Analyzing Unsynthesizable High-Level Robot Specifications in LTLMoP

Vasu Raman and Hadas Kress-Gazit
Autonomous Systems Lab, Cornell University

CAV 2011
Tuesday, July 19
HIGH-LEVEL TASKS:

- Patrolling a workspace for an indefinite period (\textit{infinite} behaviour)
- Responding to signals or objects of interest (\textit{reactive})

EXAMPLES:

Search and rescue missions
Autonomous Unmanned Vehicles
  e.g. DARPA Urban Challenge
Firefighting robots
Household chores (laundry, cleanup)
Restocking supermarket shelves
AUTOMATED HIGH-LEVEL ROBOT CONTROL:

Usual Approach

- Need to plan for a large number of contingencies.
- Does the implementation capture the high level requirements?
- Is the intended behavior even achievable?

   Do there exist robot controllers that guarantee fulfillment of the task?
FORMAL GUARANTEES:

Verifiable integration of high-level planning with continuous control.

LINEAR TEMPORAL LOGIC MISSION PLANNING TOOLKIT (LTLMoP)

- Discretized problem abstraction

- Structured English specifications ⇔ GR(1) formulas in LTL.
  - environment assumptions
  - desired system behaviour

- Synthesis of correct-by-construction controllers.
LTLMoP Overview

Robot Capability Definitions (Sensors/Actions)

(Specification Editor)

(Region Editor)
Robot Capability Definitions (Sensors/Actions)

Structured English-to-LTL Parser

LTLMoP Overview

(Specification Editor)

(log window)

(Region Editor)
Structured English-to-LTL Parser

Robot Capability Definitions (Sensors/Actions)

Synthesis

LTLMoP OVERVIEW

(Specification Editor)

(log window)

(Region Editor)

(Automaton)
Structured English-to-LTL Parser

Robot Capability Definitions (Sensors/Actions)

Synthesis

Hybrid Controller

(Specification Editor)

(LTLMoP Overview)

(Simulation)

(Region Editor)

(Automaton)
LTLMoP Overview

Structured English-to-LTL Parser

Robot Capability Definitions (Sensors/Actions)

Synthesis

Hybrid Controller

(Specification Editor)

(Simulation)

(log window)

(log filter options)

(Physical Robot)

(Region Editor)

(Automaton)
EXAMPLE: FIRE-FIGHTING SCENARIO
FIRE-FIGHTING SCENARIO
Regions:
- porch, deck, etc.
Regions:
• porch, deck, etc.
FIRE-FIGHTING SCENARIO

Regions:
• porch, deck, etc.

Robot actions:
• pick_up
• drop
• radio
• carrying_item
FIRE-FIGHTING SCENARIO

Regions:
• porch, deck, etc.

Robot actions:
• pick_up
• drop
• radio
• carrying_item

Sensors:
• hazardous_item
• person
FIRE-FIGHTING SCENARIO

Regions:
- porch, deck, etc.

Robot actions:
- pick_up
- drop
- radio
- carrying_item

Sensors:
- hazardous_item
- person

System Propositions
**FIRE-FIGHTING SCENARIO**

Regions:
- porch, deck, etc.

Robot actions:
- pick_up
- drop
- radio
- carrying_item

Sensors:
- hazardous_item
- person

System Propositions

Environment Propositions
**Fire-fighting Scenario**

Env starts with false
Robot starts with false
Robot starts in *porch*

If you were in *porch* then do not *hazardous_item*

Do *pick_up* if and only if you are sensing *hazardous_item* and you are not activating *carrying_item*

If you did not activate *carrying_item* then always not *porch*

Do *radio* if and only if you are sensing *person*

If you are activating *radio* or you were activating *radio* then stay there

If you are not activating *carrying_item* and you are not activating *radio* then visit *dining*

---

**Regions:**
- *porch*, *deck*, etc.

**Robot actions:**
- *pick_up*
- *drop*
- *radio*
- *carrying_item*

**Sensors:**
- *hazardous_item*
- *person*
**Fire-fighting Scenario**

**Initial Conditions**

- Environment starts with false
- Robot starts with false
- Robot starts in porch

If you were in **porch** then do not **hazardous_item**

Do **pick_up** if and only if you are sensing **hazardous_item** and you are not activating **carrying_item**

If you did not activate **carrying_item** then always not **porch**

Do **radio** if and only if you are sensing **person**

If you are activating **radio** or you were activating **radio** then stay there

If you are not activating **carrying_item** and you are not activating **radio** then visit **dining**

**Regions:**
- porch, deck, etc.

**Robot actions:**
- **pick_up**
- **drop**
- **radio**
- **carrying_item**

**Sensors:**
- **hazardous_item**
- **person**
Fire-fighting Scenario

Regions:
• porch, deck, etc.

Robot actions:
• pick_up
• drop
• radio
• carrying_item

Sensors:
• hazardous_item
• person

Env starts with false
Robot starts with false
Robot starts in porch

If you were in porch then do not hazardous_item

Do pick_up if and only if you are sensing hazardous_item and you are not activating carrying_item

If you did not activate carrying_item then always not porch

Do radio if and only if you are sensing person

If you are activating radio or you were activating radio then stay there

If you are not activating carrying_item and you are not activating radio then visit dining
**Fire-fighting Scenario**

Env starts with false
Robot starts with false
Robot starts in **porch**

If you were in **porch** then
do not **hazardous_item**

Do **pick_up** if and only if you are sensing **hazardous_item** and you are not activating **carrying_item**

If you did not activate **carrying_item** then
always not **porch**

Do **radio** if and only if you are sensing **person**

If you are activating **radio** or you were activating **radio** then stay there

If you are not activating **carrying_item**
and you are not activating **radio** then visit **dining**

**Regions:**
- **porch**, **deck**, **etc.**

**Robot actions:**
- **pick_up**
- **drop**
- **radio**
- **carrying_item**

**Sensors:**
- **hazardous_item**
- **person**
**Fire-fighting Scenario**

**Regions:**
- porch, deck, etc.

**Robot actions:**
- pick_up
- drop
- radio
- carrying_item

**Sensors:**
- hazardous_item
- person

**Env starts with false**
**Robot starts with false**
**Robot starts in porch**

If you were in porch then do not hazardous_item

Do pick_up if and only if you are sensing hazardous_item and you are not activating carrying_item

If you did not activate carrying_item then always not porch

Do radio if and only if you are sensing person

If you are activating radio or you were activating radio then stay there

If you are not activating carrying_item and you are not activating radio then visit dining
**Fire-fighting Scenario**

Env starts with false
Robot starts with false
Robot starts in **porch**

If you were in **porch** then
do not **hazardous_item**

Do **pick_up** if and only if you are sensing **hazardous_item** and you are not activating **carrying_item**

If you did not activate **carrying_item** then
always not **porch**

Do **radio** if and only if you are sensing **person**

If you are activating **radio** or you were activating **radio** then stay there

If you are not activating **carrying_item** and you are not activating **radio** then visit **dining**

**Regions:**
- **porch, deck, etc.**

**Robot actions:**
- **pick_up**
- **drop**
- **radio**
- **carrying_item**

**Sensors:**
- **hazardous_item**
- **person**
PROBLEM: UNSYNTHESIZABLE SPECIFICATIONS

- **UNSATISFIABLE:**
  System requirements cannot be fulfilled in any environment

- **UNREALIZABLE:**
  System requirements cannot be fulfilled in some admissible environment

GOALS:

- Identify the cause of failure in the LTL specification
- Map it back to structured English
LTLMoP Overview

Structured English-to-LTL Parser

Robot Capability Definitions (Sensors/Actions)

Synthesis

Hybrid Controller

Specifi:ation Editor

Region Editor

(Automaton)

(Portch)

(living)

(bedroom)

(deck)

(start)

(suspend)

(pause/resume & export log buttons)

(log filter options)

(Simulation)

(Physical Robot)
LTLMoP Overview

Structured English-to-LTL Parser

Robot Capability Definitions (Sensors/Actions)

Synthesis

Layer of Analysis for Specifications

Hybrid Controller

(Specification Editor)

(Simulation)

(Region Editor)

(Automaton)

(Physical Robot)
Region propositions

1. Initial conditions
2. Environment starts with false
3. Robot starts with false
4. Robot starts in porch
5. Assumptions about the environment
6. If you were in porch then do not hazardous_item
7. Define robot safety including how to pick up
8. Do pick_up if and only if you are sensing hazardous_item and you are not activating carrying_item
9. If you did not activate carrying_item then always not porch
10. Define when and how to radio
11. Do radio if and only if you are sensing person
12. If you are activating radio or you were activating radio then stay there
13. Patrol goals
14. If you are not activating carrying_item and you are not activating radio then visit dining

ERROR: Specification was unrealizable.

RESULT
System is unrealizable because the environment can force a safety violation.
No automaton synthesized.
ERROR: Specification was unrealizable.

RESULT
System is unrealizable because the environment can force a safety violation.
No automaton synthesized.
ERROR: Specification was unrealizable.

RESULT
System is unrealizable because the environment can force a safety violation.
No automaton synthesized.
# Initial conditions
1. Env starts with false
2. Robot starts with false
3. Robot starts in porch

# Assumptions about the environment
4. If you were in porch then do not hazardous_item
5. Define robot safety including how to pick up
6. Do pick_up if and only if you are sensing hazardous_item and you are not activating carrying_item
7. If you did not activate carrying_item then always not porch
8. Define when and how to radio
9. Do radio if and only if you are sensing person
10. If you are activating radio or you were activating radio then stay there
11. Patrol goals
12. If you are not activating carrying_item and you are not activating radio then visit dining

## Propose propositions
- Region propositions
- Sensor propositions
- Action propositions

## Specification text

ERROR: Specification was unrealizable.

RESULT
System is unrealizable because the environment can force a safety violation.
No automaton synthesized.
# Initial conditions
1. Env starts with false
2. Robot starts with false
3. Robot starts in porch

# Assumptions about the environment
4. If you were in porch then do not hazardous_item
5. Define robot safety including how to pick up
6. Do pick_up if and only if you are sensing hazardous_item and you are not activating carrying_item
7. If you did not activate carrying_item then always not porch

# Define when and how to radio
8. Do radio if and only if you are sensing person
9. If you are activating radio or you were activating radio then stay there

# Patrol goals
10. If you are not activating carrying_item and you are not activating radio then visit dining

ERROR: Specification was unrealizable.

RESULT
System is unrealizable because the environment can force a safety violation.
No automaton synthesized.
# Initial conditions
1. Env starts with false
2. Robot starts with false
3. Robot starts in porch

# Assumptions about the environment
4. If you were in porch then do not hazardous_item

# Define robot safety including how to pick up
5. Do pick_up if and only if you are sensing hazardous_item and you are not activating carrying_item
6. If you did not activate carrying_item then always not porch

# Define when and how to radio
7. Do radio if and only if you are sensing person
8. If you are activating radio or you were activating radio then stay there

# Patrol goals
9. If you are not activating carrying_item and you are not activating radio then visit dining

System is unrealizable because the environment can force a safety violation.
No automaton synthesized.
# Initial conditions
1. Env starts with false
2. Robot starts with false
3. Robot starts in porch

# Assumptions about the environment
4. If you were in porch then do not hazardous_item

# Define robot safety including how to pick up
5. Do pick_up if and only if you are sensing hazardous_item and you are not activating carrying_item
6. If you did not activate carrying_item then always not porch

# Define when and how to radio
7. Do radio if and only if you are sensing person
8. If you are activating radio or you were activating radio then stay there

# Patrol goals
9. If you are not activating carrying_item and you are not activating radio then visit dining

ERROR: Specification was unrealizable.

RESULT:
System is unrealizable because the environment can force a safety violation.
No automaton synthesized.
# Define robot safety including how to pick_up
Do pick_up if and only if you are sensing hazardous_item and you are not activating carrying_item
If you did not activate carrying_item then always not porch

# Define when and how to radio
Do radio if and only if you are sensing person
If you are activating radio or you were activating radio then stay there

System is unrealizable because the environment can force a safety violation
# Initial conditions
1. Env starts with false
2. Robot starts with false
3. Robot starts in porch

# Assumptions about the environment
4. If you were in porch then do not hazardous_item and do not person
5. If you did not activate carrying_item then always not porch

# Define robot safety including how to pick up
6. Do pick_up if and only if you are sensing hazardous_item and you are not activating carrying_item
7. If you were activating radio or you were activating radio then stay there

# Define when and how to radio
8. If you are not activating carrying_item and you are not activating radio then visit dining
9. Visit porch

## Regions:
- boundary
- kitchen
- porch
- deck
- bedroom
- dining
- living

## Sensors:
- fire
- person
- hazardous_item

## Actions:
- pick_up
- drop
- radio
- extinguish

## Custom Propositions:
carrying_item

**ERROR:** Specification was unrealizable.

RESULT
- System highlighted goal(s) unrealizable
- No automaton synthesized.
# Initial conditions
1. Env starts with False
2. Robot starts with False
3. Robot starts in porch

# Assumptions about the environment
4. If you were in porch then do not hazardous_item and do not person
5. If you did not activate carrying_item then always not porch

# Define robot safety including how to pick up
6. Do pick_up if and only if you are sensing hazardous_item and you are not activating carrying_item
7. If you are not activating radio or you were activating radio then stay there
8. If you are not activating radio then visit dining
9. Visit porch

# Define when and how to radio
10. Do radio if and only if you are sensing person
11. If you are activating radio or you were activating radio then stay there
12. If you are not activating carrying_item and you are not activating radio then visit dining

# Patrol goals
13. If you are not activating carrying_item and you are not activating radio then visit dining

ERROR: Specification was unrealizable.

RESULT
System highlighted goal(s) unrealizable.
No automaton synthesized.
#Define robot safety including how to pick_up
Do **pick_up** if and only if you are sensing **hazardous_item** and you are not activating **carrying_item**
If you did not activate **carrying_item** then always not **porch**

#Patrol Goals
Visit **porch**

System highlighted goal(s) unrealistic
OTHER TOOLS FOR ANALYSIS

- Interactive game to help explain unrealizability
  - Let the user interact with an environment constructed to thwart the system.
  - Similar to RATSY\(^1\), augmented with domain-specific interface.

CURRENT AND FUTURE WORK

- Identifying domain-specific special cases of unsatisfiability
  - e.g. disconnected topology

- Further narrowing down the cause of unsynthesizability
  - Unsatisfiable/unrealizable cores

- Suggesting changes to the specification that would allow synthesis
  - Add environment assumptions
  - Weaken system requirements

---

REMINDER: Tool Demo Session 3:00-6:00pm

LTLMoP: [https://github.com/LTLMoP](https://github.com/LTLMoP) (GPL)