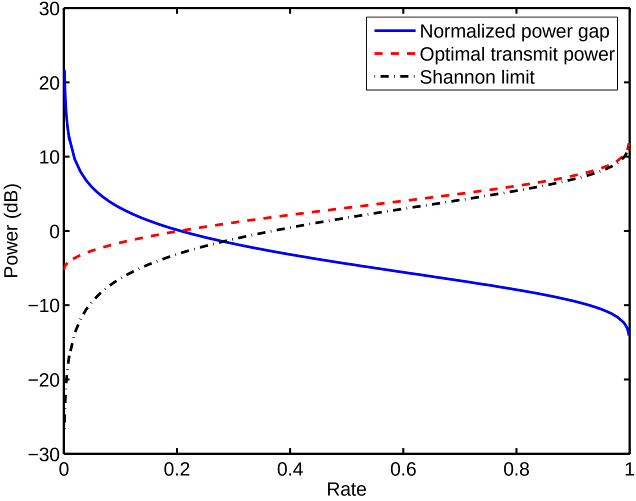
Detailed look at BPSK case



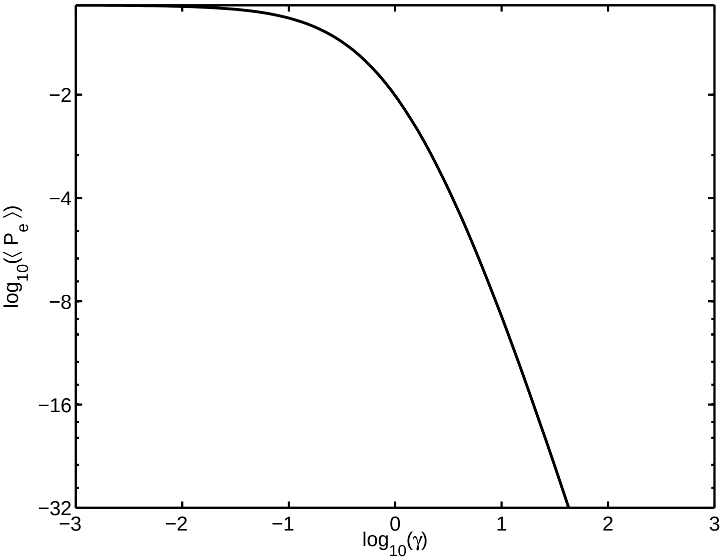
Deviations from Shannon



Deviations from Shannon



Deviations from Shannon



Sketch of proof

- Pretend the code runs over channel G instead of P.

Sketch of proof

- Pretend the code runs over channel G instead of P.
- $\delta(G) > 0$ implies average probability of error is $h_b^{-1}(\delta(G))$.

Sketch of proof

- Pretend the code runs over channel G instead of P.
- $\delta(G) > 0$ implies average probability of error is $h_b^{-1}(\delta(G))$.
- Channel needs only to misbehave for the decoding neighborhoods to cause a bit-error.

Sketch of proof

- Pretend the code runs over channel G instead of P.
- $\delta(G) > 0$ implies average probability of error is $h_b^{-1}(\delta(G))$.
- Channel needs only to misbehave for the decoding neighborhoods to cause a bit-error.
- ϵ/ϕ assure that the misbehavior is "typical" (*even if n is small*)

Sketch of proof

- Pretend the code runs over channel G instead of P.
- $\delta(G) > 0$ implies average probability of error is $h_b^{-1}(\delta(G))$.
- Channel needs only to misbehave for the decoding neighborhoods to cause a bit-error.
- ϵ/ϕ assure that the misbehavior is “typical” (*even if n is small*)
- Probability of an error event under P is lower-bounded by a convex-$\cup$ function f of the probability under G.

Sketch of proof

- Pretend the code runs over channel G instead of P.
- $\delta(G) > 0$ implies average probability of error is $h_b^{-1}(\delta(G))$.
- Channel needs only to misbehave for the decoding neighborhoods to cause a bit-error.
- ϵ/ϕ assure that the misbehavior is “typical” (*even if n is small*)
- Probability of an error event under P is lower-bounded by a convex-$\cup$ function f of the probability under G.
- Average probability under P is minimized by f of the average probability under G.

The low P_e limit

- Dominant term: $D(G||P)$ term in exponent.
- $P_e \approx \exp(-D(C^{-1}(R)||P)n) = \exp(-D(C^{-1}(R)||P)\alpha^i)$

The low P_e limit

- Dominant term: $D(G||P)$ term in exponent.
- $P_e \approx \exp(-D(C^{-1}(R)||P)n) = \exp(-D(C^{-1}(R)||P)\alpha^i)$
- Take double logs of both sides.

$$\ln\ln\frac{1}{P_e} \approx \ln D(C^{-1}(R)||P) + i\ln\alpha$$

The low P_e limit

- Dominant term: $D(G||P)$ term in exponent.
- $P_e \approx \exp(-D(C^{-1}(R)||P)n) = \exp(-D(C^{-1}(R)||P)\alpha^i)$
- Take double logs of both sides.

$$\ln \ln \frac{1}{P_e} \approx \ln D(C^{-1}(R)||P) + i \ln \alpha$$

- Minimize sum $\zeta + \gamma i$ where $\gamma = \frac{\xi_D E_{node}}{\sigma_P^2 \xi_T \ln \alpha}$ and $\zeta = \frac{P_T}{\sigma_P^2}$.

The low P_e limit

- Dominant term: $D(G||P)$ term in exponent.
- $P_e \approx \exp(-D(C^{-1}(R)||P)n) = \exp(-D(C^{-1}(R)||P)\alpha^i)$
- Take double logs of both sides.

$$\ln\ln\frac{1}{P_e} \approx \ln D(C^{-1}(R)||P) + i\ln\alpha$$

- Minimize sum $\zeta + \gamma i$ where $\gamma = \frac{\xi_D E_{node}}{\sigma_P^2 \xi_T \ln\alpha}$ and $\zeta = \frac{P_T}{\sigma_P^2}$.
- Solved by:

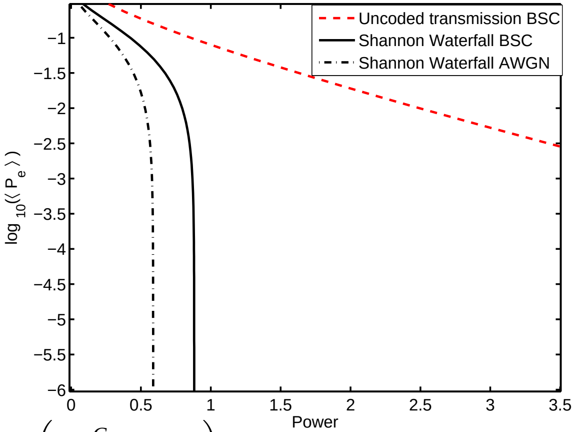
$$f(R,\zeta) / \frac{\partial f(R,\zeta)}{\partial\zeta} = \gamma$$

where $f(R, \frac{P_T}{\sigma_P^2}) = D(C^{-1}(R)||P)$.

Outline

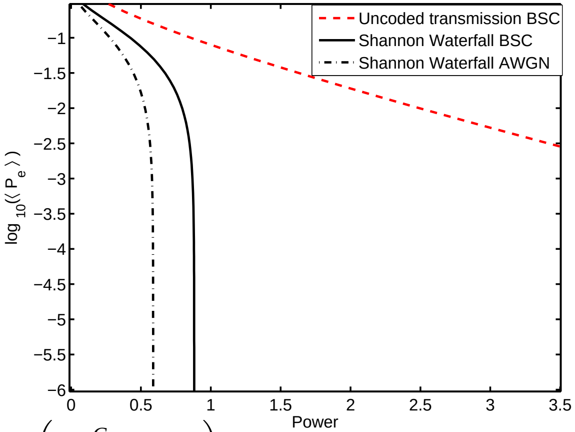
1. Motivation and introduction
2. Classical results revisited
3. A model for decoder power consumption
4. General lower bounds
5. Asymptotic behavior near capacity
6. Optimal choice of rate

The waterfall curve revisited



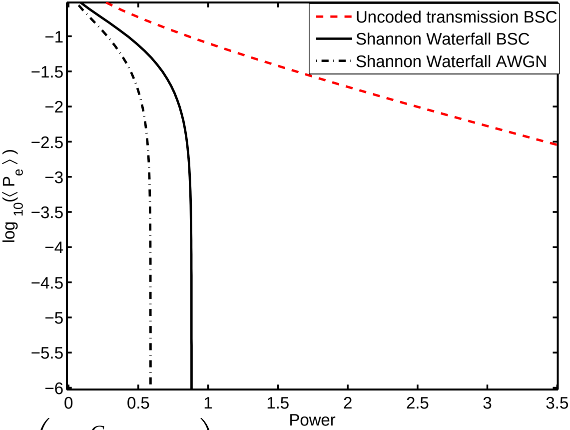
- Two gaps: $\left(\frac{C}{1-h_b(P_e)} - C\right)$ and $gap = C - R$.

The waterfall curve revisited



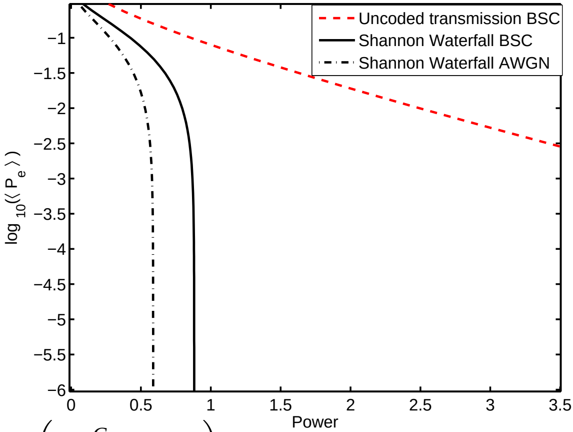
- Two gaps: $\left(\frac{C}{1-h_b(P_e)} - C\right)$ and $gap = C - R$.
- Pick a path to certainty: $P_e \to 0$ so $i \approx \frac{\ln \ln \frac{1}{P_e}}{\ln \alpha} + \frac{\ln \frac{1}{D(C^{-1}(R)||P)}}{\ln \alpha}$

The waterfall curve revisited



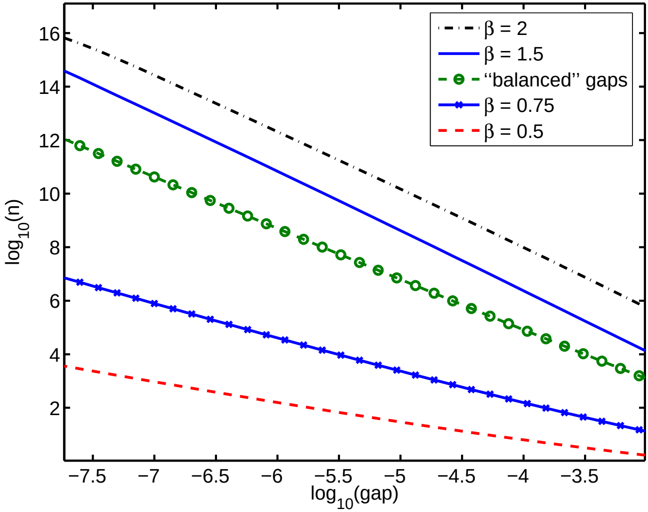
- Two gaps: $\left(\frac{C}{1-h_b(P_e)} - C\right)$ and $gap = C - R$.
- Pick a path to certainty: $P_e \rightarrow 0$ so $i \approx \frac{\ln \ln \frac{1}{P_e}}{\ln \alpha} + \frac{\ln \frac{1}{D(C^{-1}(R)||P)}}{\ln \alpha}$
- Let $R \rightarrow C$ and so $i = \log_\alpha \ln \frac{1}{P_e} + 2 \log_\alpha \frac{1}{gap} + o(\cdots)$.

The waterfall curve revisited



- Two gaps: $\left(\frac{C}{1-h_b(P_e)} - C\right)$ and $gap = C - R$.
- Pick a joint path to certainty: $P_e = gap^{\beta}$.
- Let $R \to C \ldots$

Zoom into the neighborhood of capacity



Asymptotic scaling of iterations and gap

Assume $P_e = gap^{\beta}$ and BSC channel:

- If $\beta \geq 1$, $i \geq 2 \log \frac{1}{gap} + \log \log \frac{1}{gap} + c_\beta$.
- If $\beta \leq 1$, $i \geq 2\beta \log \frac{1}{gap} + \log \log \frac{1}{gap} + c_\beta$.

Asymptotic scaling of iterations and gap

Assume $P_e = gap^{\beta}$ and BSC channel:

- If $\beta \geq 1$, $i \geq 2 \log \frac{1}{gap} + \log \log \frac{1}{gap} + c_{\beta}$.
- If $\beta \leq 1$, $i \geq 2\beta \log \frac{1}{gap} + \log \log \frac{1}{gap} + c_{\beta}$.
- Proved by taking $g^* = p + gap^r$ and using careful Taylor expansions around $g = p$.

Asymptotic scaling of iterations and gap

Assume $P_e = gap^{\beta}$ and BSC channel:

- If $\beta \geq 1$, $i \geq 2\log\frac{1}{gap} + \log\log\frac{1}{gap} + c_{\beta}$.
- If $\beta \leq 1$, $i \geq 2\beta\log\frac{1}{gap} + \log\log\frac{1}{gap} + c_{\beta}$.
- Proved by taking $g^* = p + gap^r$ and using careful Taylor expansions around $g = p$.
- Much better than semi-trivial bound of $\Omega(\log\log\frac{1}{gap})$.

Asymptotic scaling of iterations and gap

Assume $P_e = gap^{\beta}$ and BSC channel:

- If $\beta \geq 1$, $i \geq 2 \log \frac{1}{gap} + \log \log \frac{1}{gap} + c_{\beta}$.
- If $\beta \leq 1$, $i \geq 2\beta \log \frac{1}{gap} + \log \log \frac{1}{gap} + c_{\beta}$.
- Proved by taking $g^* = p + gap^r$ and using careful Taylor expansions around $g = p$.
- Much better than semi-trivial bound of $\Omega(\log \log \frac{1}{gap})$.
- Much more optimistic than Khandekar-McEliece conjectured $\Omega(\frac{1}{gap})$.

Outline

1. Motivation and introduction
2. Classical results revisited
3. A model for decoder power consumption
4. General lower bounds
5. Asymptotic behavior near capacity
6. Optimal choice of rate

Assume a fixed message size

- Suppose message very large, but not urgent.

Assume a fixed message size

- Suppose message very large, but not urgent.
- Why not increase rate?

Assume a fixed message size

- Suppose message very large, but not urgent.
- Why not increase rate?
  - Advantage: fewer samples to process

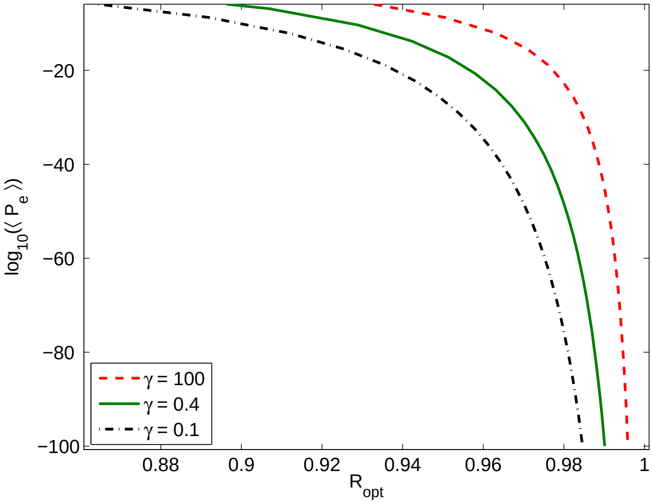
Assume a fixed message size

- Suppose message very large, but not urgent.
- Why not increase rate?
  - Advantage: fewer samples to process
  - Disadvantage: need higher capacity and thus more power
- Energy per message bit: $\frac{1}{R}\frac{P_T}{\sigma_P^2} + \max(1, \frac{1}{R})\gamma i$.

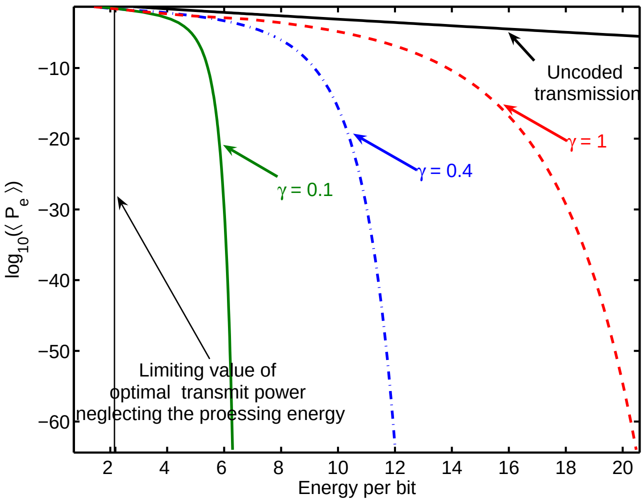
Assume a fixed message size

- Suppose message very large, but not urgent.
- Why not increase rate?
  - Advantage: fewer samples to process
  - Disadvantage: need higher capacity and thus more power
- Energy per message bit: $\frac{1}{R}\frac{P_T}{\sigma_P^2} + \max(1, \frac{1}{R})\gamma i$.
- Optimize over the choice of R as P_e varies.

Results for BPSK signaling



Results for BPSK signaling



Concluding remarks

More details:
arXiv:0801.0352
www.eecs.berkeley.edu/~sahai/

- Bounds can probably be tightened.

Concluding remarks

More details:
arXiv:0801.0352
www.eecs.berkeley.edu/$\sim$sahai/

- Bounds can probably be tightened.
- Encoding power still needs to be better understood.

Concluding remarks

More details:
arXiv:0801.0352
www.eecs.berkeley.edu/∼sahai/

- Bounds can probably be tightened.
- Encoding power still needs to be better understood.
- Is there a way around our model of decoding?
- How to reconcile with “linear-time” codes like expanders?

Concluding remarks

More details:
arXiv:0801.0352
www.eecs.berkeley.edu/∼sahai/

- Bounds can probably be tightened.
- Encoding power still needs to be better understood.
- Is there a way around our model of decoding?
- How to reconcile with “linear-time” codes like expanders?
- Expand scope to cover multiterminal problems and other components.