Data Freshness Over-engineering

Goal

- Problem statement:

  “Ensure that consumed data meets predetermined freshness guarantees.”

  Where “freshness” represents the age of the latest published result.

- Example: Task X needs to consume speed data produced by task Y that is as most 100ms old

Definitions

- Given job i of task A:
  - P_A = period of task A
  - r_A^i = release time
  - f_A^i = finish time
  - D_A = deadline relative to r_A^i (same for each job)
  - E_A = Execution time, including scheduling
- Given two tasks A and B, where B wants "fresh" data:
  - $d_{A \rightarrow B}$ = [desired] upper bound on data staleness

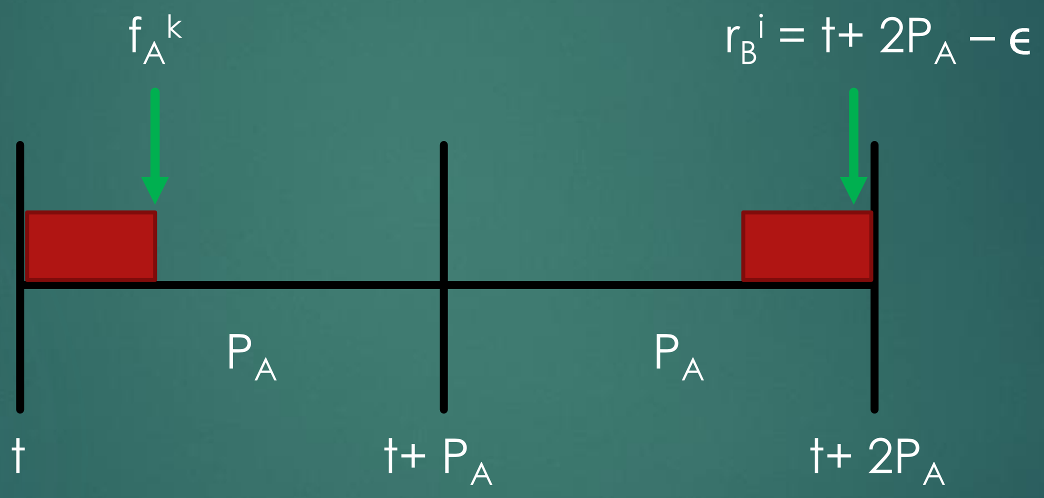
Requirement

- Given: E_A and $d_{A\rightarrow B}$, determine P_A so that:
- $(r_B^i - f_A^{K(r_B^i)}) \leq d_{A\rightarrow B}$ for all i

  where $K(r_B^i)$ = the most recently competed job of A before r_B^i

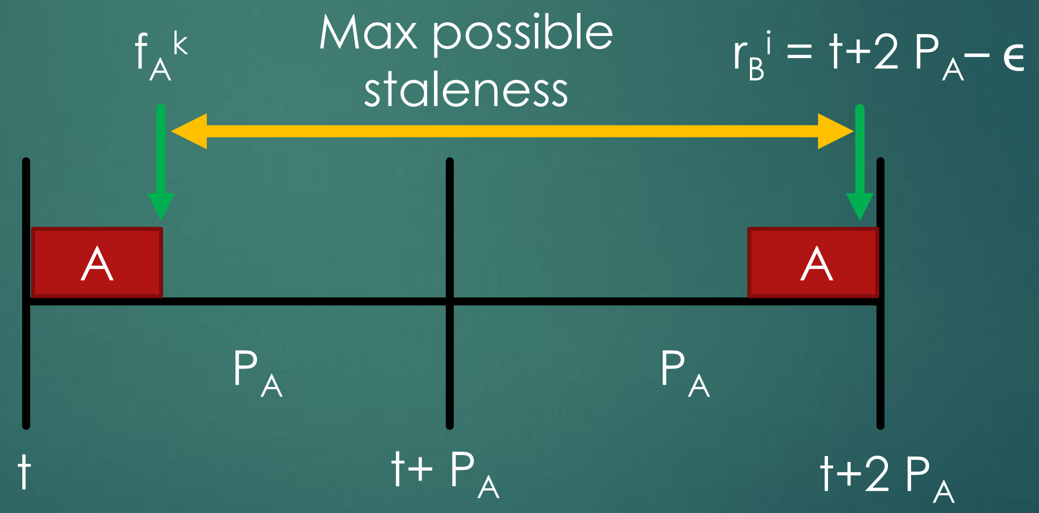
  and i represents the i^{th} task of B
- i.e. find the period of task A so that the age of the data consumed by any task of B is at most $d_{A\rightarrow B}$
- Assumption: data is published at the end of tasks, consumed at beginning (i.e., data must be published by consumer release)

Maximum Wait / Starvation Scenario



Two Task Result

- $d_{A \to B} \leq 2P_A - \epsilon - E_A$
- $d_{A \to B} \leq 2P_A - E_A$



- $P_A \leq \frac{(d_{A \to B} + E_A)}{2}$

Lemma (From Previous Slide)

Given E_A and $d_{A \to B}$, set $P_A \leq \frac{(d_{A \to B} + E_A)}{2}$

Note: doesn't guarantee schedulability under any particular scheduler and P_B is irrelevant.

Multiple (3) Task Scenario

- Now we have task A, B, and C
  - Each has the properties defined before in the two-task case
- A produces output that B consumes
- B produces output that C consumes
- Want a freshness bound from for the data from task A to task C

Requirement

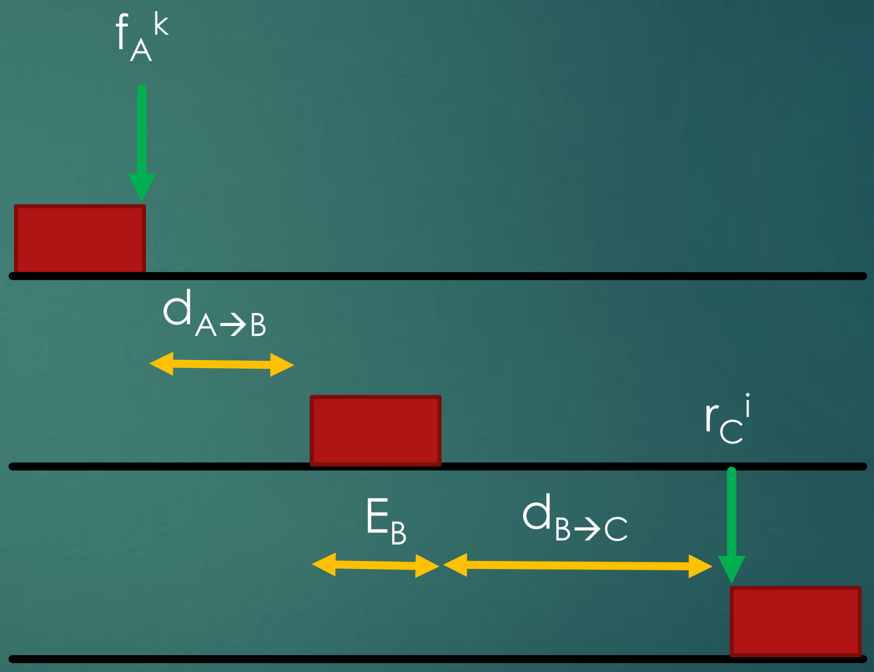
- $A \rightarrow B \rightarrow C$
- $(r_C^i - f_A^K) \leq d_{A \rightarrow C}$
- i.e., the age of the data consumed by C was derived from data from A that is at most $d_{A \rightarrow C}$ old
  - No constraint on when B is executed between them

Illustration

- $d_{A \to C} = d_{A \to B} + E_B + d_{B \to C}$

By Lemma…

- $P_A \leq \frac{(d_{A \to B} + E_A)}{2}$
- $P_B \leq \frac{(d_{B \to C} + E_B)}{2}$



Optimization

- Constraints: $d_{A \rightarrow C}$, E_A, and E_B are given.
- Free variables: $d_{A \rightarrow B}$, $d_{B \rightarrow C}$, P_A, and P_B
- Irrelevant: P_C

- Goal: Minimize utilization from producing tasks $\left(\frac{E_A}{P_A} + \frac{E_B}{P_B}\right)$
  - Common optimization objective

Solution

- Minimize: Utilization $= \frac{E_A}{P_A} + \frac{E_B}{P_B} \geq \frac{2E_A}{d_{A\to B}+E_A} + \frac{2E_B}{d_{B\to C}+E_B}$
  - Minimize over P_A and P_B using derivative = 0
- Solution: $P_A = \frac{(d_{A\to C}+E_A)}{2\left(1+\sqrt{\frac{E_B}{E_A}}\right)}$

$$P_B = \frac{(d_{A\to C}+E_A)}{2\left(1+\sqrt{\frac{E_A}{E_B}}\right)}$$

Problems

- Doesn’t account for scheduling or prioritization.
- Does not guarantee “solution” is schedulable by a particular, or any, scheduling algorithm.
- Doesn’t account for period of consuming process.

- These are some of the limitations that likely make such over-enginnering methods very pessimistic.