An Efficient Message Scheduling Algorithm for Arbitrary Topology PROFINET IRT Networks

Jinbo Sim[1], Jooyoung Son[2*]

[1] Zinnos Inc., 1125-18 Dongsam Dong, Yeongdo Gu, Busan, S. Korea
jbsim@zinnos.com
[2] Div. of IT Engineering, Korea Maritime University,
1 Dongsam Dong, Yeongdo Gu, Busan, S. Korea
mmlab@hhu.ac.kr

Abstract. The proposed efficient message scheduling algorithm is for real-time PROFINET IRT networks with arbitrary topologies. Message scheduling in real-time networked control systems determines a sending order of messages to/from a controller. Minimizing the cycle (message sending) time and the scheduling (computation) time is essential for the performance of the system. The proposed algorithm optimizes not only the cycle time but also the scheduling time even in the dynamic network environments with mixed various messages, update times, and arbitrary topologies.

Keywords: PROFINET IRT, arbitrary topology, real-time networks, message scheduling

1 Introduction

The signal collision or switch queue delay in Ethernet makes it difficult to communicate in real-time. To overcome this problem, Real-time Ethernets (RTEs) have been developed [1]. PROFINET IRT (Isochronous Real-Time) is a typical example with the most stringent real-time performance. For the PROFINET IRT networks in dynamic environments with varying data sizes and data update times, the message scheduling results should be optimal as well as the computation time itself should be minimized. An efficient message scheduling scheme is proposed in this paper as a way to satisfy that.

2 Previous works

In the PROFINET IRT networks, messages should be transmitted in the order where the cycle time is minimized. The problem is formalized into Resource Constraint Project Scheduling with Temporal Constraints (RCPS/TC) problem in [2], but it takes

* Corresponding author: Jooyoung Son

excessive computation time. The dynamic frame packing (DFP) method is proposed by [3], but it can be applied only to the tree topology and upstream communication.

3 Efficient Message Scheduling

The efficient message scheduling algorithm is proposed in this paper to optimize the cycle time and its computation time for arbitrary topology PROFINET IRT networks. The algorithm follows the two steps:

(1) Topology Conversion to MST: The network topology is converted into a minimum spanning tree (MST) in order to remove the needs of routing. The controller becomes the root node in the converted tree. Other devices are converted to intermediate nodes or leaf nodes. The MST information is delivered to each node. Compared to existing MST algorithms, a simpler algorithm can be applied because the root node election can be omitted.

(2) Efficient Frame Packing: The message transfer unit in the PROFINET IRT networks is called a datagram. Each datagram is transmitted in a separate Ethernet frame. To reduce the number of frames, an Efficient Frame Packing algorithm (EFP) is proposed, which has no restrictions on the direction of communication and the topology. Fig. 1 and Fig. 2 are the procedures for down and upstream communication, respectively. The datagrams to send are sorted in the order of non-increasing size. The procedure `NIFF_EFP()` in Fig. 3 packs the sorted datagrams into frames just like for the bin packing problem. The optimality of the performance and computation time of the NIFF (Non-Increasing First Fit) algorithm is well-known [4].

```
proc EFP_Downstream_by_Controller(PROFINET)
 // step 1: convert topology to MST
 run MST algorithm for the topology of input PROFINET;
 sort the devices in order of nonincreasing depth of devices;
 make group g be the set of devices with the same depth;
 set n_groups as the number of groups;
 // step 2: datagrams packing into frames
  make frame frames[]; // sequence of frames to send
  n_frames = 0; // # of frames containing datagrams to send
  for g = 0 to n_groups - 1 do
    gather datagrams to send to devices in group g;
    set n_dgrams as # of datagrams for devices in group g;
    make dgram dgrams[n_dgrams] for datagrams in group g;
    sort dgrams in order of nonincreasing datagram_size;
    NIFF_EFP(dgrams, n_dgrams, frames, &n_frames);
  end
  for f = 0 to n_frames - 1 do
   flood frames[f]; // broadcast the frames to devices
  end
end proc
```

Fig. 1. Pseudocode of EFP for downstream communication performed by controller

```
proc EFP_Upstream_by_Devices(PROFINET)
  gather datagrams dgrams[n_dgrams] from conflicted frames;
  set n_dgrams as # of datagrams gathered;
  sort dgrams[] in order of nonincreasing datagram_size;
  make frame frames[]; // sequence of frames to send
  n_frames = 0; // # of frames containing datagrams to send
  NIFF_EFP(g_dgrams, n_dgrams, frames, &n_frames);
  for f = 0 to n_frames - 1 do
    send frames[f] to controller; // unicast to controller
  end
end proc
```

Fig. 2. Pseudocode of EFP for upstream communication executed by branch devices

```
proc NIFF_EFP(dgram *dgrams, int n_dgrams,
                   frame *frames, int *n_frames)
  for d = 0 to n_dgrams - 1 do // d is index of dgrams[]
    f = 0;
    while (frames[f].filled_space + dgrams[d].datagram_size)
            > MAXFRAME do //look for a frame to fit a datagram
      f++;// frames[f] is insufficient and try the next frame
    end
    frames[f].datagrams[frames[f].n_datagrams++] =
            dgrams[d].datagram_index;
    frames[f].filled_space += dgrams[d].datagram_size;
    if *n_frames < f+1 then do
      *n_frames = f+1; // update the # of filled frames
    end
  end
end proc
```

Fig. 3. Pseudocode of the adapted NIFF algorithm for EFP

4 Conclusions

A message scheduling algorithm for the PROFINET IRT is proposed. The efficient frame packing considers arbitrary topology, messages of variable sizes, and various update times as well. The adapted NIFF algorithm which is originally for the well-known bin packing problem can be applied in both directions of communications. The number of frames and the cycle time are minimized as results.

References

1. Felser M.: Real-Time Ethernet – Industrial Prospective. In: Proceedings of the IEEE, vol.93, no.6, pp.1118-1129, IEEE Press, New York (2005)
2. Hanzalek Z., Burget P., Sucha P.: Profinet IO IRT Message Scheduling With Temporal Constraits. In: IEEE Transactions on Industrial Informatics, vol.6, no.3, pp.369-380, IEEE Press, New York (2010)
3. Wisniewski L., Schumacher M., Jasperneite J.: Fast and Simple Scheduling Algorithm for PROFINET IRT Networks. In: 9th IEEE International Workshop on Factory Communication Systems, pp.141-144, IEEE Press, Lempo (2012)
4. Baase S.: Computer Algorithms. Addison-Wesley, Philippines (1978)