

Multi-Rate Control over AWGN Channels via Analog Joint Source–Channel Coding

Anatoly Khina, Caltech

Joint work with:

Gustav M. Pettersson, KTH

Victoria Kostina and Babak Hassibi, Caltech

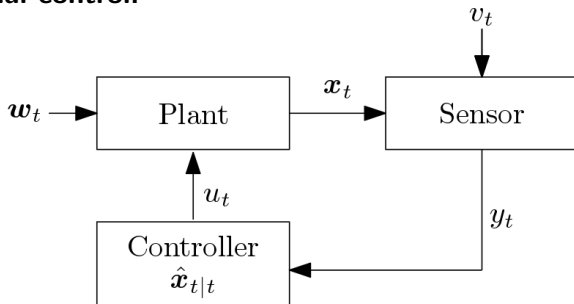
CDC 2016

Las Vegas, NV, USA

December 14, 2016

Networked Control vs. Traditional Control

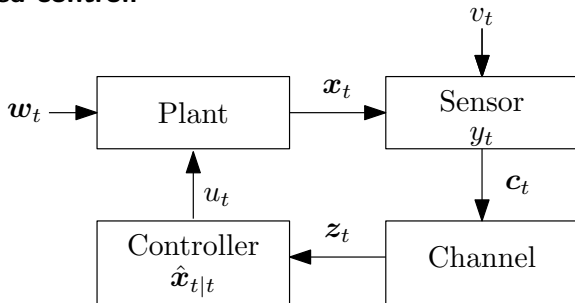
Traditional control:



- Observer and controller are co-located.
- Classical systems are hardwired and well crafted

Networked Control vs. Traditional Control

Networked control:



- Observer and controller are not co-located:
connected through noisy link
- Suitable for new remote applications
(e.g., remote surgery, self-driving cars)

Linear Quadratic Gaussian Control over Gaussian Channels

Linear quadratic Gaussian (LQG) system

$$\begin{aligned} \mathbf{x}_{t+1} &= \mathbf{A}\mathbf{x}_t + \mathbf{B}\mathbf{u}_t + \mathbf{w}_t, & \mathbf{w}_t &\sim \text{i.i.d. } \mathcal{N}(0, \mathbf{W}) \\ \mathbf{y}_t &= \mathbf{C}\mathbf{x}_t + \mathbf{v}_t, & \mathbf{v}_t &\sim \text{i.i.d. } \mathcal{N}(0, \mathbf{V}) \end{aligned}$$

Additive white Gaussian noise (AWGN) channel

$$\mathbf{b}_t = \mathbf{H}\mathbf{a}_t + \mathbf{n}_t, \quad \mathbf{n}_t \sim \text{i.i.d. } \mathcal{N}(0, \mathbf{N})$$

- Power constraint: $\mathbb{E}[\mathbf{a}_t^T \mathbf{a}_t] = \sum_i \mathbb{E}[a_{t,i}^2] \leq P \cdot \text{length}(\mathbf{a}_t)$

LQG cost

$$J = \sum_{t=1}^T \left[\mathbf{x}_t^T \mathbf{Q} \mathbf{x}_t + \mathbf{u}_t^T \mathbf{R} \mathbf{u}_t \right] + \mathbf{x}_{T+1}^T \mathbf{F} \mathbf{x}_{T+1}$$

Linear Quadratic Gaussian Control over Gaussian Channels

Scalar Linear quadratic Gaussian (LQG) system

$$\begin{aligned}x_{t+1} &= x_t + u_t + w_t, & w_t &\sim \text{i.i.d. } \mathcal{N}(0, W) \\ y_t &= x_t + v_t, & v_t &\sim \text{i.i.d. } \mathcal{N}(0, V)\end{aligned}$$

Scalar Additive white Gaussian noise (AWGN) channel

$$b_t = a_t + n_t, \quad n_t \sim \text{i.i.d. } \mathcal{N}(0, N)$$

- Power constraint: $\mathbb{E}[a_t^2] \leq P$;

LQG cost

$$J = \sum_{t=1}^T [Qx_t^2 + Ru_t^2] + Fx_{T+1}^2$$

Linear Quadratic Gaussian Control over Gaussian Channels

Scalar Linear quadratic Gaussian (LQG) system

$$\begin{aligned}x_{t+1} &= x_t + u_t + w_t, & w_t &\sim \text{i.i.d. } \mathcal{N}(0, W) \\ y_t &= x_t + v_t, & v_t &\sim \text{i.i.d. } \mathcal{N}(0, V)\end{aligned}$$

Scalar Additive white Gaussian noise (AWGN) channel

$$b_t = a_t + n_t, \quad n_t \sim \text{i.i.d. } \mathcal{N}(0, N)$$

- Power constraint: $\mathbb{E}[a_t^2] \leq P$; w.l.o.g. $P = 1$, $N = 1/\text{SNR}$

LQG cost

$$J = \sum_{t=1}^T [Qx_t^2 + Ru_t^2] + Fx_{T+1}^2$$

Linear Quadratic Gaussian Control over Gaussian Channels

Scalar LQG system

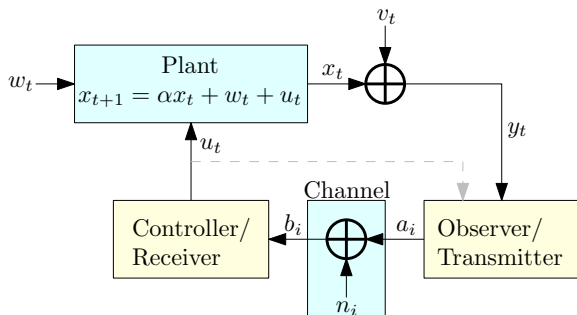
$$x_{t+1} = x_t + u_t + w_t$$

$$y_t = x_t + v_t$$

Scalar AWGN channel

$$b_t = a_t(y^t, u^{t-1}) + n_t$$

- Power constraint: $\mathbb{E}[a_t^2] \leq P$



Linear Quadratic Gaussian Control over Gaussian Channels

Control sampling rate \neq Communication signaling rate!

Scalar LQG system

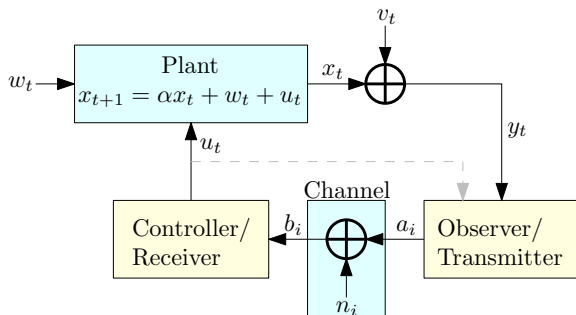
$$x_{t+1} = x_t + u_t + w_t$$

$$y_t = x_t + v_t$$

Scalar AWGN channel

$$b_i = a_i + n_i$$

- Power constraint: $\mathbb{E}[a_i^2] \leq P$



Control Sampling Rate vs. Communication Signaling Rate

- How fast the plant dynamic is \Rightarrow Control sampling rate
- Bandwidth available \Rightarrow Communication signaling rate
- Communication rate can be much higher in practice

How to benefit from excess signaling rate (bandwidth)?

Networked Control Approaches: Information-Theoretic Separation

Information-theoretic separation

- Requires large blocks (**delay!**) of source samples and channel uses
- **Suboptimal for control!**
- Requires codes with strong “anytime reliability” properties [Schulman IT'96][Sahai-Mitter IT'06][Sukhavasi-Hassibi AC'16]
- Problematic in practice: Convolutional code with infinite memory [Kh.-Halbawi-Hassibi ISIT'16]

Packeting

- Assumes communication rate \gg control rate, very good SNR
- Problem reduces to control-oriented quantization
- Bad channel events are translated to packet drops / delays

Networked Control Approaches: Joint Source–Channel Coding (JSCC)

- What to do when control and communication rates are close?
- Can we do better than IT-separation?

Less familiar IT avenue

- Low-delay joint source–channel coding (JSCC)
- Control sample corresponds to source sample

1 : 1 JSCC: Rate-Matched Case

- One AWGN channel use per one control sample

1 : 1 Optimal JSCC [Goblick IT'65]

1 : 1 optimal JSCC distortion = n : n optimal JSCC distortion

- No loss of performance
- **Analog scheme is optimal:** $a_t = \sqrt{\frac{P}{P_x}} x_t$

1 : 1 JSCC: Rate-Matched Case

Scheme

- For simplicity, assume fully-observable case
(can be extended to partially-observable case)

Observer/Transmitter:

- Generates the “source” signal: $s_t = x_t - \hat{x}_{t|t-1} = \tilde{x}_{t|t-1}$
- Adjusts power and transmits: $a_t = s_t / \sqrt{P_{t|t-1}}$

Controller/Receiver:

- Receives $b_t = a_t + n_t = \tilde{x}_{t|t-1} / \sqrt{P_{t|t-1}} + n_t$
- Applies Kalman filtering:
$$\begin{cases} \hat{x}_{t|t} = \hat{x}_{t|t-1} + \sqrt{P_{t|t-1}} \frac{\text{SNR}}{1+\text{SNR}} b_t \\ \hat{x}_{t|t-1} = \alpha \hat{x}_{t-1|t-1} + u_{t-1} \end{cases}$$
- Generates LQG control signal: $u_t = -L_t \hat{x}_{t|t}$

1 : 1 JSCC: Rate-Matched Case

- We reduced the problem to that of classical LQG control

LQR coefficients

$$L_t = \frac{\alpha S_{t+1}}{S_{t+1} + R},$$

$$S_t = \frac{\alpha^2 R S_{t+1}}{S_{t+1} + R} + Q,$$

$$S_T = F.$$

Partially-observable case

Scheme can be extended to partially-observable case

- Generates state estimators at the transmitter $\hat{x}_{t|t}^t$
(in addition to $\hat{x}_{t|t}^r$ at the receiver)

1 : 1 JSCC: Rate-Matched Case

LQG cost

- This scheme achieves optimal LQG cost
 - Formally proved by applying
 - Shannon's lower bound
 - Entropy-power inequality
 - Tightness of both in Gaussian case
 - Optimality of "1 : 1 JSCC" scheme in the Gaussian case
 - Recovers results of [Freudenberg-Middleton-Solo AC'10] as a special case
- in the dynamic-programming solution
(extension of [Kostina-Hassibi Allerton'16])

Conclusion: No coding is needed!

1 : 1 JSCC: Rate-Matched Case

Infinite-horizon steady-state average-stage LQG cost

$$\bar{J}^r = \bar{J}^t + \frac{Q + (\alpha^2 - 1) S}{1 + \text{SNR} - \alpha^2} W$$

$$\bar{J}^t = SW$$

- S is the positive solution of the DARE

$$S^2 - [Q + (\alpha^2 - 1) R] S - QR = 0$$

- System is stabilizable if and only if $\text{SNR} > \alpha^2 - 1$
- This is in stark contrast to classical LQG

Conclusion: No coding is needed!

1 : 1 JSCC: Rate-Matched Case

Infinite-horizon LQG cost: Partially-observable case

$$\bar{J}^t = \bar{J}^t + \frac{Q + (\alpha^2 - 1) S}{1 + \text{SNR} - \alpha^2} (P_t^t - \bar{P}_t^t)$$

$$\bar{J}^t = S(P_t^t - \bar{P}_t^t) + Q\bar{P}_t^t$$

- S is the positive solution of the DARE

$$S^2 - [Q + (\alpha^2 - 1) R] S - QR = 0$$

- System is stabilizable if and only if $\text{SNR} > \alpha^2 - 1$

- $P^t \triangleq \lim_{t \rightarrow \infty} P_{t+1|t}^t$, $\bar{P}^t \triangleq \lim_{t \rightarrow \infty} P_{t|t}^t$

- P^t is the positive solution of the DARE

$$(P^t)^2 - [(\alpha^2 - 1) V + W] P^t - VW = 0, \quad \bar{P}^t = \frac{P^t V}{P^t + V}$$

Conclusion: No coding is needed!

1 : 1 JSCC: Rate-Matched Case

Infinite-horizon LQG cost: Partially-observable case

$$\bar{J}^t = \bar{J}^t + \frac{Q + (\alpha^2 - 1) S}{1 + \text{SNR} - \alpha^2} (P_t^t - \bar{P}_t^t)$$

$$\bar{J}^t = S(P_t^t - \bar{P}_t^t) + Q\bar{P}_t^t$$

- S is the positive solution of the DARE

$$S^2 - [Q + (\alpha^2 - 1) R] S - QR = 0$$

- System is stabilizable if and only if $\text{SNR} > \alpha^2 - 1$

- $P^t \triangleq \lim_{t \rightarrow \infty} P_{t+1|t}^t$, $\bar{P}^t \triangleq \lim_{t \rightarrow \infty} P_{t|t}^t$

- P^t is the positive solution of the DARE

$$(P^t)^2 - [(\alpha^2 - 1) V + W] P^t - VW = 0, \quad \bar{P}^t = \frac{P^t V}{P^t + V}$$

What about 1 : 2 case?

1 : 2 JSCC: Rate-Mismatched Case

- Two AWGN channel uses per one control sample

Naïve scheme: Repetition

Observer/Transmitter: $a_{t;1} = a_{t;2} = \tilde{x}_t / \sqrt{P_{t|t-1}}$

Controller/Receiver: $b_t^{\text{eff}} = \frac{b_{t;1} + b_{t;2}}{2}$

- Reduces to 1 : 1 JSCC with $\text{SNR}^{\text{eff}} = 2\text{SNR}$
- 3dB improvement comes from doubling total transmit power
- Same improvement is attained by
 - Using $2P$ during first channel use
 - Remaining silent during second channel use
- No real improvement due to extra degree of freedom...

Can we do better?

1 : 2 JSCC: Rate-Mismatched Case

Infinite blocklength: “ $n : 2n$ JSCC” for $n \rightarrow \infty$ [Shannon '48]

$$1 + \text{SNR}^{\text{eff}} = (1 + \text{SNR})^2$$

- Much better than $\text{SNR}_{\text{naïve}}^{\text{eff}} = 2\text{SNR}$ at high SNR

1 : 2 JSCC: Rate-Mismatched Case

Infinite blocklength: “ $n : 2n$ JSCC” for $n \rightarrow \infty$ [Shannon '48]

$$1 + \text{SNR}^{\text{eff}} = (1 + \text{SNR})^2$$

- Much better than $\text{SNR}_{\text{naïve}}^{\text{eff}} = 2\text{SNR}$ at high SNR

1 : 2 JSCC: Rate-Mismatched Case

Infinite blocklength: “ $n : 2n$ JSCC” for $n \rightarrow \infty$ [Shannon '48]

$$1 + \text{SNR}^{\text{eff}} = (1 + \text{SNR})^2$$

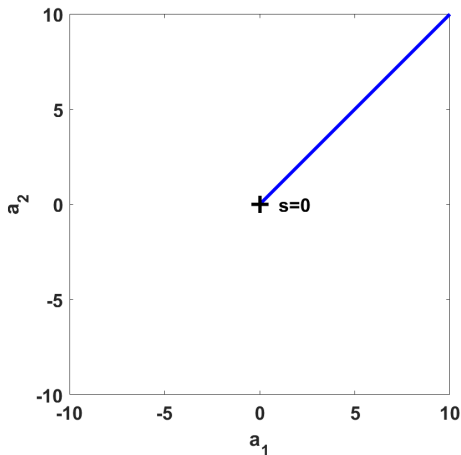
- Much better than $\text{SNR}_{\text{naive}}^{\text{eff}} = 2\text{SNR}$ at high SNR

What about 1 : 2 JSCC?

Non-linear mappings can do better! [Kotel'nikov '47][Shannon '49]

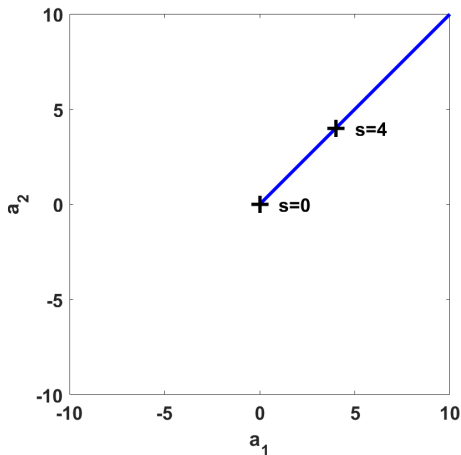
1 : 2 JSCC: Rate-Mismatched Case

$$\begin{cases} a_1(s) = s \\ a_2(s) = s \end{cases}$$



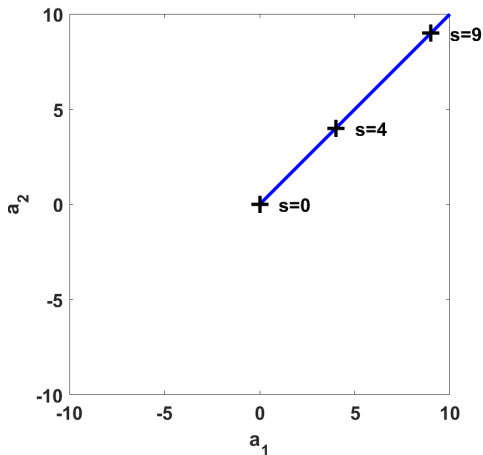
1 : 2 JSCC: Rate-Mismatched Case

$$\begin{cases} a_1(s) = s \\ a_2(s) = s \end{cases}$$



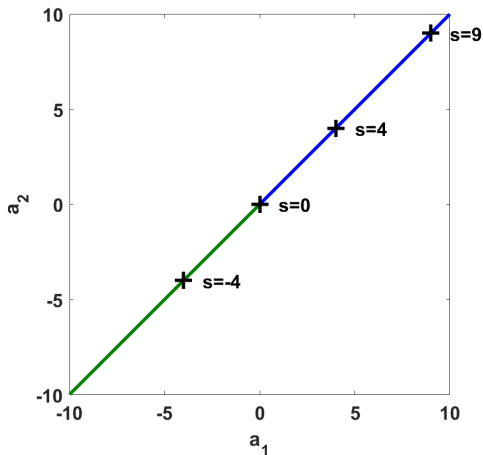
1 : 2 JSCC: Rate-Mismatched Case

$$\begin{cases} a_1(s) = s \\ a_2(s) = s \end{cases}$$



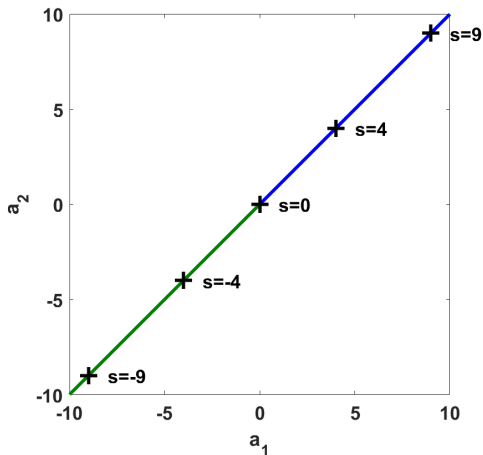
1 : 2 JSCC: Rate-Mismatched Case

$$\begin{cases} a_1(s) = s \\ a_2(s) = s \end{cases}$$



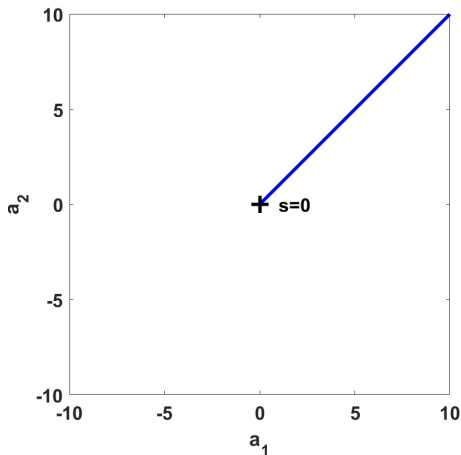
1 : 2 JSCC: Rate-Mismatched Case

$$\begin{cases} a_1(s) = s \\ a_2(s) = s \end{cases}$$

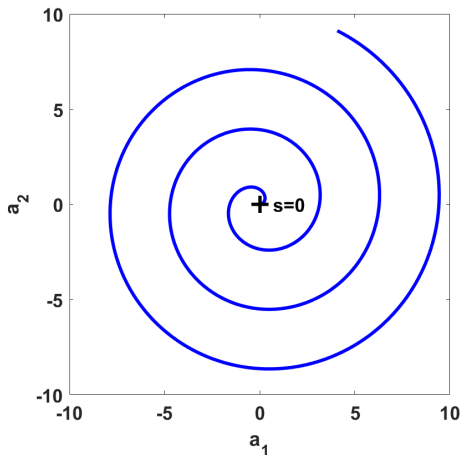


1 : 2 JSCC: Rate-Mismatched Case

$$\begin{cases} a_1(s) = s \\ a_2(s) = s \end{cases}$$

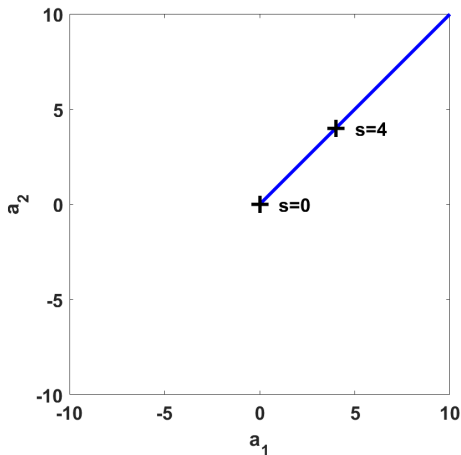


$$\begin{cases} a_1(s) = s \cos(2s) \\ a_2(s) = s \sin(2s) \end{cases}$$

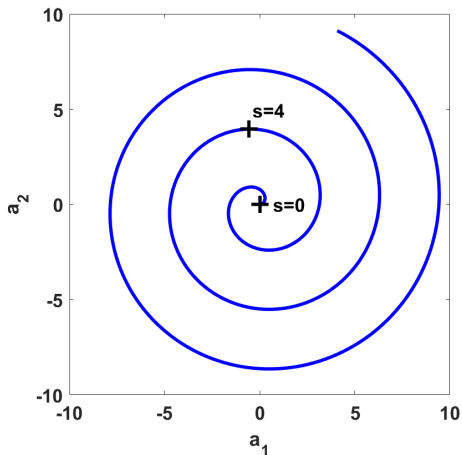


1 : 2 JSCC: Rate-Mismatched Case

$$\begin{cases} a_1(s) = s \\ a_2(s) = s \end{cases}$$

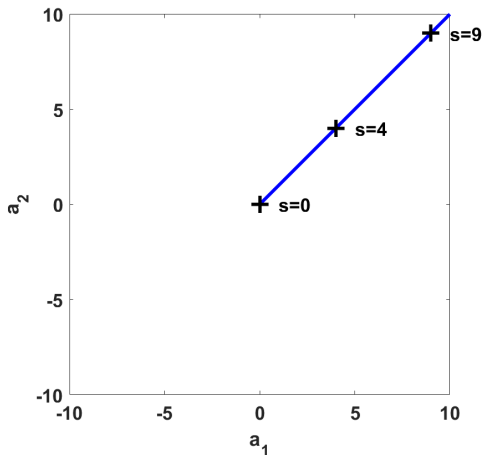


$$\begin{cases} a_1(s) = s \cos(2s) \\ a_2(s) = s \sin(2s) \end{cases}$$

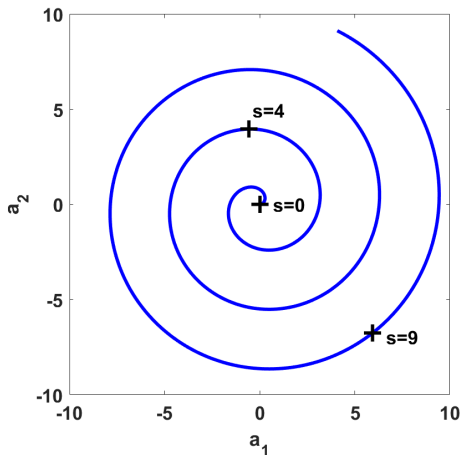


1 : 2 JSCC: Rate-Mismatched Case

$$\begin{cases} a_1(s) = s \\ a_2(s) = s \end{cases}$$

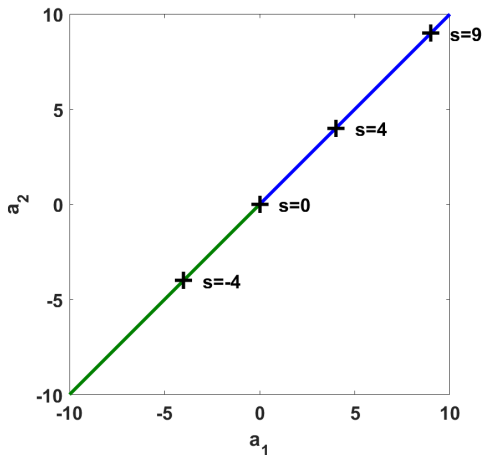


$$\begin{cases} a_1(s) = s \cos(2s) \\ a_2(s) = s \sin(2s) \end{cases}$$

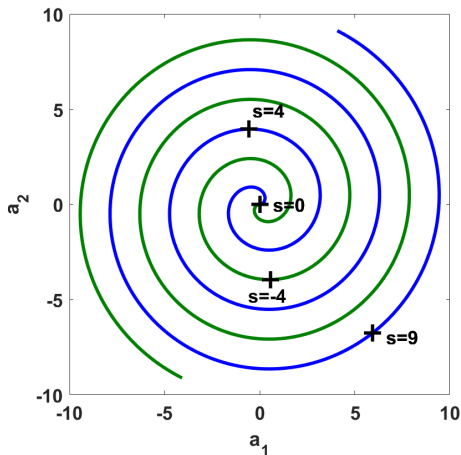


1 : 2 JSCC: Rate-Mismatched Case

$$\begin{cases} a_1(s) = s \\ a_2(s) = s \end{cases}$$

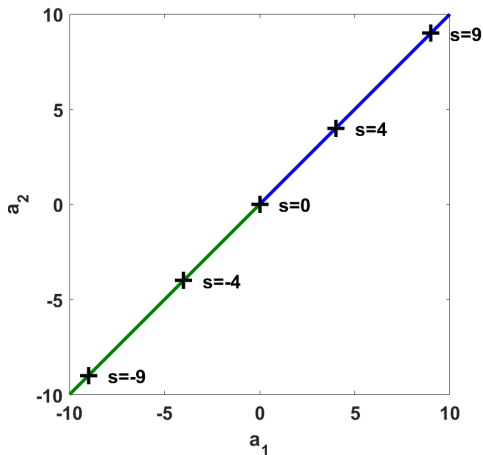


$$\begin{cases} a_1(s) = s \cos(2s) \\ a_2(s) = s \sin(2s) \text{ sign}(s) \end{cases}$$

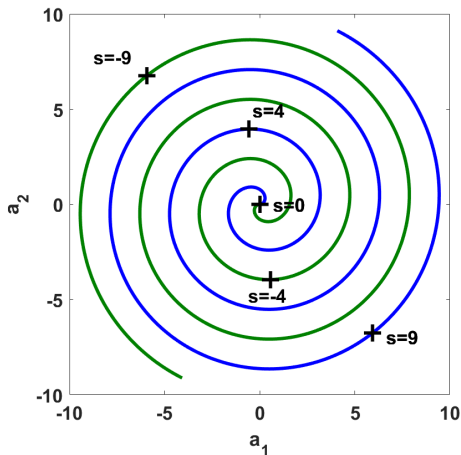


1 : 2 JSCC: Rate-Mismatched Case

$$\begin{cases} a_1(s) = s \\ a_2(s) = s \end{cases}$$



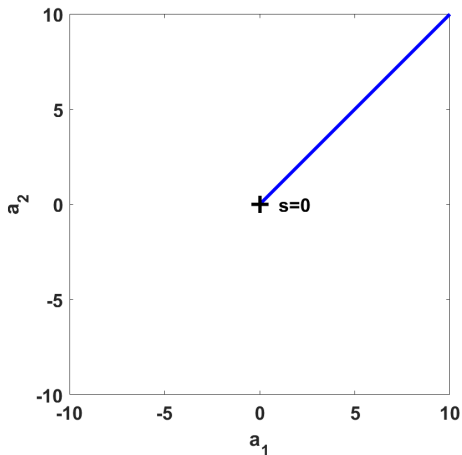
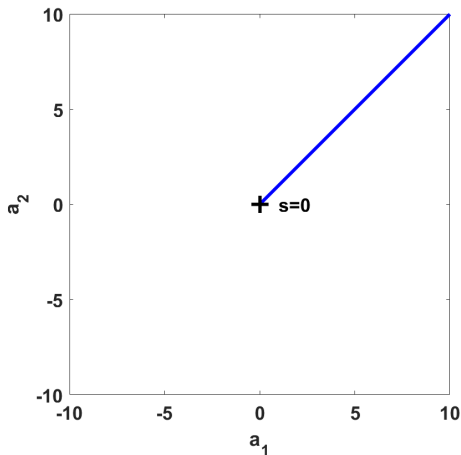
$$\begin{cases} a_1(s) = s \cos(2s) \\ a_2(s) = s \sin(2s) \operatorname{sign}(s) \end{cases}$$



1 : 2 JSCC: Rate-Mismatched Case

$$\begin{cases} a_1(s) = s \\ a_2(s) = s \end{cases}$$

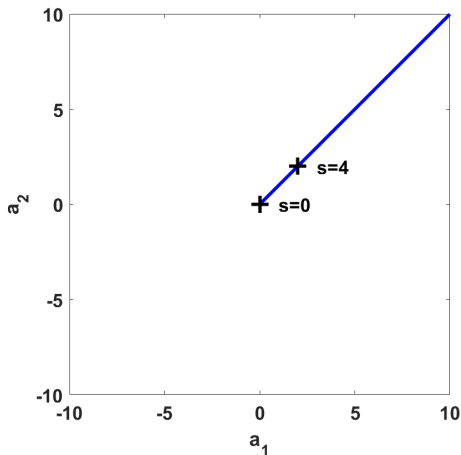
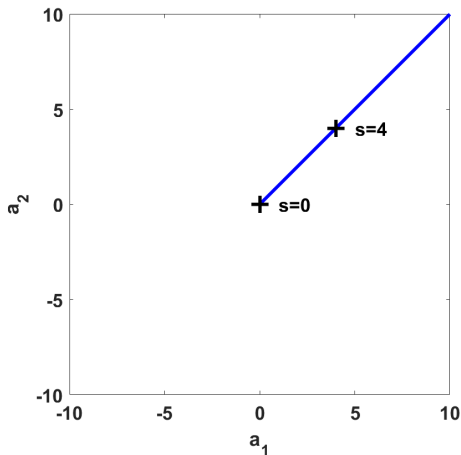
$$\begin{cases} a_1(s) = \sqrt{s} \\ a_2(s) = \sqrt{s} \end{cases}$$



1 : 2 JSCC: Rate-Mismatched Case

$$\begin{cases} a_1(s) = s \\ a_2(s) = s \end{cases}$$

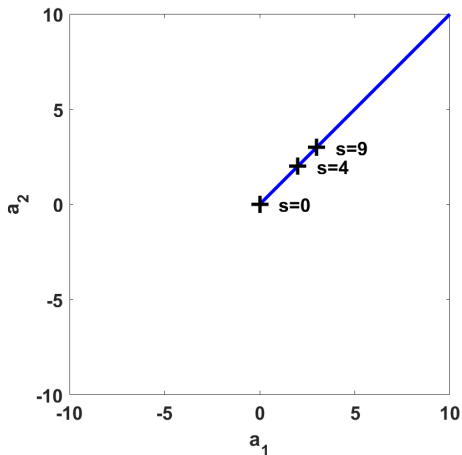
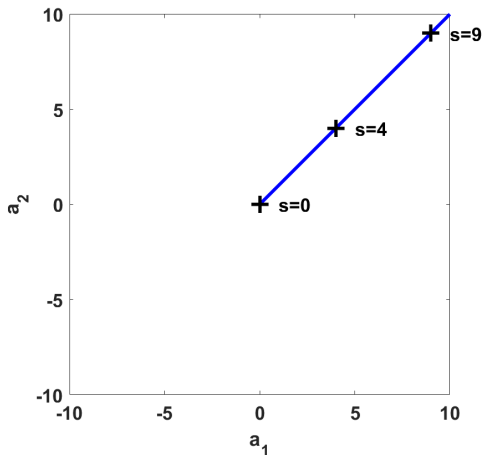
$$\begin{cases} a_1(s) = \sqrt{s} \\ a_2(s) = \sqrt{s} \end{cases}$$



1 : 2 JSCC: Rate-Mismatched Case

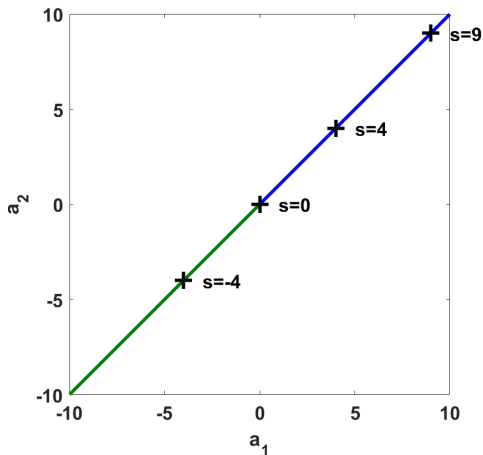
$$\begin{cases} a_1(s) = s \\ a_2(s) = s \end{cases}$$

$$\begin{cases} a_1(s) = \sqrt{s} \\ a_2(s) = \sqrt{s} \end{cases}$$

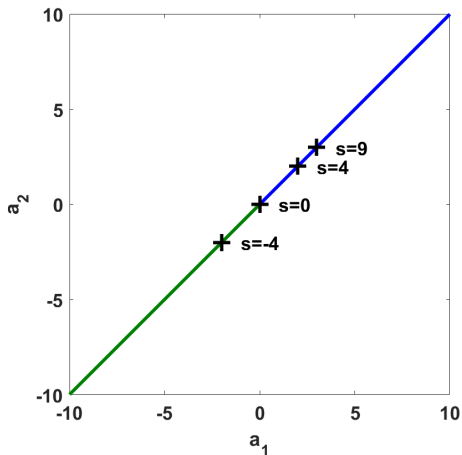


1 : 2 JSCC: Rate-Mismatched Case

$$\begin{cases} a_1(s) = s \\ a_2(s) = s \end{cases}$$



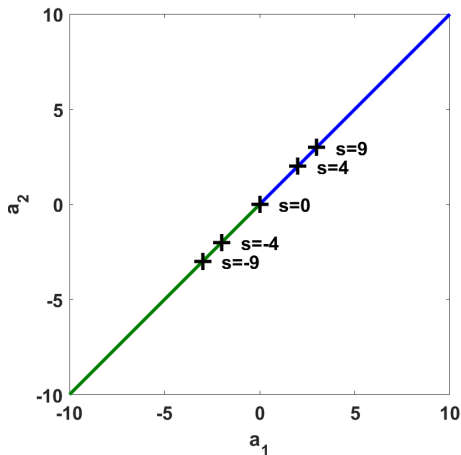
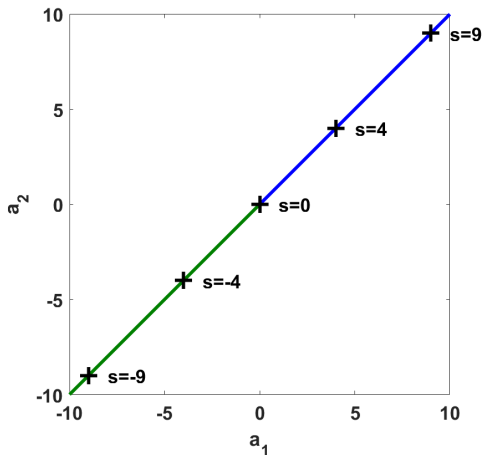
$$\begin{cases} a_1(s) = \sqrt{|s|} \text{sign}(s) \\ a_2(s) = \sqrt{|s|} \text{sign}(s) \end{cases}$$



1 : 2 JSCC: Rate-Mismatched Case

$$\begin{cases} a_1(s) = s \\ a_2(s) = s \end{cases}$$

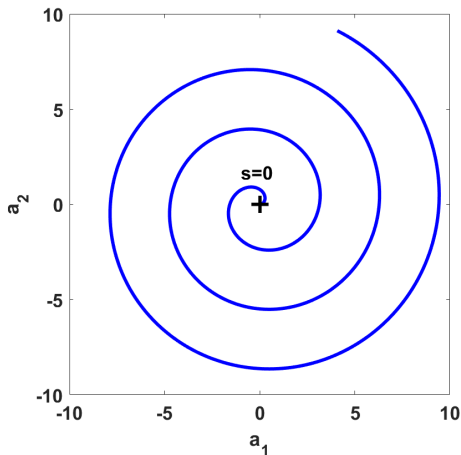
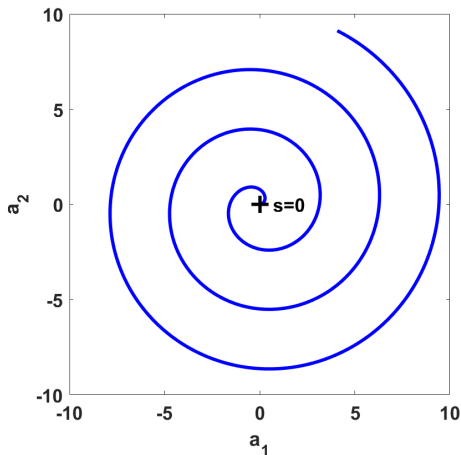
$$\begin{cases} a_1(s) = \sqrt{|s|} \operatorname{sign}(s) \\ a_2(s) = \sqrt{|s|} \operatorname{sign}(s) \end{cases}$$



1 : 2 JSCC: Rate-Mismatched Case

$$\begin{cases} a_1(s) = s \cos(2s) \\ a_2(s) = s \sin(2s) \end{cases}$$

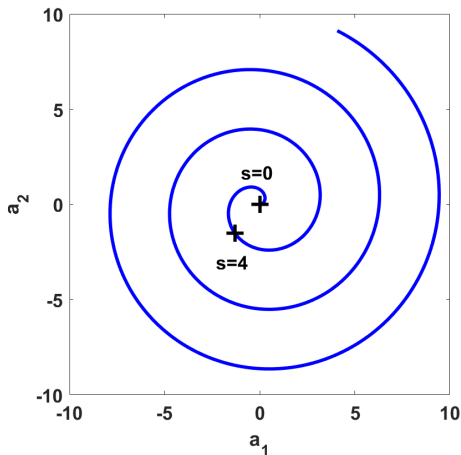
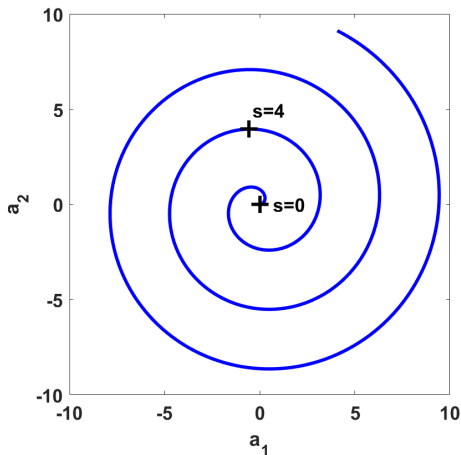
$$\begin{cases} a_1(s) = \sqrt{s} \cos(2\sqrt{s}) \\ a_2(s) = \sqrt{s} \sin(2\sqrt{s}) \end{cases}$$



1 : 2 JSCC: Rate-Mismatched Case

$$\begin{cases} a_1(s) = s \cos(2s) \\ a_2(s) = s \sin(2s) \end{cases}$$

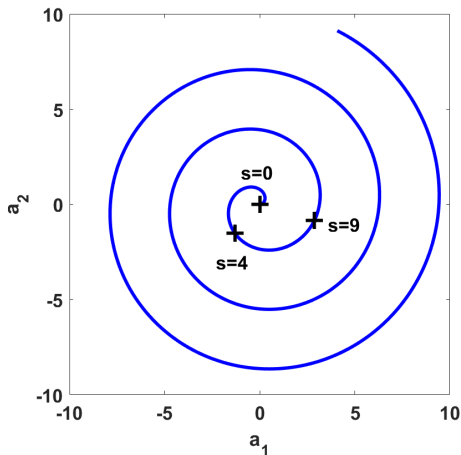
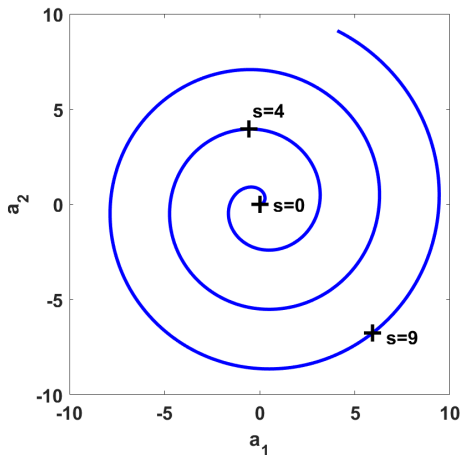
$$\begin{cases} a_1(s) = \sqrt{s} \cos(2\sqrt{s}) \\ a_2(s) = \sqrt{s} \sin(2\sqrt{s}) \end{cases}$$



1 : 2 JSCC: Rate-Mismatched Case

$$\begin{cases} a_1(s) = s \cos(2s) \\ a_2(s) = s \sin(2s) \end{cases}$$

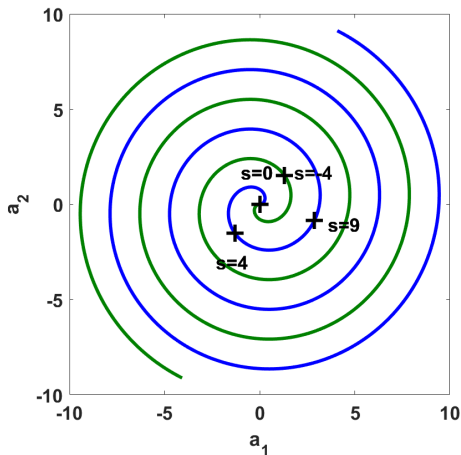
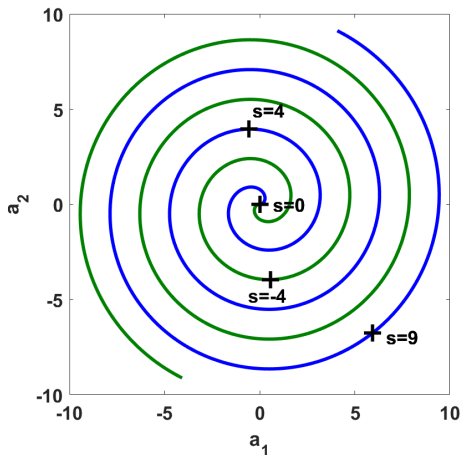
$$\begin{cases} a_1(s) = \sqrt{s} \cos(2\sqrt{s}) \\ a_2(s) = \sqrt{s} \sin(2\sqrt{s}) \end{cases}$$



1 : 2 JSCC: Rate-Mismatched Case

$$\begin{cases} a_1(s) = s \cos(2s) \\ a_2(s) = s \sin(2s) \operatorname{sign}(s) \end{cases}$$

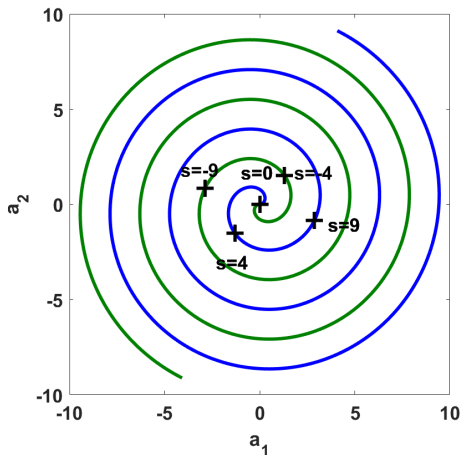
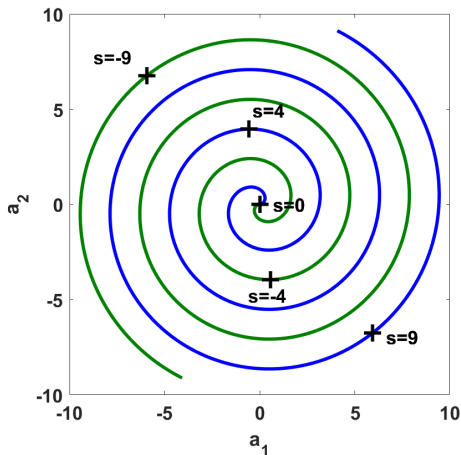
$$\begin{cases} a_1(s) = \sqrt{|s|} \cos(2\sqrt{|s|}) \operatorname{sign}(s) \\ a_2(s) = \sqrt{|s|} \sin(2\sqrt{|s|}) \operatorname{sign}(s) \end{cases}$$



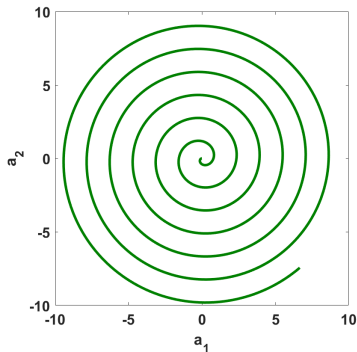
1 : 2 JSCC: Rate-Mismatched Case

$$\begin{cases} a_1(s) = |s| \cos(2|s|) \operatorname{sign}(s) \\ a_2(s) = |s| \sin(2|s|) \operatorname{sign}(s) \end{cases}$$

$$\begin{cases} a_1(s) = \sqrt{|s|} \cos(2\sqrt{|s|}) \operatorname{sign}(s) \\ a_2(s) = \sqrt{|s|} \sin(2\sqrt{|s|}) \operatorname{sign}(s) \end{cases}$$



1 : 2 JSCC: Rate-Mismatched Case

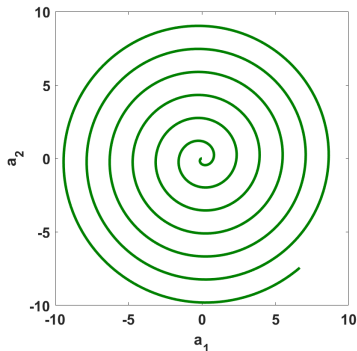


- Small distance between branches
 \Rightarrow better for “weak noise”
- Large distance between branches
 \Rightarrow better for “strong noise”

Standard spiral

$$\begin{cases} a_1(s) \propto s \cos(\omega s) & = |s| \cos(\omega |s|) \text{sign}(s) \\ a_2(s) \propto s \sin(\omega s) \text{sign}(s) & = |s| \sin(\omega |s|) \text{sign}(s) \end{cases}$$

1 : 2 JSCC: Rate-Mismatched Case



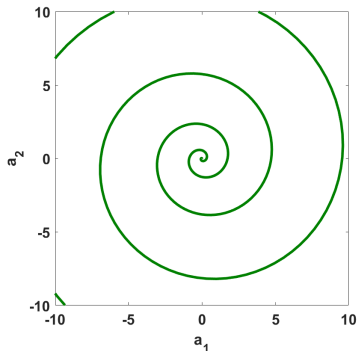
- Small distance between branches
 \Rightarrow better for “weak noise”
- Large distance between branches
 \Rightarrow better for “strong noise”

Stretched-source spiral

Stretch input before mapping to spiral: $s \rightarrow |s|^\lambda \text{sign}(s)$

$$\begin{cases} a_1(s) \propto |s|^\lambda \cos(\omega |s|^\lambda) \text{sign}(s) \\ a_2(s) \propto |s|^\lambda \sin(\omega |s|^\lambda) \text{sign}(s) \end{cases}$$

1 : 2 JSCC: Rate-Mismatched Case



⇐ Control requirements

- Small distance between branches
⇒ better for “weak noise”
- Large distance between branches
⇒ better for “strong noise”

Bounded average distortion given any input

Avoid increase in distortion with $|s|$ ⇒ Slower rotation with $|s|$

$$\begin{cases} a_1(s) \propto |s|^{\lambda\beta} \cos(\omega|s|^\lambda) \text{sign}(s) \\ a_2(s) \propto |s|^{\lambda\beta} \sin(\omega|s|^\lambda) \text{sign}(s) \end{cases}$$

1 : 2 JSCC: Rate-Mismatched Case

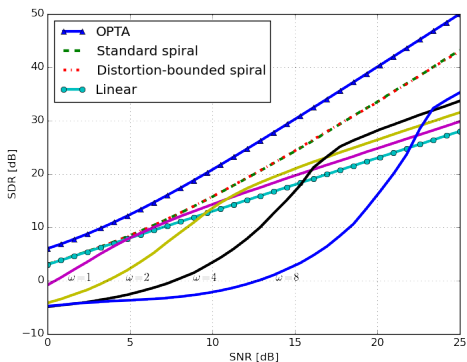
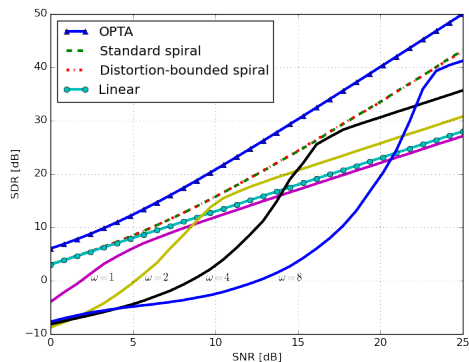


⇐ Control requirements

- Small distance between branches
⇒ better for “weak noise”
- Large distance between branches
⇒ better for “strong noise”

1 : 2 JSCC: Rate-Mismatched Case

- Average distortion given (almost) any s needs to be small!
- E.g., transmitters that truncate the signal do not perform well (avalanche effect)

(a) $\lambda = 1$.(b) $\lambda = 0.5$.

1 : 2 JSCC: Rate-Mismatched Case

Inner bound: Black-box approach

Assume a JSCC scheme with bounded distortion $D = \frac{1}{\text{SNR}^{\text{eff}}}, \forall s$.
Then,

$$\bar{J}^r \leq \bar{J}^t + \frac{Q + (\alpha^2 - 1) S}{1 + \text{SNR}^{\text{eff}} - \alpha^2} (P_t^t - \bar{P}_t^t)$$

Outer bound: Extension of [Kostina-Hassibi Allerton'16]

$$\bar{J}^r \geq \bar{J}^t + \frac{Q + (\alpha^2 - 1) S}{1 + \text{SNR}_{n \rightarrow \infty}^{\text{eff}} - \alpha^2} (P_t^t - \bar{P}_t^t)$$

- $1 + \text{SNR}_{n \rightarrow \infty}^{\text{eff}} = (1 + \text{SNR})^2$

- Difference between bounds is only due to effective SNR

Further Results and Future Research

- Inner bound can be improved: Optimization over curves, e.g. [Akyol-Vishwanatha-Rose-Ramstad IT'14]
- Outer bound for low-delay JSCC can be improved [Ziv-Zakai IT'73]
- High dimensional curves
- Other low-delay JSCC techniques: e.g., repetitive quantization [Kleiner-Rimoldi GLOBECOM'09]
 - Easy to generalize to higher dimensions
- Vector \mathbf{x} , vector \mathbf{u} , scalar y : *Simple extension of scalar setting!*
- Rate-matched case with vector \mathbf{y} : “n : 1 JSCC” is needed
 - Switch roles between Transmitter and Receiver
 - Improves over [Freudenberg-Middleton-Solo AC'10]