CSE535: Distributed Systems

Project: DiemBFT v4 Consensus Algorithm Phase 4

Team Name: Loyal Byzantine Generals

Manikanta Sathwik Yeluri

Preetham Reddy Katta

Sai Bhavana Ambati

User Manual

- **DistAlgo Installation**
  - Follow https://github.com/DistAlgo/distalgo/ to see the steps to install DistAlgo in the system.
- **Known system-specific issues**
  - For macOS, If the message size exceeds the threshold specified by the system for UDP, the system will throw an error *"OSError(40, 'Message too long') ".*

`Execute sudo sysctl -w net.inet.udp.maxdgram=65535 to increase the limit.`

- **Configuration and output for diembft simulation**

All the test cases follow the same format for the config file; ledger files for validators; and log files for all validators as well as clients mentioned in the following format.

Config File:

'*config/config.da'* contains a list of all configurations to be executed.

Ledger File:

For each test case *'$t'* in each configuration at index *'$c'* mentioned in the configuration file, each validator with index *'$v'* creates its own ledger file under '*ledgers/config$c/test_case_'$t'/ validator_$v.ledger'*

Log File:

For each test case *'$t' in* each configuration at index *'$c'* mentioned in the configuration file, each validator with index *'$v'* creates its own log file under '*logs/config$c/test_case_'$t'/ validator_$v.log'.*

For each test case *'$t' in* each configuration at index *'$c'* mentioned in the configuration file, each client with index *'$r'* creates its own log file under '*logs/config$c/test_case_'$t'/ client_$r.log'*

Example of configuration file located at *"/config/config.da"*

Keep the import statements and only modify the existing configs list.

Each element of the configs list is the configuration for the simulation.

The simulation runs all configurations mentioned in the configs list in a single execution

```
from object_types import FailType, Failure, FailureConfig, MsgType

configs = [ {
    'nvalidators': 7,
    'nTwins': 2,
    'nclients': 3,
    'nclientops': 2,
    'sleeptime': 1,
    'clienttimeout': 4,
    'delta': 0.25,
    'window_size': 5,
    'exclude_size': 0,
    'delay': 1,
    'quorum_bug': False, #quorum = 2f
    'accept_conflicting_votes':False,
    'liveness_bound' : 10,
    'n_test_cases' : 5
```

```
}]
```

Figure 1. Example of configuration file located at *"/config/config.da"*

● Explanation of each label present in the configuration file

```
'nvalidators': Number of Validators/Replicas,
'nTwins': Number of Twin Validators,
'nclients': Number of Clients,
'nclientops': Number of operations each client performs,
'sleeptime': Delay between consecutive operations for the same client in
seconds,
'clienttimeout': Amount of time the client waits in seconds to receive
the response. If no response is received, it retransmits that request
'delta': Amount of time in seconds used to decide the pacemaker timer
timeout time,
'window_size': Window size used for Leader Election,
'exclude_size': Exclude size used for Leader Election,
'delay': Delay time when a fail type occurs
'quorum_bug': Change the quorum from 2f+1 to 2f when set to true
'accept_conflicting_votes': Vote for multiple proposal messages in a round
'liveness_bound': test case run duration bound
'n_test_cases': number of test cases to run for a particular config
```

Figure 2. Explanation of each label present in the configuration file

● Commands to execute in sequence

- ○ cd <path_of_project_folder>/src
- ○ python3 -m da --message-buffer-size 65535 run_diembft.da

● Configuration and output for twins generator

Config File:

'src*/config_test.py'* contains a list of all configurations to be executed.

```
configs = [ {
        'n_replicas' : 5,
        'n_twins' : 1,
        'n_rounds' : 5,
        'n_partitions' : 2,
        'is_leader_faulty' : True,
        'partition_num_limit' : 100,
        'n_test_cases' : 5,
        'leader_partitions_num_limit' : 100,
        'random_seed' : 1234567,
        'is_deterministic' : False}
]
```

Example of configuration file located at *"*'src*/config_test.py'"*

● Explanation of each label present in the configuration file

n_replicas : Total number of actual replicas
n_twins : Total number of twins
n_rounds : Total number of rounds
n_partitions : Total number of partitions in each round

is_leader_faulty : If the leader of the round can be one of the twins or not
partition_num_limit : Maximum number of partition combinations
n_test_cases : Total number of test cases
leader_partitions_num_limit : Maximum number of leader-partition combinations
random_seed : seed for generation of random numbers
is_deterministic : whether limiting of partition_combinations and partition_leader_combinations should be done deterministically or randomly.

- Commands to execute in sequence
    - cd <path_of_project_folder>/src
    - python3 generator.py