
```
clear all;
clc;

% constantes
cdata.L = 200; % comprimento total
cdata.v=5; % velocidade
cdata.tmax = 20; % tempo total
cdata.cfl = 1; % cfl (valor para a condicao de estabilidade)
cdata.intervalos_impressao = [0, 5, 10, 15, 20, 30];

% definicoes do usuario
%cdata.n = 20; % quantidade de blocos
%cdata.n = 40;
cdata.n = 80;

% preprocess
cdata.h = cdata.L/cdata.n; % tamanho dos blocos
cdata.sat0 = zeros(cdata.n, 1); % saturacao inicial
cdata.sat0(1) = 1;
cdata.indices = 1:cdata.n;
cdata.dt = cdata.cfl*cdata.h/cdata.v; % passo de tempo

% simulacao
t_simulation = 0;
loop = 0;
count_impr = 2;
plot_results = false(1,1);

resp.saturations(count_impr-1,:) = cdata.sat0;
resp.times(count_impr-1) = t_simulation;
resp.positions = zeros(cdata.n,1);
resp.positions(1) = cdata.h/2;
for i = 2:cdata.n
    resp.positions(i) = resp.positions(i-1) + cdata.h;
end

v = cdata.v;
h = cdata.h;
ds = zeros(cdata.n, 1);

while t_simulation <= cdata.tmax

    % ver o dt no tempo de plotar os resultados
    loop = loop + 1;
    sat0 = cdata.sat0;
    dt = cdata.dt;
    time_to_plot = cdata.intervalos_impressao(count_impr);
    if t_simulation + dt >= time_to_plot
        dt = time_to_plot - t_simulation;
        plot_results(:) = 1;
    end
    t_simulation = t_simulation + dt;
```

```
k = -v*dt/h;
ds(:) = 0;
% calcular a saturacao
for i = 2:cdata.n
    ds(i) = k*(sat0(i) - sat0(i-1));
end
new_sat = sat0 + ds;

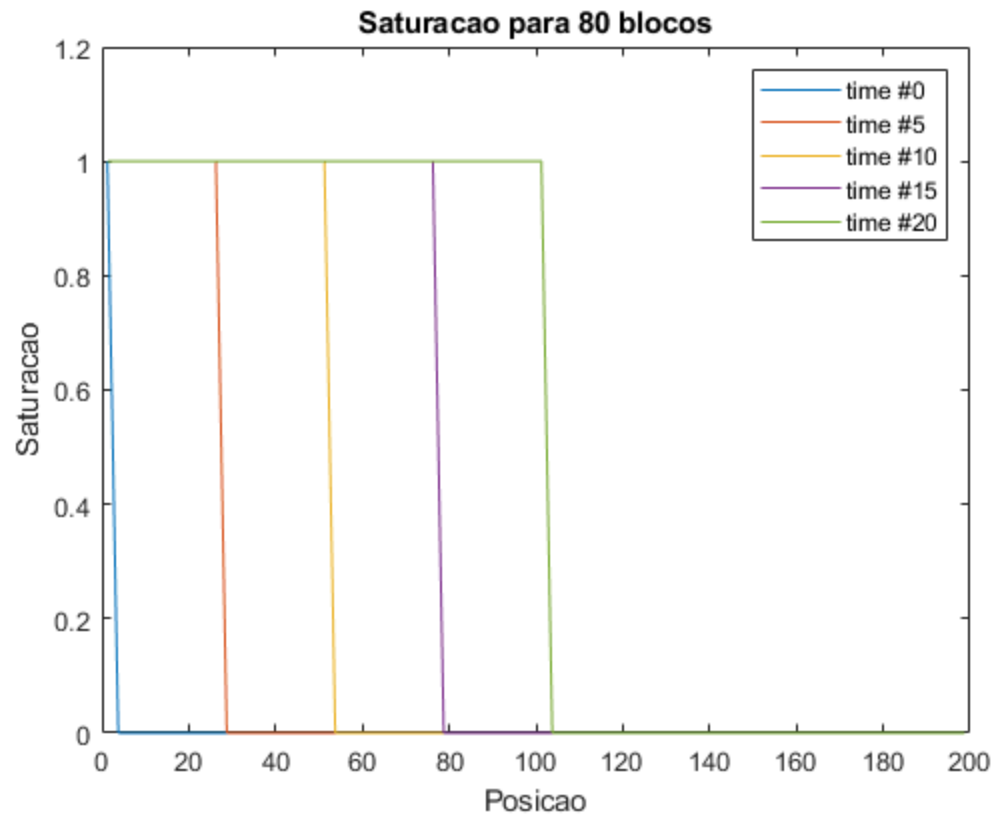
if plot_results
    resp.times(count_impr) = t_simulation;
    resp.saturations(count_impr,:) = new_sat;
    plot_results(:) = 0;
    count_impr = count_impr + 1;
end

cdata.sat0(:) = new_sat;

end

% plot results
n_plots = size(resp.saturations);
figure;

for i = 1:n_plots(1)
    plot(resp.positions, resp.saturations(i,:), 'DisplayName', ['time
    #' num2str(resp.times(i))]);
    hold on;
end
ylim([0 1.2]);
title(['Saturacao para ' num2str(cdata.n) ' blocos']);
xlabel('Posicao');
ylabel('Saturacao');
legend();
```



Published with MATLAB® R2020a