Autonomous Networking

Gaia Maselli
Dept. of Computer Science

Today’s plan

- Course logistics
- Introduction to main course topics

Course logistics

- **Class schedule**
  - Wednesday 10 am - 12 am
  - Friday 8 am – 11 am -> starting 8.30
- **Textbook and reading**
  - Scientific papers available on digital libraries
    - ACM (acm digital library)
    - IEEE (ieeexplore)
  - Reinforcement Learning: An Introduction, Richard S. Sutton and Andrew G. Barto, 2nd Edition MIT Press, Cambridge, MA, 2018
- **Website:** https://twiki.di.uniroma1.it/twiki/view/AN/WebHome

Course logistics

- **Google Classroom site**
  - Class info (lectures, exams)
  - Material (slides, scientific papers)
  - News (including canceled lectures)

  will be available on classroom

  - You need an institutional account
  - https://classroom.google.com/c/NzE4MzcyMzU0NzA2?cjc=hywi5a5
  - hywi5a5

Evaluation

- **All the topics covered in the course are subject to examination**
- To pass the exam students can choose <u>ONE</u> of the following:
  1. **Written exam during the exam session (at the end of the course)**
  2. **Mid-term exam + oral presentation of a research paper or a project or a research work**

Evaluation: midterm + integration

- Only the students taking a grade higher than 26 can do the integration (may vary depending on the number of students)
- The presentation of the assigned scientific article will take place in class during the class hours (according to a schedule established after the mid term exam).
- Oral presentation can take place only during the course (in case you miss it you need to do the final exam)
- Oral presentation requires students to be critical (comment the presented paper)
- By the end of October, students interested in the second examination mode must inform the instructor (a google form will be provided)

Prerequisites

- Computer networks course
  - TCP/IP network stack
  - MAC and routing protocols

- Notions of probability

Course outline (1/3)

- Wireless networked communication systems:
  - sensor networks
  - IoT
  - Backscattering (RFID) networks
  - Unmnanned networks (dronets)

  - But also some notions of network performance evaluation

Course outline (2/3)

Reinforcement learning

- Reinforcement learning problems
- Reinforcement learning techniques
  - Bandit algorithm
  - Q learning

Course outline (3/3)

This part will be mainly presented by students

- Reinforcement learning assisted networking (for the aforementioned technologies)
  - Reinforcement learning assisted channel access and routing
- Autonomous networks not based on reinforcement learning (e.g., protocols for unmanned networks)

Why autonomous?

No human intervention (**Self-Governance**) - makes independent decisions based on predefined rules and real-time data.

Existing and functioning as an independent organism (**Self-Sufficiency**) capable of managing its own resources and processes.

Autonomy involves intelligence (**Embedded Intelligence):** Utilize advanced algorithms and AI to interpret environmental inputs, learn from experiences, and adapt to changing conditions

Able to learn from the environment and react accordingly (**Proactive Response)**: Continuously monitor the environment, anticipate changes, and react autonomously to maintain optimal performance and reliability.

Example

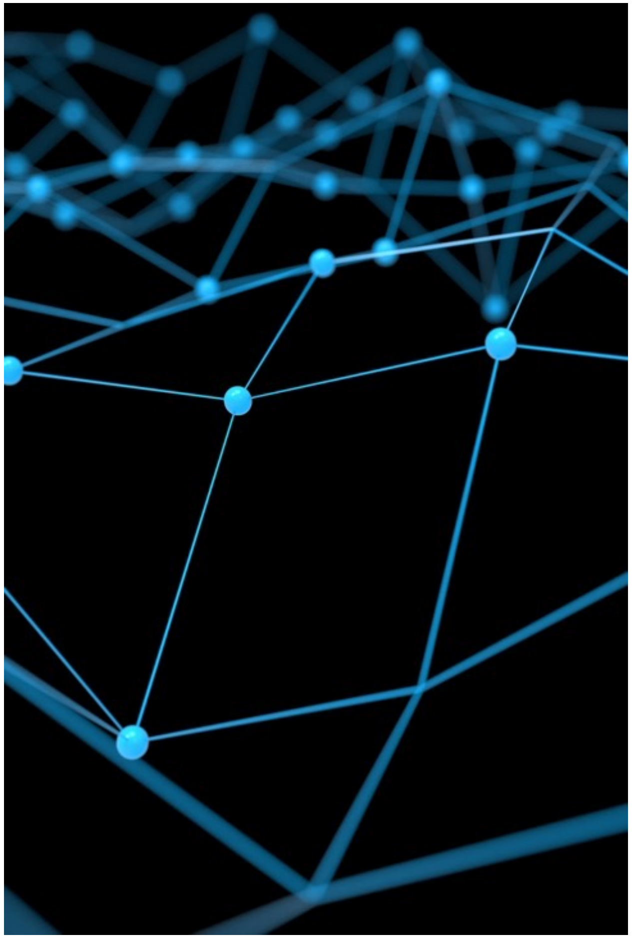
- Suppose you have a smart home with many devices deployed in the environment
- Communication channel is shared by all devices
- Video streaming is taking most of bandwidth
- Suddenly there is smoke in the kitchen
- All devices monitoring the kitchen should be able to send their data without any delay (able to gain channel access in short time)

Autonomous networking

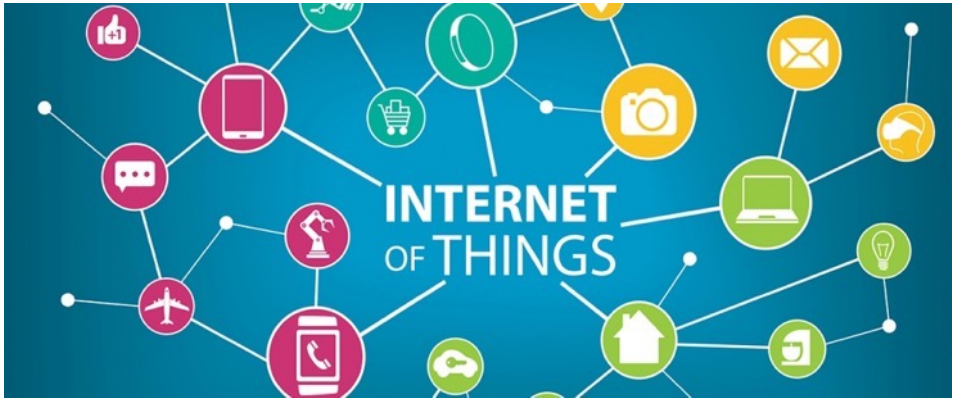
Which tecnologies?

Mainly wireless

- Backscattering (RFID)
- Sensor & IoT
- Drone networks (dronet)



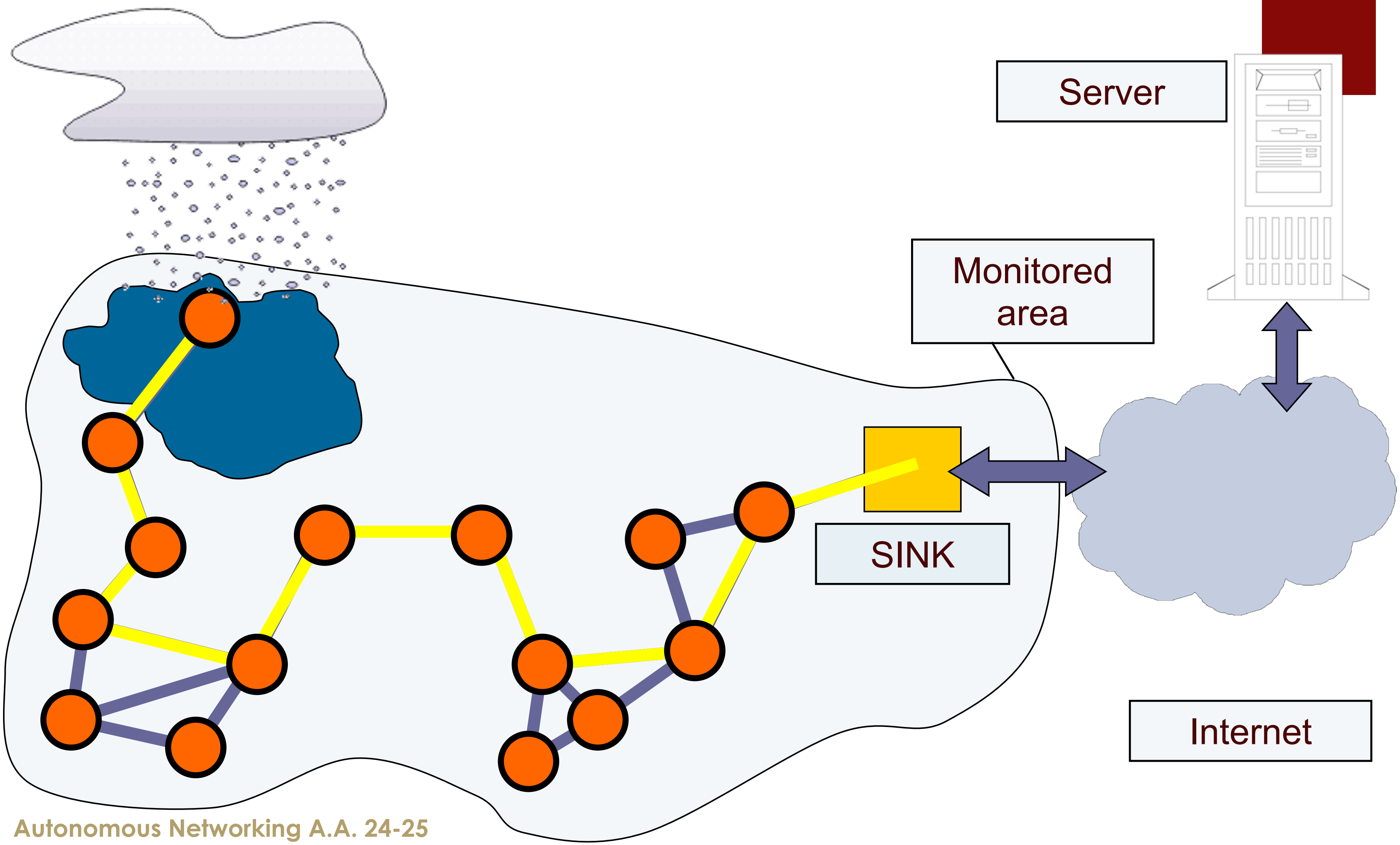
RFID, sensor networks, and IoT



RFID

- Radio Frequency Identification

Sensor networks (sense the environment)



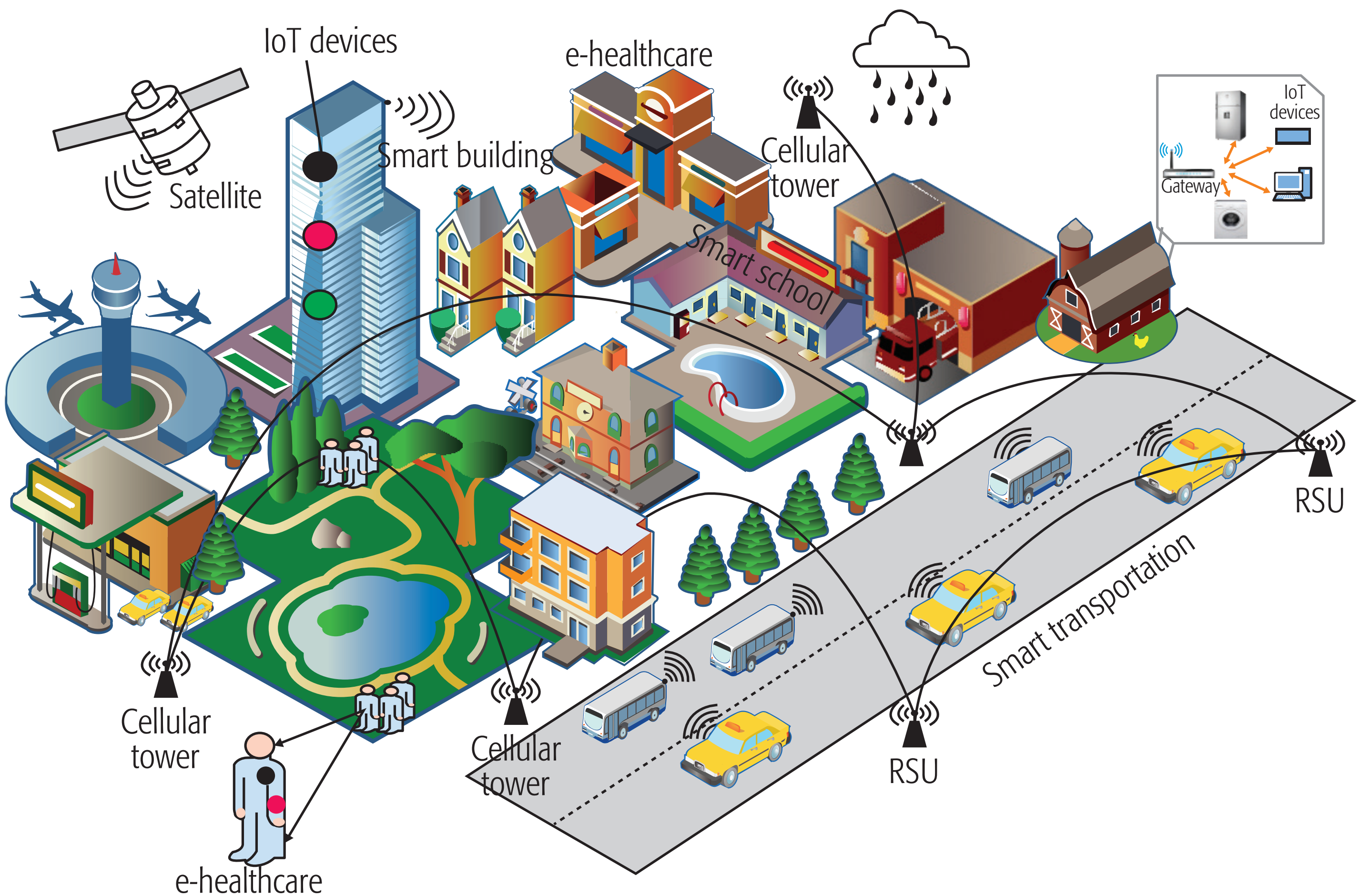
IoT network

- Smart devices
  - Sensors
  - RFID
  - Wi-Fi
- Smart environments
- Smart homes

Smart homes/Smart buildings

- Instrumenting buildings with IoT technologies may help in both **reducing the consumption of resources** associated to buildings (electricity, water) as well as in **improving the satisfaction level of humans** populating it
- A key role is played by **sensors**, which are used to both **monitor resource consumptions** as well as to **proactively detect current users' needs**
- Energy monitoring
- Home automation
- Home media services
- Home security
- Home confort

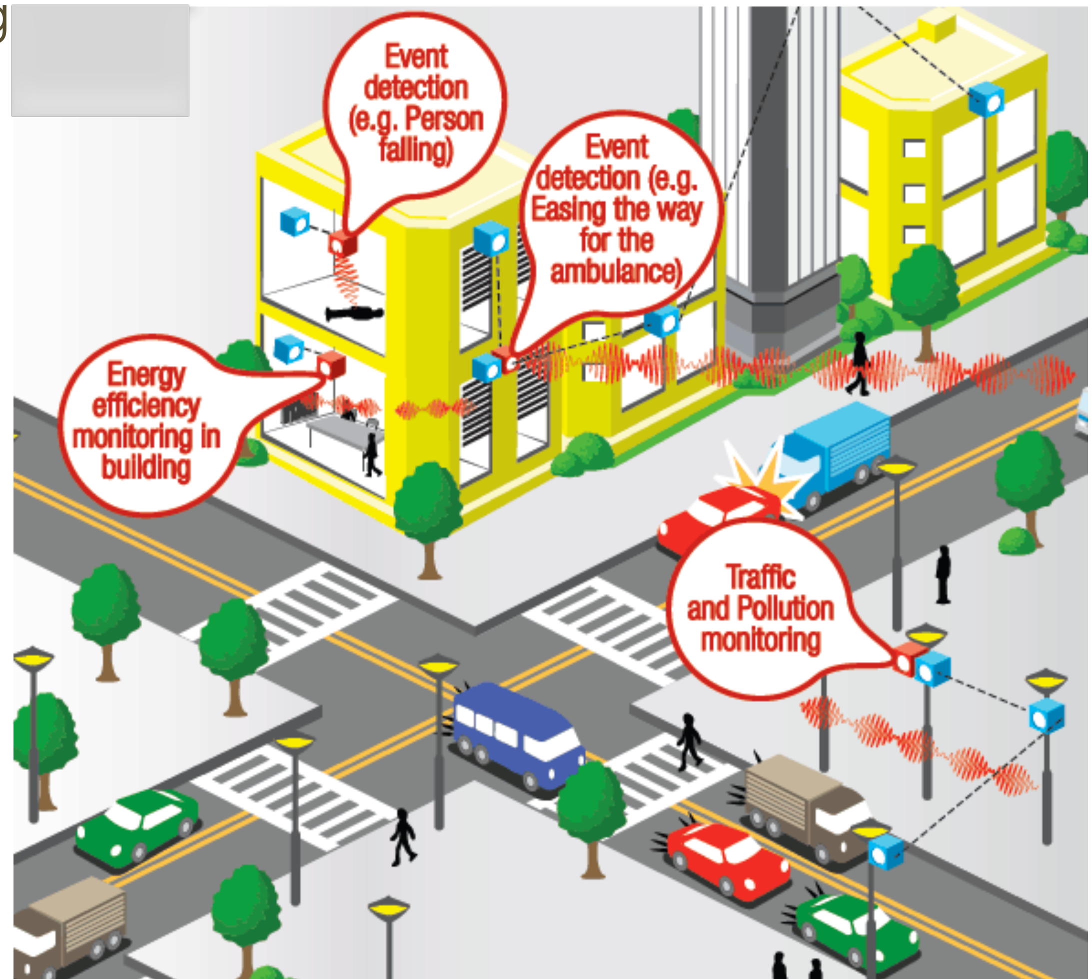
Smart city



IoT devices
e-healthcare
Smart building
Satellite
Cellular tower
Smart school
IoT devices
Gateway
RSU
Smart transportation
Cellular tower
Cellular tower
RSU
e-healthcare

Example: advanced traffic control system

- Car traffic monitoring in big cities or highways
- Traffic routing advice to avoid congestion
- Smart parking device system to provide drivers with automated parking advice
- Detecting violations
- Store information useful in case of accident



Dronet: networks of drones

Piloted drone

- Drones are used for many applications
- Aerial images
- Parcel delivery (amazon is experimenting a new service in the States)
- Most of applications **require the presence of human piloting**

Squad of **unmanned** drones

- **Unmanned** means without the physical presence of people in control
- Drones fly without being piloted by a central system
- Drones need to coordinate
- Drones need autonomy

Autonomous **networking**

We address **autonomy** mainly at the networking level (communication and routing) but also mobility

How do we make a squad of drones or an IoT network autonomous?

Autonomous means intelligent

One of the answers is

Reinforcement Learning

Overview of reinforcent learning (RL)

Reinforcement learning is concerned with the really foundational issue of

How an intelligent agent can **learn** to make a **good sequence of decisions**

That's a single sentence that summarizes what reinforcement learning is.

Learning to make a good sequence of decisions (under uncertainty)

1. Learn to make good **sequence of decisions**

- How can an intelligent agent make not just one decision but a whole sequence of decisions

2. Learn to make **good** sequence of decisions

- Goodness: we have some utility measure over the decisions that are being made

3. **Learn** to make good sequence of decisions

- Learning
- The agent does not know in advance how its decisions are going to affect the world
- What decisions might be associated with good outcomes
- The agent has to acquire this information through experience

RL, Behavior & Intelligence

Why do we have a brain?

- Primitive creature which evolves as following during its lifetime
- Larval stage: primitive brain & eye, swims around, attaches to a rock
- Adult stage: sits, digests brain
- We need a brain to guide decisions
- **When all the decisions have been completed maybe we no longer need a brain**

Example from Yael Niv

Applications

- RL is applicable to a huge number of domains
- Incredible work on robotics: agents do grasping, fold clothes
- Robotics engineers at University of California at Berkeley have developed a robot that can fold towels and socks

Key aspects of RL

- Optimization
- Delayed consequences
- Exploration
- Generalization

RL: optimization

Goal is to find an optimal way to make decisions

Yielding best outcomes

Or at least a very good strategy

RL: delayed consequences

Decisions now can impact things much later…

ice cream example

Introduces two challenges

When **planning**: decisions involve reasoning about not just immediate benefit of a decision but also its longer term ramifications

When **learning**: temporal credit assignment is hard (what caused later high or low rewards?)

RL: exploration

- Learning about the world by making decisions
  - Learn to ride a bike by trying (and falling)

- Censored data (you only get to learn about what you try to do)
  - Only get a reward for decision made
  - Don't know what would have happened if we had taken a different decision
  - Each choice is made at a particular time

RL: generalization

- Policy is mapping from past experience to action (defines the actions an agent should take based on its past experiences)
- Why not just pre-program a policy?
- We can have a huge number of states, thus we need generalization
- Even if we run into a particular state we have never seen before our agent still know what to do (Generalization allows an RL agent to handle situations it has never encountered before by applying knowledge from similar past experiences.)

How do we proceed?

Explore the world

Use experience (rewards) to guide future decisions

Other issues

Where do rewards come from?

And what happens if we get it wrong?

Robusteness/risk sensitivity

Robustness refers to the agent's ability to perform well in different, often unseen situations, without being too sensitive to slight changes in the environment

Risk sensitivity involves how the agent manages uncertainty, often opting for strategies that minimize risky outcomes

End of class goal

Given a networking problem (MAC or routing)	decide if it should be formulated as a RL problem
	if yes be able to define it formally (in terms of state space, action space, dynamics and reward model)
	state what algorithm (from class) is best suited to addressing it and justify your answer

Next class

- We will begin an overview of wireless technologies
  - **RFID**
  - Sensor networks
  - IoT networks
  - Backscattering networks
  - Dronets