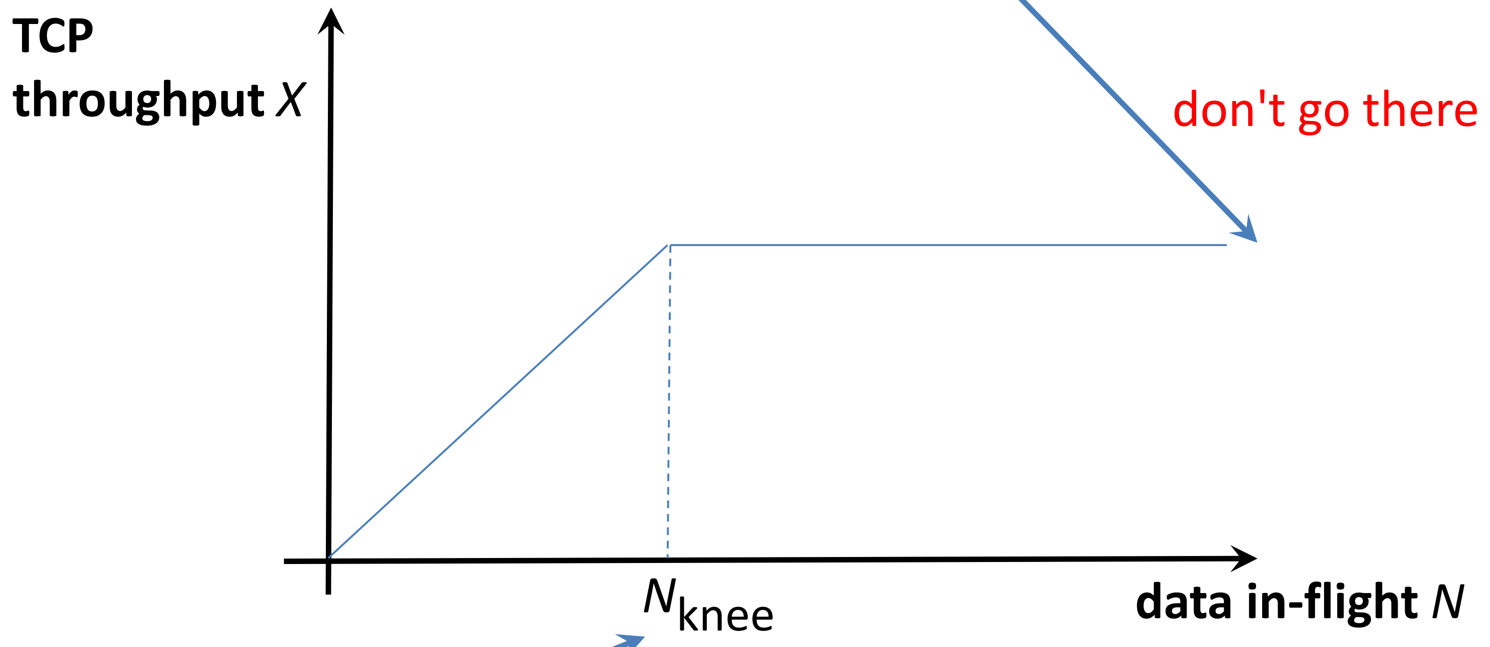
A technique to estimate a system's asymptotic delay and throughput

Y.C. Tay
National University of Singapore

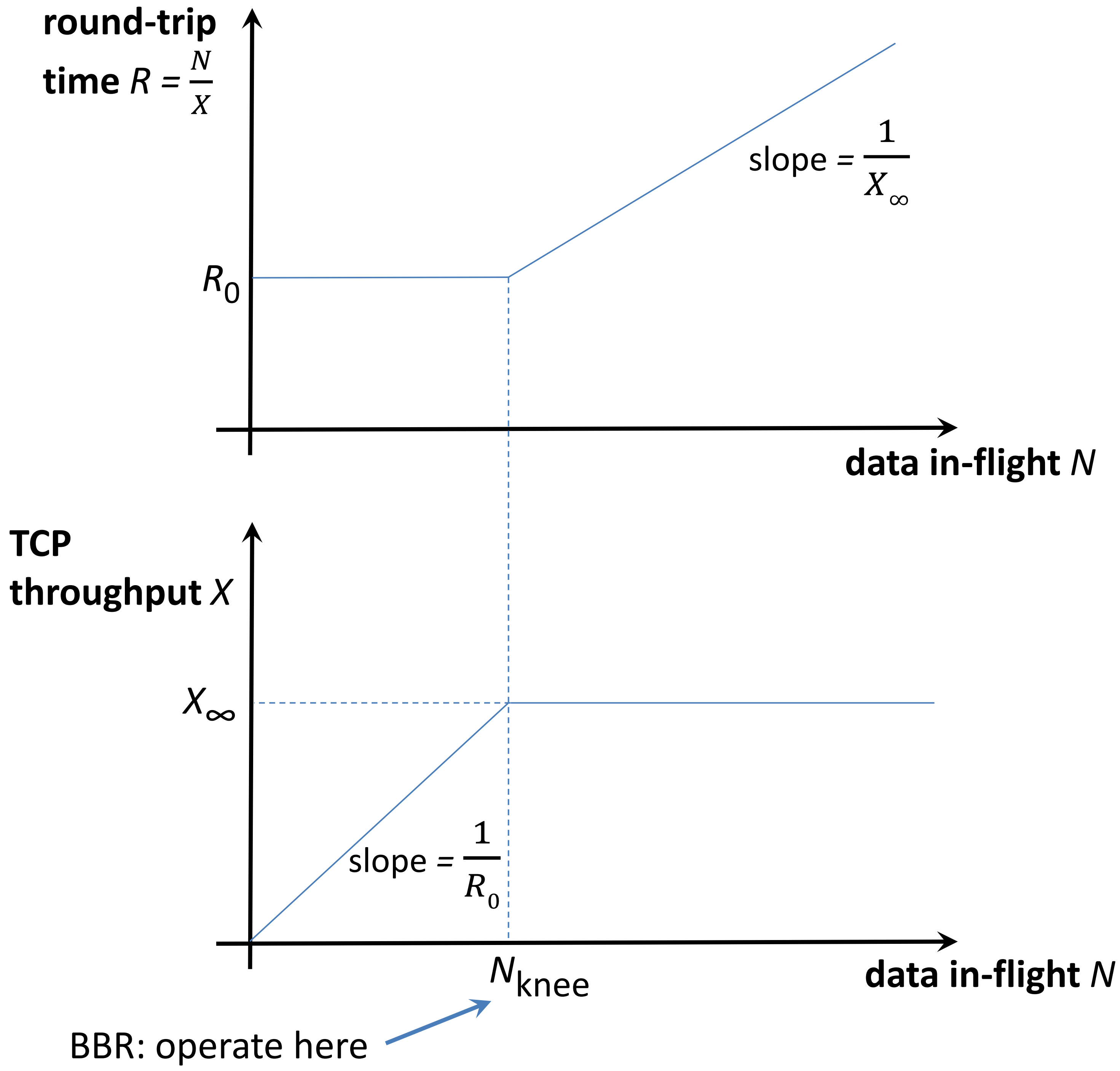
2015--2017: Google moved wide-area (B4) production traffic to TCP variant BBR $\rightarrow$ big performance improvement

(bottlenect bandwidth and round-trip propagation time)

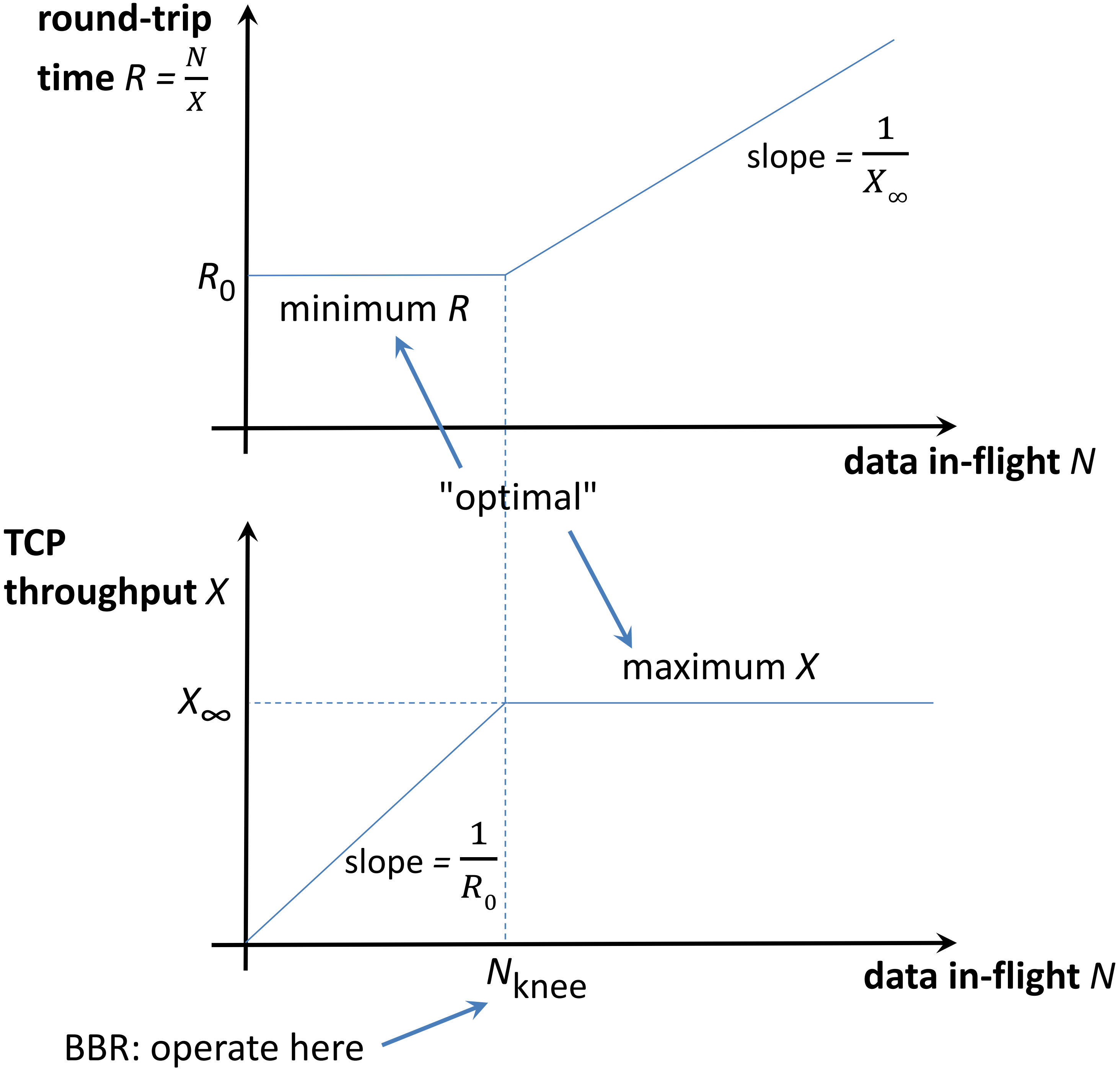
magic: BBR does not use packet loss as congestion signal



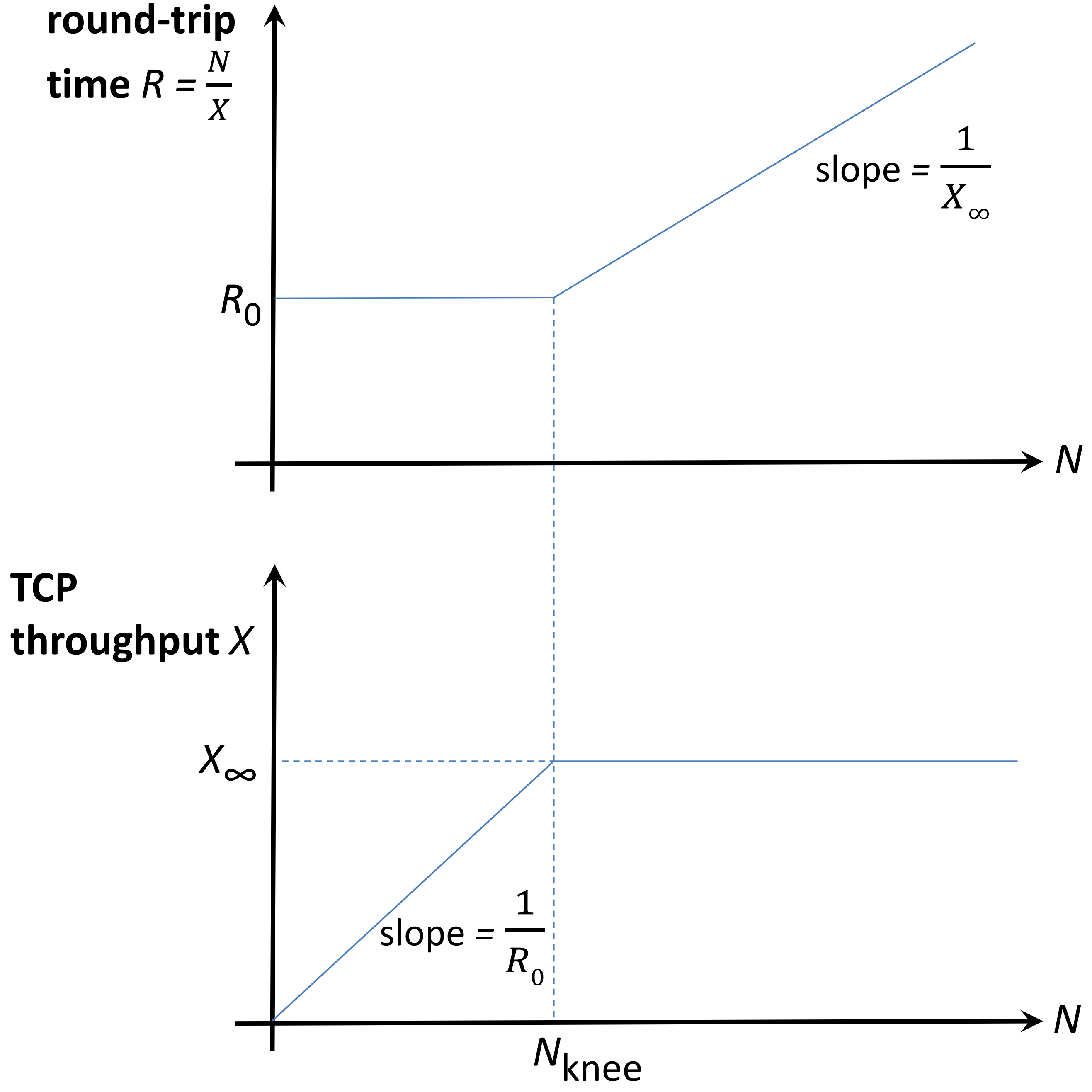
BBR: operate here



round-trip time $R = \frac{N}{X}$
slope = $\frac{1}{X_\infty}$
R_0
data in-flight N
TCP throughput X
X_∞
slope = $\frac{1}{R_0}$
N_{knee}
data in-flight N
BBR: operate here

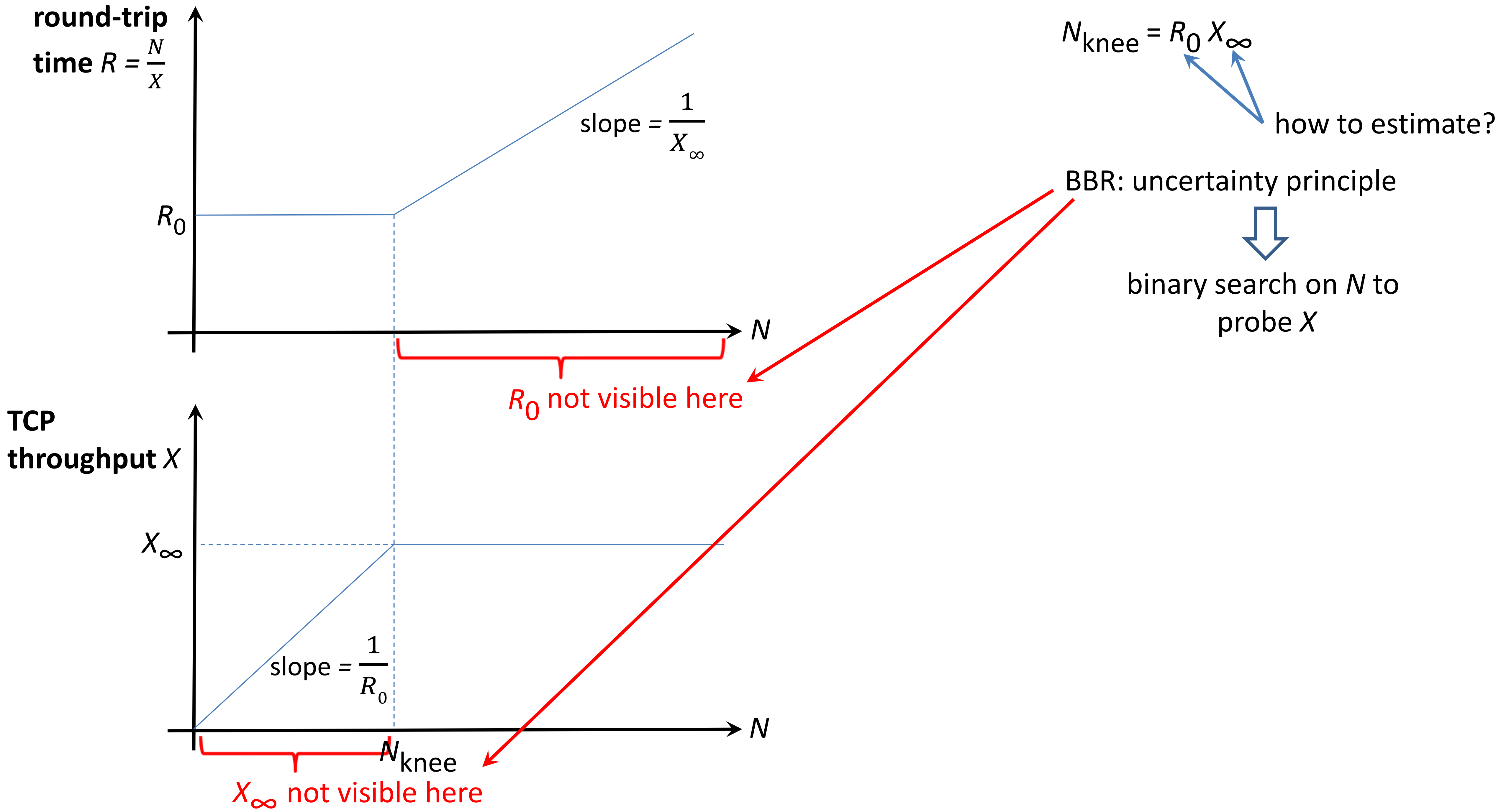


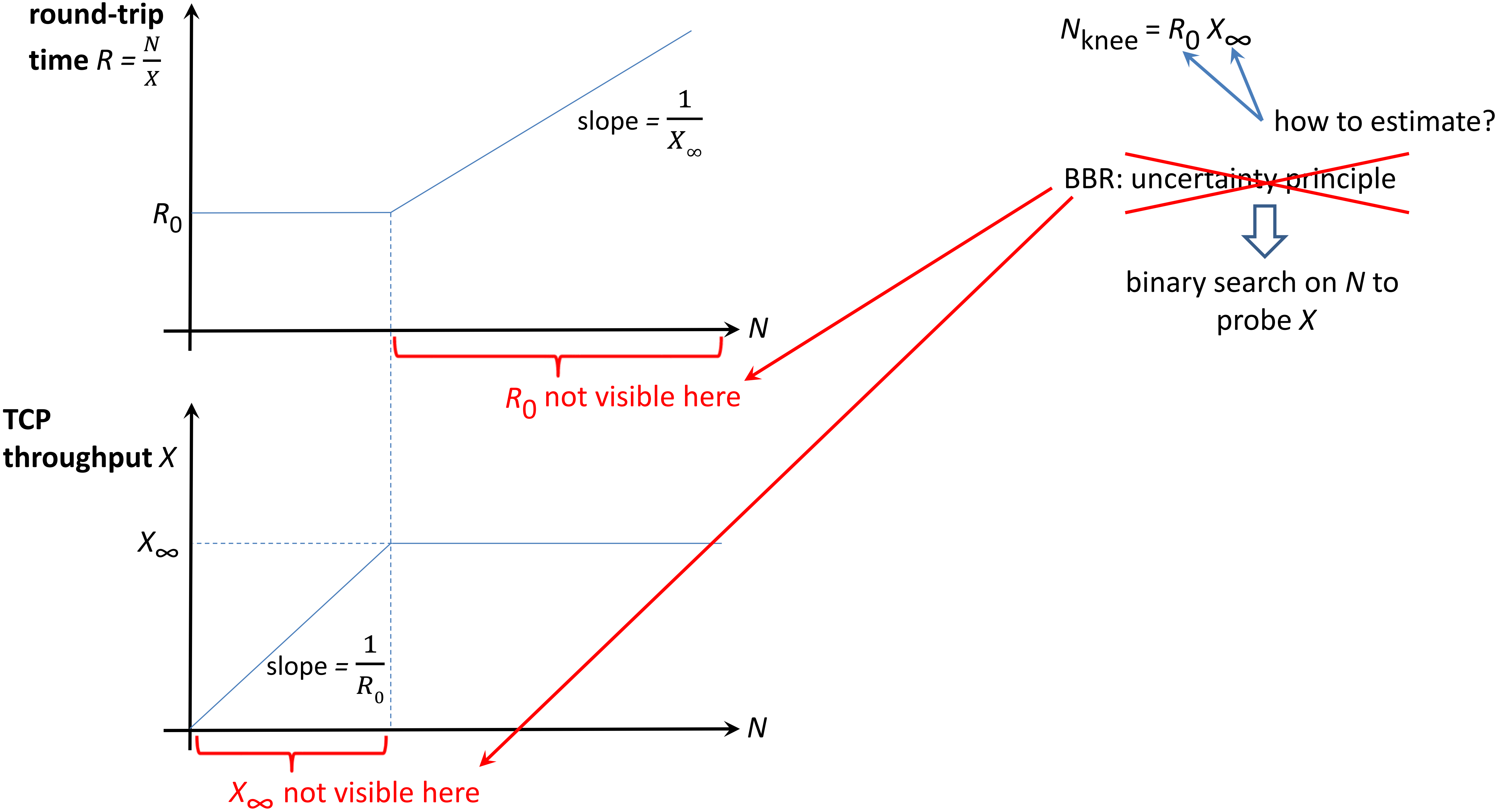
round-trip time $R = \frac{N}{X}$
slope = $\frac{1}{X_\infty}$
R_0
minimum R
data in-flight N
"optimal"
TCP throughput X
maximum X
X_∞
slope = $\frac{1}{R_0}$
N_{knee}
data in-flight N
BBR: operate here

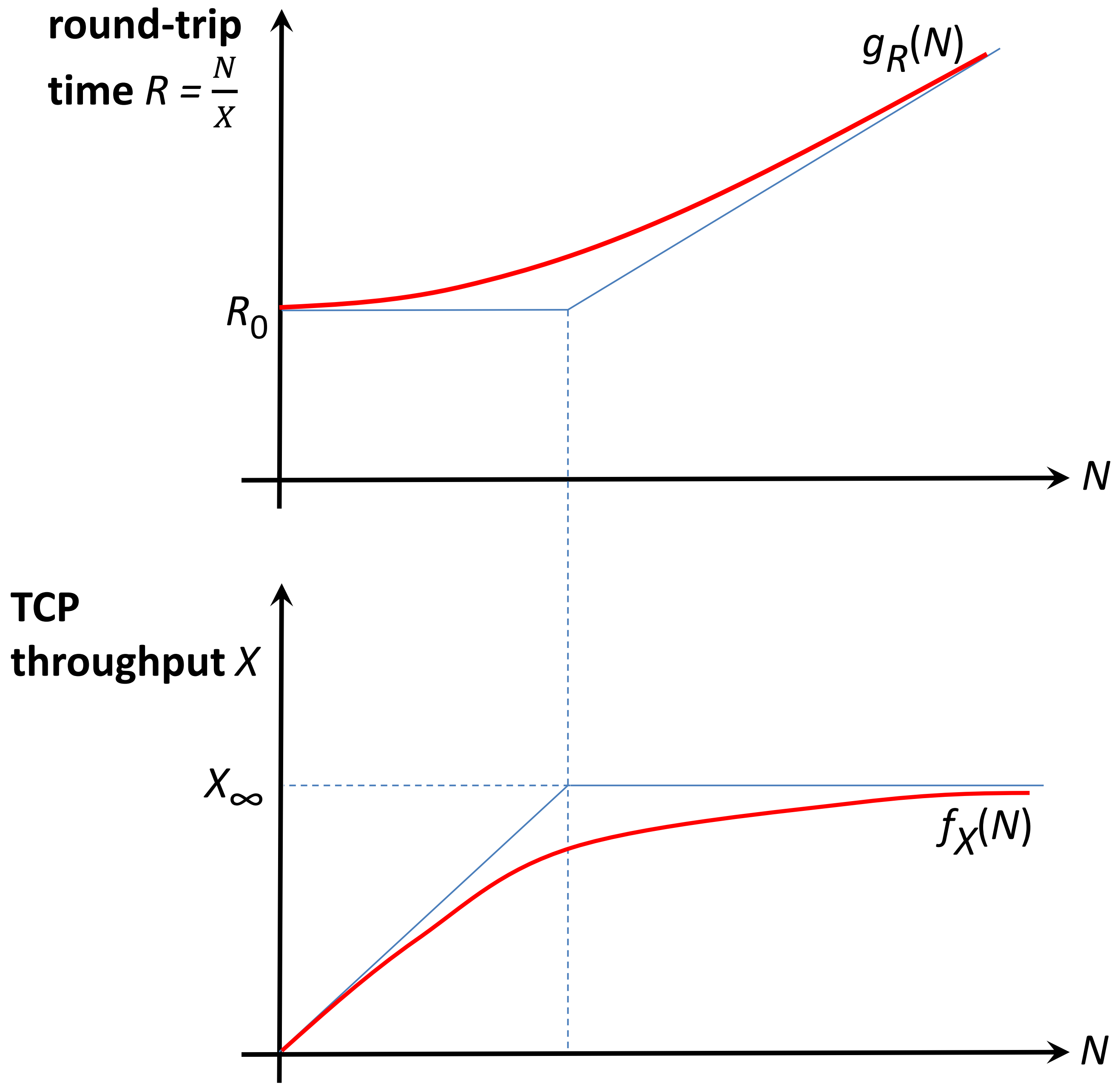


$$N_{\text{knee}} = R_0 \, X_\infty$$

how to estimate?


round-trip time $R = \frac{N}{X}$
slope = $\frac{1}{X_\infty}$
R_0
N
R_0 not visible here
TCP throughput X
X_∞
slope = $\frac{1}{R_0}$
N
N_{knee}
X_∞ not visible here
$N_{knee} = R_0 X_\infty$
how to estimate?
BBR: uncertainty principle
binary search on N to probe X


round-trip time $R = \frac{N}{X}$
slope = $\frac{1}{X_\infty}$
R_0
N
R_0 not visible here
TCP throughput X
X_∞
slope = $\frac{1}{R_0}$
N
X_∞ not visible here
$N_{knee} = R_0 X_\infty$
how to estimate?
~~BBR: uncertainty principle~~
binary search on N to probe X

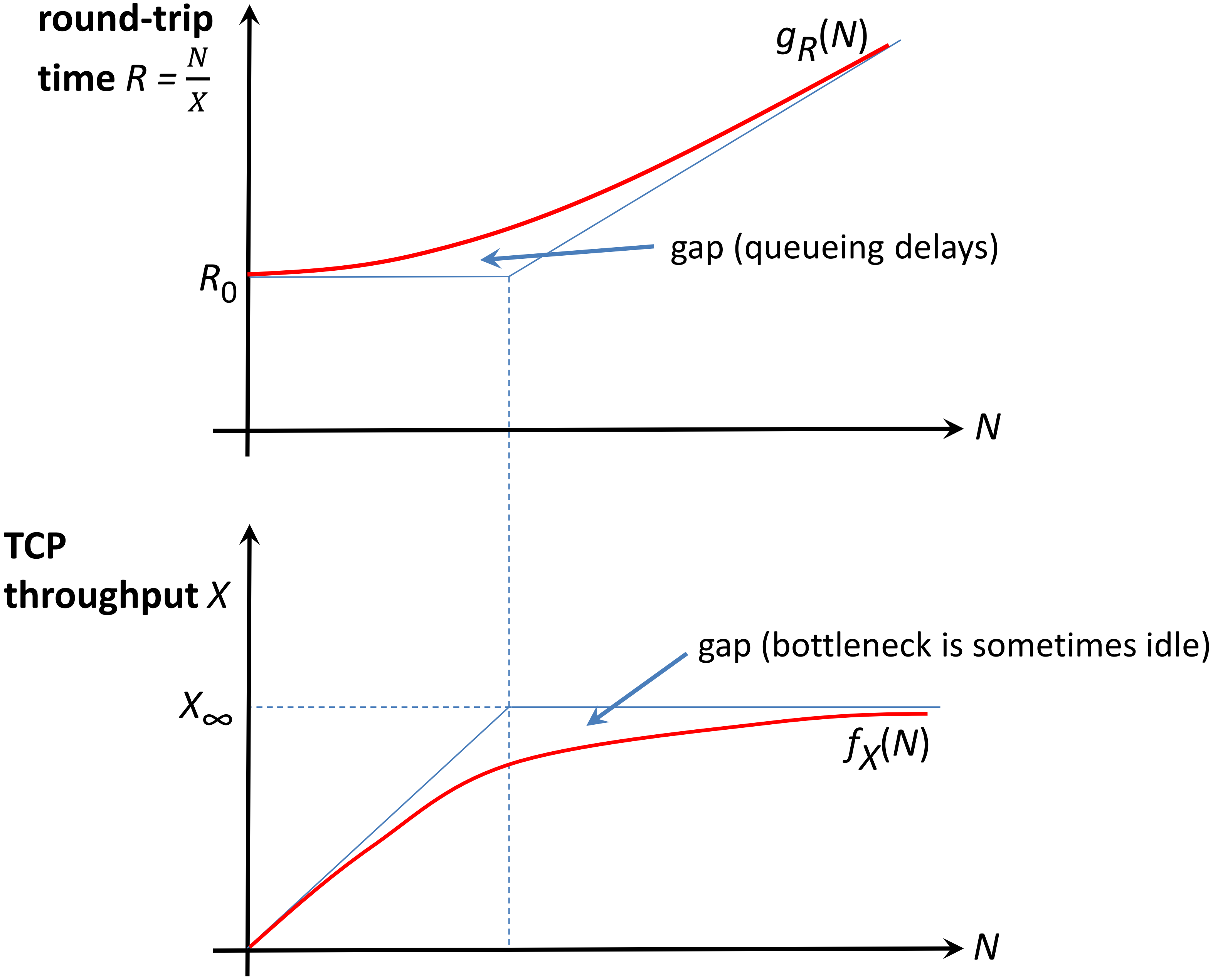


$N_{\text{knee}} = R_0 \, X_\infty$

how to estimate?

~~BBR: uncertainty principle~~

binary search on N to probe X

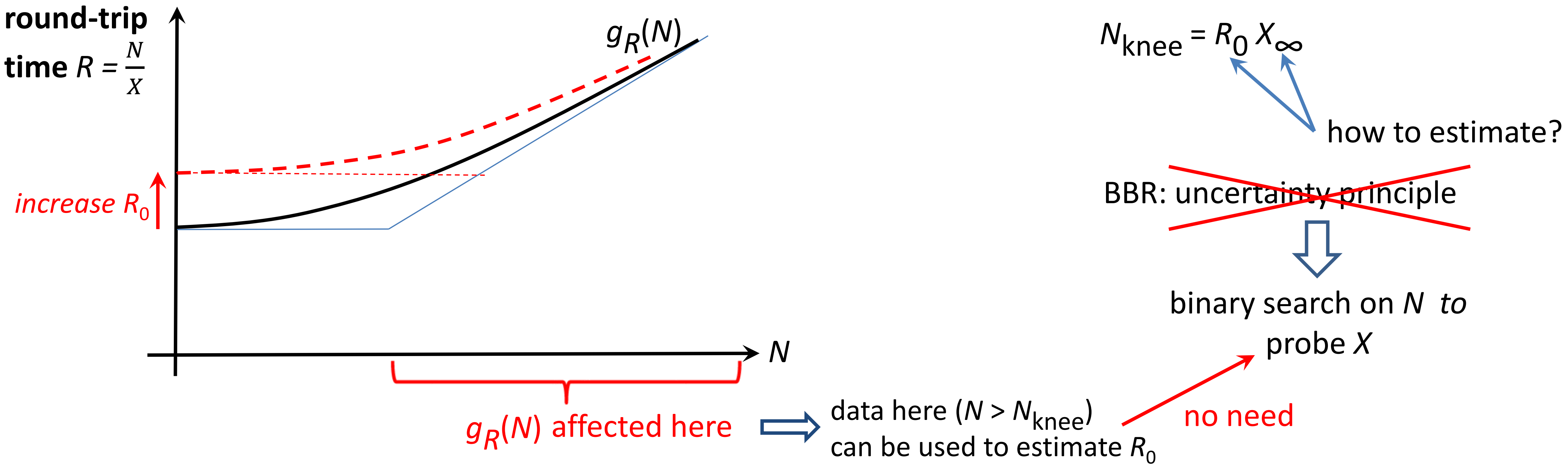


$N_{\text{knee}} = R_0 \, X_\infty$

how to estimate?

~~BBR: uncertainty principle~~

binary search on N to probe X



round-trip
time $R = \frac{N}{X}$
$g_R(N)$
increase R_0
N
$g_R(N)$ affected here
data here ($N > N_{\text{knee}}$)
can be used to estimate R_0
no need
$N_{\text{knee}} = R_0 X_\infty$
how to estimate?
~~BBR: uncertainty principle~~
binary search on N *to*
probe X

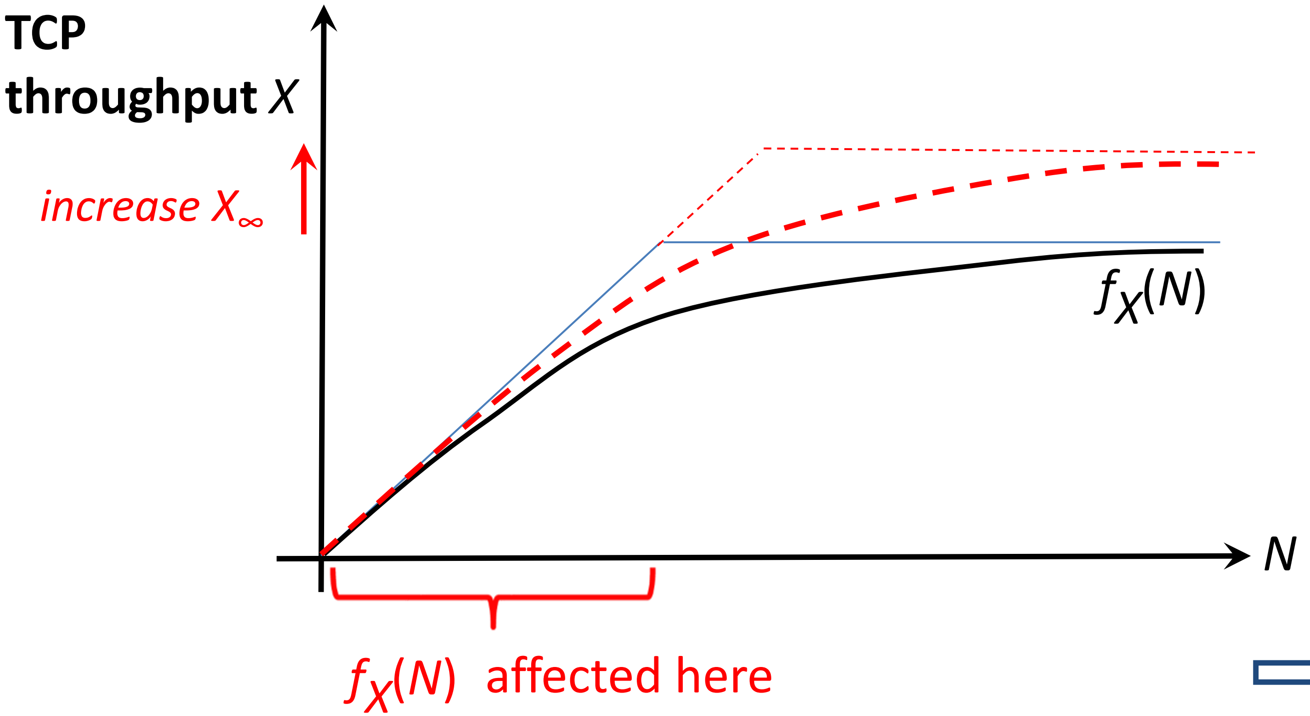
$N_{knee} = R_0 X_\infty$

how to estimate?

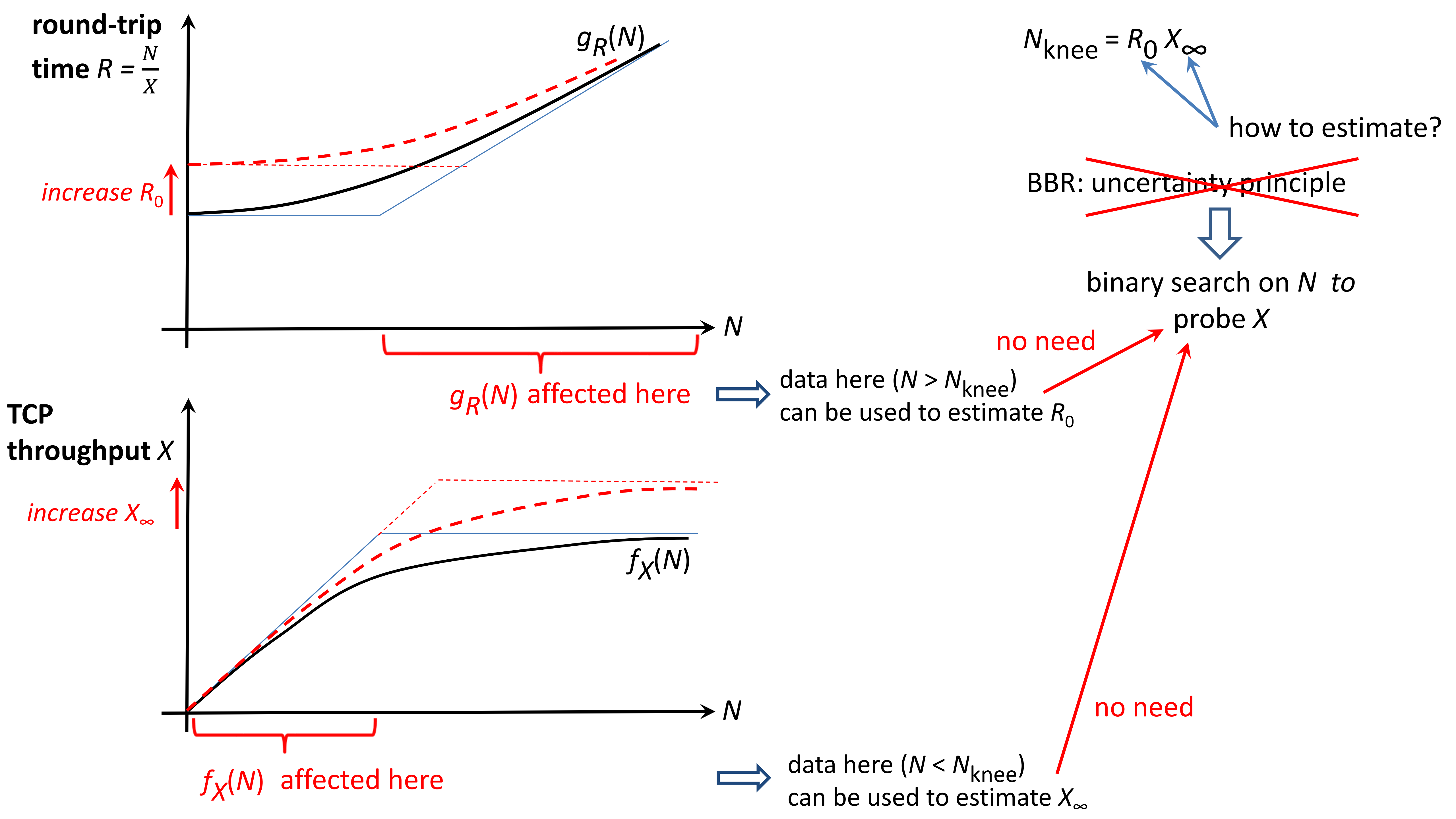
~~BBR: uncertainty principle~~

binary search on *N* *to* probe *X*

no need



data here ($N < N_{knee}$) can be used to estimate X_∞


round-trip time $R = \frac{N}{X}$
$g_R(N)$
increase R_0
N
$g_R(N)$ affected here
data here ($N > N_{knee}$) can be used to estimate R_0
TCP throughput X
increase X_∞
$f_X(N)$
N
$f_X(N)$ affected here
data here ($N < N_{knee}$) can be used to estimate X_∞
$N_{knee} = R_0 X_\infty$
how to estimate?
~~BBR: uncertainty principle~~
binary search on N to probe X
no need
no need

Balanced Job Bound for separable networks [Zahorjan et al., 1982]

$$X > \frac{N}{R_0 + \frac{N}{X_\infty}}$$

$$N_{knee} = R_0 X_\infty$$

how to estimate?

real networks: not balanced, not separable $\Rightarrow$ not a bound

idea: use $f_X^B(N) = \dfrac{N}{R_0 + \frac{N}{X_\infty}}$ to approximate X

asymptotically correct:

$$\lim_{N \to 0} \frac{N}{f_X^B(N)} = \lim_{N \to 0} \left(R_0 + \frac{N}{X_\infty}\right) = R_0$$

$$\lim_{N \to \infty} f_X^B(N) = \lim_{N \to \infty} \frac{N}{R_0 + \frac{N}{X_\infty}} = X_\infty$$

$$X = \frac{N}{R_0 + \frac{N}{X_\infty}} \Rightarrow \frac{1}{X} = R_0 \frac{1}{N} + \frac{1}{X_\infty}$$

linear regression

measure $\{ <N_1, X_1>, <N_2, X_2>, \ldots, <N_k, X_k> \}$

$$\longrightarrow \left\{ <\frac{1}{N_1}, \frac{1}{X_1}>, <\frac{1}{N_2}, \frac{1}{X_2}>, \ldots, <\frac{1}{N_k}, \frac{1}{X_k}> \right\} \longrightarrow R_0, X_\infty \longrightarrow N_{knee}$$

Different TCP variants (Tahoe, Reno, CUBIC, BBR, …) have the same R_0 and X_∞ but $g_R(N)$ and $f_X(N)$ have different curvatures.

How to model?

idea: introduce 2 parameters for how variants affect X and N

$$\frac{1}{X+\alpha} = R_0 \frac{1}{N(1+\beta)} + \frac{1}{X_\infty}$$

Many more experiments needed to understand the model’s accuracy.

Some engineering needed to apply the model to TCP congestion control.

General technique for any closed black box

Another example: video game developer has choice of

hardware: Nvidia or AMD

game engine: Unity or Unreal Engine

how to choose?

choice depends on:

R_0 = time to render 1 frame

X_∞ = maximum frame rate

issue: how to determine R_0 and X_∞

using a game prototype (with limited number of objects/scenarios)?

answer: fit $\frac{1}{X} = R_0 \frac{1}{N} + \frac{1}{X_\infty}$ with prototype