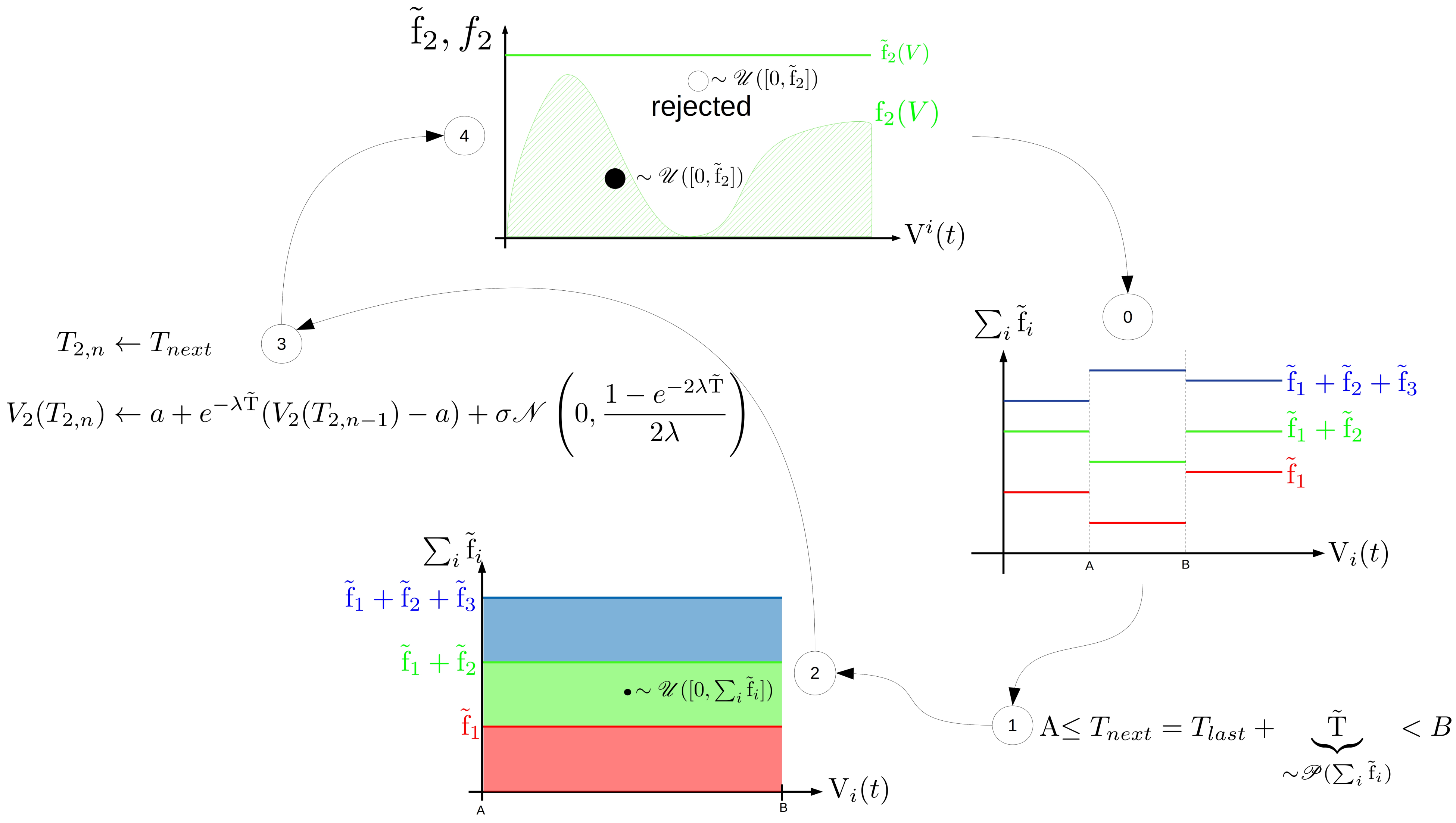

Figure 1: Illustration of algorithm 2

Algorithm 2 Simulator

N: number of neurons
$\tilde{\mathrm{T}}$: time elapsed since last event in the system
T_{last}: absolute time of the last event in the system
T^{i}_{next}: absolute time of the nth event of neuron i
$\mathscr{P}$: poisson distribution
$\mathscr{U}$: uniform distribution
$\tilde{\mathrm{f}}$: approximation function of function f ($\tilde{\mathrm{f}}(x) \geq f(x)$)
repeat
  repeat ▷ Steps 0-1
    determine an interval $[A, B[$ on which sampling
    compute the array of $(\tilde{\mathrm{f}}_i(V_i(A)))_{i\in\{1,\cdots,N\}}$ and $\sum_i \tilde{\mathrm{f}}_i(V_i(A))$ on interval $[A, B[$
    $\tilde{\mathrm{T}} \sim \mathscr{P}(\lambda = \sum_i \tilde{\mathrm{f}}_i)$
    $T_{next} \leftarrow T_{last} + \tilde{\mathrm{T}}$
  until in good interval AND at least one $\tilde{\mathrm{f}}_i \neq 0$
  $i \leftarrow \mathbf{argmin}_{i\in\{1,\cdots,N\}} \left(\sum_j \tilde{\mathrm{f}}_j(V_j(A)) * u \sim \mathscr{U}([0,1]) < \tilde{\mathrm{f}}_i(V_i(A)) \right)$ ▷ Step 2
  $T_{i,n} \leftarrow T_{next}$ ▷ Step 3
  $V_i(T_{i,n}) \leftarrow a + e^{-\lambda\tilde{\mathrm{T}}}(V_i(T_{i,n-1}) - a) + \sigma\mathscr{N}(0, \frac{1-e^{-2\lambda\tilde{\mathrm{T}}}}{2\lambda})$
  $\mathbb{P}(\text{accepting spike of neuron i}) = \frac{f_i(V_i(T_{i,n}))}{\tilde{\mathrm{f}}_i(V_i(A))}$ ▷ Step 4
  if spike is accepted **then**
    update potentials of all postsynaptic neurons
until an end condition is met

$$S = \begin{pmatrix} \mathrm{S}_0, & \mathrm{I}_1 \\ \mathrm{S}_{I_1}, & \mathrm{I}_2 \\ \vdots, & \vdots \\ \mathrm{S}_{\sum_{k=1}^{i-1}}, & \mathrm{I}_i \\ \vdots, & \vdots \\ \mathrm{S}_{\sum_{k=1}^{N-1}}, & \mathrm{I}_N \end{pmatrix}$$

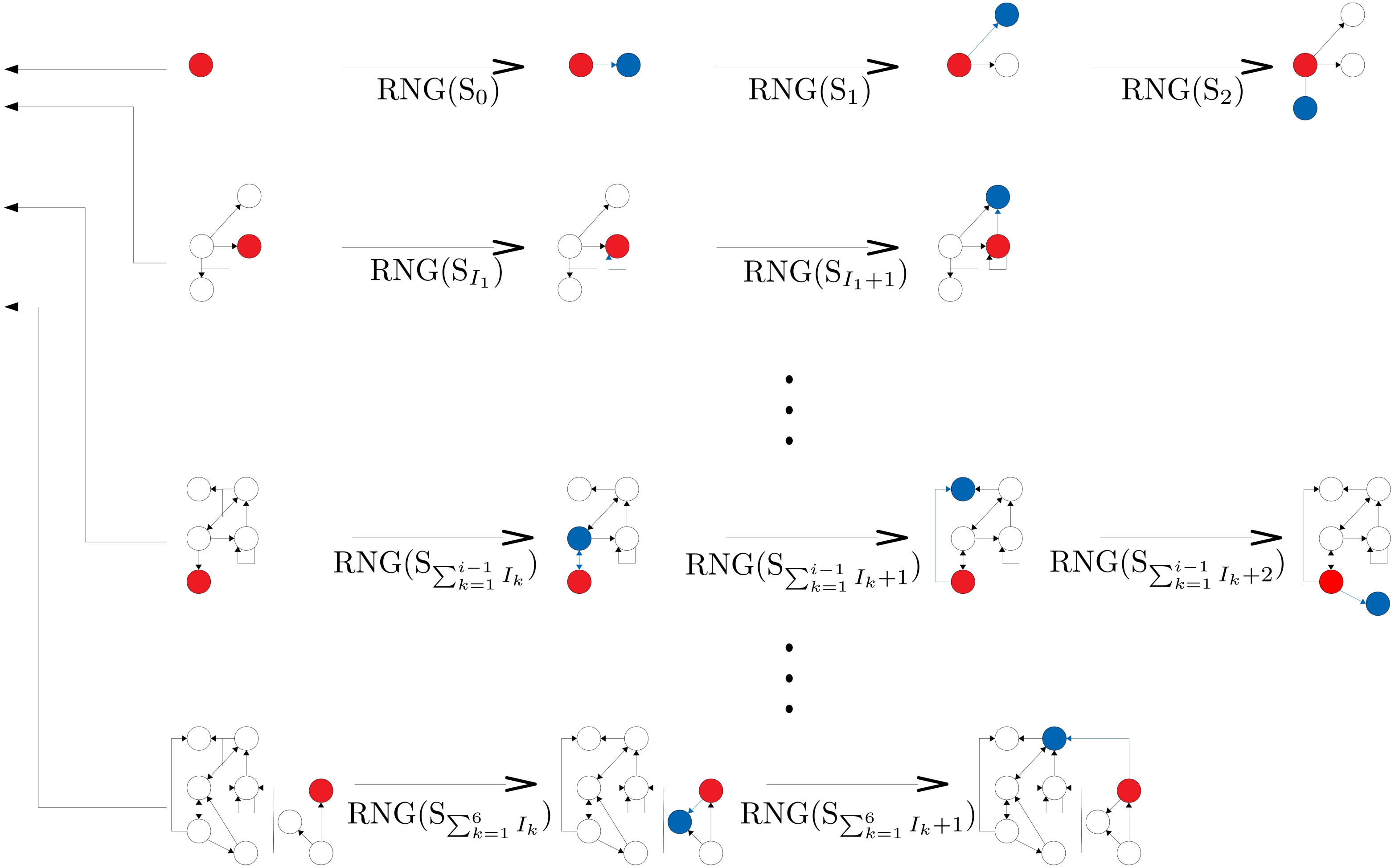

Figure 2: Construction of graph and vector of states

Algorithm 3 Generation of a matrix of children

RNG: Random Number Generator
p: Probability of connection between two neurons
$\mathscr{M}_{i,j}$: matrix of interaction: equals 1 for a connection, 0 otherwise
RNG.B(p): returns 0 or 1, bernouilli distributed with probability p.
function Interaction matrix(RNG, p)
    for i $\leftarrow$ 1 **to** n **do**
        for j $\leftarrow$ 1 **to** n **do**
            $\mathscr{M}_{i,j}$ $\leftarrow$RNG.B(p)

Algorithm 4 Generation of a vector of rng states

RNG: random Number Generator
p: probability of connection between two neurons
N: number of neurons in the system
S: vector of pairs (Status of the RNG, Number of children of neuron i)
RNG.Bin(N,p): returns a random value following the binomial distribution of parameter N and p
RNG(S_i): returns the index of a child for i among all other unchosen neurons
RNG.Status: returns the current internal state of the RNG
I_i: number of children of neuron i
function Make vector(RNG, p)
    for i $\leftarrow$ 1 **to** N **do**
        I_i $\leftarrow$RNG.Bin(N,p)
        $S[i]$ $\leftarrow$(RNG.Status,I_i)
        for j $\leftarrow$ 1 **to** I_i **do**
            RNG($\mathrm{S}_{j+\sum_{k=1}^{i-1} I_k}$)

Algorithm 5 Comparison of usage between clasical method and reconstruction

function Comparison
    Interaction Matrix(RNG_1,p)
    Make Vector(RNG_2,p)
            $\vdots$ Simulation
    Neuron i is spiking
    $\triangleright$Using matrix graph
    for j $\leftarrow$ 1 **to** N **do**
        if $\mathscr{M}_{i,j} = 1$ **then**
            Update potential of neuron j
    $\triangleright$Using reconstruction
    S[i]=($\mathrm{S}_{\sum_{k=1}^{i-1} I_k}, I_i$)
    RNG.SetState(S[i][1])
    for j $\leftarrow$ 1 **to** I_i **do**
        Child_j $\leftarrow$RNG($\mathrm{S}_{j-1+\sum_{k=1}^{i-1} I_k}$)
        Update potential of neuron Child_j

	Algorithmic complexity at creation	Algorithmic complexity during usage	Memory storage
Reconstruction method	$\mathscr{O}(N^2)$	$\mathscr{O}(N)$	$\mathscr{O}(N)$
Interaction matrix	$\mathscr{O}(N^2)$	$\mathscr{O}(1)$	$\mathscr{O}(N^2)$

Table 1: Table of memory and algorithmic complexity

$$\text{Event-driven complexity: } \mathcal{O}\left(\frac{\nu c N}{\text{acceptation rate}}\right)$$

$$\text{Clock-driven complexity: } \mathcal{O}\left(\frac{\nu c N^2}{\text{Blow-up time interval}}\right)$$

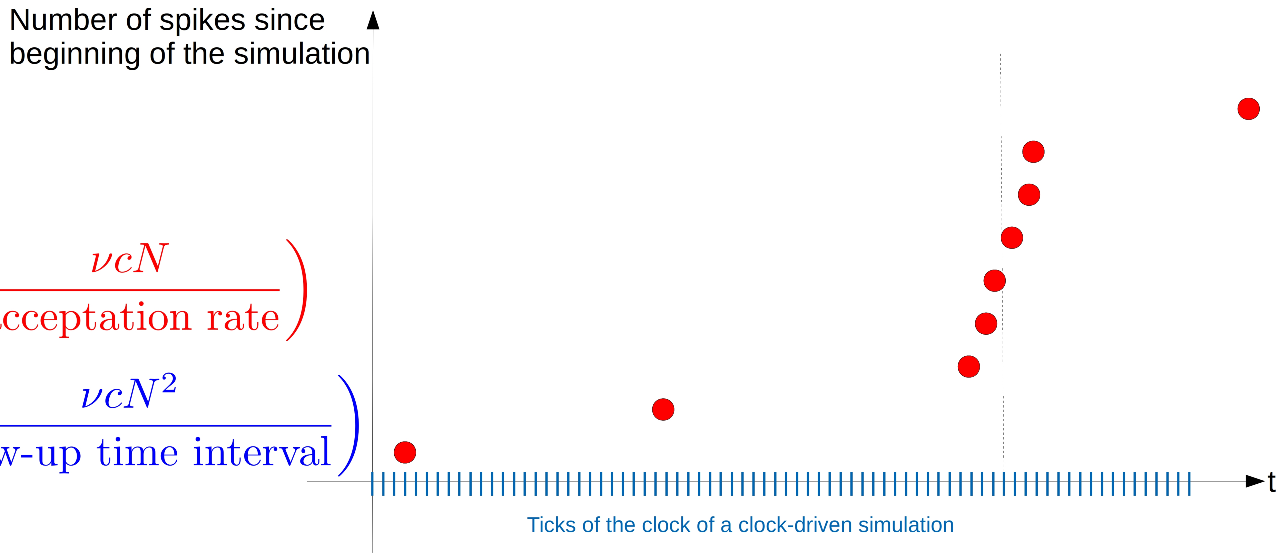

Figure 3: Complexity as number of events during a time interval