

Annexos

Aquest annex conté imatges i figures que s'han obtingut durant el treball però no s'han inclòs al cos de la memòria. En aquest annex tampoc s'inclouran tots els cicles representats i estudiats.

I) Verificació de les dades

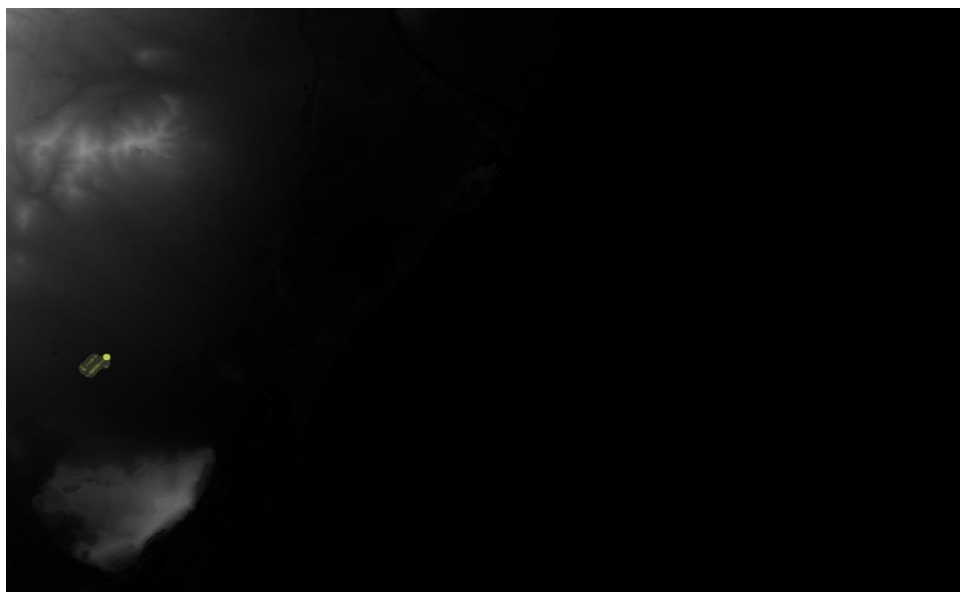


Figura 0.1: Perfil altimètric de la ciutat de Barcelona amb punts extrems a l'Eixample. La imatge s'obté amb el QGIS. Les zones fosques corresponen a alçades baixes, mentre que les més clares, corresponen als turons i pics que es troben a la ciutat i els seus voltants.

II) Resultats de les simulacions

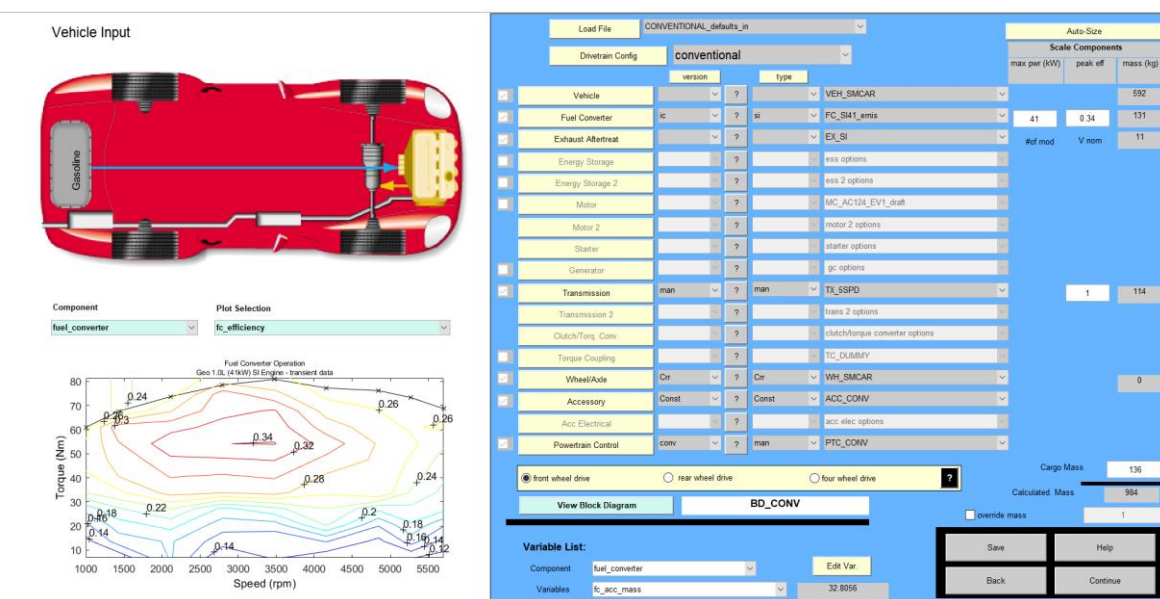


Figura 0.2: Finestra de selecció de característiques del Vehicle a l'Advisor.

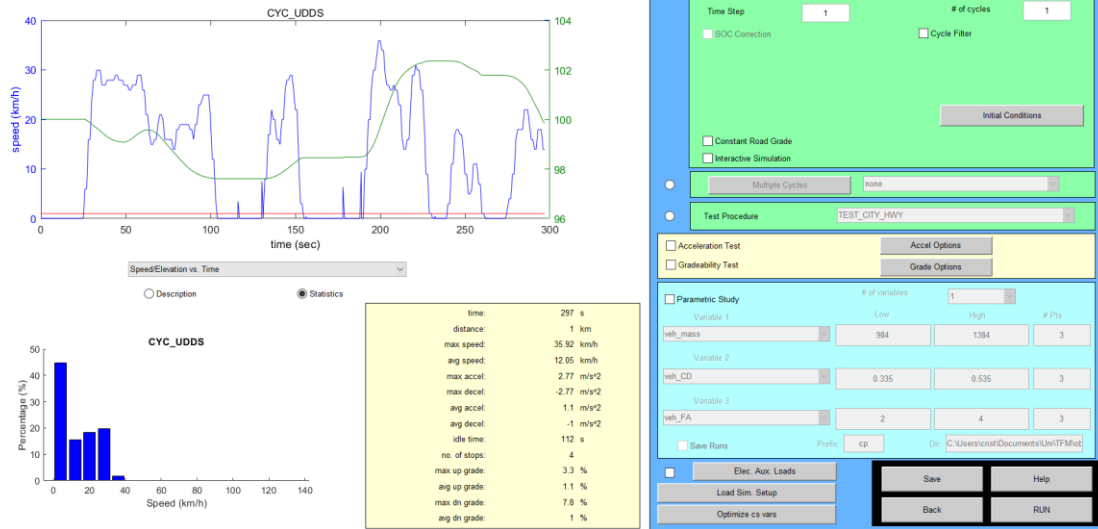


Figura 0.3: Finestra de característiques del Cicle a l'Advisor. En aquest cas està seleccionat el cicle del migdia obtingut a la Gran Via

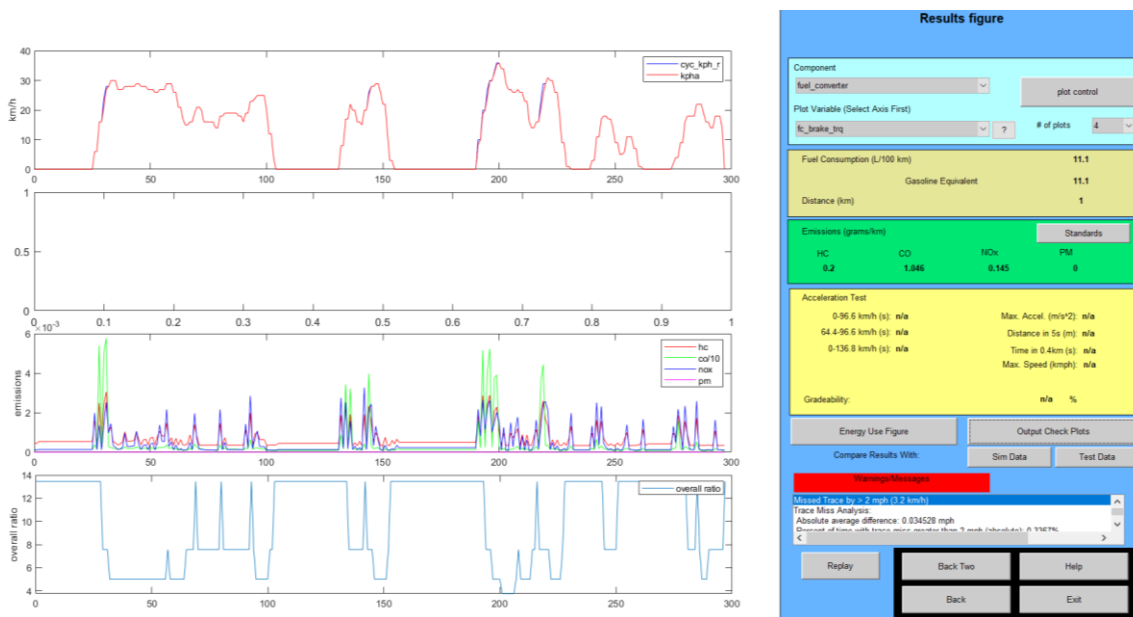


Figura 0.4: Resultats obtinguts de la simulació

A continuació s'exposa el codi de Matlab utilitzat per calcular el pendent i facilitar el tractament.

```
%% Obrir l'arxiu d'excel

filename = 'Dades_obd1309_urgellavroma.xlsx';
[status,sheets,xlFormat] = xlsfinfo(filename);
sheets

%% Escull la fulla que vull

Full10 = readtable(filename,"Sheet","Ruta4");
props = Full10.Properties;
%props.VariableNames
head(Full10)

%%

%Calcul pendent
Alt = Full10.AltitudQGIS;
Alt1= Alt(4:length(Alt));
Velocitat=Full10.Speed_OBD__km_h_;
Velocitat=Velocitat*1000/3600;

%x: Distància en metres integrada de la velocitat
x=cumtrapz(Velocitat);
x1=x(4:length(x));

%diferència d'altituds
a=diff(Alt);

%diferència de distàncies
b=diff(x);

%Càlcul del pendent
Pend = a./b;
Pend1=Pend;

%Filtre pels valors que es divideixen per 0
Pend1(Pend1==Inf|Pend1==-Inf)=0;
Pend2=Pend1;
Pend2(find(isnan(Pend2)))=0;

%Aplicació del filtre
y1=sgolayfilt(Pend3,1,2);
```

```
%%  
%Càlcul del pendent per n=3  
a1=Alt1-Alt(1:length(Alt1));  
b1=x1-x(1:length(x1));  
Pendordre3 = a1./b1;  
Pendordre31=Pendordre3;  
Pendordre31(Pendordre31==Inf|Pendordre31==-Inf)=0;  
Pendordre32=Pendordre31;  
i = (Pendordre32>=1) |(Pendordre32<=-1);  
for z = (1:length(i))  
    indexs = [1:length(Pendordre32)]';  
    if i(z)==1 & i(z+1)==i(z)  
        indexs= indexs(i)-2;  
        Pendordre32(i) = Pendordre32(indexs);  
    else  
        indexs= indexs(i)-1;  
        Pendordre32(i) = Pendordre32(indexs);  
    end  
end  
  
Pendordre32(find(isnan(Pendordre32)))=0;  
Pendordre33=Pendordre32;  
  
y2=sgolayfilt(Pendordre33,1,1);
```

Figura 0.5: Codi Matlab