華中科技大學
HuaZhong University of Science & Technology

Control Science and Engineering

Petri Net Modeling of the Reconfigurable Protocol Stack for Cloud Computing Control Systems

Dr. Naixue Xiong
Georgia State Univ. GA

Authors: Chunjie Zhou, Hui Chen, et al.

OUTLINE

I. What is Cloud NCS?

II. Why we need a new architecture?

III. Our proposal - RPS architecture

IV. How to model the RPS architecture

V. Experimental results

VI. Conclusions

What is Cloud NCS?

Definition: A networked control system (NCS) uses a distributed control architecture where sensors, actuators and controllers are interconnected through real time network.

Cloud Computing + Industrial Ethernet

= **Cloud NCS**: the control systems based on Industrial Ethernet that can obtains services from Cloud Computing for hiding the complexity of resource scheduling and reducing the calculation effort and storage cost of end users.

What is Cloud NCS?

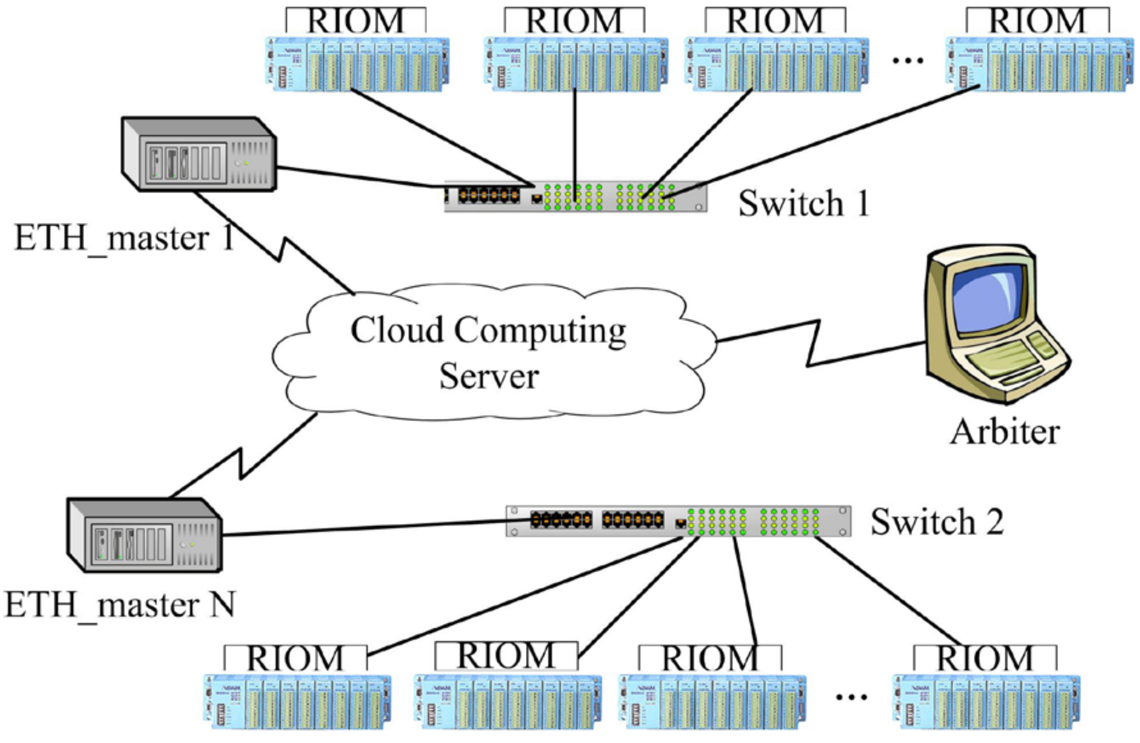


Fig.1 The cloud structure of Industrial Ethernet system

Characters of Cloud NCS:

1. Producer/Consumer application model, Master/Slave interaction structure
2. Heterogeneous devices due to various Industrial Ethernet standards
3. Communication resources may change unexpectedly, due to changes in user demands, or disturbances in the network environments.

Why we need new architecture?

Existing services in Cloud Computing

- Infrastructure as a Service (IaaS)
- Platform as a Service (PaaS)
- Software as a Service (SaaS)

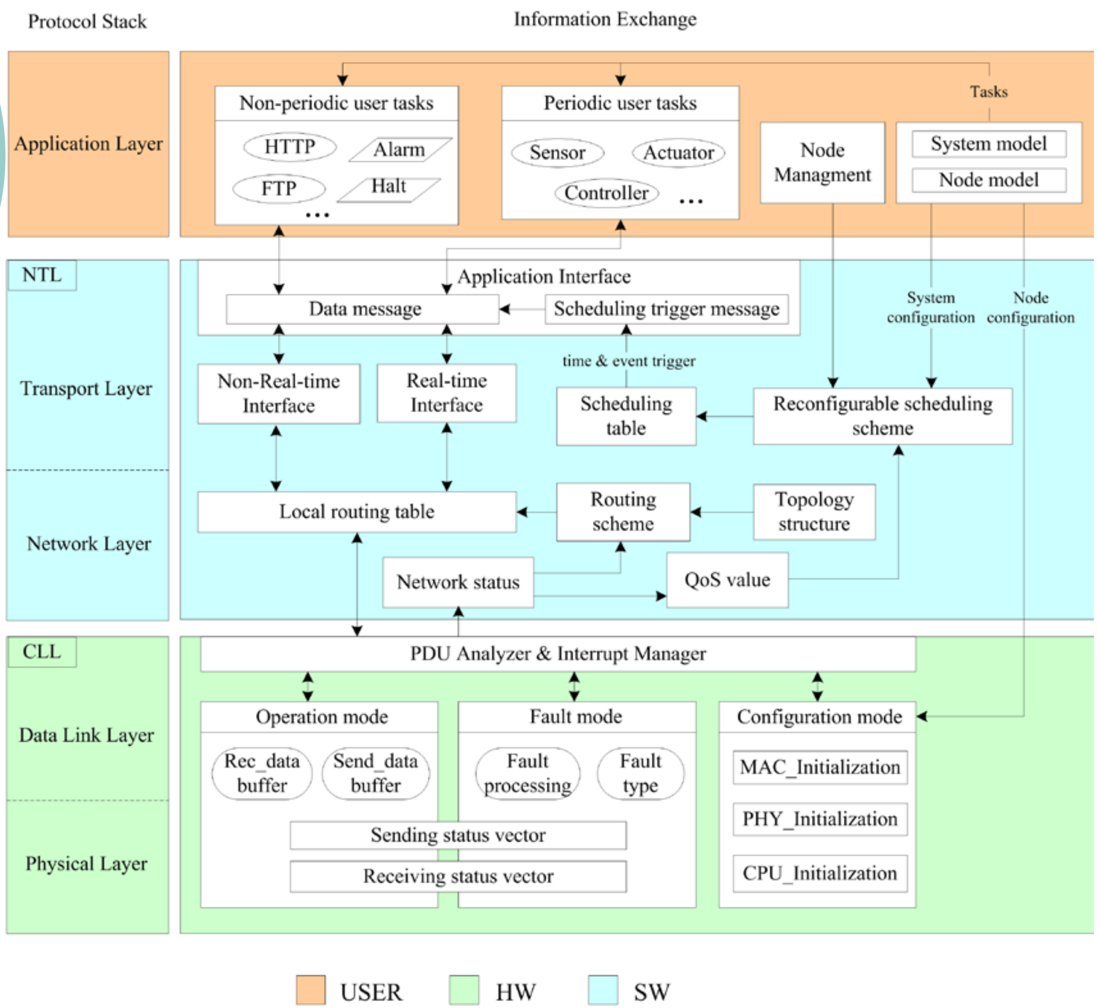
- Benefit users by delivering resources transparently.
- Enjoy existing Cloud services based on HTTP.
- Still, there are a lack of Cloud service for digital appliances unsupported by HTTP protocol in NCS.

- **Challenge:** guaranteed real time service for control applications under heterogeneous and dynamic environments.

- **Solution:** communication protocol stack with reconfiguration capability to provide the flexibility in management of network elements. (Reconfigurable Protocol Stack, RPS)

- **Our focus:** a protocol model enabling Cloud to provide Services based on various protocols for NCS users.

Our proposal - RPS architecture



Producer/Consumer

TDMA

CSMA/CD

Fig. 2 Architecture for the reconfigurable protocol stack of NCS

Our proposal - RPS architecture

The RPS architecture deals with the cooperation between CSMA/CD and TDMA mechanism under the produce/consumer cooperation model.

1) **Non-real time Channel**: uses the CSMA/CD mechanism and standard TCP/IP suite to send reconfiguration requests and non-real-time traffic.

2) **Real time Channel**: transmits real-time traffic under the TDMA based scheduling scheme to stabilize the local network utilization and ensure the quality of control is within constraints of the variations of network QoS.

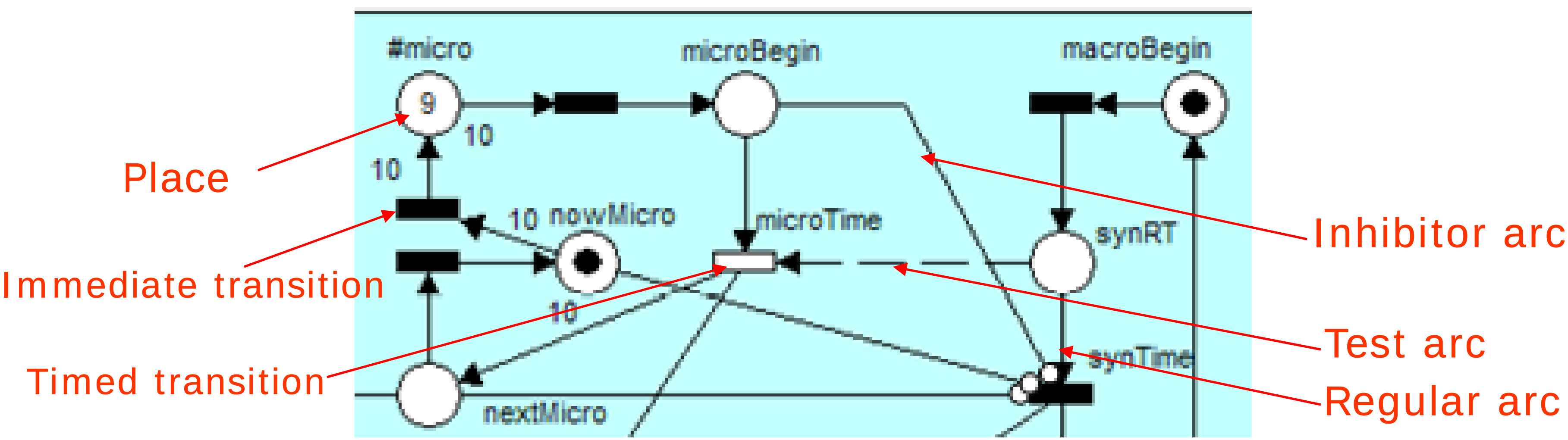
Carrier Sense Multiple Access/Collision Detection (CSMA/CD);

Time Division Multiple Access (TDMA)

How to model the RPS architecture

Modeling method - DSPN

- Places <-> protocol entities
- Transitions <-> protocol behaviors
- Arcs <-> control signals



(1) CSMA model of CLL

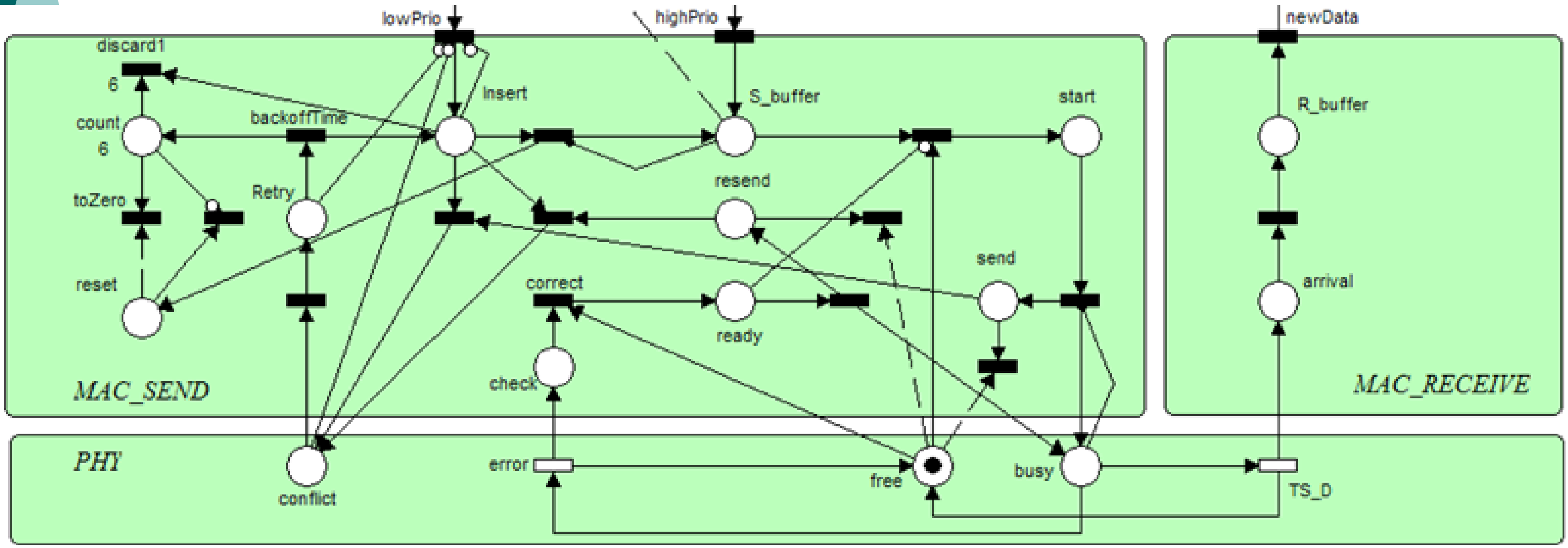


Fig.5(part 1). Combined DSPN model for the reconfigurable protocol stack

(2) TDMA model of NTL

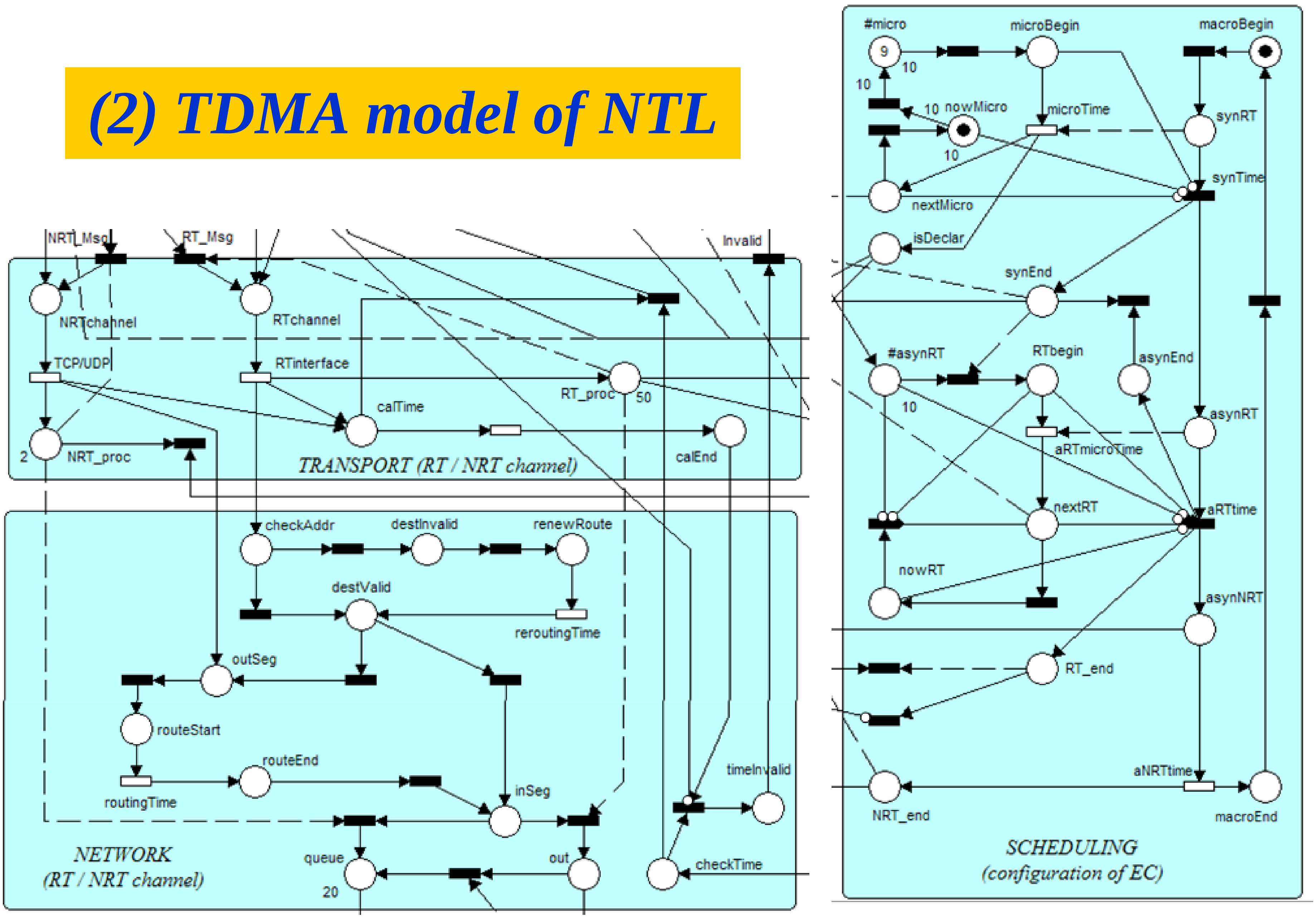


Fig.5(part 2). Combined DSPN model for the reconfigurable protocol stack

(1) Producer/consumer model of APPL

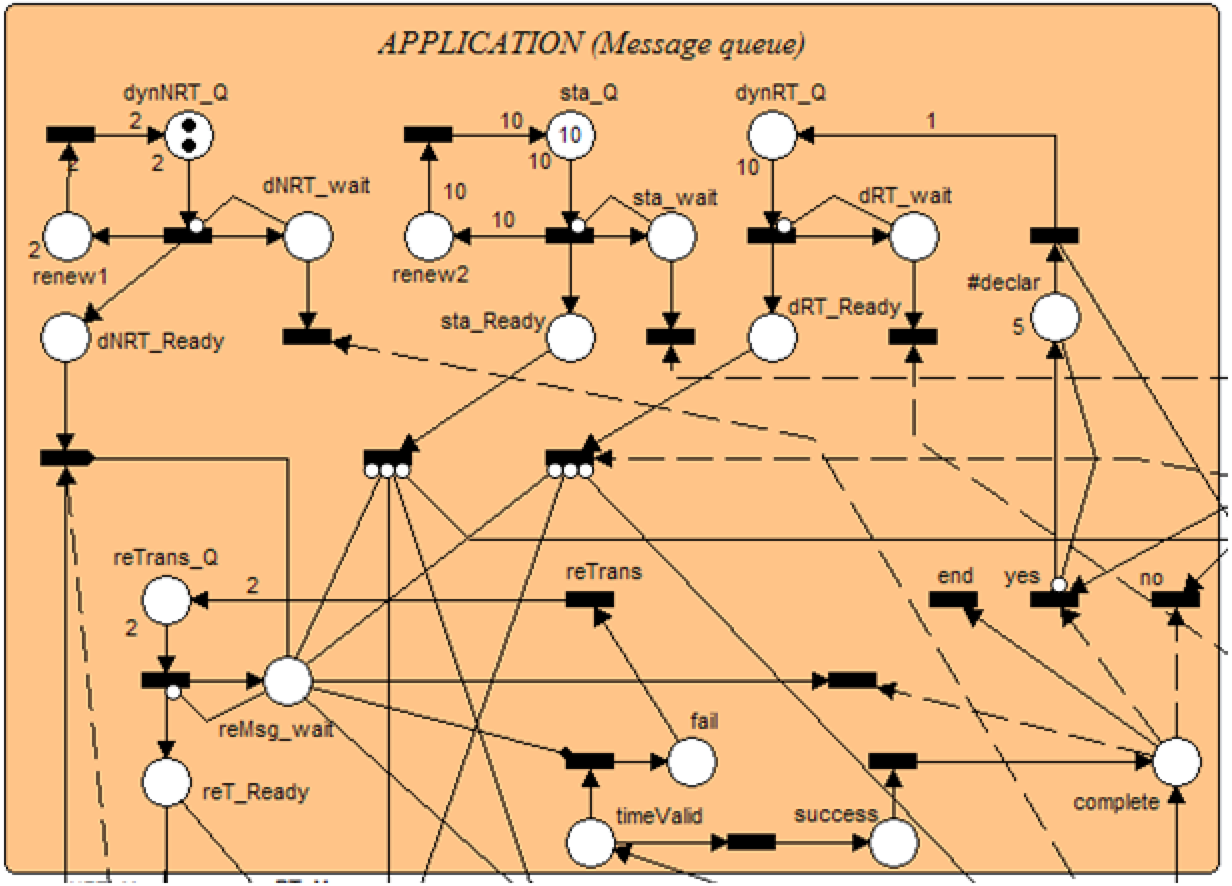


Fig.5(part 3). Combined DSPN model for the reconfigurable protocol stack

Experiment results

Network performance metrics

- **Time Constraints**

If the token in the state *queue* is 0 at the intervals of *asyNRT* activation, the time constraint on the real time traffic is satisfied.

If the token in the state *queue* is 0 at the intervals of *macroEnd* activation, the reservation bandwidth for the non-real time constraints is enough for scheduling all messages.

- **Network Utilization**

Network Utilization = busy / (busy+ free)

- **Network Efficiency**

Network Efficiency = 1 – (fail–a) / (complete + fail)

Experiment results

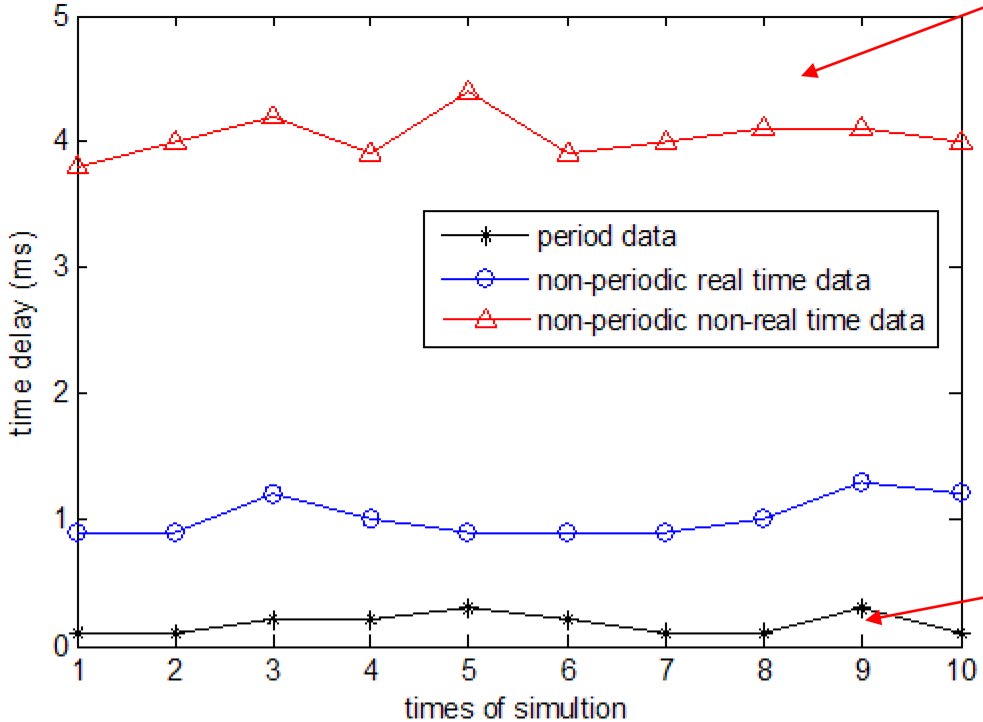
The parameters of DSPN simulation model

Layer	Features	Configurations

Application	real time and non-real time	sta_Q = 10;
		dynRT_Q = #delcare during simulation;
		dynNRT_Q = 2 (periodic to ready)

Scheduling	TDMA scheduling	microTime=5; aRTmicroTime=5; aNRTime=10
Transport	TCP/UDP	TCP/UDP = 3; RTInterface = 1 timeOut = 10 (deterministic type)
Network	IP based routing	Prob (inSeg) = Prob (outSeg); Prob (destInvalid) = Prob (destValid); routingTime = 1~3 (uniform type) reroutingTime = 2 (deterministic type)
MAC	CSMA/CD	BackoffTime = immediate type; count = 10;
Physical	average 100Mbit/s; bus length up to 100 meters ; 46 ~ 1500 bytes per data frame	TsD = 1~3 (uniform type); error = 20~40 (uniform type, occurring probability ≈ 0.05~0.1%)

Experiment results



The delay result calculated from the Message distribution diagram

The average real time delay = *0.18 ms*

Fig.3 Time delay distributions (real time v.s. non-real time, data)

Conclusions

Cloud computing offers lower costs and higher computing ability for large-scale industrial systems.

Industrial Ethernet plays an important role for development of a Cloud Computing based control system.

The dramatic growth of Industrial Ethernets confronts designers with serious difficulties from architecture heterogeneity and environment variability.

Conclusions

1) We presented reconfigurable protocol stack and general framework: real time performance analysis with DSPN.
2) Reconfigurable protocol stack was evaluated through a set of layer models related to reconfiguration activities.
3) It can be considered as a PaaS providing a flexible real time Ethernet service for interfacing with the field control sub-systems and Cloud Computing server.

Future topics related to cloud NCS,

Dynamic scheduling algorithms for incorporating these formal models into a virtual machine

The management framework for virtualization technology

...

Thanks For Paying Your Attention!

Questions?

More details are given by

Chunjie Zhou, Hui Chen,

Email: huichen@ieee.org