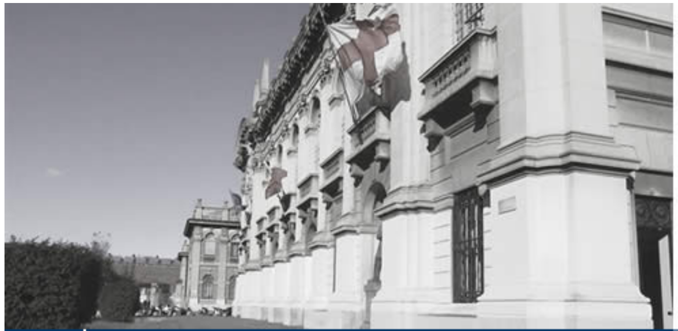


Computing Infrastructures

POLITECNICO DI MILANO

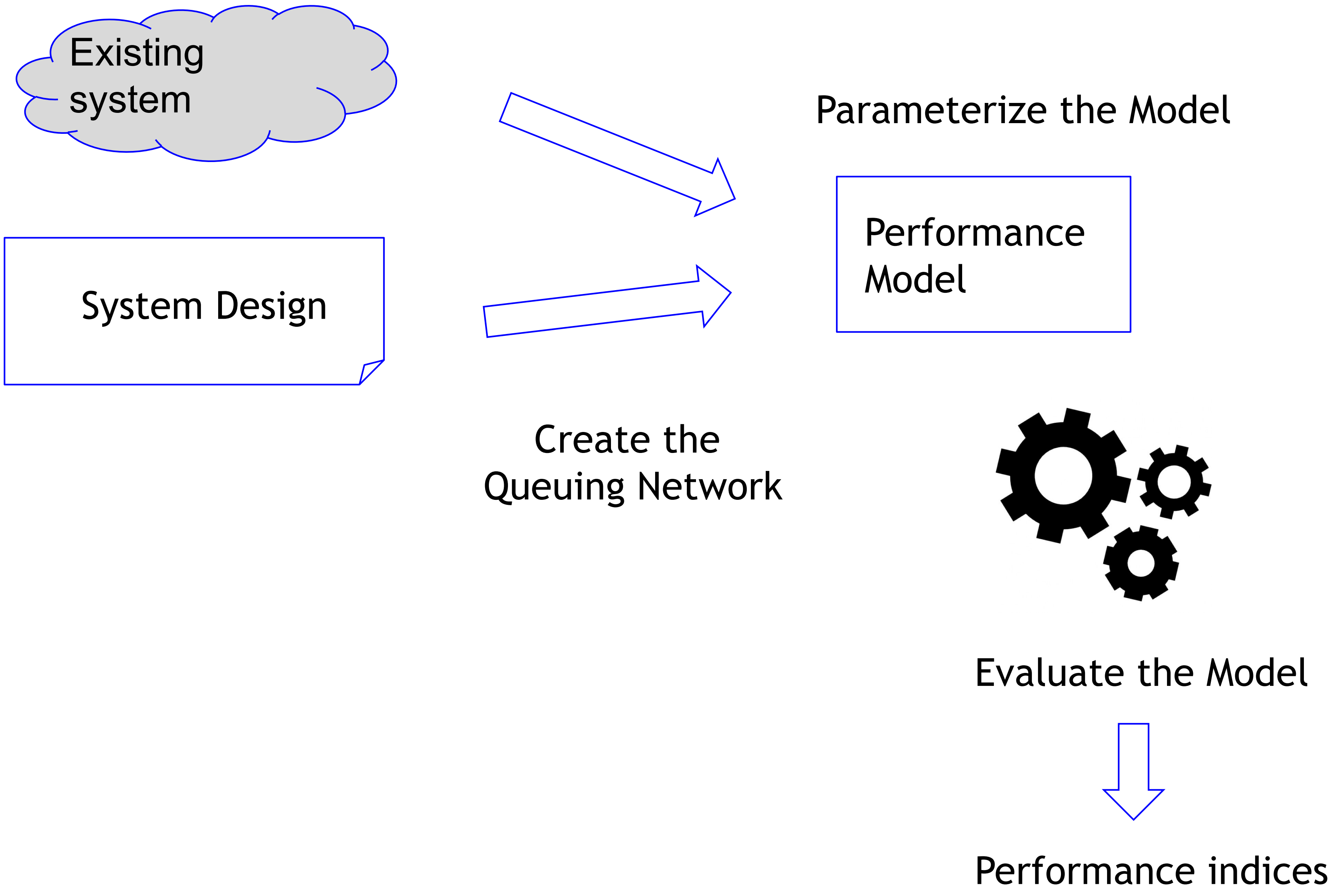
    

Performance Bounds

Prof. Danilo Ardagna

Credits: Raffaela Mirandola,
Jane Hilston, Ed Lazowska,
Marco Gribaudo, Moreno Marzolla

Model Definition for Performance Studies



Model Definition for Performance Studies

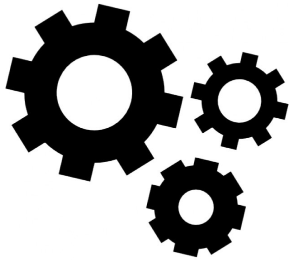
Existing system

System Design

Create the Queuing Network

Parameterize the Model

Performance Model



Evaluate the Model

Performance indices

Performance bounds

- Provide valuable insight into the primary factors affecting the performance of computer system
- Can be computed quickly and easily therefore serve as a first cut modeling technique
- Several alternatives can be treated together

Bounding Analysis

- We will consider single class systems only

- Determine *asymptotic bounds*, *i.e.*, ***upper*** and ***lower bounds*** on a system's performance indices X and R:
  - In our case, we will treat X and R bounds as functions of *number of users* or *arrival rate* (i.e., λ or N)

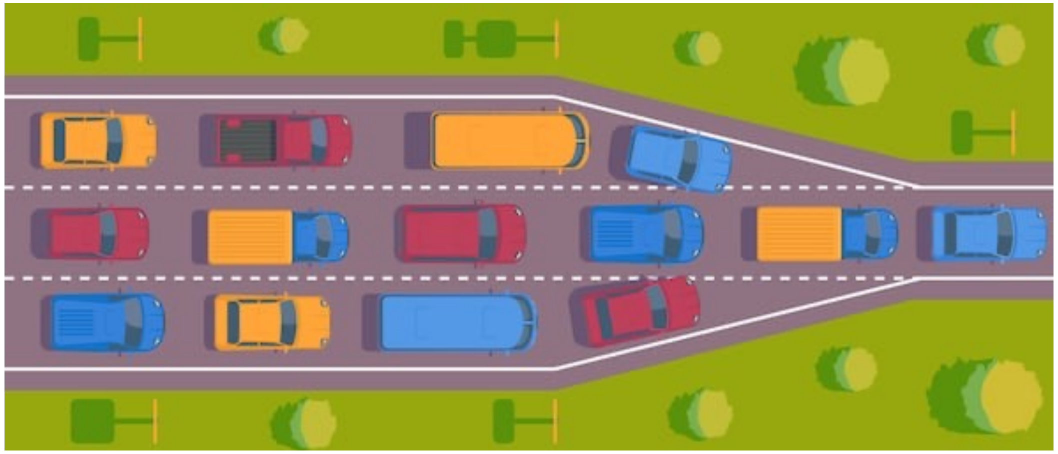
- Advantages of bounding analysis:
  - Highlight and quantify the critical influence of the system ***bottleneck***

Bottleneck

- The resource within a system which has the greatest service demand is known as the bottleneck resource or bottleneck device, and its service demand is $\max_k \{D_k\}$, denoted D_{max}

- The bottleneck resource is important because it limits the possible performance of the system

- This will be the resource which has the highest utilisation in the system



Bounding Analysis

- Advantages of bounding analysis:
  - Highlight and quantify the critical influence of the system *bottleneck*
  - Can be computed quickly, even by hand

  - Useful in System Sizing:
    - Based on preliminary estimates (quickness)
    - This kind of studies involve typically a large number of candidate configurations with a single critical resource (e.g., CPU) dominant and the other configured accordingly: *treated as one alternative*

  - Useful for System Upgrades…

Notation

The considered models and the bounding analysis make use of the following parameters:

- K, the number of service centers
- D, the sum of the service demands at the centers, so

  $D=\sum_k D_k$

- D_{max}, the largest service demand at any single center
- Z, the average think time, for interactive systems

And the following performance quantities are considered:

- X, the system throughput
- R, the system response time

Bounding Analysis - *Asymptotic bounds*

- Are derived by considering the (asymptotically) extreme conditions of light and heavy loads:
  - *Optimistic*: ***X*** upper bound and ***R*** lower bound
  - *Pessimistic*: ***X*** lower bound and ***R*** upper bound

- Under the extreme conditions of:
  - *Light load*
  - *Heavy load*

- **Under the assumption that:**
  - the service ***demand*** of a customer at a center does not depend on how many other customers currently are in the system, or at which service centers they are located

Bounding Analysis - *Asymptotic bounds*

Open models: less information than in closed models…

***X* bound** = the $\underbrace{\textit{maximum arrival rate}}$ that the system can process

if λ > *X* bound → the system $\underbrace{\textbf{\textit{SATURATES}}}$

new jobs have to wait an indefinitely long time

Remembering that $U_k = XD_k$

$$U_{max}(\lambda) = \lambda D_{max} \leqslant 1$$

The *X* bound is calculated as:

Bounding Analysis - *Asymptotic bounds*

Open models: less information than in closed models…

***X* bound** = the $\underbrace{\textit{maximum arrival rate}}$ that the system can process

if λ > *X* bound → the system $\underbrace{\textit{SATURATES}}$

new jobs have to wait an indefinitely long time

Remembering that $U_k = XD_k$

$$U_{max}(\lambda) = \lambda D_{max} \leqslant 1$$

The *X* bound is calculated as:

$$\lambda_{sat} = \frac{1}{D_{max}}$$

Bounding Analysis - *Asymptotic bounds*

Open models:

***R* bounds** = the largest and smallest possible **R** experienced at a given λ investigated only when $\lambda < \lambda_{sat}$ (otherwise the system is unstable!)

2 extreme situations:

1. If no customers interferes with any other (= no queue time)

   Then $R = D$, with $D = \sum_k D_k$

Bounding Analysis - *Asymptotic bounds*

Open models:

2. There is no pessimistic bound on ***R***:
    - if ***n*** customers arrives together every ***n/λ*** time units (the system arrival rate is n / (n/ λ)= λ)
    - customers at the end of the batch are forced to queue for customers at the front of the batch, and thus experience large response times

Bounding Analysis - *Asymptotic bounds*

Open models:

2. There is no pessimistic bound on ***R***:
   - if ***n*** customers arrives together every ***n/λ*** time units (the system arrival rate is n /(n/ λ)= λ)
   - customers at the end of the batch are forced to queue for customers at the front of the batch, and thus experience large response times
   - as the batch size n increases, more and more customers are waiting an increasingly long time
   - thus, for any postulated pessimistic bound on response times for system arrival rate λ, it is possible to pick a batch size n sufficiently large that the bound is exceeded

There is no pessimistic bound on response times, regardless of how small the arrival rate λ might be

Bounding analysis: Open models

Bound for X(λ)

$$X(\lambda) \leq \frac{1}{D_{max}}$$

Bound for R(λ)

$$R(\lambda) \geq D$$

Bounding analysis: Open models

$$R(\lambda) \geq D$$

$$X(\lambda) \leq \frac{1}{D_{max}}$$

Bounding Analysis - *Asymptotic bounds*

Closed models:

X **bounds** considered first, then converted in ***R*** **bounds** using Little's Law

Light Load situation (lower bounds)**:**

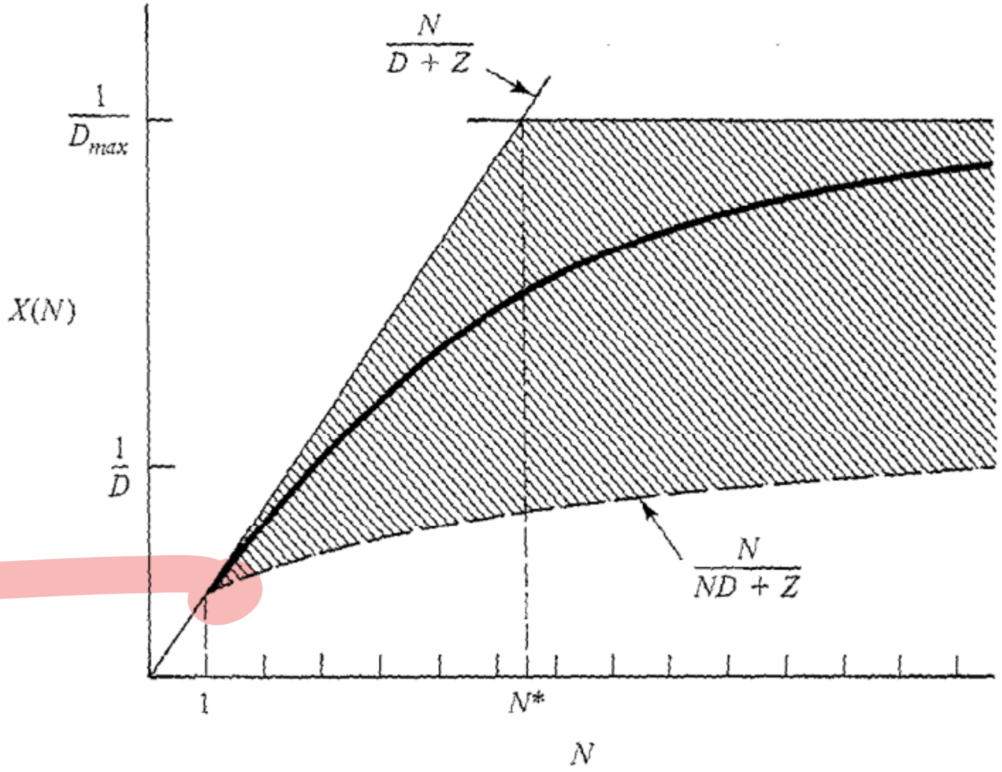
1 customer case:

$N = X\,(R + Z)$

$1 = X\,(D + Z)$

Then *X* is:

$X = 1 / (D + Z)$

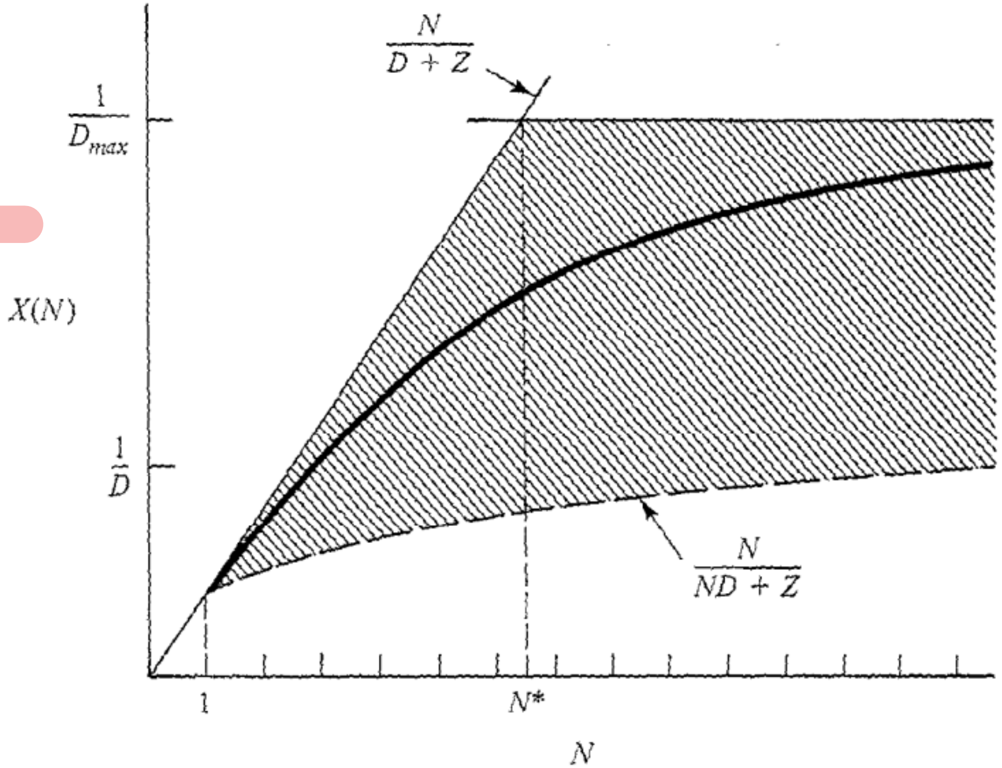


Closed models:

Light Load situation (lower bounds):

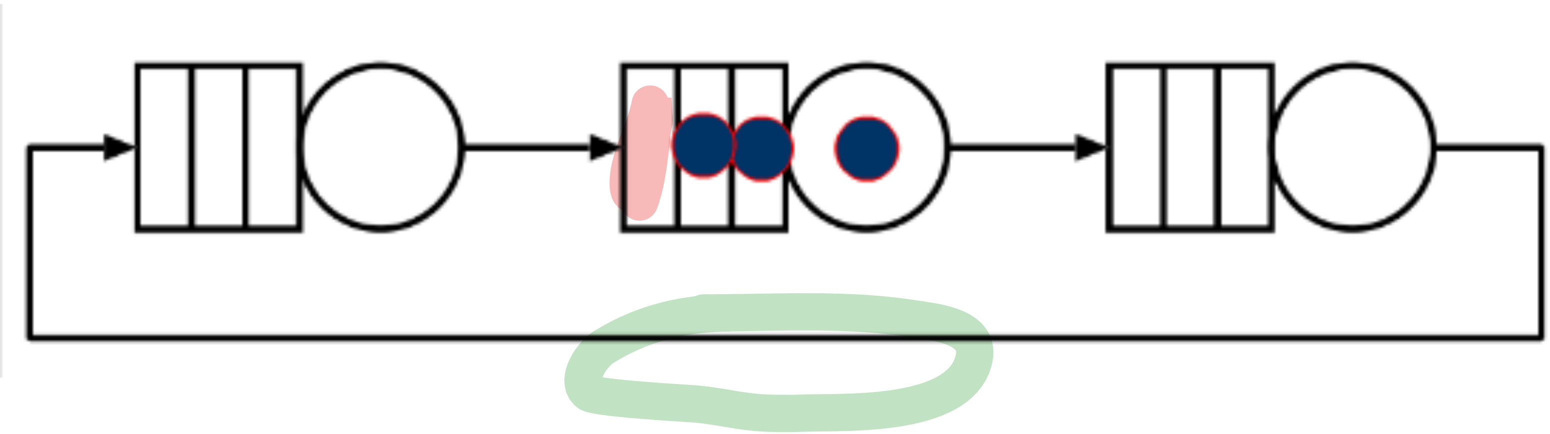
Adding customers:

Smallest *X* obtained with largest R, i.e., new jobs queue behind others already in the system



Bounding Analysis - *Asymptotic bounds*

In closed models, the highest possible system response time occurs when each job, at each station, founds all the other *N-1* costumers in front of it



Closed models:

Light Load situation (lower bounds)**:**

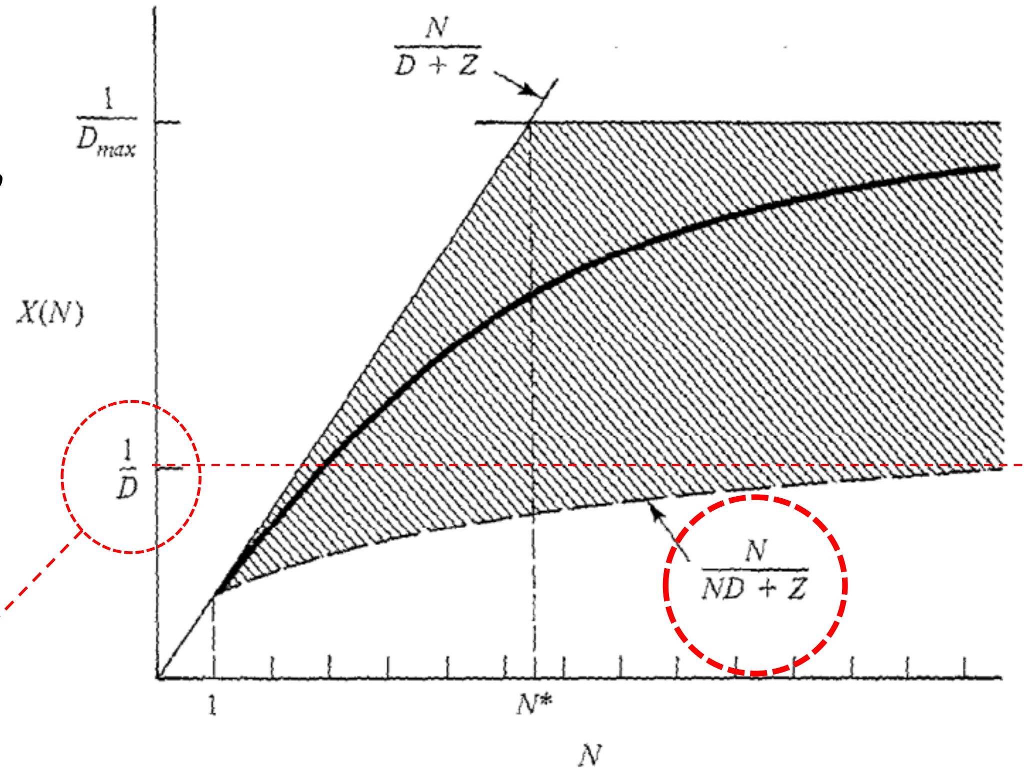
<u>Adding customers:</u>

Smallest ***X*** obtained with largest R, i.e., new jobs queue behind others already in the system

In this case the **X** is:

$$X = N / (ND + Z)$$

$$\lim_{N \to \infty} N / (ND + Z) = 1 / D$$

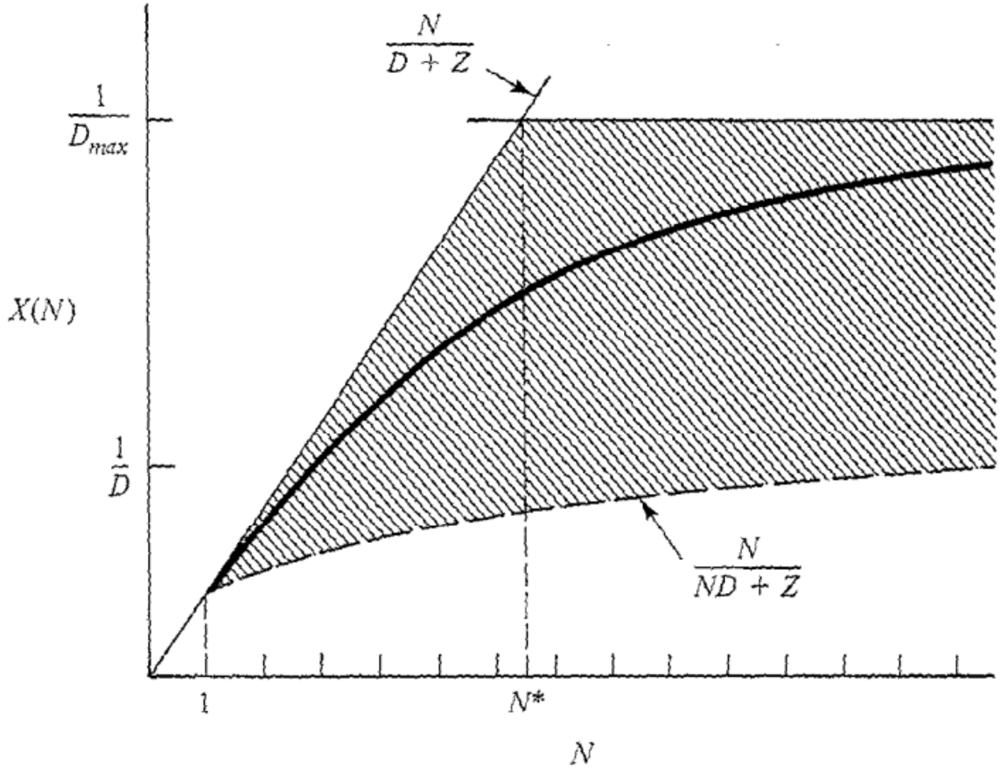


Closed models:

Light Load situation (upper bounds)**:**

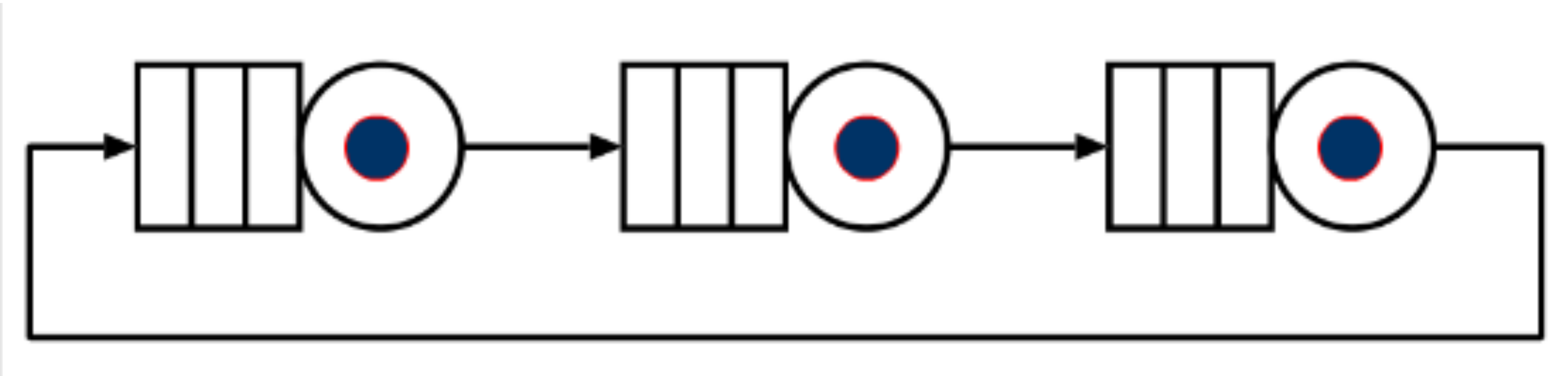
Adding customers:

Largest ***X*** obtained with the lowest response time R



Asymptotic Bounds - Closed Models

The lowest response time can be obtained if a job always finds the queue empty and always starts being served immediately



Closed models:

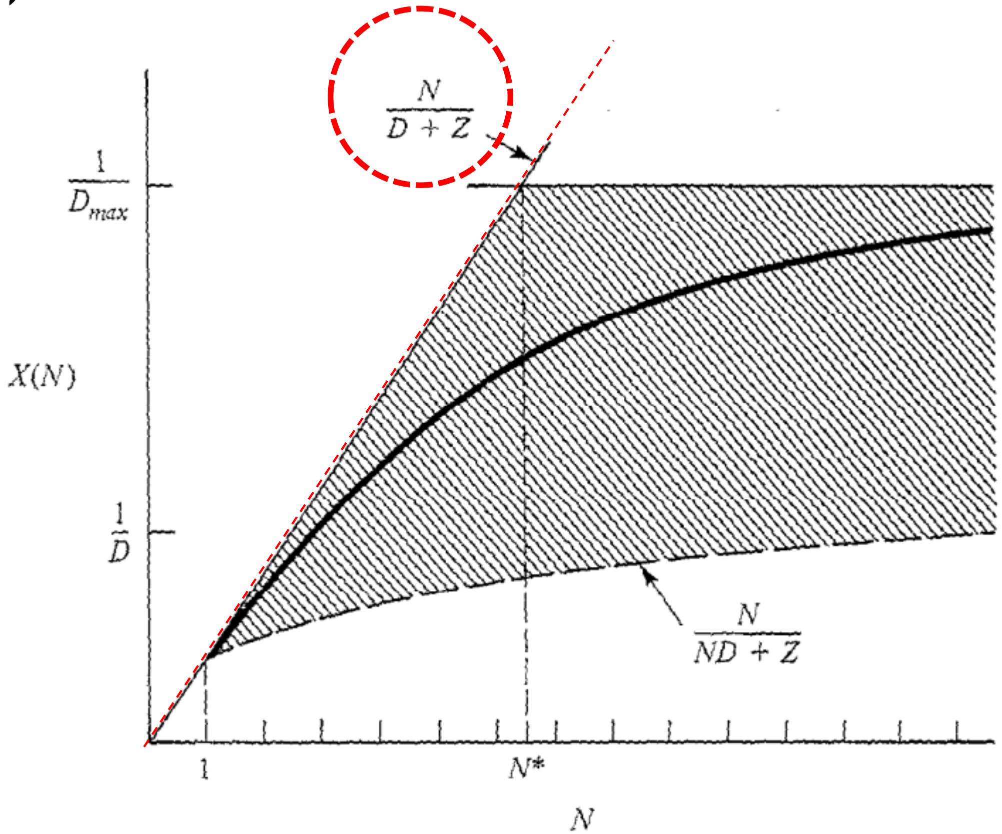
Light Load situation (upper bounds)**:**

Adding customers:

Largest ***X*** if new jobs never queue behind other already in the system:

In this case the **X** is:

$$X = N / (D + Z)$$



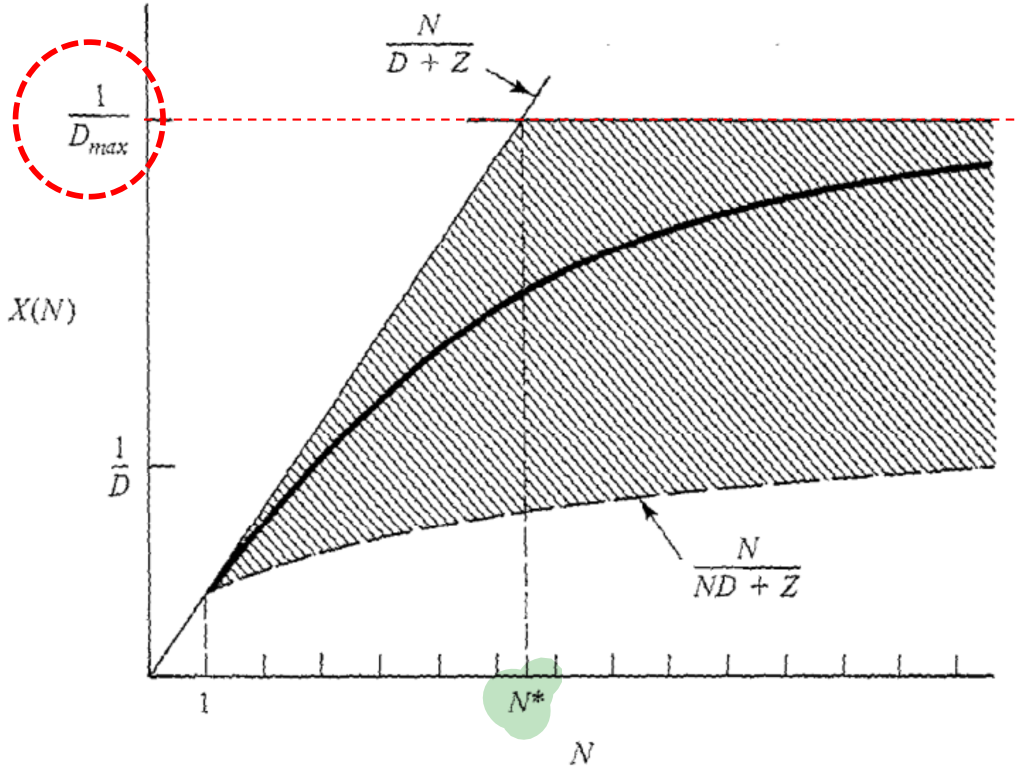
Closed models:

Heavy Load situation (upper bound)**:**

$$U_k(N) = X(N)\,D_k \leqslant 1$$

Since the first to saturate is the **Bottleneck** (max)**:**

$$X(N) \leqslant \frac{1}{D_{max}}$$



Bounding Analysis - *Asymptotic bounds.*

Closed models:

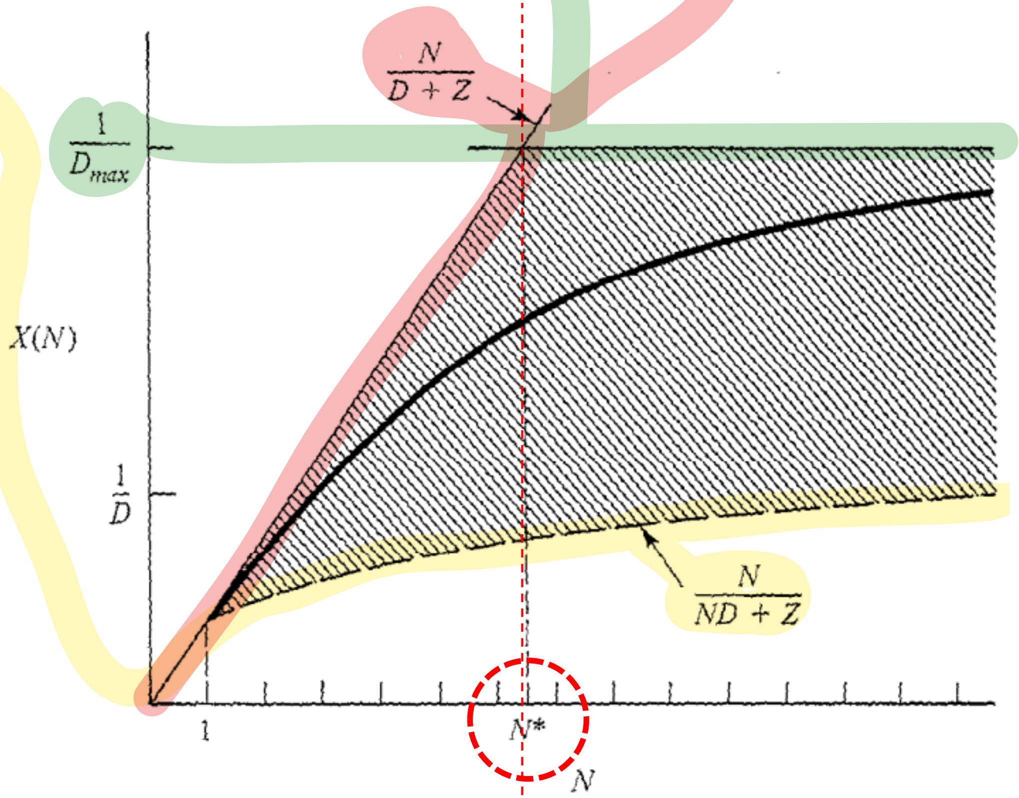
X(N) bounds:

$$\frac{N}{ND+Z} \leqslant X(N) \leqslant \min\left(\frac{1}{D_{max}}, \frac{N}{D+Z}\right)$$

N*:

Particular population size determining if the light or the heavy load optimistic bound is to be applied

$$N^* = \frac{D+Z}{D_{max}}$$



Bounding Analysis - *Asymptotic bounds.*

R(N) bounds:

Let us simply rewrite the previous equation, considering that: X(N)=N/(R(N)+Z), we have:

$$\frac{N}{ND+Z} \leq \frac{N}{R(N)+Z} \leq \min\left(\frac{1}{D_{max}}, \frac{N}{D+Z}\right)$$

And to have R as numerator we invert the members and we have

$$\max\left(D_{max}, \frac{D+Z}{N}\right) \leq \frac{R(N)+Z}{N} \leq \frac{ND+Z}{N}$$

From which we have

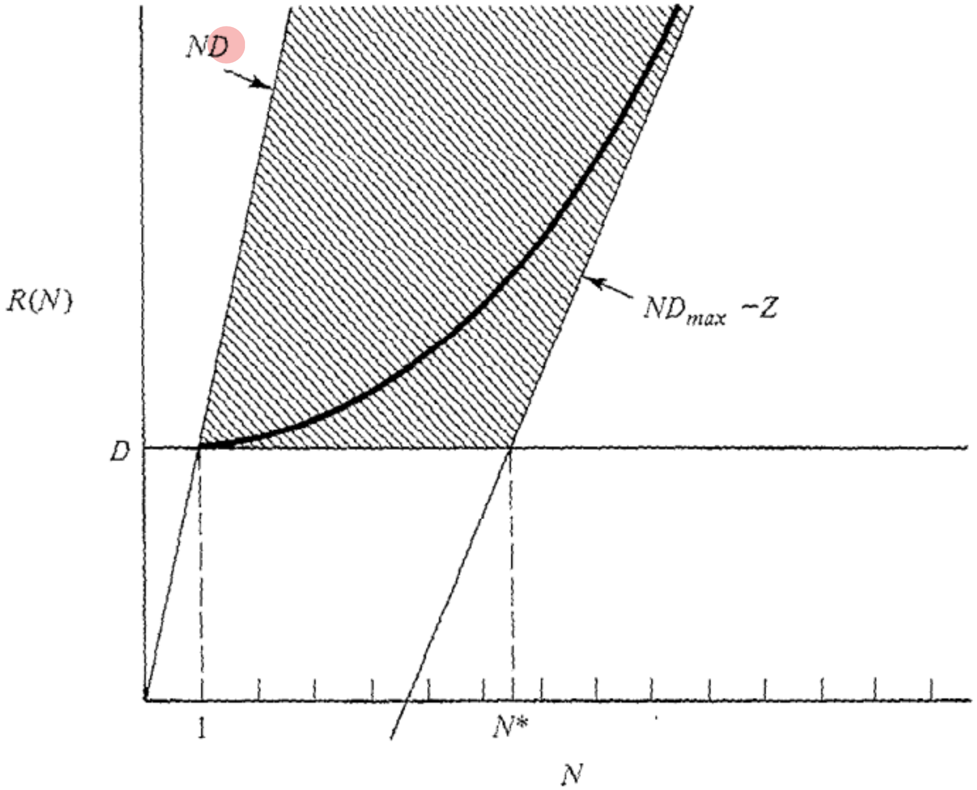
Bound for R(N)

$$\max\left(D, ND_{max} - Z\right) \leq R(N) \leq ND$$

Closed models:

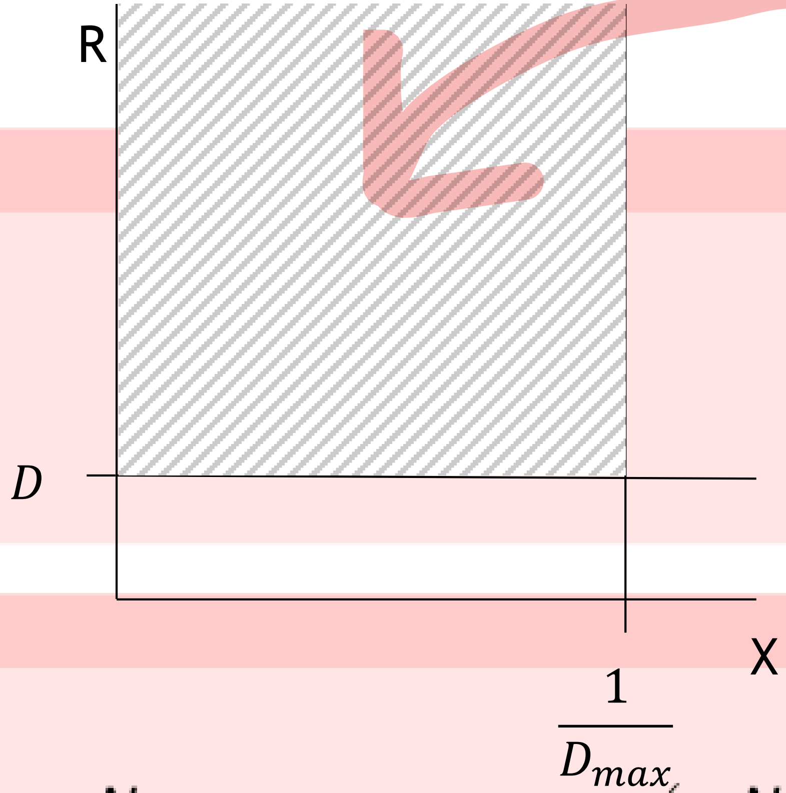
R(N) bounds:

$$\max\,(D\,,\,ND_{max} - Z) \leqslant R(N) \leqslant ND$$



Asymptotic bounds summary

Open Models

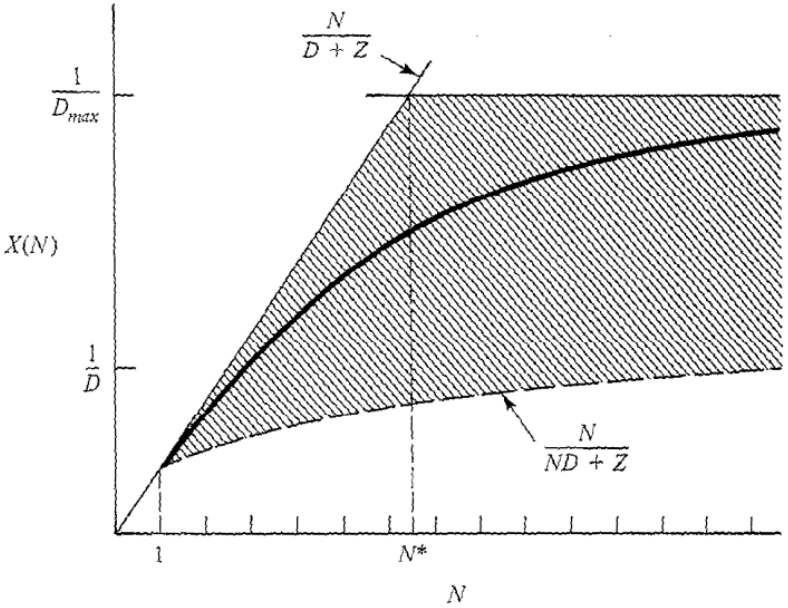


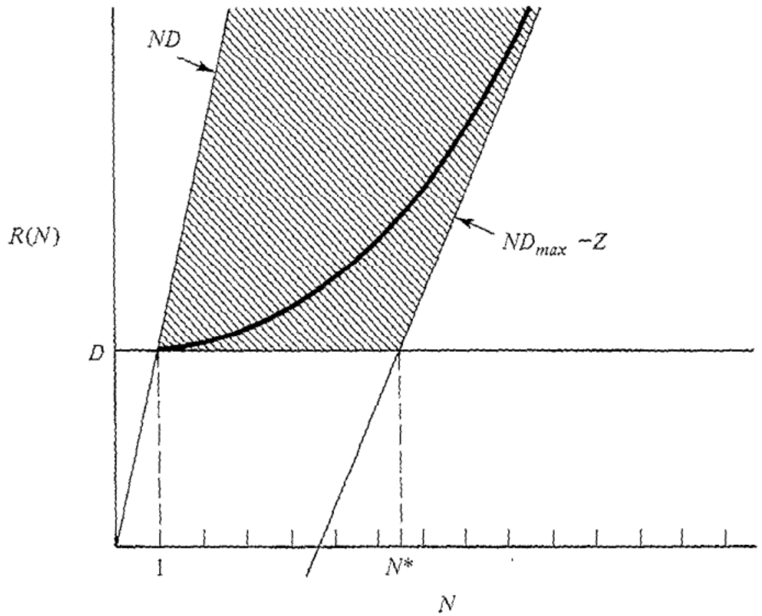
Closed Models

$$\frac{N}{ND+Z} \leq \quad X(N) \quad \leq \min\left(\frac{N}{D+Z}, \frac{1}{D_{max}}\right)$$

$$\max\left(D, ND_{max} - Z\right) \leq \quad R(N) \quad \leq ND$$

Asymptotic bounds summary



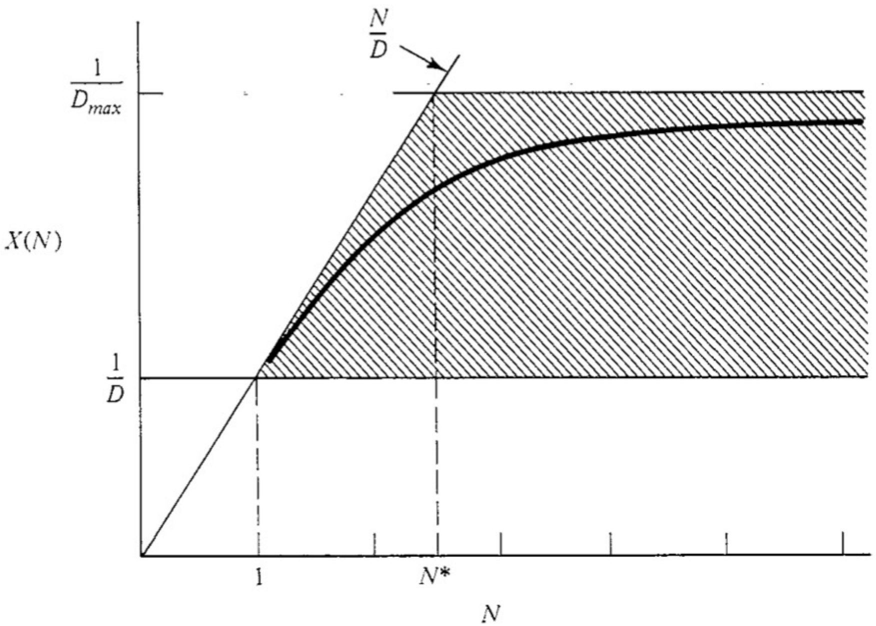


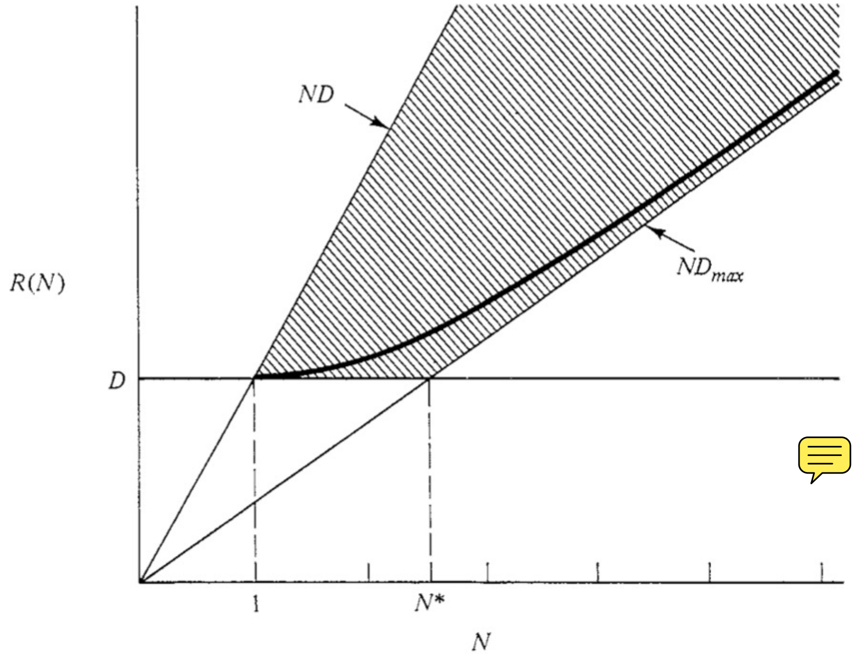
Closed Models

$$\frac{N}{ND+Z} \leq \quad X(N) \quad \leq \min\left(\frac{N}{D+Z}, \frac{1}{D_{max}}\right)$$

$$\max\left(D, ND_{max} - Z\right) \leq \quad R(N) \quad \leq ND$$

Asymptotic bounds summary



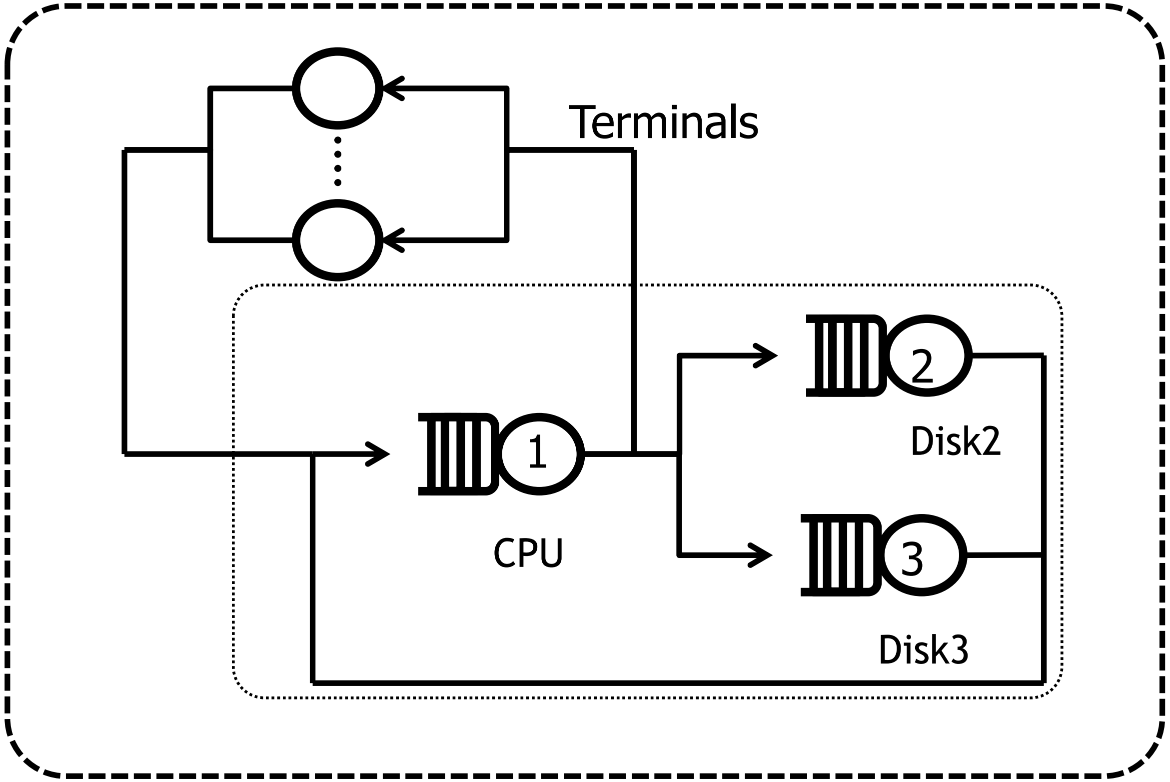


Closed Models

$$\frac{N}{ND+Z} \leq \quad X(N) \quad \leq \min\left(\frac{N}{D+Z}, \frac{1}{D_{max}}\right)$$

$$\max\left(D, ND_{max} - Z\right) \leq \quad R(N) \quad \leq ND$$

Example



Parameters:

$D_1 = 2.0s,\ D_2 = 0.5s,\ D_3 = 3.0s$

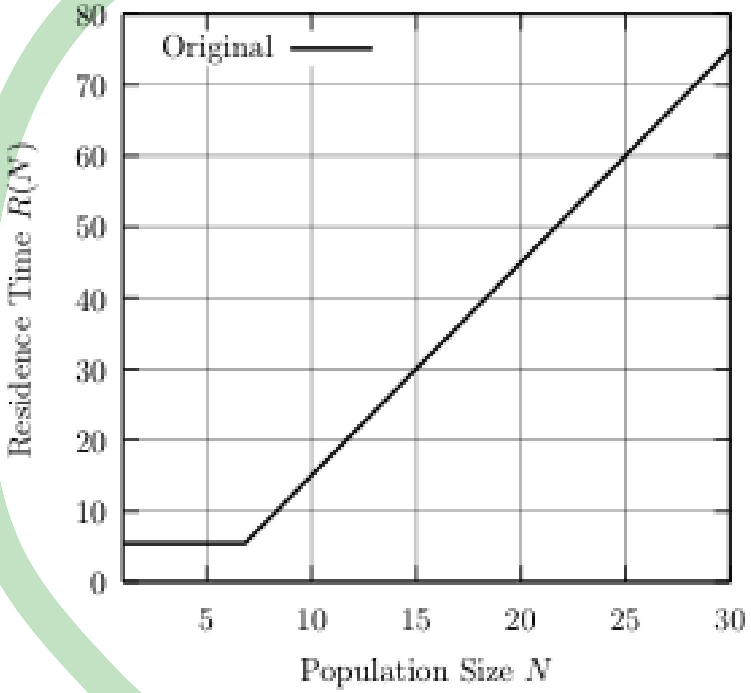
$V_2 = 10,\ V_3 = 100$

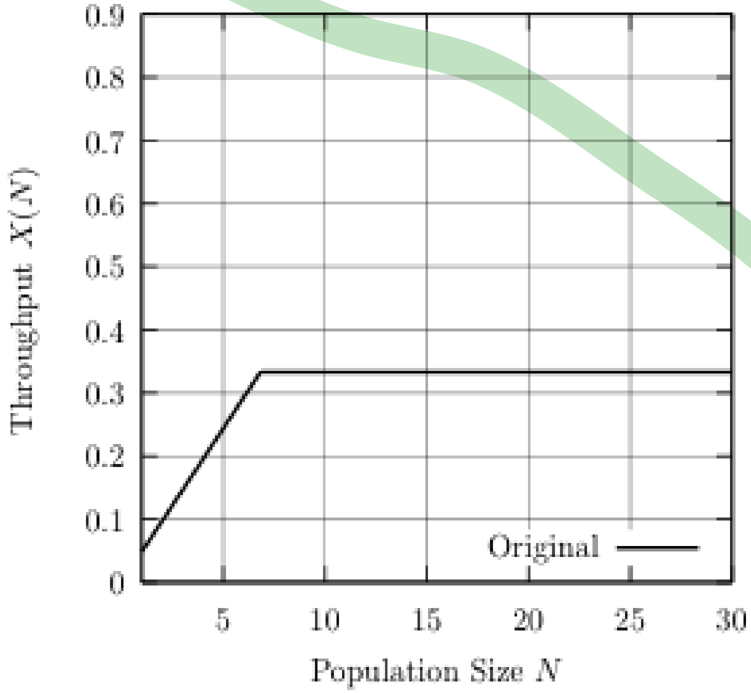
$S_2 = 0.05s,\ S_3 = 0.03s$

$Z = 15s$

Example (original system)

$D_1 = 2.0, D_2 = 0.5, D_3 = 3.0$





Max (5.5, 3*N-15) <=R(N)

X(N) <=min(N/(5.5+15), 1/3)

What-if analysis

Let us consider 4 possible scenarios:

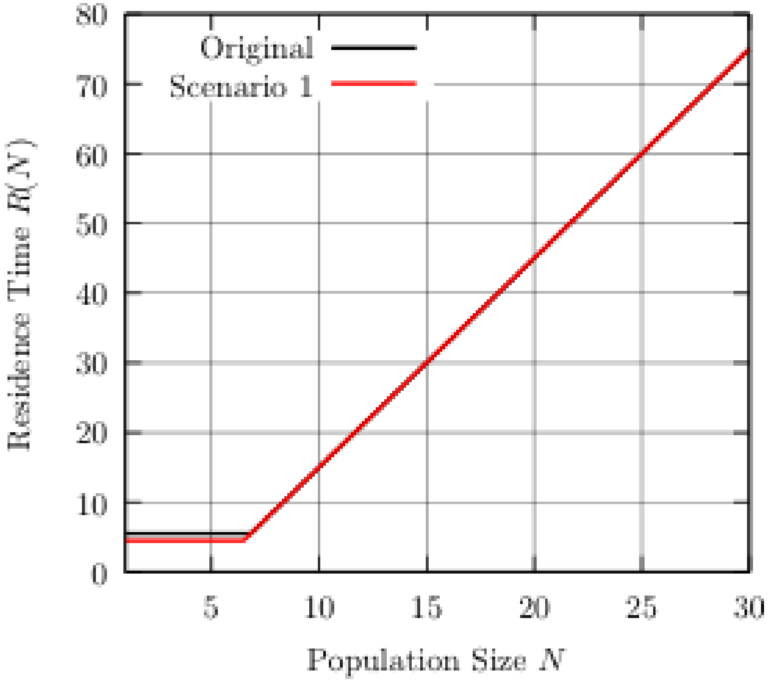
1. Replace the CPU with one that is twice as fast
2. Shift some files from the faster disk (server 3) to the slower disk (server 2), balancing their demands
3. Add a second fast disk (center 4, S_4=0.03) to handle half the load of the busier existing disk (server 3)
4. The three changes made together: the faster CPU and a balanced load across two fast disks and one slow disk

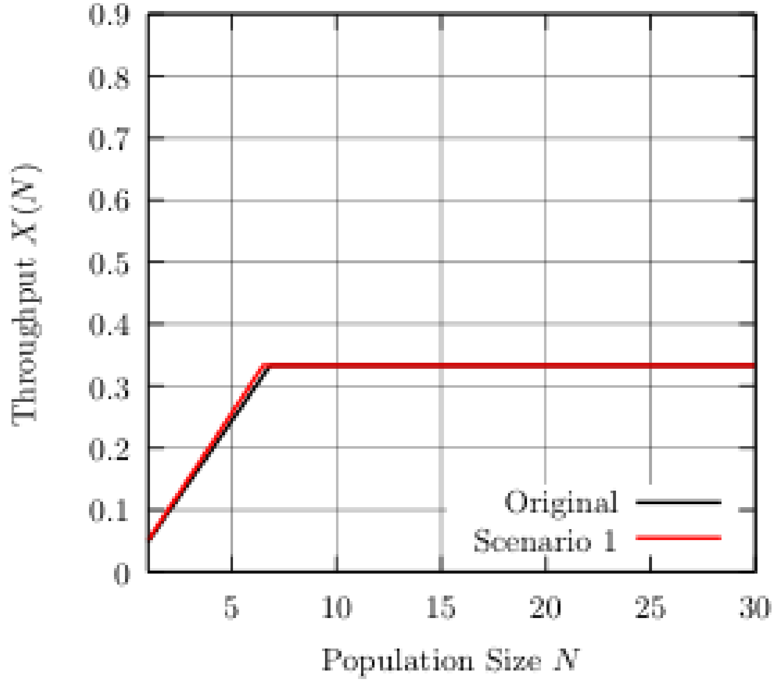
Example: alternative 1

Replace the CPU with one that is twice as fast, so we have:
$D_1=1s$, $D_2=0.5s$, $D_3=3.0s$

Example: alternative 1

Replace the CPU with one that is twice as fast, so we have:
D_1=1*s*, D_2=0.5*s*, D_3=3.0*s*





Max (4.5, 3*N-15) <=R(N)

X(N) <=min(N/(4.5+15), 1/3)

Example: alternative 2

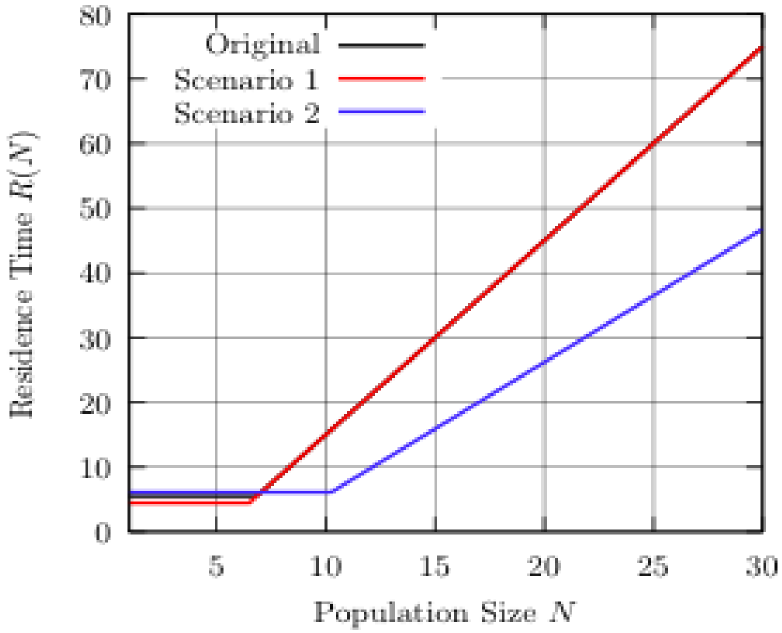
Shift some files from the faster disk (server 3) to the slower disk (server 2), balancing their demands, so having D_2=D_3. Since D_k=V_kS_k, we can solve the following system:

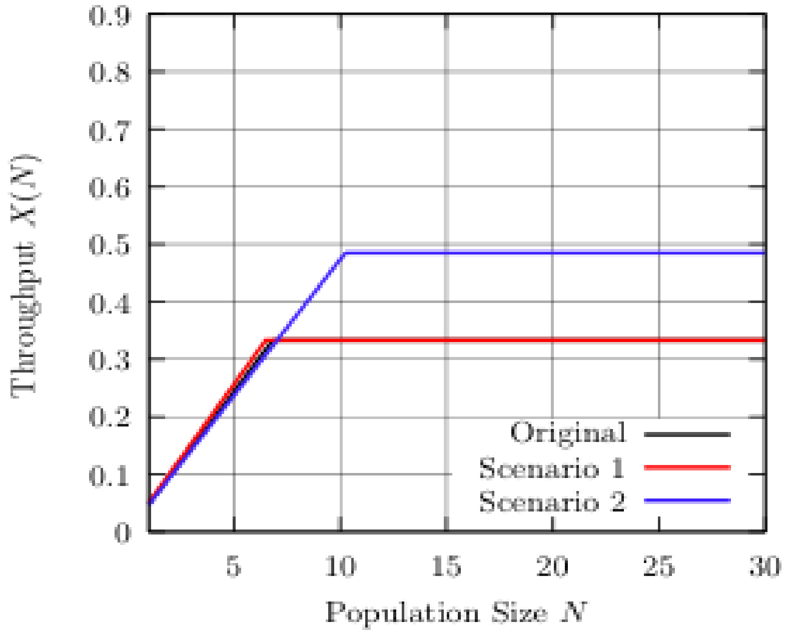
Example: alternative 2

Shift some files from the faster disk (server 3) to the slower disk (server 2), balancing their demands, so having $D_2=D_3$. Since $D_k=V_kS_k$, we can solve the following system:

$$\begin{cases} V_2+V_3 = 110 & \text{the total number of visits remain unchanged} \\ V_2S_2 = V_3S_3 & \text{balancing the service demands} \end{cases}$$

We obtain: $V_2=41$, $V_3=69$ and $D_2=D_3=2.06s$





Max (6.12, 2.06*N-15) <=R(N)   X(N) <=min(N/(6.12+15), 1/2.06)

Example: alternative 2

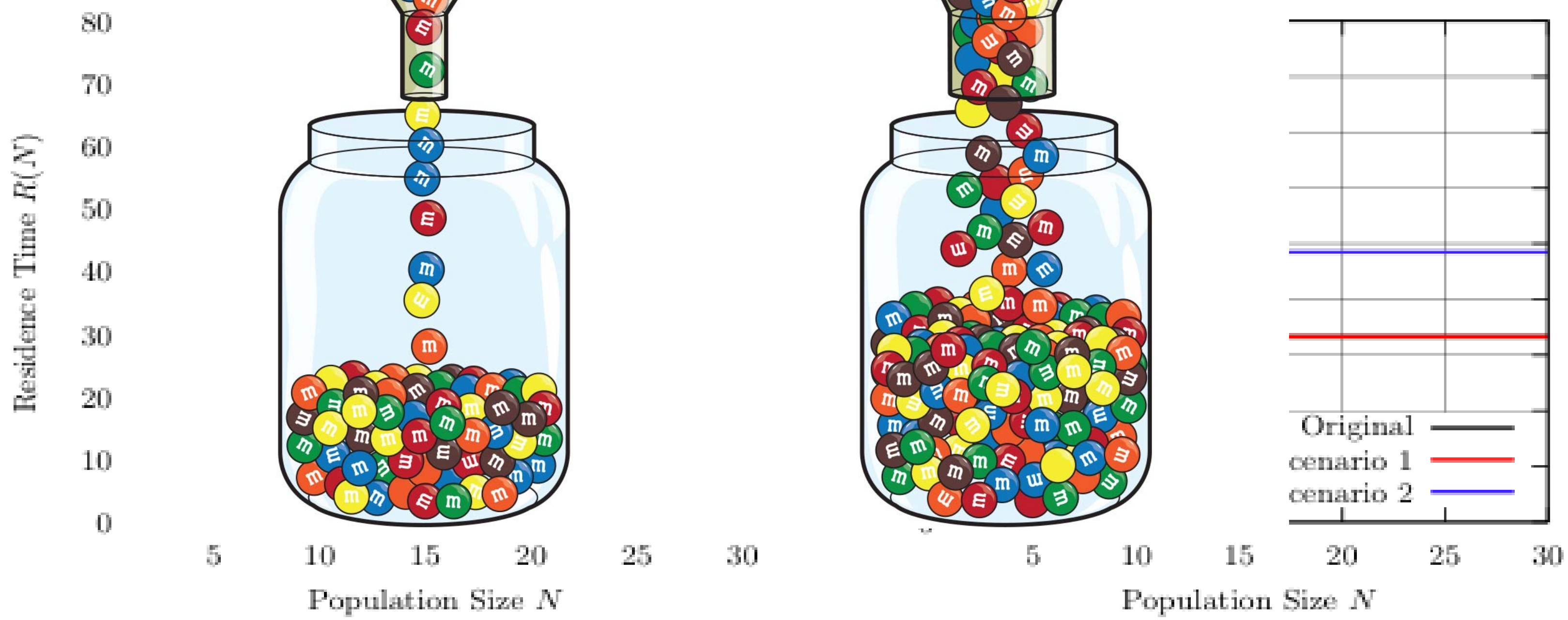
Shift some files from the faster disk (server 3) to the slower disk (server 2), balancing their demands, so having D_2=D_3. Since D_k=$V_k S_k$, we can solve the following sy

V_2-

V_2S

anged

We obtain: \



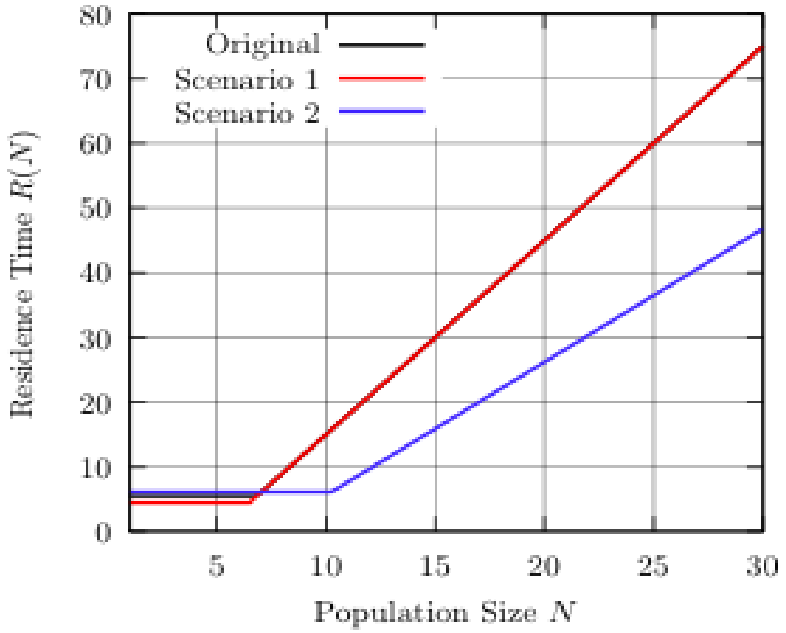
Max (6.12, 2.06*N-15) <=R(N) X(N) <=min(N/(6.12+15), 1/2.06)

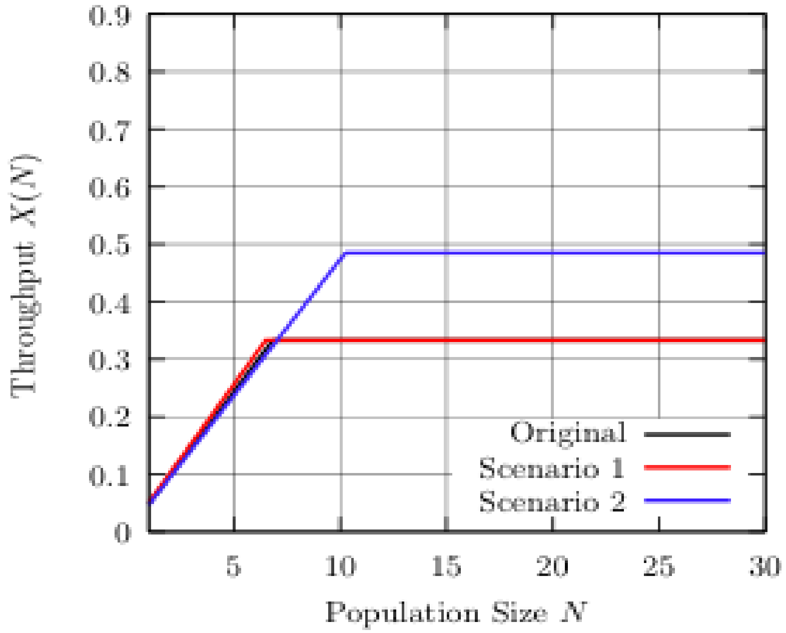
Example: alternative 2

Shift some files from the faster disk (server 3) to the slower disk (server 2), balancing their demands, so having D_2=D_3. Since D_k=$V_k S_k$, we can solve the following system:

$$\begin{cases} V_2+V_3 = 110 & \text{the total number of visits remain unchanged} \\ V_2S_2 = V_3S_3 & \text{balancing the service demands} \end{cases}$$

We obtain: V_2=41, V_3=69 and D_2=D_3=2.06s





Max (6.12, 2.06*N-15) <=R(N)   X(N) <=min(N/(6.12+15), 1/2.06)

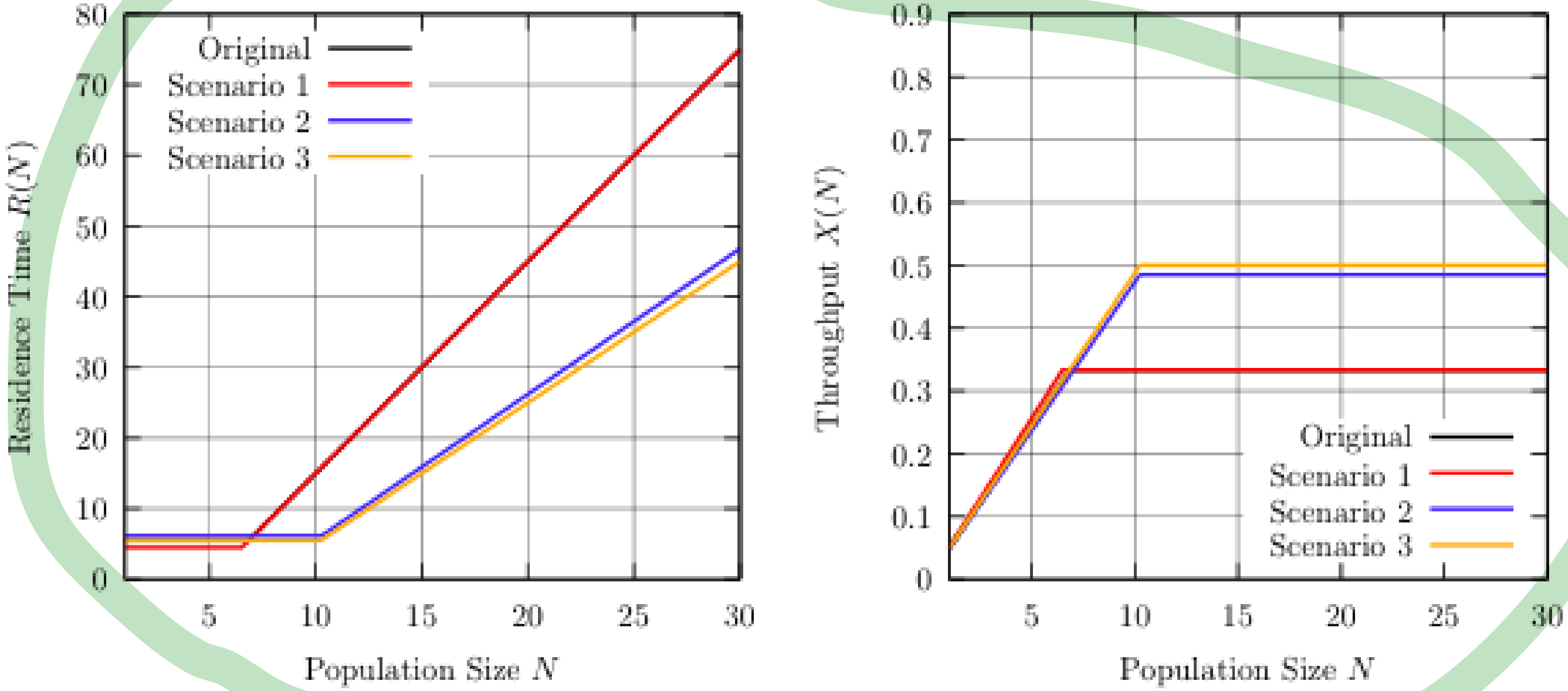
Example: alternative 3

Add a second fast disk (center 4, S_4=0.03) to handle half the load of the busier existing disk (server 3). So we will have K=4 service centers, with

D_1=2, D_2=0.5, D_3 =D_4 =1.5*s*

Example: alternative 3

Add a second fast disk (center 4, S_4=0.03) to handle half the load of the busier existing disk (server 3). So we will have K=4 service centers, with
D_1=2, D_2=0.5, D_3 =D_4 =1.5*s*



Max (5.5, 2*N-15) <=R(N)

X(N) <=min(N/(5.5+15), 1/2)

Example: alternative 4

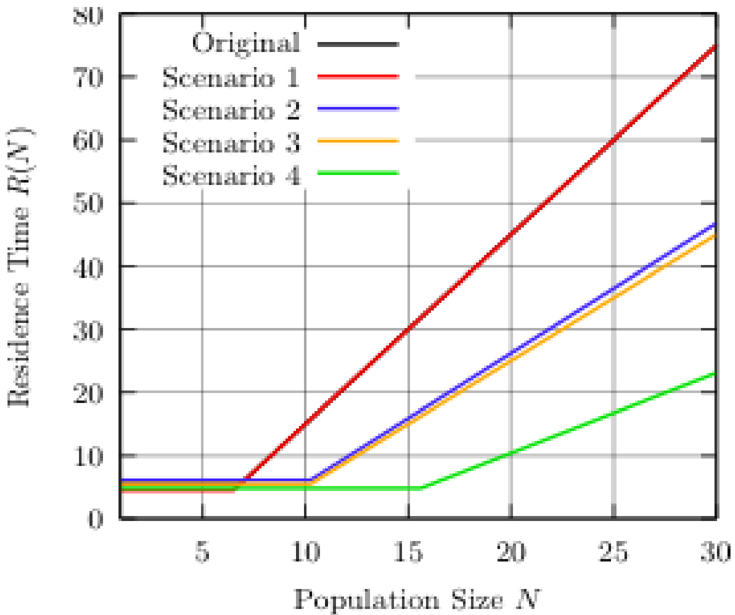
Here we have: a faster CPU (D_1=1) and a balanced load across two fast disks and one slow disk. Similarly to alternative 2, we have:

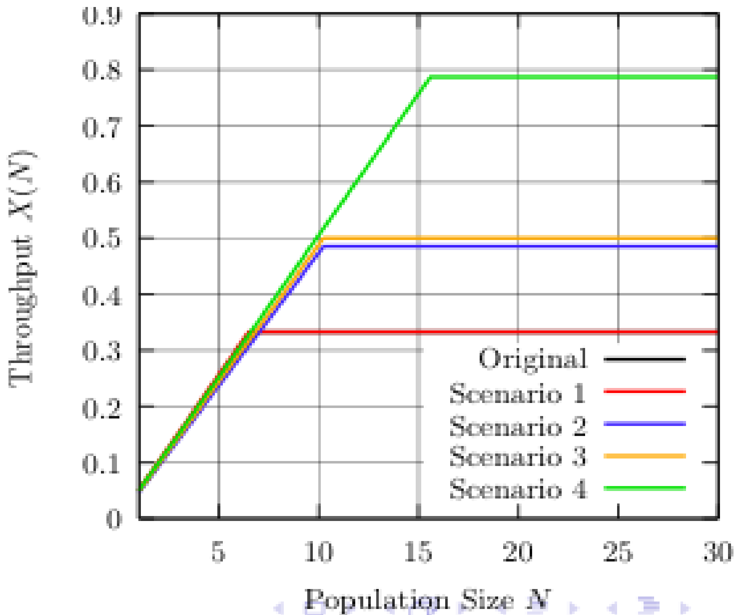
Example: alternative 4

Here we have: a faster CPU (D_1=1) and a balanced load across two fast disks and one slow disk. Similarly to alternative 2, we have:

$$\begin{cases} V_2+V_3+V_4 = 110 & \text{the total number of visits remain unchanged} \\ V_2S_2 = V_3S_3. & \text{balancing the service demands} \\ V_3S_3 = V_4S_4 & \end{cases}$$

Solving the system we obtain: $D_2=D_3=D_4=1.27s$





Max (4.81, 1.27*N-15) <=R(N) X(N) <=min(N/(4.81+15), 1/1.27)