

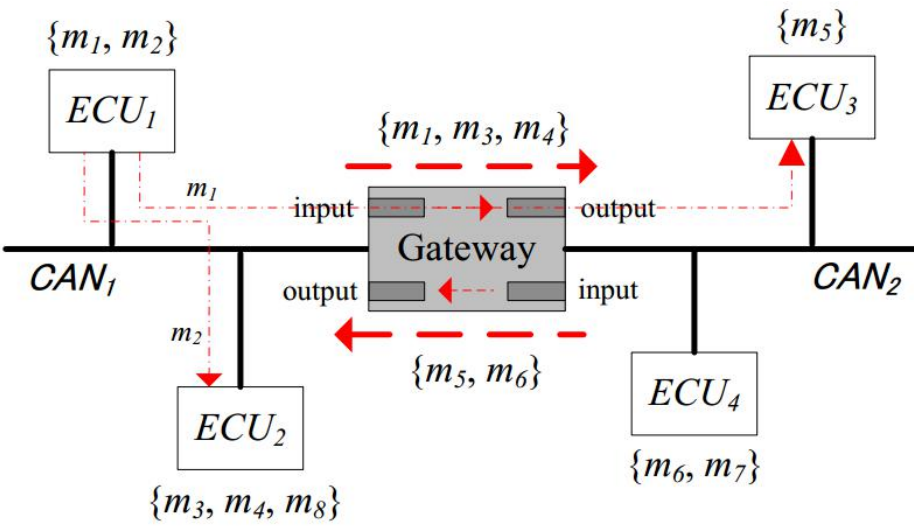
周报告

宋金林 导师：李仁发教授

此前小结

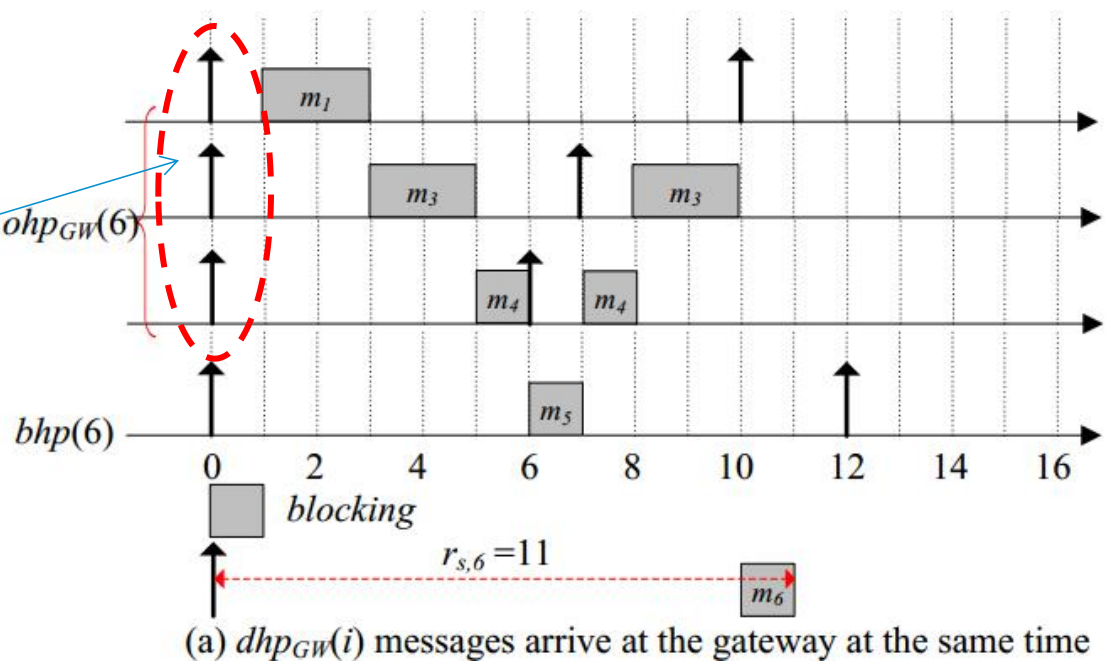
- 完成了CAN网络的时间分析（同构网络）
 - 充分分析和精确分析（基于忙周期理论）
 - 单个CAN以及网关互连CAN
- 当前正在学习研究异构网络的时间分析

网关互连CAN网络

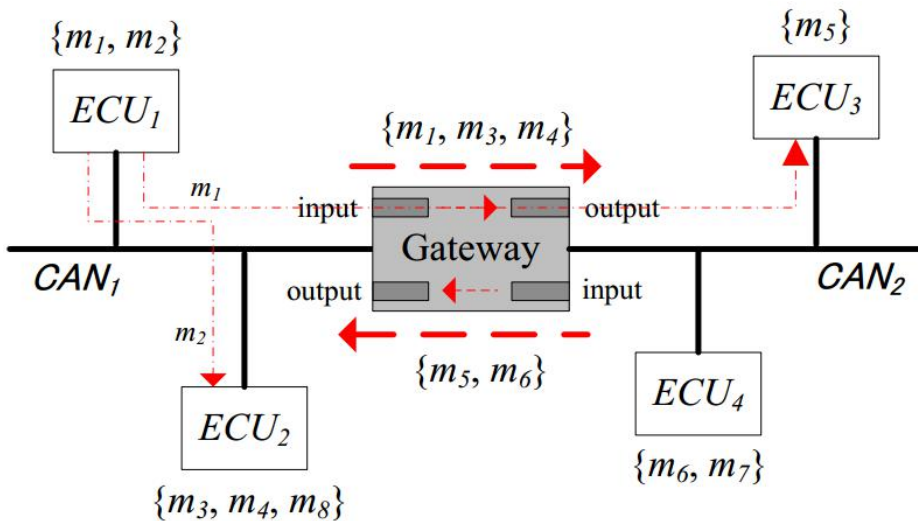


对于网关互连CAN₁和CAN₂，分析CAN₂网络内消息响应时间需要考虑来自CAN₁网络的网关消息的抢占时延（反之分析CAN₁也一样需要考虑来自CAN₂消息的抢占）。

以 m_6 为例：为简化，可假设 m_1, m_3, m_4 到达CAN₂是同步的。



网关互连CAN网络



```
//sub,ecu,pri,per,size,off,jit,gw
```

```
message msg[8]={
{1, 1, 1, 12, 2, 0, 0, 1},
{1, 1, 2, 12, 1, 0, 0, 0},
{1, 2, 3, 12, 2, 0, 0, 1},
{1, 2, 4, 12, 1, 0, 0, 1},
{1, 2, 8, 12, 1, 0, 0, 0},
{2, 5, 5, 12, 1, 0, 0, 1},
{2, 6, 6, 12, 1, 0, 0, 1},
{2, 6, 7, 12, 1, 0, 0, 0}};
```

选择C:\Users\song\Desktop\can-can_Azketa方法.exe

after pri sort:

| | | | | | | | |
|---|---|---|----|---|---|---|---|
| 1 | 1 | 1 | 12 | 2 | 0 | 0 | 1 |
| 1 | 1 | 2 | 12 | 1 | 0 | 0 | 0 |
| 1 | 2 | 3 | 12 | 2 | 0 | 0 | 1 |
| 1 | 2 | 4 | 12 | 1 | 0 | 0 | 1 |
| 2 | 5 | 5 | 12 | 1 | 0 | 0 | 1 |
| 2 | 6 | 6 | 12 | 1 | 0 | 0 | 1 |
| 2 | 6 | 7 | 12 | 1 | 0 | 0 | 0 |
| 1 | 2 | 8 | 12 | 1 | 0 | 0 | 0 |

inversion delay:

| | | | | |
|---|---|----|----|----|
| 2 | 2 | 0 | 0 | 0 |
| 2 | 1 | 0 | 0 | 0 |
| 2 | 2 | 0 | 0 | 0 |
| 1 | 1 | 0 | 0 | 0 |
| 1 | 1 | 0 | 0 | 0 |
| 1 | 1 | 0 | 0 | 0 |
| 1 | 1 | 0 | 0 | 0 |
| 1 | 1 | 0 | 0 | 0 |
| 2 | 2 | 4 | 4 | 8 |
| 2 | 1 | 5 | 0 | 0 |
| 2 | 2 | 7 | 6 | 13 |
| 1 | 1 | 7 | 6 | 13 |
| 1 | 1 | 7 | 8 | 15 |
| 1 | 1 | 11 | 10 | 21 |
| 1 | 1 | 16 | 0 | 0 |
| 1 | 1 | 12 | 0 | 0 |

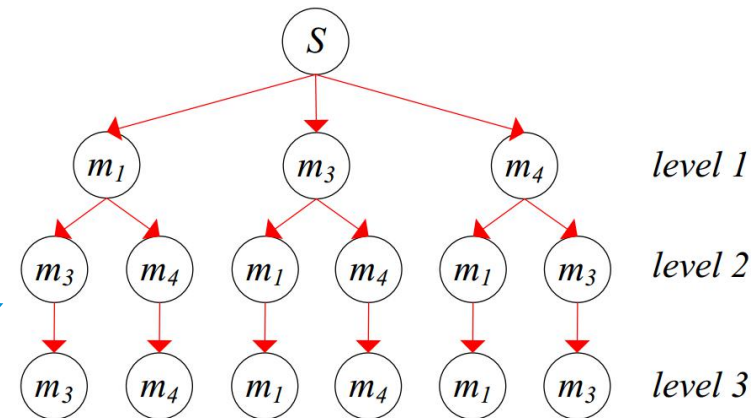
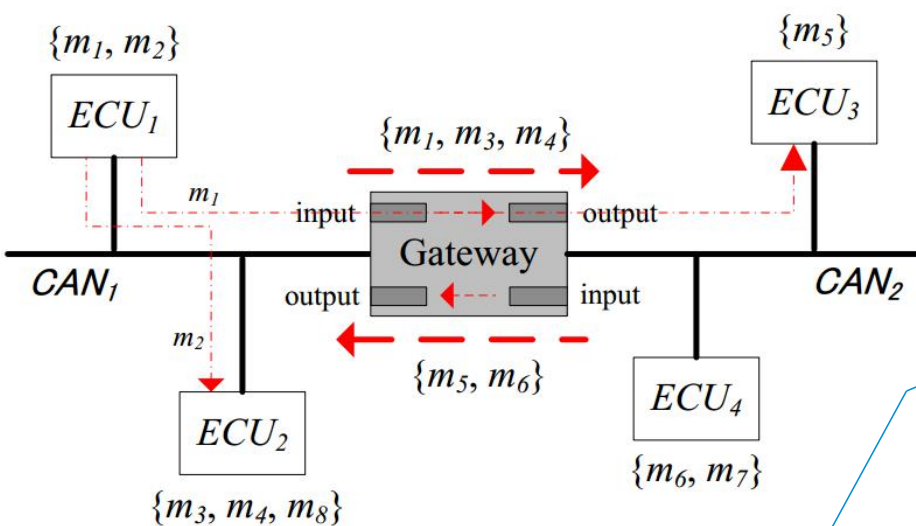
源端时延 (Source delay) points to the first column of the inversion delay table.

目的地端时延 (Destination delay) points to the second column of the inversion delay table.

总时延 (Total delay) points to the fourth column of the inversion delay table.

Process exited after 0.3187 seconds with return value 0

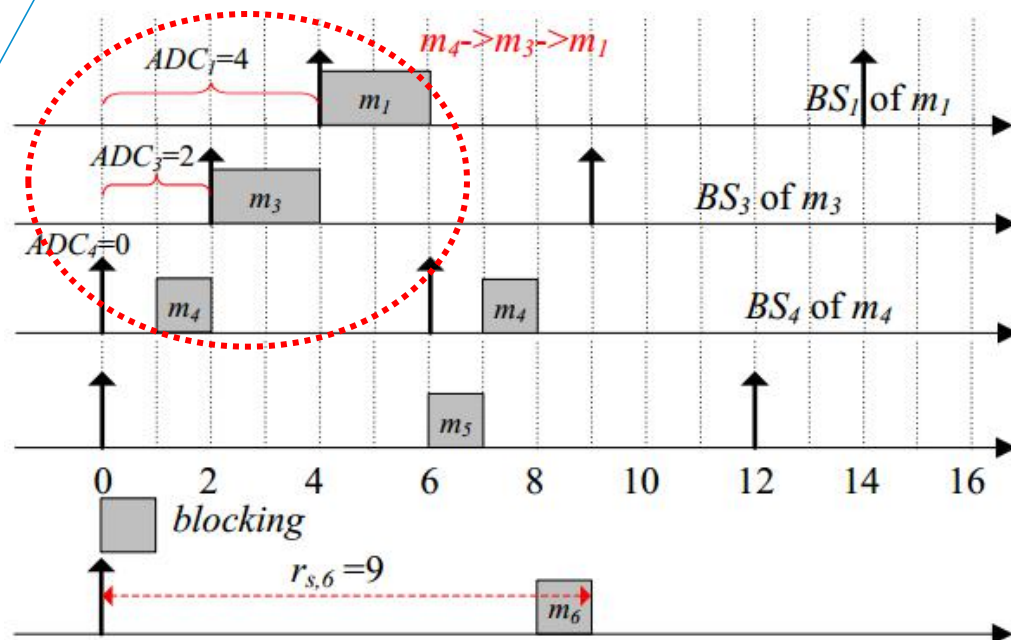
网关互连CAN网络（考虑时间间隔）



以 m_6 为例:

m_1, m_3, m_4 到达 CAN_2 是不同步的, 要考虑时间间隔 ADC_k

是个搜索问题



(b) ADC_k is considered for $dhp_{GW}(i)$ messages

实验结果

计算公式:

$$w_{s,i}^{n+1} = B_{s,i} + \sum_{j \in shp(i)} \left[\frac{w_{s,i}^n}{T_j} \right] \cdot C_j + \sum_{k \in dhp_{GW}(i)} \left[\frac{w_{s,i}^n + r_{s,k} - C_k - ADC_k}{T_k} \right] \cdot C_k$$

测试用例实验结果:

| 消息 | 不考虑间隔 | | | 考虑时间间隔 | | |
|----|-------|-----|-----|--------|-----|-----|
| | sou | des | e2e | sou | des | e2e |
| m1 | 4 | 4 | 8 | 4 | 4 | 8 |
| m2 | 5 | 0 | 0 | 5 | 0 | 0 |
| m3 | 7 | 6 | 13 | 7 | 6 | 13 |
| m4 | 7 | 6 | 13 | 7 | 6 | 13 |
| m5 | 7 | 8 | 15 | 7 | 8 | 15 |
| m6 | 11 | 10 | 21 | 9 | 10 | 19 |
| m7 | 16 | 0 | 0 | 12 | 0 | 0 |
| m8 | 12 | 0 | 0 | 12 | 0 | 0 |

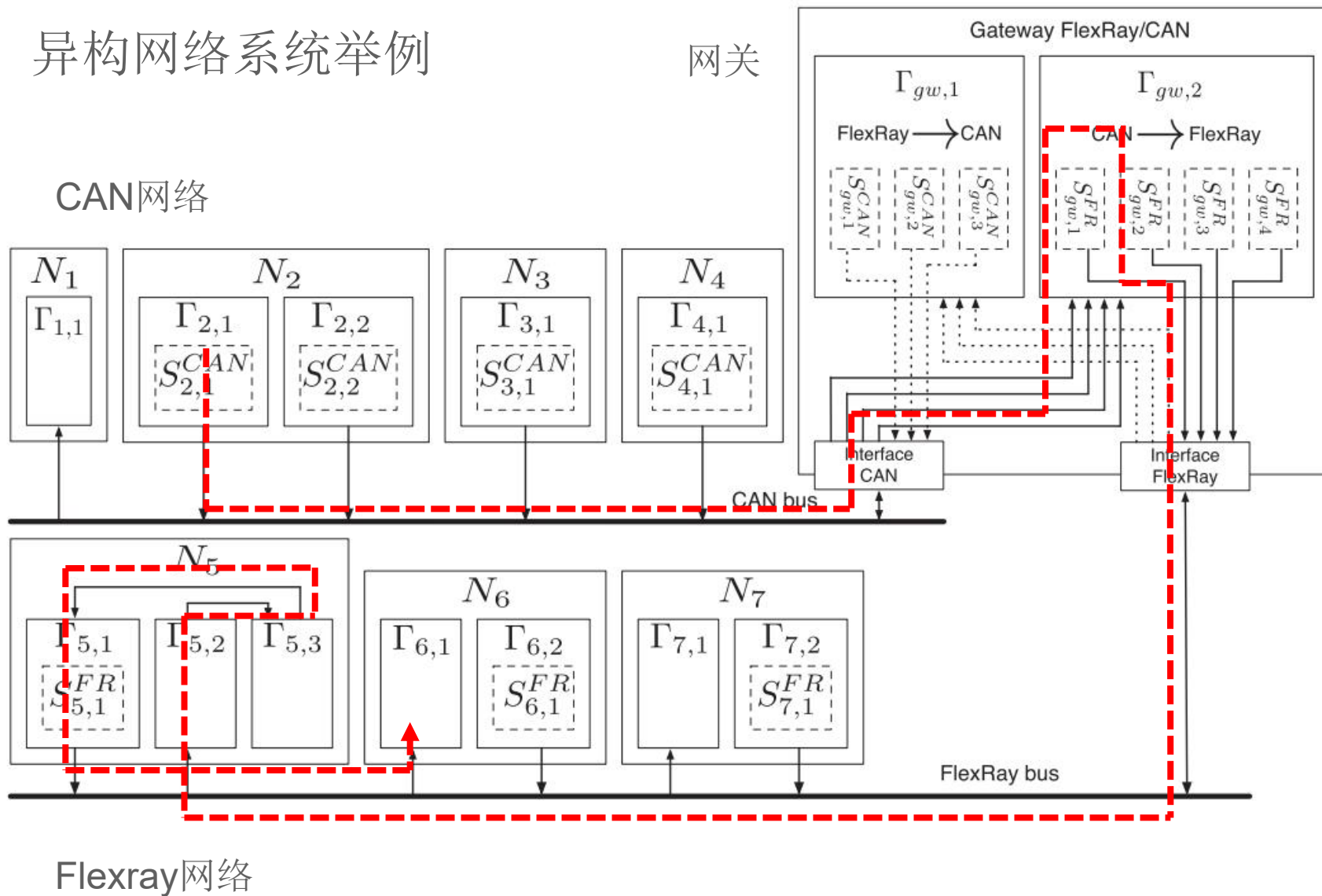
这个结果还有继续优化的空间（和真实的WCRT相比还是悲观了），但怎么优化是个难题。

异构网络模型

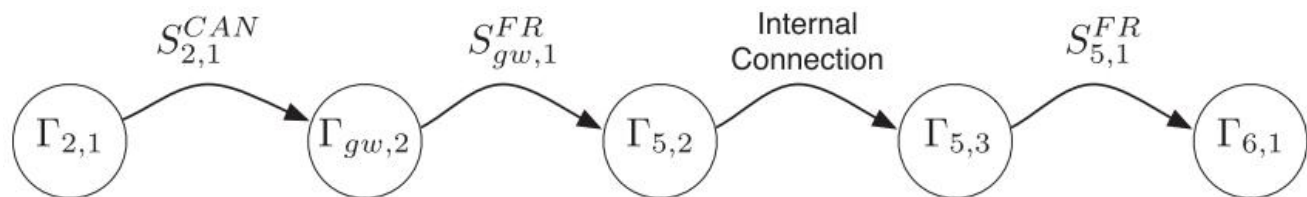
- 一个异构网络系统，主要包括以下几个组件：
 - Network：CAN、FlexRay总线
 - Node：ECU控制
 - Gateway：不同协议网络连接，数据转换
 - Message：节点之间发送的消息

异构网络系统

异构网络系统举例



○Flow的概念



○表示: $\varphi_1 = \{ \Gamma_{2,1}, S_{2,1}^{CAN}, \Gamma_{gw,2}, S_{gw,1}^{FR}, \Gamma_{5,2}, \Gamma_{5,3}, S_{5,1}^{FR}, \Gamma_{6,1} \}$

○分析方法: 计算Flow中的每个元素的最坏响应时间, 相加得出整个Flow的最坏响应时间。

○需要解决的问题:

○单个子网络内的时间分析与计算

○网关内的消息处理

下阶段任务

- 分析异构网络的消息最坏响应时间
- 另外，继续思考怎么优化CAN网络的分析结果



THANKS