CS 181AG
Lecture 20

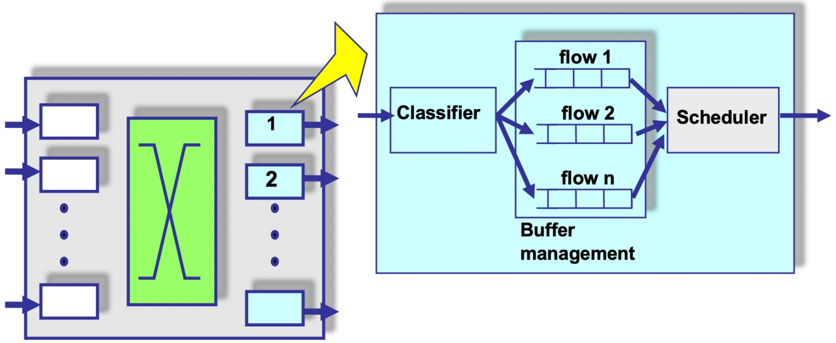
Output Scheduling (cont.)

Arthi Padmanabhan

Nov 21, 2022

Recap

- Within each output queue, there are several flows (set of packets that take the same path through whole network)
- RED discards packets *before* they are placed in the output queue if the running average of queue length > threshold. Why?
- Sometimes we need a way of controlling long-term rate and maximum burst size for a flow -> token bucket algorithm

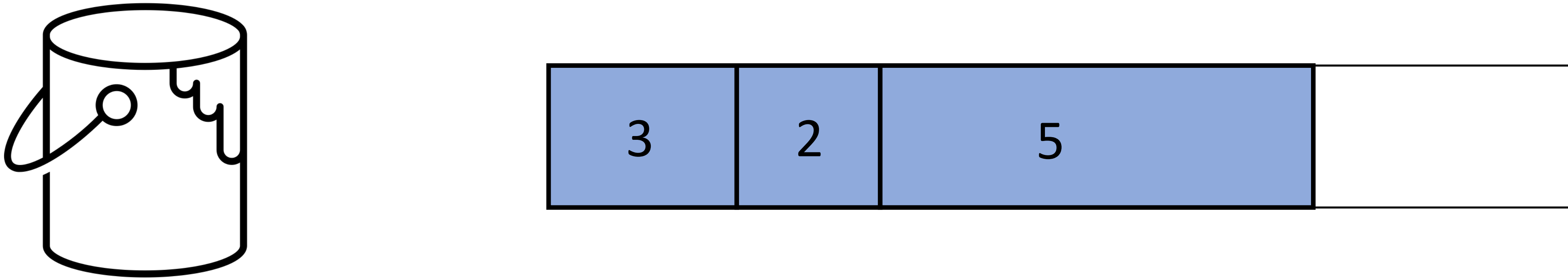


Token Bucket Shaping

- Goal: for a particular flow, limit 1) the long-term rate of sending and 2) the maximum burst size
- Why: we might need to limit the amount of news traffic, or UDP traffic

Token Bucket Example

r = 2 Mbps
b = 5 Mb
1 token = 1 Mb



3

2

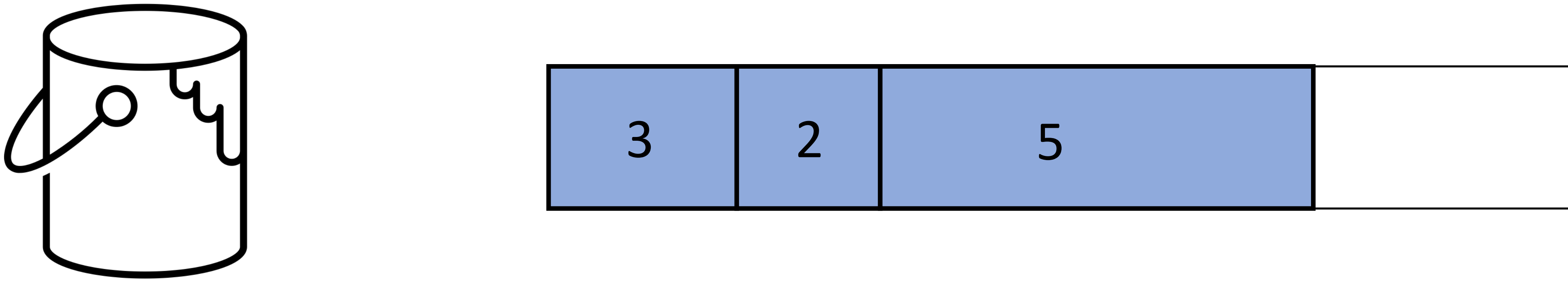
5

Token Bucket

Buffer

Token Bucket Example

r = 2 Mbps
b = 5 Mb
1 token = 1 Mb



r tokens enter bucket
every time slot

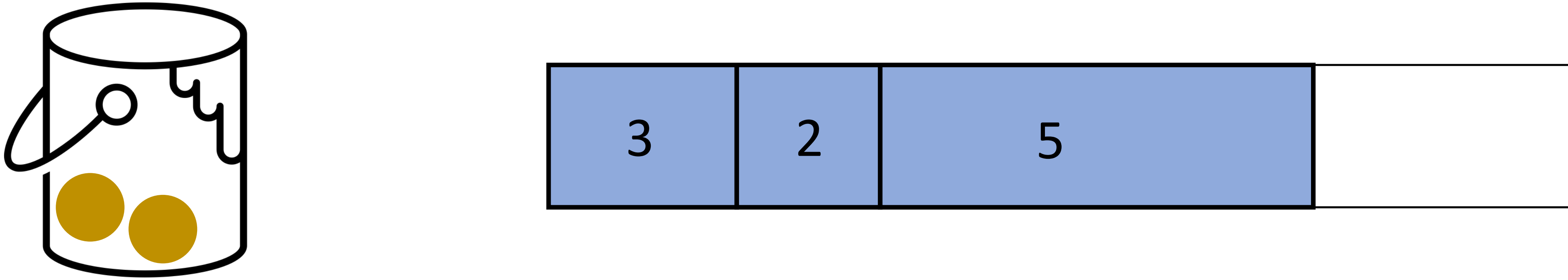
Token can hold
max 5 tokens

Token Bucket

Buffer

Token Bucket Example

r = 2 Mbps
b = 5 Mb
1 token = 1 Mb



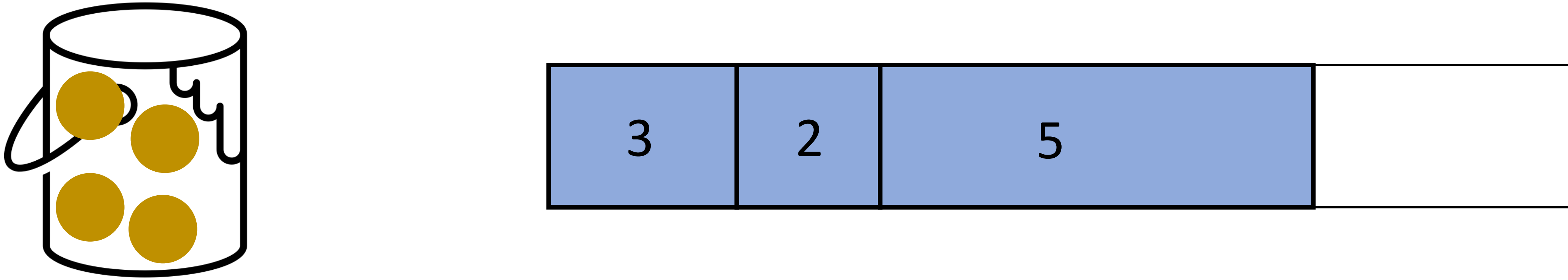
Token Bucket

Buffer

t = 0

Token Bucket Example

r = 2 Mbps
b = 5 Mb
1 token = 1 Mb



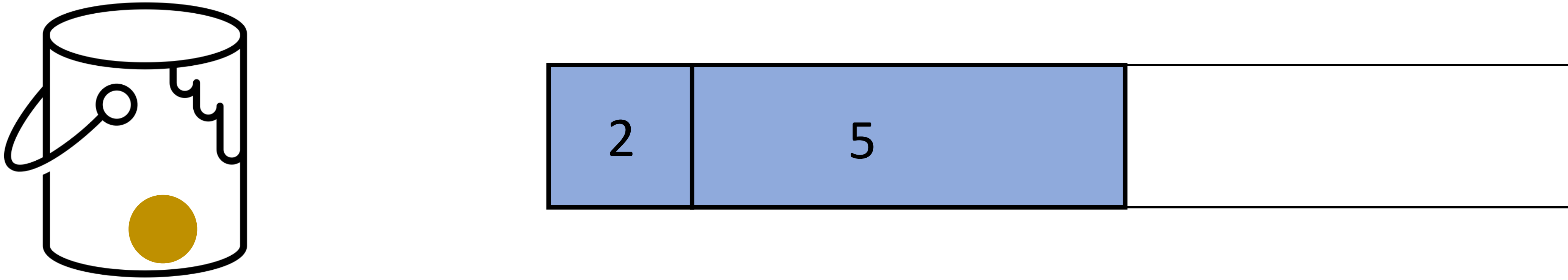
Token Bucket

Buffer

t = 1

Token Bucket Example

r = 2 Mbps
b = 5 Mb
1 token = 1 Mb



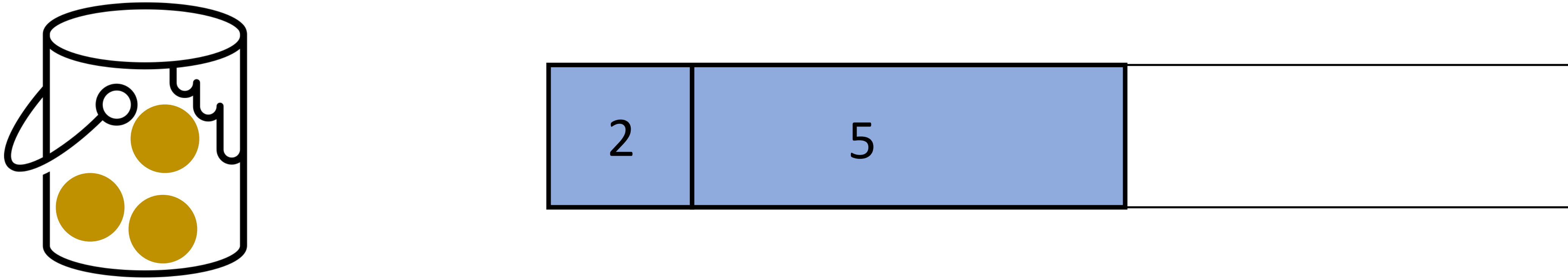
Token Bucket

Buffer

t = 1

Token Bucket Example

r = 2 Mbps
b = 5 Mb
1 token = 1 Mb



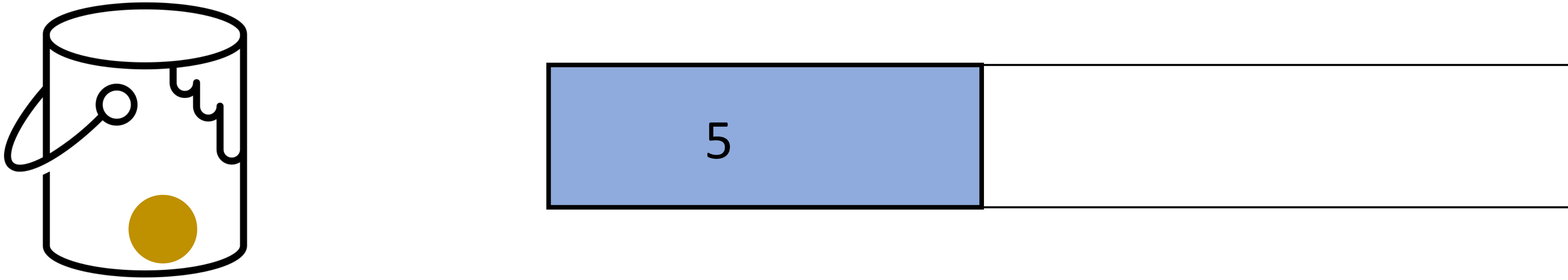
Token Bucket

Buffer

t = 2

Token Bucket Example

r = 2 Mbps
b = 5 Mb
1 token = 1 Mb



5

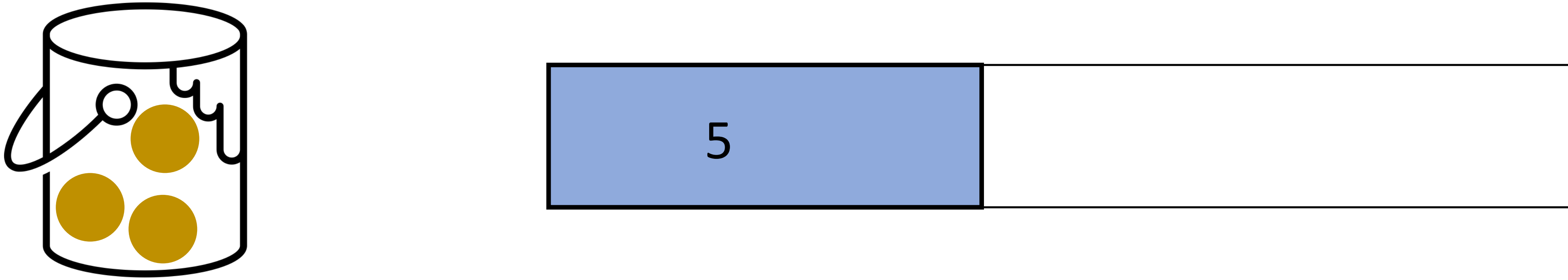
Token Bucket

Buffer

t = 2

Token Bucket Example

r = 2 Mbps
b = 5 Mb
1 token = 1 Mb



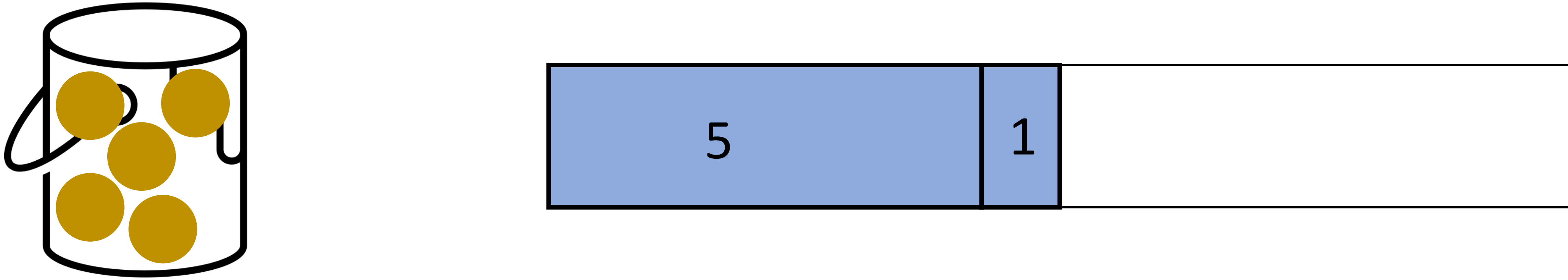
Token Bucket

Buffer

t = 3

Token Bucket Example

r = 2 Mbps
b = 5 Mb
1 token = 1 Mb



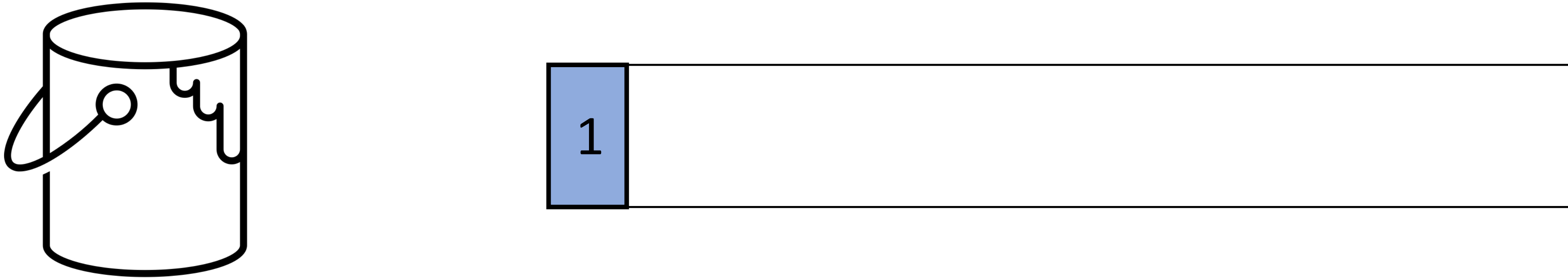
Token Bucket

Buffer

t = 4

Token Bucket Example

r = 2 Mbps
b = 5 Mb
1 token = 1 Mb



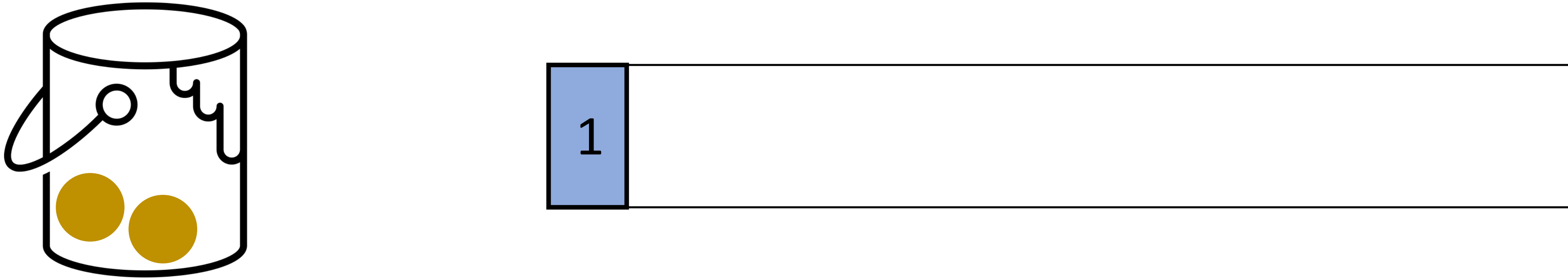
Token Bucket

Buffer

t = 4

Token Bucket Example

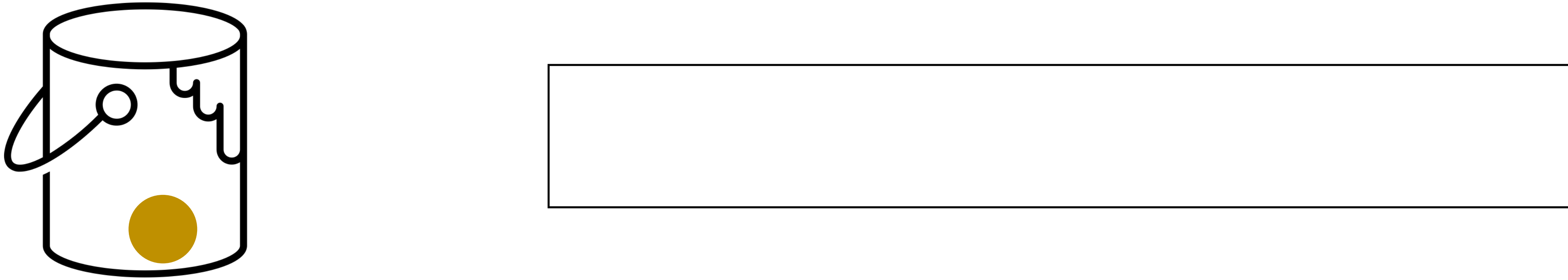
r = 2 Mbps
b = 5 Mb
1 token = 1 Mb



t = 5

Token Bucket Example

r = 2 Mbps
b = 5 Mb
1 token = 1 Mb



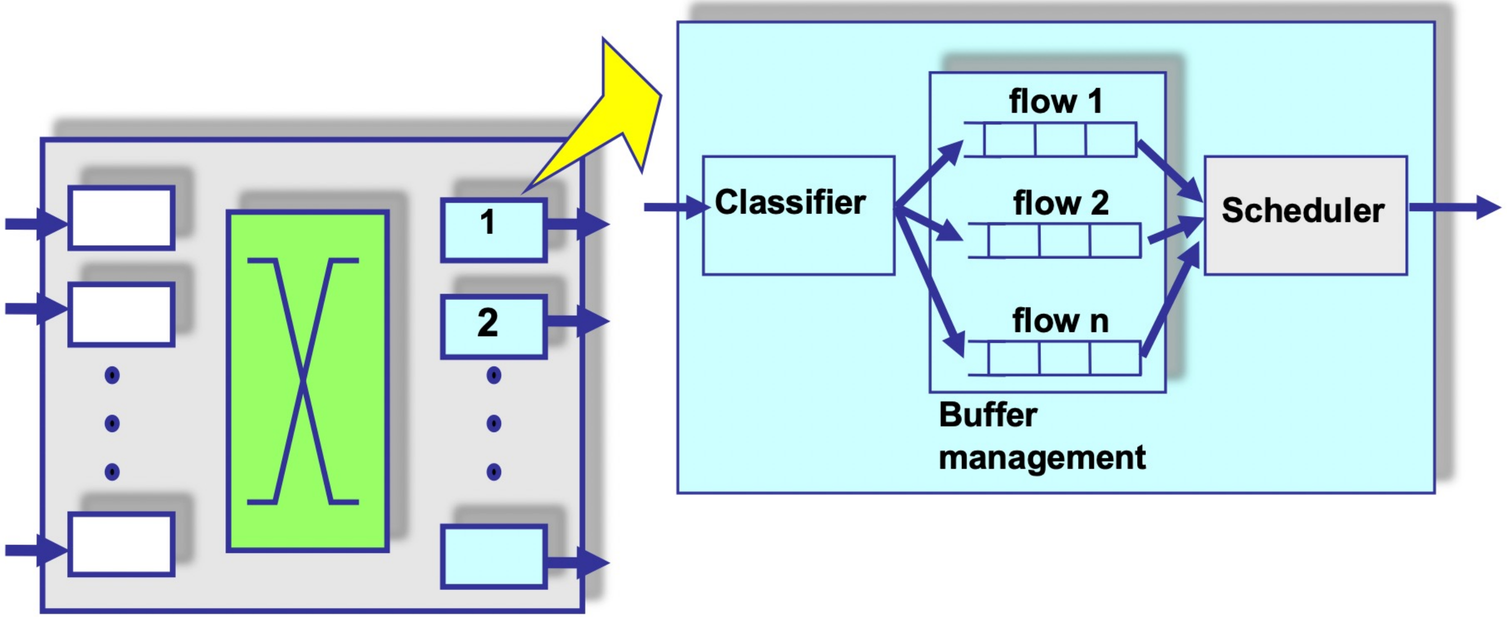
Token Bucket

Buffer

t = 5

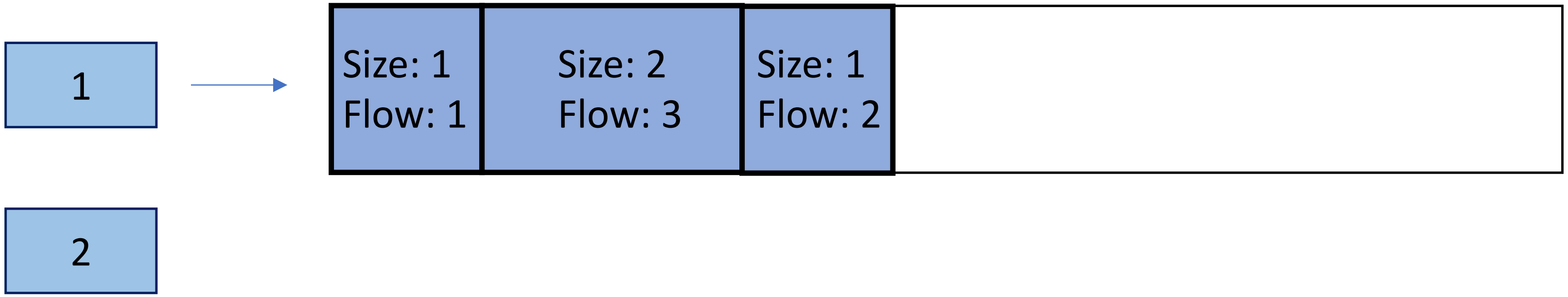
Token Bucket Per Flow

- So far, we have considered token bucket algorithm *per flow*



Token Bucket: One Buffer?

- What would be the issue if we kept all flows in the same buffer?



1

2

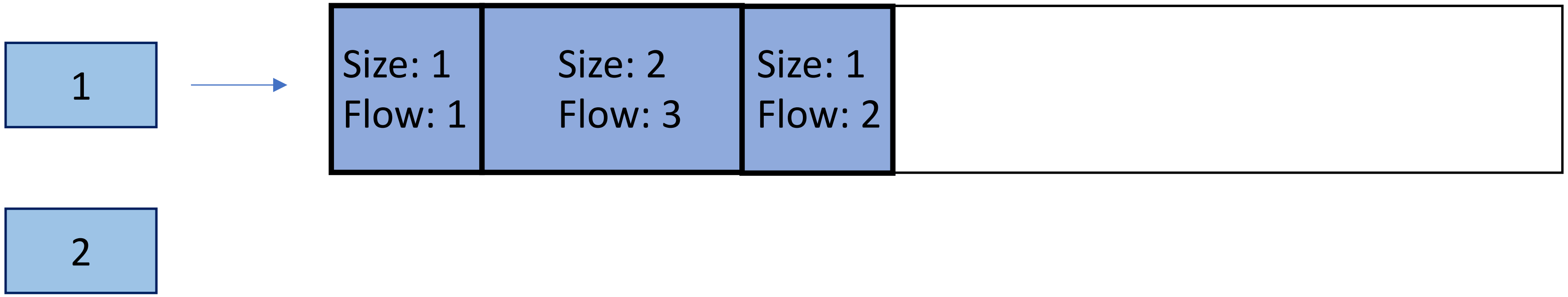
Size: 1
Flow: 1

Size: 2
Flow: 3

Size: 1
Flow: 2

Token Bucket: One Buffer?

- What would be the issue if we kept all flows in the same buffer?



- Waiting for one packet to get enough tokens could block other flows that are able to send
- Solution: drop any packet that has insufficient tokens to send

Token Bucket Shaping vs Policing

- Shaping: one buffer per flow in the output queue; if there are insufficient tokens to send packet at head of buffer, wait
- Policing: one buffer for all packets in the output queue; if there are insufficient tokens to send packet at head of buffer, drop (still keep 1 token bucket per flow)

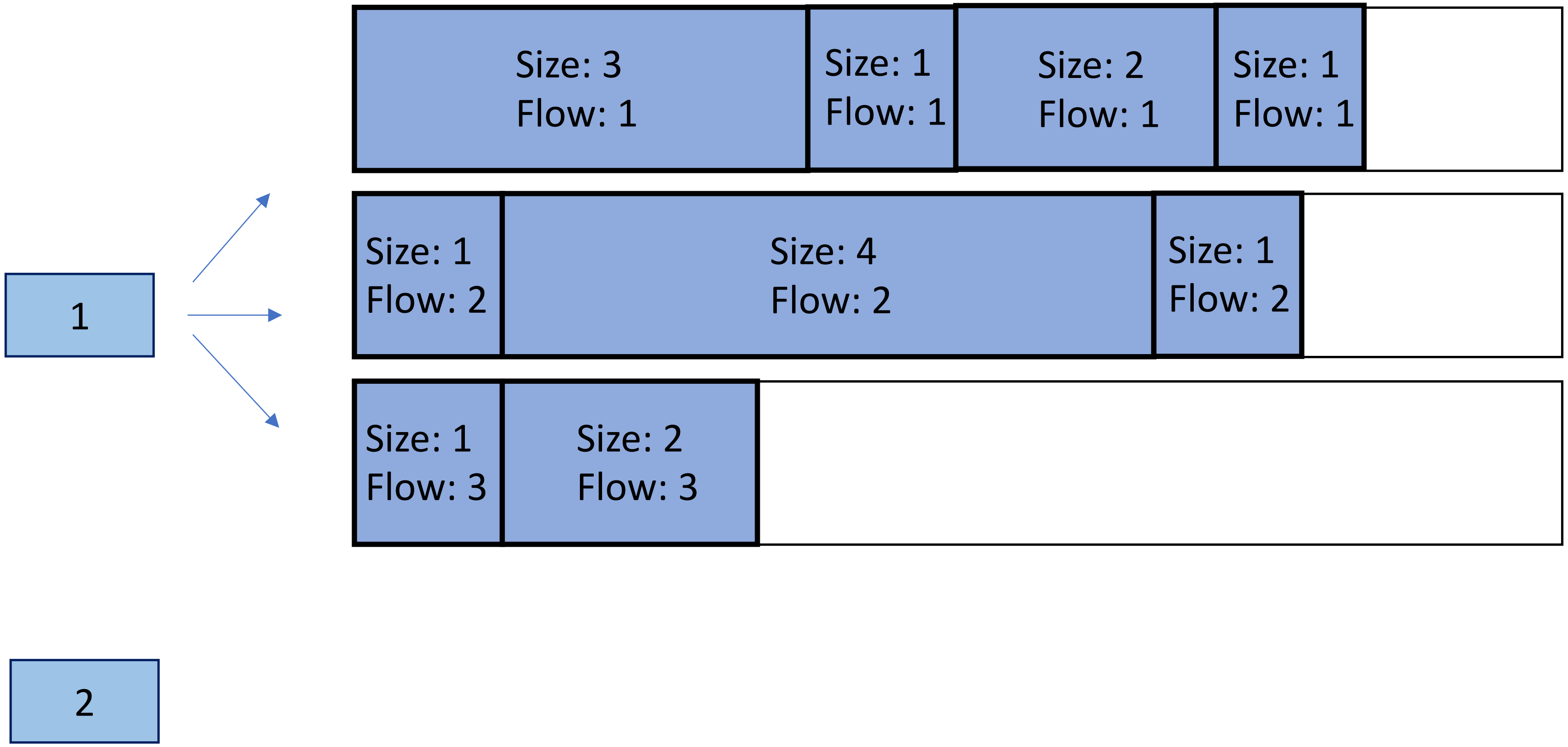
Token Bucket Policing Example

- It takes one unit of time to look at the head of the buffer and decide what to do with it
- Two tokens are added at each time slot and may be used in the same time slot they are added. Token bucket can hold max 8
- Show the head of the buffer and the token count in each bucket at the start of each time step

Size: 2 Flow: 2	Size: 3 Flow: 1	Size: 4 Flow: 1	Size: 4 Flow: 2	Size: 6 Flow: 2	Size: 3 Flow: 1	

Handling Multiple Flows

- In the case where we have multiple flows and we wait, how do we decide which flow to send from?



Size: 3 Flow: 1

Size: 1 Flow: 1

Size: 2 Flow: 1

Size: 1 Flow: 1

Size: 1 Flow: 2

Size: 4 Flow: 2

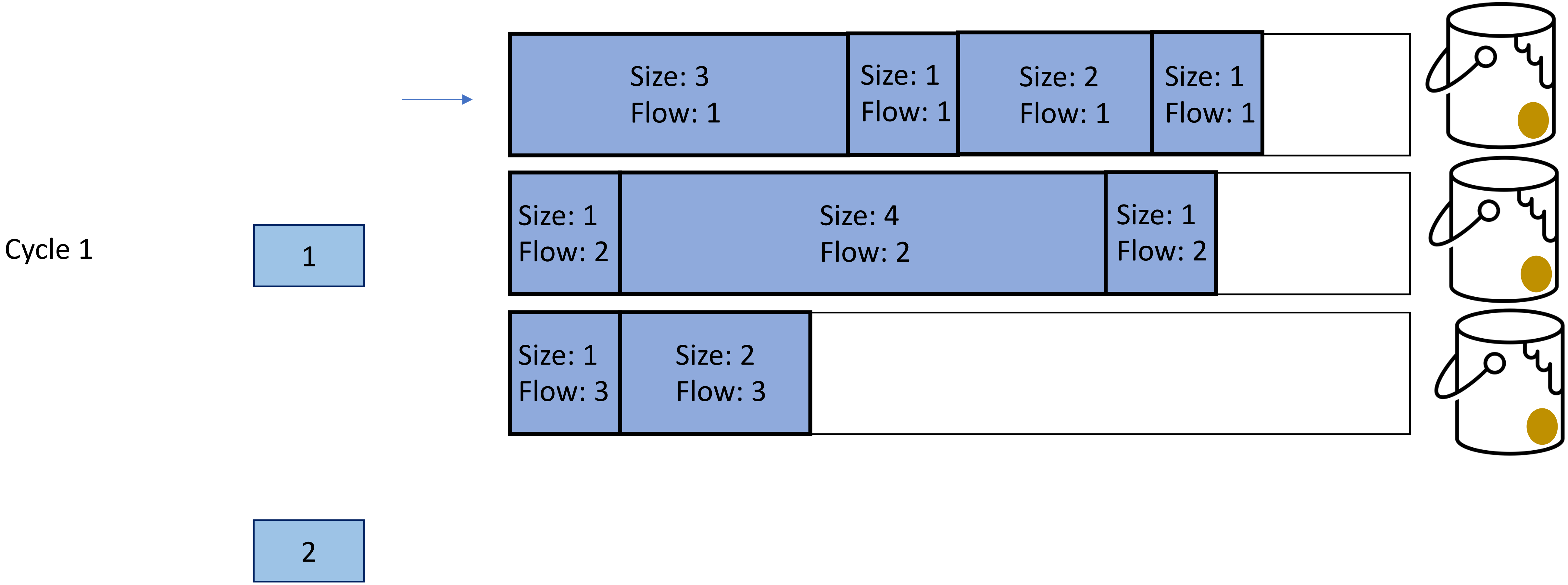
Size: 1 Flow: 2

Size: 1 Flow: 3

Size: 2 Flow: 3

Deficit Round Robin

Salary = 1 token



Cycle 1

1

2

Size: 3
Flow: 1

Size: 1
Flow: 1

Size: 2
Flow: 1

Size: 1
Flow: 1

Size: 1
Flow: 2

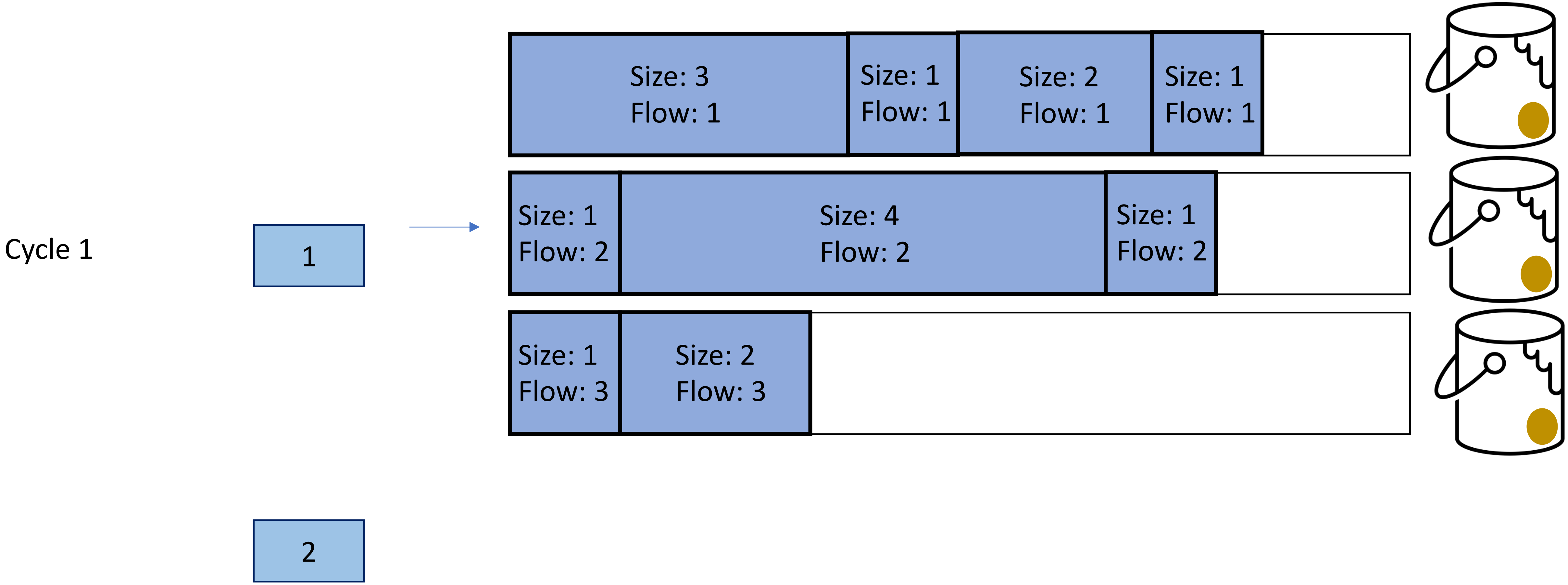
Size: 4
Flow: 2

Size: 1
Flow: 2

Size: 1
Flow: 3

Size: 2
Flow: 3

Deficit Round Robin



Size: 3 Flow: 1

Size: 1 Flow: 1

Size: 2 Flow: 1

Size: 1 Flow: 1

Cycle 1

1

Size: 1 Flow: 2

Size: 4 Flow: 2

Size: 1 Flow: 2

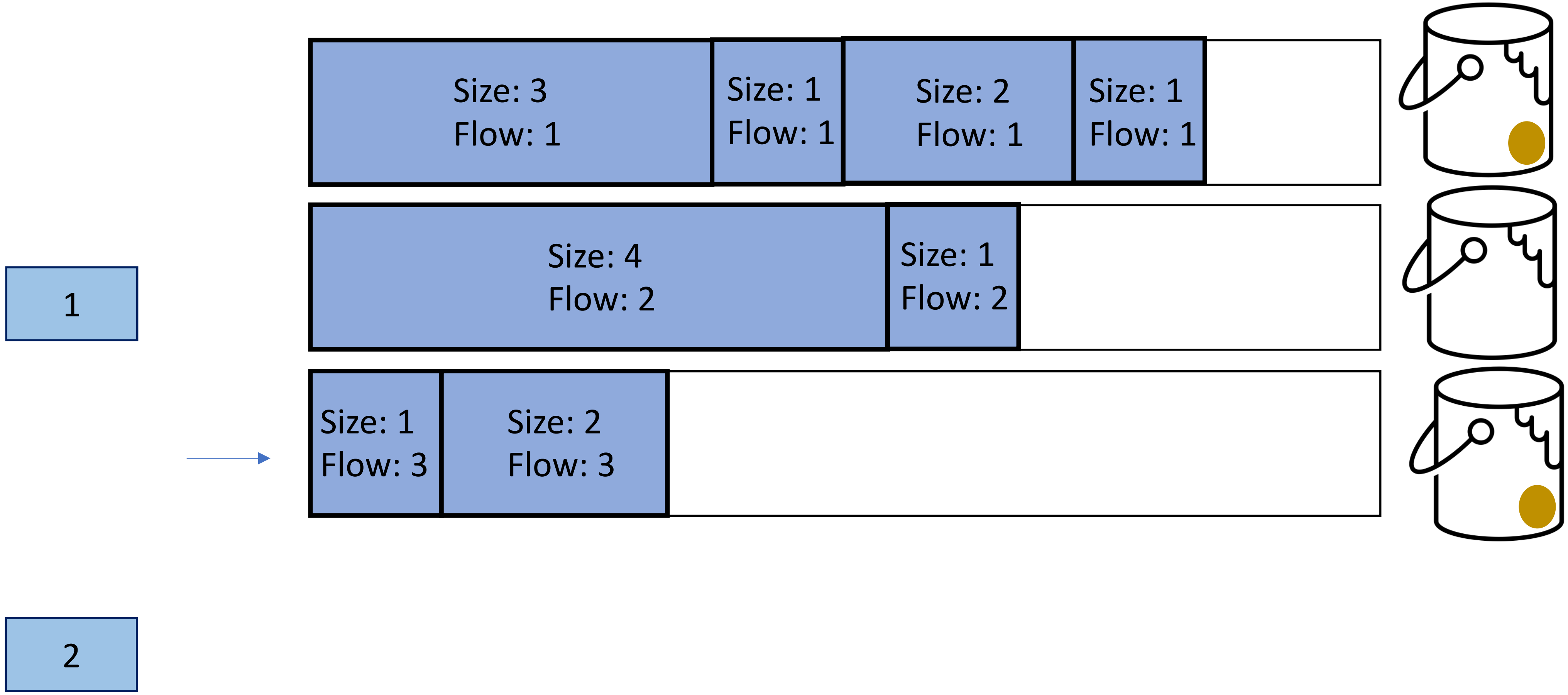
Size: 1 Flow: 3

Size: 2 Flow: 3

2

Deficit Round Robin

Cycle 1



1

2

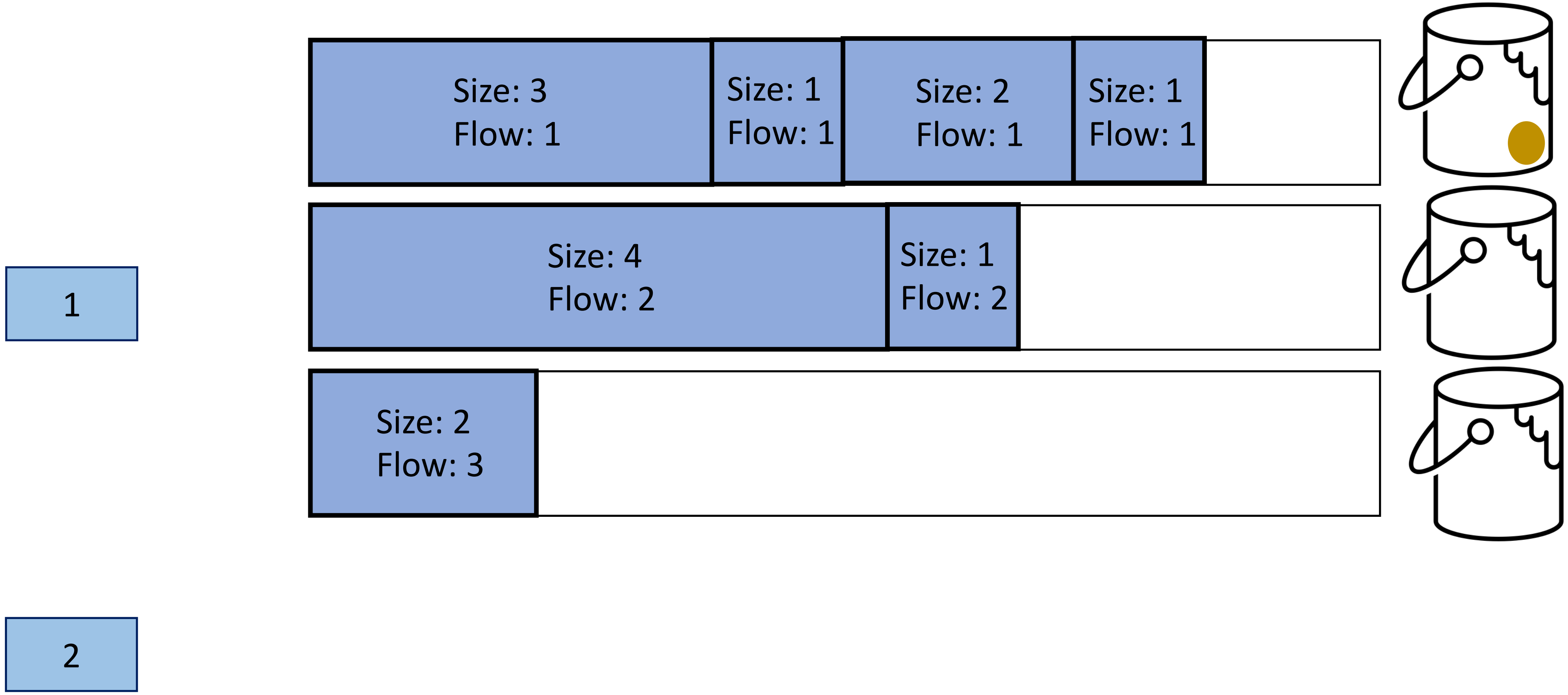
Size: 3 Flow: 1	Size: 1 Flow: 1	Size: 2 Flow: 1	Size: 1 Flow: 1

Size: 4 Flow: 2	Size: 1 Flow: 2

Size: 1 Flow: 3	Size: 2 Flow: 3

Deficit Round Robin

End of
Cycle 1



1

2

Size: 3
Flow: 1

Size: 1
Flow: 1

Size: 2
Flow: 1

Size: 1
Flow: 1

Size: 4
Flow: 2

Size: 1
Flow: 2

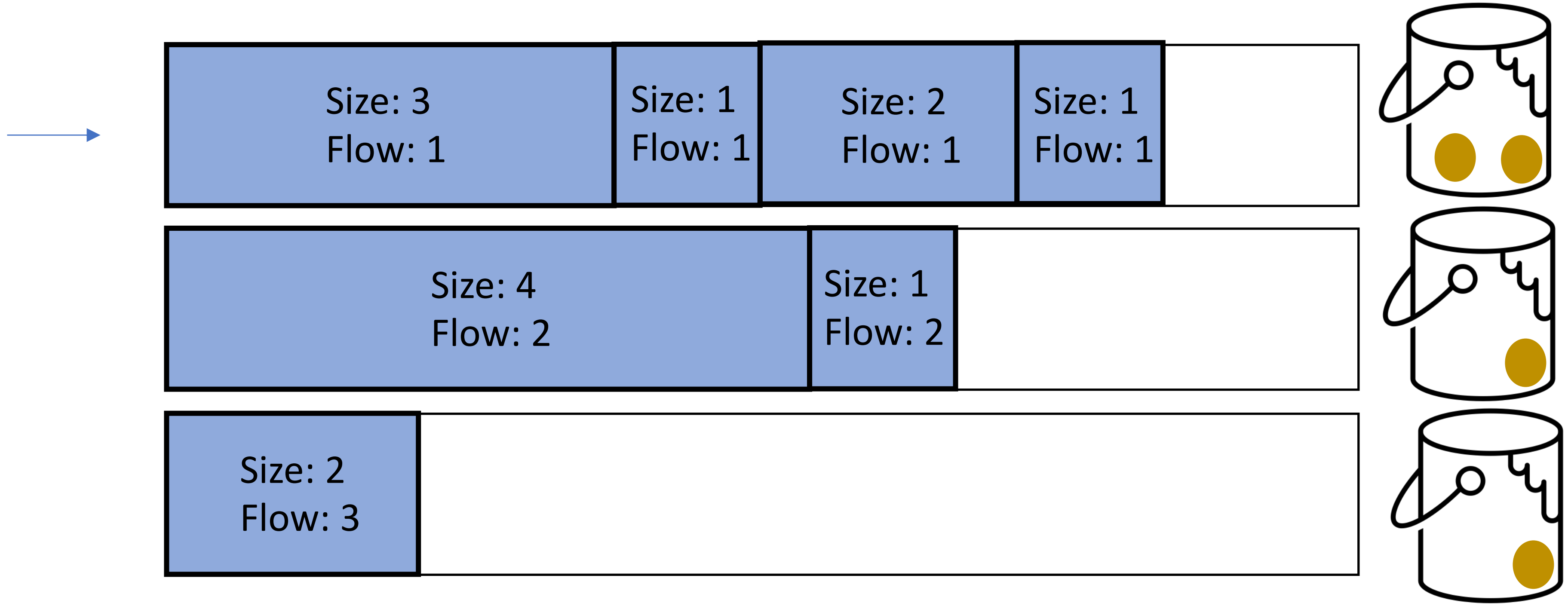
Size: 2
Flow: 3

Deficit Round Robin

Cycle 2

1

2



Size: 3
Flow: 1

Size: 1
Flow: 1

Size: 2
Flow: 1

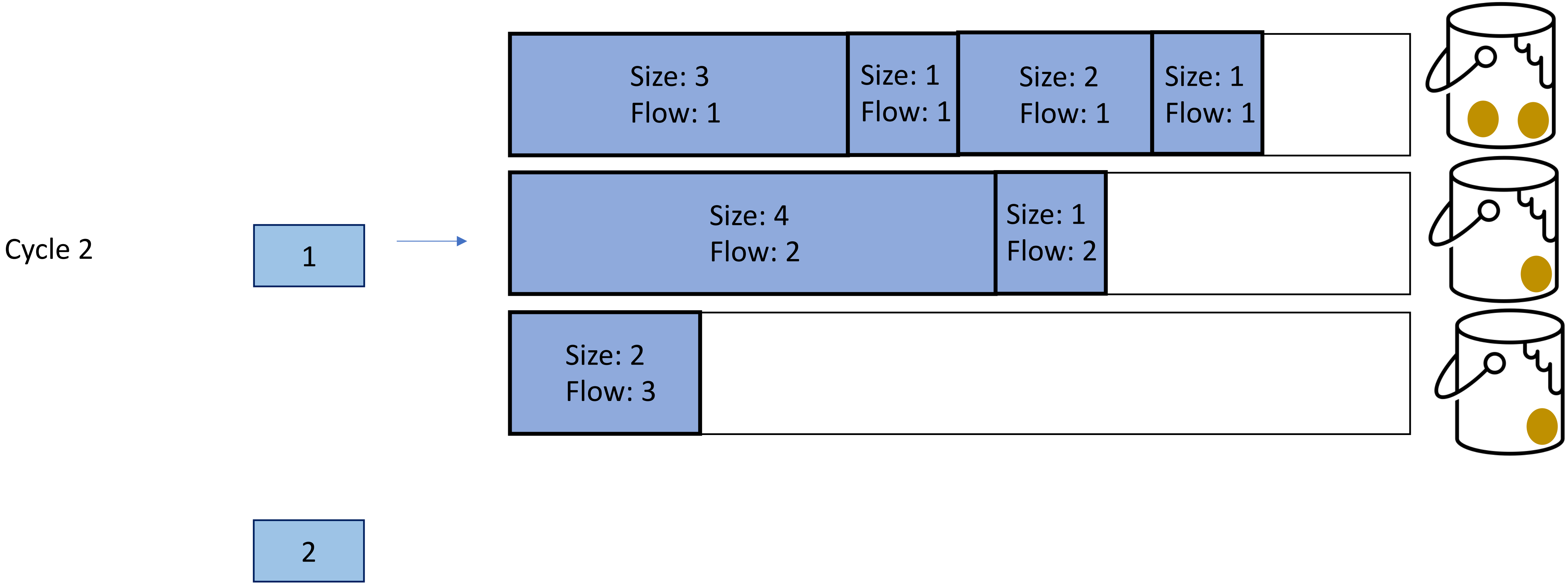
Size: 1
Flow: 1

Size: 4
Flow: 2

Size: 1
Flow: 2

Size: 2
Flow: 3

Deficit Round Robin



Cycle 2

1

2

Size: 3
Flow: 1

Size: 1
Flow: 1

Size: 2
Flow: 1

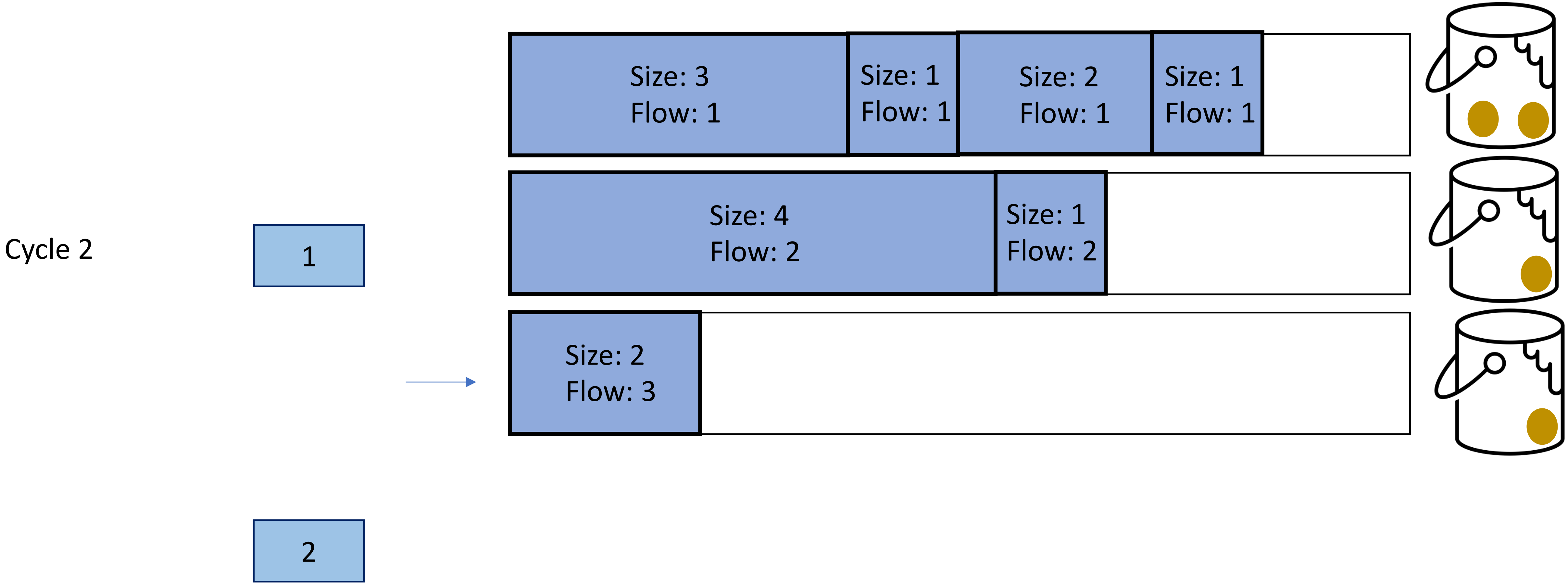
Size: 1
Flow: 1

Size: 4
Flow: 2

Size: 1
Flow: 2

Size: 2
Flow: 3

Deficit Round Robin



Cycle 2

1

2

Size: 3
Flow: 1

Size: 1
Flow: 1

Size: 2
Flow: 1

Size: 1
Flow: 1

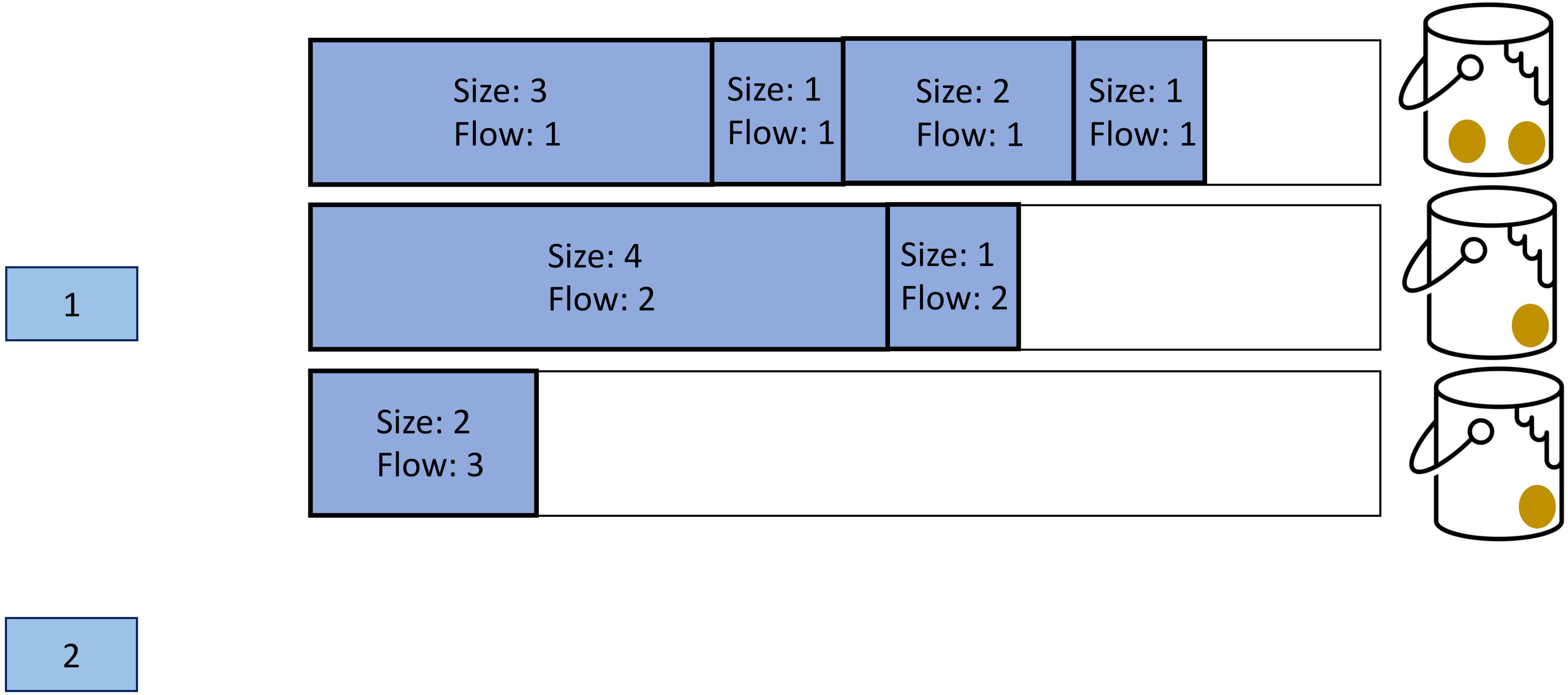
Size: 4
Flow: 2

Size: 1
Flow: 2

Size: 2
Flow: 3

Deficit Round Robin

End of
Cycle 2



1

Size: 3
Flow: 1

Size: 1
Flow: 1

Size: 2
Flow: 1

Size: 1
Flow: 1

Size: 4
Flow: 2

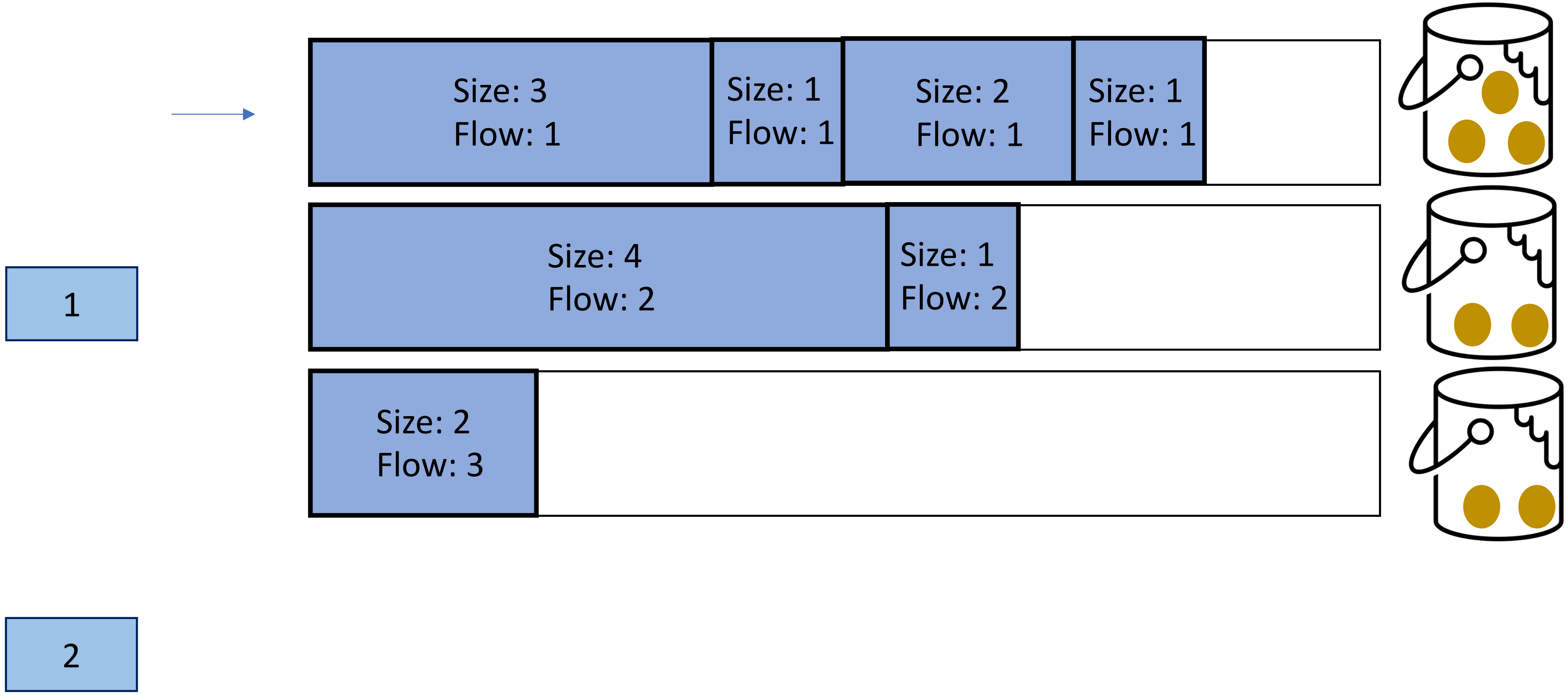
Size: 1
Flow: 2

Size: 2
Flow: 3

2

Deficit Round Robin

Cycle 3



1

2

Size: 3
Flow: 1

Size: 1
Flow: 1

Size: 2
Flow: 1

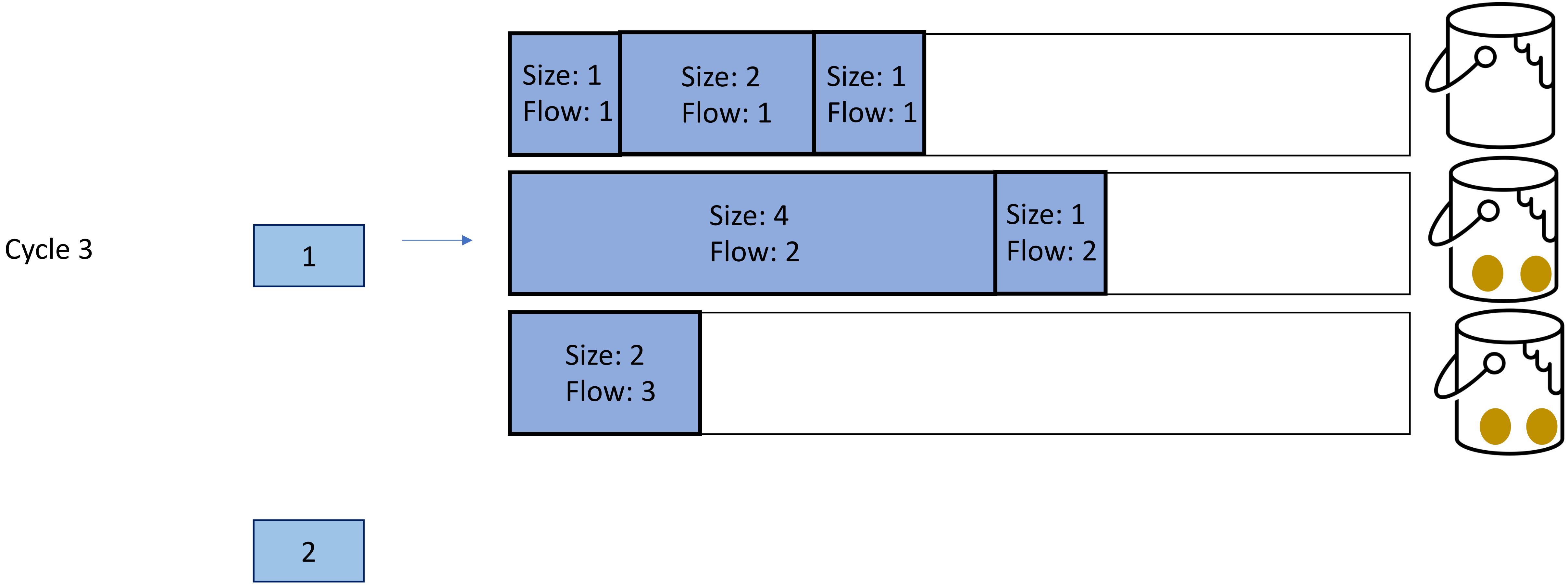
Size: 1
Flow: 1

Size: 4
Flow: 2

Size: 1
Flow: 2

Size: 2
Flow: 3

Deficit Round Robin



Cycle 3

1

2

Size: 1
Flow: 1

Size: 2
Flow: 1

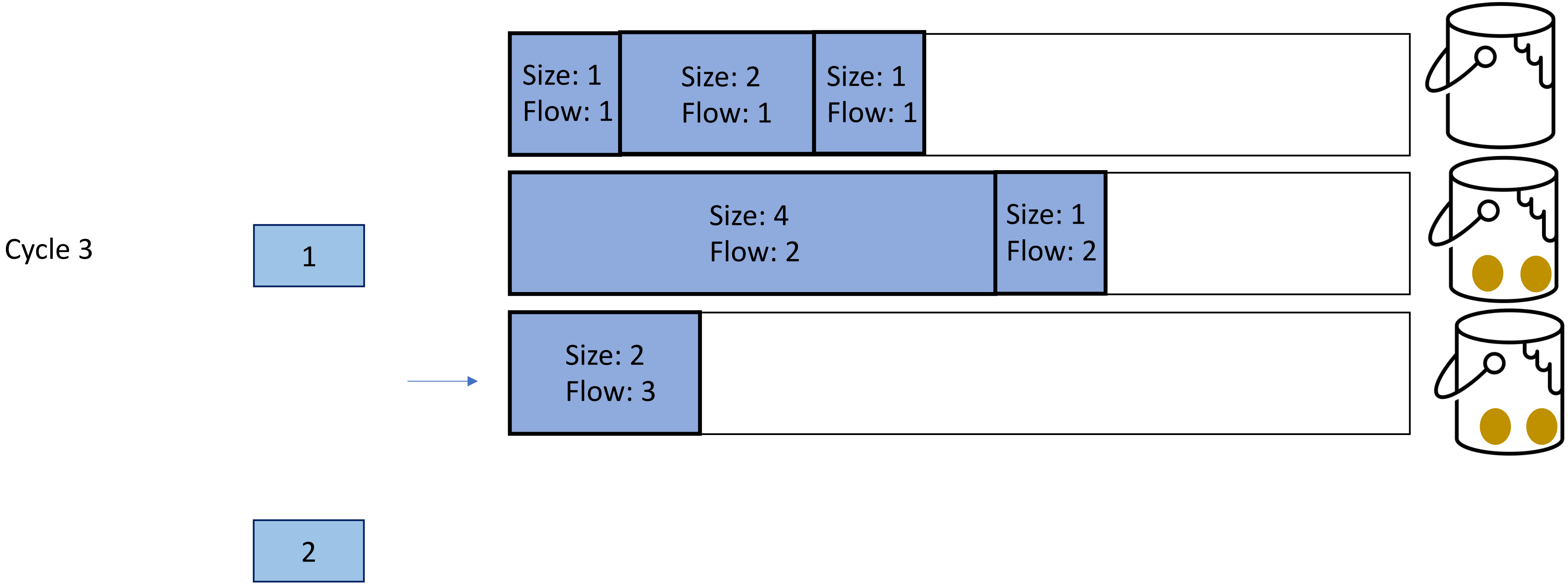
Size: 1
Flow: 1

Size: 4
Flow: 2

Size: 1
Flow: 2

Size: 2
Flow: 3

Deficit Round Robin



Cycle 3

1

2

Size: 1
Flow: 1

Size: 2
Flow: 1

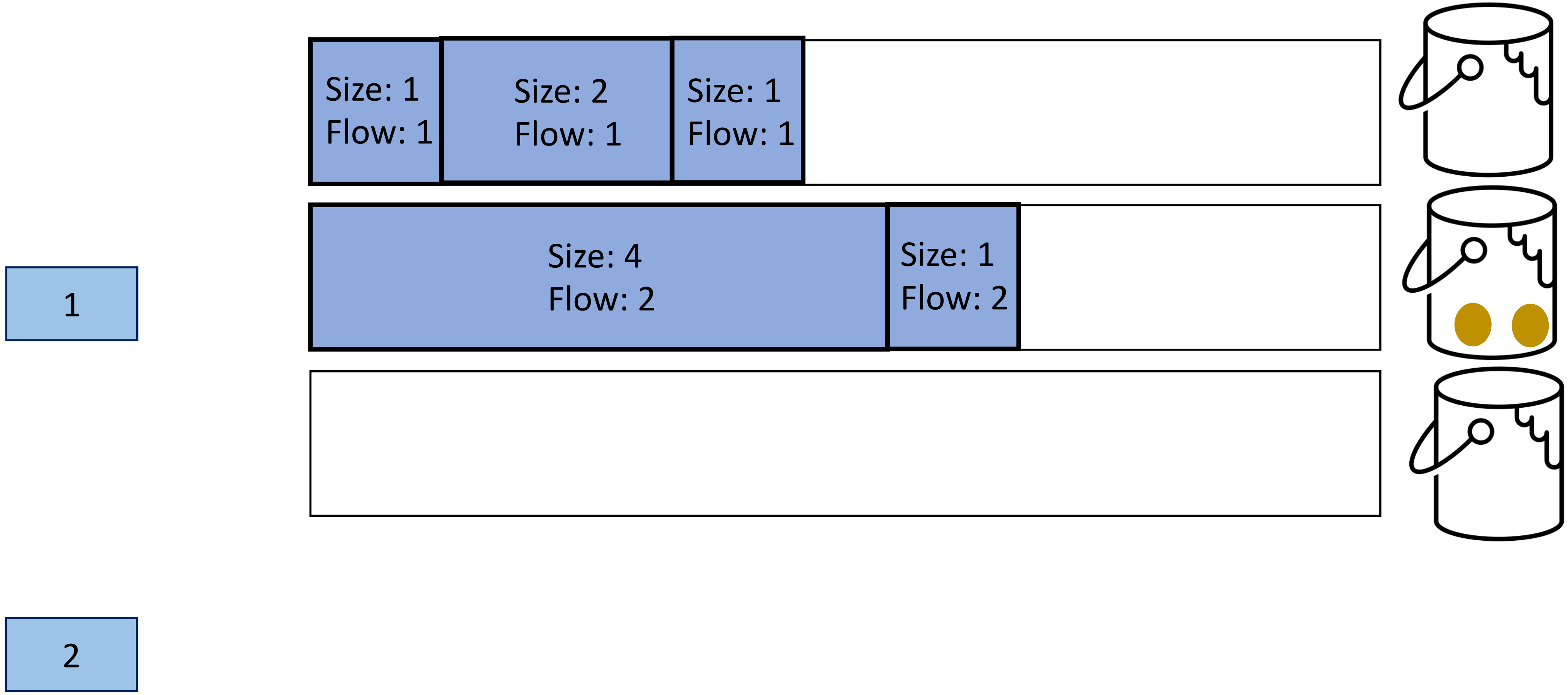
Size: 1
Flow: 1

Size: 4
Flow: 2

Size: 1
Flow: 2

Size: 2
Flow: 3

Deficit Round Robin



End of Cycle 3

1

2

Size: 1
Flow: 1

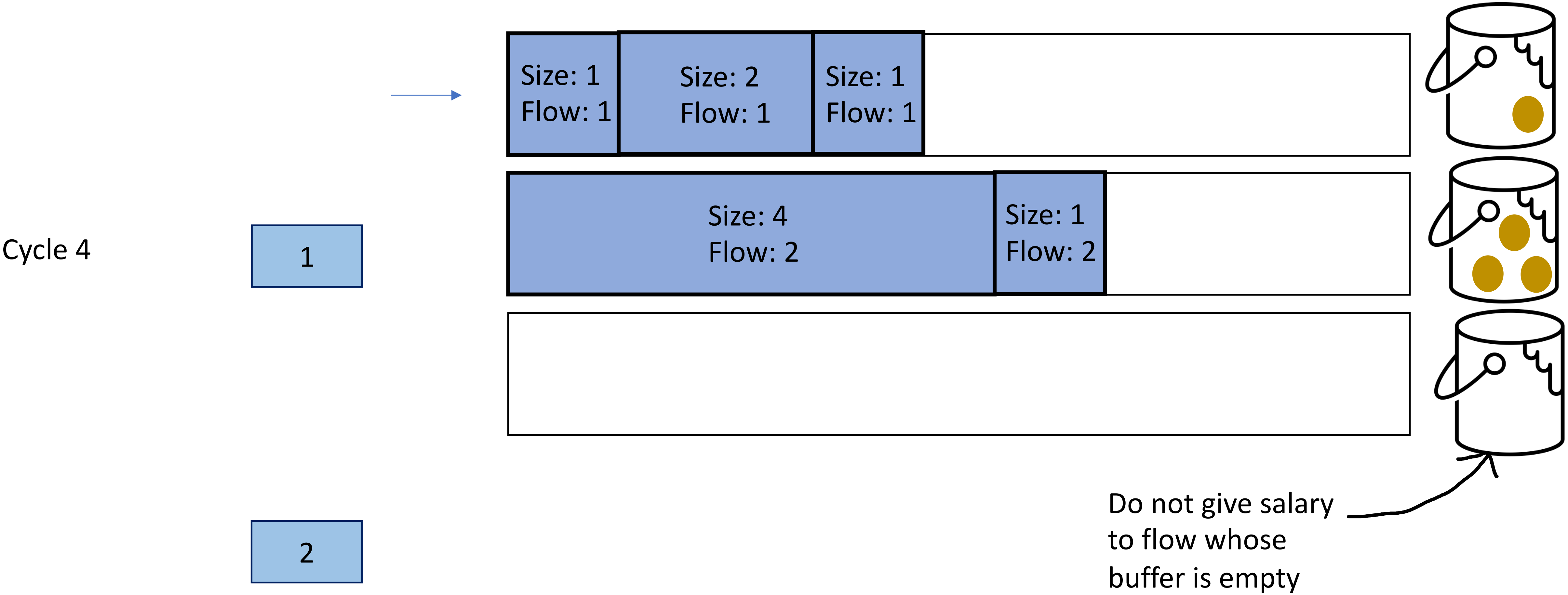
Size: 2
Flow: 1

Size: 1
Flow: 1

Size: 4
Flow: 2

Size: 1
Flow: 2

Deficit Round Robin



Do not give salary to flow whose buffer is empty

Deficit Round Robin

- Keep buffer and token bucket per flow and round robin, sending as many packets as possible each turn
- Each bucket gets a “salary,” or an amount put in on each cycle of the round robin