August 21, 2026 | SAFL AI Futures Dialogue

Beyond the Lab: User-Aligned Assessment of Taskable AI Systems

Pulkit Verma

pulkitv@cse.iitm.ac.in

pulkitverma.net

01

Introduction

02

Foundational Approach

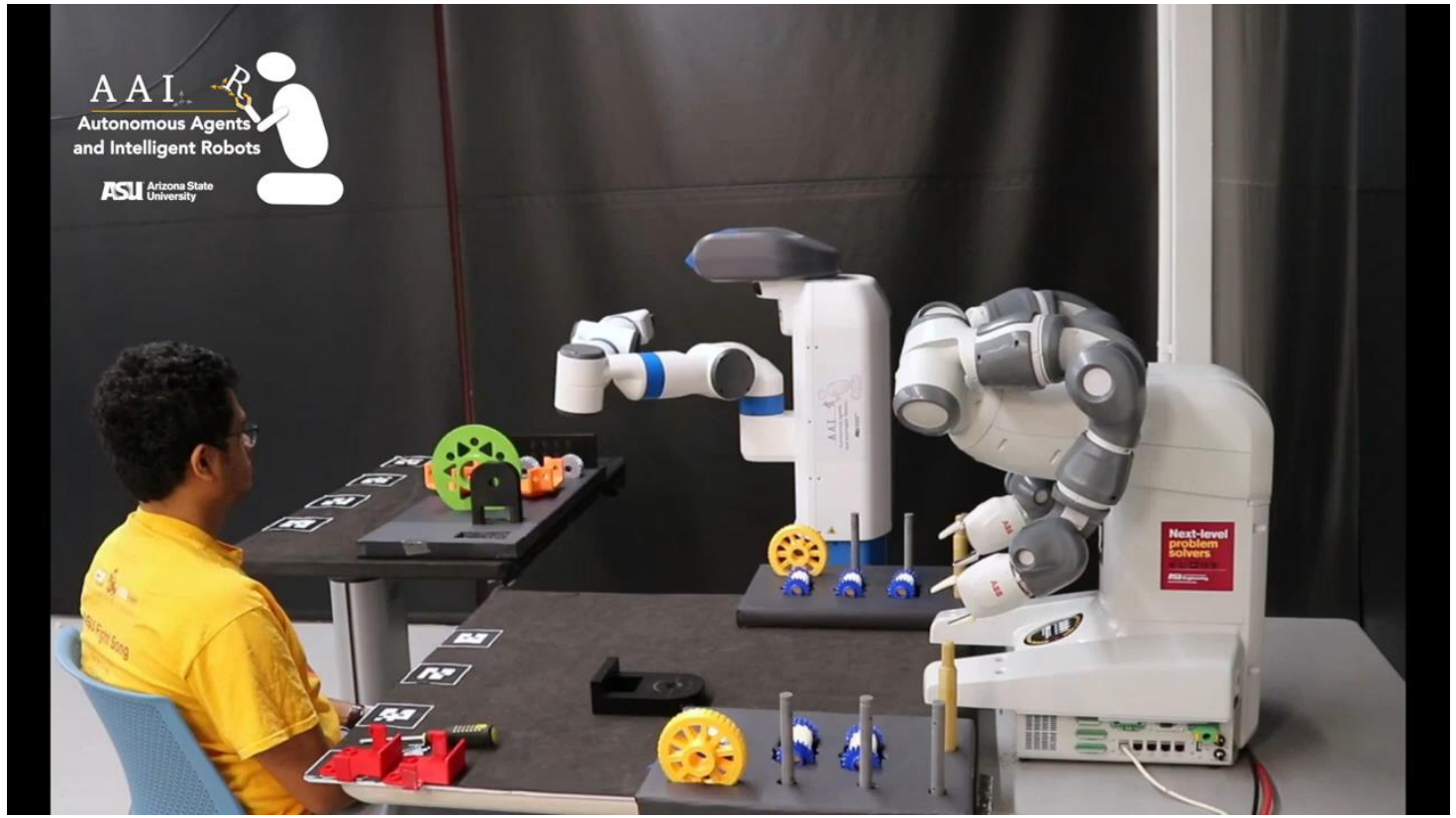
03

Generalizations

04

Future Work

Taskable AI Systems: Systems that can Learn and Plan



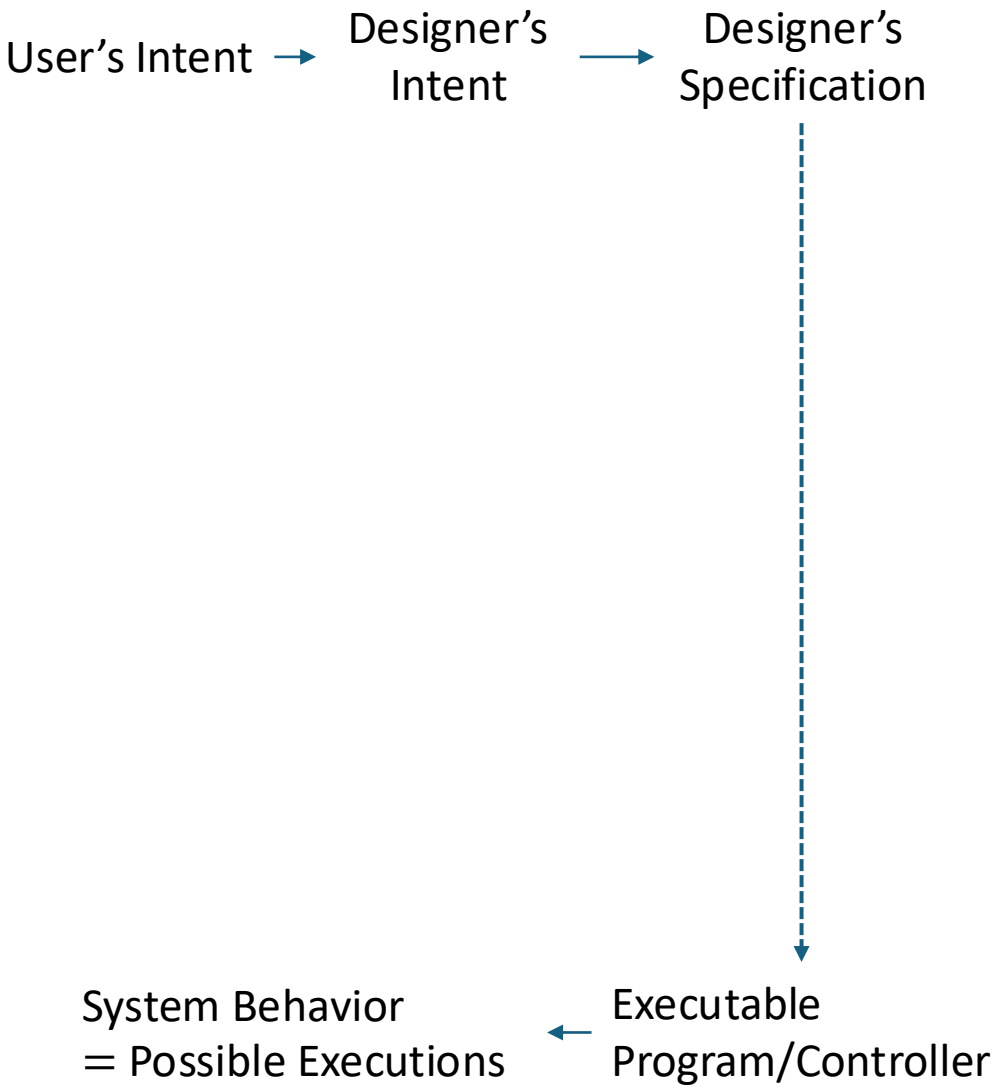
User gives a task and Robots have to complete it

- Sequential Decision-Making Systems.
- Systems designed to be able to help user.
- User has a task in mind and expects the AI system to help in that task.

Personalized Assessment of AI Systems that can Learn and Plan

- Users would like to give AI systems multiple tasks.
  - How would users know what the AI systems can do?

Classical Notion of Verification

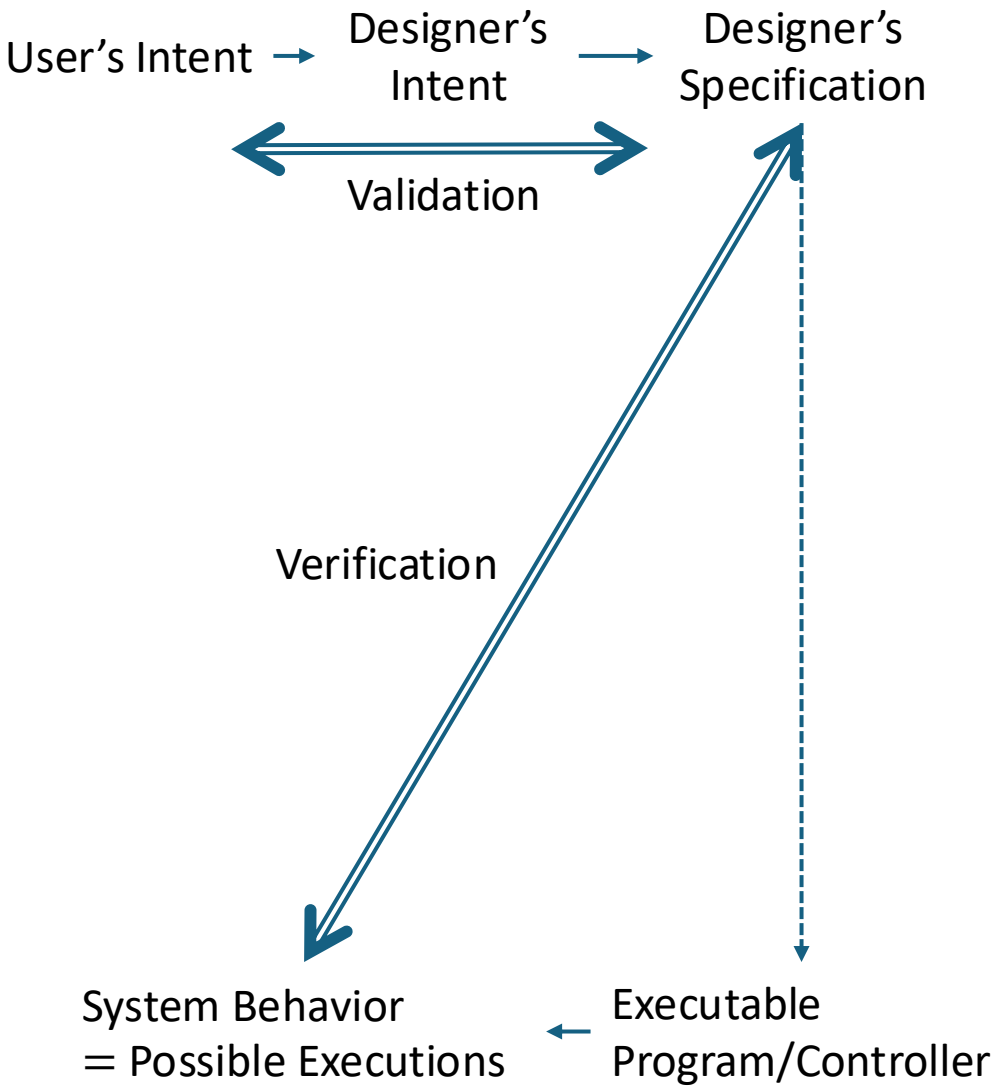


[Srivastava, Fainekos, **Verma**, Bramblett; AIA @ IJCAI '25]

Classical Notion of Verification

Input for Verification: Executable Program/Controller
(includes task spec)
+ Model/assumptions on env
+ Safety property

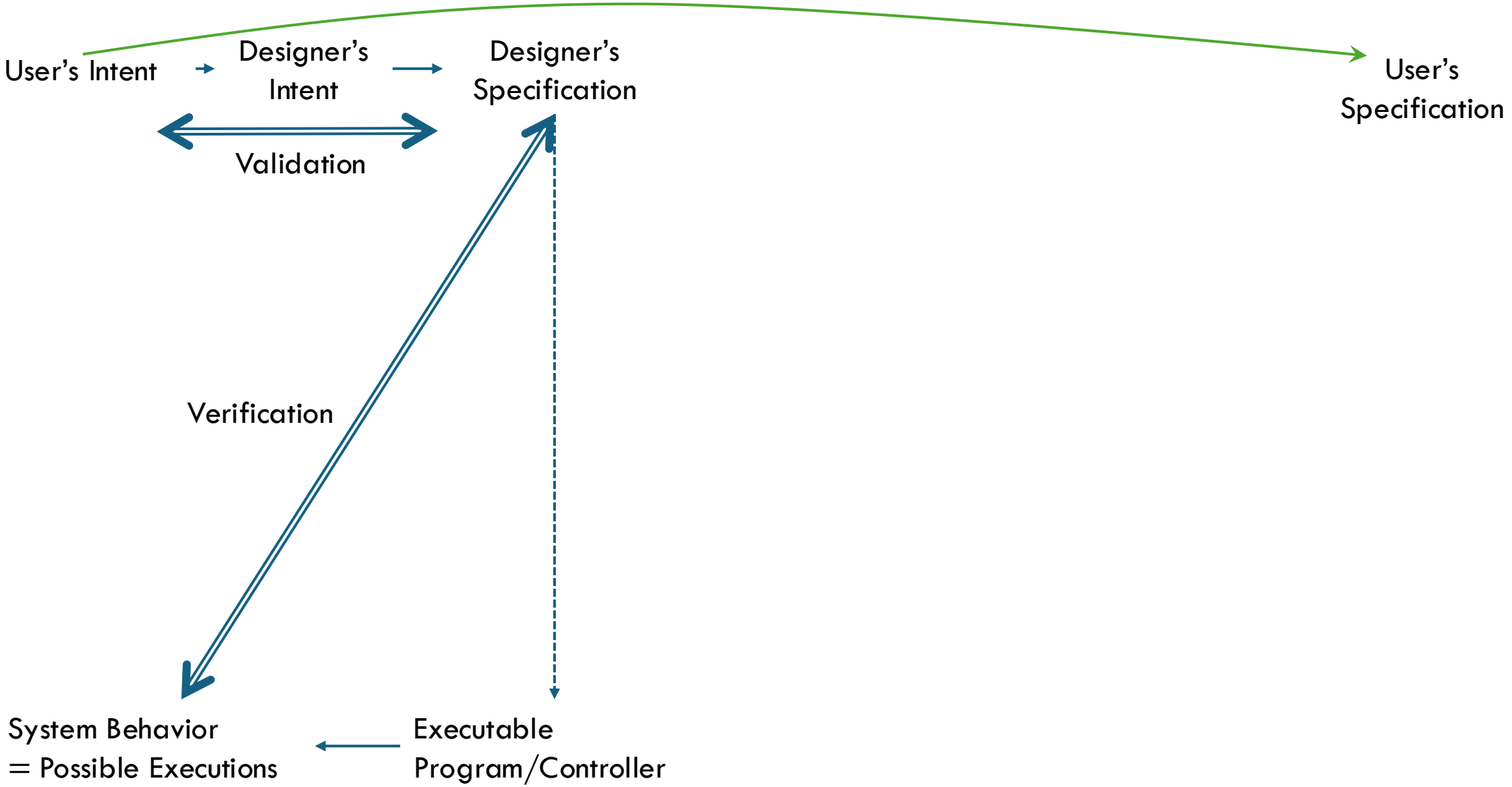
The designer plays a central role



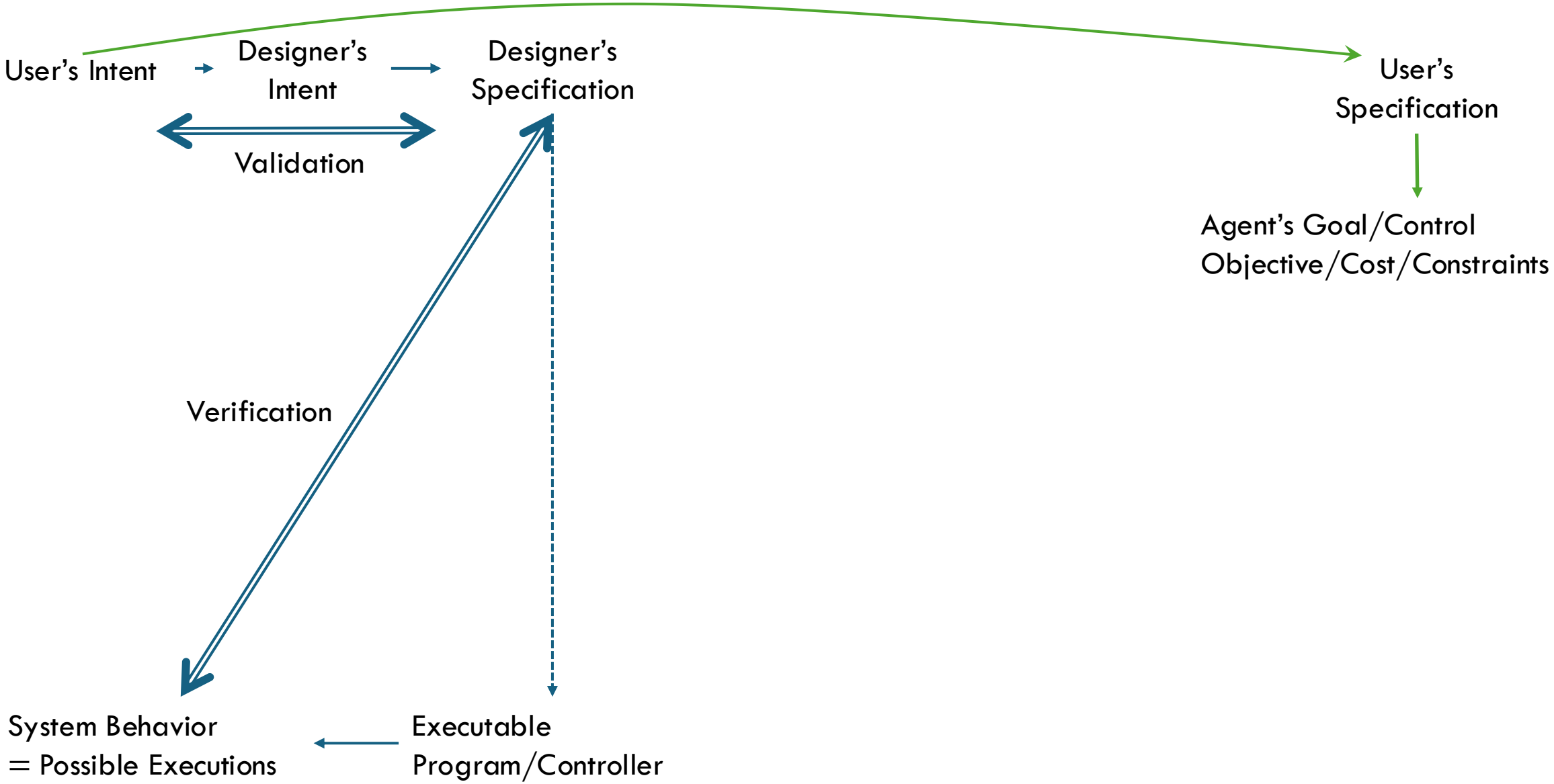
[Srivastava, Fainekos, **Verma**, Bramblett; AIA @ IJCAI '25]

Personalized Assessment of AI Systems that can Learn and Plan

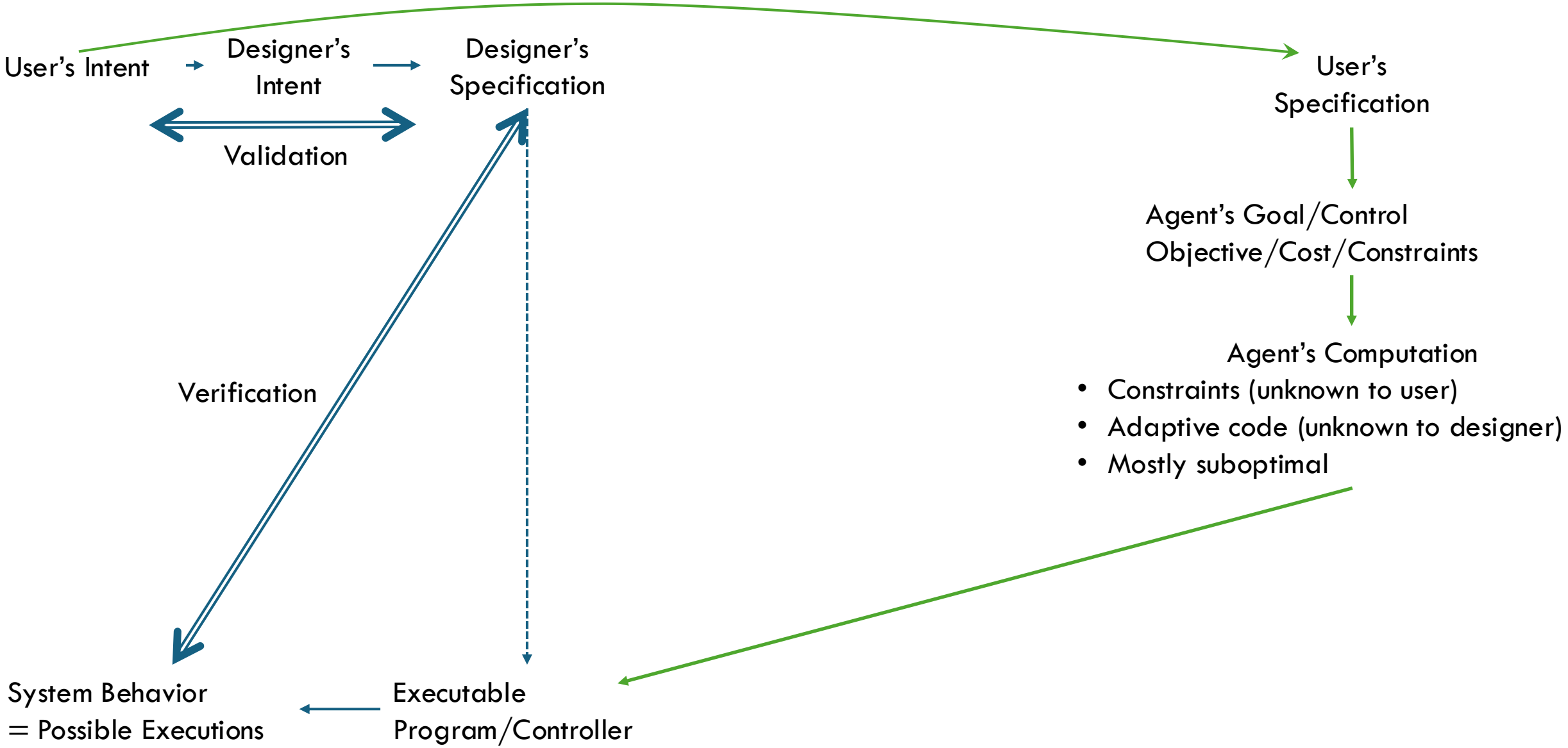
- Users would like to give AI systems multiple tasks.
  - How would users know what the AI systems can do?
- AI systems should support third-party assessment.
- The assessment should work with:
  - Adaptive AI systems.
  - Black-Box AI systems.



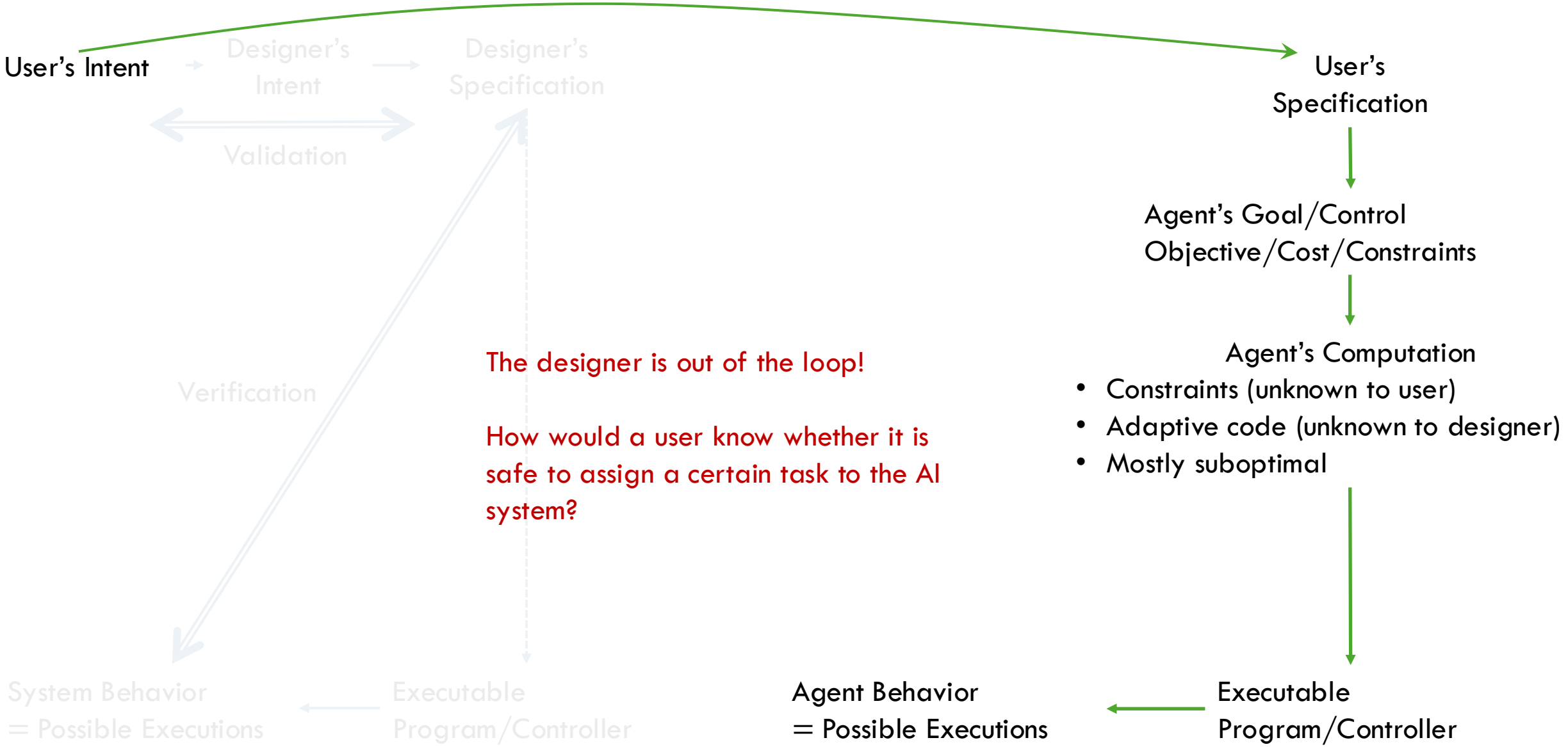
[Srivastava, Fainekos, **Verma**, Bramblett; AIA @ IJCAI '25]



[Srivastava, Fainekos, **Verma**, Bramblett; AIA @ IJCAI '25]



[Srivastava, Fainekos, **Verma**, Bramblett; AIA @ IJCAI '25]

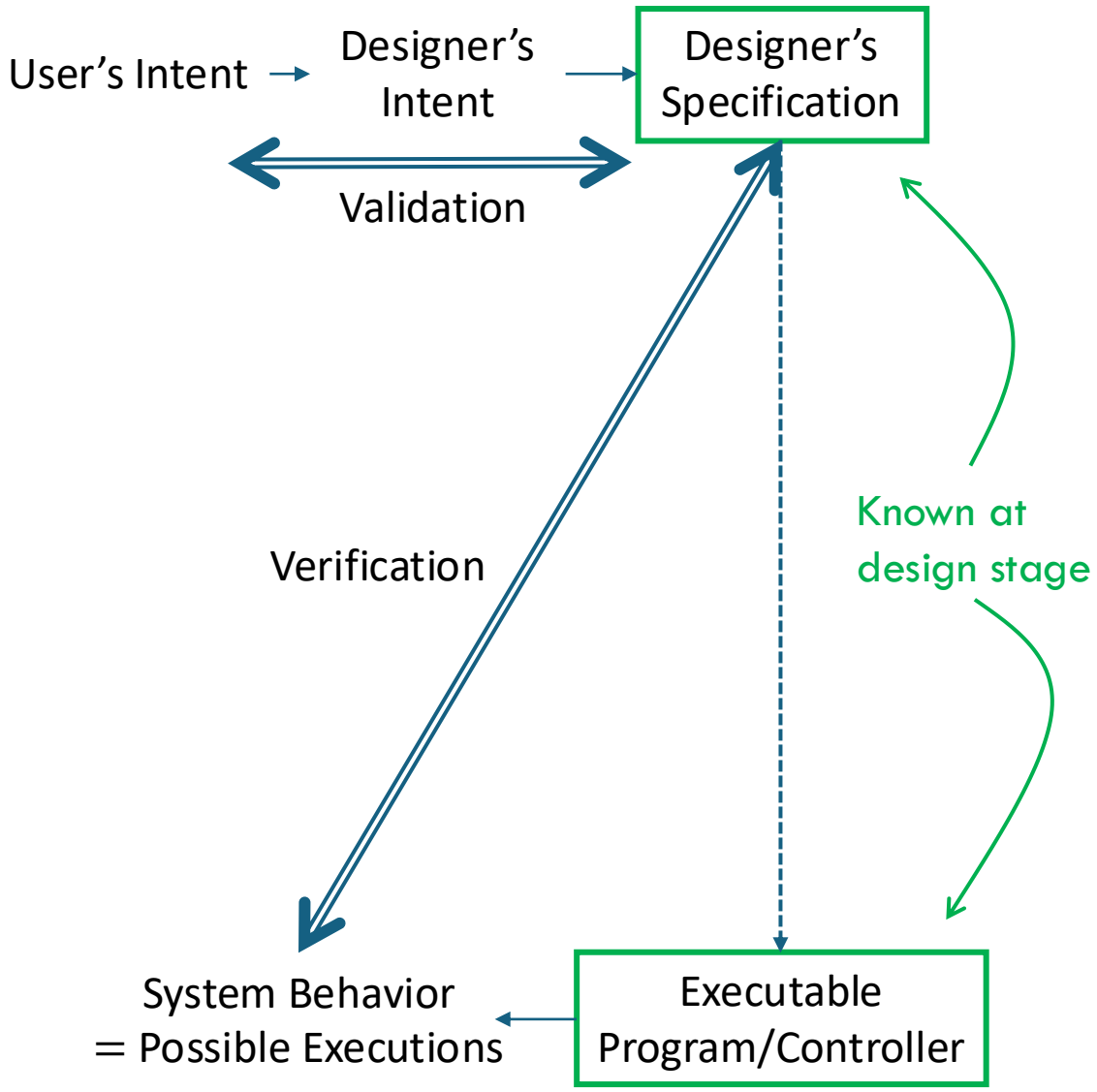


The designer is out of the loop!

How would a user know whether it is safe to assign a certain task to the AI system?

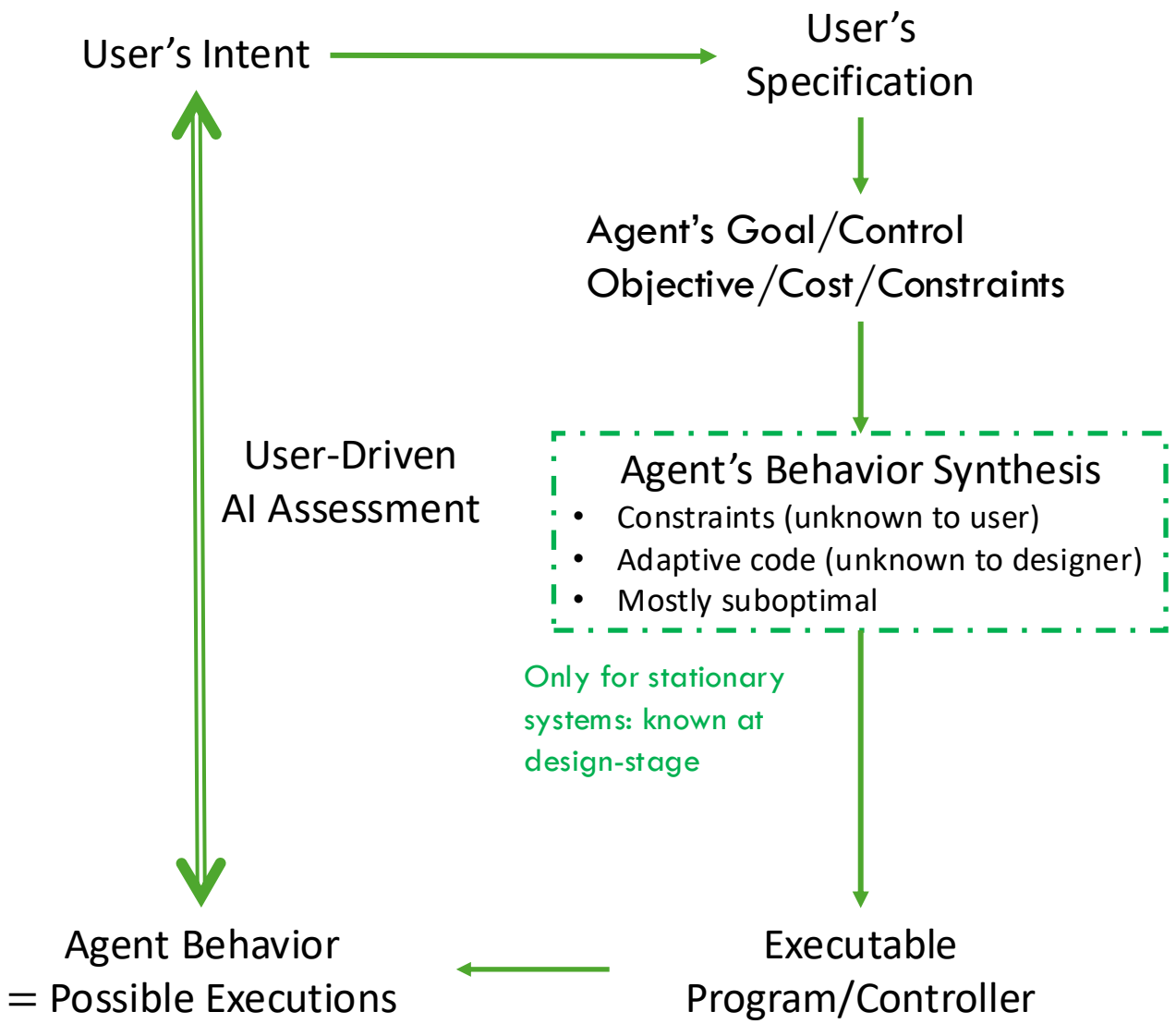
[Srivastava, Fainekos, **Verma**, Bramblett; AIA @ IJCAI '25]

Conventional Verification

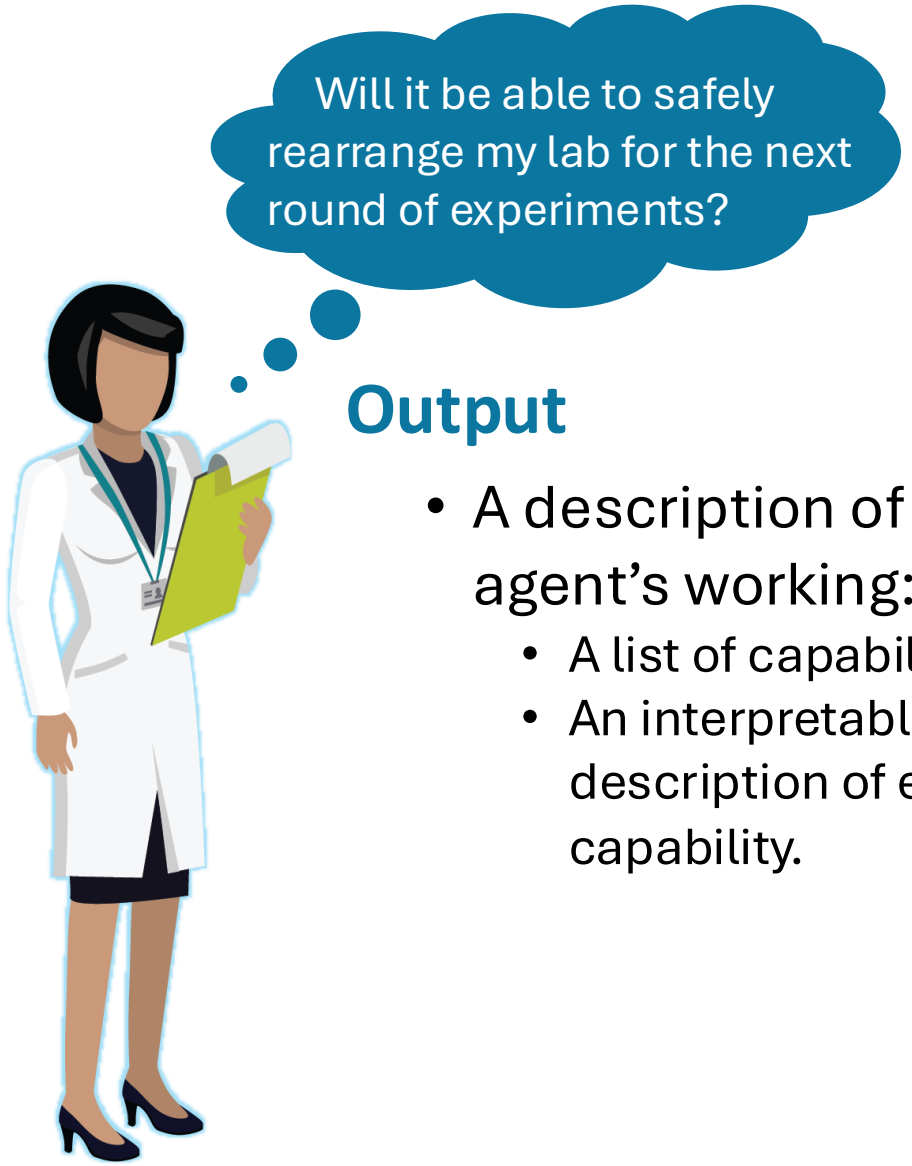


[Srivastava, Fainekos, **Verma**, Bramblett; AIA @ IJCAI '25]

Needed for AI Systems

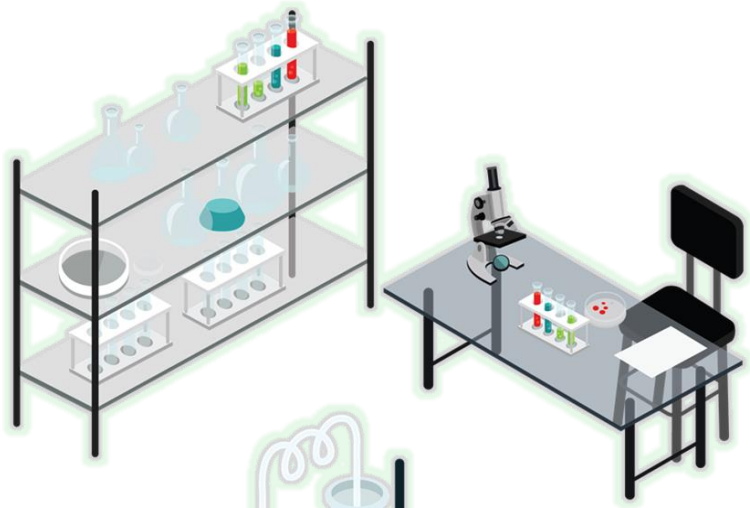


The Assessment Problem



Output

- A description of agent's working:
  - A list of capabilities.
  - An interpretable description of each capability.



Black-Box AI

- Arbitrary internal implementation
- Comes with a Trained Policy

```
at(p0,cell_6_3)
clear(cell_0_0)
door_at(cell_9_2)
next_to(m0)
alive(m0)
key_at(9_4)
```

[Input]
Concepts that the user understands

Personalized AI-Assessment Module (AAM)

Black-Box AI

Arbitrary internal implementation

Doesn't know user's vocabulary

```
at(p0,cell_6_3)
clear(cell_0_0)
door_at(cell_9_2)
next_to(m0)
alive(m0)
key_at(9_4)
```

[Input]
Concepts that the user understands

Personalized AI-Assessment Module (AAM)

Query

Response

simulator

Black-Box AI

Arbitrary internal implementation

Doesn't know user's vocabulary

[Input]
Concepts that the user understands

[Output]
User-Interpretable description of Black-Box AI's capabilities

Personalized AI-Assessment Module (AAM)

This Work

Query

Response

simulator

Black-Box AI

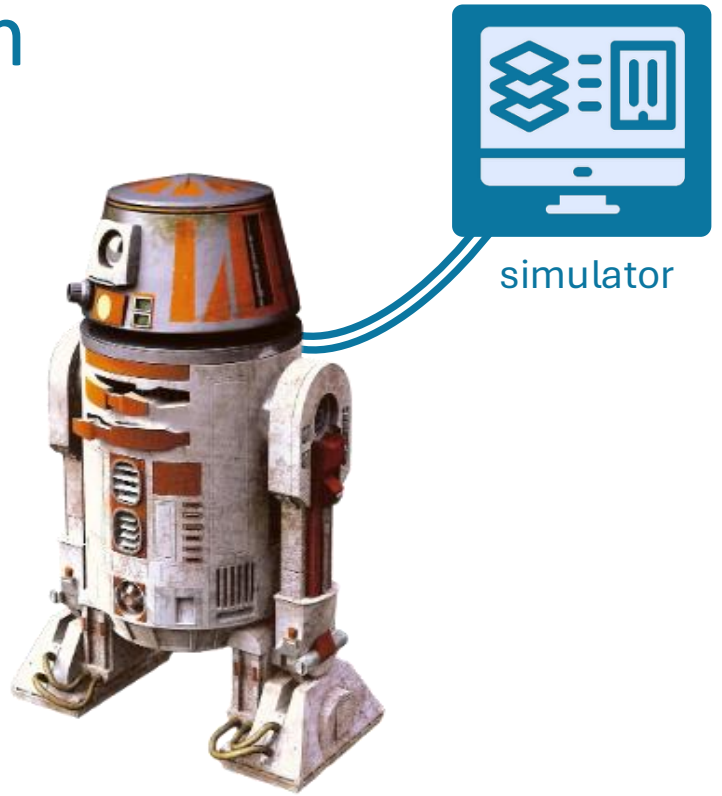
Arbitrary internal implementation

Doesn't know user's vocabulary

The Assessment Problem

Black-Box (Taskable) AI

- Can be connected to a simulator.
- Can have arbitrary internal implementation.
- Does not know input vocabulary.



Black-Box AI

Input

- Predicates (User vocabulary)
  - With their evaluation functions

Output

- A description of agent's working:
  - A list of capabilities.
  - An interpretable description of each capability.

Interpretable Description: PDDL/PPDDL

```
(:action open-door
   :parameters (?l1)
   :precondition (and
      (has_key)
      (player_at ?l1)
      (door_adjacent ?l1))
   :effect (probabilistic
      0.95 (and (door_open))
      0.05 (and (not (has_key))
                (game-over))
)
```

Precondition: This condition must be true for this action to execute

Effect: This is a set of conditions, one of which becomes true when this action is executed

Probabilities: Each set of effect has an associated probability with which that effect set is executed

Interpretable: Easily Convertible to Natural Language

```
(:action open-door
   :parameters (?l1)
   :precondition (and
      (has_key)
      (player_at ?l1)
      (door_adjacent ?l1))
   :effect (probabilistic
      0.95 (and (door_open))
      0.05 (and (not(has_key))
                (game-over))
)
```

The player can open the door when in location ?l1 if:

- It has the key
- The player is at location ?l1
- The door is adjacent to location ?l1

After executing that capability:

- With 95% probability, the door will open
- With 5% probability, the player will not have the key and the game will be over

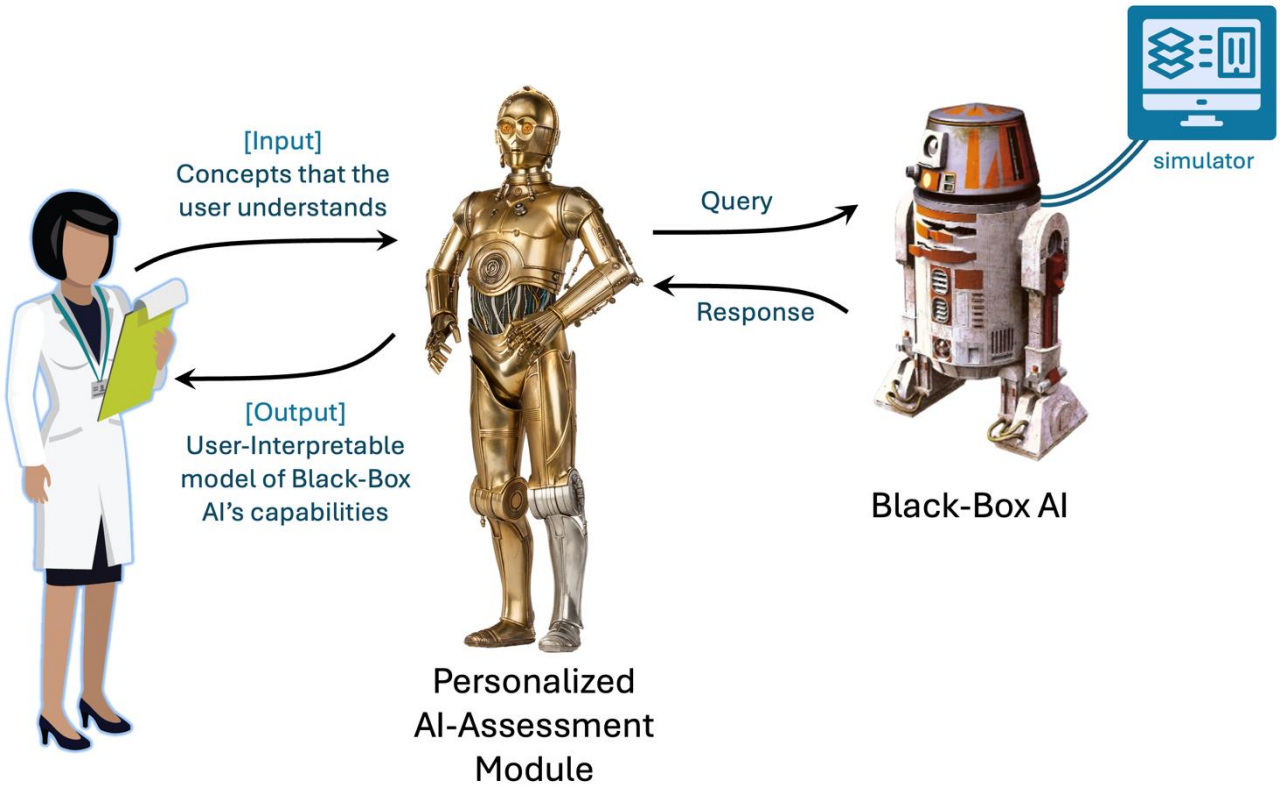
01
Introduction

02
Foundational Approach

03
Generalizations

04
Future Work

Deterministic and Stationary Setting



Assumptions

- User's vocabulary matches simulator's vocabulary.
- Black-Box AI provides a list of capabilities.
- Stationary agent model.
- Deterministic environment.
- Fully observable setting.

Deterministic and Stationary Setting

Input

- Predicates (User vocabulary)
  - With their evaluation functions
- List of capabilities.

Output

- PDDL-like description of each capability.

Siddharth Srivastava

Shashank R Marpally

Asking the Right Questions: Learning Interpretable Action Models Through Query Answering

Pulkit Verma, Shashank Rao Marpally, Siddharth Srivastava

In Proceedings of the Thirty-Fifth AAAI Conference on Artificial Intelligence, 2021

Assumptions

- User's vocabulary matches simulator's vocabulary.
- Black-Box AI provides a list of capabilities.
- Stationary agent model.
- Deterministic environment.
- Fully observable setting.

Exponential Search for Learning Correct Description

- Consider the following 4 predicates/concepts:
  - `(has_key)`
  - `(door_open)`
  - `(door_adjacent ?x)`
  - `(player_at ?x)`
- Consider just one capability: `(open-door ?x)`
- $9^{|C|\times|P|}$ = $9^{1\times4}$=6561 possible models (Assuming deterministic models/ descriptions, i.e., no probabilities).

```
(:action open-door
  :parameters (?l1)
  :precondition (and
    (+/-/∅)(has_key)
    (+/-/∅)(door_open)
    (+/-/∅)(door_adjacent ?l1)
    (+/-/∅)(player_at ?l1))
  :effect (and
    (+/-/∅)(has_key)
    (+/-/∅)(door_open)
    (+/-/∅)(door_adjacent ?l1)
    (+/-/∅)(player_at ?l1))
```

Simple Queries

	Plan Outcome Queries	State Reachability Query
Query	In state s_I, what will happen if you execute the plan $\pi = \langle c_1, \ldots, c_n \rangle$?	Can you go from state s_I to state s_F?
Response	I can execute first ℓ steps of the plan, ending up in state s_F.	Yes / No.

- How to generate the queries?
- How to use the responses to generate models?

We have a reduction that converts this to a planning problem, so it automatically generates queries.

AAM learns Accurate Deterministic Models

- Theorem (*termination*) : The algorithm terminates after a finite number of iterations.

- Theorem (*soundness*): The resulting (set of) model(s) is(are) functionally equivalent to the ground truth model.

01
Introduction

02
Foundational Approach

03
Generalizations

04
Future Work

Stochastic and Stationary Setting

Input

- Predicates (User vocabulary)
  - With their evaluation functions
- List of capabilities.

Output

- PPDDL-like description of each capability.

Siddharth Srivastava

Rushang Karia

Autonomous Capability Assessment of Sequential Decision-Making Systems in Stochastic Settings

Pulkit Verma, Rushang Karia, Siddharth Srivastava

In Proceedings of the Thirty-seventh Conference on Neural Information Processing Systems,, 2023

Assumptions

- User's vocabulary matches simulator's vocabulary.
- Black-Box AI provides a list of capabilities.
- Stationary agent model.
- ~~Deterministic~~ Stochastic environment.
- Fully observable setting.

Discovering Capabilities

Input

- Predicates (User vocabulary)
  - With their evaluation functions
- Samplers: high-level state to low-level state.
- Low-level state transitions.

Output

- List of capabilities.
- PDDL-like description of each capability.

Siddharth Srivastava

Shashank R Marpally

Discovering User-Interpretable Capabilities of Black-Box Planning Agents

Pulkit Verma, Shashank Rao Marpally, Siddharth Srivastava

In Proceedings of 19th International Conference on Principles of Knowledge Representation and Reasoning, 2022

Assumptions

- ~~User's vocabulary matches simulator's vocabulary~~.
- Black-Box AI provides a list of ~~capabilities~~. transitions.
- Stationary agent model.
- Deterministic environment.
- Fully observable setting.

Discovering Capabilities in Stochastic Settings

Input

- Predicates (User vocabulary)
  - With their evaluation functions
- Samplers: high-level state to low-level state.
- Low-level state transitions.

Output

- List of capabilities.
- PDDL-like description of each capability.

Assumptions

- ~~User's vocabulary matches simulator's vocabulary~~.
- Black-Box AI provides a list of ~~capabilities~~. transitions.
- Stationary agent model.
- Stochastic ~~Deterministic~~ environment.
- Fully observable setting.

Autonomous Assessment of Generalizability of AI Agent Capabilities

Daniel Bramblett, Rushang Karia, Adrian Ciotinga, Pulkit Verma, YooJung Choi, and Siddharth Srivastava

In ICAPS 2026 Workshop on Generalization in Planning, 2026

Differential Assessment

Input

- Initial model of the AI system.
  - Predicates (User vocabulary)
    - With their evaluation functions
  - List of capabilities.
- Observations collected from AI system.

Output

- Updated PDDL-like description of each capability.

Siddharth Srivastava

Rashmeet K Nayyar

Differential Assessment of Black-Box AI Agents

Rashmeet K Nayyar*, *Pulkit Verma**, *Siddharth Srivastava*

In Proceedings of Thirty-Sixth AAAI Conference on Artificial Intelligence, 2022

Assumptions

- User's vocabulary matches simulator's vocabulary.
- Black-Box AI provides a list of capabilities.
- Adaptive ~~Stationary~~ agent model.
- Deterministic environment.
- Fully observable setting.

01
Introduction

02
Foundational Approach

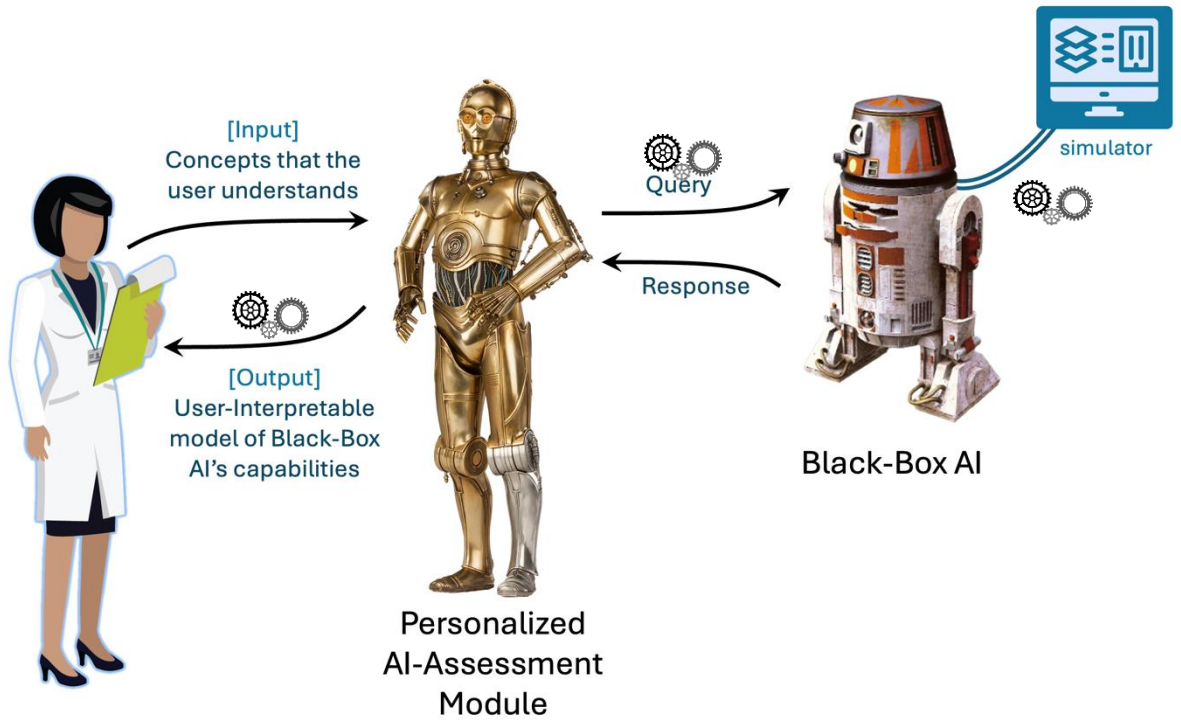
03
Generalizations

04
Future Work

Future Work: Complexity Analysis

- Perform extensive analysis of queries in terms of:
  - Complexity of generating them.
  - Complexity of answering them.
  - Complexity of inferring models from Black-Box AI's responses.

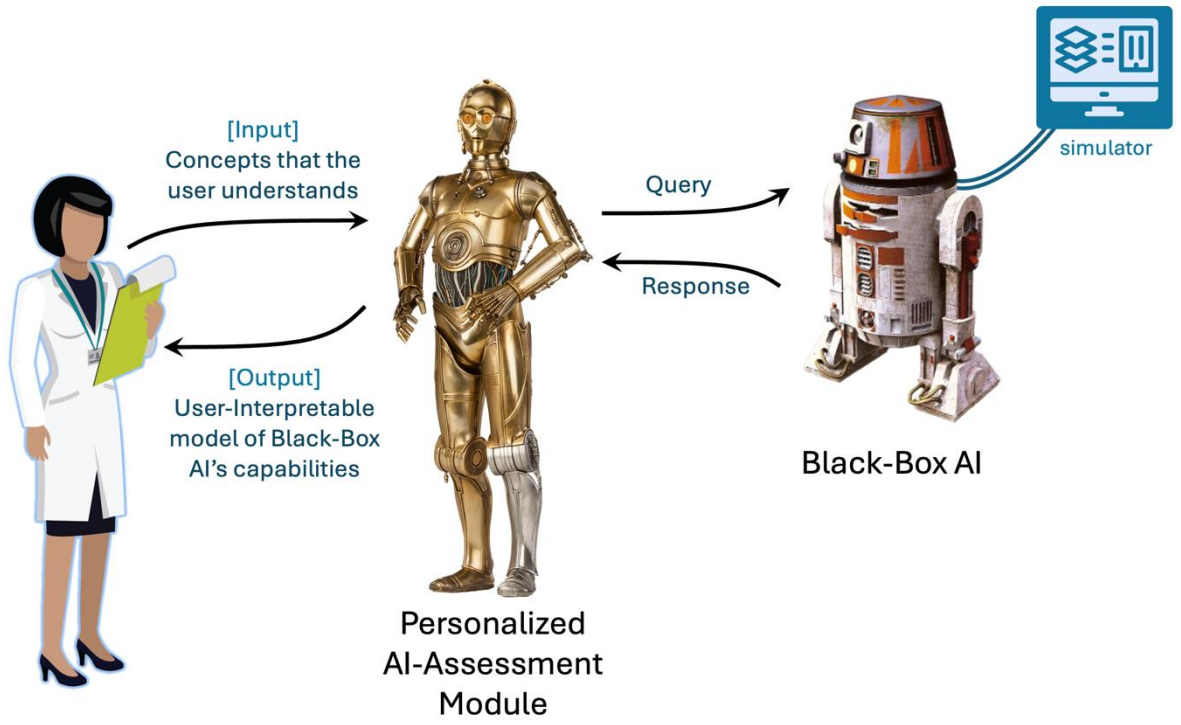
- Extend the work for partially observable settings.



Future Work: Interpretable Representations

- Representations beyond PDDL.
  - LTL – Linear Temporal Logic
  - STL- Signal Temporal Logic

- How interpretable each representation is?



[Input]
Concepts that the user understands

[Output]
User-Interpretable description of Black-Box AI's capabilities

Future Work

Personalized AI-Assessment Module (AAM)

Query

Response

simulator

Black-Box AI

Arbitrary internal implementation

Doesn't know user's vocabulary

[Input]
Concepts that the user understands

[Output]
User-Interpretable description of Black-Box AI's capabilities

Future Work

Personalized AI-Assessment Module (AAM)

Future Work

simulator

Query

Response

Black-Box AI

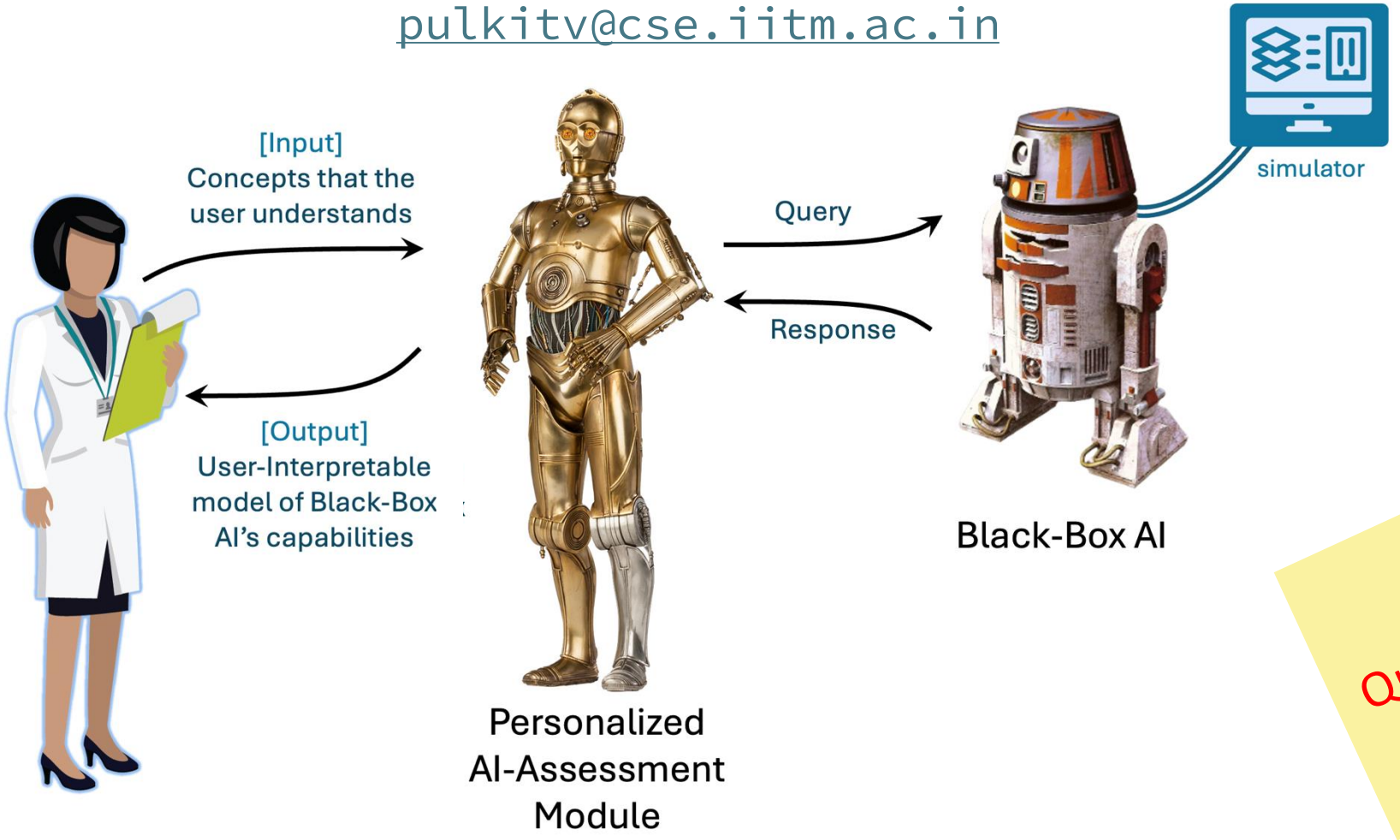
Arbitrary internal implementation

Doesn't know user's vocabulary

Beyond the Lab: User-Aligned Assessment of Taskable AI Systems

Pulkit Verma

pulkitv@cse.iitm.ac.in



Questions?