CNI Seminar

Online *Age-of-Information* Scheduling

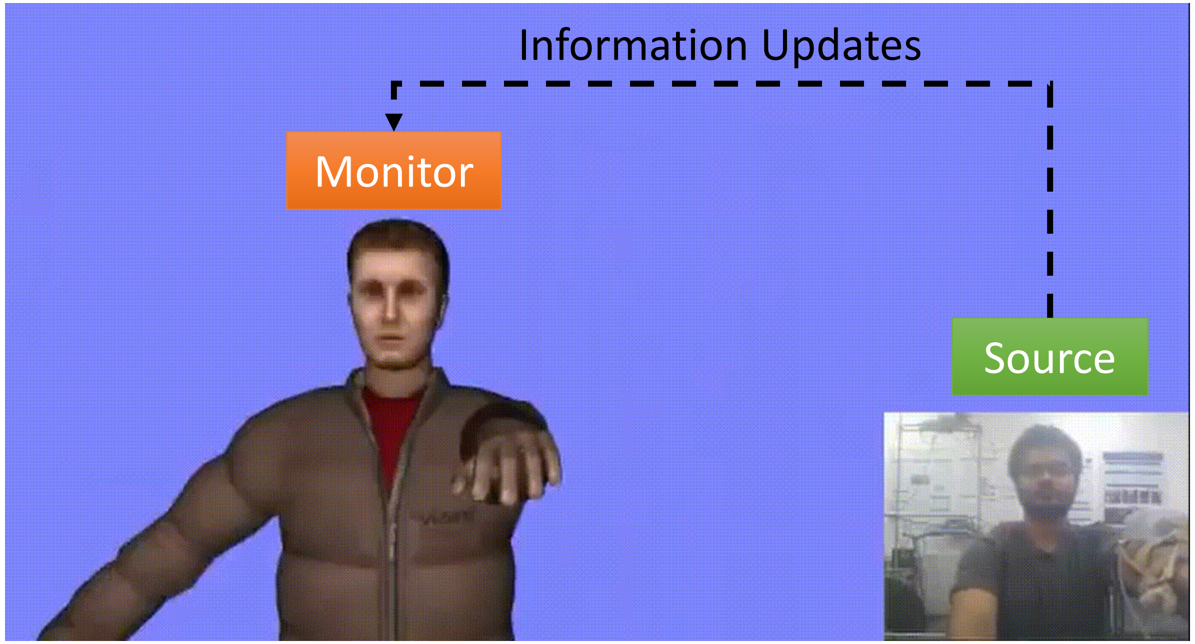
Kumar Saurav

(TIFR, Mumbai)

Joint work with Prof. Rahul Vaze *(TIFR, Mumbai)*

Modern Applications

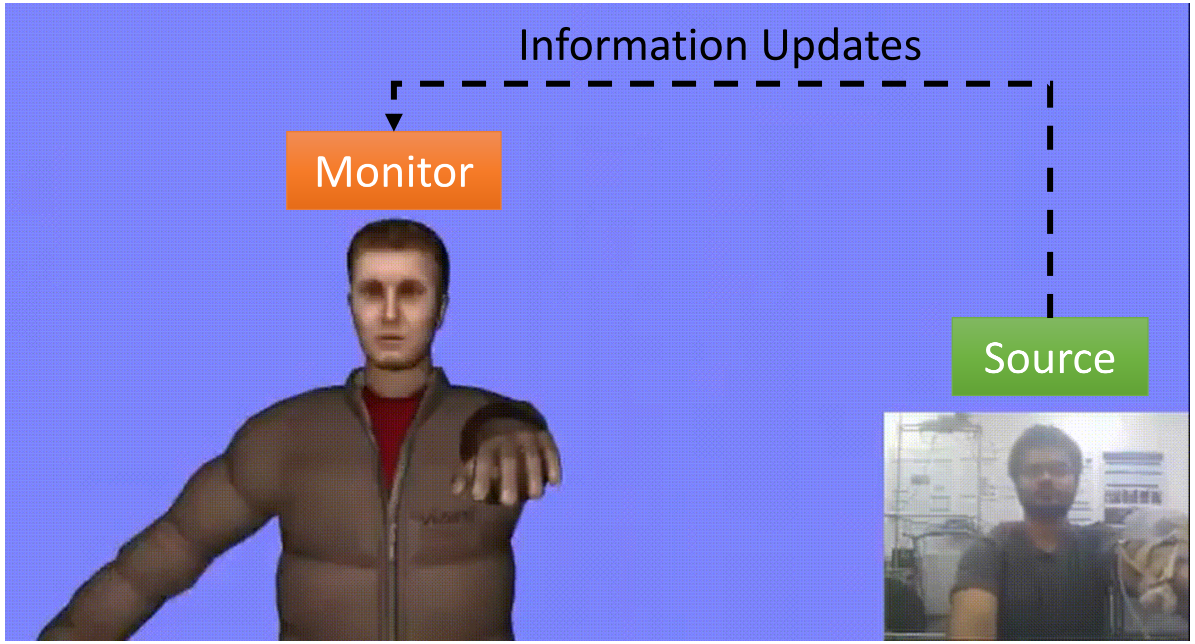
e.g.: virtual reality, tele-robotics, networked cars, etc.



Modern Applications

e.g.: virtual reality, tele-robotics, networked cars, etc.

Information at the monitor should accurately reflect the most recent state of the source.

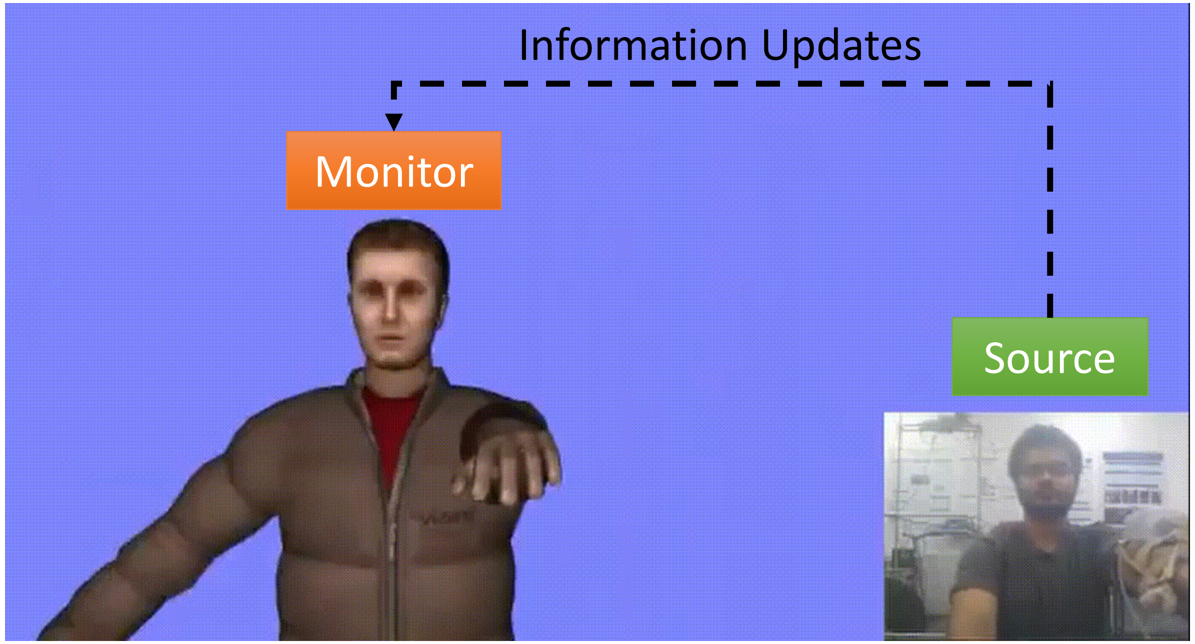


Modern Applications

e.g.: virtual reality, tele-robotics, networked cars, etc.

Information at the monitor should accurately reflect the most recent state of the source.

Information Freshness



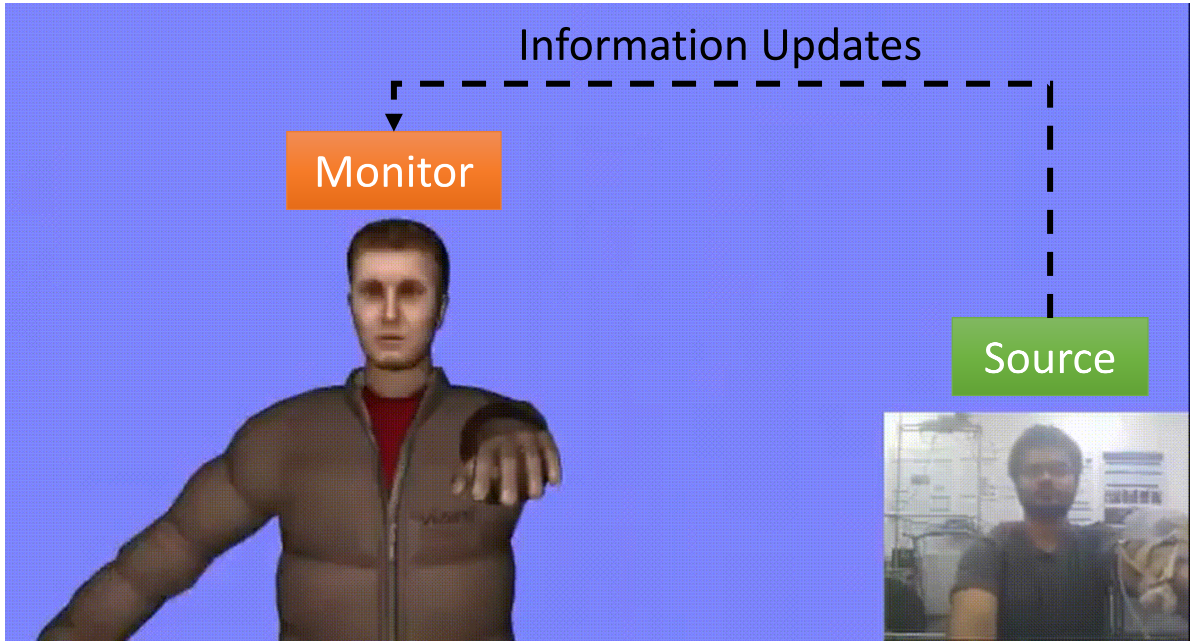
Modern Applications

e.g.: virtual reality, tele-robotics, networked cars, etc.

Information at the monitor should accurately reflect the most recent state of the source.

Information Freshness

Emphasis on recent information at monitor, instead of individual packets/updates.



Modern Applications

e.g.: virtual reality, tele-robotics, networked cars, etc.

↓

Information at the monitor should accurately reflect the most recent state of the source.

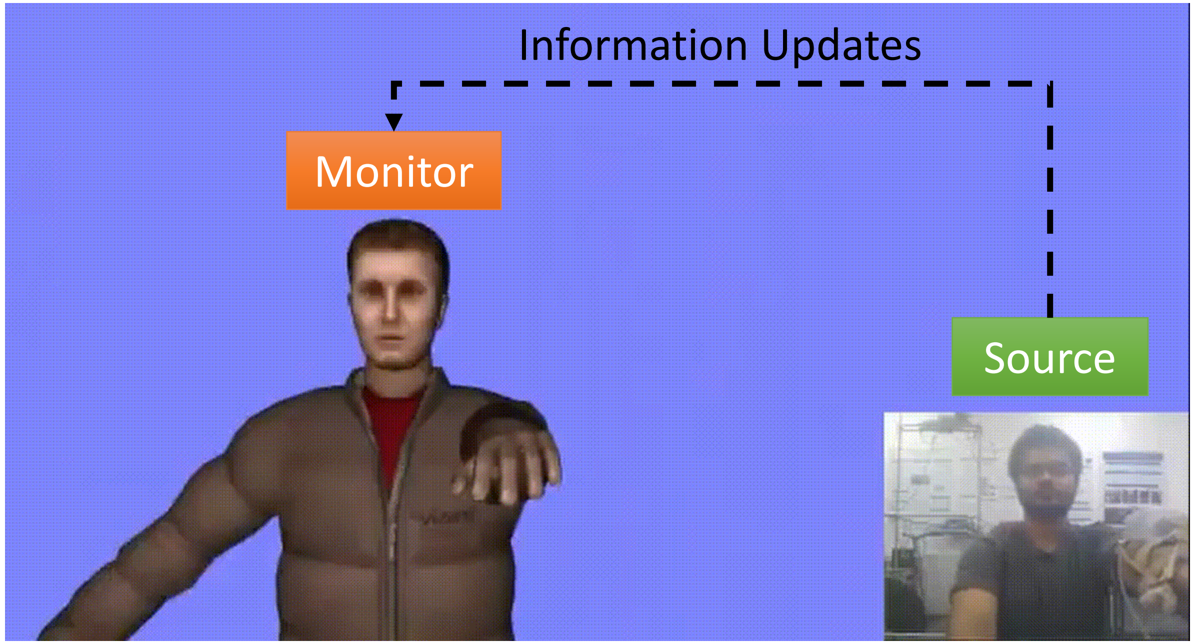
↓

Information Freshness

↓

Emphasis on recent information at monitor, instead of individual packets/updates.

Classical packet-based metrics (e.g. latency) not sufficient to quantify information freshness!



Modern Applications

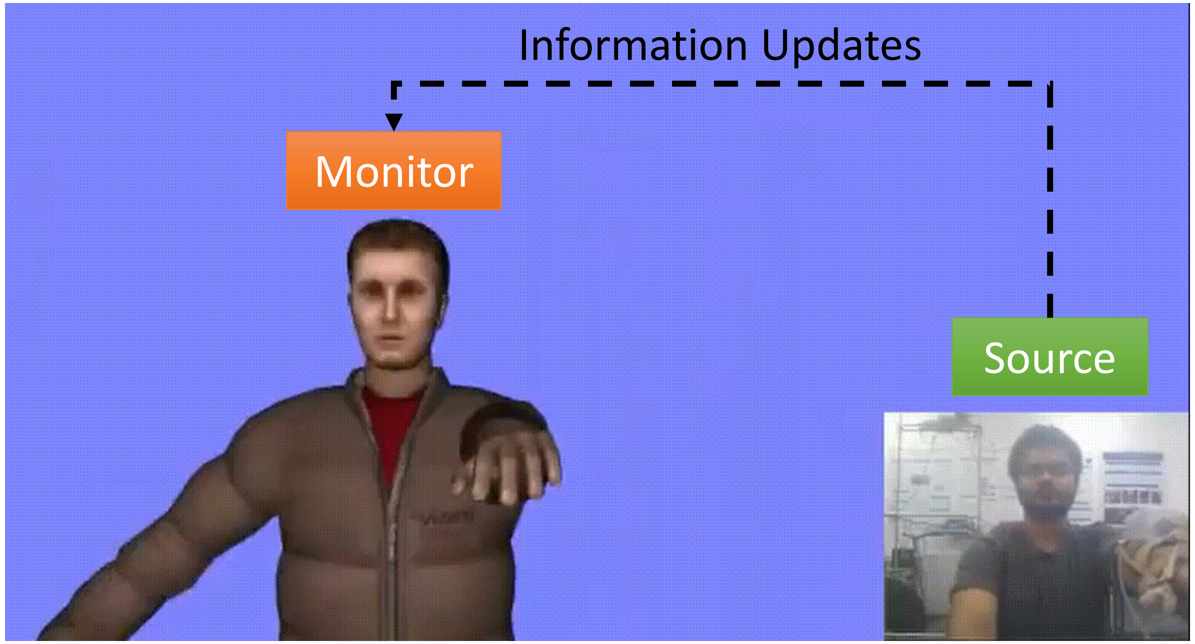
e.g.: virtual reality, tele-robotics, networked cars, etc.

Information at the monitor should accurately reflect the most recent state of the source.

Information Freshness

Emphasis on recent information at monitor, instead of individual packets/updates.

Classical packet-based metrics (e.g. latency) not sufficient to quantify information freshness!



New approaches towards formalizing INFORMATION FRESHNESS.

Modern Applications

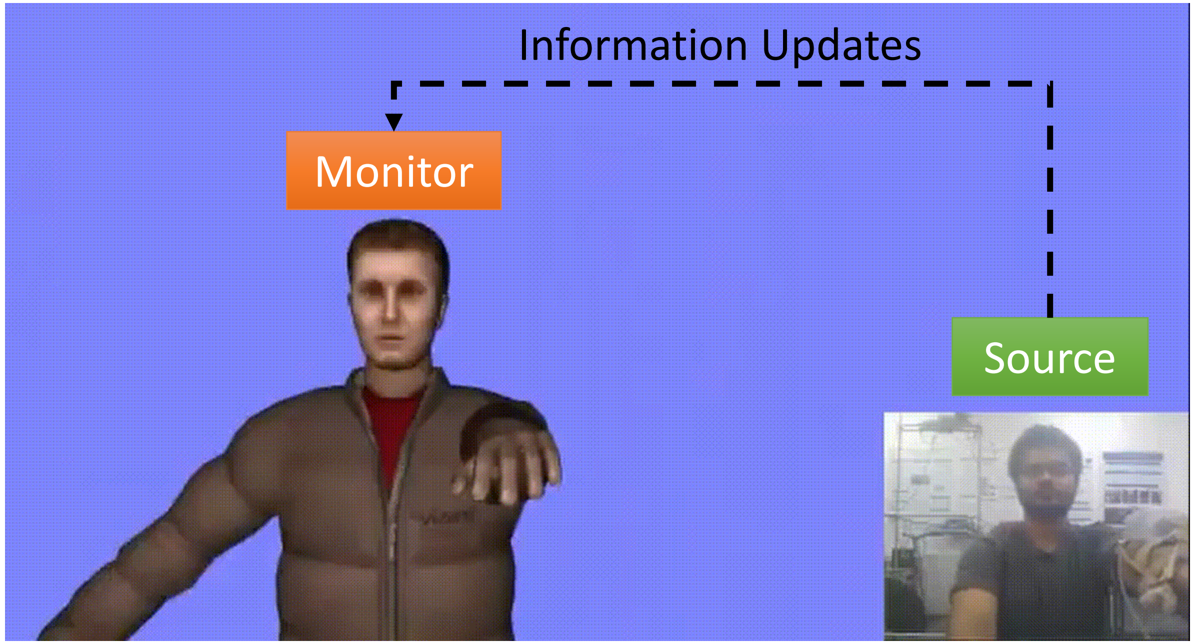
e.g.: virtual reality, tele-robotics, networked cars, etc.

Information at the monitor should accurately reflect the most recent state of the source.

Information Freshness

Emphasis on recent information at monitor, instead of individual packets/updates.

Classical packet-based metrics (e.g. latency) not sufficient to quantify information freshness!

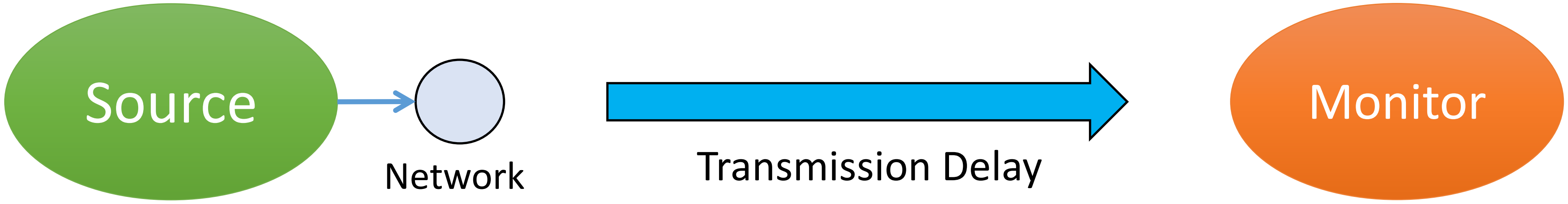


New approaches towards formalizing INFORMATION FRESHNESS.

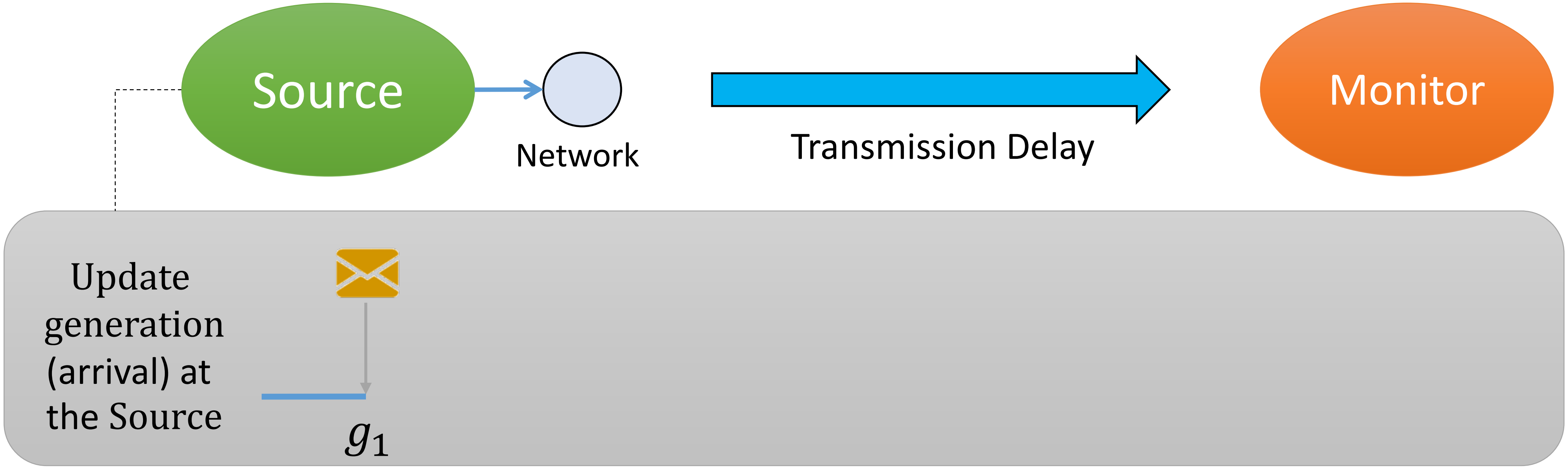
Most popular

Using AGE OF INFORMATION (AoI) metric.

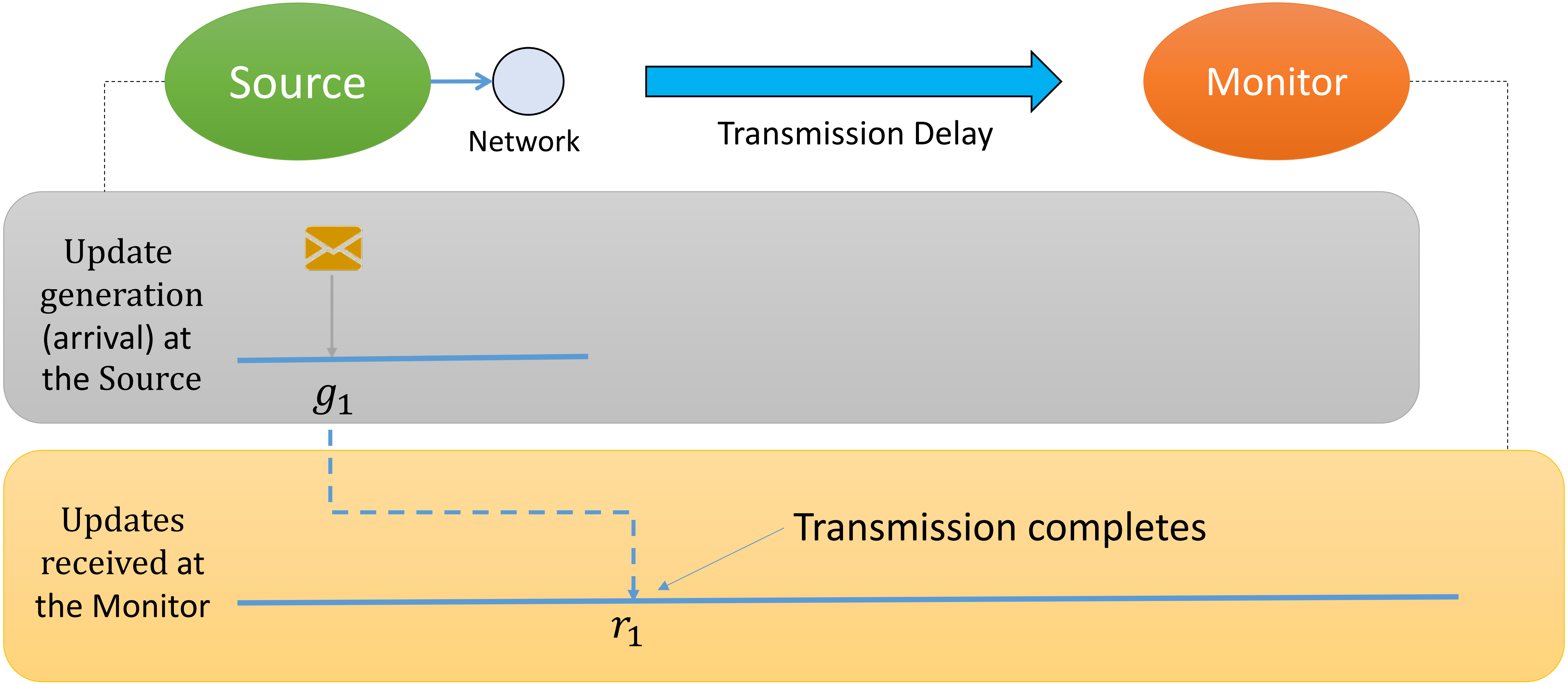
Age of Information



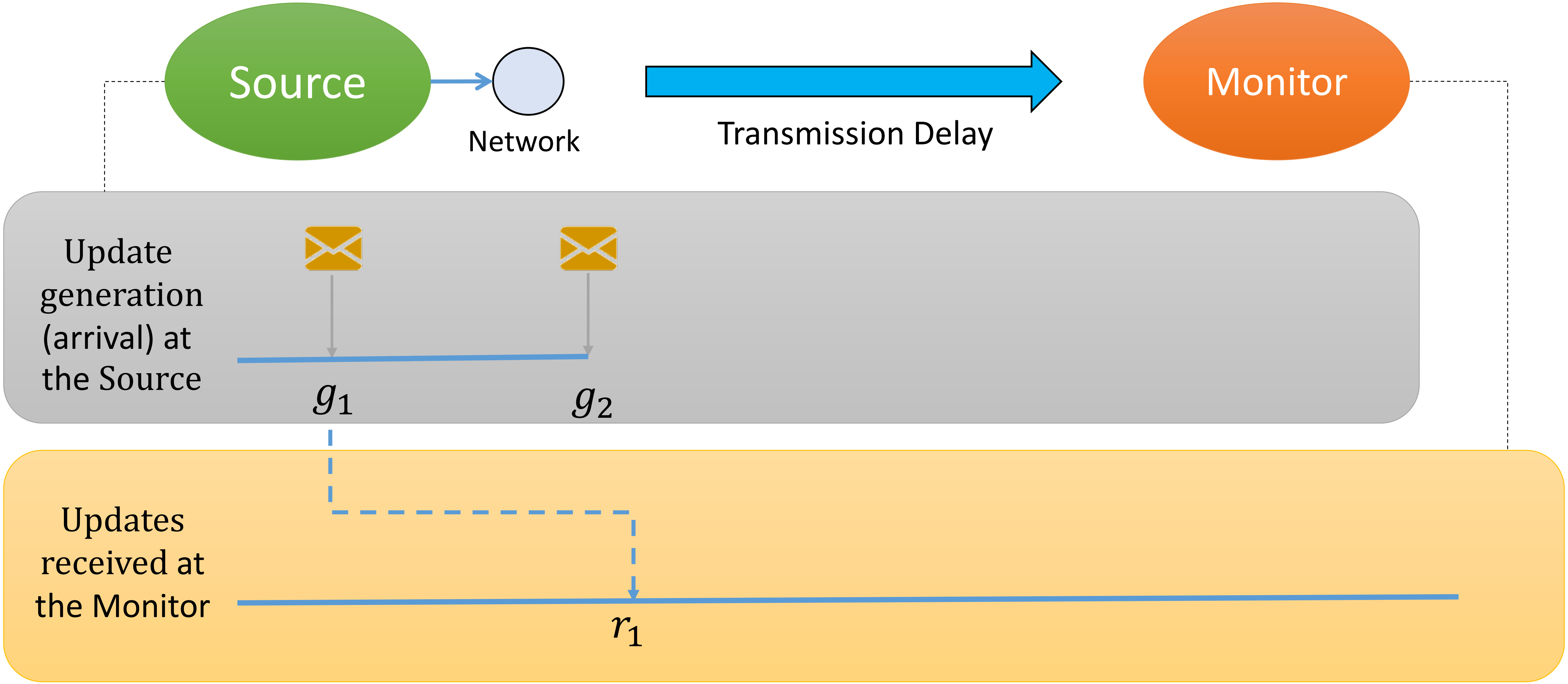
Age of Information



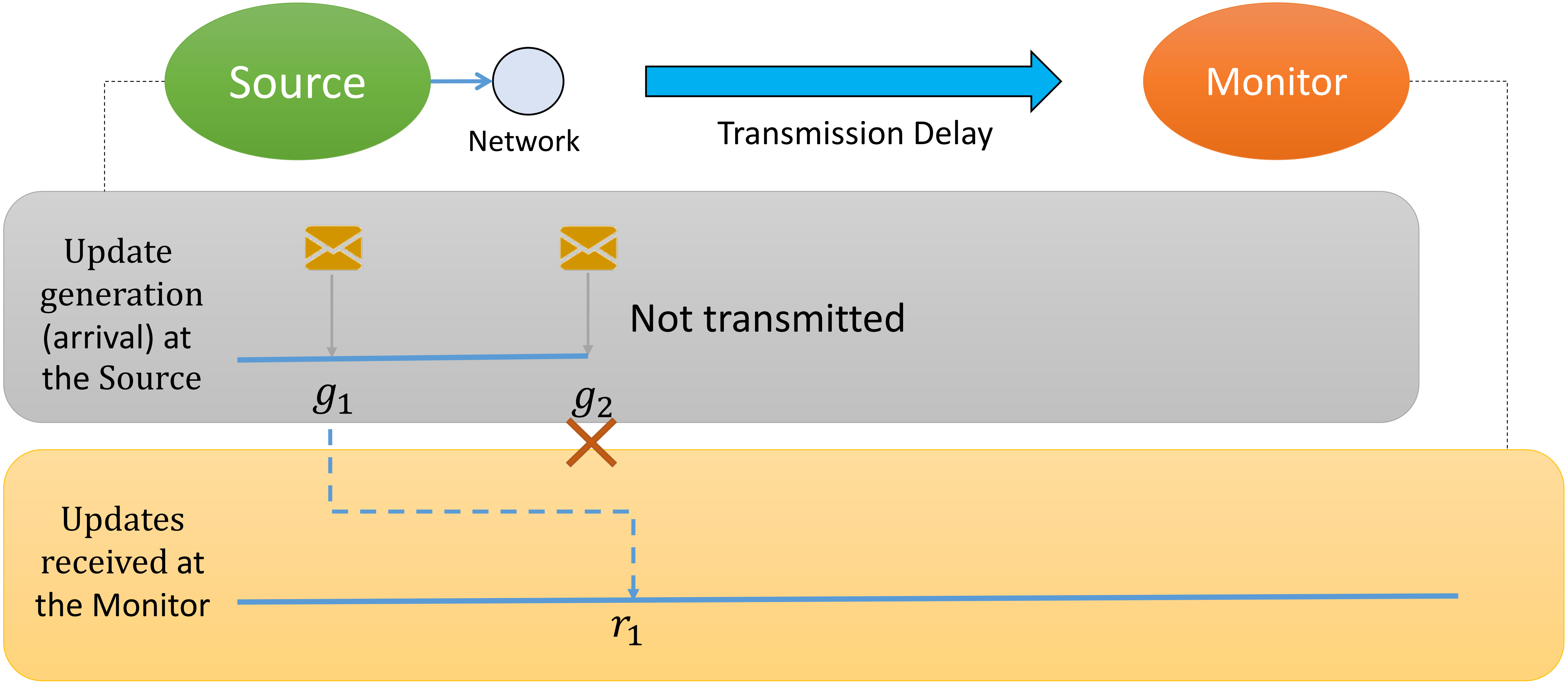
Age of Information



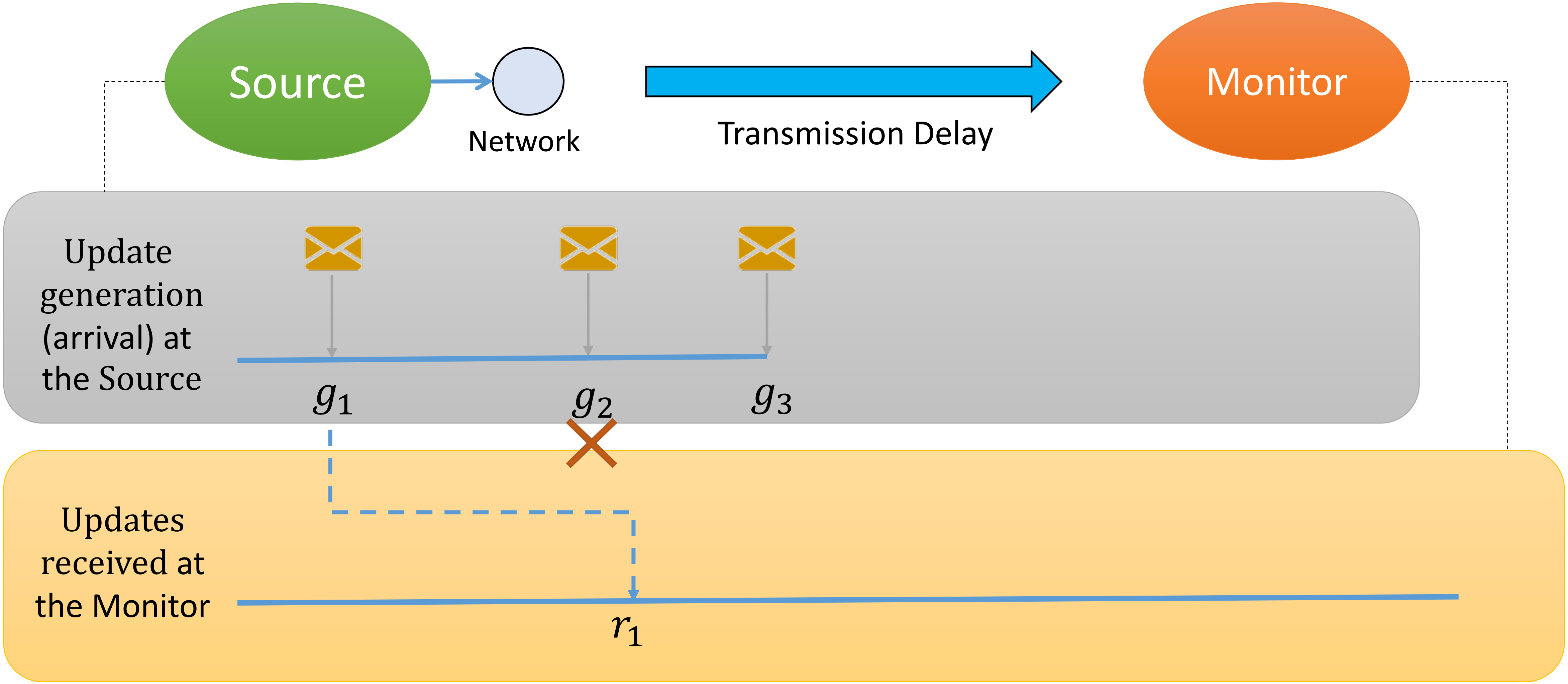
Age of Information



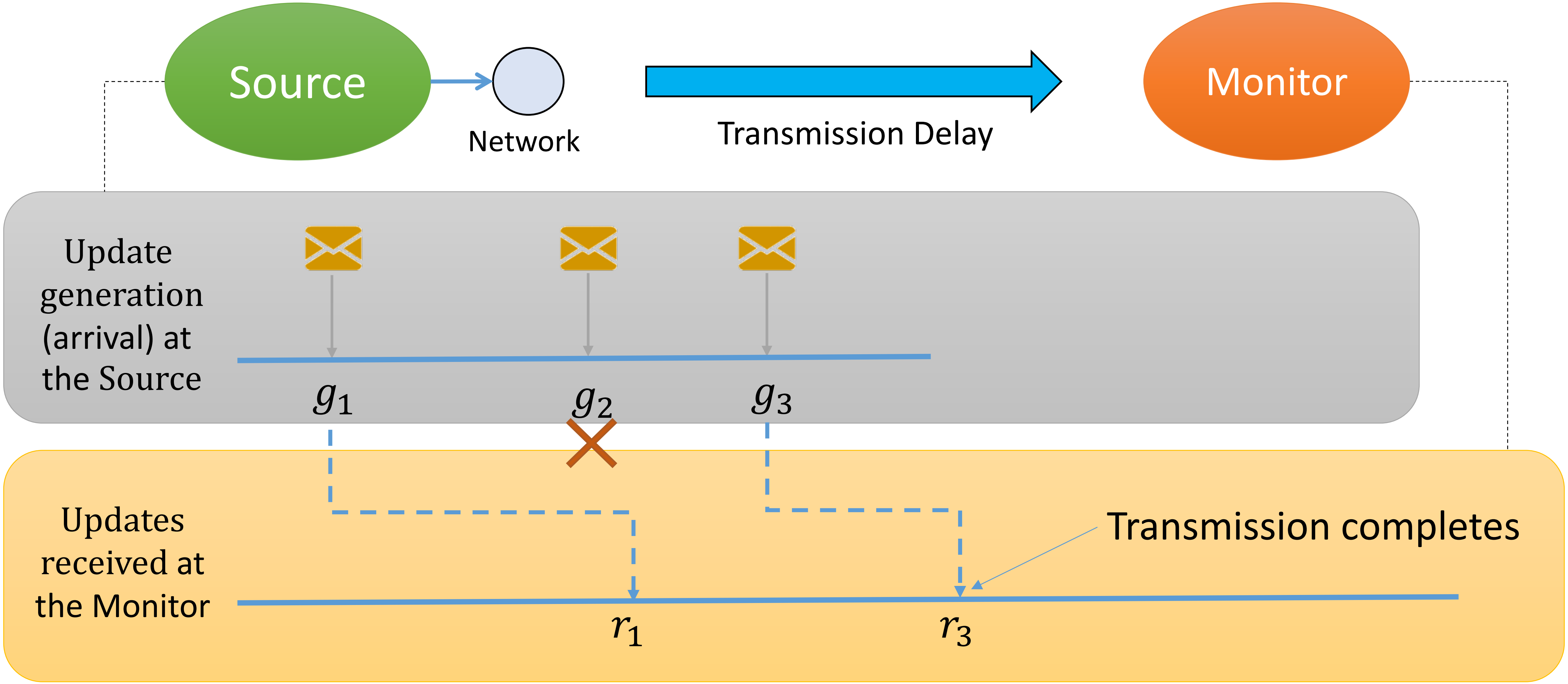
Age of Information



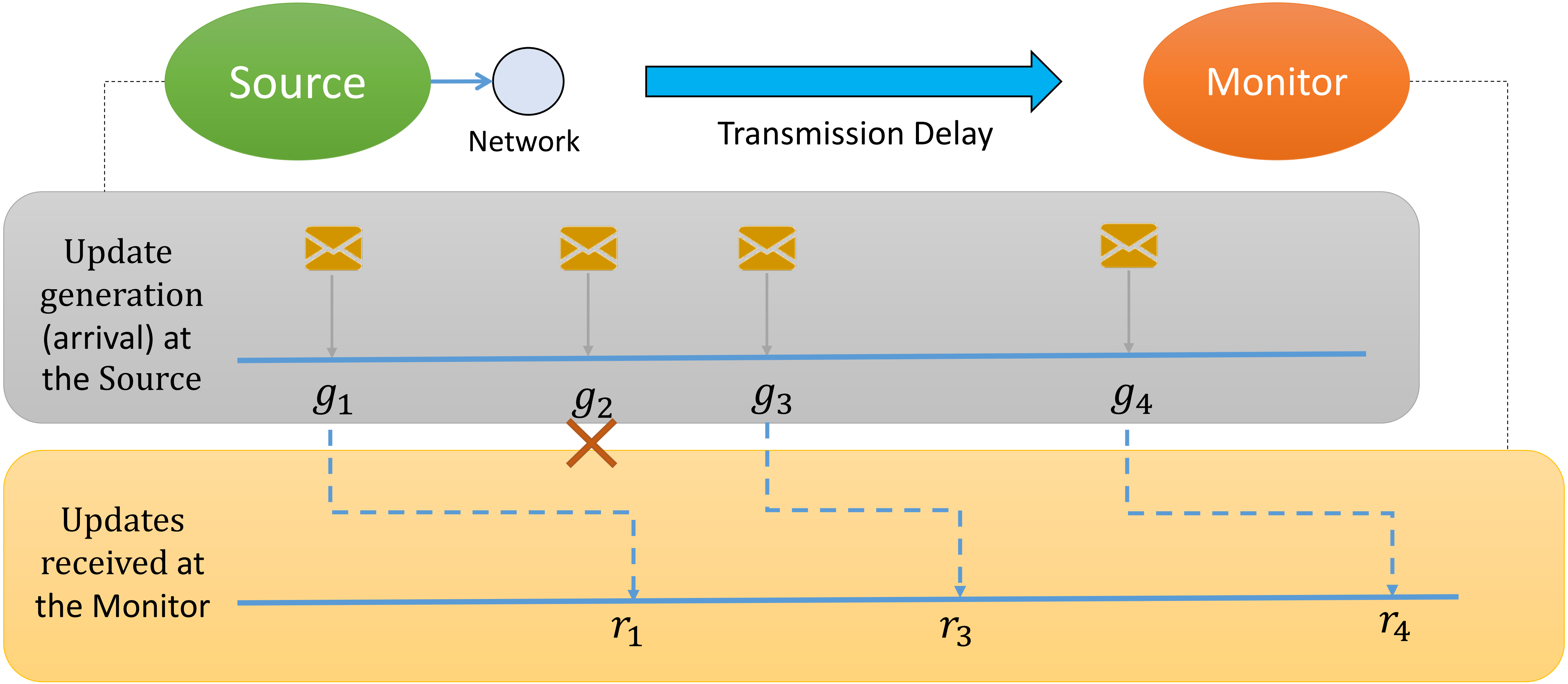
Age of Information



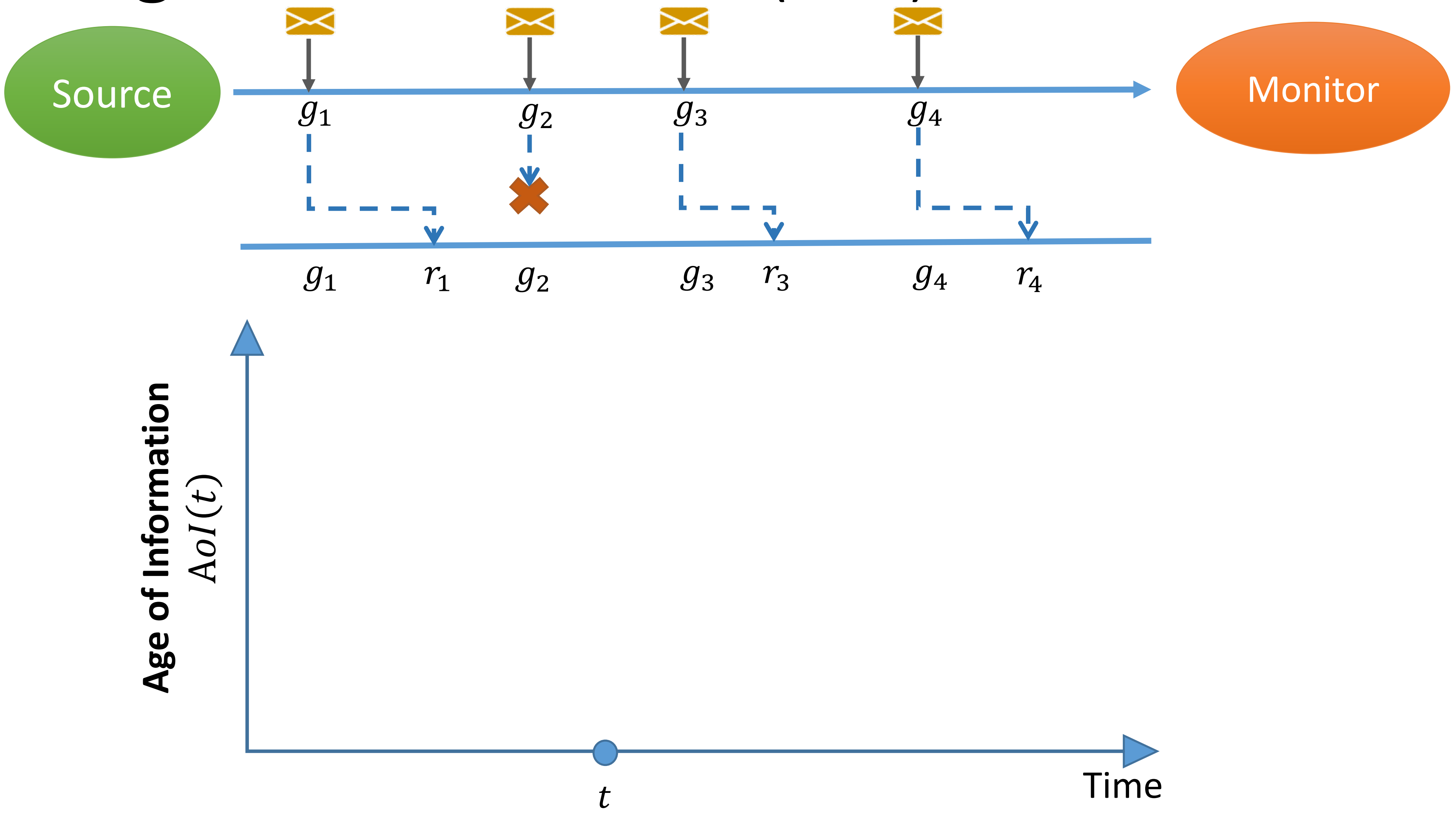
Age of Information



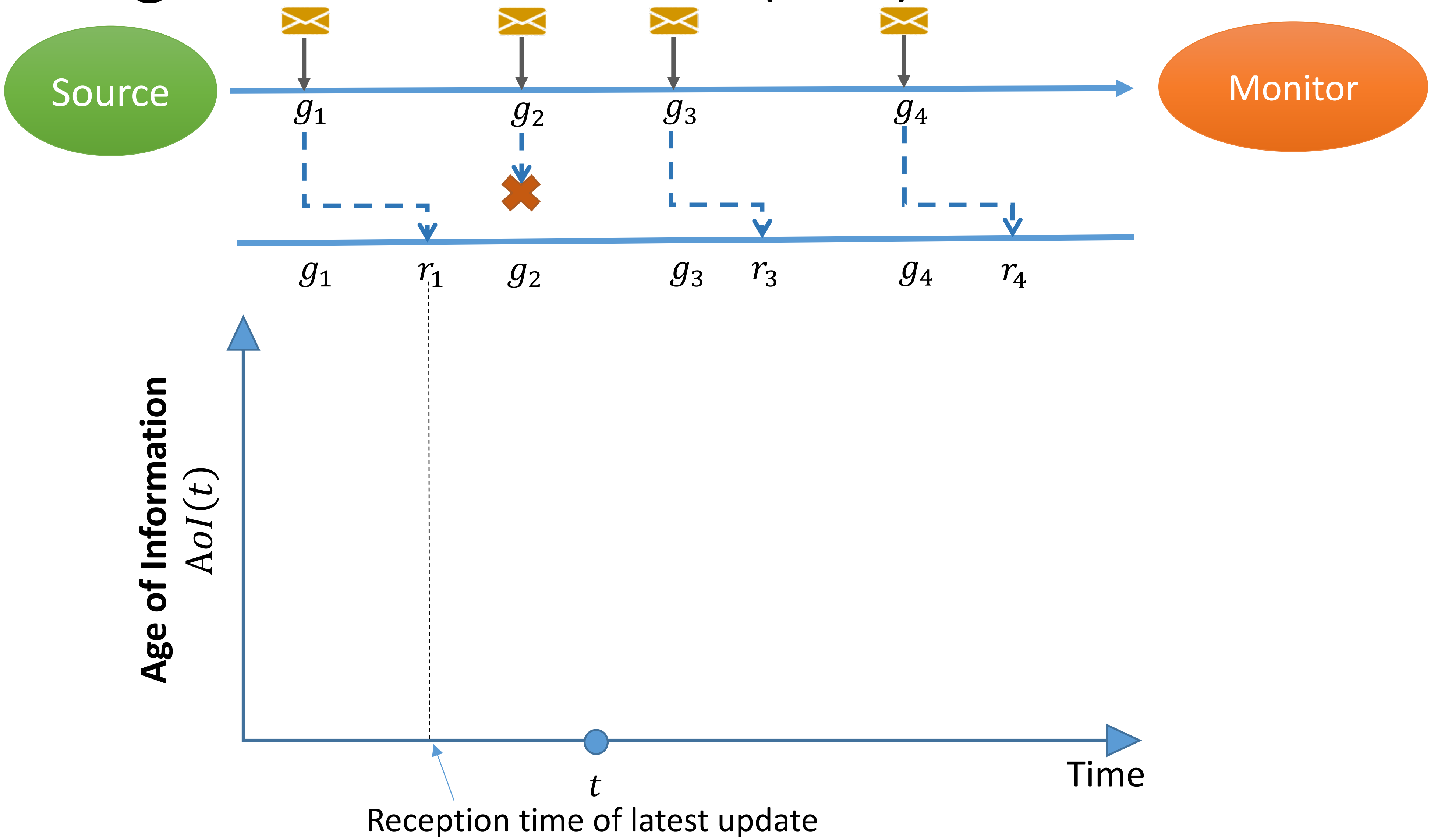
Age of Information



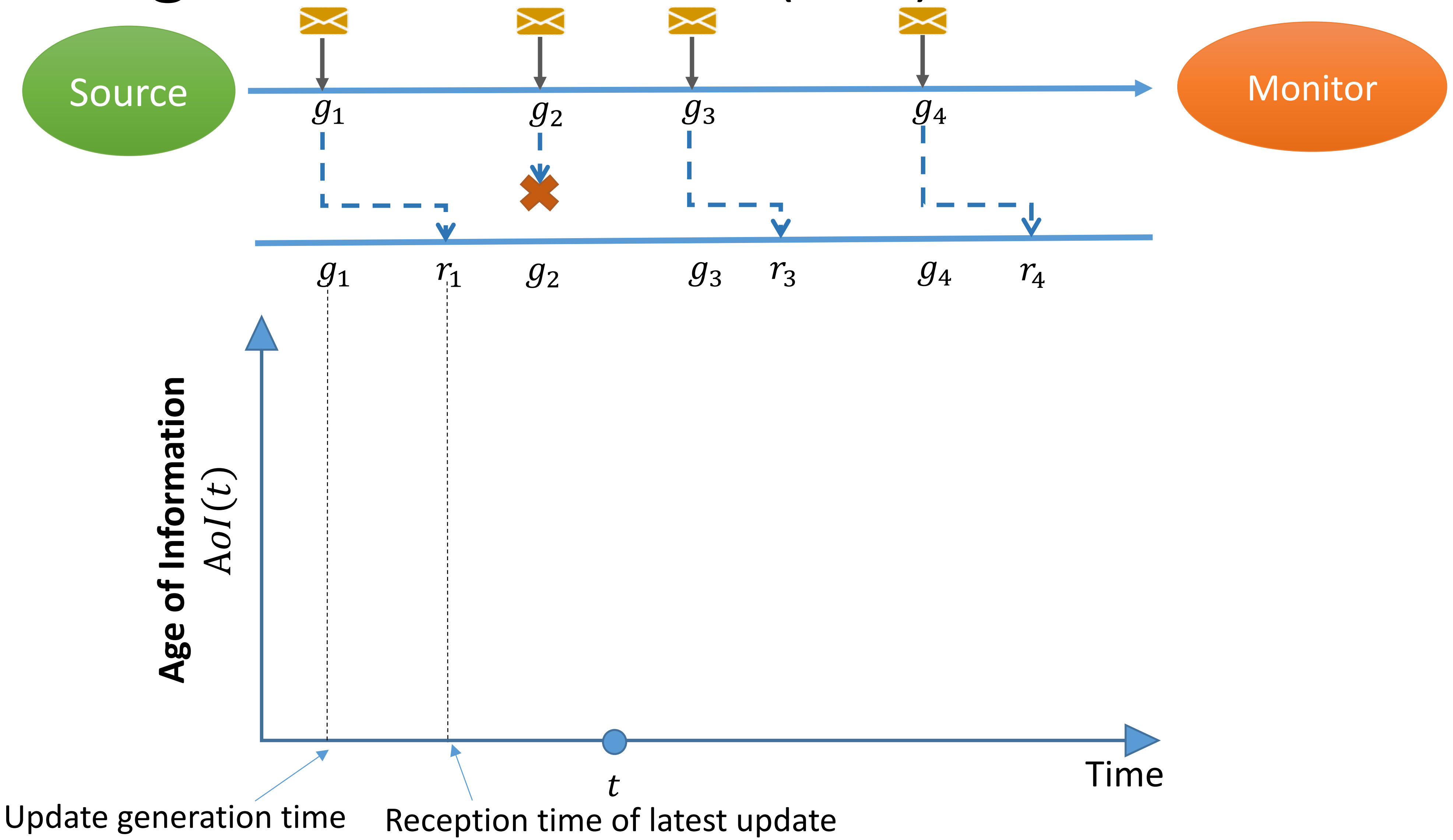
Age of Information (AoI)



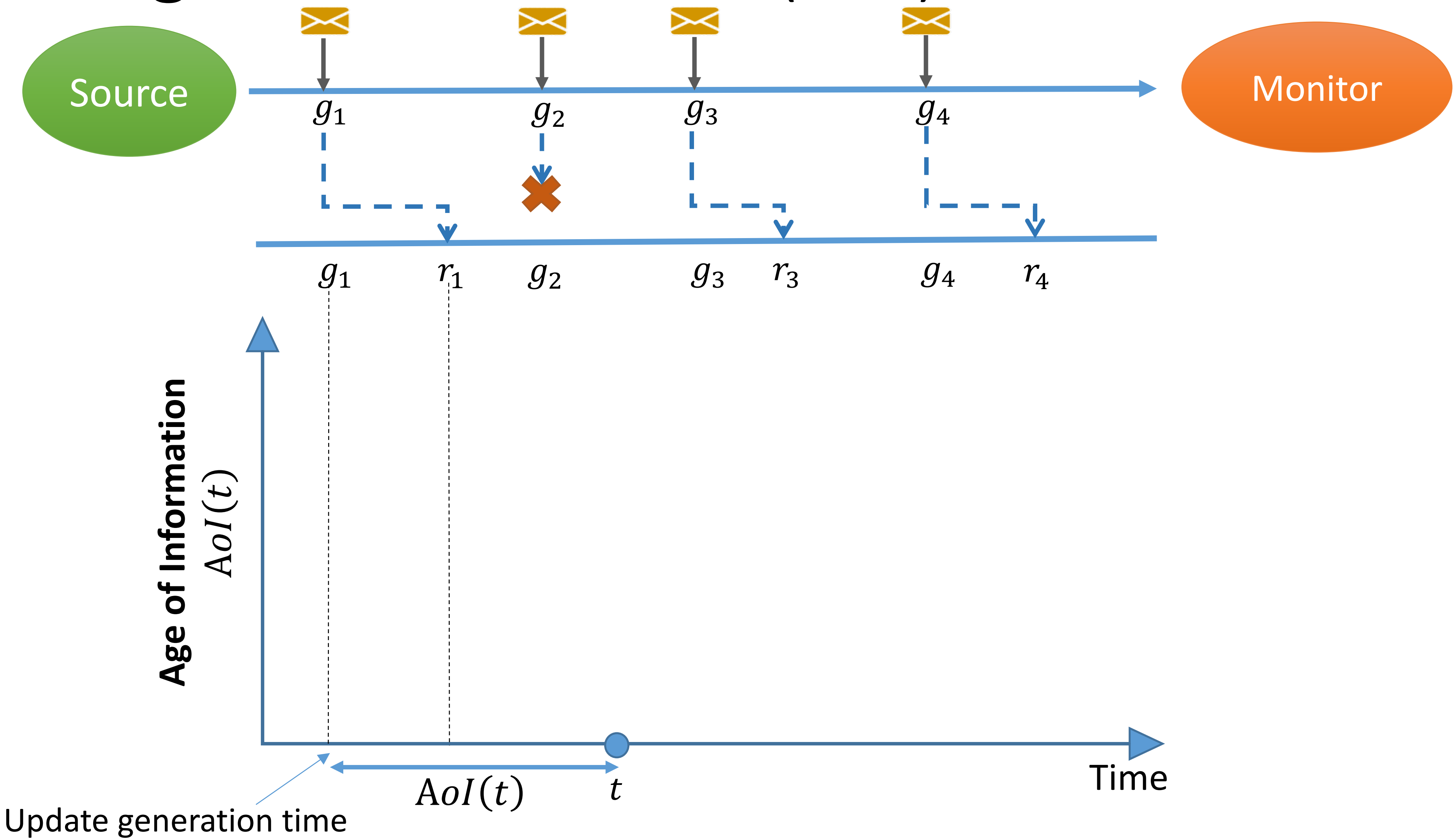
Age of Information (AoI)



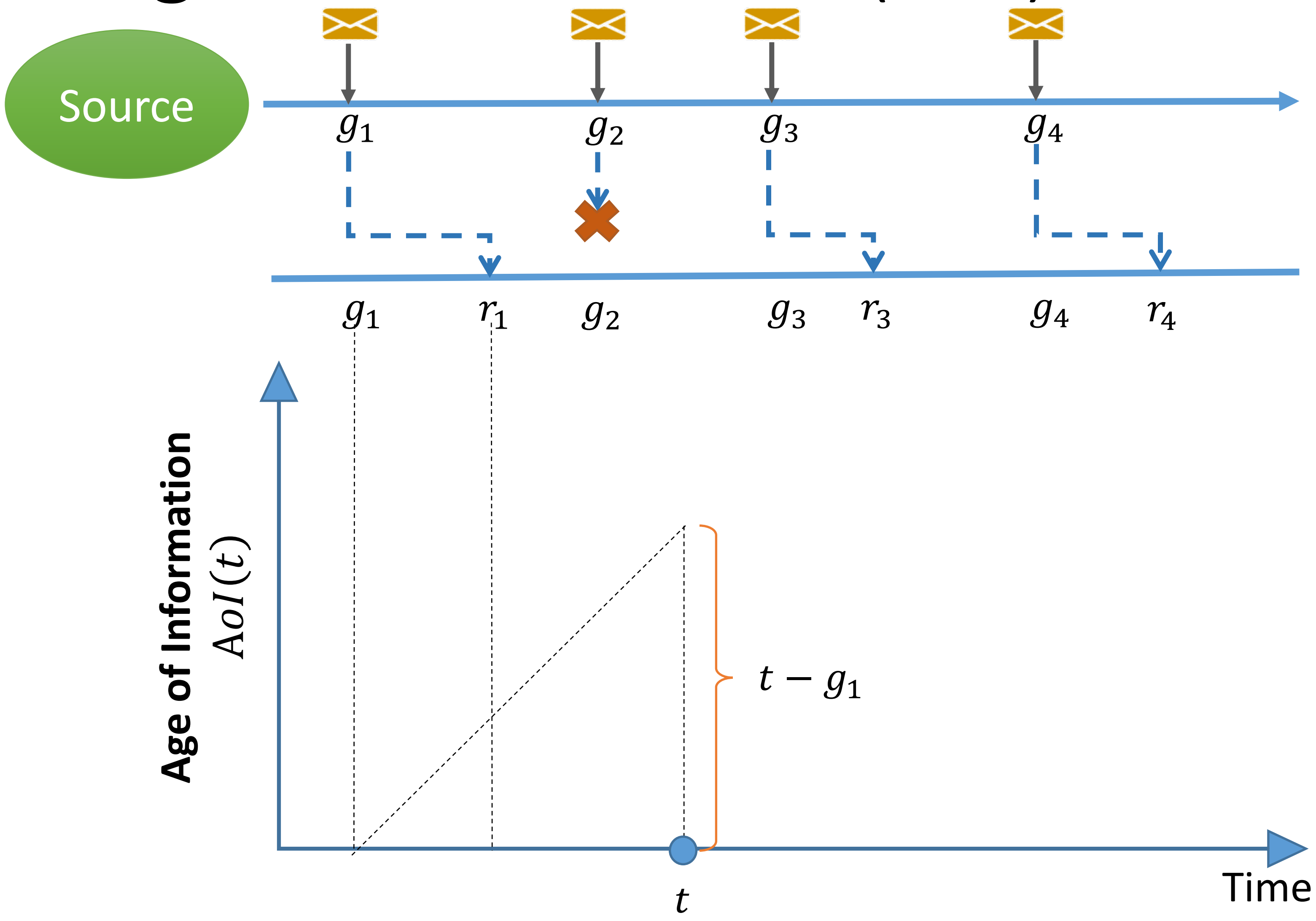
Age of Information (AoI)



Age of Information (AoI)

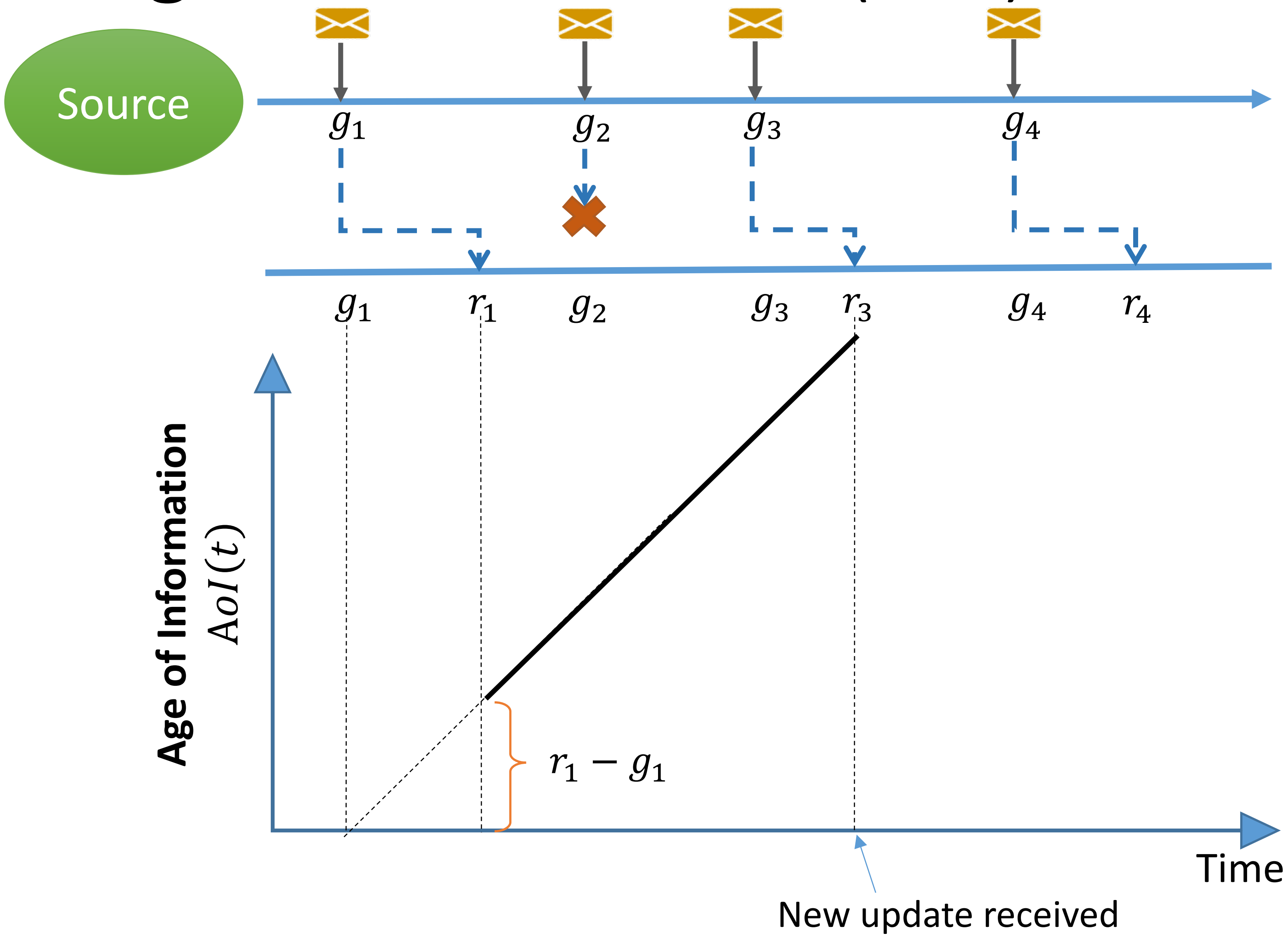


Age of Information (AoI)



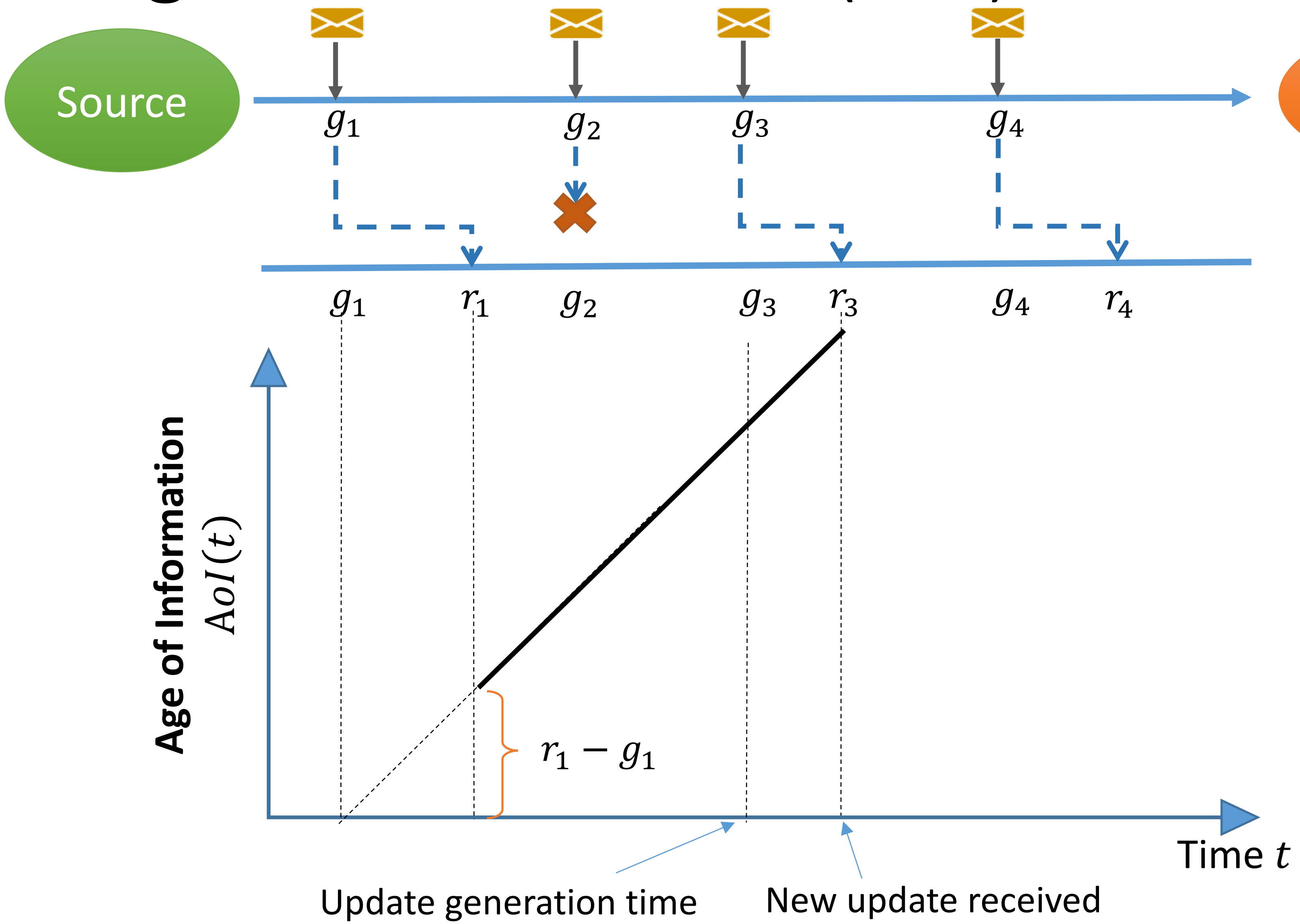
$$AoI(t) = t - \max_{i: r_i \le t} g_i$$

Age of Information (AoI)



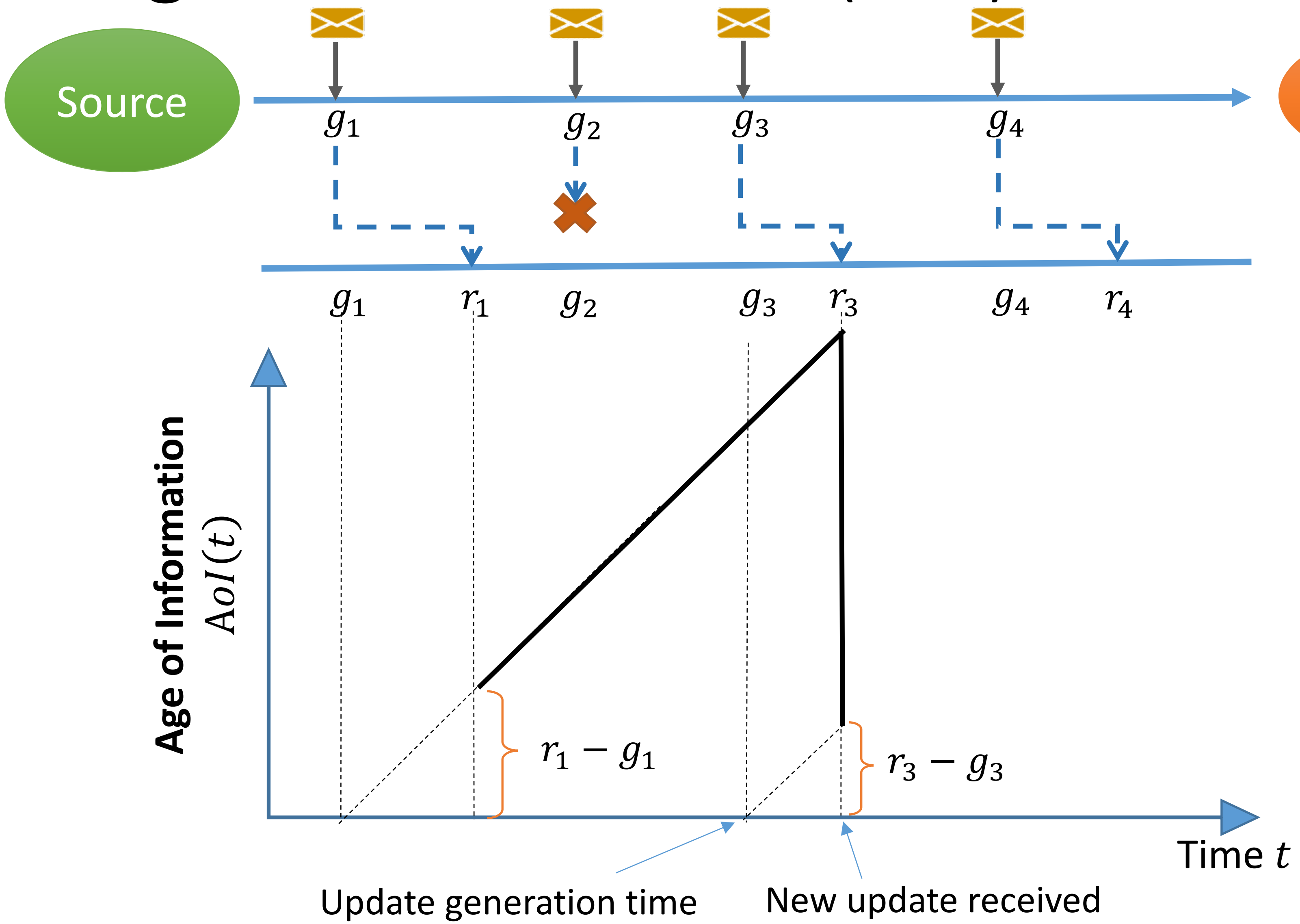
$$AoI(t) = t - \max_{i:r_i \leq t} g_i$$

Age of Information (AoI)



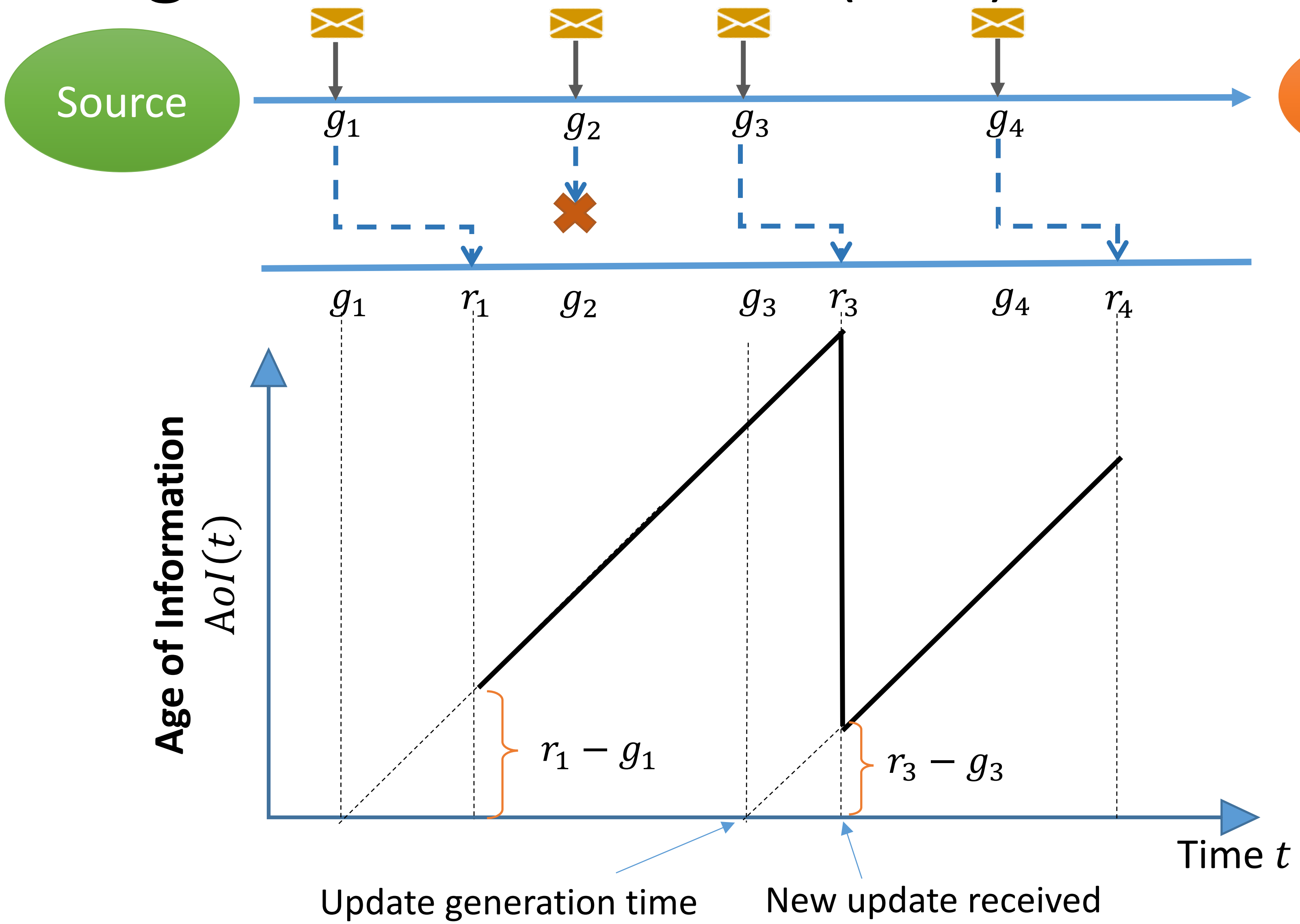
$$AoI(t) = t - \max_{i: r_i \leq t} g_i$$

Age of Information (AoI)



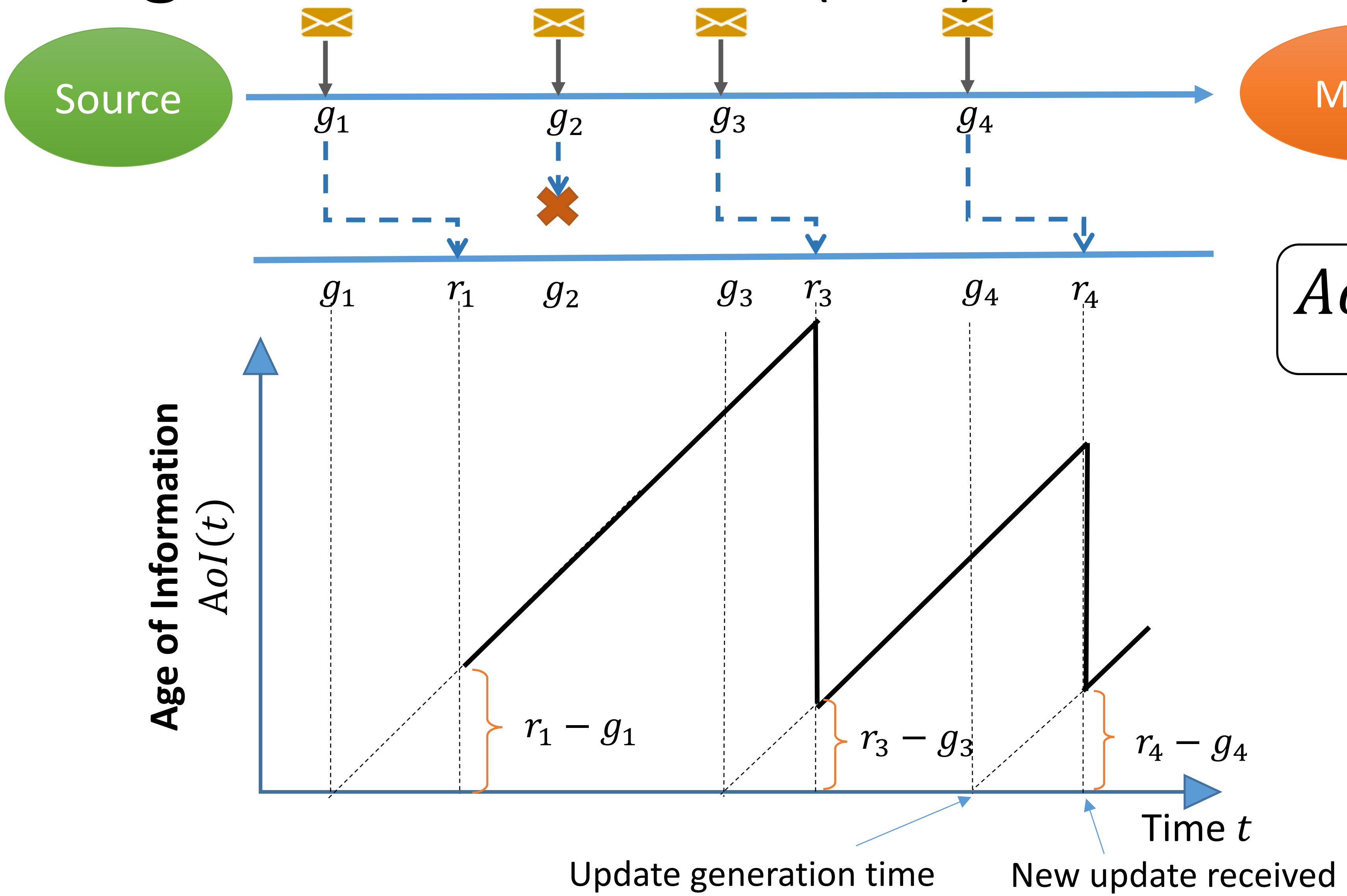
$$AoI(t) = t - \max_{i:r_i \leq t} g_i$$

Age of Information (AoI)



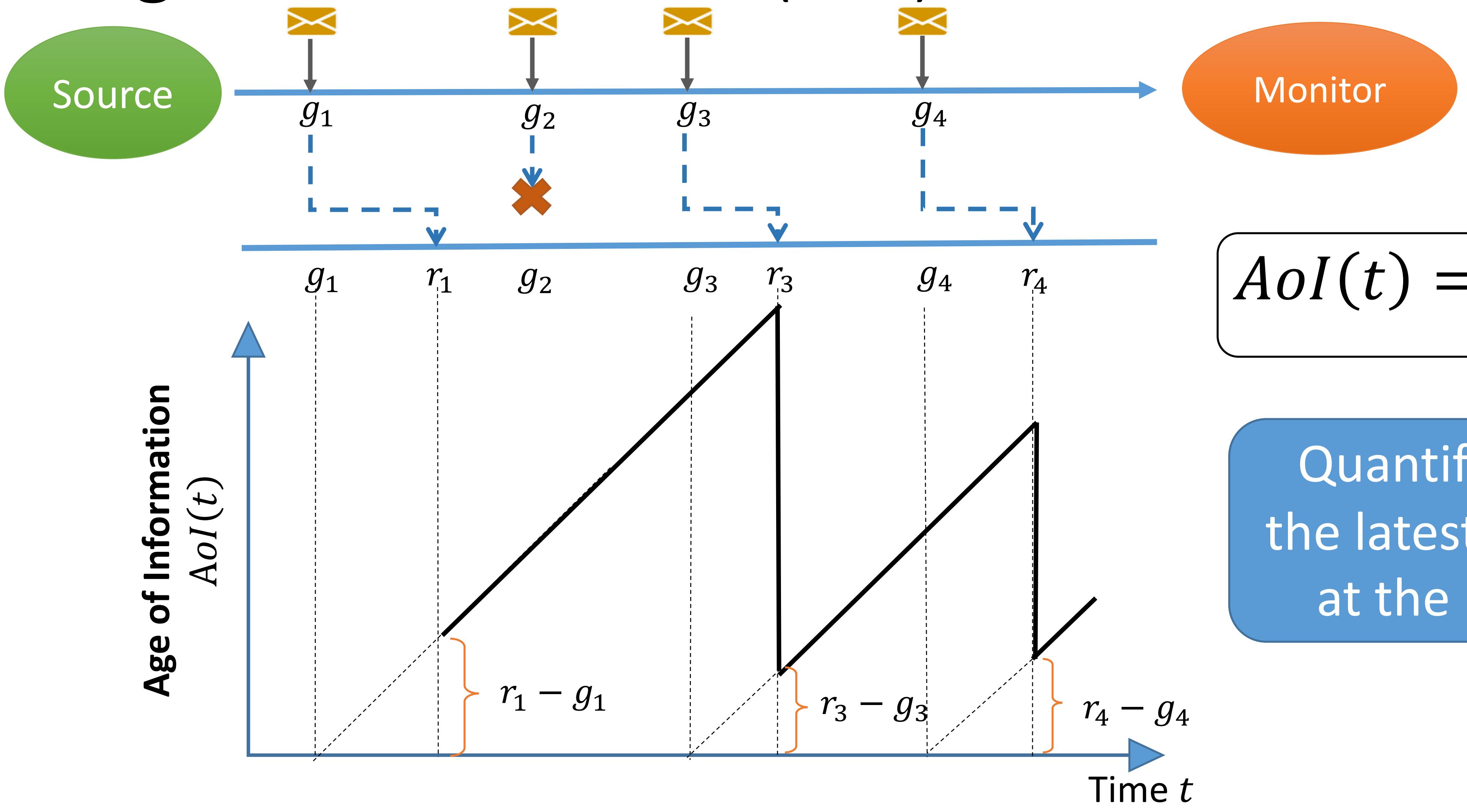
$$AoI(t) = t - \max_{i:r_i \leq t} g_i$$

Age of Information (AoI)



$$AoI(t) = t - \max_{i:r_i \leq t} g_i$$

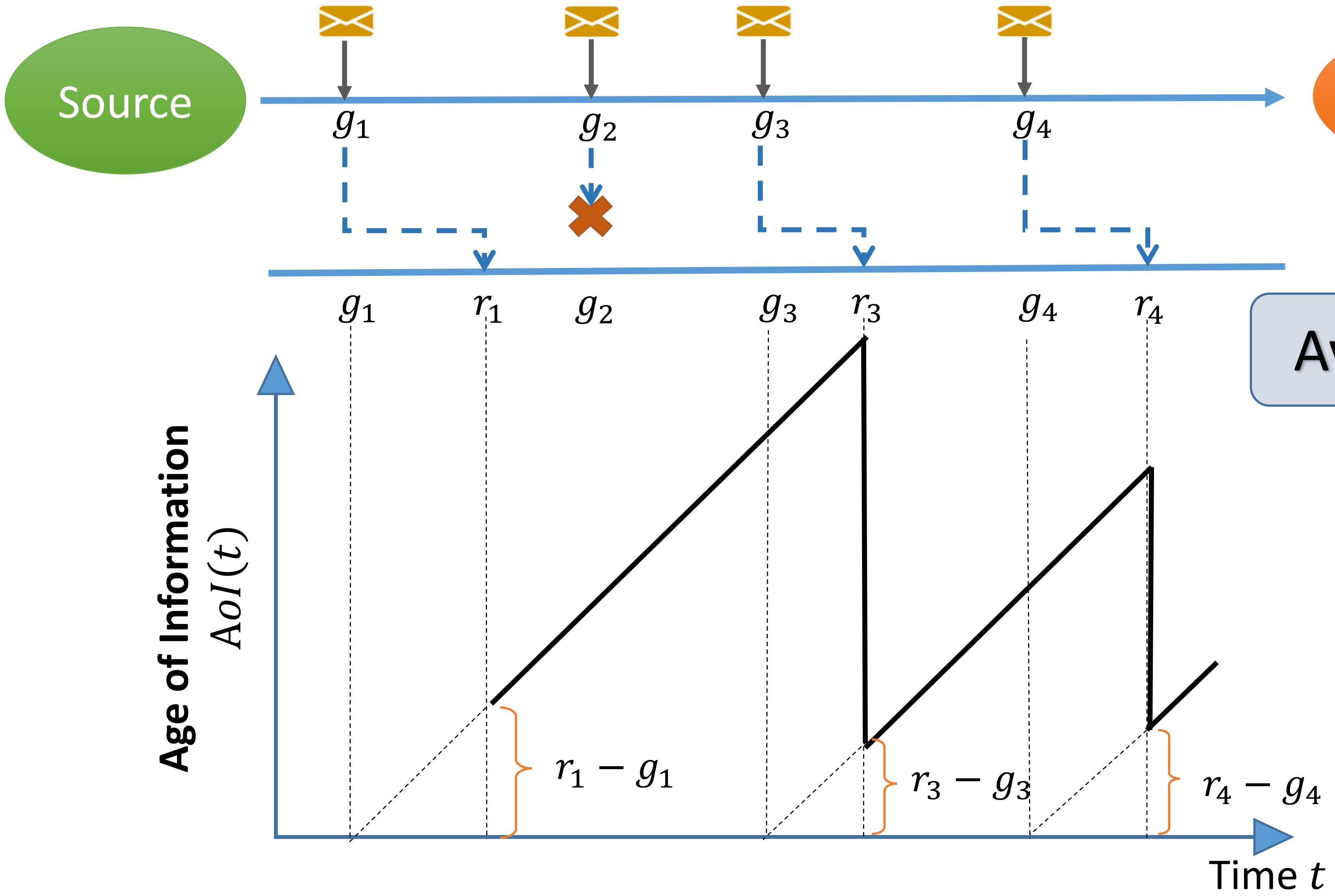
Age of Information (AoI)



$$AoI(t) = t - \max_{i:r_i \leq t} g_i$$

Quantifies how old the latest information at the monitor is.

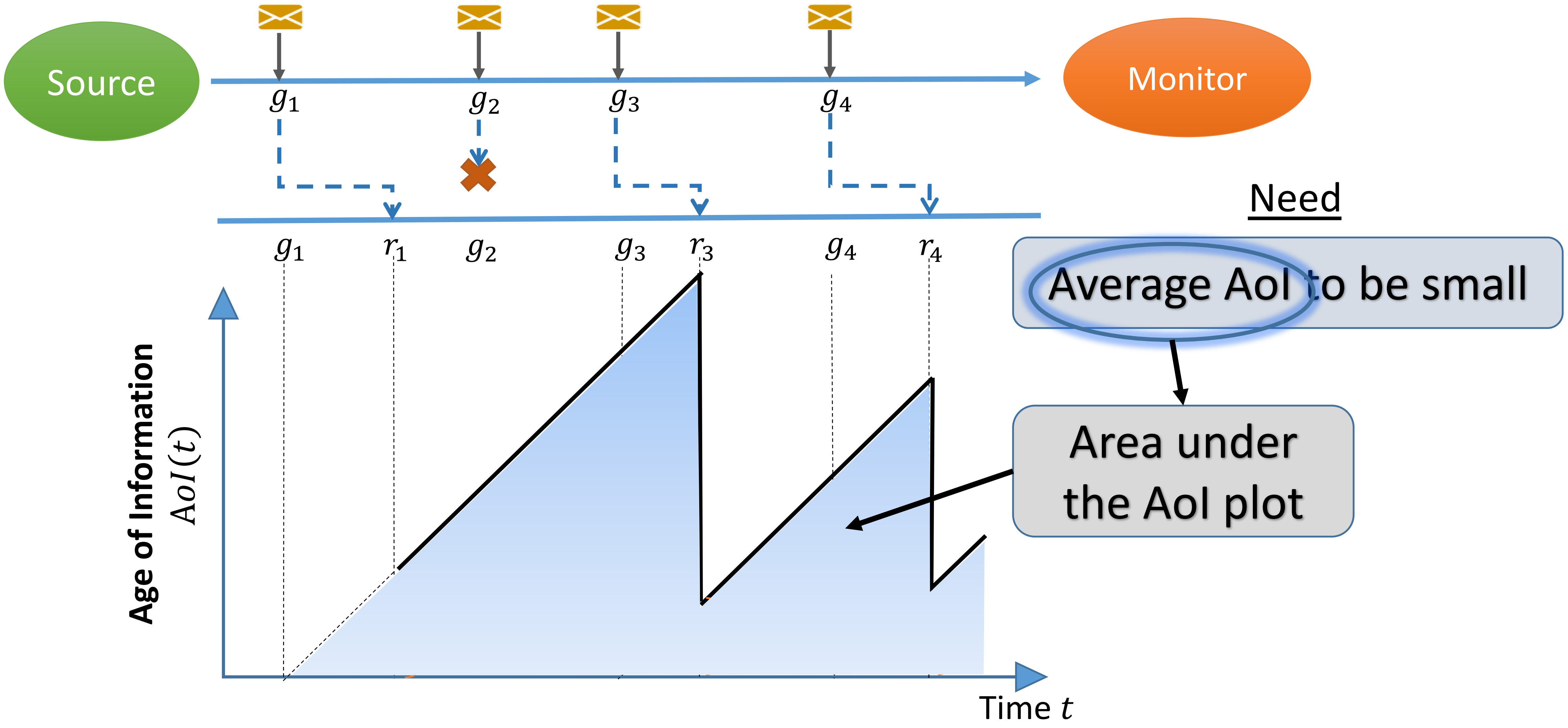
For Information Freshness



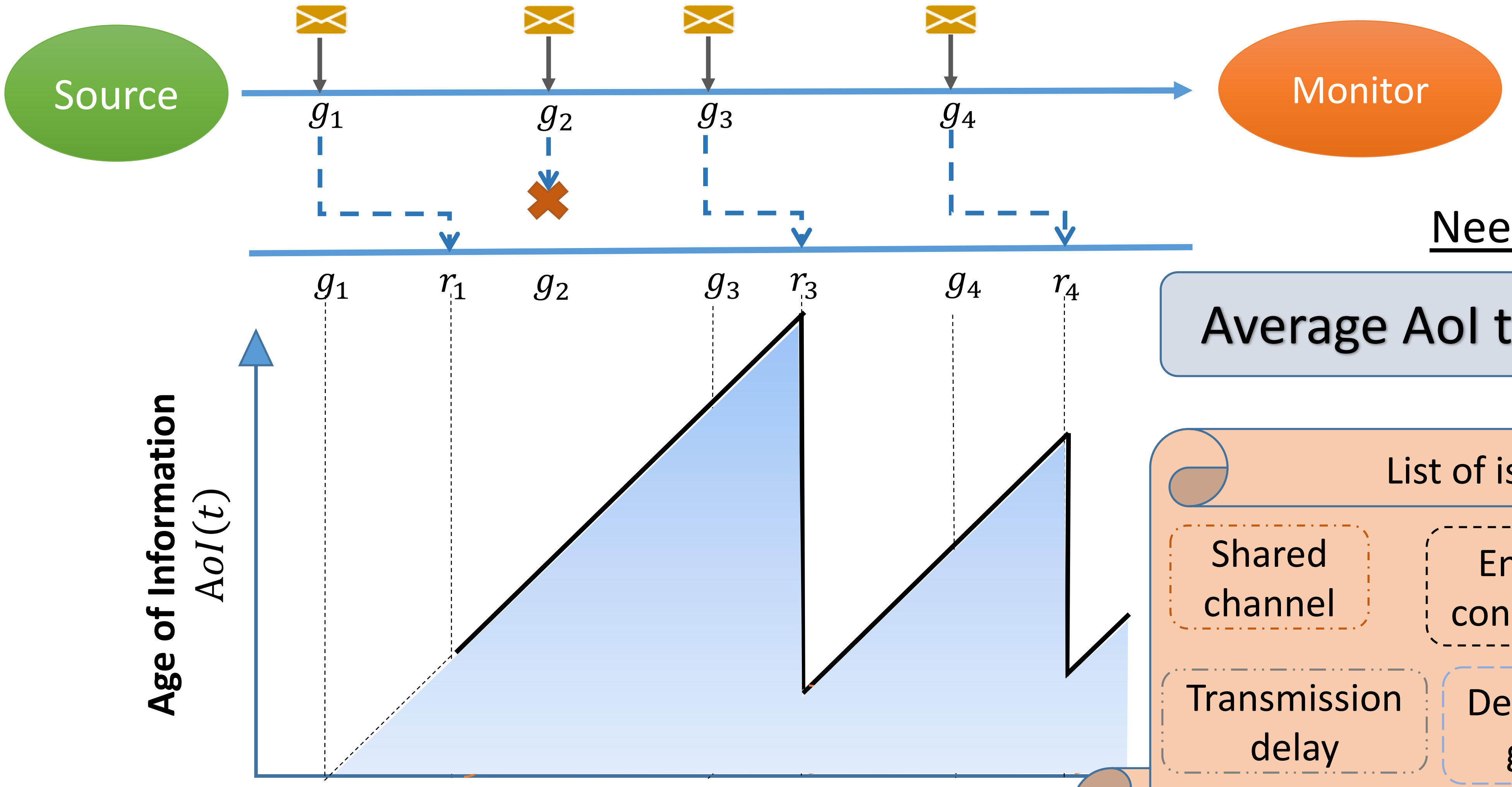
Need

Average AoI to be small

For Information Freshness



For Information Freshness



<u>Need</u>

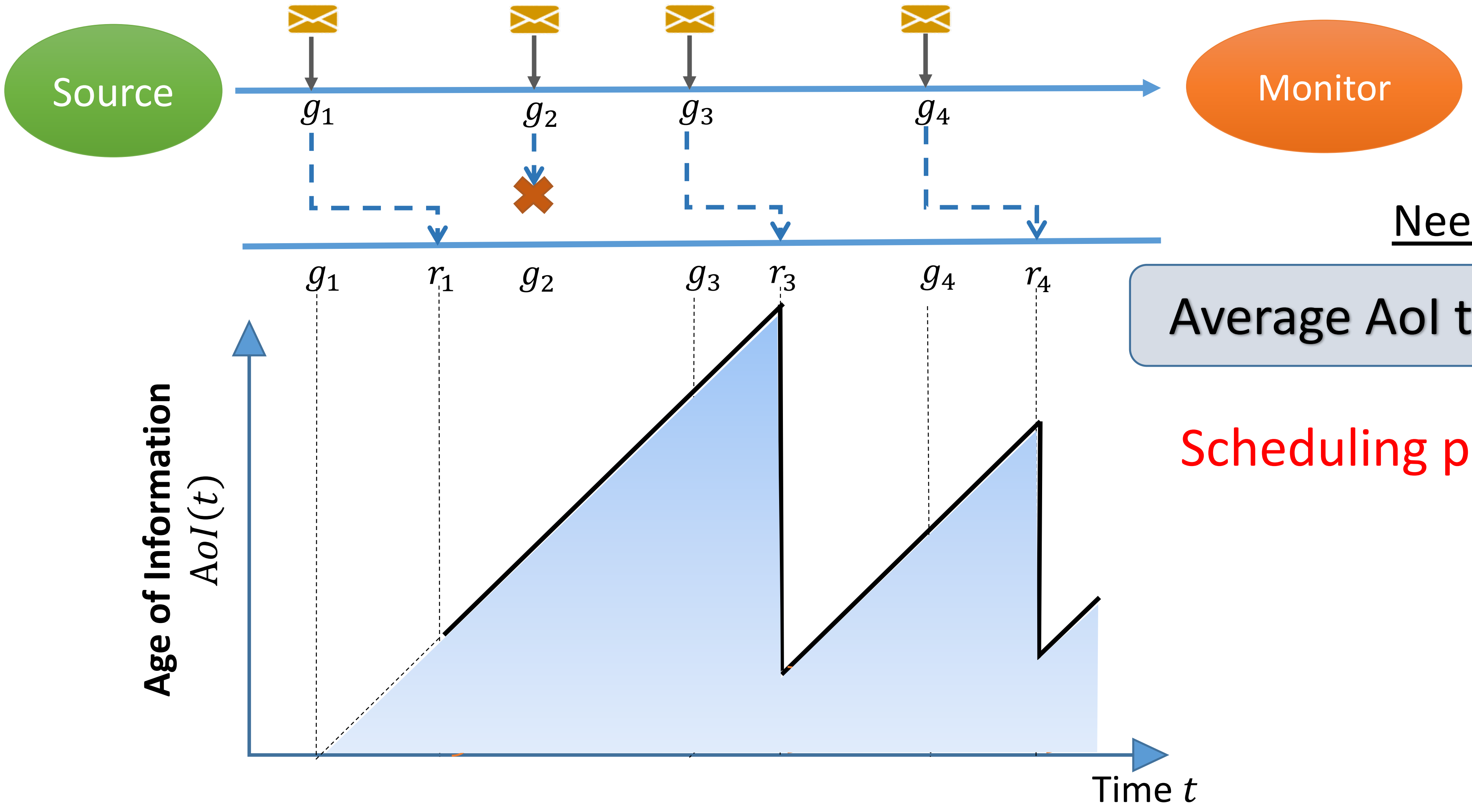
Average AoI to be small

List of issues…

- Shared channel
- Energy constraint
- Transmission delay
- Delay in update generation

⋮ ⋮ ⋮

For Information Freshness



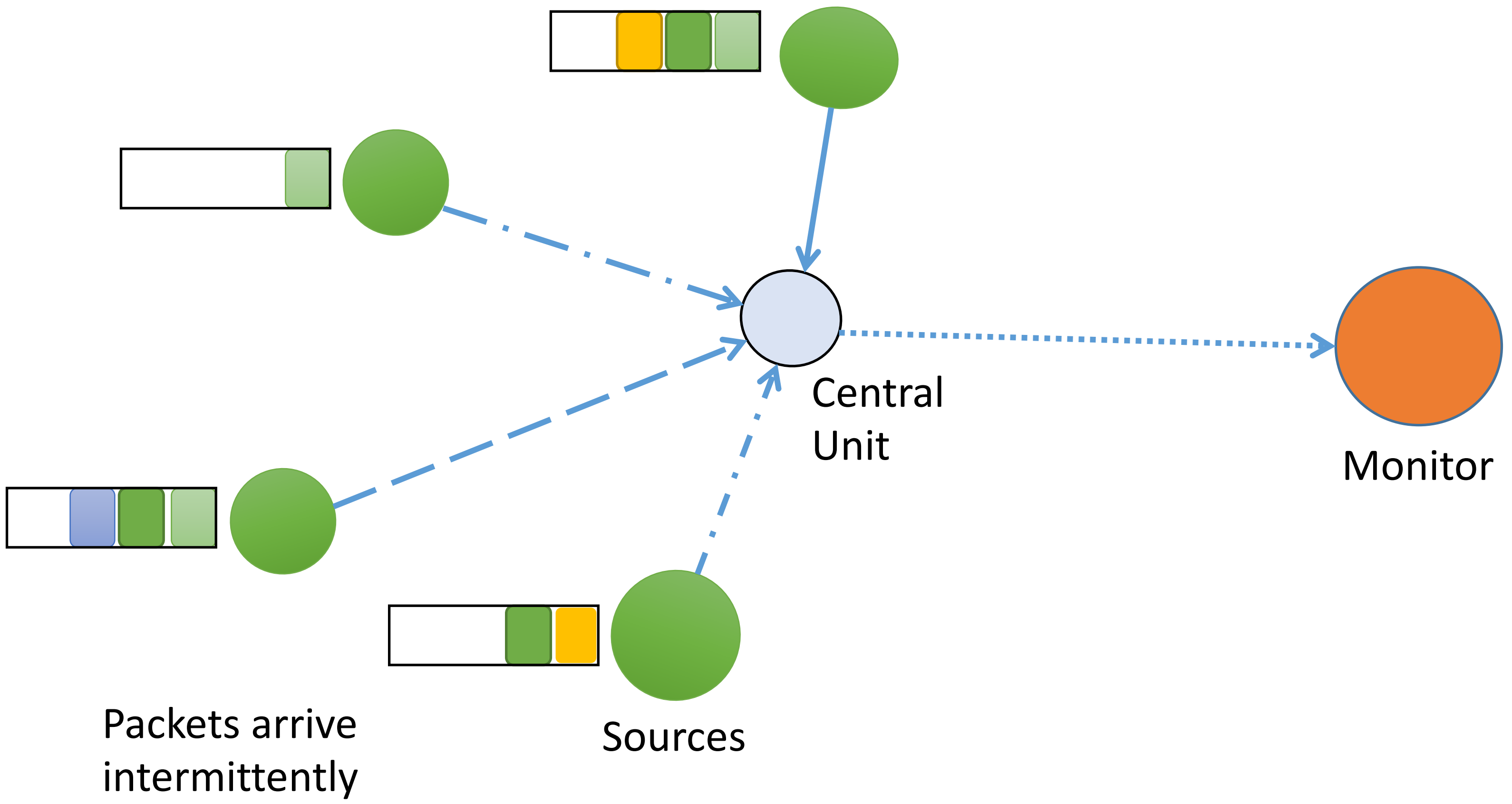
Need

Average AoI to be small

Scheduling problems!!

In this talk ....

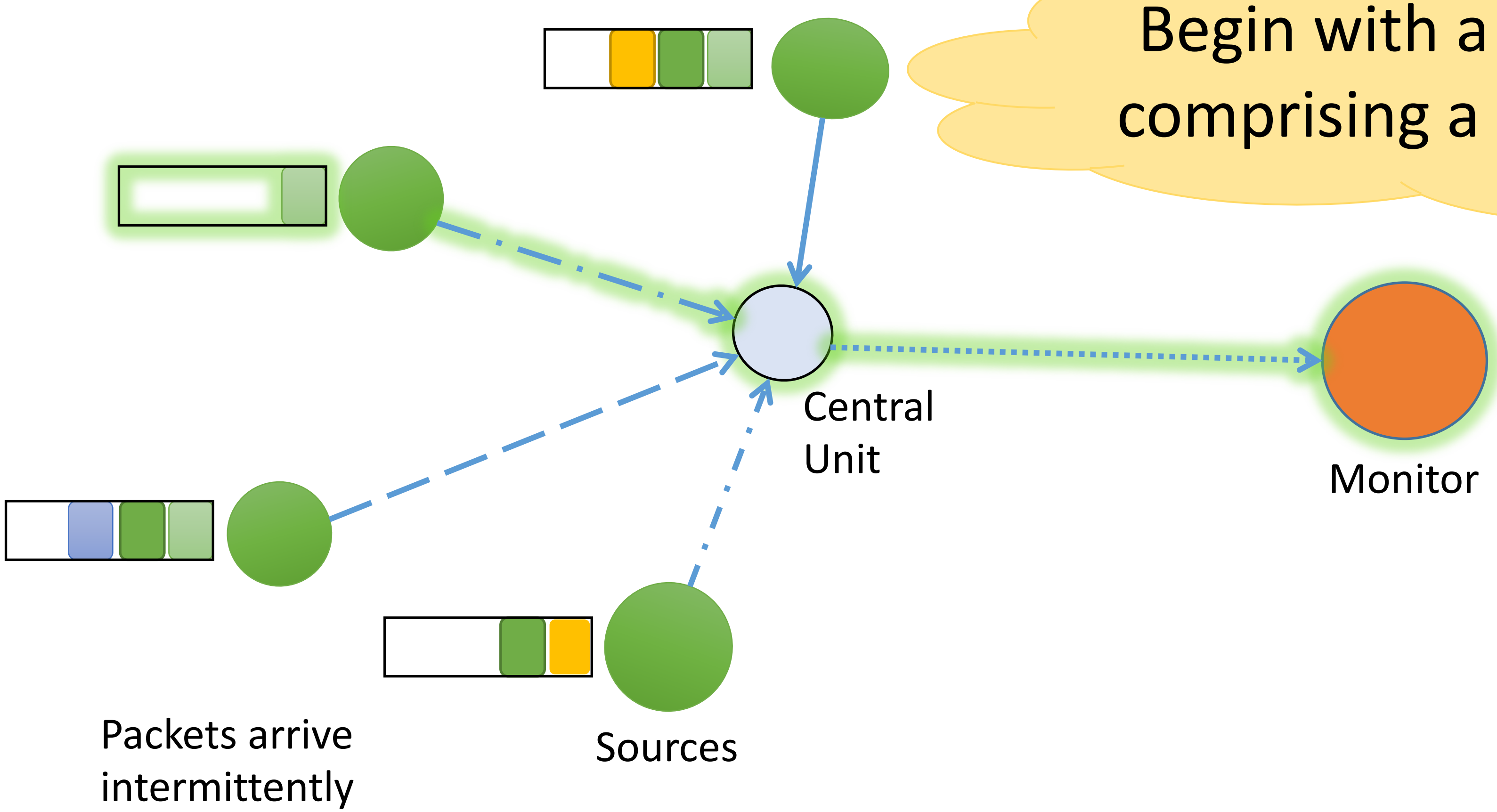
AoI scheduling for a generic network!!



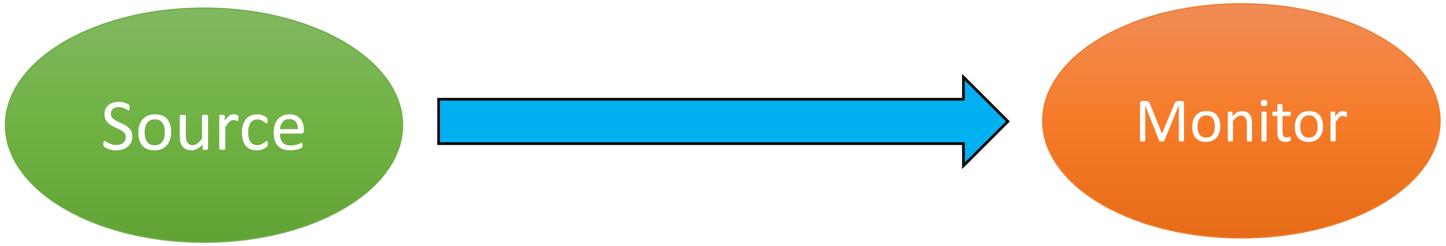
In this talk ....

AoI scheduling for a generic network!!

Begin with a simpler setting comprising a single source ....

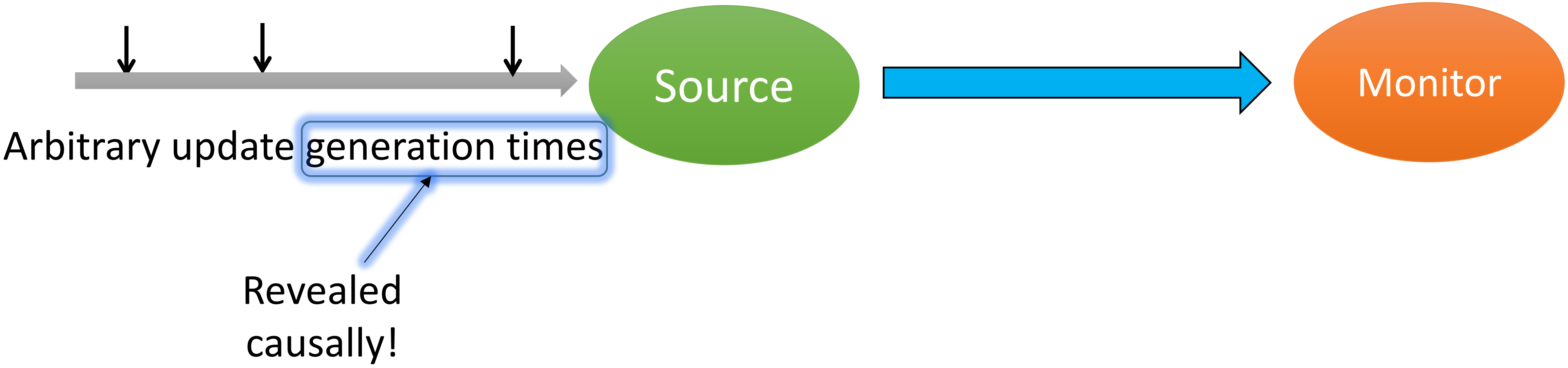


System Model

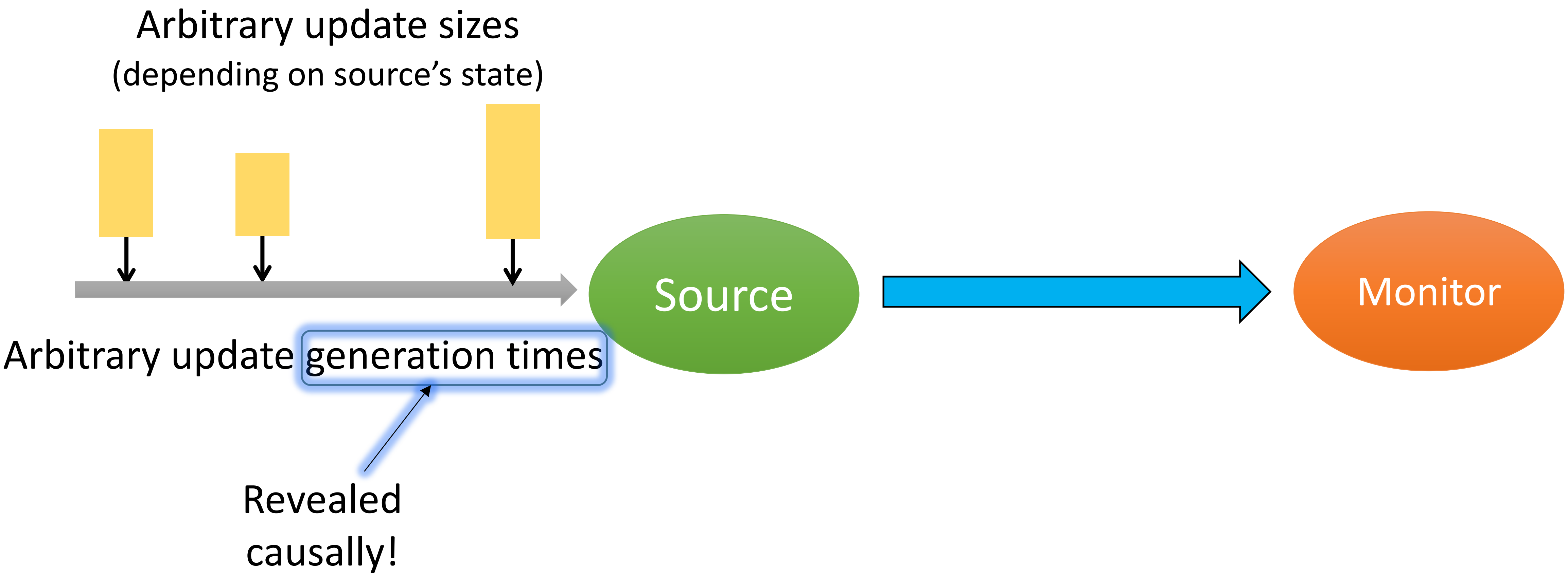


System Model

System Model

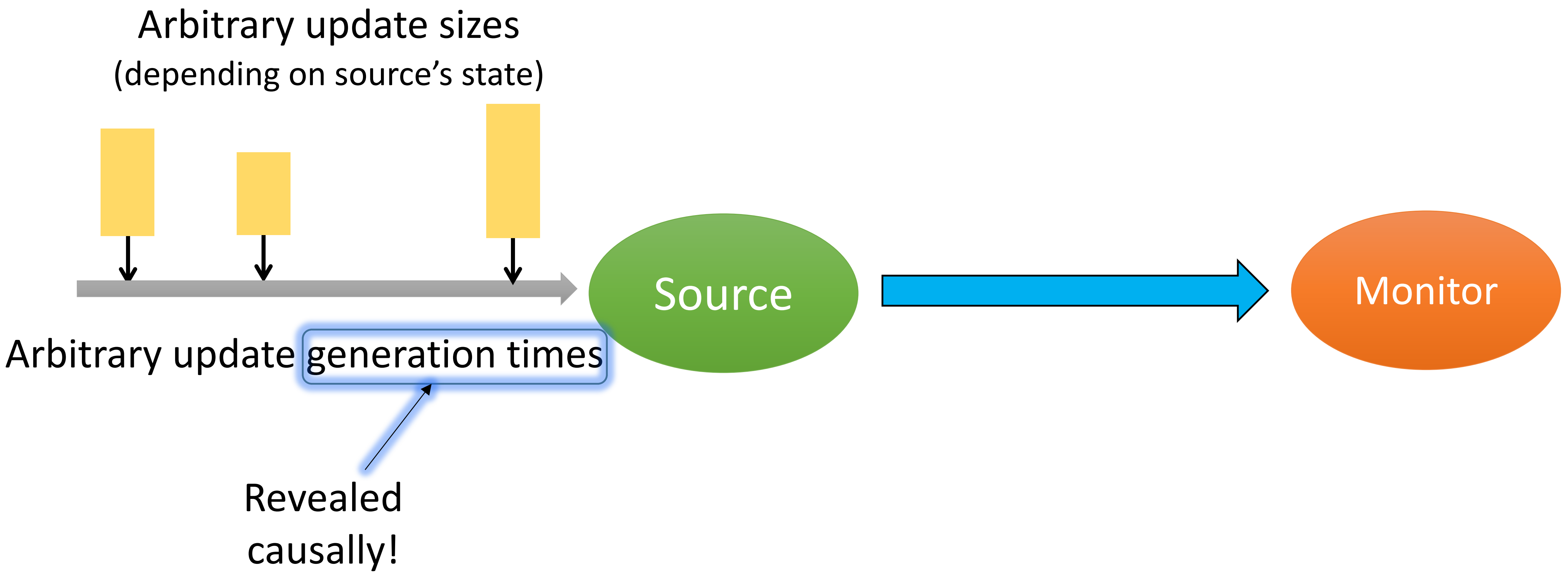


System Model

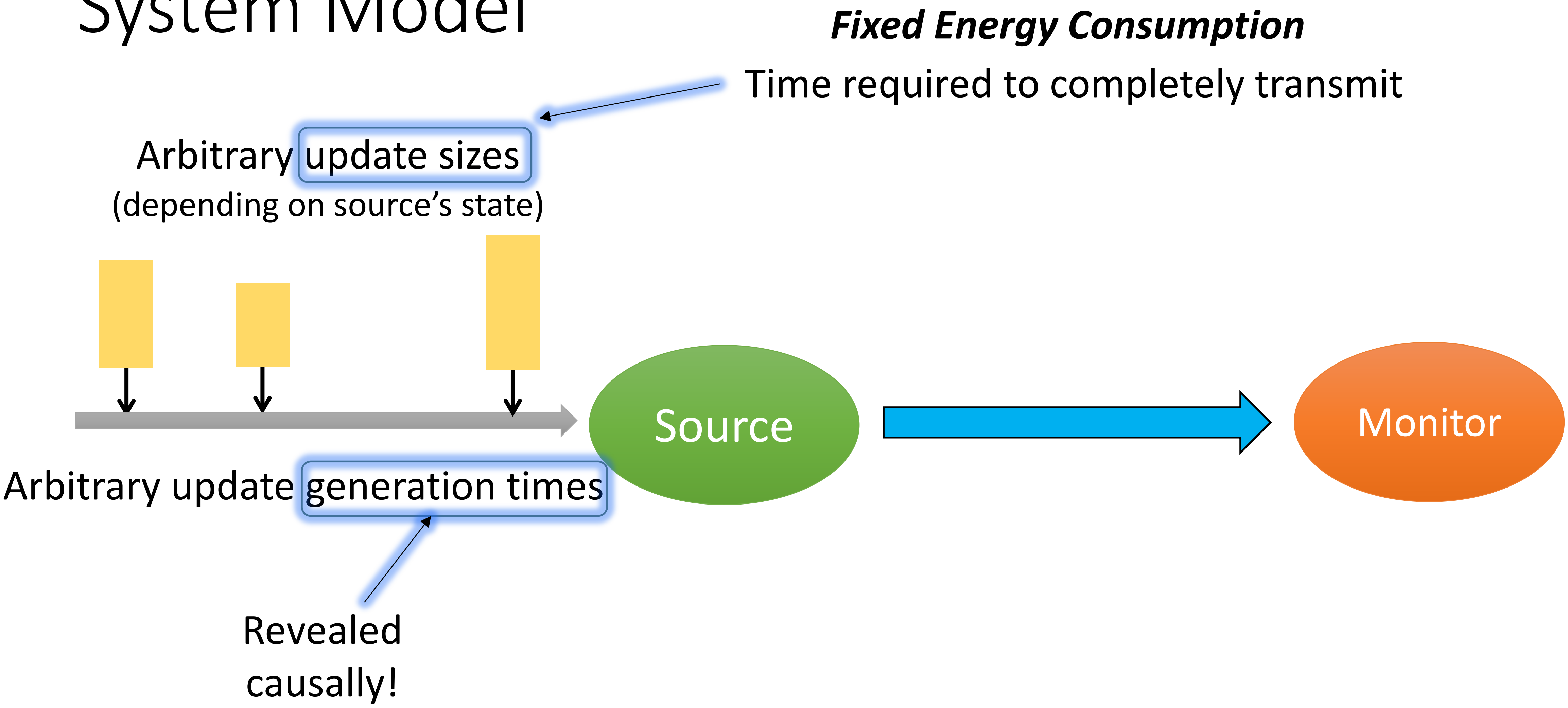


System Model

Fixed Energy Consumption



System Model



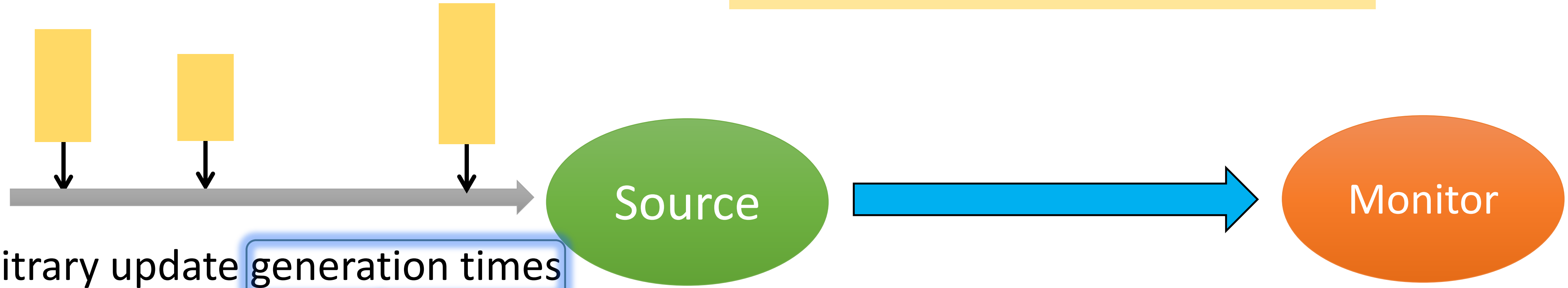
System Model

Fixed Energy Consumption

Time required to completely transmit

Energy consumed at rate $'c'$ per unit transmission time

Arbitrary update sizes
(depending on source's state)



Arbitrary update generation times

Revealed causally!

System Model

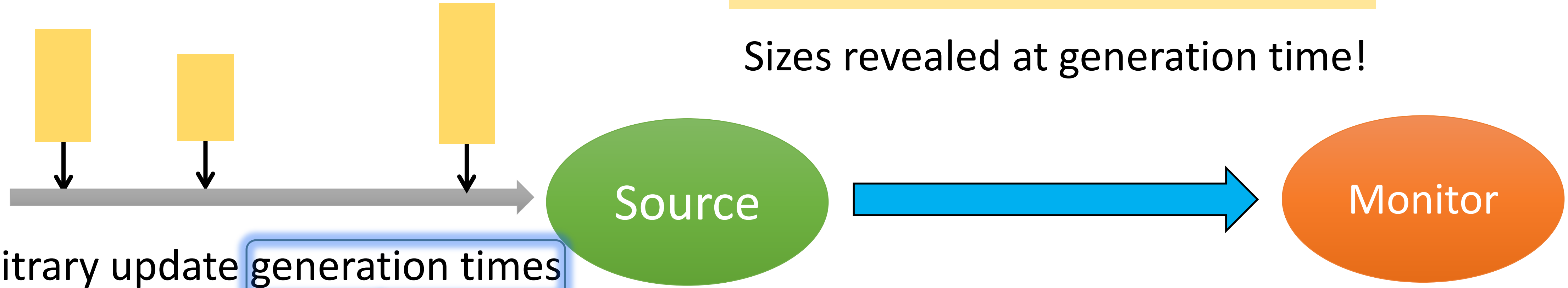
Fixed Energy Consumption

Time required to completely transmit

Energy consumed at rate $'c'$ per unit transmission time

Sizes revealed at generation time!

Arbitrary update sizes
(depending on source's state)



Arbitrary update generation times

Revealed causally!

System Model

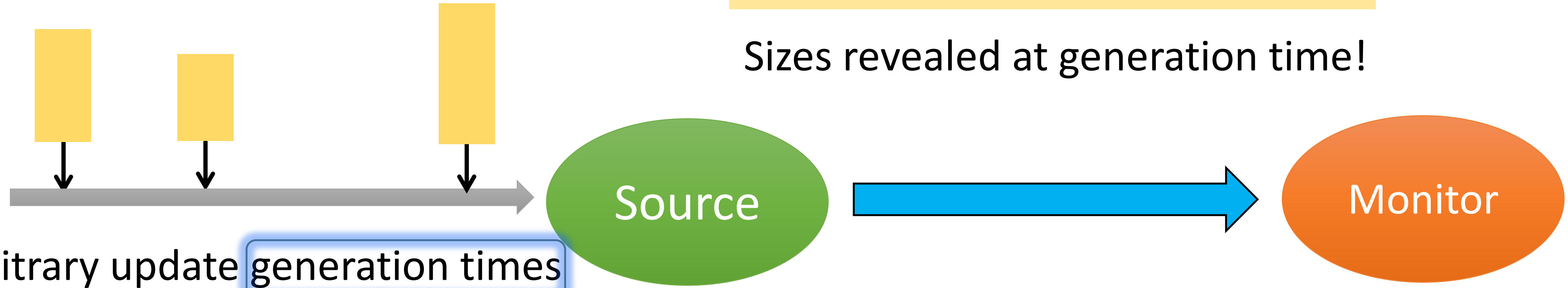
Fixed Energy Consumption

Time required to completely transmit

Energy consumed at rate $'c'$ per unit transmission time

Sizes revealed at generation time!

Arbitrary update sizes
(depending on source's state)



Arbitrary update generation times

Revealed causally!

At any time, the source can transmit
- **any** available update (at most **one**)
- preemption is allowed

Objective

Minimize the sum of average AoI and average energy cost

Objective

Minimize the sum of average AoI and average energy cost

$$\min \lim_{T\to\infty} \frac{1}{T}\int_0^T (AoI(t) + c \cdot u(t))dt$$

where $u(t) = 1$ if an update is under transmission at time t, and 0 otherwise.

Objective

Minimize the sum of average AoI and average energy cost

$$\min \lim_{T\to\infty} \frac{1}{T}\int_0^T \left(AoI(t) + c\cdot u(t)\right)dt$$

where $u(t) = 1$ if an update is under transmission at time t, and 0 otherwise.

Online Decision Problem:

At any time, which update to transmit (or, not to transmit)?

Objective

Minimize the sum of average AoI and average energy cost

$$\min \lim_{T\to\infty} \frac{1}{T}\int_0^T (AoI(t) + c\cdot u(t))dt$$

where $u(t) = 1$ if an update is under transmission at time t, and 0 otherwise.

Online Decision Problem:

At any time, which update to transmit (or, not to transmit)?

Challenges??

Prior Work

Prior Work

- Response-time minimization

Prior Work

latency

- Response-time minimization

Prior Work

latency

flow-time

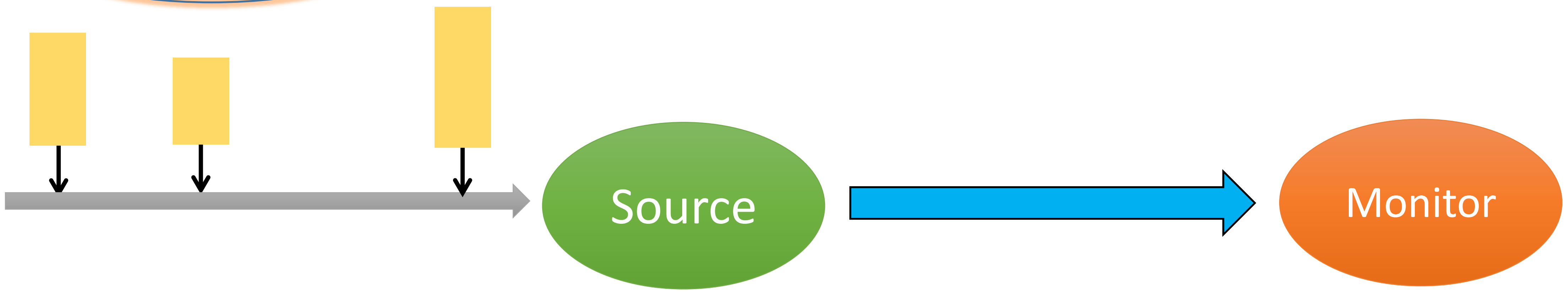
- Response-time minimization

Prior Work

- Response-time minimization

latency

flow-time



Prior Work

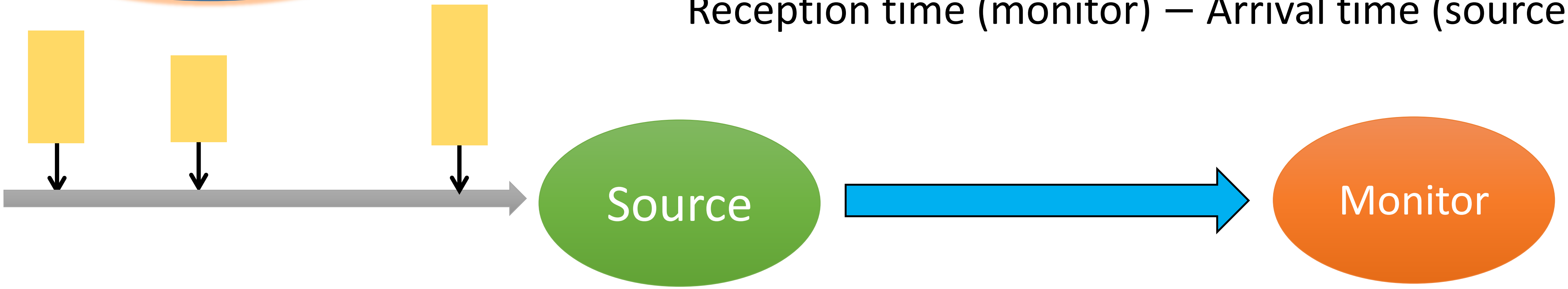
flow-time

latency

- Response-time minimization

Response-time:

Reception time (monitor) – Arrival time (source)



Prior Work

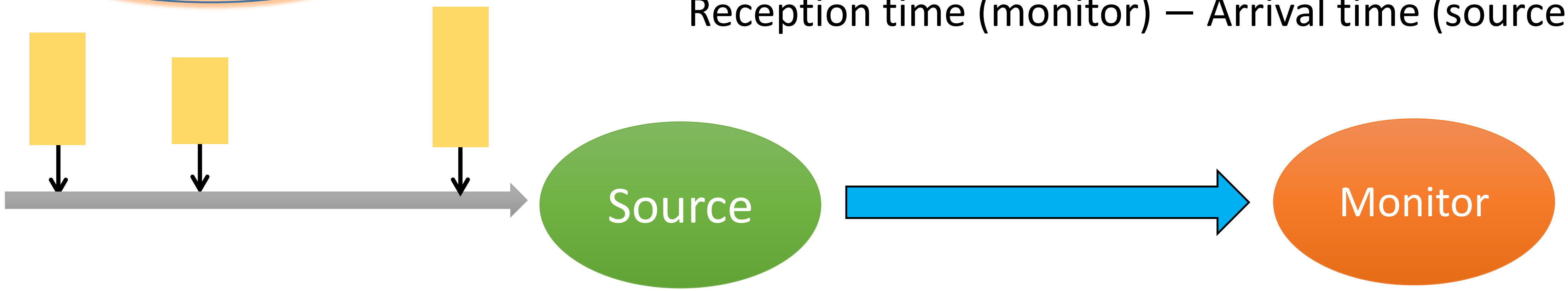
latency

flow-time

- Response-time minimization

Response-time:

Reception time (monitor) – Arrival time (source)



Objective: Schedule packet transmissions to minimize mean response-time.

Prior Work

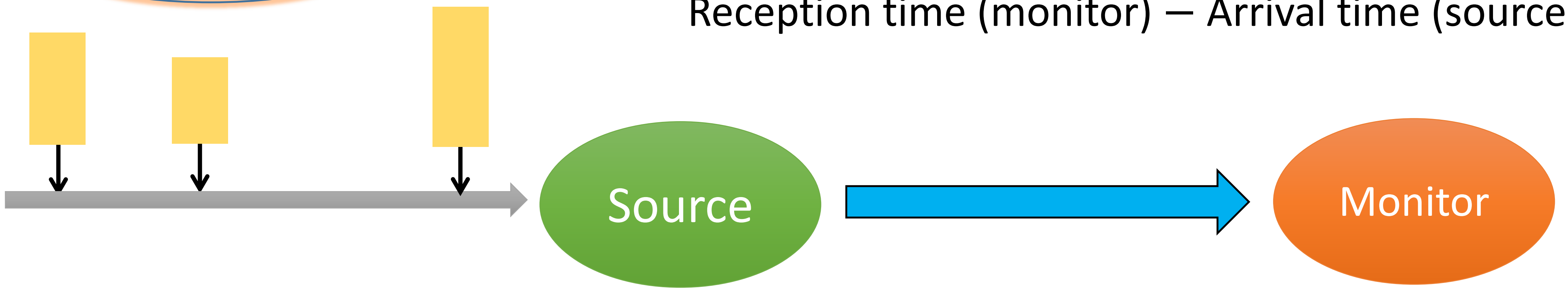
flow-time

latency

- Response-time minimization

Response-time:

Reception time (monitor) – Arrival time (source)



Objective: Schedule packet transmissions to minimize mean response-time.

SRPT (Shortest Remaining Processing Time):

At any time, transmit the packet with least remaining size.

Prior Work

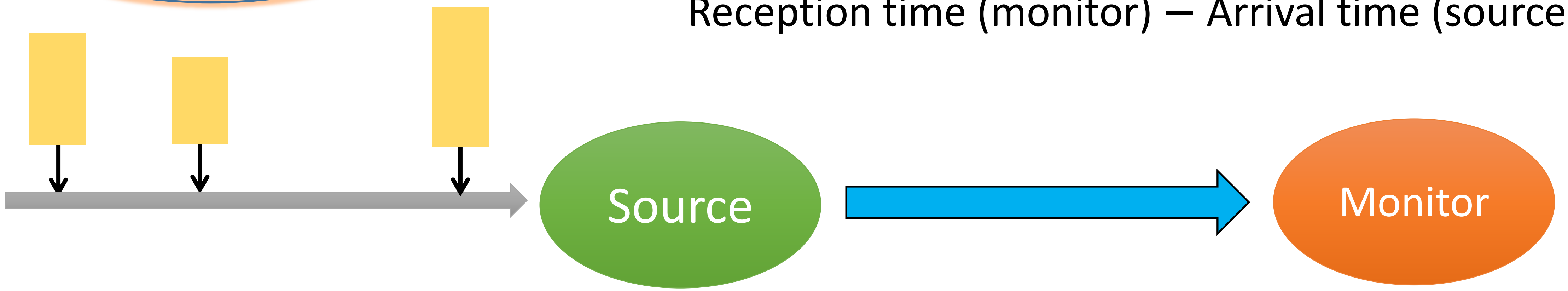
flow-time

latency

- Response-time minimization

Response-time:

Reception time (monitor) − Arrival time (source)



Objective: Schedule packet transmissions to minimize mean response-time.

SRPT (Shortest Remaining Processing Time):

At any time, transmit the packet with least remaining size.

Optimal!

Prior Work

Differences

Response-time minimization

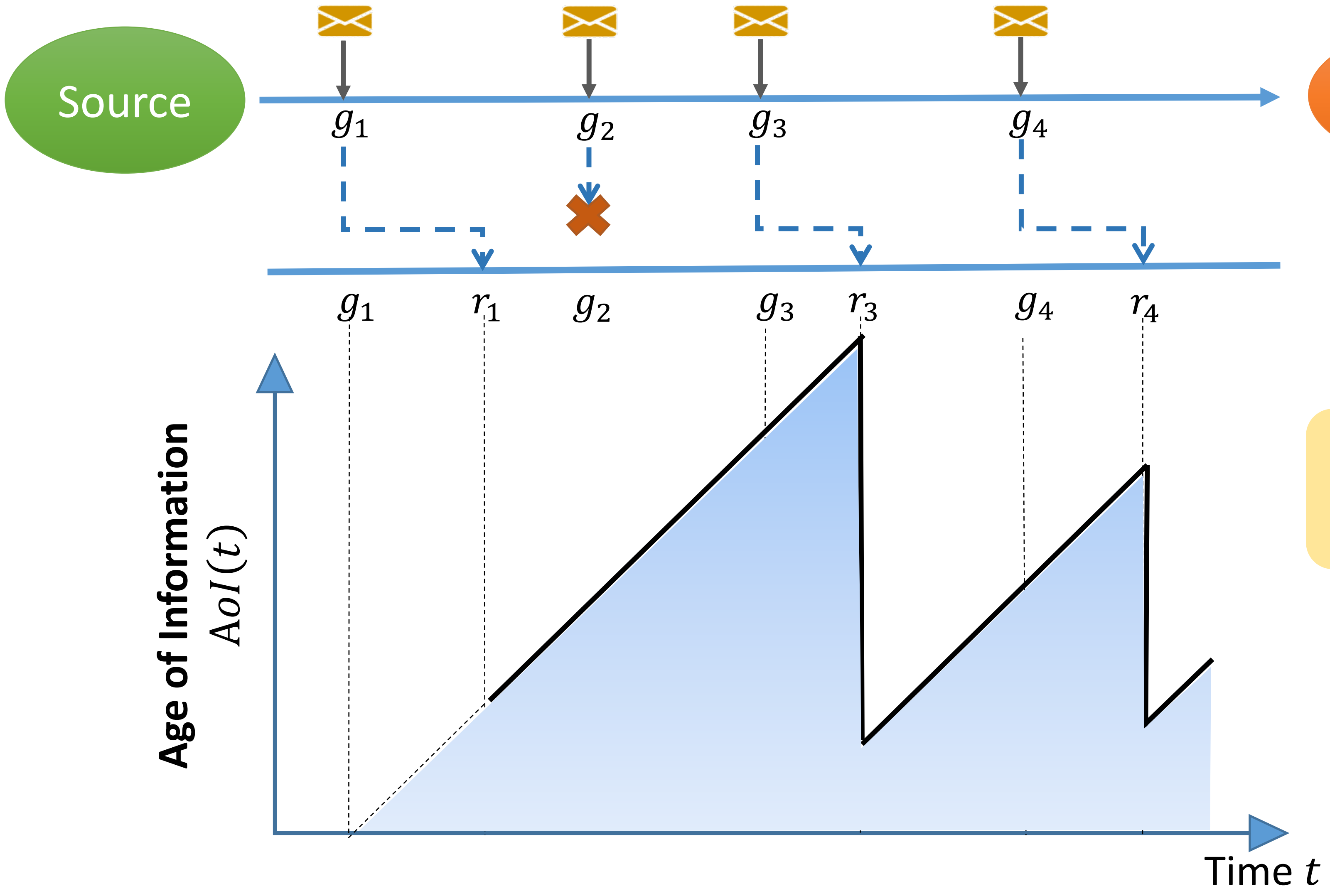
Considered problem

Prior Work

Differences

Response-time minimization		Considered problem
All packets need to be transmitted	→	Sufficient to transmit a **subset of updates**

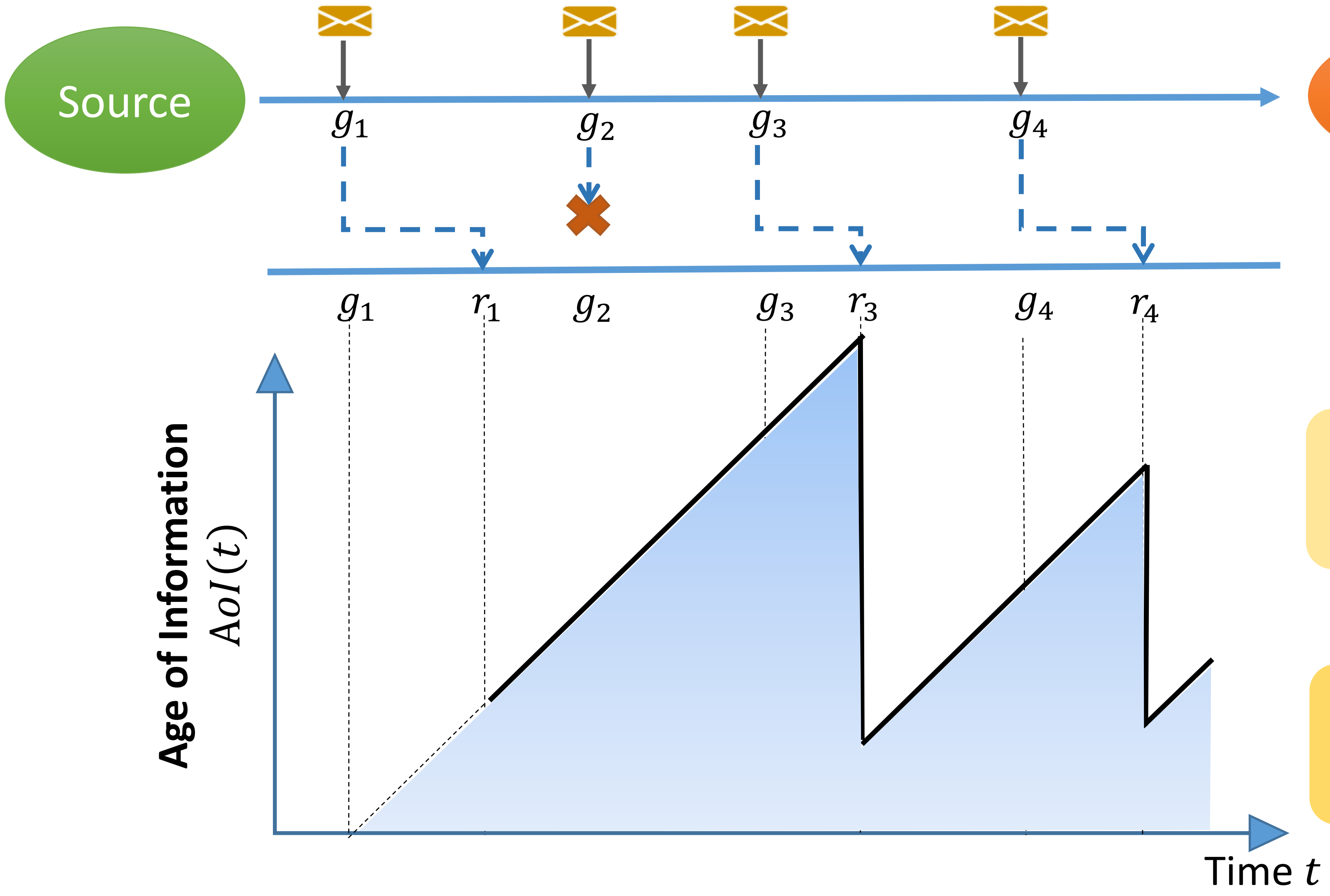
Recall...



$$AoI(t) = t - \max_{i: r_i \le t} g_i$$

At any time, AoI depends only on latest transmitted update

Recall…



$$AoI(t) = t - \max_{i:r_i \le t} g_i$$

At any time, AoI depends only on latest transmitted update

Few updates may be skipped to save energy

Prior Work

Differences

Combinatorial nature!

<u>Response-time minimization</u> → <u>Considered problem</u>

All packets need to be transmitted → Sufficient to transmit a **subset of updates**

Prior Work

Combinatorial nature!

Differences

<u>Response-time minimization</u>		<u>Considered problem</u>
All packets need to be transmitted	→	Sufficient to transmit a **subset of updates**
Energy cost **independent** of policy	→	Energy cost depends on updates transmitted

Prior Work

Differences

Combinatorial nature!

Response-time minimization		Considered problem
All packets need to be transmitted	→	Sufficient to transmit a **subset of updates**
Energy cost **independent** of policy	→	Energy cost depends on updates transmitted

$SRPT$ is optimal

Prior Work

Differences

Combinatorial nature!

<u>Response-time minimization</u>

All packets need to be transmitted

Energy cost **independent** of policy

$SRPT$ is optimal

<u>Considered problem</u>

Sufficient to transmit a **subset of updates**

Energy cost depends on updates transmitted

$SRPT$ arbitrarily bad compared to OPT

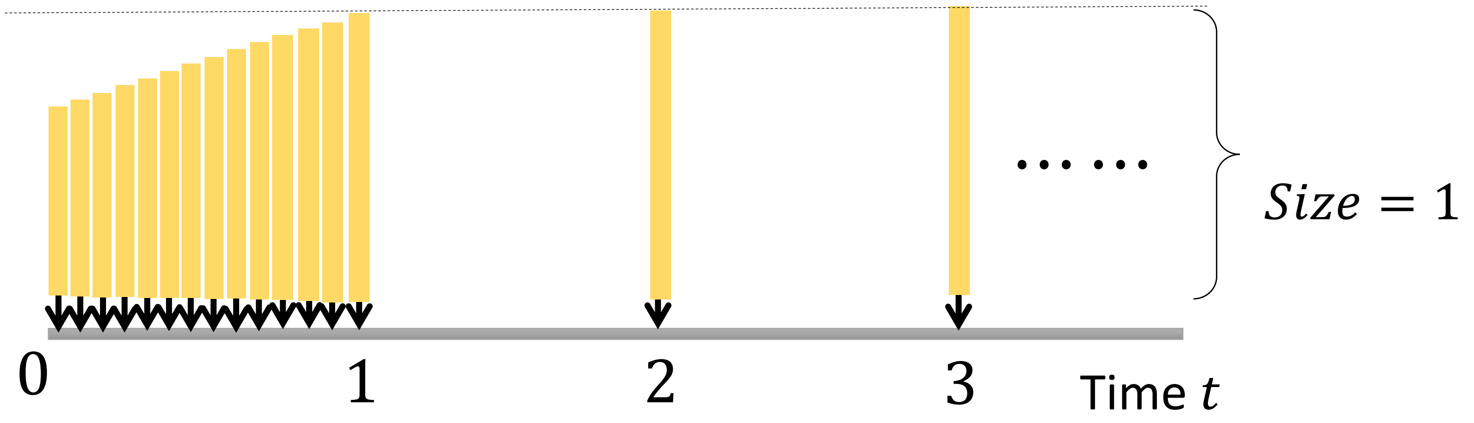
Prior Work

Differences

Combinatorial nature!

Response-time minimization		Considered problem
All packets need to be transmitted	→	Sufficient to transmit a **subset of updates**
Energy cost **independent** of policy	→	Energy cost depends on updates transmitted
$SRPT$ is optimal		$SRPT$ arbitrarily bad compared to OPT

Example



Prior Work

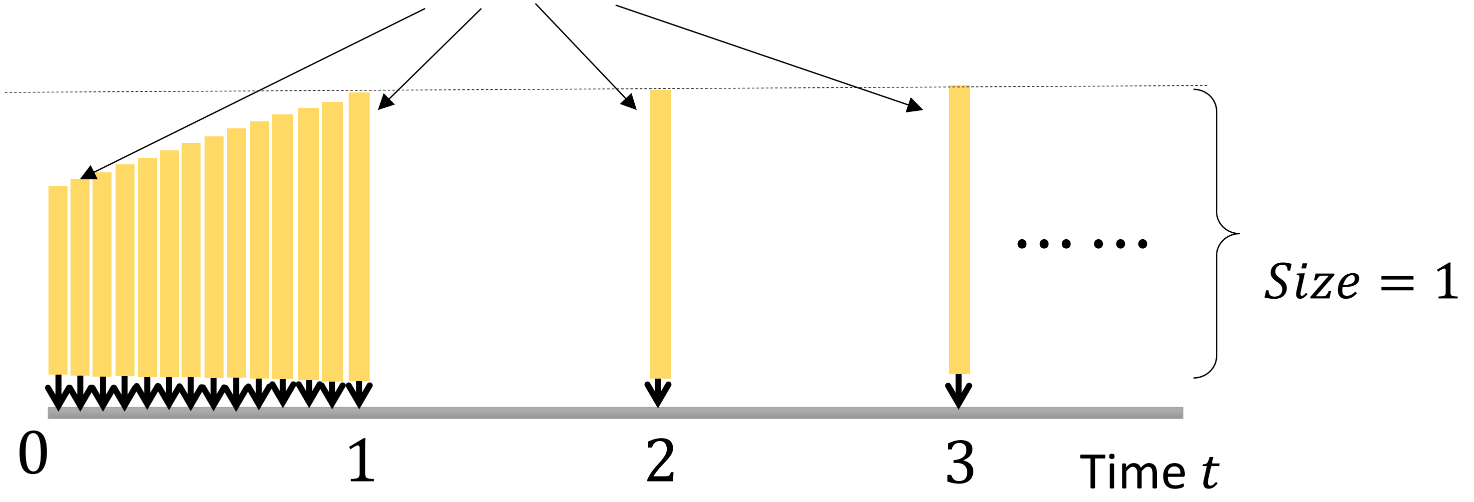
Differences

Combinatorial nature!

Response-time minimization		Considered problem
All packets need to be transmitted	→	Sufficient to transmit a **subset of updates**
Energy cost **independent** of policy	→	Energy cost depends on updates transmitted
$SRPT$ is optimal		$SRPT$ arbitrarily bad compared to OPT

Example

OPT may transmit



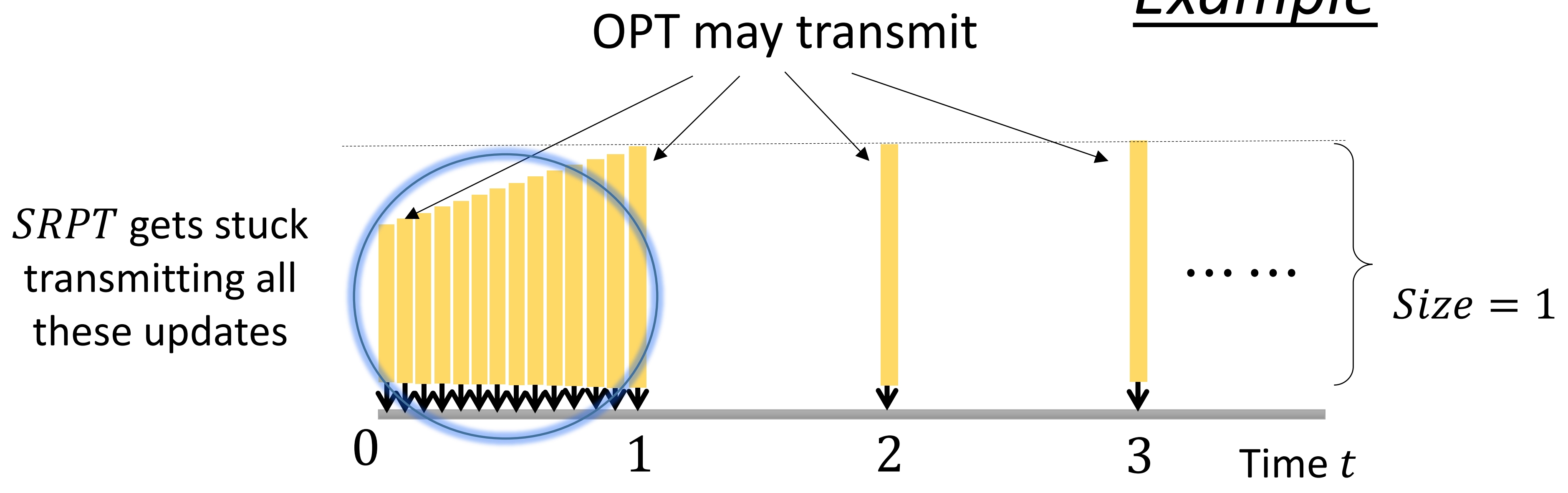
Prior Work

Differences

Combinatorial nature!

Response-time minimization		Considered problem
All packets need to be transmitted	→	Sufficient to transmit a **subset of updates**
Energy cost **independent** of policy	→	Energy cost depends on updates transmitted
$SRPT$ is optimal		$SRPT$ arbitrarily bad compared to OPT

Example



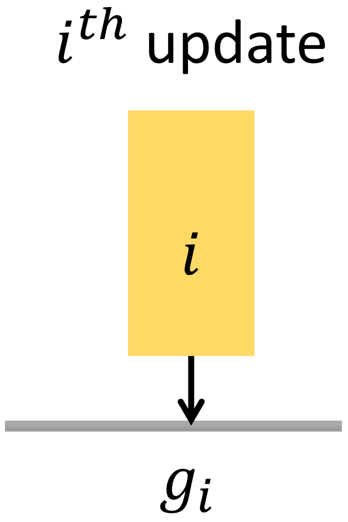
Which update to transmit?

Which update to transmit?

Notations:

Which update to transmit?

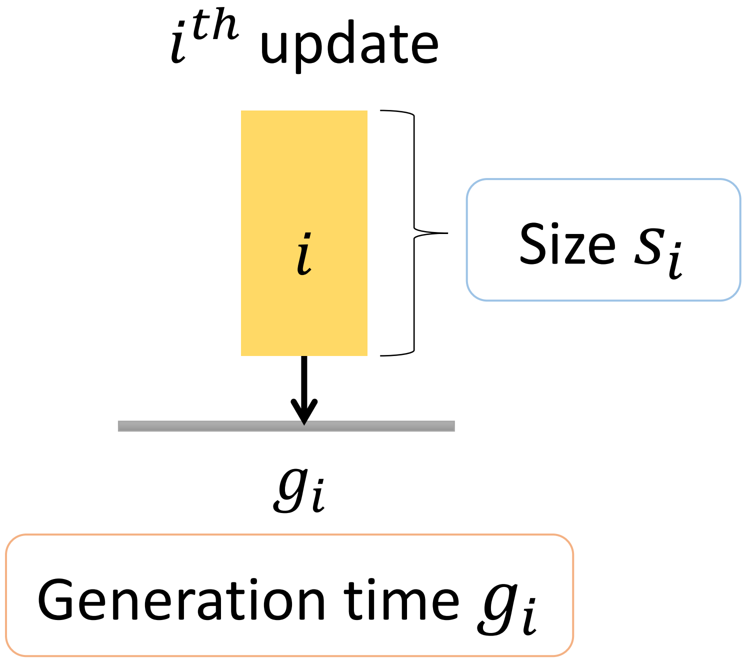
Notations:



Generation time g_i

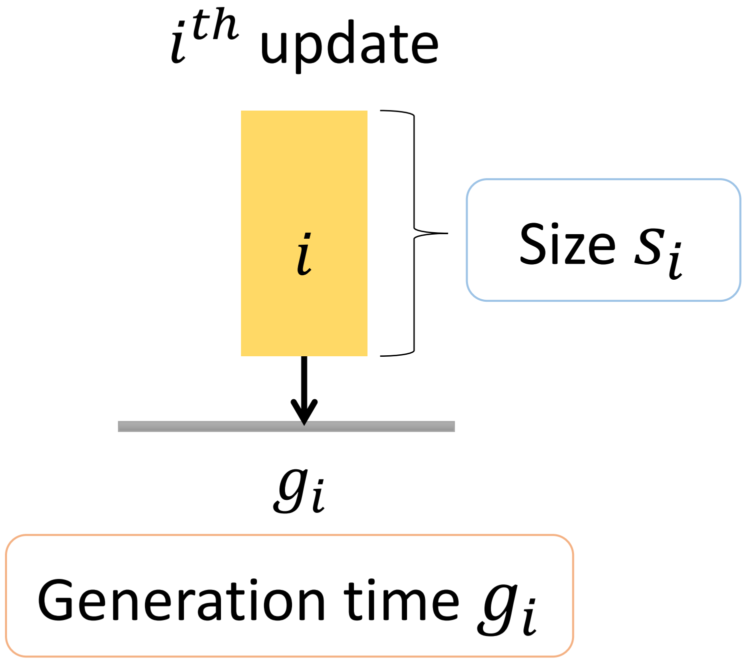
Which update to transmit?

Notations:



Which update to transmit?

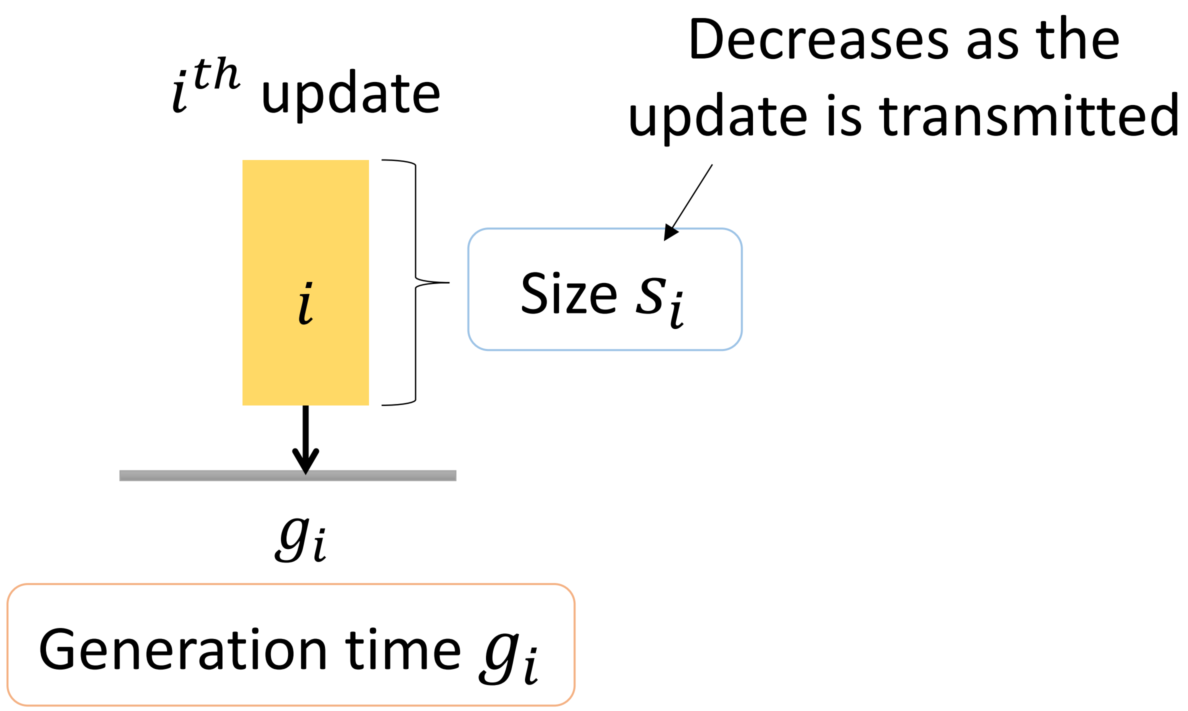
Notations:



Constant!!

Which update to transmit?

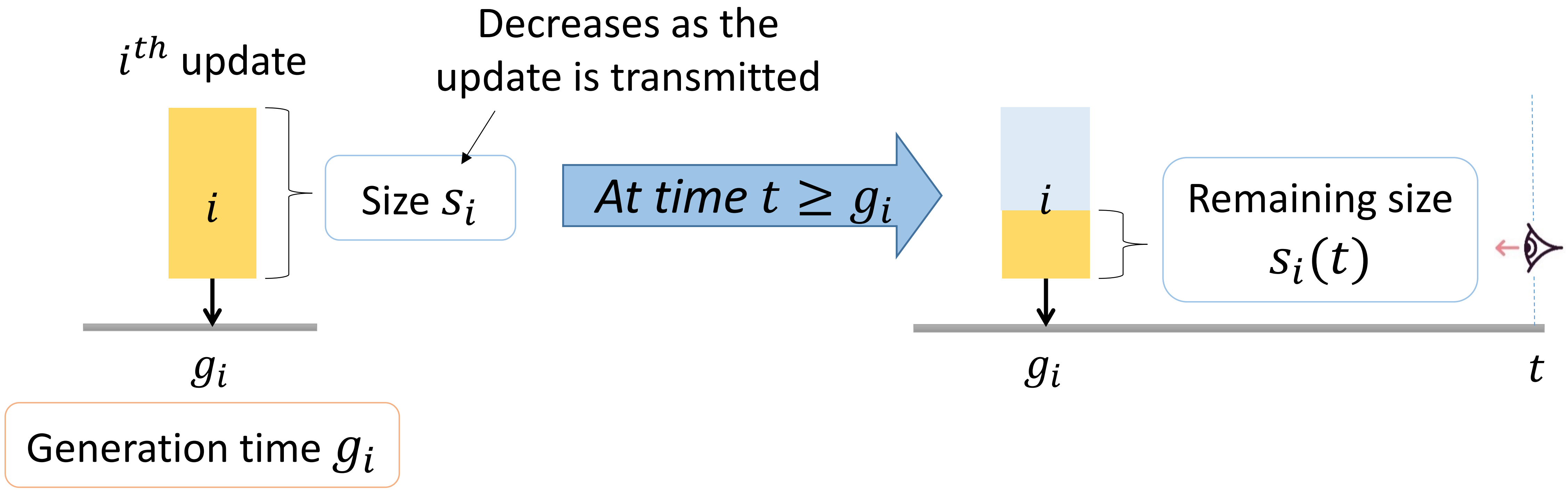
Notations:



Constant!!

Which update to transmit?

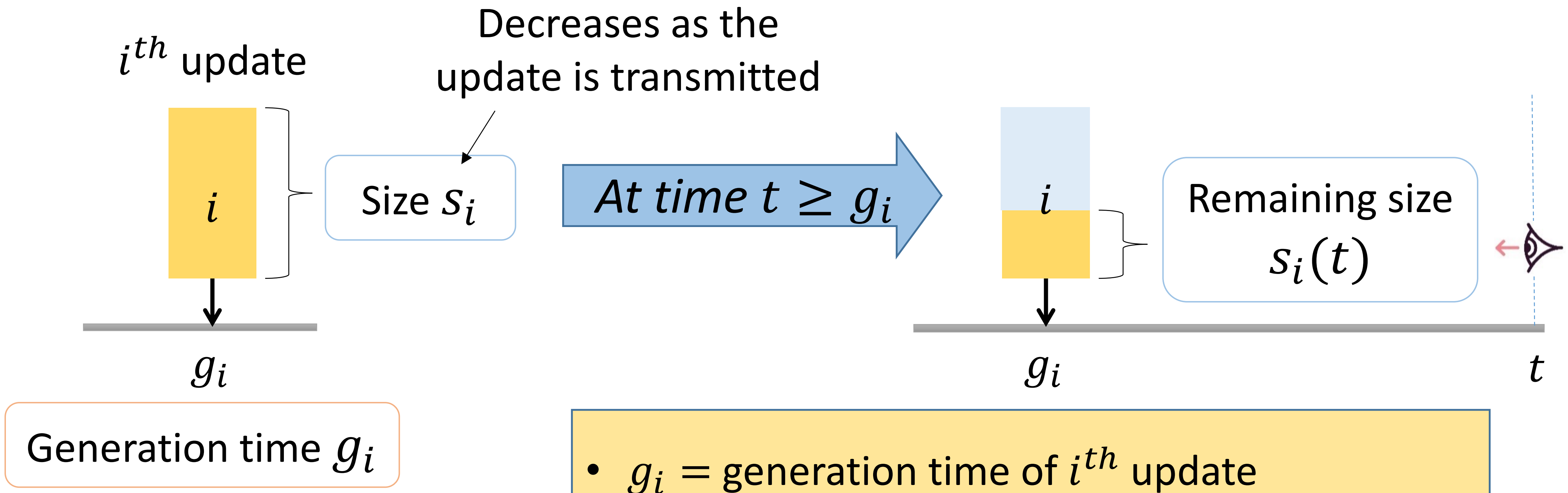
Notations:



Constant!!

Which update to transmit?

Notations:

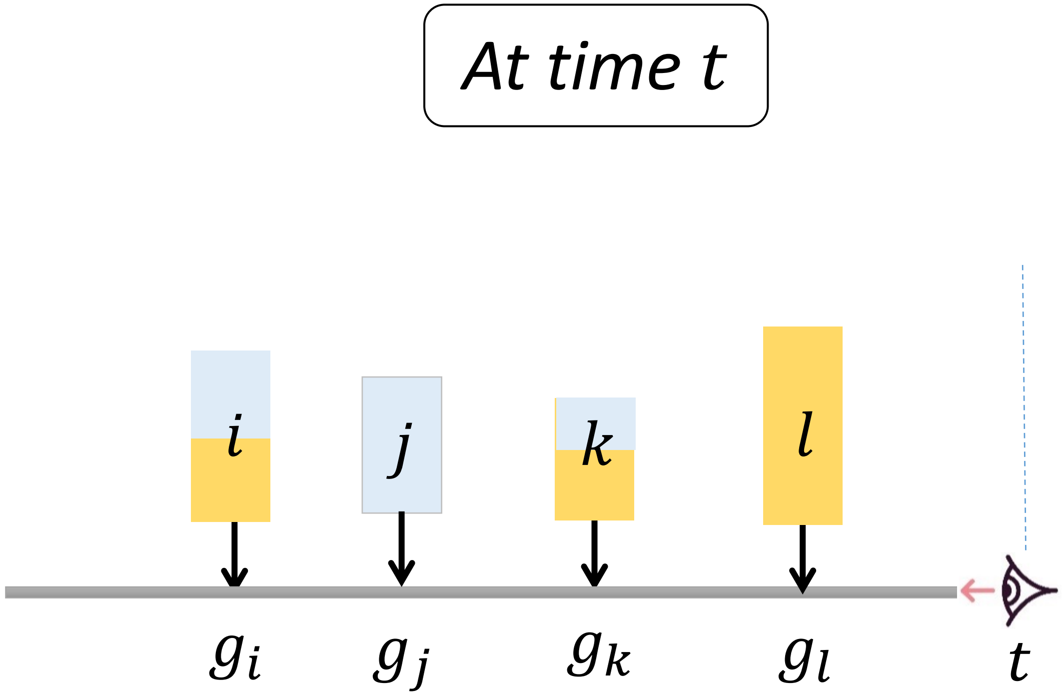


Constant!!

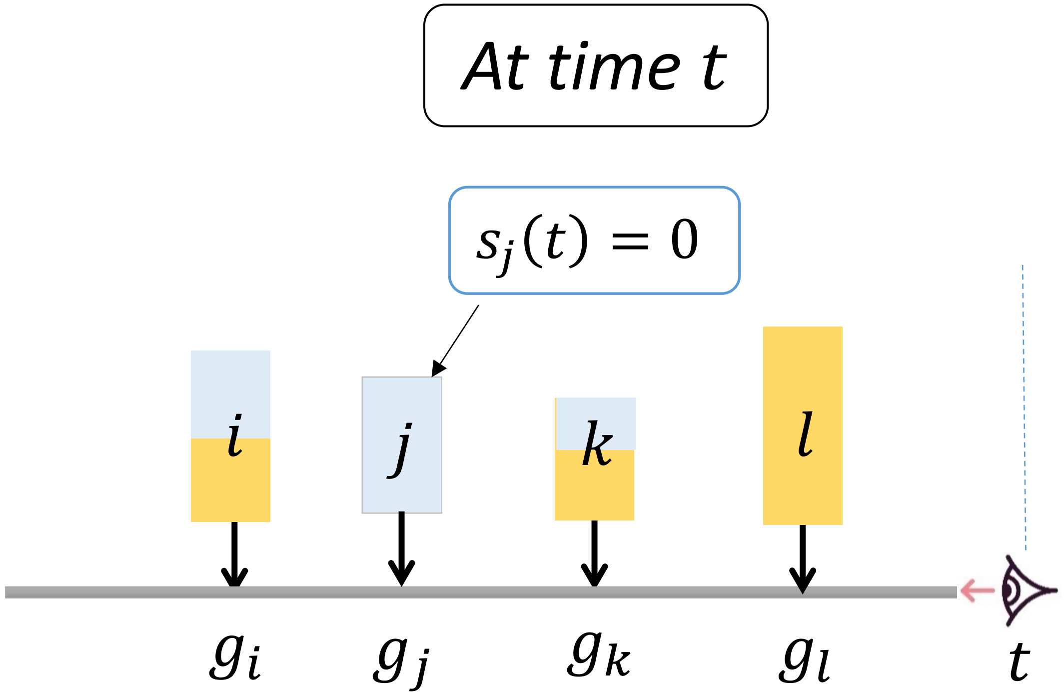
- g_i = generation time of i^{th} update
- $s_i(t)$ = remaining size of i^{th} update at time t

Which update to transmit?

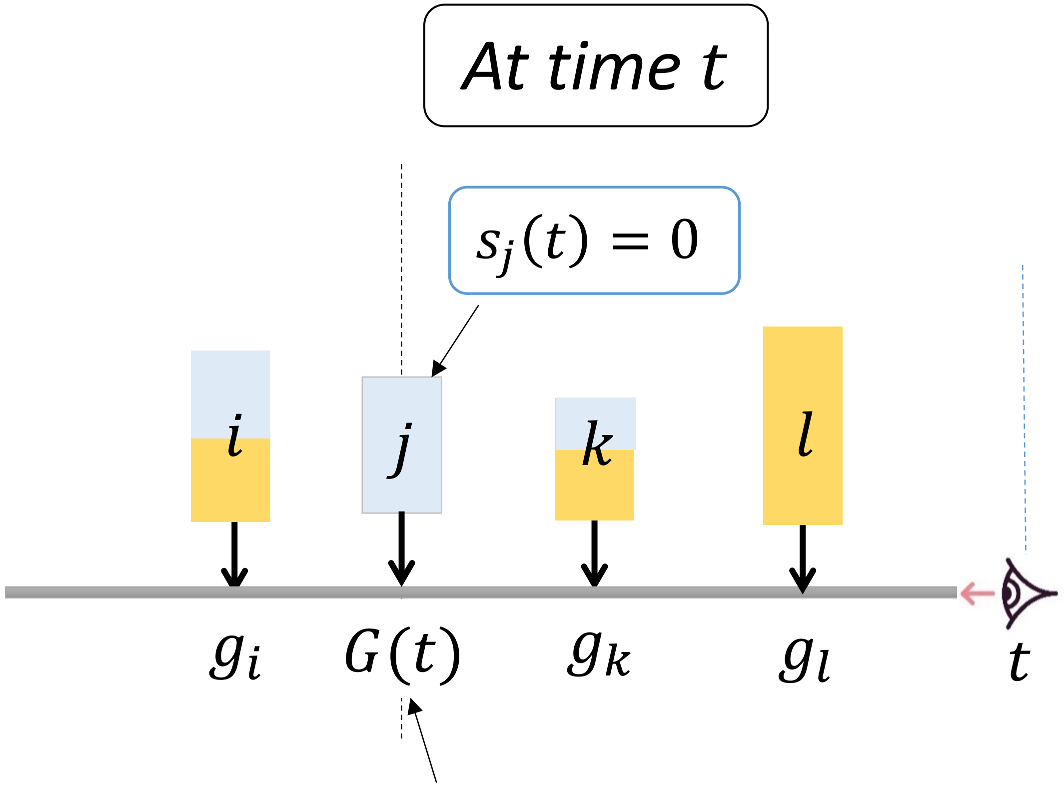
Which update to transmit?



Which update to transmit?



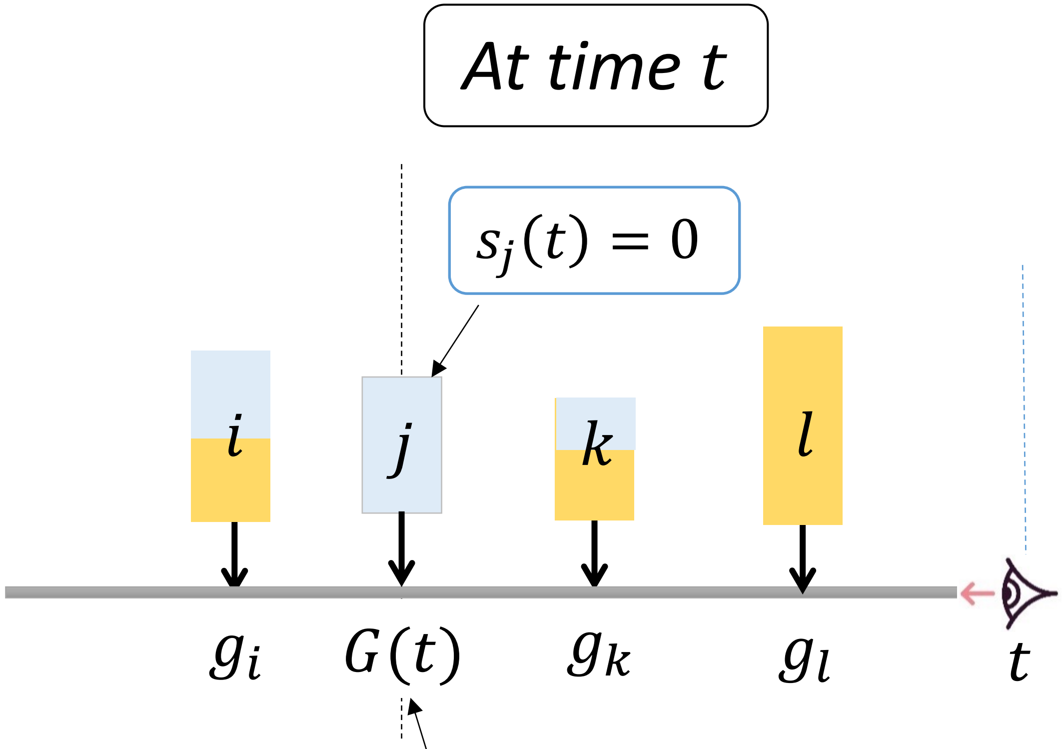
Which update to transmit?



generation time of latest update
completely transmitted until time t

(Remaining size at time t is 0)

Which update to transmit?

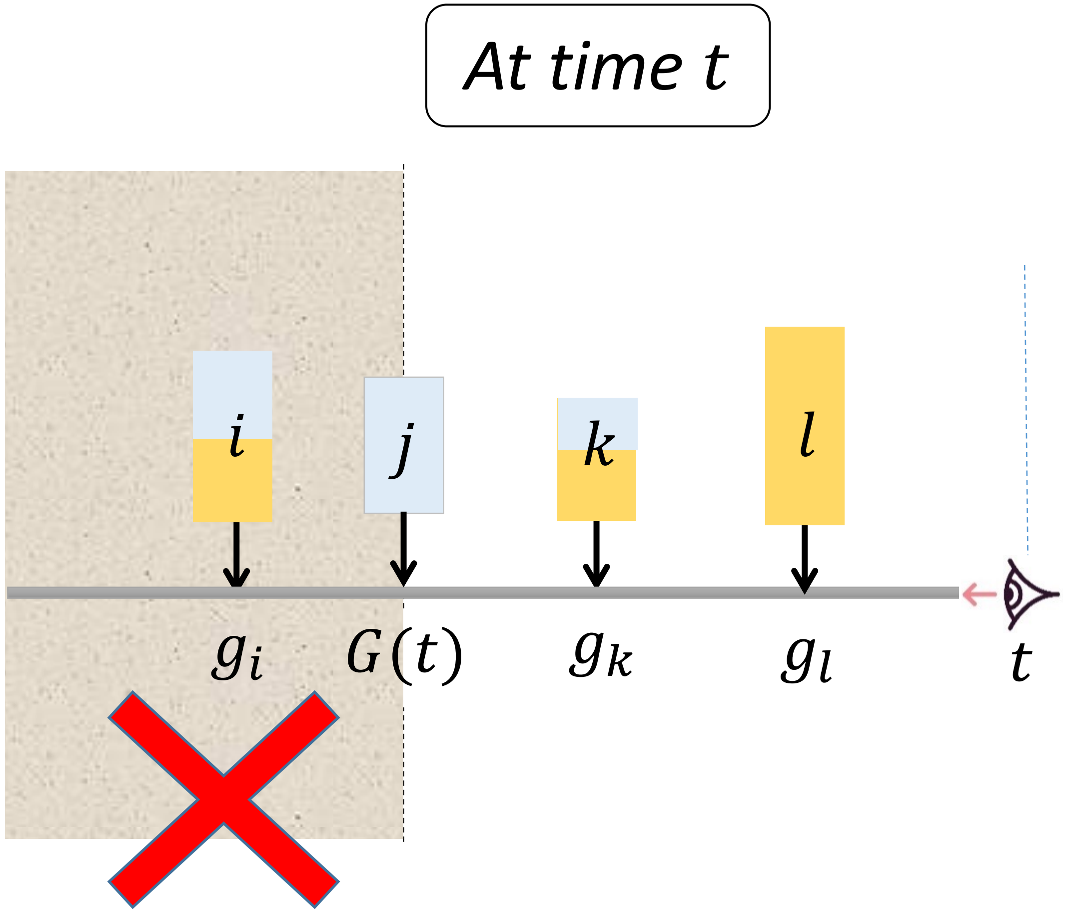


generation time of latest update
completely transmitted until time t

(Remaining size at time t is 0)

$$AoI(t) = t - G(t)$$

Which update to transmit?

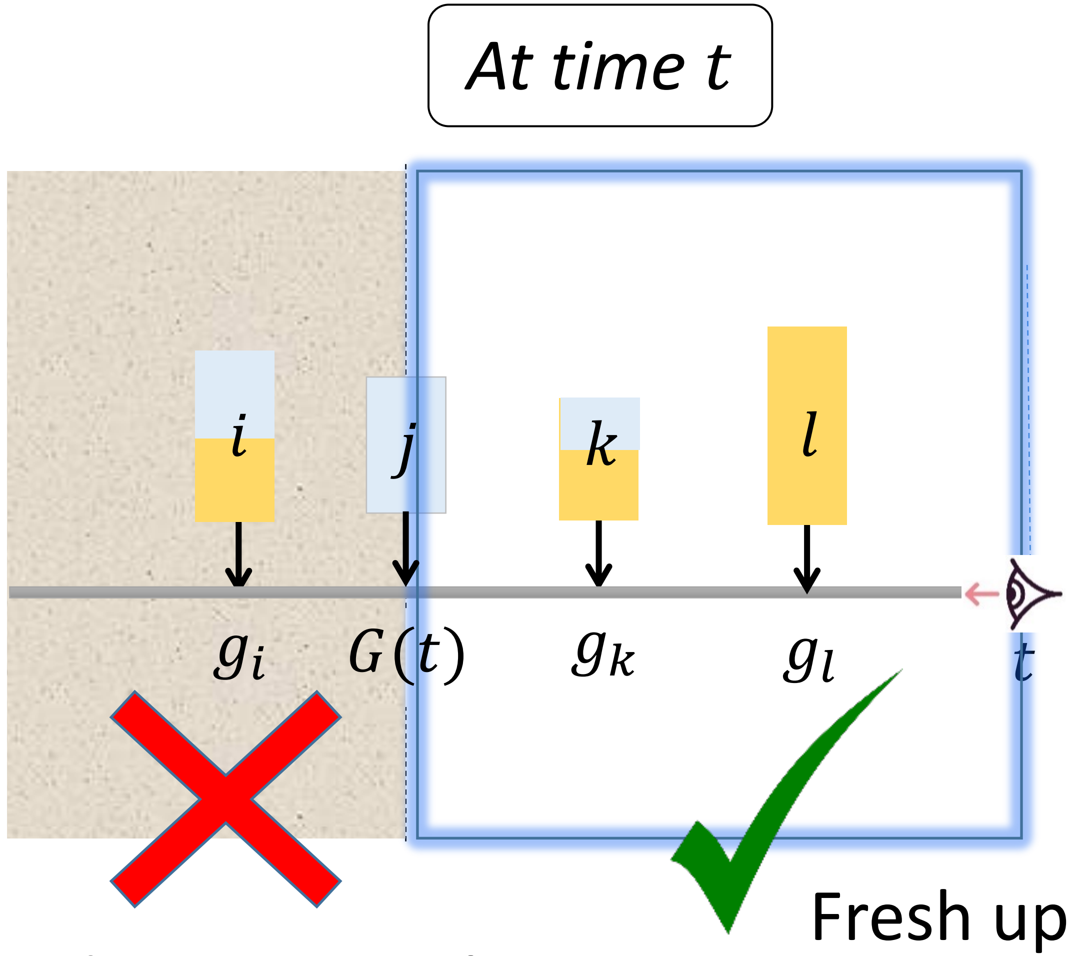


$$AoI(t) = t - G(t)$$

Old updates are useless

Which update to transmit?

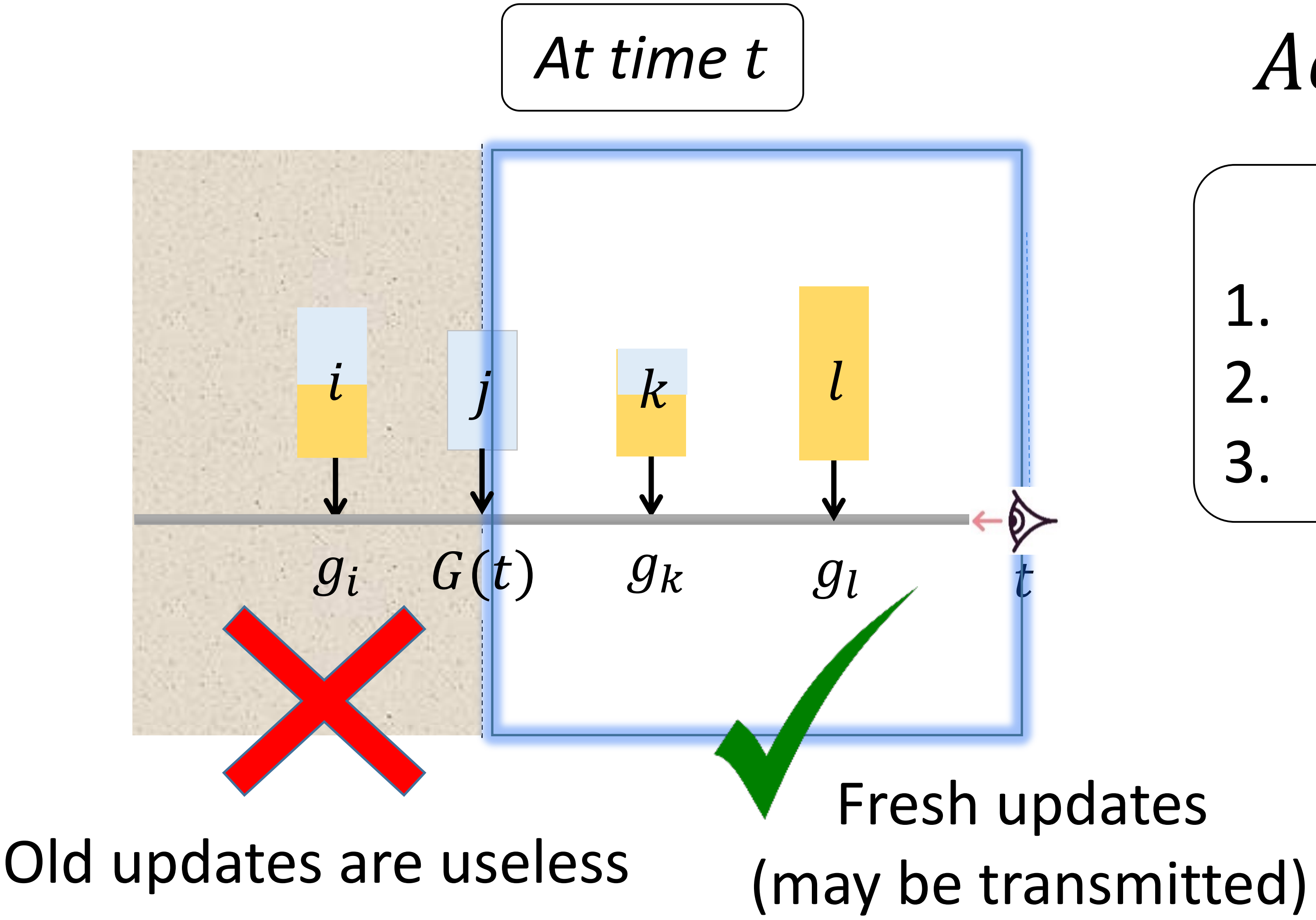
$$AoI(t) = t - G(t)$$



Old updates are useless

Fresh updates
(may be transmitted)

Which update to transmit?

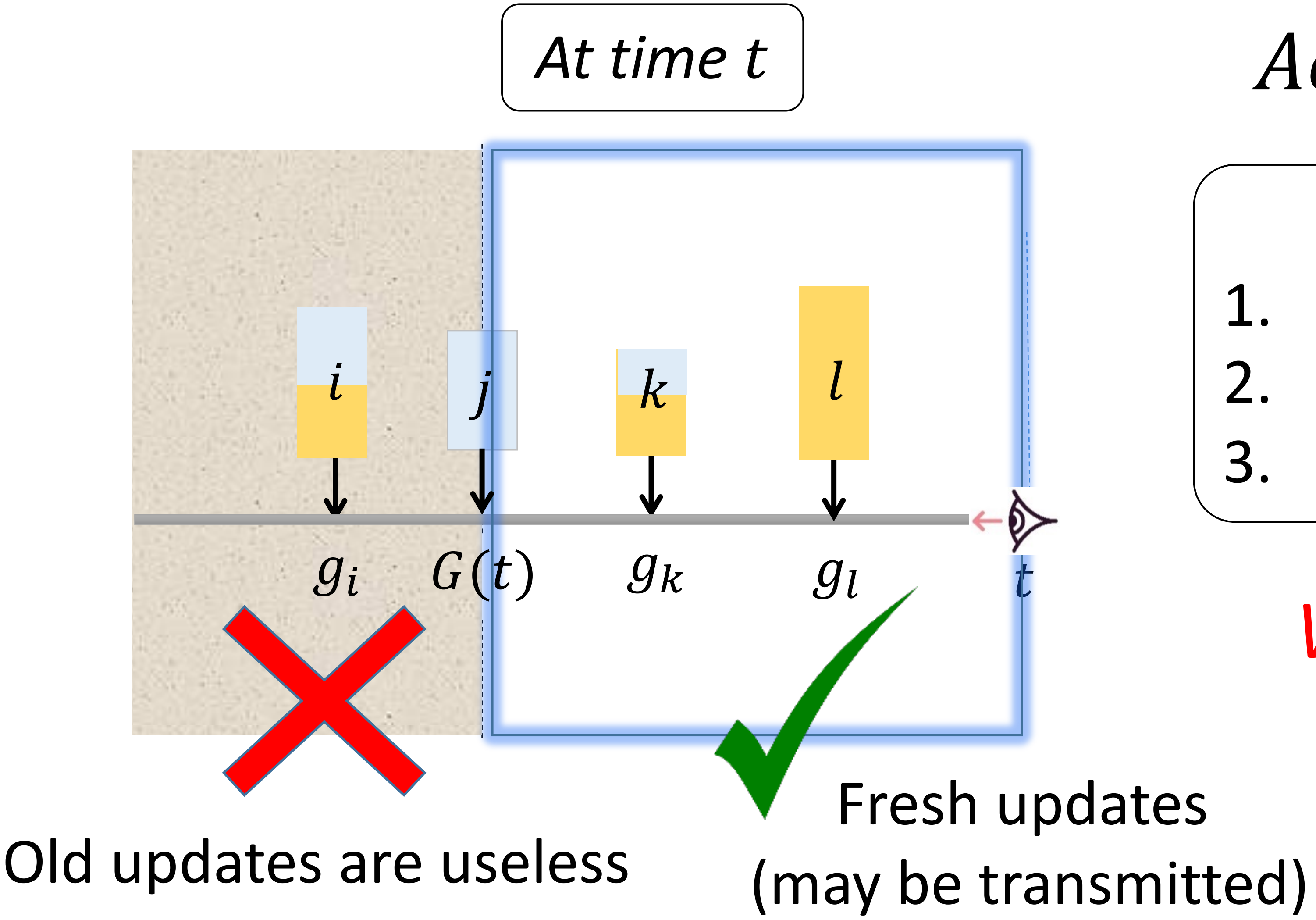


$$AoI(t) = t - G(t)$$

Choices (at time t):

1. Transmit update k
2. Transmit update l
3. Wait (do not transmit)

Which update to transmit?



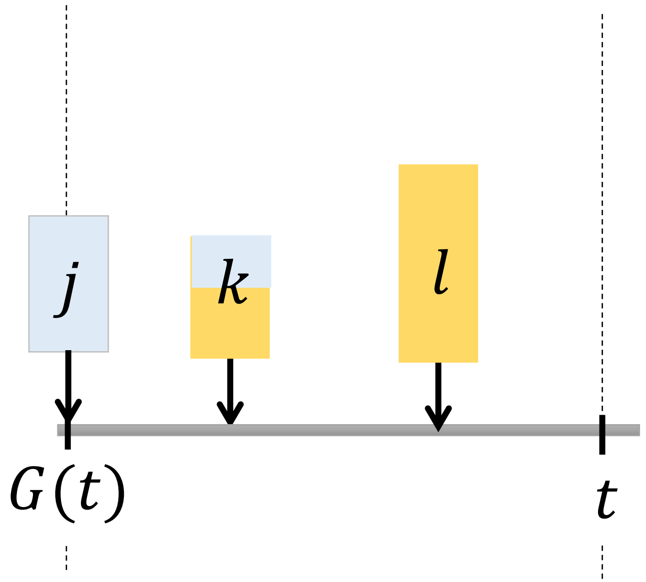
$$AoI(t) = t - G(t)$$

<u>*Choices (at time t):*</u>

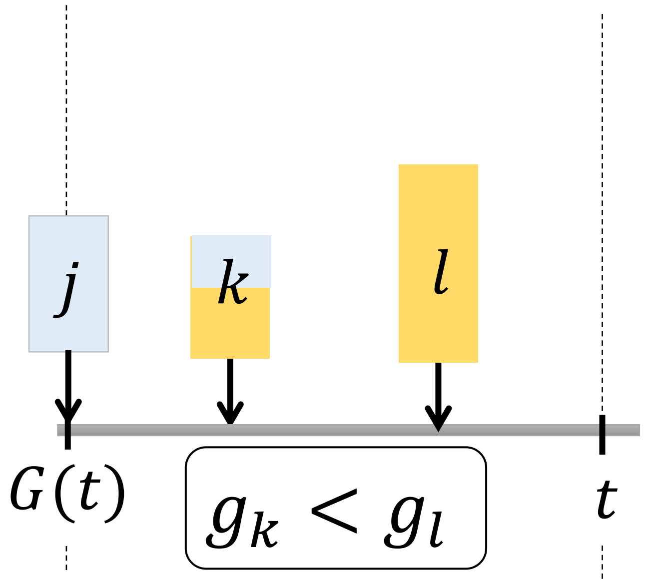
1. Transmit update k
2. Transmit update l
3. Wait (do not transmit)

Which is optimal?

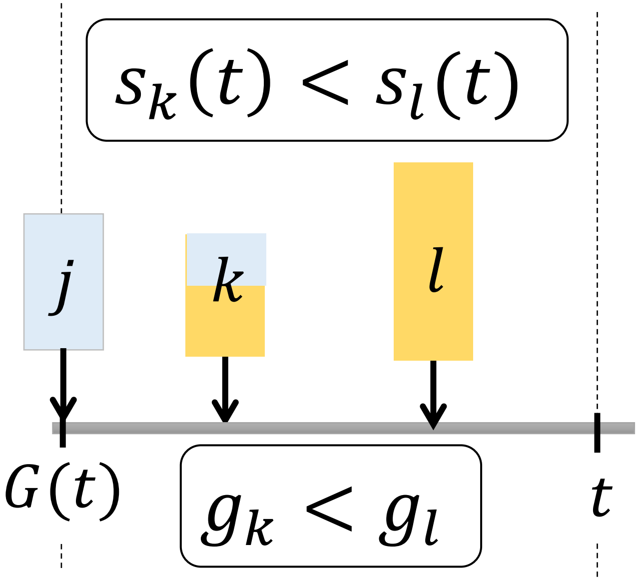
Which update to transmit?



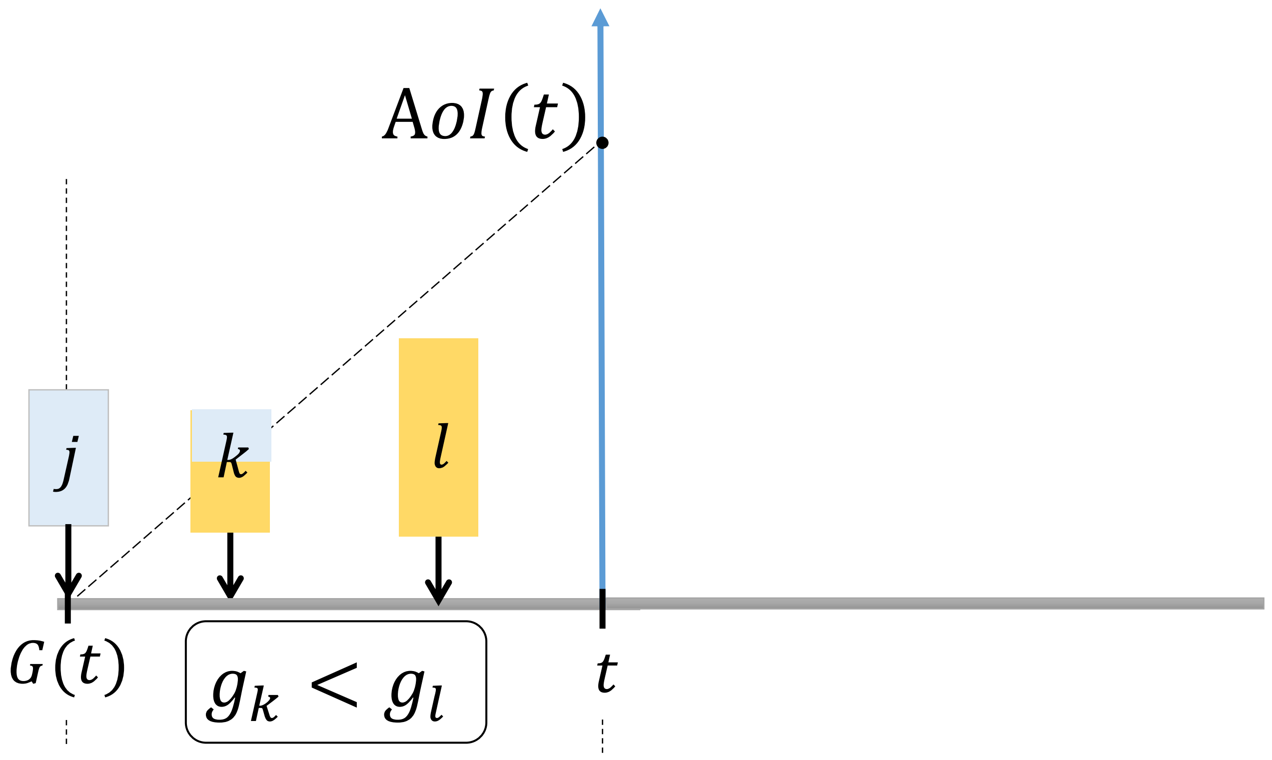
Which update to transmit?



Which update to transmit?



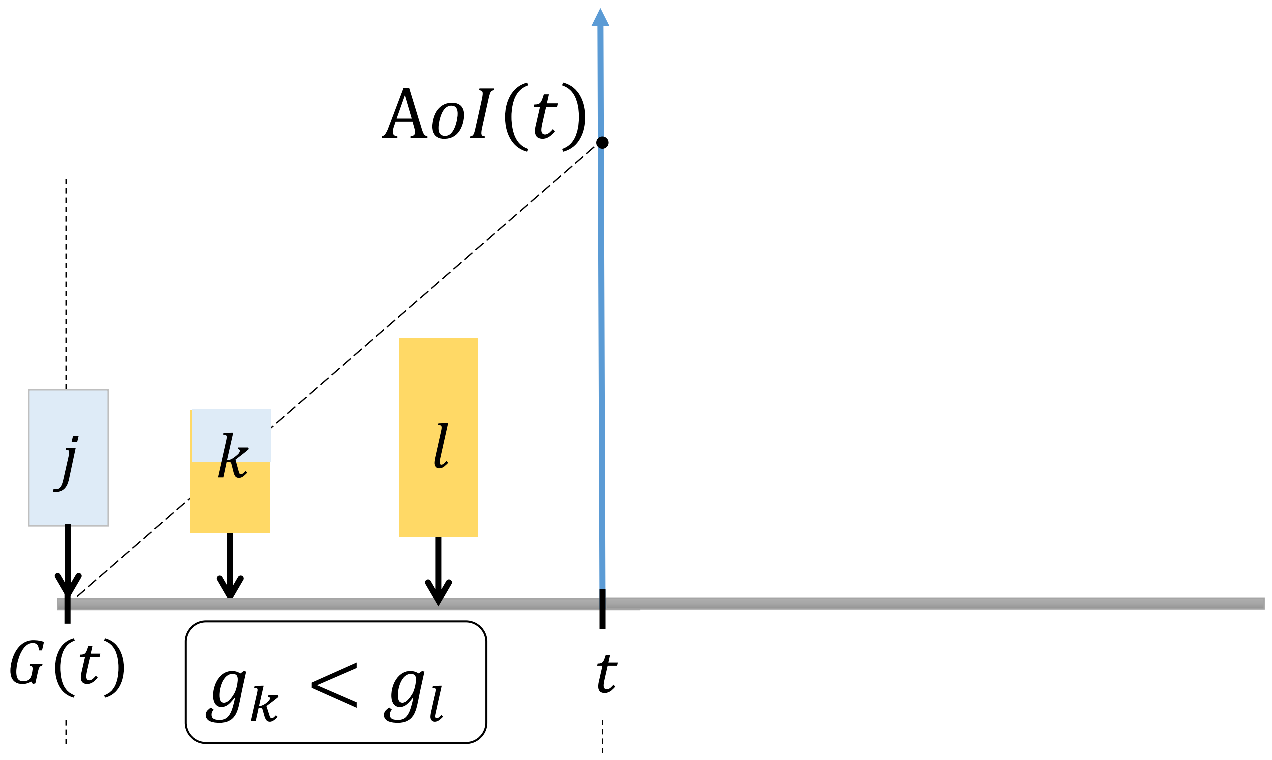
Which update to transmit?



Which update to transmit?

$c = 0$

(Better to transmit than not)

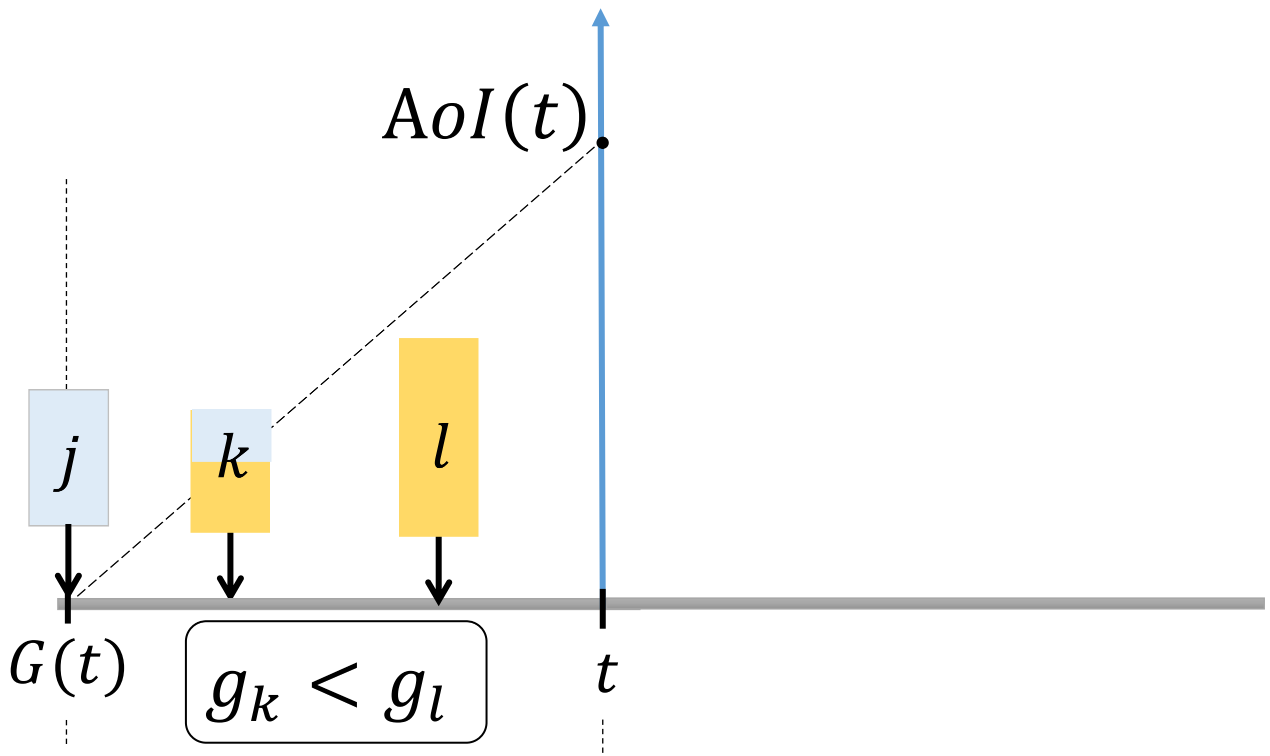


Which update to transmit?

(at time t)

$c = 0$

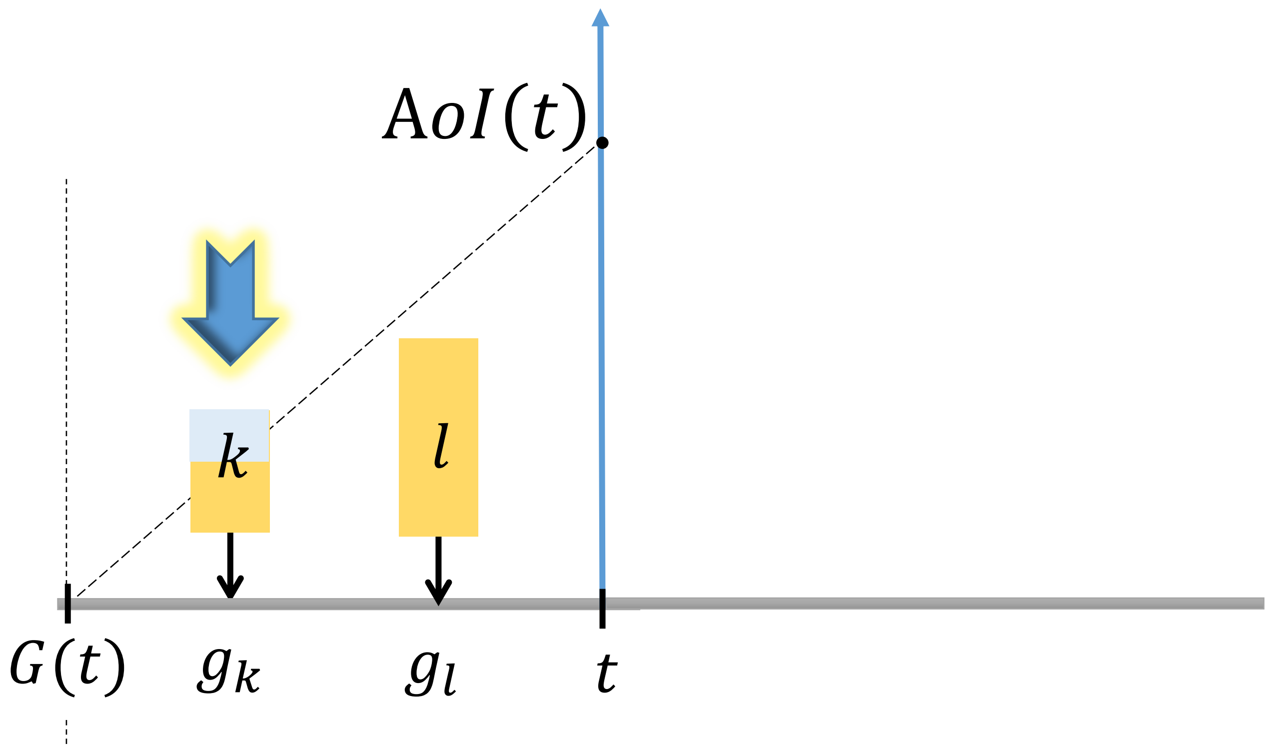
(Better to transmit than not)



Which update to transmit?

$c = 0$

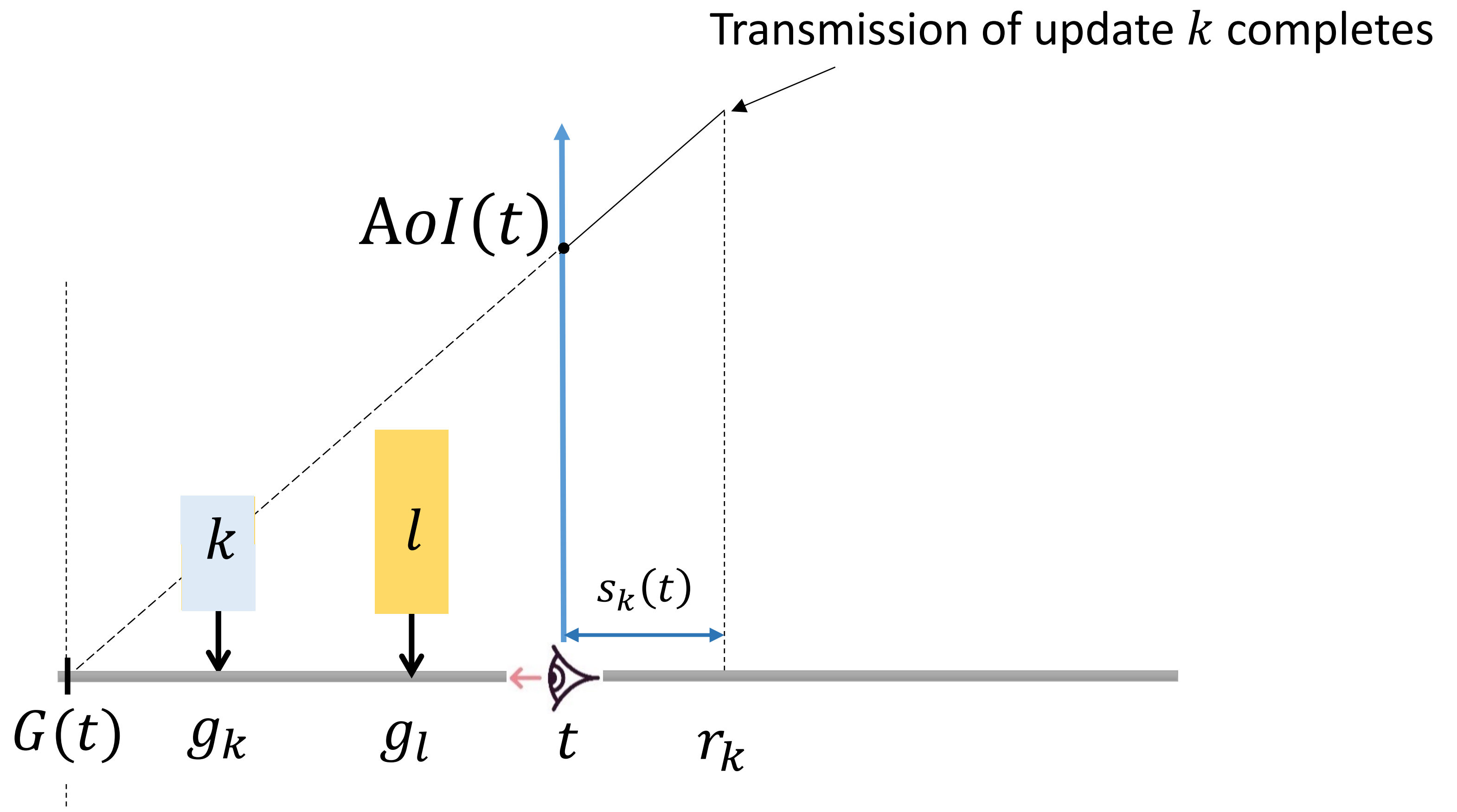
Choice 1: Transmit update k from time t



Which update to transmit?

$c = 0$

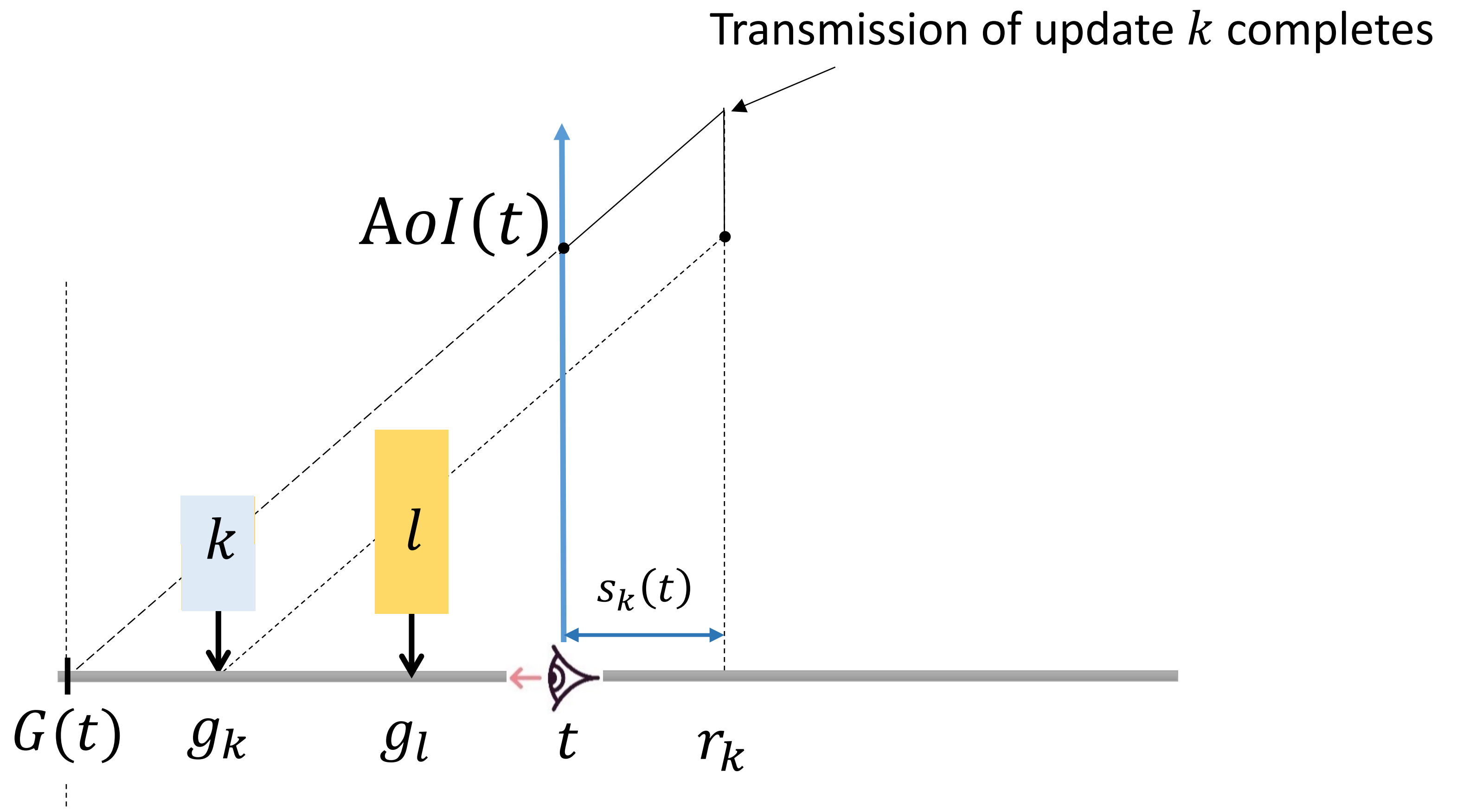
Choice 1: Transmit update k from time t



Which update to transmit?

$c = 0$

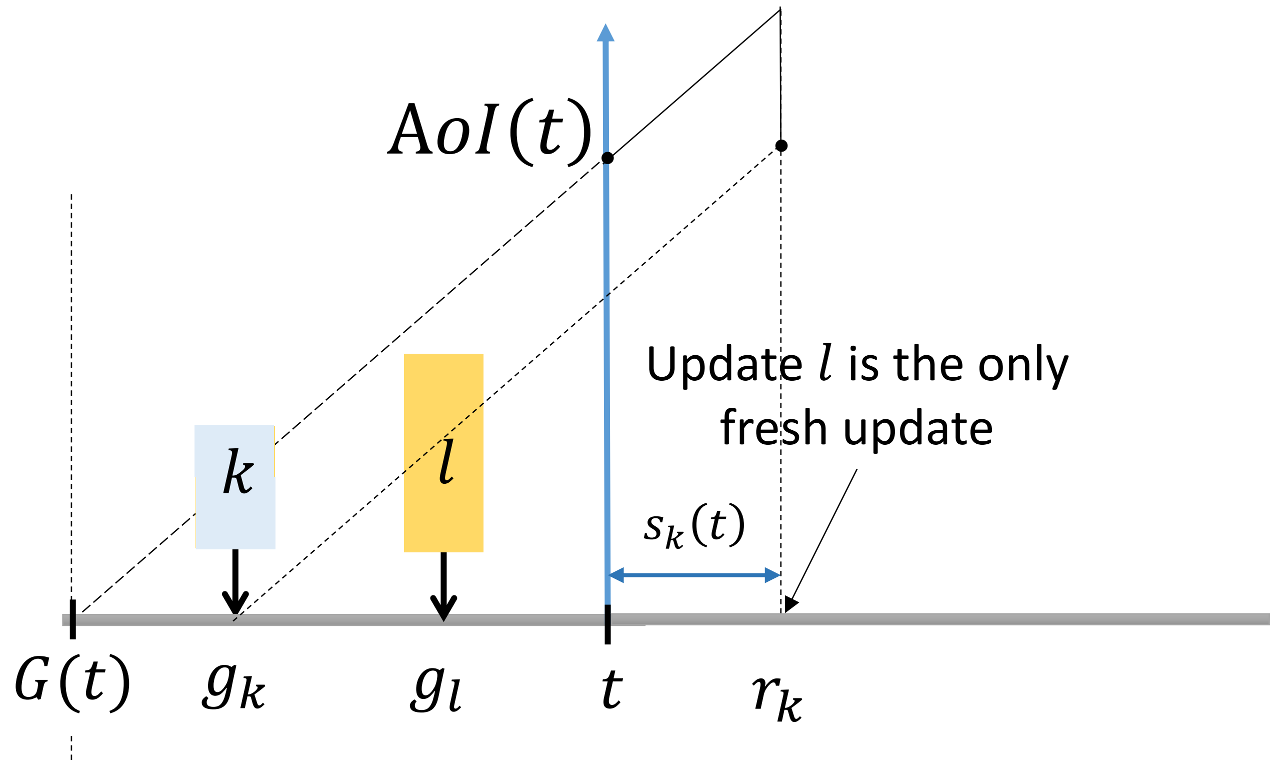
Choice 1: Transmit update k from time t



Which update to transmit?

$c = 0$

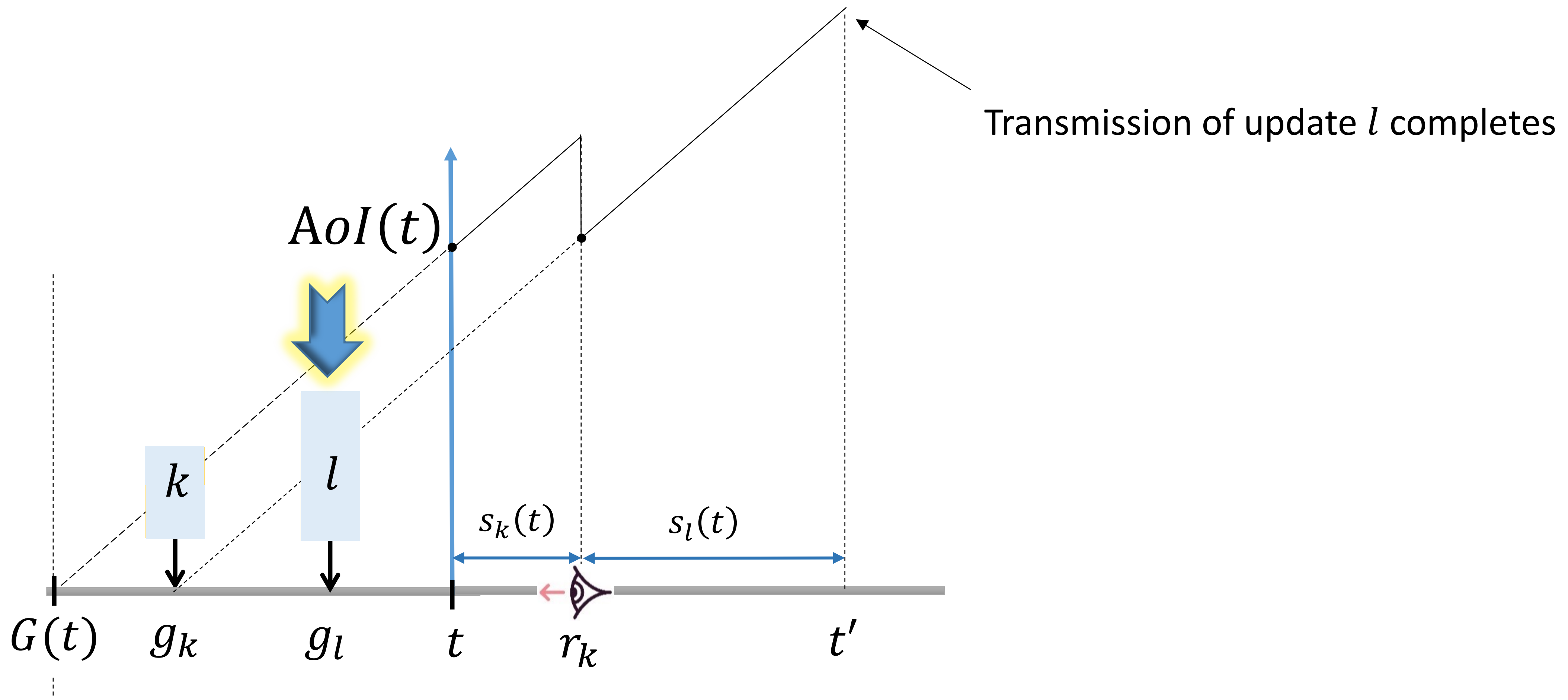
Choice 1: Transmit update k from time t



Which update to transmit?

$c = 0$

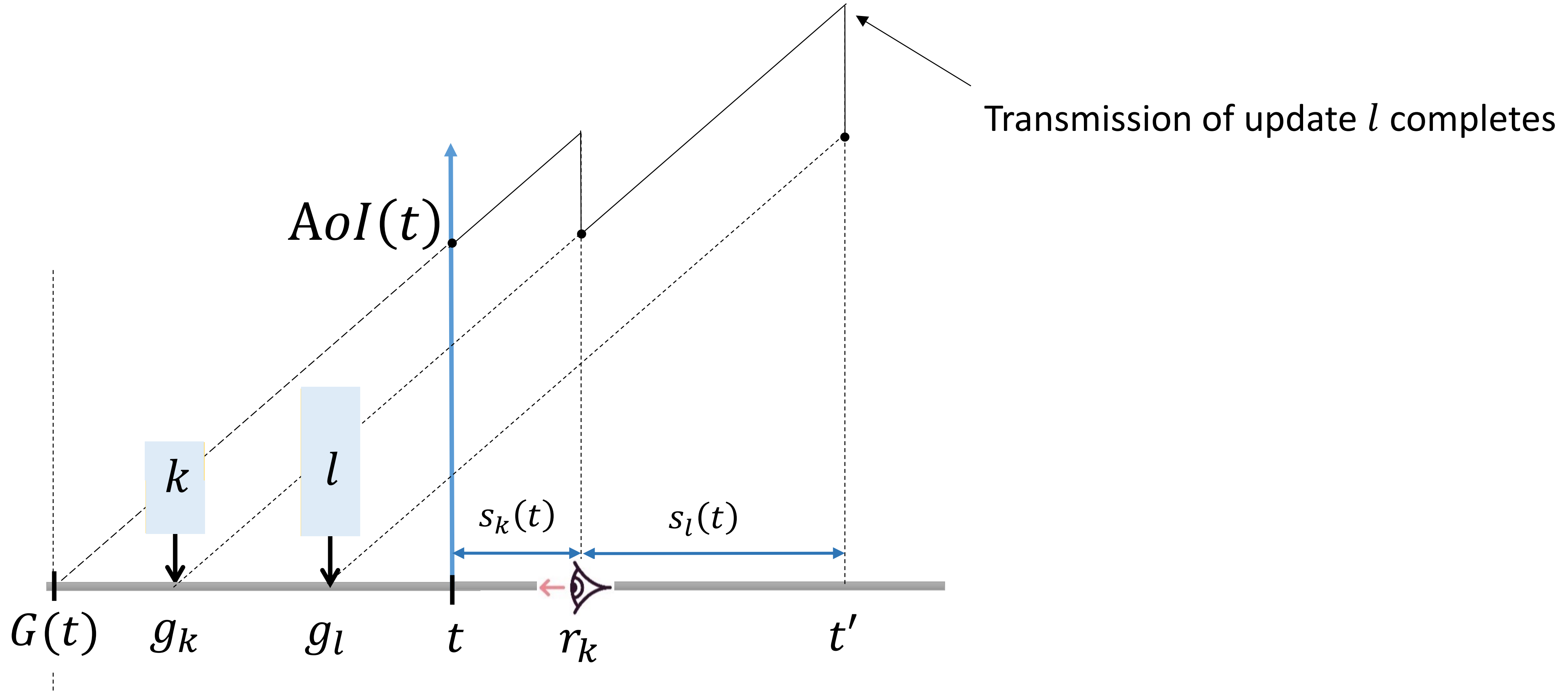
Choice 1: Transmit update k from time t



Which update to transmit?

$c = 0$

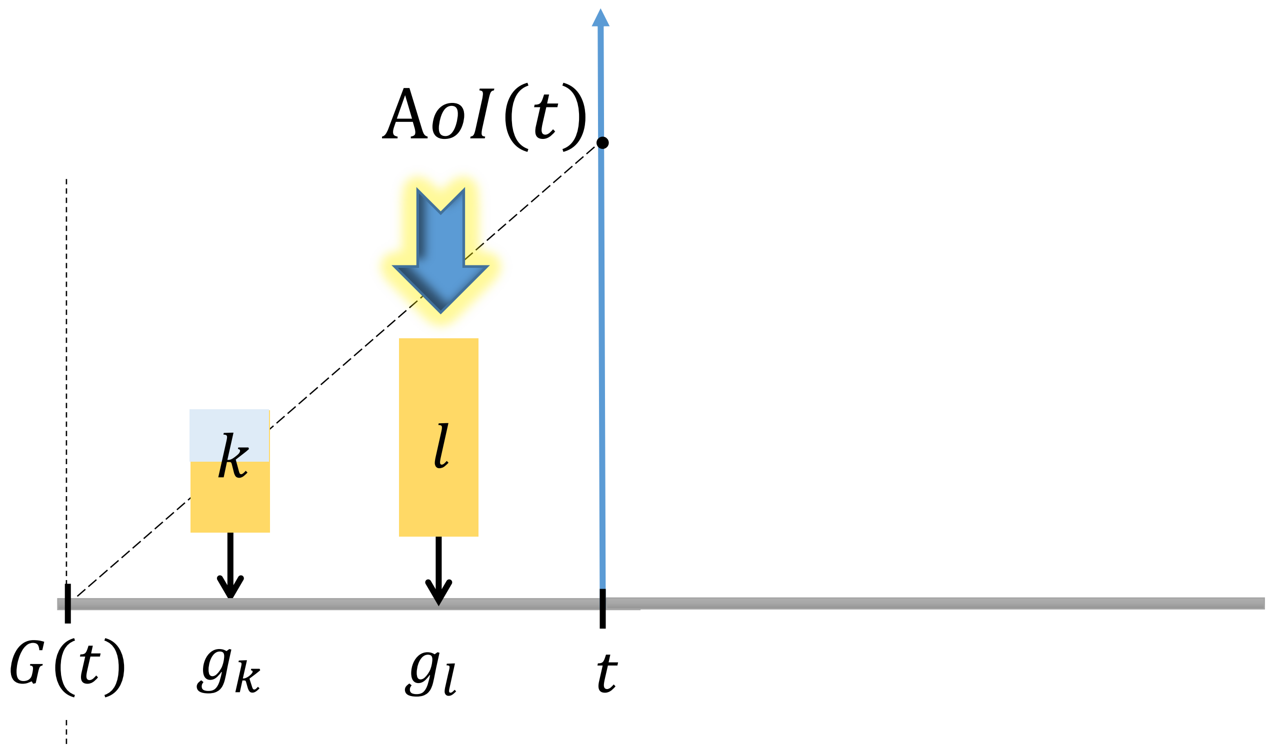
Choice 1: Transmit update k from time t



Which update to transmit?

$c = 0$

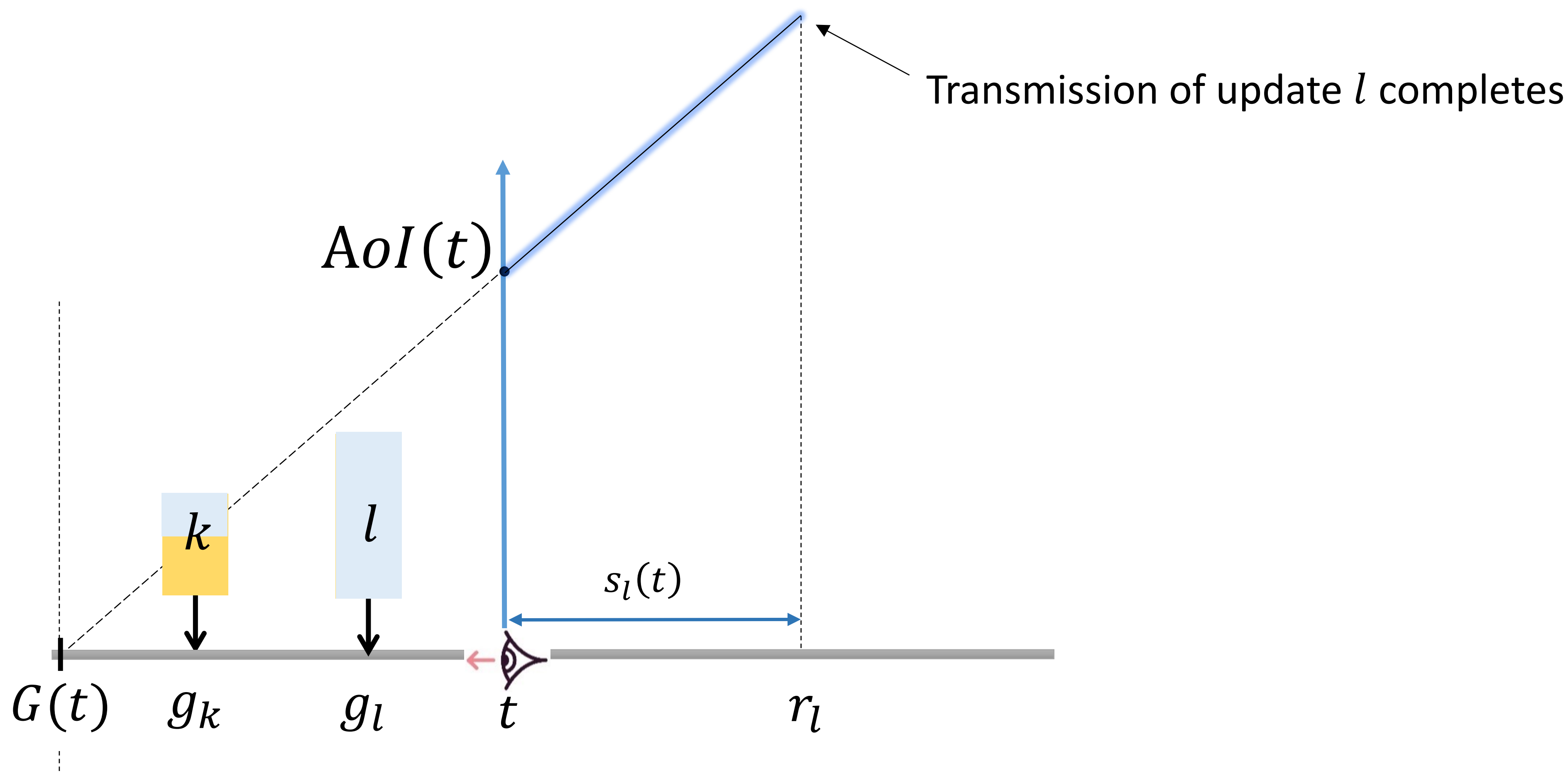
Choice 2: Transmit update l from time t



Which update to transmit?

$c = 0$

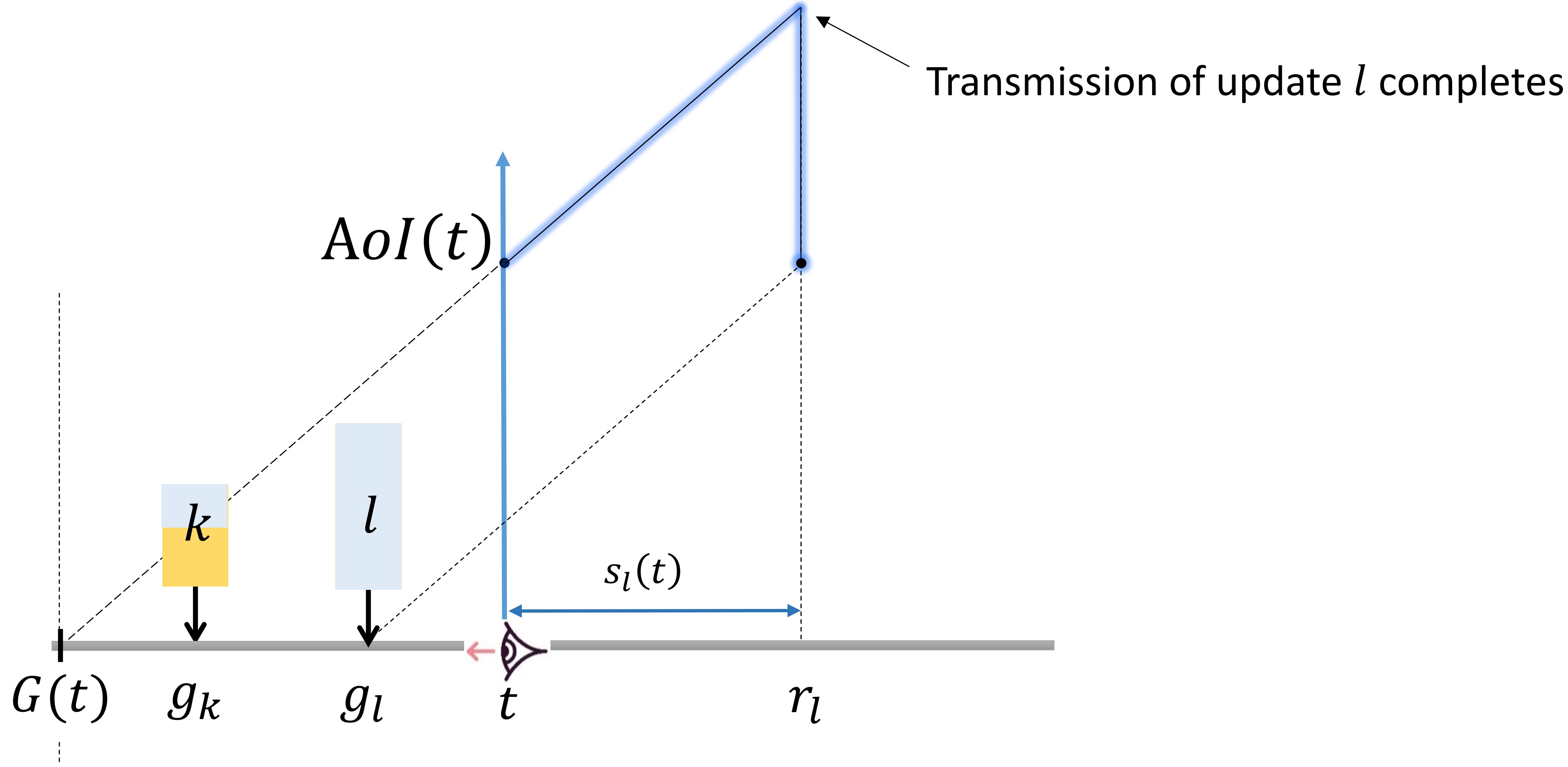
Choice 2: Transmit update l from time t



Which update to transmit?

$c = 0$

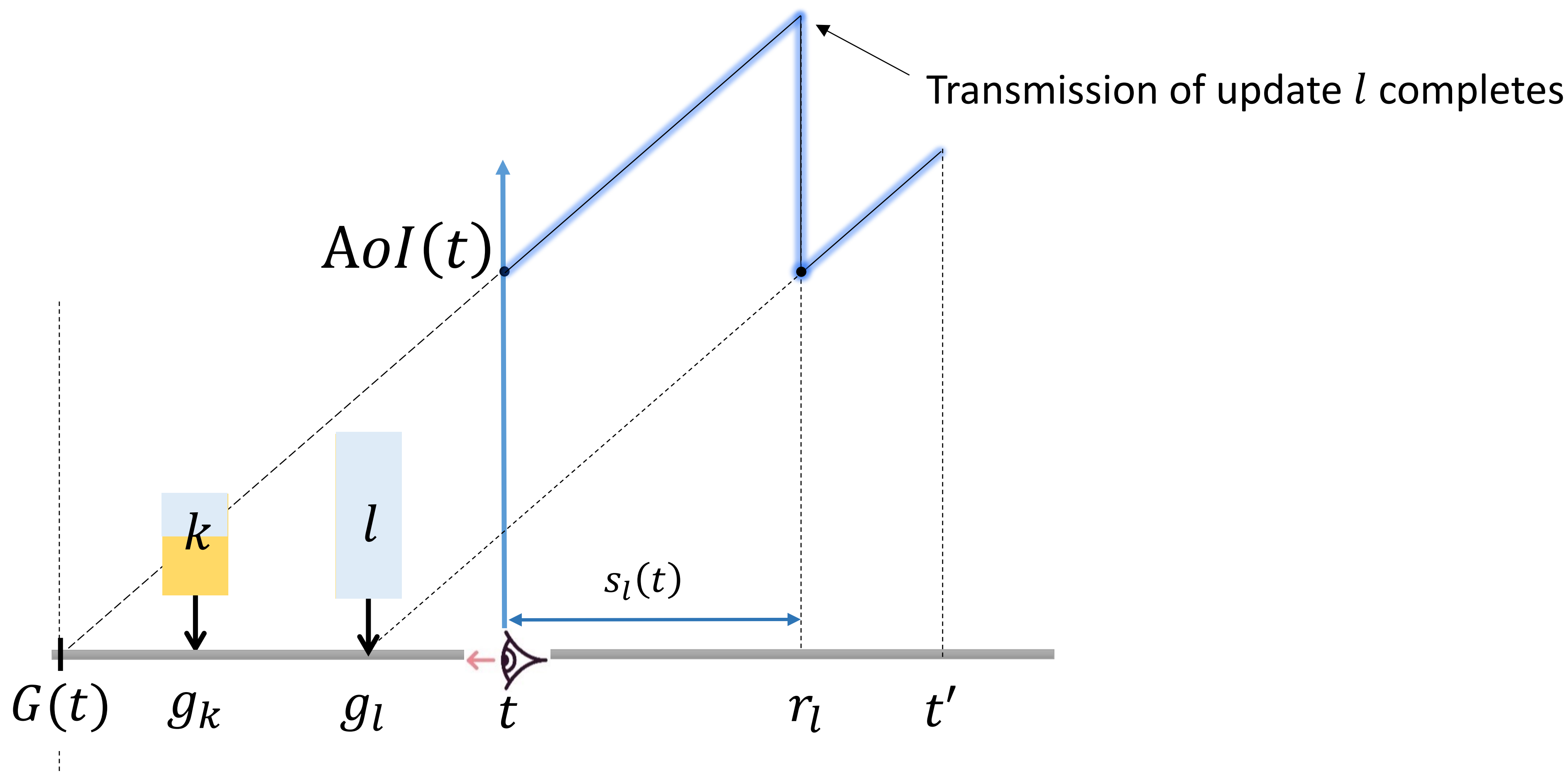
Choice 2: Transmit update l from time t



Which update to transmit?

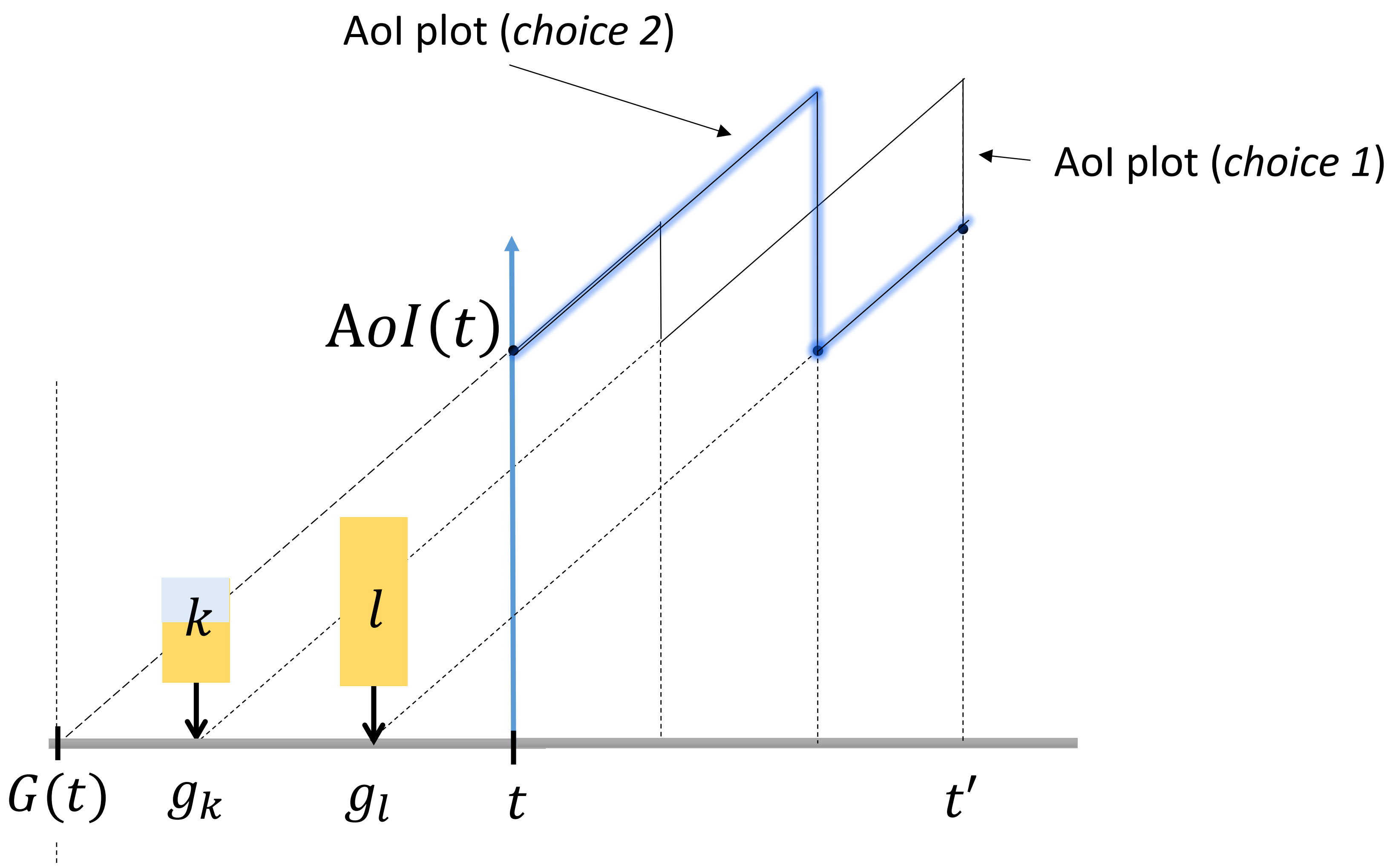
$c = 0$

Choice 2: Transmit update l from time t



Which update to transmit?

$c = 0$



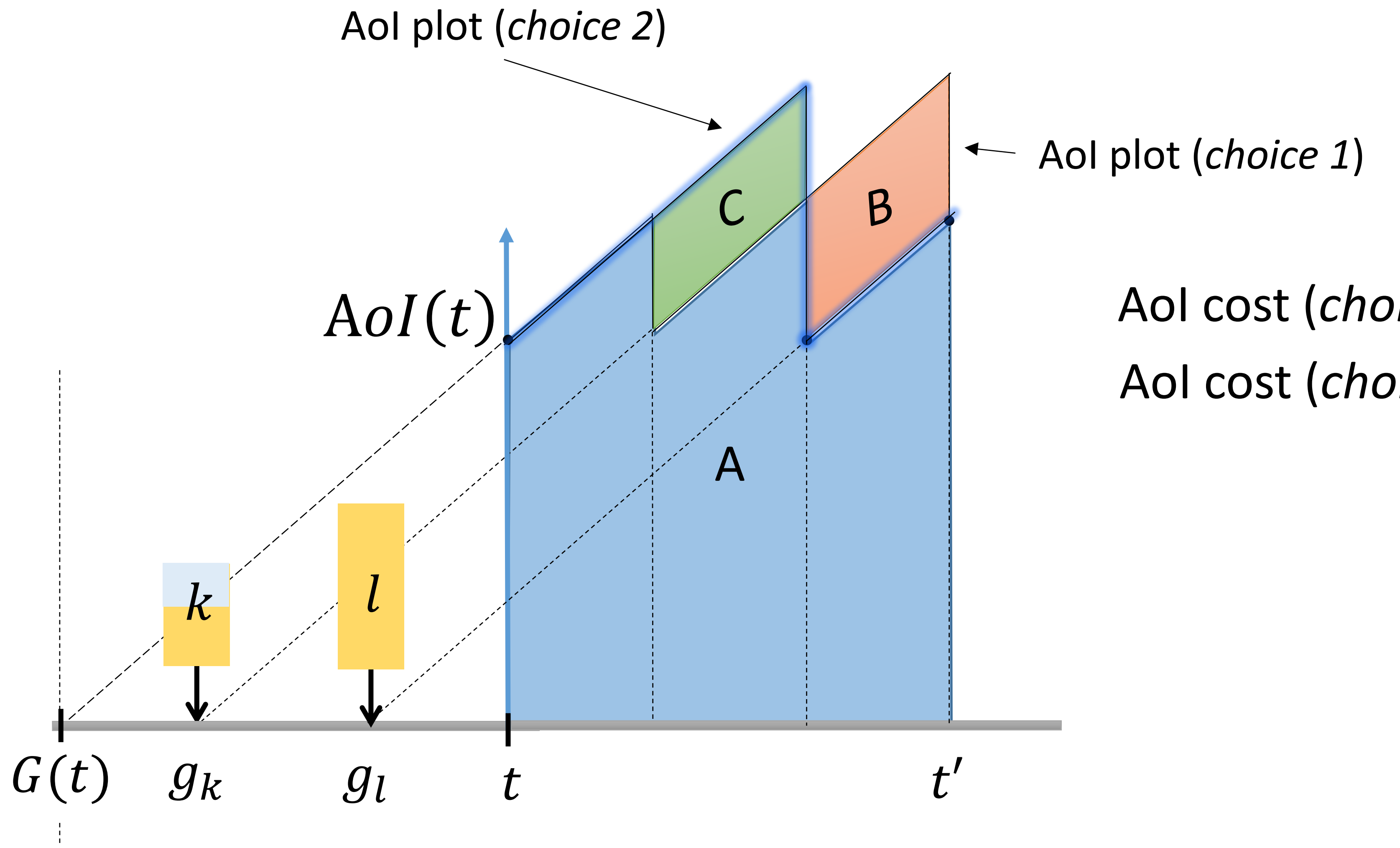
Choices (at time t):

1. Transmit update k
2. Transmit update l

Which update to transmit?

$c = 0$

Choices (at time t):
1. Transmit update k
2. Transmit update l

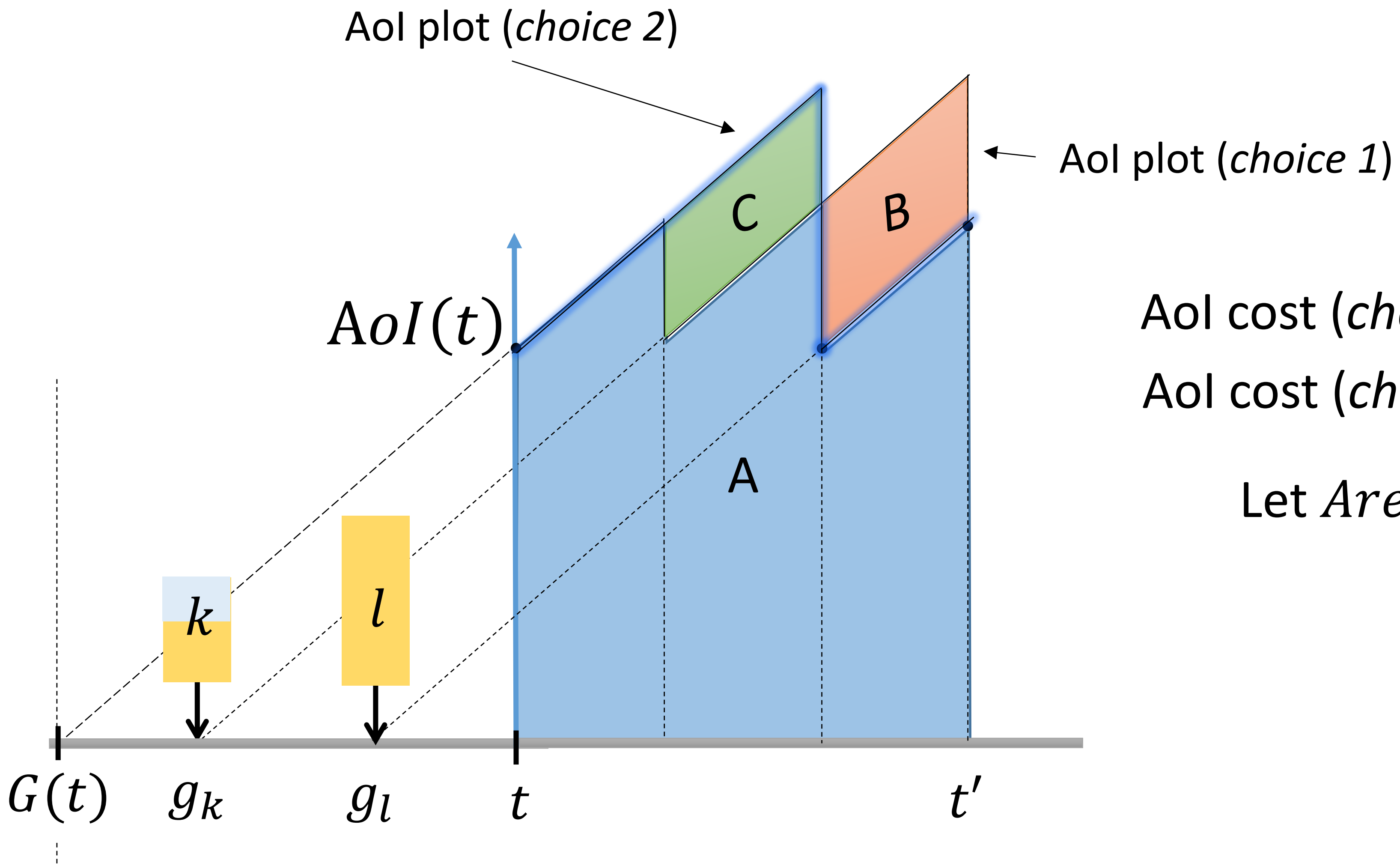


AoI cost (*choice 1*)$= Area(A + B)$

AoI cost (*choice 2*)$= Area(A + C)$

Which update to transmit?

$c = 0$



Choices (at time t):

1. Transmit update k
2. Transmit update l

AoI cost (*choice 1*)= $Area(A + B)$

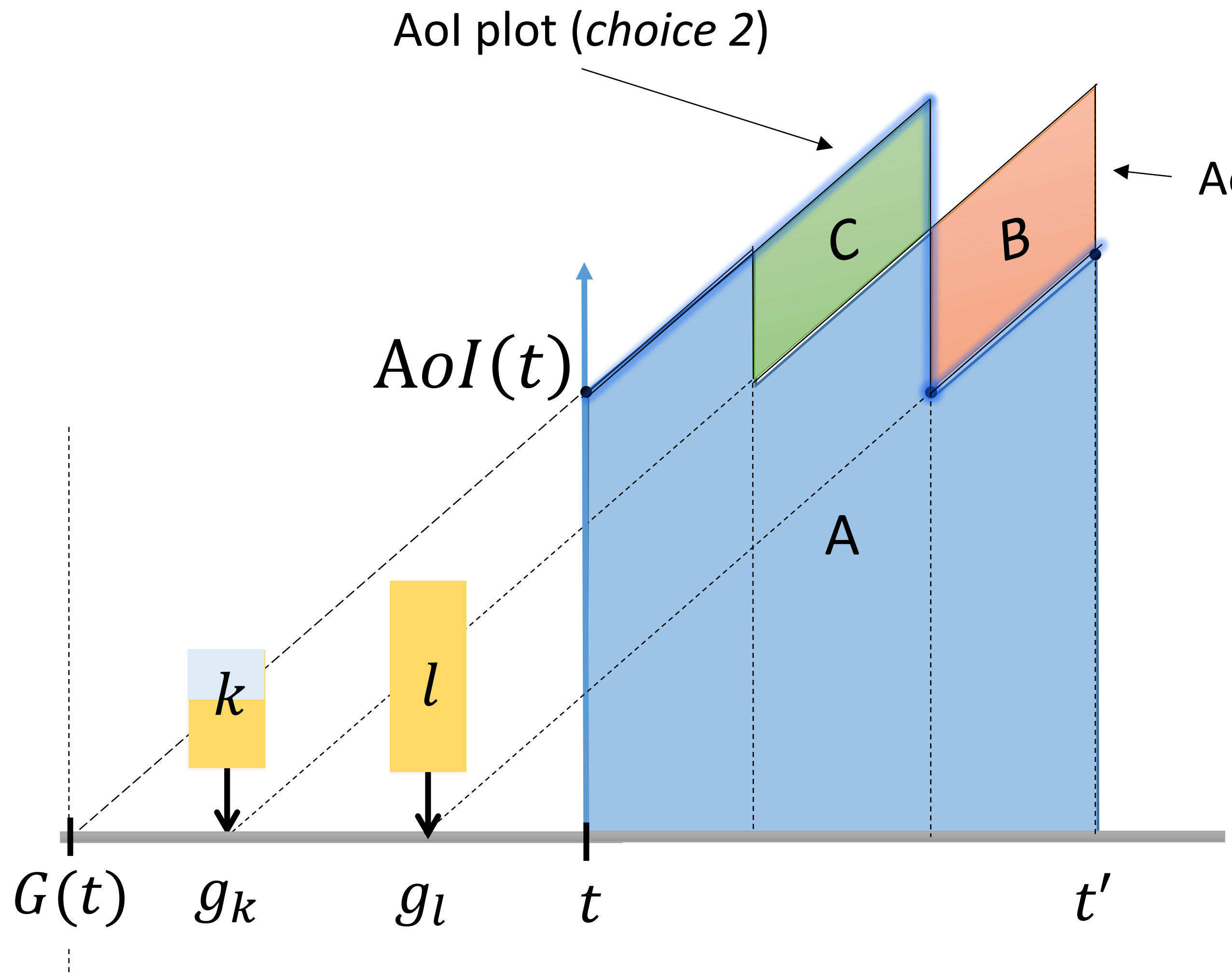
AoI cost (*choice 2*)= $Area(A + C)$

Let $Area(B) > Area(C)$:

Which update to transmit?

$c = 0$

Choices (at time t):
1. Transmit update k
2. Transmit update l



AoI cost (*choice 1*)= $Area(A + B)$

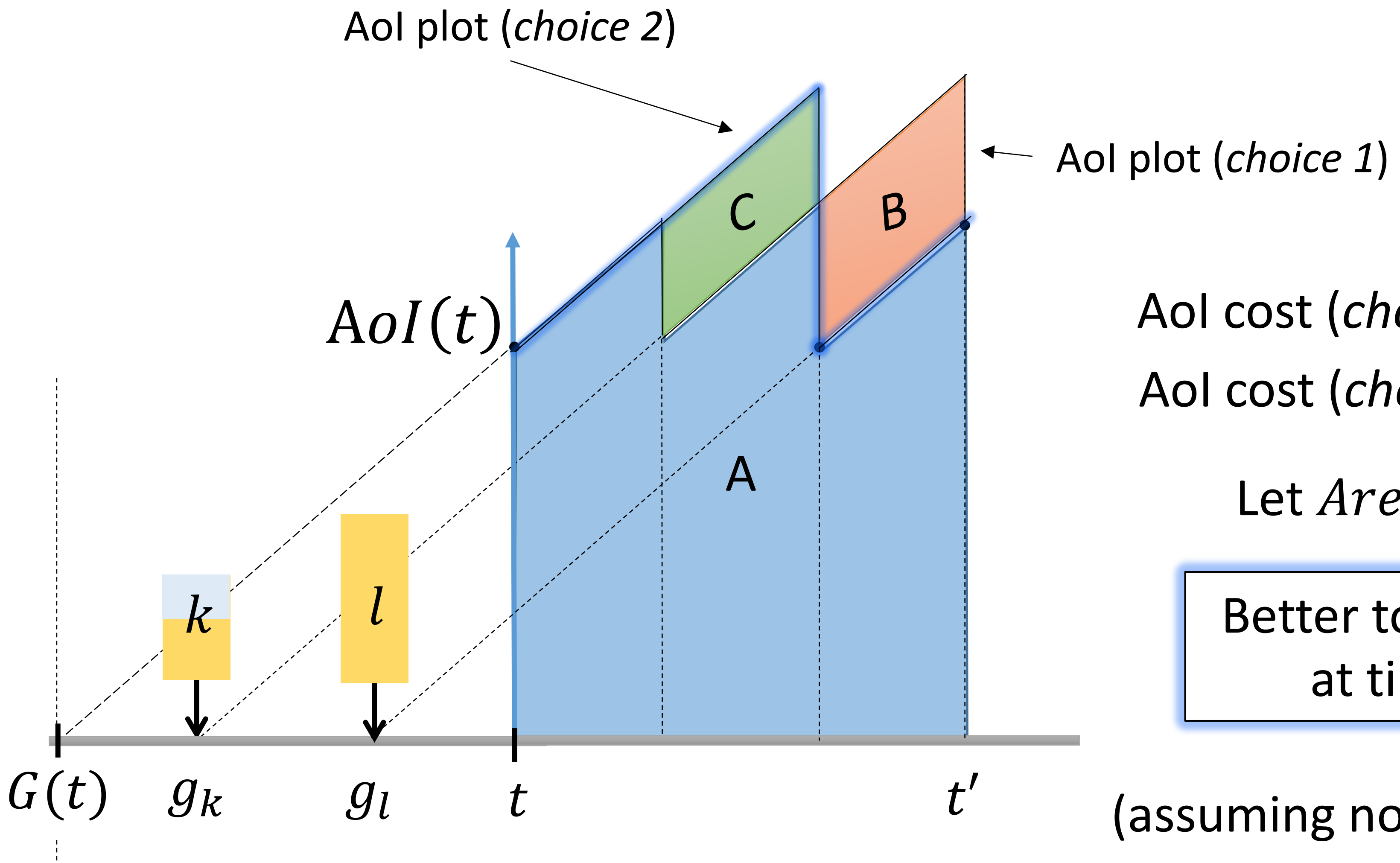
AoI cost (*choice 2*)= $Area(A + C)$

Let $Area(B) > Area(C)$:

Better to transmit update l at time t (*choice 2*)

Which update to transmit?

$c = 0$



Choices (at time t):

1. Transmit update k
2. Transmit update l

AoI cost (*choice 1*)= $Area(A + B)$

AoI cost (*choice 2*)= $Area(A + C)$

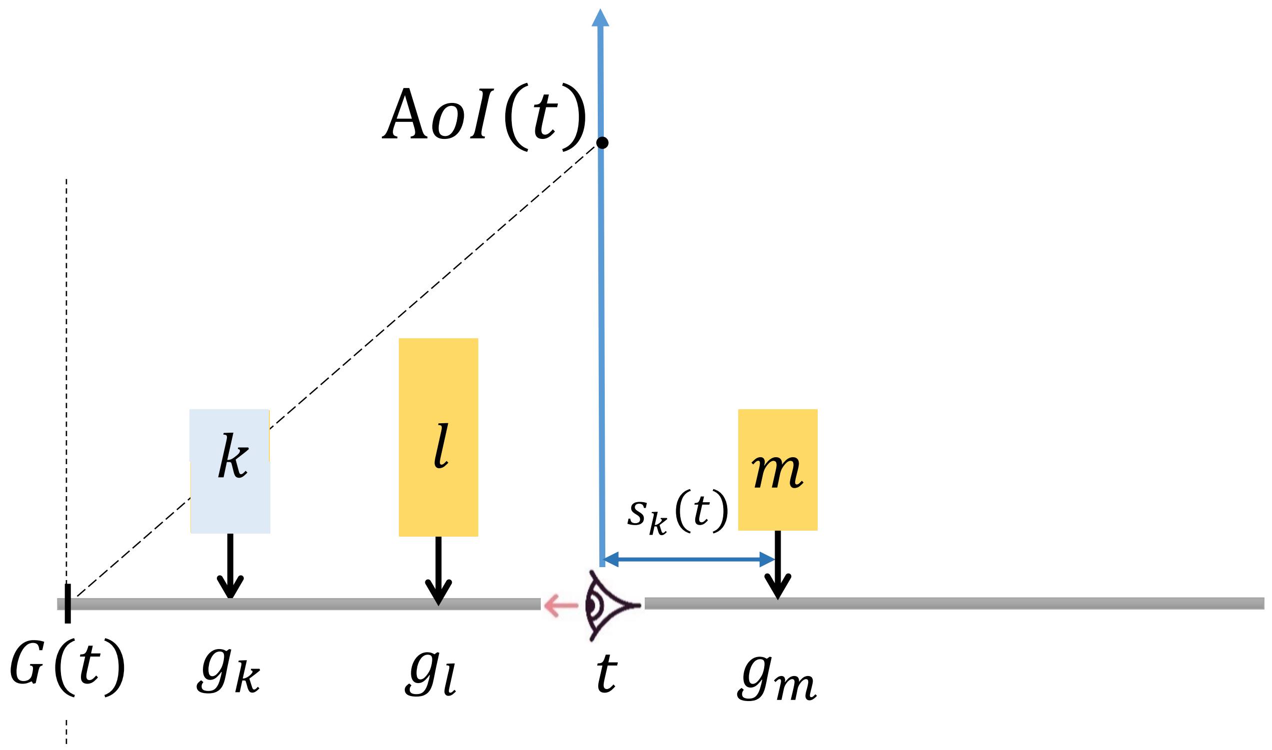
Let $Area(B) > Area(C)$:

Better to transmit update l at time t (*choice 2*)

(assuming no update generated in $[t, t']$)

What if .... a new update m is generated

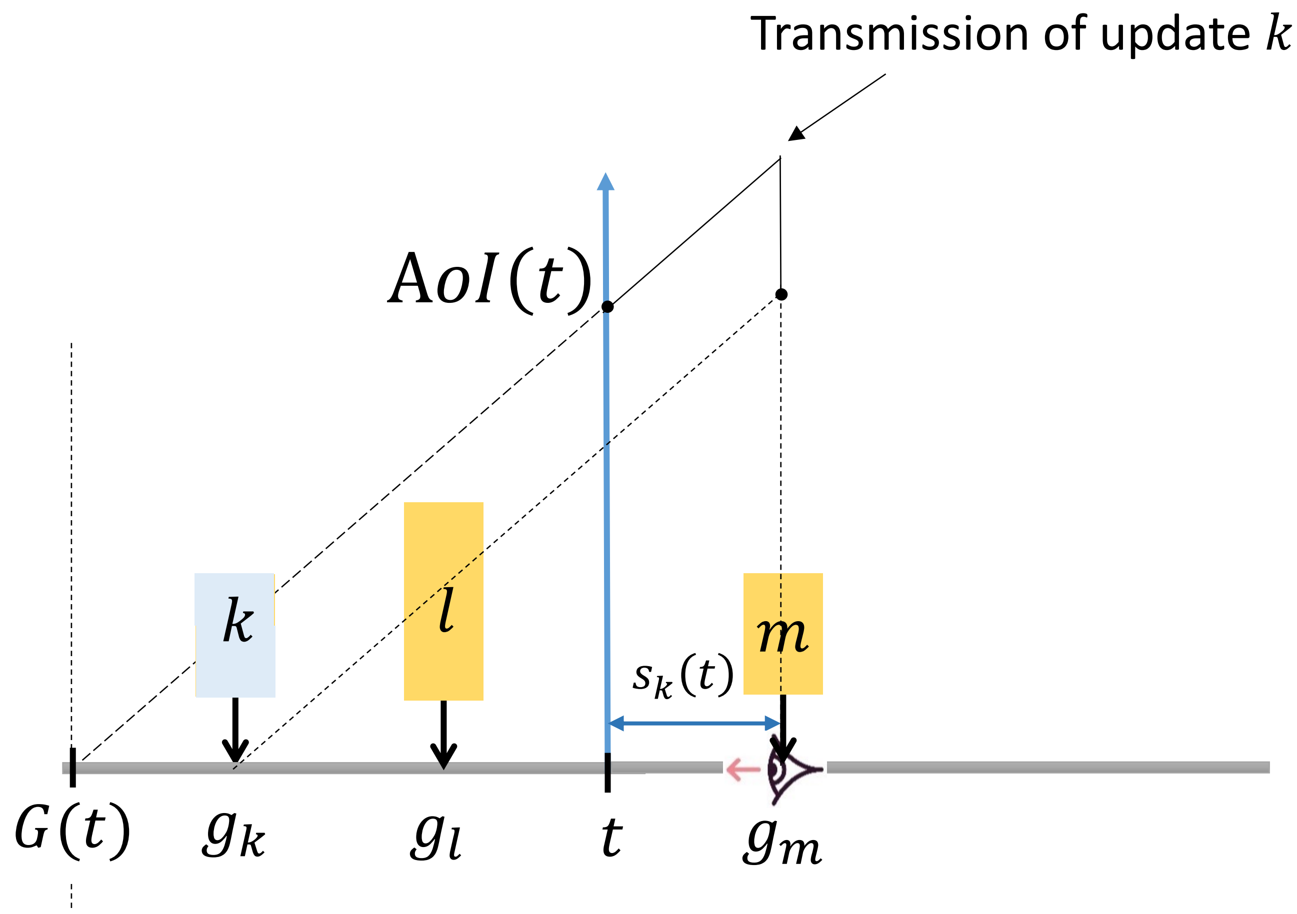
$c = 0$



What if .... a new update m is generated

$c = 0$

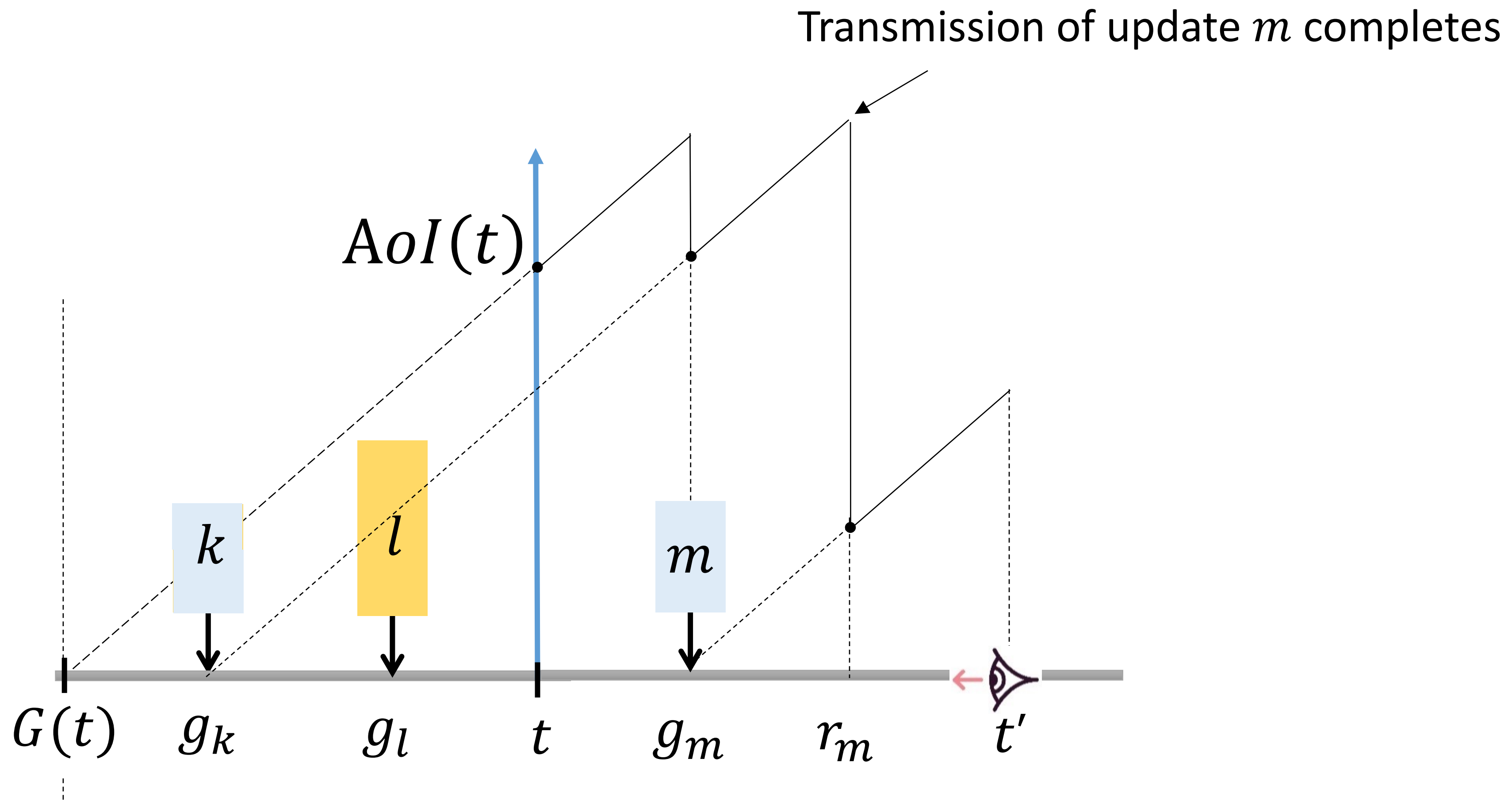
Choice 1: Transmit update k from time t



What if .... a new update m is generated

$c = 0$

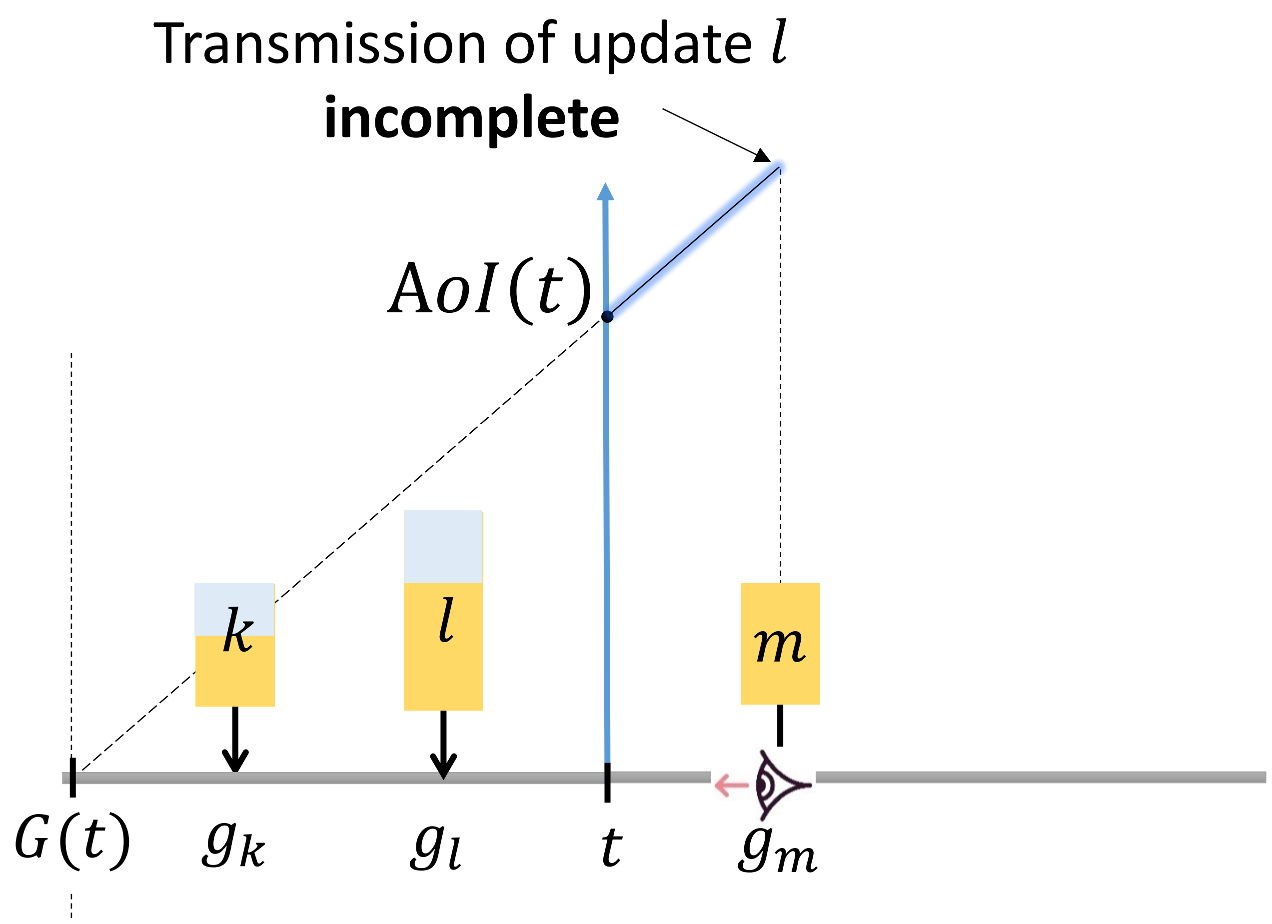
Choice 1: Transmit update k from time t



What if .... a new update m is generated

$c = 0$

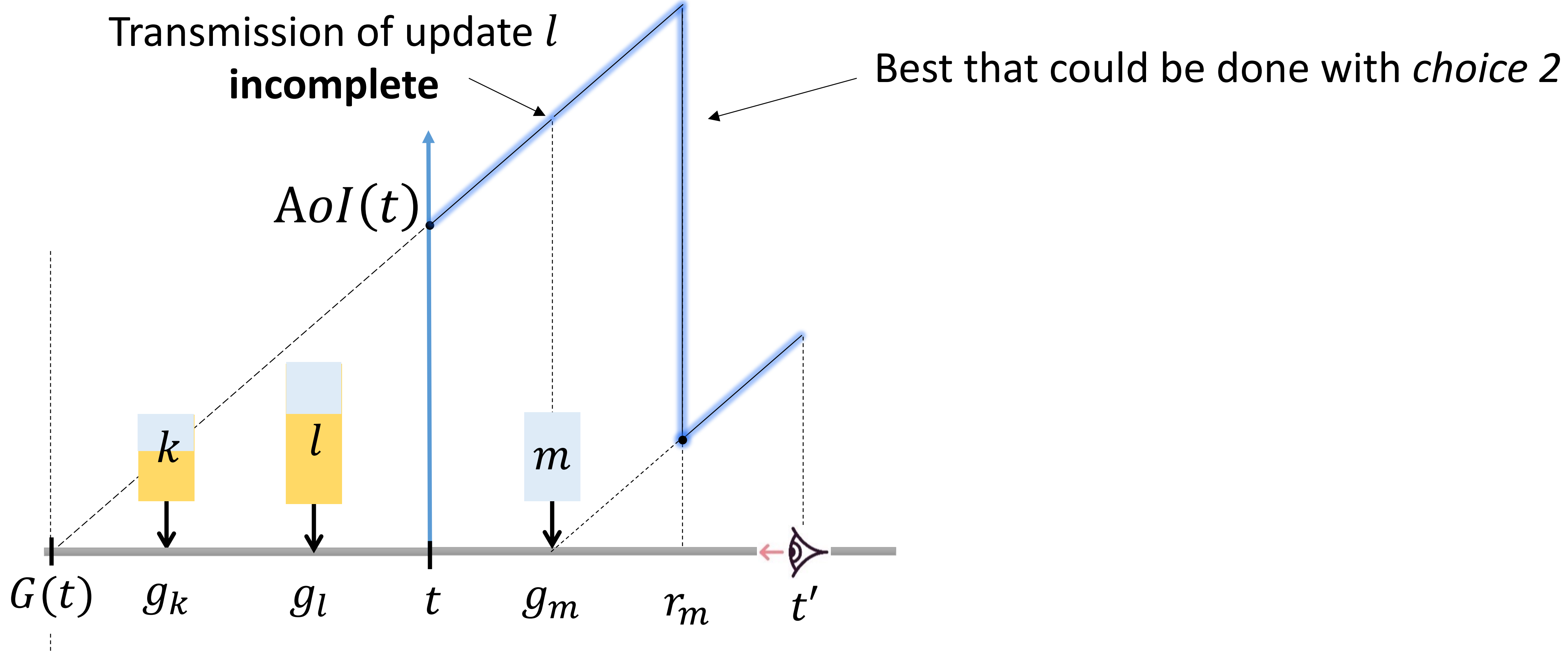
Choice 2: Transmit update l from time t



What if .... a new update m is generated

$c = 0$

Choice 2: Transmit update l from time t

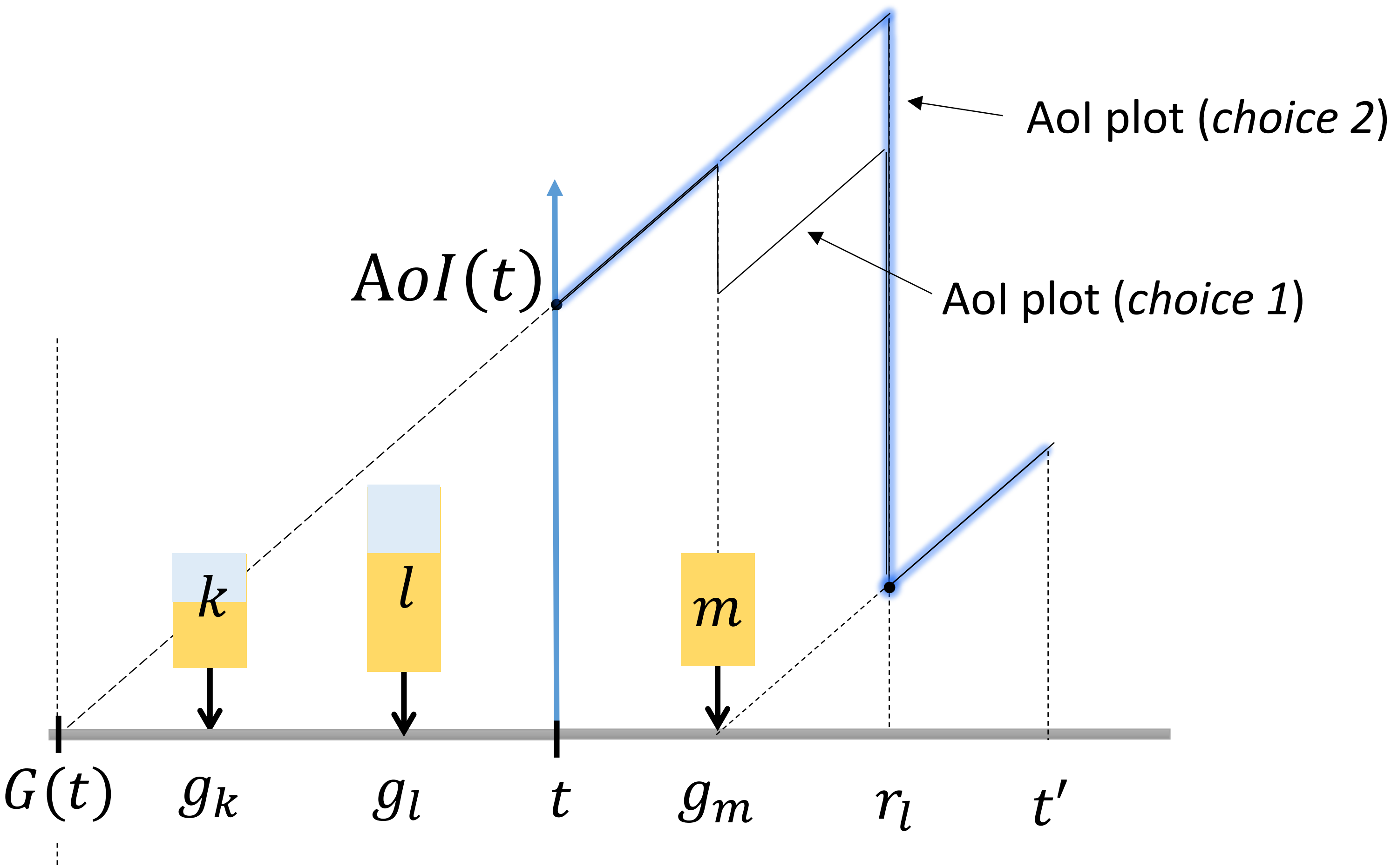


What if .... a new update m is generated

$c = 0$

Choices (at time t):

1. Transmit update k
2. Transmit update l

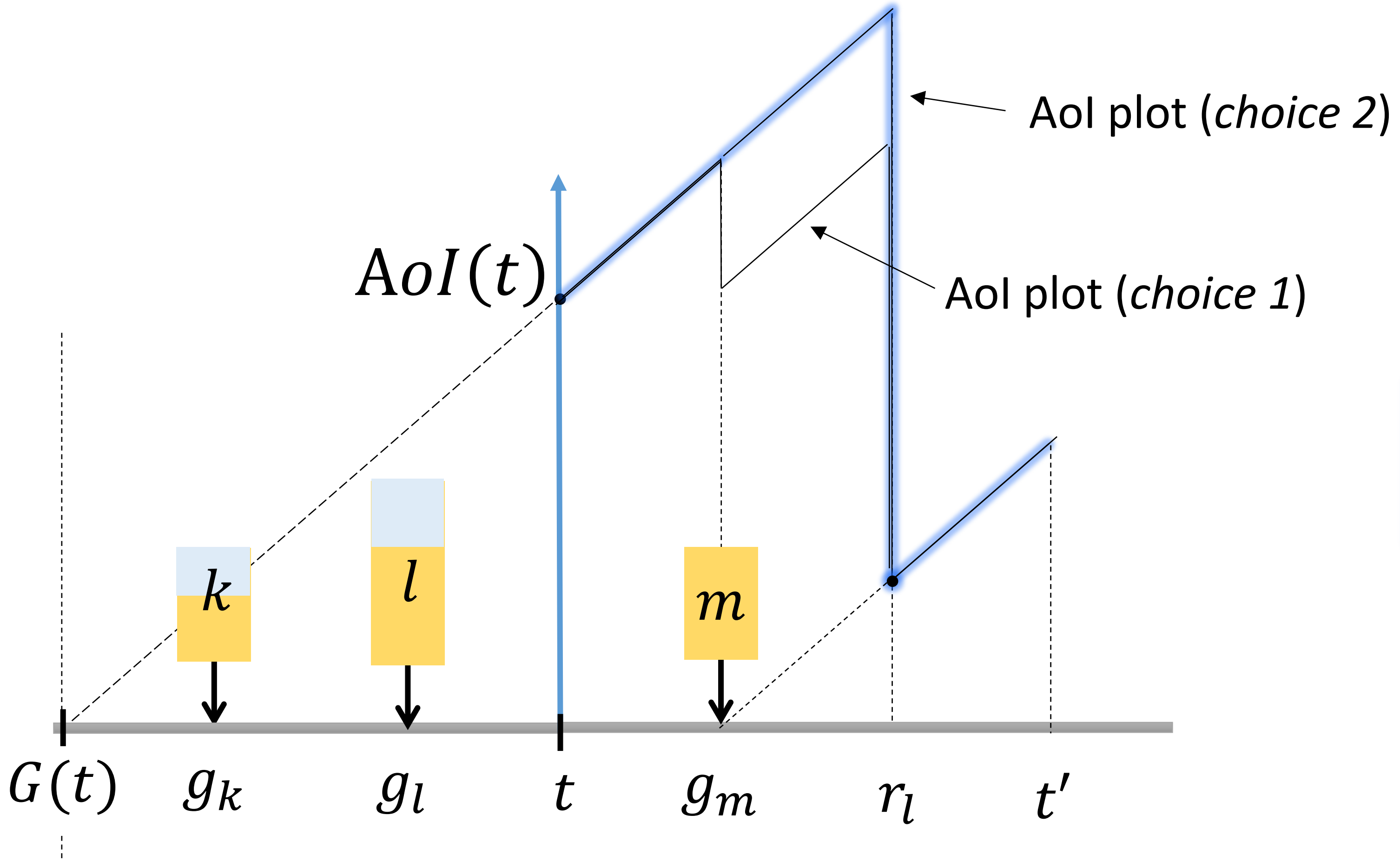


What if .... a new update m is generated

$c = 0$

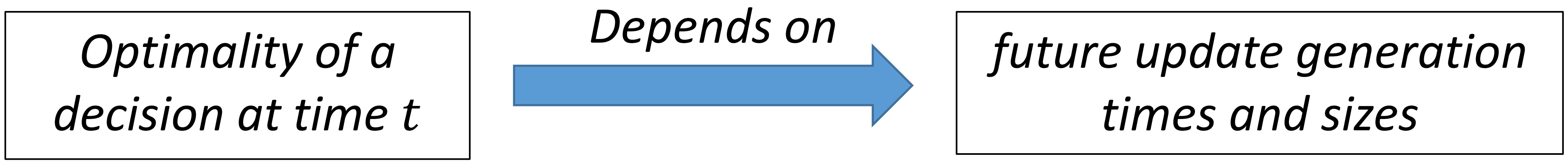
Choices (at time t):

1. Transmit update k
2. Transmit update l

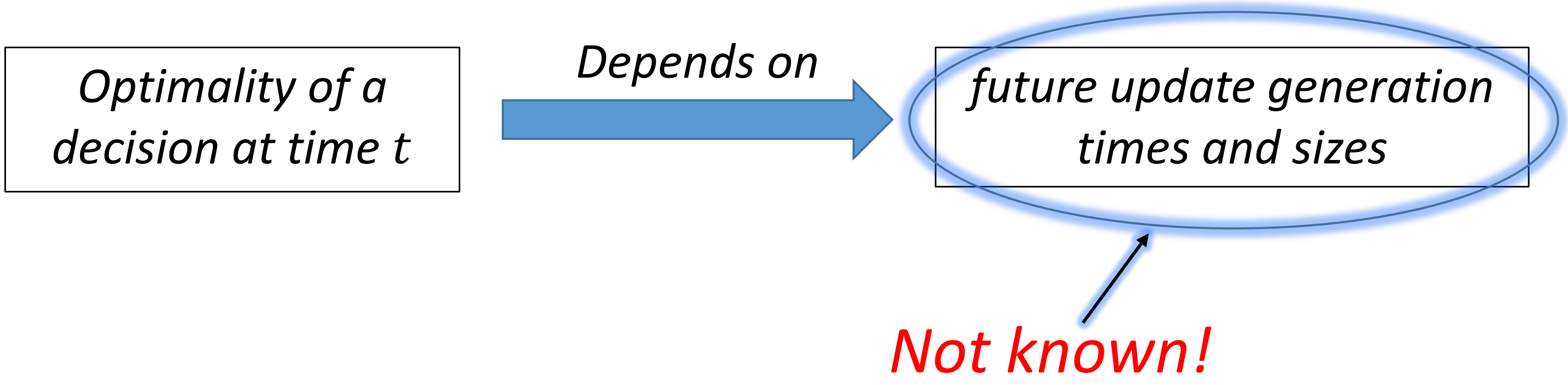


Better to transmit update k at time t (*choice 1*)

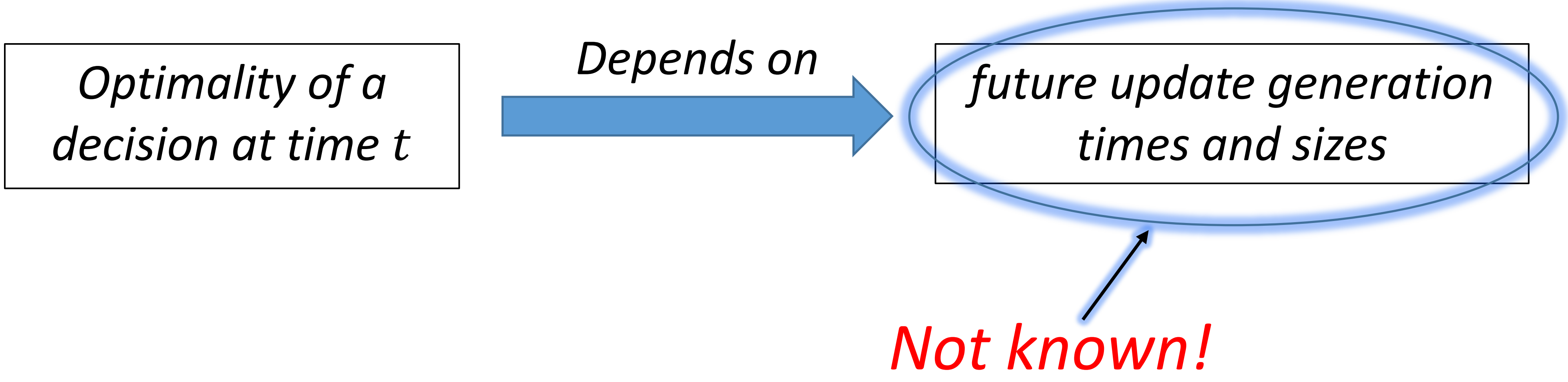
Which update to transmit? (Summary)



Which update to transmit? (Summary)



Which update to transmit? (Summary)



GOAL: Find a causal policy with least ***Competitive Ratio***.

Causal policy: Algorithm that at each time instant, chooses which update to transmit, using only causal information.

Competitive Ratio *(A Metric for Causal Policy)*

Competitive Ratio *(A Metric for Causal Policy)*

- Input σ : sequence of update generation times and sizes (not known to a causal policy).

Competitive Ratio *(A Metric for Causal Policy)*

- Input σ : sequence of update generation times and sizes (not known to a causal policy).
- For given input σ, let
  - $Cost(\pi; \sigma)$: cost incurred by causal policy π.
  - $Cost(*; \sigma)$: cost incurred by an OPT (optimal offline policy that knows σ in advance).

Competitive Ratio *(A Metric for Causal Policy)*

- Input σ : sequence of update generation times and sizes (not known to a causal policy).
- For given input σ, let
  - $Cost(\pi;\sigma)$: cost incurred by causal policy π.
  - $Cost(*;\sigma)$: cost incurred by an OPT (optimal offline policy that knows σ in advance).

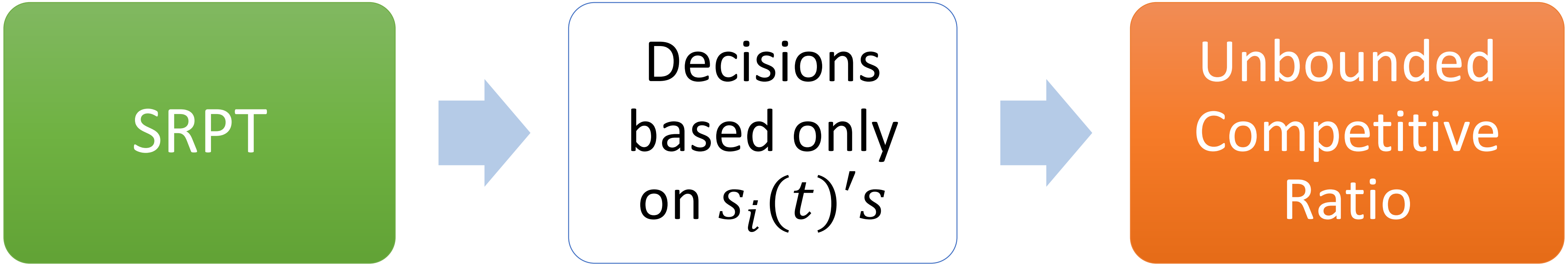
Competitive Ratio of policy π:

$$CR_\pi = \max_\sigma \frac{Cost(\pi;\sigma)}{Cost(*;\sigma)}$$

Recall

SRPT → Decisions based only on $s_i(t)$'s → Unbounded Competitive Ratio

Recall



$Similarly$:

Recall

SRPT → Decisions based only on $s_i(t)'s$ → Unbounded Competitive Ratio

$Similarly$:

LCFS (Last Come First Serve) → Decisions based only on $g_i's$ → Unbounded Competitive Ratio

Greedy Policy

Greedy Policy

→

Accounts for both generation time and size of updates

Greedy Policy

Accounts for both generation time and size of updates

At any time t

For each fresh update i

Define

$$GRADE(i,t) = \frac{g_i - G(t)}{s_i(t)}$$

Greedy Policy

Accounts for both generation time and size of updates

At any time t

For each fresh update i

Define

$$GRADE(i,t) = \frac{g_i - G(t)}{s_i(t)}$$

$G(t)$: generation time of latest update completely transmitted until time t

(Remaining size at time t is 0)

Greedy Policy

Accounts for both generation time and size of updates

At any time t

For each fresh update i

Define

$$GRADE(i,t) = \frac{g_i - G(t)}{s_i(t)}$$

$g_i - G(t)$: Reduction in AoI on tx. of update i

$s_i(t)$: Remaining size of update i

Greedy Policy

Accounts for both generation time and size of updates

At any time t

For each fresh update i

Define

$$GRADE(i,t) = \frac{g_i - G(t)}{s_i(t)}$$

$g_i - G(t)$: Reduction in AoI on tx. of update i

$s_i(t)$: Remaining size of update i

Transmit the update with largest *GRADE*

Greedy Policy

Accounts for both generation time and size of updates

However

At any time t

For each fresh update i

Define

$$GRADE(i,t) = \frac{g_i - G(t)}{s_i(t)}$$

Transmit the update with largest *GRADE*

Greedy Policy

Accounts for both generation time and size of updates

At any time t

For each fresh update i

Define

$$GRADE(i,t) = \frac{g_i - G(t)}{s_i(t)}$$

Transmit the update with largest *GRADE*

However

Large update may preempt small update

Greedy Policy

Accounts for both generation time and size of updates

__At any time t__

For each fresh update i

Define

$$GRADE(i,t) = \frac{g_i - G(t)}{s_i(t)}$$

Transmit the update with largest *GRADE*

__However__

Large update may preempt small update

Possible back-to-back preemptions

Greedy Policy

Accounts for both generation time and size of updates

At any time t

For each fresh update i

Define

$$GRADE(i,t) = \frac{g_i - G(t)}{s_i(t)}$$

Transmit the update with largest *GRADE*

However

Large update may preempt small update

Possible back-to-back preemptions

Large AoI

Greedy Policy

Accounts for both generation time and size of updates

At any time t

For each fresh update i

Define

$$GRADE(i,t) = \frac{g_i - G(t)}{s_i(t)}$$

Transmit the update with largest *GRADE*

However

Large update may preempt small update

Possible back-to-back preemptions

Large AoI

Difficult to bound CR

Greedy Policy

At any time t

For each fresh update i

Define

$$GRADE(i,t) = \frac{g_i - G(t)}{s_i(t)}$$

Transmit the update with largest *GRADE*

Accounts for both generation time and size of updates

However

Large update may preempt small update

Need to fix!

Possible back-to-back preemptions

Large AoI

Difficult to bound CR

SRPT+

SRPT+

At any time t

Case 1: An update i is under transmission

Case 2: No update under transmission

SRPT+

At any time t

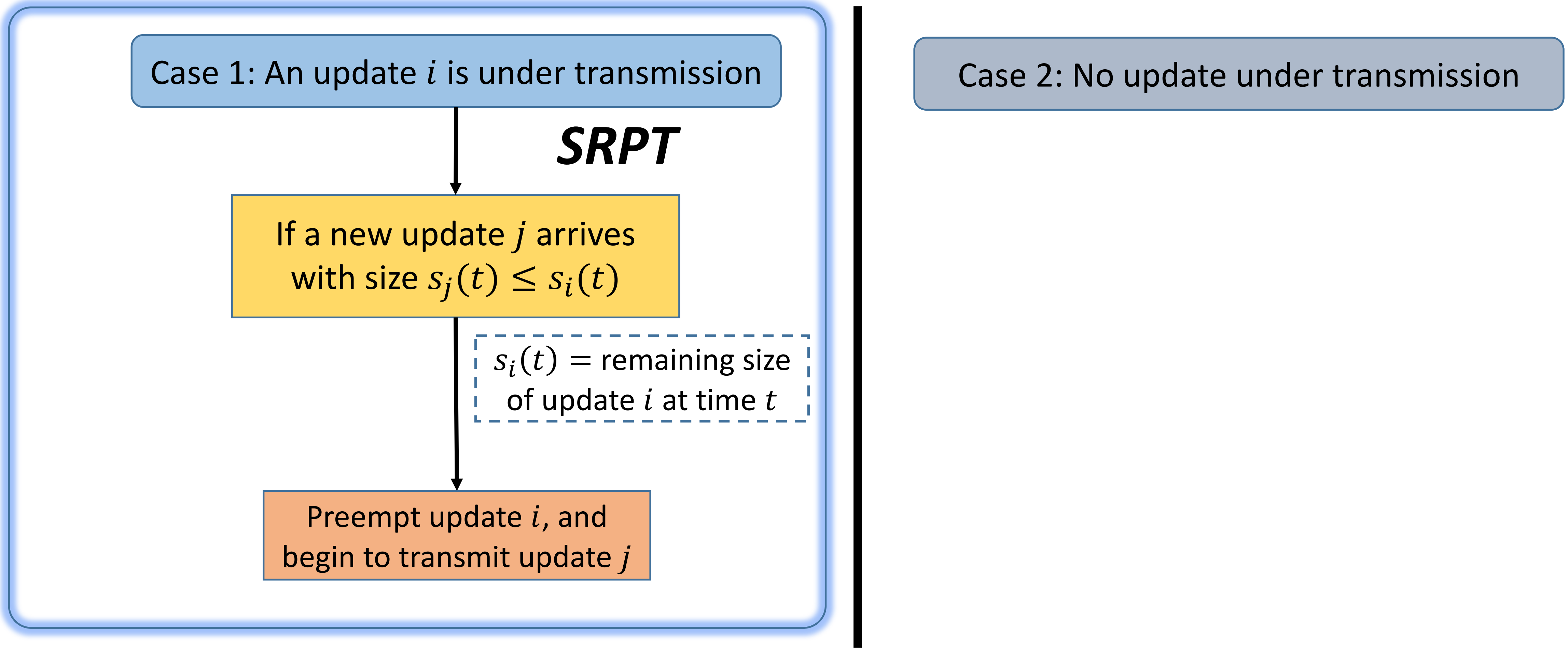
Case 1: An update i is under transmission

SRPT

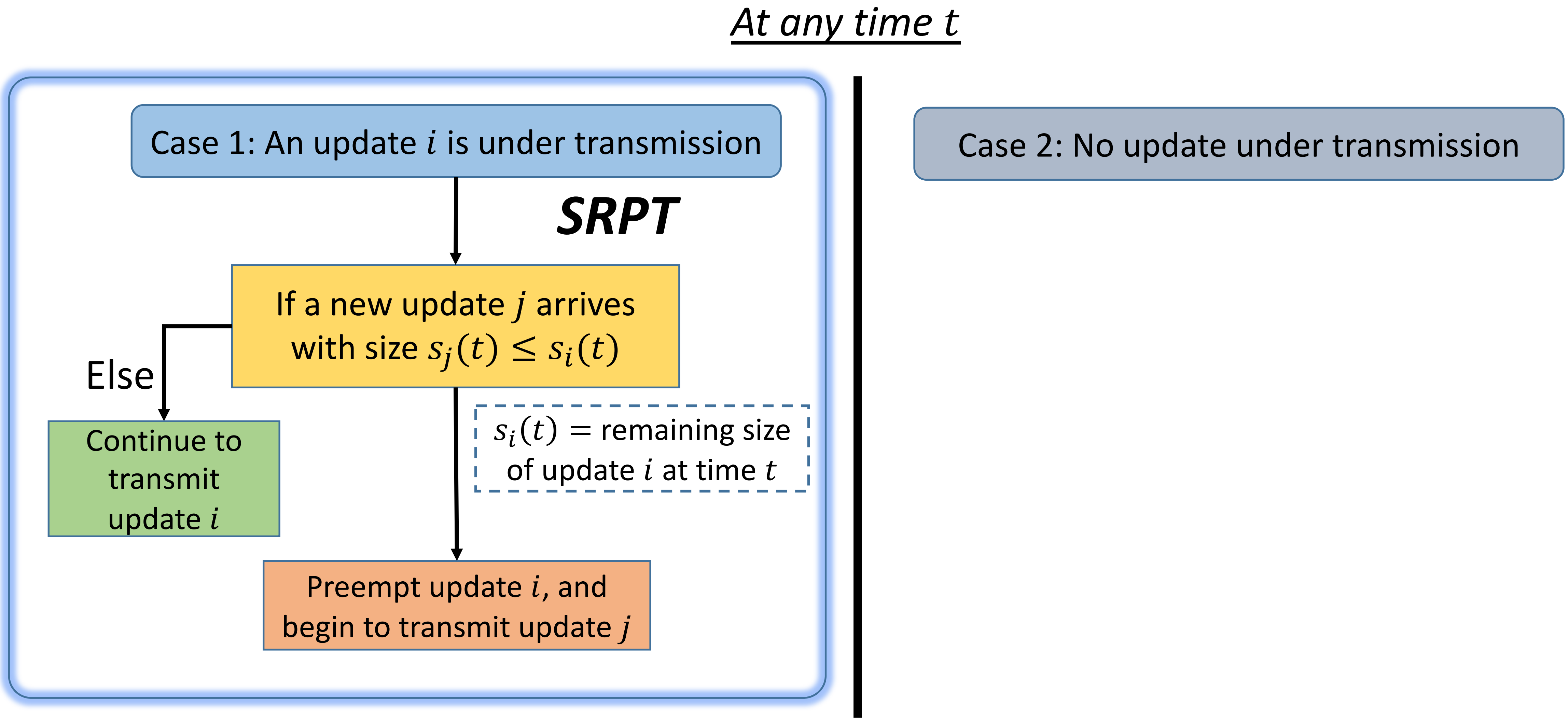
Case 2: No update under transmission

SRPT+

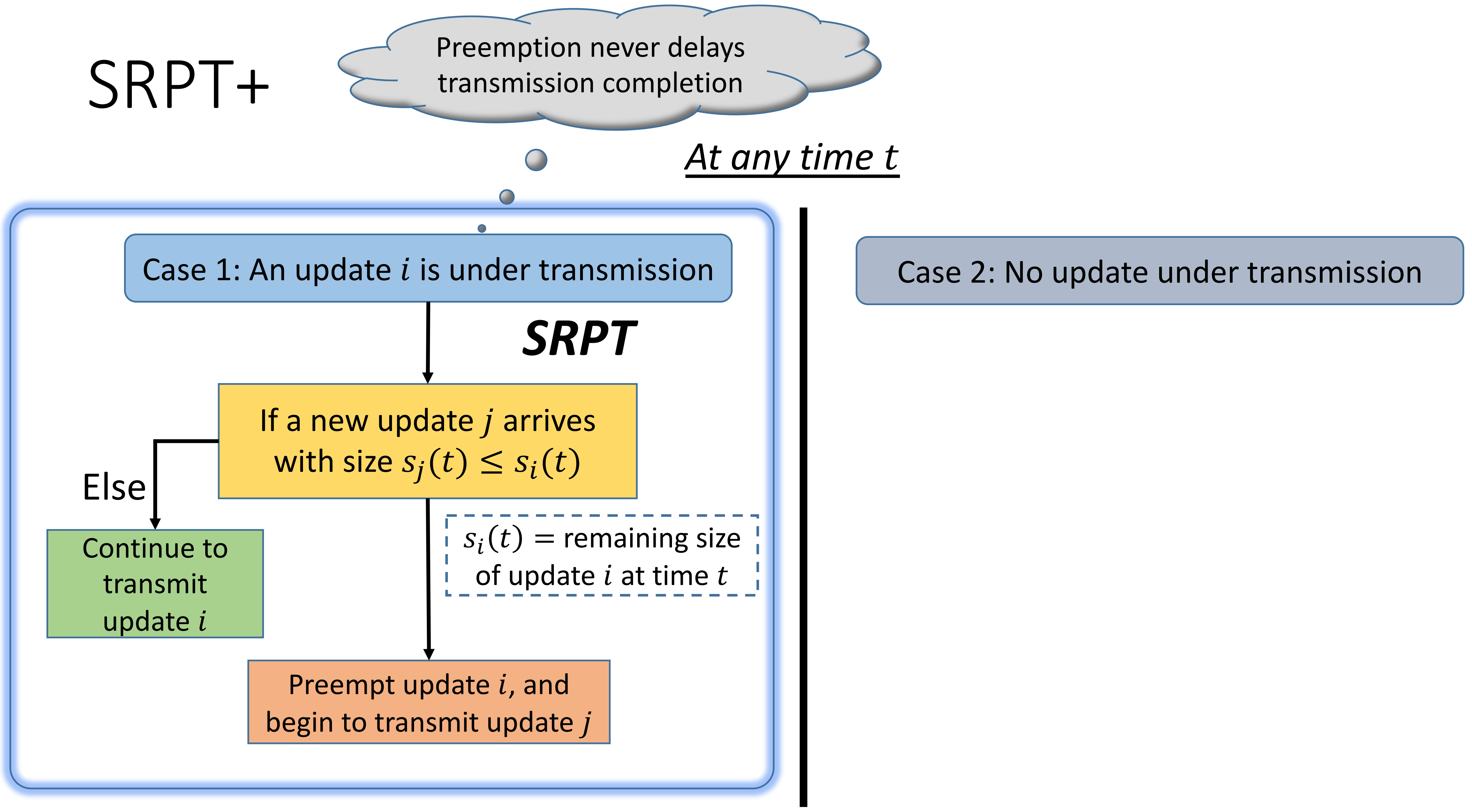
At any time t



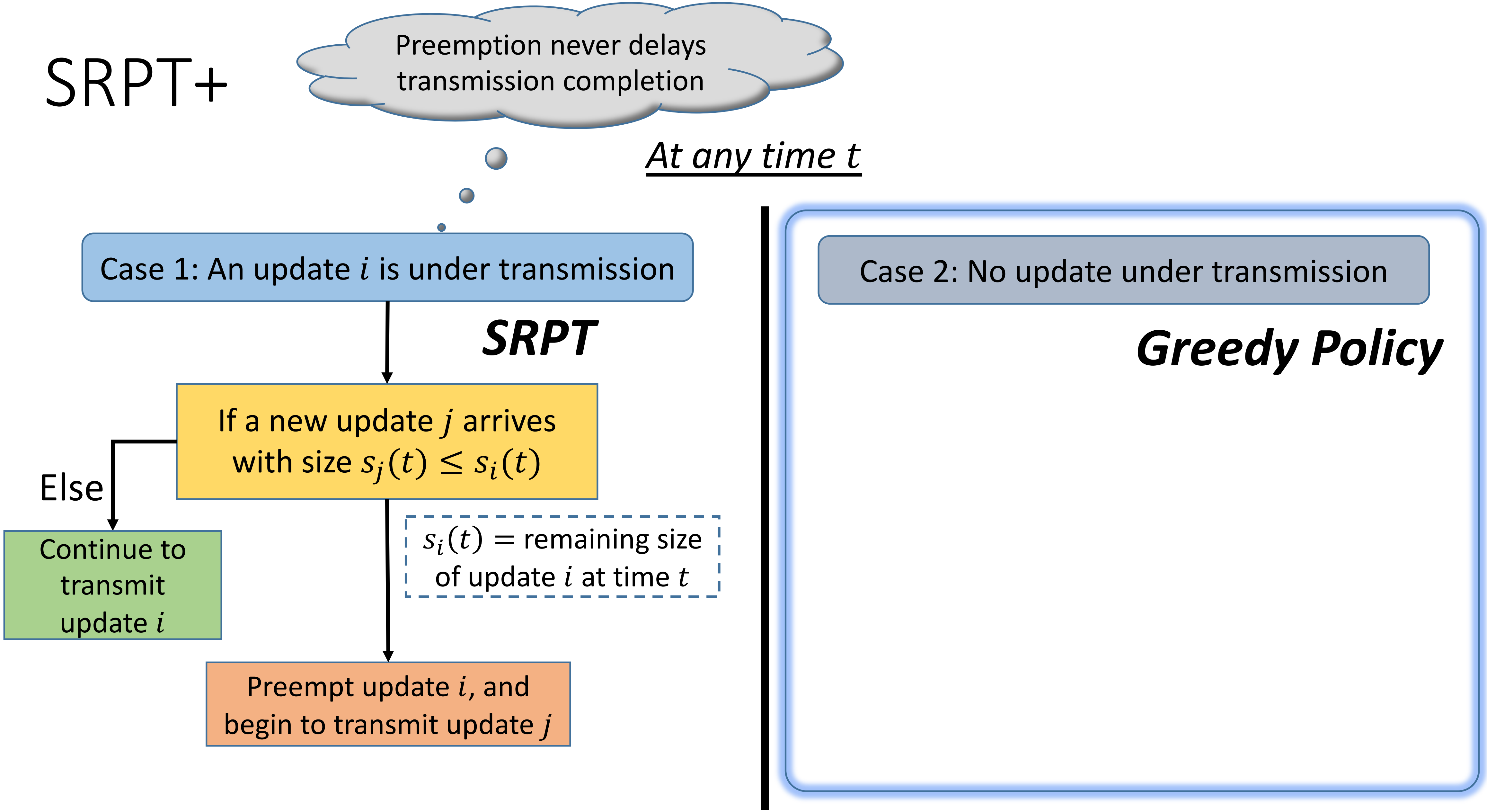
SRPT+

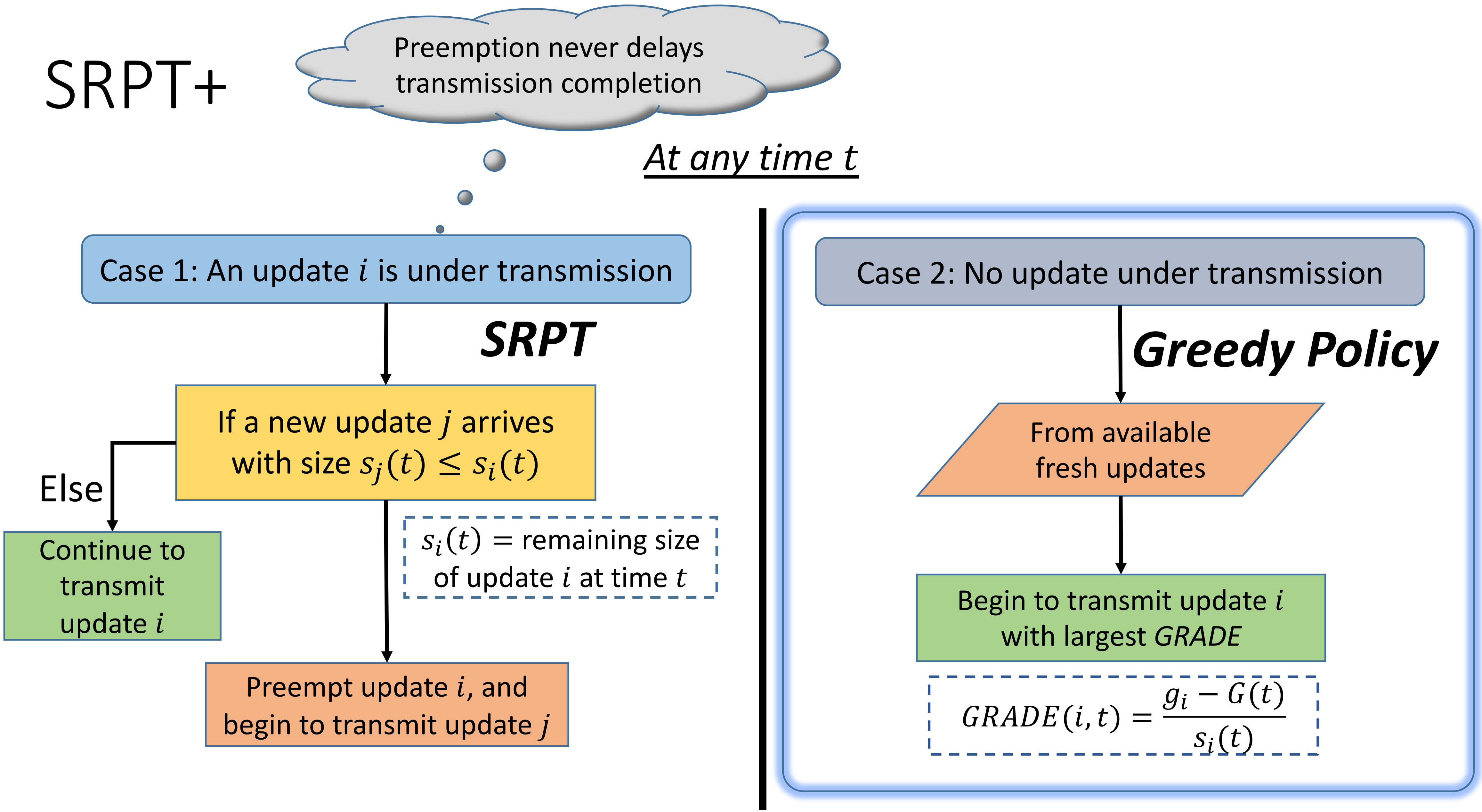


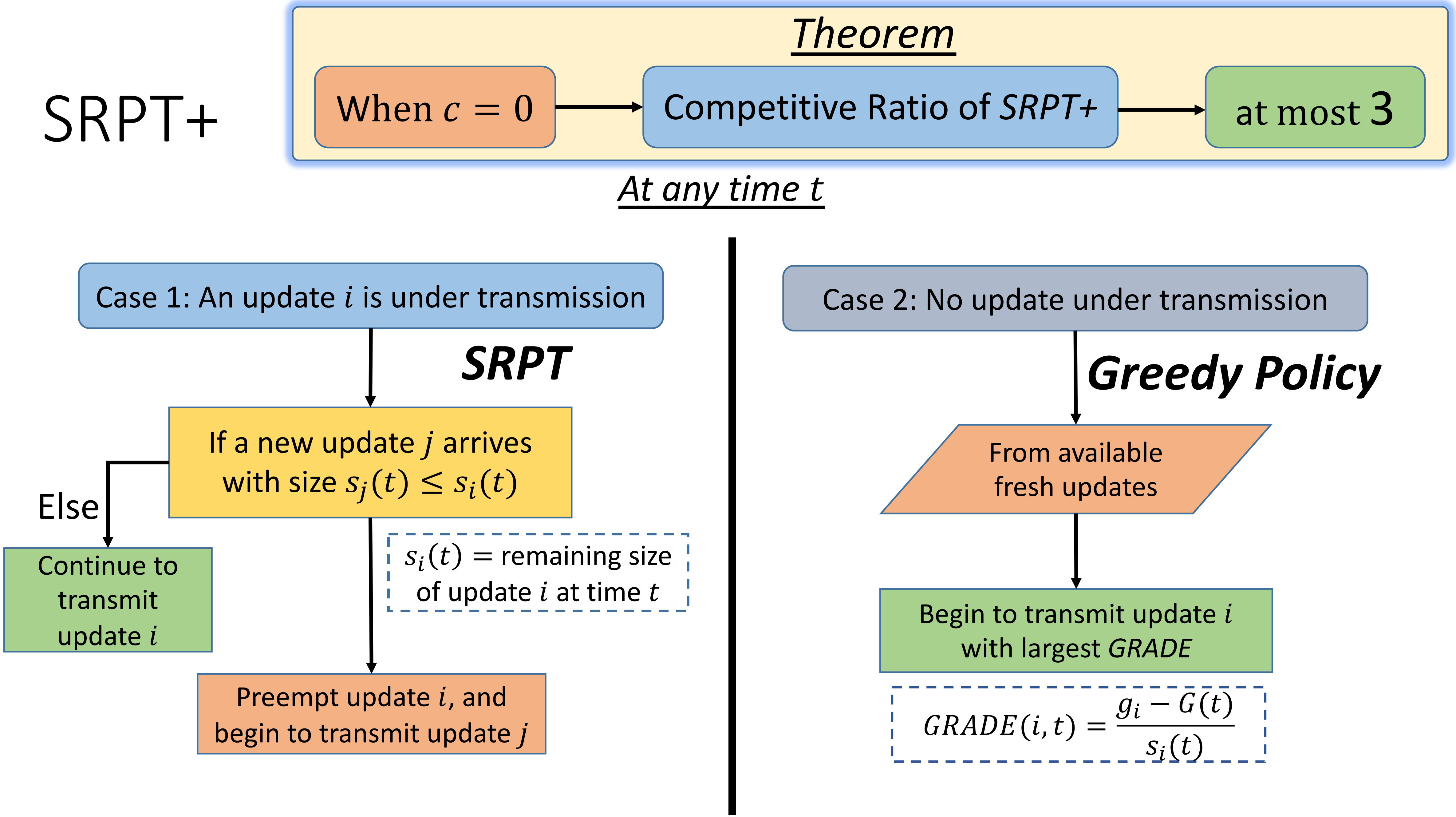
At any time t
Case 1: An update i is under transmission
SRPT
If a new update j arrives with size $s_j(t) \leq s_i(t)$
Else
Continue to transmit update i
$s_i(t)$ = remaining size of update i at time t
Preempt update i, and begin to transmit update j
Case 2: No update under transmission


SRPT+
Preemption never delays transmission completion
At any time t
Case 1: An update i is under transmission
SRPT
If a new update j arrives with size $s_j(t) \leq s_i(t)$
Else
Continue to transmit update i
$s_i(t)$ = remaining size of update i at time t
Preempt update i, and begin to transmit update j
Case 2: No update under transmission

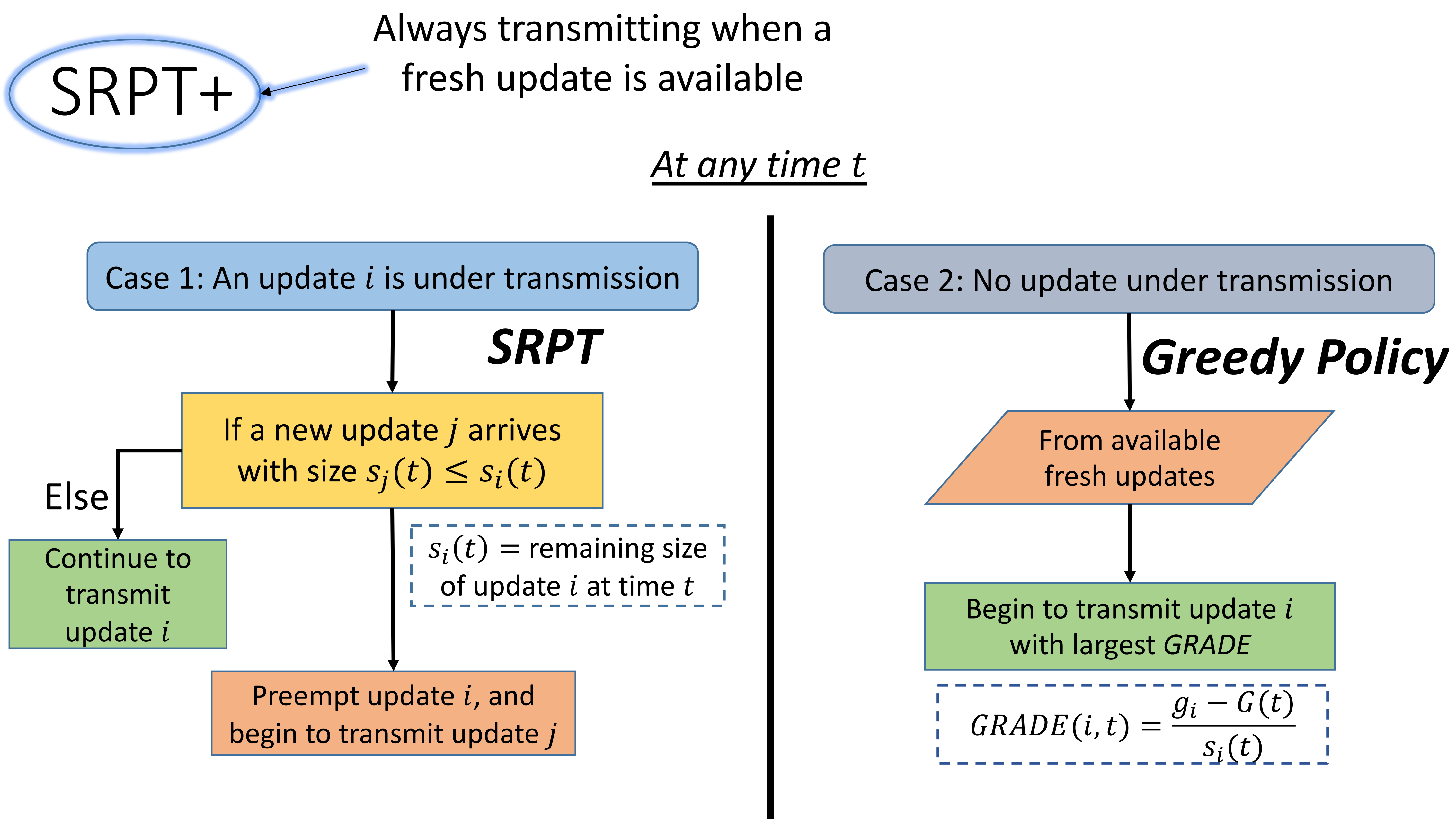
SRPT+

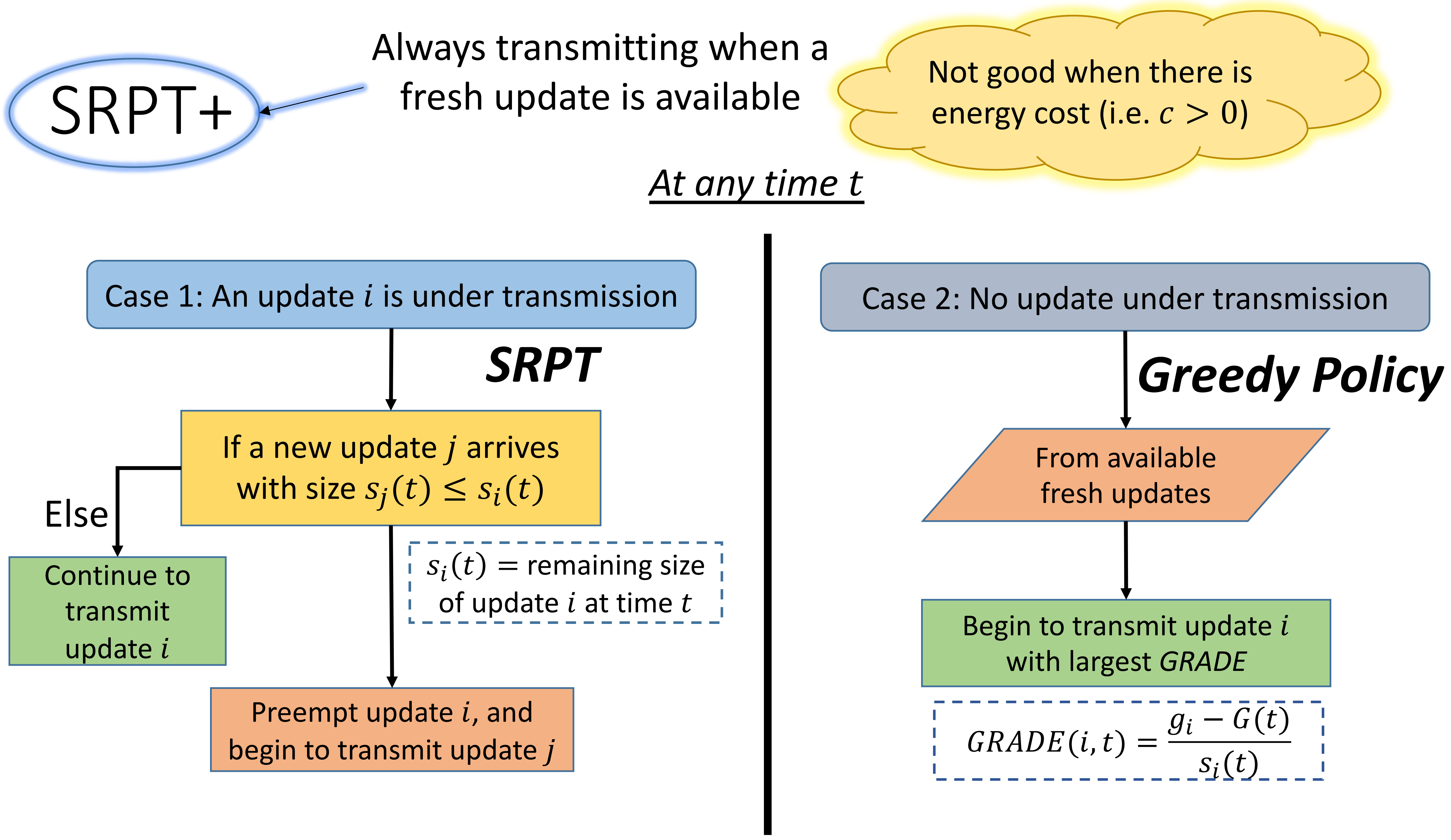

Preemption never delays transmission completion
At any time t
Case 1: An update i is under transmission
SRPT
If a new update j arrives with size $s_j(t) \leq s_i(t)$
Else
Continue to transmit update i
$s_i(t)$ = remaining size of update i at time t
Preempt update i, and begin to transmit update j
Case 2: No update under transmission
Greedy Policy


SRPT+
Preemption never delays transmission completion
At any time t
Case 1: An update i is under transmission
SRPT
If a new update j arrives with size $s_j(t) \le s_i(t)$
Else
Continue to transmit update i
$s_i(t)$ = remaining size of update i at time t
Preempt update i, and begin to transmit update j
Case 2: No update under transmission
Greedy Policy
From available fresh updates
Begin to transmit update i with largest *GRADE*
$GRADE(i,t) = \frac{g_i - G(t)}{s_i(t)}$



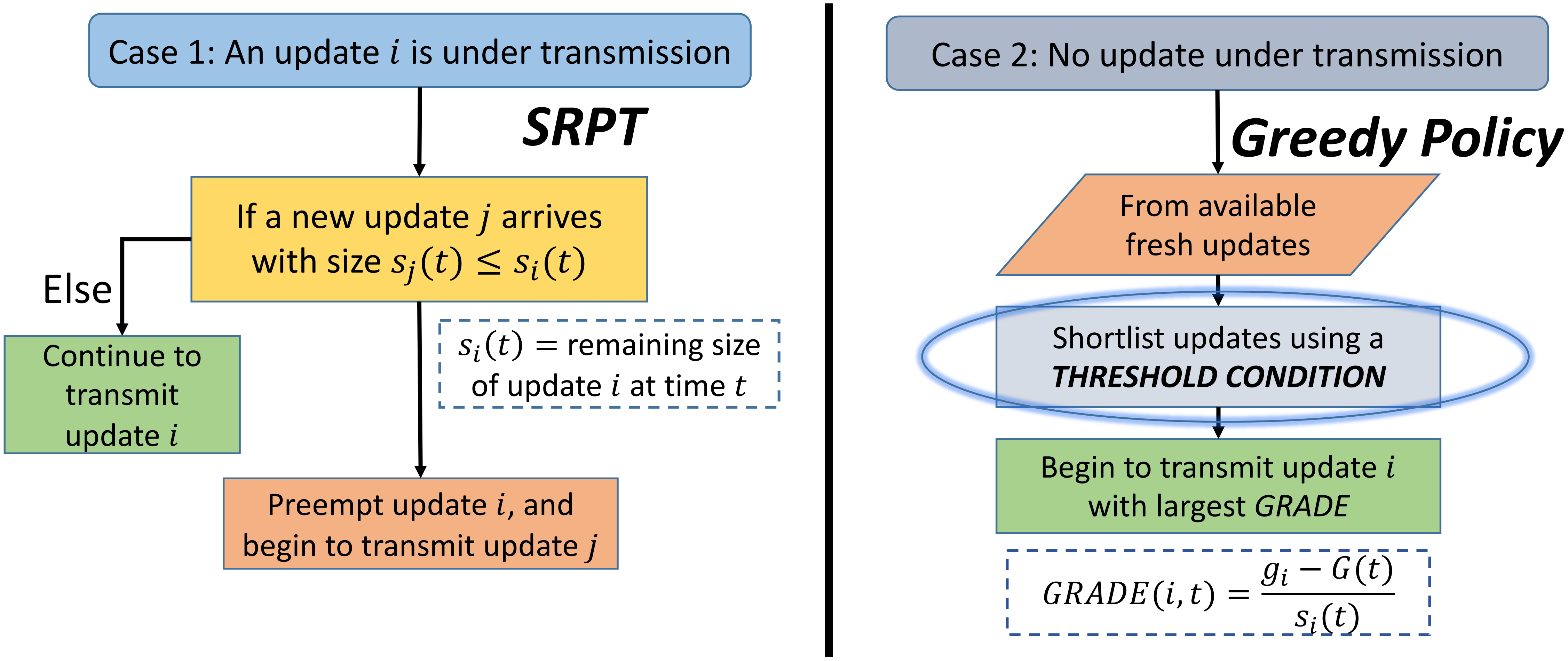
SRPT+
Theorem
When $c = 0$
Competitive Ratio of *SRPT+*
at most 3
At any time t
Case 1: An update i is under transmission
SRPT
If a new update j arrives with size $s_j(t) \leq s_i(t)$
Else
Continue to transmit update i
$s_i(t)$ = remaining size of update i at time t
Preempt update i, and begin to transmit update j
Case 2: No update under transmission
Greedy Policy
From available fresh updates
Begin to transmit update i with largest *GRADE*
$GRADE(i,t) = \frac{g_i - G(t)}{s_i(t)}$


SRPT+
Always transmitting when a fresh update is available
At any time t
Case 1: An update i is under transmission
SRPT
If a new update j arrives with size $s_j(t) \leq s_i(t)$
Else
Continue to transmit update i
$s_i(t)$ = remaining size of update i at time t
Preempt update i, and begin to transmit update j
Case 2: No update under transmission
Greedy Policy
From available fresh updates
Begin to transmit update i with largest *GRADE*
$GRADE(i,t) = \frac{g_i - G(t)}{s_i(t)}$

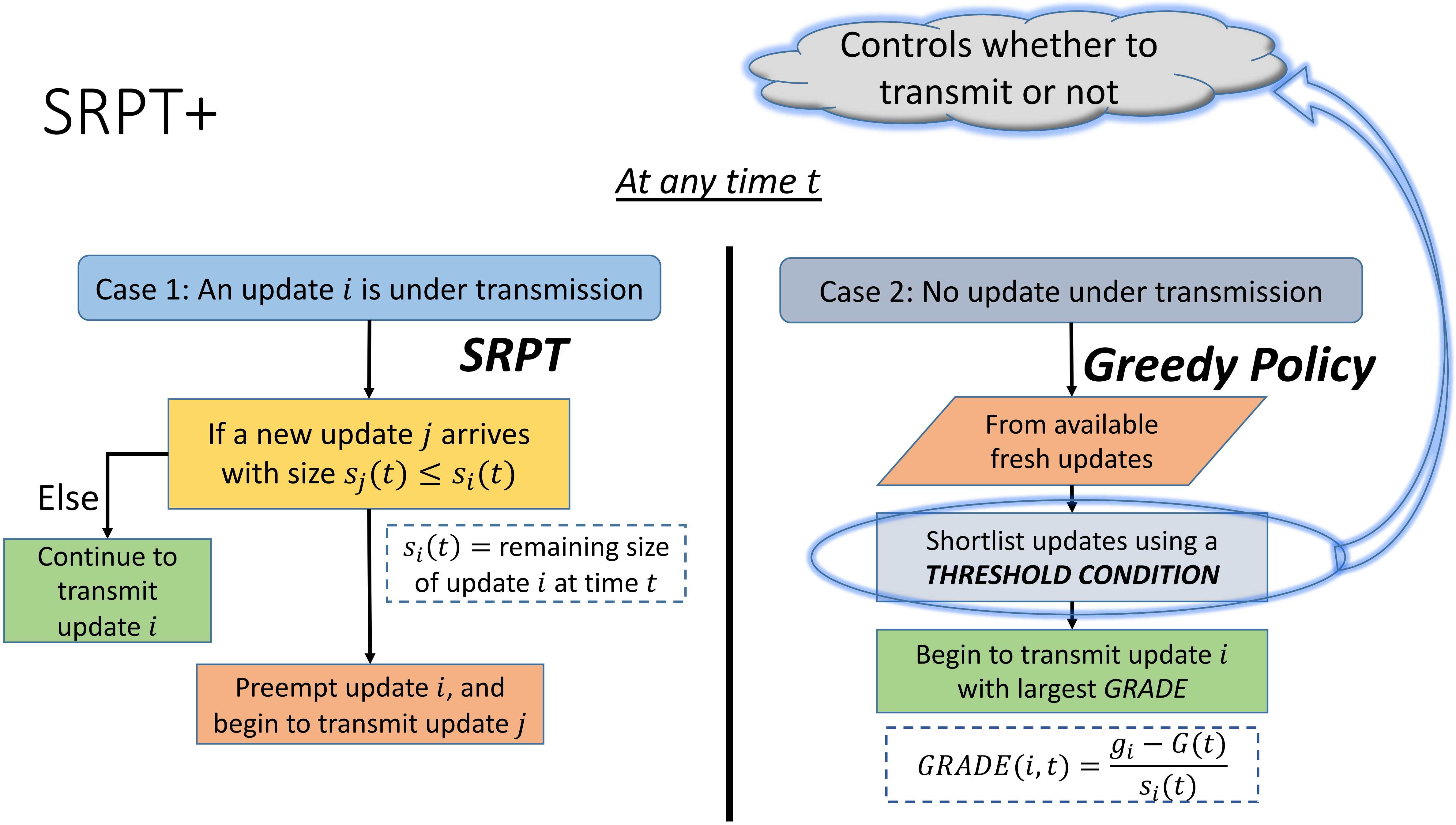

SRPT+
Always transmitting when a fresh update is available
Not good when there is energy cost (i.e. $c > 0$)
At any time t
Case 1: An update i is under transmission
SRPT
If a new update j arrives with size $s_j(t) \leq s_i(t)$
Else
Continue to transmit update i
$s_i(t)$ = remaining size of update i at time t
Preempt update i, and begin to transmit update j
Case 2: No update under transmission
Greedy Policy
From available fresh updates
Begin to transmit update i with largest *GRADE*
$GRADE(i,t) = \frac{g_i - G(t)}{s_i(t)}$

SRPT+

At any time t

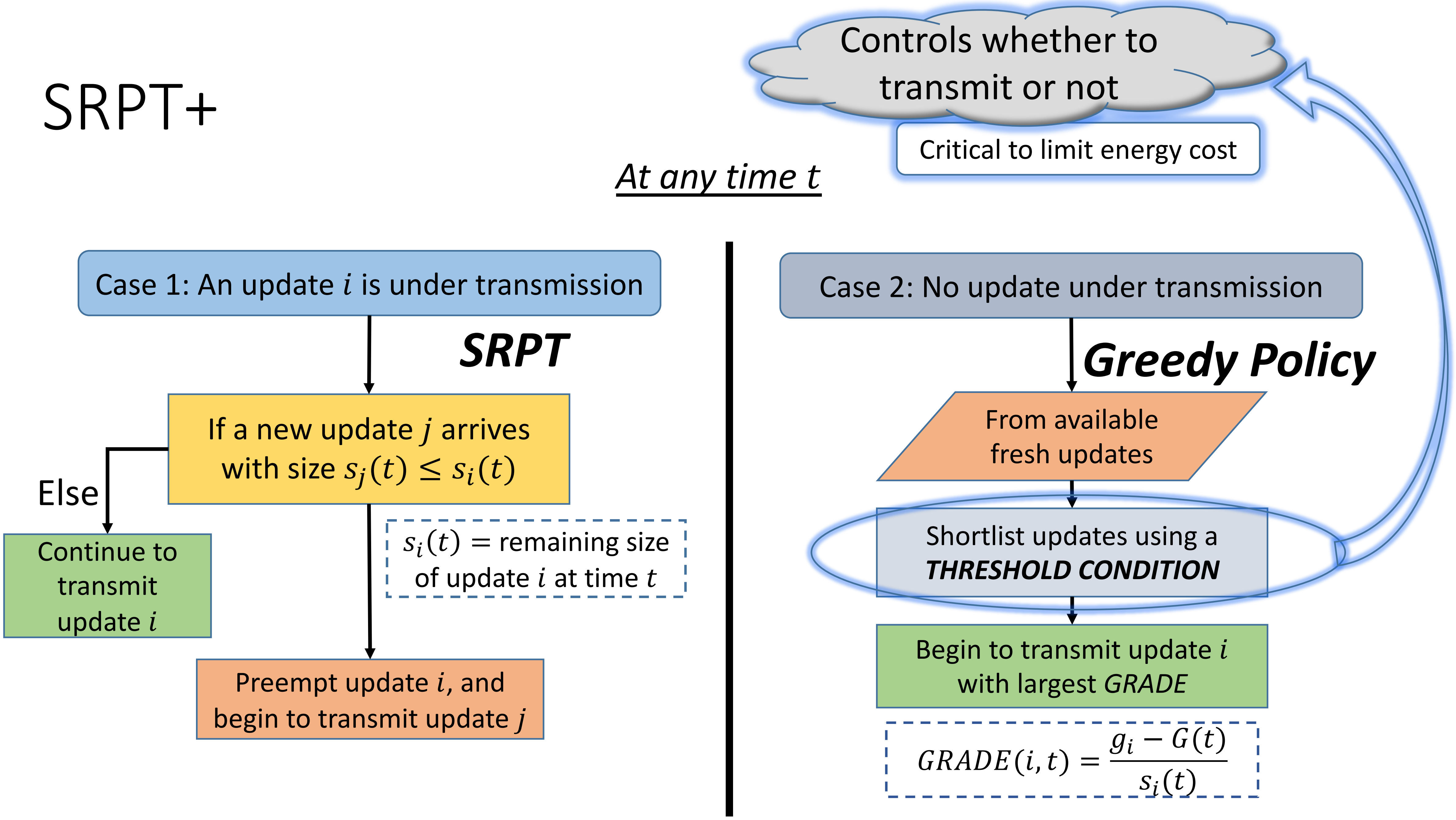


SRPT+



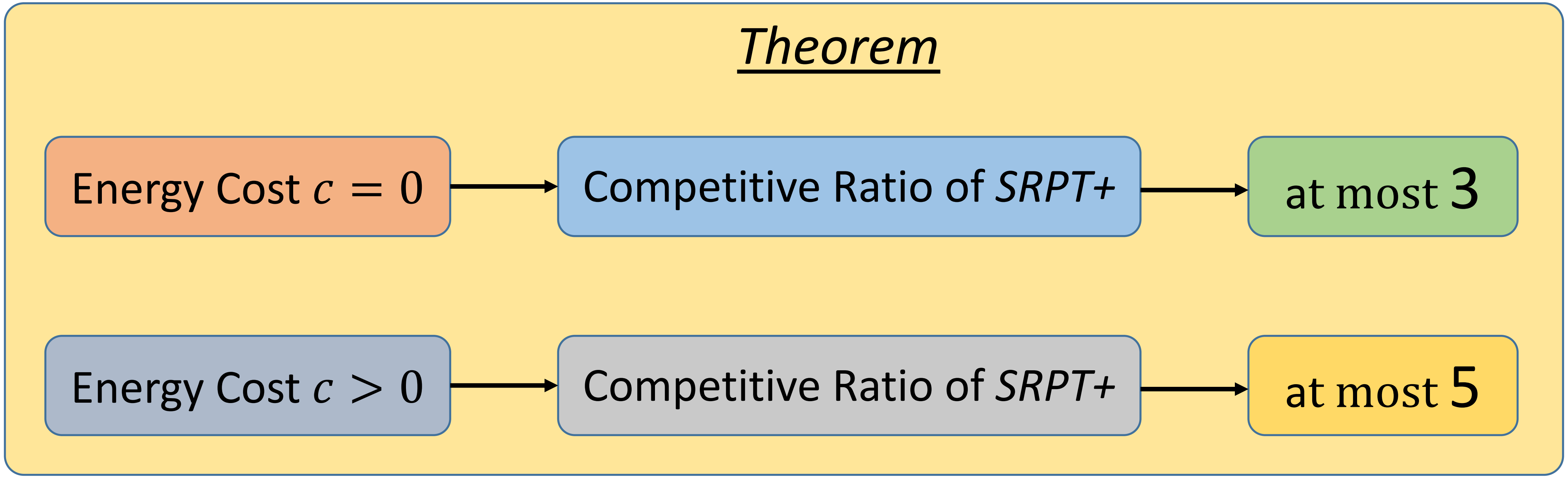
Controls whether to transmit or not
At any time t
Case 1: An update i is under transmission
SRPT
If a new update j arrives with size $s_j(t) \leq s_i(t)$
Else
Continue to transmit update i
$s_i(t)$ = remaining size of update i at time t
Preempt update i, and begin to transmit update j
Case 2: No update under transmission
Greedy Policy
From available fresh updates
Shortlist updates using a THRESHOLD CONDITION
Begin to transmit update i with largest GRADE
$GRADE(i,t) = \frac{g_i - G(t)}{s_i(t)}$

SRPT+



Controls whether to transmit or not
Critical to limit energy cost
At any time t
Case 1: An update i is under transmission
SRPT
If a new update j arrives with size $s_j(t) \leq s_i(t)$
Else
Continue to transmit update i
$s_i(t)$ = remaining size of update i at time t
Preempt update i, and begin to transmit update j
Case 2: No update under transmission
Greedy Policy
From available fresh updates
Shortlist updates using a THRESHOLD CONDITION
Begin to transmit update i with largest GRADE
$GRADE(i,t) = \frac{g_i - G(t)}{s_i(t)}$

Main Result



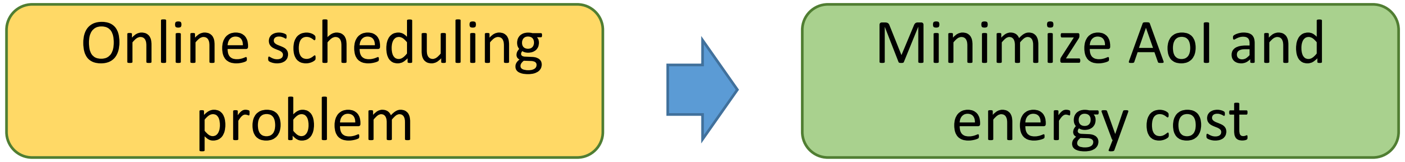
Theorem
Energy Cost $c = 0$
Competitive Ratio of *SRPT+*
at most 3
Energy Cost $c > 0$
Competitive Ratio of *SRPT+*
at most 5

Summary so far

Summary so far

Online scheduling problem

Summary so far



Summary so far

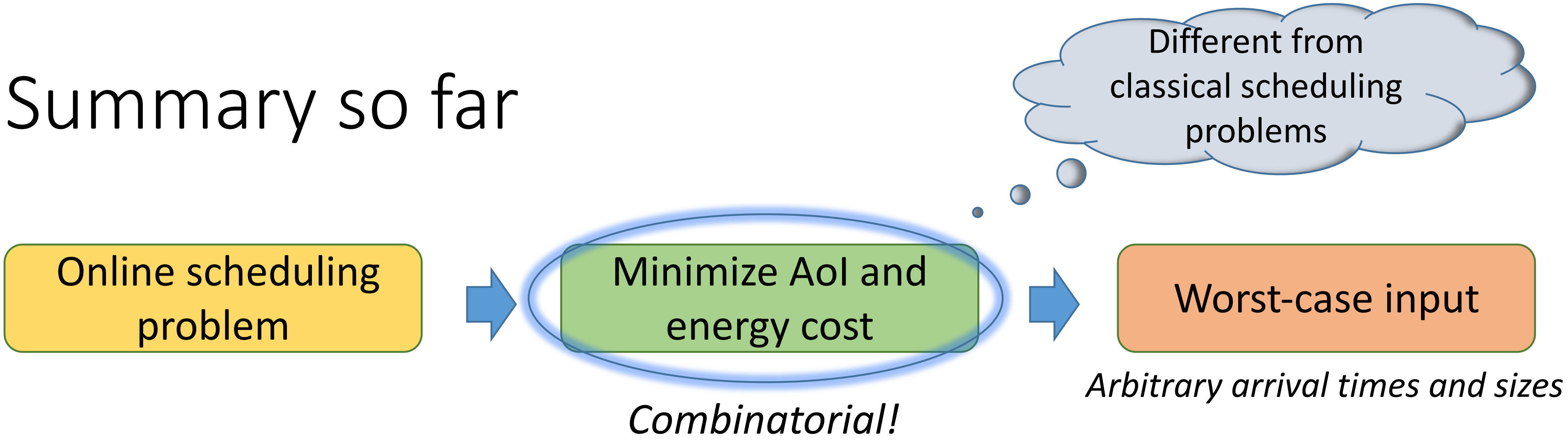
Online scheduling problem

Minimize AoI and energy cost

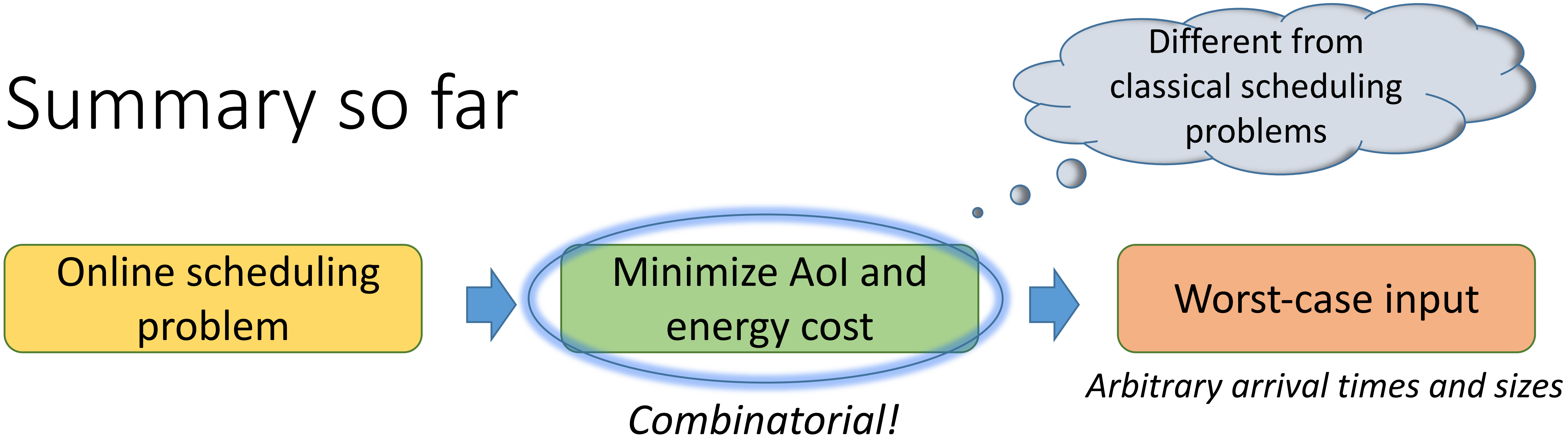
Combinatorial!

Different from classical scheduling problems

Summary so far



Summary so far



Policies such as SRPT, LCFS, etc. has unbounded competitive ratio

Summary so far

Different from classical scheduling problems

Online scheduling problem → Minimize AoI and energy cost → Worst-case input

Combinatorial!

Arbitrary arrival times and sizes

Policies such as SRPT, LCFS, etc. has unbounded competitive ratio

Index-based (Greedy) policy is difficult to analyse

Summary so far

Different from classical scheduling problems

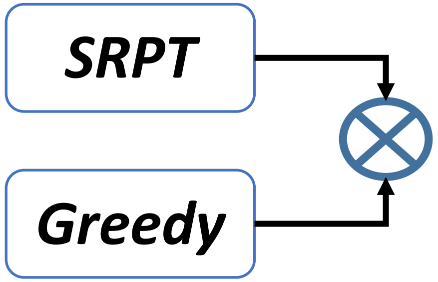
Online scheduling problem → Minimize AoI and energy cost → Worst-case input

Combinatorial!

Arbitrary arrival times and sizes

Policies such as SRPT, LCFS, etc. has unbounded competitive ratio

Index-based (Greedy) policy is difficult to analyse



Summary so far

Different from classical scheduling problems

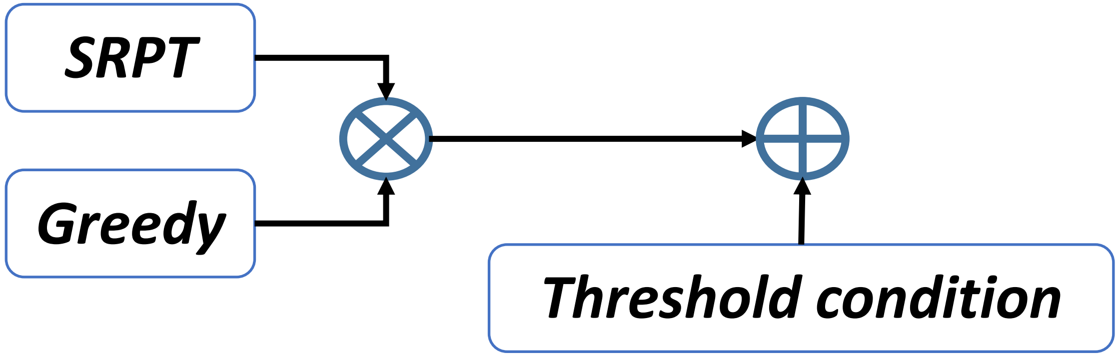
Online scheduling problem → Minimize AoI and energy cost → Worst-case input

Combinatorial!

Arbitrary arrival times and sizes

Policies such as SRPT, LCFS, etc. has unbounded competitive ratio

Index-based (Greedy) policy is difficult to analyse



Summary so far

Different from classical scheduling problems

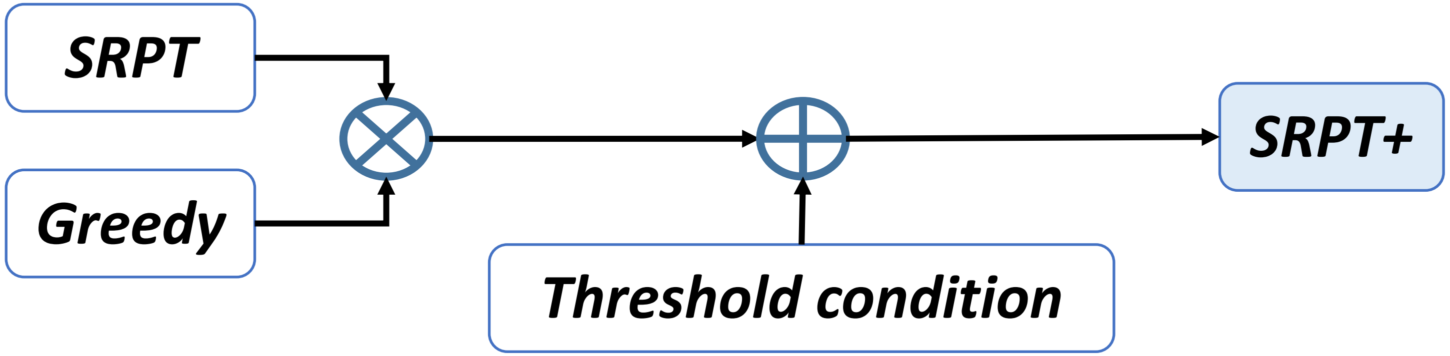
Online scheduling problem → Minimize AoI and energy cost → Worst-case input

Combinatorial!

Arbitrary arrival times and sizes

Policies such as SRPT, LCFS, etc. has unbounded competitive ratio

Index-based (Greedy) policy is difficult to analyse



Summary so far

Different from classical scheduling problems

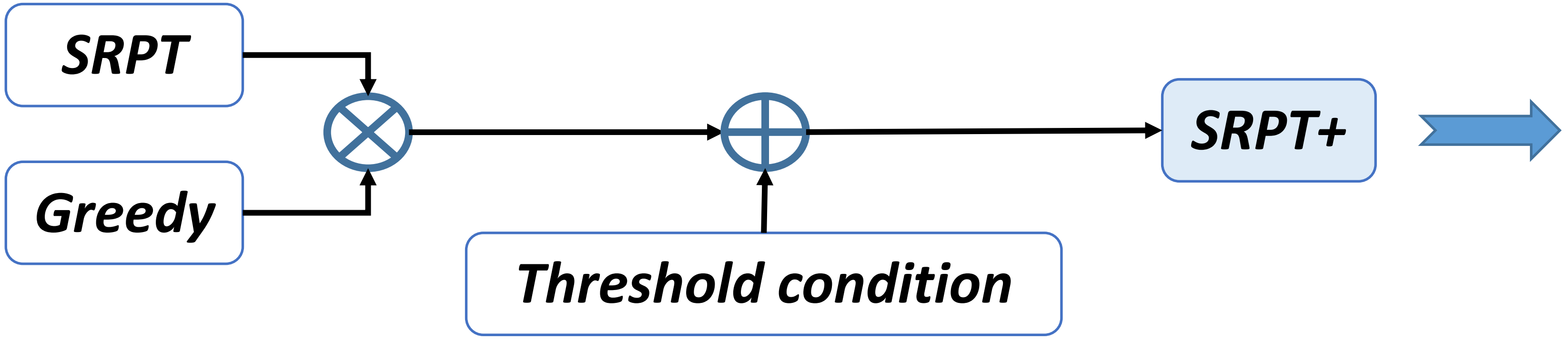
Online scheduling problem → Minimize AoI and energy cost → Worst-case input

Combinatorial!

Arbitrary arrival times and sizes

Policies such as SRPT, LCFS, etc. has unbounded competitive ratio

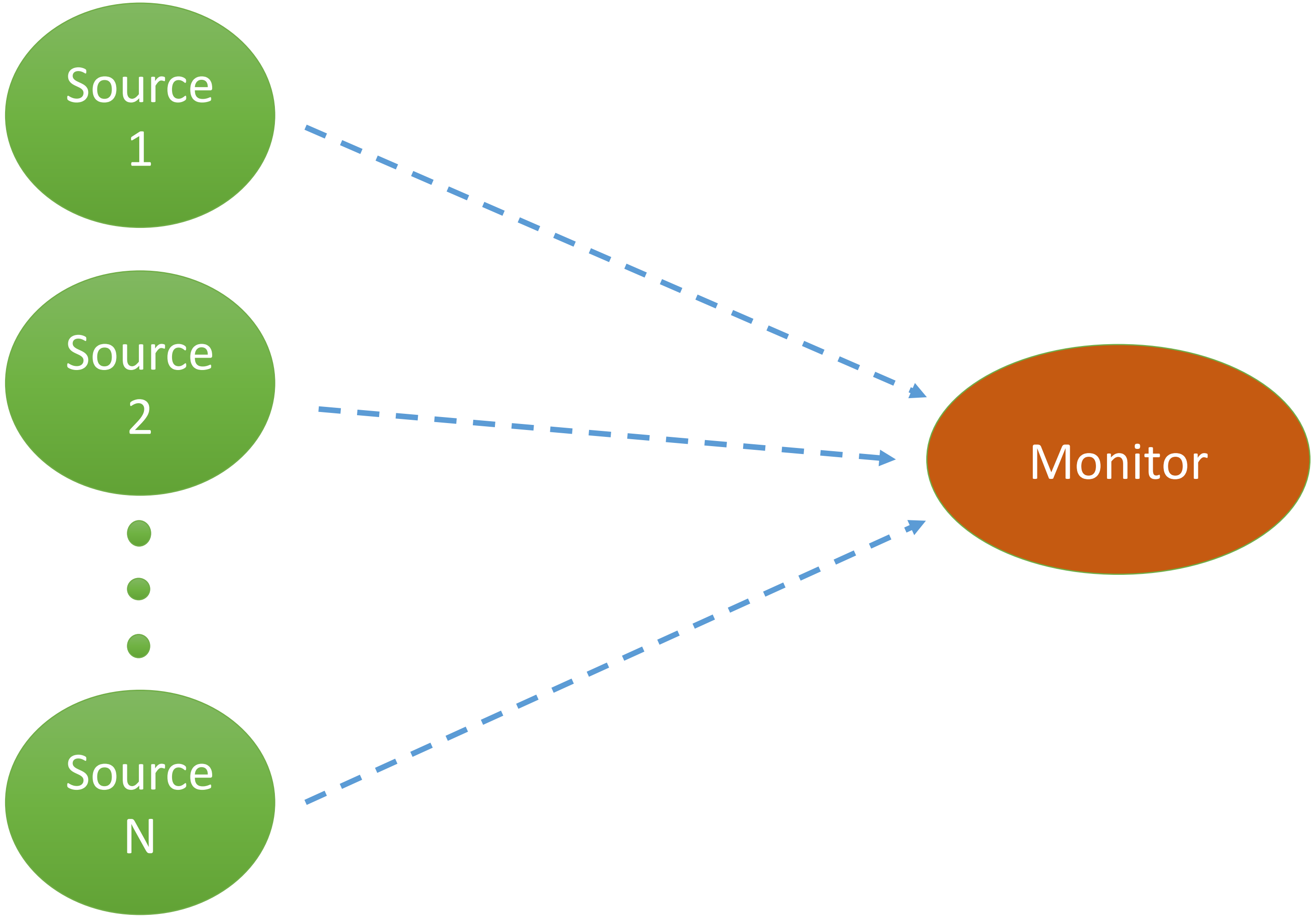
Index-based (Greedy) policy is difficult to analyse



Competitive Ratio at most 5

System Model (General Setting)

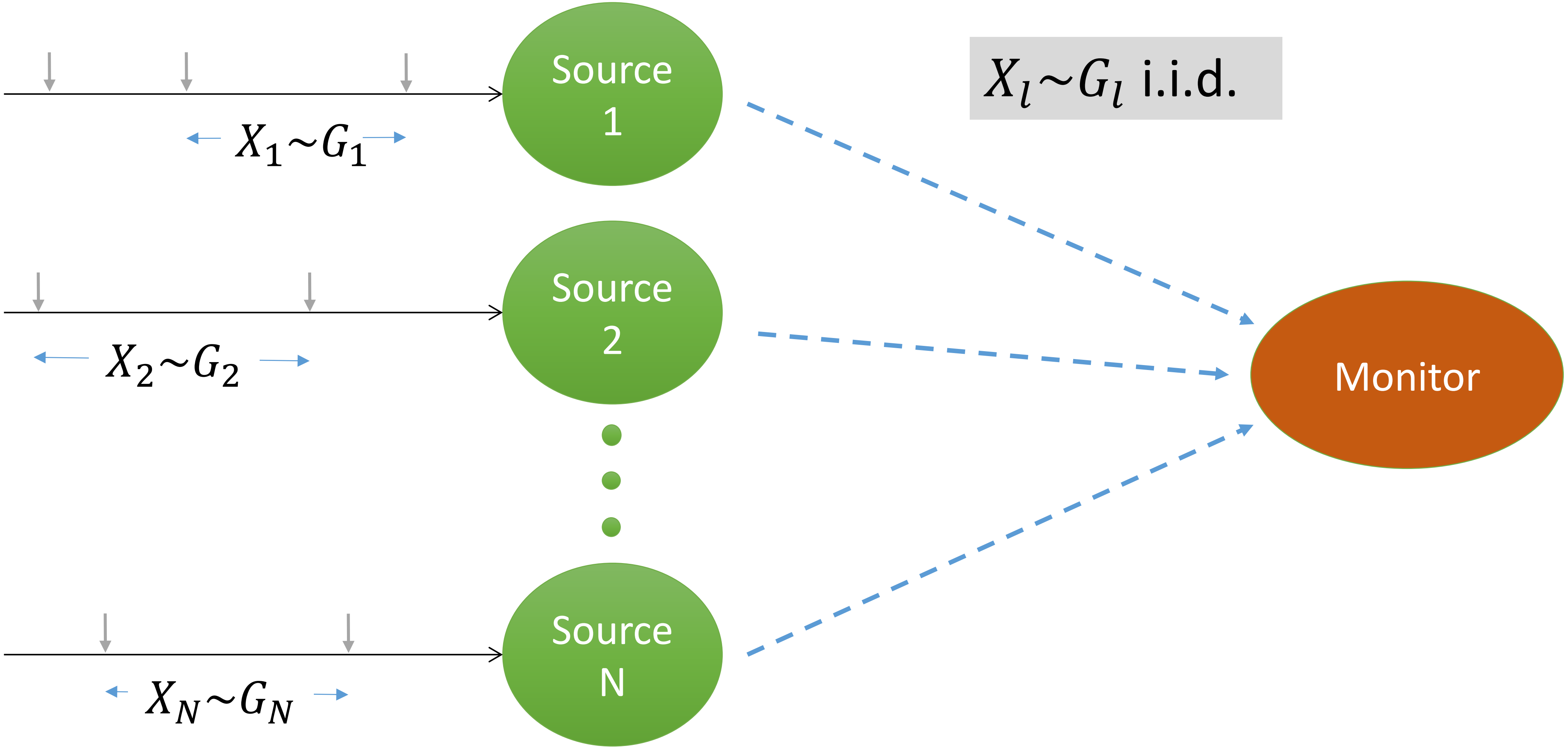
We consider the setup with multiple sources.



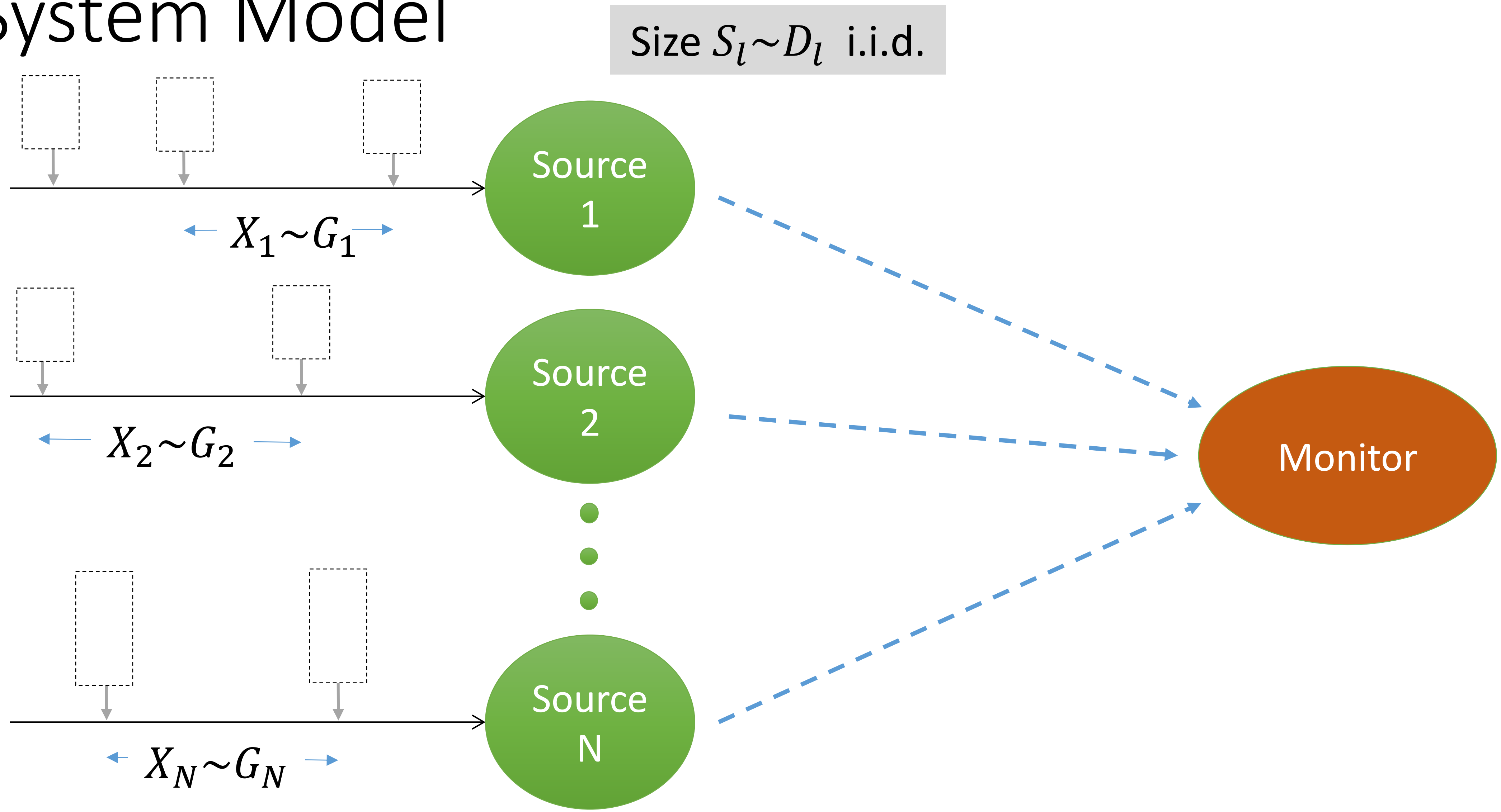
System Model

X_l =inter-generation time of updates at source l, with distribution G_l

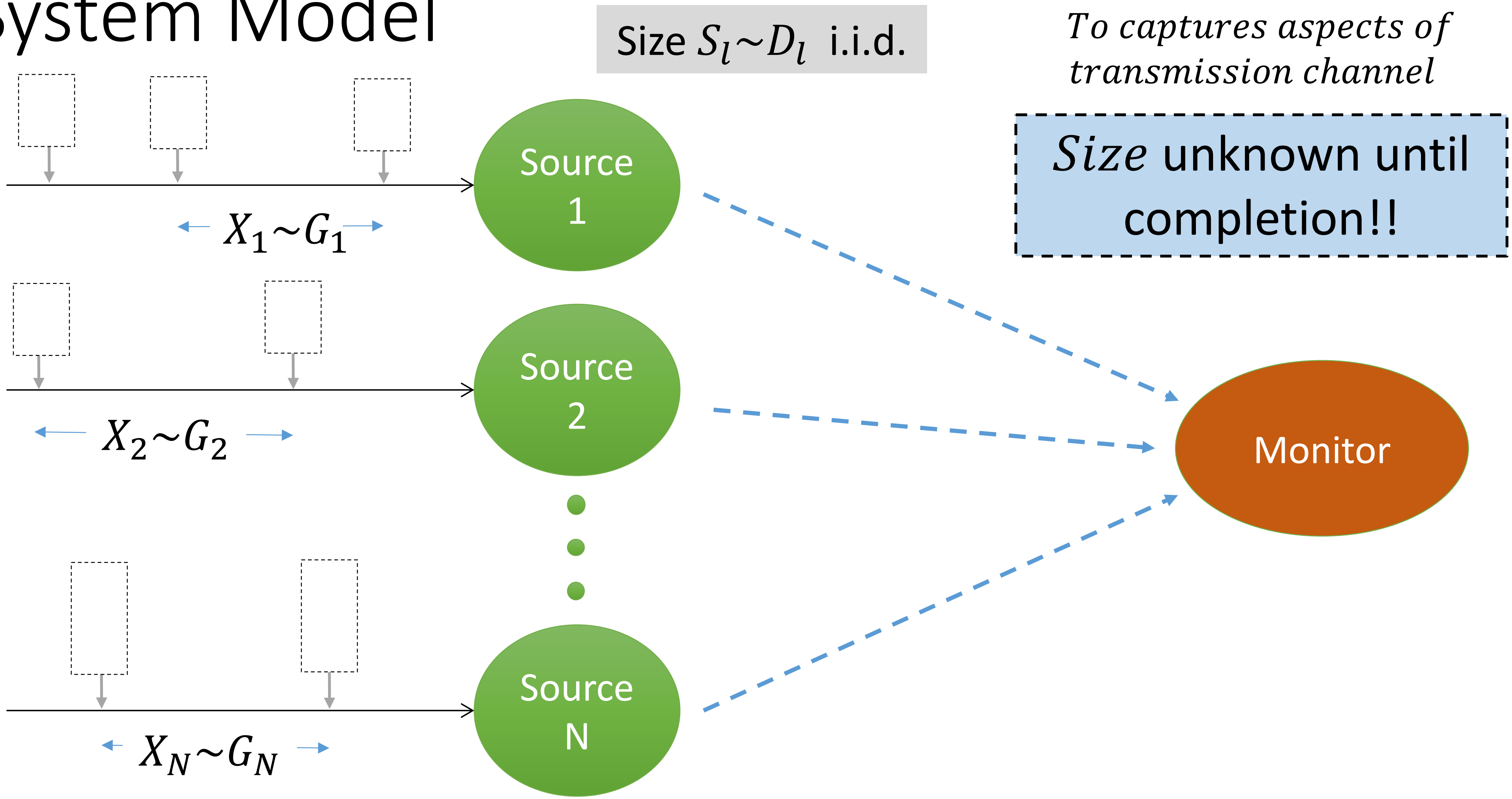
$X_l \sim G_l$ i.i.d.



System Model



System Model



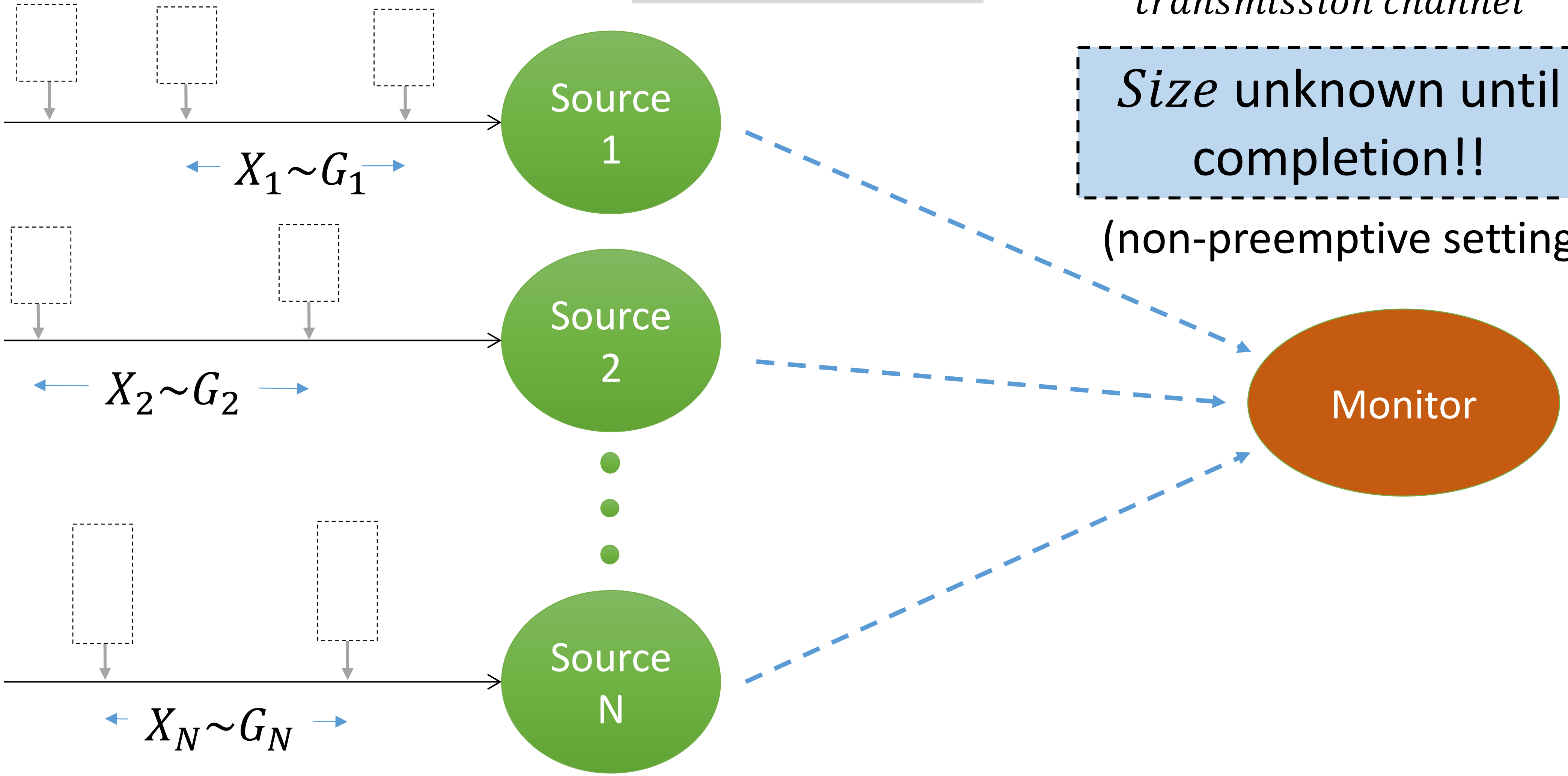
System Model

Size $S_l \sim D_l$ i.i.d.

To captures aspects of transmission channel

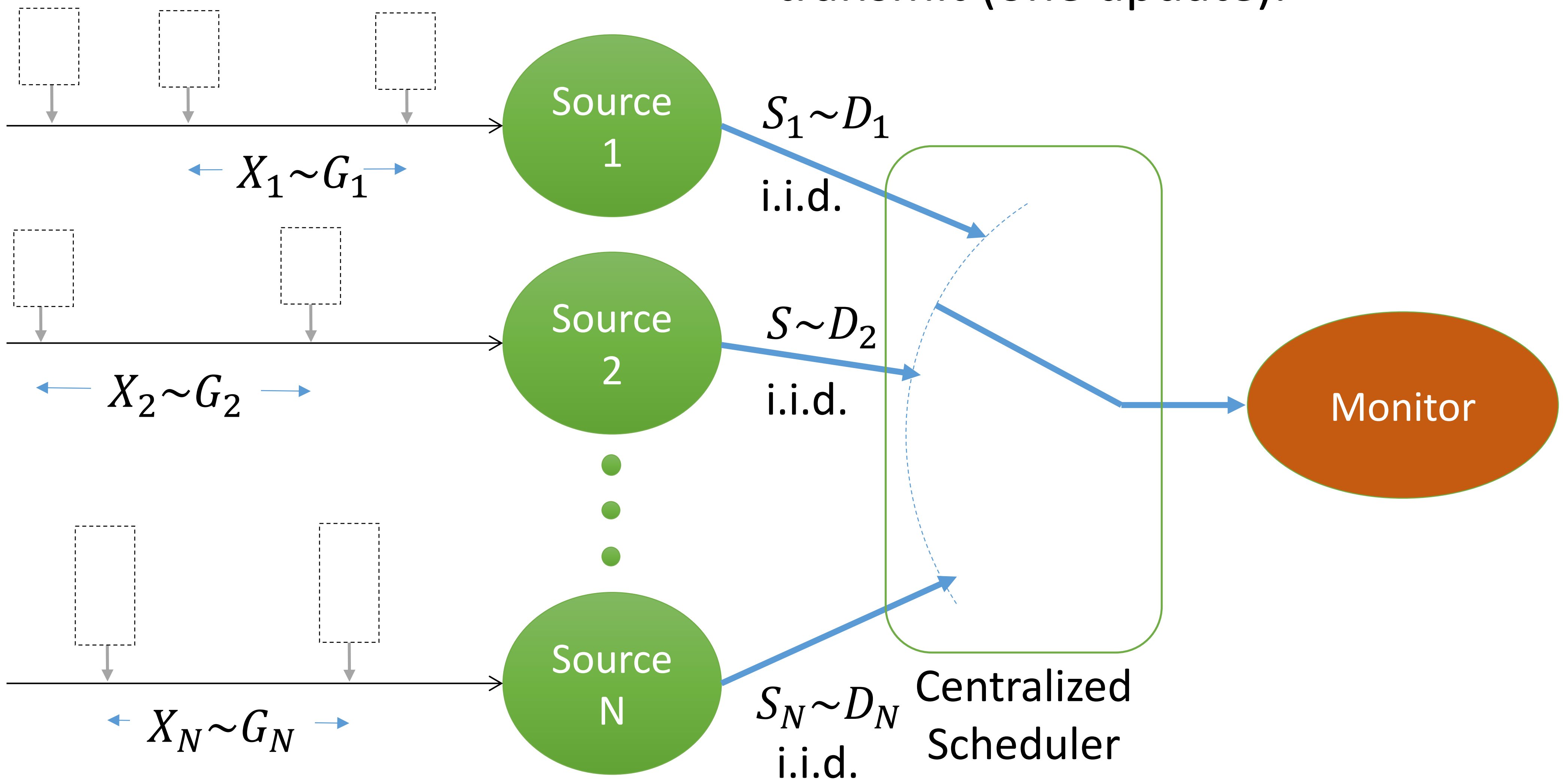
Size unknown until completion!!

(non-preemptive setting)



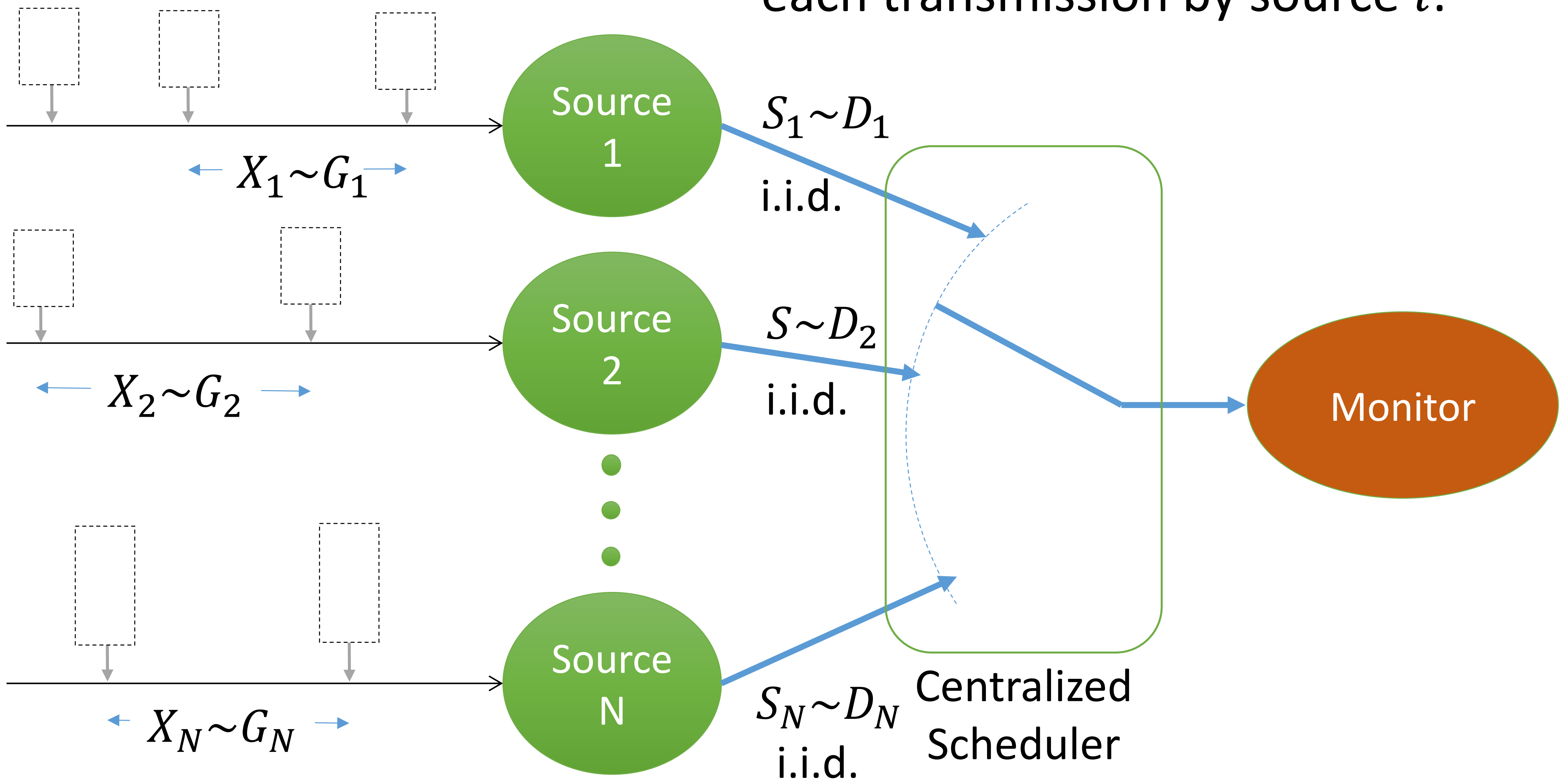
System Model

At any time, at most one source can transmit (one update).



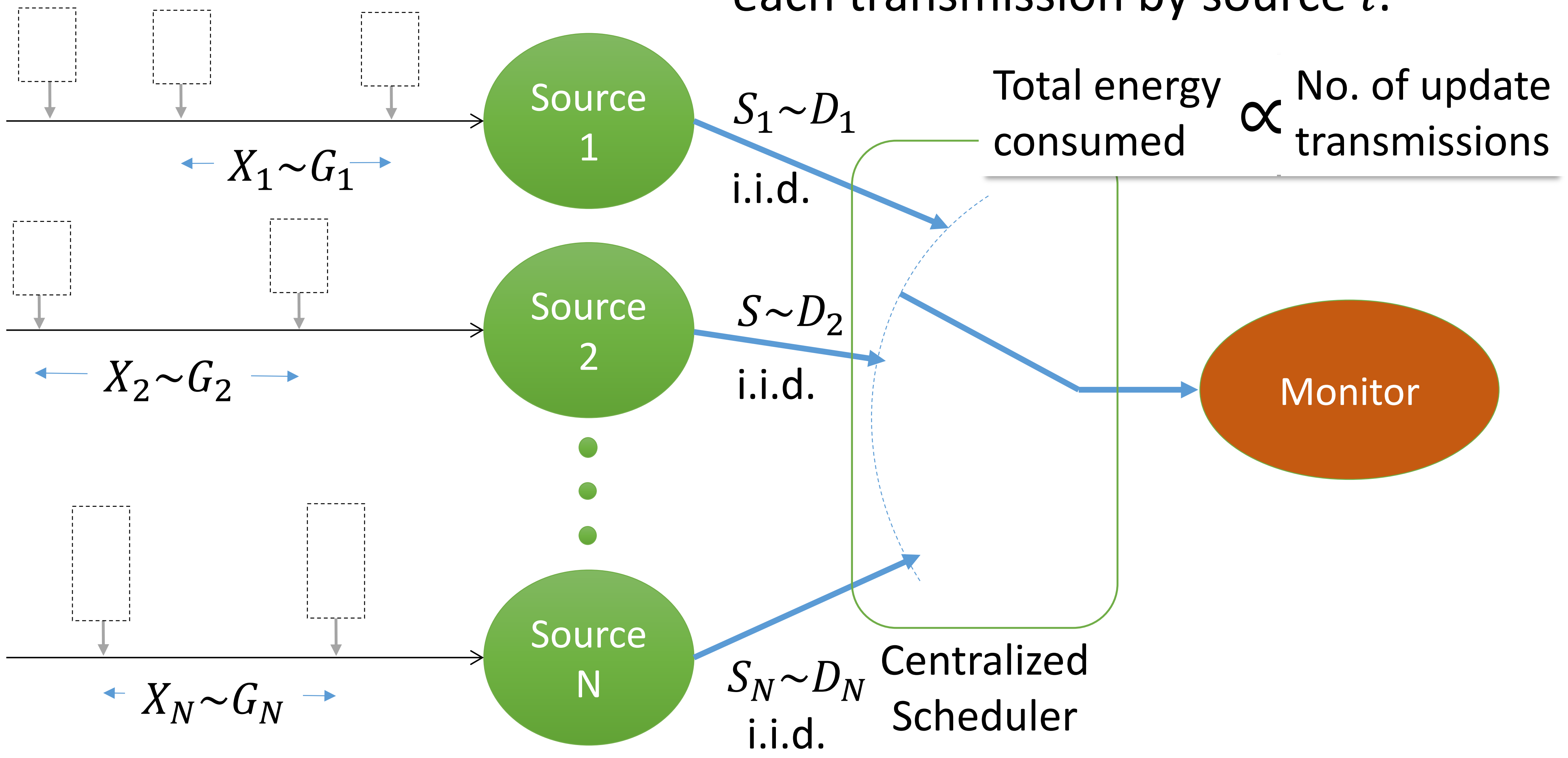
System Model

Fixed transmission (energy) cost c_l for each transmission by source l.



System Model

Fixed transmission (energy) cost c_l for each transmission by source l.



Objective

Minimize the weighted sum of the average AoI of sources, subject to a constraint on the average transmission cost.

Objective

Minimize the weighted sum of the average AoI of sources, subject to a constraint on the average transmission cost.

$$\min \sum_{l} (w_l \cdot AAoI_l)$$

$$\text{s.t.} \sum_{l} c_l \cdot R_l \leq C_{MAX}$$

Objective

Minimize the weighted sum of the average AoI of sources, subject to a constraint on the average transmission cost.

$$\min \sum_{l} (w_l \cdot AAoI_l)$$

$$\text{s.t.} \sum_{l} c_l \cdot R_l \leq C_{MAX}$$

Weightage of source l
(constant)

Objective

Minimize the weighted sum of the average AoI of sources, subject to a constraint on the average transmission cost.

$$\min \sum_{l} (w_l \cdot AAoI_l)$$

$$\text{s.t.} \sum_{l} c_l \cdot R_l \leq C_{MAX}$$

Average AoI of source l

Objective

Minimize the weighted sum of the average AoI of sources, subject to a constraint on the average transmission cost.

$$\min \sum_l (w_l \cdot AAoI_l)$$

$$\text{s.t.} \sum_l c_l \cdot R_l \leq C_{MAX}$$

Cost per transmission for source l (constant)

Objective

Minimize the weighted sum of the average AoI of sources, subject to a constraint on the average transmission cost.

$$\min \sum_{l} (w_l \cdot AAoI_l)$$

$$\text{s.t.} \sum_{l} c_l \cdot R_l \leq C_{MAX}$$

Number of updates source l transmits per unit time

Objective

Minimize the weighted sum of the average AoI of sources, subject to a constraint on the average transmission cost.

$$\min \sum_l (w_l \cdot AAoI_l)$$

$$\text{s.t.} \sum_l c_l \cdot R_l \leq C_{MAX}$$

Fixed constant

Objective

Minimize the weighted sum of the average AoI of sources, subject to a constraint on the average transmission cost.

$$\min \sum_{l} (w_l \cdot AAoI_l)$$

$$\text{s.t.} \sum_{l} c_l \cdot R_l \leq C_{MAX}$$

<u>Decision Problem:</u> At each time,

1. which source gets to transmit.
2. which update the source should transmit.

Objective

Minimize the weighted sum of the average AoI of sources, subject to a constraint on the average transmission cost.

$$\min \sum_{l} (w_l \cdot AAoI_l)$$

$$\text{s.t.} \sum_{l} c_l \cdot R_l \leq C_{MAX}$$

Only Causal Information

Decision Problem: At each time,

1. which source gets to transmit.
2. which update the source should transmit.

Prior Work

Prior Work

Single source with transmission cost:

Prior Work

Single source with transmission cost:

Sun et al. (2017)

Prior Work

Single source with transmission cost:

Sun et al. (2017) → *An new update available at all times.*

Prior Work

Single source with transmission cost:

Sun et al. (2017) → *An new update available at all times.* → Derived an optimal causal policy.

Prior Work

Single source with transmission cost:

Sun et al. (2017) → *An new update available at all times.* → Derived an optimal causal policy.

Multiple sources but without transmission cost:

Prior Work

Single source with transmission cost:

Sun et al. (2017) → *An new update available at all times.* → Derived an optimal causal policy.

Multiple sources but without transmission cost:

Kadota & Modiano (2019)

Prior Work

Single source with transmission cost:

Sun et al. (2017) → *An new update available at all times.* → Derived an optimal causal policy.

Multiple sources but without transmission cost:

Kadota & Modiano (2019) → *Discrete-time setting with geometric distributions.*

Prior Work

Single source with transmission cost:

Sun et al. (2017) → *An new update available at all times.* → Derived an optimal causal policy.

Multiple sources but without transmission cost:

Kadota & Modiano (2019) → *Discrete-time setting with geometric distributions.* → Derived causal policies with competitive ratio at most 4.

In this Work

For Continuous-Time setting with

- General distributions $G_i's$ *and* $D_i's$
- *Non-negative transmission cost.*

In this Work

For Continuous-Time setting with
- General distributions $G_i's$ *and* $D_i's$
- *Non-negative transmission cost.*

We propose a randomized scheduling policy.

In this Work

For Continuous-Time setting with
- General distributions $G_l's$ *and* $D_l's$
- *Non-negative transmission cost.*

We propose a randomized scheduling policy.

Competitive ratio:

$$CR \leq 3 + \max_l \frac{Variance(G_l)}{Mean^2(G_l)}$$

G_l = update inter-generation time distribution for source l.

In this Work

For Continuous-Time setting with

- General distributions $G_l's$ *and* $D_l's$
- *Non-negative transmission cost.*

We propose a randomized scheduling policy.

Competitive ratio:

$$CR \leq 3 + \max_l \frac{Variance(G_l)}{Mean^2(G_l)}$$

Independent of size distributions D_l.

G_l = update inter-generation time distribution for source l.

In this Work

For Continuous-Time setting with
- General distributions G_l's and D_l's
- *Non-negative transmission cost.*

We propose a randomized scheduling policy.

Competitive ratio:

$$CR \leq 3 + \max_{l} \frac{Variance(G_l)}{Mean^2(G_l)}$$

Independent of size distributions D_l.

At most 1 for common distributions like exponential, uniform and Rayleigh.

G_l = update inter-generation time distribution for source l.

In this Work

For Continuous-Time setting with
- General distributions $G_l's$ *and* $D_l's$
- *Non-negative transmission cost.*

We propose a randomized scheduling policy.

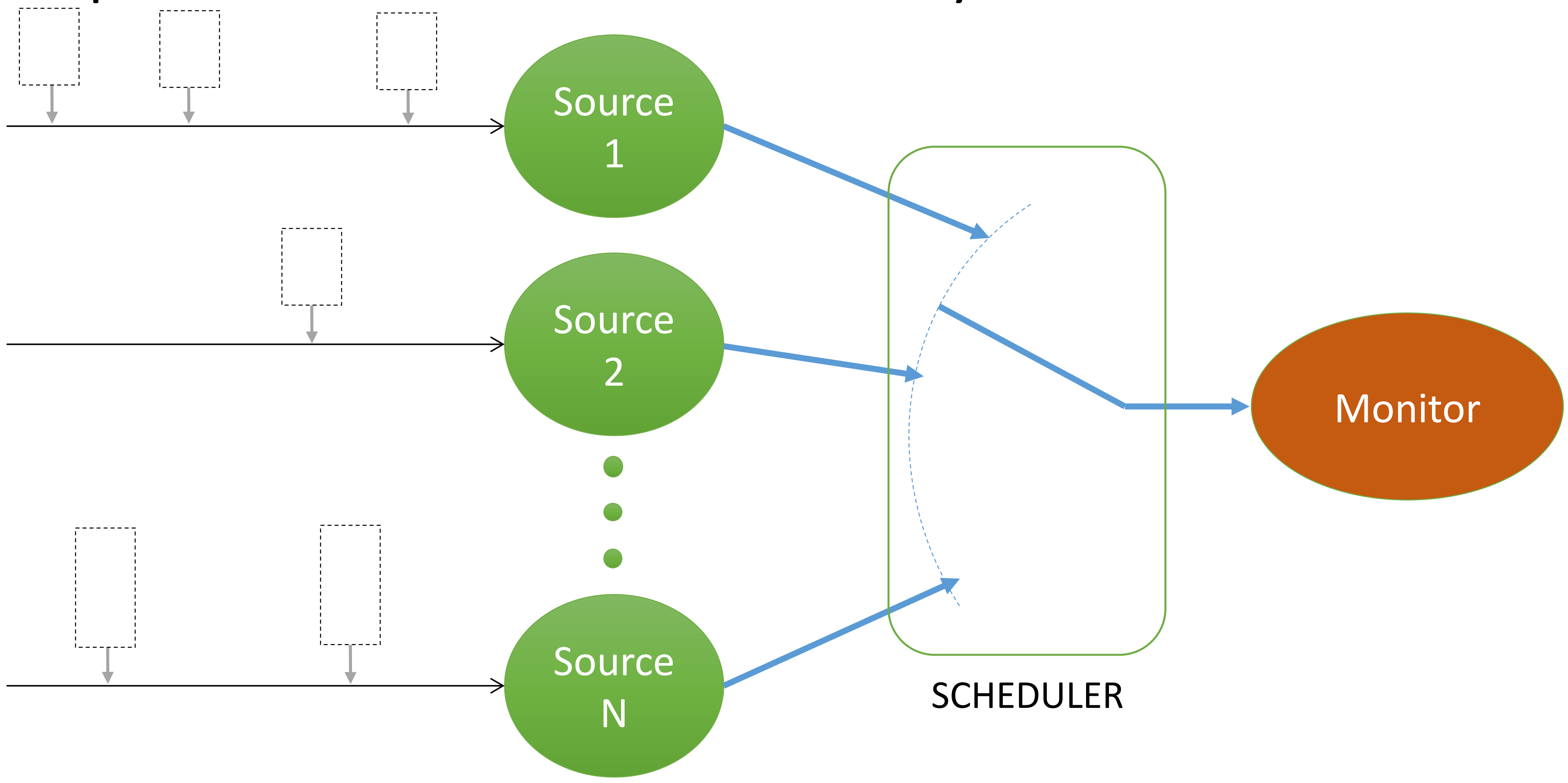
<u>Competitive ratio:</u>

$$CR \leq 3 + \max_{l} \frac{Variance(G_l)}{Mean^2(G_l)}$$

Analysis is tight for the considered policy (dependence of its CR on $\max_{l} \frac{Variance(G_l)}{Mean^2(G_l)}$ is unavoidable).

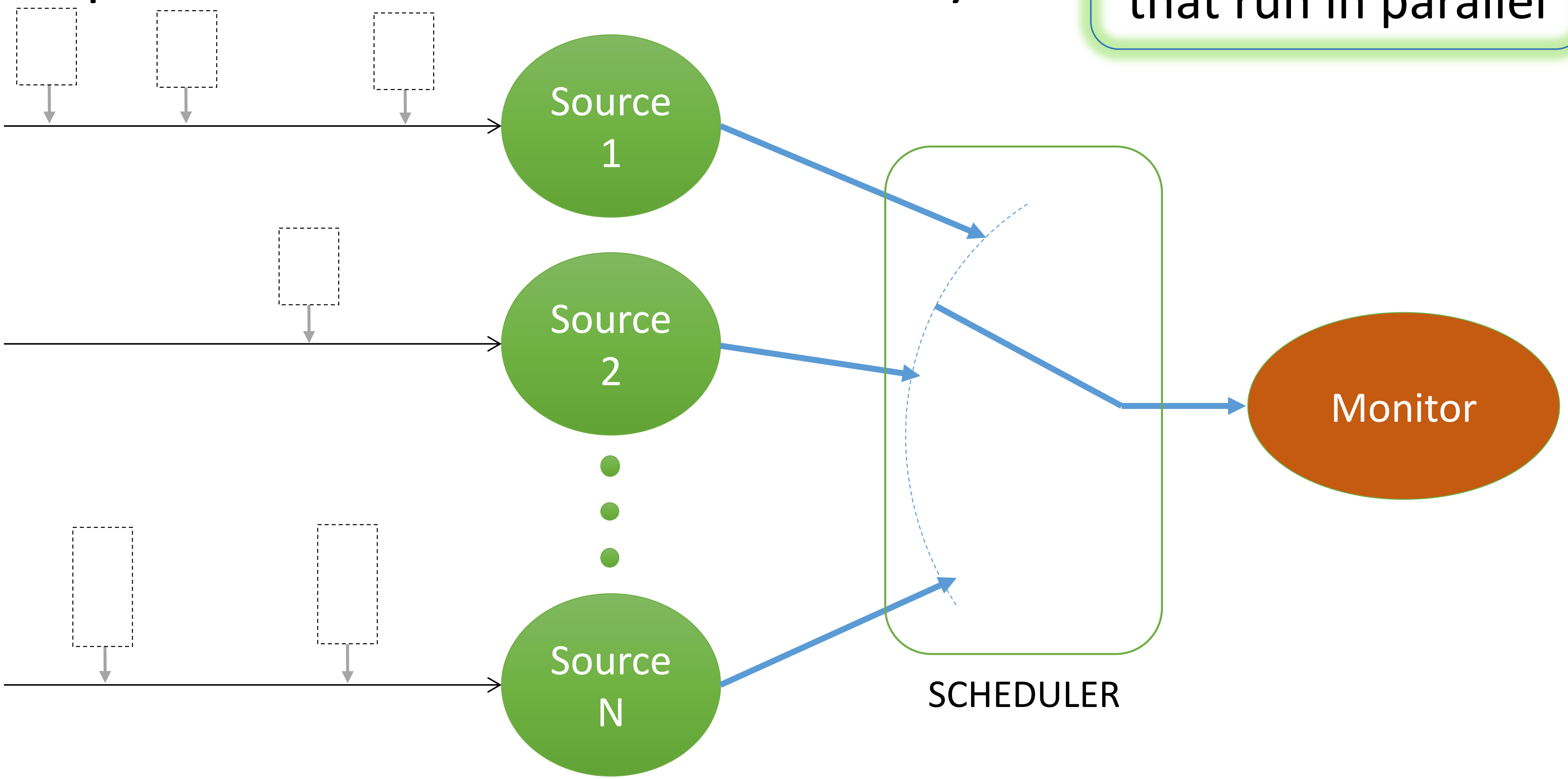
G_l = update inter-generation time distribution for source l.

Proposed Randomized Policy



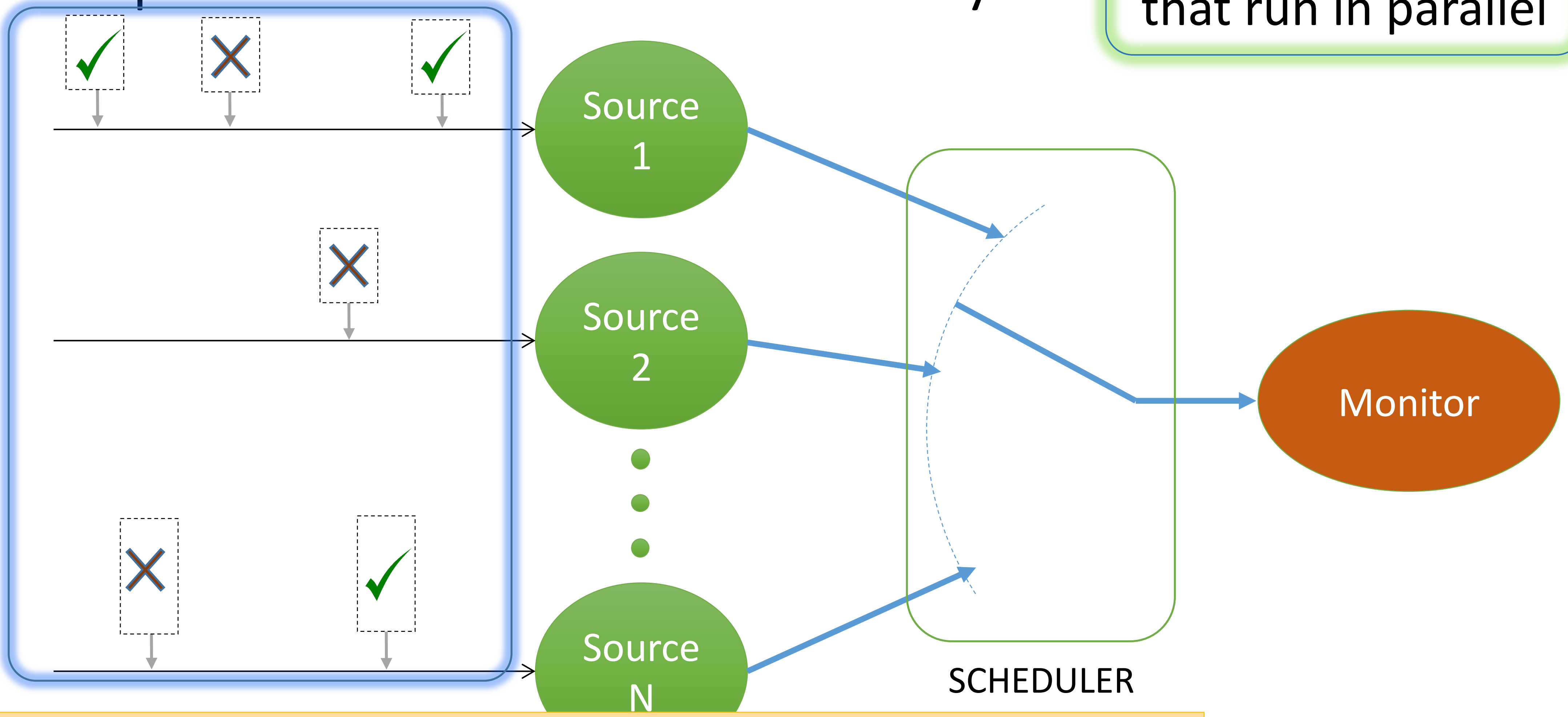
Proposed Randomized Policy

Two subroutines that run in parallel



Proposed Randomized Policy

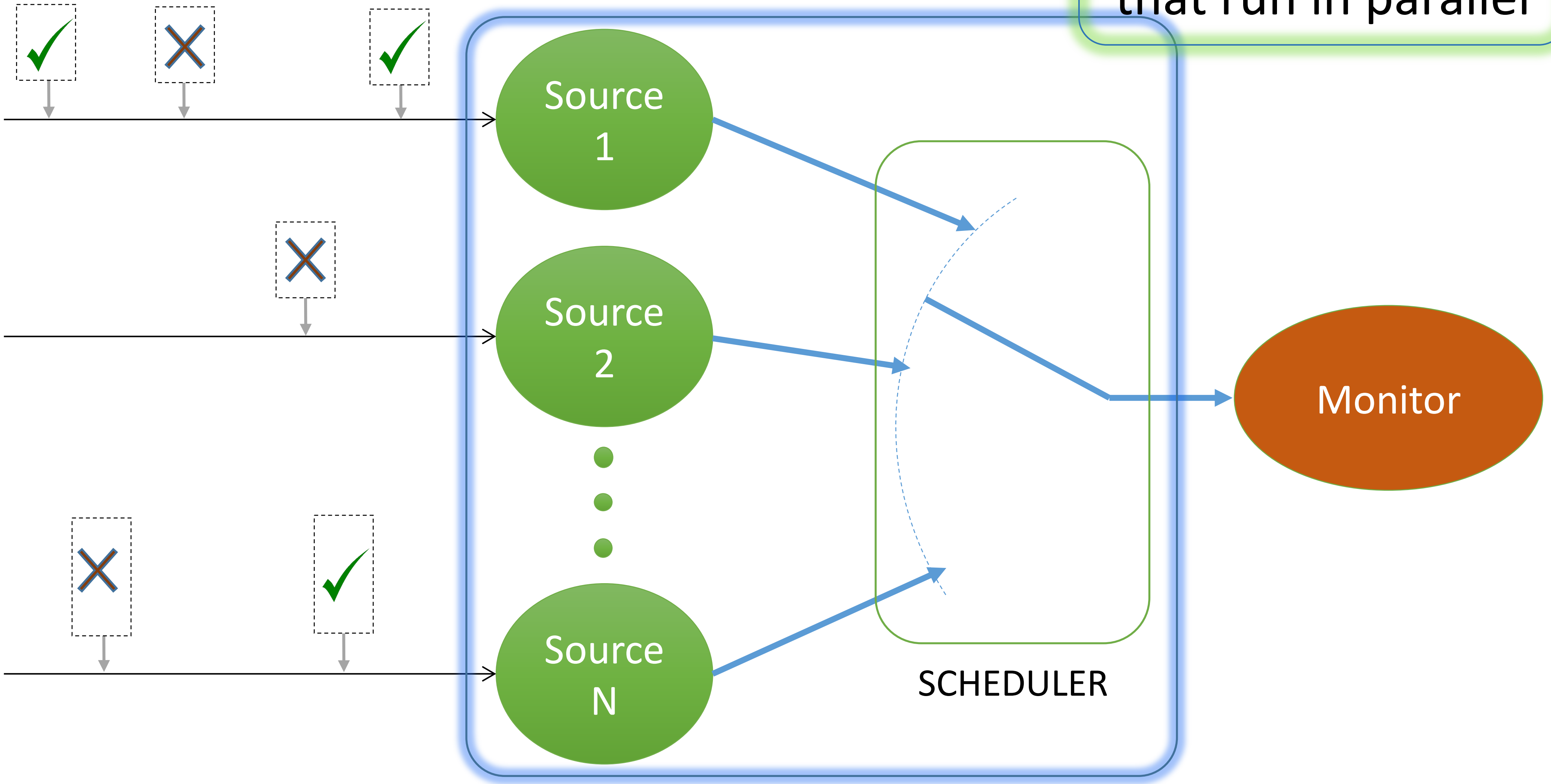
Two subroutines that run in parallel



1) Update Selection
(which update is worth transmitting given the transmission cost)

Proposed Randomized Policy

Two subroutines that run in parallel



2) Source Selection (which source gets to transmit at any time)

Proposed Randomized Policy

First Subroutine

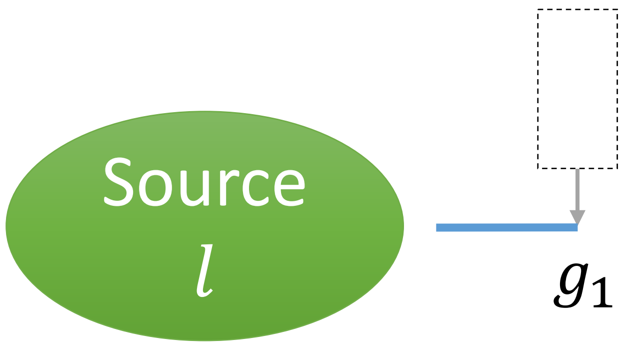
1) <u>Update Selection at Sources</u>

Proposed Randomized Policy

First Subroutine

1) Update Selection at Sources

- At each source l, whenever an update is generated,

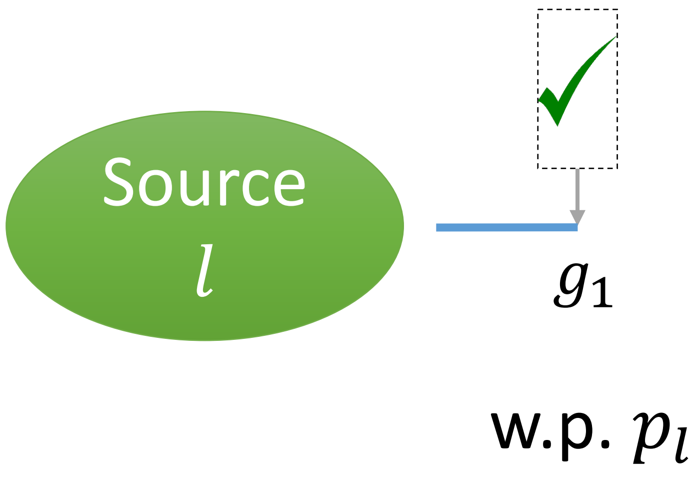


Proposed Randomized Policy

First Subroutine

1) Update Selection at Sources

- At each source l, whenever an update is generated,
  - Mark the generated update with fixed probability p_l (independently).



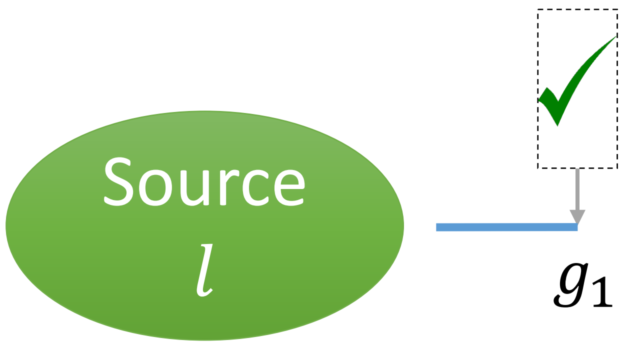
Proposed Randomized Policy

First Subroutine

1) *Update Selection at Sources*

- At each source l, whenever an update is generated,
  - Mark the generated update with fixed probability p_l (independently).

Found by *min* an upper bound on the cost!!



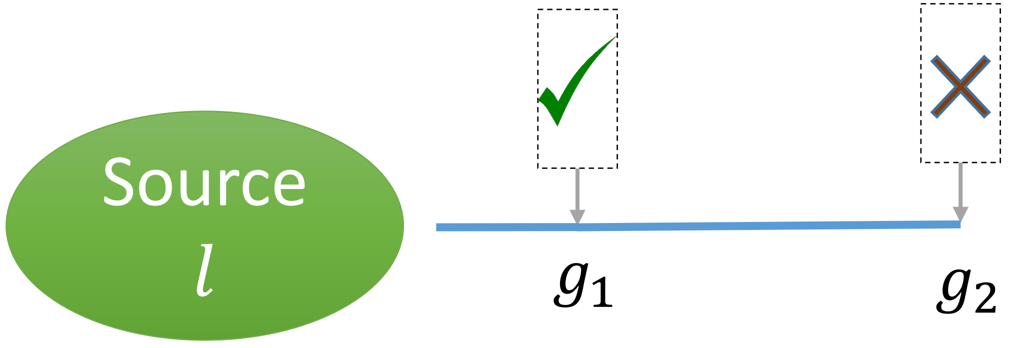
w.p. p_l

Proposed Randomized Policy

First Subroutine

1) Update Selection at Sources

- At each source l, whenever an update is generated,
  - Mark the generated update with fixed probability p_l (independently).
  - Updates that are not marked are discarded.



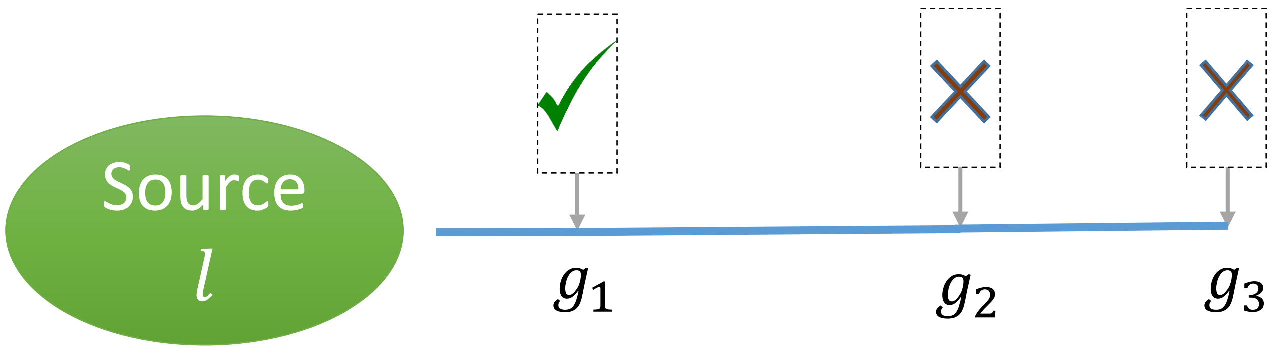
w.p. $(1 - p_l)$

Proposed Randomized Policy

First Subroutine

1) Update Selection at Sources

- At each source l, whenever an update is generated,
  - Mark the generated update with fixed probability p_l (independently).
  - Updates that are not marked are discarded.



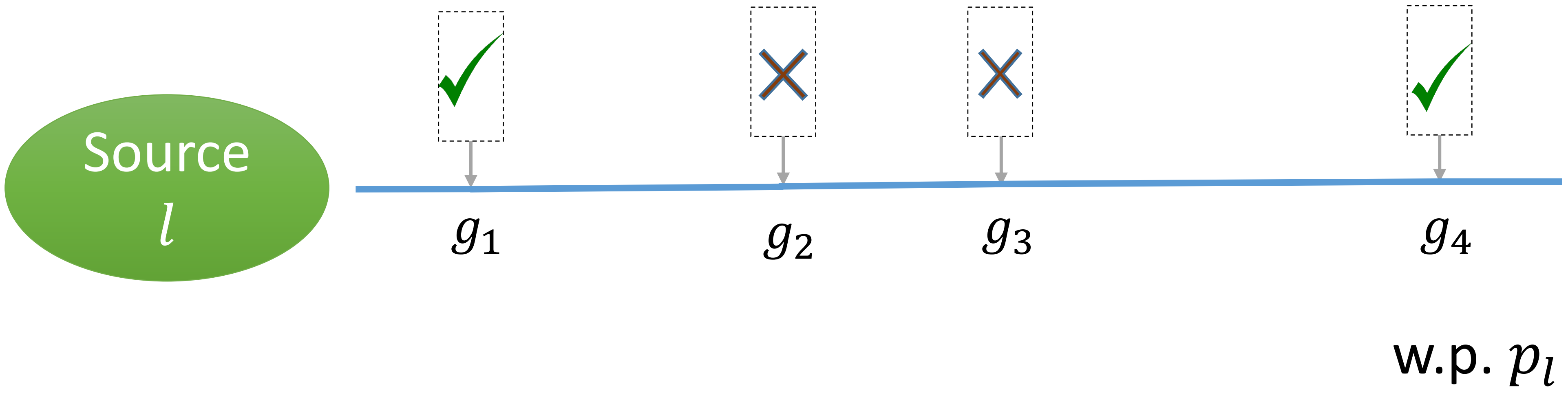
w.p. $(1 - p_l)$

Proposed Randomized Policy

First Subroutine

1) Update Selection at Sources

- At each source l, whenever an update is generated,
  - Mark the generated update with fixed probability p_l (independently).
  - Updates that are not marked are discarded.

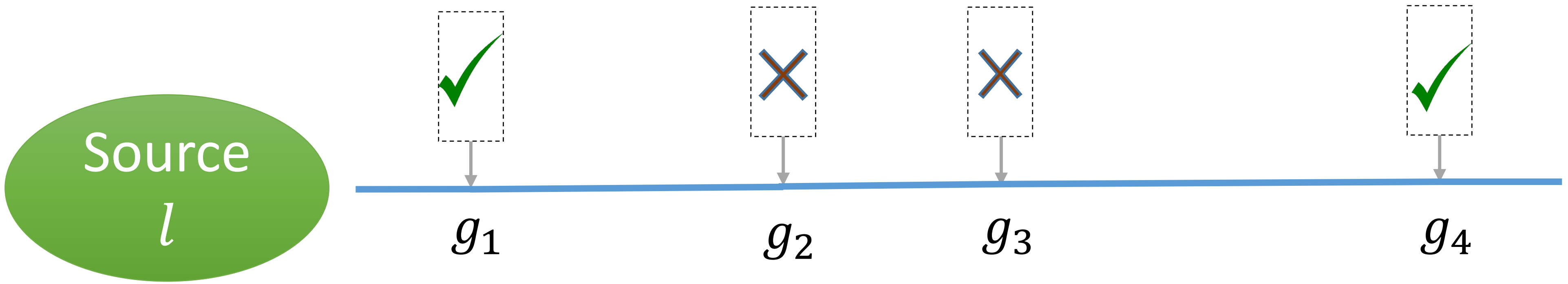


Proposed Randomized Policy

First Subroutine

1) Update Selection at Sources

- At each source l, whenever an update is generated,
  - Mark the generated update with fixed probability p_l (independently).
  - Updates that are not marked are discarded.
- When source l is selected to transmit,

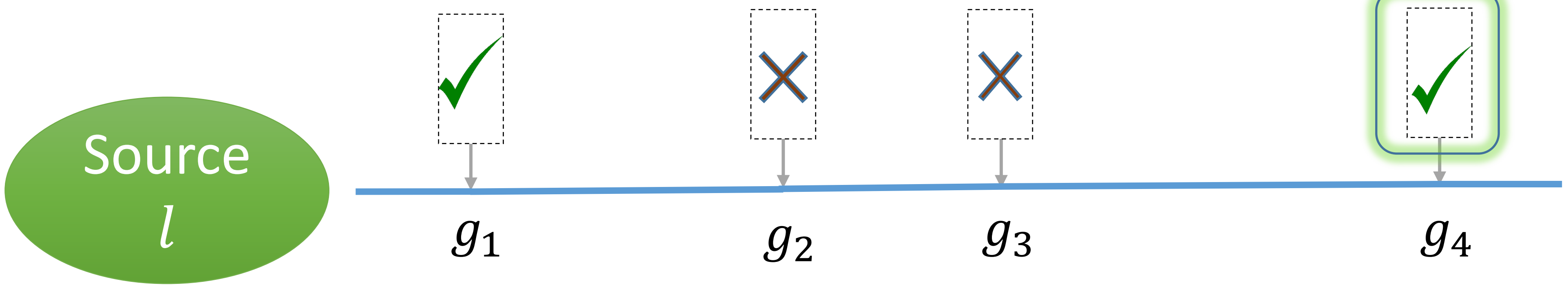


Proposed Randomized Policy

First Subroutine

1) Update Selection at Sources

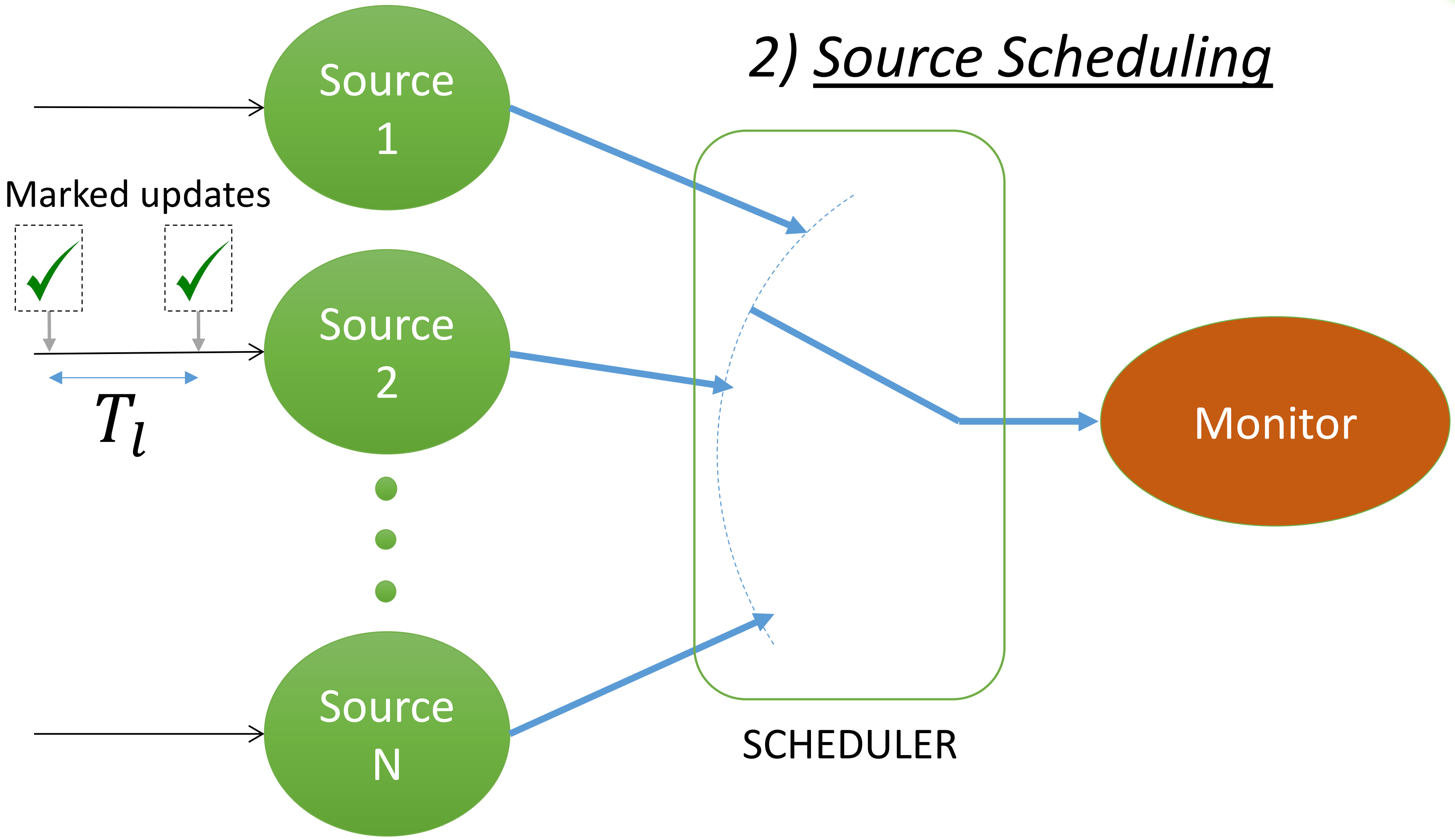
- At each source l, whenever an update is generated,
  - Mark the generated update with fixed probability p_l (independently).
  - Updates that are not marked are discarded.
- When source l is selected to transmit,
  - Immediately transmit source's latest marked update (if any).



Proposed Randomized Policy

Second Subroutine

2) *Source Scheduling*

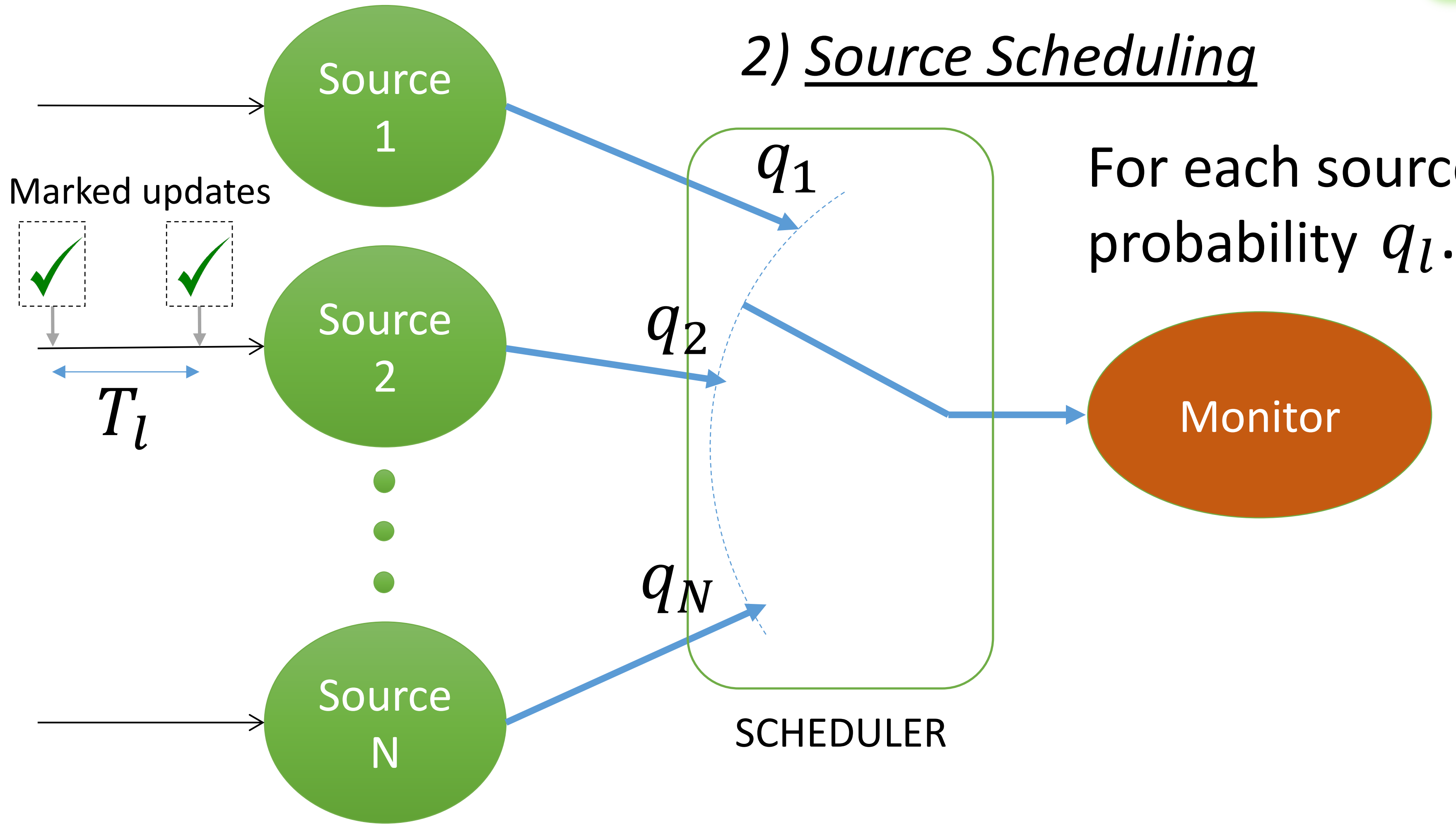


Proposed Randomized Policy

Second Subroutine

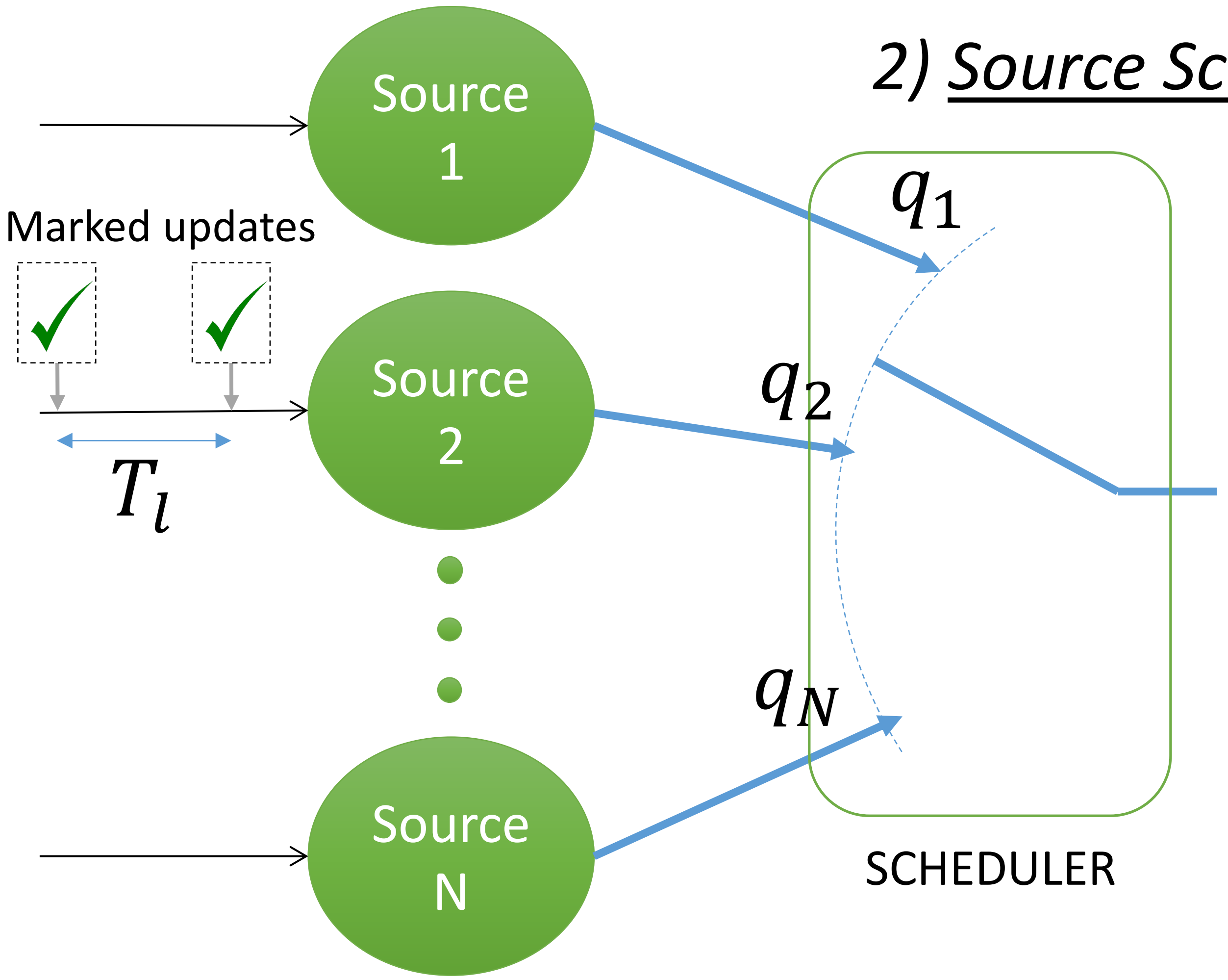
2) *Source Scheduling*

For each source l, we define probability q_l.



Proposed Randomized Policy

Second Subroutine



2) *Source Scheduling*

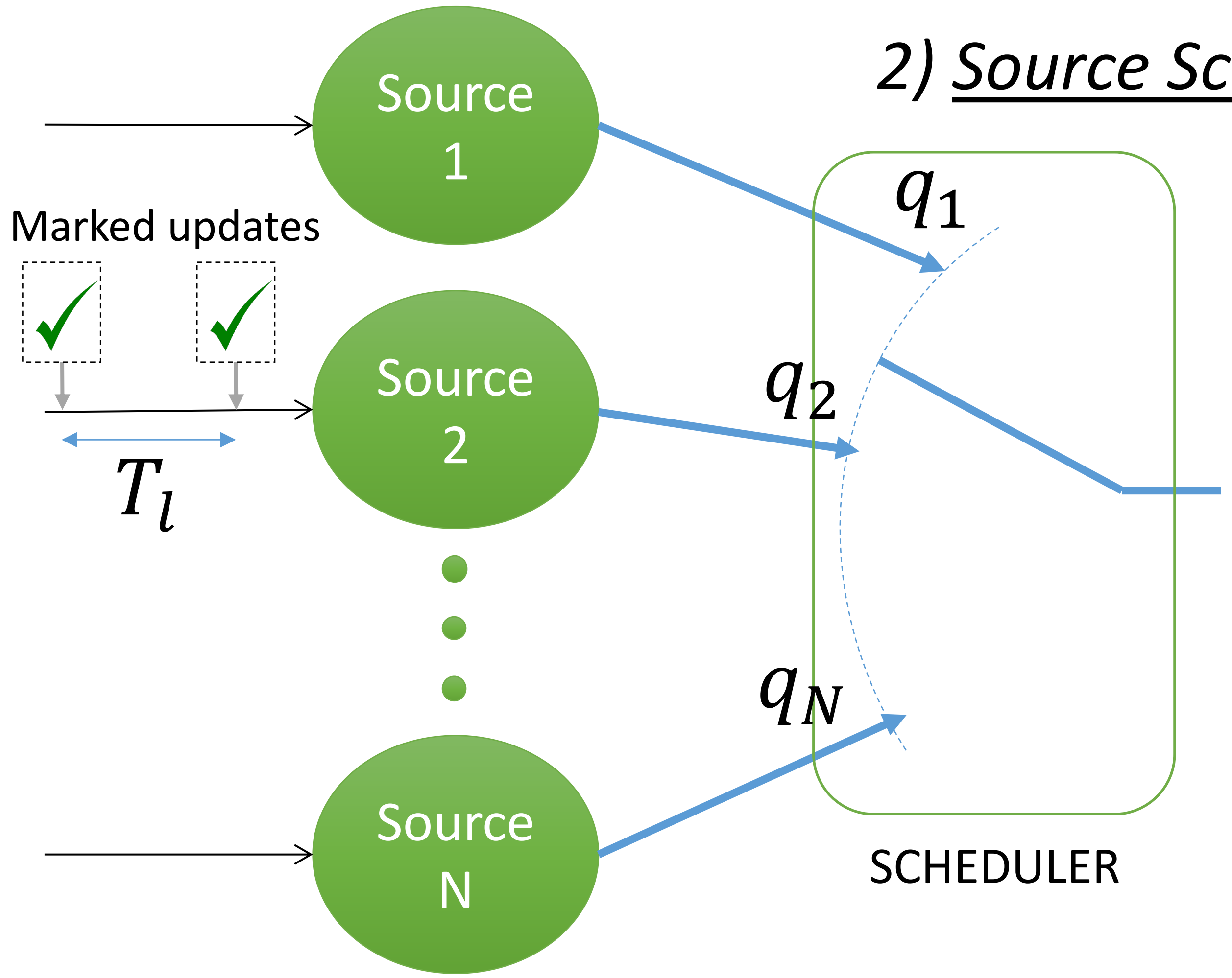
For each source l, we define probability q_l.

- $q_l \propto \frac{1}{E[T_l]}$

Mean inter-generation time of marked updates at source l.

Proposed Randomized Policy

Second Subroutine



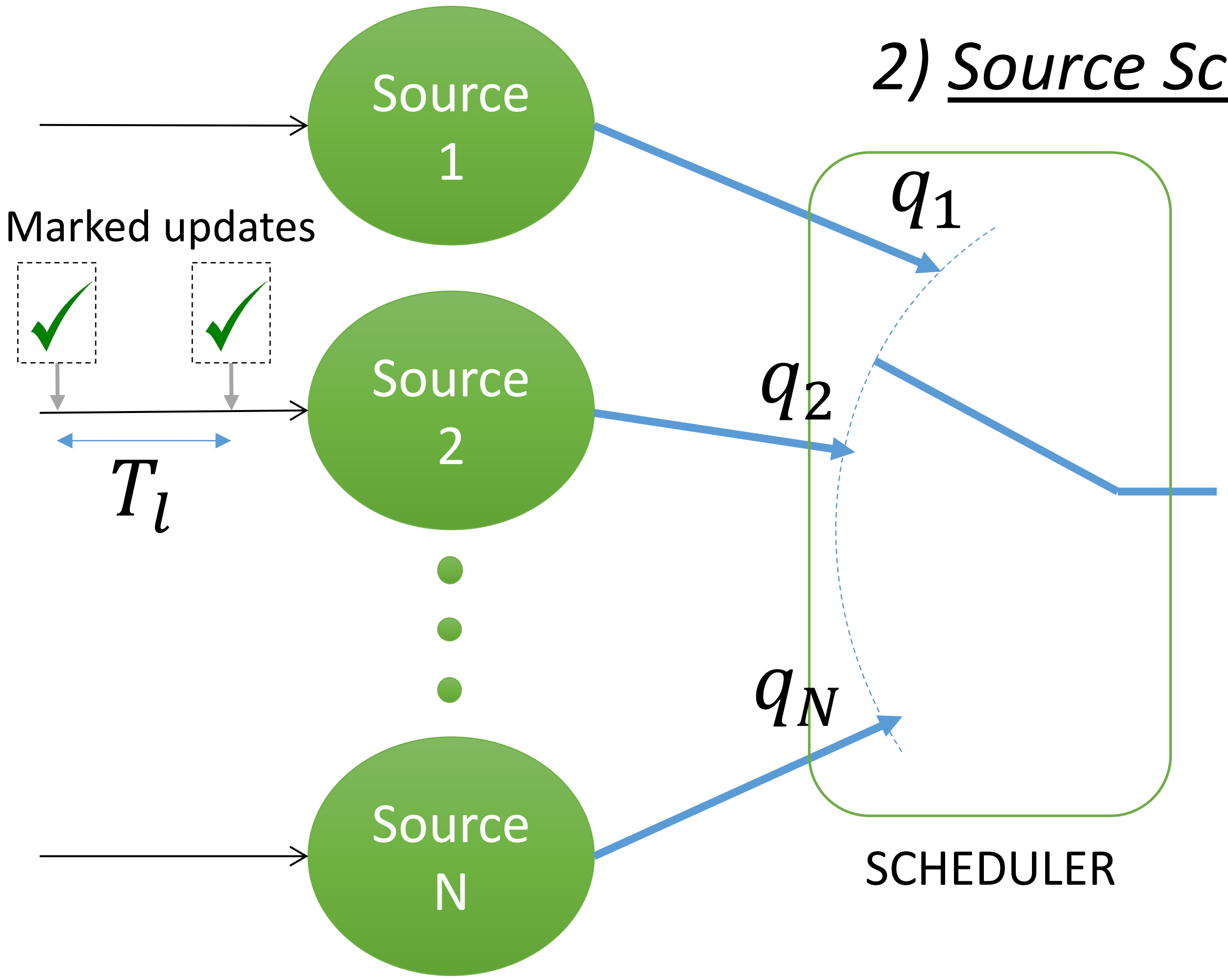
2) *Source Scheduling*

For each source l, we define probability q_l.

- $q_l \propto \frac{1}{E[T_l]} = \frac{p_l}{Mean(G_l)}$

Proposed Randomized Policy

Second Subroutine



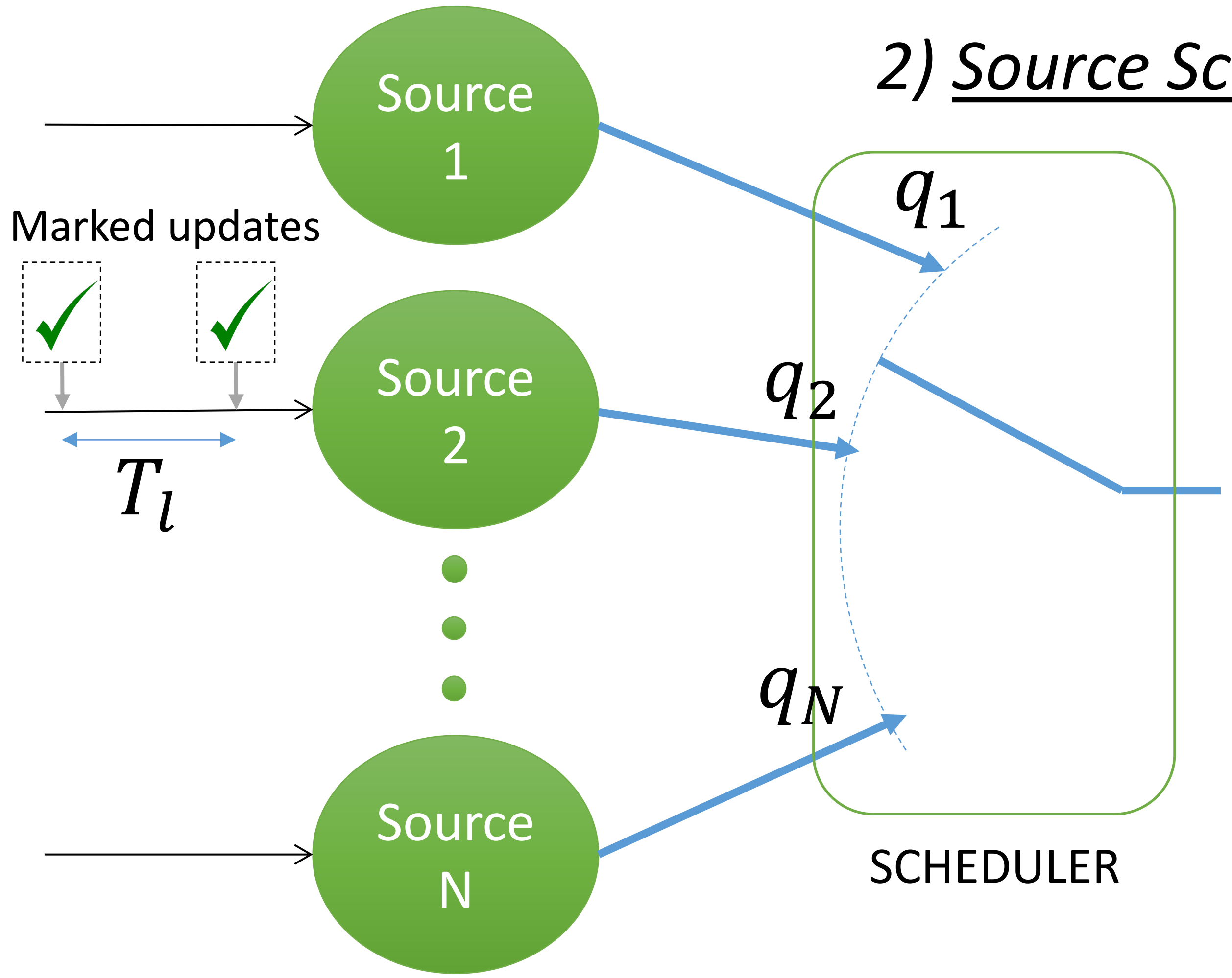
2) *Source Scheduling*

For each source l, we define probability q_l.

- $q_l \propto \frac{1}{E[T_l]} = \frac{p_l}{Mean(G_l)}$
- Normalized such that $\sum_l q_l = 1$.

Proposed Randomized Policy

Second Subroutine



2) Source Scheduling

For each source l, we define probability q_l.

- $q_l \propto \frac{1}{E[T_l]} = \frac{p_l}{Mean(G_l)}$
- Normalized such that $\sum_l q_l = 1$.

Among all sources, schedule source l with probability q_l.

Intuition

Update Selection

Source Scheduling

Intuition

Update Selection

Marks an update eligible for transmission at source l with prob p_l.

Source Scheduling

Intuition

Update Selection

Marks an update eligible for transmission at source l with prob p_l.

p_l's obtained by minimizing an upper bound on the AoI cost.

Source Scheduling

Intuition

Update Selection

Marks an update eligible for transmission at source l with prob p_l.

p_l's obtained by minimizing an upper bound on the AoI cost.

p_l depends on all sources.

Source Scheduling

Intuition

Critical to manage transmission cost.

Update Selection

Marks an update eligible for transmission at source l with prob p_l.

p_l's obtained by minimizing an upper bound on the AoI cost.

p_l depends on all sources.

Source Scheduling

Intuition

Critical to manage transmission cost.

Update Selection

Marks an update eligible for transmission at source l with prob p_l.

p_l's obtained by minimizing an upper bound on the AoI cost.

p_l depends on all sources.

Source Scheduling

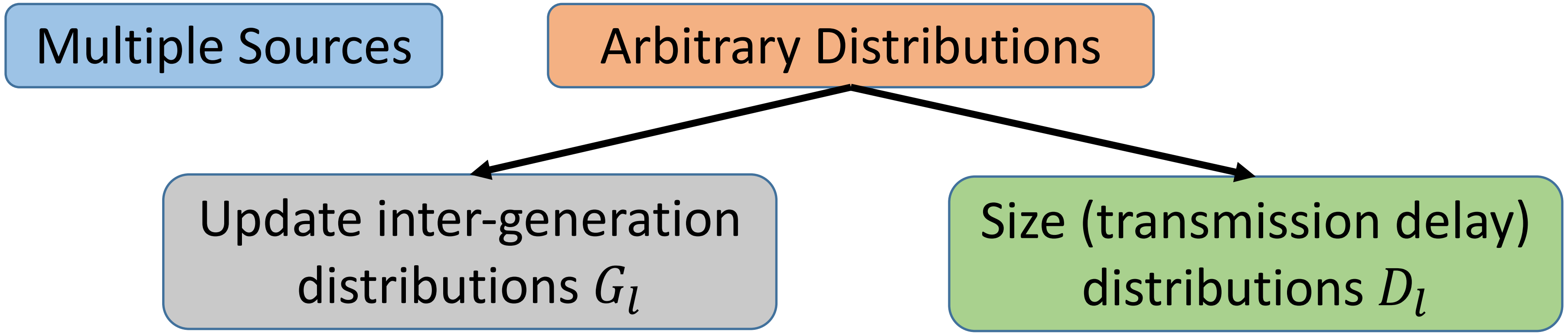
Controls the frequency at which a source transmits its marked updates.

Quick Recap

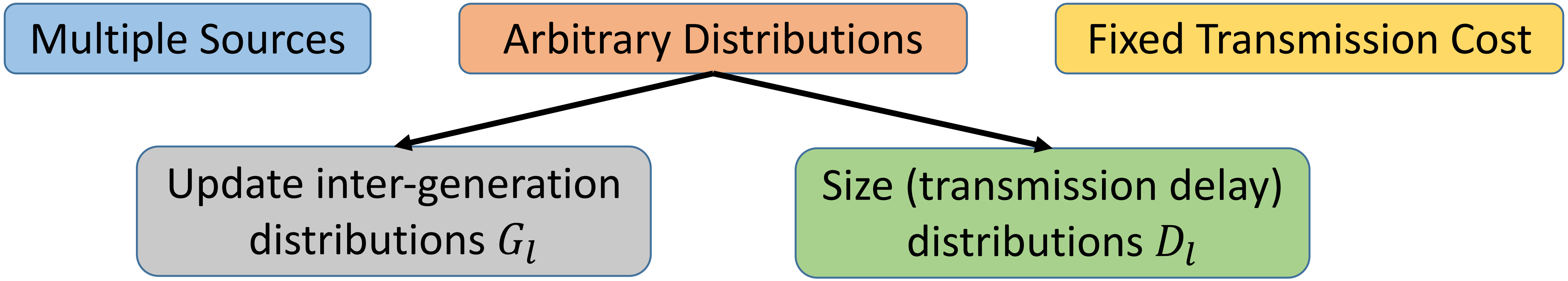
Quick Recap

Multiple Sources

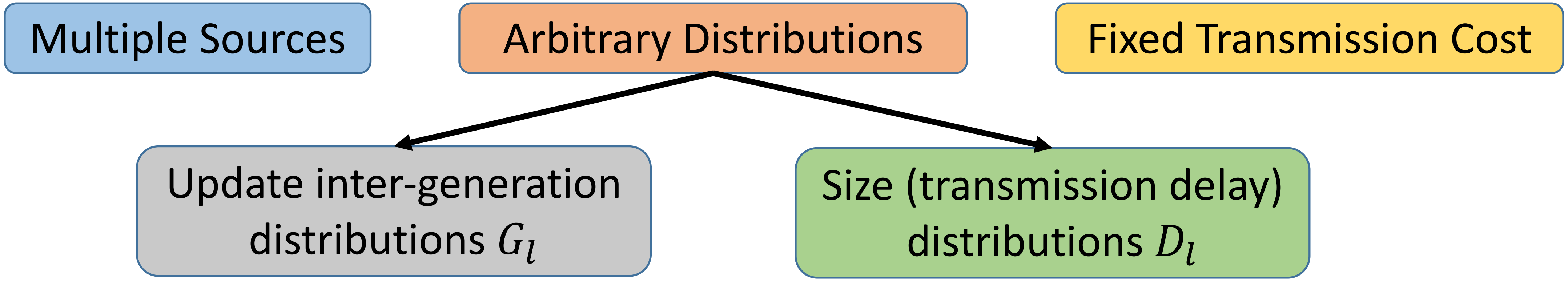
Quick Recap


Multiple Sources
Arbitrary Distributions
Update inter-generation distributions G_l
Size (transmission delay) distributions D_l

Quick Recap


Multiple Sources
Arbitrary Distributions
Fixed Transmission Cost
Update inter-generation distributions G_l
Size (transmission delay) distributions D_l

Quick Recap



- Proposed a novel two-stage randomized policy with competitive ratio (CR)

Quick Recap

Multiple Sources

Arbitrary Distributions

Fixed Transmission Cost

Update inter-generation distributions G_l

Size (transmission delay) distributions D_l

- Proposed a novel two-stage randomized policy with competitive ratio (CR)

Independent of size distributions D_l.

Quick Recap

Multiple Sources

Arbitrary Distributions

Fixed Transmission Cost

- Update inter-generation distributions G_l
- Size (transmission delay) distributions D_l

- Proposed a novel two-stage randomized policy with competitive ratio (CR)

Independent of size distributions D_l.

At most 4 for common distributions G_l like exponential, uniform and Rayleigh.

Quick Recap

Multiple Sources

Arbitrary Distributions

Fixed Transmission Cost

Update inter-generation distributions G_l

Size (transmission delay) distributions D_l

- Proposed a novel two-stage randomized policy with competitive ratio (CR)

Independent of size distributions D_l.

At most 4 for common distributions G_l like exponential, uniform and Rayleigh.

What’s the catch?

Quick Recap

Multiple Sources

Arbitrary Distributions

Fixed Transmission Cost

Update inter-generation distributions G_l

Size (transmission delay) distributions D_l

- Proposed a novel two-stage randomized policy with competitive ratio (CR)

Independent of size distributions D_l.

At most 4 for common distributions G_l like exponential, uniform and Rayleigh.

What's the catch?

Unbounded CR if $Variance(G_l) \gg Mean^2(G_l)$

Quick Recap

Multiple Sources

Arbitrary Distributions

Fixed Transmission Cost

Update inter-generation distributions G_l

Size (transmission delay) distributions D_l

- Proposed a novel two-stage randomized policy with competitive ratio (CR)

Independent of size distributions D_l.

At most 4 for common distributions G_l like exponential, uniform and Rayleigh.

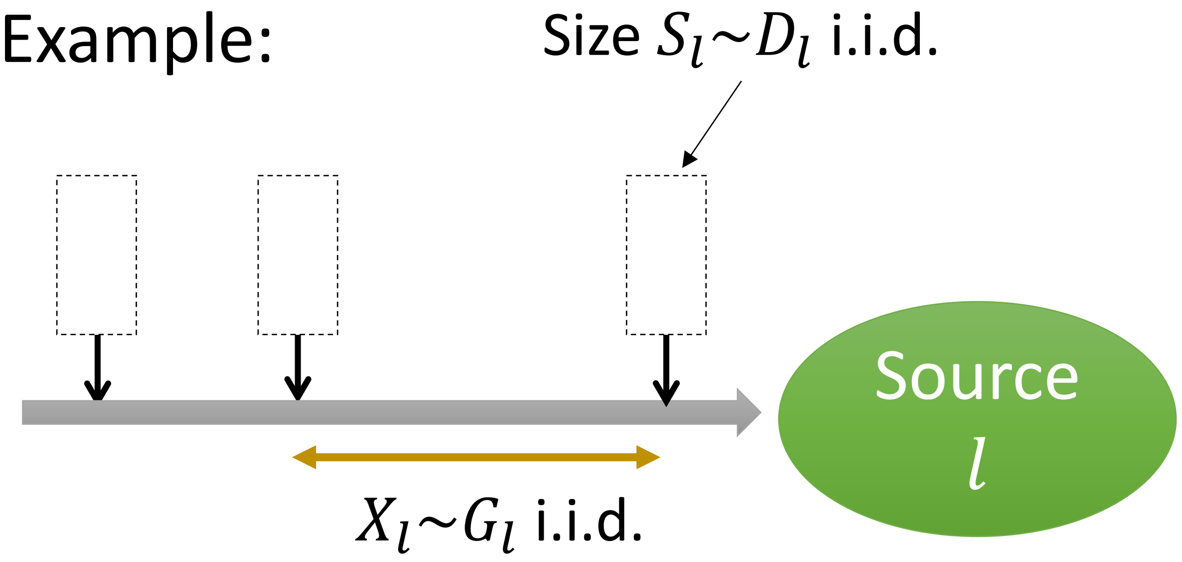
What's the catch?

Because of non-preemptive assumption

Unbounded CR if $Variance(G_l) \gg Mean^2(G_l)$

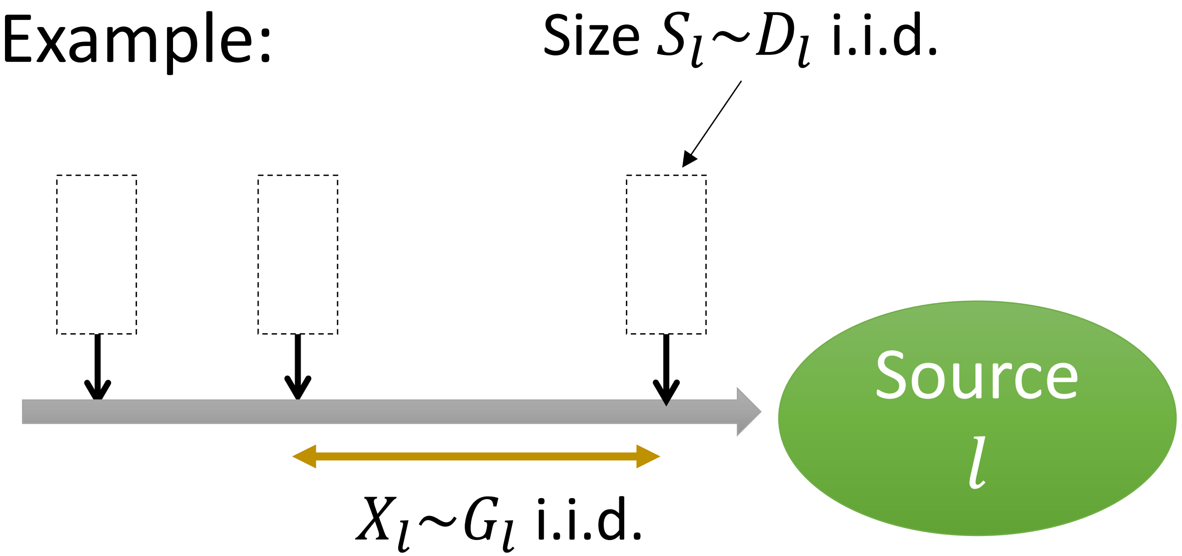
CR dependent on G_l

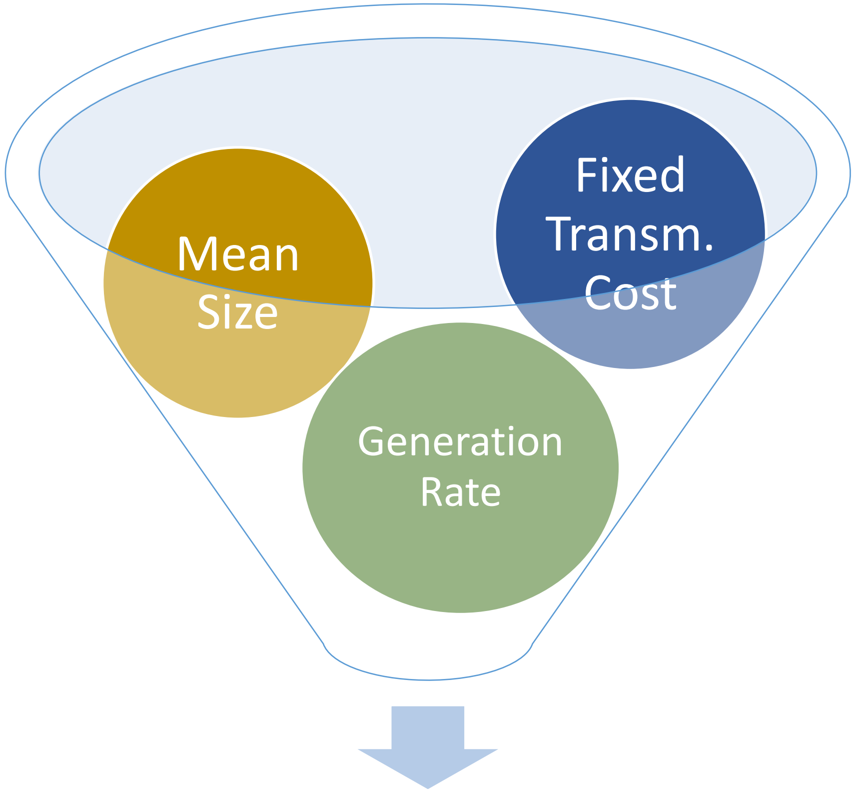
- Example:



CR dependent on G_l

- Example:

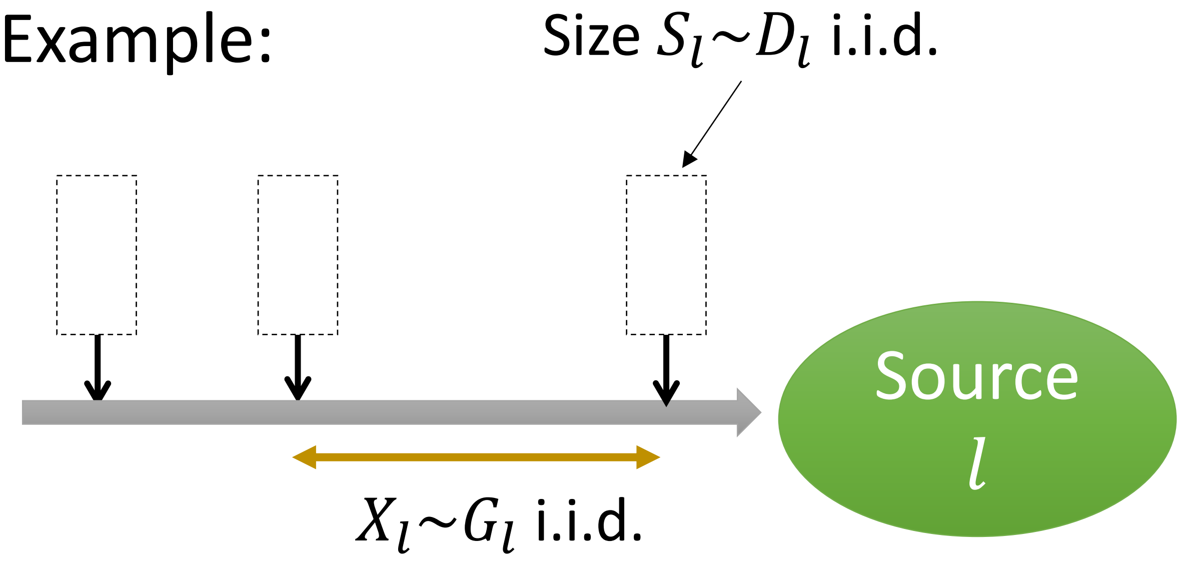


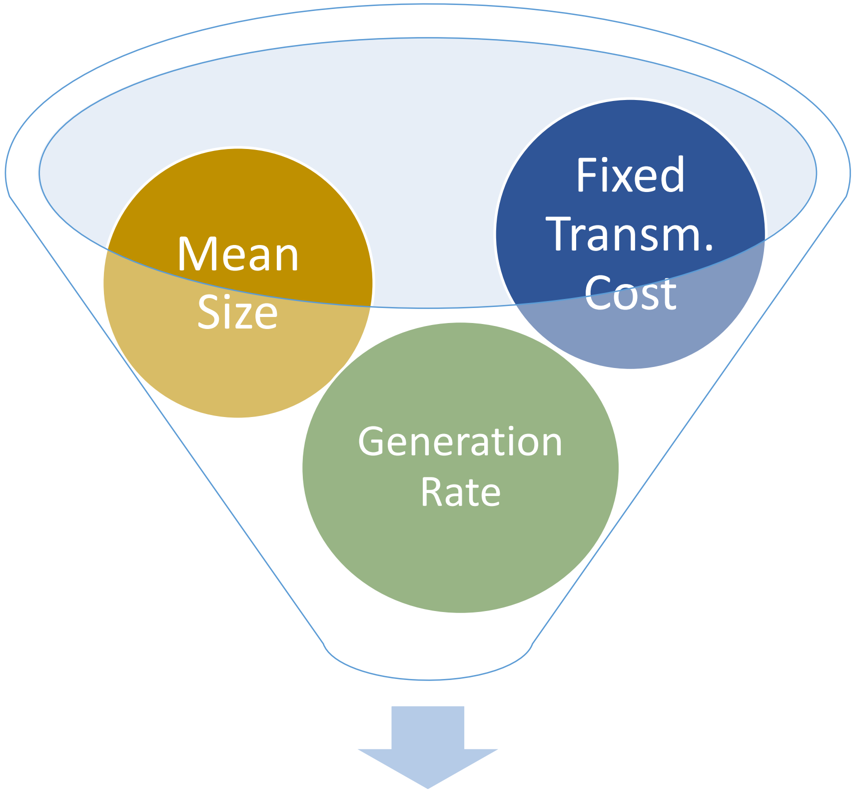


Optimal inter-generation time of **transmitted** updates T_l^*

CR dependent on G_l

- Example:



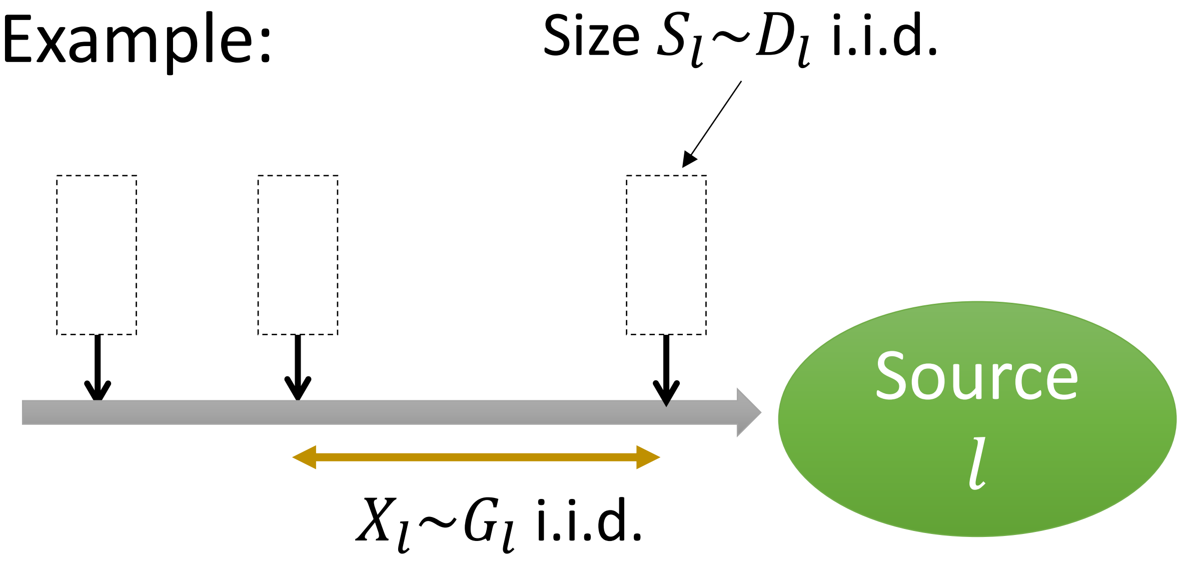


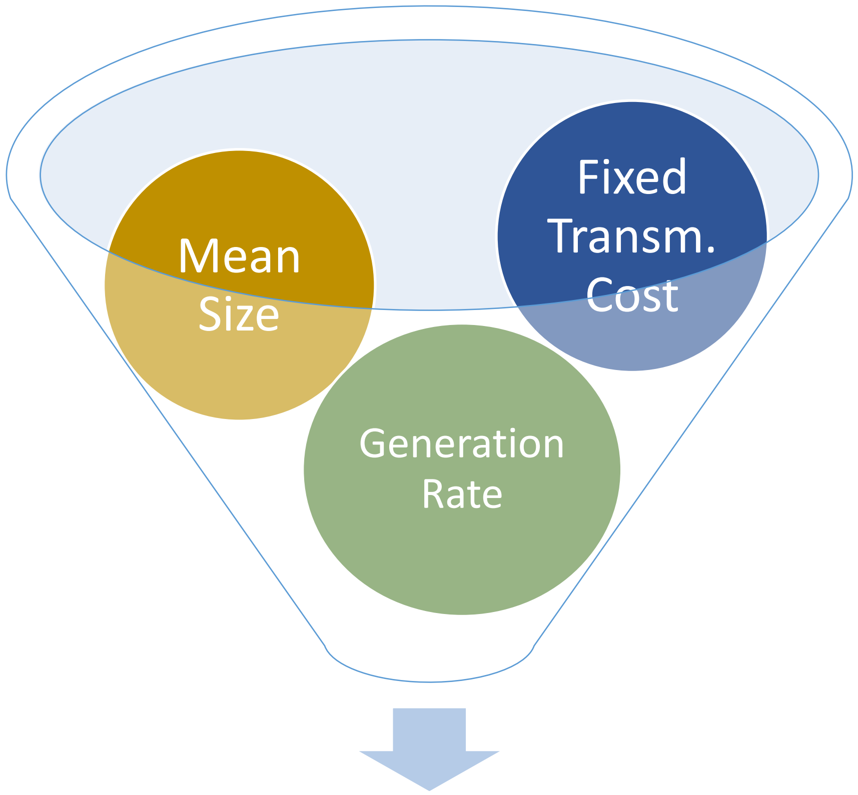
Optimal inter-generation time of **transmitted** updates T_l^*

Transmit updates that min variance w.r.t. T_l^*, for all l

CR dependent on G_l

- Example:





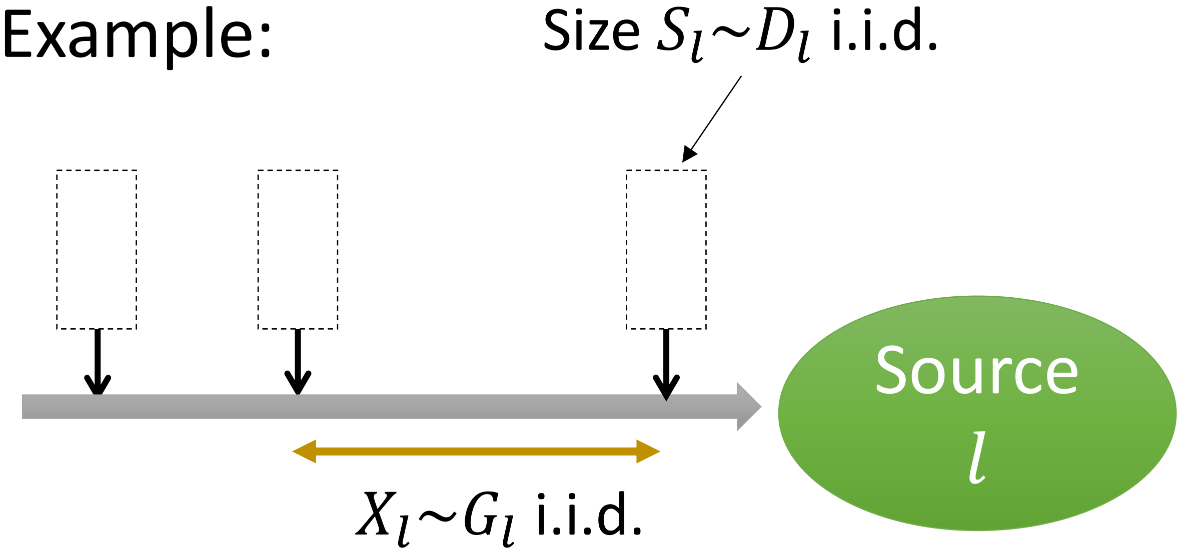
Optimal inter-generation time of **transmitted** updates T_l^*

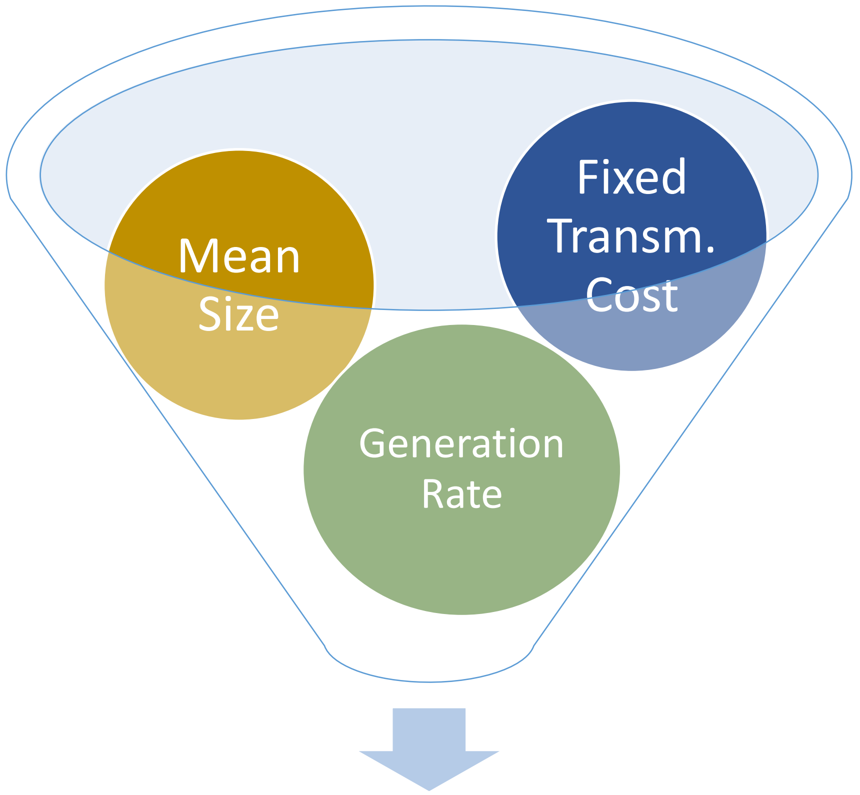
Transmit updates that min variance w.r.t. T_l^*, for all l

Complicated for general distributions G_l!

CR dependent on G_l

- Example:





Optimal inter-generation time of **transmitted** updates T_l^*

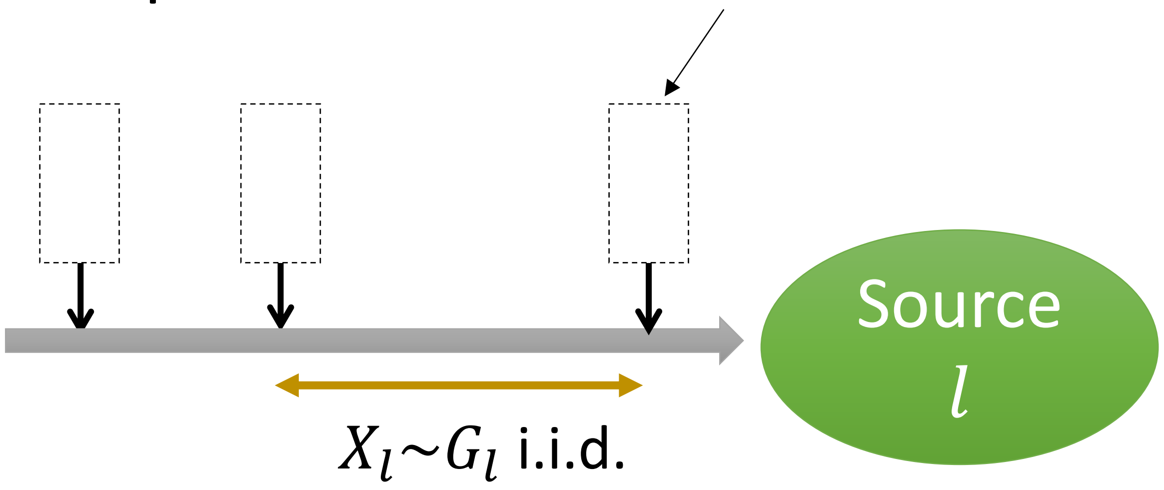
Transmit updates that min variance w.r.t. T_l^*, for all l

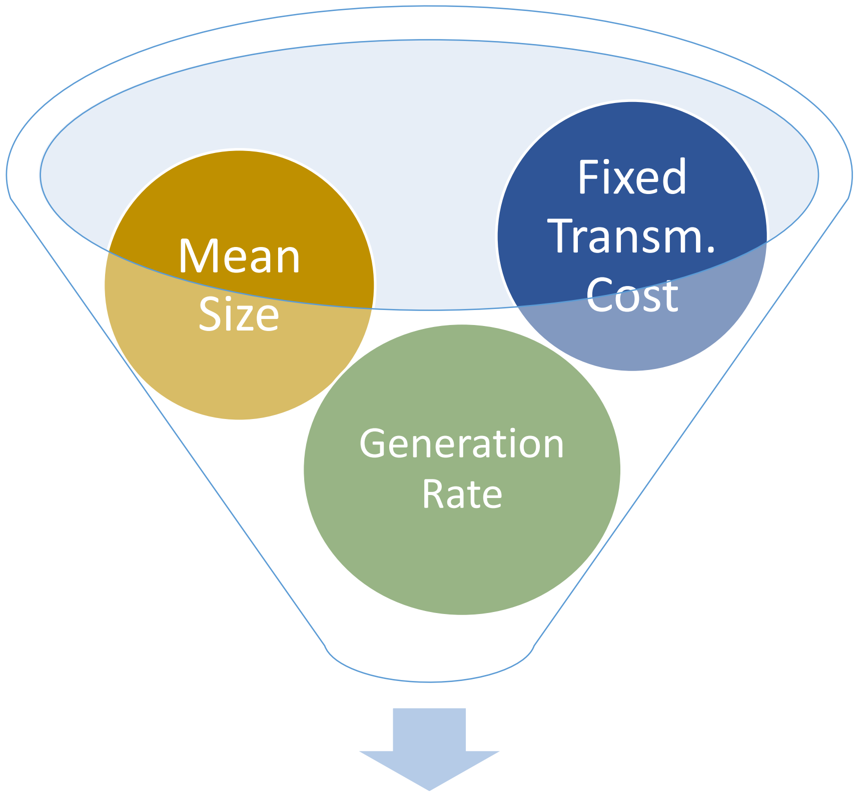
Complicated for general distributions G_l!

Optimal offline policy (OPT) may have zero variance

CR dependent on G_l

- Example:





Optimal inter-generation time of **transmitted** updates T_l^*

Transmit updates that min variance w.r.t. T_l^*, for all l

Complicated for general distributions G_l!

Optimal offline policy (OPT) may have zero variance

Need a better bound on OPT's cost

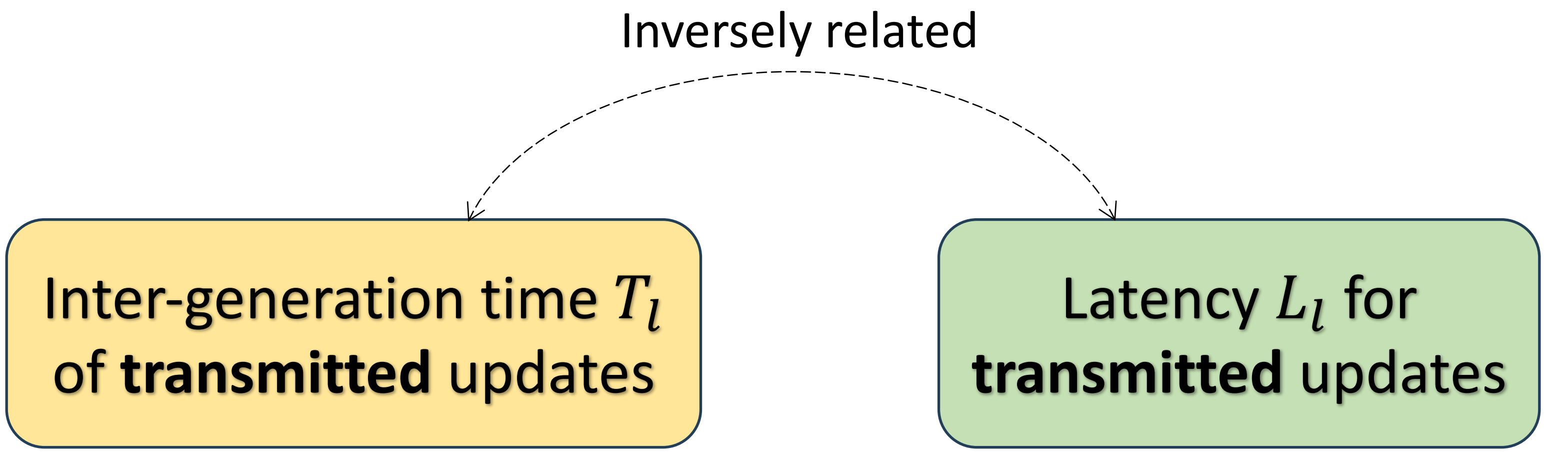
CR independent of D_l

CR independent of D_l

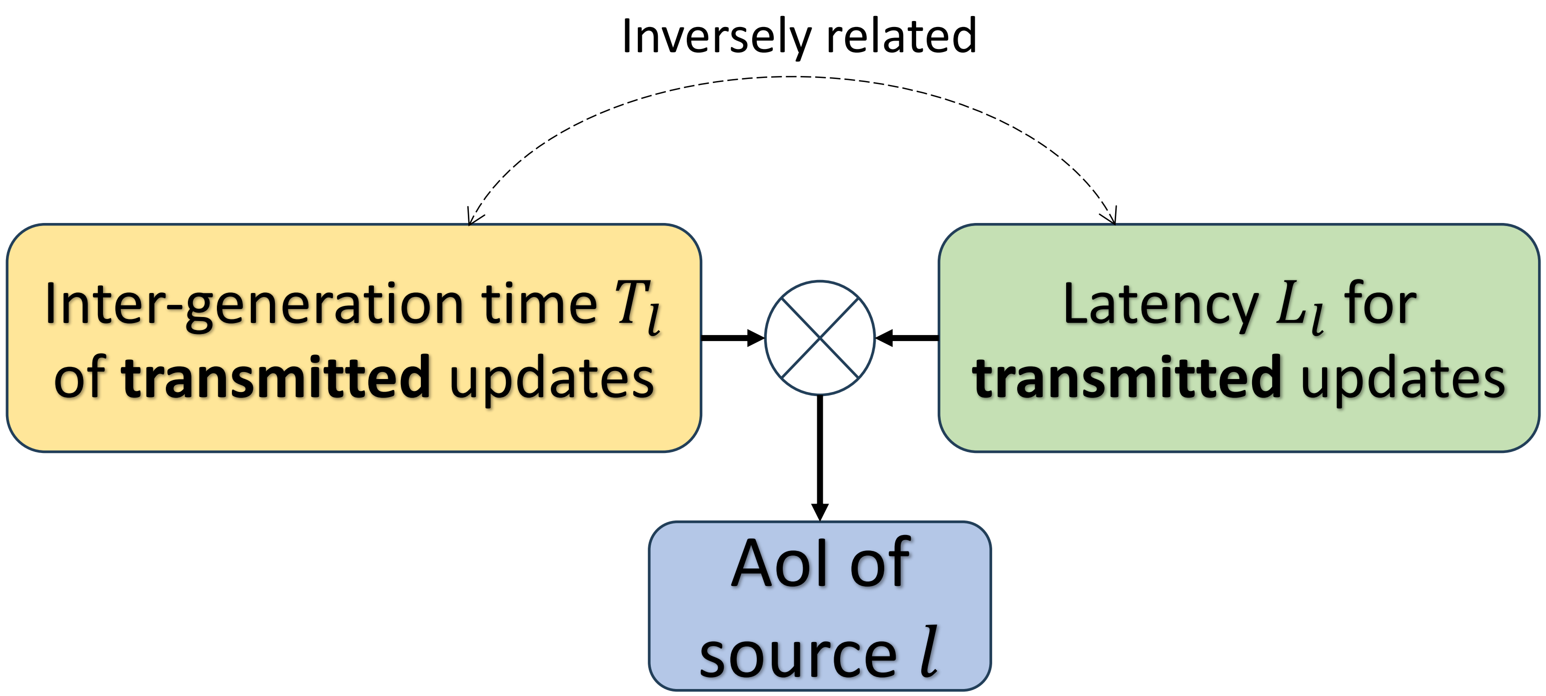
Inter-generation time T_l of **transmitted** updates

Latency L_l for **transmitted** updates

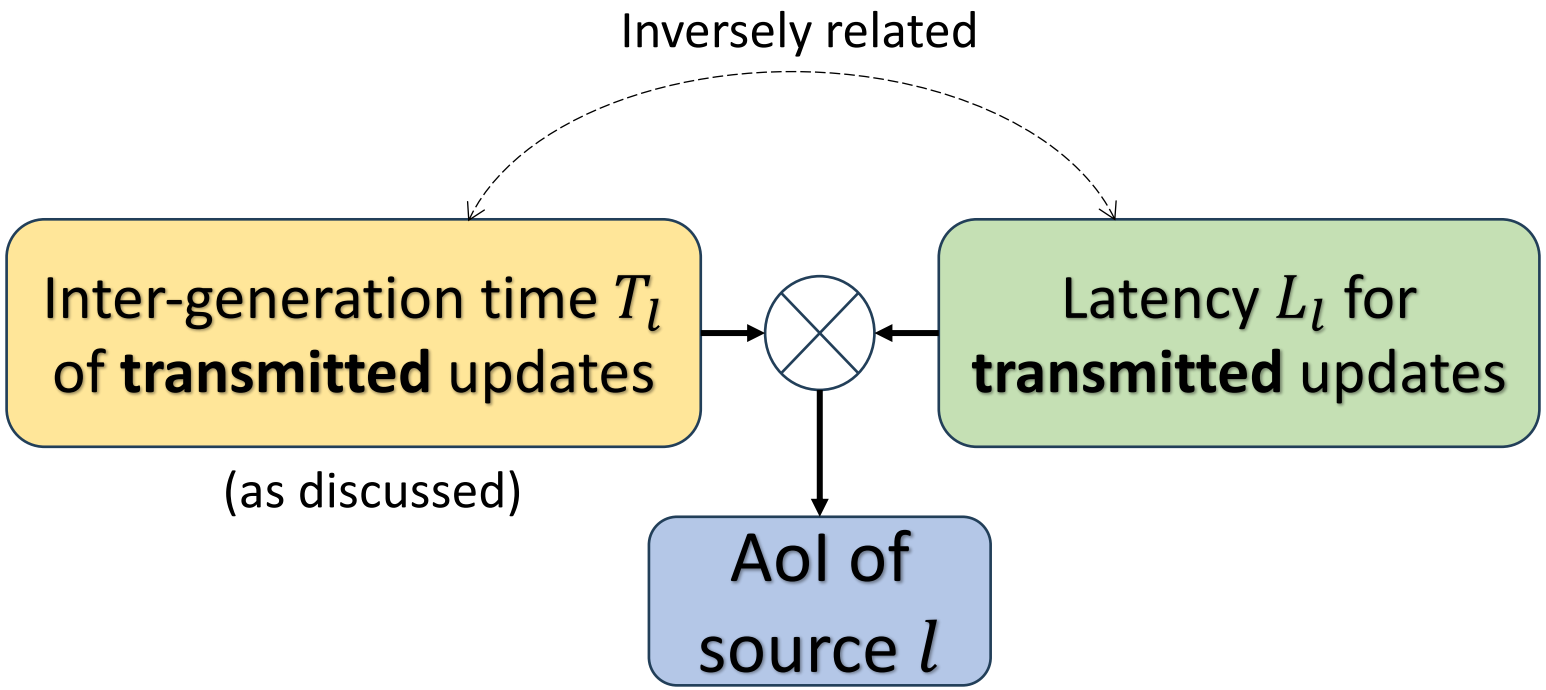
CR independent of D_l



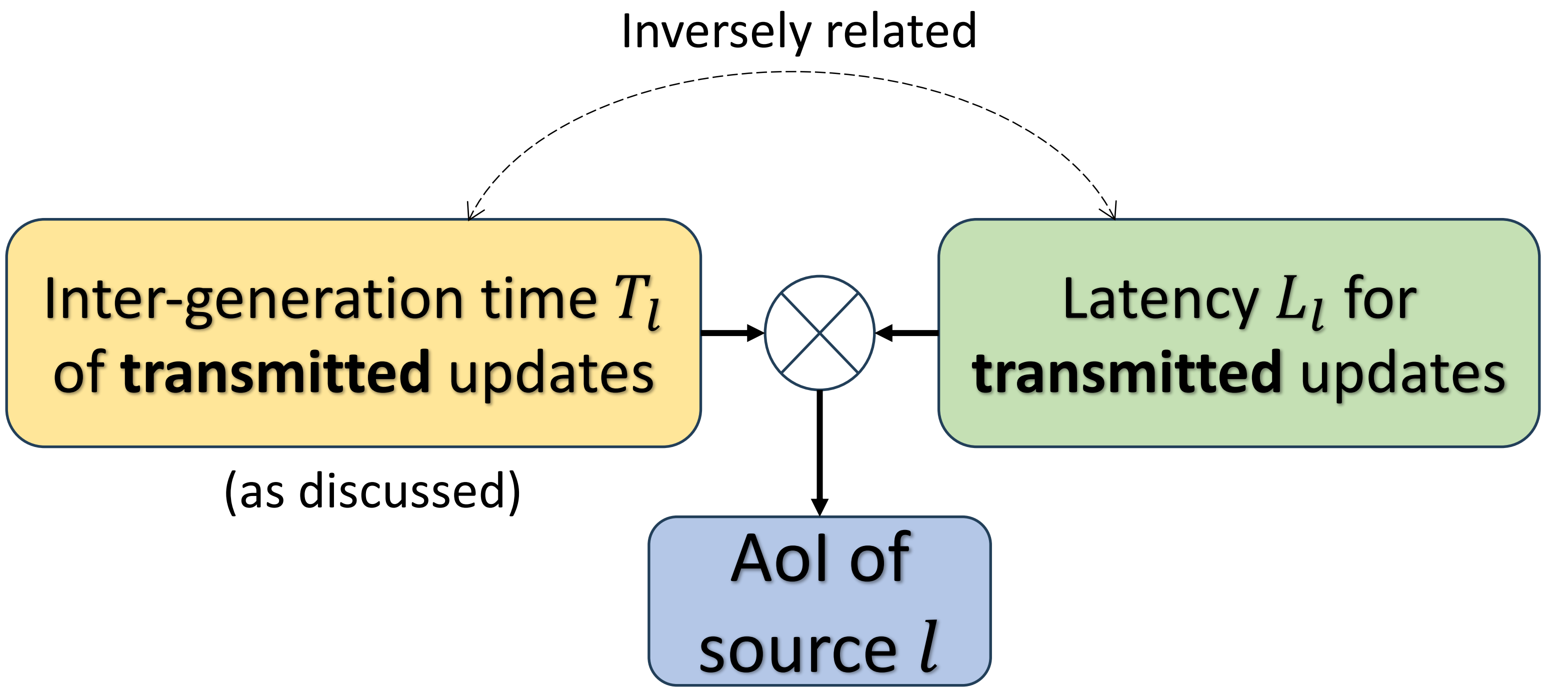
CR independent of D_l



CR independent of D_l

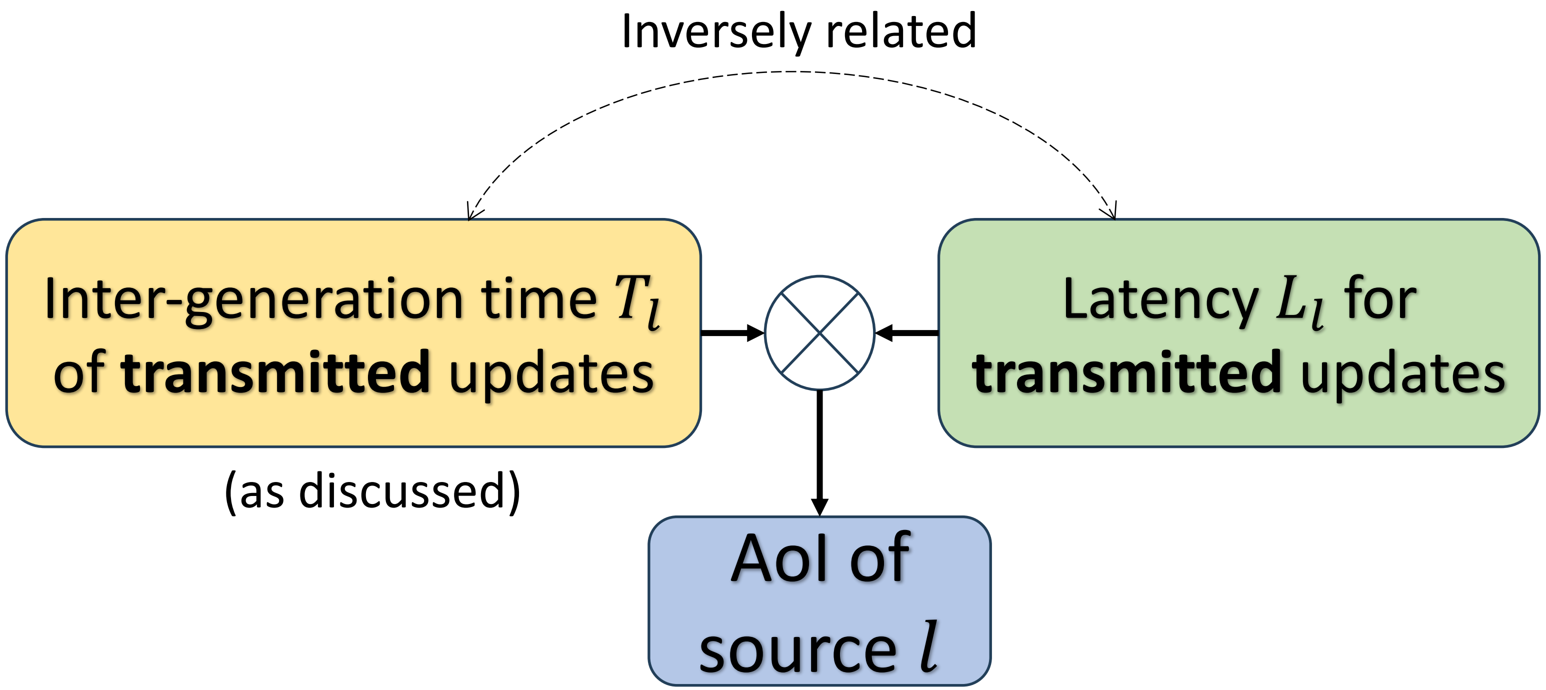


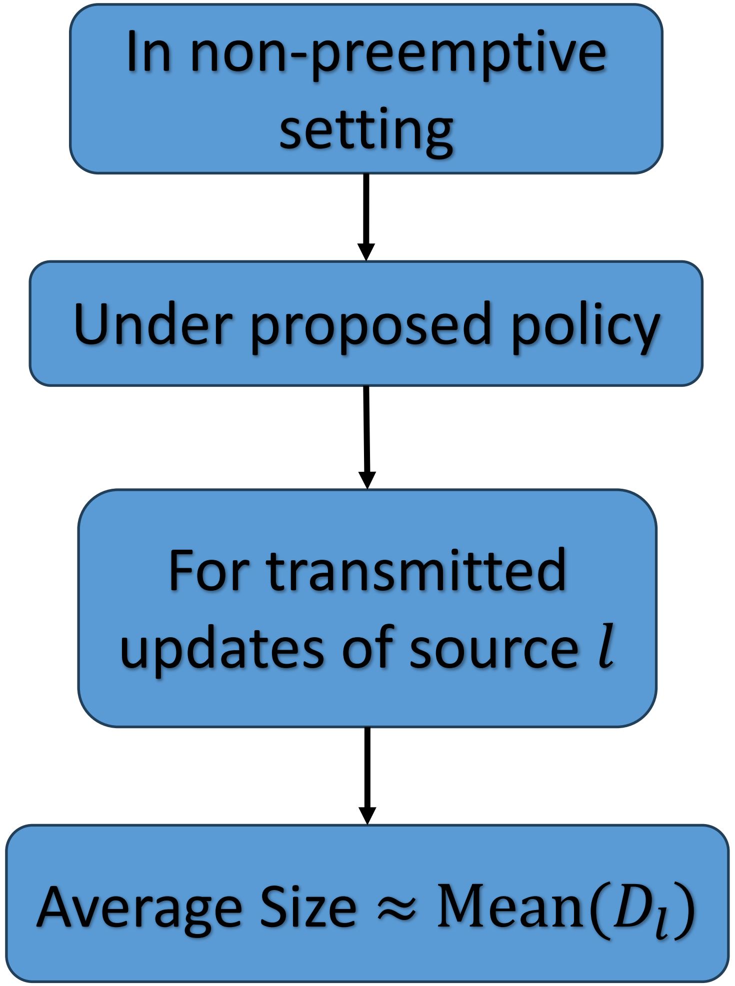
CR independent of D_l



For appropriate T_l, latency L_l primarily depends on update size $S_l \sim D_l$

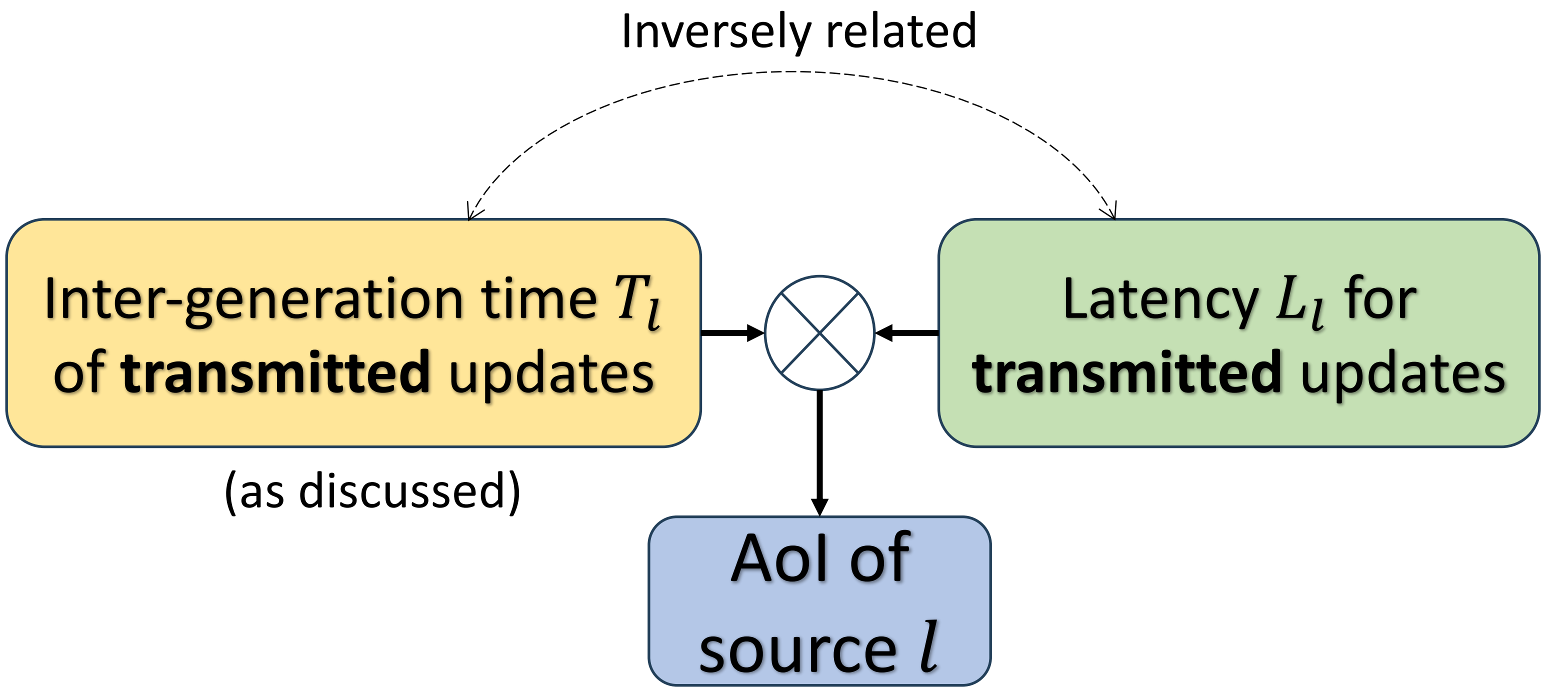
CR independent of D_l



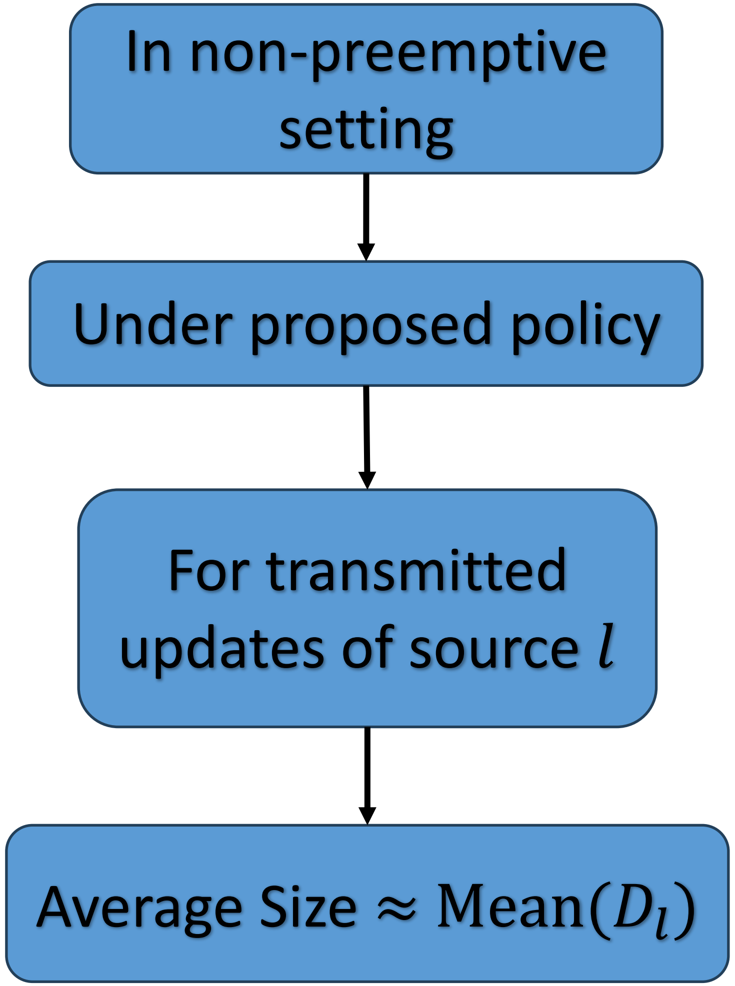


For appropriate T_l, latency L_l primarily depends on update size $S_l \sim D_l$

CR independent of D_l

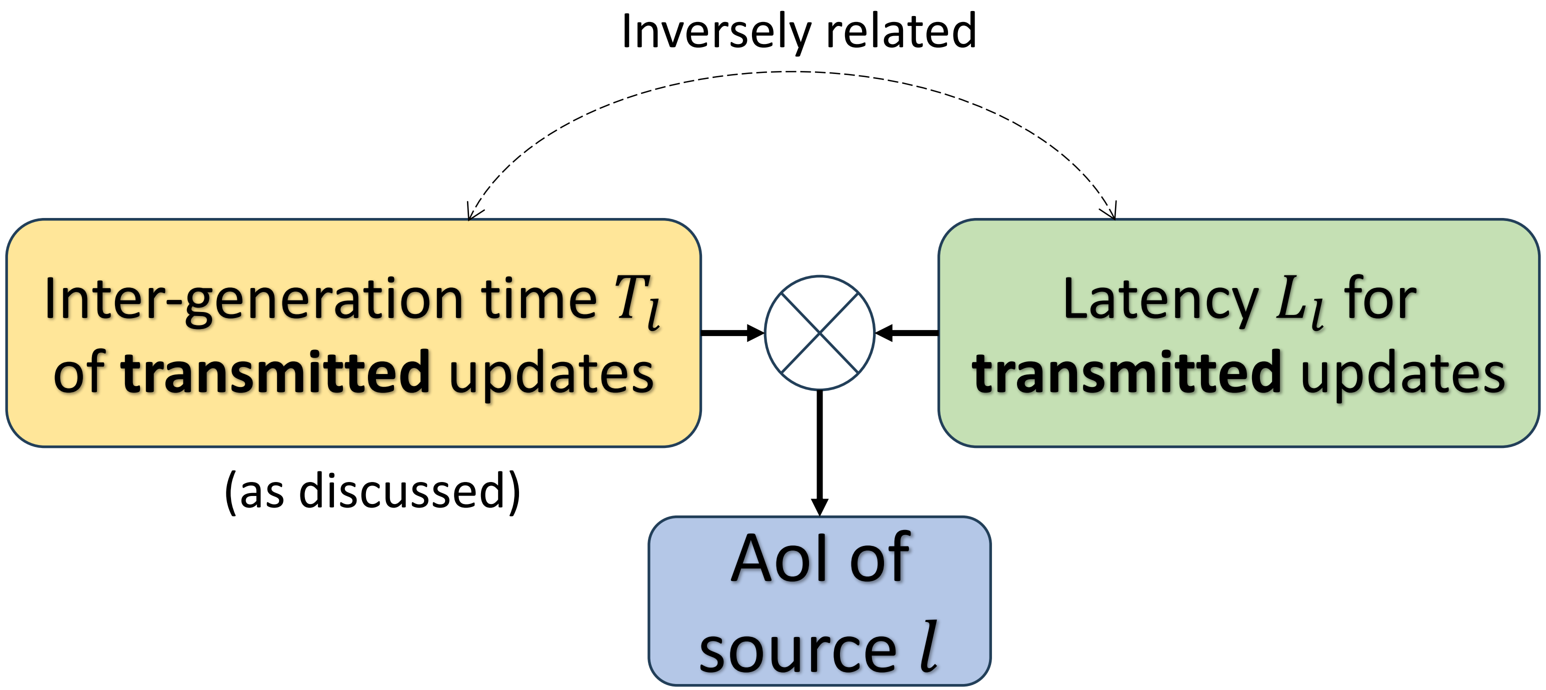


For appropriate T_l, latency L_l primarily depends on update size $S_l \sim D_l$

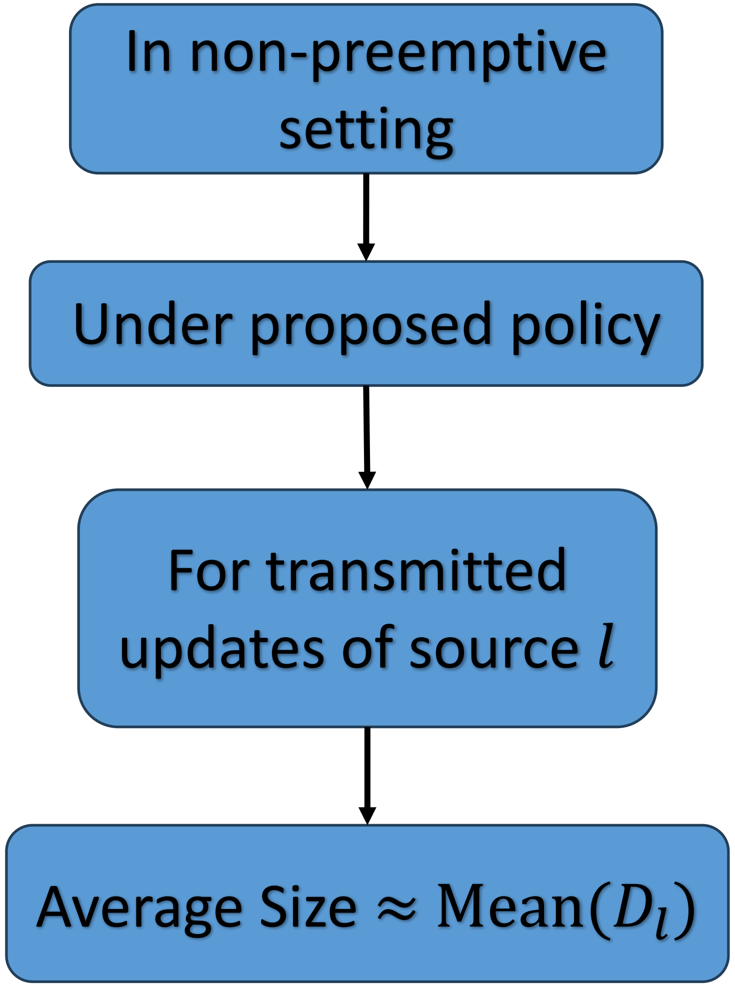


Latency comparable to OPT

CR independent of D_l



For appropriate T_l, latency L_l primarily depends on update size $S_l \sim D_l$



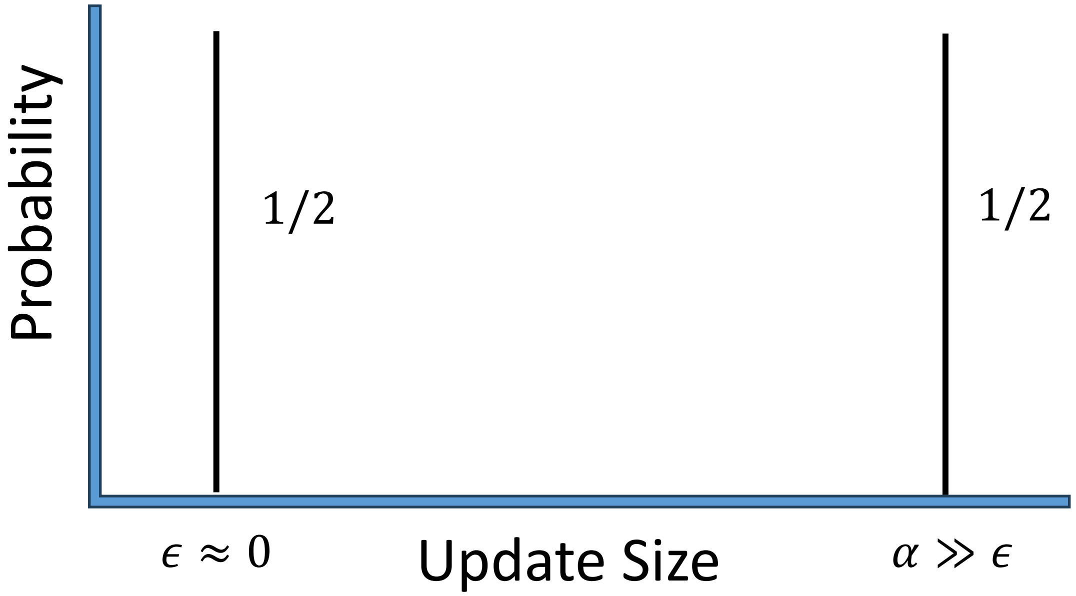
Latency comparable to OPT

CR independent of D_l!

Towards preemptive setting ….

Towards preemptive setting ….

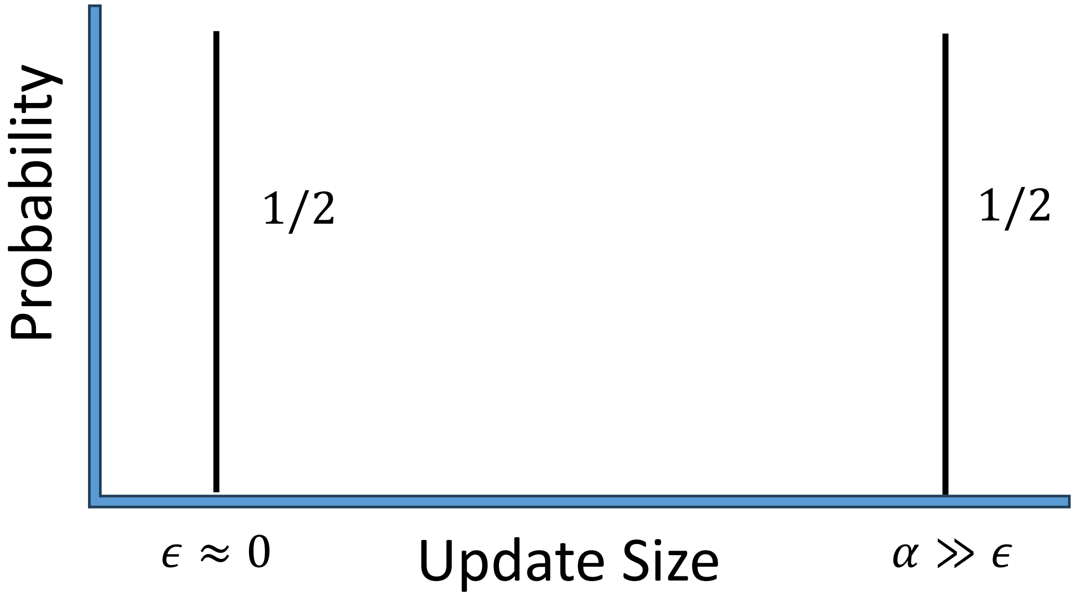
Example (Size distribution D_l):



Towards preemptive setting ….

OPT may preempt updates and transmit new ones if transmission exceeds ϵ time units

Example (Size distribution D_l):

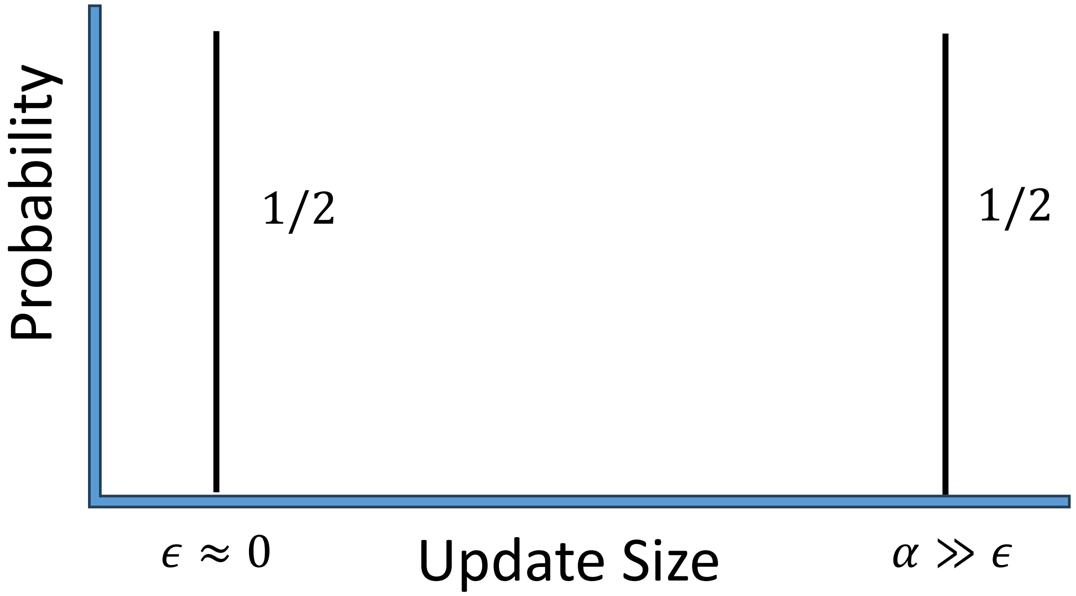


Towards preemptive setting ....

OPT may preempt updates and transmit new ones if transmission exceeds ϵ time units

Time req to completely transmit one update

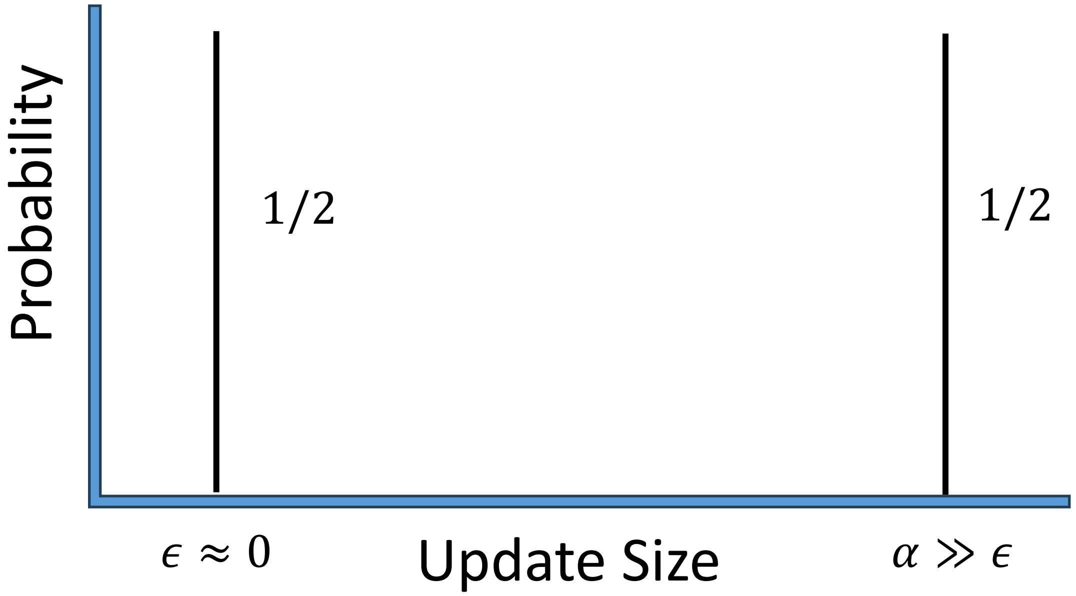
Example (Size distribution D_l):



Towards preemptive setting ….

OPT may preempt updates and transmit new ones if transmission exceeds ϵ time units → Time req to completely transmit one update → Arbitrarily small

Example (Size distribution D_l):



Towards preemptive setting ….

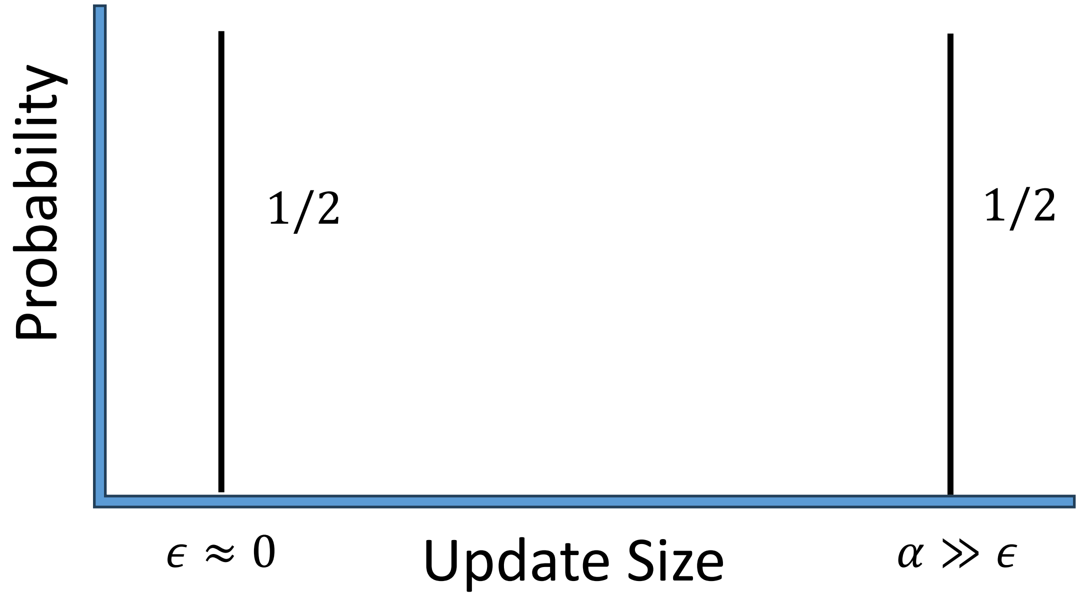
OPT may preempt updates and transmit new ones if transmission exceeds ϵ time units

→ Time req to completely transmit one update

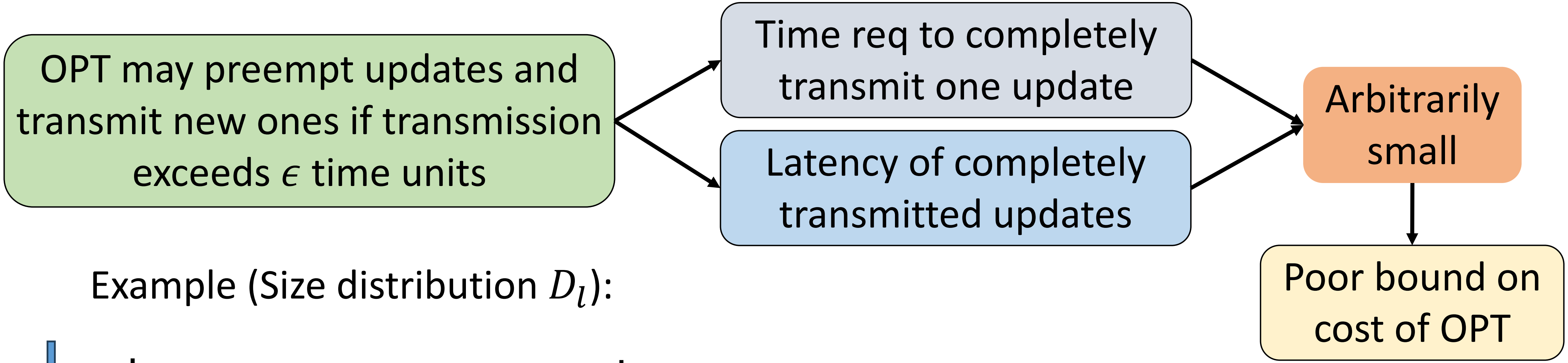
→ Latency of completely transmitted updates

→ Arbitrarily small

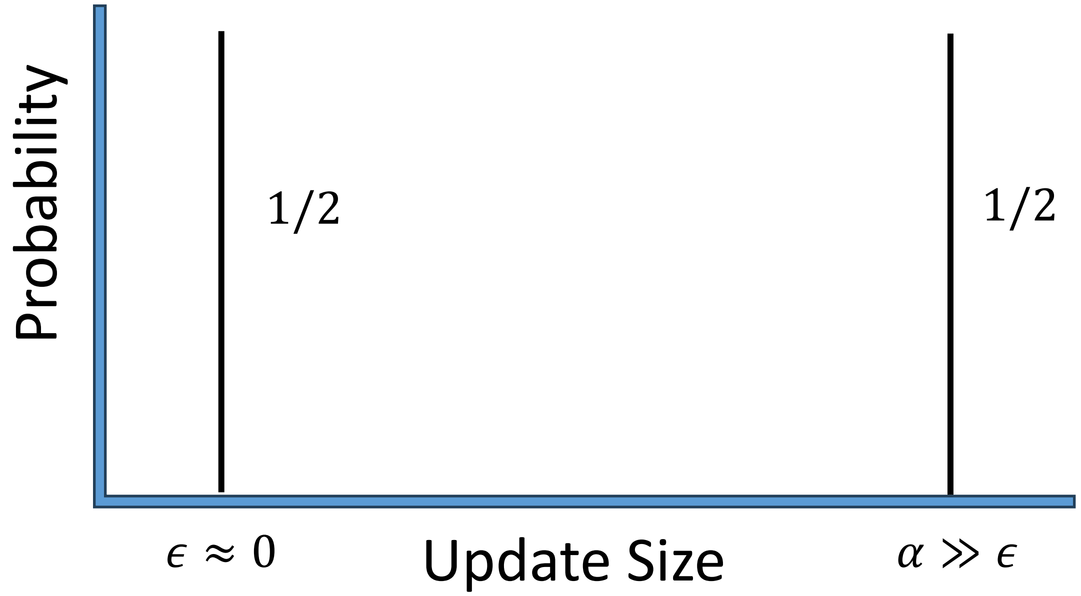
Example (Size distribution D_l):



Towards preemptive setting ….



Example (Size distribution D_l):



Towards preemptive setting ....

OPT may preempt updates and transmit new ones if transmission exceeds ϵ time units

Time req to completely transmit one update

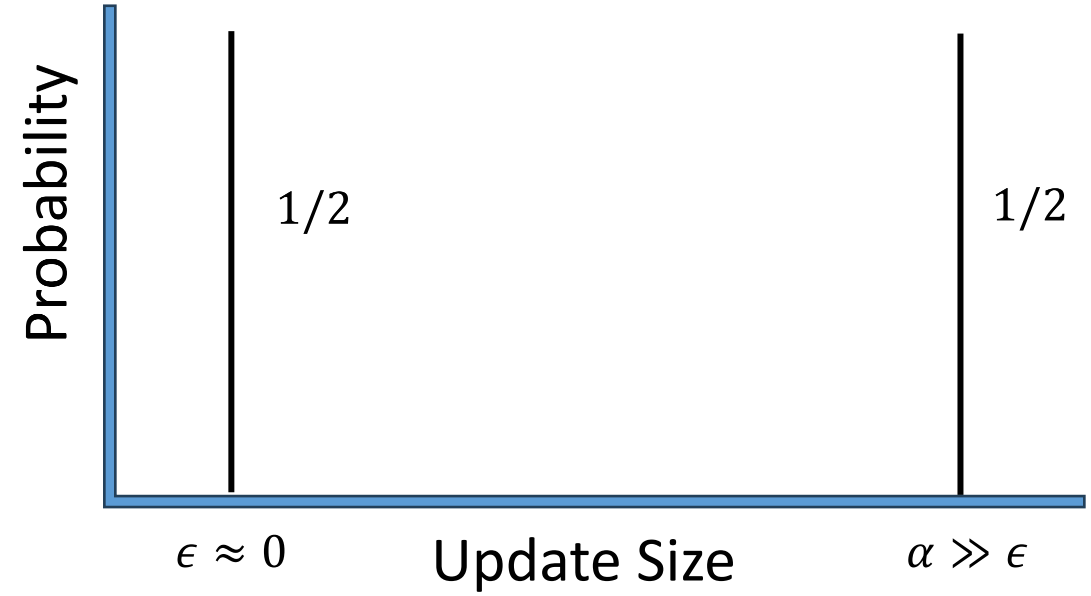
Latency of completely transmitted updates

Arbitrarily small

Poor bound on cost of OPT

For causal policy, not obvious when to preempt

Example (Size distribution D_l):



Towards preemptive setting ….

OPT may preempt updates and transmit new ones if transmission exceeds ϵ time units

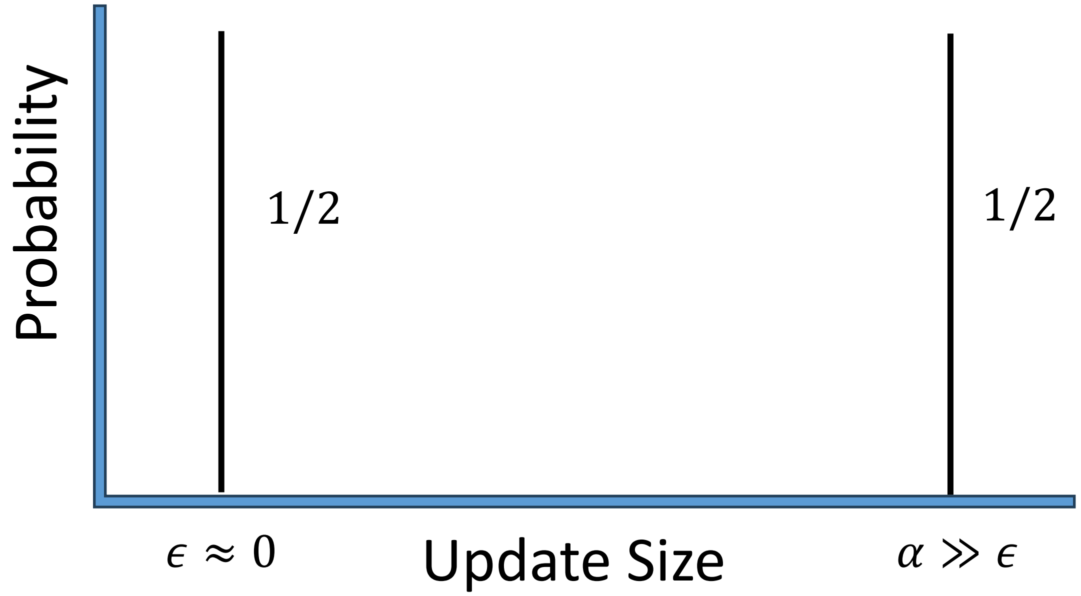
→ Time req to completely transmit one update

→ Latency of completely transmitted updates

→ Arbitrarily small

→ Poor bound on cost of OPT

Example (Size distribution D_l):



For causal policy, not obvious when to preempt

→ Need to work for all arrival & size distributions (G_l, D_l), & Tx. cost c_l

Towards preemptive setting ….

OPT may preempt updates and transmit new ones if transmission exceeds ϵ time units

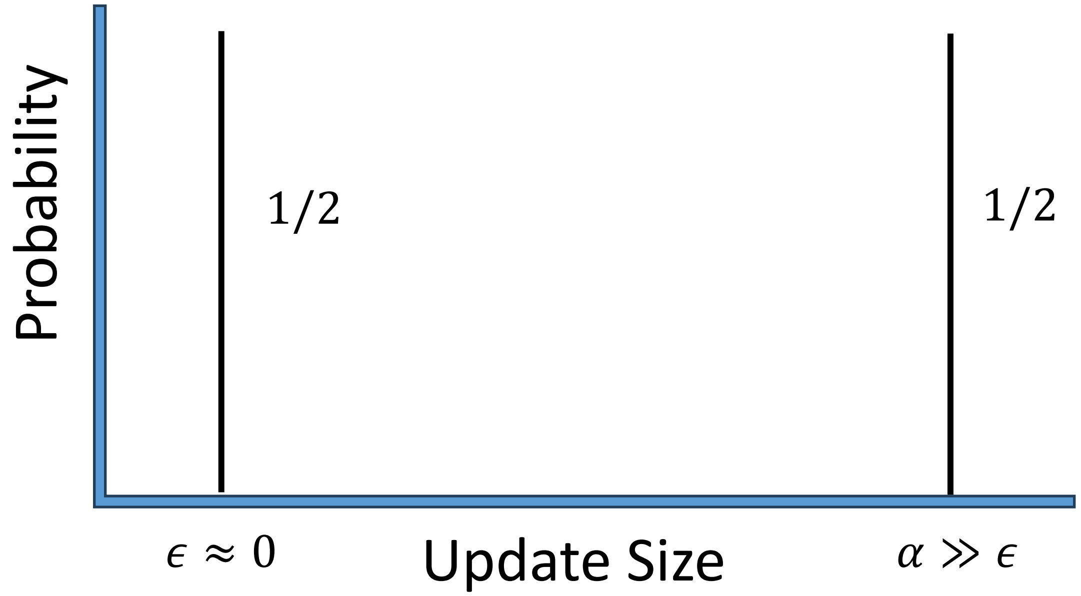
→ Time req to completely transmit one update

→ Latency of completely transmitted updates

→ Arbitrarily small

→ Poor bound on cost of OPT

Example (Size distribution D_l):

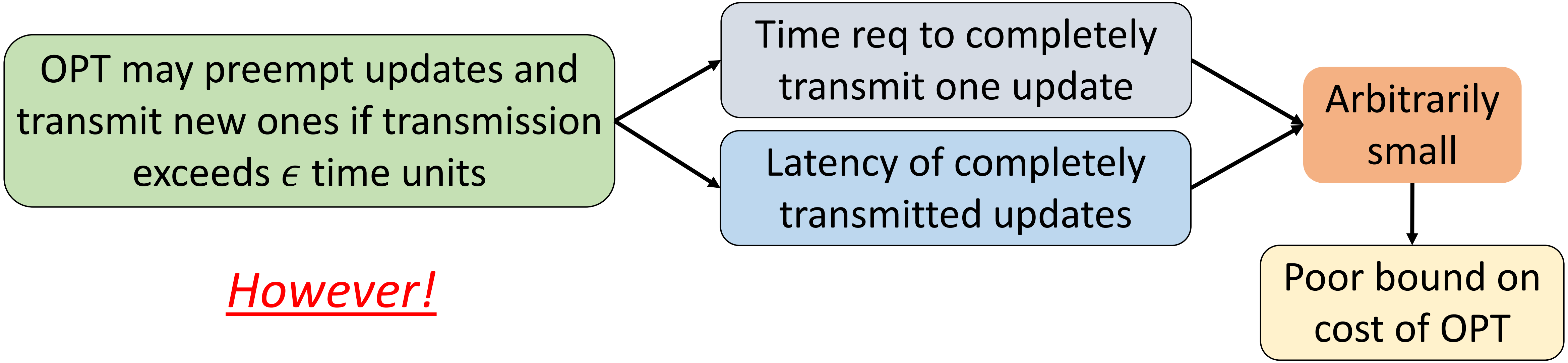


For causal policy, not obvious when to preempt

→ Need to work for all arrival & size distributions (G_l, D_l), & Tx. cost c_l

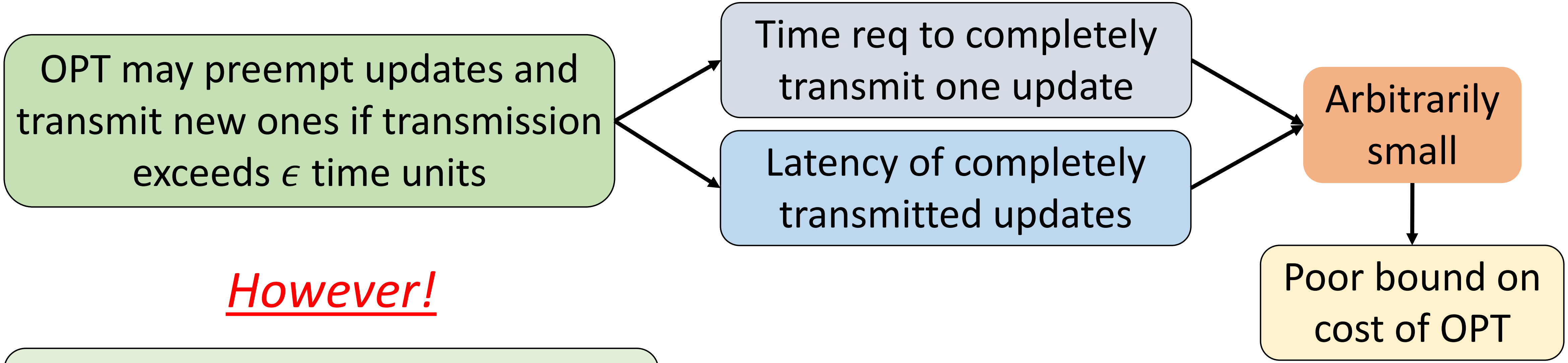
Hard to bound competitive ratio!

Towards preemptive setting ....



However!

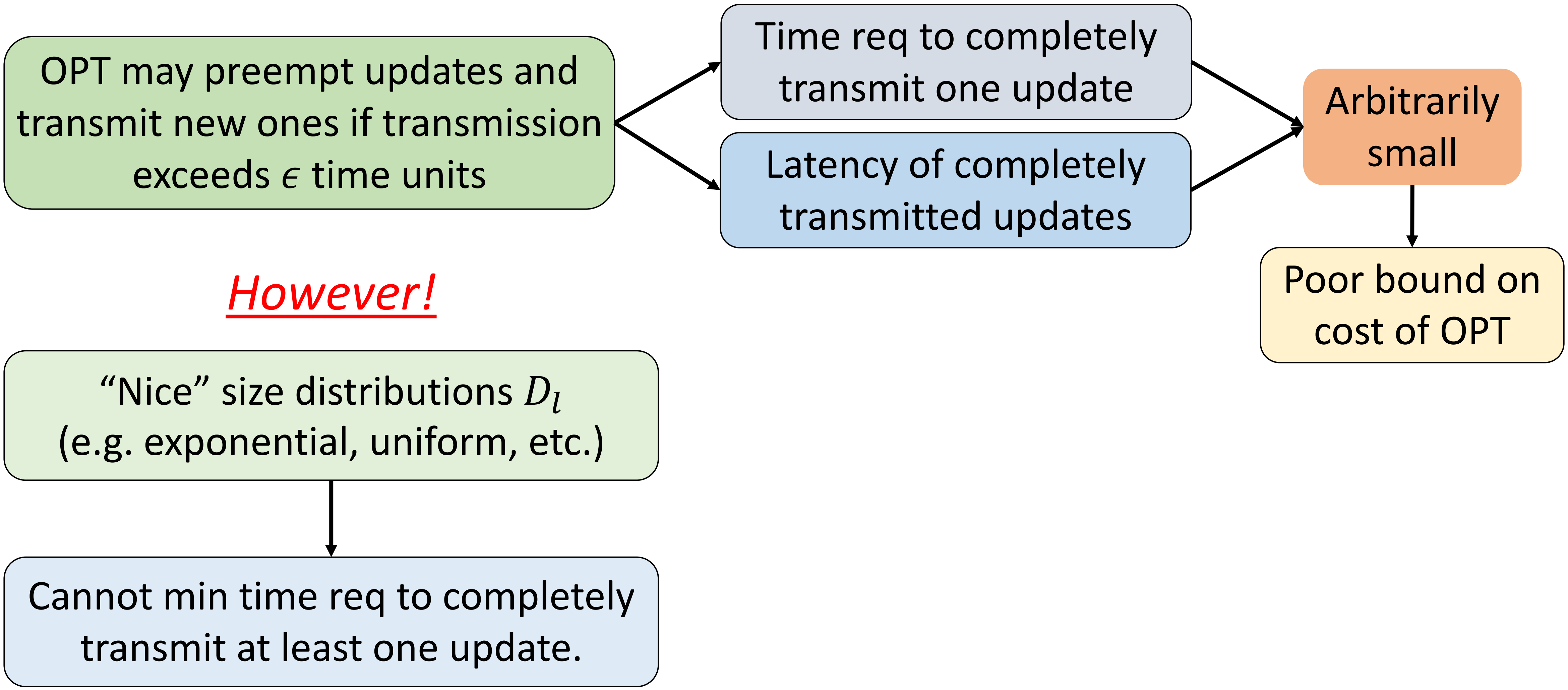
Towards preemptive setting ….



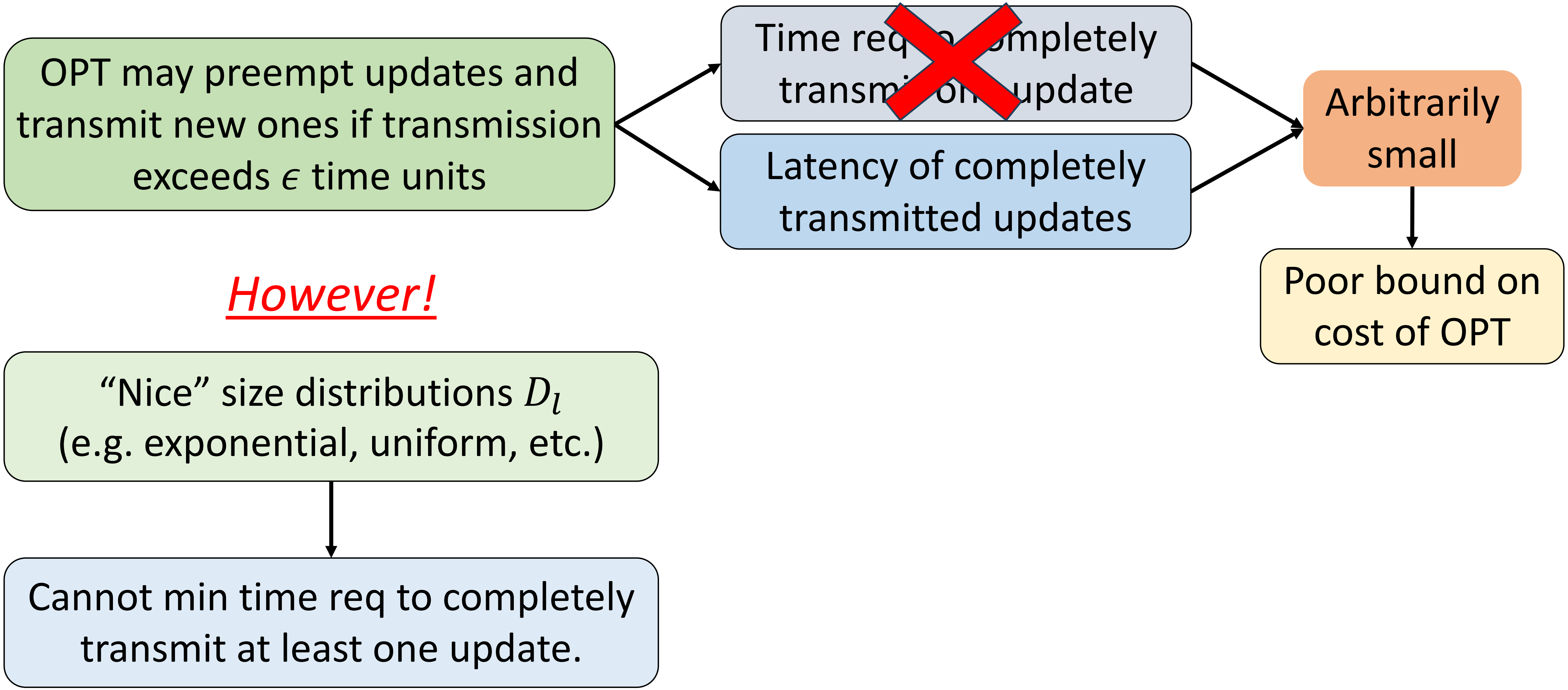
<u>However!</u>

“Nice” size distributions D_l
(e.g. exponential, uniform, etc.)

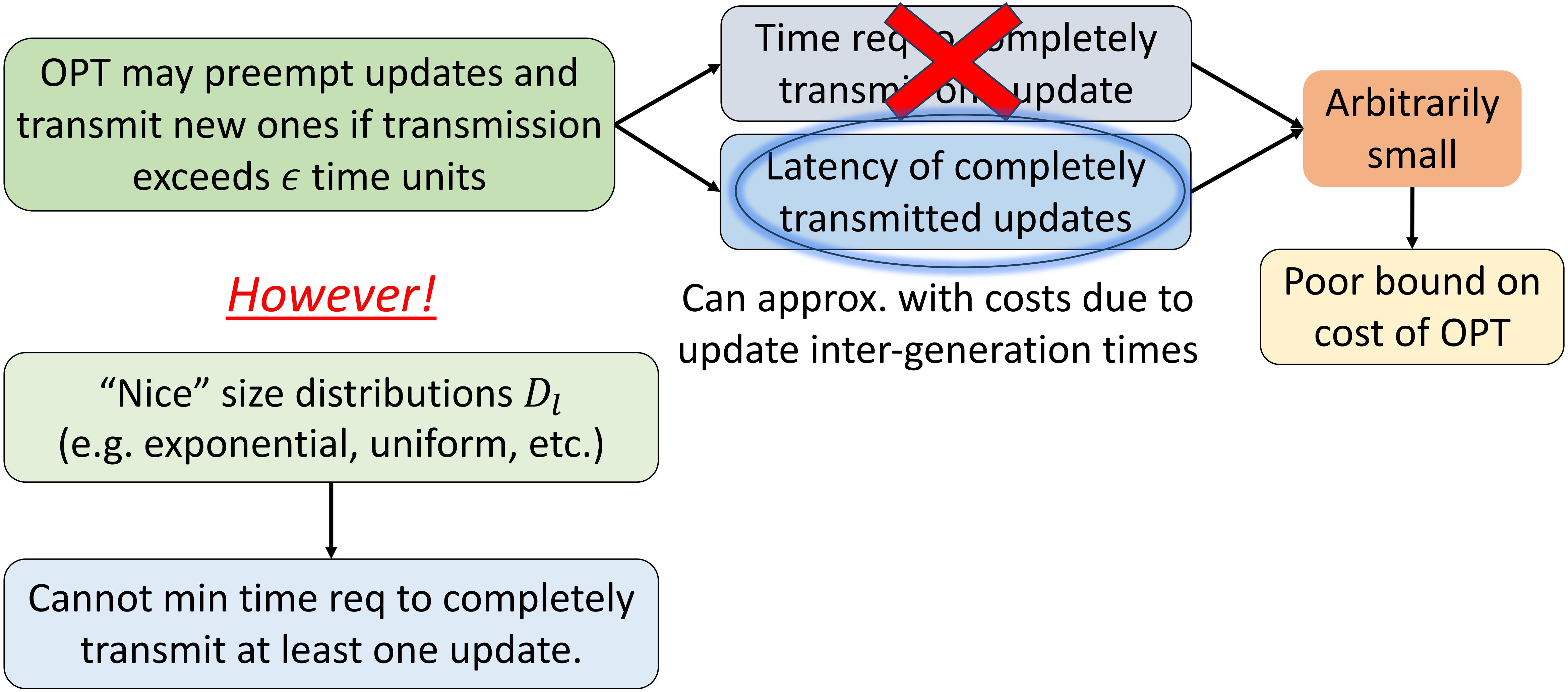
Towards preemptive setting ….



Towards preemptive setting ....



Towards preemptive setting ….



Towards preemptive setting ….

OPT may preempt updates and transmit new ones if transmission exceeds ϵ time units

Time req to completely transmission update

Latency of completely transmitted updates

Can approx. with costs due to update inter-generation times

Arbitrarily small

Poor bound on cost of OPT

However!

“Nice” size distributions D_l
(e.g. exponential, uniform, etc.)

Cannot min time req to completely transmit at least one update.

Towards preemptive setting ....

OPT may preempt updates and transmit new ones if transmission exceeds ϵ time units

Time req to completely transmit one update

Latency of completely transmitted updates

Can approx. with costs due to update inter-generation times

Arbitrarily small

Poor bound on cost of OPT

However!

"Nice" size distributions D_l (e.g. exponential, uniform, etc.)

Cannot min time req to completely transmit at least one update.

Competitive ratio (proposed policy):

$$CR \leq 5 + \max_{l} \frac{Variance(G_l)}{Mean^2(G_l)}$$

Towards preemptive setting ….

OPT may preempt updates and transmit new ones if transmission exceeds ϵ time units

Time req to completely transmit one update

Latency of completely transmitted updates

Can approx. with costs due to update inter-generation times

Arbitrarily small

Poor bound on cost of OPT

However!

“Nice” size distributions D_l (e.g. exponential, uniform, etc.)

Cannot min time req to completely transmit at least one update.

Competitive ratio (proposed policy):

$$CR \leq 5 + \max_{l} \frac{Variance(G_l)}{Mean^2(G_l)}$$

Exceeds non-preemptive case only by additive constant 2

Conclusion

Conclusion

Online scheduling problem

Conclusion

Online scheduling problem → Minimize AoI and energy cost

Conclusion

Online scheduling problem → Minimize AoI and energy cost

Combinatorial!

Conclusion

Online scheduling problem → Minimize AoI and energy cost

Combinatorial!

1 Single Source → Arbitrary update arrival times & sizes → Classical policies not useful → SRPT+ is 5-competitive

Conclusion

Online scheduling problem → Minimize AoI and energy cost

Combinatorial!

1. Single Source → Arbitrary update arrival times & sizes → Classical policies not useful → SRPT+ is 5-competitive

2. Multiple Sources → Arbitrary update arrival & size distributions → A doubly randomized policy → CR independent of size distribution*

Conclusion

Online scheduling problem → Minimize AoI and energy cost

Combinatorial!

1. Single Source → Arbitrary update arrival times & sizes → Classical policies not useful → SRPT+ is 5-competitive

2. Multiple Sources → Arbitrary update arrival & size distributions → A doubly randomized policy → CR independent of size distribution*

Does there exist any causal policy with CR independent of update arrival distributions?

Relevant Publications

- Kumar Saurav & Rahul Vaze. Minimizing Age of Information under Arbitrary Arrival Model with Arbitrary Packet Size. Performance Evaluation, 2023.

Relevant Publications

- Kumar Saurav & Rahul Vaze. Minimizing Age of Information under Arbitrary Arrival Model with Arbitrary Packet Size. Performance Evaluation, 2023.
    - Results for stochastic setting appeared in IEEE INFOCOM 2021.

Relevant Publications

- Kumar Saurav & Rahul Vaze. Minimizing Age of Information under Arbitrary Arrival Model with Arbitrary Packet Size. Performance Evaluation, 2023.
  - Results for stochastic setting appeared in IEEE INFOCOM 2021.
  - Results for tuneable transmission rate/energy consumption model appeared in IEEE Journal on Selected Areas in Information Theory, 2023.

Relevant Publications

- Kumar Saurav & Rahul Vaze. Minimizing Age of Information under Arbitrary Arrival Model with Arbitrary Packet Size. Performance Evaluation, 2023.
  - Results for stochastic setting appeared in IEEE INFOCOM 2021.
  - Results for tuneable transmission rate/energy consumption model appeared in IEEE Journal on Selected Areas in Information Theory, 2023.
- Kumar Saurav & Rahul Vaze. *Scheduling to minimize age of information with multiple sources*. IEEE Journal on Selected Areas in Information Theory, 2023, IEEE.


THE END