T3.1: Automating IoT device configuration

Renzo E. NAVAS

WP3: Virtualization and automation of IoT access networks.

Outline

01. Introduction
02. Roadmap
03. Bandits
04. Evaluation
05. Takeaway (+Discussion)

Task 3.1: Introduction

Description:

- Having a large number of IoT nodes to learn what parameters to use (time, power, spreading factor, etc.) for uploading data, and optimizing their global performance.
- Focus on mechanisms where IoT nodes make their own decisions (decentralized)
- But, end devices may have their strategy optimized globally by the orchestrator.

Goal:

- **Propose and analyze the performance of machine learning algorithms that need few resources, like multi-armed bandit methods.**
- **D3.1: Lightweight learning algorithms for massive IoT and analysis of their performance. (T15)**

Task 3.1: Roadmap

Baseline: Do “better” than LoRaWAN’s ADR –and SoA–, in massive IoT scenarios.

1. Related Work (Positioning) [Non-priority for D3.1]:
   - Survey of proposals to improve LoRaWAN Medium Access, like [1] (~22 art.). We identified ~50 articles.
2. **Proposal**: Reinforcement-Learning-based, particularly **Bandit Algorithms [2]**
3. **Evaluation**: We will use a realistic setting/evaluation scenario, NS-3 based [URL].
4. **Contribution:** A differentiating factor of our Bandit-based Algorithm(s) will be this applicability/evaluation in realistic LoRaWAN scenarios.
   - In the literature, proposals use strong hypotheses or simplified models:
     - E.g., Only one node uses RL, immediate feedback, independence of configuration parameters.
     - I.e., Proposals tend to the theoretical-side.
   - However, evaluation/comparison against non-bandit proposals will be a challenge.

Bandit Algorithms – Introduction I

Round	1	2	3	4	5	6	7	8	9	10	11	
SF9	0		1		0	0				1	?	PDR = 2/5
SF7		1		0			0	0	0		?	PDR = 1/5

Arms (k=2)

Exploration-phase

Exploitation?

- Strategy: shall the IoT node keep using SF9, ignoring SF7?
- Or we attribute the poorer PDR performance of SF7 to “bad luck” and try it a few more times? How many more times? (Exploration-Exploitation trade-off)
- … What if we redefine the (negative) **reward** as the energy spent to deliver a packet?

Bandit Algorithms – Introduction II

- Goal: Maximize the cumulative reward, for a horizon of N “rounds” (could be infinite).
  - The definition of “reward” is fundamental (E.g., plain PDR vs Energy-aware PDR).
- In the basic bandit setting:
  - **Context-agnostic**: Arms’ rewards are independent (a reward for one arm does not give information about other arms’. E.g., a packet delivered with SF7 does not mean it also would have been delivered with SF9).
  - **Context-agnostic**: The learning agent **only** interacts with the system by “pulling” an arm and observing the empirical reward.
  - **Perfect Monitoring**: Empirical-reward observation (feedback) is immediate (… or almost).
  - **Stationary:** The underlying environment does not change (E.g., An arm’s reward’s “behavior” is always the same)

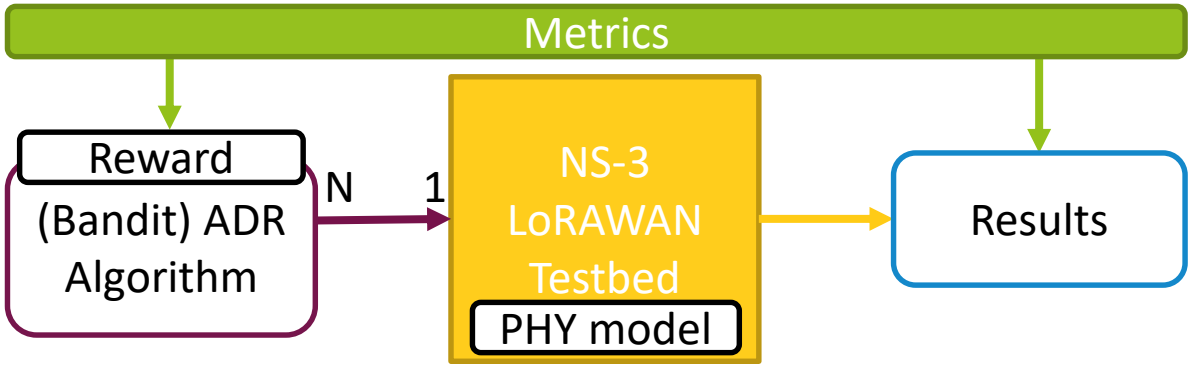
Bandit Algorithms – A realistic setting

- In our realistic setting:
  - **Contextual Bandits** → We can use contextual information (E.g., use DL for SNR stats, SFs are not-independent)
  - **Partial Monitoring** → We do not have perfect monitoring (E.g., DownLink is expensive)
  - **Non-stationary** → The environment, most likely, is not stationary.

- Defining an analytical model for this setting is not trivial, and theoretical bandit solutions does not exist (to the best of my knowledge).
- Our proposal(s) will use bandit solutions at their core, but probably will be mixed with **some heuristics**.
- Thanks to a realistic ns-3 evaluation/simulation environment, we will have strong **statistical guarantees** about their performance (and will compare with some SoA, including vanilla ADR). TBD: Obtain some theoretical guarantees.

LoRaWAN Evaluation: NS-3

- NS-3 LoRaWAN module
  - GIT : https://github.com/signetlabdei/lorawan and Documentation.
  - Current version has an implementation of LoRaWAN's ADR.
  - Does not implement: Class-B Nodes, Frame Counters (but Lost packets are traced in-software).
  - Energy-measurement module could be implemented w/reasonable effort.
  - NS-3 module is highly customizable: E.g., stack of path loss models (Shadowing, Buildings..).



Takeaway slide

- Proposal(s): Bandit-based ADR, will realistic hypothesis.
- In this setting, will be hard to prove analytical/theo. properties.
- But, we will provide strong statistical guarantees:
  - The NS-3 LoRaWAN testbed will be a primordial component (a contribution).
  - Comparisons: baseline will be vanilla ADR, and compare against State of the Art.
  - Bandits-implementations (C++) will be relatively easily ported to real IoT nodes.

Discussion

- Acklio, AGUILA involvement?
  - Acklio:
  - AGUILA:
- Shared NS-3 LoRaWAN Framework/Base Code? (Alessandro)
  - To have a common PHY config (?). E.g.: I plan on using a more realistic inter-SF interference Matrix than the default one.
- Any Other?

Merci !

renzo.navas@inria.fr

Bibliography

[1] Kufakunesu, Rachel, Gerhard P. Hancke, and Adnan M. Abu-Mahfouz. "A survey on Adaptive Data Rate optimization in LoRaWAN: Recent solutions and major challenges." Sensors 20.18 (2020): 5044.

[2] Lattimore, Tor, and Csaba Szepesvári. Bandit algorithms. Cambridge University Press, 2020. URL: https://tor-lattimore.com/downloads/book/book.pdf