SELECTION OF TECHNIQUES AND METRICS

Gaia Maselli

maselli@di.uniroma1.it

SAPIENZA
UNIVERSITÀ DI ROMA

Overview

- Selection of an evaluation technique
- Selection of performance metrics

Selecting an evaluation technique

- Three techniques for performance evaluation
  1. Analytical modeling
  2. Simulation
  3. Measurement

- How do we choose one of them?

Criterion 1: Stage

- The key consideration in deciding the evaluation technique is the *life-cycle stage* in which the system is
- New system → *analytical modeling* and *simulation* are the only techniques from which to choose
- Improved system → *measurement* (if something similar to the proposed system already exists

Modeling and simulation can be anytime, measurement requires a prototype

Criterion 2: Time required

- Time available for evaluation has to be taken into account
- Often results are required *yesterday*

Short	➔	*analytical modeling*
Medium	➔	*simulation*
Long	➔	*measurement*

Murphy’s law strikes measurements more often than other techniques

🡻

Variable time

Criterion 3: Tools

- Availability of tools plays an important role

Modeling skills

Simulation languages

Measurement instruments

Criterion 4: Accuracy

- Level of accuracy desired is another important consideration

- Low ➔ *analytical modeling* requires so many simplifications and assumptions that if the results turn out to be accurate, even the analysts are surprised
- Moderate ➔ *simulations can* incorporate more details and and require less assumptions and thus are more closer to reality
- Variable ➔ *measurement* may not give accurate results simply because many of the environmental parameters, such as system configuration, type of workload, and time of measurement may be unique to the experiment

Criterion 5: Trade-off evaluation

- The goal of every performance study is to compare different alternatives or to find the optimal parameter value

Easy	→	*analytical modeling*
Moderate	→	*simulation*
Difficult	→	*measurement*

Criterion 6: Cost

- Cost allocated for the project is also important
- Small → *analytical modeling* requires only paper and pencil (in addition to the analyst’s time)
- Medium → *simulation* requires a simulator (often free) and some time
- High → *measurement* requires real equipment, instruments and time

Criterion 7: Saleability

- Saleability is the key justification when considering the expense and labor of measurements.

- Low → *analytical modeling* - some people are skeptical of analytical results simply because they do not understand the technique or the final results
- Medium → *simulation*
- High → *measurement* - It is easy to convince others if it is a real measurement

Often it is helpful to use to use two or more techniques

simultaneously : to validate results

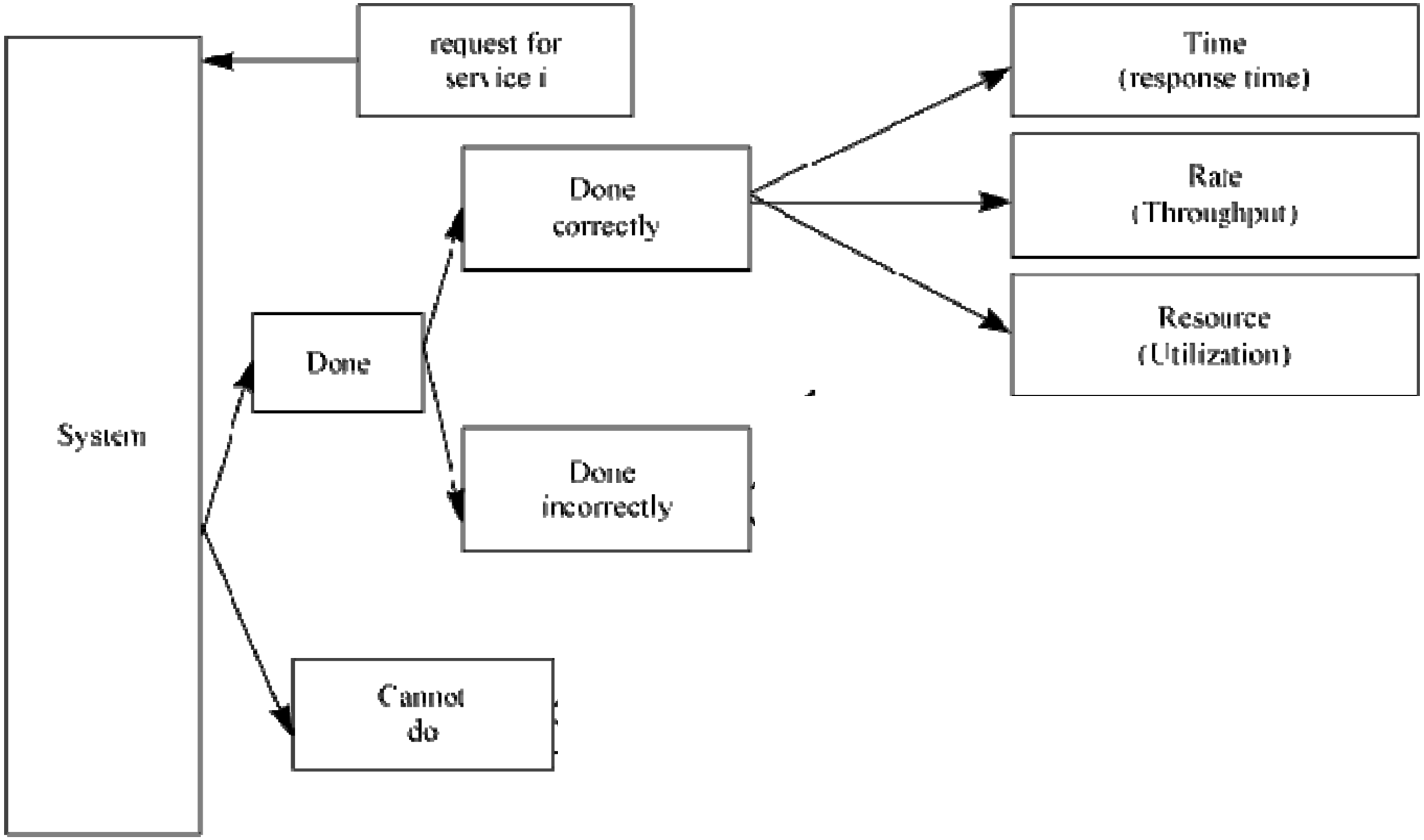
sequentially: to find the appropriate range for system parameters

Three rules of validation

- ❑ Do not trust the results of a **simulation model** until they have been validated by analytical modeling or measurements.
- ❑ Do not trust the results of an **analytical model** until they have been validated by a simulation model or measurements.
- ❑ Do not trust the results of a **measurement** until they have been validated by simulation or analytical modeling.

Selecting performance metrics

Selecting performance metrics



responsiveness

productivity

utilization

Selecting performance metrics

- For each service request the system may **perform the service correctly, incorrectly, or refuse** to perform the service

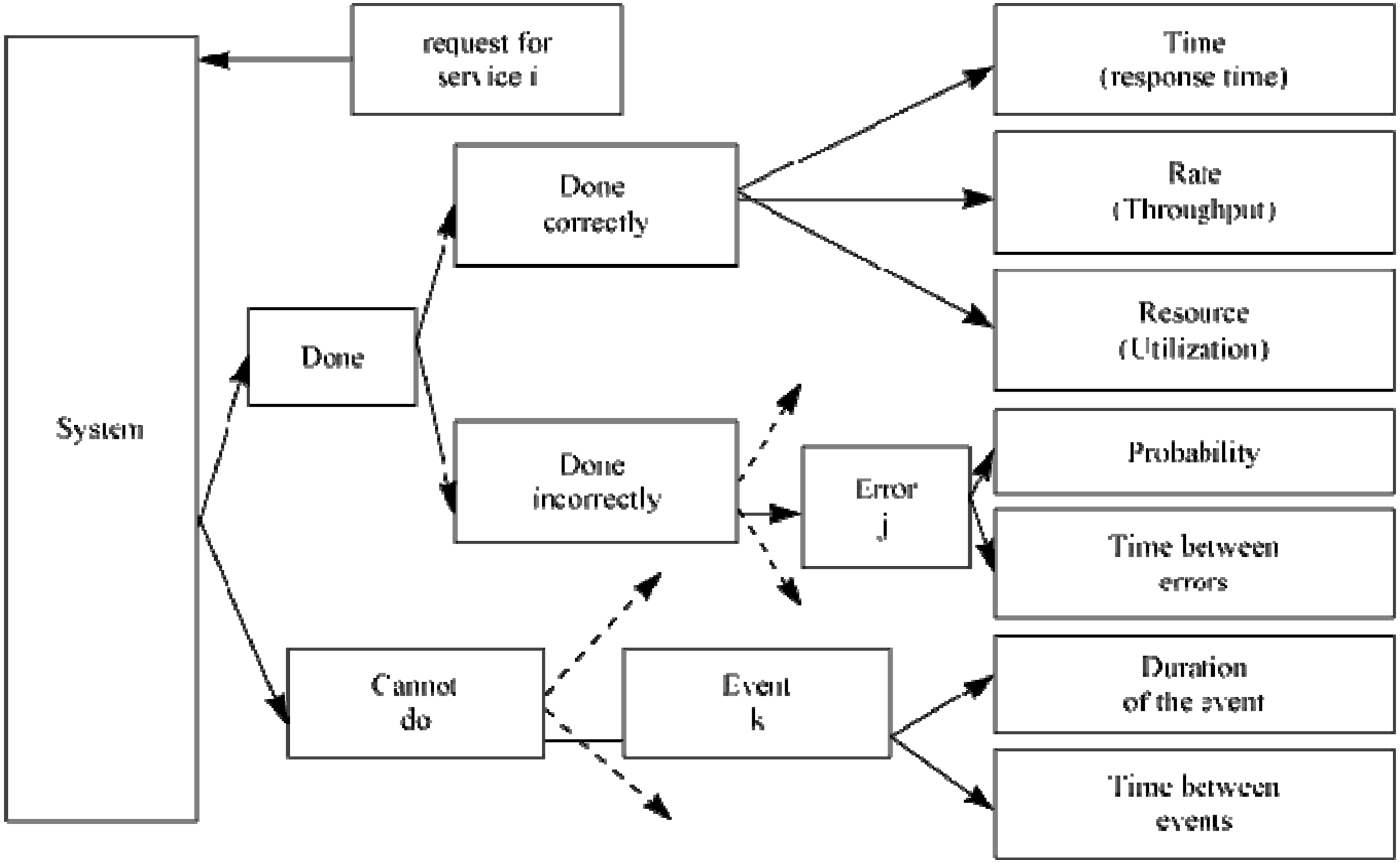
Example: a gateway in a network offers the service of forwarding packets to the specified destination. When presented a packet, it may forward the packet correctly, it may forward it to the worng destination, or it may be down (not forward it at all)

- If the systems performs the service correctly, its performance is measured by the **time taken** to perform the service**, the rate at which** the service is performed, and **the resource consumed** while performing the service
  - **Time ➔ responsiveness**
  - **Rate ➔ productivity**
  - **Resource ➔ utilization**

Example (gateway):

- Responsiveness is the time interval between arrival of a packet and its successful delivery
- Productivity is the number of packets forwarded per unit of time
- Utilization is percentage of time gateway resources are busy for the given load level

Selecting performance metrics



responsiveness

productivity

utilization

global

Selecting performance metrics

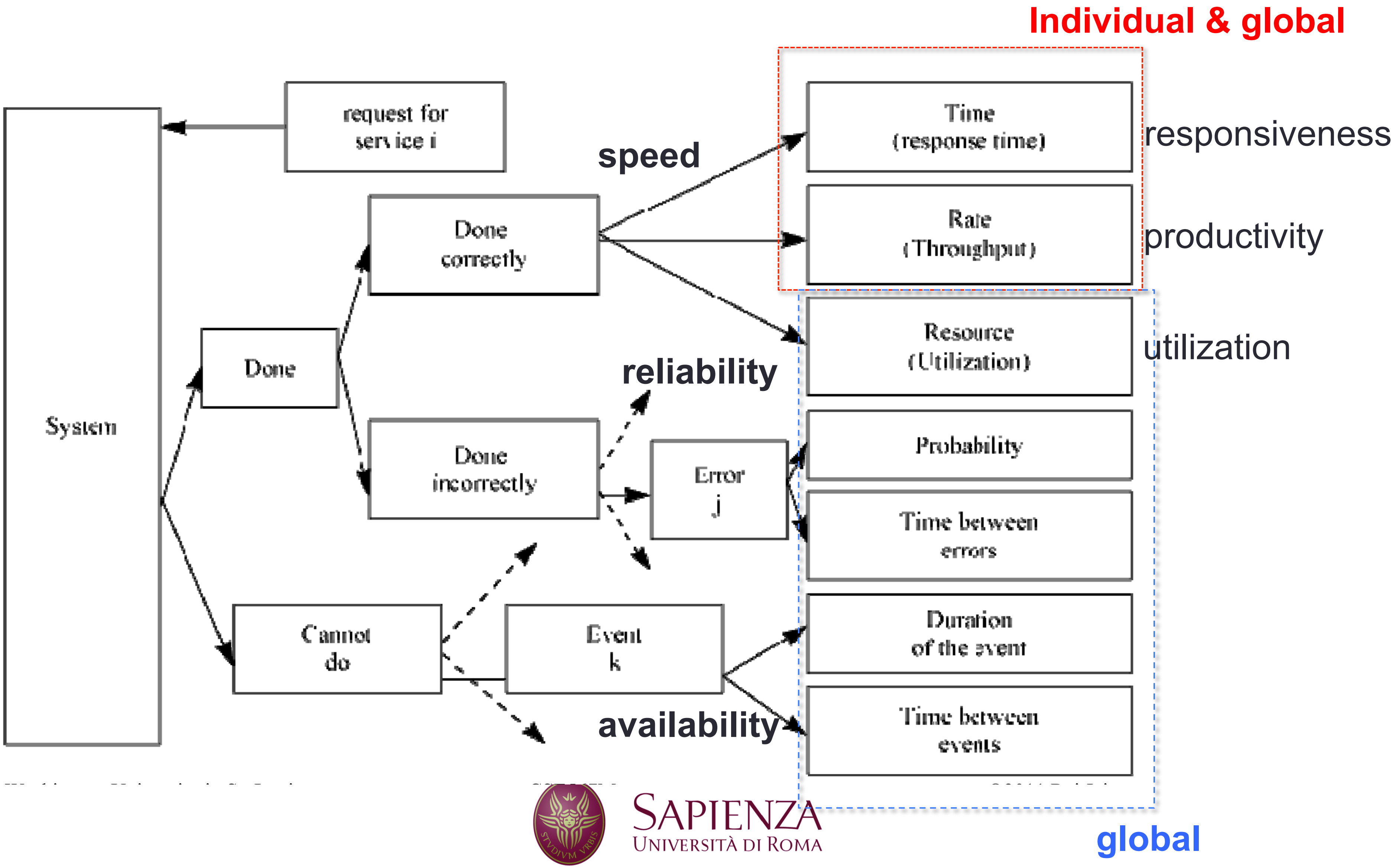
- If the system performs the service incorrectly, an error is occurred
- It is helpful to classify errors and to determine the probabilities of each class of errors.

Example (gateway): we may want to find the probability of single-bit errors and packet error

- If the system does not perform the service, it is said to be down, failed or unavailable
- It is helpful to classify the failure modes and to determine the probability of each class

Example (gateway): the gateway may be unavailable 0.01% of the time due to processor failure and 0.03% due to software failure

Selecting performance metrics



Selecting performance metrics

- Names of the metrics associated with the three outcomes
  - **successful service ➔ speed**
  - **Error ➔ reliability**
  - **Unavailability ➔ availability**
- For each service offered by the system, one would have a number of speed metrics, a number of reliability metrics, and a number of availability metrics
- Most systems offer more than one service, and thus the number of metrics grows proportionately
- As a network is shared by multiple users, two types of performance metrics need to be considered**: individual** and **global**
  - **Individual** metrics reflect the **utility of each user**
  - **Global** metrics reflect the **system wide utility**
  - Some metrics are **individual and global**

N.B. there are cases when the decision that optimize individual metrics is different from the one that optimizes the system metric (e.g., throughput !!!)

Selecting performance metrics

- Given a number of metrics, use the following considerations to select a subset:
  - **Low variability**
    - helps reducing the number of repetitions
    - Metrics that ratio of two variables generally have larger variability than either of the two variable and should be avoided
  - **Non redundancy**
    - If two metrics give essentially the same information, it is less confusing to study only one
  - **Completeness**
    - All possible outcomes should be reflected in the set of performance metrics

Case study: two congestion control algorithms

- Consider the case of comparing two congestion control algorithms for computer networks
- A **network** is composed by a number of end systems interconnected via intermediate systems. The end systems send packets to other end systems on the network. Intermediate systems forward the packets along the right path. Congestion occurs when the number of packets waiting at an intermediate node exceeds the node's buffering capacity
- **Service**: send packets from specified source to specified destination in order (packet forwarding)
- Possible **outcomes**:
  - some packets are delivered in order to the correct destination
  - Some packet are delivered out-of-order to the destination
  - Some packets are delivered more than once (duplicates)
  - Some packets are dropped on the way (lost packets)

Case study (cont)

- Performance: for packets delivered in order
  - Time-rate-resource →
    1. Response time to deliver packets
    2. Throughput: the number of packets per unit of time
    3. Processor time per packets on the source end system
    4. Processor time per packets on the destination end system
    5. Processor time per packets on intermediate systems
  - Variability of response time → retransmissions
    - Response time: the delay inside the network
    6. Variance of response time

Case study (cont)

- Out-of-order packets consume buffers

7. Probability of out-of-order arrivals

- Duplicate packets consume the network resources

8. Probability of duplicate packets

- Lost packets require retransmission

9. Probability of lost packets

- Too much loss cause excessive retransmissions disconnection

10. Probability of disconnect

- The network is a multiuser system: all users have to be treated fairly

10. Fairness: for any given set of user throughputs

$$f(x_1, x_2, \cdots, x_n) = \frac{(\sum_{i=1}^{n} x_i)^2}{n \sum_{i=1}^{n} x_i^2}$$

Example on fairness

Suppose one is asked to distribute 20 dollars among 100 persons

Case 1: Give 20 cents to each of the 100 persons

$$x_i = 0.2 \quad i = 1, 2, \cdots, 100$$

$$Fairness\ Index = 1.0$$

➔Scheme is totally fair

Case 2: discrimination criteria

$$x_i = \begin{cases} 2 & i = 1, 2, \cdots 10 \\ 0 & i = 11, 12, \cdots 100 \end{cases}$$

$$Fairness\ index = 0.10$$

➔ Scheme is only 10% fair

Selecting performance metrics

- After a few experiment it was clear that throughput and delay were really redundant metrics
  - All schemes that resulted in higher throughput also resulted in higher delay
  - The two metrics were removed and combined in a single metric called power, which is defined as the ratio of throughput to response time
- The variance in response time was also dropped since it was redundant with the probability of duplication and the probability of disconnection
  - A higher variance resulted in a higher probability of duplications and a higher probability of premature disconnection

Commonly used performance networks

Let us identify *some* common metrics in computer networks

Commonly used performance metrics

Response time: interval between a user’s request and the system response

In computer networks:

Packet delay: the delay experienced by a packet as it passes through the network, given by the sum of

1. *Routing delay (processing + queuing)*: time a packet spends inside a router (time between the arrival of the trailing bit at the router and the moment the first bit of the packet is placed on the output link)
2. *Transmission delay*: time required to place a packet onto a link (packetSize/rate)
3. *Propagation delay*: time required for a packet to pass from one end of a link to the other end (distance/propSpeed)

Commonly used performance metrics

Throughput: rate at which the requests are serviced by the system

In computer networks:

Throughput: the rate at which traffic flow through the network.

- In a given interval T, the throughput is calculated as the number of packets the pass at a point *without loss* over time T, and is measured in packets per seconds (**pps**) or bits per second (**bps**)
- Throughput over a sequence of hops is determined by the element with minimum available capacity
- The maximum achievable throughput under ideal workload conditions is called **nominal capacity** of the system and corresponds to the **bandwidth**
- The ratio of maximum achievable throughput (usable capacity) to nominal capacity is called **efficiency**. Example: if the maximum throughput from a 100Mbps LAN is 85 Mbps, its efficiency is 85%

Commonly used performance metrics

Reliability: probability of errors or the mean time between errors

In computer networks:

Packet Loss: fraction of packets lost in a time period

Relative Packet loss rate: if C_n is the number of packets entering a network element in time period n, and L_n is the number of packets lost during that time period, the relative packet loss rate can be estimated as L_n/C_n

Commonly used performance metrics

A metric related to throughput and packet loss

Goodput: the rate at which the application endpoint successfully receives data

- The rate at which TCP sends packets is the load it places on the network per unit of time
- Packets being sent may be retransmissions
- Bytes are not necessarily delivered to the application at the rate that the connection sends them

Utility classification of performance metrics

- Three classes depending upon the utility function of a performance metric

1. **Higher is better** (**HB**): systems users and systems managers prefer higher values of such metrics (e.g., throughput is an HB metric)
2. **Lower is better (LB**): systems users and systems managers prefer lower values of such metrics (e.g., packet delay is an HB metric)
3. **Nominal is best** (**NB**): Both high and low values are undesirable. A particular value in the middle is considered best.

Setting performance requirements

- Specifying performance requirements for a system is often a problem
- Typical requirements
  - The system should be both processing and memory efficient
  - There should be an extremely low probability that the network will duplicate a packet

  ➔unacceptable !!! Because
  - *Non specific*: no clear numbers are specified
  - *Non measur*able: there is no way to measure a system and verify that it meets the requirements
  - *Non thorough*: no attempt is made to specify all the possible requirements
- Requirements must be specific, measurable, and thorough
- See as an example case study 3.2, chapter 3, Jain's book