An Analytical Model for Progressive Mesh Streaming

黃瑋璨

新加坡國立大學

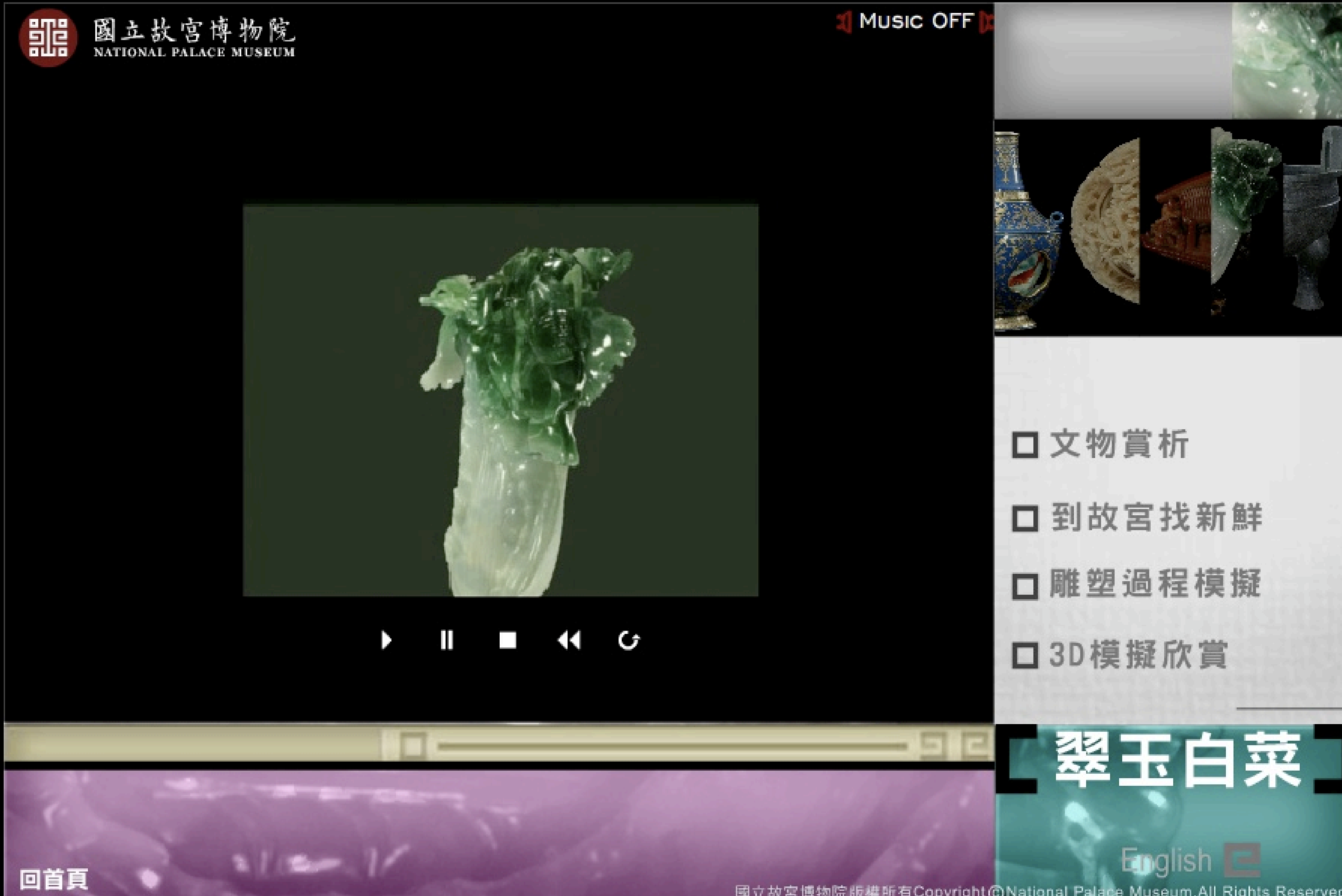
WEI TSANG **OOI**

National University of Singapore

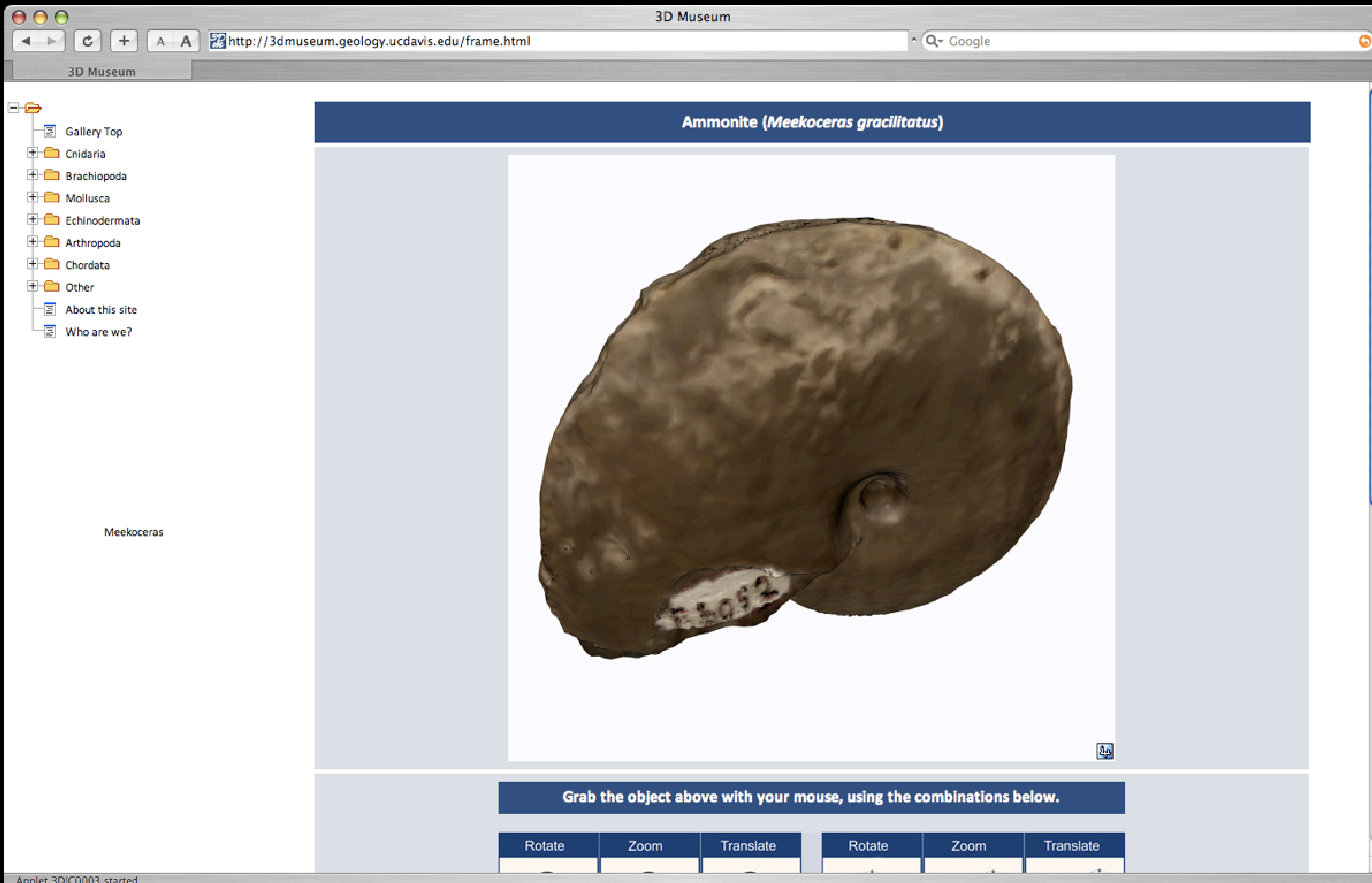
joint work with

Cheng Wei
National University of Singapore

Sebastian Mondet
Romulus Grigoras
Geraldine Morin
IRIT, University of Toulouse



國立故宮博物院
NATIONAL PALACE MUSEUM
Music OFF
文物賞析
到故宮找新鮮
雕塑過程模擬
3D模擬欣賞
翠玉白菜
English
回首頁
國立故宮博物院版權所有Copyright©National Palace Museum,All Rights Reserved.

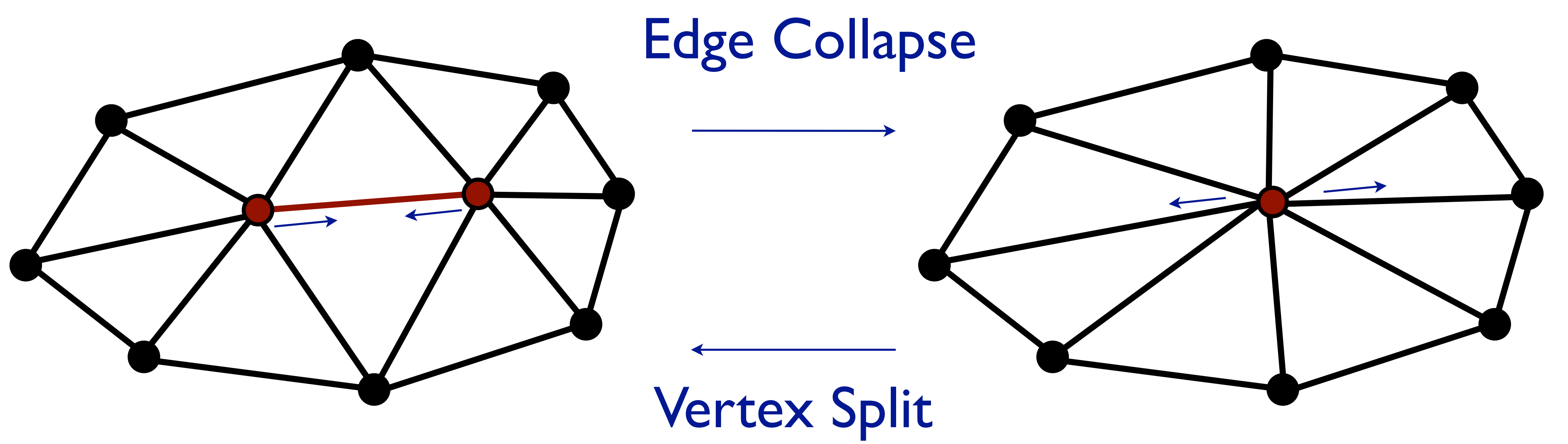

3D Museum
http://3dmuseum.geology.ucdavis.edu/frame.html
Google
3D Museum
Gallery Top
Cnidaria
Brachiopoda
Mollusca
Echinodermata
Arthropoda
Chordata
Other
About this site
Who are we?
Meekoceras
Ammonite (*Meekoceras gracilitatus*)
Grab the object above with your mouse, using the combinations below.
Rotate
Zoom
Translate
Rotate
Zoom
Translate
Applet 3DiC0003 started


DOMINUS

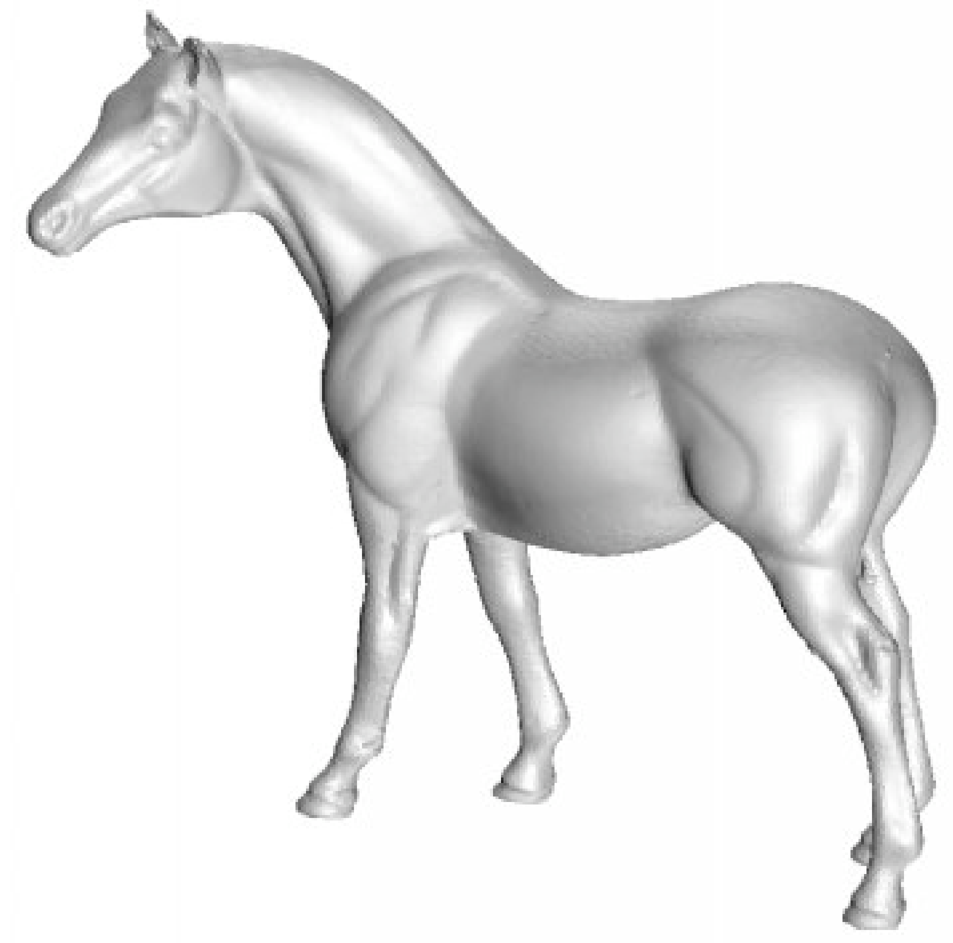
10 MB

2 GB

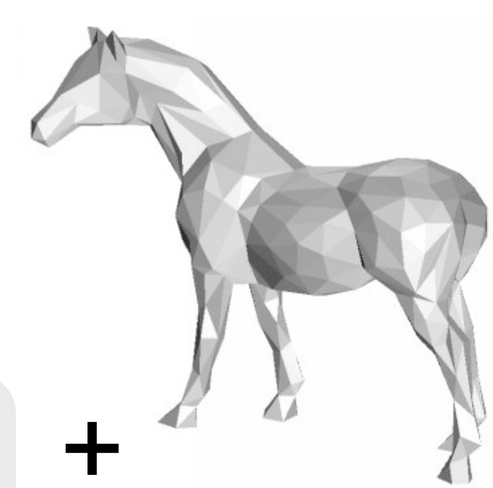
Hoppe's Progressive Mesh



At the sender

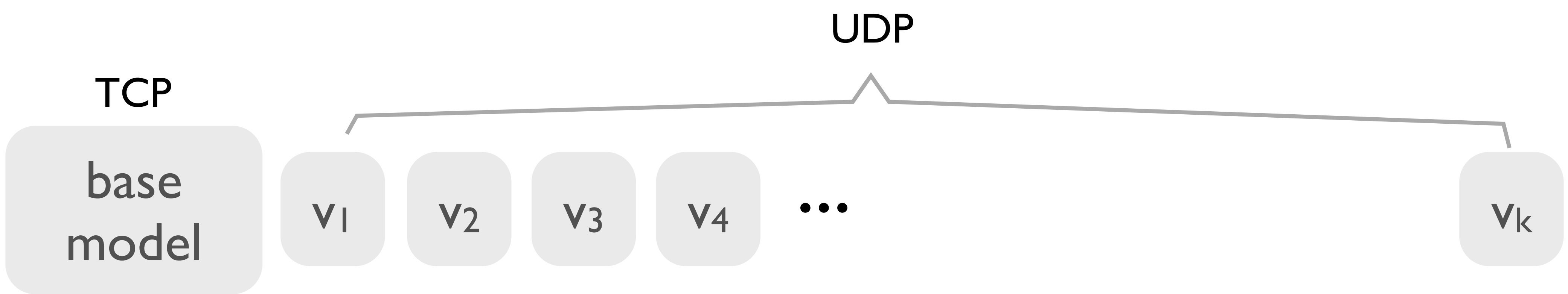


=



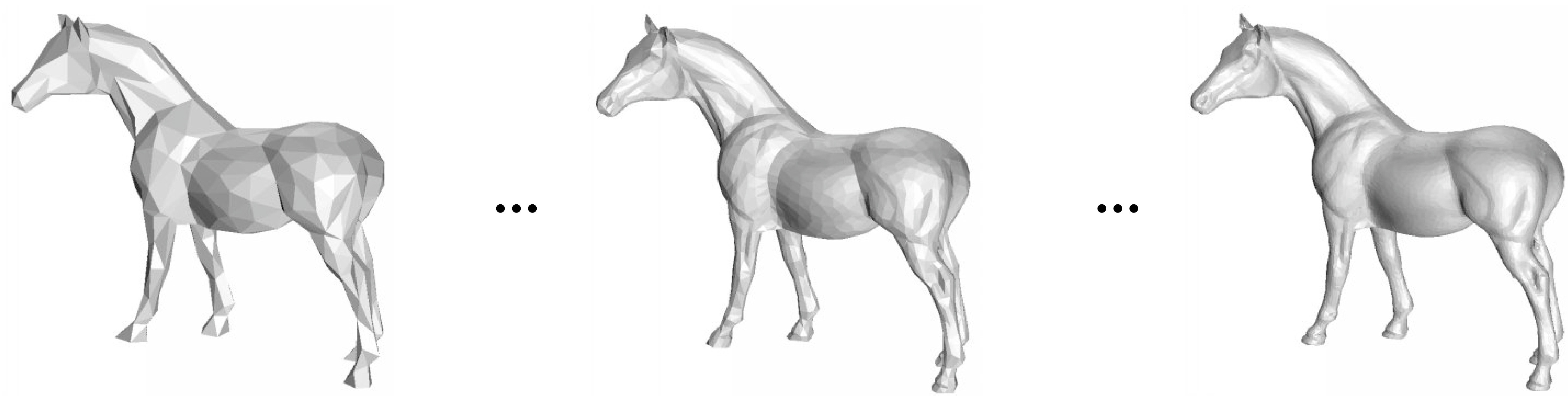
v_k … v_4 v_3 v_2 v_1 + base model

Transmission



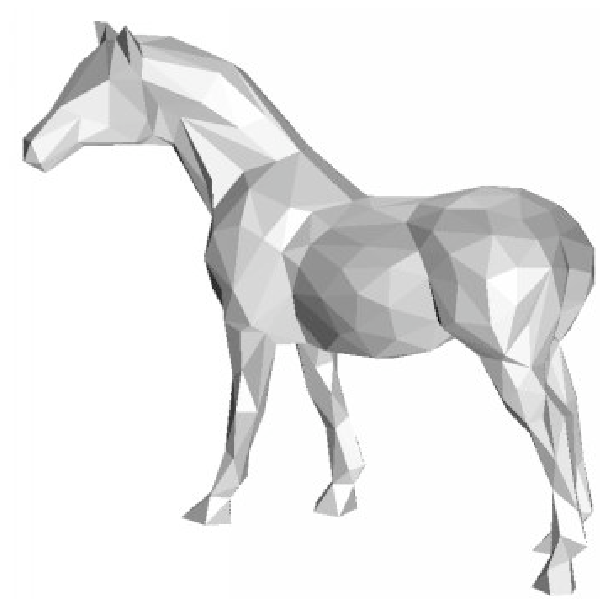
At the receiver

base model v_1 v_2 v_3 v_4 … v_k

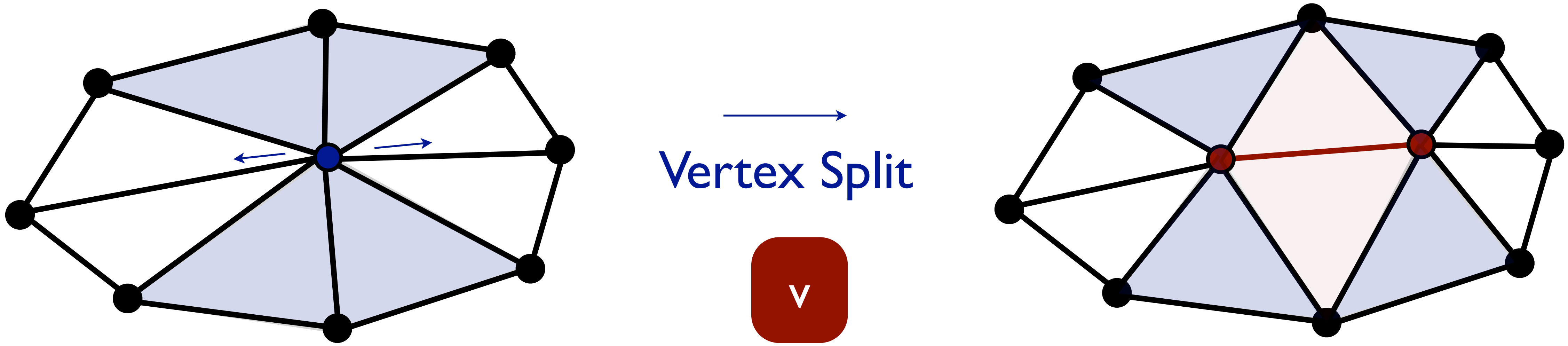


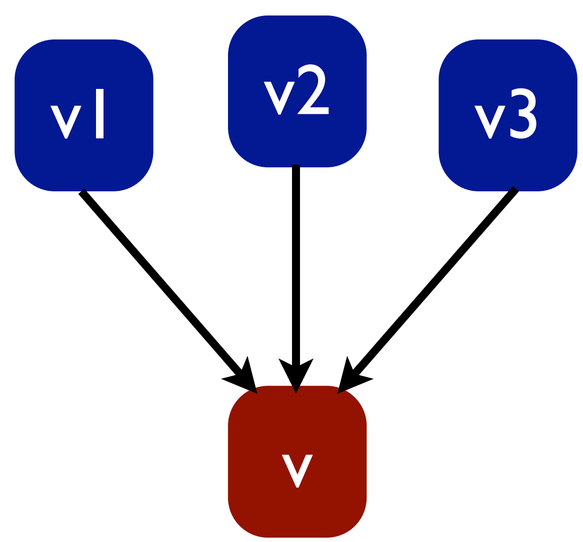
What happen if some data is lost?

base model v_1 v_2 $\cancel{v_3}$ v_4 ...

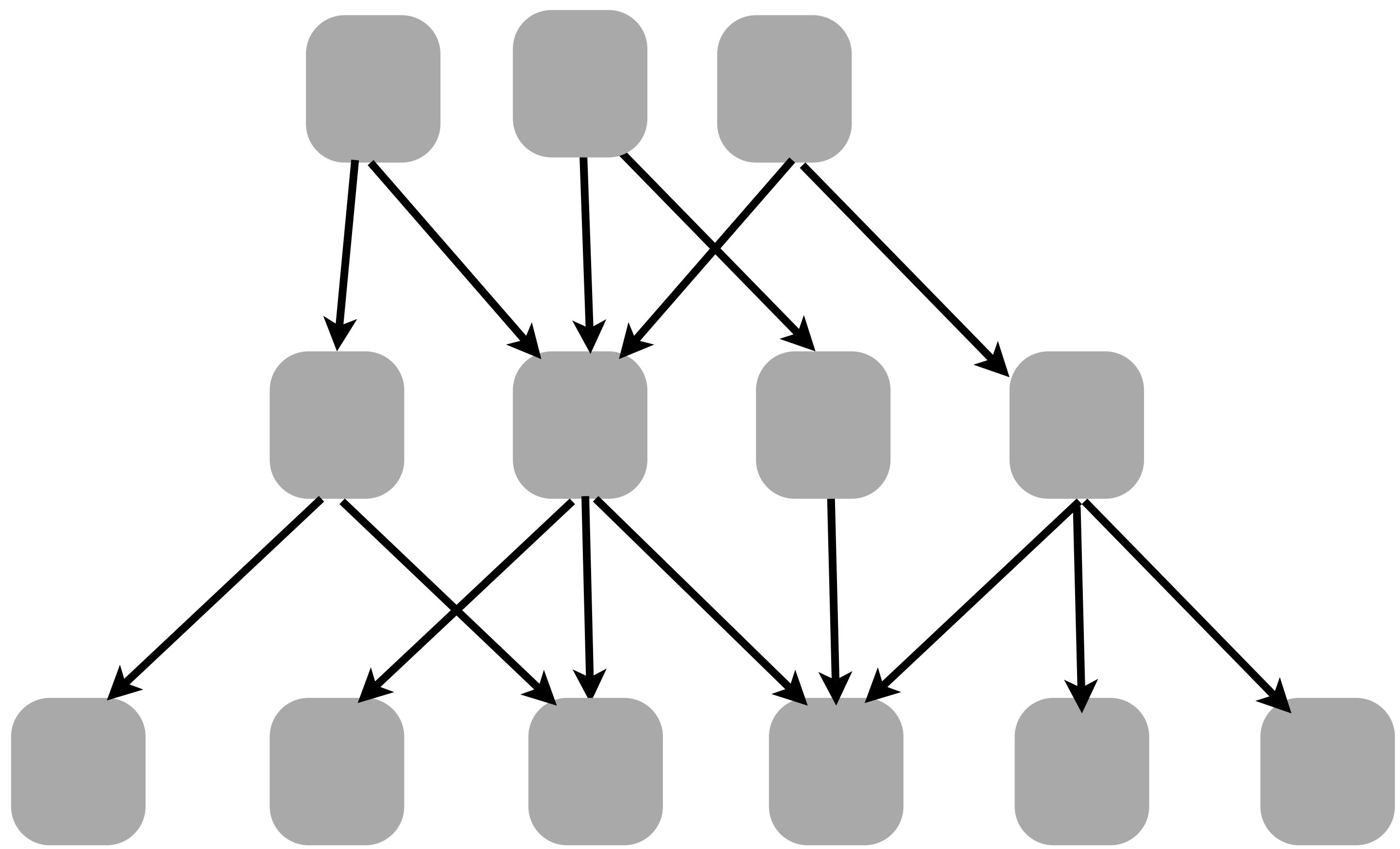


... **?**

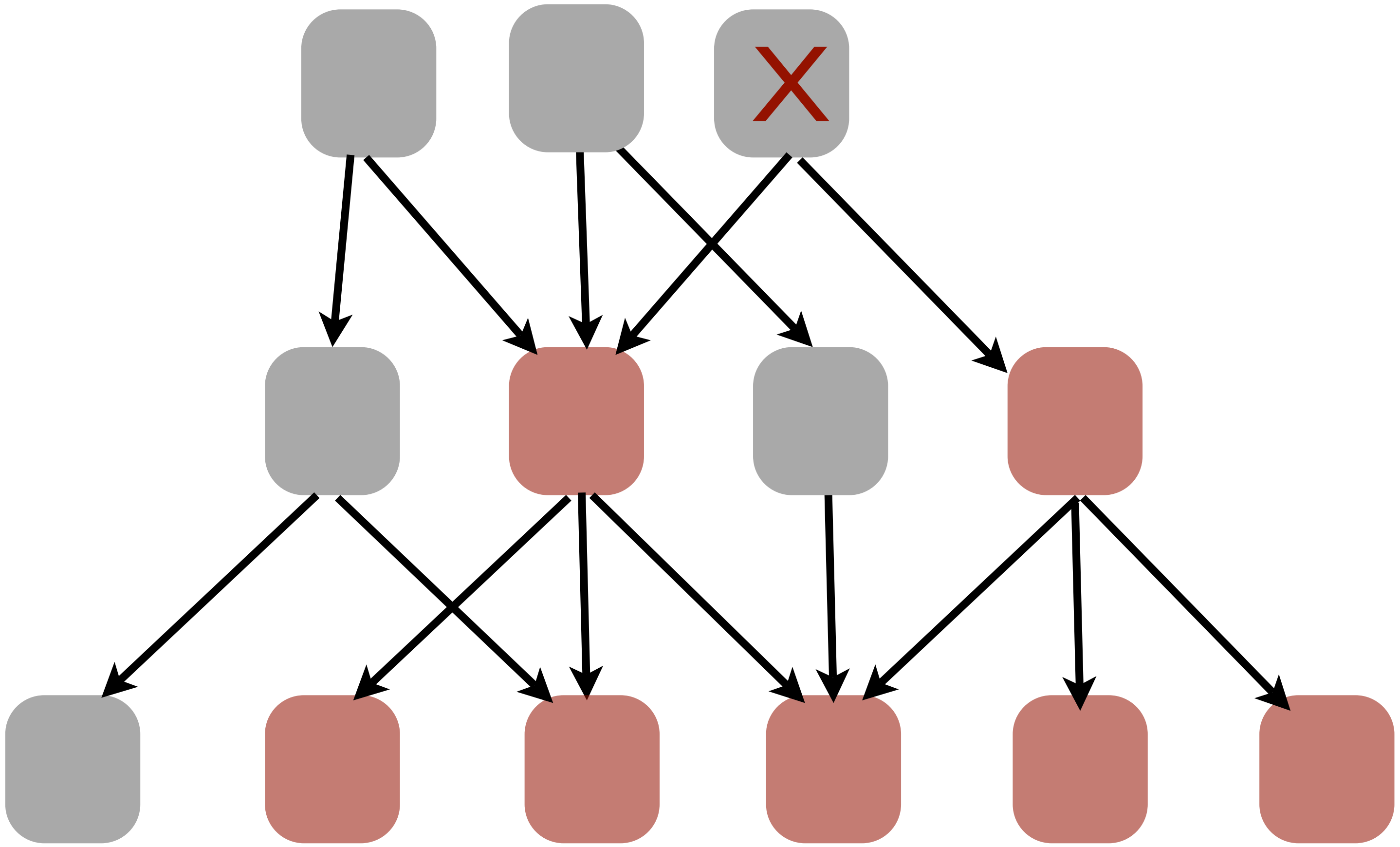

Vertex Split
v


v1
v2
v3
v

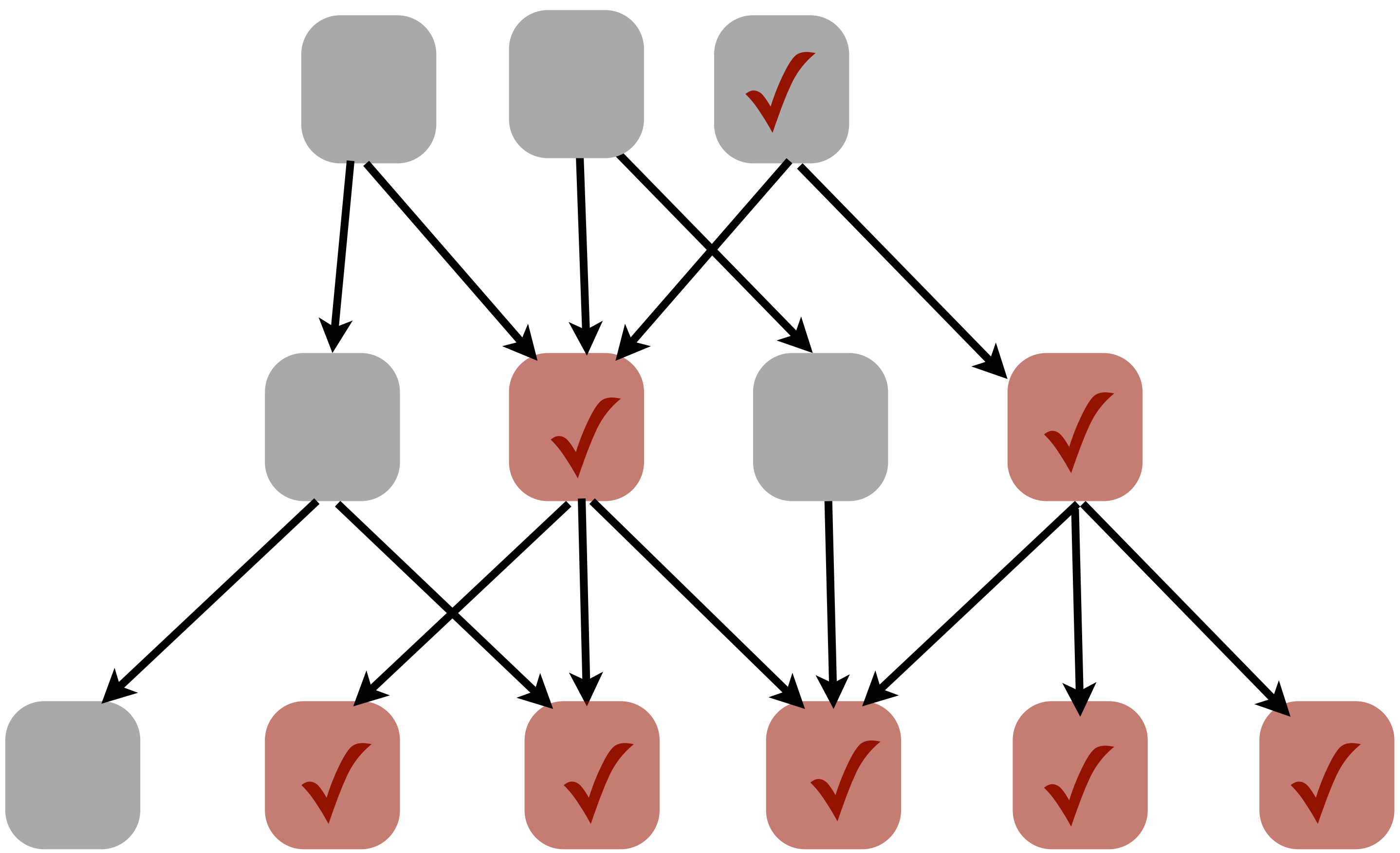
Dependency Graph



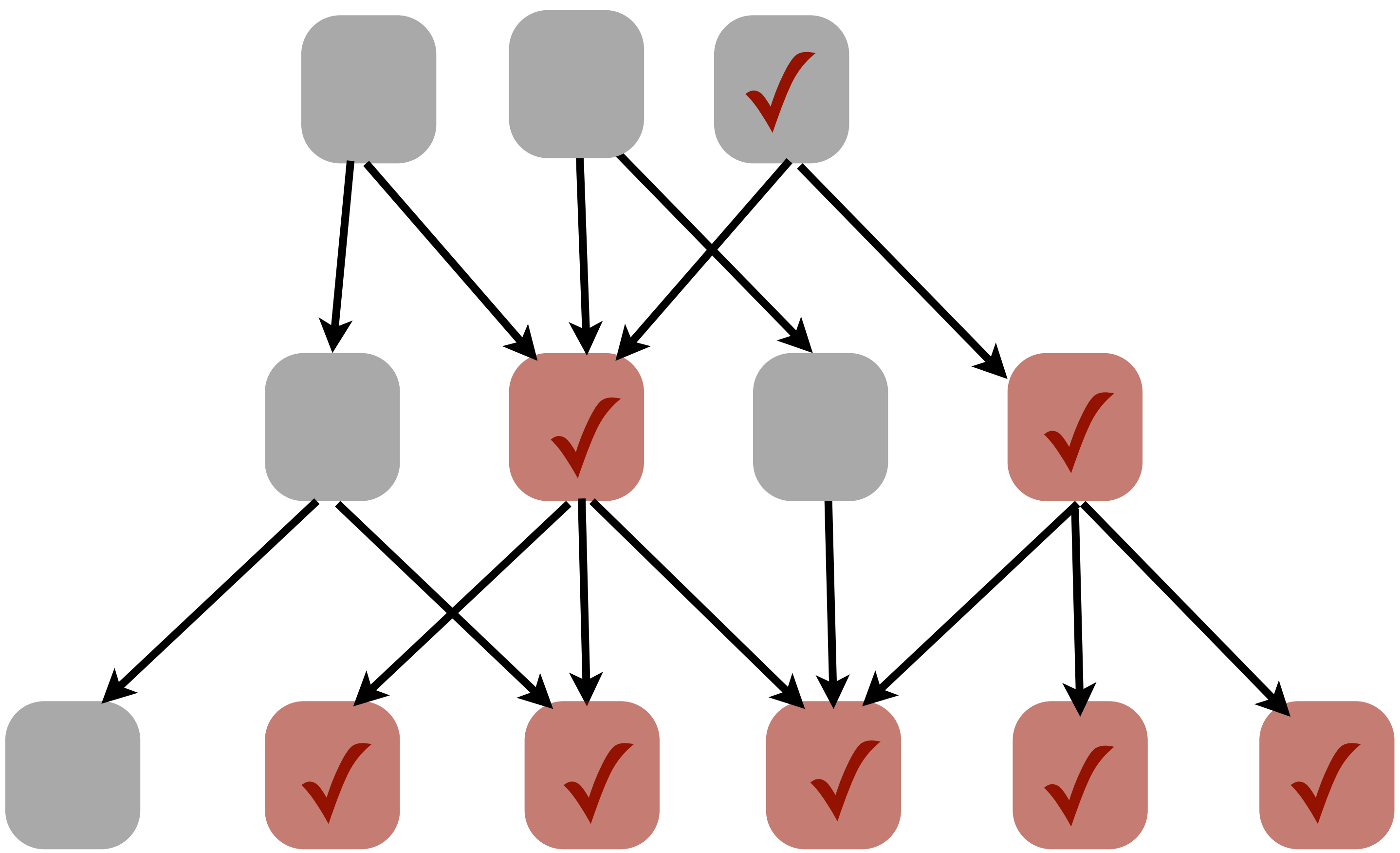
Error Propagation



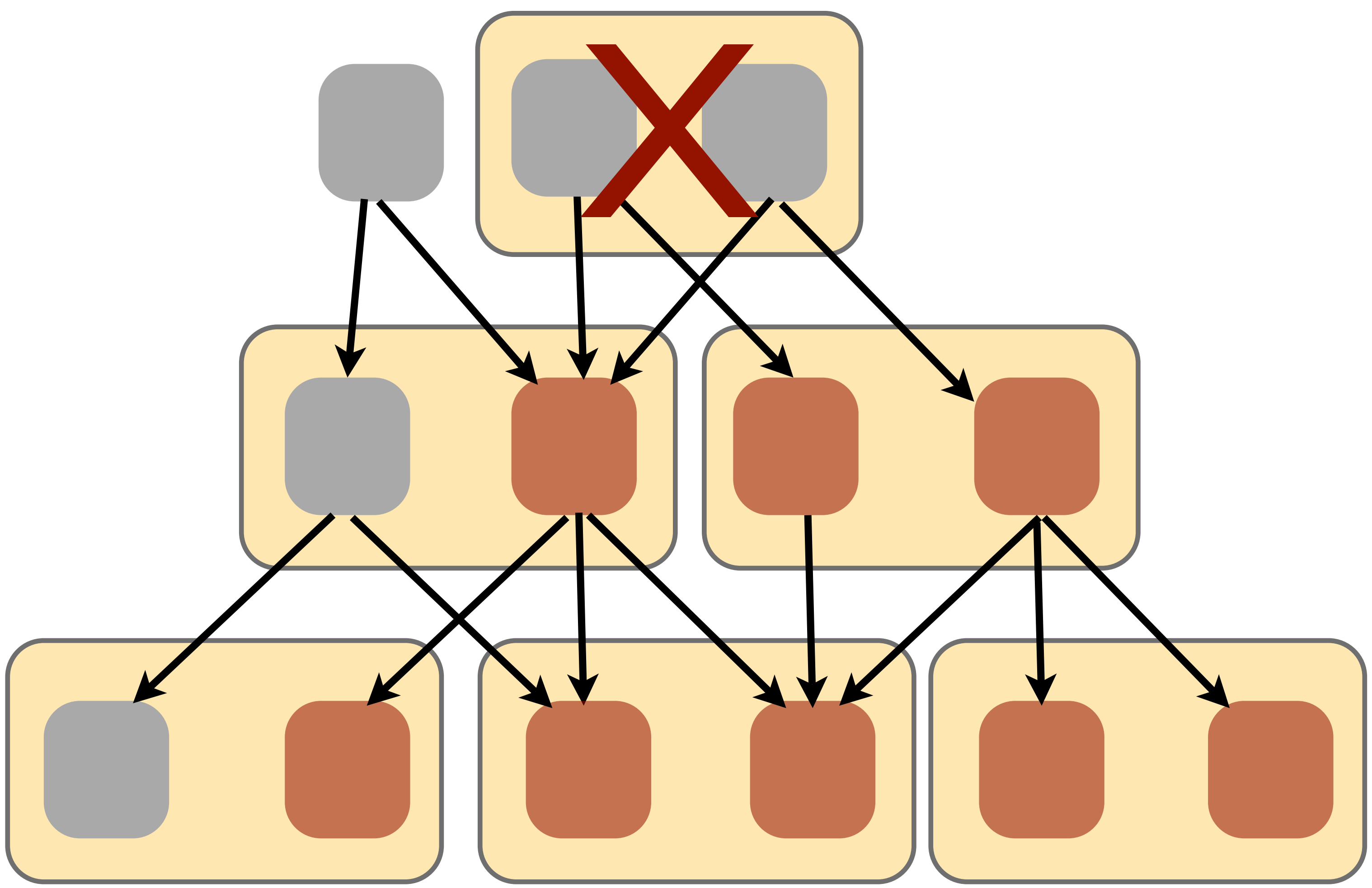
Retransmission upon detecting loss



Retransmission takes precedence over new vertex splits



Normally send multiple vertex splits per packet

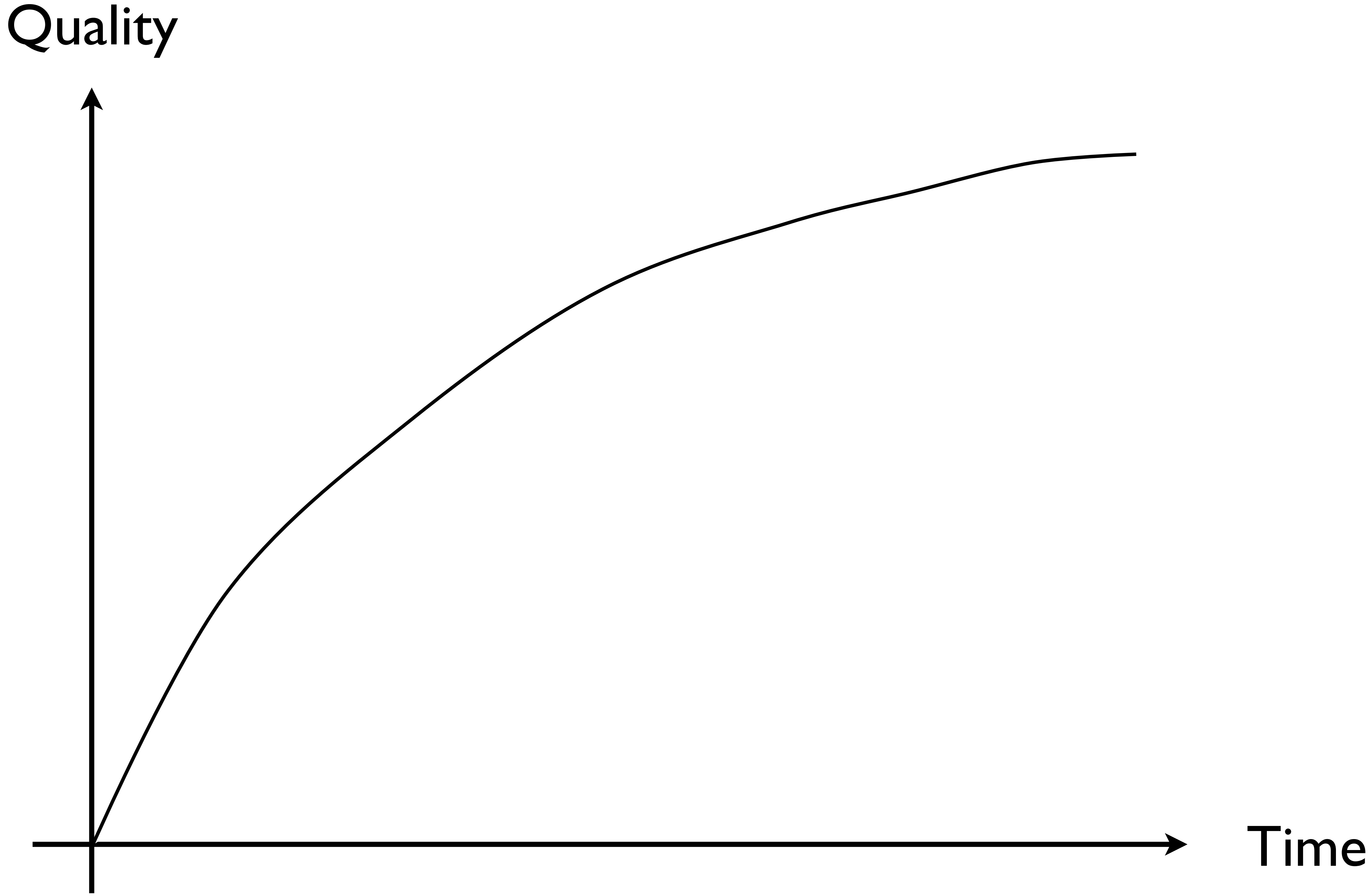


How serious is error propagation?

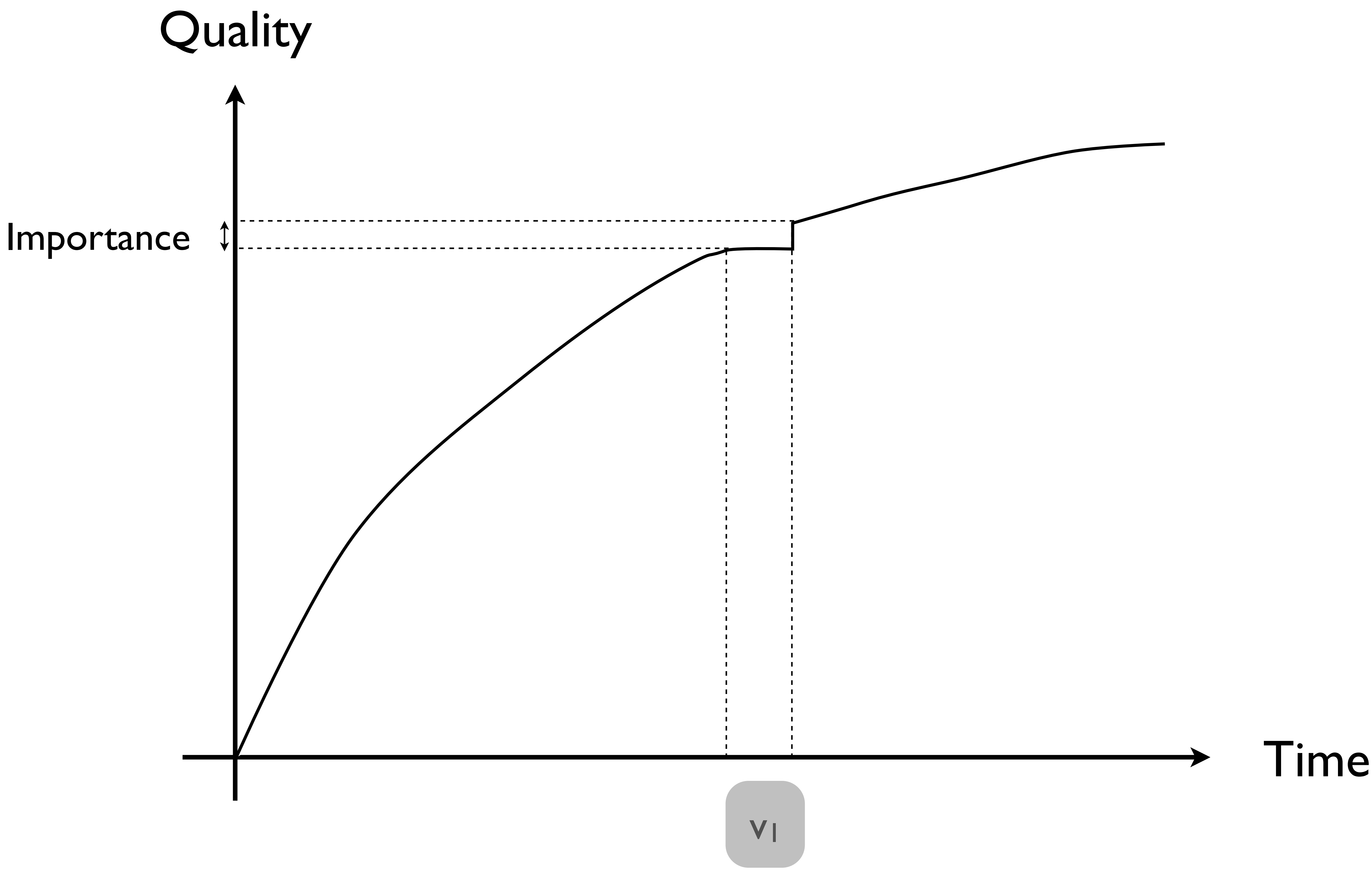
What is the effect of dependencies?

Decoded Mesh Quality

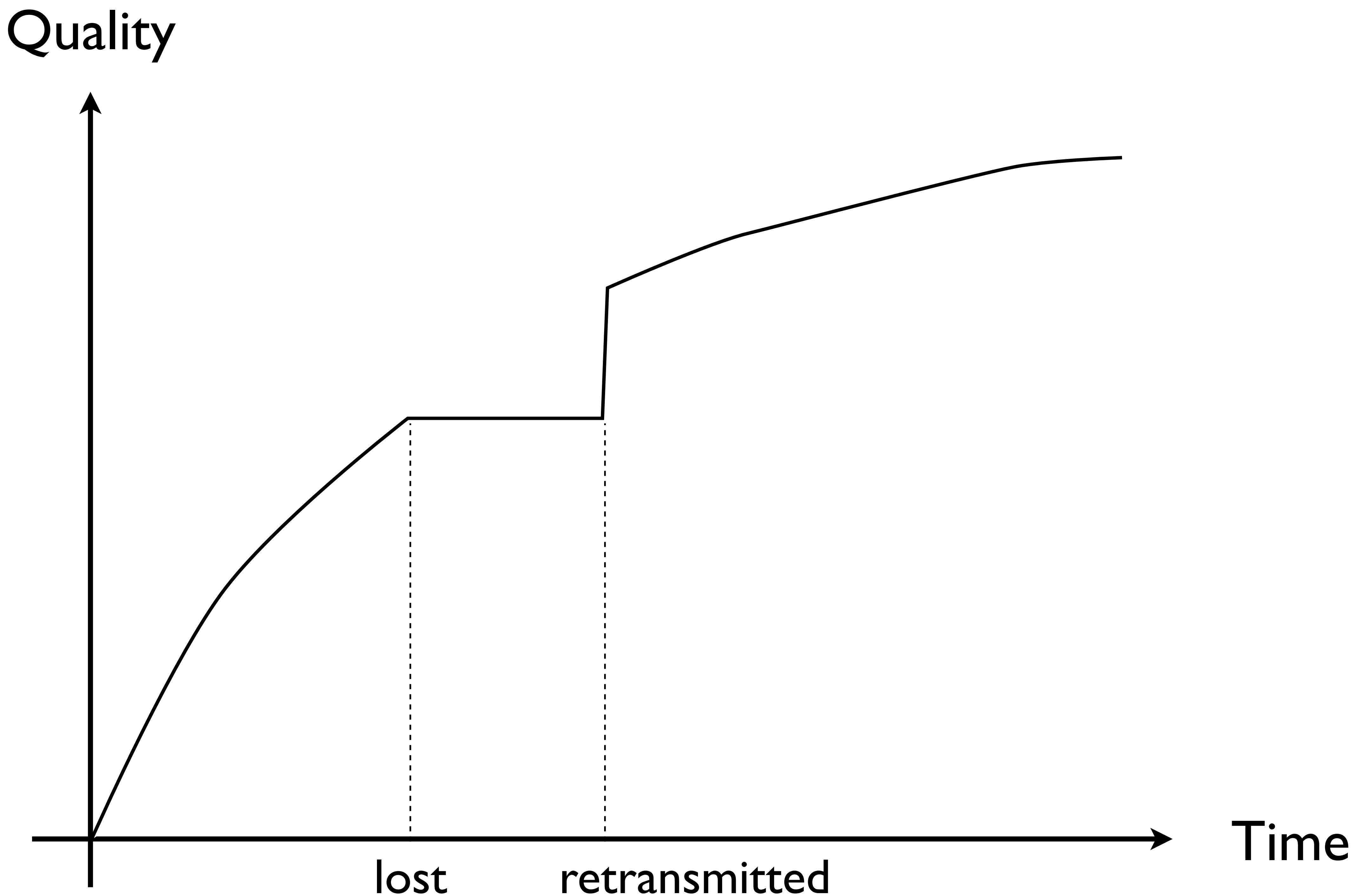
Quality versus Time



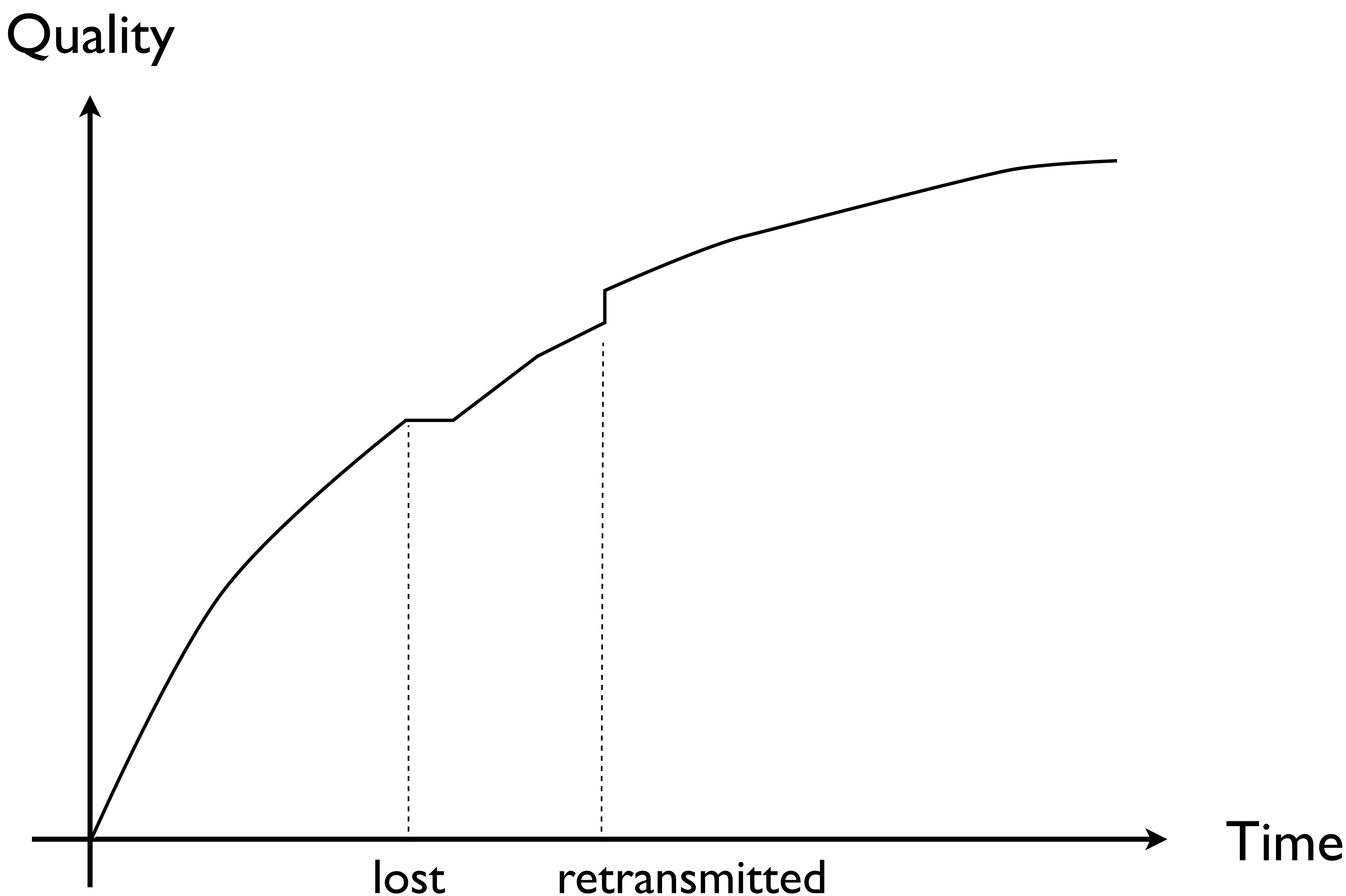
Importance of a vertex



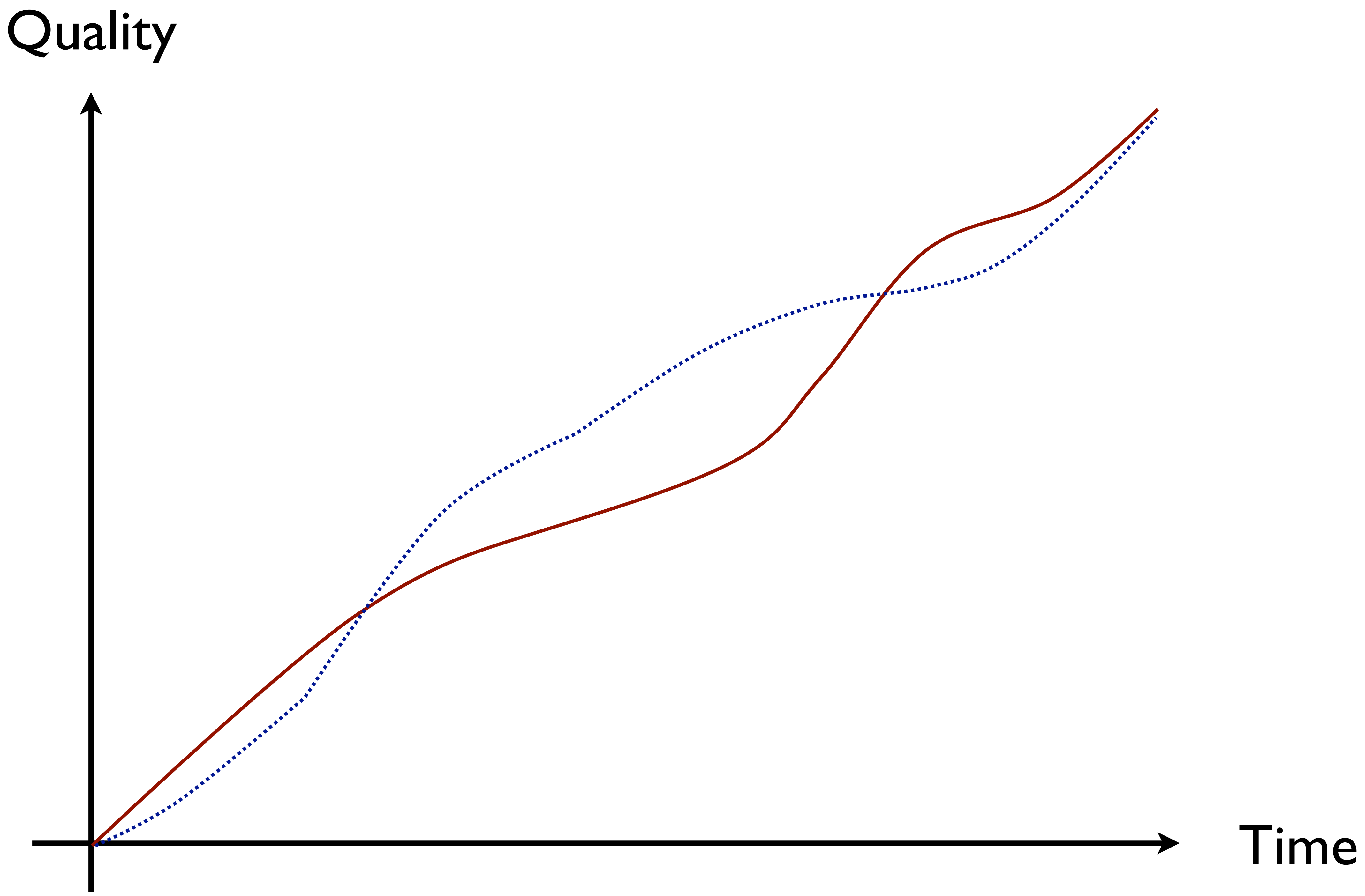
Case 1: complete dependency



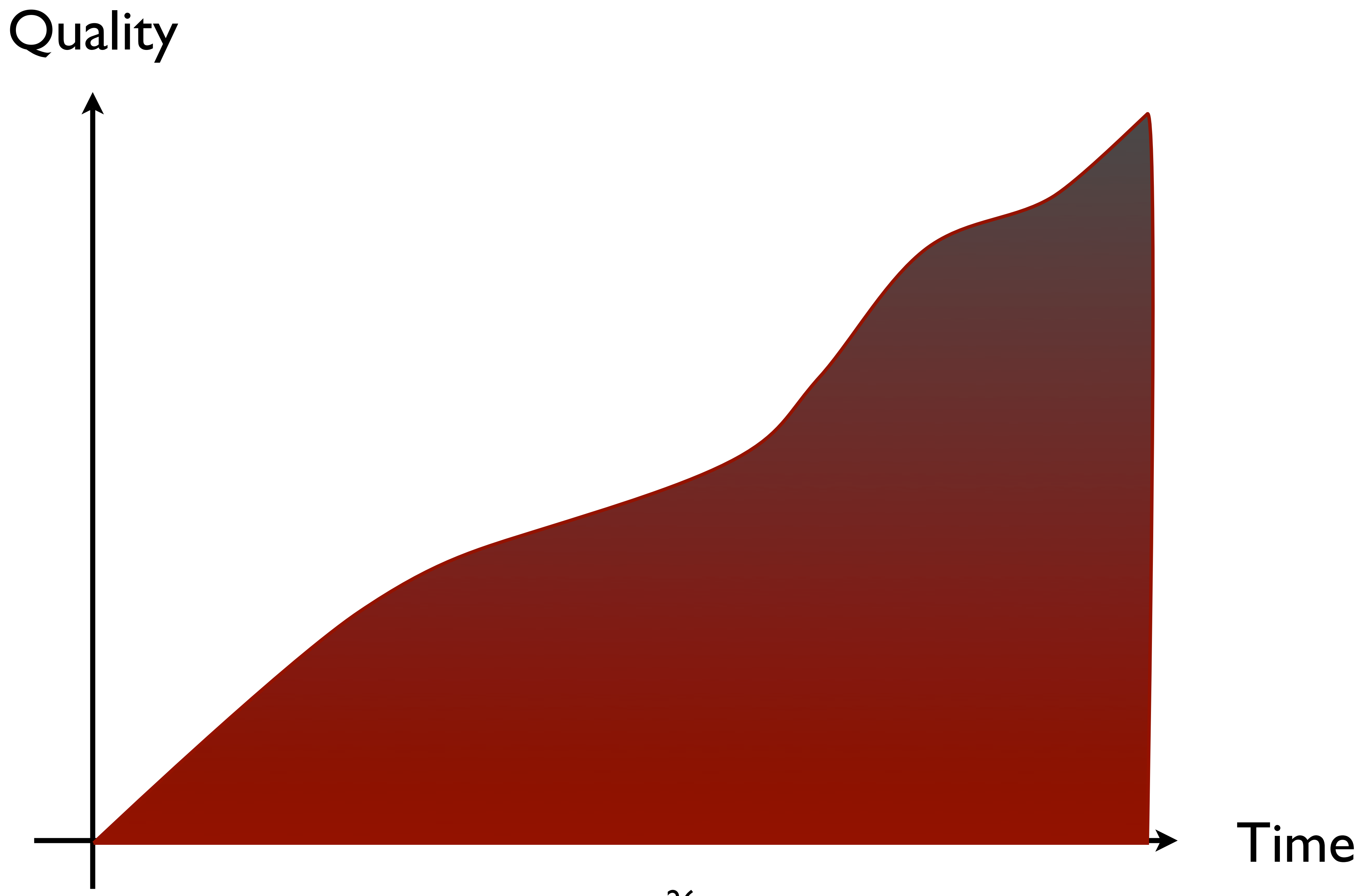
Case 2: no dependency



higher quality **earlier** is better



Evaluation metric: area under the graph
Larger area = better



Given a progressive mesh,
what affects the area?

Dependency Pattern

Given a progressive mesh, the dependencies among the vertex splits are fixed, but packetization can affect dependencies among the data packets.

Sending Order

Given a set of packets, which one to send first?

Loss Rate

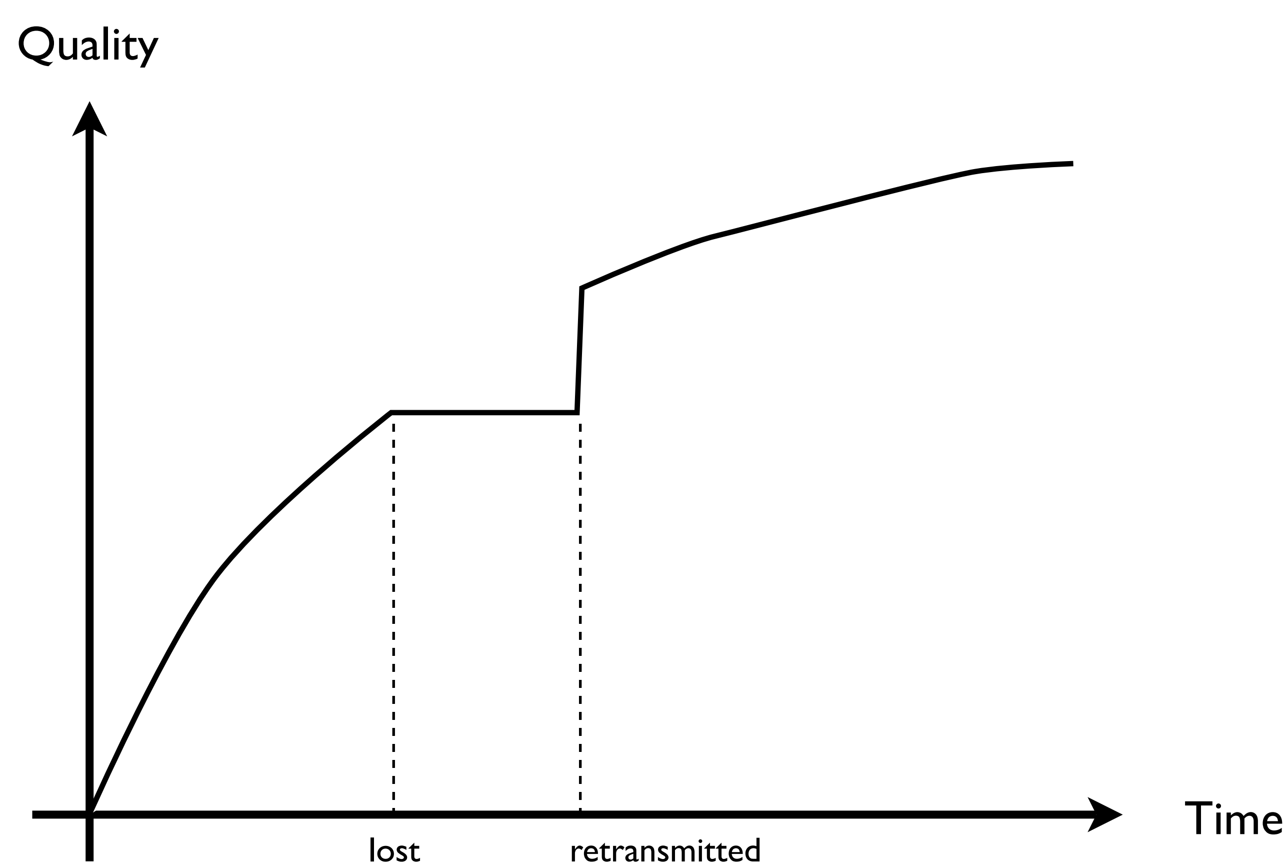
Different loss pattern gives different area. We are interested in the expected area given a loss rate.

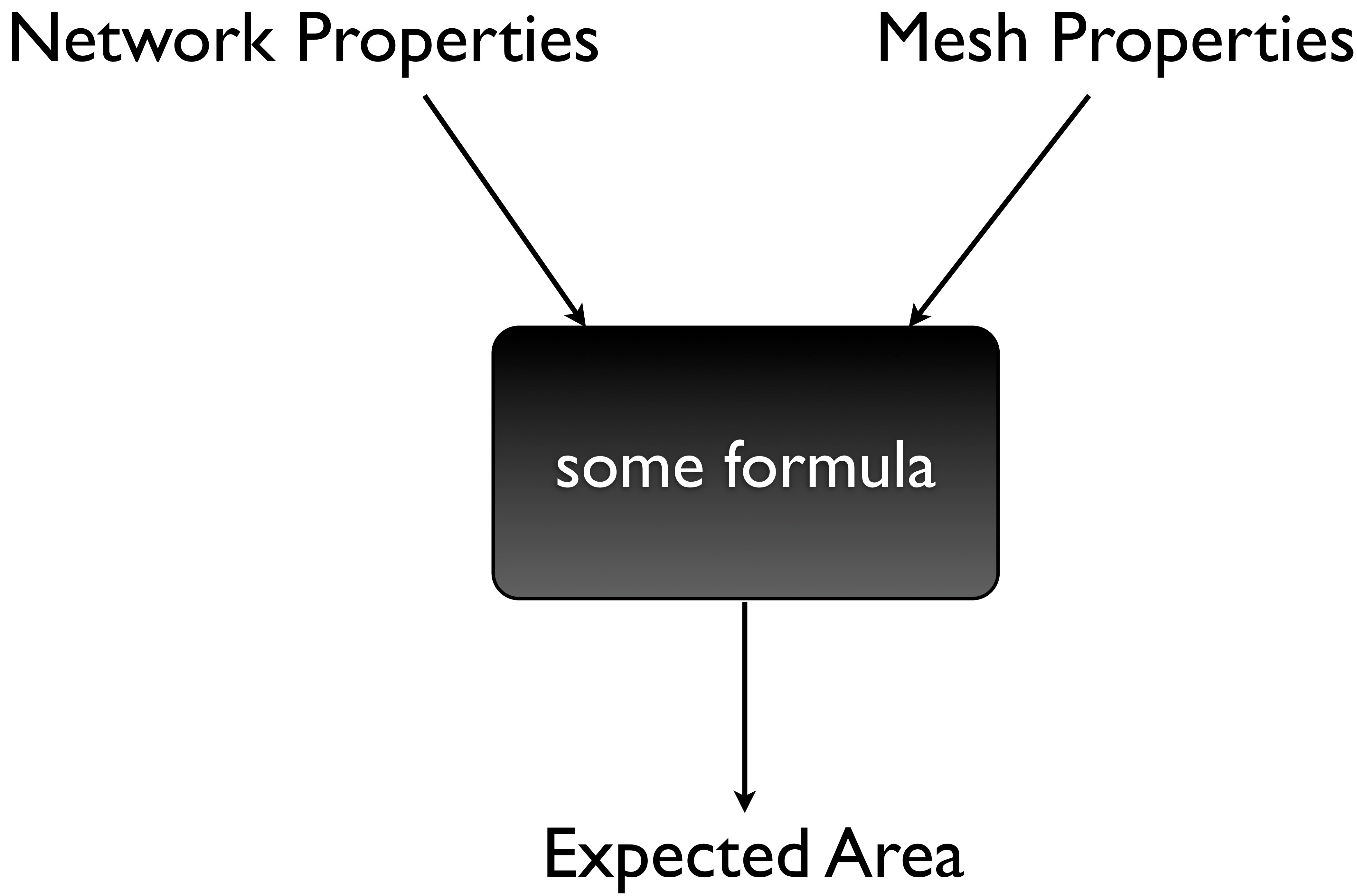
Available Bandwidth

Faster sending rate means the quality increases quickly.

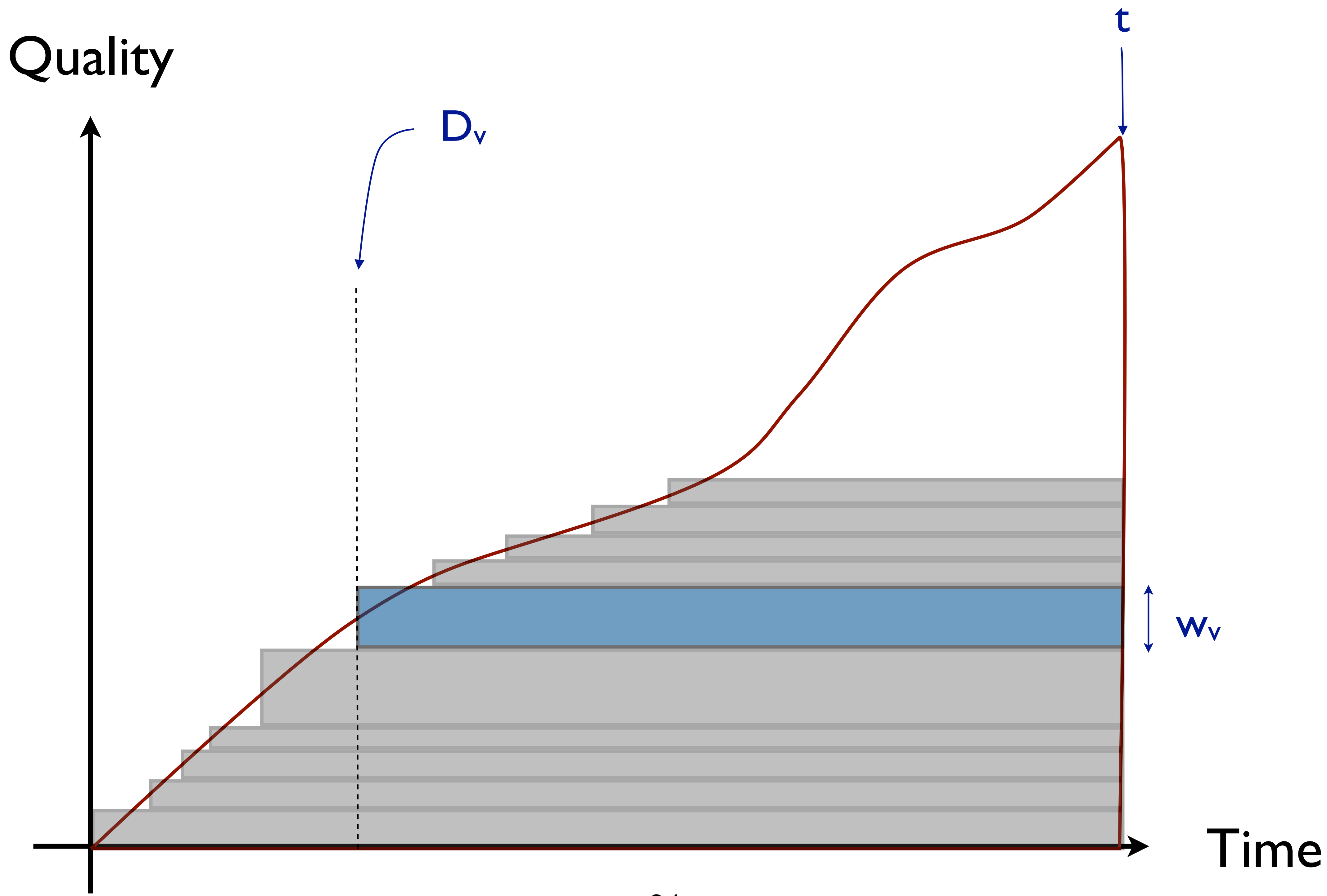
Round Trip Time

Larger round trip time means longer time till realizing that a packet is lost and retransmit.

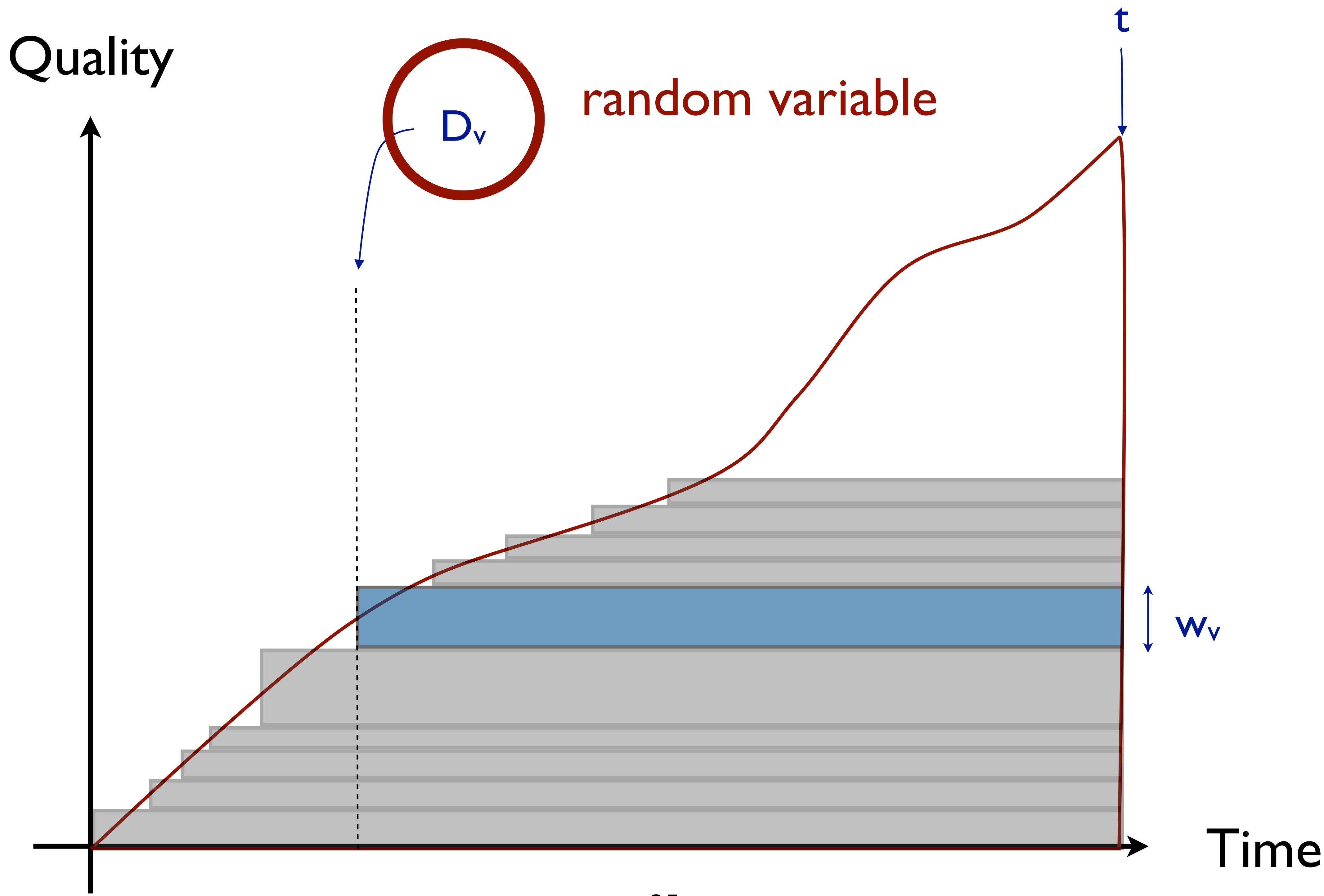



Network Properties
Mesh Properties
some formula
Expected Area

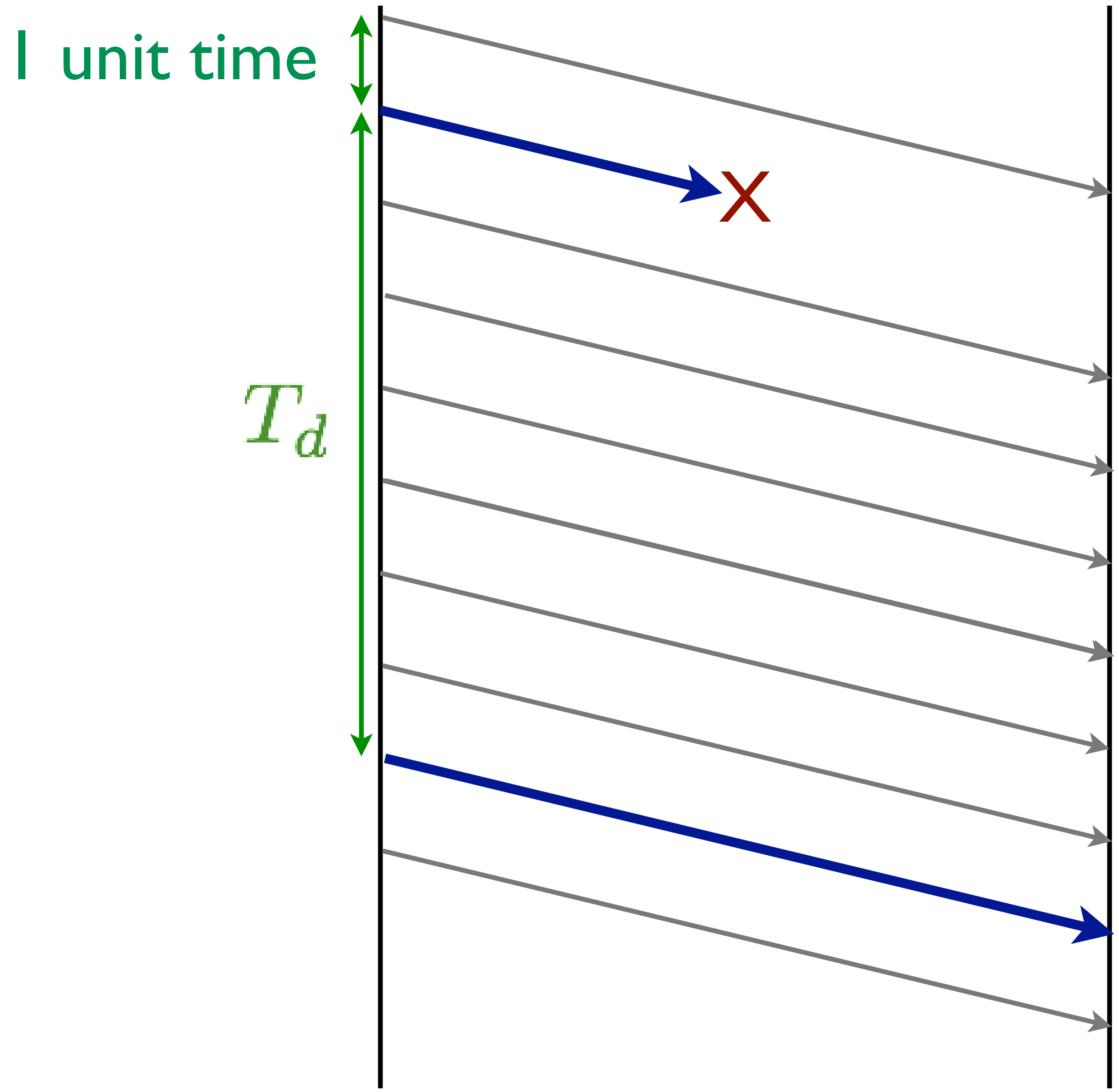
$$\sum_v w_v(t - D_v)$$



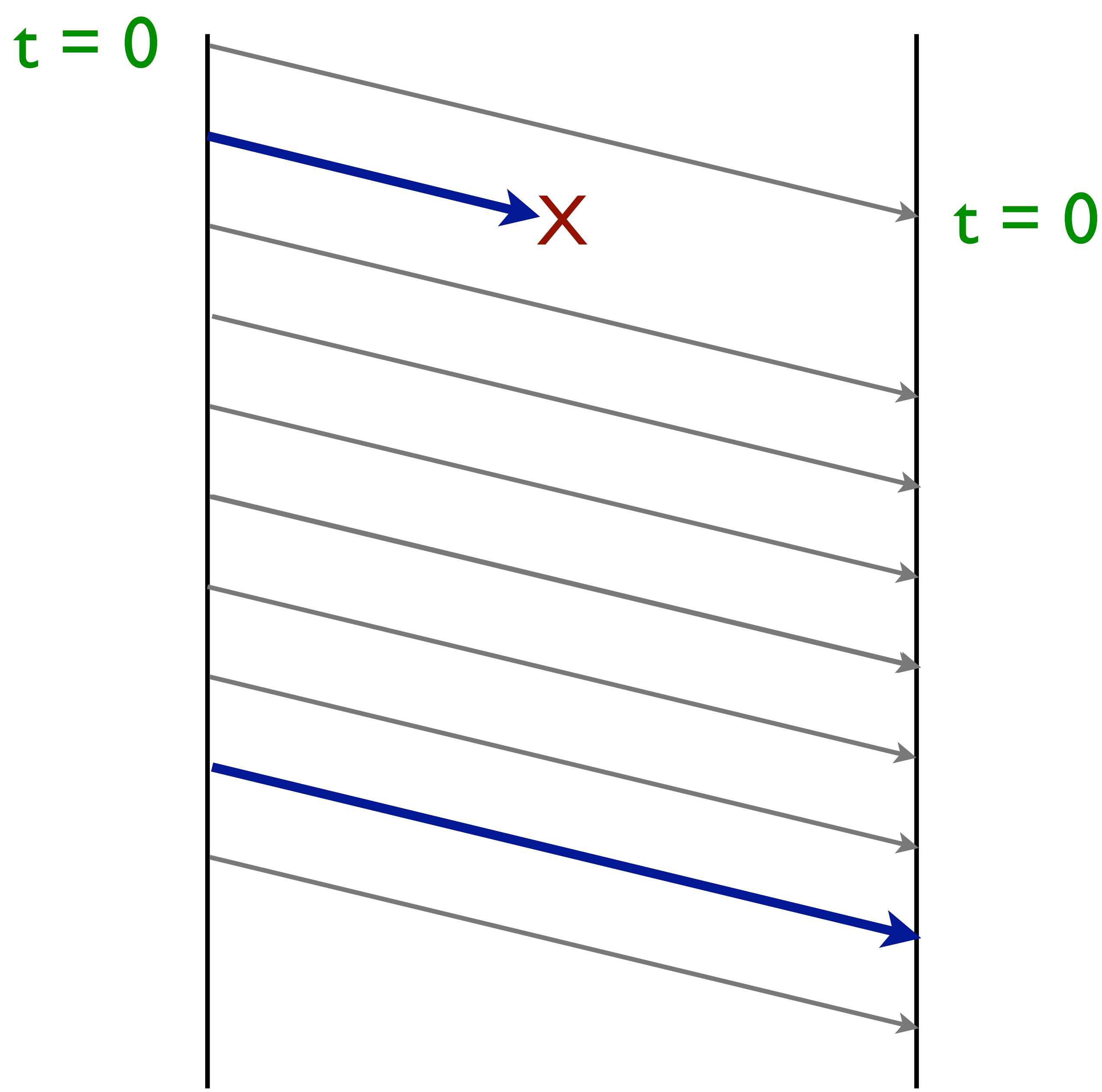
$$\sum_v w_v(t - D_v)$$



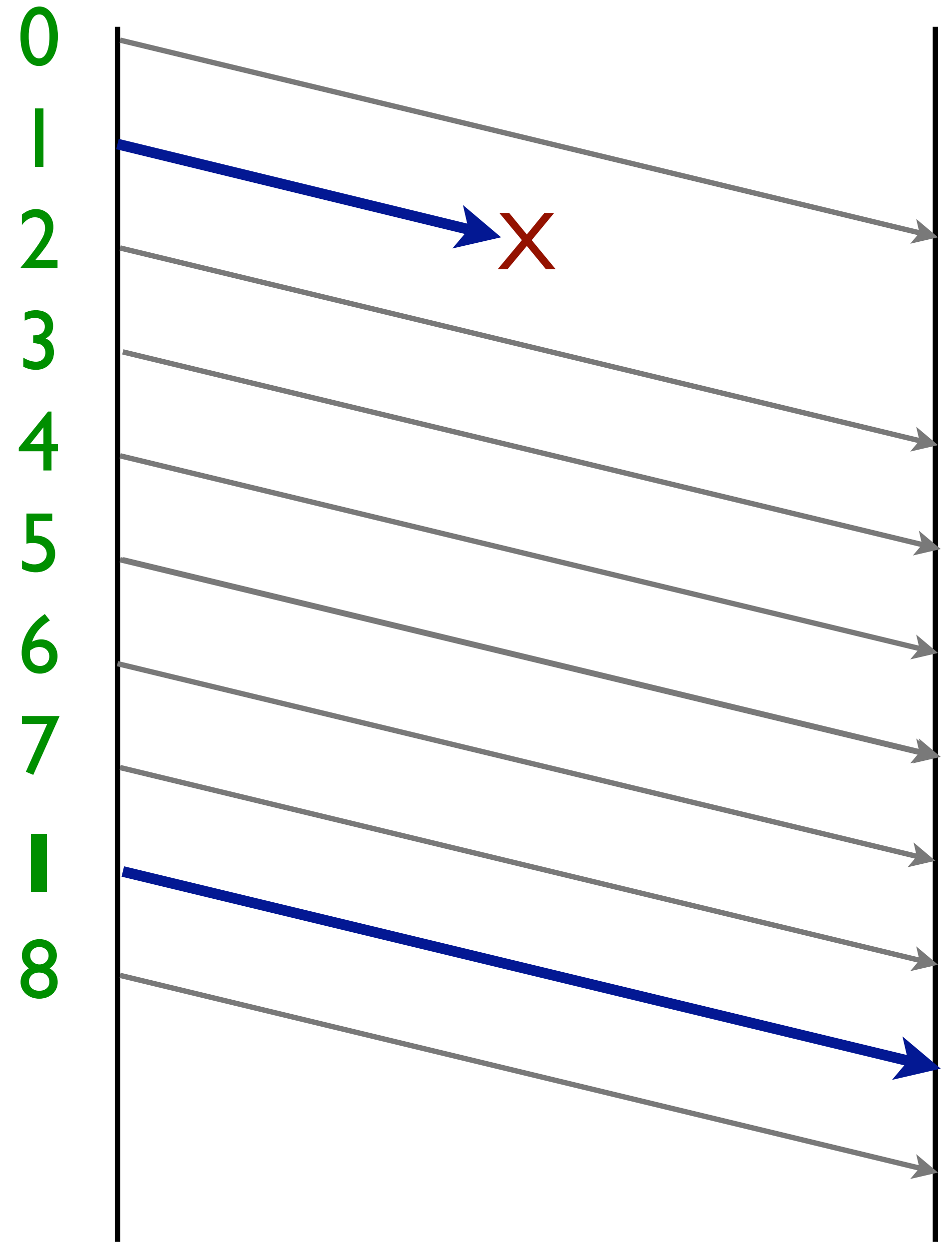
The Analytical Model


1 unit time
T_d
X

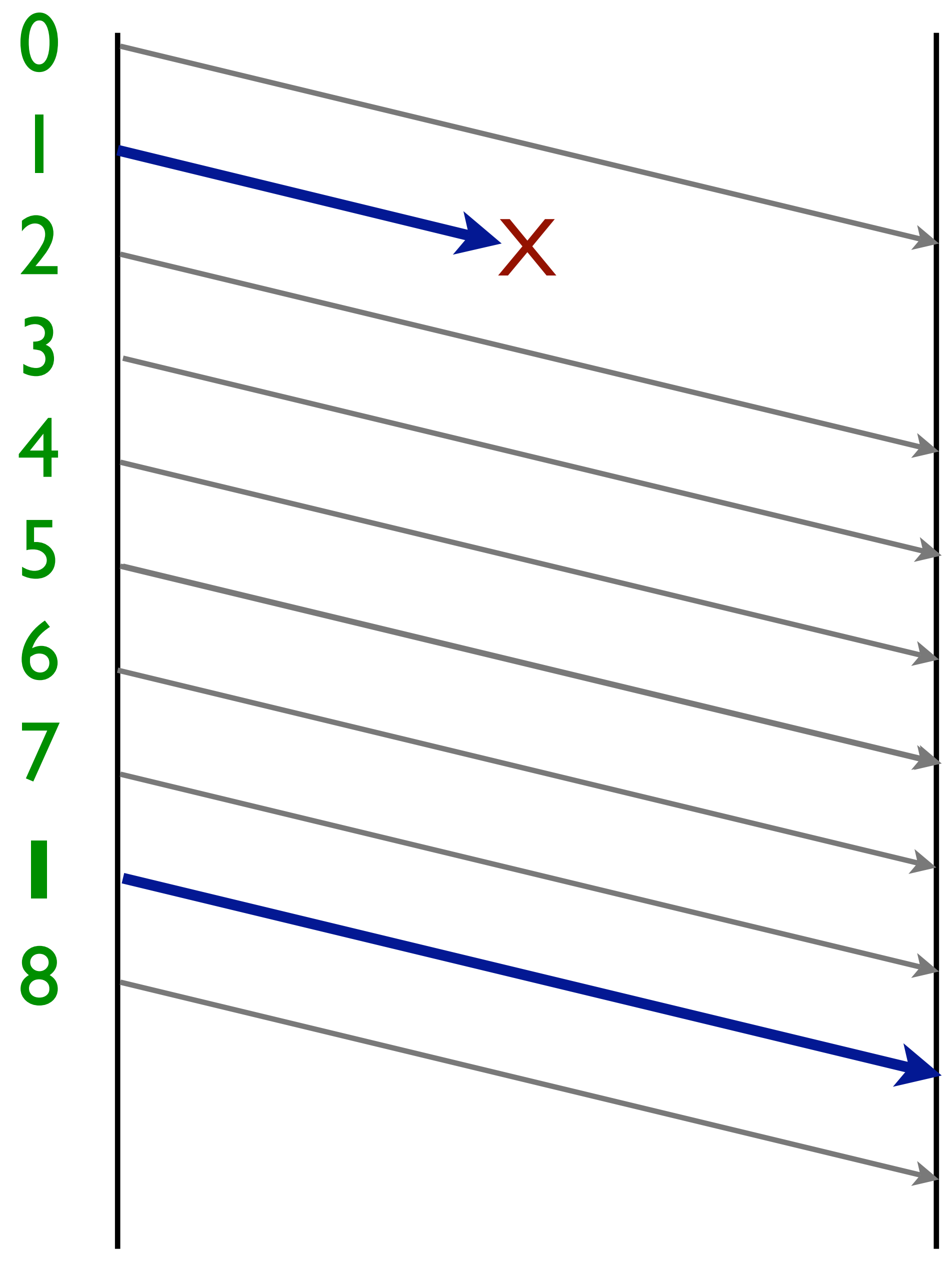
Clock at sender starts when sending first packet
Clock at receiver starts RTT/2 later.

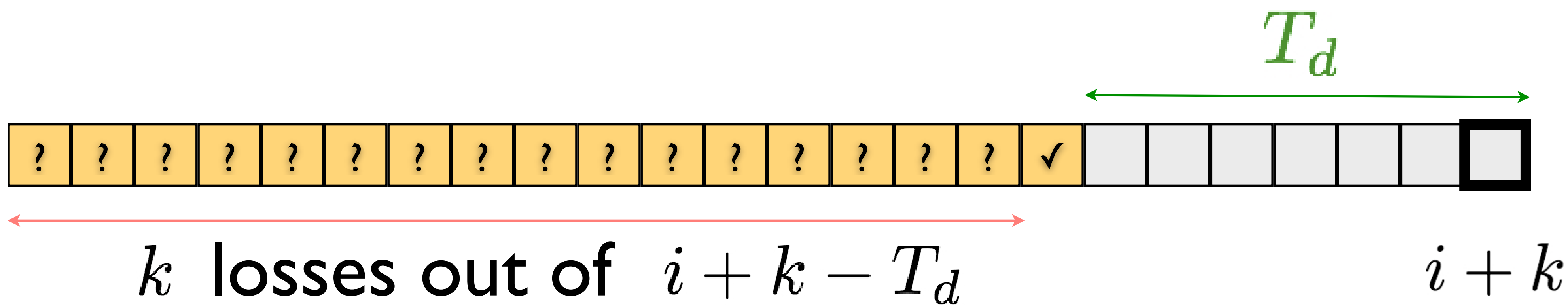


Packet *i* is sent at time *i* if there is no retransmission



Packet *i* is sent at time *i*+*k* if there are *k* retransmissions before *i*





$$P(S_i = i + k) = \binom{i - T_d + k}{k} p^k (1 - p)^{i - T_d + 1}$$

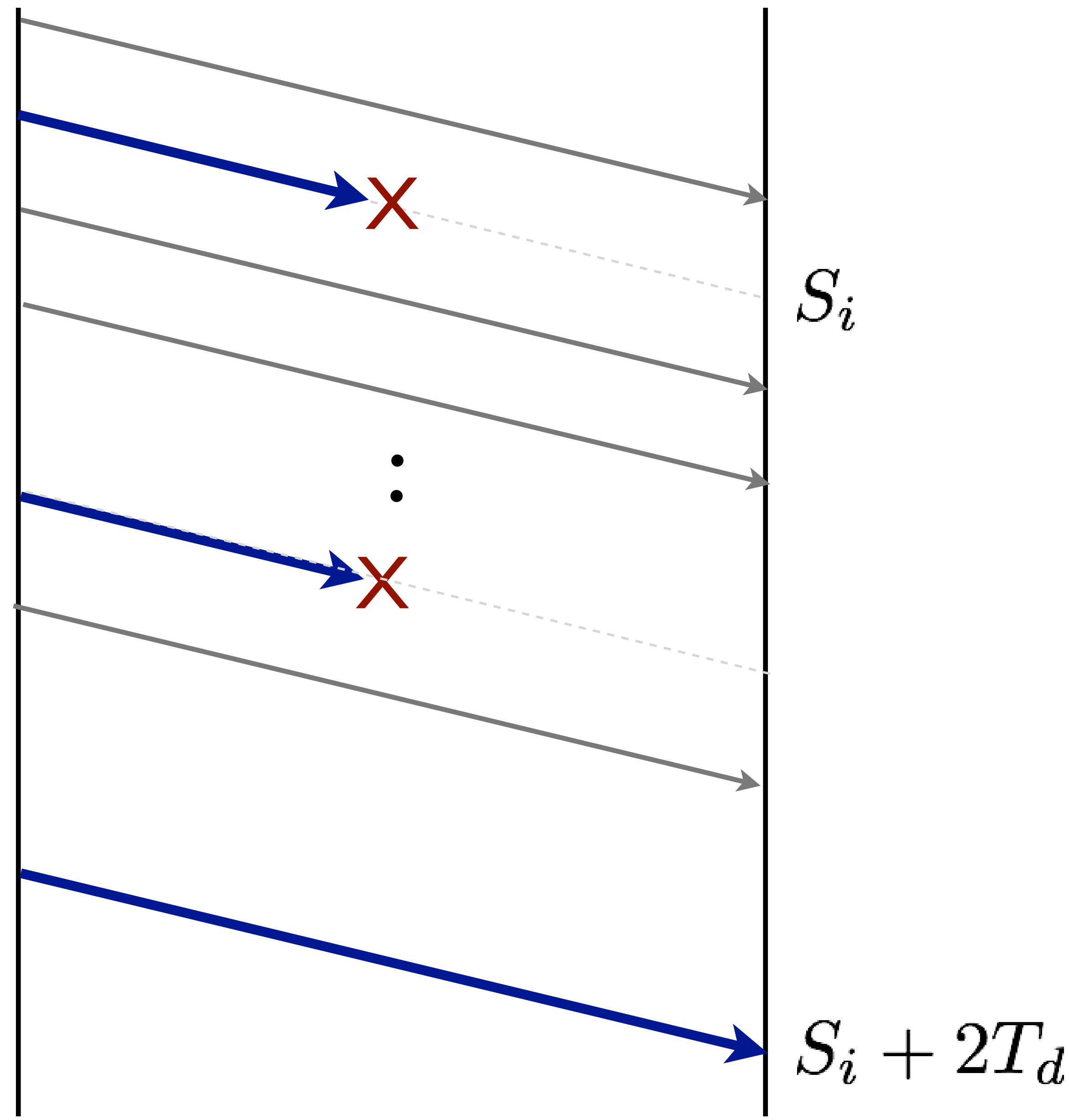
$$E[S_i] = \frac{i - T_d + 1}{1 - p} + T_d - 1$$

time slot when packet *i* is sent

loss probability

Packet *i* is received at time

$$R_i = S_i + nT_d$$



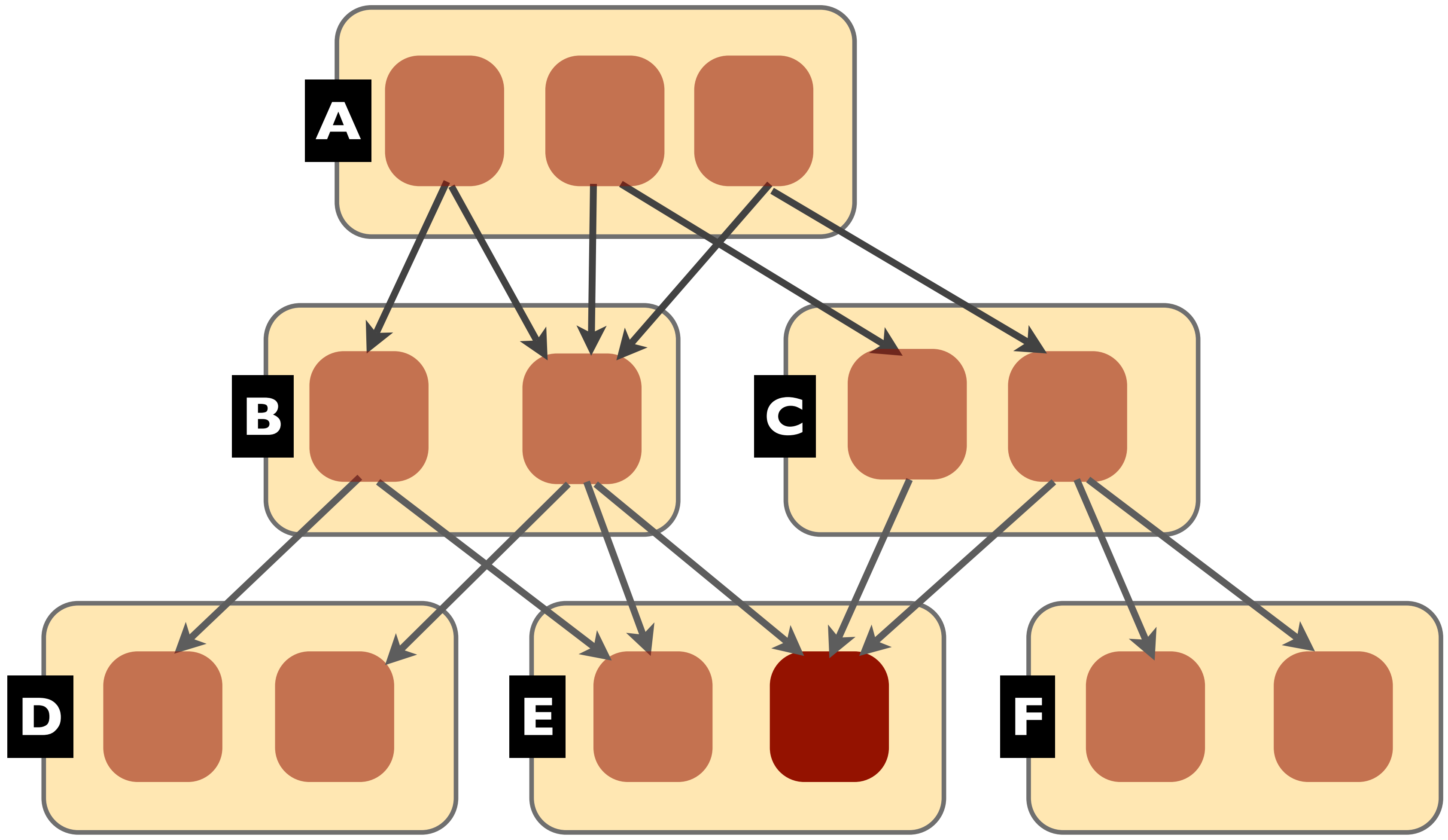
approximate using E[S_i]

$$Pr(R_i = t) = \begin{cases} (1-p)p^{n_{i,t}} & \text{if } (t - S_i) \mod T_d = 0 \\ 0 & \text{otherwise} \end{cases}$$

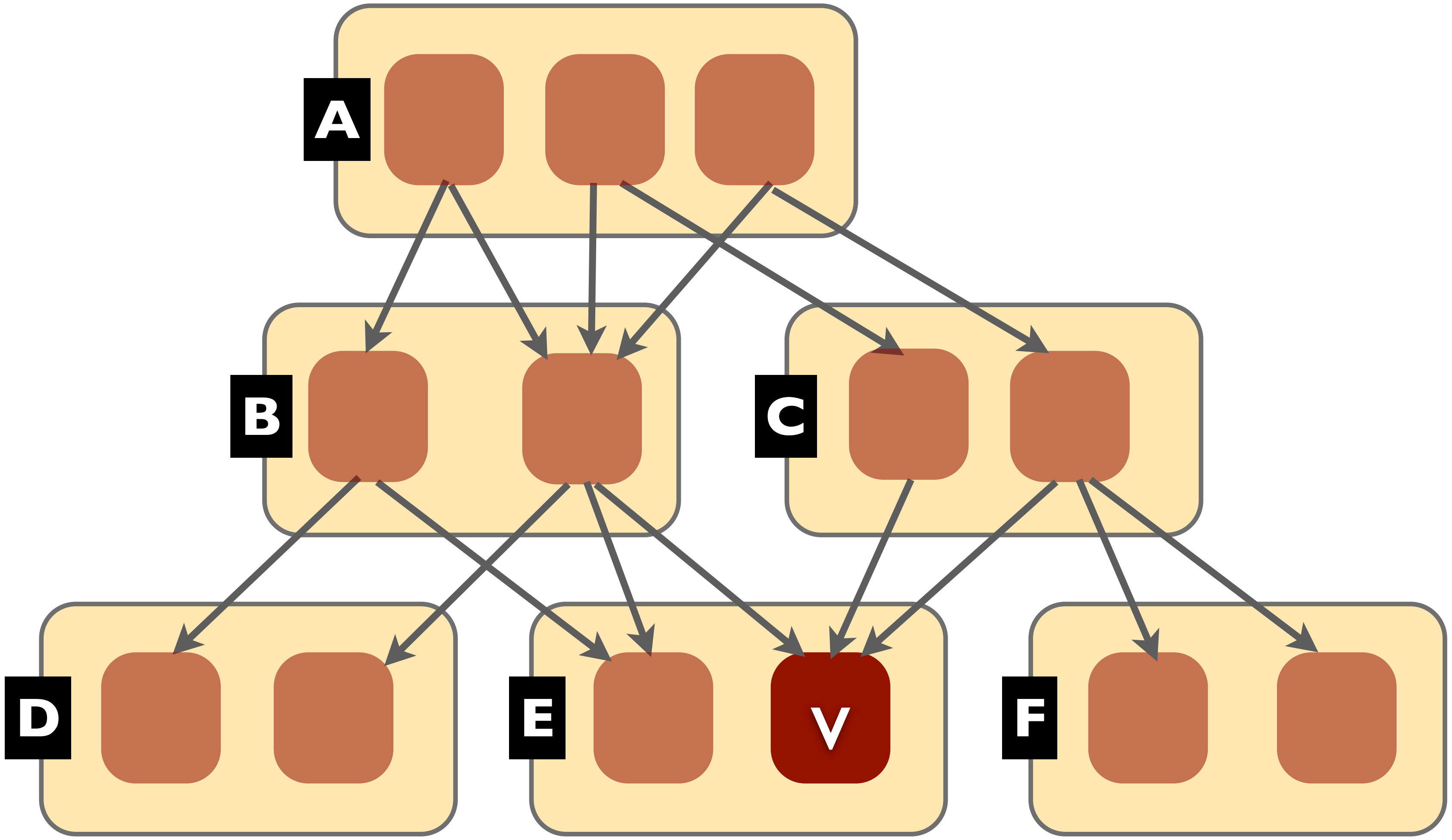
$$n_{i,t} = \lfloor (t - S_i)/T_d \rfloor$$

$$Pr(R_i \leq t) = 1 - p^{n_{i,t}+1}$$

A packet *p* is a parent packet of a vertex *v* if a vertex that *v* depends on belongs to *p*



$$P(v) = \{A, B, C, E\}$$

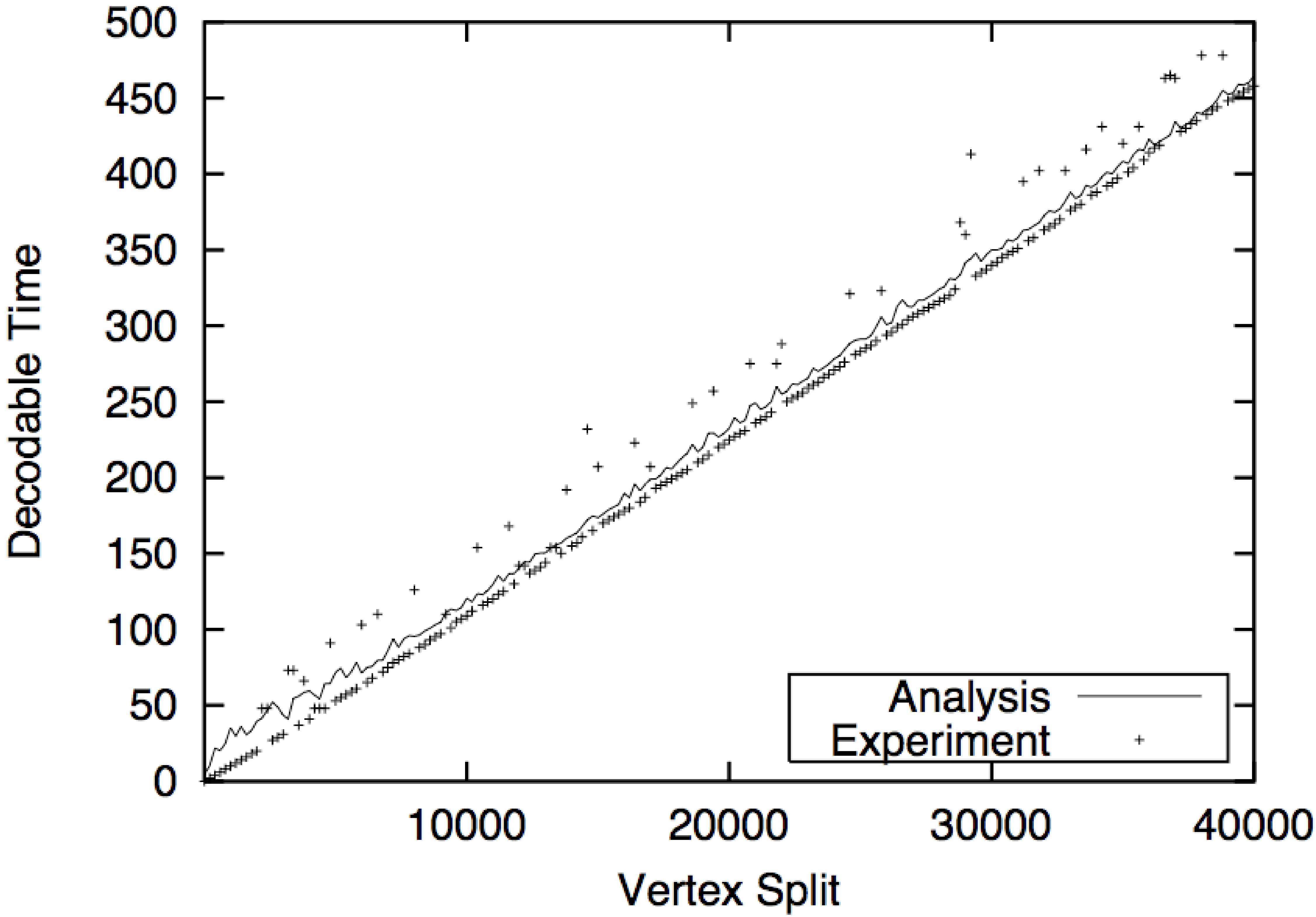


A vertex *v* is decoded at time *t*, if

1. a parent packet of *v* is received at time *t*, and
2. all other parent packets are received before *t*.

$$Pr(D_v = t) = \sum_{j \in \mathcal{P}(i)} \frac{Pr(R_j = t)}{Pr(R_j < t)} \prod_{k \in \mathcal{P}(i)} Pr(R_k < t)$$

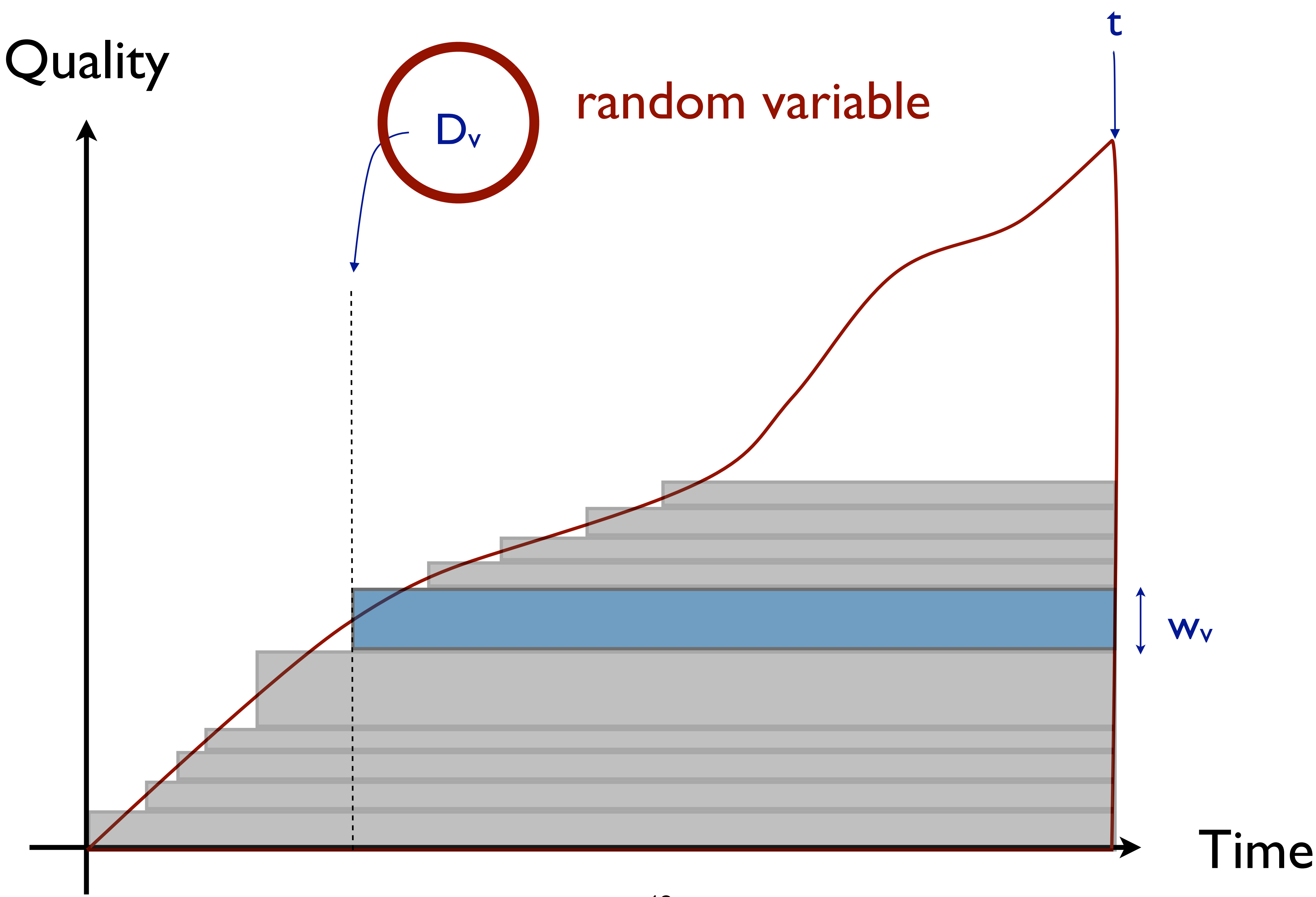
$$E[D_v] = \sum_{j=S_v}^{\infty} j Pr(D_v = j)$$


Decodable Time
500
450
400
350
300
250
200
150
100
50
0
10000
20000
30000
40000
Vertex Split
Analysis
Experiment

Simulation with HORSE model with 10% Losses

Number of runs	Average difference	Maximum difference
1000	0.474	3.192
10000	0.161	1.567
100000	0.122	1.308
trace	0.177	2.184

$$\sum_v w_v(t - D_v)$$



$$x_{i,t} = \begin{cases} 1 & \text{if } D_i \leq t \\ 0 & \text{otherwise} \end{cases} .$$

$$x_{i,t} = \begin{cases} 1 & \text{if } D_i \leq t \\ 0 & \text{otherwise} \end{cases} .$$

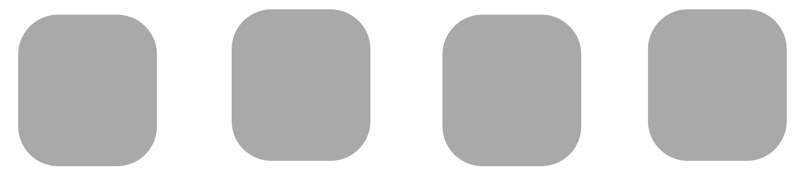
$$a_t = \sum_{i=0}^{t} x_{i,t} w_i (t - D_i)$$

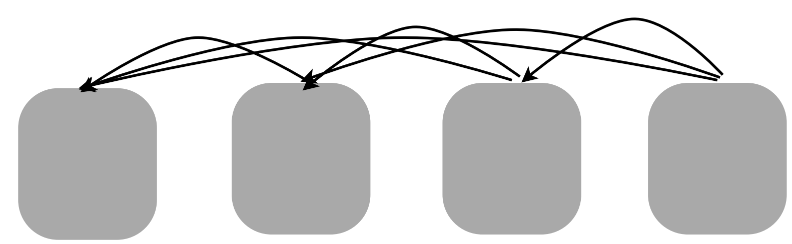
$$x_{i,t} = \begin{cases} 1 & \text{if } D_i \leq t \\ 0 & \text{otherwise} \end{cases} .$$

$$a_t = \sum_{i=0}^{t} x_{i,t} w_i (t - D_i)$$

$$\begin{aligned} E[a_t] &= \sum_{i=0}^{t} w_i (tE[x_{i,t}] - E[x_{i,t} D_i]) \\ &= \sum_{i=0}^{t} w_i (tP(D_i \leq t) - \sum_{k=0}^{t} kP(D_i = k)) \end{aligned}$$

Do dependencies matter?

best case 

worst case 

best case

$$\Delta_t = (1 - p)$$

worst case

$$\Delta_t = \begin{cases} (1-p)^{t+1} & \text{if } t < T_d \\ 1 - p & \text{if } t \geq T_d \text{ and } t = nT_d \\ (1-p^{n+1})\Delta_{t-1} & \text{if } t \geq T_d \text{ and } t = nT_d + b \end{cases}$$

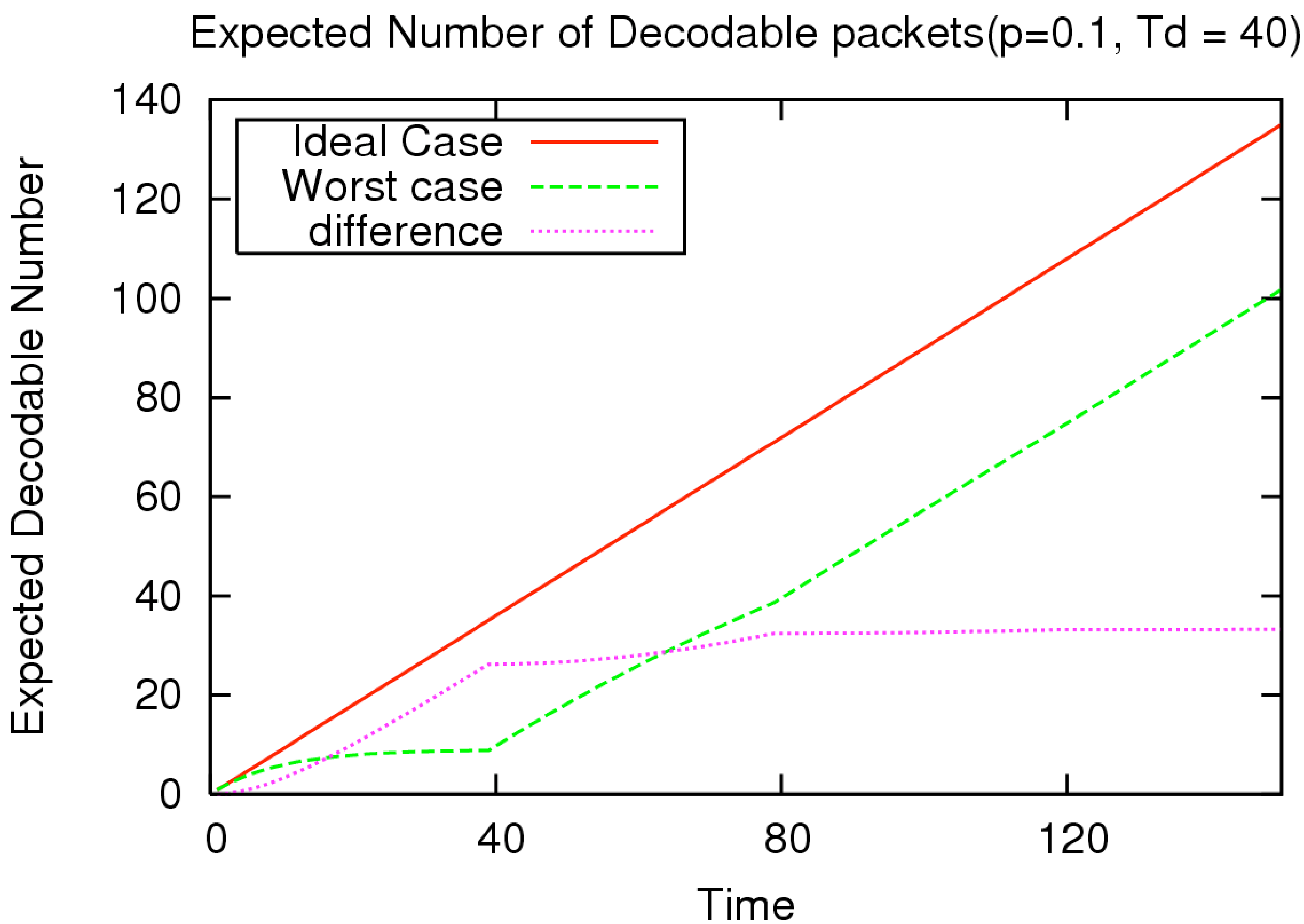
best case

$$\Delta_t = (1 - p)$$

worst case

$$\Delta_t = \begin{cases} (1-p)^{t+1} & \text{if } t < T_d \\ 1 - p & \text{if } t \geq T_d \text{ and } t = nT_d \\ (1-p^{n+1})\Delta_{t-1} & \text{if } t \geq T_d \text{ and } t = nT_d + b \end{cases}$$

$$\Delta_t \approx (1 - p) \qquad \text{for large } t$$


Expected Number of Decodable packets(p=0.1, Td = 40)
Expected Decodable Number
Time
Ideal Case
Worst case
difference
0
20
40
60
80
100
120
140
0
40
80
120

Gap between the two extreme cases at $t = T_d$ - 1

$$\left(T_d - \frac{1-(1-p)^{T_d}}{p}\right)(1-p)$$

RTT = 250 ms , Packet Size = 1500 bytes , Sending rate = 1.5 Mbps

T_d = **30**, p = **5**%

100 vertex splits per packet

Gap = **1500** vertex splits

A Better Packetization Algorithm

FIFO strategy:

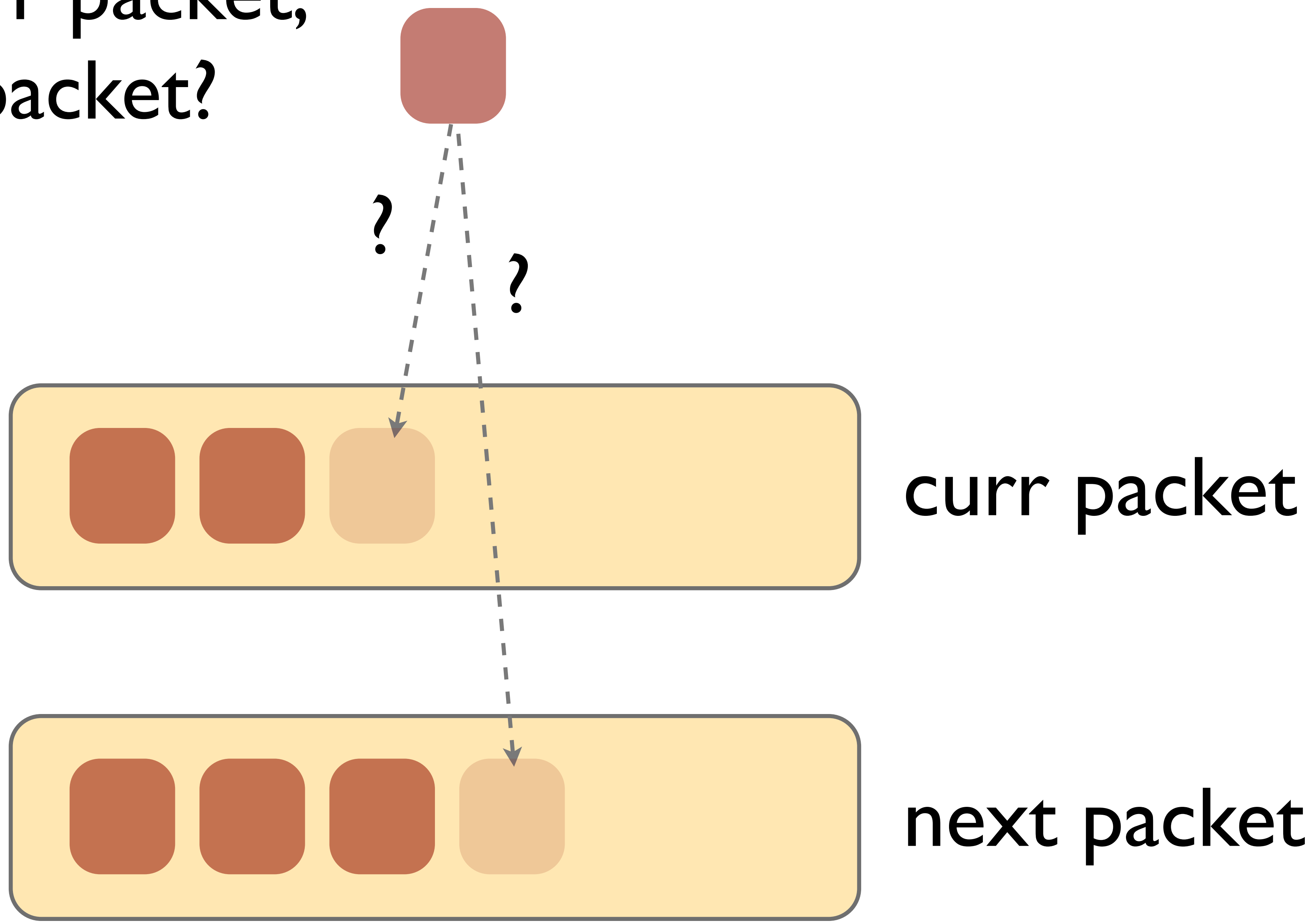
send the most important vertex split first

[Gu05]'s strategy:

minimize the dependencies among the vertex splits

Need to consider **both** importance and dependencies

put into curr packet,
or next packet?

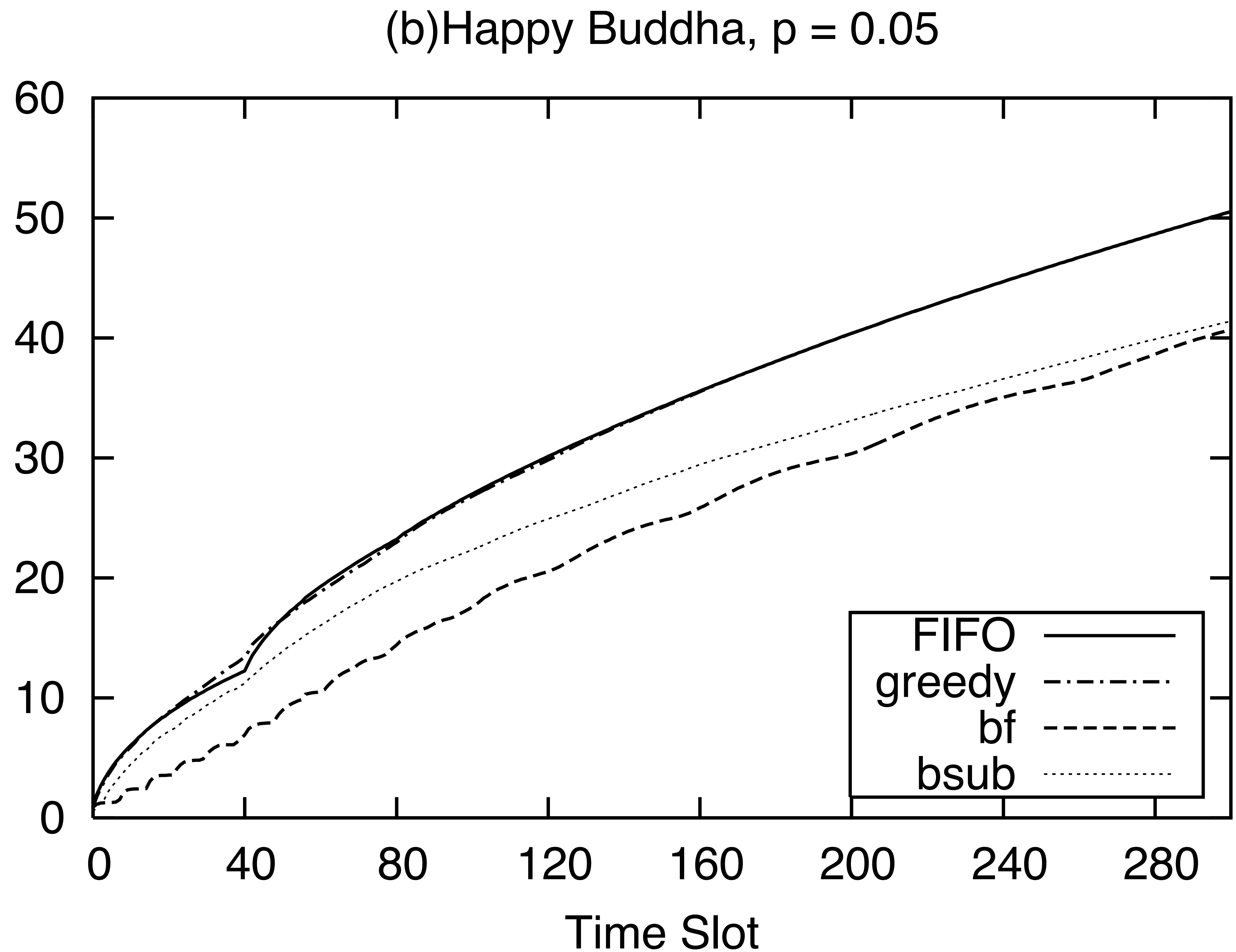


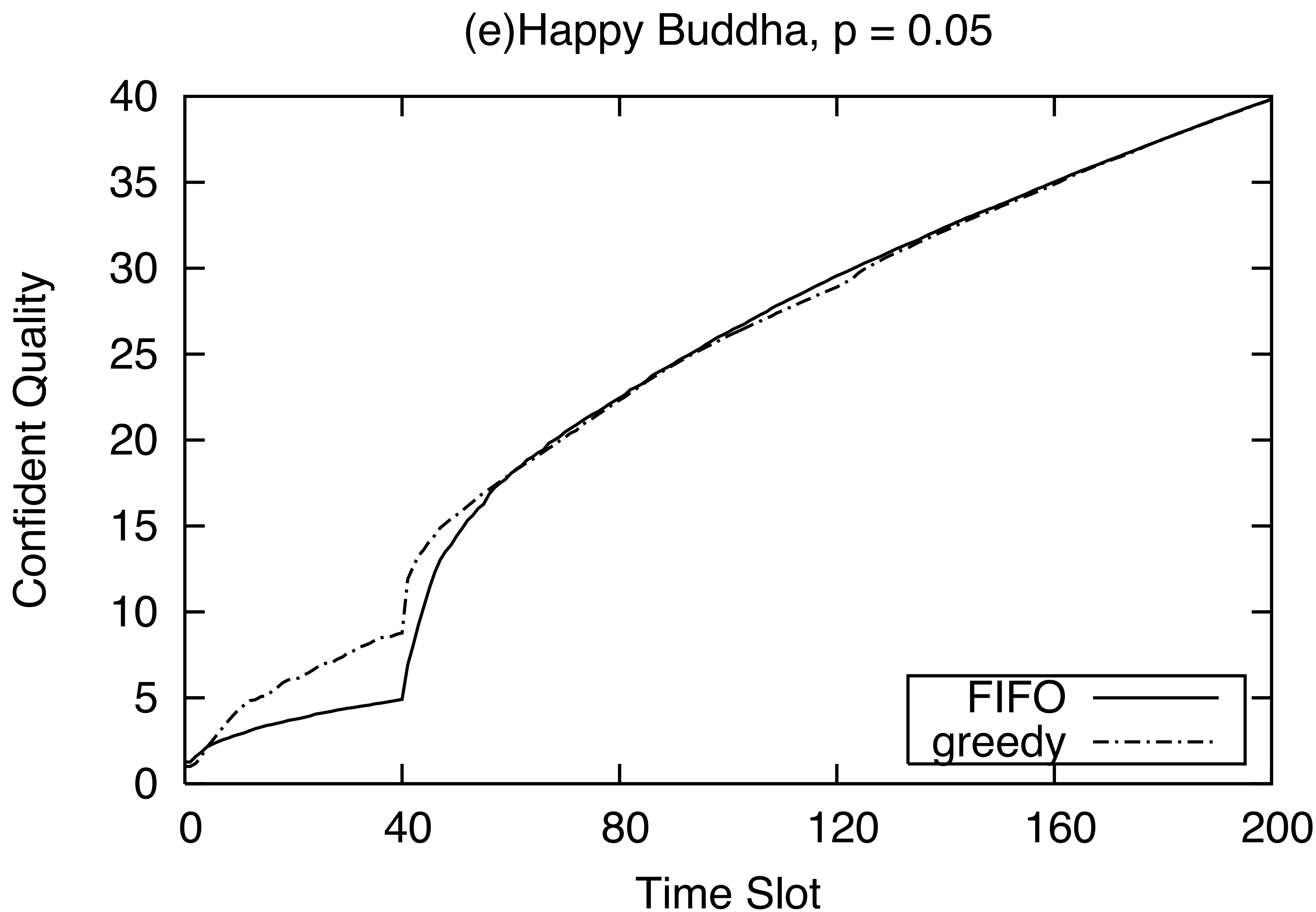
$$\delta_i = w_i(E[D_i^{next}] - E[D_i^{curr}])$$

(only consider nodes whose parents are packed)

maintain a max heap of all nodes using δ_i as key

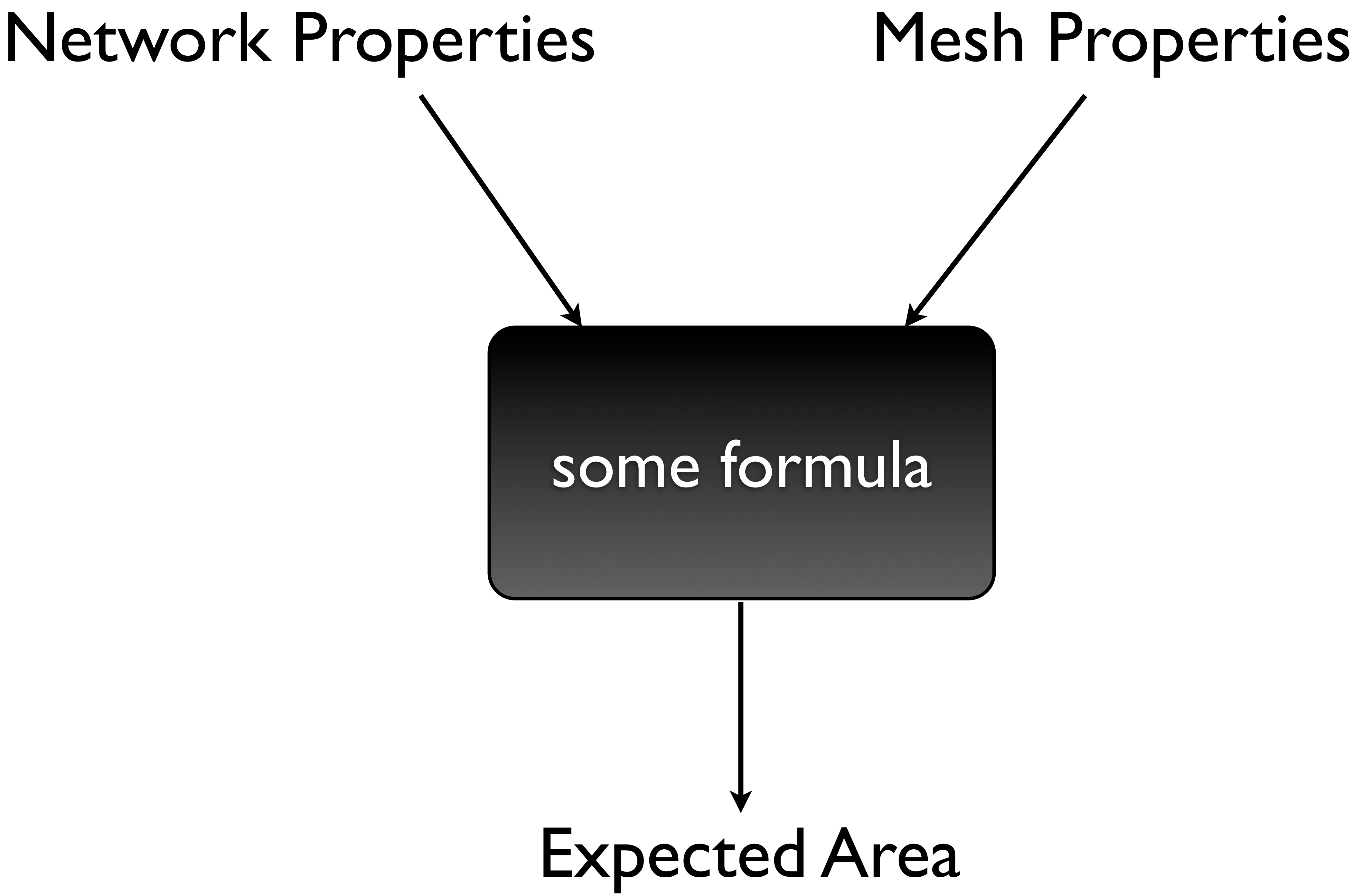
while heap is not empty and packet is not full
  pop a node *i* from heap and packed *i*
  for each child *k* of *i*
    insert *k* into heap


(b)Happy Buddha, p = 0.05
60
50
40
30
20
10
0
0
40
80
120
160
200
240
280
Time Slot
FIFO
greedy
bf
bsub



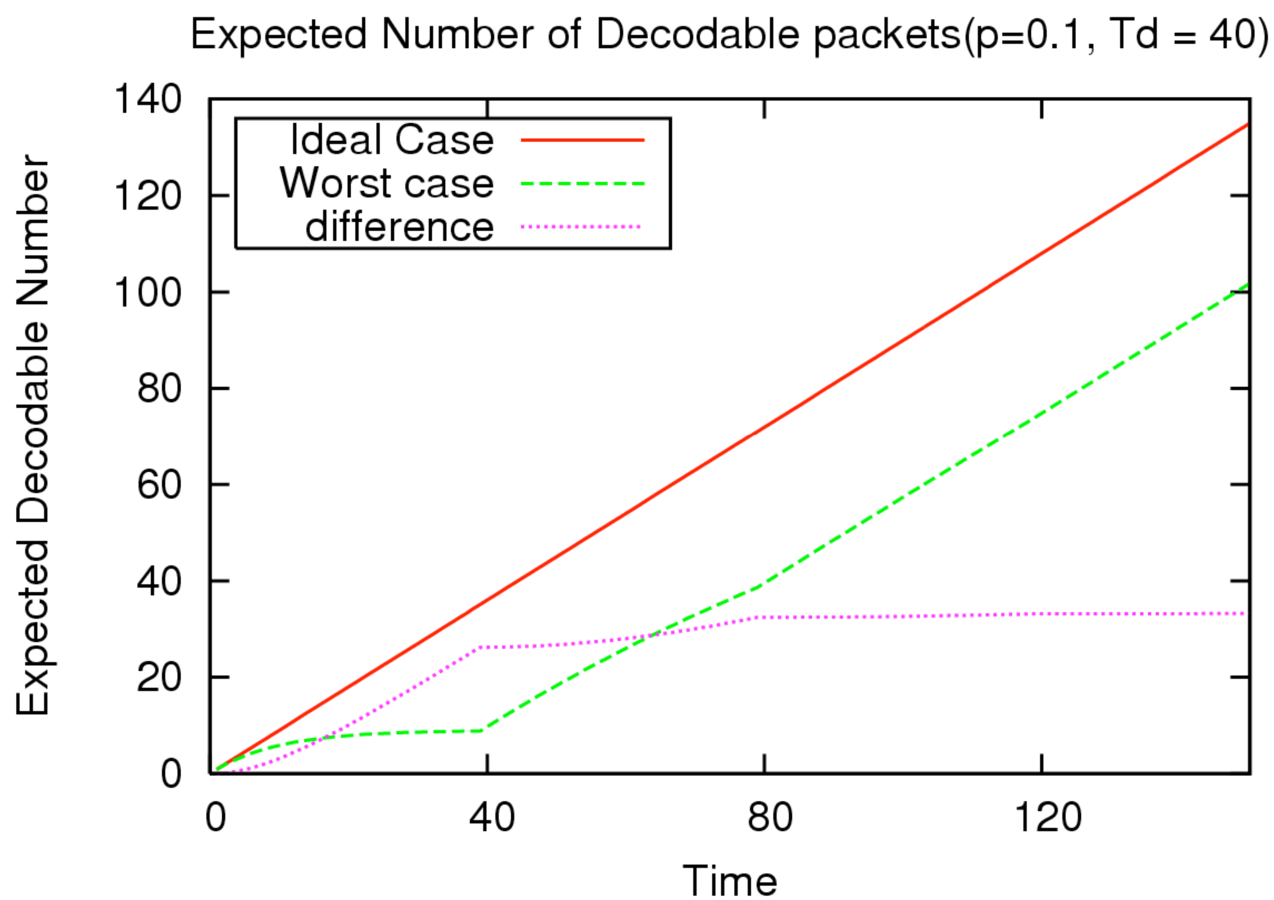
(e)Happy Buddha, p = 0.05
Confident Quality
Time Slot
FIFO
greedy
0
5
10
15
20
25
30
35
40
0
40
80
120
160
200

Summary



Network Properties
Mesh Properties
some formula
Expected Area

Dependencies matter only for a short time initially



謝謝