Easysim

A simulator for distributed algorithms

Vivien Quéma

CNRS (LSR laboratory)

Vivien.Quema@inrialpes.fr

Outline

1 Introduction

2 Operating principle

3 Examples

Introduction

- Round-based simulator for distributed algorithms
  - Empirically check the behavior of an algorithm
  - Compute the latency
  - Compute the throughut
- Written in Java 5
- Modification of the Peersim simulator

Outline

Nodes

- Each node is represented by an object of a class extending the `Node<T>` class
- A node
  - Implements the `cycleHandler` method
  - Has an array `neighbors`
  - Is able to
    - Send message calling the `send` method
    - Receive a message calling the `receive` method
- At each round, the simulator *sequentially* executes all the nodes

Controllers

- It is possible to define controllers
- A controller is an object that implements the `Control` interface
  - Define the `execute` method
- Controllers are *sequentially* executed at the end of each round
- Controllers can be used to
  - Inject load (e.g. deciding which node will send messages at the start of the following round)
  - Compute statistics (e.g. how many messages have been received?)

Messages

- Messages are objects of a class extending the `Message` class
  - Defines the `clone` method
- Messages are sent using the `send(m, neighbors, latencies)` method
  - `m` is the message to be sent
  - `destinations` is an array of nodes to which the message must be sent
  - `latencies` is an array of integers representing latencies (i.e. number of rounds the message transfer will requires)
- Messages are receveid using the `receive` method
  - Returns `null` if there is no message for the calling node

Configuration

- Done through a configuration file

```
random.seed 1234567890
simulation.cycles 20
simulation.timeDiagram

network.size 5

init.0_rnd WireRing
init.0_rnd.protocol example
init.0_rnd.k 2
#init.0_rnd.undirected

protocol.example example.ring.Ring
protocol.example.latencyRange 1
#protocol.example.perMessageLatency
#protocol.example.constantLatency
protocol.example.maxMessagesToSend 1
protocol.example.maxMessagesToReceive 1
control.1_observer example.ring.PostObserver

control.1_observer.protocol example
control.1_observer.observe.nbReceivedMessages
```

Outline

Examples

- Ring
  - Nodes are organized in a ring
  - Node 0 send a new message at the start of each round
  - All nodes but 0 forward messages they receive to their neighbor
- Multicast
  - Nodes are organized in a complete graph
  - Node 0 broadcasts a new message at the start of each round
  - All nodes but 0 broadcast messages they receive
  - A message does not do more than two hops