

Robocup Rescue Simulation - Communication Channels

Mohammad Mehdi Saboorian

February 2007
Rev.2

1 Introduction

Till now, agents could only discriminate between unread received messages based on senders of them. Now with introducing channels, agents can easily decide which channel(s) they want to listen to. So there will be more information about a message available for agents. On the other hand, finally, kernel can force constraint on number of messages each agent reads.

2 Protocol changes

Agent developers should consider these changes in order to use new communication system. Please note that for detailed information about rescue simulation communication protocol (including channels) you should see "*Robocup Rescue Simulation - Communication Protocol*".

- From now on, **AK_TELL** will be the only communication command. And it'll include an extra field for specifying target channel.
- **AK_SAY** will be completely ignored by kernel.
- **KA_HEAR** now includes a channel number, that specifies source of it's data.
- Kernel will not send **KA_HEAR_SAY** and **KA_HEAR_TELL** anymore.
- Agents should use the new **AK_CHANNEL** command to inform kernel about their preferences for reading messages. For instance, in a simple scenario, agent *A* selects channel 1 and 2. (We assume that each agent can read maximum of 4 messages at each cycle.) Now with this command kernel will remember to forward maximum of 2 messages from channel 1 and 2 messages from channel 2 to *A*.

Those teams using *librescue* can use these new features easily. Required changes for *Yab* are listed at appendix ???. Also, *RescueCore* will support channel based communication in near future.

3 Communication rules

- Previous constraint on number of messages for agents are still valid.
- Now all channels are accessible for each agent.
- Agents can read their own message with no restriction.
- Currently two types of channels are supported:
 1. Say channel (channel 0)
 2. Radio channels
- Say channel is a replacement for AK_SAY messages and have similar restrictions. (only hearable within a specified range)

A Yab API

in RCRSSProtocolSocket.java

```
public void akTell (int selfId, int channel, String message)
{
    send(AK_TELL, new Object[] {
        new Integer(selfId), new Integer(channel),
        message});
}

public void akChannel (int selfId, byte[] Channels)
{
    send (AK_CHANNEL, new Object[] {
        new Integer (selfId), Channels});
}
```

in KaHear.java

```
public final int channelId;
public KaHear(DataInputStream dis) throws IOException
{
    selfId = dis.readInt();
    senderId = dis.readInt();
    channelId = dis.readInt();
    message = RCRSSProtocol.readStringElement(dis);
}
```

in ProtocolConstants.java

```
AK_CHANNEL = 0x90;
```