2_4_Parallel Mixed Model

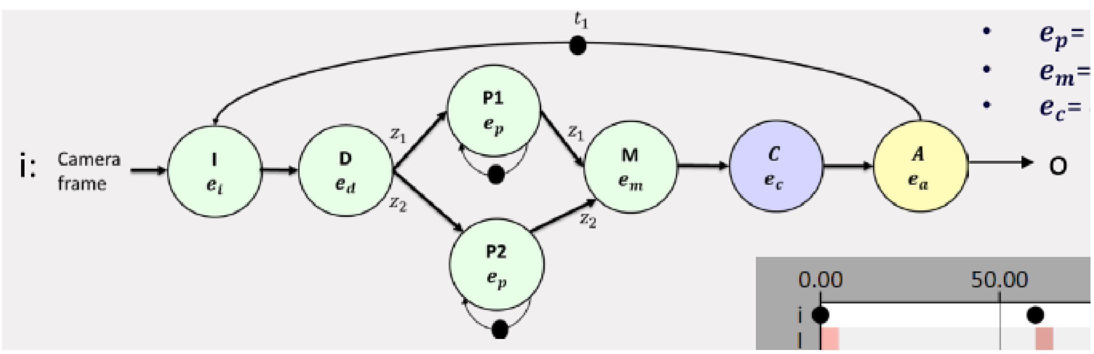
1. Timing Analysis
Equally Distributed Example:
Unequal Distributed Example:
2. Controller Design
Summary

For parallel-mixed model, we will still use the example IBC system used in last part. But now, we assume we may use multiple core or we can build an pipeline structure.

- Long sensor-to-actuator delay can degrade the control performance (e.g., longer settling time) if the controller is implemented sequentially
- Parallel processing the sensing operation is a way tackle this design issue

1. Timing Analysis

Different to Pipeline Model, in parallel mixed model, we only add the number of component to do a single operation step. For example, we add the resources to do the P step. For example, we double the resources to do P:



- Workload z is split into two processors z_1 and z_2

Equally Distributed Example:

$$z = 6, z_1 = 3, z_2 = 3$$

- We just compute the τ as before, but this time, the P part will actually cost 1/2 time compared to original solution. So the $\tau = 59$ ms.
- $h = \lceil \frac{\tau}{f_h} \rceil f = 60$

Unequal Distributed Example:

$$z = 6, z_1 = 4, z_2 = 2$$

- We just compute the τ as before, but this time, the P part will actually cost 2/3 time (we should take the longer path time), compared to original solution. So the $\tau = 69$ ms.
- $h = \lceil \frac{\tau}{f_h} \rceil f = 70$

2. Controller Design

Then we will have a system in which $\tau < h$, we can build a control system which has "small delay"

Summary

- After get the processing step of the task, we can use resources to **speed up some critical part** of the test:
  - $\tau < h$ as normal augmented state space
  - we are actually shorten the sensor-to-actuator delay