

BFTGym: An Interactive Playground for BFT Protocols

Haoyun Qin, Chenyuan Wu, Mohammad Javad Amiri, Ryan Marcus, Boon Thau Loo

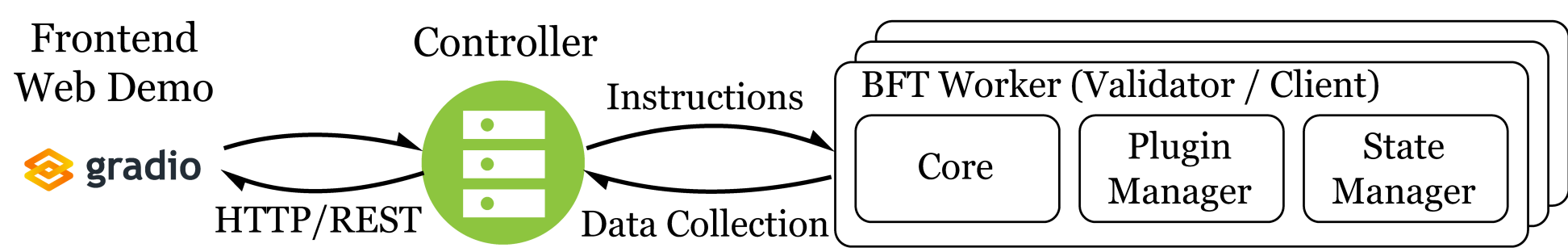
{qhy, wucy, rcmarcus, boonloo}@seas.upenn.edu, amiri@cs.stonybrook.edu

University of Pennsylvania, Stony Brook University

Supported by NSF grants CNS-2104882 and CNS-2107147

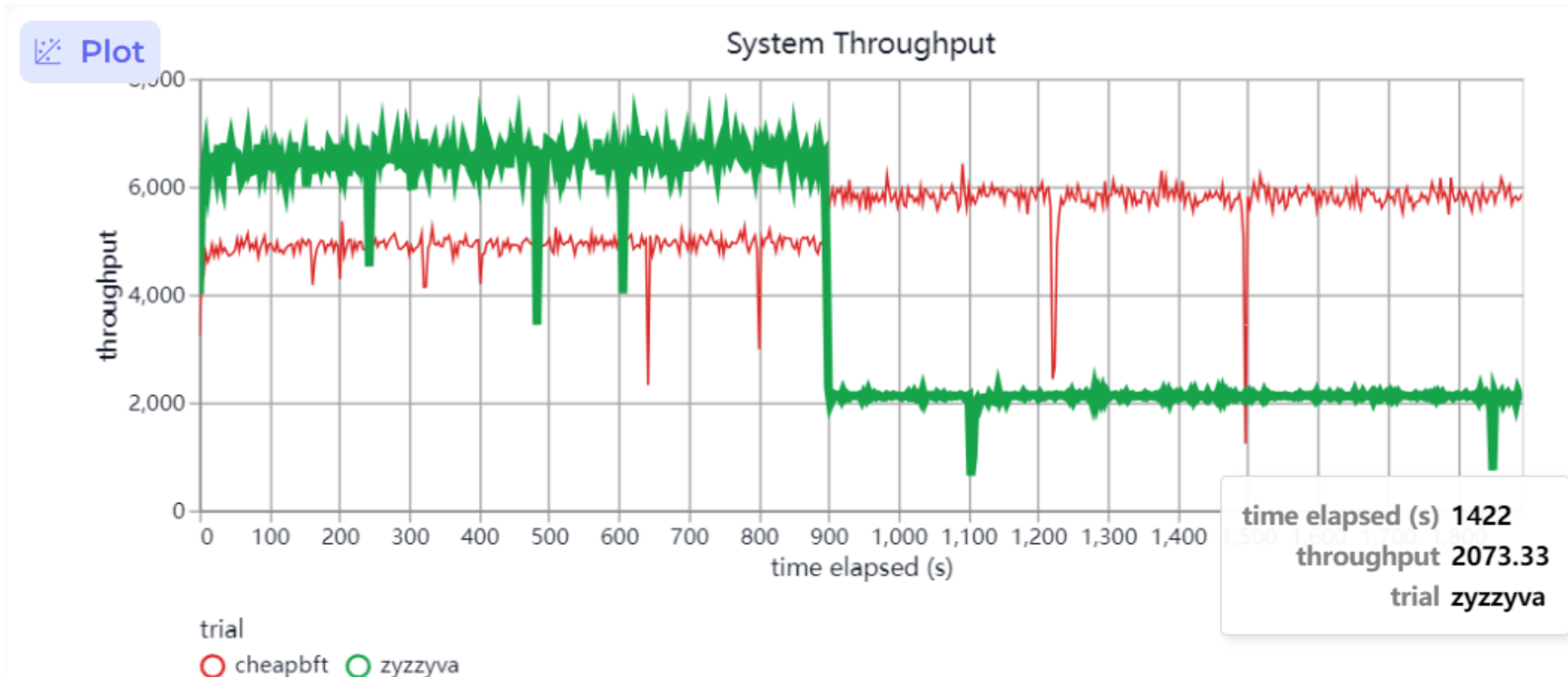


System Architecture

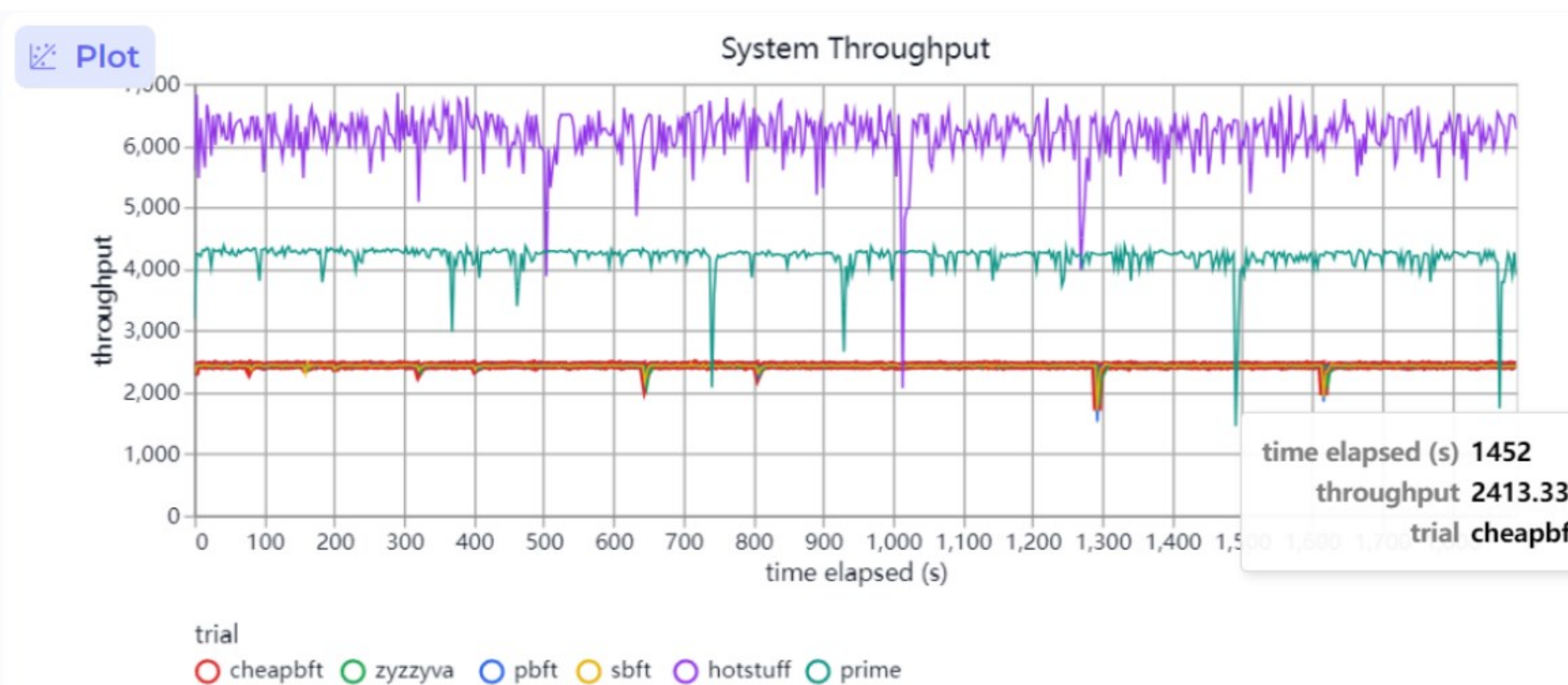


- An **interactive playground** for evaluating, comparing and gathering insights for various BFT protocols under a **unified framework** and a **wide range of conditions**;
- Support **workload adjustment** and **fault injection** with **immediate performance feedback**;
- Enable rapid **prototyping** for new BFT protocols through **Bedrock Domain Specific Language (DSL)**

Evaluation

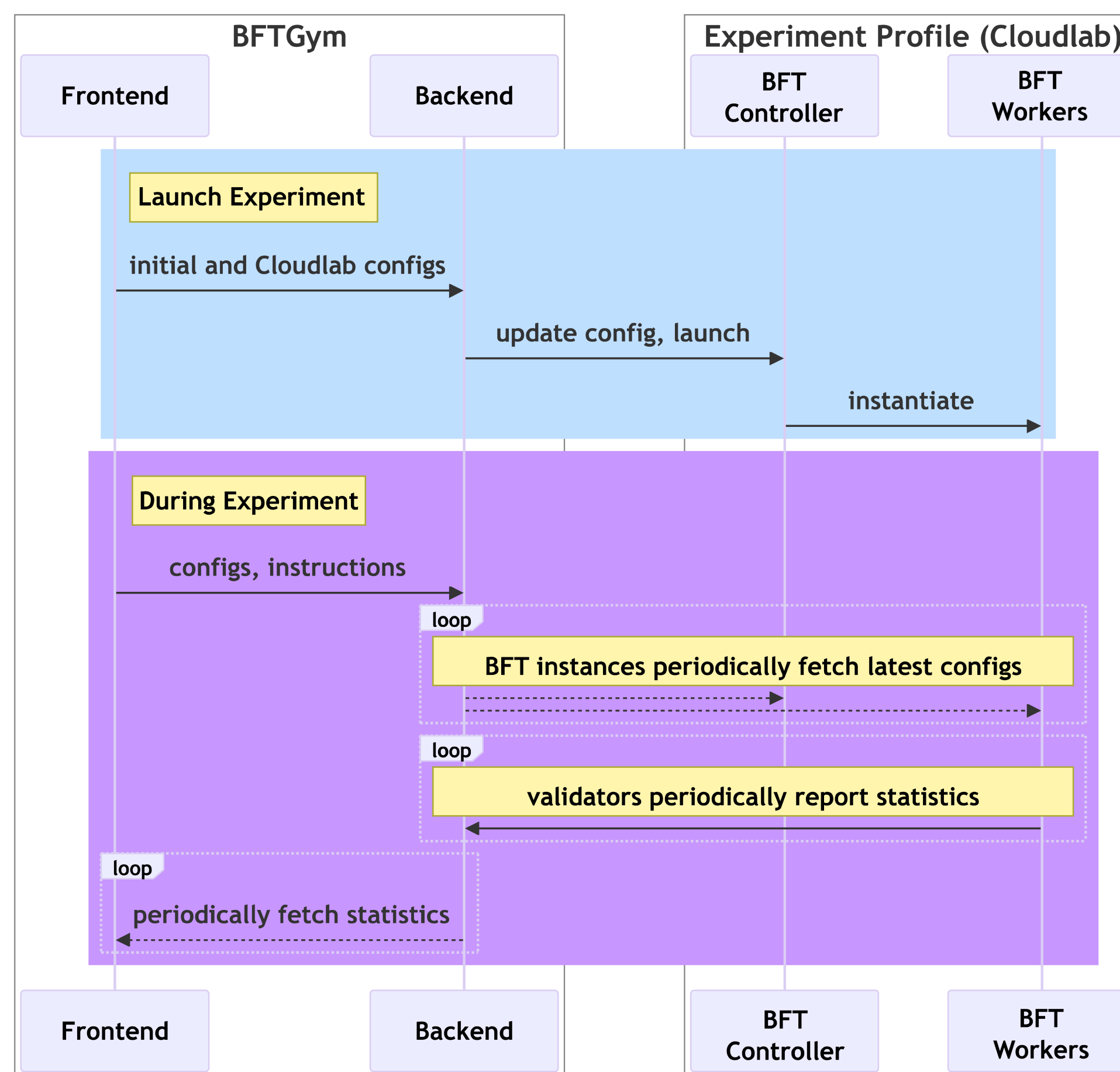


Comparing speculative and non-speculative protocol under 0/0 standard workload with non-responsive fault launched at the middle



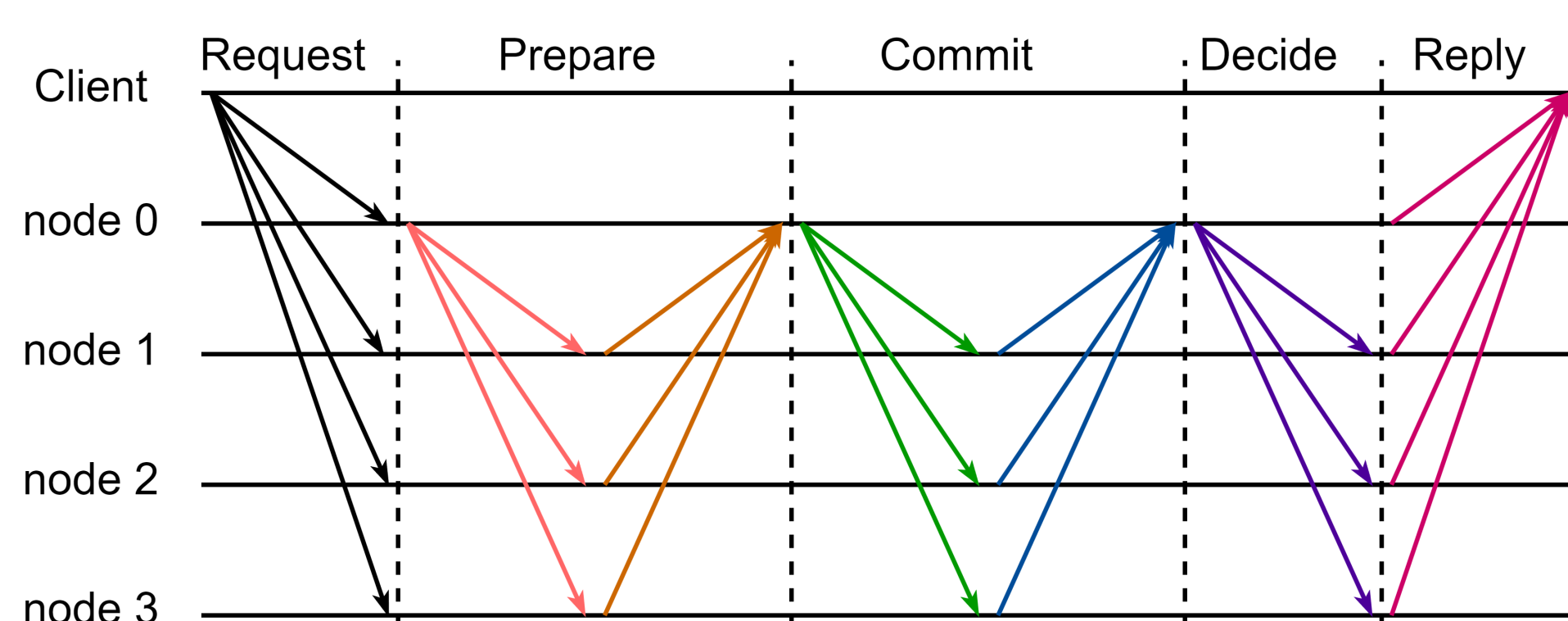
Throughput comparison of 5 protocols under 1/0 standard workload with 20 millisecond slowness fault

Demo Setup



Protocol Prototyping

- BFTGym can enable fast protocol prototyping. Code snippet on the right shows the Bedrock DSL modeling Hotstuff-2, whose communication diagram is shown below.
- Protocols are first-class citizens in BFTGym. Custom protocols can be seamlessly integrated into the experimental workflow, benefiting from the same functionalities as other built-in ones.



```

# protocol properties
general:
  leader: rotate
  request-target: primary
# roles involved
roles:
  - primary
  - nodes
  - client
# phase definitions
phases:
  - name: normal
    states:
      - idle
      - wait_prepare
      - wait_commit_primary
      - wait_commit_all
      - wait_decide
      - executed
    messages:
      - name: request
        request-block: true
      - name: reply
        request-block: true
      - name: prepare
        request-block: true
      - commit
      - decide
  - name: checkpoint
    messages:
      - checkpoint
# data flow
transitions:
  from:
    - role: client
      state: idle
      to:
        - state: executed
          update: sequence
          condition:
            type: message
            message: commit
            quorum: 2f + 1
          response:
            - target: nodes
              message: decide
            - target: client
              message: reply
  - role: primary
    state: idle
    to:
      - state: wait_prepare
        condition:
          type: message
          message: request
          quorum: 1
        response:
          - target: nodes
            state: wait_decide
            to:
              - state: executed
                update: sequence
                condition:
                  type: message
                  message: decide
                  quorum: 1
                response:
                  - target: client
                    message: reply
  - role: nodes
    state: idle
    to:
      - state: wait_commit_all
        condition:
          type: message
          message: prepare
          quorum: 1
        response:
          - target: primary
            message: prepare
      - state: wait_commit_primary
        condition:
          type: message
          message: prepare
          quorum: 2f + 1
        response:
          - target: nodes
            message: commit
  - role: nodes
    state: wait_decide
    to:
      - state: wait_commit_all
        condition:
          type: message
          message: commit
          quorum: 1
        response:
          - target: primary
            message: commit
  - role: nodes
    state: wait_commit_primary
    to:
      - state: executed
        update: sequence
        condition:
          type: message
          message: commit
          quorum: 2f + 1
        response:
          - target: nodes
            message: decide
          - target: client
            message: reply
    
```

Demo Interface

BFTGym: An Interactive Playground for BFT Protocols

Step 1 - Configurations

- Fault:** Non-responsive Nodes (IDs separated by ",")
- In-dark - enabled:** True / False
- Slow proposal Nodes (IDs separated by ","):** 0
- Slowness of proposal in milliseconds:** 0
- Workload:** C.2
- Update:** E
- Units Status:** F
 - Controller: Updated
 - Replica 0: Updated
 - Replica 1: Updated
 - Replica 2: Updated
 - Replica 3: Updated
 - Client: Updated
- Cloudlab Configurations:** C.3

Step 2 - Control Panel

- Trial Name:** default
- Protocol:** pbft
- Custom Protocol:** Only applicable when protocol selected to custom
- Start** / **Stop**

Results - Plots

- G.1:** System Throughput plot showing throughput over time.
- G.2:** Last Committed Sequence Number plot showing sequence number over time.

Use via API • Built with Gradio