NEURAL INFORMATION PROCESSING SYSTEMS

PVSGym: A Proof Learning Environment

Manoj Acharya, Karthik Nukala, Natarajan Shankar (SRI)

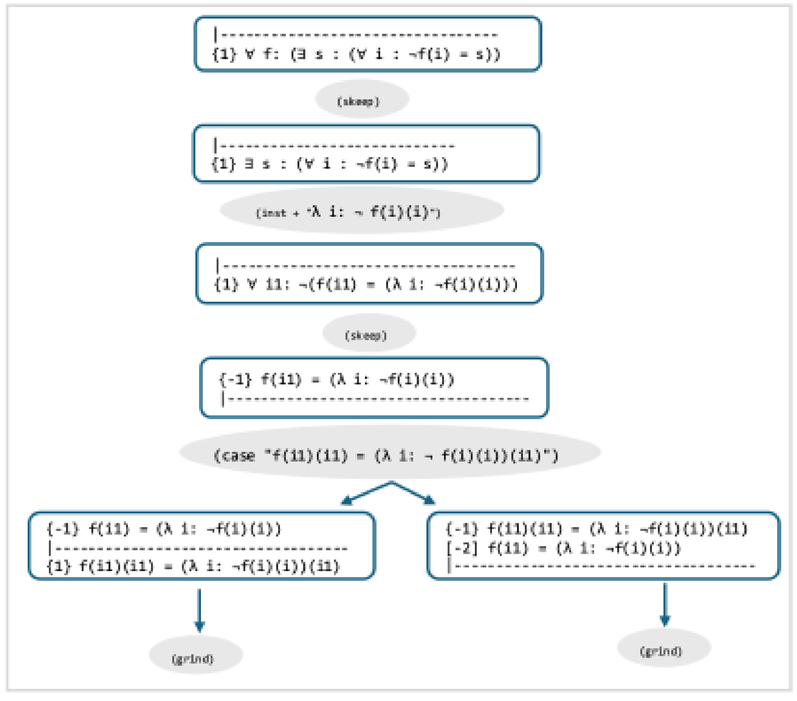
SRI

Abstract

PVSGym is a reinforcement learning environment for the PVS theorem prover. It includes an ML-friendly dataset of expert-level formalizations from NASALib, spanning mathematics, computer science, and aerospace domains. PVSGym bridges AI and formal methods, enabling research on automated proof synthesis.

Introduction

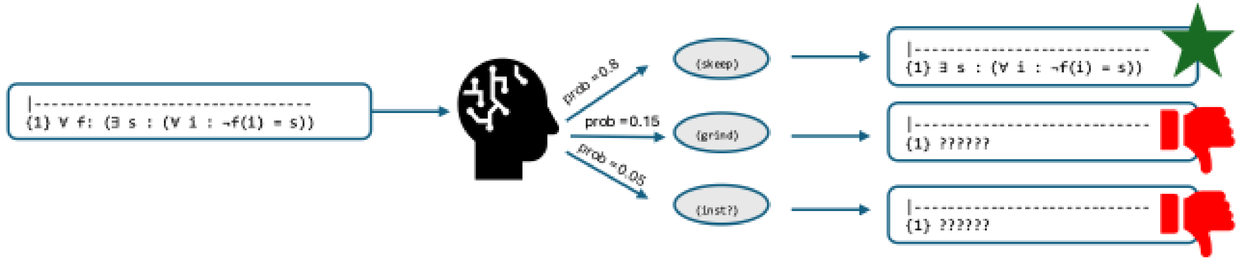
Large language models enable soft reasoning but lack formal rigor. PVSGym integrates symbolic reasoning within a reinforcement learning framework, where agents interact with the PVS theorem prover to learn command-level proof strategies.



PVSGym Environment

PVSGym provides a JSON-RPC API for theorem proving, allowing agents to send proof commands, receive sequent, rewards, and updated proof states. The environment supports interactive and high-throughput training for proof search.

```python
async def run_agent(PORT, file_path, formula_name):
  async with PVSExecutor("localhost", str(PORT)) as env:
    sequents, info = await env.reset(file_path, formula_name)
    print(f"Goal: {sequents}")
    done = False
    while not done:
      ## recieve command from the agent
      state, reward, terminated, truncated, info = await env.step(command)
      done = terminated or truncated
```



NASALib Dataset

NASALib spans 62 libraries with over 38K theorems covering mathematics, analysis, logic, and safety-critical verification. PVSGym extracts 21K+ proof rollouts including command sequences, intermediate sequent, and rewards.

```
sum: THEORY
 BEGIN

 n: VAR nat

 sum(n): RECURSIVE nat =
  (IF n = 0
   THEN 0
   ELSE n + sum(n - 1)
   ENDIF)
  MEASURE n

 closed_form: THEOREM
    sum(n) = (n * (n + 1))/2

 END sum
```

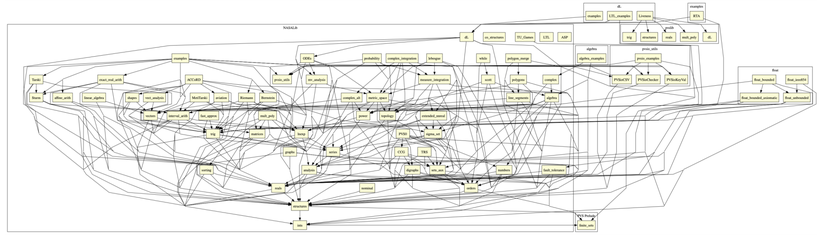
2.1 PVS specification file (`.pvs`) with formula declarations and theorem statements.

```
(induct-and-simplify "n")
```

2.2 A combined PVS proof command applying induction on n and applying aggressive simplification.

```
closed_form :
|-------
{1}  FORALL (n: nat):
     sum(n) = (n * (n + 1)) / 2
```

2.3 PVS sequent on which proof commands will be applied.



Results

Method	cmdhist + sequent	sequent	cmdhist
K-NN (Yeh et al., 2023)	0.28	0.19	0.27
CoProver* (Yeh et al., 2023)	0.48	0.28	0.21
Ours	**0.49**	**0.47**	**0.42**

Command prediction accuracies across input configurations.

Conclusion

PVSGym is the first such environment for PVS. Released with NASALib rollouts, baselines, and scripts, it provides a foundation for future AI-assisted formal reasoning and mathematical discovery.