Effectiveness of Delaying Timestamp Computation

Sandeep Kulkarni
Michigan State University

Nitin Vaidya
University of Illinois

System Model

- Asynchronous system
- n processes
- Pairwise message-passing channels
- Unicasts
- Incomplete network

Goal

Assign timestamp T_e to each event e

such that

$$T_e < T_f \quad \text{iff} \quad e \rightarrow f$$

Much Related Work

- Message-passing
  - Vector timestamps [Fidge-Mattern 1988]
  - Vector lower bound [Charron-Bost 1991]
  - Lower bound [Melideo 2001]
  - Synchronous messages [Garg 2002]
  - Causal separators [Rodriguez 1995]
  - Exploiting locality [Meldal 1999]
  - Plausible clocks [Torres-Rojas 1999]
  - Cluster timestamps [Ward 2001]
  - …
- Shared memory
  - Lazy replication [Ladin 1992]
  - SwiftCloud [Zawirski 2015]
  - Version vectors, dotted version vectors [Almeida 2014]
  - …

Outline

- Vector timestamp bounds
- Alternate solution

Vector Timestamps
[Fidge-Mattern 1988]

- Timestamp T_e is a vector
- Vector comparison: $T_e < T_f$ if
  - $T_e[i] \leq T_f[i]$ for all i
  - There exists i such that $T_e[i] < T_f[i]$

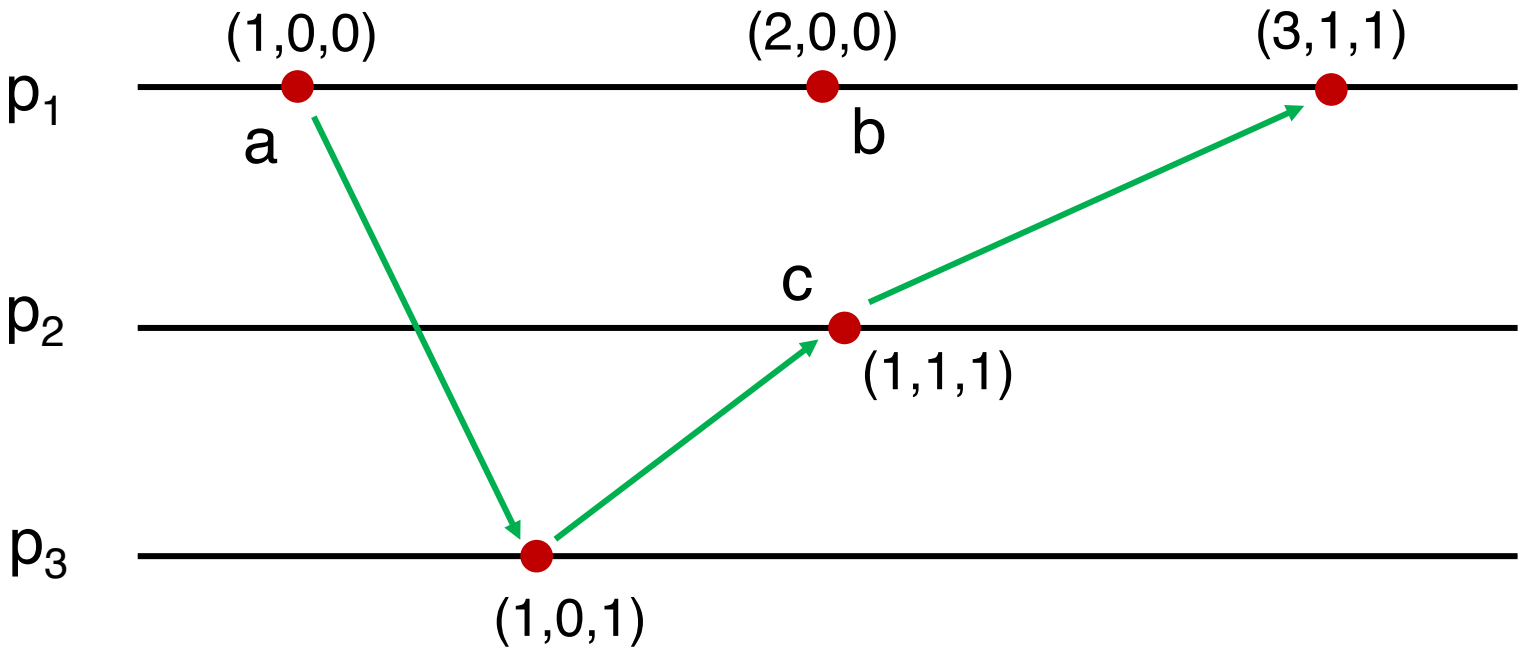
Vector Timestamps
[Fidge-Mattern 1988]

- Timestamp T_e is a vector
- Vector comparison: $T_e < T_f$ if
  - $T_e[i] \leq T_f[i]$ for all i
  - There exists i such that $T_e[i] < T_f[i]$

$(0,1,2) < (0,1,3)$

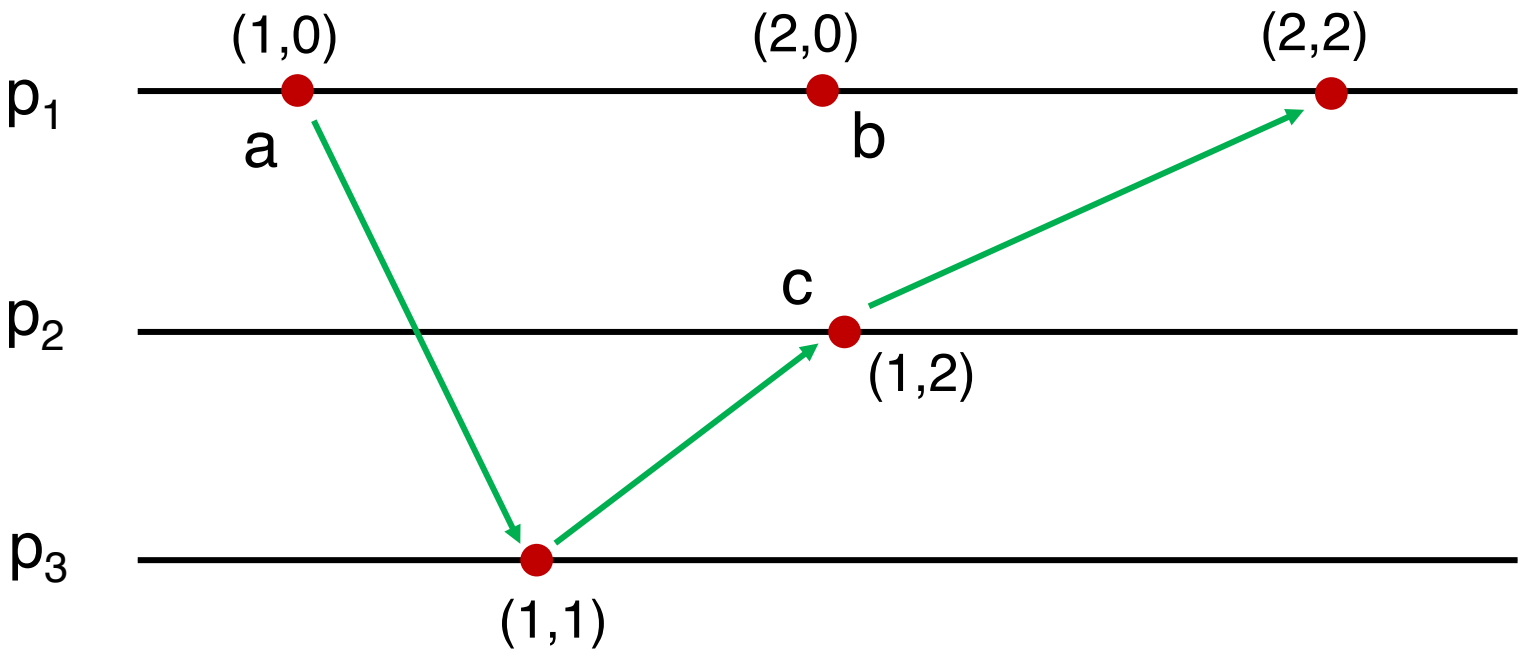
$(0,1,2) \not< (2,1,1)$

Vector Timestamps



Online algorithm ➔ Assign timestamp when event occurs

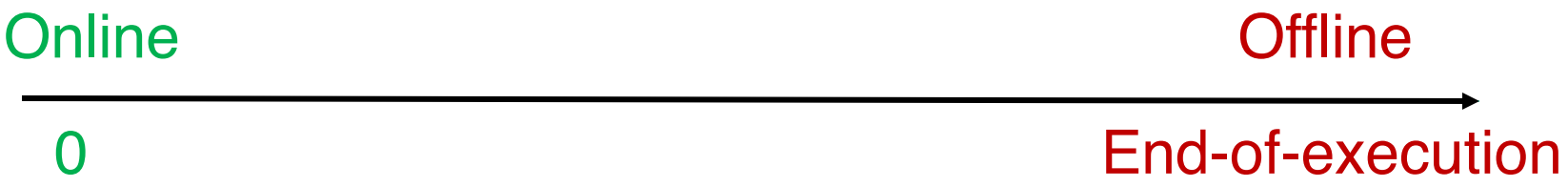
Offline Vector Timestamps



Offline algorithm can often reduce timestamp size

Delay versus Timestamp Size

Delay = Time between event & timestamp assignment



Lower Bound [Charron-Bost 1991]

- Worst-case bound: Vector length n
- Assuming complete communication graph

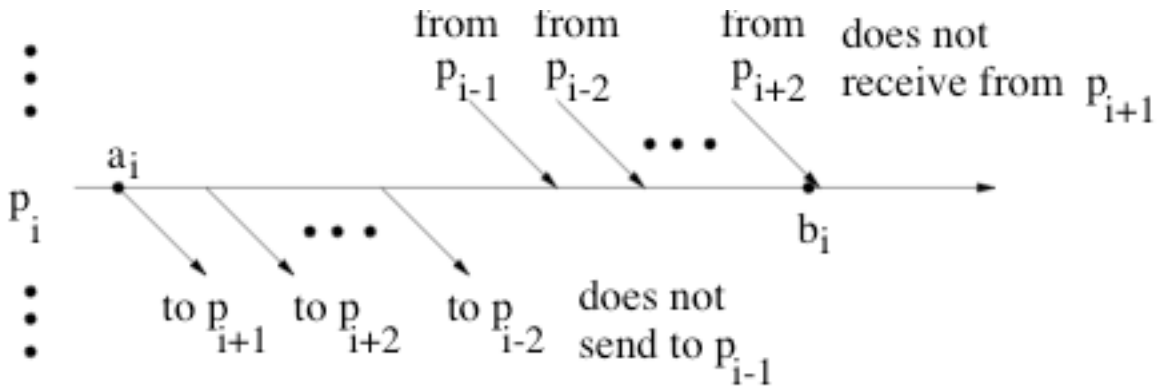


Figure from [Attiya-Welch]

Reduce online vector timestamp

- Exploit network topology ?

Vector Timestamps Lower Bounds

Star Graph		Connectivity	
Real-valued	Integer-valued	>1	1
n-1	n	n	n - Z

Z = #minimal cuts

Vector Timestamps Lower Bounds

Star Graph		Connectivity	
Real-valued	Integer-valued	>1	1
n-1	n	n	n - Z

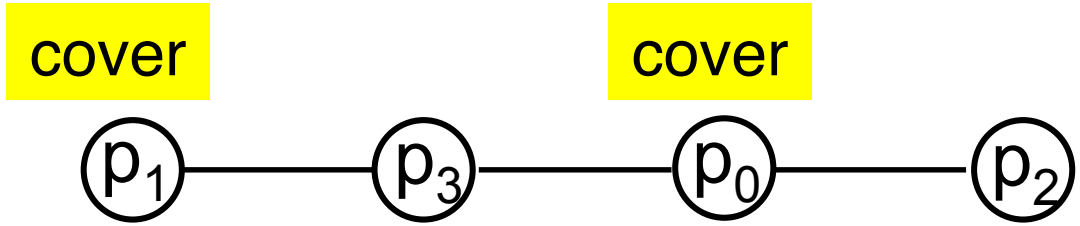
Z = network parameter

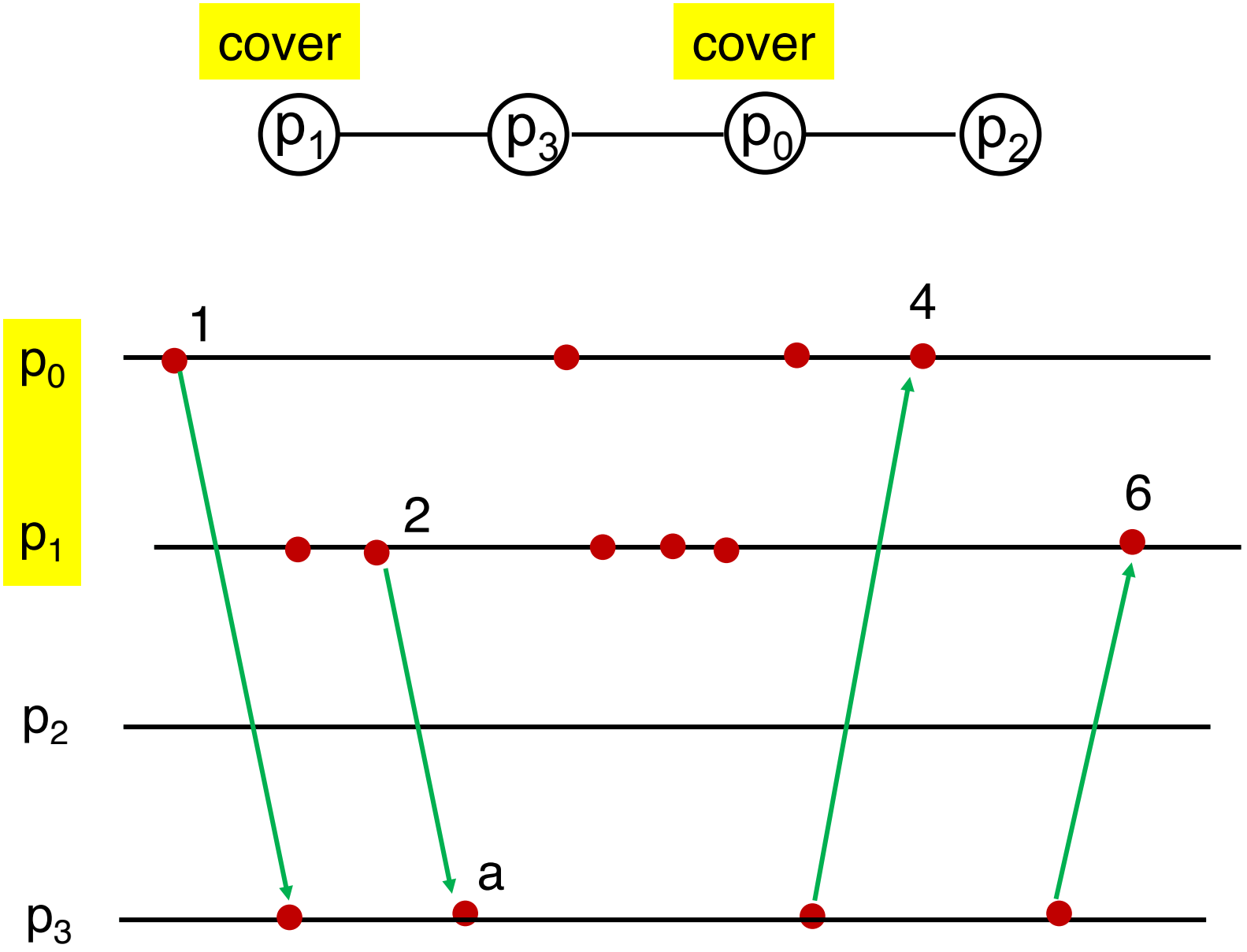
Outline

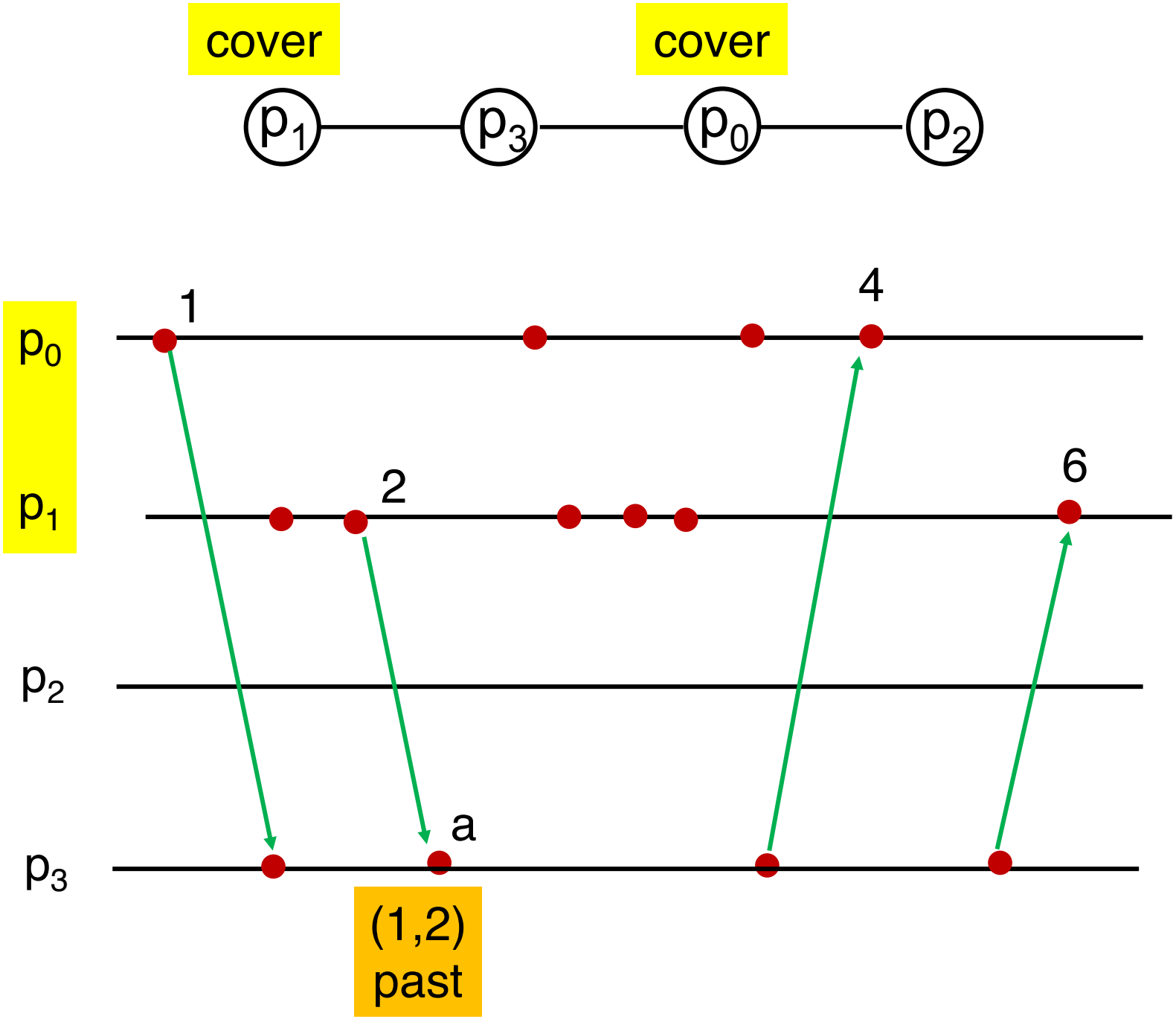
- Vector timestamp bounds
- Alternate solution

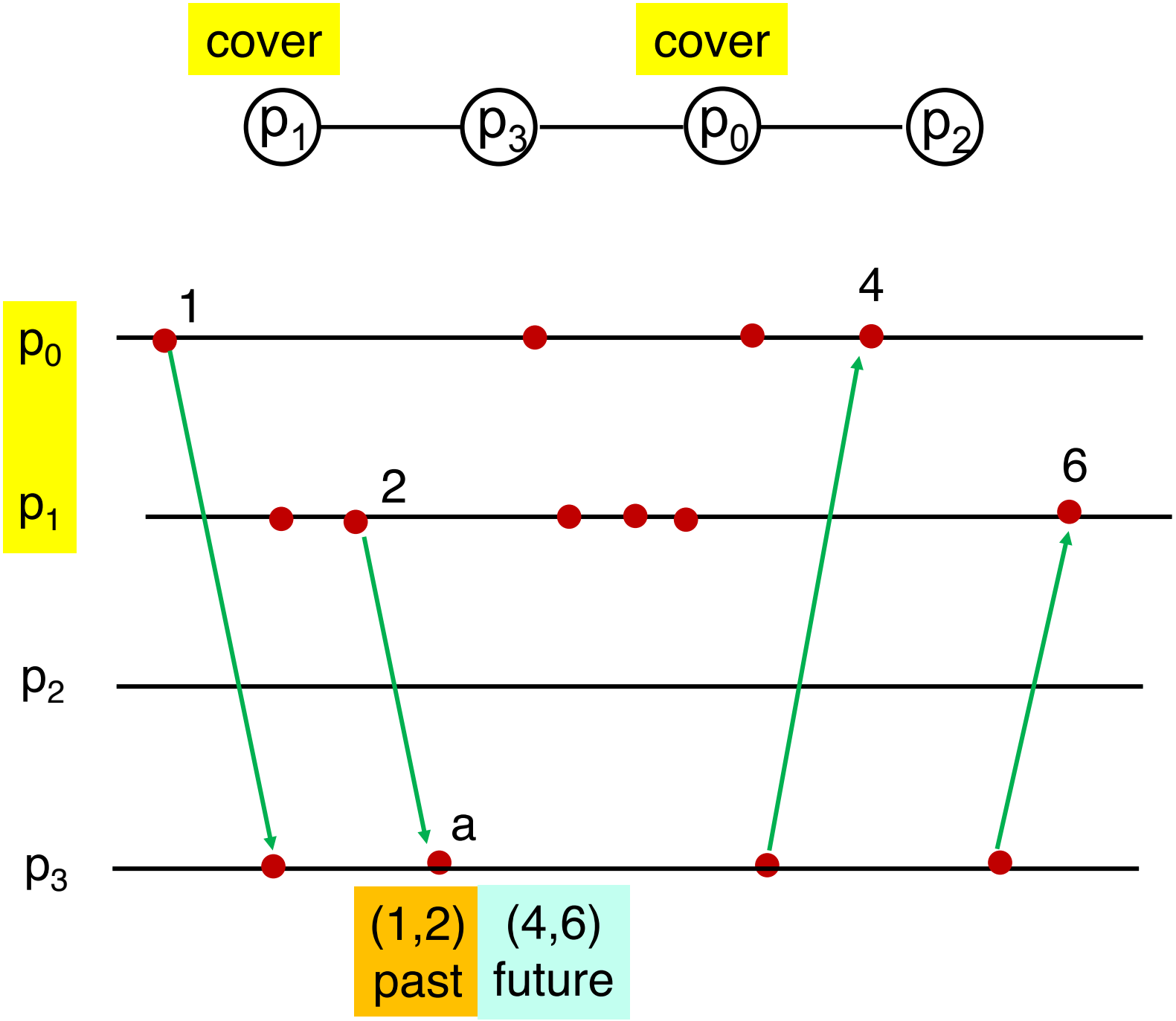
Alternative Solution

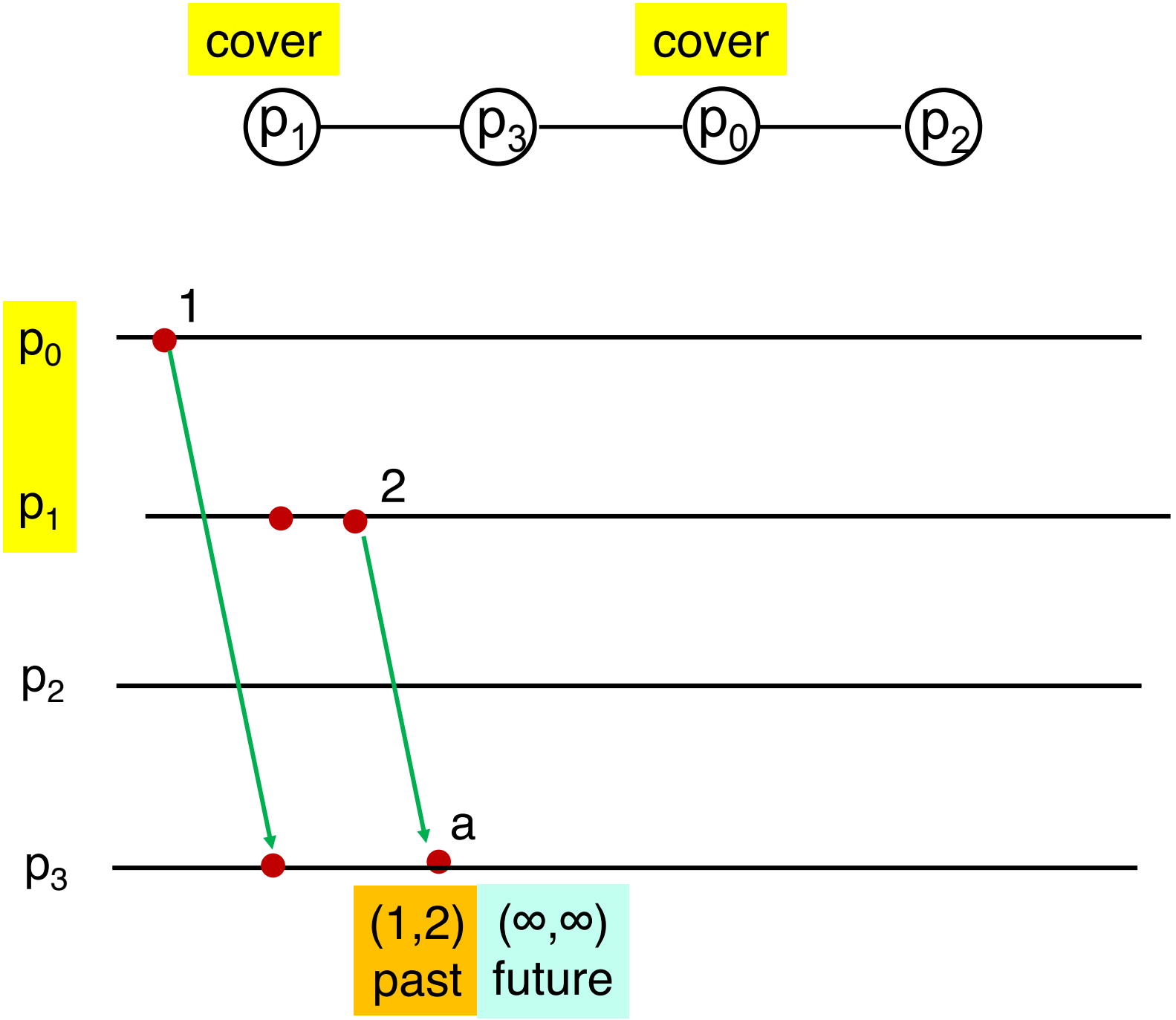
- Events at processes in a chosen vertex cover used to order all events

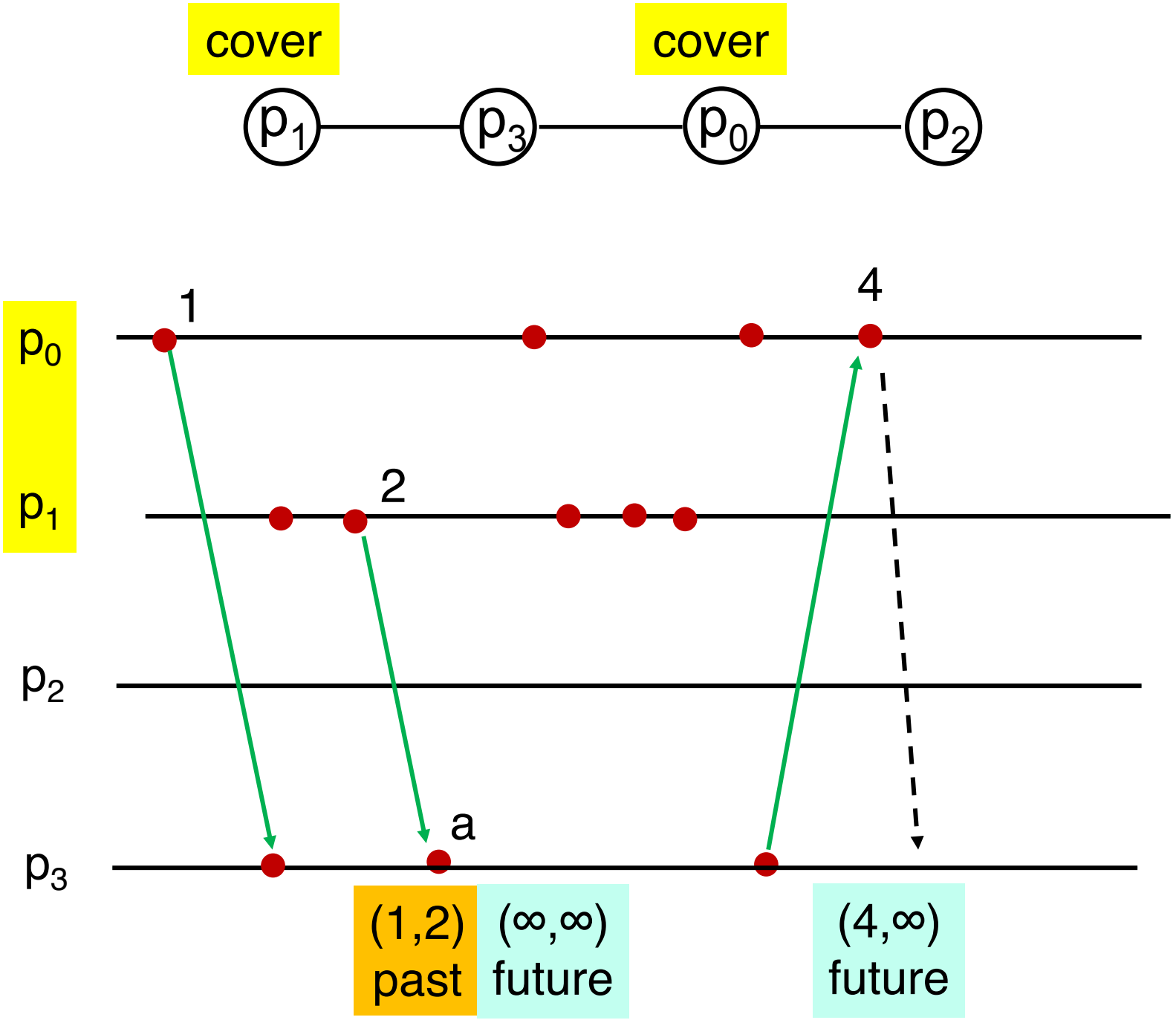


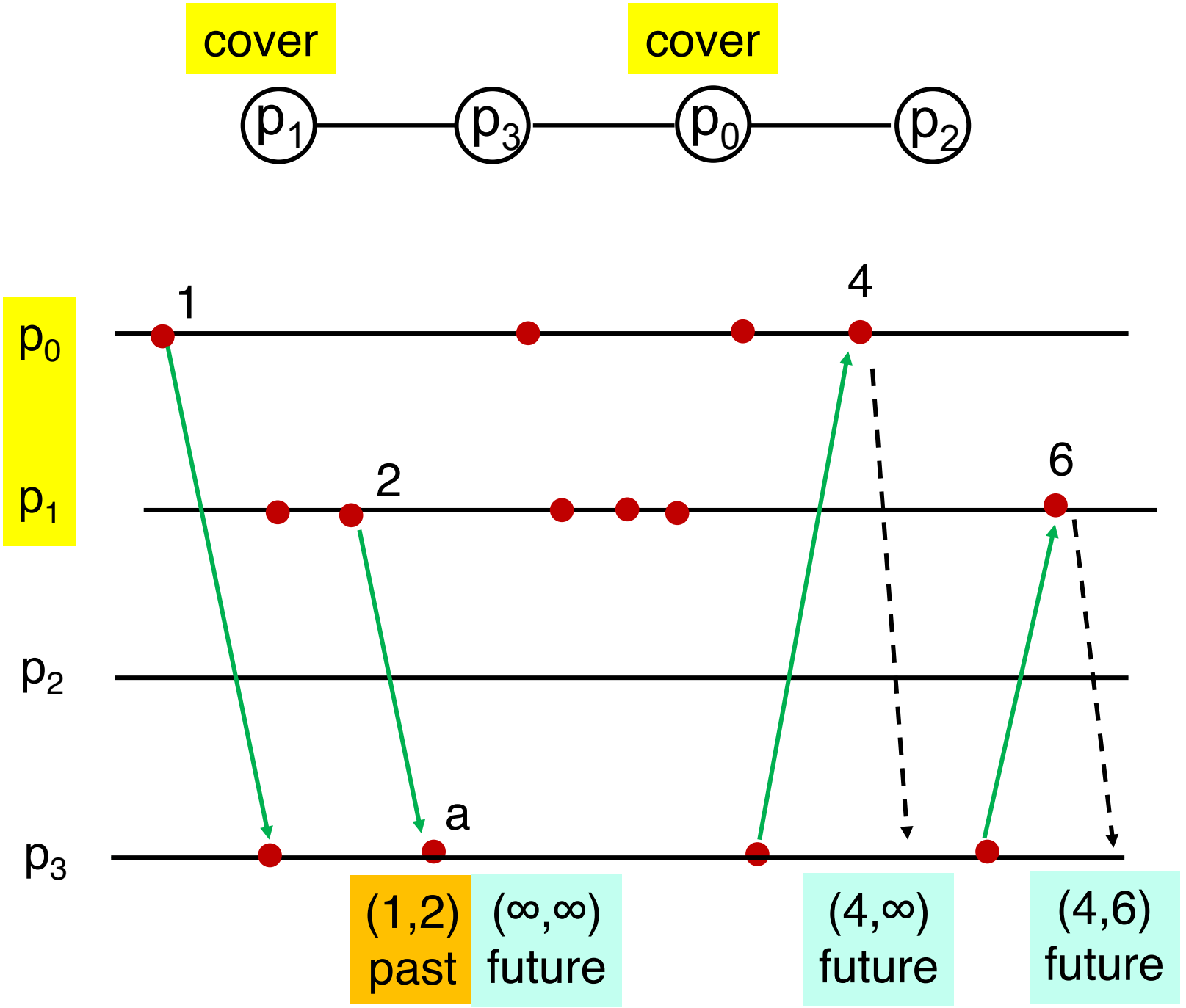

cover
cover
p_1
p_3
p_0
p_2
p_0
p_1
p_2
p_3
1
2
a
4
6

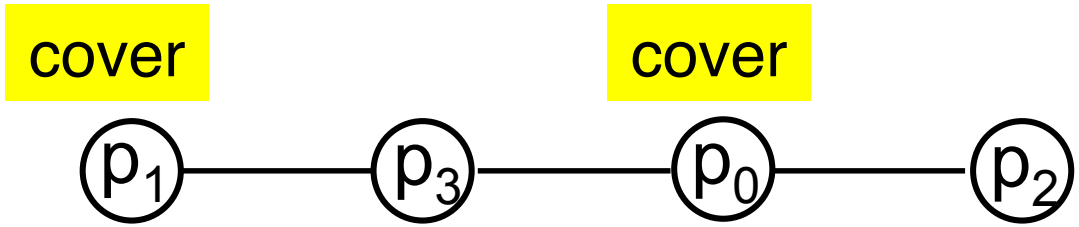

cover
cover
p_1
p_3
p_0
p_2
p_0
p_1
p_2
p_3
1
2
4
6
a
(1,2)
past

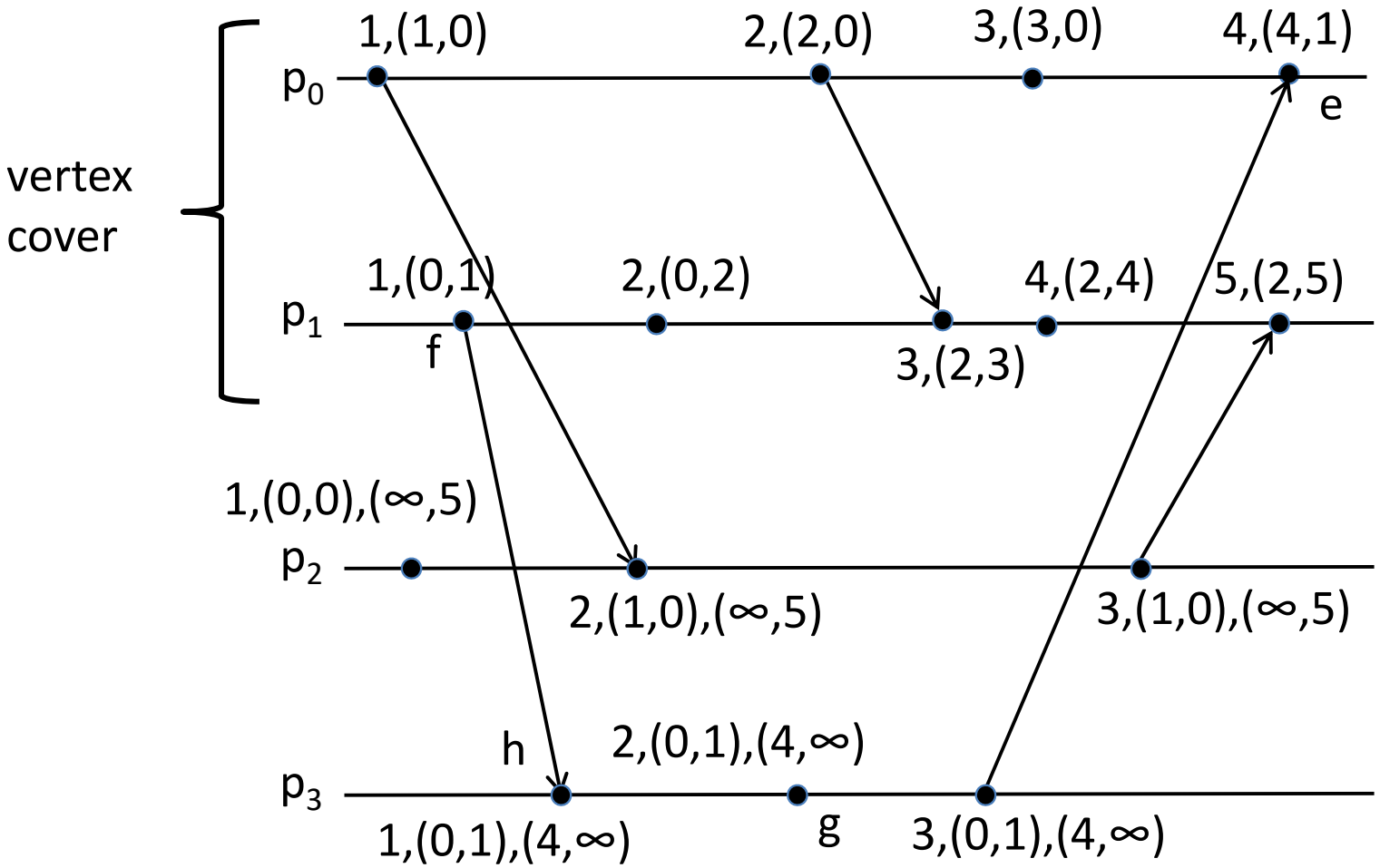

cover
cover
p_1
p_3
p_0
p_2
p_0
p_1
p_2
p_3
1
2
a
4
6
(1,2)
past
(4,6)
future


cover
cover
p1
p3
p0
p2
p0
p1
p2
p3
1
2
a
(1,2)
past
(∞,∞)
future


cover
cover
p1
p3
p0
p2
p0
p1
p2
p3
1
2
4
a
(1,2)
past
(∞,∞)
future
(4,∞)
future


cover
cover
p1
p3
p0
p2
p0
p1
p2
p3
1
2
a
4
6
(1,2)
past
(∞,∞)
future
(4,∞)
future
(4,6)
future


cover
cover
p_1
p_3
p_0
p_2


vertex
cover
1,(1,0)
2,(2,0)
3,(3,0)
4,(4,1)
p_0
e
1,(0,1)
2,(0,2)
4,(2,4)
5,(2,5)
p_1
f
3,(2,3)
1,(0,0),(∞,5)
p_2
2,(1,0),(∞,5)
3,(1,0),(∞,5)
h
2,(0,1),(4,∞)
p_3
1,(0,1),(4,∞)
g
3,(0,1),(4,∞)

Happened-Before

Intuitively:

$e \rightarrow f$ iff future(e) $\leq$ past(f)

Alternate Timestamps

- Size proportional to cover size
- Minimum 1 round-trip delay in determining "future" component
  - Potentially much longer

Applications

Smaller timestamps not interesting if the cost of using them is too high

- → easy to verify
- Maximal consistent cuts (with a slightly modified timestamp)
- Predicate detection