

PROPOSAL OF POLLED-CSMA PROTOCOL FOR AN AD HOC SCADA SYSTEM

**Yang Yuxiang, Dang Xiaoyu, Zhou Tingxian
Harbin Institute of Technology**

ABSTRACT

In this paper we introduce an ad hoc distributed supervisory control and data acquisition (SCADA) system, which covers two types of traffic, a small number of slave stations with stream traffic and much larger number of slave stations with burst traffic. To improve the system's real-time transmission and channel allocation efficiency, a MAC scheme, named Polled-CSMA, is designed by integrating the merits of polling and CSMA protocols. Extensive analysis shows that this MAC protocol meets the system's QOS requirements.

KEY WORDS

MAC, QOS, CSMA, SCADA

INTRODUCTION

A distributed supervisory control and data acquisition (SCADA) system often has star topology, consisting of a master station that acts as supervisory and control center and several slave stations that act as real sensing and control units. All slave stations use a single channel (upstream channel) to transmit information packets to a master base station, and the master station transmits short acknowledgment packets and control packets to slave stations. Downstream traffic can be transmitted in a separate channel (using a different frequency band). Or, it can share a single channel with upstream traffic. In SCADA system, the transferred data from different units may have different delay and bandwidth requirements, thus it is a challenge to design a communication protocol to meet these different QOS requirements^[6]. In this paper, we focus on a narrow-band wireless SCADA system, in which there are two types of traffic, a small number of slave stations with stream traffic and much larger number of slave stations with burst traffic. In this kind of SCADA system, there are different services and traffic requirements, and the traditional MAC schemes are not suited to it. To meet the system's

QOS and traffic requirements, a Polled-CSMA protocol is proposed in this paper. The remainder of this paper is organized as follows: In section 2 different kinds of MAC protocols are analyzed. In section 3 the proposed protocol--Polled-CSMA is presented in detail. In section 4 a quantificational analysis is performed on Polled-CSMA.

COMPARISON OF MAC PROTOCOLS

MAC protocols can be divided into three classes: (1) Fixed assignment, FDMA CDMA, etc. (2) Random assignment, such as ALOHA, CSMA, etc. (3) Demand assignment, such as token ring, polling. The above protocols and its variant versions are adapted to different services. In common LAN system, the allocation efficiency is the most concerned issue, while in SCADA system, the real-time transfer matters^{[1][2]}.

Polling is a basic preference in SCADA system for its simplicity. In polling protocol, the master station polls the slave stations in regular sequence, no matter whether they have data to send or not. If the polled slave station has data ready to send, it send the data, if not it keeps silent, and the master station goes on to poll another slave station in subsequent polling slot. Featured by simplicity and by giving each station equal priority to access the shared channel, polling protocol is very widely used in SCADA system. But polling is not suited to our system, For the traffic from different station is very different in our system. In master station polling way, the master station often polled the slave station with low traffic and few packets to send, those with high traffic can not get the priority to send data. Thus the channel is often wasted and not used fully.

Token ring may be considered as an update version of polling protocol. In token ring, each station should have an order number, and all the station order numbers are linked to form a logical ring. A token is passed on along this ring, only the station that obtains a token can access channel. Token ring protocol has more channel efficiency than the polling protocol at the cost of more protocol complexity. But in terms of channel assignment, both token ring and polling are fixed assignment scheme, and they will not work properly in multi-service environment. So neither polling nor token ring meet the QOS requirements in our system.

Random access is another channel assignment scheme. In random access protocols, a channel is occupied by contention. There are many typical random access protocols such as ALOHA, CSMA/CD, etc. In ALOHA, a station sends packets all at once no matter the channel is busy or not. If the station detects a packet collision with packets from other stations, it defers some time to retransmit the corrupted packet. If the packet collides again it repeats the above process until successful transmission.

In high load condition, because all stations send data blindly, the performance of ALOHA is not satisfactory^[5]. CSMA is the improved version of ALOHA. In this protocol, a station will not send packets until it detects the channel is idle. If the channel is busy, it defers some time, waits for another chance to send the packets. By using the channel sensing technology, CSMA has more channel efficiency than ALOHA.

The principal merits of random access methods are their ability to serve a large number of stations, each with a low average data rate and a high peak rate. When too many stations try to communicate all at once or are in heavy load, the system throughput goes down and transmission delay increases substantially. In our system there is stream traffic, the channel traffic load is relatively heavy, so a pure random access protocol does not meet the system's traffic requirements.

POLLED-CSMA PROTOCOL

From the above analysis, we know that it is difficult to use the shared channel efficiently and to control the maximum packet delay when applying traditional MAC protocols to our system. The reason is that there are two classes of traffic, namely burst traffic and stream traffic. If the shared channel is assigned based on the traffic of the burst station, the channel efficiency is good, but the maximum delay increases. On the other hand, if the channel is assigned on the traffic of stream stations, the maximum delay decreases, and the channel efficiency decreases significantly as well.

To solve those problems, we propose a polled-CSMA protocol. By integrating both polling and CSMA protocol, this protocol successfully improves the system's performance. The protocol is described below:

- (1) The master station polls all the slave stations with large traffic. The master station sends polling packet to the stations with large traffic at interval of T in regular sequence. If a polled station has data ready to send, it sends the data. Otherwise, it keeps silence. All stations with large traffic have equal priority to access the shared channel.
- (2) All the stations with low traffic access shared channel with CSMA. When a station with low traffic receives a polling packet, it first checks the channel, if no packet head is detected, it transmit its own packet. That is to say only if any station with large traffic does not transmit a packet can a station with burst traffic send packet.

When the total traffic of the stations with burst traffic is low, a burst traffic station can access the shared channel in large probability. For the stations with stream traffic have priority to access the channel, their

QOS requirements can also be met. So the proposed scheme achieves efficient use of channel, and makes it possible to control the maximum delay.

PROTOCOL ANALYSIS

Given the polled-CSMA scheme, now we could analyze the protocol in detail. To simplify the analysis, some assumptions and variables are taken and defined: (1) every station sends packet independently, furthermore every station with burst traffic has the same probability G_m to send a packet, and similarly every station with stream traffic has the same probability G_n to send a packet. (2) Any station, whether it is the station with burst traffic or not, has the same packet duration T. (3) The length of polling packet is negligible, so the polling interval is equal to the length of packet T.^[3]

Some variables and parameters are defined as follows:

m: number of the stations with burst traffic

n: number of the stations with stream traffic

T: length of packet from slave station to master station

G_m : Probability of a station with burst traffic to send a packet within T

G_n : Probability of a station with stream traffic to send a packet within T

Our basic assumption is $m \gg n$, $nG_n > mG_m$. To compare the performance of the Polled-CSMA with other protocols, we estimate each of their throughputs and delay respectively.

In polling protocol, every station has the equal priority to access the shared channel, so

$$\begin{aligned}
 S_{poll} &= (mG_m + nG_n)/(m + n) \\
 &= \frac{m}{m+n}G_m + \frac{n}{m+n}G_n \\
 &\because G_n > G_m \\
 &\therefore \\
 S_{poll} &< \frac{mG_n}{m+n} + \frac{nG_n}{m+n} = G_n \quad (1-1)
 \end{aligned}$$

Where S_{poll} is the normalized throughput of a system using polling protocol.

The random protocol is a more complex protocol, we only evaluate its performance approximately. The system's total traffic is ^[4]

$$G = \frac{mG_m + nG_n}{m + n}$$

For collision corruption, the throughput of a random access protocol is less than the system's total traffic, so

$$S_r < G = \frac{mG_m + nG_n}{m + n}$$

For $G_m < G_n$ then,

$$S_r < G = \frac{mG_m + nG_n}{m + n} < \frac{mG_n + nG_n}{m + n} = G_n \quad (1-2)$$

Where S_r is the normalized throughput of a system that uses random access protocol.

The polled-CSMA is integrated by the polling and CSMA protocols. The stream traffic stations use polling protocol to access the shared channel, while the burst traffic stations use CSMA protocol to access the channel. We assume that under pure competitive environment the throughput of all the stations with burst traffic is S_c , then,

$$\begin{aligned} S_{P-CSMA} &= G_n + S_c(1 - G_n) \\ &> G_n \end{aligned} \quad (1-3)$$

Where S_{P-CSMA} is the normalized throughput of a system that uses Polled-CSMA protocol.

From (1-1), (1-2), (1-3), we can see that

$$S_{P-CSMA} > S_{poll}$$

$$S_{P-CSMA} > S_r$$

So the Polled-CSMA is better than either the polling or random access protocols.

In terms of delay, Polled-CSMA can guarantee the packet delay $D_s \leq nT$ to the stream traffic station. The stream traffic stations can send their packet under their maximum delay requirement. The burst traffic stations, which contend for the rest of the slots not used by the stream traffic stations, have low band requirements and high delay tolerance, their requirements can also be met within the competitive environment. Pure random access does not guarantee any station's delay, nor does the delay of the stream traffic station. Polling protocol can guarantee maximum delay, but poor channel efficiency in various traffic conditions.

CONCLUSION

In this paper, we have presented a new MAC protocol to apply to an ad hoc SCADA system. It has also presented the performance evaluation results of the proposed protocol, in which it is confirmed that the channel efficiency of the proposed Polled-CSMA scheme is higher than that of the pure random or pure fixed assignment scheme. It is also confirmed that the assumed system using the proposed protocol can accommodate different types of traffic more efficiently than the system using other schemes.

REFERENCES

- [1] Joaquin luque and Isable Gomez, "The role of media access control protocols in SCADA system", IEEE Transaction on power delivery, Vol.11, No.3, July, 1996
- [2] Qiu Gongwei, Zhao Xiangyuan and Wu Shuping, Communication technology in multi micro-computer real control and sophisticated instrument system, Tsinghua University Publishing House, 1996
- [3] Xie Xireng, Computer network, Publishing House of Dalian University of Science and Technology, 1989
- [4] D. RAYCHAUDHURI, "Stability,Throughput, and Delay of Asynchronous Selective Reject ALOHA", IEEE Trans. Commun.,vol. COM-35,No.7, July 1987
- [5] Yang Yunnian VSAT network, Post Telecommunication Publishing House,1997
- [6] Takahiro Yamada, "An extension to the CCSDS packet telemetry to support data transfer with various service requirements", ITC/USA'97