Silent Data Corruption in Robot Operating Systems (ROS):

A Case for End-to-End System-level Fault Analysis Using UAVs

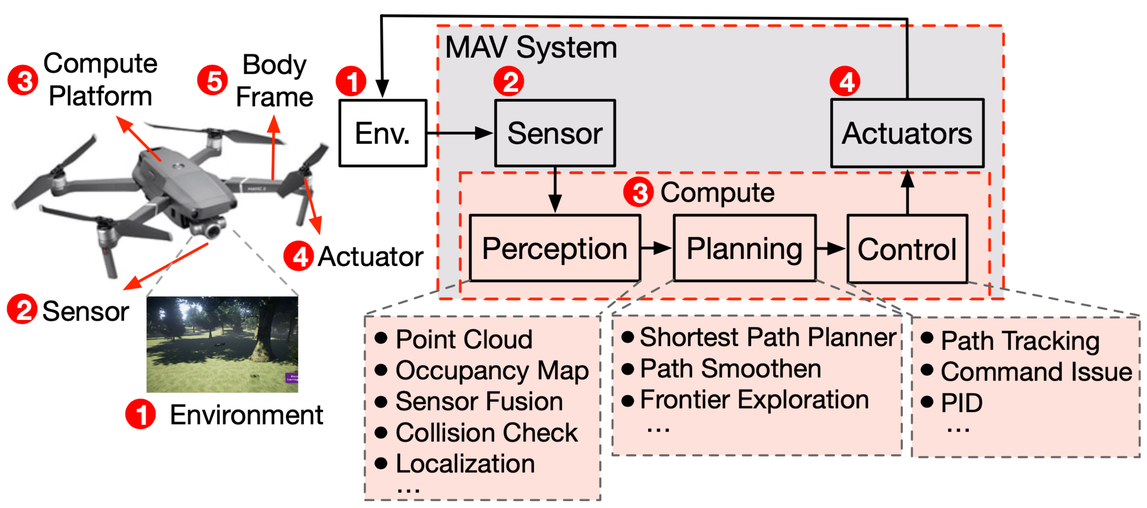
Yu-Shun Hsiao[1*], Zishen Wan[2*], Tianyu Jia[3], Radhika Ghosal[1], Abdulrahman Mahmoud[1], Arijit Raychowdhury[2], David Brooks[1], Gu-Yeon Wei[1], Vijay Janapa Reddi[1]

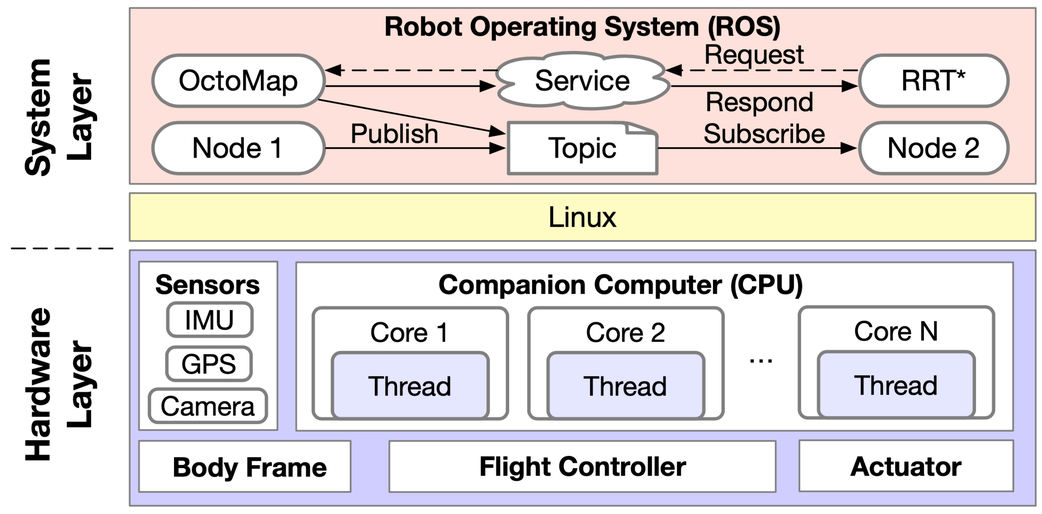
[1]Harvard University, MA [2]Georgia Institute of Technology, GA [3]Peking University, China

*Equal Contributions, listed in alphabetical order

Silent Data Corruption In Autonomous Systems

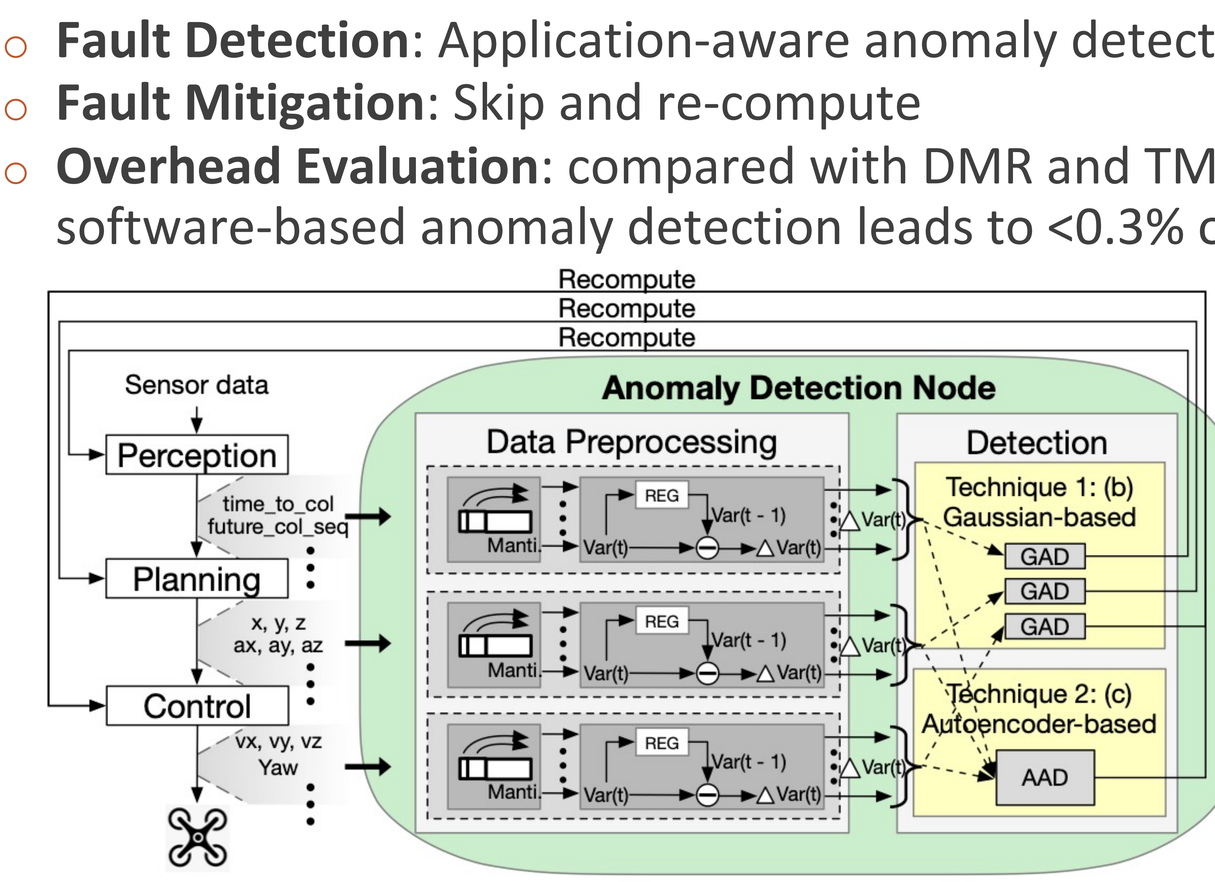
- **Motivation**: Silent Data Corruption (SDC) has shown a significant threat in computing, from server scale systems to emerging application areas. Safety and reliability of autonomous systems is critical.
- **Challenge**: No suitable fault analysis tool; Autonomous machines are complex cyber-physical systems.
- **This work**: What is SDC impact on end-to-end system-level autonomy metrics for autonomous aerial robots? How to enhance the resilience of autonomous system against SDC with lightweight techniques?





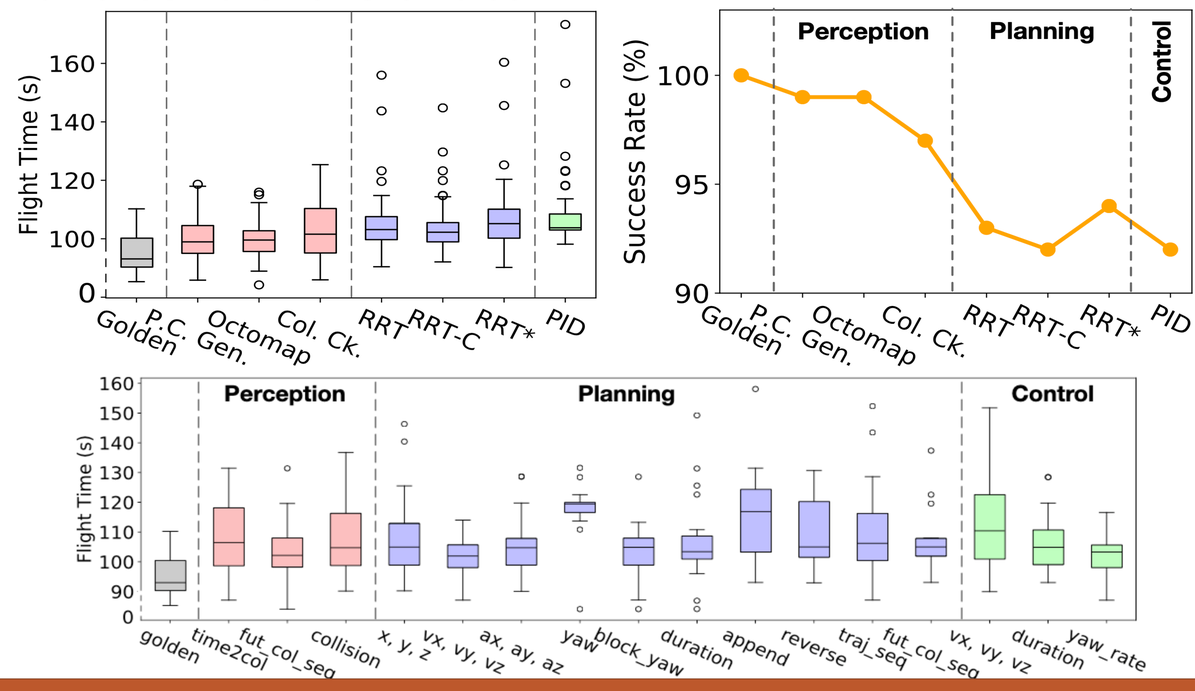
Fault Analysis Framework

- **Fault Injection**: Hardware transient faults during compute, portable to any ROS-based systems.



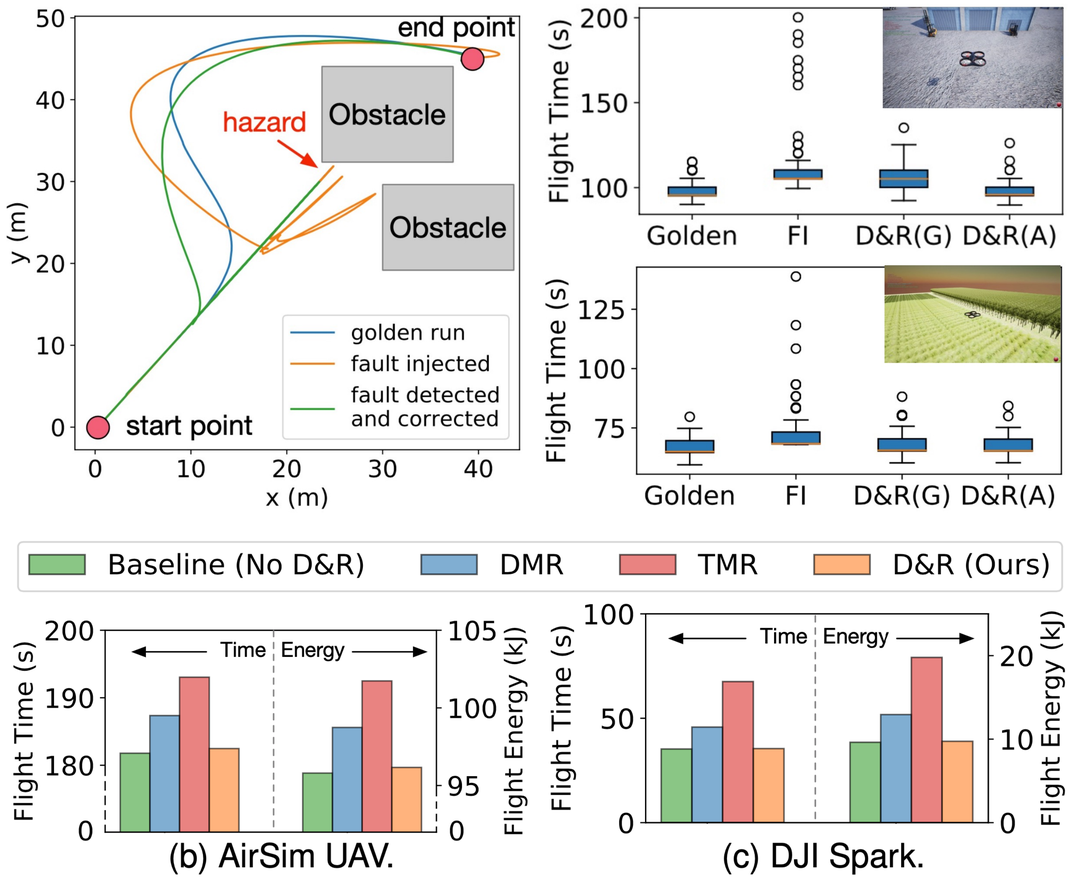
Fault Characterization

- **Fault Characterization**: (1) Metric: quality-of-flight; (2) Eva: end-to-end analysis; (3) Takeaway: planning and control are critical.



Fault Detection and Mitigation

- **Fault Detection**: Application-aware anomaly detection
- **Fault Mitigation**: Skip and re-compute
- **Overhead Evaluation**: compared with DMR and TMR, software-based anomaly detection leads to <0.3% overhead



Paper

Code

Acknowledgments

This work was supported in part by C-BRIC and ADA, two of six centers in JUMP, a Semiconductor Research Corporation (SRC) program sponsored by DARPA