# Poisson process

With intensity parameter $\lambda$ is a counting stochastic process with continues time $\{N(t), t \geq 0\}$ for which $N(0)=0$ and ①, ②, ③ (counting here means tha we begin from 0 and always increment the value)

① is time-homogerous : $(\forall_{t,s>0})(N(t+s)-N(s) \sim N(t)-N(0))$

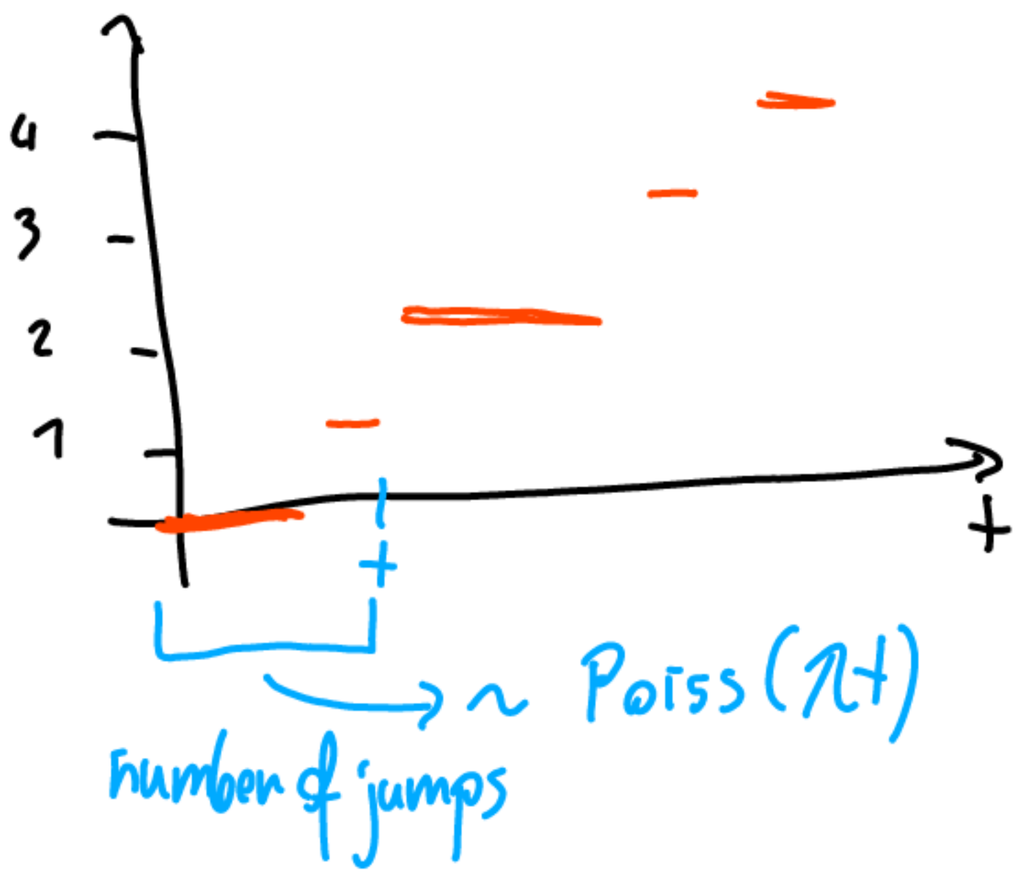
② has independent increments : $[t_1, t_2] \cap [t_3, t_4] = \emptyset \rightarrow$

$N(t_2)-N(t_1) \wedge N(t_4)-N(t_3)$ are ind.

③ $\lim_{t \to 0} \frac{Pr[N(t)=1]}{t} = \lambda \equiv \begin{cases} Pr[N(t)=1] = \lambda t + o(t) \\ Pr[N(t)=0] = 1-\lambda t + o(t) \\ Pr[N(t) \geq 2] = o(t) \end{cases}$

so, what is the probability that we have a jump in this interval.

---



**Theorem 1** The only process that meets the criteria of the above definition is a process for which

$$(\forall t,s>0)\,(\underbrace{N(t+s)-N(s)}_{\text{number of jumps}} \sim Poiss(\lambda t))$$

we believe in it (proof is in Mitzenmacher, Theorem 8.7)

---

**Theorem 2** Let $T_i$ be r.v. denoting the time between $(i-1)$-th and $i$-th event in process $\{N(t), t \geq 0\}$. Then

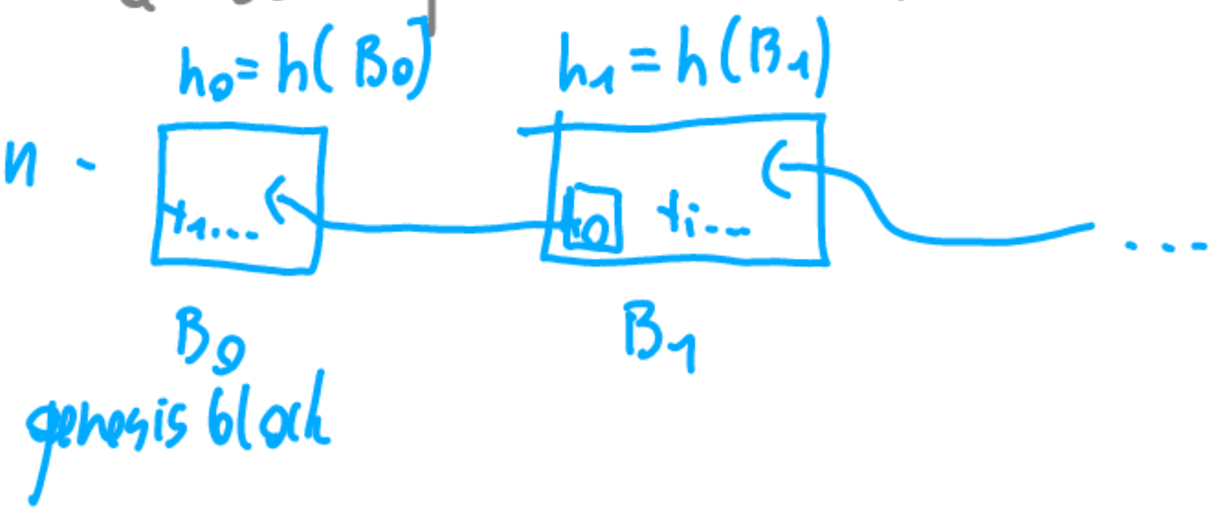
$T_1, T_2, \ldots$ are independent, $T_i \sim Exp(\lambda) \leftrightarrow \{N(t), t \geq 0\}$ is poisson process with parameter $\lambda$.

(proof as above, theorem 8.10.)

---

## Blockchain (short summary)

- transaction $t$ - declaration that user A sends funds to user B digitaly signed.
- block B - a set of transactions
- blockchain -



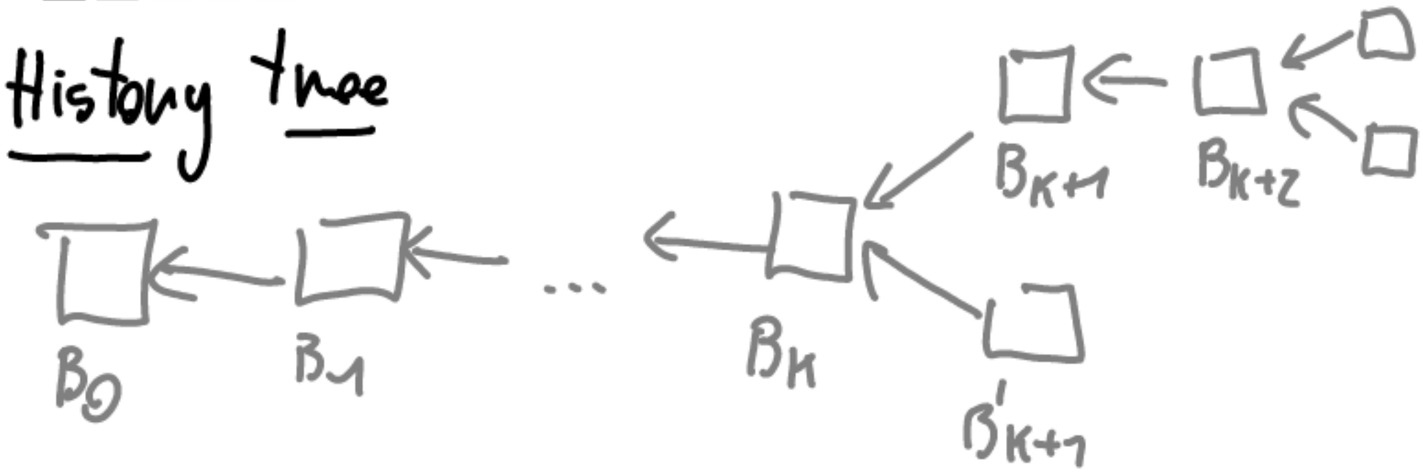

- distributed ledger - based on consensus mechanism.

## How do we build a single block? - block mining

1) Draw at random a nonce $\in \{0,1,\dots 2^6-1\}$
2) $h = h(\text{nonce} \| h\_prev \| t_1 \| \dots \| t_n) \in [0,1)$
3) if $h < \delta$ you mined a block and publish it
   else draw at random a new nonce
4) User that mined a block gets a reward.

---

## History tree




- we would like to find one valid version of the history - a root from the root to the leaf.
- typically - the longest at the moment path

---

## Confirmed transaction

a) transaction is in the largest path
b) it is in a block that is appended n blocks

---

## Double spending - attack

1) $U_1$ buys from $U_2$ a product and broadcasts over network digitally signed transaction $t$, in which $U_1$ transfers coins to $U_2$.
2) Block $B_{K+1}$ with transaction $t$ is attached to the longest branch. $U_2$ waits for n blocks and ships the product.
3) In this time $U_1$ creates $t'$ ($U_1 \xrightarrow{t'} U_3$) puts it in block $B_{K+1}$

and creates (secretly) an alternative branch

---

## Mathematical model

1) $T_k$ - time to generate block $B_k$. $T_k$ is memoryless and continuous.

we assume that set for looking for nonce is so big that we can assume that is almost memoryless (like sele. with returning)

- Thus $T_k \sim Exp(\lambda)$, $E[T_k] = \frac{1}{\lambda} \approx 10$ minutes
- $\lambda$ is connected to hashrate and difficulty level $\delta$.

2) Since $T_1, T_2, \dots$ are independent, $T_i \sim Exp(\lambda)$ so from Th.2 $\{N(t), t \geq 0\}$ where $N(t)$ represents the number of mined blocks up to time $t$, is Poisson process with parameter $\lambda$.

From Th.1 we have that

$$Pr[N(t)=n] = \frac{(\lambda t)^n \cdot e^{-\lambda t}}{n!}$$

---

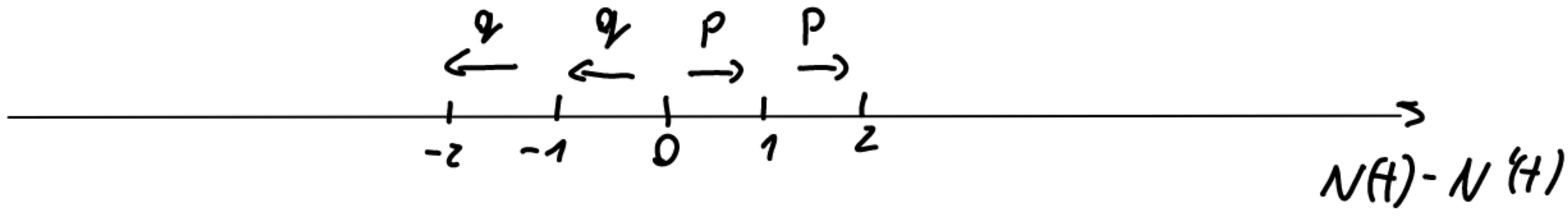
normal users

- $\lambda, \lambda'$ - hashrates $\lambda' < \lambda$

adversary

- $T \sim Exp(\lambda)$, $T' \sim Exp(\lambda')$, corresponding Poisson process $N(t), N'(t)$
- $Pr[\min\{T, T'\} = T'] = Pr[T' < T] = \frac{\lambda'}{\lambda + \lambda'} = q$ , $q < \frac{1}{2}$
- so the p-b that normal users $p = 1 - q$

we called this model : <u>Random walk on line</u>




we are intrested in finding probability that $N(t) - N'(t) \leq 0$ even after we have n moves to the right