Principles for Measurability in Protocol Design

Mark Allman, Rob Beverly, and ***Brian Trammell***

in ACM CCR, April 2017

Measurement and Analysis for Protocols RG
IETF 100 Singapore — 13 November 2017

Network Measurement

- Fundamental to network operations, research, protocol design, and informing Internet policy development.
- Minimal support from stack today (`ping` is what you get)
  - Unintended features (e.g. `traceroute`)
  - Brittle hacks (e.g. passive TCP loss/RTT)
  - Inference (cf. any academic measurement paper)

Result:
Important questions are hard

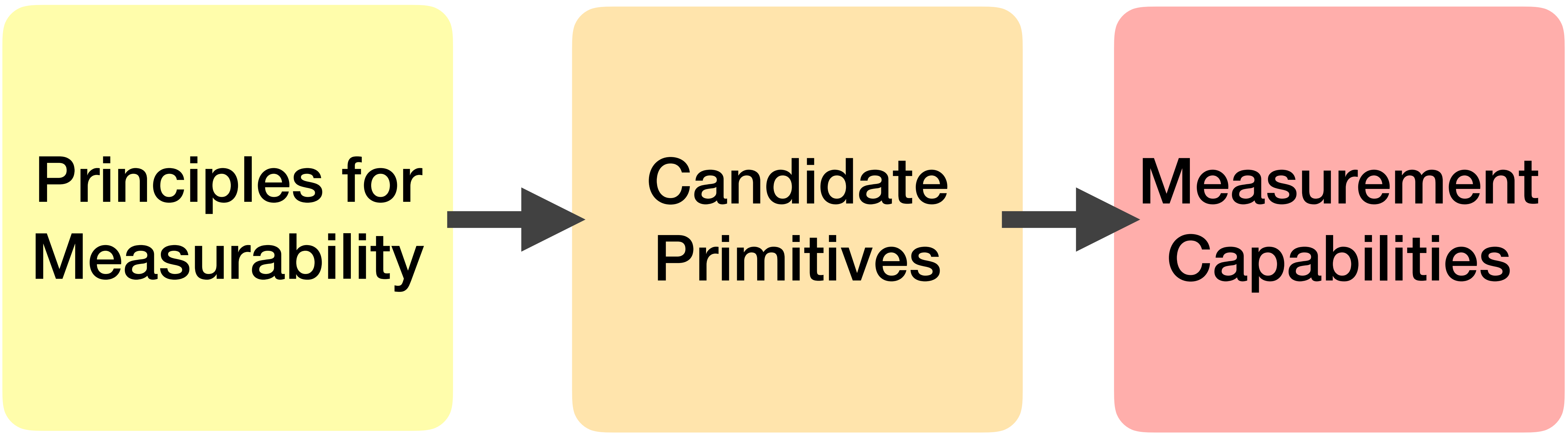
- What's the best path to route traffic?
- What is the capacity or utilization of a link?
- How do networks interconnect?
- What AS operates a given router?

Even simple inferences are difficult!

- What's the delay between two hosts?
  - (Per-protocol traffic differentiation, path vs. host delay, asymmetry)
- What are the endpoints in a communication?
  - (NATs, CGNs, aliases, IPv6)
- How did packets arrive at a remote destination?
  - (order? modified? mangled? path? queued?)

What if we re-think the Internet protocol stack with measurability as a first-class component?

Approach



- Imagine packets carry measurement information: what should they include?
    - Goal: maximum benefit for minimum overhead

Principles for Measurability

1. Measurement should be **explicit**
2. Measurement should be **in-band**
3. The measurement **consumer** should **bear the cost**
4. The measurement **provider** should **retain control**
5. Measurement must be **visible**
6. Measurement should be **cooperative**

Principles for Measurability

1. Measurement should be **explicit**
   Reduces ambiguity, increases future-proof-ness
2. Measurement should be **in-band**
3. The measurement **consumer** should **bear the cost**
4. The measurement **provider** should **retain control**
5. Measurement must be **visible**
6. Measurement should be **cooperative**

Principles for Measurability

1. Measurement should be **explicit**
   Reduces ambiguity, increases future-proof-ness
2. Measurement should be **in-band**
   Ensures measurement traffic treated similar to"real" traffic
3. The measurement **consumer** should **bear the cost**
4. The measurement **provider** should **retain control**
5. Measurement must be **visible**
6. Measurement should be **cooperative**

Principles for Measurability

1. Measurement should be **explicit**
   Reduces ambiguity, increases future-proof-ness
2. Measurement should be **in-band**
   Ensures measurement traffic treated similar to"real" traffic
3. The measurement **consumer** should **bear the cost**
   Shifts state and per-packet overhead offline, increases deployability
4. The measurement **provider** should **retain control**

5. Measurement must be **visible**

6. Measurement should be **cooperative**

Principles for Measurability

1. Measurement should be **explicit**
   Reduces ambiguity, increases future-proof-ness
2. Measurement should be **in-band**
   Ensures measurement traffic treated similar to"real" traffic
3. The measurement **consumer** should **bear the cost**
   Shifts state and per-packet overhead offline, increases deployability
4. The measurement **provider** should **retain control**
   Ensures users know how measurable their traffic is
5. Measurement must be **visible**

6. Measurement should be **cooperative**

Principles for Measurability

1. Measurement should be **explicit**
   Reduces ambiguity, increases future-proof-ness
2. Measurement should be **in-band**
   Ensures measurement traffic treated similar to"real" traffic
3. The measurement **consumer** should **bear the cost**
   Shifts state and per-packet overhead offline, increases deployability
4. The measurement **provider** should **retain control**
   Ensures users know how measurable their traffic is
5. Measurement must be **visible**
   Increases transparency and trustworthiness in measurement
6. Measurement should be **cooperative**

Principles for Measurability

1. Measurement should be **explicit**
   Reduces ambiguity, increases future-proof-ness
2. Measurement should be **in-band**
   Ensures measurement traffic treated similar to"real" traffic
3. The measurement **consumer** should **bear the cost**
   Shifts state and per-packet overhead offline, increases deployability
4. The measurement **provider** should **retain control**
   Ensures users know how measurable their traffic is
5. Measurement must be **visible**
   Increases transparency and trustworthiness in measurement
6. Measurement should be **cooperative**
   Follows from P3/P4, leverages existing middle/endpoint tussle

Principles for Measurability

1. Measurement should be **explicit**
   Reduces ambiguity, increases future-proof-ness
2. Measurement should be **in-band**
   Ensures measurement traffic treated similar to"real" traffic
3. The measurement **consumer** should **bear the cost**
   Shifts state and per-packet overhead offline, increases deployability
4. The measurement **provider** should **retain control**
   Ensures users know how measurable their traffic is
5. Measurement must be **visible**
   Increases transparency and trustworthiness in measurement
6. Measurement should be **cooperative**
   Follows from P3/P4, leverages existing middle/endpoint tussle

Candidate Primitives
(see the paper)

- End to End Information §4.1
  - Host Identification §4.1.1
  - Timing §4.1.2
  - Arrival §4.1.3
  - Integrity §4.1.4
- Hop-Specific Information §4.2
  - Topology
  - Queue Performance
- Accumulated Path Information §4.3
  - Path Change Detection
  - Path Queue Delay

Candidate Primitives
(see the paper)

- End to End Information §4.1
  - Host Identification §4.1.1
  - Timing §4.1.2
  - Arrival §4.1.3
  - Integrity §4.1.4
- Hop-Specific Information §4.2
  - Topology
  - Queue Performance
- Accumulated Path Information §4.3
  - Path Change Detection
  - Path Queue Delay

focus of this talk

also covered briefly

Candidate Primitive: Timing Information §4.1.2

- TCP TSOPT (RFC 7313) is almost right...
  - ...but not designed for passive measurement
- Approach: add a T_{now}, T_{echo}, T_{Δ} tuple to packets:
  - T_{now} = timestamp in constant-rate clock
  - T_{echo} = last timestamp seen from peer
  - T_{Δ} = ticks in constant-rate clock since T_{echo} seen
- Resolution-overhead tradeoff: can be sent on $^1/_n$ packets.

Candidate Primitive: Timing Information §4.1.2

- TCP TSOPT (R ght...
  - ...but not designed for passive measurement

P2 In-Band
P5 Visible

- Approach: add a T_{now}, T_{echo}, T_{Δ} tuple to packets:
  - T_{now} = timestamp in constant-rate clock
  - T_{echo} = last timestamp seen from peer
  - T_{Δ} = ticks in constant-rate clock since T_{echo} seen
- Resolution-overhead tradeoff: can be sent on $^{1}/_{n}$ packets.

Candidate Primitive: Timing Information §4.1.2

- TCP TSOPT (R ght...
  - ...but not designed fo passive measurement

P2 In-Band
P5 Visible

- Approach: add a T_{now}, T_{echo}, T_{Δ} tuple to packets:
  - T_{now} = timestamp in constant-rate clock
  - T_{echo} en from peer
  - T_{Δ} = ticks in constant-rate clock since T_{echo} seen

P1 Explicit

- Resolution-overhead tradeoff: can be sent on $^1/_n$ packets.

Candidate Primitive: Timing Information §4.1.2

- TCP TSOPT (R ght...

P2 In-Band
P5 Visible

  - ...but not designed for passive measurement
- Approach: add a T_{now}, T_{echo}, T_{Δ} tuple to packets:
  - T_{now} = timestamp in constant-rate clock

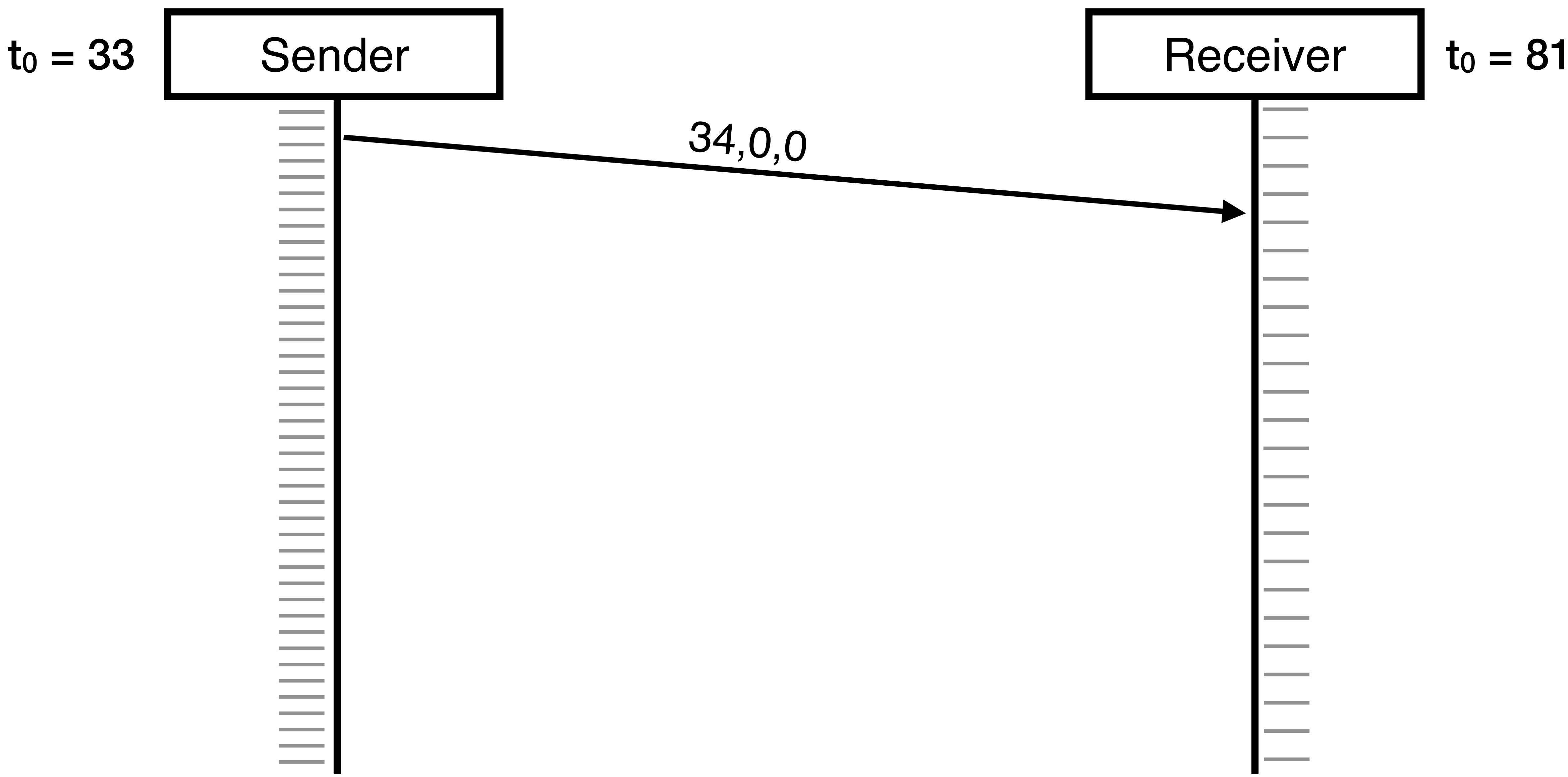
P1 Explicit

  - T_{echo} en from peer
  - T_{Δ} = ticks in constant-rate clock since T_{echo} seen
- Resolution-overhead tradeoff: can be sent on $^1/_n$ packets.

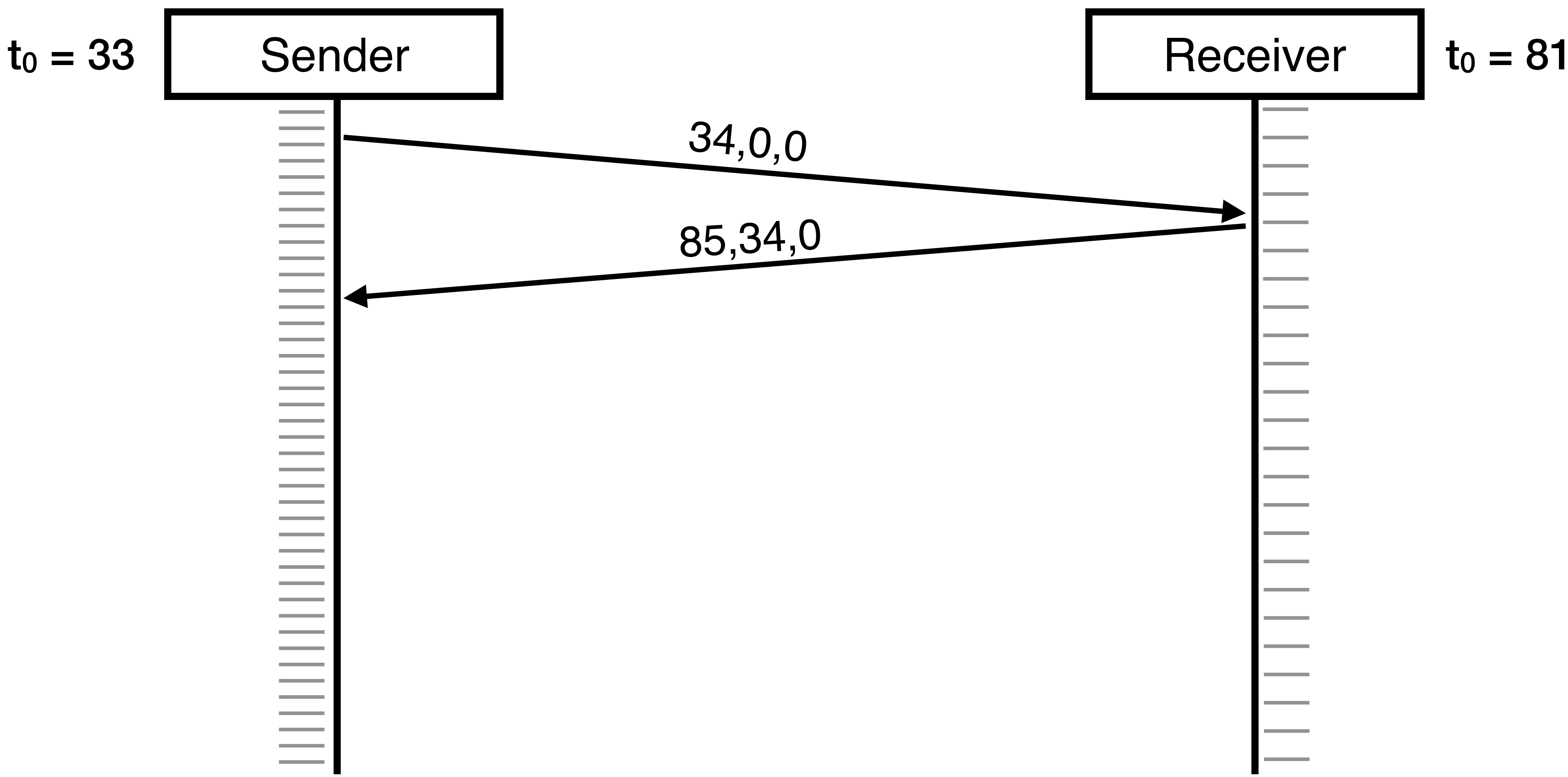
P4 Sender Control

Timing Information §4.1.2: RTT and Timing Measurement

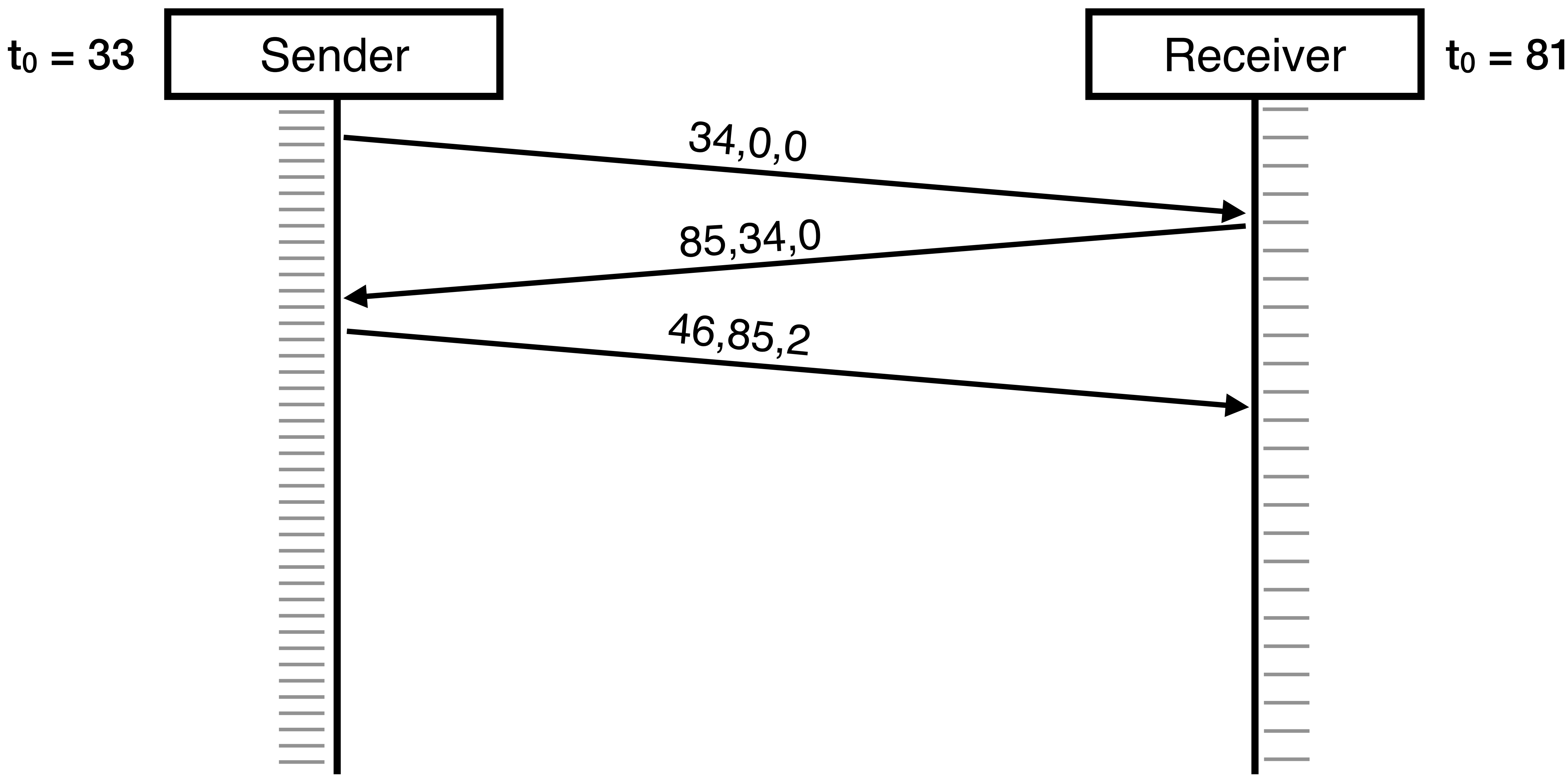
Timing Information §4.1.2: RTT and Timing Measurement



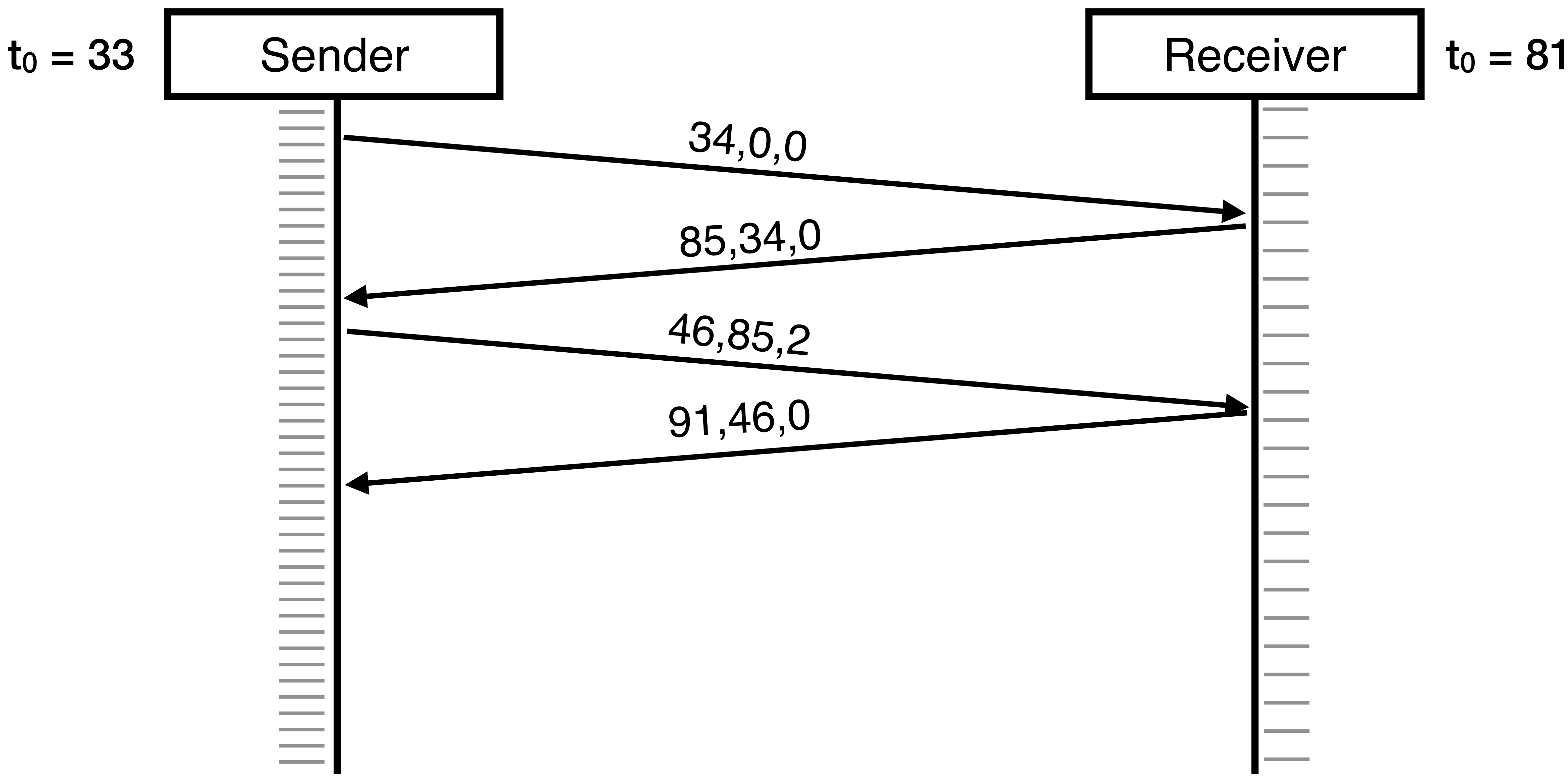
Timing Information §4.1.2: RTT and Timing Measurement



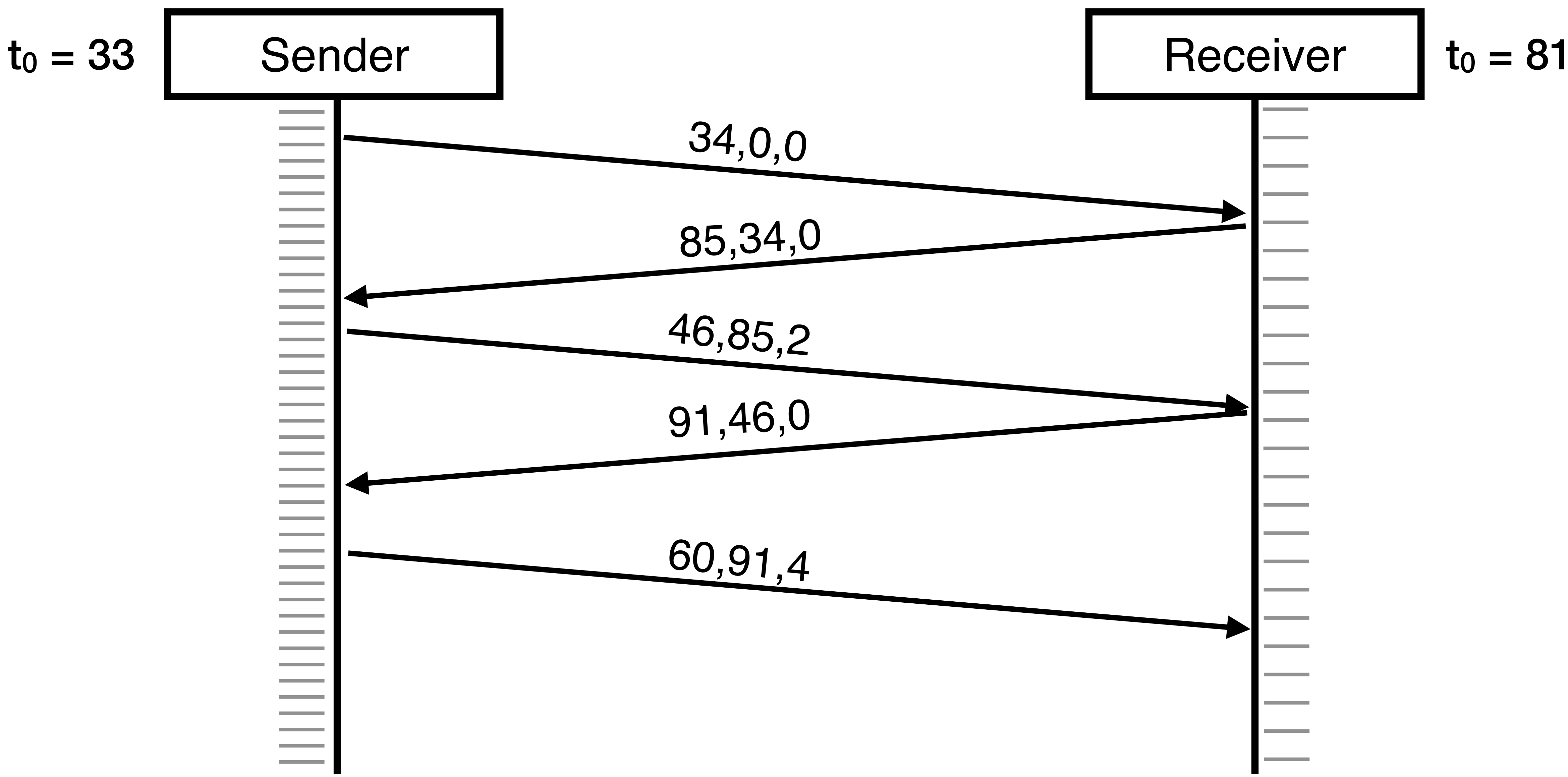
Timing Information §4.1.2: RTT and Timing Measurement



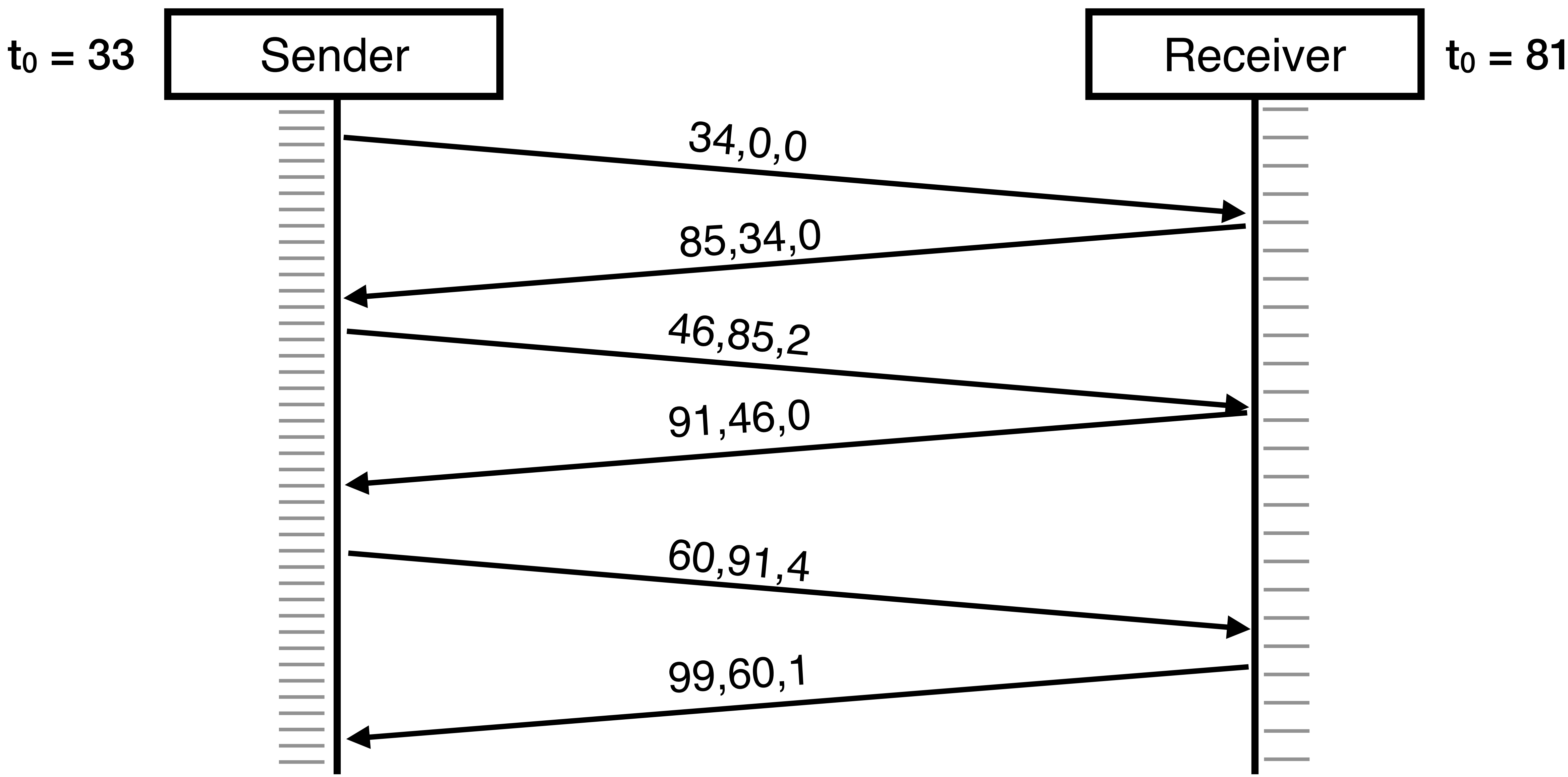
Timing Information §4.1.2: RTT and Timing Measurement



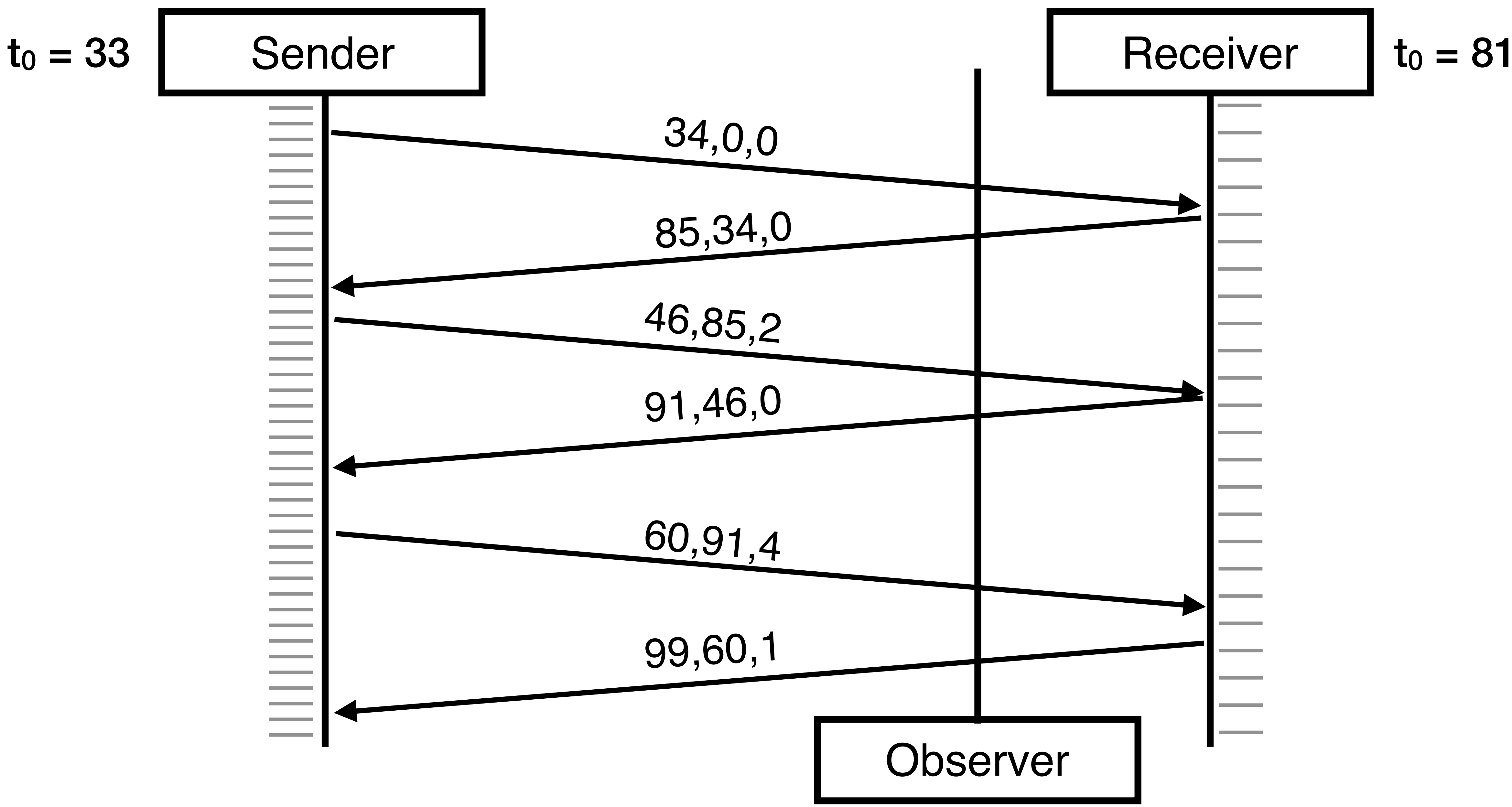
Timing Information §4.1.2: RTT and Timing Measurement



Timing Information §4.1.2: RTT and Timing Measurement



Timing Information §4.1.2: RTT and Timing Measurement



Candidate Primitive: Arrival Information §4.1.3

- Makes pattern of loss and reordering visible in a transport-independent way.
- Each sender maintains a counter N_{tx} per flow:
  - Increment N_{tx} by a randomly-chosen but increasing number for each packet sent.
  - Maintain running sum of received N_{tx} values as $\sum N_{echo}$.
  - Send $\{N_{tx}, \sum N_{echo}\}$ on every packet.

- (inspired by Savage et al. *TCP Congestion Control with a Misbehaving Receiver*. ACM CCR 29(5), Oct. 1999.)

Candidate Primitive: Arrival Information §4.1.3

- Makes pattern of loss and reordering visible in a transport-independent way.
- Each sender maintains a counter N_{tx} per flow:
  - Increment N_{tx} by a randomly-chosen but increasing number for each packet sent.
  - Maintain running sum of received N_{tx} values as $\sum N_{echo}$.
  - Send $\{N_{tx}, \sum N_{echo}\}$ on every packet.

P3 Consumer Cost

- (inspired by Savage et al. *TCP Congestion Control with a Misbehaving Receiver*. ACM CCR 29(5), Oct. 1999.)

Candidate Primitive: Arrival Information §4.1.3

- Makes pattern of loss and reordering visible in a transport-independent way.

P1 Explicit
P5 Visible

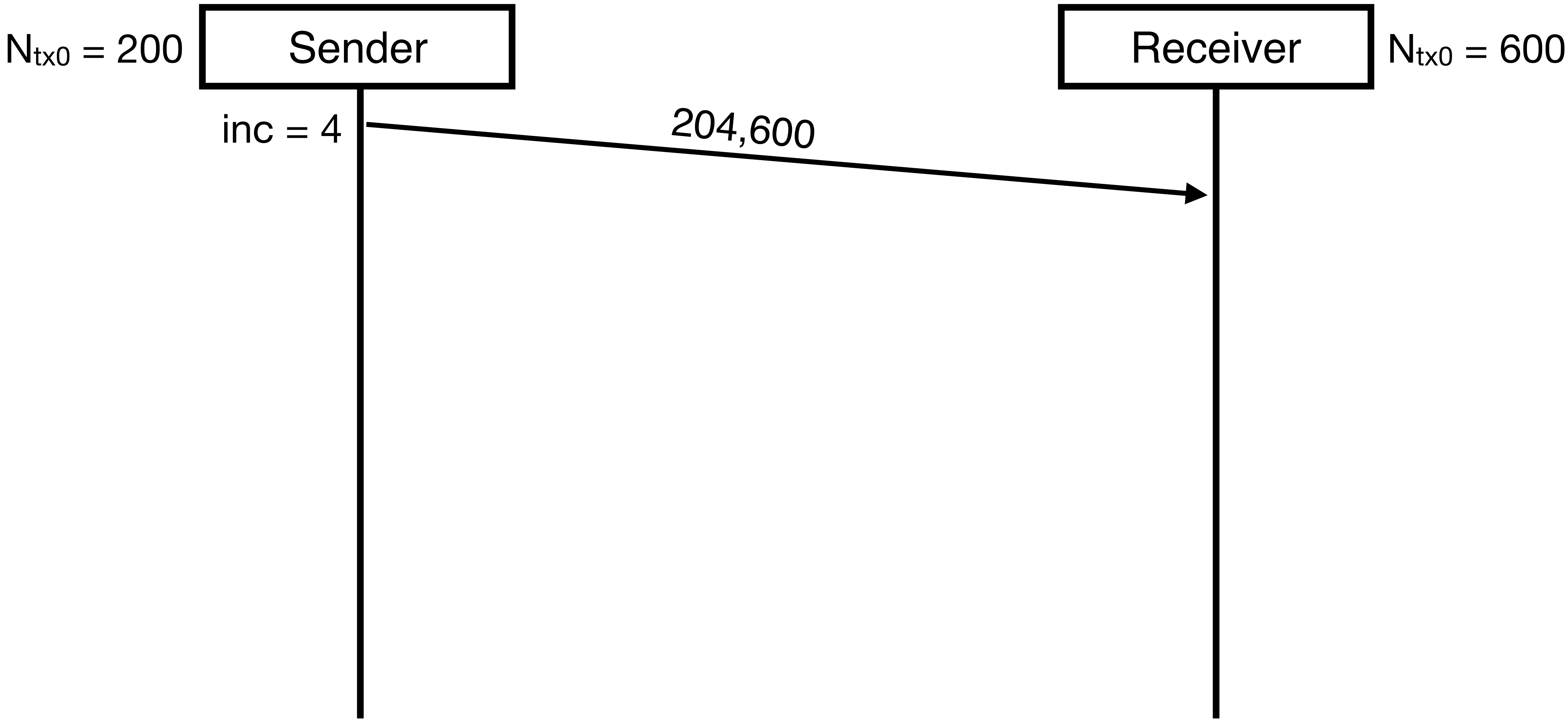
- Each sender maintains a counter N_{tx} per flow:
  - Increment N_{tx} by a randomly-chosen but increasing number for each packet sent.
  - Maintain running sum of received N_{tx} values as $\sum N_{echo}$.
  - Send $\{N_{tx}, \sum N_{echo}\}$ on every packet.

P3 Consumer Cost

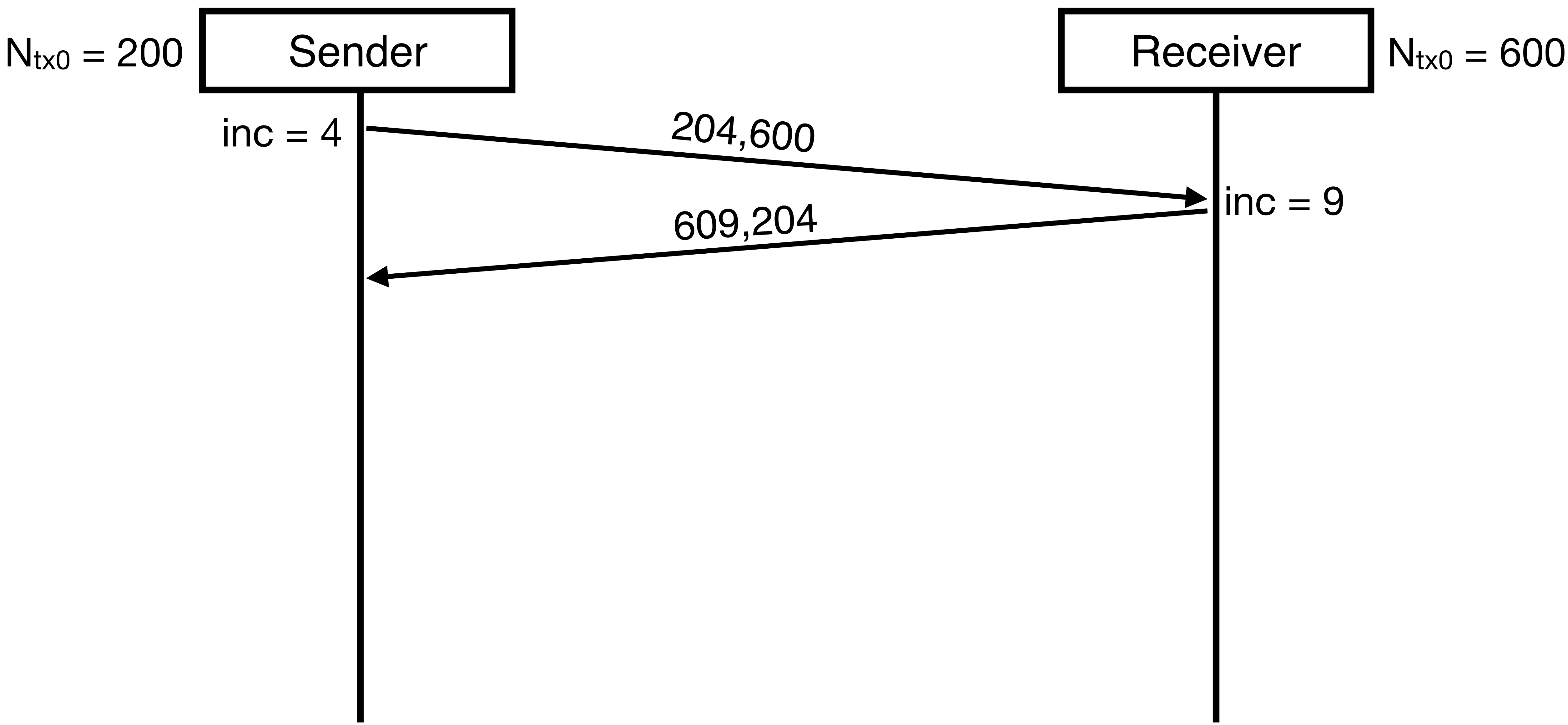
- (inspired by Savage et al. *TCP Congestion Control with a Misbehaving Receiver*. ACM CCR 29(5), Oct. 1999.)

Arrival Information §4.1.3: General Case

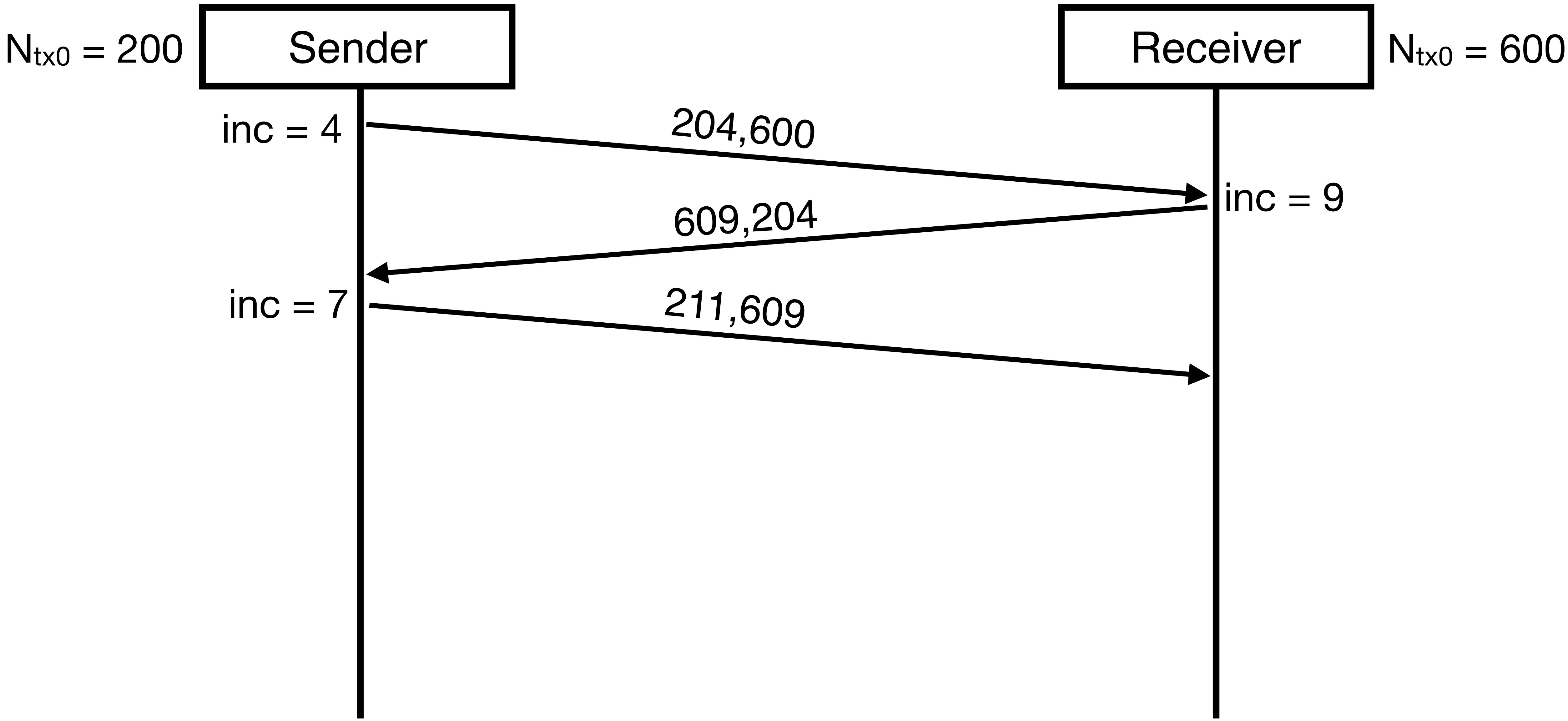
Arrival Information §4.1.3: General Case



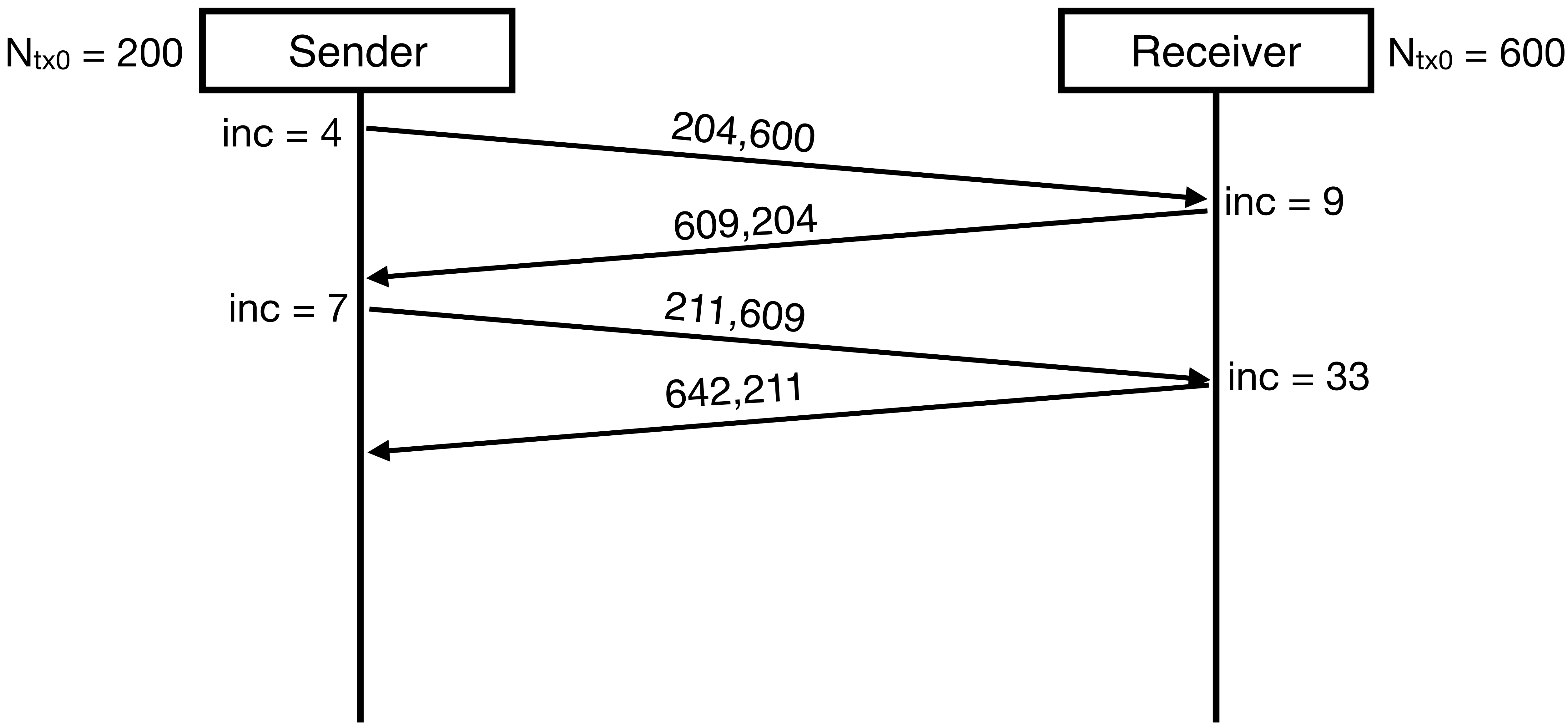
Arrival Information §4.1.3: General Case



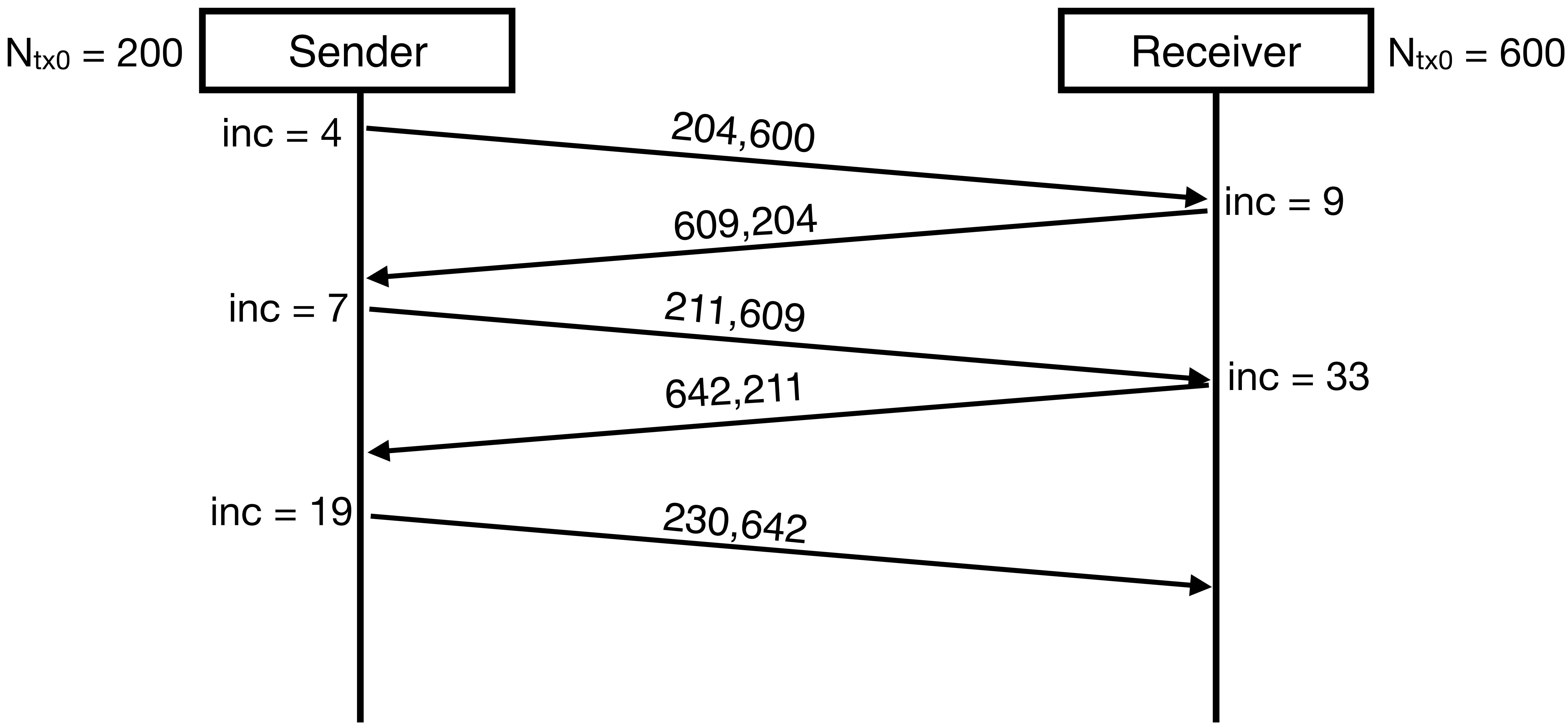
Arrival Information §4.1.3: General Case



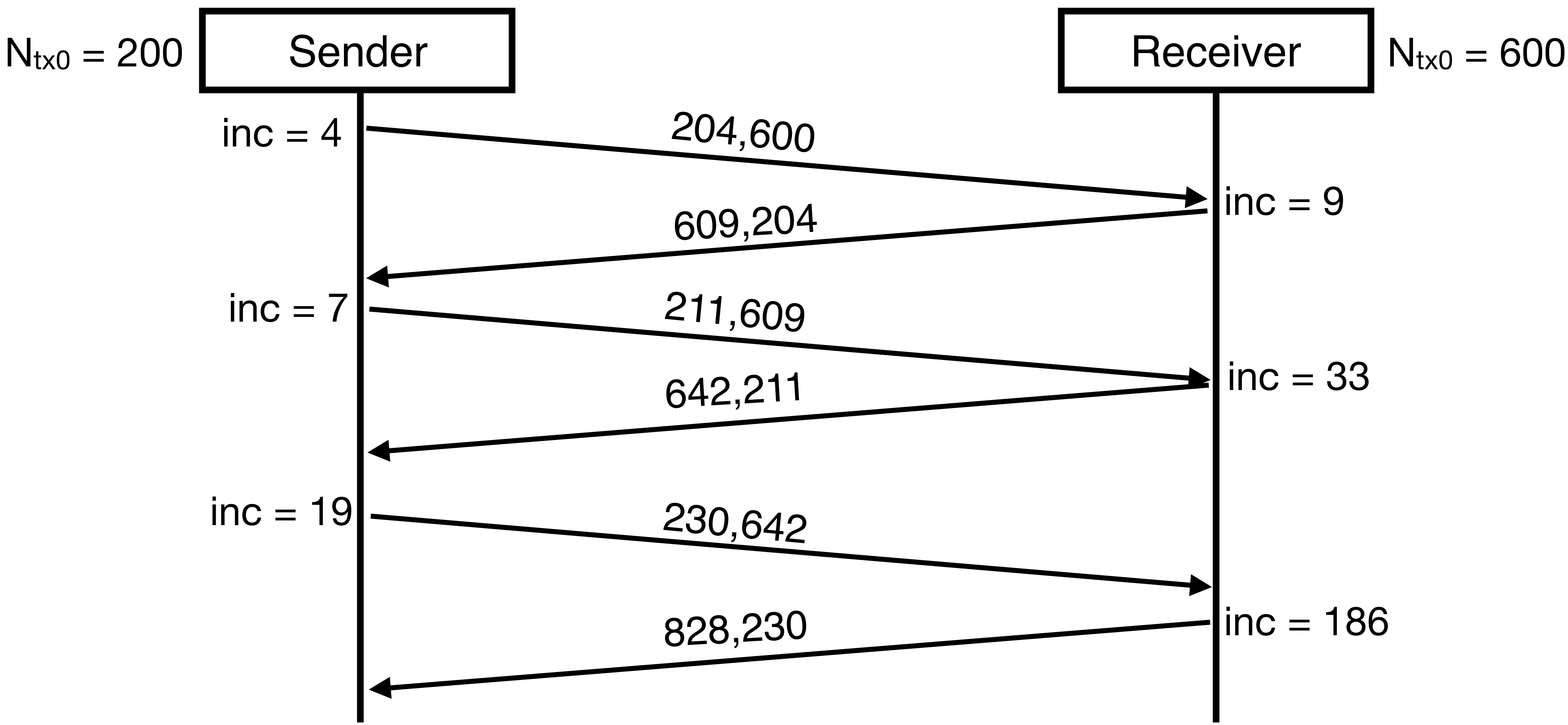
Arrival Information §4.1.3: General Case



Arrival Information §4.1.3: General Case

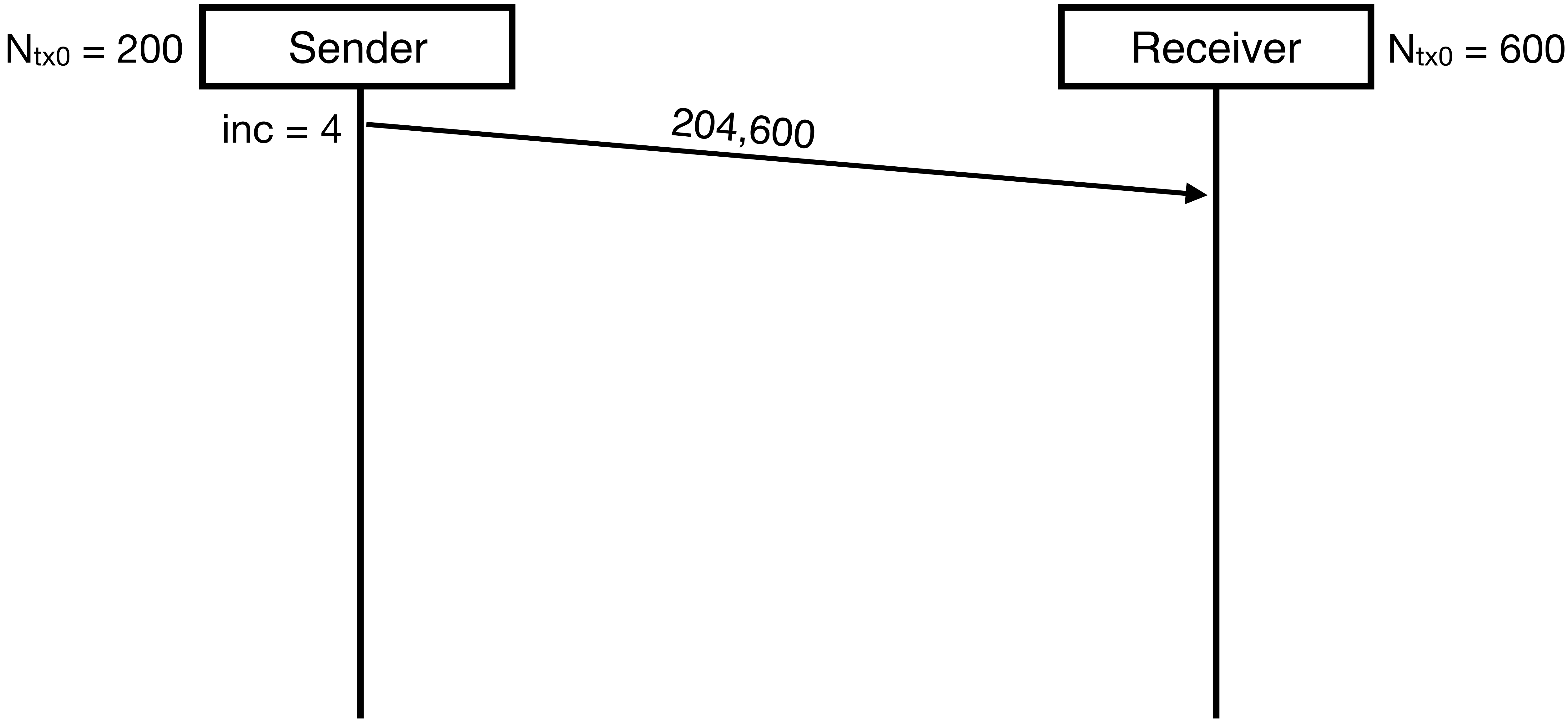


Arrival Information §4.1.3: General Case

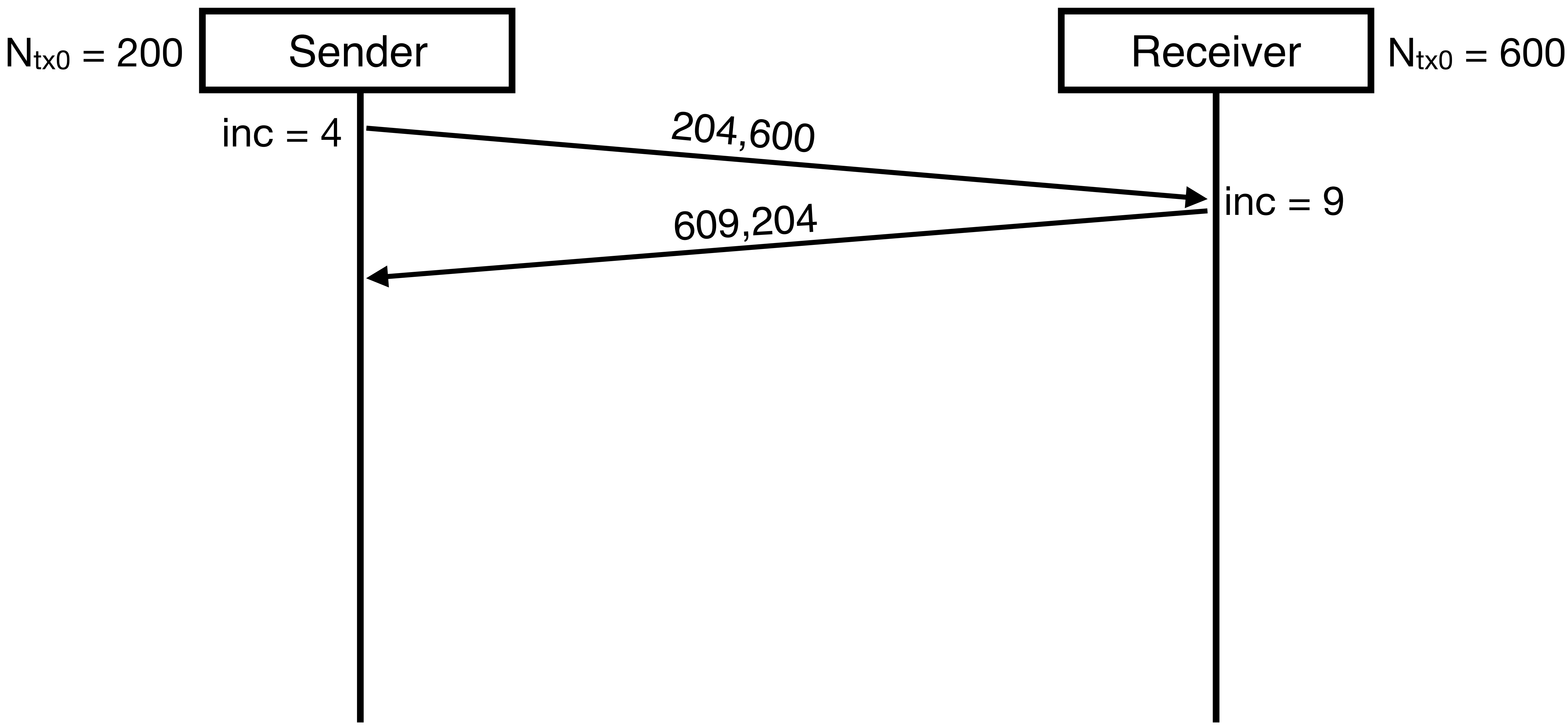


Arrival Information §4.1.3: Loss Detection

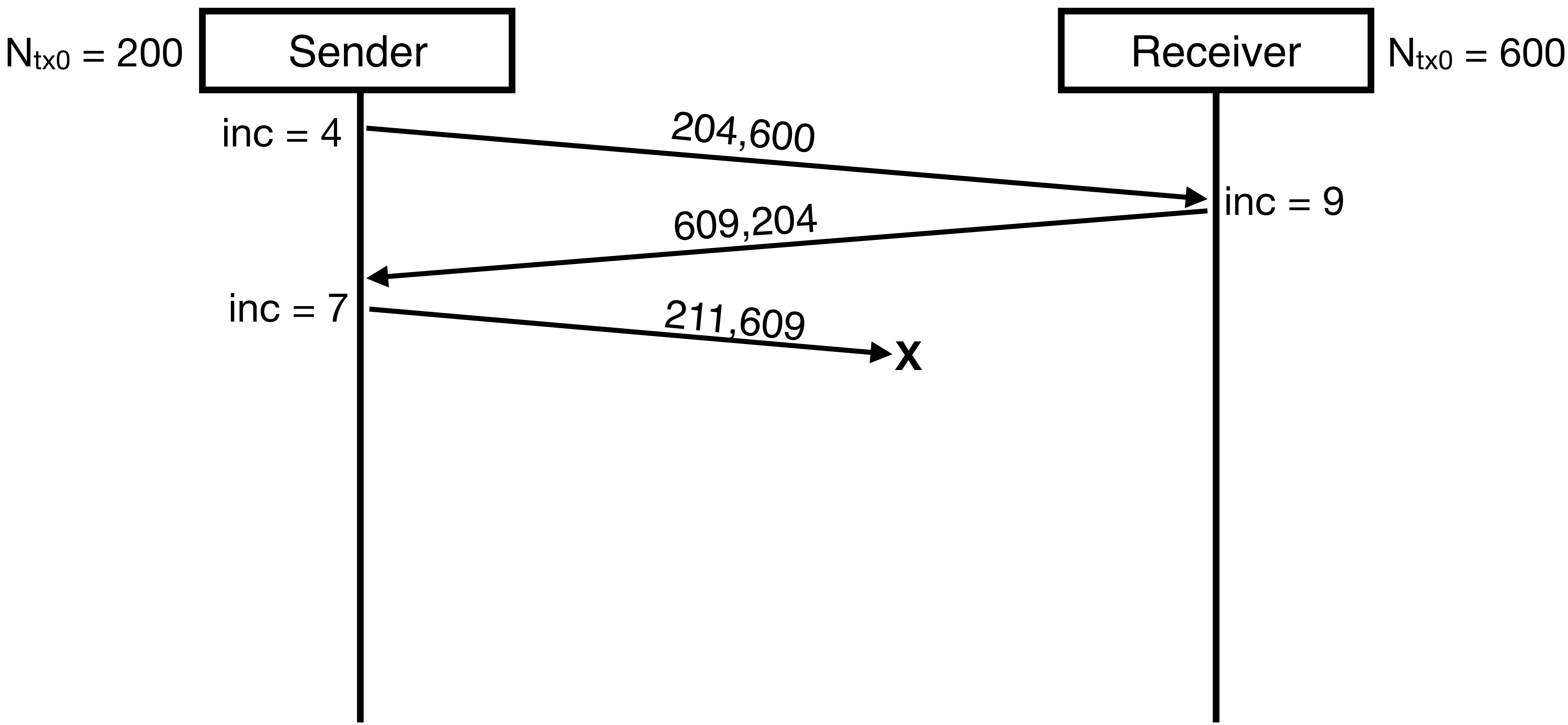
Arrival Information §4.1.3: Loss Detection



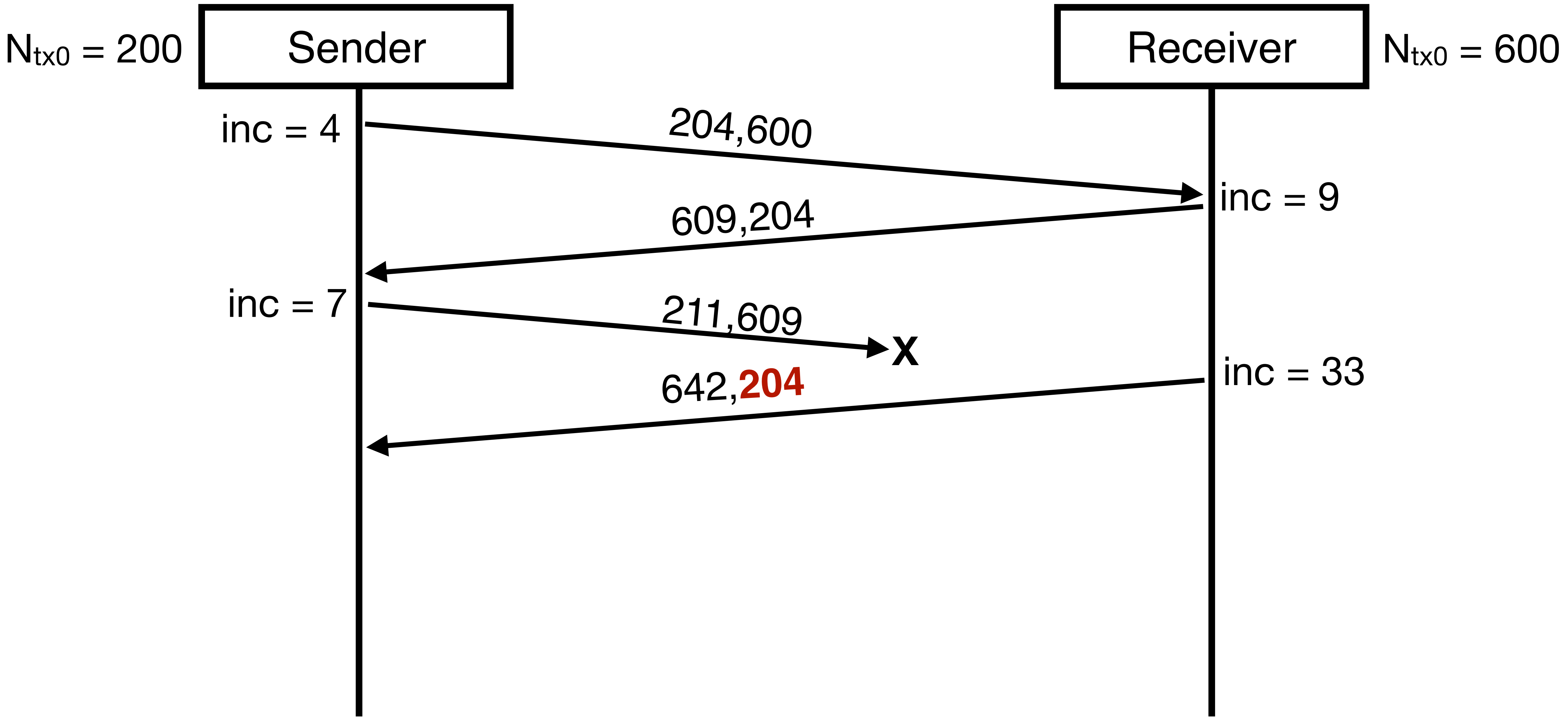
Arrival Information §4.1.3: Loss Detection



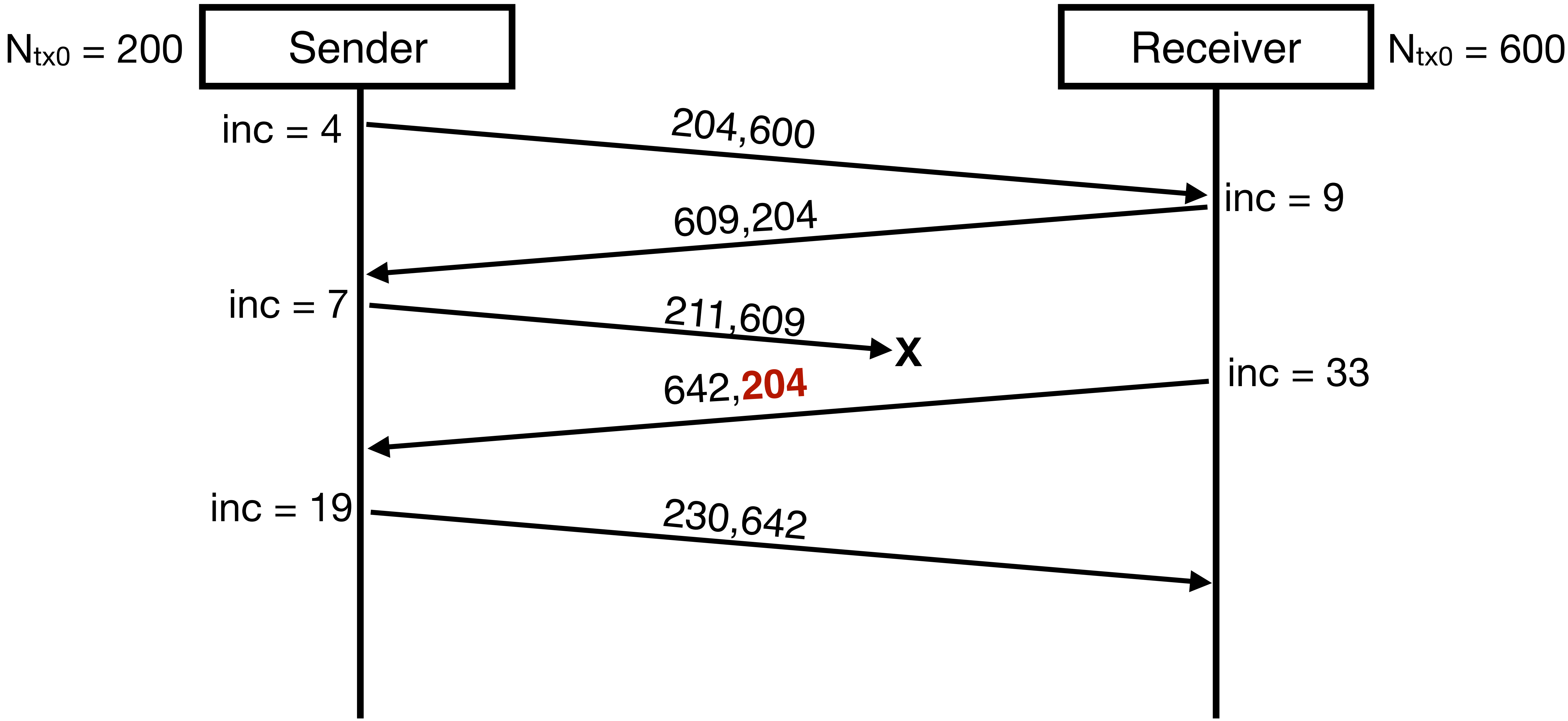
Arrival Information §4.1.3: Loss Detection



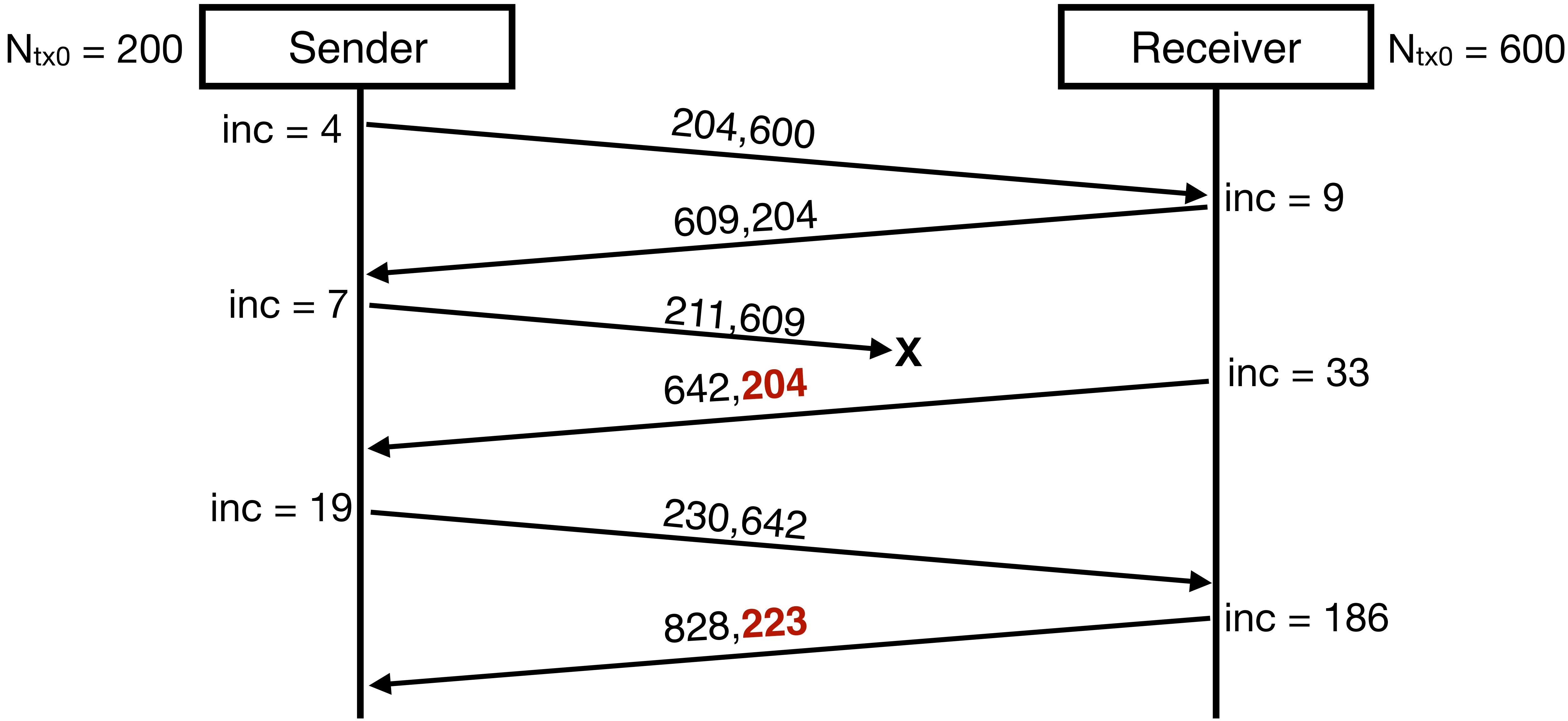
Arrival Information §4.1.3: Loss Detection



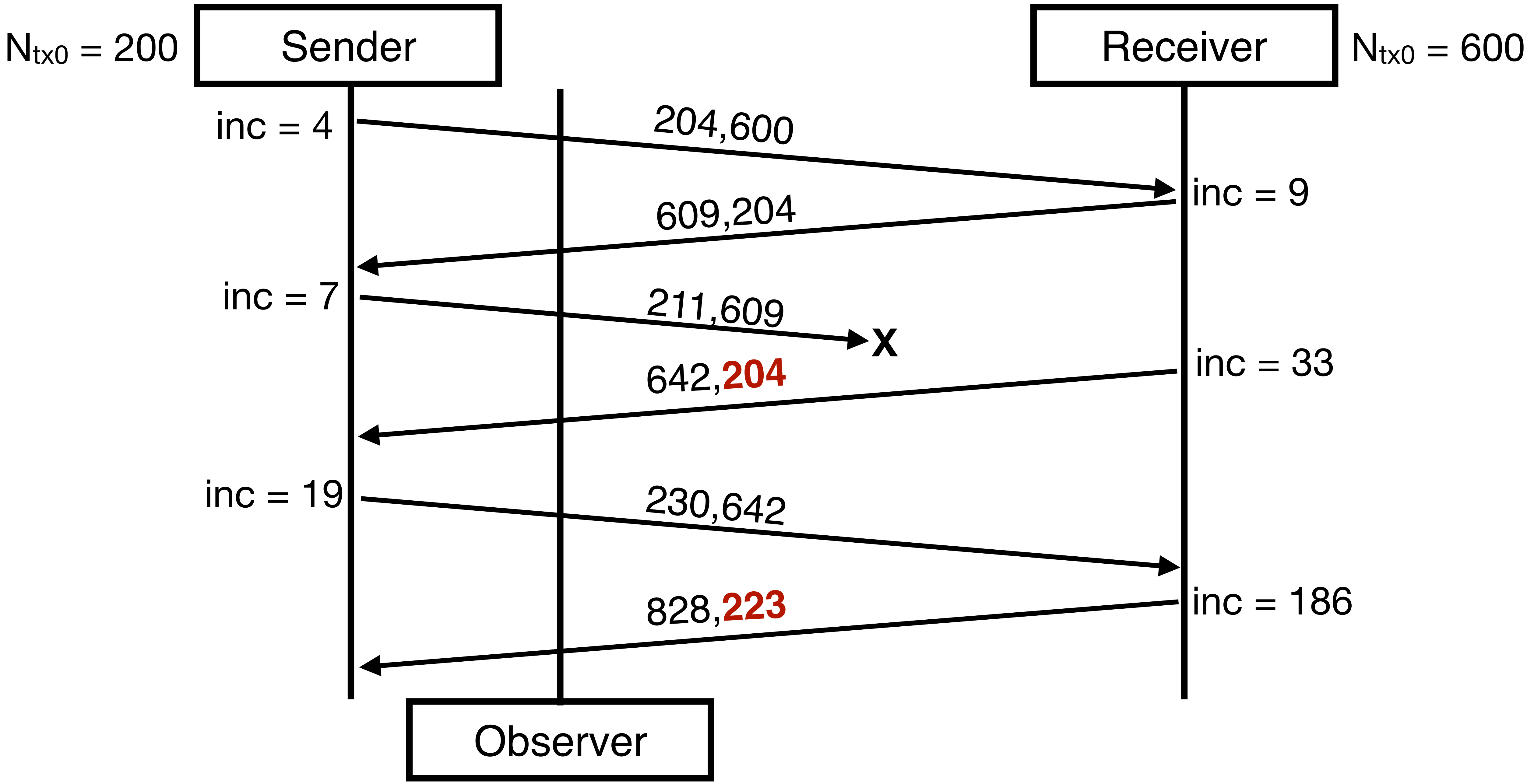
Arrival Information §4.1.3: Loss Detection



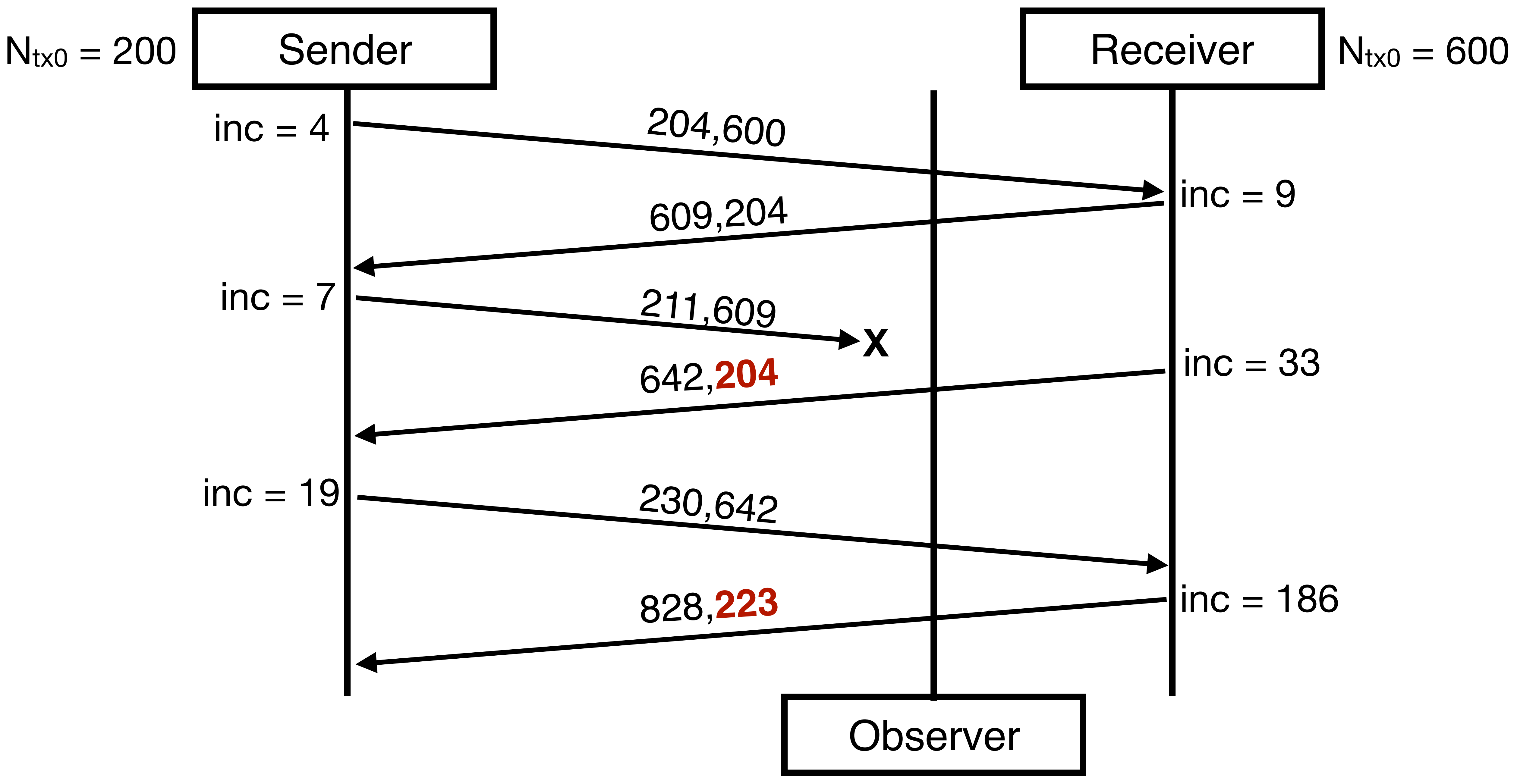
Arrival Information §4.1.3: Loss Detection



Arrival Information §4.1.3: Loss Detection



Arrival Information §4.1.3: Loss Detection

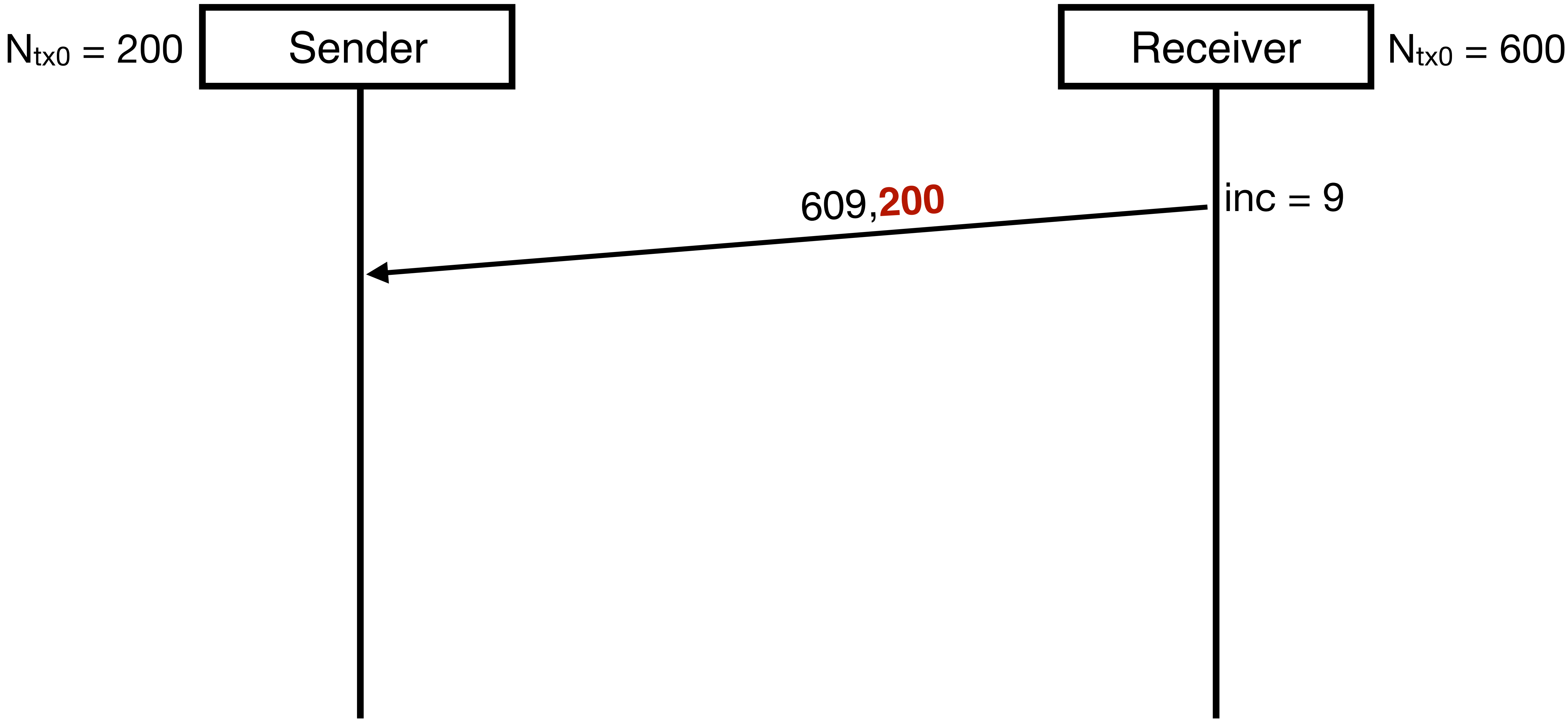


Arrival Information §4.1.3: Reordering Detection

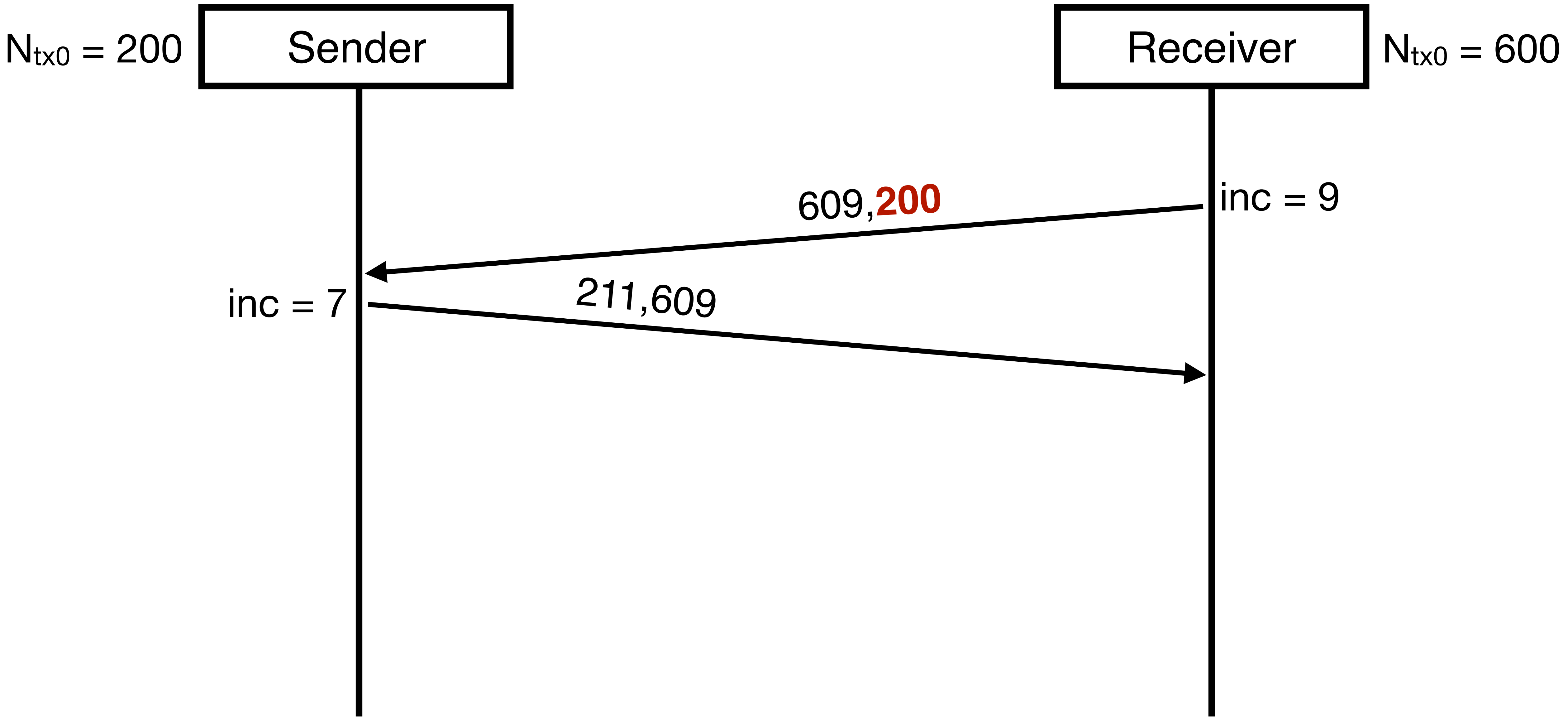
N_{tx0} = 200

N_{tx0} = 600

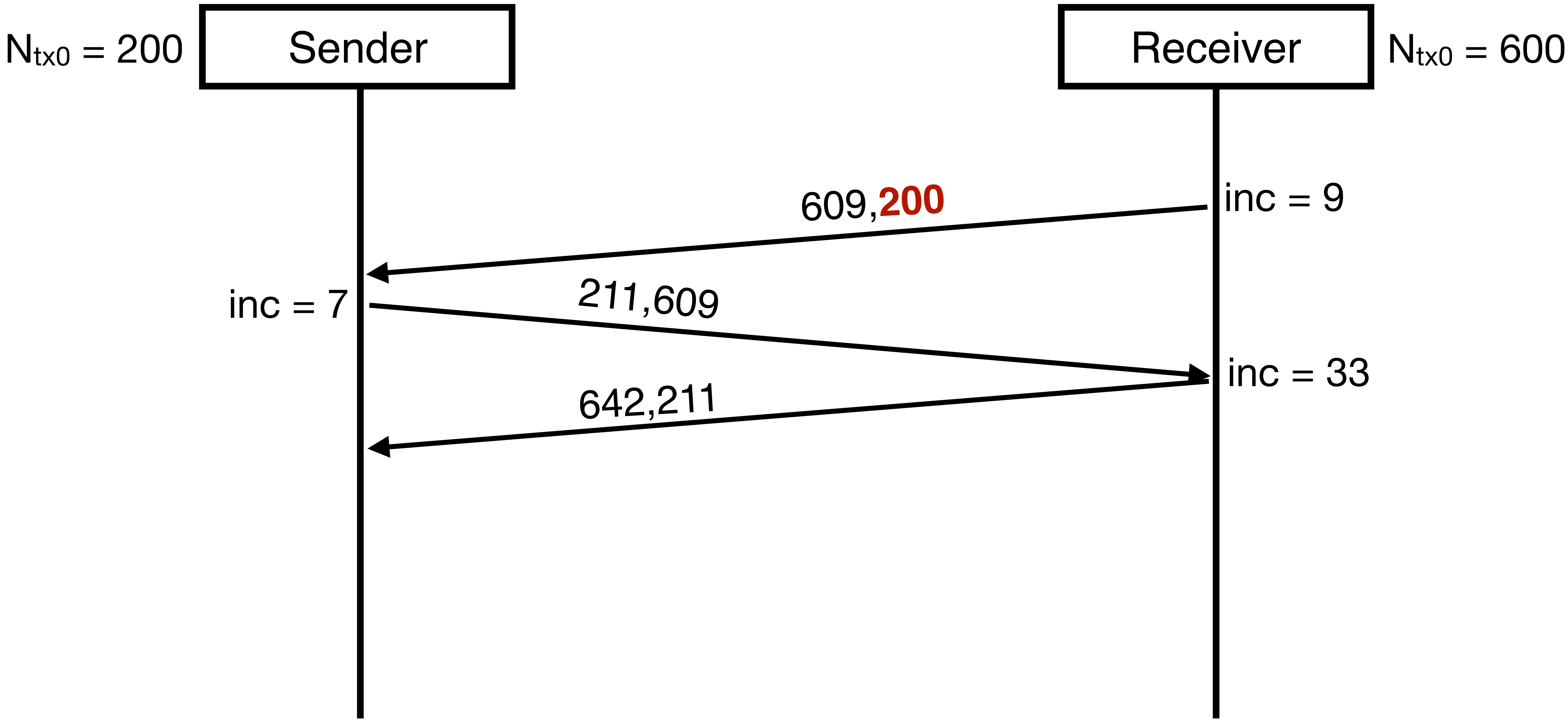
Arrival Information §4.1.3: Reordering Detection



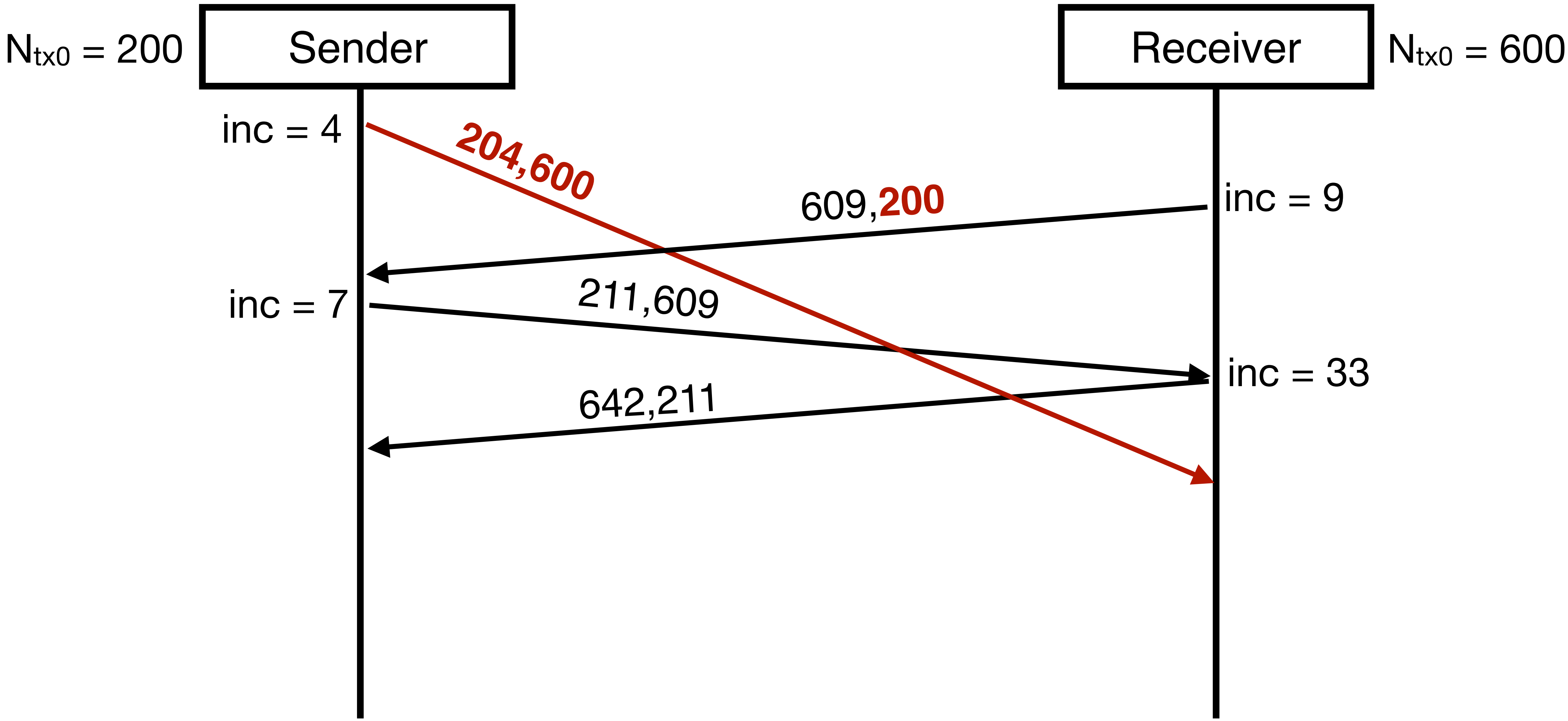
Arrival Information §4.1.3: Reordering Detection



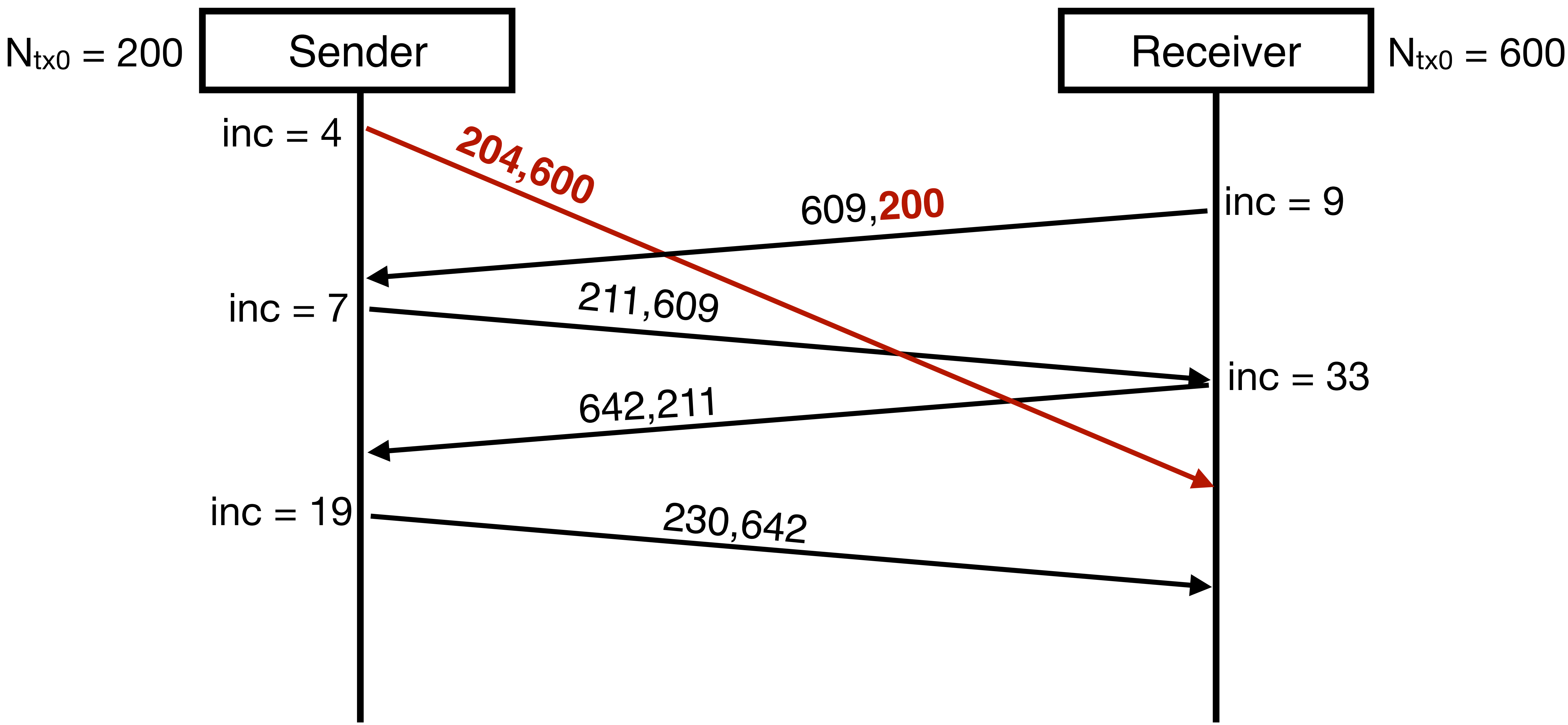
Arrival Information §4.1.3: Reordering Detection



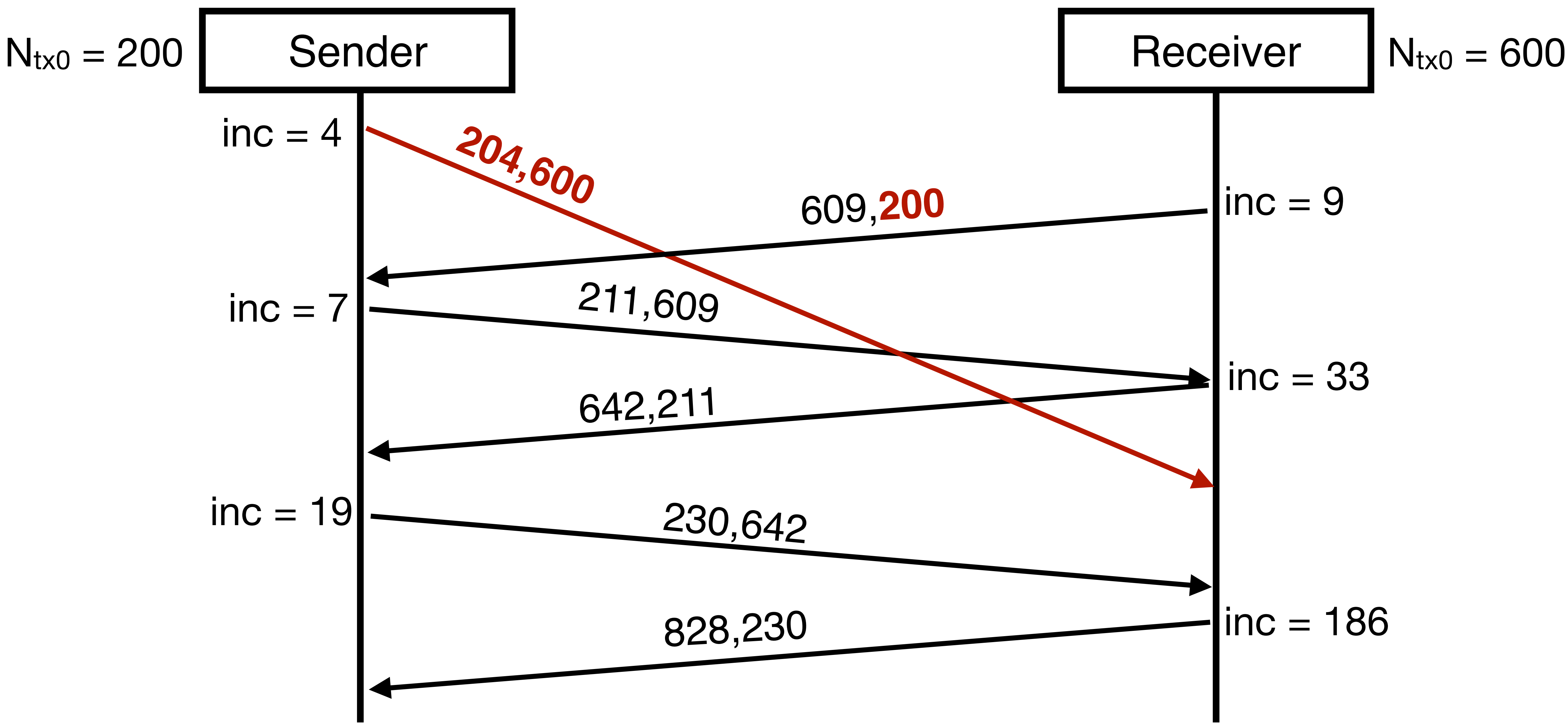
Arrival Information §4.1.3: Reordering Detection



Arrival Information §4.1.3: Reordering Detection



Arrival Information §4.1.3: Reordering Detection



Candidate Primitive: Probabilistic and Triggered Stamping §4.2

- Request for information to be added by router
  - at TTL *n* or with probability *p*
  - Performance diagnostics: $\{T_{now}, D_{queue}, C_{queue}\}$ tuple: replaces high-load queueing delay measurement.
  - Topology discovery: $\{AS, ID, IP_{in}, IP_{out}\}$ tuple: explicit replacement for tracreoute.

- Useful in intradomain applications, with migration to interdomain usage.

Candidate Primitive: Probabilistic and Triggered Stamping §4.2

- Request for information to be added by router
  - at TTL *n* or with probability *p*
  - Performance diagnostics: $\{T_{now}, D_{queue}, C_{queue}\}$ tuple: replaces high-load queueing delay measurement.
  - Topology discovery: $\{AS, ID, IP_{in}, IP_{out}\}$ tuple: explicit replacement for tracreoute.

P1 Explicit
P5 Visible

- Useful in intradomain applications, with migration to interdomain usage.

Candidate Primitive: Probabilistic and Triggered Stamping §4.2

P6 Cooperative

- Request for information to be added by router
    - at TTL *n* or with probability *p*
    - Performance diagnostics: $\{T_{now}, D_{queue}, C_{queue}\}$ tuple: replaces high-load queueing delay measurement.
    - Topology discovery: $\{AS, ID, IP_{in}, IP_{out}\}$ tuple: explicit replacement for tracreoute.

P1 Explicit
P5 Visible

- Useful in intradomain applications, with migration to interdomain usage.

Conclusions

- Network **measurement is critical.** We need better tools, and better tools **need better support from the network.**
- Propose guiding **principles** for viable measurement.
- Demonstrate candidate **primitives** that address long-standing, important real-world measurement problems.
- Position paper: spur discussion, debate, and **inform protocol development.**

- **Read the paper:** https://ccronline.sigcomm.org/wp-content/uploads/2017/05/acmdl17-60.pdf

This work supported in part by National Science Foundation (NSF) grants CNS-1213155, CNS-1213157 and CNS-1564329; the European Commission (EC) under grant agreement H2020-688421 (MAMI); and the Swiss State Secretariat for Education, Research, and Innovation under contract 15.0268. This work represents the position of the authors and not of the NSF, EC, or the Swiss or U.S.governments.