Information-Theoretic Control Through the Lens of Reinforcement Learning

Photios A. Stavrou

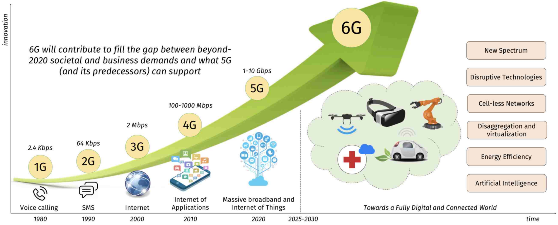
Department of Communication Systems
Algorithms & Foundations Group

EURECOM
Sophia Antipolis

19^{th} of June, 2025

Axis 5: Theoretical foundations of future communication networks
(National Center on Networks and Systems for Digital Transformation)

6G: From Connected Human and Things to Connected Intelligence[1]



Evolution of cellular network generation, from 1G to the envisioned 6G networks. Courtesy of Giordani et al.[2]

Trend towards future AI-native connect-compute systems

- Embedding physical, digital, and human worlds into the same ecosystem
- Moving from connected things to **connected intelligence**
- Enabling **pervasive** AI services, e.g., holographic communication, autonomous systems, connected robotics, wireless brain-computer devices, augmented reality, etc.

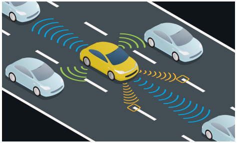
[1] W. Tong and P. Zhu, *6G: New Horizon- From connected people and things to connected intelligence [White paper]*,Available Online, 2021

[2] M. Giordani et al., *Toward 6G Networks: Use Cases and Technologies* IEEE Communications Magazine, vol. 58, no. 3, pp. 55-61, March 2020.

From Sensing to Decision and Control

Applications...

Factory Automation



Autonomous vehicles

Tele-surgery

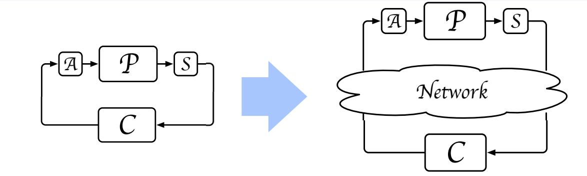
- ☞ Too much information gathered from network sensing; transform it into effective decisions (e.g., autonomous vehicles are envisioned to generate up to 4TB of data per day/each day!)
- ☞ Network limitations determine how to sense, process, and act on data

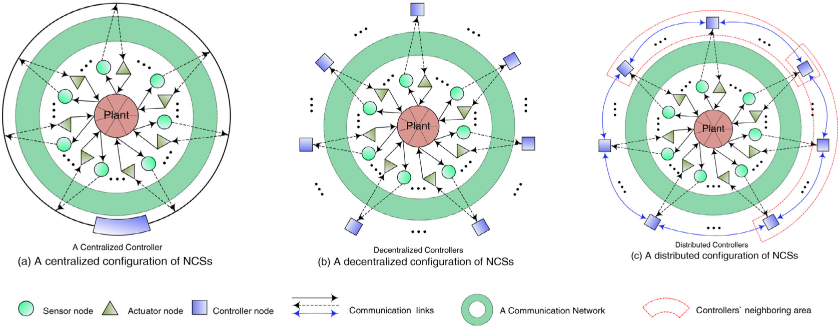
Several Issues/Challenges

- ➤ Communication constraints (e.g., limited bandwidth, quantization, coding, packet losses, delays)
- ➤ Co-design of communication and control
- ➤ Security and privacy
- ➤ Scalability and Complexity
- ➤ Stability and robustness
- ➤ Energy and resource efficiency
- ➤ Heterogeneity
- ➤ Real-time requirements.

Networked Control Systems

Networked Control Systems (NCSs) are spatially distributed systems in which control loops are closed through a wireless communication network as follows





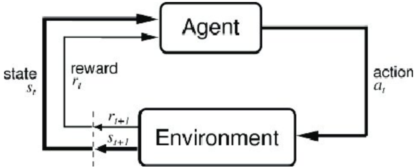
(a) A centralized configuration of NCSs

(b) A decentralized configuration of NCSs

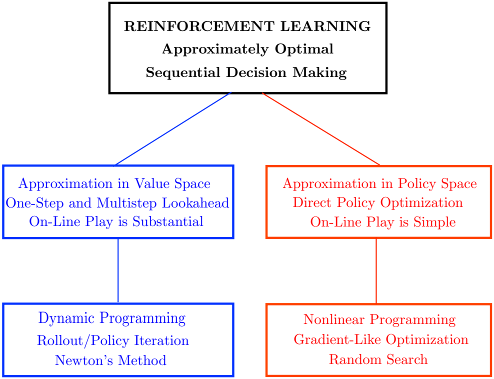
(c) A distributed configuration of NCSs

Why is Reinforcement Learning relevant in NCSs?

- Adaptive to Dynamic Environments (often without the need to know the dynamical model)
- Operate with or without needing a mathematical model of the network (model-based or model-free optimization)
- RL naturally frames problems as Markov decision models
- RL algorithms offer scalability to high-dimensional control (Distributed multi-agent systems, deep RL, etc)



Reinforcement Learning in a Nutshell



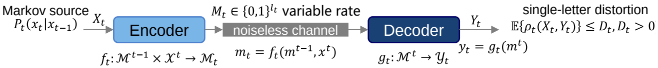
RL deals with exactly the same mathematical problem as DP

Courtesy of D. Bertsekas[1]

- ☞ **Approximation in value space:** We aim at learning the best value or cost function and indirectly improve the policy
- ☞ **Approximation in policy space:** Aims at directly optimizing to find the best policy or its approximate value

[1] D. Bertsekas, *Reinforcement learning and optimal control* Athena Scientific, 2019.

Case Study: The Zero-delay Lossy Compression Problem



A discrete-time zero-delay lossy source coding system

☞ We encode causally, followed by Huffman coding, and again decode causally[7,8]

Empirical Rates

The empirical rate for each fidelity D_t over the whole horizon $\{0, 1, \ldots, n\}$ is given by

$$R^{op}_{[0,n]}(D_0, D_1, \ldots, D_n) = \inf_{f_t,\ g_t\colon\ \mathbf{E}[\rho_t(X_t,Y_t)] \leq D_t,\ \forall t} \frac{1}{n+1}\sum_{t=0}^{n} R_t, \quad R_t = \mathbf{E}[\ell_t]$$

Achievable Bound

- Method 1: Upper bounds on the empirical rates using reinforcement learning techniques
- **Method 2:** Consider a sequential version of SFRL and one-shot achievability

$$R^{op}_{[0,n]}(D_0,D_1,\ldots,D_n) \geq R^{na}_{[0,n]}(D_0,D_1,\ldots,D_n) + \log\Big(R^{na}_{[0,n]}(D_0,D_1,\ldots,D_n)+1\Big) + 6 \qquad (1)$$

[1] Z. He, C. D. Charalambous, and P. A. Stavrou *A new finite-horizon dynamic programming analysis of nonanticipative rate-distortion function for Markov sources*, ECC 2025 (to appear).

Lower Bound

Causal Rate Distortion Function

For each fidelity D_t over the whole horizon $\{0, 1, \ldots, n\}$, the following lower bound holds

$$R^{op}_{[0,n]}(D_0, D_1, \ldots, D_n) \geq R^{na}_{[0,n]} = \inf_{\substack{P_t(y_t|x_t, y_{t-1}): \\ \mathbf{E}[\rho_t(X_t, Y_t)] \leq D_t, \ \forall t}} \frac{1}{n+1} I(X^n \rightarrow Y^n)$$

where $I(X^n \rightarrow Y^n) = \sum_{t=0}^{n} I(X_t; Y_t | Y_{t-1})$

☞ Problem under certain conditions is convex (assuming the past posteriors at each instant of time are given)

Element	Description
Information state b_t	$P_t(x_{t-1}\|y_{t-1})$
Disturbance w_t	$P_t(x_t\|x_{t-1})$
Feedback control policy μ_t	$P_t(y_t\|y_{t-1}, x_t)$
Cost $g_t(b_t, \mu_t)$	$\log\left(\frac{P_t(y_t\|y_{t-1}, x_t)}{P_t(y_t\|y_{t-1})}\right) - s_t(\rho_t(x_t, y_t) - D_t[y_{t-1}, b_t])$

Information-State MDP (POMDP)

Stochastic DP Algorithm

(Offline training-Backward in Time)

Terminal stage: $R_n(D_n[y_{n-1}, b_n]) = \min_{\mu_n} \mathbf{E}\{g_n(b_n, \mu_n)\}$

Cost-to-go: $R_t(D_t[y_{t-1}, b_t]) = \min_{\mu_t} \mathbf{E}\{(g_t(b_t, \mu_t) + R_{t+1}(D_{t+1}[y_t, b_{t+1}]))\}$

where

$$b_{t+1} = f_t(b_t, \mu_t, w_t)$$

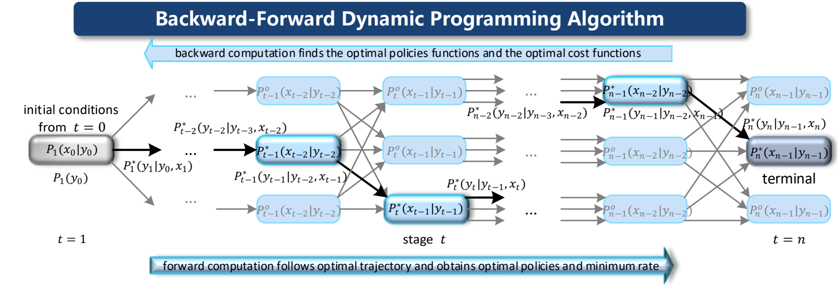
(Online Computation-Forward in Time)

$$\mu_t^* \in \arg\min_{\mu_t} \mathbf{E}\left\{g_t(b_t, \mu_t) + R_{t+1}^*(D_{t+1}[y_t, b_{t+1}])\right\}, \ t = 0, 1, \ldots, n$$

- The above finite horizon stochastic DP recursions are subject to a continuous state (e.g., $b_t \in [0, 1], \ \forall t$)
- We can use approximation methods[1], e.g., directly discretizing the belief-state
- In the sequel, I will restrict myself to discrete alphabets

[2] D. Bertsekas, *Reinforcement learning and optimal control* Athena Scientific, 2019.

Approximation in Policy Space



Algorithm 1 Approximation of the Control Policy Backward in Time (Offline Training)

Input: $\{P_t(x_t|x_{t-1}): \ t \in \mathbb{N}_0^n\}$, $\{s_t \leq 0: \ t \in \mathbb{N}_0^n\}$, given belief state $P_t^o(x_{t-1}|y_{t-1}) \in \mathcal{B}_t$, $\epsilon > 0$.

Initialize $\{P_t^{(0)}(y_t|y_{t-1}) : t \in \mathbb{N}_0^n\}$
for $t = n : 1$ **do**
  $k \leftarrow 0$
  while $T_{L_t}[y_{t-1}, P_t^o] - T_{U_t}[y_{t-1}, P_t^o] > \epsilon$ **do**
    $P_t^{(k)}(y_t|y_{t-1}, x_t) \leftarrow$ (20)
    $P_t^{(k+1)}(y_t|y_{t-1}) \leftarrow$ (21)
    $R_t(D_t[y_{t-1}, P_t^o]) \leftarrow$ (22)
    $k \leftarrow k + 1$
  end while
end for

Output:
$\{P_t^*[P_t^o](y_t|y_{t-1}, x_t) : t \in \mathbb{N}_1^n\}$, $\{P_t^*[P_t^o](y_t|y_{t-1}) : t \in \mathbb{N}_1^n\}$, $\{R_t(D_{s_t}[y_{t-1}, P_t^o]) : t \in \mathbb{N}_1^n\}$.

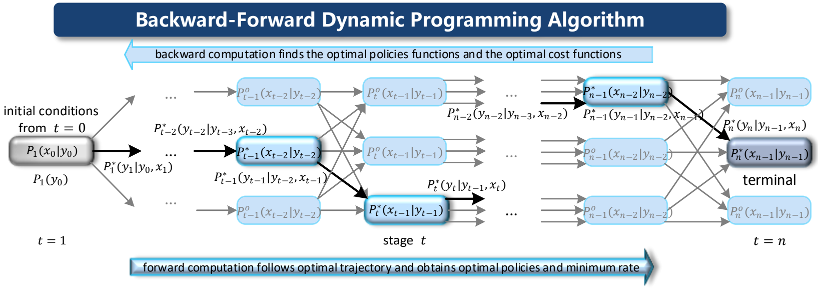
Pros:

- ✓ We discretize the belief-state
- ✓ We apply a stage-wise alternating minimization to obtain the best (approximate) policy functions
- ✓ Provable convergence guarantees for any backward horizon

Cons:

- ✓ Computationally expensive (exponential increase in the computation when increasing your discretization set)

Approximation in Policy Space



Algorithm 2 Forward Computation of the Approximate Control Policy (Online Computation)

Input: $\{\mathcal{B}_t : t \in \mathbb{N}_1^n\}$ of given $\{P_t^o(x_{t-1}|y_{t-1}) : t \in \mathbb{N}_1^n\}$, outputs of Algorithm 1.

Initialize $P_0(x_0)$, $P_0(y_0)$, $P_1^*(x_0|y_0) = P(x_0|y_0)$

for $t = 1 : n-1$ **do**

  $P_{t+1}^*(x_t|y_t) \leftarrow$ (26)

  $P_t^*(y_t|y_{t-1}, x_t) \leftarrow$
    $P_t^*[P_t^*(x_{t-1}|y_{t-1}), P_{t+1}^*(x_t|y_t)](y_t|y_{t-1}, x_t)$

end for

$P_n^*(y_n|y_{n-1}, x_n) \leftarrow P_n^*[P_n^*(x_{n-1}|y_{n-1})](y_n|y_{t-1}, x_n)$

Output:
  $\{P_t^*(x_{t-1}|y_{t-1}) : t \in \mathbb{N}_0^n\}$, $\{P_t^*(y_t|y_{t-1}, x_t) : t \in \mathbb{N}_0^n\}$, $R_{[0,n]}^{na}(D_0, D_1, \ldots, D_n)$.

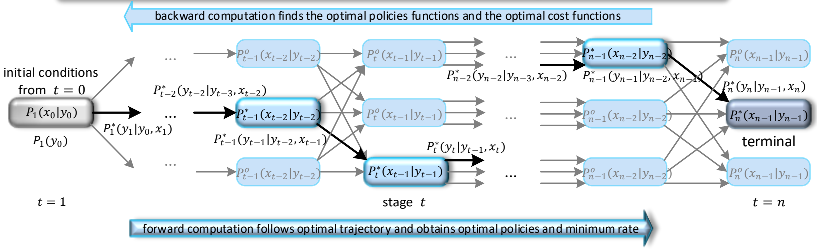
Pros:

- ✓ Light-speed computation (simple computations)

Cons:

- ✓ Does not allow for online re-planning

Backward-Forward Dynamic Programming Algorithm

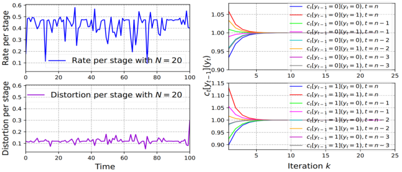


Parallel computation
for backward training

❖ **Settings**

- binary alphabet $\{\mathcal{X}_t = \mathcal{Y}_t = \{0, 1\}: t \in \mathbb{N}_0^n\}$.
- Hamming distortion metric $\rho_t(x_t, y_t) = \rho(x_t, y_t) = \begin{cases} 0, & \text{if } x_t = y_t \\ 1, & \text{if } x_t \neq y_t \end{cases}$.

☞ **Example 1**. Time-varying binary symmetric Markov source



(a) Stagewise rate & distortion (b) Stagewise convergence

- belief state space $\mathcal{B}_t$ with quantization level $|\mathcal{B}_t| = N = 20$
- Lagrange multiplier $s_t = s = -2$
- time horizon $n = 100$

Approximation in Policy Space: Interpretable, Explainable, and Trustworthy Model-Based RL

✓ **Interpretable**

- ☞ Policies operate over explicit belief states $P_t(x_{t-1} \mid y_{t-1})$
- ☞ Feedback Control laws are structured and visualizable
- ☞ No black-box networks-fully transparent policy structure

✓ **Explainable**

- ☞ Learning via Alternating Minimization with mathematical grounding
- ☞ Each step has semantic meaning (e.g., distortion matching)
- ☞ Derived from KKT conditions and dynamic programming

✓ **Trustworthy**

- ☞ Offline optimization with convergence guarantees
- ☞ Online execution is deterministic and efficient
- ☞ Learning and deployment are cleanly decoupled

✓ **Goal-Aware (Semantic Layer)**

- ☞ Policies preserve only task-relevant information
- ☞ Semantic rate-distortion ensures minimal, purposeful encoding
- ☞ Supports explainable pruning of irrelevant details

Q-Factor Recursions

Stochastic DP Algorithm via Q-Factors

(Offline training-Backward in Time)

$$Q_t^*(b_t, \mu_t) = \mathbf{E}\left\{ g_t(b_t, \mu_t) + \min_{\mu_{t+1}} Q_{t+1}^*(b_{t+1}, \mu_{t+1}) \right\}$$

with the terminal condition $Q_{t+1}^*(b_{t+1}, \mu_{t+1}) = 0$ when $t = N$.

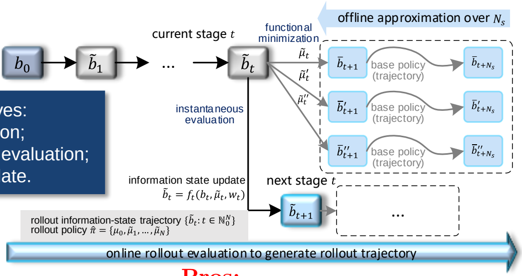
(Online Computation-Forward in Time)

$$\mu_t^*(b_t) = \arg\min_{\mu_t} Q_t^*(b_t, \mu_t), \; t = 0, 1, \ldots, N$$

☞ We will tackle the problem assuming approximate DP with truncated rollout[9]

Approximation in Value Space via Truncated Rollout

The online **rollout** involves:
1) Functional minimization;
2) Instantaneous policy evaluation;
3) Information state update.



Algorithm 1 Offline Base Control Policy Approximation

Input: given $\{w_t : t \in \mathbb{N}^N_{N-N_s+1}\}$,
given base information state $\bar{b}_t \in \bar{\mathcal{B}}_t$, Lagrange multipliers $\{s_t \leq 0 : t \in \mathbb{N}^N_{N_s}\}$, error tolerance $\epsilon > 0$

Initialize $\{\nu_t^{(0)} : t \in \mathbb{N}^N_{N-N_s+1}\}$
for $t = N : N - N_s + 1$ **do**
  $k \leftarrow 0$
  while $T_{U_t}[u^{t-1}, b_t] - T_{L_t}[u^{t-1}, b_t] > \epsilon$ **do**
    $\mu_t^{(k)} \leftarrow$ (25)
    $\nu_t^{(k+1)} \leftarrow$ (26)
    $Q_t(b_t, \mu_t^{(k)}) \leftarrow$ (27)
    $k \leftarrow k + 1$
  end while
end for
$Q^{\bar{\pi}}_{N_s}(b_t, \mu_t) \leftarrow Q^*_{N_s}[g_t, Q^*_{N_s+1}](b_t, \mu_t^*)$

Output: $\{\mu_t^*(b_t) : t \in \mathbb{N}^N_{N-N_s+1}, b_t \in \bar{\mathcal{B}}_t\}$,
$\{\nu_t^*[b_t] : t \in \mathbb{N}^N_{N-N_s+1}, b_t \in \bar{\mathcal{B}}_t\}$,
$\{Q^{\bar{\pi}}_{N_s}(b_t, \mu_t) : b_t \in \bar{\mathcal{B}}_t, \mu_t \in \mu_t^*(b_t)\}$.

Pros:

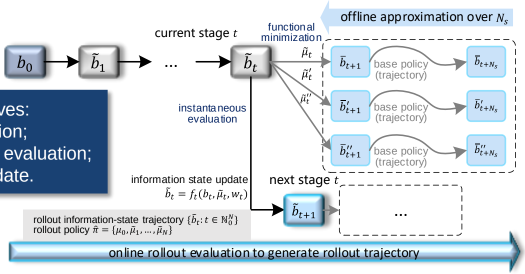
- ✓ No need for full discretization of the belief-state
- ✓ Stable and repeatable method
- ✓ Memory efficient
- ✓ Provable convergence guarantees for any rolling horizon

Cons:

- ✓ Approximation due to truncation of the horizon
- ✓ Dependent on the discretization
- ✓ Pretraining is required

Approximation in Value Space via Truncated Rollout

The online **rollout** involves:
1) Functional minimization;
2) Instantaneous policy evaluation;
3) Information state update.



Algorithm 2 Online Rollout Evaluation

Input: $\{\mathcal{B}_t : t \in \mathbb{N}^N_{N-N_s+1}\}$ of given $\{b_t : t \in \mathbb{N}^N_{N-N_s+1}\}$,
$\{\mu^*_t(b_t) : t \in \mathbb{N}^N_{N-N_s+1}, b_t \in \mathcal{B}_t\}$,
$\{\nu^*_t[b_t] : t \in \mathbb{N}^N_{N-N_s+1}, b_t \in \mathcal{B}_t\}$,
$\{Q^{\bar{\pi}}_{N_s} : b_t \in \mathcal{B}_t\}$.

Initialize $\mu_0 = P_0(u_0|x_0)$, $P_1(u^0)$, $\tilde{b}_1 = P(x_0|u_0)$
for $t = 1 : N$ **do**
  $\tilde{Q}^{\bar{\pi}}_t(\tilde{b}_t, \mu_t) \leftarrow$ step 3-9 in Algorithm 1
  $\tilde{\mu}_t \leftarrow$ (31)
  $\tilde{b}_{t+1} \leftarrow$ (3)
end for

Output: $\hat{\pi} = \{\mu_0, \tilde{\mu}_1, \ldots, \tilde{\mu}_N\}$, $\{\tilde{b}_t, t \in \mathbb{N}^N_1\}$,
$\{\tilde{\nu}_t : t \in \mathbb{N}^N_0\}$, $C^{\tilde{\pi}}(X^N, U^N)$.

Pros:

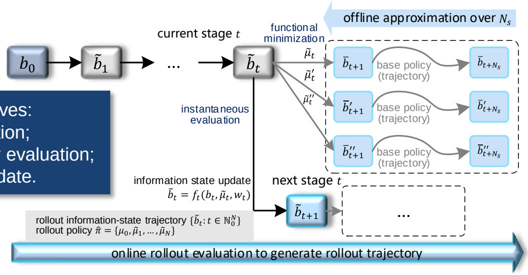
- ✓ Policy improvement via one step lookahead minimization
- ✓ Allows for online re-planning (real time adaptivity)
- ✓ Scalable and stable method

Cons:

- ✓ Computationally expensive
- ✓ Relies on the quality of the base policy
- ✓ No long-term guarantees

Approximation in Value Space via Truncated Rollout

The online **rollout** involves:
1) Functional minimization;
2) Instantaneous policy evaluation;
3) Information state update.

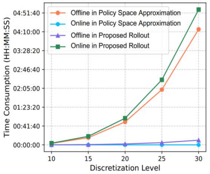


Settings

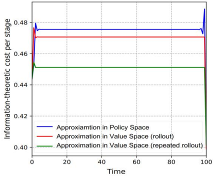
- binary alphabet $\{\mathcal{X}_t = \mathcal{Y}_t = \{0,1\} \colon t \in \mathbb{N}_0^N\}$
- Hamming distortion metric $\rho_t(x_t, y_t) = \rho(x_t, y_t) = \begin{cases} 0, & \text{if } x_t = y_t \\ 1, & \text{if } x_t \neq y_t \end{cases}$

Parallel computation
(Offline & Online)

Example 2. Time-invariant binary symmetric Markov source



(a) Time consumption



(b) Stagewise cost

- information-state space $\overline{\mathcal{B}}_t$ with quantization level $|\overline{\mathcal{B}}_t| = n = 20$
- Lagrange multiplier $s_t = s = -2$
- time horizon $N = 100$, $N_s = 5$

✓ stable RL approach
✓ good scalability

Q-factor Truncated Rollout: Interpretable, Explainable, and Trustworthy Model-Based RL

✓ **Interpretable**

- ☞ Explicit Q-factor functions over belief states and actions
- ☞ Policies derived by structured, transparent minimization
- ☞ Full visibility into how decisions depend on expected future cost

✓ **Explainable**

- ☞ Modular architecture: offline base policy + online rollout
- ☞ Each policy improvement step is locally justified and auditable
- ☞ Stage-by-stage reasoning: traceable Q-updates and decision logic

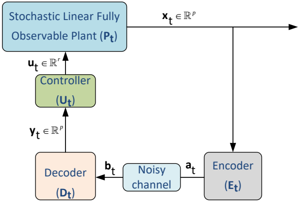
✓ **Trustworthy**

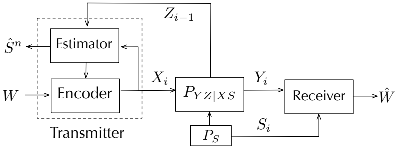
- ☞ Offline computation is stable, convergent, and verifiable
- ☞ Online rollout guarantees improvement over base policy
- ☞ Deterministic, certified decision-making at deployment

✓ **Goal-Aware (Semantic Information Structure)**

- ☞ Policies shaped by directed information and task-driven costs
- ☞ Prunes irrelevant information via semantic compression
- ☞ Enables transparent understanding of what matters for control

Possible collaboration opportunities: How can we jointly design and identify the fundamental limits of communication, sensing, and control?





Fundamental Questions...

- ☞ Consider Finite State Channels with feedback + sensing?
- ☞ Joint source-channel-control-sensing design?
- ☞ Low coding delays scenarios?

---

[3] M. Kobayashi *et al.*, *Joint state sensing and communication over memoryless MAC*, IEEE ISIT, 2019

[4] M. Ahmadipour *et al.*, *An information-theoretic approach to joint sensing and communication*, IEEE Tras. Info. Theory, 2023

[5] Y. Xiong *et al.*, *On the fundamental tradeoff of integrated sensing and communications under Gaussian channels*, IEEE Tras. Info. Theory, 2023

Thank you!

QUESTIONS

For more information:

Photios A. Stavrou (fotios.stavrou@eurecom.fr)

Acknowledgement: Part of this work has received funding from the European Commission (EC) under the EU's Horizon 2020 program (Grant Agreement No 101139232).