

Università degli Studi di Padova – Dipartimento di Ingegneria dell'Informazione
Corso di Laurea in Ingegneria Informatica

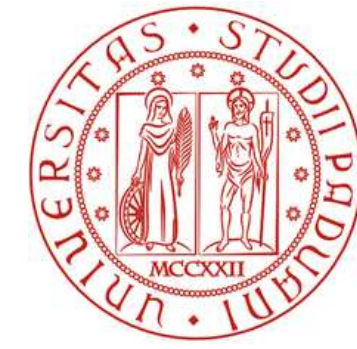
***Relazione per la prova finale
«Generazione di Chiave mediante Tecniche IA
su Canali Asimmetrici»***

Relatore: Prof. Stefano Tomasin
Correlatore: Prof. Francesco Ardizzon

Laureando: Gabriele Pascal D'Ambrosio
Matricola n.: 2101772

Padova, 21/07/2026

INTRODUZIONE: SICUREZZA WIRELESS



- Il segnale viene propagato nello spazio e potrebbe essere intercettato. Le tecniche crittografiche richiedono l'utilizzo di chiavi segrete.
- L'idea della **PKG** è di sfruttare l'informazione reciproca fra Alice e Bob, contenuta nel canale di comunicazione per generare chiavi più semplicemente.
- **PROBLEMA:** Le osservazioni fisiche di canale sono affette da Hardware Impairments.
- Mediante il Machine Learning è possibile ricostruire l'informazione utile del canale, compensando Impairments e rumore.
- **OBIETTIVO DEL LAVORO:** Migliorare la fase di Advantage Distillation, ottenendo chiavi vicine e distribuite in modo uniforme.

MODELLO DI CANALE PER L'IMPAIRMENT: $y_B = \alpha \cdot h \cdot (x_A + w_A) + n_B$

Il canale è asimmetrico, con Impairment significativo solo per Alice, mentre per Bob è praticamente assente.

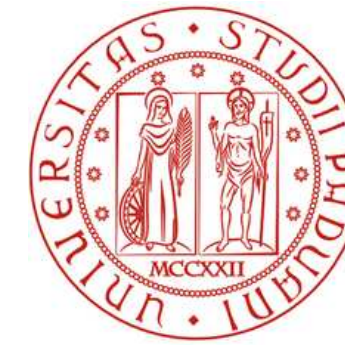
MODELLO DI CANALE **SENZA** IMPAIRMENT: $y_A = h \cdot x_B + n_A$

OSSERVAZIONI SUL CANALE (STIMA DI h):

- Stima Alice: $\hat{h}_A = \frac{y_A}{x} = h + \frac{n_A}{x}$

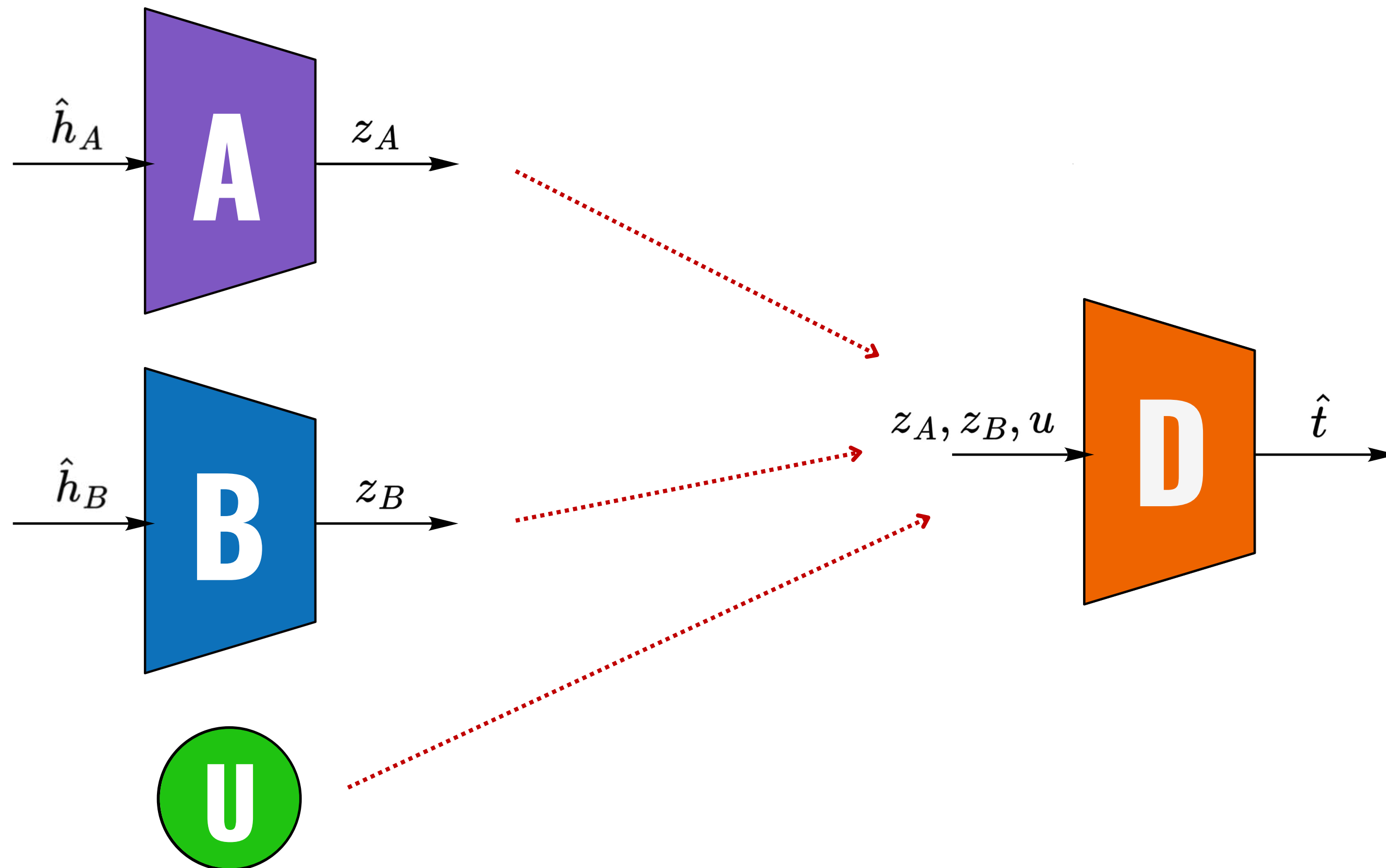
- Stima Bob: $\hat{h}_B = \frac{y_B}{x} = \alpha_A \cdot h + \frac{\alpha_A \cdot h \cdot w_A}{x} + \frac{n_B}{x}$

MODELLO MACHINE LEARNING: ADDESTRAMENTO AVVERSAIO

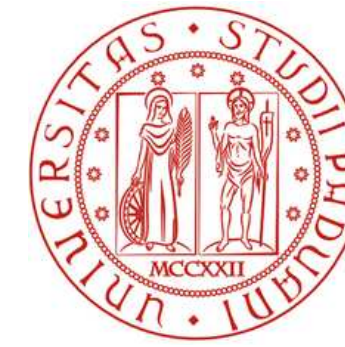


- **Due Encoder** distinti per Alice e Bob, allenati indipendentemente.
- Per la Reciprocità si sfrutta una funzione loss di tipo **MSE**.
- Per la Randomicità si introduce una nuova rete: **Discriminatore**.
- Il Discriminatore riceve le chiavi grezze di Alice e Bob e dati da una distribuzione uniforme.
- **Addestramento avversario**: Alice e Bob vogliono “ingannare” il Discriminatore, mentre il Discriminatore vuole classificare correttamente i dati.
- Per ingannare il Discriminatore, Alice e Bob devono distribuire le chiavi in modo più uniforme possibile.
- E' necessario che il Discriminatore “sappia fare il suo lavoro”, quindi bisogna **allenarlo**.

MODELLO MACHINE LEARNING: ADDESTRAMENTO AVVERSAARIO



MODELLO MACHINE LEARNING: FUNZIONI DI LOSS



ALICE E BOB $\longrightarrow$ $\mathcal{L}_A = (1 - \gamma) \cdot \mathcal{L}_1 + \gamma \cdot \mathcal{L}_2$

• **LOSS RECIPROCITA':** $\mathcal{L}_1 = \frac{1}{n} \sum_{i=1}^n (z_{Ai} - z_{Bi})^2$

• **LOSS RANDOMICITA':** $\mathcal{L}_2 = -\frac{1}{n} \sum_{i=1}^n \log(\hat{t}_i)$

DISCRIMINATORE $\longrightarrow$ $\mathcal{L}_D = \mathcal{L}_{real} + \beta \cdot (\mathcal{L}_{fakeA} + \mathcal{L}_{fakeB})$

• **LOSS REAL:** $\mathcal{L}_{real} = -\frac{1}{n} \sum_{i=1}^n \log(\hat{t}_i)$

• **LOSS FAKE:** $\mathcal{L}_{fakeA} = -\frac{1}{n} \sum_{i=1}^n \log(1 - \hat{t}_i)$

RISULTATI: ANDAMENTO LOSS

- **Alice e Bob:** Convergono rapidamente verso valori bassi (ca. 0.35)
- **Discriminatore:** Cresce verso valori alti e si stabilizza (ca 1.4)

Alice e Bob stanno imparando rappresentazioni **reciproche e stabili**, ingannando il Discriminatore.

Infatti dopo le prime epoche in cui impara a distinguere i dati, il suo lavoro viene complicato e fa **fatica** a distinguerli correttamente, assestandosi al risultato complessivo di circa 1.4.

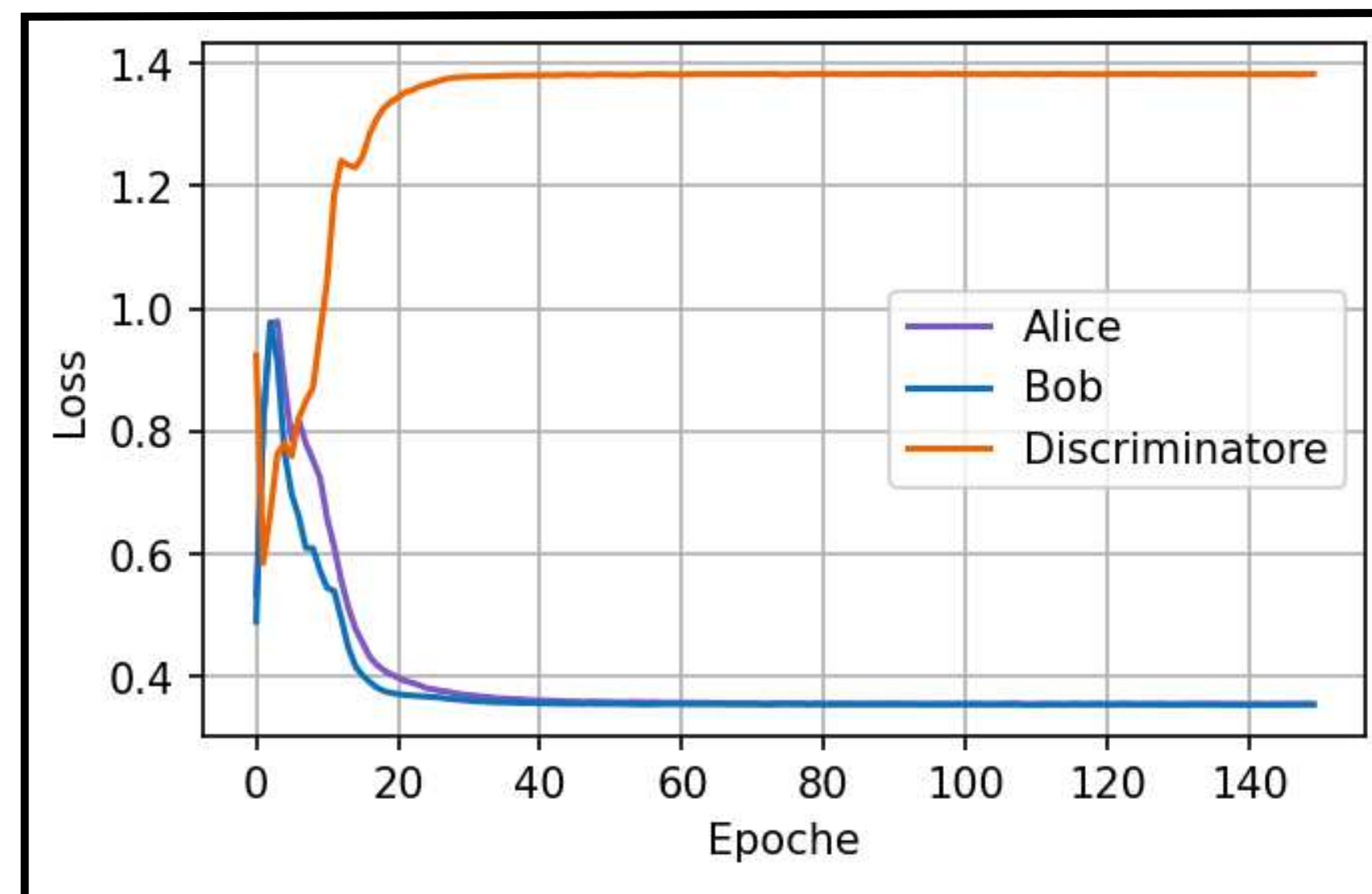


Fig. 1. Andamento Loss in funzione delle epoche, nel caso di addestramento con $\sigma_w^2 = \sigma_n^2 = 0.1$

$$\mathcal{L}_D = -\log(0.5) - 0.5[\log(0.5) + \log(0.5)] = -2 \cdot \log(0.5) \simeq 1.384$$

La distribuzione delle chiavi di Alice e Bob inizia ad assomigliare a quella di **uniforme**.

RISULTATI: DISTRIBUZIONE CHIAVI GREZZE

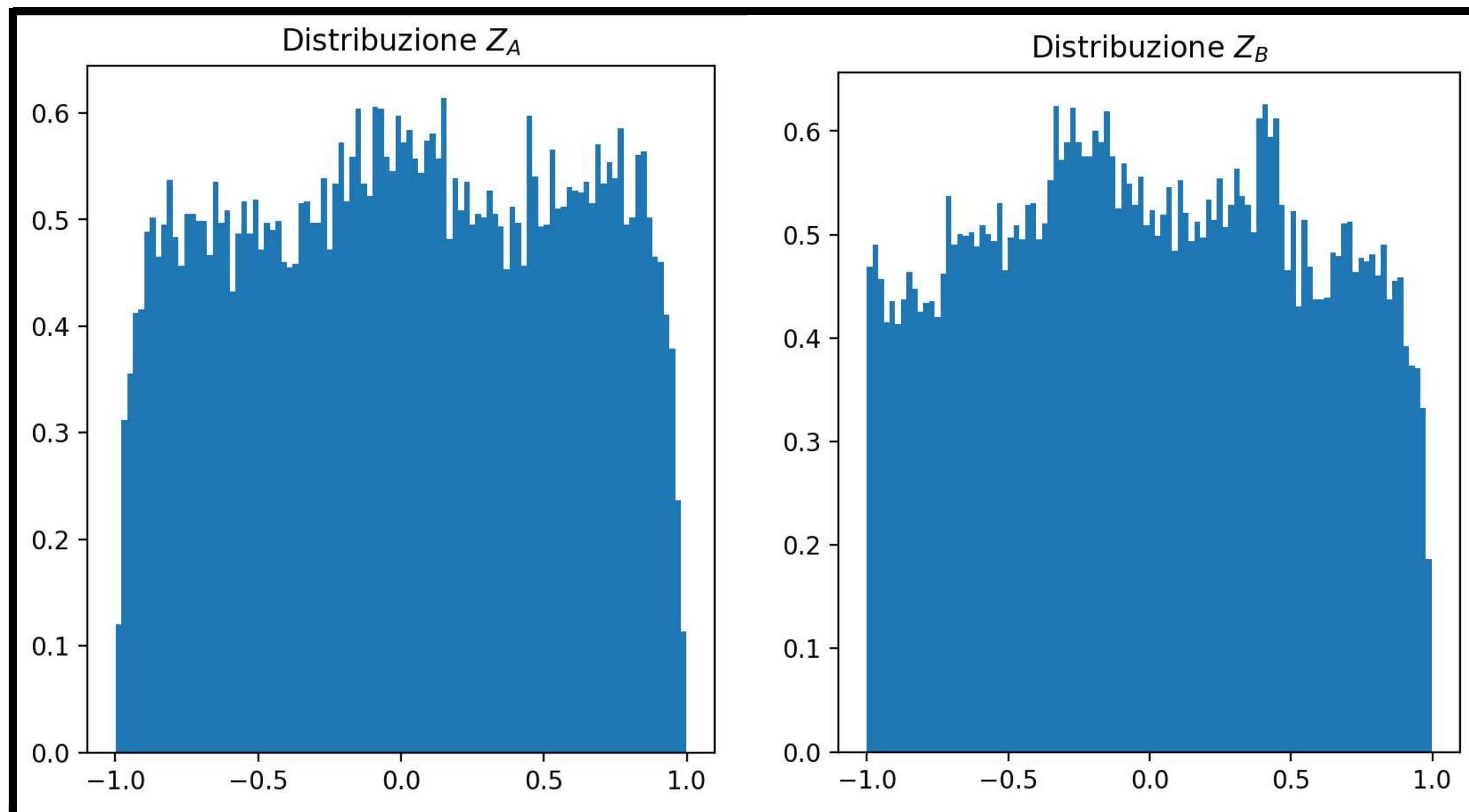
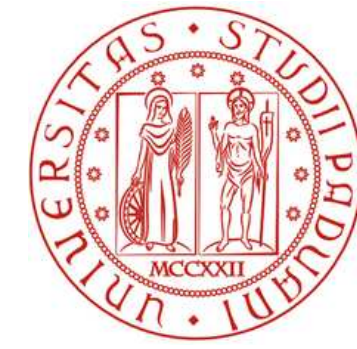


Fig. 2. Distribuzione delle chiavi grezze nello scenario $\sigma_w^2 = \sigma_n^2 = 0.1$

RISULTATI: MODELLO SNR



- Per valutare la reciprocità tra le chiavi grezze è necessario avere un sistema di riferimento, in grado di rappresentare la qualità del canale di comunicazione, in funzione di Impairments e rumore.
- E' fondamentale definire un modello di SNR.
- Il segnale utile nel modello è il coefficiente di canale h che Alice e Bob vogliono stimare.
- Il Noise invece è rappresentato dalla distanza (errore) fra le stime $\hat{h}_A, \hat{h}_B$.

$$SNR = E \left[\frac{||h||^2}{||\hat{h}_A - \hat{h}_B||^2} \right]$$

- Per stimare il SNR definito dal modello teorico, si utilizza una simulazione Monte Carlo.

RISULTATI: ANDAMENTO SNR

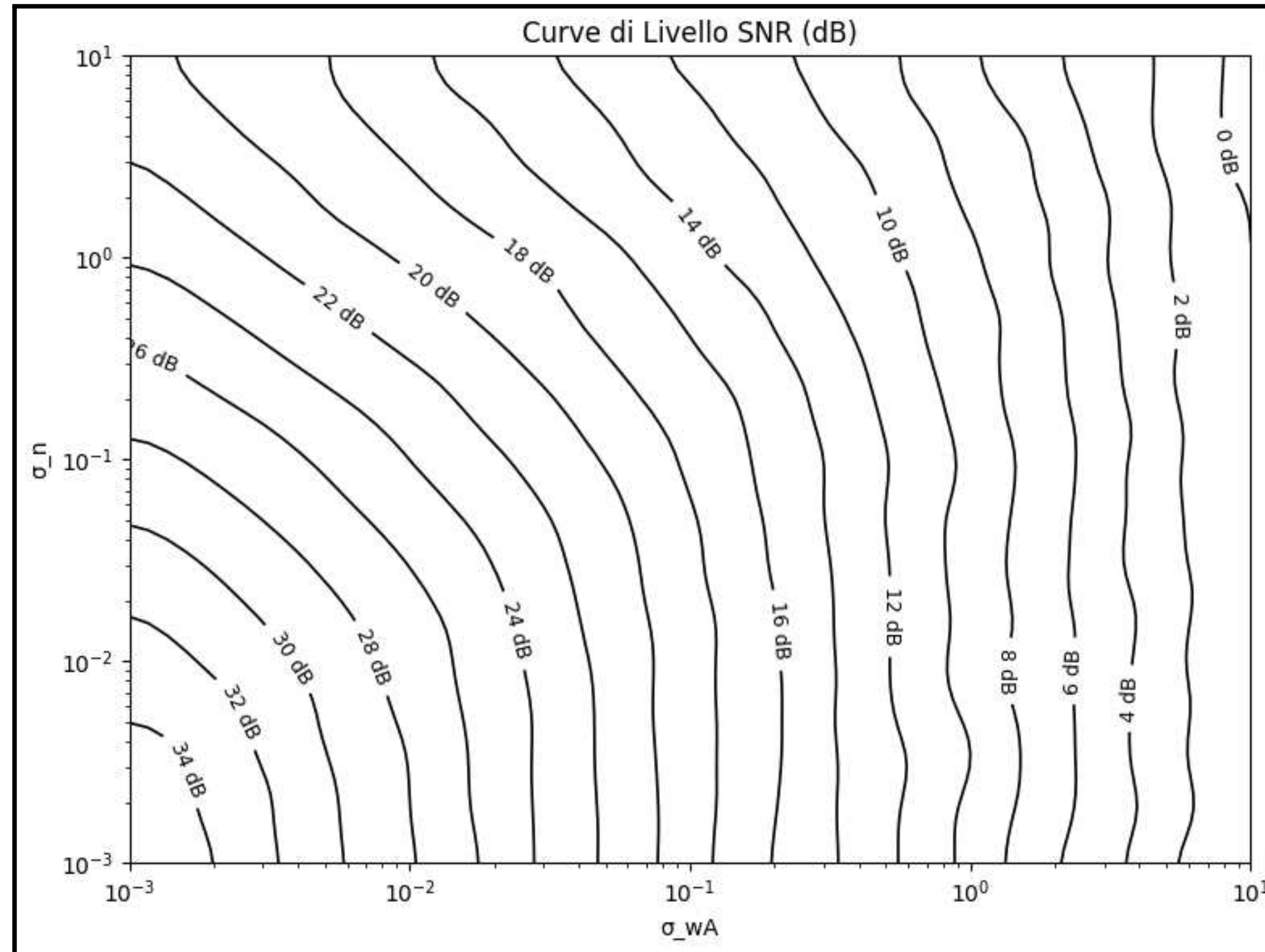


Fig. 4. Curve di Livello SNR (in dB) ottenute tramite simulazione Monte Carlo

RISULTATI: INFORMAZIONE MUTUA



- Ogni singolo addestramento usa uno specifico dataset che simula le osservazioni fatte a un certo livello di SNR.
- Una volta allenate, le reti vengono valutate su un insieme di dati di test generati a diversi livelli di SNR.
- Si analizza l'andamento dell'**Informazione Mutua** in funzione del SNR dei dati.
- L'Informazione Mutua ha un andamento coerente, crescendo al crescere del SNR.
- Confrontando addestramenti condotti a livelli diversi di SNR, si osservano sempre gli **stessi andamenti** dell'Informazione Mutua.
- Il modello di training è **robusto** e non overfitta i dati di training.

RISULTATI: ANDAMENTO INFORMAZIONE MUTUA

Vengono eseguiti tre allenamenti diversi:

- Allenamento 1:

SNR = 30 dB

- Allenamento 2:

SNR = 15 dB

- Allenamento 3:

SNR = 5 dB

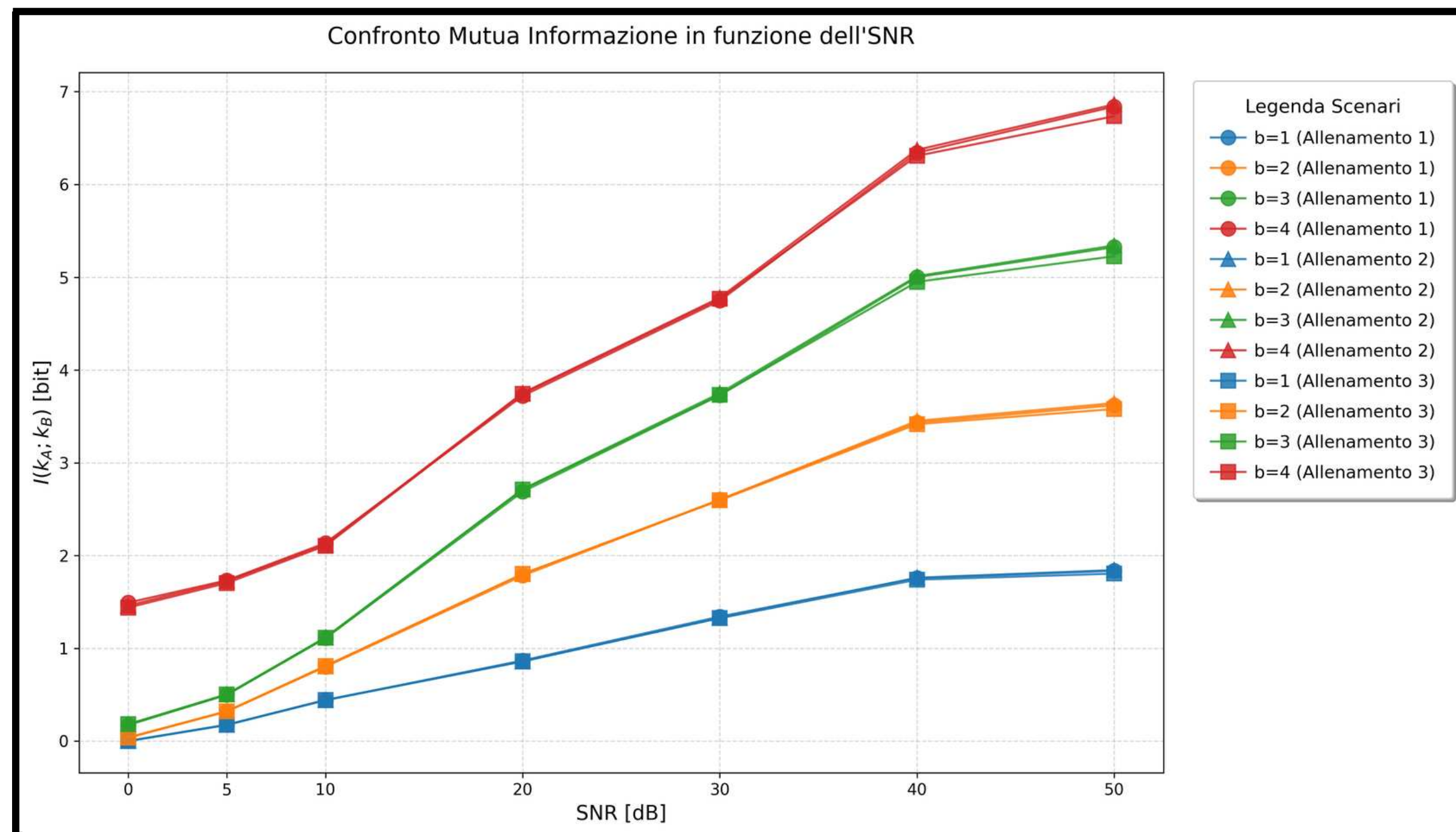


Fig. 4. Curve di Livello SNR (in dB) ottenute tramite simulazione Monte Carlo

CONCLUSIONI

- Il lavoro svolto ha permesso di costruire un modello completo per descrivere, simulare e analizzare un sistema di Key Agreement basato su osservazioni correlate del canale.
- Partendo dalla definizione del modello di rumore e impairments, passando per la modellazione della struttura di Machine Learning, fino alla formulazione del modello SNR.
- E' stato possibile caratterizzare in modo chiaro la qualità del canale e il comportamento delle rappresentazioni generate dalla rete.
- Il modello di training è risultato stabile e robusto.
- Le chiavi generate approssimano una distribuzione uniforme. (*Randomicità*)
- L'andamento dell'Informazione Mutua è coerente con la teoria e rispecchia le proprietà del canale. (*Reciprocità*)