

Automated Extraction and Checking of Property Models from Source Code for Robot Swarms

E. Merlo, C. Pinciroli, J. Panerati, M. Famelis, **G. Beltrame**

May 9, 2022

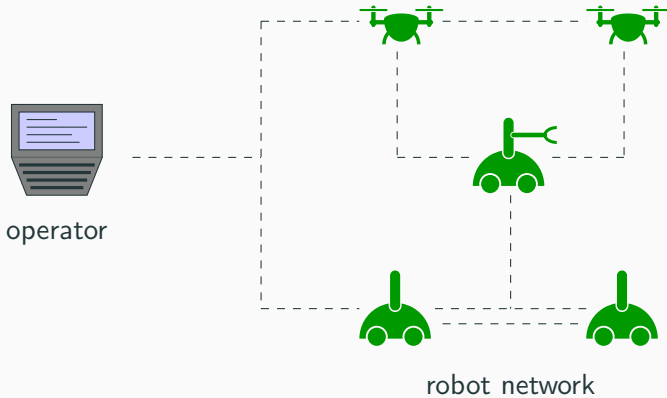
Buzz: a DSL for robot swarms

- Simple, dynamically-typed language
- Open source compiler and runtime

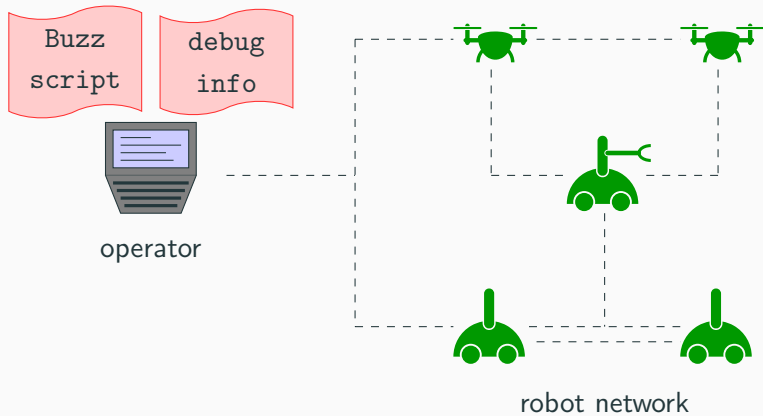
Main goal

- Automatically extract static property models using Pattern Traversal Flow Analysis (PTFA)
- Models fed to a model checker to verify safety properties

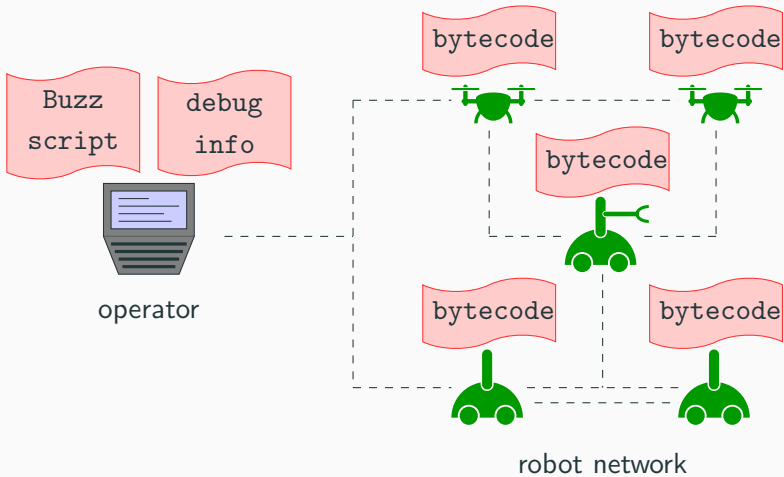
Buzz Deployment



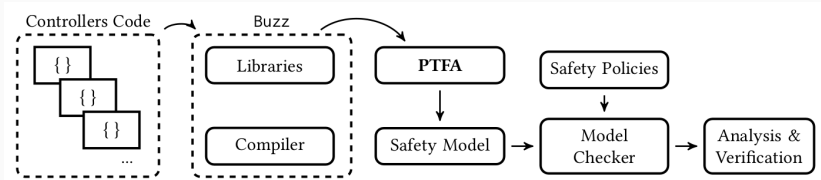
Buzz Deployment



Buzz Deployment



Overall architecture



- Possible manual specification of functions to be tracked
- The physical properties associated with domain specific functions tracked in the model
 - We track all functions in our experiments
- Property constraints are defined by developers (domain knowledge)
- Language-dependent front-end, language-independent from PTFA to model verification
- All automated!

Performance

	ROSBuzz	Swarm Relays
Size (LOCs)	4007	3490
Parsing + CFG (s)	26	34
Reachability analysis + model	53	71

Extracted model

	ROSBuzz	Swarm Relays
nodes	96	128
edges	202	266

Sample model: ROSBuzz



- Allows verification of constraints
- Satisfaction of a forbidden predicate → violation
- Non-satisfaction of a property that must hold → violation
- E.g. taking-off without verifying the GPS location of the robot
- Tools
 - Alloy (slower)
 - Z3 (faster)

Z3 examples

```
ASSERT: (not (=> barrier_set_end barrier_ready_begin))  
unsat :total-time 0.02)
```

```
ASSERT: (not (=> navigate_begin pathPlanner_begin))  
unsat :total-time 0.01)
```

Conclusions

- Novel model extraction method of “as-implemented”
- Analysis of programs using Buzz
- Experiments on 2 medium-size systems for robotics research (ROSBuzz and SwarmRelays)
- Model checking of swarm robotics properties on test pairs from ROSBuzz and SwarmRelays developers
- Results manually validated, suggest that the approach is feasible and scalable