Hybrid Communication Network for Autonomous Agents

Joshua P. Bruckmeyer, Dr. Eric M. Schwartz, Dr. A. Antonio Arroyo
Machine Intelligence Lab, University of Florida

2000 Florida Conference on Recent Advances in Robotics
May 4-5, Florida Atlantic University

Abstract

This paper sets the basis for a communications network designed for autonomous agents. The criteria for this network are defined, discussed, and meet in this paper.

To meet these criteria two current network systems are discussed and compared. By taking the advantages of the two current network systems a hybrid networking system was formed. The devices and algorithms required to communicate on this hybrid network have been defined and discussed.

Introduction

Reliable communication between autonomous agents is an essential step in the development of swarm behaviors. Previous attempts at robot communication at the Machine Intelligence Lab (MIL) have used RS-232 over various media. There are two fundamental problems with this:

1. Transfer Error:
   When dealing with a network media there is always a probability that there will be interference during communication. If measures are not made to detect, and correct these errors the communicating devices will go out of synchronization thus communication will fail.

2. Clock Recovery
   The RS-232 protocol assumes the communication devices are operating at the same clock rate, to a certain degree of certainty. No attempt is made to recover the clock; the data is collected at the rate of the receiving unit. When transmitting large amounts of data the difference in the clock rates may cause errors in the data—which would also cause the devices to go out of synchronization.

The above issues are not the only issues in robot communication, but they are major stepping stones that must be climbed in order to address higher level ones. This paper will address these issues and suggest methods for resolving them.

Background

Communication between autonomous agents is an essential requirement for future work in the area of swarm behavior. Previous attempts at the Machine Intelligence Lab (MIL) have involved:

- Light Emitting Diode (LED) transfer of data based on the RS-232 protocol.
- Wireless RS-232 protocol communication.

The above methods have worked in isolated cases but have not provided a general means of interconnecting autonomous agents. In both cases above the robots would have trouble sending data over a long period. Data would be lost and the robots would not be able to recover. The previous research at the MIL indicates that an RS-232 protocol, by itself, does not provide an adequate interface for communication.

Network Development

To establish communication between devices one first define the requirements of the system. The requirements are as follows:

- Operate on a single wireless channel, with a common transmit and receive port.
- Have error checking and data arrival verification.
- Each device must be the same, so each device on the network is interchangeable.
- The clock must be recoverable from the data transmitted.

To find solutions to the requirements one should look into current technology, evaluate the advantages and disadvantages, and incorporate them into the new system. Two local area network (LAN) networks, Ethernet and Token Ring, will be evaluated in this paper.

Token Ring

Token Ring maybe an unpopular network solution today, but unpopular does not mean it is a bad solution. Token Ring operates on a star-wired based network, were transmit and receive lines of a device are tied to a port on a concentrator. To access the ring the device has to follow a protocol, which includes closed-loop and open-loop diagnostics. A dc current, called a phantom current, is applied to the port. This serves two purposes:

1. Informs the concentrator when to insert the port.
2. Allows the device to monitor the resistance of the wire.

By monitoring the resistance, the device can detect a short or open circuit [1]. This adds to the robustness of the system.

Originally once on the ring the devices were just connected to each other—using relays. Cable length was limited by the amount of tolerable clock skew and jitter each device on the network could handle. Today many concentrators do not directly connect devices together. Using Phase Lock Loops (PLL) the data is received, re-clocked, then transmitted [1]. This allows for accurate clock-generation with a minimization of clock skew and jitter.

On the ring, itself there is a device that acts as an active monitor. If no active monitor is present when a device enters the ring, that device may become the active monitor via a protocol. The active monitor's primary purpose is to circulate a token around the ring. Each device on the ring receives the token, then retransmits it. All devices on the ring must participate; this makes it easy to isolate down devices.

To transmit data on the ring a device must first obtain the token. Only one device at a time may posses the token, so there are never two stations transmitting at the same time—except in the case of a ring error. Once the token is received the device changes the frame format to indicate that the token is in use, and then adds the rest of the information to the frame—including the destination address. The data frame is passed along just as in the case of the token. When the frame reaches the destination address it copies the frame data and passes the frame along. When the frame arrives at the transmitting device, the transmitting device releases the token and the transmission is complete. If the transmitting device does not receive, the frame a ring error has occurred and actions are taken to correct the situation [1].

Token Ring is a very efficient networking system, "…as load increases, the utilized bandwidth also increases up to the ring data rate. Token Ring achieve full bandwidth of the media" [1]. For this efficiency, there is a price. All stations on the ring must always participate, so when a device is not in use it can not shut off to save power. There is also the overhead the protocols needed to become an active monitor and insert into the ring.

Ethernet

Ethernet operates on a bus network, where all devices are connected over the same media. It is based on Carrier-Sense Multiple Access with Collision Avoidance (CSMA/CD) protocol. When a device wants to transmit data, it waits for a quite time on the network then initiates its transfer of data. If another device transmits at the same time, a collision occurs where both devices back off and wait for a random amount of time. During a collision, both messages and network time are lost when a collision is being resolved [1].

Devices can be removed from the network if they cause 60 or more consecutive collisions or if it jabbers. A device is jabbering if it continually transmits data for 40,000 to 75,000 bit times. When a port jabbers it is removed from, the network until is ceases jabbering [2]. The probability of a collision increases with increased network traffic and devices. This network congestion causes data to be slowed, so Ethernet networks normally do not operate at the highest rate possible. Therefor Ethernet are less efficient than their Token Ring counterpart [1].

Comparison

The communication system under design will be based on a commercial off the shelf (COTS) single channel wireless technology with a single link to the antenna. This media resembles the bus media of an Ethernet network. One major difference in the two media is that in the wireless media can only transmit or receive at any given time. There is not connection to the media when the device is transmitting so collisions can not be detected. The only method of detecting collisions would be to find errors in the frame. The received frame would be of invalid structure and be discarded. Therefore, no other device on the network would know to send out a request for retransmission frame. The sending device would have to continually try to retransmit the frame, not knowing if it is a media error or if the receive device is no longer present on the network. For the reasons above it is clear that a true Ethernet based network is impractical.

The Token Ring network method of token passing offers many advantages in the transmission of data. Having only one device is allowed to transmit at a time allows for collision free transmission of data over the bus created by the wireless system. Another advantage of a Token Ring based media is error detection. When a device transmits data, and no errors occur, the device receives the data after it is transmitted around the ring. By use of flags, the transmitting device can check to see if the destination device received the frame. If the frame is a match for what it sent it, and the received flag has been set, the transmitting device can be assured that the data has been properly sent.

One major issue with a Token Ring based network is what to do with when a device wants to enter the ring. In an ordinary Token Ring network the device would insert phantom, receive the token, and send out a request for access to the ring. In the case of the shared bus media, any unauthorized access to the network will cause collision of data. In this manor, no device will be able to tell that the new device wants access to the network.

Both networks use a MAC address to distinguish one device from another. The assignment of MAC address is normally regulated. If a duplicate address is occurs on the network both devices will respond to the same frames, causing major network errors. To prevent this Token Ring networks make sure there are no duplicate addresses during the insertion process. If there is another device on the ring with the same address, the new device is not allowed on the ring [1].

Hybrid Solution

<u>Network Basis</u>

Both LAN network formats have their advantages and disadvantages. To provide a robust communication interface a hybrid solution of the two networking types is suggested. Token Ring offers the distinct advantage when it comes to passing data on an established ring. There is no bus contention. If all the devices operate properly, error detection is a simple matter of comparing what one received to what one sent.

<u>Insertion</u>

To handle the case of an insertion the active monitor periodically places a quite time on the ring. When a station wants to insert it detects this silence and sends out a request to be inserted into the ring. When the active monitor receives this frame, it adjusts the ring to include that device.

If two devices try to insert at the same time, the active monitor will observe an invalid frame format. Once the invalid request has been received, a collision frame is sent out to the all of the devices. This tells the devices to back off for a random period and try again. This can occur for a set number of attempts. If the devices can not be inserted into the ring, the active monitor starts circulating the ring as before.

If the inserting station does not detect an active monitor on the network, it may become the active monitor of the system

<u>Removal</u>

When a device wants to be removed from the ring, it sends a frame out indicating its intention. The active monitor will receive its request, send out an acknowledge frame followed by a removal frame. Once the frame has been transmitted, the active monitor restructures the ring to eliminate the device.

All devices on the network MUST respond to the removal command, in this manor if a device is not functioning properly it can be removed from the network so normal functional can be restored.

<u>Active Monitor</u>

The ring requires an active monitor to control the ring. The active monitor handles token generating, insertions, removal, error recovery, and ring restructuring. Any device can become the active monitor is one is not present.

Becoming the Active Monitor

To become the active monitor a device must wait for a set period the try to insert into the ring. If the request is not granted it may become the active monitor. If there are, two active monitors both are to remove their status of active monitor then wait a random time to repeat the above procedure.

Token Generation

It is the responsibility of the active monitor to assure that either data or a token is passed around the network. A token is a frame with control bits set to indicate that the ring is free. If no data is present, and no token detected the active monitor must generate another token. If that token is lost, it must go into error recovery mode.

The active monitor must also ensure that data does not transmit around the ring more than once. This is accomplished by setting and checking a flag in every frame it receives. On the first pass through the active monitor it sets the flag, on the second pass it takes action to correct the ring. If the frame has passed around the ring, twice the transmitting device has not removed the frame; thus, no new data can be transmitted. The active monitor must remove the frame and go into error recovery to determine which device has caused the error.

Insertions

Periodically the active monitor must stop network traffic so new devices can access the ring. The active monitor must first send out a frame indicating its intentions to temporarily disable the ring—this is to prevent other stations from trying to become the active monitor or join the ring.

The active monitor waits for a period for request for insertions. If no request is received, it starts the token passing again. If a request is received, the active monitor makes sure the new address is not a duplicate of one on its internal list. If there is no error, the active monitor restructures the ring to include the new device. If there is a duplicate address, a frame is sent to the offending device informing it. If a request is received but the frame is invalid, a collision frame is sent out. This informs the inserting devices to back off for a random period and try to insert again. The number of attempts should be capped so that malfunctioning stations are not allowed on the ring.

Removal

When the active monitor receives a request for removal from the ring, it first sends out an acknowledgement frame. Then it sends a removal frame to tell the device it must remove itself from the ring. Once the device has received the frame the active monitor modifies its address table then restructures the ring.

In the case of error recovery, the active monitor may remove a station that is not following the proper protocol.

Error Recovery

When either the active monitor detects an error or a device reports an error the active monitor must go into error recovery mode. The objective of this mode is to isolate the error and correct it so that network traffic may resume.

To find the error each device on the ring is sent a check frame. If the active monitor receives the frame correctly, it knows the device is functioning properly. If communication between the device and the active monitor can not be established the device is removed from the ring.

If all devices on the ring can communicate and an error still exists, the active monitor must restructure the ring. If the restructuring does not work, an error signal is produced to the system attached to the active monitor for user intervention.

Ring Restructuring

When a station is inserted, removed, or an error can not be solved the ring needs to be restructured. The active monitor keeps a record off all the devices on the ring. This list is kept in an arbitrary order—since order is not important on this ring. The active monitor takes the first device and keeps it for its closest downstream neighbor. The active monitor then sends a frame to the closest downstream neighbor informing that device of its closest downstream neighbor. This continues until a loop is formed back to the active monitor.

Once the ring is structured, the active monitor sends out a frame informing all stations on the ring what address are on the ring.

Devices on Ring

Devices on the ring are responsible to:

- Immediate removal at the request of the active monitor
- Remember its nearest downstream neighbor
- ONLY transmit data when it posses the token
- Remove transmitted data from the ring when it has passed around the ring
- Report network errors to the active monitor
- Obey the insertion rules as set forth above

Transmit

In order for a device to transmit data on the ring, it must wait for the token to be passed to it. When the token is received, the device may transmit the data. If the frame is received with no errors, the data has been transmitted correctly. The device is to remove the frame and pass a token. If there has been an error, the device may try to retransmit the data. If that attempt fails, it must inform the active monitor of a ring error.

Receive

When a device on the ring is passing a data, it has to look for its address. If the device finds the address, it must copy the frame into a buffer. Once in the buffer the frame integrity must be verified. If an error is found the data is discarded and the device must wait for the transmitter to retransmit the data.

Conclusion

To verify that the requirements of the communication network have been met each requirement will be reviewed and discussed.

- Operate on a single wireless channel, with a common transmit and receive port.

  The hybrid network above has been designed to operate on a single channel media. At no time is a device required to transmit and receive at the same time. This allows communication over a common transmit and receive port.

- Have error checking and data arrival verification.

  By nature of the hybrid network, each frame is verified by comparison to the original. When the transmitting device receives the copy of its properly formatted frame, it is assured that it has arrived at its destination.

- Each device must be the same, so each device on the network is interchangeable.

  All devices on the network are equivalent and interchangeable.

- The clock must be recoverable from the data transmitted.

  Each time a device receives a frame it uses a PLL to recover the clock from the data.

The requirements of the hybrid network have been met so ends the scope of this paper. Much work needs to be done to realize the hybrid network established in the paper. This work includes:

- Definition of frame format and control codes
- Development of the required hardware, and software
- Development of an application to use this network with

References

[1] Understanding Token Ring Protocols and Standards. 1998, Artech House, Inc. James T. Carlo, Robert D. Love, Michael S. Siegel, and Kenneth T. Wilson. Pages 1-35.

[2] VHDL for Programmable Logic. 1996. Addison-Wesley Publishing Company. Kevin Skahill. Pages 326-332