**Multi-agent Systems**

- Systems composed of multiple (controllable and uncontrollable) agents
- System specification (its constraints and objectives):
  - E.g., Collision avoidance, surveillance, reachability
- Goal: a controller for each controllable agent such that the specification is satisfied
- Need for automated design and verification tools

**Example**

Two controllable robots, R1 and R2, are moving around a room, where there is a dynamic object moving back and forth in the second column. The robots must avoid collision with each other, and with the obstacle.

---

**Reactive Synthesis**

- Formal specification of the system using logical formulas (e.g., linear temporal logic)
  - $p_1 \equiv \text{always}(\text{object}=2 \rightarrow \text{next}(\text{object}=5))$ and $\text{always}(\text{object}=5 \rightarrow \text{next}(\text{object}=2))$
  - $p_2 \equiv \text{always}(\text{no collision with the object})$ and $\text{always}(\text{no collision between robots})$
- Specification $\phi = (p_1 \land p_2)$
- Reactive synthesis problem: Automatic synthesis of a system satisfying the specification
- Can be viewed as a game between two players: the system and its environment
  - The system player is the winner, a winning strategy can be synthesized for it

**Counter-example Guided Synthesis of Most General Strategies**

- Exploiting the structure in multi-agent systems
- Decomposition of the specification into sub-specifications with respect to agents
- Computing most general strategies over abstraction of games
  - Most general strategy covers the set of all possible strategies for the system
- Composition of solution spaces

---

**Hierarchical and Compositional Synthesis**

- Decomposition of the specification

---

**References**