FFL-attachment based algorithm

Initialize params: $N, M, \gamma, K_{max}, p = (p_1, p_2, p_3, p_4)$
Subset nucleus $\overrightarrow{G}_0(V, E)$ **from target network**
Grow network:
while $size(\overrightarrow{G}) < N$ **do**
    if $0 < RandomUniform[0,1] \leq p_1$ **then**
        if $\mathcal{L}_{out} \leq K_{max}$ **then**
            Choose node $\mathcal{L}$ to connect at random with probability $\frac{L^\gamma}{\sum_{i=1}^{n} L_i^\gamma}$
            Apply **R1 rule**
        end if
    else
        Choose random FFL-motif $\subset \overrightarrow{G}$
        if $0 < RandomUniform[0,1] \leq p_2$ **then**
            Apply **R2 rule**
            FFL_{TTG} created
        else
            Apply **R3** or **R4 rule** with p_3, p_4 probabilities
            FFL_{TTT} created
        end if
    end if
end while
Control for sparsity in $\overrightarrow{G}(V, E)$:
while $\frac{sum(E)}{V(V-1)} \neq M$ **do**
    if $\frac{sum(E)}{V(V-1)} > M$ **then**
        Choose node $\mathcal{V}$ with probability proportional to out-degree $\mathcal{V}_{out}$
        Select random edge e_i where $\mathcal{V}_{in} = \{e_1, ..., e_i, ..., e_k\}$
        $\overrightarrow{G} = \overrightarrow{G} - e_i$
    else
        Choose node $\mathcal{V}$ with probability proportional to out-degree $\mathcal{V}_{out}$
        Add edge to a random node $\mathcal{W}$ where $e = (e_\mathcal{V}, e_\mathcal{W})$
        $\overrightarrow{G} = \overrightarrow{G} + e$
    end if
end while
return $\overrightarrow{G}(V, E)$