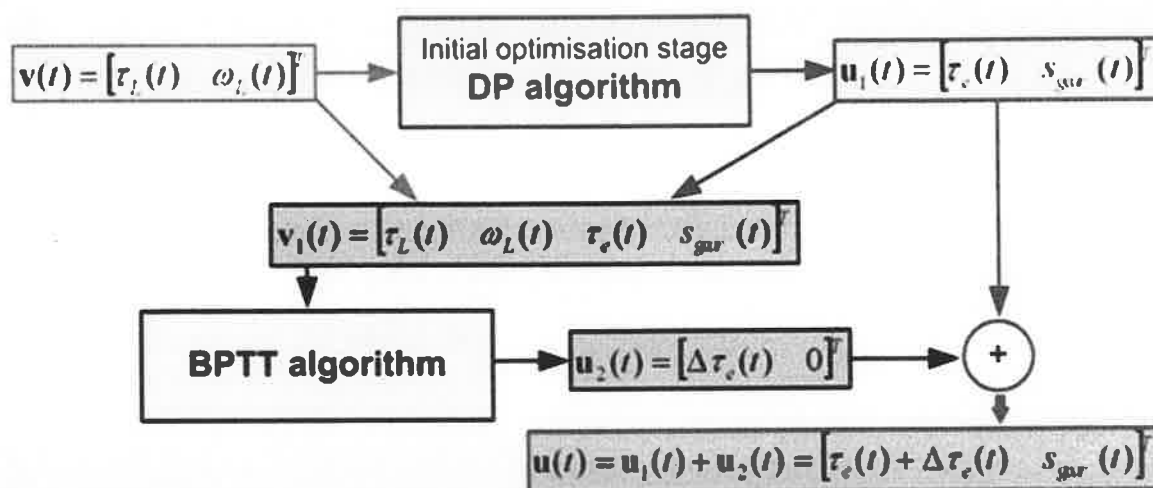


## Računalni algoritam – Kaskadna optimizacija upravljačkih varijabli pogona hibridnog skidera

### Uvod

Ovaj izvještaj prikazuje računalni algoritam optimizacije upravljačkih varijabli hibridnog pogona skidera korištenjem takozvanog postupka kaskadne optimizacije koristeći pojednostavljeni model pogonskog sklopa u kojem je stanje napunjenosti baterije jedina varijabla stanja. Ova tehnika optimizacije sastoji se od algoritma dinamičkog programiranja (DP) za početno (grubo) traženje optimalnog rješenja te konjugirano kompleksnog optimizacijskog algoritma temeljenog na gradijentu (BPTT) za dodatno rafiniranje početnog optimalnog rješenja, a koji se primjenjuju sekvencijalno kako je predloženo u publikacijama [1 – 3].

Kako je prikazano na slici 1, unaprijed poznato opterećenje pogona vozne misije je vektor  $\mathbf{v}(t)$  (moment i brzina vrtnje izlaznog vratila) koji se koristi kao ulaz u DP algoritam kojim se pronalazi optimalan upravljački vektor  $\mathbf{u}_1(t)$  (moment dizel motora, stupanj prijenosa) kojim se minimizira potrošnja goriva. Potom se dobiveni optimalni upravljački vektor i poznato opterećenje pogona koriste kao poznati ulazni vektor  $\mathbf{v}_1(t)$  za BPTT algoritam kojim se pronalazi vektor promjene  $\mathbf{u}_2(t)$  upravljačkog vektora  $\mathbf{u}_1(t)$  i kojom je moguće dodatno minimizirati potrošnju goriva. Nakon provedbe BPTT algoritma, zbroj vektora  $\mathbf{u}_1(t)$  i  $\mathbf{u}_2(t)$  daje konačni optimalan upravljački vektor  $\mathbf{u}(t)$ .



Slika 1. Funkcionalni dijagram kaskadnog optimizacijskog pristupa.

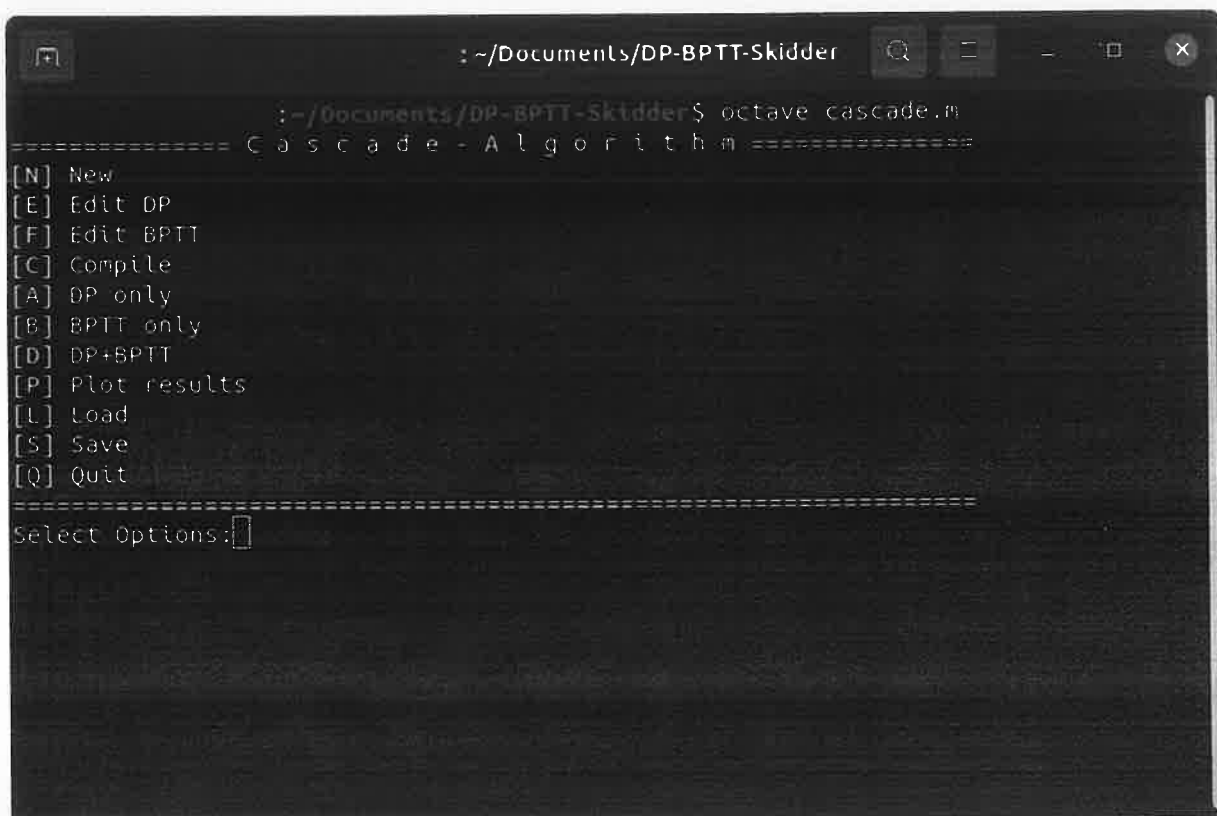
## Realizacija kaskadnog optimizacijskog algoritma

DP i BPTT algoritam napisani su u programskom jeziku C unutar Linux OS okruženja kako bi se iskoristila najnovija inačica GCC kompilatora opremljenog OpenMP API-jem, a koji može koristiti više CPU jezgara za izvršavanje algoritma. OpenMP API primijenjen je samo na prvi dio DP algoritma (koji nije uzročno povezan s vremenom) i rezultira u prosjeku faktorima smanjenja vremena izvršavanja DP algoritma od 1,9, 2,7 i 3,1 puta na Intel 4 Core procesor i7-7500U CPU @ 2,70 GHz (procesor s četiri jezgre) kada se koriste 2, 3 ili 4 programske niti (engl. program threads). Jednostavno korisničko sučelje, predprocesiranje i postprocesiranje te funkcije crtanja razvijeni su u GNU Octave okruženju, dok su rezultati optimizacije također verificirani pomoću modela razvijenog u okruženju Matlab/Simulink.

## Osnovni izbornik

Algoritam se pokreće iz Linux terminala koristeći naredbu: ***octave cascade.m***

Nakon pokretanja otvara se glavni izbornik prikazan na slici 2 pomoću kojeg se može izvršiti odabir pojedine funkcije programa odabirom slova u uglatoj zagradi i pritiskom na tipkovnici.

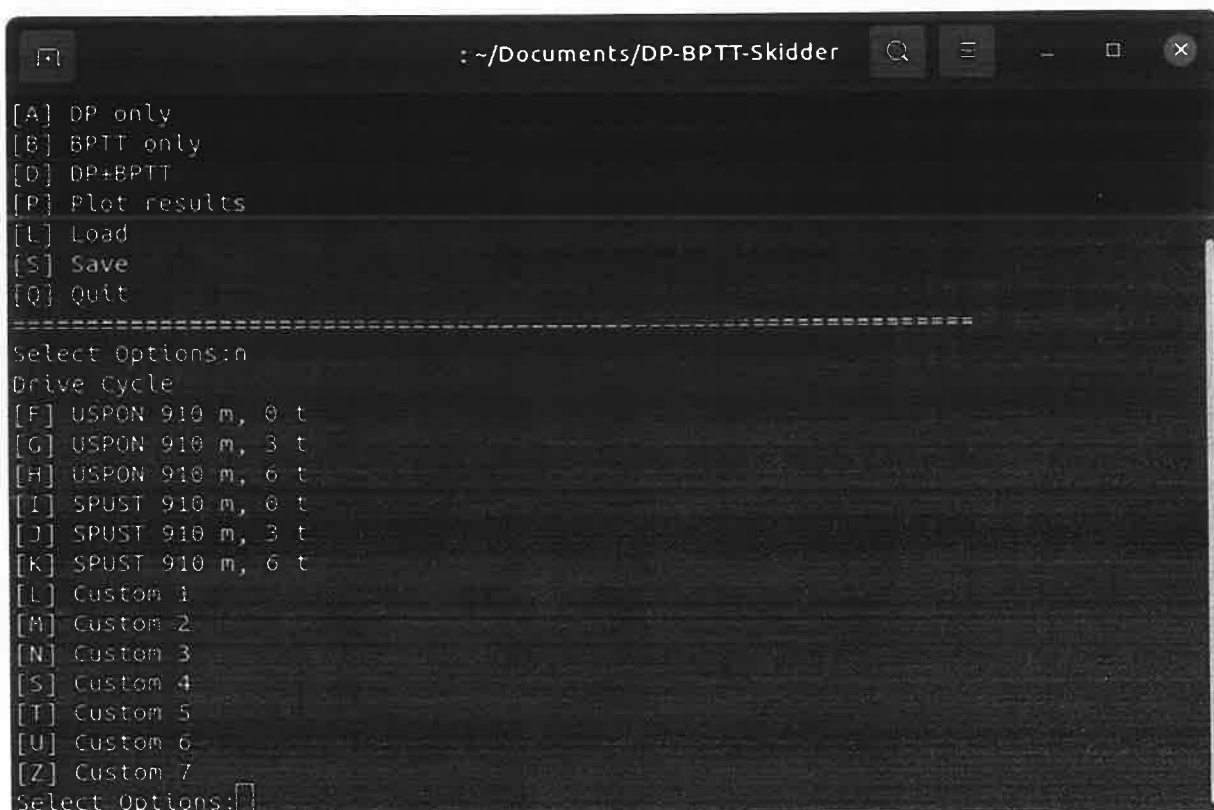
A screenshot of a Linux terminal window. The window title is "::~~/Documents/DP-BPTT-Skidder". The terminal shows the command "octave cascade.m" being executed. The output is a menu titled "==== Cascade - Algorithm =====". The menu items are: [N] New, [E] Edit DP, [F] Edit BPTT, [C] Compile, [A] DP only, [B] BPTT only, [D] DP+BPTT, [P] Plot results, [L] Load, [S] Save, [Q] Quit. Below the menu is a prompt "Select options:" followed by a cursor.

Slika 2. Glavni izbornik.

U nastavku izvještaja su dani kratki opisi pojedinih funkcionalnosti programskog okruženja.

## Odabir - [N] New

Odabir „New” otvara novi izbornik prikazan na slici 3 pomoću kojeg se vrši izbor voznog scenarija. U programu je za skider preddefinirano šest različitih voznih scenarija opterećenja koji su u formi tekstualnih datoteka, a korišteni su za publikacije [2,3]. Odabirom scenarija pokreće se kopiranje karakteristike odabranog voznog scenarija, statičkih karakteristika pojedine komponente pogona (mapa potrošnje goriva, mapa efikasnosti elektromotora, karakteristike baterije i sl...), izvorni kod algoritma i datoteke s parametrima iz arhive u privremenu radnu mapu te se program vraća na glavni izbornik.

A screenshot of a terminal window showing the HiSkid program's main menu. The window title is "~ / Documents / DP-BPTT-Skider". The menu options are listed as follows:

```
[A] DP only
[B] BPTT only
[D] DP+BPTT
[P] Plot results
[L] Load
[S] Save
[Q] Quit
```

A dashed line separates this menu from the next one. The second menu is titled "Select Options:n" and "Drive Cycle". It lists the following options:

```
[F] USPON 910 m, 0 t
[G] USPON 910 m, 3 t
[H] USPON 910 m, 6 t
[I] SPUST 910 m, 0 t
[J] SPUST 910 m, 3 t
[K] SPUST 910 m, 6 t
[L] Custom 1
[M] Custom 2
[N] Custom 3
[S] Custom 4
[T] Custom 5
[U] Custom 6
[Z] Custom 7
```

The prompt "Select Options:" is followed by a cursor.

Slika 3. Izbornik voznog scenarija.

## Odabiri – [E] Edit DP i [F] Edit BPTT

Odabiri „Edit DP” i „Edit BPTT” pokreću program za uređivanje tekstualnih datoteka (tekstualni editor program) te automatski otvara datoteke koje sadrže programski kod pojedinog algoritma te parametre pogona i optimizacije. Na slici 4 je prikazan navedeni tekstualni editor u kojem su otvorene četiri datoteke za DP algoritam te je prikazan dio koda koji se odnosi na datoteku *ths-model.h* koja zajedno s datotekom *indef.m* sadrže parametre optimizacije i parametre pogona hibridnog skidera. Dio izvornog koda glavne i pomoćnih funkcija DP algoritma su prikazani na slikama 5 i 6. Modificiranjem ovih otvorenih datoteka moguće je napraviti izmjene u modelu i parametrima pogona te parametrima optimizacije.

```

thS-model.h
-----
indef.m      thS-model.h      Main.c      DP_func.h

/***** TRANSMISSION - PARAMETERS *****/
/***** Mechanical driveline data *****/
#define r 0.71 // Effective wheel radius [m]
#define hpg 2.24 // Planetary gear ratio
#define to 2.16 // Bevel gear and differential ratio
/* ovo iznad se za skidera ne koristi *****/
/*****SKIDER*****/

/***** Battery data *****/
#define Qbat 57240.0 // Full charge cell capacity 15,9 Ah (57240 As)
#define ncel 250.0 // Number of cells
// #define Ric 0.056128 // Battery internal resistance of charging (from map)
// #define Ridc 0.115904 // Battery internal resistance of discharging (from map)
/*****

#define weler 105.00 // Brzina vrtnje ICE u leru
// #define hevflag 1 // 1 Hibrid, 0 Conv

#define hg1 213.0 //skider (213, 116, 70, 44, 29) - volt (30.720, 18.350, 6.9984, 3.1200, 1.0000)
#define hg2 116.0 //volt paralelni (30.72, 18.350, 6.9984, 3.1200, 1.0000)
#define hg3 70.0
#define hg4 44.0
#define hg5 29.0

/***** M A P E *****/
#define deltat 1 // Sample-time
    
```

Slika 4. Datoteka s pogonskim parametrima u tekstualnom editoru.

```

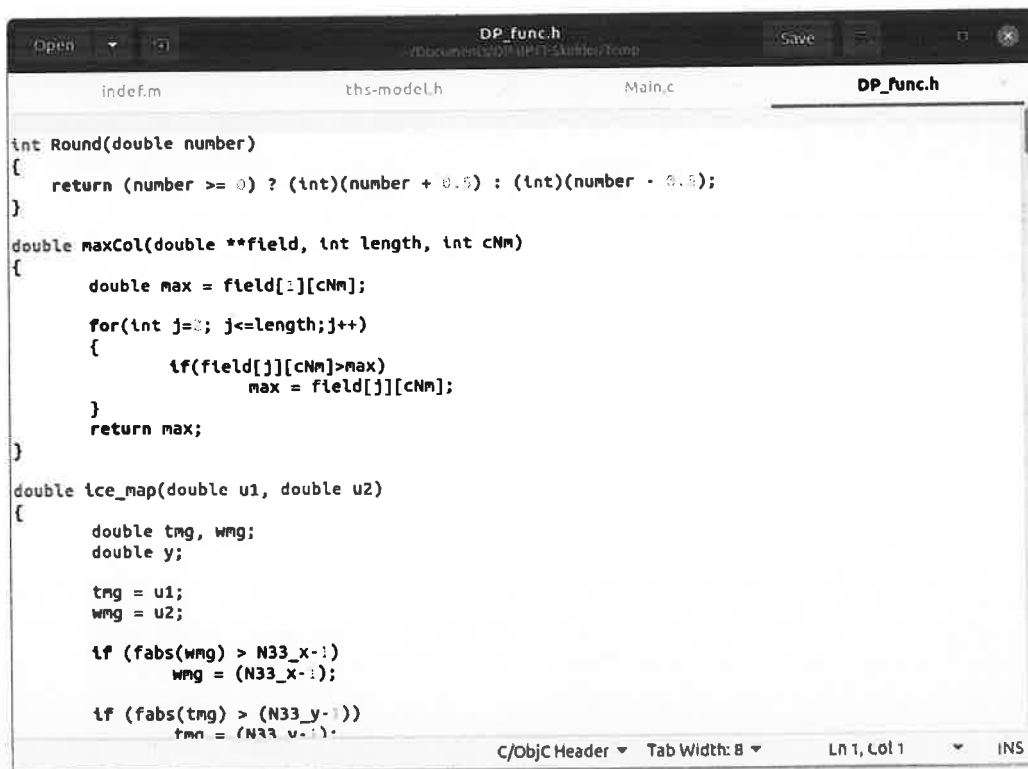
Main.c
-----
indef.m      thS-model.h      Main.c      DP_func.h

#include <stdio.h>
#include <math.h>
#include <omp.h>
#include "GF_Nrutil.h"
#include "GF_Matrix.h"
#include "GF_Data.h"
#include "GF_Mape.h"
#include "Lsdef.h"
#include "Nsdef.h"
#include "thS-model.h"
#include "DP_func.h"
#define Ndat 5
double MaxTue, MaxTmg1;

void main(int argc, char *argv[])
{
    double *t,*x, *u1, *u2, *u3, **U1opt,**U2opt, **U3opt, **Fopt, **Fopt1, *uu1, *uu2, *uu3,
    *x1;
    double Jpen, x_t, x_t1, index_x1, lxl, fFopt, cFopt, aaa, Fopt_interpol, bbb, index_x0;
    double li, ufl, uce, aa, bb, index_xk, pom, min = 1000000;
    int lfxl, lcxl, lfl, lce, lfor, Ncp1, Ncp2, Ncp3, Ncpx, Ncpt;
    int ln1, ln2, ln3, l, j, k, l, j1, j2, j3;
    double t0, tf, x_min, x_max, u1_min, u1_max, u2_min, u2_max, u3_min, u3_max, Dt, Dx, Du1,
    Du2, Du3, x_fin, x0;

    t0 = 0;
    tf = term_time;
    x_min = soc_min;
    x_max = soc_max;
    u1_min = 0;
    // ...
    
```

Slika 5. Datoteka izvornog koda glavne funkcije DP algoritma u tekstualnom editoru.



```

int Round(double number)
{
    return (number >= 0) ? (int)(number + 0.5) : (int)(number - 0.5);
}

double maxCol(double **field, int length, int cNm)
{
    double max = field[0][cNm];
    for(int j=0; j<=length;j++)
    {
        if(field[j][cNm]>max)
            max = field[j][cNm];
    }
    return max;
}

double ice_map(double u1, double u2)
{
    double tmg, wmg;
    double y;

    tmg = u1;
    wmg = u2;

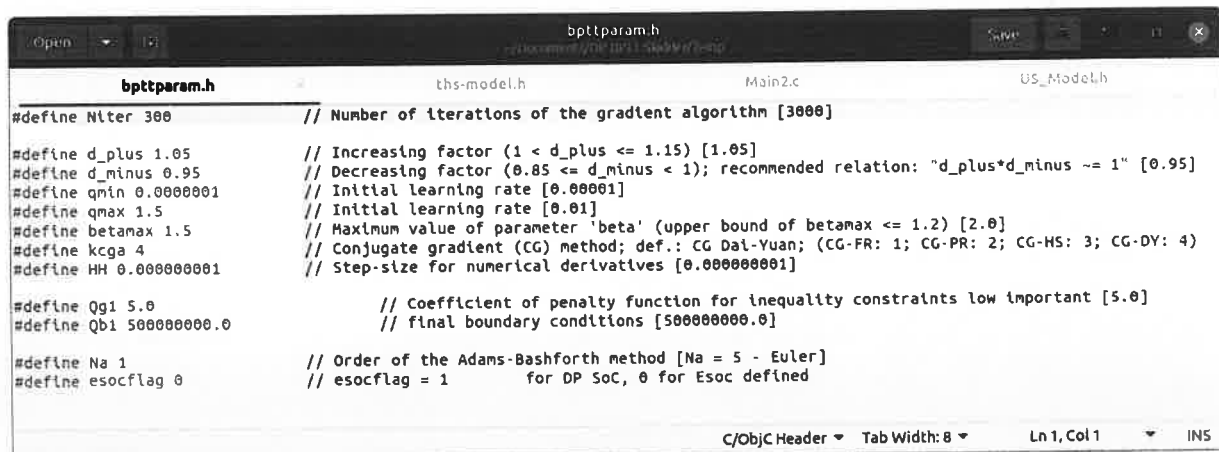
    if (fabs(wmg) > N33_x-1)
        wmg = (N33_x-1);

    if (fabs(tmg) > (N33_y-1))
        tmg = (N33_y-1);
}

```

Slika 6. Datoteka izvornog koda pomoćnih funkcija DP algoritma u tekstualnom editoru.

Slično je u slučaju odabira „Edit BPTT“, također se otvaraju četiri datoteke u tekstualnom editoru. Na slici 7 prikazana je otvorena datoteka *bpttparam.h* koja sadrži parametre optimizacije BPTT algoritma. Datoteka *ths-model.h* sadrži parametre pogona te se koristi se u oba algoritma stoga je prikazana na slici 4. Dio izvornog koda glavne i pomoćnih funkcija BPTT algoritma su prikazani na slikama 8 i 9, te je također modificiranjem ovih datoteka moguće provesti izmjene u parametrima pogona i parametrima optimizacije.



```

#define NIter 300 // Number of iterations of the gradient algorithm [3000]

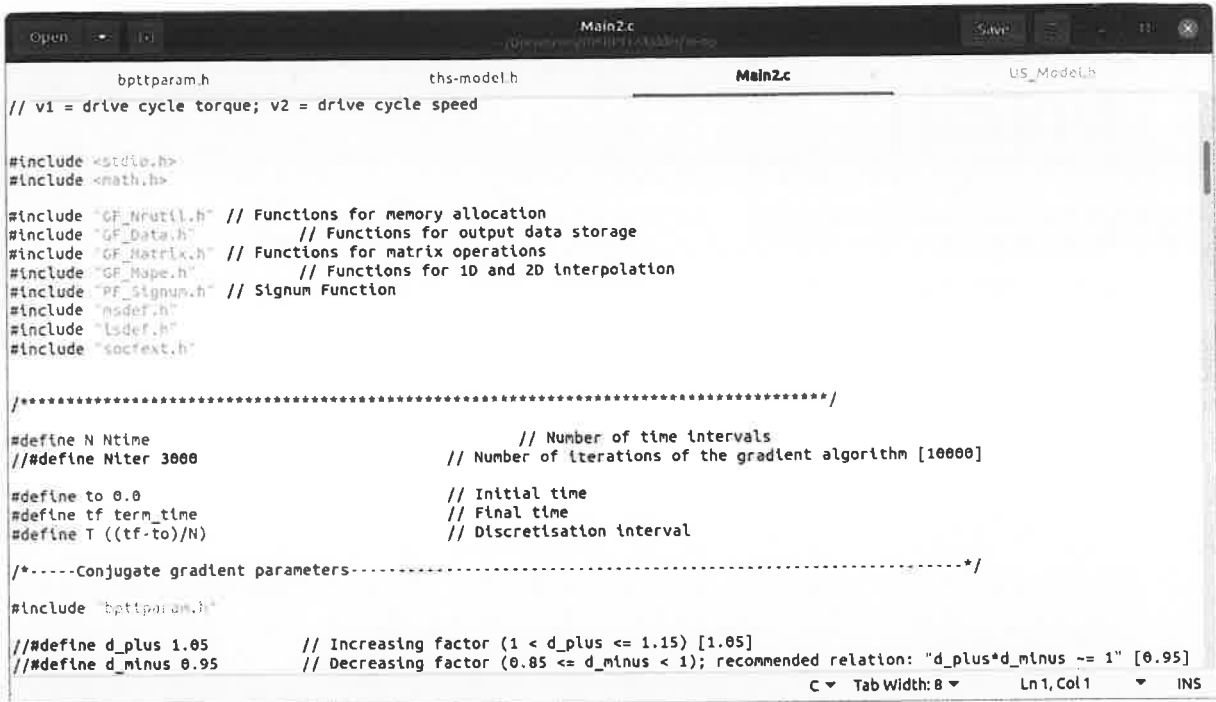
#define d_plus 1.05 // Increasing factor (1 < d_plus <= 1.15) [1.05]
#define d_minus 0.95 // Decreasing factor (0.85 <= d_minus < 1); recommended relation: "d_plus*d_minus == 1" [0.95]
#define qmin 0.0000001 // Initial learning rate [0.00001]
#define qmax 1.5 // Initial learning rate [0.01]
#define betamax 1.5 // Maximum value of parameter 'beta' (upper bound of betamax <= 1.2) [2.0]
#define kcg 4 // Conjugate gradient (CG) method; def.: CG Dai-Yuan; (CG-FR: 1; CG-PR: 2; CG-HS: 3; CG-DY: 4)
#define HH 0.000000001 // Step-size for numerical derivatives [0.000000001]

#define Qg1 5.0 // Coefficient of penalty function for inequality constraints low important [5.0]
#define Qb1 500000000.0 // final boundary conditions [500000000.0]

#define Na 1 // Order of the Adams-Bashforth method [Na = 5 - Euler]
#define esocflag 0 // esocflag = 1 for DP SoC, 0 for Esoc defined

```

Slika 7. Datoteka s parametrima optimizacije BPTT algoritma u tekstualnom editoru.



```

Open  Main2.c
-----
bpttparam.h  ths-model.h  Main2.c  US_Model.h

// v1 = drive cycle torque; v2 = drive cycle speed

#include <stdio.h>
#include <math.h>

#include "GF_Nrutil.h" // Functions for memory allocation
#include "GF_Data.h" // Functions for output data storage
#include "GF_Matrix.h" // Functions for matrix operations
#include "GF_Mape.h" // Functions for 1D and 2D interpolation
#include "PF_Signum.h" // Signum Function
#include "nsdef.h"
#include "lsdef.h"
#include "sacext.h"

/*****/

#define N Ntime // Number of time intervals
// #define Niter 3000 // Number of iterations of the gradient algorithm [10000]

#define to 0.0 // Initial time
#define tf term_time // Final time
#define T ((tf-to)/N) // Discretisation interval

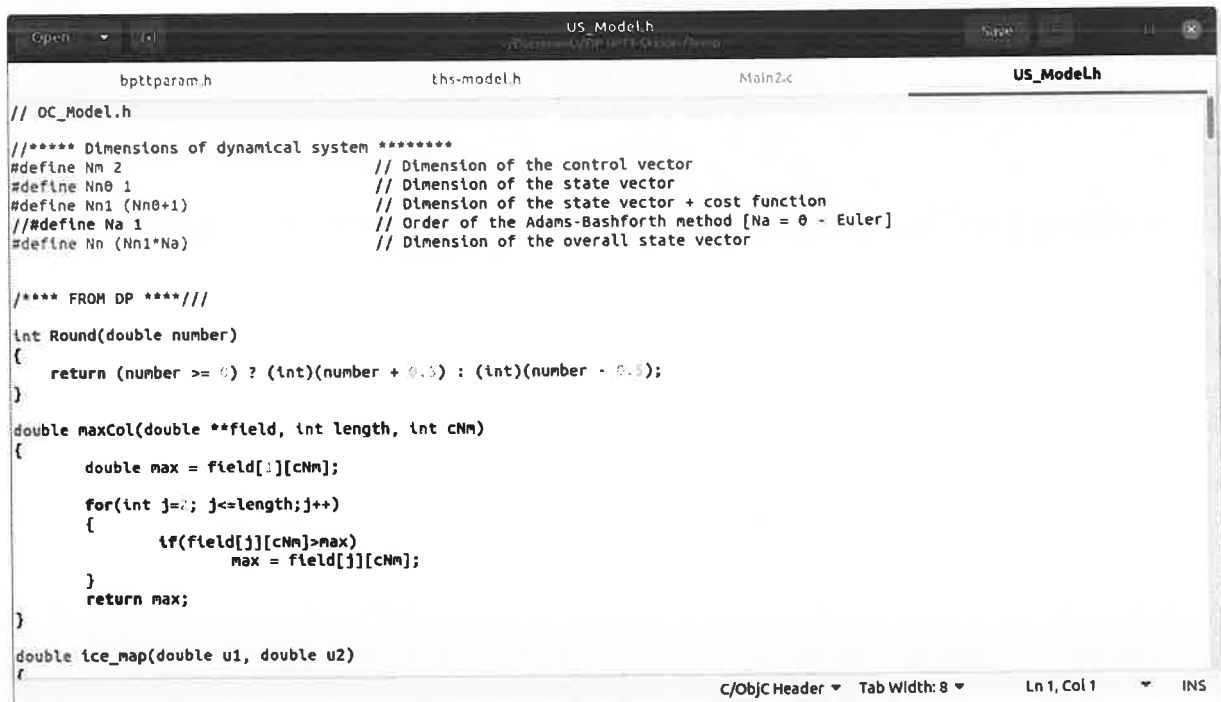
/*-----Conjugate gradient parameters-----*/

#include "bpttparam.h"

// #define d_plus 1.05 // Increasing factor (1 < d_plus <= 1.15) [1.05]
// #define d_minus 0.95 // Decreasing factor (0.85 <= d_minus < 1); recommended relation: "d_plus*d_minus == 1" [0.95]

```

Slika 8. Datoteka izvornog koda glavne funkcije BPTT algoritma u tekstualnom editoru.



```

Open  US_Model.h
-----
bpttparam.h  ths-model.h  Main2.c  US_Model.h

// OC_Model.h

/***** Dimensions of dynamical system *****/
#define Nm 2 // Dimension of the control vector
#define Nn0 1 // Dimension of the state vector
#define Nn1 (Nn0+1) // Dimension of the state vector + cost function
// #define Na 1 // Order of the Adams-Bashforth method [Na = 0 - Euler]
#define Nn (Nn1*Na) // Dimension of the overall state vector

/**** FROM DP ****/

int Round(double number)
{
    return (number >= 0) ? (int)(number + 0.5) : (int)(number - 0.5);
}

double maxCol(double **field, int length, int cNm)
{
    double max = field[0][cNm];
    for(int j=0; j<length;j++)
    {
        if(field[j][cNm]>max)
            max = field[j][cNm];
    }
    return max;
}

double ice_map(double u1, double u2)
{

```

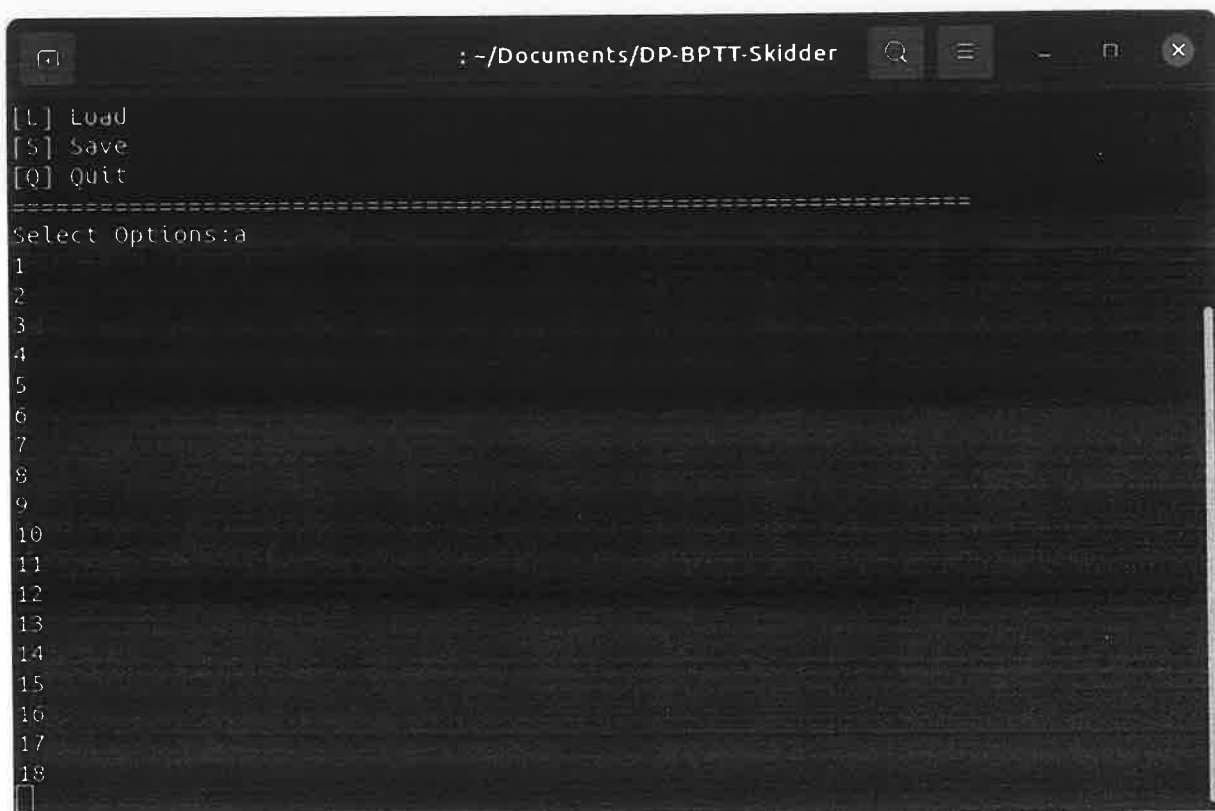
Slika 9. Datoteka izvornog koda pomoćnih funkcija BPTT algoritma u tekstualnom editoru.

### Odabir - [C] Compile

S odabirom „Compile“ opcije kompilira se program DP i BPTT algoritma (prevodi se izvorni kod napisan u programskom jeziku razumljivom čovjeku u strojni kod razumljiv računalu) te se priprema za izvršenje. Kompiliranje je potrebno provesti uvijek nakon prvog učitavanja koristeći opciju „New“ ili ako se provelo uređivanje koda pomoću tekst editora.

### Odabir - [A] DP Only

Odabirom „DP Only“ pokreće se izvršavanje optimiranja upravljačkih varijabli samo pomoću algoritma DP. Ako nema pogreške u računalnom kodu, program započinje te se prikazuju numeričke vrijednosti (slika 10) koje odgovaraju vremenskom trenutku voznog scenarija u sekundama. Trajanje izvođenja programa ovisi o trajanju voznog ciklusa te odabranoj gustoći točaka (rezoluciji) koje su definirane u parametrima optimizacije. Ako se odabere veća gustoća točaka dobiveno optimalno rješenje je preciznije, ali time značajno produžuje vrijeme izvođenja programa.

A screenshot of a terminal window with a dark background. The window title is ":-/Documents/DP-BPTT-Skider". The terminal content shows a menu with options: [L] Load, [S] Save, [Q] Quit. Below this is a dashed line and the prompt "Select Options:a". A list of numbers from 1 to 18 is displayed, with a cursor at the bottom of the list. The numbers are: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18.

```
:-/Documents/DP-BPTT-Skider
[L] Load
[S] Save
[Q] Quit
-----
Select Options:a
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
█
```

Slika 10. Izvršavanje optimiranja DP algoritmom

### Odabir - [B] BPTT Only

Odabir „BPTT Only“ pokreće izvršavanje optimiranja upravljačkih varijabli samo pomoću algoritma BPTT. U tom slučaju na ekranu se pojavi prikaz numeričkih vrijednosti (Slika 11) gdje je u prvoj koloni redni broj iteracije algoritma, zatim je dana vrijednost ciljne funkcije (koja treba konvergirati) te su na kraju dane vrijednosti faktora gradijenta.



```
231 J=1167.869887 beta=0.684224 eta=0.008240
232 J=1167.869623 beta=1.500000 eta=0.008652
233 J=1167.869215 beta=0.794384 eta=0.009084
234 J=1167.868868 beta=1.481960 eta=0.009539
235 J=1167.868339 beta=0.831367 eta=0.010016
236 J=1167.867869 beta=1.500000 eta=0.010516
237 J=1167.867141 beta=0.883939 eta=0.011042
238 J=1167.866450 beta=1.500000 eta=0.011594
239 J=1167.865365 beta=1.078864 eta=0.012174
240 J=1167.864126 beta=1.388286 eta=0.012783
241 J=1167.862325 beta=1.226610 eta=0.013422
242 J=1167.860011 beta=1.221576 eta=0.014093
243 J=1167.856945 beta=1.500000 eta=0.014798
244 J=1167.851829 beta=1.500000 eta=0.015537
245 J=1167.842135 beta=1.500000 eta=0.016314
246 J=1167.823878 beta=1.500000 eta=0.017130
247 J=1167.794933 beta=0.174247 eta=0.017986
248 J=1167.789477 beta=1.500000 eta=0.018886
249 J=1167.779912 beta=1.500000 eta=0.019830
250 J=1167.764731 beta=0.519857 eta=0.020822
251 J=1167.756423 beta=0.546051 eta=0.021863
252 J=1167.751707 beta=0.332533 eta=0.022956
253 J=1167.750047 beta=0.581403 eta=0.024104
```

Slika 11. Izvršavanje optimiranja BPTT algoritmom

Za razliku od DP, izvođenje optimiranja primjenom BPTT algoritmom je brzo i njegovo trajanje ne ovisi toliko o parametrima optimizacije kao kod DP algoritma. Međutim, pogrešan izbor parametara može dovesti do problema s konvergencijom.

### Odabir - [D] DP+BPTT

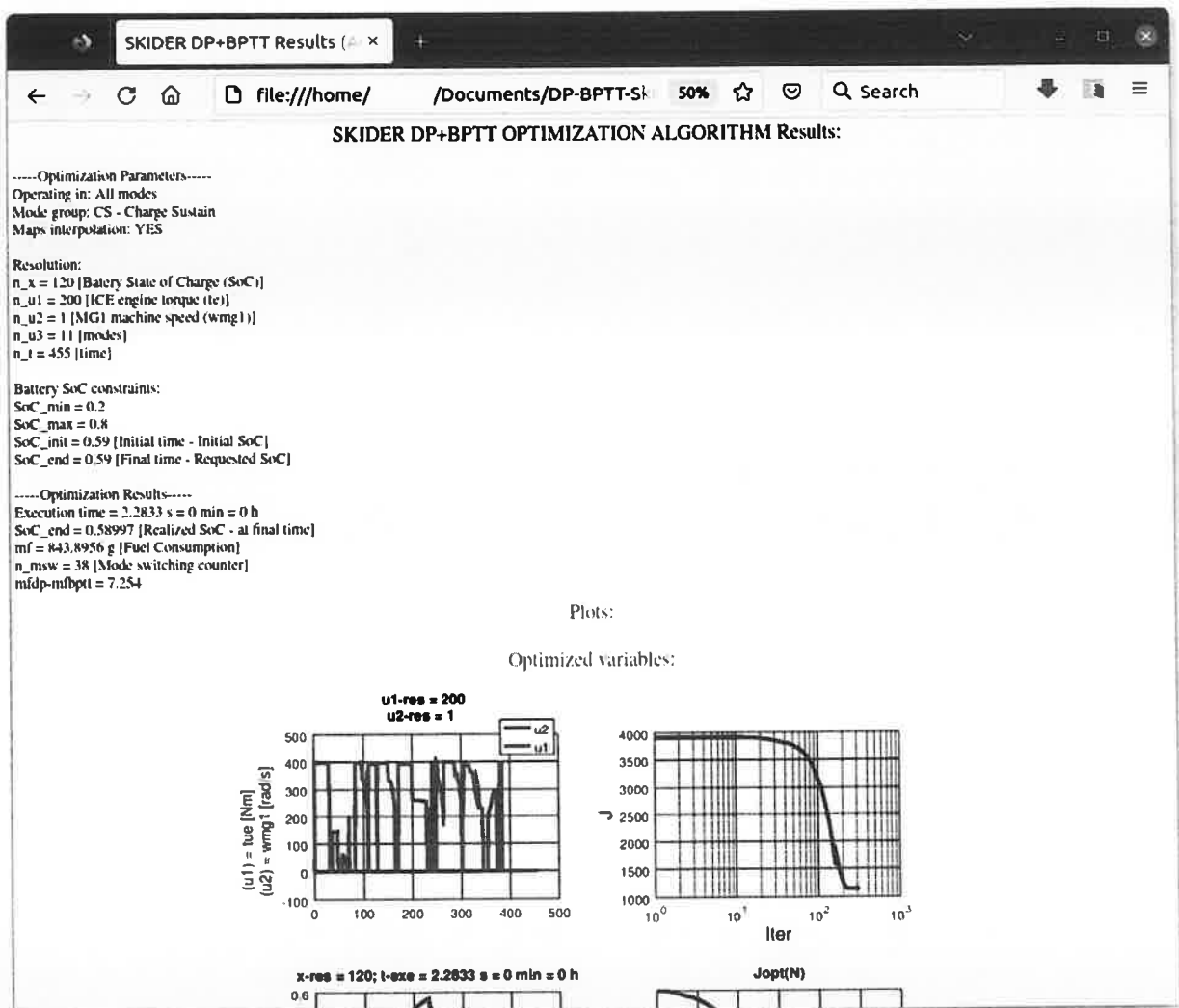
Odabirom „DP+BPTT“ pokreće se izvršavanje optimiranja upravljačkih varijabli pomoću kaskadnog postupka. Najprije se provodi optimiranje pomoću DP algoritma, a zatim se automatski nastavlja optimiranje pomoću BPTT algoritma. Za razliku od DP algoritma, BPTT algoritam je temeljen na gradijentu te su varijable kontinuirane po amplitudi, a također se može koristiti integracijska metoda višeg reda, čime se dodatno može povećati preciznost dobivenog optimalnog rješenja. Takva kaskadna optimizacija ima nekoliko prednosti u odnosu na pristup koji koristi samo DP. Optimalno rješenje se može poboljšati povećanjem numeričke preciznosti



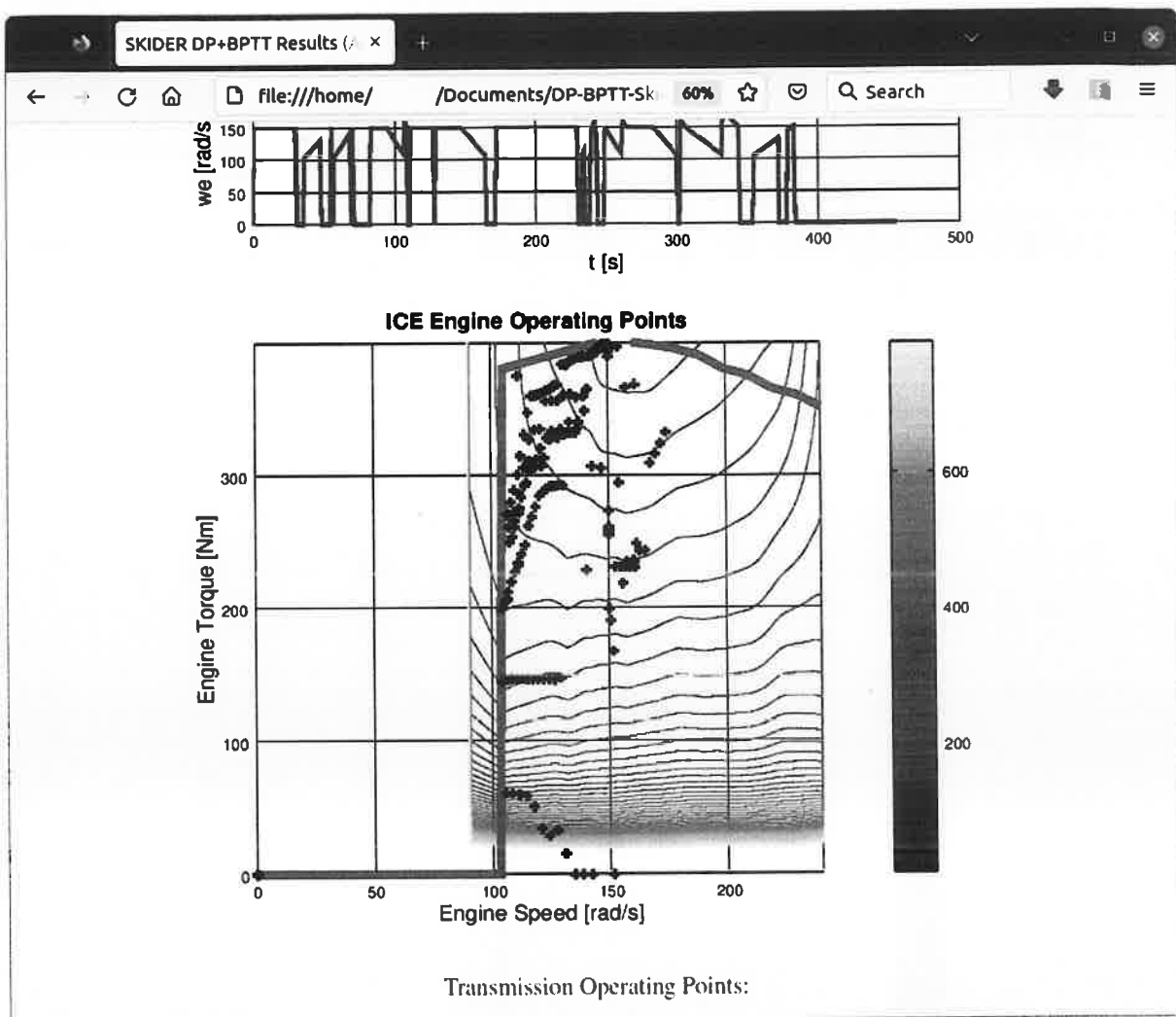
gradijentnog algoritma, a gotovo optimalno rješenje moguće je pronaći u bitno kraćem vremenu primjenom dinamičkog programiranja s niskom gustoćom mreže, a čiji se rezultati naknadno doraduju pomoću gradijentnog algoritma. Sve u svemu, ovo čini pristup kaskadne optimizacije vrlo obećavajućim alatom za studije optimizacije koje zahtijevaju visoku preciznost i ograničeno vrijeme optimizacije.

## Odabir - [P] Plot results

Pomoću opcije „Plot results“ mogu se prikazati rezultati optimiranja. Funkcija iscrtava vremenske dijagrame pojedine varijable te generira dokument *results.html* kojeg je moguće otvoriti Internetskim preglednikom (Slika 12). Na slici 13 kao primjer su prikazane radne točke motora s unutarnjim izgaranjem.



Slika 12. Rezultati optimiranja otvoreni Internetskim preglednikom



Slika 13. Rezultati optimiranja – radne točke motora s unutarnjim izgaranjem

Na kraju dokumenta nalaze se još i poveznice na tekstualne datoteke koje sadrže vrijednosti pojedine pogonske varijable u vremenu i koje se naknadno mogu koristiti u drugim računalnim programima.

### Odabiri – [L] Load i [S] Save

Odabir opcije „Save“ omogućuje arhiviranje trenutne radne mape za budući rad, a opcija „Load“ omogućuje učitavanje iz radne arhive. Kod korištenja opcije potrebno je točno navesti ime arhive.


**Literatura:**

1. Cipek, Mihael; Kasać, Josip; Pavković, Danijel; Zorc, Davor: A novel cascade approach to control variables optimisation for advanced series-parallel hybrid electric vehicle power-train. Applied energy, 276 (2020), 115488, 12
2. Karlušić, Juraj; Cipek, Mihael; Pavković, Danijel; Šitum, Željko; Benić, Juraj; Šušnjar, Marijan: Benefit Assessment of Skidder Power-train Hybridization utilizing a novel Cascade Optimization Algorithm. Sustainability, 12 (2020), 24; 10396, 15
3. Karlušić, Juraj; Cipek, Mihael; Pavković, Danijel; Šitum, Željko; Benić, Juraj: Efficiency comparison of different powertrain structures intended for a hybrid skidder by utilizing a novel cascade optimization algorithm. e-Prime - Advances in Electrical Engineering, Electronics and Energy, 2 (2022), 100079, 11

Datum: 8.12.2022.

Za izvješće: doc. dr. sc. Mihael Cipek

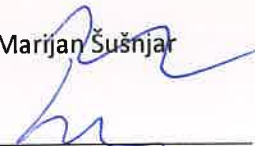
prof. dr. sc. Željko Šitum

A handwritten signature in blue ink, appearing to read "Ž. Šitum".

---

voditelj projektnog tima  
Fakultet strojarstva i brodogradnje  
Sveučilišta u Zagrebu

prof. dr. sc. Marijan Šušnjar

A handwritten signature in blue ink, appearing to read "M. Šušnjar".

---

voditelj projektnog tima  
Fakultet šumarstva i drvne tehnologije  
Sveučilišta u Zagrebu