

Communication Model for Cooperative Robotics Simulator

Test Plan

Version 1.0

1. Test plan identifier

TestPlan-CommRobot-001

2. Introduction

This test plan is used to address the required test of “Communication Model for Cooperative Robotics Simulators”. The software test items and features will include the feature as described in Software Requirement Specification Version 1.1

3. Test Items

- CommunicationsSystem class
- RobotCommRecord class
- RobotParameters class
- PriorityQueue class
- Message class

4. Features to be tested

The following list described the features that will be tested based on the Software Requirement Specification Version 1.1

Feature identifier	Description	Requirement number
T-001	Register robot.	3.2.1,3.2.2
T-002	Sending broadcast message	3.2.3,3.2.5,3.2.6
T-003	Sending point-to-point message	3.2.9,3.2.11,3.2.13
T-004	Receiving broadcast message	3.2.4,3.2.5,3.2.7
T-005	Receiving point-to-point message	3.2.10,3.2.11,3.2.12,3.2.14
T-006	Start up all links	3.2.17
T-007	Shutdown all links	3.2.17
T-008	Set system range	3.2.19
T-009	Set system delay	3.2.18
T-010	Set system delivery probability	3.2.20
T-011	Enable broadcast capability	3.2.2
T-012	Disable broadcast capability	3.2.2
T-013	Enable point-to-point capability	3.2.2
T-014	Disable point-to-point capability	3.2.2
T-015	Start up robot’s send link	3.2.17

T-016	Shutdown robot's send link	3.2.17
T-017	Start up robot's receive link	3.2.17
T-018	Shutdown robot's receive link	3.2.17
T-019	Set robot range	3.2.19,3.2.16
T-020	Set robot delay	3.2.18,3.2.15,3.2.8
T-021	Set robot deliver probability	3.2.20

5. Approach

Unit testing

All executable java files, which are identified in section 3, will be tested. The JUnit, a testing framework for java unit testing, will be used to generate automated testing. One or more drivers and stubs will be developed to conduct unit testing. All executable files would be passed before starting integration testing.

Integration testing

The integration testing will include the combined function of several classes to perform the main software feature. Bottom-up integration will be used to conduct integration testing.

System testing

After integration testing, the system testing will be performed with other modules in the system to ensure all requirements are satisfied.

6. Environmental needs.

6.1 Hardware

The testing will be done on Sun Sparc and Intel-based machine.

6.2 Software

J2sdk version 1.4 is used to compile and execute program.

6.3 Operating systems.

Windows XP Professional

Unix Solaris

7. Test cases

7.1 Unit testing

7.1.1 Register robot

Test item: CommunicationsSystem class

Input: robot name, communication type (broadcast, point-to-point, or both)

Pass criteria:

- New RobotCommRecord class is created.
- The RobotCommRecord has a name as the input robot name.
- Unique RobotCommRecord.

- Broadcast, point-to-point, or both capabilities are enabled for this robot depending on the communication type.

Fail criteria

- New RobotCommRecord is not created.
- The RobotCommRecord has a name different from the input robot name.
- There is a duplication of RobotCommRecord (has the same name)..

7.1.2 Sending broadcast message.

Test item: CommunicationsSystem class

Input: Message object, time step

Pass criteria:

- All receivers get the message and save it to their priority queue.

Fail criteria:

- One or more receiver does not get the message or not save it to their priority queue.

7.1.3 Sending point-to-point message.

Test item: CommunicationsSystem class

Input: Message object, time step

Pass criteria:

- The specified receiver gets the message and save it to their priority queue.
- The received message is the same as the sending message.

Fail criteria:

- The specified does not get the message or not save it to its priority queue.
- The other receives get the message or save it to their priority queue.
- The received message is not the same as the sending message.

7.1.4 Receiving broadcast message.

Test item: CommunicationsSystem class

Input: robot name, current time step

Output: a set of Message object.

Pass criteria:

- Only Message objects, which has time step equal to current time step, are returned.
- The returned Message objects are removed from robot's priority queue.

Fail criteria:

- One or more returned Message objects, which has time step not equal to current time step, are returned.
- The returned Message objects are not removed from robot's priority queue.

7.1.5 Receiving point-to-point message.

Test item: CommunicationsSystem class

Input: robot name, time step

Output: a set of Message object.

Pass criteria:

- Only Message objects, which have time step equal to current time step, are returned.
- The returned Message objects are removed from robot's priority queue.
- Only Message objects, which have receiver name equal to robot name, are returned.

Fail criteria:

- One or more returned Message objects, which has time step not equal to current time step, are returned.
- The returned Message objects are not removed from robot's priority queue.
- One or more Message objects, which have receiver name equal to robot name, are returned.

7.1.6 Start up all links

Test item: CommunicationsSystem class

Pass criteria:

- IsLinkEnabled attribute is set to true.

Fail criteria:

- IsLinkEnabled attribute is set to something else.

7.1.7 Shutdown all links

Test item: CommunicationsSystem class

Pass criteria:

- IsLinkEnabled attribute is set to false.

Fail criteria:

- IsLinkEnabled attribute is set to something else.

7.1.8 Set system Range

Test item: CommunicationsSystem class

Input: System range

Pass criteria:

- Range attribute is set to system range.

Fail criteria:

- Range attribute is not set or set to something else.

7.1.9 Set system delay

Test item: CommunicationsSystem class

Input: System delay

Pass criteria:

- Delay attribute is set to system delay.

Fail criteria:

- Delay attribute is not set or set to something else.

7.1.10 Set system delivery probability

Test item: CommunicationsSystem class

Input: System delivery probability

Pass criteria:

- DeliveryProb attribute is set to system delivery probability.

Fail criteria:

- DeliveryProb attribute is not set or set to something else.

7.1.11 Start up robot's send link

Test item: RobotCommRecord class

Pass criteria:

- isSendLinkEnabled attribute is set to true.

Fail criteria:

- isSendLinkEnabled attribute is set to something else.

7.1.12 Shutdown robot's send link

Test item: RobotCommRecord class

Pass criteria:

- isSendLinkEnabled attribute is set to false.

Fail criteria:

- isSendLinkEnabled attribute is set to something else.

7.1.13 Start up robot's receive link

Test item: RobotCommRecord class

Pass criteria:

- isReceiveLinkEnabled attribute is set to true.

Fail criteria:

- isSendLinkEnabled attribute is set to something else.

7.1.14 Shutdown robot's receive link

Test item: RobotCommRecord class

Pass criteria:

- isReceiveLinkEnabled attribute is set to false.

Fail criteria:

- isReceiveLinkEnabled attribute is set to something else.

7.1.15 Set robot range

Test item: RobotCommRecord class

Input: robot range

Pass criteria:

- Range attribute is set to robot range.

Fail criteria:

- Range attribute is not set or set to something else.

7.1.16 Set robot delay

Test item: RobotCommRecord class

Input: robot1 name, robot2 name, robot delay

Pass criteria:

- Robot1 has a RobotParameter record that has robot2 as receiver name and robot delay as delay time.

- Robot2 has a RobotParameter record that has robot1 as receiver name and robot delay as delay time.

Fail criteria:

- There is no record of robot1's robotparameter that has robot2 as receiver name and robot delay as delay time.
- There is no record of robot2's robotparameter that has robot1 as receiver name and robot delay as delay time.

7.1.17 Set robot delivery probability

Test item: RobotCommRecord class

Input: robot1 name, robot2 name, robot delivery probability

Pass criteria:

- Robot1 has a RobotParameter record that has robot2 as receiver name and robot delivery probability as deliveryProb.
- Robot2 has a RobotParameter record that has robot1 as receiver name and robot delivery probability as deliveryProb.

Fail criteria:

- There is no record of robot1's robotparameter that has robot2 as receiver name and robot delivery probability as deliveryProb.
- There is no record of robot2's robotparameter that has robot1 as receiver name and robot delivery probability as deliveryProb.

7.1.18 Enable broadcast capability

Test item: RobotCommRecord class

Input: robot name

Pass criteria:

- The RobotCommRecord with this robot name has isBroadcastEnabled attribute set to true.

Fail criteria:

- The RobotCommRecord with this robot name has isBroadcastEnabled attribute set to something else.

7.1.19 Enable point-to-point capability

Test item: RobotCommRecord class

Input: robot name

Pass criteria:

- The RobotCommRecord with this robot name has isP2PEnabled attribute set to true.

Fail criteria:

- The RobotCommRecord with this robot name has isP2PEnabled attribute set to something else.

7.1.20 Disable broadcast capability

Test item: RobotCommRecord class

Input: robot name

Pass criteria:

- The RobotCommRecord with this robot name has isBroadcastEnabled attribute set to false.

Fail criteria:

- The RobotCommRecord with this robot name has isBroadcastEnabled attribute set to something else.

7.1.21 Disable point-to-point capability

Test item: RobotCommRecord class

Input: robot name

Pass criteria:

- The RobotCommRecord with this robot name has isP2PEnabled attribute set to false.

Fail criteria:

- The RobotCommRecord with this robot name has isP2PEnabled attribute set to something else.

8.19 Start up send link

Test item: RobotCommRecord class

Input: robot name

Pass criteria:

- The RobotCommRecord with this robot name has isSendLinkEnabled attribute set to true.

Fail criteria:

- The RobotCommRecord with this robot name has isSendLinkEnabled attribute set to something else.

7.2 Integration testing

7.2.1 Sending and receiving broadcast message

All robot records tested in these scenarios have broadcast capability and set parameters to default unless they are set by the scenario. The default parameters are incoming link and outgoing link enables, system link enables, delay time is zero, delivery probability is 100% and no maximum range is set for the entire system and for each robot.

7.2.1.1 Scenario 1 -- Default scenario

Pass criteria:

- All robots get the broadcast messages.

Fail criteria:

- One or more robots do not get the broadcast messages.

7.2.1.2 Scenario 2 -- Disable broadcast capability.

Pass criteria:

- A robot without broadcast capability does not get any broadcast messages from other robots
- Other robots do not get any broadcast messages from a robot without broadcast capability.

Fail criteria:

- A robot without broadcast capability gets broadcast messages from other robots
- One or more robot gets broadcast messages from a robot without broadcast capability.

7.2.2 Sending and receiving point-to-point message

All robots tested in these scenarios have point-to-point capability and set parameters to default unless they are set by the scenario. The default parameters are incoming link and outgoing link enables, system link enables, delay time is zero, delivery probability is 100% and no maximum range is set for both system and for each robot.

7.2.2.1 Scenario 3 -- Default scenario

Pass criteria:

- A robot with specified address gets a point-to-point message.

Fail criteria:

- Other robots that are not the specified address get a point-to-point message.

7.2.2.2 Scenario 4 -- Disable point-to-point capability.

Pass criteria:

- A robot without point-to-point capability does not get any point-to-point messages from other robots
- Other robots do not get point-to-point messages from a robot without point-to-point capability.

Fail criteria:

- A robot without point-to-point capability gets point-to-point messages from other robots
- One or more robot gets point-to-point messages from a robot without point-to-point capability.

7.2.3 Both broadcast and point-to-point message

The following scenarios apply for sending both broadcast and point-to-point messages. All robot records tested in these scenarios have broadcast capability, point-to-point capability and set parameters to default unless they are set by the scenario. The default parameters are incoming link and outgoing link enables, system link enables, delay time is zero, delivery probability is 100% and no maximum range is set for both system and for each robot.

7.2.3.1 Scenario 5 -- Disable system link

Pass criteria:

- All robots do not get any messages from other robots.

Fail criteria:

- One or more robots get messages from other robots.

7.2.3.2 Scenario 6 -- Disable outgoing link of a robot.

Pass criteria:

- No robots get the message sending from a robot with disabled outgoing link.

Fail criteria:

- One or more robots get the message sending from a robot with disabled outgoing link.

7.2.3.3 Scenario 7 -- Disable incoming link of a robot.

Pass criteria:

- A robot with disabled incoming link does not get any messages from other robots.

Fail criteria:

- A robot with disabled incoming link gets messages from other robots.

7.2.3.4 Scenario 8 – Set system delay

Pass criteria:

- Receiver gets message from sender after the time of sender sent message plus system delay. (Received time = sent time + system delay)

Fail criteria:

- Receiver gets message from sender in different time other than sent time plus system delay. (Received time \neq sent time + system delay)

7.2.3.5 Scenario 9 – Set robot delay

Pass criteria:

- Receiver gets message from sender after sender sent message plus robot delay time. (Received time = sent time + robot delay)

Fail criteria:

- Receiver gets message from sender in time other than sent time plus robot delay. (Received time \neq sent time + robot delay)

7.2.3.6 Scenario 10 – Set system delay and robot delay

Pass criteria:

- Receiver gets message from sender after sender sent message plus system delay plus robot delay time. (Received time = sent time + system delay + robot delay)

Fail criteria:

- Receiver gets message from sender in time other than sent time plus system delay plus robot delay. (Received time \neq sent time + system delay + robot delay)

7.2.3.7 Scenario 11 – Set system range

Pass criteria:

- Only robots within *this range* of sender get the message.

Fail criteria:

- One or more robots outside *this range* of sender get message.

* *this range* refers to system range.

7.2.3.8 Scenario 12 – Set robot range

Pass criteria:

- Only robots within *this range* of sender get the message.

Fail criteria:

- One or more robots beyond *this range* of sender get message.
- * *this range* refers to robot range.

7.2.3.9 Scenario 13 – Set system range and robot range

Pass criteria:

- Only robots within *this range* of sender get the message.

Fail criteria:

- One or more robots outside *this range* of sender get message.
- * *this range* refers to system range + robot range.

7.2.3.10 Scenario 14 – Set system delivery probability

Pass criteria:

- Only randomly selected message within the probability range will delivery to the receivers.
- Randomly selected message beyond the probability range will not delivery to the receivers.

Fail criteria:

- Randomly selected message within the probability does not delivery to the receivers.
- Randomly selected message beyond the probability range delivery to the receivers.

7.2.3.11 Scenario 15 – Set robot delivery probability

Pass criteria:

- Only randomly selected message within the probability range will delivery to the receivers.
- Randomly selected message beyond the probability range will not delivery to the receivers.

Fail criteria:

- Randomly selected message within the probability does not delivery to the receivers.
- Randomly selected message beyond the probability range delivery to the receivers.

7.2.3.12 Scenario 16 – Set system delivery probability and robot delivery probability

*The probability range will be the average of system and robot delivery probability.

Pass criteria:

- Only randomly selected message within the probability range will delivery to the receivers.
- Randomly selected message beyond the probability range will not delivery to the receivers.

Fail criteria:

- Randomly selected message within the probability does not delivery to the receivers.
- Randomly selected message beyond the probability range delivery to the receivers.

7.3 System testing

The System testing will be done by having Environment use the communications interface to send and receive message. Robots create their own messages and send or receive message through the communication model via the environment. The environment control panel can also set the parameters via the environment. The test will be conducted using the same test cases as integration testing. The system testing will ensure the compatibility with other modules in the system.

8. Schedule

All testing will be performed during July 14 – July 28, 2004. Unit testing will be performed after finish coding each module. Integration testing will begin after all unit testing are done.