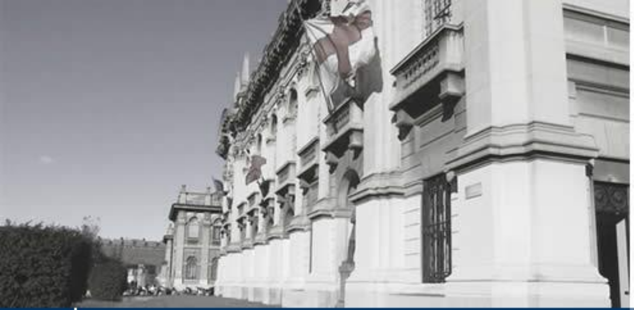


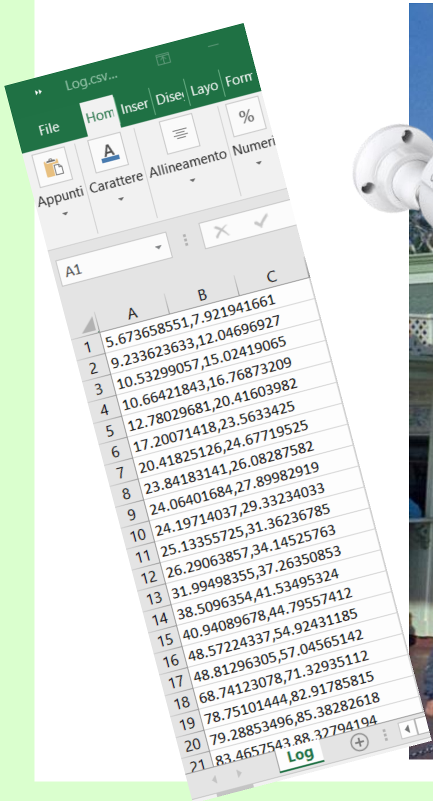
POLITECNICO DI MILANO

Performance Evaluation and Applications

Basic Performance Metrics

Motivating example

A ticket booth is monitored by a smart device that writes in a log file the times at which customers enter and leave the counters.



	A	B	C
1	5.673658551,7.921941661		
2	9.233623633,12.04696927		
3	10.53299057,15.02419065		
4	10.66421843,16.76873209		
5	12.78029681,20.41603982		
6	17.20071418,23.5633425		
7	20.41825126,24.67719525		
8	23.84183141,26.08287582		
9	24.06401684,27.89982919		
10	24.19714037,29.33234033		
11	25.13355725,31.36236785		
12	26.29063857,34.14525763		
13	31.99498355,37.26350853		
14	38.5096354,41.53495324		
15	40.94089678,44.79557412		
16	48.57224337,54.92431185		
17	48.81296305,57.04565142		
18	68.74123078,71.32935112		
19	78.75101444,82.91785815		
20	79.28853496,85.38282618		
21	83.4657543,88.32794194		

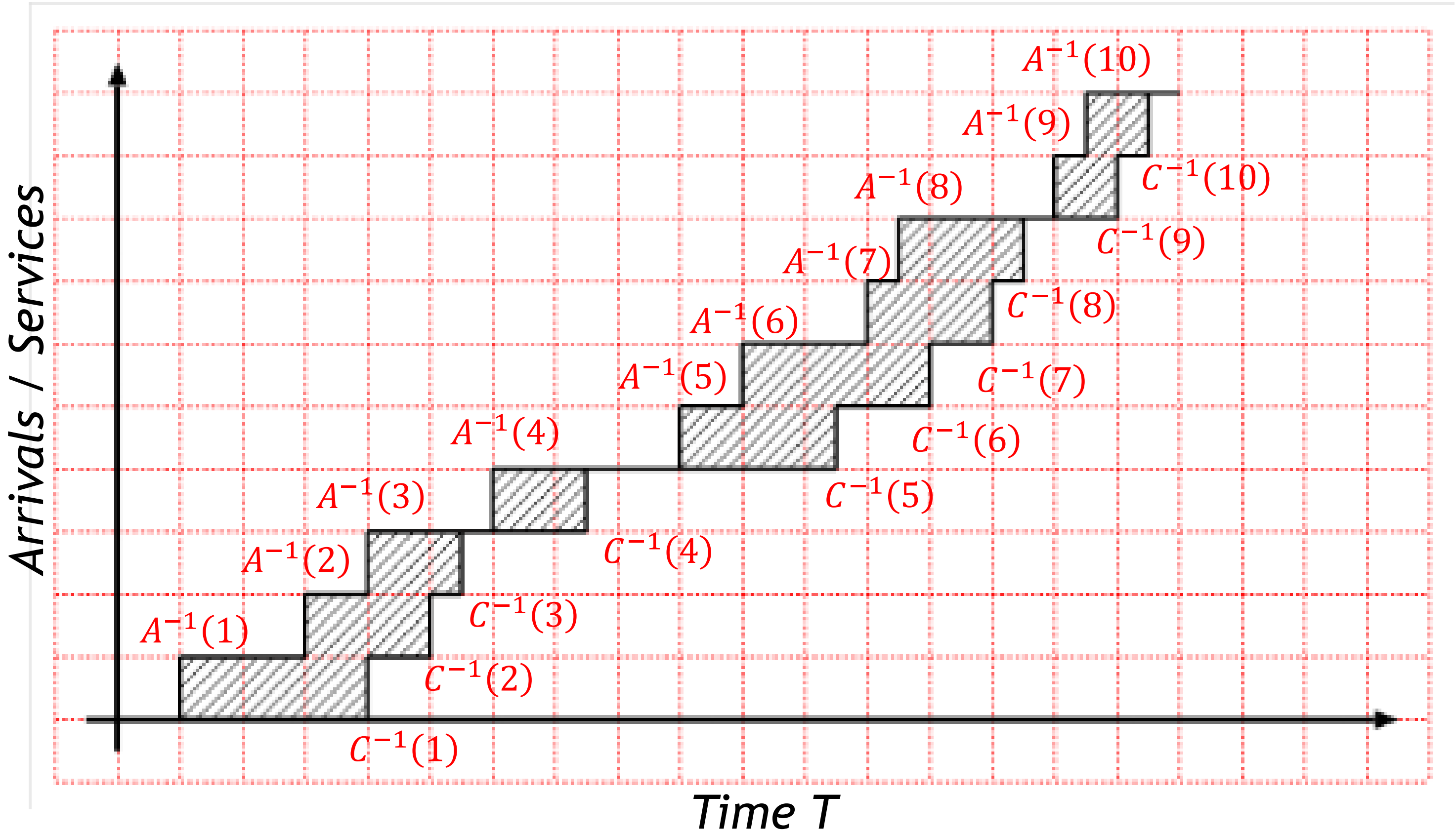
Motivating example

The company would like to use such log file to determine:

- Average queue length
- Utilization of the Booth
- Average service time
- Arrival rate
- Average response time
- Probability that a costumer has to wait more than 15 minutes.

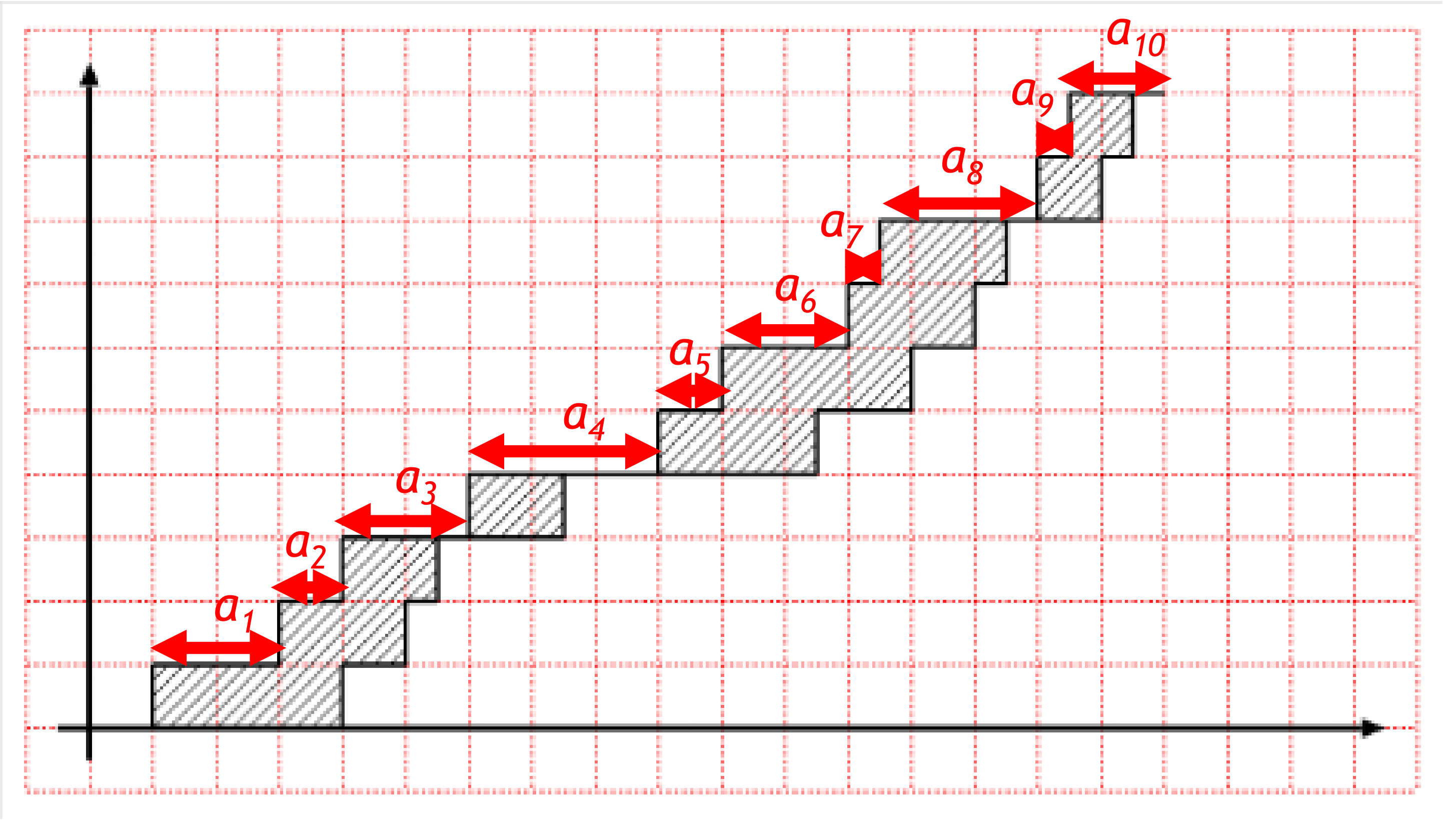
Arrival and completion times of a job

Let us call $A^{-1}(i)$ the time of the i-th arrival, and $C^{-1}(i)$ the time of the i-th service. Since both *A(T)* and *C(T)* are step functions, $A^{-1}(i)$ and $C^{-1}(i)$ can be seen as infimum of their inverse.



Interarrivals times

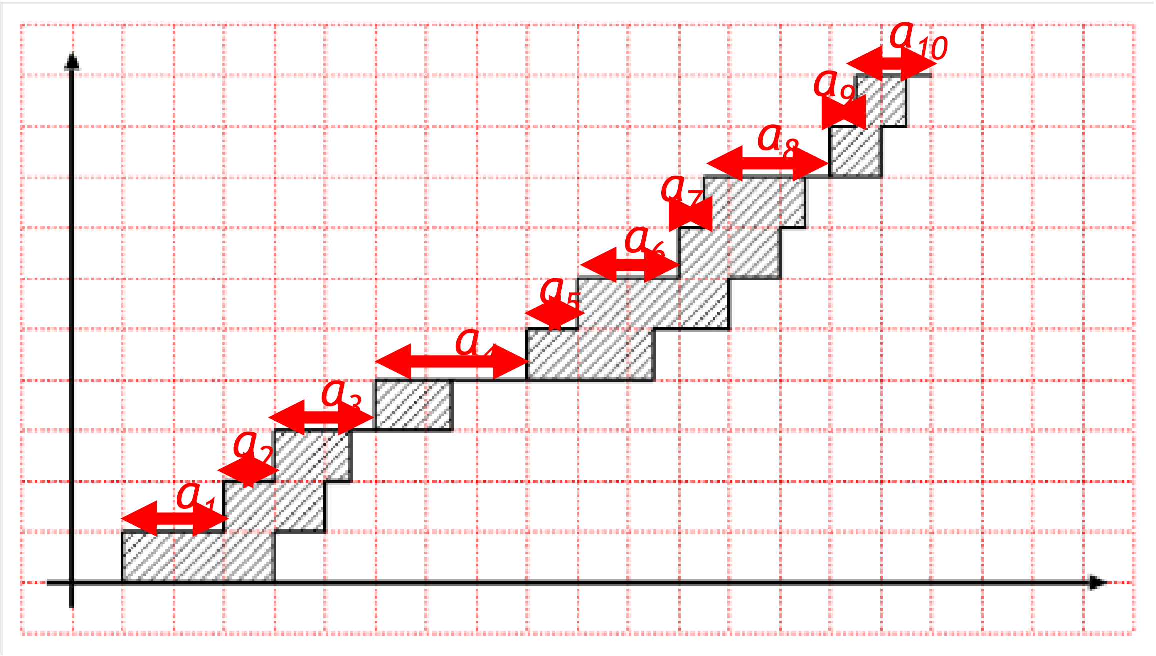
The inter-arrival a_i time measures the time between the arrivals of two consecutive jobs *i* and *i+1*.



Interarrivals times

If we have *A(T)* we can easily derive the inter-arrival times a_i:

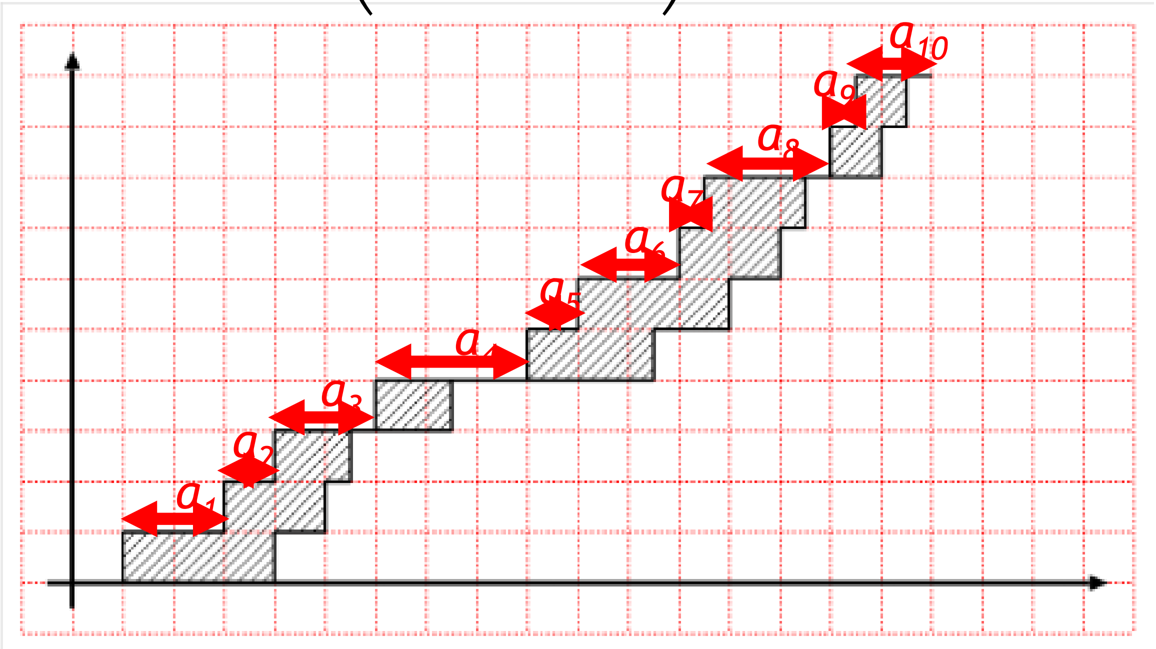
$$a_i = A^{-1}(i+1) - A^{-1}(i)$$



Interarrivals times

Conversely, if we have the inter-arrival times a_i, we can easily derive $A(T)$. Let us call $I(X)$ the indicator function, which returns *1* if proposition X is true or *0* otherwise, and let as assume that a_0 accounts for the arrival time of the first job. We have:

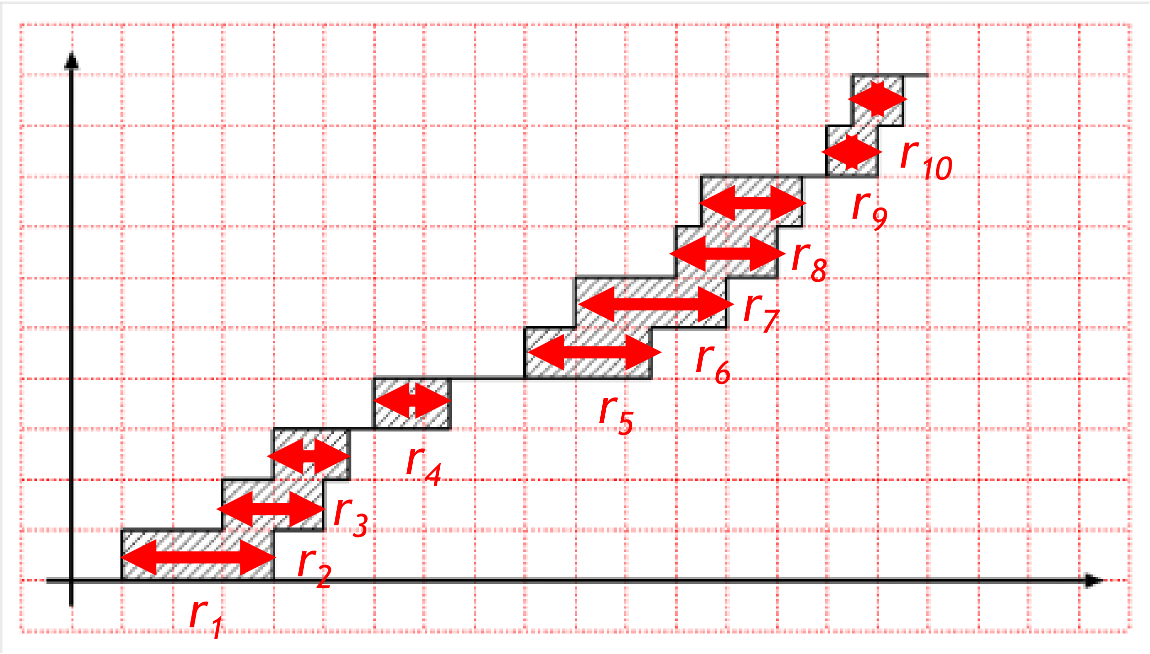
$$A(T) = \sum_{K=1} I\left(\sum_{i=0}^{K-1} a_i \leq T\right) \qquad A^{-1}(i) = \sum_{k=0}^{i-1} a_k$$



Arrival and completion times of the i-th job

Moreover, if we know that jobs are served one at a time, in the order in which they arrived, and without being interrupted, we can estimate r_i from *A(T)* and *C(T)*:

$$r_i = C^{-1}(i) - A^{-1}(i)$$



Estimating C(T)

Under the same assumptions, we can determine C(T) from A(T) and r_i :

$$C(T) = \sum_K I(A^{-1}(K) + r_K \leq T)$$

$$C^{-1}(i) = A^{-1}(i) + r_i$$

In this setting, we can also iteratively determine C(T) from a_i and s_i .
In particular, the *i-th* job will end s_i time units after either:

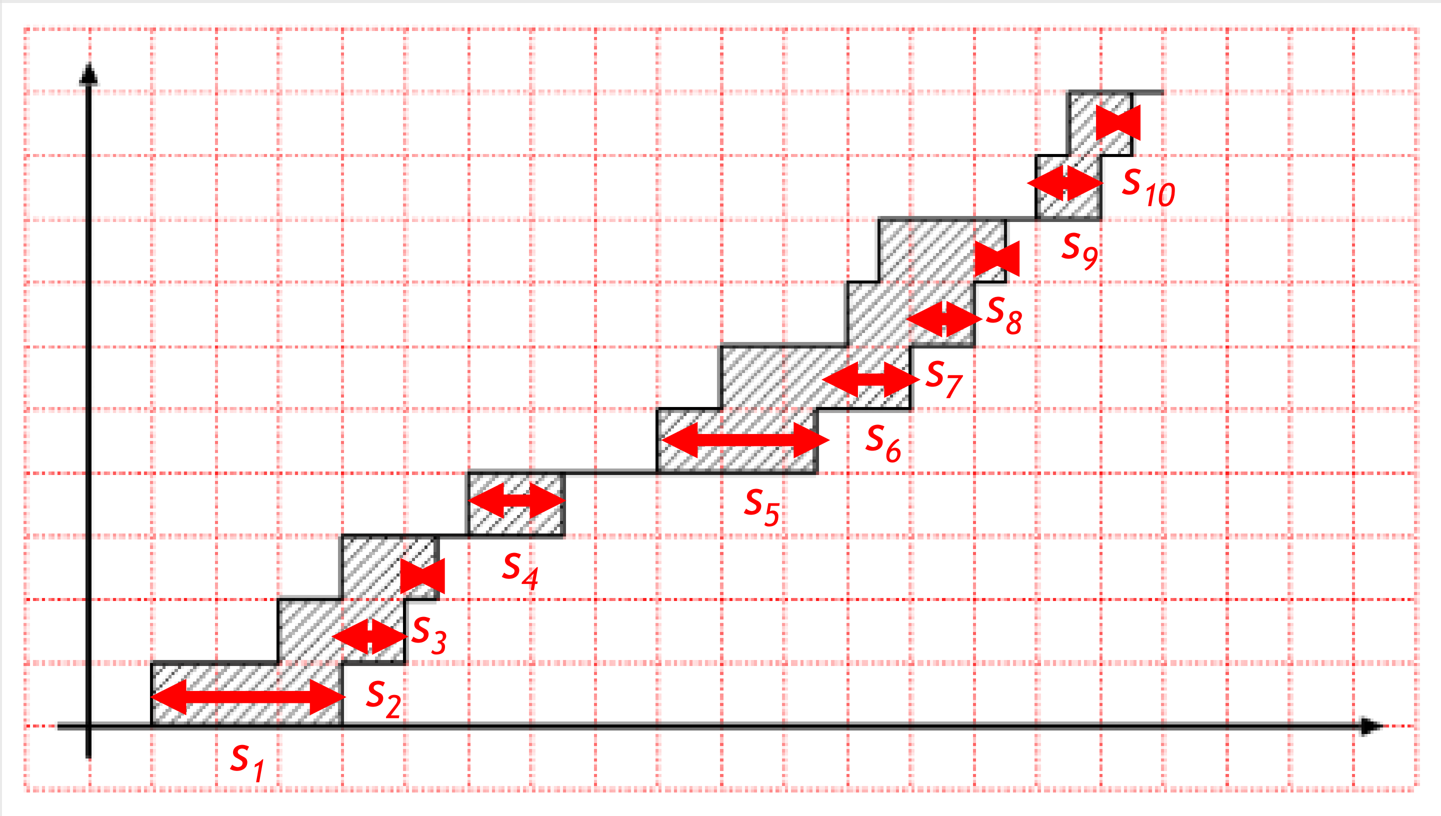
- the completion of the previous job if it had to wait in the queue
- or after its arrival to the station if it was served immediately

$$C^{-1}(i) = max\left(A^{-1}(i), C^{-1}(i-1)\right) + s_i$$

Service times

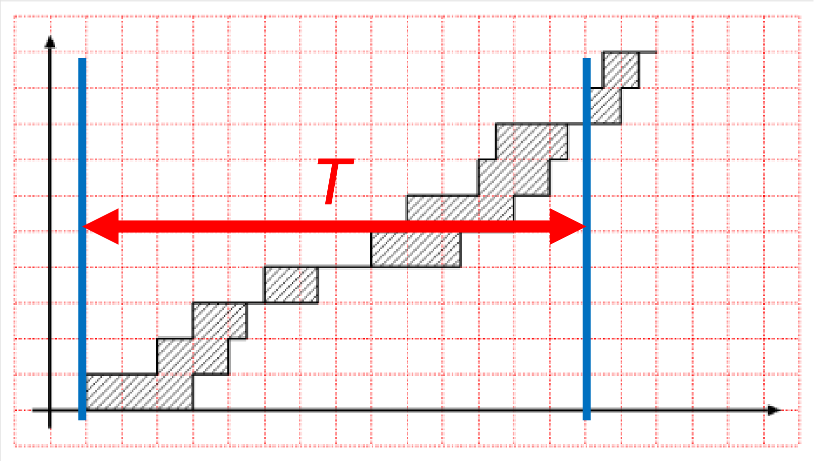
Still under these assumptions, inverting the previous formula we can compute s_i from both the arrival and service curves:

$$s_i = C^{-1}(i) - max\left(A^{-1}(i), C^{-1}(i-1)\right)$$



Basic relations

If we set the time T starting and ending at the moment just before a new arrival at an empty system, we have:



$$A(T) = C(T) \qquad \sum_{i=1}^{A(T)} a_i = T \qquad \sum_{i=1}^{C(T)} s_i = B(T)$$

Basic relations: first wrap-up

All the relations previously seen are useful because, depending on the system, it can be easier measure $A(T)$, $C(T)$, a_i, s_i, or r_i.

With the previous relations, if the assumptions are fulfilled, we can derive the missing parameters, and thus compute all the workload and performance indices values.

$$\sum_{i=1}^{A(T)} a_i = T$$

$$\sum_{i=1}^{C(T)} s_i = B(T)$$

$$C^{-1}(i) = max\left(A^{-1}(i), C^{-1}(i-1)\right) + s_i$$

$$C(T) = \sum_{K} I(A^{-1}(K) + r_K \leq T)$$

$$r_i = C^{-1}(i) - A^{-1}(i)$$

$$C^{-1}(i) = A^{-1}(i) + r_i$$

$$A(T) = \sum_{K=1} I\left(\sum_{i=0}^{K-1} a_i \leq T\right)$$

$$A^{-1}(i) = \sum_{k=0}^{i-1} a_k$$

$$a_i = A^{-1}(i+1) - A^{-1}(i)$$

Basic relations

Let us call $\bar{A}$ the *average inter-arrival time*:

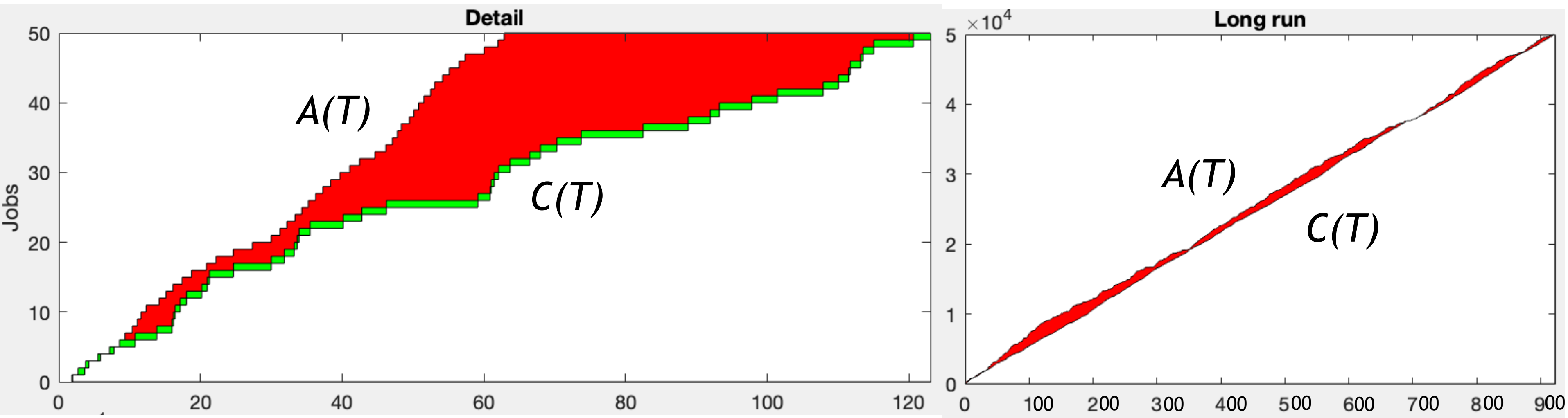
$$\bar{A} = \lim_{T\to\infty} \frac{\sum_{i=1}^{A(T)} a_i}{A(T)}$$

Since $T = \sum_{i=1}^{A(T)} a_i$, the arrival rate λ can also be defined in the following way:

$$\lambda = \lim_{T\to\infty} \frac{A(T)}{T} = \frac{1}{\lim_{T\to\infty} \frac{T}{A(T)}} = \frac{1}{\lim_{T\to\infty} \frac{\sum_{i=1}^{A(T)} a_i}{A(T)}} = \frac{1}{\bar{A}}$$

Basic relations

If the system is *stable* (it is able to serve all its jobs), there will always exist a point of time *T*, in the future, when *A(T)* = *C(T)* .

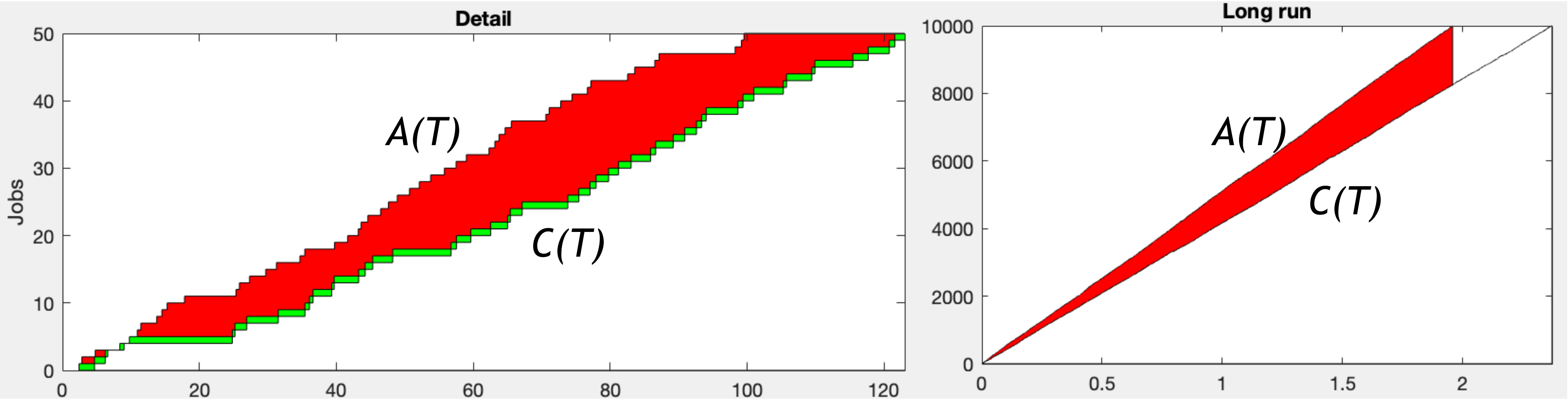


Thus, if the system is *stable* and there are no losses, throughput and arrival rates are always equal.

$$\lambda = X$$

Basic relations

If the system is unstable, *A(T)* and *C(T)* will diverge, and after a given point in time, the system will never return empty again.



In this case:

$$\lambda > X$$

Stability condition

By construction, since B(T) is less or equal to T, then the utilization should be less than one:

$$B(T) \leq T \quad \Rightarrow \quad U = \frac{B(T)}{T} \leq 1$$

Although there exists special cases in which the system is stable with U exactly equal to one ($U = 1$), they are extremely rare.

In most of the cases, B(T) = T means that the system never returns to an empty state, thus it is unstable. For this reason, we usually prefer to check that:

$$B(T) < T \quad \Rightarrow \quad U < 1$$

Stability condition

Stability condition allows to find limiting relations between the arrival rate and the average service:

$$X \cdot S = \lambda \cdot S = \frac{S}{\bar{A}} = \leq 1$$

$$\lambda \leq \frac{1}{S} \qquad S \leq \frac{1}{\lambda} \qquad S \leq \bar{A}$$

$$X \leq \frac{1}{S} \qquad S \leq \frac{1}{X} \qquad \frac{1}{X} = \bar{A}$$

Again, the equality should always be taken with extreme care!

Response time distribution

If we have the response times of the single jobs, r_i, we can approximate its distribution, estimating the probability that the response time is less than a threshold τ.

$$p(R < \tau) = \frac{\sum_{i=1}^{C} I(r_i < \tau)}{C}$$

Note that this relation can be extended to any predicate Ψ(R), and it can be used to compute the probability that the response time respects a given property:

$$p(\Psi(\mathrm{R})) = \frac{\sum_{i=1}^{C} I(\Psi(r_i))}{C}$$

Example:
Ψ(R) = “R between 2 and 3”

$$p(\Psi(\mathrm{R})) = \frac{\sum_{i=1}^{C} I(2 \le r_i \le 3)}{C}$$

Service time and inter-arrival time distributions

The same reasoning is valid also for the service time s_i and the inter-arrival times a_i:

$$p(S < \tau) = \frac{\sum_{i=1}^{C} I(s_i < \tau)}{C}$$

$$p(A < \tau) = \frac{\sum_{i=1}^{C} I(a_i < \tau)}{A}$$

$$p(\Psi(\mathrm{S})) = \frac{\sum_{i=1}^{C} I(\Psi(s_i))}{C}$$

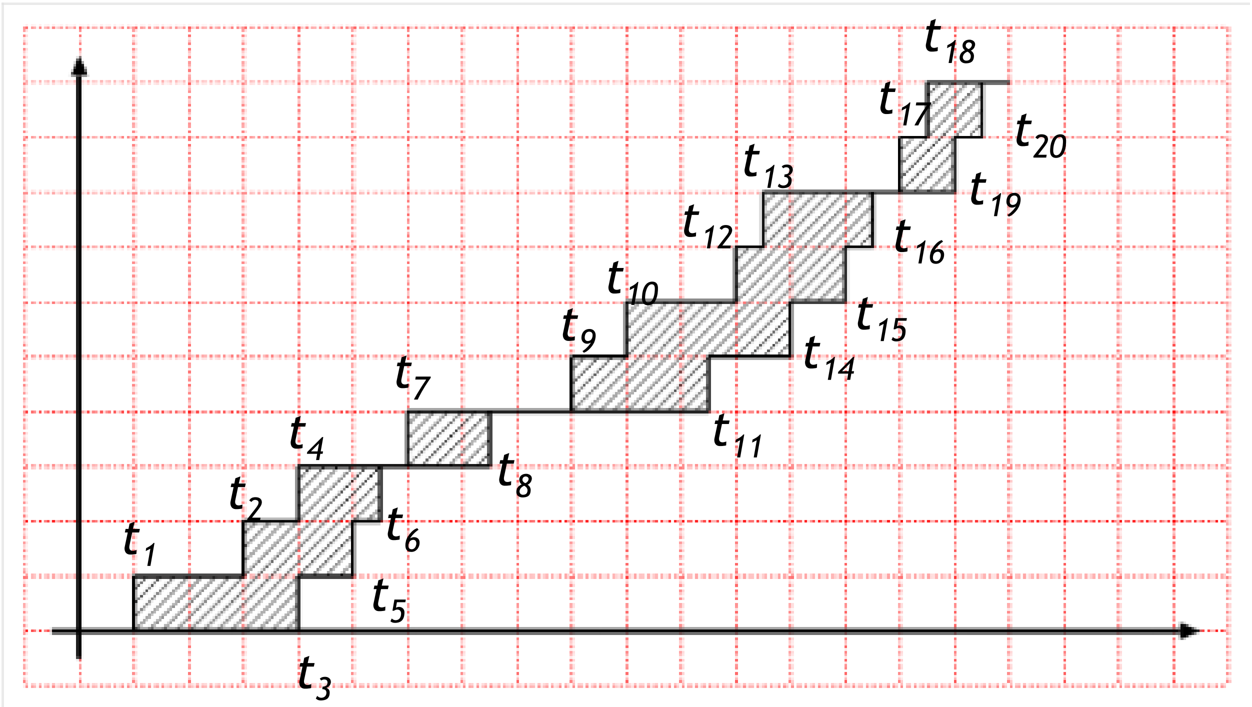
$$p(\Psi(\mathrm{A})) = \frac{\sum_{i=1}^{C} I(\Psi(a_i))}{A}$$

Queue length distribution

With a slightly more complex procedure, we can determine the probability of having *n* jobs in the system from A(t) and C(t).

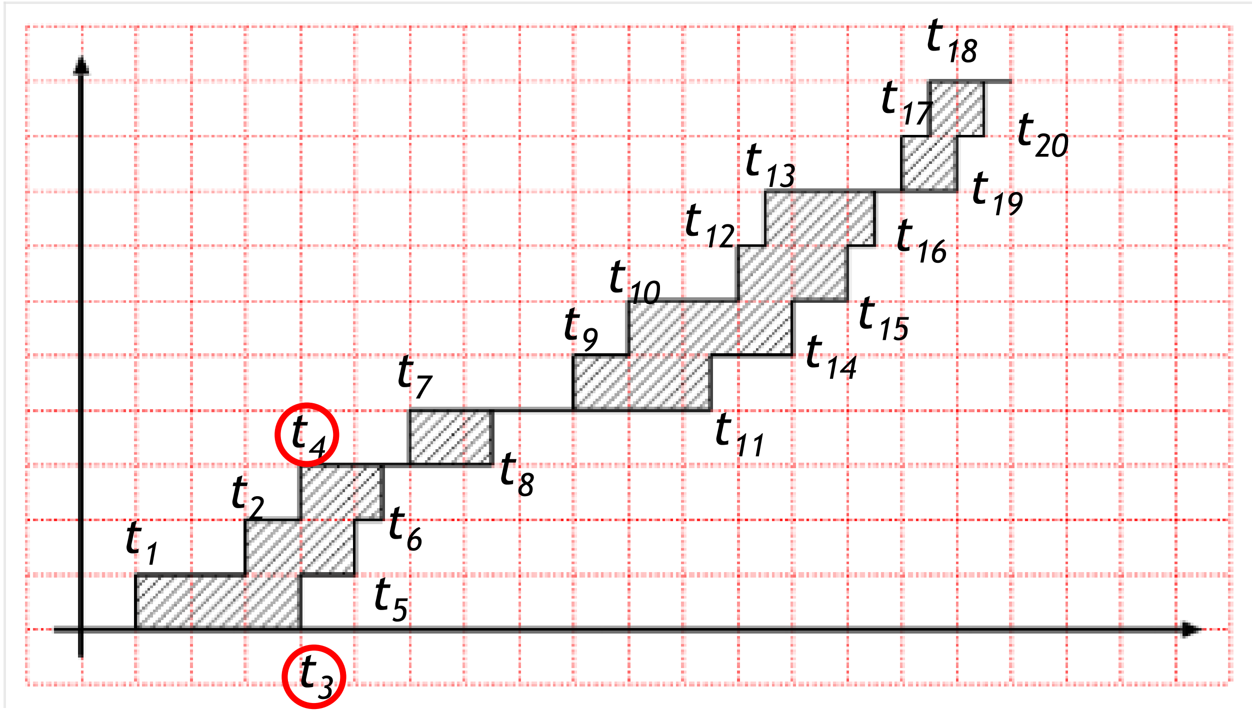
First we observe that between arrivals or services, the population in the system remains constant.

Let's call t_i the time at which either an arrival, or a departure occur.



Queue length distribution

Note that, although very close, we suppose that the departure of the first job t_3 is just slightly before the arrival of the third job t_4.



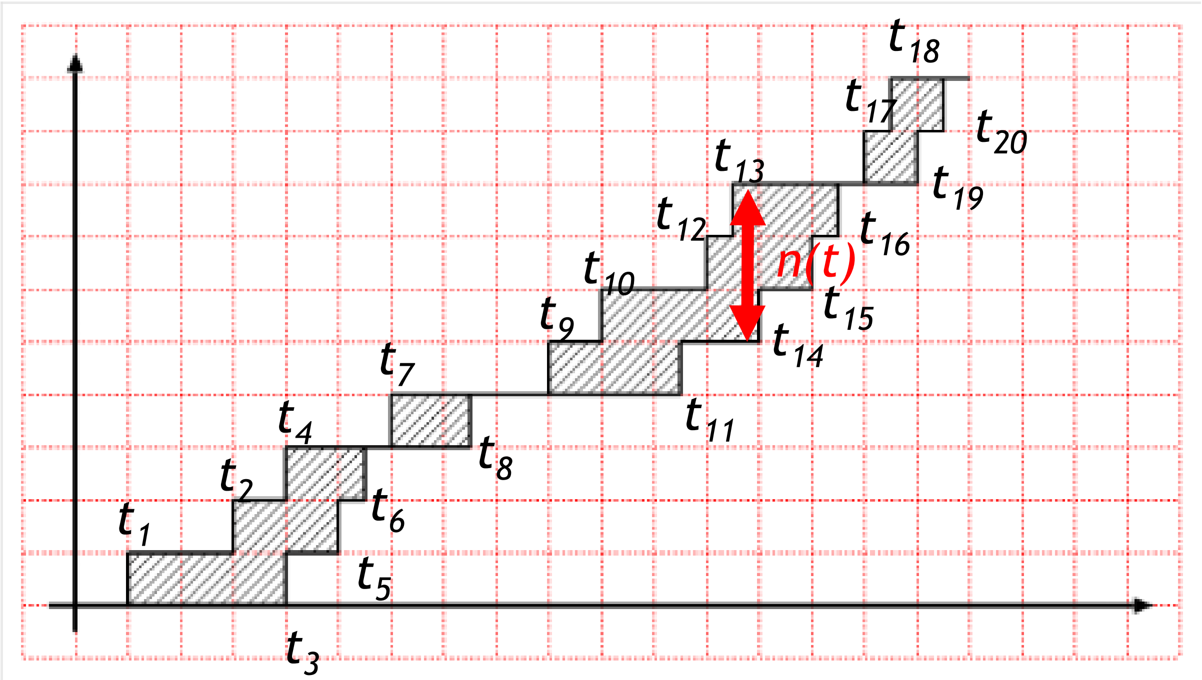
Queue length distribution

At a given point in time t between two instants t_i and t_{i+1}, the number of jobs in the system $n(t)$ is constant and equal to:

$$n(\mathrm{t}) = A(t) - C(t)$$

Please remember the assumption that the system starts empty.

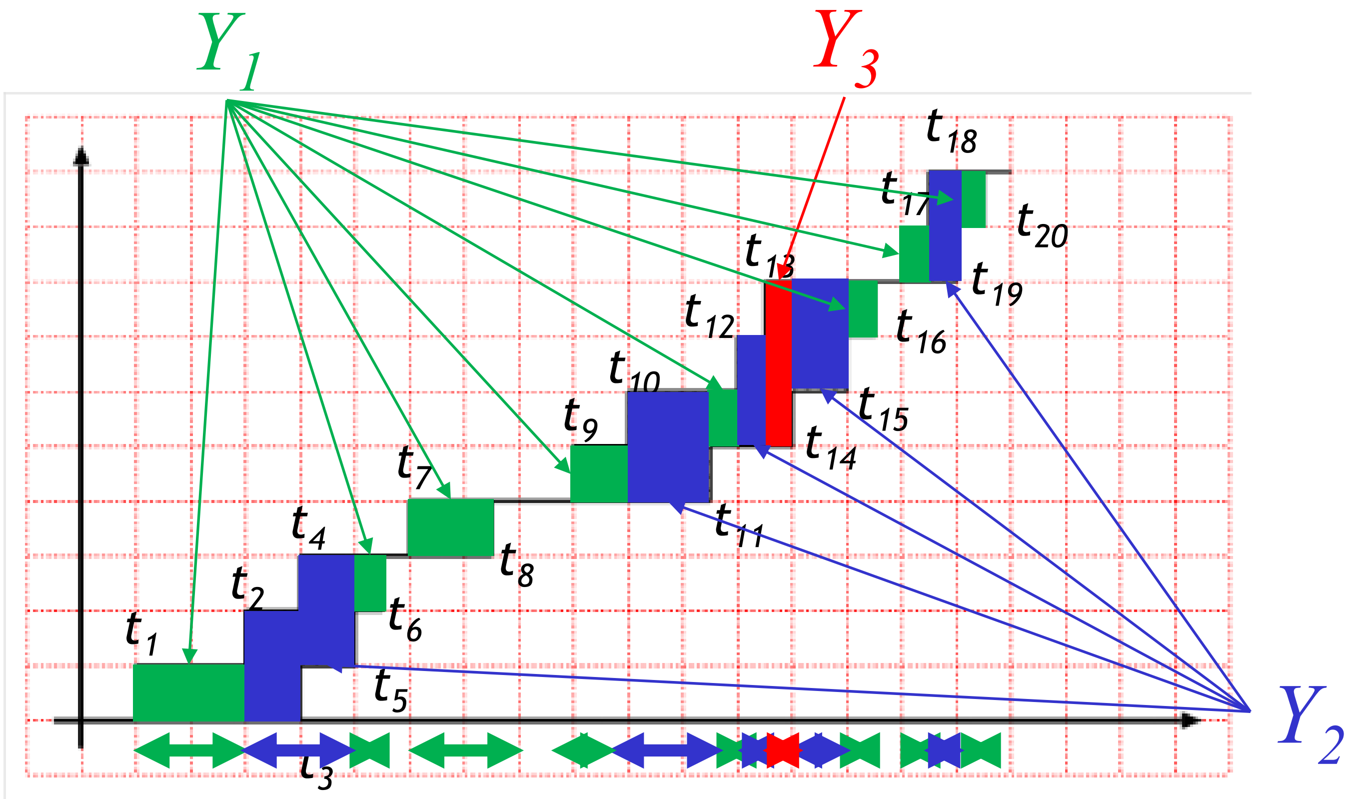
If the system starts with n_0 jobs inside, then we have:
$n(t) = A(t) - C(t) + n_0$



Queue length distribution

We can then compute Y_m as the fraction of time the system has m jobs.

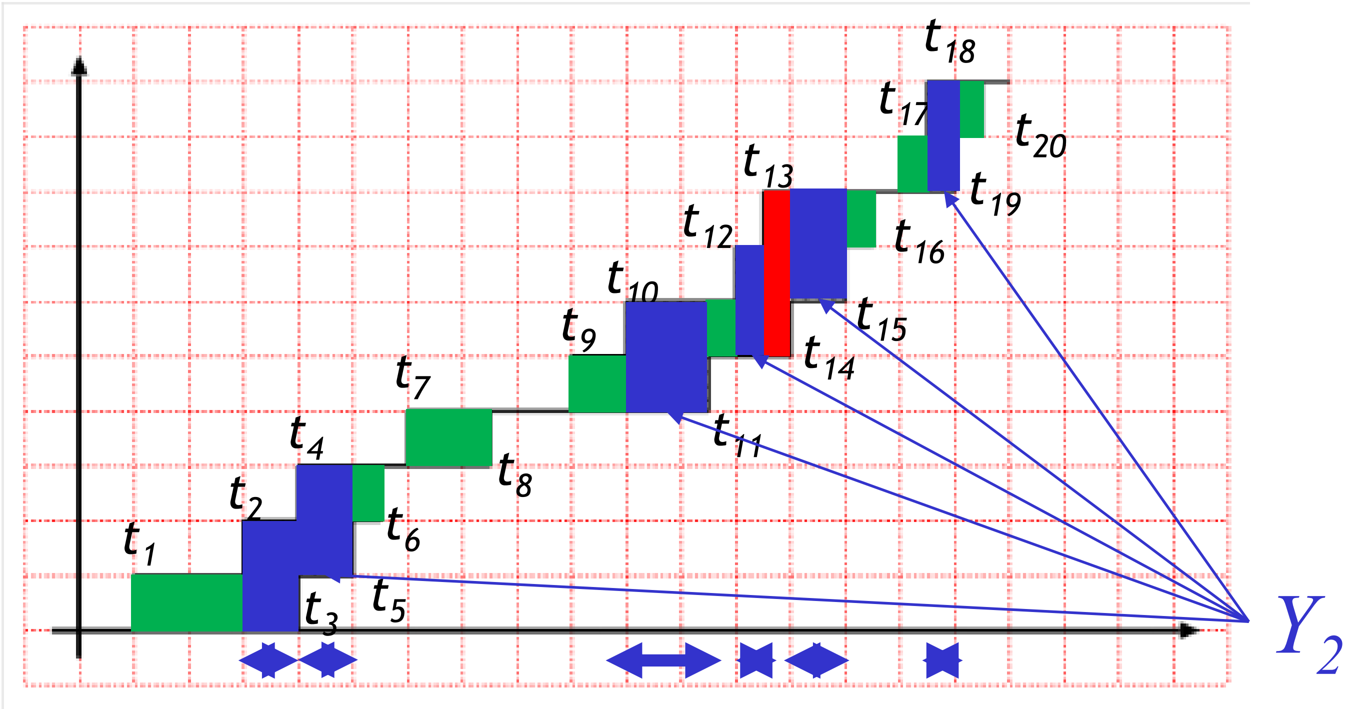
$$Y_m = \int_0^T I(n(t) = m)\, dt$$



Queue length distribution

Please note, that this integral can computed as a summation of the differences between consecutive time instants where we have the given number of jobs in the system.

$$Y_2 = \int_0^T I(n(t) = m)\,dt = (t_3 - t_2) + (t_5 - t_4) + (t_{11} - t_{10}) + (t_{13} - t_{12}) + (t_{15} - t_{14}) + (t_{19} - t_{18})$$



Queue length distribution

We can then approximate the probability of having *n* jobs in the system in the following way:

$$p(N = m) = \frac{Y_m}{T}$$

Note that also in this case, the technique can be extended to compute the probability that a given predicate $\Psi(\mathrm{N})$ on the number of jobs is true. If we call $Y_{\Psi(\mathrm{N})}$ the time in which the system fulfills such property, we have:

$$p(\Psi(\mathrm{N})) = \frac{Y_{\Psi(\mathrm{N})}}{T}$$

Queue length distribution

With these relations, we can estimate *B*, *W* and *N* in other ways:

$$B = \sum_{m=1} Y_m = T - Y_0$$

$$W = \sum_{m=1} m \cdot Y_m$$

$$N = \sum_{m=1} m \cdot p(N = m)$$

Analysis of Motivating Example

Analyzing the log file with the techniques just seen, the following performance indices have been determined:

- •Average queue length
- •Utilization of the Booth
- •Average service time
- •Arrival rate
- •Average response time
- •Probability that a costumer has to wait more than 15 minutes.

```
Average Number of jobs: 3.80549
Utilization: 0.841256
Average Service Time: 2.54616
Arrival Rate: 0.330402, Throughput 0.330402
Average Response Time: 11.5178
Pr(R>15): 0.2628
```