

```

%-----
% Author: Akira Nagamori
% Last update: 6/8/17
% REFERENCE:
% Fuglevand et al. 1993
% Fuglevand et al. 2006
%-----

function [time,spike_train] = MotorUnitModel(pool_parameter,Input)
%-----
% Input: user-defined parameters of the model
%       excitation input to motor unit
% Output: time vector and spike train data
%-----
% model parameters
N = pool_parameter.N; %number of motor unit
i = 1:N; %motor unit identification index
RR = 30; %range of recruitment in unit of fold
a = log(RR)/N; %coefficient to establish a range of threshold values
RTE = exp(a*i); %recruitment threshold excitation
MFR = 8; %minimum firing rate constant for all motoneurons
g_e = pool_parameter.gain; %missing parameter from the paper
PFR1 = 35; %the peak firing rate of the first recruited motoneuron in
unit of impulse/sec
PFRD = 10; %the desired difference in peak firing rates between the
first and last units in unit of impulse/sec
RTEn = exp(a*N); %recruitment threshold of the last motor unit
PFR_v1 = PFR1 - PFRD * (RTE./RTEn); %peak firing rate
PFRn = PFR1 - PFRD; %peak firing rate of the last motor unit
Emax = RTEn + (PFRn - MFR)/g_e; %maximum excitatory input
cv = pool_parameter.ISI_CoV; %ISI variability as per coefficient of
variation (=mean/SD)
%-----
% time
Fs = 1000; % sampling frequency
time = 0:1/Fs:(length(Input)-1)/Fs;
%-----
% excitatory drive to motor unit model
E = Emax.*Input;
%-----
% parameter initialization
spike_time = zeros(1,N); % spike time for next
spike_train = zeros(N,length(time)); % binary spike trains
time_stamp = zeros(1,N);
PIC = zeros(1,N);
%-----
% simulate motor unit model
for j = 1:length(time)

    FR = g_e.*(E(j) + PIC - RTE) + MFR; % firing rate of motoneurons
    FR(FR<MFR) = 0;

    for n = 1:N
        if FR(n) > PFR_v1(n)
            FR(n) = PFR_v1(n);
        end
        if FR(n) == 0
            mu(n) = 0;
        end
    end
end

```

```

else
    mu(n) = 1/FR(n); % predicted inter-spike intervals
end
if spike_time(n) == 0 % for the first spike
    Z = randn(1); % Z-score
    Z(Z>3.9) = 3.9;
    Z(Z<-3.9) = -3.9;
    spike_time(n) = mu(n) + mu(n).*cv.*Z + spike_time(n); %
spike time with noise added
    if spike_time(n) ~= 0
        spike_train(n,j) = 1; % add 1 to a matrix when the unit
fires
        time_stamp(n) = j;
        PIC(n) = 2; % add PIC once the unit started firing
    end
else
    if j == (time_stamp(n) + round(spike_time(n)*Fs))
        Z = randn(1);
        Z(Z>3.9) = 3.9;
        Z(Z<-3.9) = -3.9;
        spike_time(n) = mu(n) + mu(n).*cv.*Z + spike_time(n); %
spike time with noise added
        spike_train(n,j) = 1;

    elseif j > (time_stamp(n) + round(spike_time(n)*Fs)) &&
FR(n) >= 8
        Z = randn(1);
        Z(Z>3.9) = 3.9;
        Z(Z<-3.9) = -3.9;
        spike_time(n) = mu(n) + mu(n).*cv.*Z + spike_time(n); %
spike time with noise added
    end
end
end
end
end
end

```