Mazewar Protocol

Ying Gao, Yunlu Li

04/10/2016

Mazewar is a distributed, multiplayer game. Each player in the game controls a rat in the maze, and receives points for tagging other rats with a missile and loses points for being tagged.

1. General Specification
   1) All packets are sent over UDP in network order.
   2) Clients join the same UDP multicast group to communicate with each other.
   3) Interval for sending heartbeat message is 1s, referred to as STATE_UPDATE_INTERVAL.
   4) Missile in flight is only visible to its sender in the bird's eye view map, but invisible to other players.
   5) Each rat has at most 1 missile in flight at the same time. It can only fire another missile when the previous one hits a wall, or another rat.
   6) We support at most 4 concurrent players, since the maze is quite crowded.

2. Packet Descriptor

Descriptor	Description
State Update (0)	Sent when a rat's state changes, or as heartbeat message every 1 second.
Missile Hit (1)	Missile sender sends this packet to victim to update its score.
Leave Game (2)	Player leaving the game sends this packet to all other players.

3. Packet Header

0	8	16	24
Player ID (INT32)			
Sequence ID (INT32)			
Packet Type Descriptor			
Player Name (16 Bytes in ASCII)			

   1) Size of packet header totals to 28 bytes.
   2) Player ID is a 32-bit integer generated randomly.
   3) Sequence ID is a strictly increasing unsigned integer generated by each player.
   4) There are 3 types of packets in total:
      a. State update (0)
      b. Missile hit (1)

    c. Leaving game (2)
5) Player Name takes 16 bytes in total, so maximum length of nickname allowed is 16 characters.

Usage:

1) Each packet has a header prepended. Receiver of the packet can use the header to determine the sender.
2) Receiver is responsible of tracking the largest sequence id seen for a given sender. Packets come from the same sender with sequence id smaller than or equal to the largest should be dropped.

4. State update packet payload

0	8	16	24
Position x	Position y	Direction	Cloaked
Score			

1) 0 <= posX <= 31, 0 <= posY <= 15, using bottom left corner as the origin.
2) There are 4 directions in total: N (0), S (1), W (2), E (3)
3) Score is a 32-bit signed integer
4) Cloaked is a 1-bit Boolean value indicating the rat is invisible (1) or visible (0).
5) Cloaking duration is 5s, recuperation time is set to 5s as well. To be more clear, after the rat appears again, the recuperation time starts accumulating.

Usage:

1) A rat is responsible of sending a state update packet if its state changes, which includes
    a. Position change.
    b. Direction change.
    c. Change cloaked state
    d. Join a new game or reappear after being tagged at a new random position
    e. Score change.
2) Each rat also sends this packet as a heartbeat message every STATE_UPDATE_INTERVAL, if there's no state change.

5. Missile hit packet payload

0	8	16	24
Victim ID			

1) Victim ID is the ID of the rat hit by the missile.

Usage:

1) Each rat is responsible of keeping track of the missile it fired. It can either hit a wall or a rat.
2) When a missile hits a rat, sender sends the packet above to the victim twice with the same sequence ID. This is to increase the consistency level of scores in case of the packet above being dropped.
3) The victim decreases it score by (cloaked ? 7 : 5) points.

6. Leaving game packet
   This message has no payload.

   Usage:
   Player sends the packet above to notify other players when it tries to quit.

7. Game play rules
   1) Joining the game
      When joining the game, the new player connects to the multicast group and listens for heartbeat message from other players for 5 seconds. It
      f. updates its peer table accordingly, if no peer exists, just moves to next step.
      g. generates a random ID that doesn't conflict with others'
      h. chooses a random position untaken (no wall or rat occupying the cell)
      i. initializes its own score to 0
      j. starts sending state update packets

      If a new player happens to collide with a missile, it's tagged immediately. It will lose point and reappear at a new random position.

   2) Leaving the game
      Player voluntarily leaving the game sends a leaving packet. Other players remove the one leaving from their peer table.

   3) Position conflict
      If two rats try to occupy the same position, both should back off to its previous location and retry the move.

8. Exception Handling
   1) If a player cannot hear heartbeat message from its peer for more than 10 * STATE_UPDATE_INTERVAL, it rejoins the game as a fresh new player.
   2) If a heartbeat message hasn't been heard from a player for 10 * STATE_UPDATE_INTERVAL, the lost player is treated as gone.
   3) Player should drop packets from its peers sent in reverse order.
   4) Inconsistency may occur in the case of missile hit. If both the two packets are dropped, the score of the sender is increased whereas that of the receiver is not decreased. We decide not to deal with this issue further, since the possibility of two failed packets is relatively low.