Séminaire STORE - 13 avril 2021.

INCdeep: Intelligent Network Coding with Deep Reinforcement Learning

Qi Wang [*], Jianmin Liu [*], Katia Jaffres-Runser[+], Yongqing Wang [*], Chentao He [*], Cunzhuang Liu [*], Yongjun Xu [*]

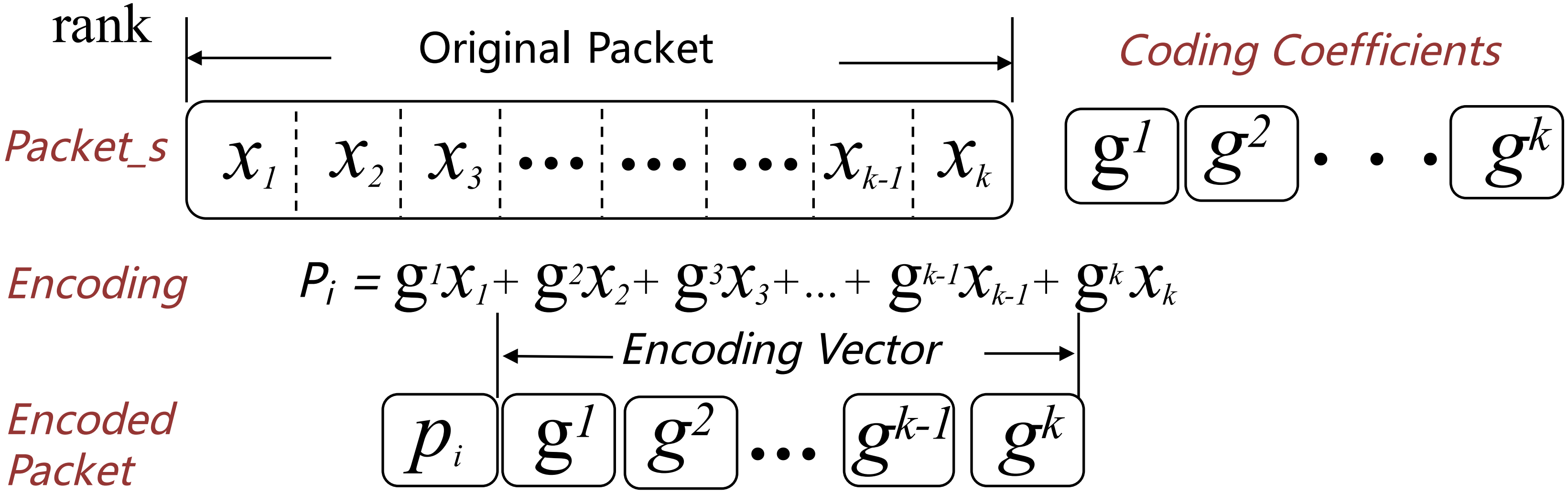
[*] Institute of Computing Technology, Chinese Academy of Sciences, China
[+] University of Toulouse, IRIT/ENSEEIHT, France

Institute of Computing Technology, Chinese Academy of Sciences

CNRS - INPT - UPS - UT1 - UTM

1 Network coding

- Network coding has been introduced to achieve higher transmission rates and improve transmission reliability
  - **Source, relay node**: selects a vector of coefficients and combines received packets into one or several output packets
  - **Destination (decoder)** : decodes original packets from a set of received ones by Gaussian elimination if coefficient matrix is full rank



Related works

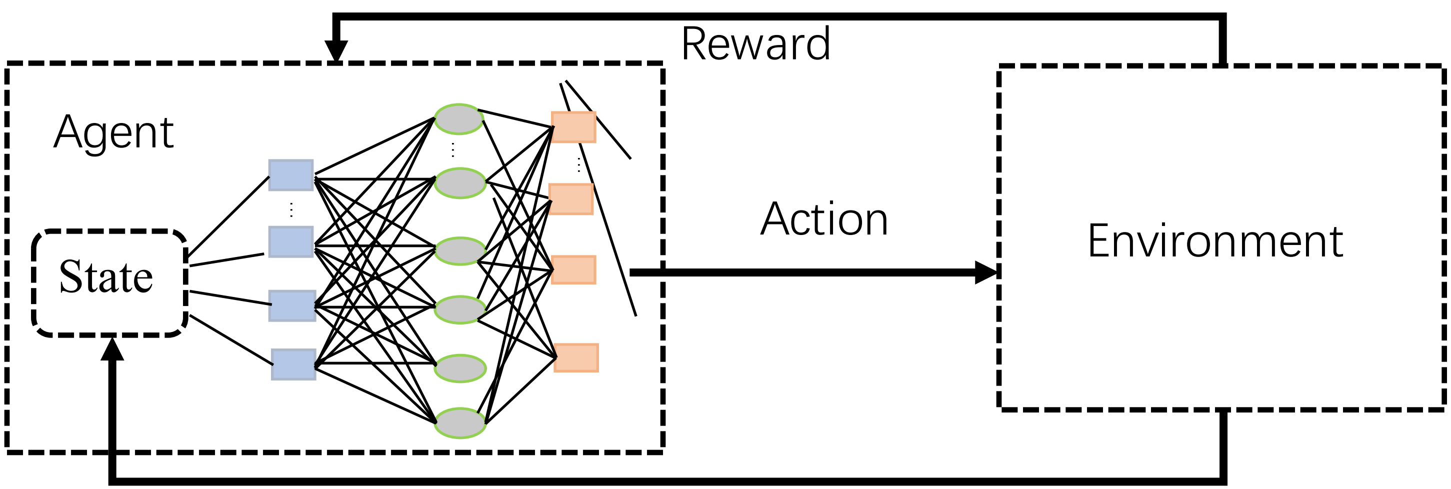
- Existing linear network coding
  - Deterministic network coding(cf. [1][2])
    - **Coding coefficients** determined by dedicated deterministic algorithms
    - Requiring **global network information, incapable of adapting to varying network conditions, high complexity, low robustness.**
  - Random linear network coding(cf. [3][4])
    - **Coding coefficients** randomly selected in a finite field in random linear network coding
    - Easily implemented in a distributed fashion
    - Coefficients **can not be adjusted to network dynamics (time-varying link quality, changing number of relays)**

Problem: How to automatically build coding coefficients to adapt to network dynamics?

[1] S.-Y. Li, R. W. Yeung, and N. Cai, "Linear network coding," IEEE transactions on information theory, vol. 49, no. 2, pp. 371–381, 2003.
[2] R. Koetter el., "An algebraic approach to network coding," IEEE/ACM transactions on networking, vol. 11, no. 5, pp. 782–795,2003.
[3] T. Ho, el., "The benefits of coding over routing in a randomized setting," in ISIT, July 2003.
[4] T. Ho, el., "A random linear network coding approach to multicast," IEEE Transactions on Information Theory, vol. 52, no. 10, pp. 4413–4430, Oct 2006.

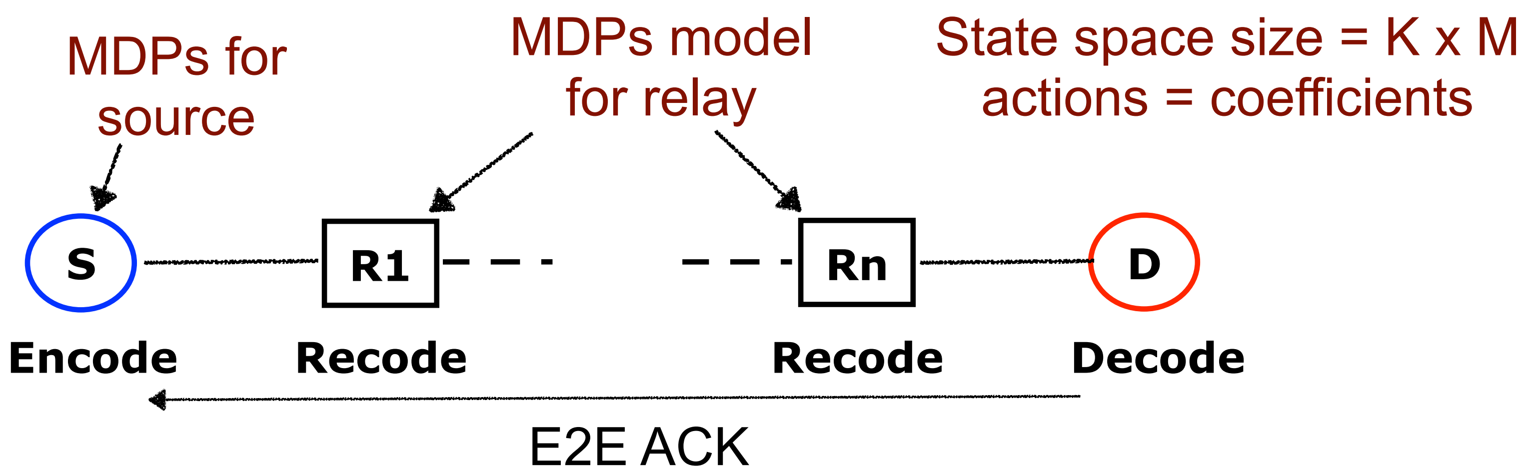
Opportunity of Deep Reinforcement Learning

- Deep reinforcement learning combines deep learning with reinforcement learning (RL) to implement machine learning from perception to action.
- Network variations can be automatically and continually expressed as Markov decision process (MDP) state transitions.
- RL agent learns to optimize coding coefficients to maximize a decoding probability objective, through the rewards obtained by interacting directly with the network.



Motivation: How to devise a DQN to automatically select coding coefficients to adapt to network dynamics?

Network code model



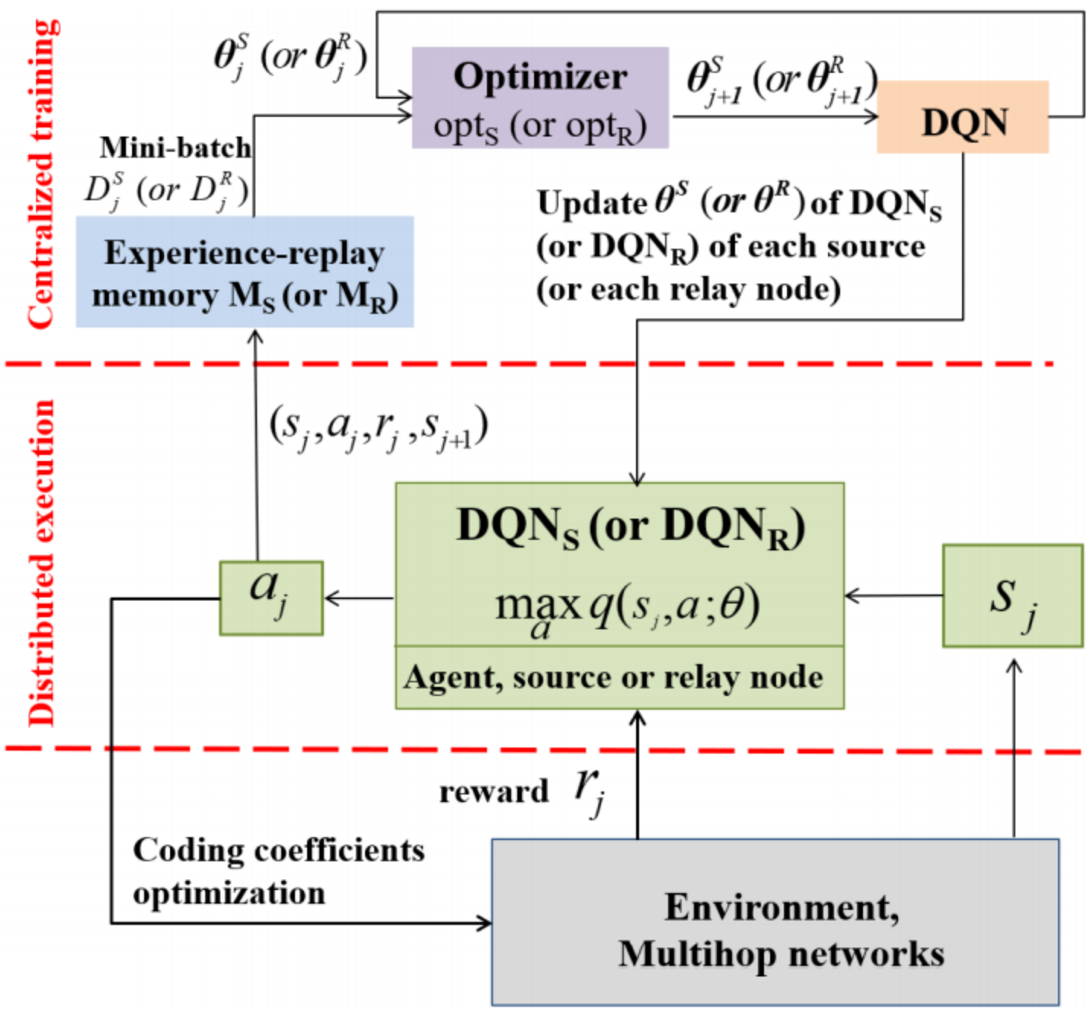
- Source S encodes linearly K equally sized packets at each episode until it receives an end-to-end ACK from D.

- Relays recode linearly the M previously received coded packets until they receive the ACK from D.

- D decodes if coding matrix is full rank.

Architecture of INCdeep

Deep Q Network (DQN)

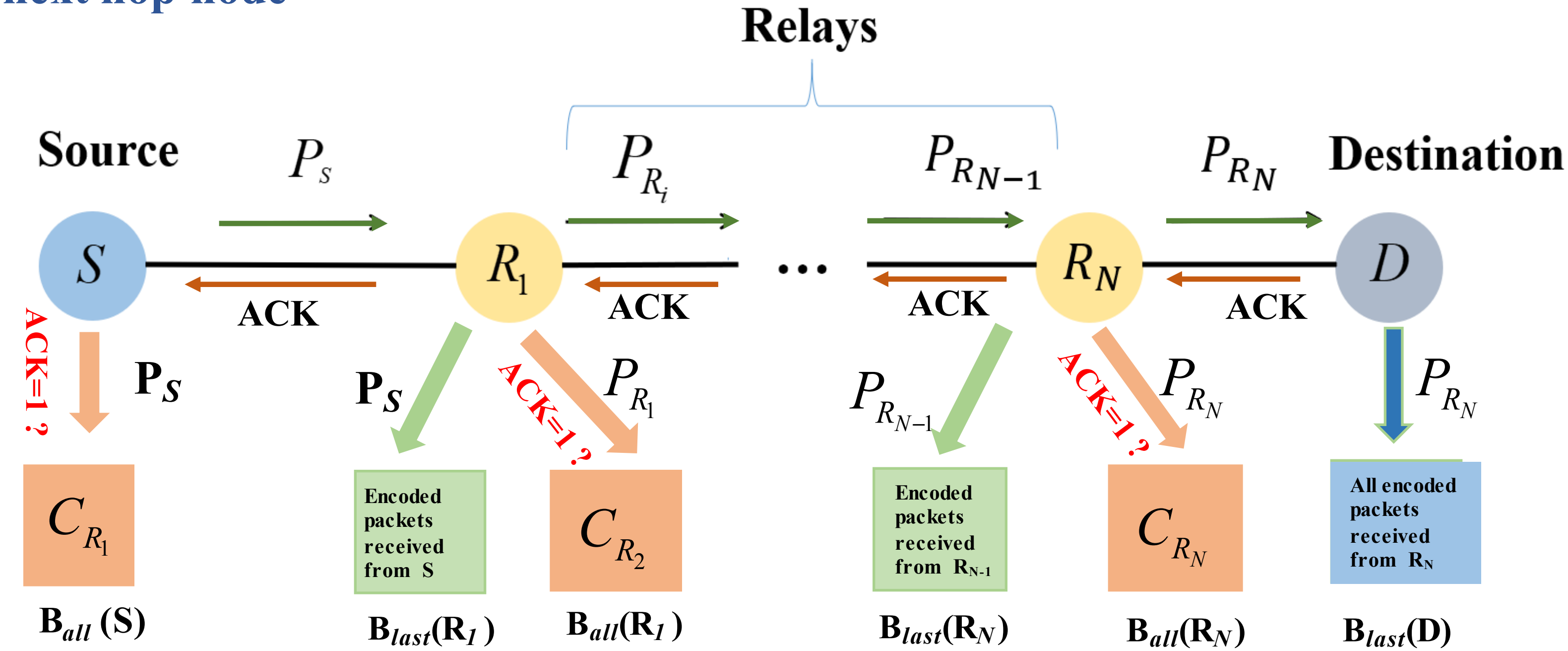


Training of DQN is centralized

Coding coefficients optimization of packets is distributively executed at source and at each relay node

How to keep track of the decoding matrix at the next relay?

- Buffer B_{last} (Encoded packets received from its previous node) of size M
  - Each relay R_i has a buffer $B_{last}(R_i)$ to store the latest M encoded packets received from its previous hop.
- Buffer B_{all} (Encoded packets received by next node) of size M
  - Source S and each relay R_i have a buffer B_{all}(S) (or $B_{all}(R_i)$) to keeping track of encoded packets received by its next hop node ($C_{R_{i+1}}$, $i = 1 \ldots N$) using ACK messages.
- ACK is used to confirm the transmitted encoded packet received successfully by the next hop node



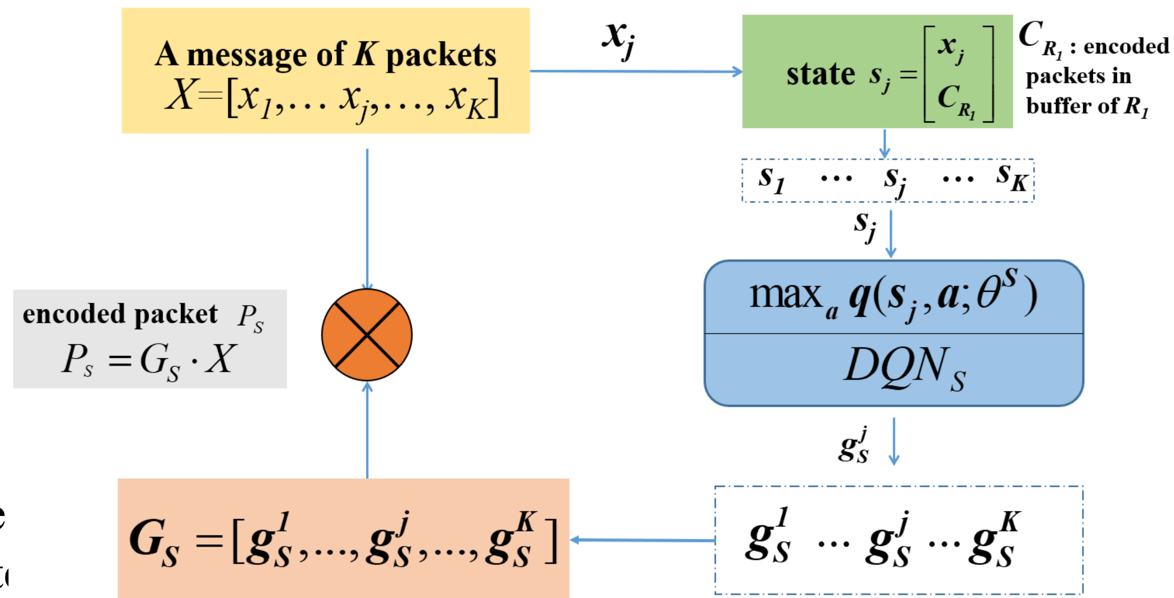
DQN model for source (DQN_S)

State input:

- The information of the j-th packet x_j of a message
- M encoded packets in the buffer of the first relay R_1

Action output:

- Coding coefficient of j-th packet of the message
- Run K times DQN_S to derive coefficient vector G_S to creat a single encoded packet P_s



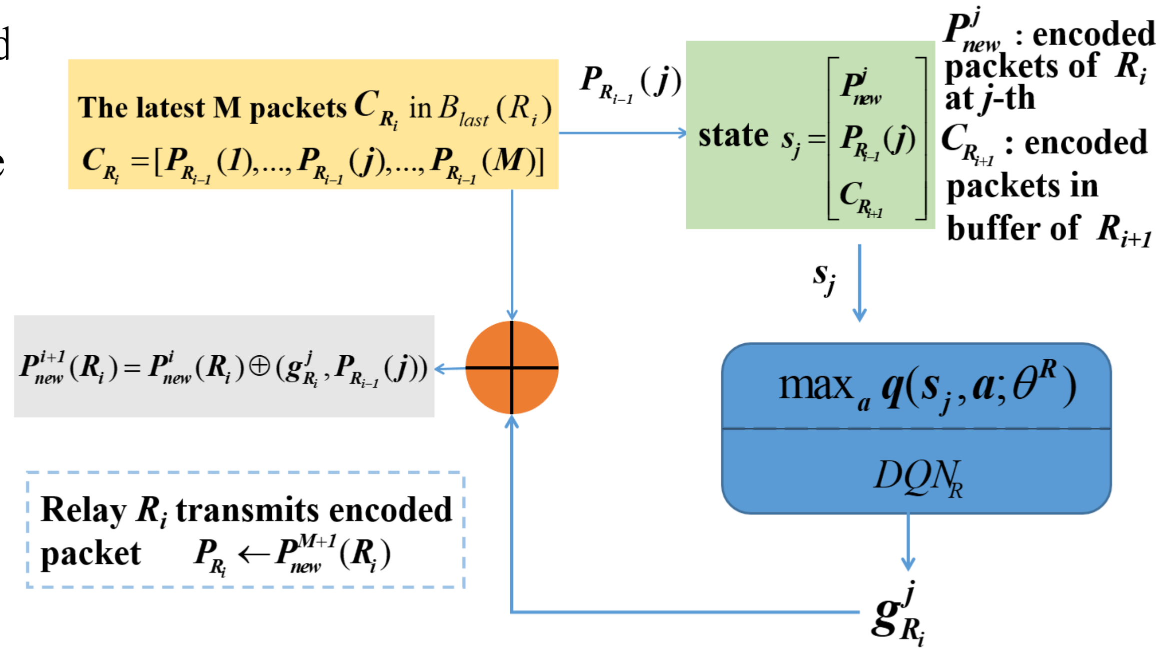
Reward:

- The rank of linear system of equations formed by these packets in the buffer of next node $\mathbf{B}_{all}$ increases, reward is +1
- Otherwise, reward is -1

DQN model for relays (DQN_R)

State input:

- New encoded packet produced by current node at step j
- The j-th encoded packet in the buffer of current node
- M encoded packets in the buffer of the next node



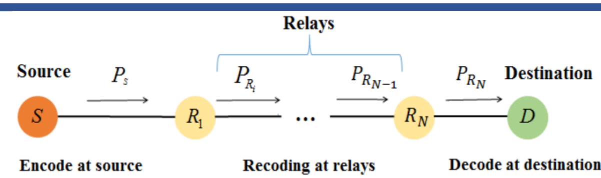
Action output:

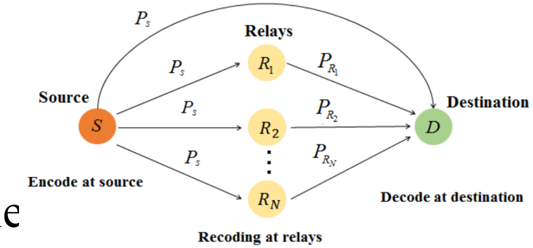
- Coding coefficient of j-th encoded packet in its buffer
- Run M times DQN_R to derive coefficient to create a single encoded packet P_{Ri}

Reward:

- The rank of linear system of equations formed by these packets in the buffer of next node increases, reward is +1
- Otherwise, reward is -1

Simulations





- ## Simulation Setup
  - **Network topologies**: Linear topologies and Parallel topologie
  - **Simulation platform**: Python 3.5-based framework and TensorFlow 1.15; Ubuntu 16.04 server with 2 RTX2080 Ti GPUs and Intel 2 Xeon E5-2660.
  - **Results assessment**:
    - **Root Mean Squared Error (RMSE)** $RMSE = \frac{1}{n}\sqrt{\sum_{i=1}^{n}\frac{(d(i)-\tilde{d}(i))^2}{d(i)^2}}$;
    - **Overhead** $overhead = \frac{1}{k}(E) * 100 = \frac{1}{k}(Nr - K) * 100$

- ## Benchmark coding algorithms
  - **Traditional random network coding**: Source nodes uses a fountain code and relays combine coded packets using RLNC [1]
  - **RL-aided SNC[2]**: Reinforcement learning-aided sparse network coding.

[1] T. Ho,el., "A random linear network coding approach to multicast," IEEE Transactions on Information Theory, vol. 52, no. 10, pp. 4413–4430, Oct 2006.
[2] R. Gao, Y. Li, J. Wang, and T. Q. S. Quek, "Dynamic sparse coded multihop transmissions using reinforcement learning," IEEE Communications Letters, vol. Early Access, no. 1-5, June 2020.

Simulations

- **Performance Evaluation Results**
  - **Training efficiency**: reward converges in about 10000 episodes for field size q = 8 and PER = 0.1, and 7000 episodes for finite field size q = 8 and PER = 0 as well as finite field size q = 2.
  - **Runtime cost**: for both topologies, the runtime is 0.8 ms and 1.6 ms to obtain coding coefficients for an encoded packet under K = 5 and K = 10, respectively
  - **Generalization ability**: verify generalization on link quality and number of relays

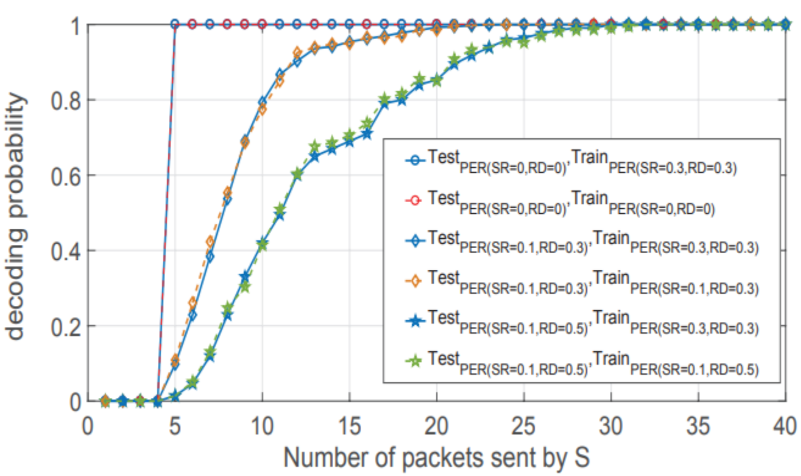


Fig.1 Generalization ability on PER

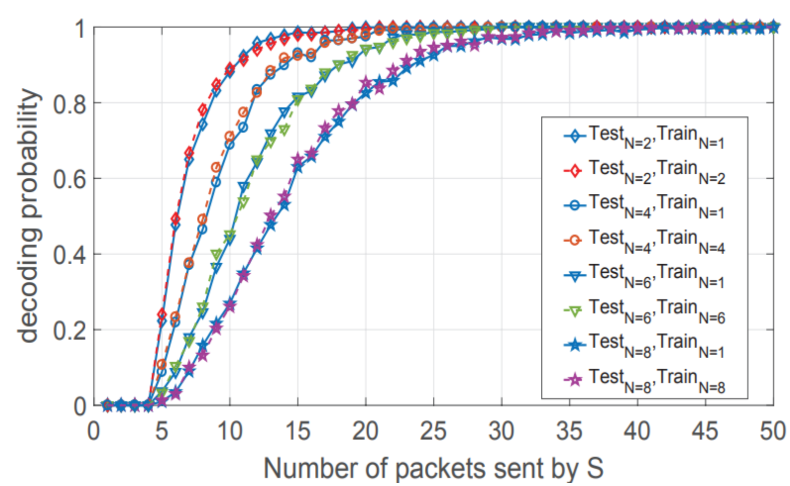


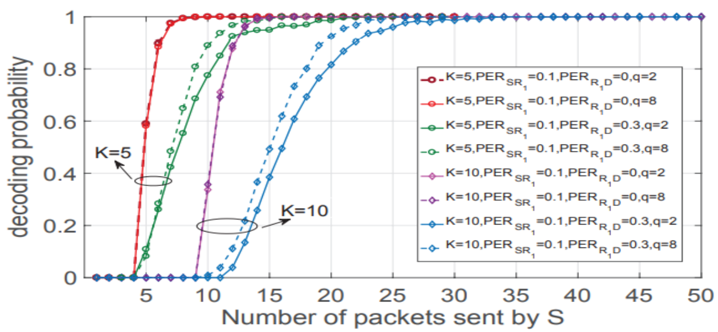
Fig.2 Generalization ability on relay number N

INCdeep model obtained with a training setting that consider a bad channel setting can be leveraged at runtime.

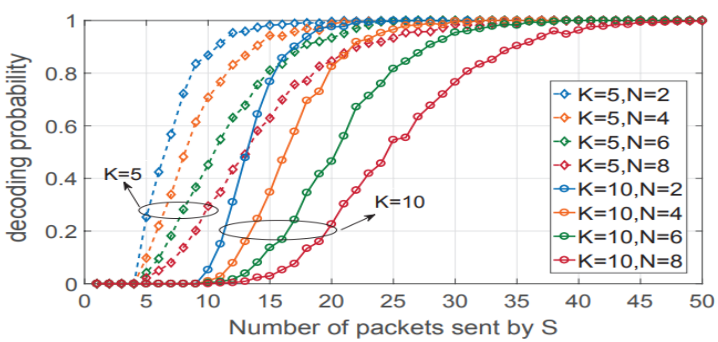
Simulations

- **Performance Evaluation Results**
  - **Performance of INCdeep with *different PER* and *different number of relays N***

Linear network

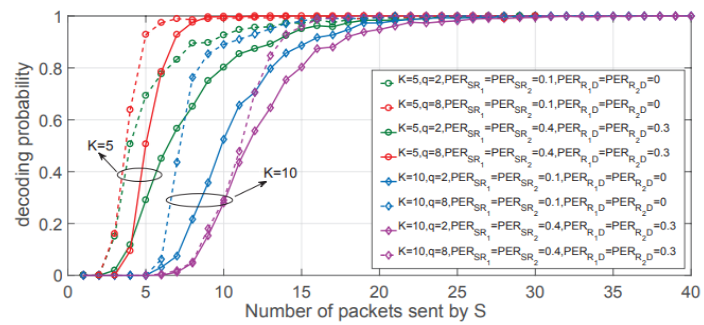


(a) Single-relay linear network, $K\in\{5,10\}$, $q\in\{2,8\}$, PER_{SR_1}=0.1 and $PER_{R_1D}\in\{0,0.3\}$.

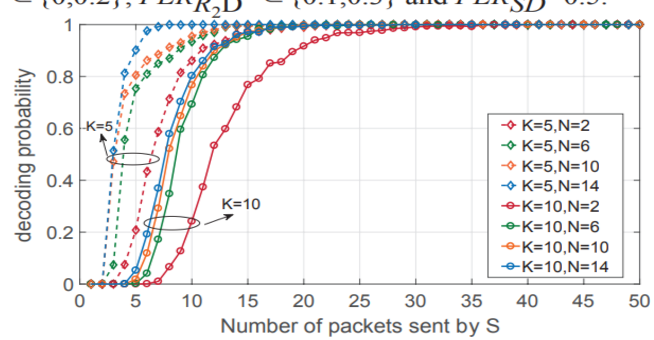


(b) Multiple-relays linear network, $K\in\{5,10\}$, various $N\in\{2,4,6,8\}$, q=2 and $PER_{SR_j=R_jD}$=0.1 (j=1,…,N).

Parallel network



(a) Two-relay parallel network, $K\in\{5,10\}$, $q\in\{2,8\}$, $PER_{SR_1}\in\{0,0.1\}$, $PER_{SR_2}\in\{0.1,0.2\}$, $PER_{R_1D}\in\{0,0.2\}$, $PER_{R_2D}\in\{0.1,0.3\}$ and PER_{SD}=0.5.

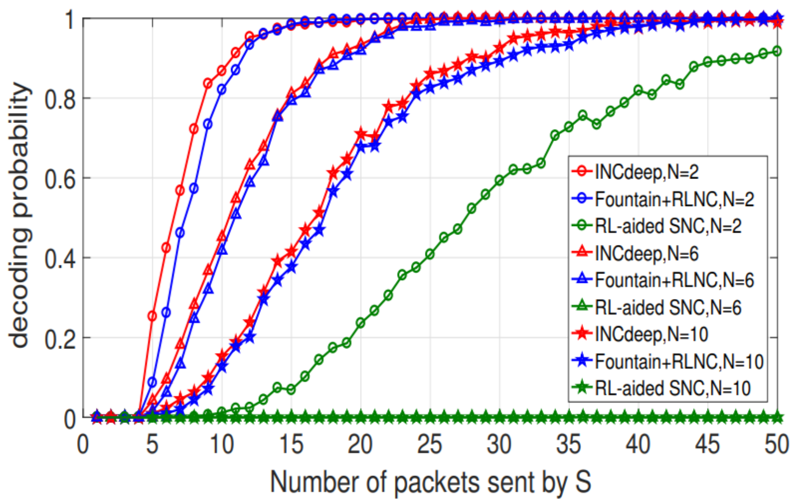


(b) Multiple-relays parallel network, $K\in\{5,10\}$, various $N\in\{2,6,10,14\}$, q=2, PER_{SR_j}=0.1, PER_{R_jD}=0.3 (j=1,…,N), PER_{SD}=0.8.

Simulations

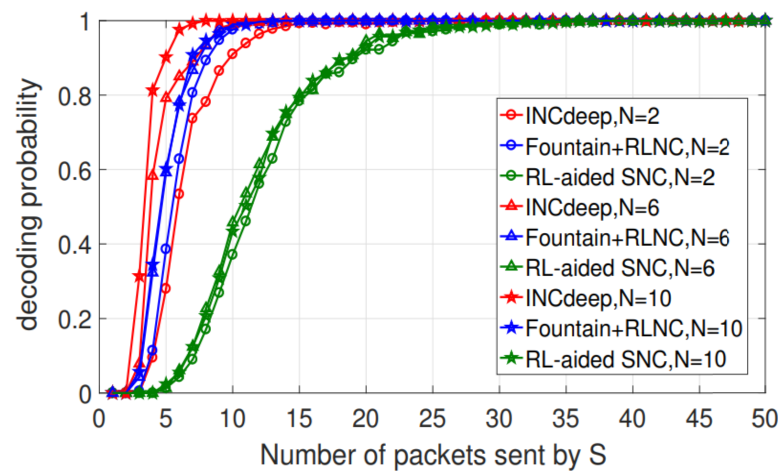
- **Performance Evaluation Results**
  - **Comparison between INCdeep and benchmark coding algorithms**

Linear network



PER=0.1 on each link

Parallel network



PER_sr = 0.1 ; PER_rd = 0.3 ; PER_sd = 0.8

TABLE I: Comparison of the overhead

Topology	N	Coding Method		
		INCdeep	Fountain+RLNC	RL-aided SNC
Linear	2	14.26 %	24.86 %	30.1%
	6	19.6 %	25.1 %	596 %
	10	22.2 %	26.3 %	9855 %
Parallel	2	222.68 %	89.5 %	292.64 %
	6	290.48 %	345.66 %	936.6 %
	10	454.26 %	598.8 %	1605.32 %

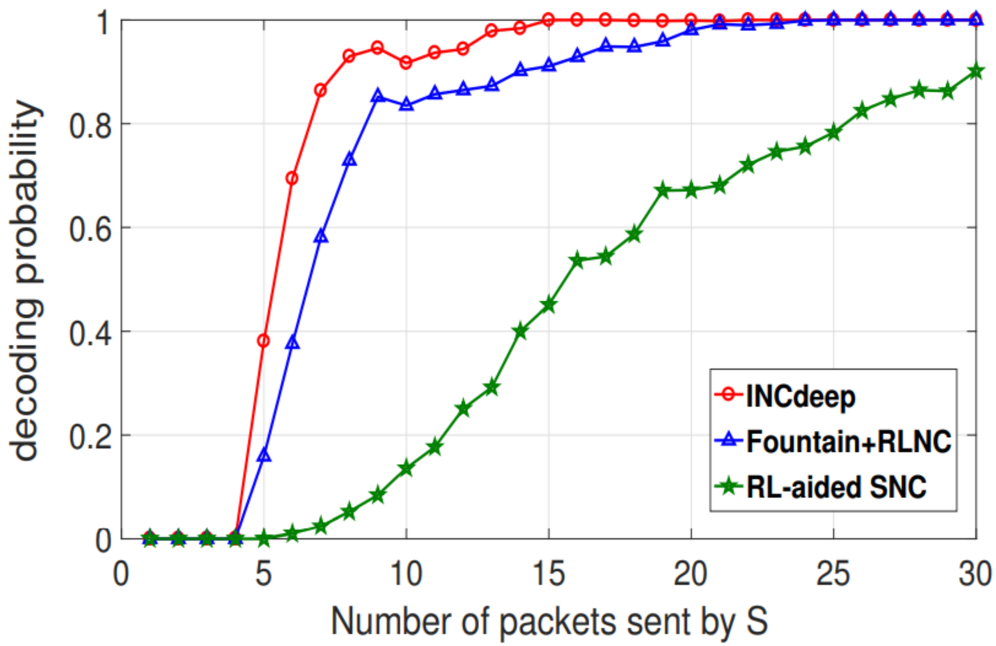
✓ **In linear topology, INCdeep has smallest overhead.**

✓ **In parallel topology, INCdeep outperforms benchmark algorithms as the number of relays increases.**

Simulations

- **Performance Evaluation Results**
  - **Fast Changing Environment:** evaluate INCdeep and benchmark coding algorithms in dynamic environments where PER vary much faster.

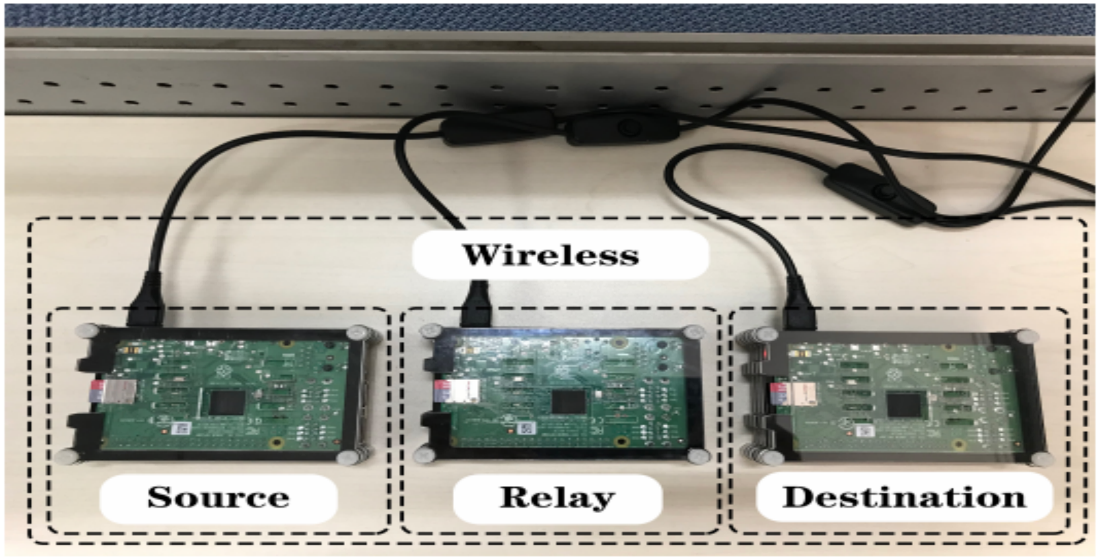
Here, PER_{SR} and PER_{RD} are independently randomly drawn from a uniform distributions over [0.0,0,4] for every five packets emitted by source



Decoding probability of INCdeep outperforms benchmark coding algorithms. INCdeep can adapt well to a dynamic environment.

Testbed experiments

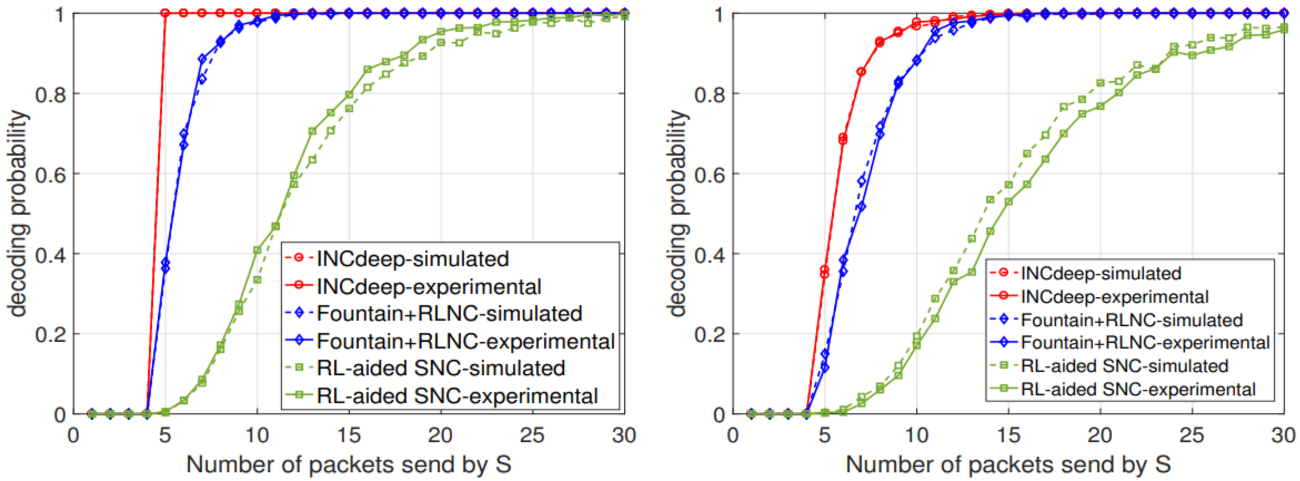
- **Experimental Setup**
  - Use Raspberry Pi 3 Model B+ [1] to conduct experiments
  - Exploit TensorFlow Lite [2] to deploy the trained INCdeep to the Raspberry Pi 3B+



Testbed for Network coding on Raspberry Pi 3B+

Testbed experiments

- **Experimental Results**



(a) PER=0 (b) PER=0,1

Experimental and simulated decoding probability for single-relay network, $PER_{SR=RD} \in \{0, 0.1\}$, q=2 and K=5.

Experimental results coincide with simulation results.

RMSE for experimental and simulated results

	INCdeep	Fountain+RNLC	RL-aided SNC
PER=0	0	0.0109	0.0283
PER=0.1	0.0042	0.0153	0.0379
PER=0.3	0.0117	0.0237	0.0154

Smallest RMSE

Performance on CPU usage and decoding probability

	CPU usage	Decoding Prob. (K=5)	
		PER=0	PER=0.1
INCdeep	15.9% (+28.2%)	1.00 (0%)	0.86 (0%)
Fountain+RNLC	12.4% (0%)	0.37 (-63%)	0.70 (-18.6%)
RL-aided SNC	18.6% (+50%)	0.05 (-95%)	0.16 (-81.4%)

Higher CPU usage than traditional coding, but decoding prob. =1 when PER =0

Conclusions

- **Propose INCdeep, an adaptive, DRL approach to automatically build adaptive coding coefficients**
  - **Good generalization ability** that adapts well in dynamic scenarios.
  - **Higher decoding probability and lower coding overhead** compared with benchmark algorithms.
  - **Converges fast** and runtime costs within 0.8 ms and 1.6 ms to obtain coding coefficients for K = 5 and K = 10, respectively.
  - Experimental results coincide with the simulation results, showing that INCdeep is **feasible to offer a practical deployment.**

- **Future Work**
  - Whether coding is beneficial or not for an overall multi-objective performance goal

Thanks!

Q&A: wangqi08@ict.ac.cn

Institute of Computing Technology, Chinese Academy of Sciences

CNRS - INPT - UPS - UT1 - UTM