Contents

```
function result = idm_final(t,x,guideMap,disMatrix,triggerDis)
```

```
%IDM function used in simulate.m
```

Parameter definitions

```
%Simulation Parameters
L = 1000000; %Length of the highway in m
lcar = 0; %Length of the cars in m
%Model Parameters
%desired speed in free traffic
s0 = 0.5; %minimum distance to next car
T = 1; %desired time headway to vehicle in front
a = 0.3; %maximum acceleration of a car
b = 3; %comfortable braking deceleration
delta = 4; %exponent used in equation
sStar = @(va,dva,dva1) s0 + va*T + va*dva/2/sqrt(a*b); %influence of the following car
v0=30; %Max. speed
```

Memory Allocation

```
result = zeros(length(x),1);
Ncars = floor(length(x)/2);
```

Application of different models

```
for ii = 1:Ncars
    if ii > 1 %All but first car
        dva = (x(ii+Ncars) - x(ii-1 + Ncars));

        sa = x(ii-1) - x(ii) - lcar;

    else %First car
        dva = x(ii+Ncars);
        sa = L - x(ii);
        dva1 = 0;

    end

    %State Space Equation
    if ~guideMap(ii) %"Normal" cars
        result(ii) = x(ii+Ncars);
        result(ii+Ncars) = min(a*(1 - (x(ii+Ncars)/v0)^delta - (sStar(x(ii+Ncars),dva,dva1)
/sa)^2),a);

    else %Guide Cars
```

```
        result(ii) = x(ii+Ncars);
        %Determine preceding guide car
        k = ii-1;
        while k>0 && ~guideMap(k)
            k=k-1;
        end

        if k <= 0 %if it is the first guide car
            result(ii+Ncars) = min(a*(1 - (x(ii+Ncars)/v0)^delta - (sStar(x(ii+Ncars),dva,d
va1)/sa)^2),a);
        else
            result(ii+Ncars) = min(a*(1 - (x(ii+Ncars)/v0)^delta - (sStar(x(ii+Ncars),dva,d
va1)/sa)^2 - (sa<triggerDis)*(x(ii+Ncars)-x(k+Ncars))),a);%(x(ii+Ncars)>x(k+Ncars))*
        end

    end
    %Introducing a disturbance
    if disMatrix(ii,1) ~= 0
        disLoc = disMatrix(ii,1);
        disLength = disMatrix(ii,2);
        disV = disMatrix(ii,3);
        v02 = @(x) max(v0 - (x-disLoc)*(v0-disV)/400,disV);
        if x(ii) > disLoc && x(ii) < disLoc + disLength
            result(ii) = x(ii+Ncars);
            result(ii+Ncars) = min(a*(1 - (x(ii+Ncars)/v02(x(ii)))^delta - (sStar(x(ii+Ncar
s),dva,dva1)/sa)^2),a);
        end

    end
    if result(ii)<0
        result(ii) = 0;
    end

end
```

```
end
```

Published with MATLAB® R2016b