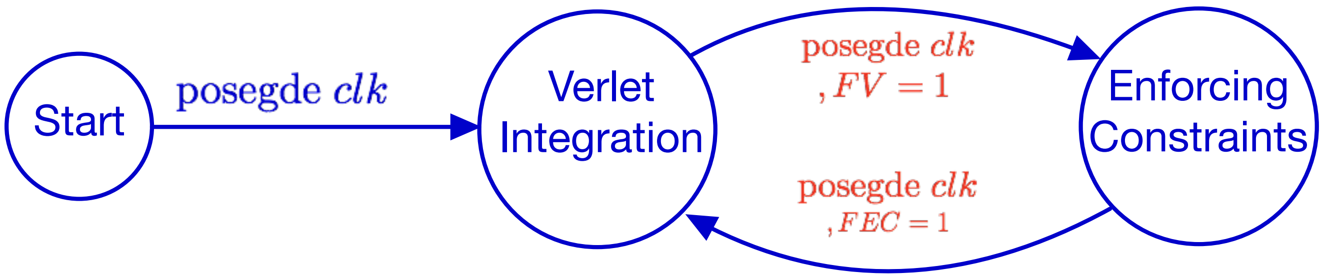
FSM diagram for *Rope Simulation* using *Verlet Integration* and *Jacobsen's Method*.

FSM diagram for simulating the rope motion using Verlet Integration and Jacobsen method.

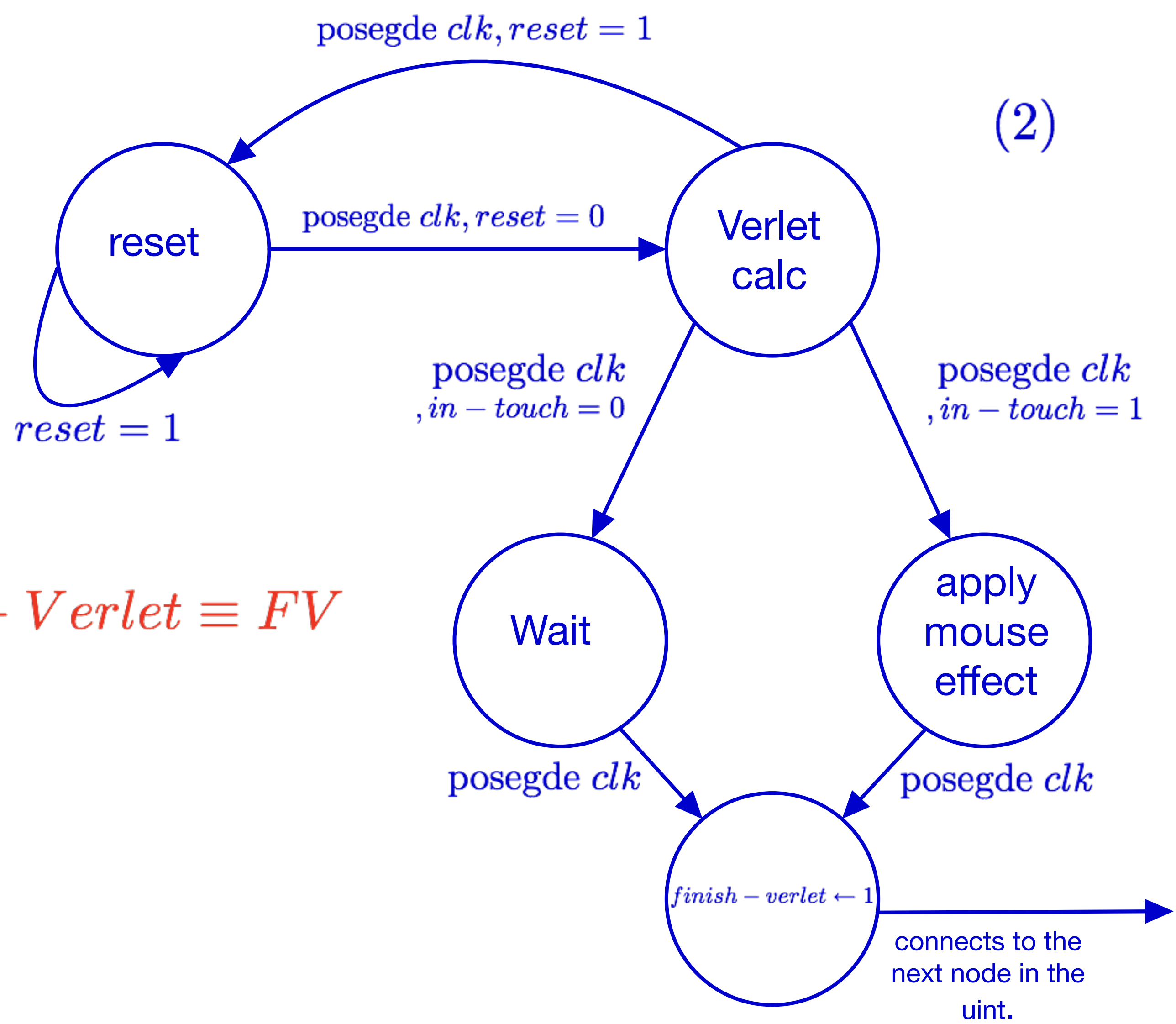
The overall process is represented as



(1)

For each node, we draw an FSM for reset and verlet integration, then apply force coming from mouse touch.

Input: $clk, reset, mouse$ Signal: $in - touch$ Output: $finish - verlet$



(2)

$$finish - Verlet \equiv FV$$

$in - touch \leftarrow |mouse.x - node.x| < minX$ and $|mouse.y - node.y| < minY$

reset: Set node to an initial position (x, y).

Verlet calc: Calculate next position of the node using verlet integration.

Verlet calculation is done as follows:

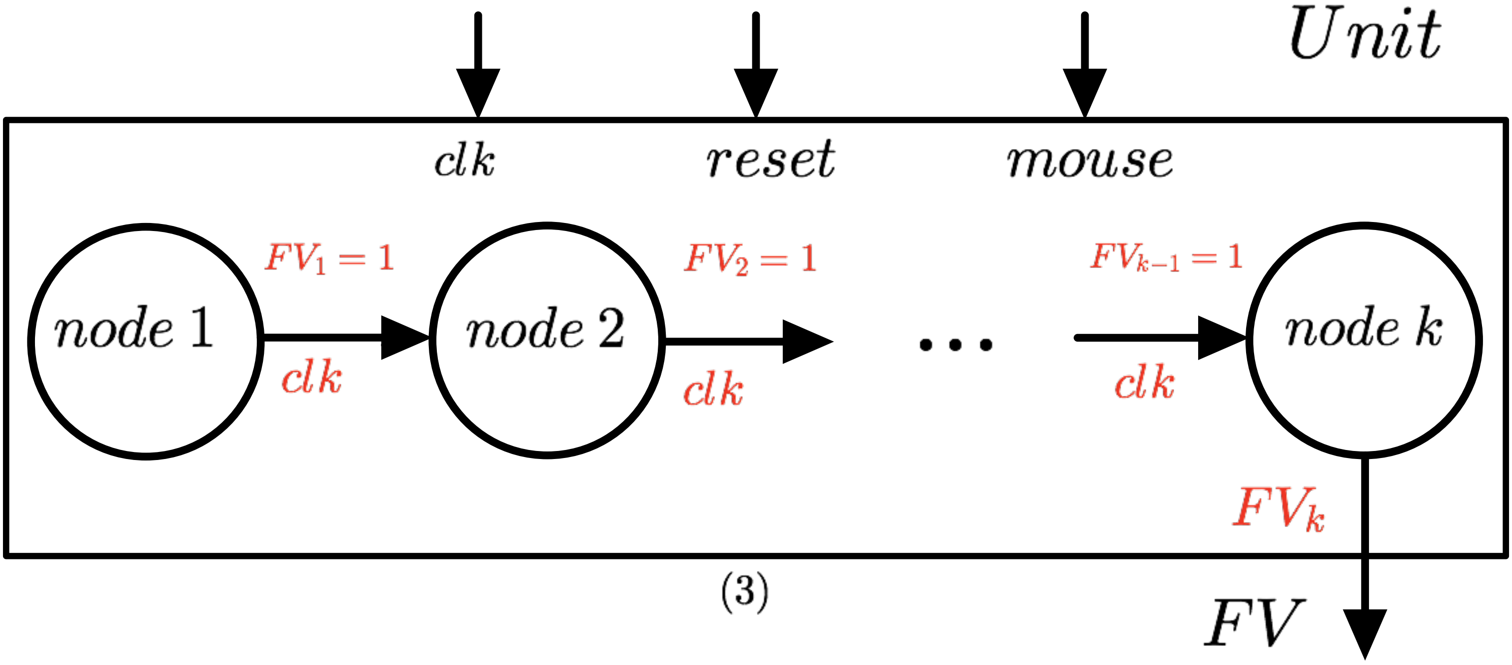
$$\begin{aligned} vx &\leftarrow node, n - node.px \\ vy &\leftarrow node, y - node.py \\ node, px &\leftarrow node.x \\ node.py &\leftarrow node \cdot y \\ node.x &\leftarrow node \cdot x + vx \\ node \cdot y &\leftarrow node.y + vy + \Delta t^2 \times gravity \end{aligned} \tag{1}$$

Applying mouse effect is done as follows:

$$\begin{aligned} &\text{if} \quad mouse.x < node.px \quad \text{then} \\ &\quad node.px \leftarrow node.x + a \\ &\text{else} \\ &\quad node.px \leftarrow node.x - a \end{aligned} \tag{2}$$

Where a is a constant.

For each node in the rope soft body, The above process is done concurrently. In case of using a single processing unit for a patch of nodes, for example k node per unit. Each unit stores its nodes as an array structure each unit will be presented as follows:

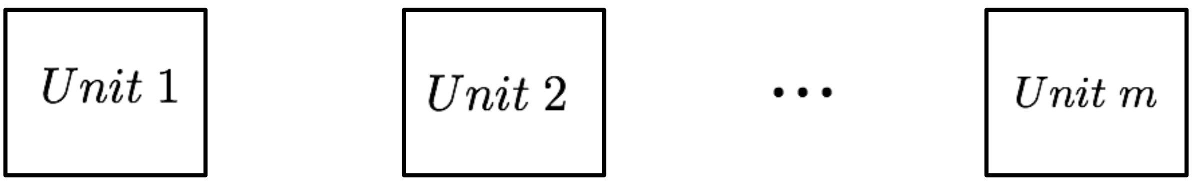


(3)

FV in diagram (1), is logical AND of all FV outputs of Units in diagram (3).

After full execution of previous steps. now we enforce constraints. This step is done to simulate the rope motion by maintaining a constant distance between every two nodes as the constraint.

Suppose we have m units, each consisting of k nodes. then the diagram for this would be:

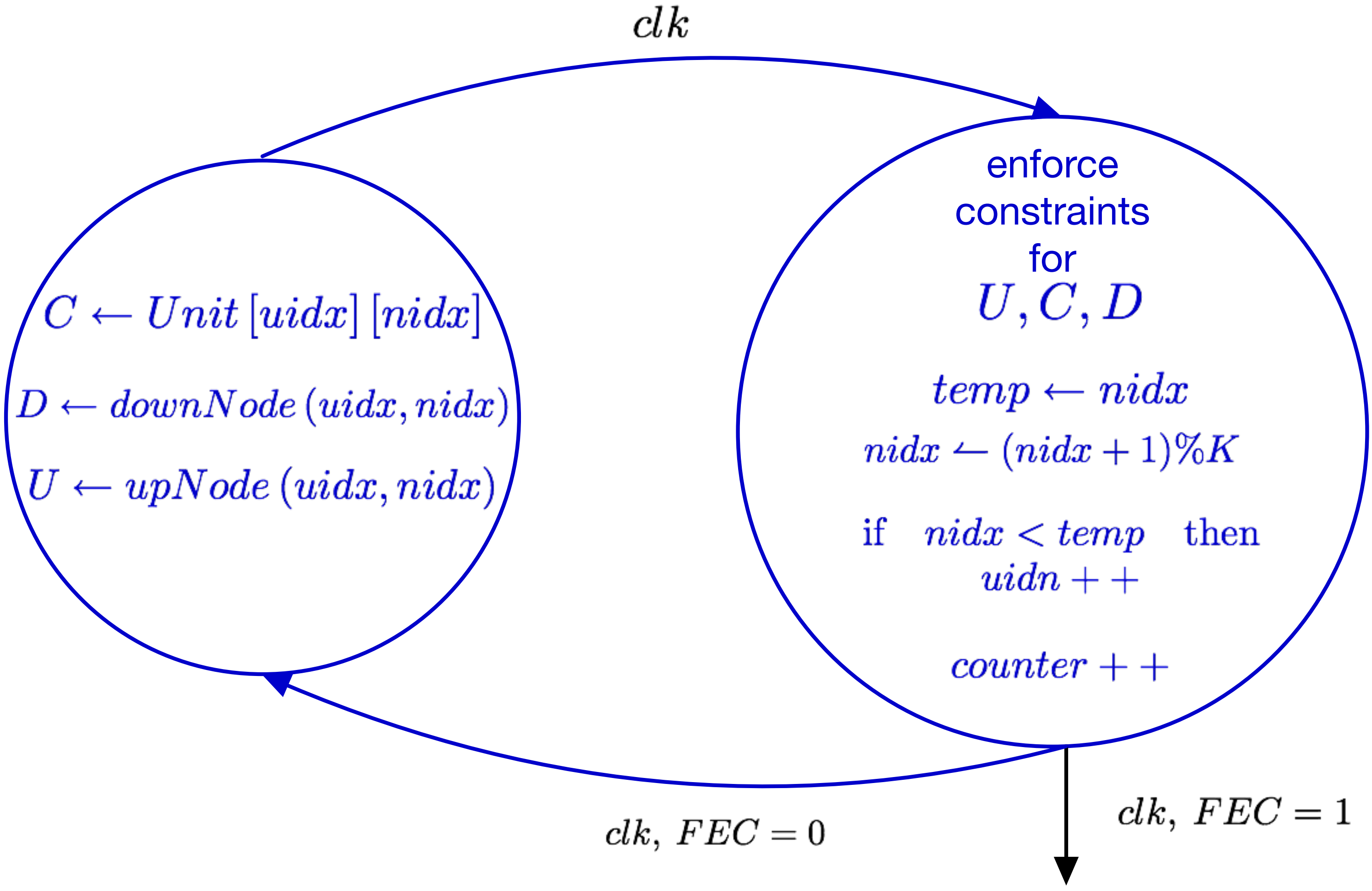


(4)

Regs: $uidx,\ nidx,\ counter$ Output: FEC

Consider above units as an array of length m, accessible by $uidx$. and each node in each unit accessible by $nidx$.

In each iteration we keep track of three nodes: U, C, D



(5)

$FEC \leftarrow counter = m \times k$

$downNode(i, j):$

$$\begin{aligned}&\text{if}\quad j = k-1 \quad \text{then} \quad \text{return} \quad unit\,[i+1]\,[0]\\ &\text{else} \quad \text{return} \quad unit\,[i]\,[j]\end{aligned}$$

$upNode(i, j):$

$$\begin{aligned}&\text{if}\quad j = 0 \quad \text{then} \quad \text{return} \quad unit\,[i-1]\,[k-1]\\ &\text{else} \quad \text{return} \quad unit\,[i]\,[j]\end{aligned}$$

And enforcing constraints for U, C, D is done as follows:

$$
\begin{aligned}
&dxu \leftarrow C.x - U.x \\
&dyu \leftarrow C.y - U.y \\
&du \leftharpoonup \sqrt{dxu^2 + dyu^2} \\
&nxu \leftarrow U.x + \frac{dxu}{du} \times \ dist \\
&nyu - U.y + \frac{dyu}{du} \times \ dist \\
&nxd \text{ and } nyd \text{ are calcuated the same way} \\
&C.x = \frac{nxu + nxd}{2} \\
&C.y = \frac{nyu + nyd}{2}
\end{aligned}
$$

Where $dist$ is a constant.