The Black Sheep Team Description

Cameron Skinner, Jonathan Teutenberg, Gary Cleveland, Mike Barley, Hans Guesgen, Pat Riddle, and Ute Loerch

The University of Auckland
New Zealand
{cam,jteu004,gcle027,barley,hans,pat,ute}@cs.auckland.ac.nz

1 Introduction

This short paper describes the functionality of the Robocup Rescue agents built by Team Black Sheep[1] at the University of Auckland[2]. We have implemented agents for each of the four components of the simulated environment: fire brigades, police forces, ambulance teams and emergency centers (fire stations, police offices and ambulance bases).

Our mobile agents are called Fireman Sam[1], PC Plod and Dr Ropata[2].

2 Communication

All our agent implementations share a common communication protocol. Messages are encoded using a set of message types, a timestamp and zero or more bytes of message-dependant data. Agents queue messages internally during a timestep. When it comes time to send the messages, they are packed into 80-byte chunks. Up to four (the maximum number of messages allowed by the Robocup Rescue rules) of these chunks are then sent to the kernel.

Agents are only allowed to read four messages per timestep. This is enforced by each agent. When an 80-byte chunk arrives the agent unpacks all the components of the chunk and looks at each component in turn. If any of the components are useful[3] then the message is considered to be "read". If none are used then the message is considered to be "ignored and discarded", as per the rules specification. When decoding messages if the timestamp of a received message is more than one step behind the current time then that message is ignored.

We cannot list all message details due to space limitations, so an example will have to suffice. Suppose a fireman wants to report a fire in building 1884 during timestep 23. He will send the message shown in figure 1.

Other messages, such as reporting blocked roads, extinguished fires and calling for help, are similar.

---

[1] http://www.cs.auckland.ac.nz/˜rescue

[2] http://www.auckland.ac.nz

[3] A message component is useful if it is either a) used to update internal state or b) forwarded on to other agents

Bits	Value	Meaning
0-5	1	Constant describing the “report a fire” message
6-15	23	The timestep that the message was generated
16-47	1884	The building ID as a 32-bit number
48-55	3	The fieryness property of the building

Fig. 1. Example “report a fire” message

3 Strategies

3.1 Trivial

This strategy is extremely simple. Fire brigades and ambulance teams can request help from police to clear roads, but otherwise each agent is entirely independent. Centers pass on the road-clearing requests but do nothing else.

Fireman Sam Trivial Fireman Sam maintains a list of known fires which is updated every time the agent receives a KA_SENSE message. At each timestep Sam picks the closest target from this list. If he is close enough to extinguish the fire then he does so, otherwise he attempts to plan a path to the fire. If no path can be found, then the next closest fire is chosen and the algorithm repeats. Once a path is found, any blocked road segments on that path are reported to the police and Sam proceeds to attempt execution of the path.

PC Plod This agent maintains a list of known blocked roads (updated by a KA_SENSE message), and a list of blocked roads that other agents have requested cleared. At each timestep, PC Plod examines his current target. The target is invalid if one of the following conditions holds:

1. The target is known to be clear due to an update from the kernel (i.e a KA_SENSE message)
2. The current target was taken from the list of known blocked roads and the list of requests is not empty

If the target is invalid then a new target is chosen: either the head of the list of requests, or the closest known blocked road if there are no requests. Once a valid target is found, PC Plod plans a path to the target and attempts to clear it. If this agent's current location (or a neighbouring road if the agent is on a node) is blocked then that location is cleared before attempting to move

Dr Ropata The trivial Dr Ropata works similarly to Fireman Sam, except that he looks out for buried civilians and attempts to rescue them.

Trivial Center The trivial centers ignore all messages except requests for roads to be cleared, which are all forwarded on.

3.2 Simple

The first real set of strategies is quite simple. There is little coordination between agents and only minimal centralised control. Agents report fires, blocked roads and buried civilians to each other, and fire and ambulance officers can request help from police to clear roads. Centers pass these reports on to the relevant agents. Agents also report their current position and when they have no valid targets. Centers will assign jobs to idle agents.

A summary of each of the agents behaviour when receiving a message is shown in figure 2. In addition, if ordered to do a particular job (extinguish a fire, clear a road or rescue a civilian) by the corresponding emergency center then they will stop what they are doing and carry out the order.

Message	Response			
	Fireman Sam	PC Plod	Dr Ropata	Simple Center
Fire	Update state	-	-	Update state
	Replan	-	-	Order idle agents
Fire extinguished	Update state	-	-	Update state
Blocked road	Update state	Update state	Update state	Update state
	-	-	-	Forward message
Road cleared	Update state	Update state	Update state	Update state
	-	-	-	Forward message
Need road cleared	-	Add road to queue	-	Forward message
Buried civilian	-	-	Update state	Update state
	-	-	Replan	Order idle agents
Civilian rescued	-	-	Update state	Update state

Fig. 2. Simple agent behaviour

3.3 Clustering

3.4 Auction

The auction strategy is a centralised scheme. Emergency centers maintain information about the jobs that need doing by listening for reports from agents. When a new job becomes available the center asks for bids from the agents - these bids reflect each agents confidence that it can complete the job within some timeframe. The agent with the highest bid then gets assigned the job. This mechanism is still being implemented and how best to generate bids remains to be seen.

4 Summary

Team Black Sheep has implemented (or is in the process of implementing) four types of strategy. The first, trivial implementation is intended as a benchmark against which other implementations can be tested. The “simple” agents use centralised control to assign jobs to idle agents, but the agents are otherwise largely independant. They share as much information as possible, however. The “cluster” implementation groups buildings into city blocks and reasons on which fires to extinguish based on these clusters. The auction implementation is a centralised control mechanism where agents bid for jobs. The auctions are controlled by the various emergency centers.

References

1. Fireman Sam. http://www.firemansam.co.uk (2003)
2. New Zealand Film Archive.
   http://www.filmarchive.org.nz/collections/collections_images.html (2003)