RASPBERRY SI: Resource Adaptive Software Purpose-Built for Extraordinary Robotic Research Yields

PI: Pooyan Jamshidi, University of South Carolina

https://nasa-raspberry-si.github.io/raspberry-si/

Target: Ocean worlds subsurface---Europa & Enceladus.

Science:

- Enhance autonomy by handling unexpected situations through causal learning
- Increasing the timeliness of scientific explorations by recovering from failures resulting from environmental and system dynamic changes

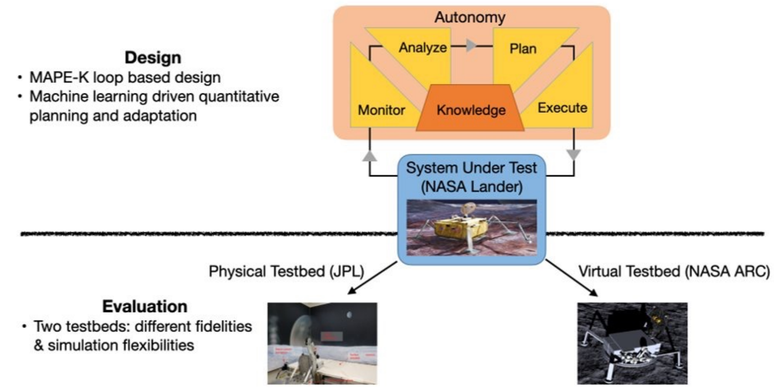
Objectives:

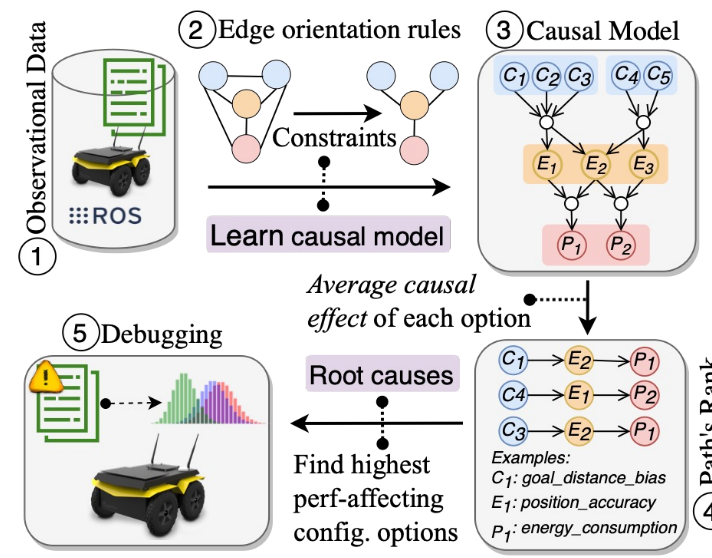
- Develop causal learning models to discover mission intent and enable transfer learning and online learning during a mission.
- Develop a probabilistic planner by integrating causal models to enable run-time adaptation based on the spacecraft's state, environment assumption, and resource availability.

Technology Accomplishments:

- CaRE: Finding Root Causes of Configuration Issues in Highly-Configurable Robots
- CURE: Simulation-Augmented Auto-Tuning in Robotics
- Casual MAPE-K for Fault-Tolerant Autonomy
- Causal MDPs for Probabilistic Model Checking

Autonomy Module





10/01/2020 - 09/30/2022
09/30/2022 - 09/30/2024
(no cost extension)

TRL (2) to (6)

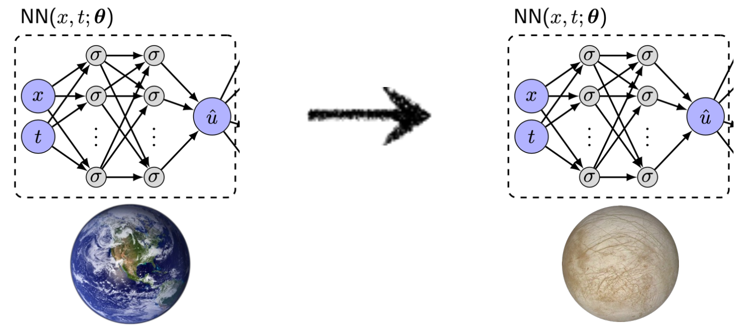
Concepts for Ocean worlds Life Detection Technology (COLDTech)

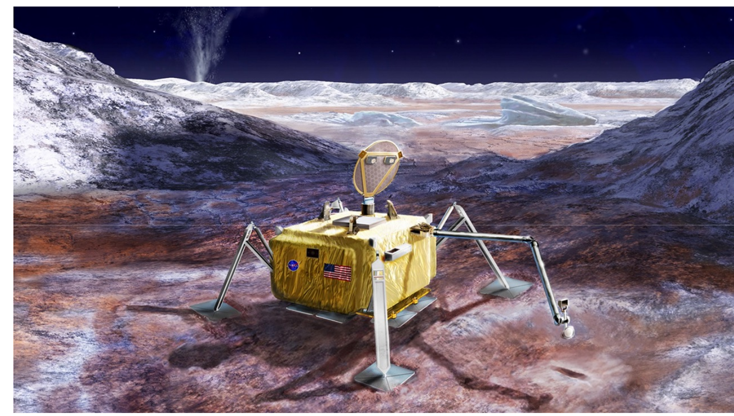
AISR: Autonomous Robotics Research for Ocean Worlds (ARROW)

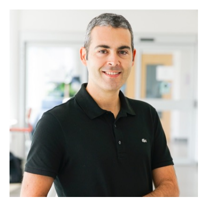
RASPBERRY SI

Resource Adaptive Software Purpose-Built for Extraordinary Robotic Research Yields - Science Instruments

https://nasa-raspberry-si.github.io/raspberry-si





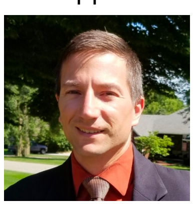


Pooyan Jamshidi
USC
PI

Bradley Schmerl
CMU
Co-I

David Garlan
CMU
Co-I

Javier Camara
York
Collaborator



Matt DeMinico
NASA
Co-I

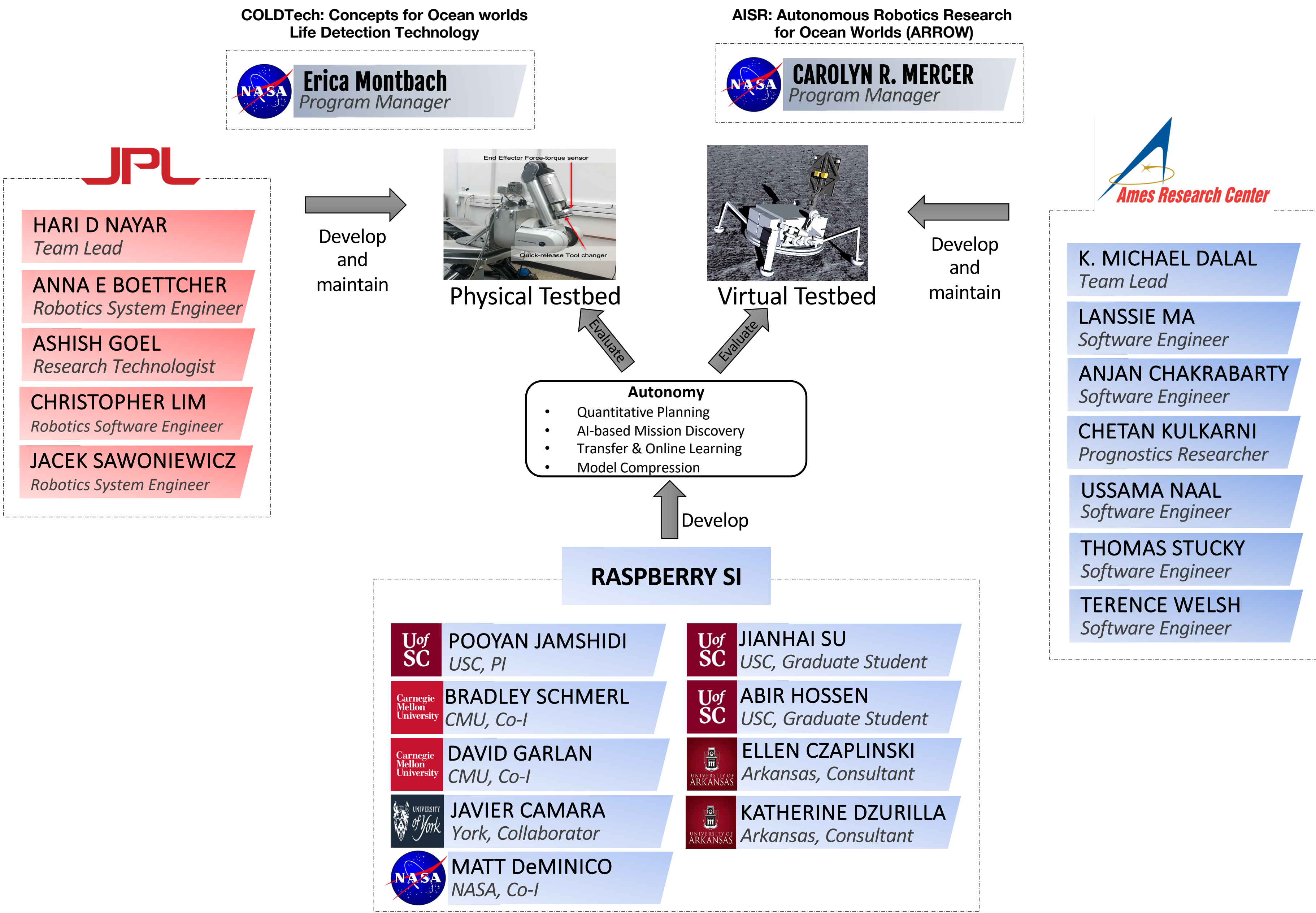
Jianhai Su
USC
Graduate Student

Abir Hossen
USC
Graduate Student

Sonam Kharde
USC
Postdoc

Ellen Czaplinski
Arkansas
Consultant

Katherine Dzurilla
Arkansas
Consultant

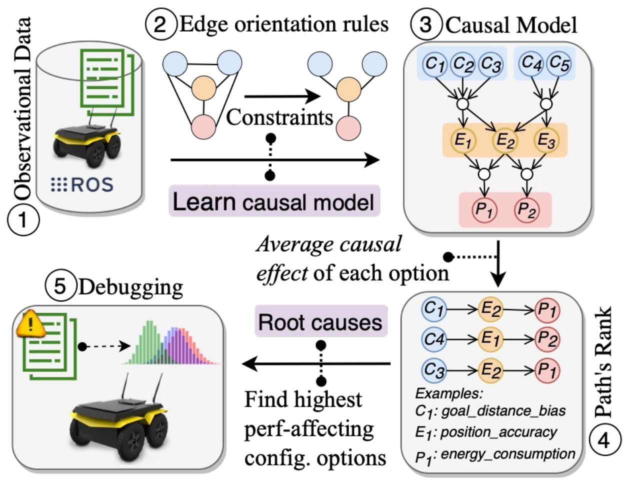

COLDTech: Concepts for Ocean worlds Life Detection Technology
Erica Montbach
Program Manager
AISR: Autonomous Robotics Research for Ocean Worlds (ARROW)
CAROLYN R. MERCER
Program Manager
JPL
HARI D NAYAR
Team Lead
ANNA E BOETTCHER
Robotics System Engineer
ASHISH GOEL
Research Technologist
CHRISTOPHER LIM
Robotics Software Engineer
JACEK SAWONIEWICZ
Robotics System Engineer
Develop and maintain
End Effector Force-torque sensor
Quick-release Tool changer
Physical Testbed
Virtual Testbed
Develop and maintain
Ames Research Center
K. MICHAEL DALAL
Team Lead
LANSSIE MA
Software Engineer
ANJAN CHAKRABARTY
Software Engineer
CHETAN KULKARNI
Prognostics Researcher
USSAMA NAAL
Software Engineer
THOMAS STUCKY
Software Engineer
TERENCE WELSH
Software Engineer
Evaluate
Evaluate
Autonomy
Quantitative Planning
AI-based Mission Discovery
Transfer & Online Learning
Model Compression
Develop
RASPBERRY SI
POOYAN JAMSHIDI
USC, PI
BRADLEY SCHMERL
CMU, Co-I
DAVID GARLAN
CMU, Co-I
JAVIER CAMARA
York, Collaborator
MATT DeMINICO
NASA, Co-I
JIANHAI SU
USC, Graduate Student
ABIR HOSSEN
USC, Graduate Student
ELLEN CZAPLINSKI
Arkansas, Consultant
KATHERINE DZURILLA
Arkansas, Consultant

CARE: Finding Root Causes of Configuration Issues in Highly Configurable Robots

https://github.com/softsys4ai/care

Challenges:

- Non-trivial human efforts due to the large configuration space.
- Performance influence models are non-transferable across different robotic platforms and environments
- Incorrect explanations of root causes in debugging
- Data collection is expensive from physical hardware

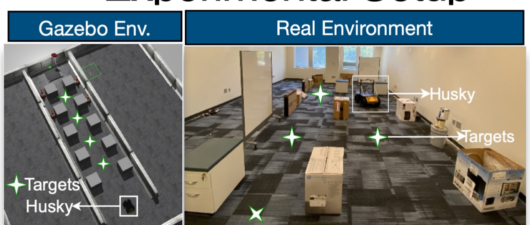


CARE Overview

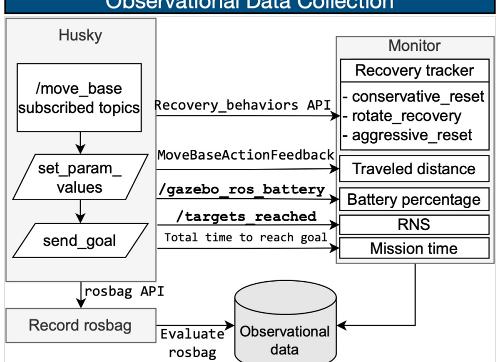
Contributions:

- Proposed a novel framework for finding the root causes of the configuration bugs in robotic systems.
- Demonstrated the transferability of the causal models by learning the causal model in the Husky simulator and reusing it in the Turtlebot 3 physical platform.

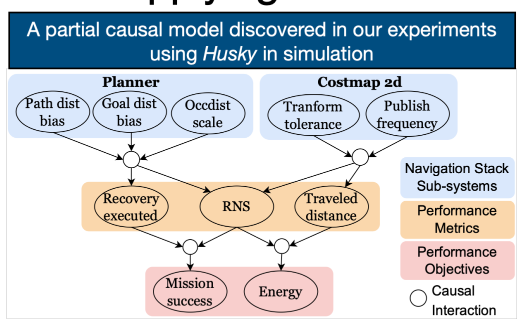
Experimental Setup

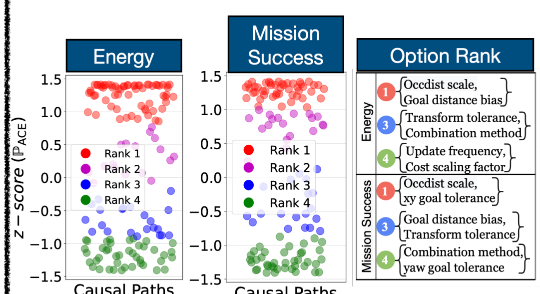


Observational Data Collection

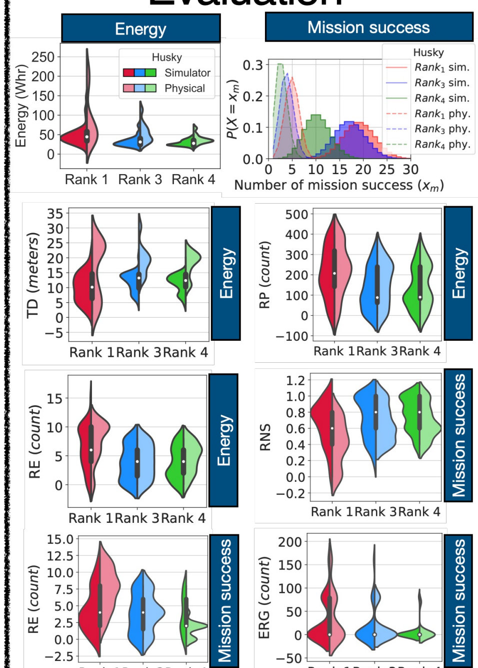


Applying CaRE





Evaluation



CURE: Simulation-Augmented Auto-Tuning in Robotics

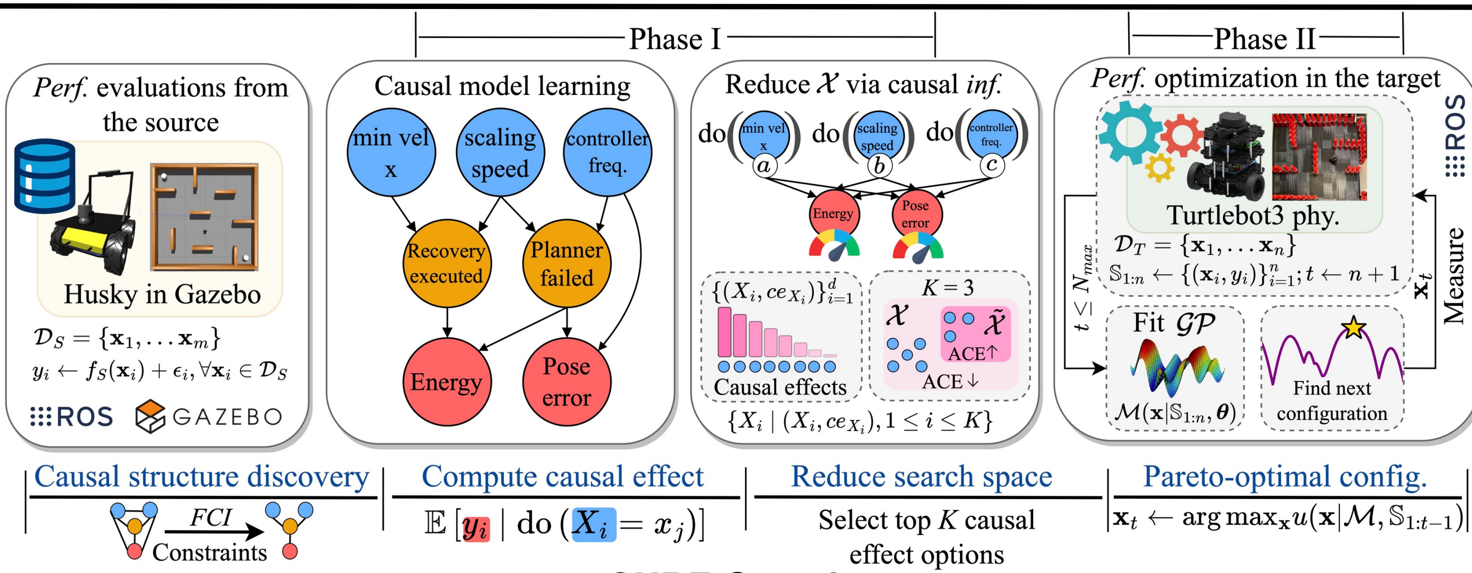
https://github.com/softsys4ai/cure

Challenges:

- Large configuration space
- Conflicting multiple performance objectives
- Costly acquisition of training data
- Safety-critical structure of robotic systems

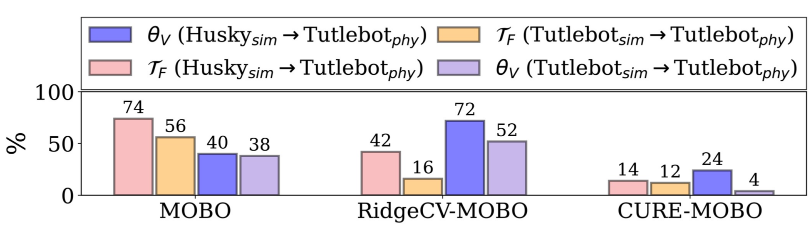
Contributions:

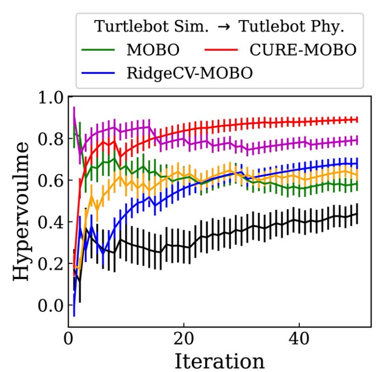
- Proposed a multi-objective optimization method that operates in the reduced search space involving causally relevant configuration options and allows faster convergence.
- Demonstrated transferability on *Husky* and *Turtlebot 3 platforms,* both in simulation and real robots, with different severity of deployment changes

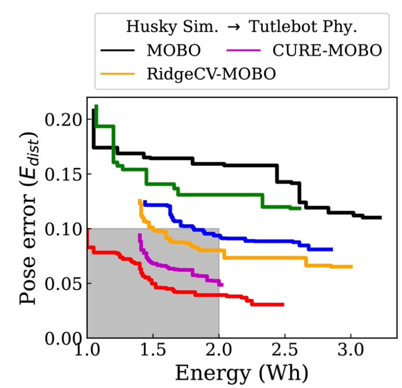


CURE Overview

Evaluation and Results:







PESTO Planetary Exploration Science Technology Office

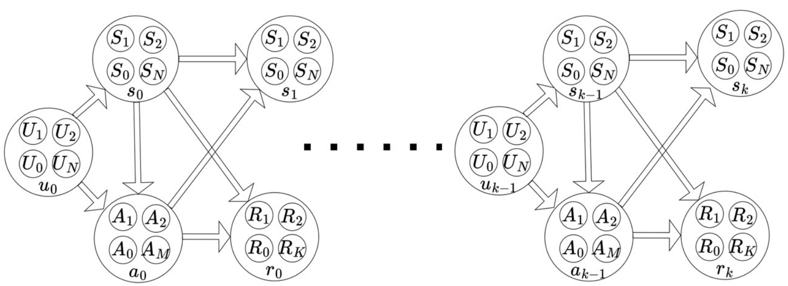
From Roots to Probabilities:
A Causal Exploration in the Probabilistic Model Checking

Challenges:

- Model uncertainty
- Complex reward structures
- State space explosion
- Computational complexity

Contributions:

- Causal-MDP as a modeling framework in probabilistic model checking
- Reward and transition probabilities computation considering causal dependencies, enabling the use of system dynamics for robust predictions
- Efficient and scalable verification procedures, ensuring property satisfaction and reducing computational complexities.



Causal evolution for 'k' step CIMDP

Transition Probability:

$$\mathcal{P}(s'|s,a) = \frac{\mathbb{E}_{u\sim\tilde{P}}[P(s'|s,a,u)\cdot\pi(a|s,u)]}{\mathbb{E}_{u\sim\tilde{P}}[\pi(a|s,u)]}$$

$$P(s'|s,a,u) = \prod_i \tilde{P}_i \forall i \in V \setminus S \cup A \cup R$$

Rewards:

$$R(s'|s,a) = \frac{\mathbb{E}_{u\sim\tilde{P}}[R(s'|s,a,u)\cdot\pi(a|s,u)]}{\mathbb{E}_{u\sim\tilde{P}}[\pi(a|s,u)]}$$

$$R(s'|s,a,u) = \mathbb{E}_{\tilde{P}}[R(s,a,u)]$$

Computational Complexity:	Model size	Verification
	$\mathcal{O}(\lvert\mathcal{S}\rvert \sum_{i=0}^{\lvert S\rvert-1} \lvert\mathcal{A}(s_i)\rvert)$.	Linear with model size

Evaluations and Results:

Table: Statistical Analysis for numerical and verification queries for autonomous vehicle

Property	Result	Parameter	MDP	CIMDP	MDP	CIMDP	MDP	CIMDP
NQ_1: $Pr_s^{max}(reach(\texttt{goal}))\ \forall s \in \mathcal{S}$	0.909	State Space Size	2160	524	1584	396	1008	252
		Reachability iterations	16	16	12	12	9	8
NQ_2: $Pr_s^{min}(reach(\texttt{goal}))\ \forall s \in \mathcal{S}$	0.019	Model construction time	28.589	1.859	14.053	0.895	5.043	0.388
		States	115	75	83	55	51	35
VQ_3:$Pr_{\geq 0.8}[F^{\leq 10}\texttt{goal}]$	False	Transitions	390	316	278	228	166	150
		Choices	197	117	144	85	85	53
VQ_4: $R'^{ec'}_{\leq 500}[C^{\leq 8}]$	True	Nonzero Transition Rewards	386	314	274	228	162	138
NQ_5: $R'^{ec'}_{min=?}[F\ (\texttt{goal})]$	413.1285	No. of Nonzero value states	110	72	78	52	46	32
NQ_6:$R'^{ec'}_{max=?}[C \leq 5]$	364.0112	Verification time (seconds)	0.01	0.007	0.009	0.005	0.007	0.003
NQ_7:$multi(R'^{ec'}_{min=?}[C], R'^{col'}_{max=?}[C])$	[(283.467, 0.0)]	Reward computation time	0.138	0.05	0.096	0.04	0.04	0.028

PESTO
Planetary Exploration Science Technology Office

Fault Tolerant Autonomy Design for Ocean World Lander System

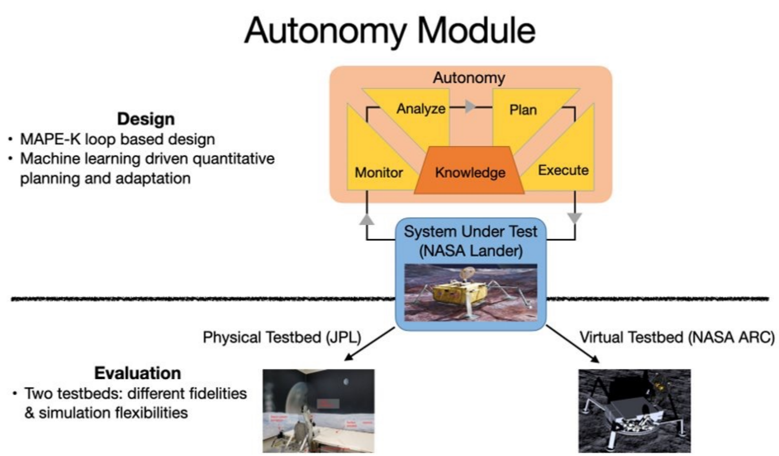
https://nasa-raspberry-si.github.io/raspberry-si

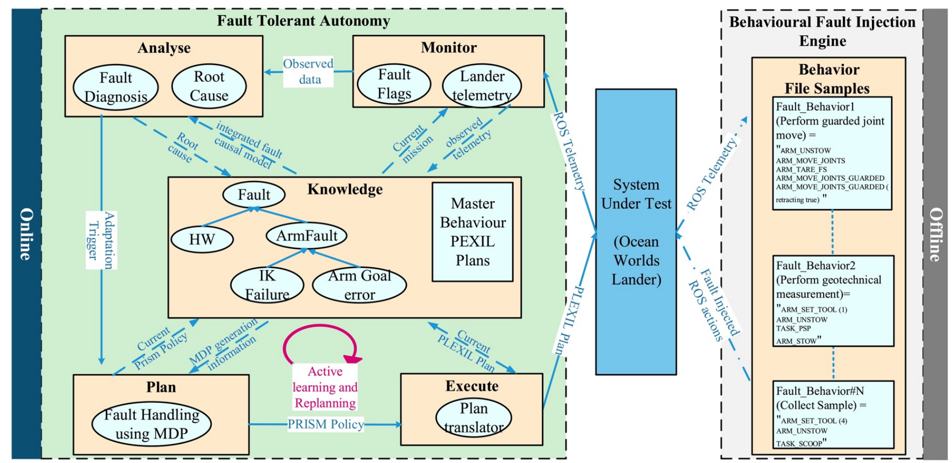
Challenges:

- Limited arm workspace
- Unexpected surface behavior
- Communication delays
- Battery constraints

Contributions:

- Autonomy design using the MDP framework
- Causal MAPE-K for fault-tolerant autonomy
- Search space reduction for planner
- Capability demonstration on physical and virtual testbeds





Evaluations and Results:

Evaluation Criteria:

Stage	Description
Stage A (Baseline): PLEXIL plan, No-Fault, No Autonomy	Test connection between autonomy and testbed
Stage B (Baseline): PLEXIL plan, Fault, No Autonomy	Show need for autonomy
Stage C.A (Challenge Stage - Design time)	Autonomy in Naive mode with design time fixes
Stage C.B (Challenge Stage - Runtime)	Autonomy with synthesized adaptation plan

PESTO
Planetary Exploration Science Technology Office