

Zpráva o realizaci vybrané varianty nadřazeného řízení v dopravě

Jitka Homolová, Ivan Nagy

UTIA AV ČR

1 Úvod

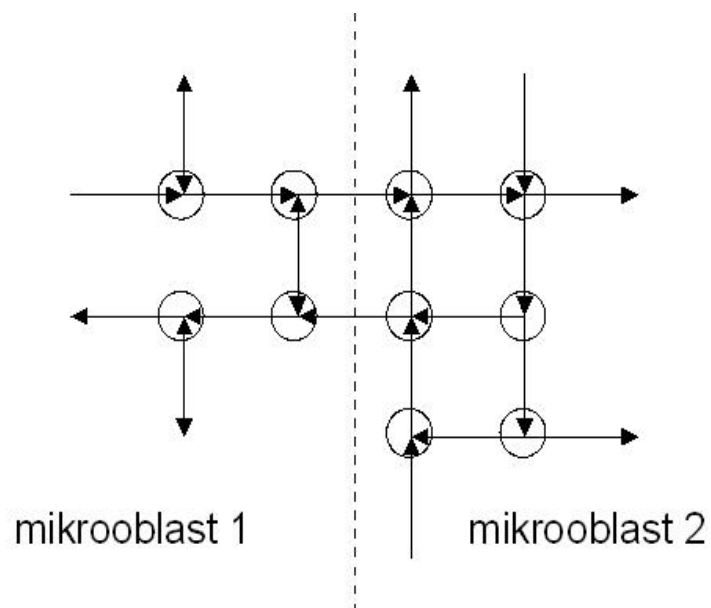
Tato zpráva navazuje na zprávu UTIA č. 2131, J. Homolová, I. Nagy: Nadřazená úroveň hierarchického regulátoru, červen 2005, ve které jsou popsány vytipované strategie nadřazené úrovně hierarchického řízení. Vybírá nejúspěšnější strategii, minimalizující rozdíly mezi množstvím aut v jednotlivých mikrooblastech, a ukazuje výsledky experimentů, testujících vlastnosti této strategie.

2 Popis vybrané strategie

Vybranou strategií je strategie 4.1 ze zmíněné zprávy. Model pro realizaci této strategie je analogií k modelu mikrooblasti. Modelovanou veličinou (stavem) jsou délky kolon - počty automobilů akumulovaných v jednotlivých mikrooblastech. Cílem řízení je udržovat rovnováhu v zaplnění jednotlivých mikrooblastí, tedy udržovat množství automobilů v mikrooblasti úměrné její velikosti. Tento požadavek lze prakticky formulovat jako minimalizaci absolutní hodnoty váženého rozdílu "délky kolon" v mikrooblastech. Ve zprávě č. 2131 je dále popsáno i zobecnění této strategie na libovolný počet mikrooblastí.

3 Ověření funkčnosti navrhované strategie

Funkci navrhované strategie ověříme na následujícím příkladě. Uvažujeme dopravní oblast tvořenou 10 křižovatkami a rozdělenou na dvě mikrooblasti tak, že první mikrooblast obsahuje 4 křižovatky a druhá 6 křižovatek. Schema oblasti je naznačeno na obrázku.

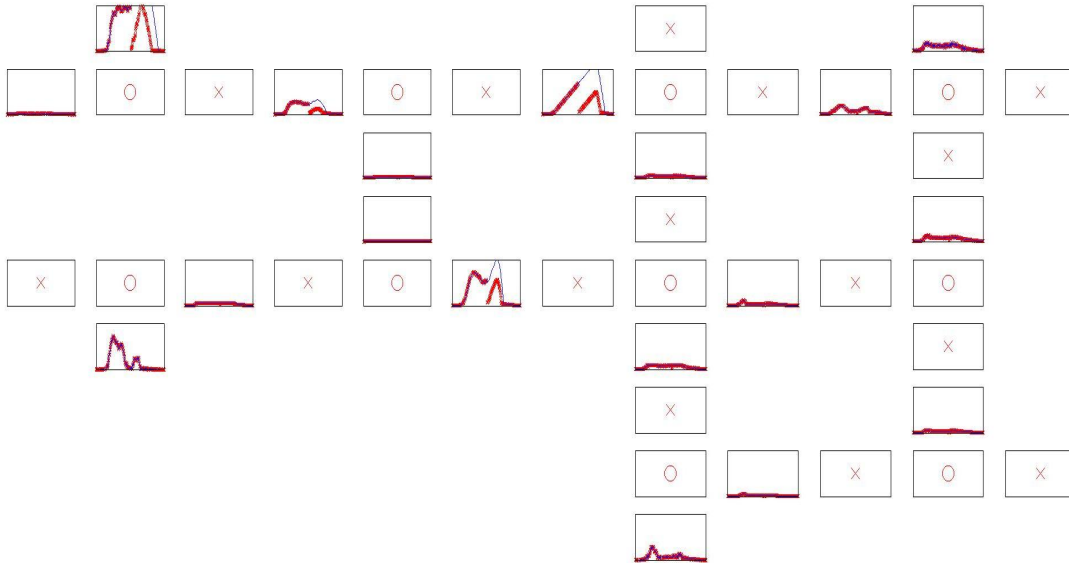


Obrázek 1: Schéma oblasti tvořené dvěma mikrooblastmi

Na obrázku jsou ramena křižovatek značena šipkami, vlastní křižovatky jsou průsečíky šipek. Šipky značí tok vozidel; je-li rameno obousměrné, má šipku na začátku i na konci.

Při experimentu byly na vstupní ramena přiváděny reálné intenzity dopravního proudu, získané měřeními na detektorech v oblasti Smíchova (ulice Zborovská a Svornosti a ulice s nimi se křižující). Pomocí simulace byly pak k daným časovým průběhům zelených počítány délky kolon v jednotlivých ramenech.

Experiment č. 1 V tomto experimentu byly zelené nastaveny ručně a zůstaly konstantní po celou dobu experimentu. Korekce odhadu kolon zpětnou vazbou od měřených výstupů byla vypnuta. Výsledek je na Obrázku 2.

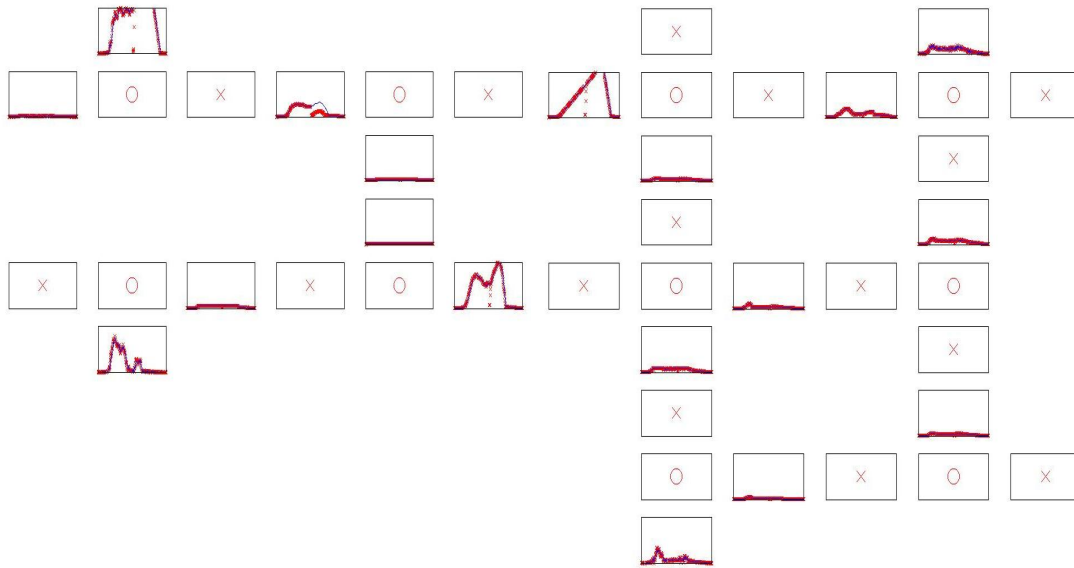


Obrázek 2: Experiment č. 1

Na Obrázku 2 jsou průběhy kolon schematicky vyznačeny tak, že každý malý obrázek se skutečnou kolonou (tenká čára) a odhadem kolony (silná čára) představuje jedno rameno křižovatky. Vlastní křižovatka je označena kroužkem, rameno bez vjezdu - a tedy bez kolony - je označeno křížkem.

Při popisu výsledků budeme sledovat levý horní roh oblasti (jinde je situace obdobná). Vidíme, že vodorovný směr je zbytečně příliš preferován, zatímco ve svislém směru se tvoří poměrně velká kolona. Navíc, abychom konečné řízení učinili obtížnější a více se přiblížili realitě, udělali jsme přibližně uprostřed průběhu dosti vážnou poruchu - vynulovali jsme všechny kolony. Protože model kolon má "integrační" charakter, není schopen bez zpětné vazby tuto poruchu kompenzovat. Odhad sleduje sice dále průběh skutečné kolony ale s chybou, která zmizí až po vymizení kolony, tj. při novém odhadu v dalším dni.

Experiment č. 2 Při tomto experimentu byla zapnuta korekce odhadu stavů od měřených výstupů.

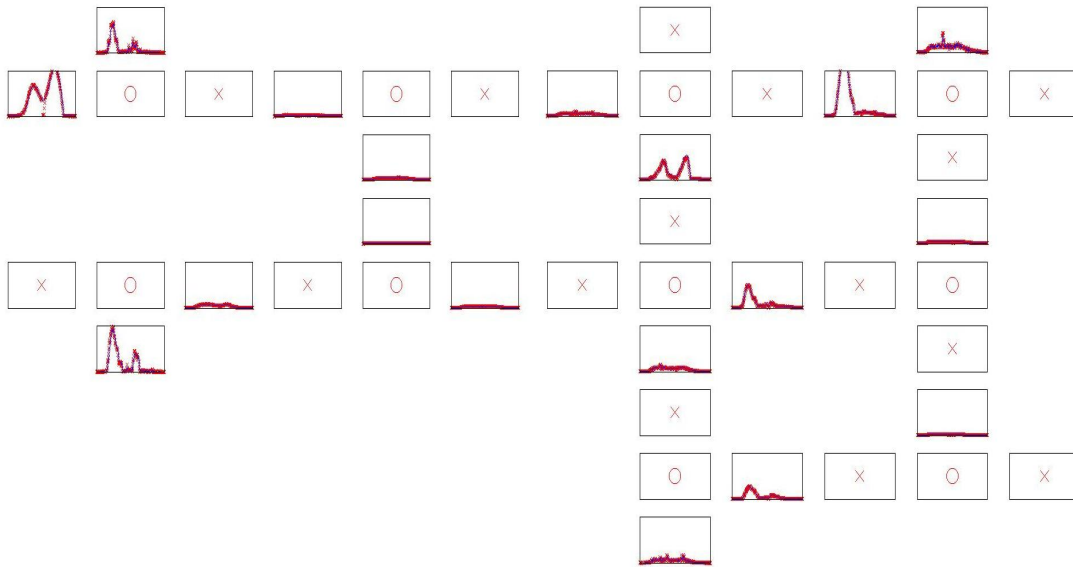


Obrázek 3: Experiment č. 2

Z obrázku je patrné, že ke korekci poruchy dochází okamžitě a není třeba čekat do dalšího dne.

Pro přesnější zhodnocení efektu řízení jsme při každém experimentu počítali numerické kritérium - průměrnou hodnotu součtu všech kolon (PSK). Pro experiment 1 i 2 byla tato hodnota $PSK = 367.102$.

Experiment č. 3 Při experimentu 3 bylo zapnuto ještě lokální řízení obou mikroblastů.

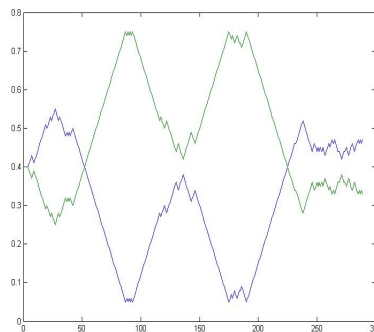


Obrázek 4: Experiment č. 3

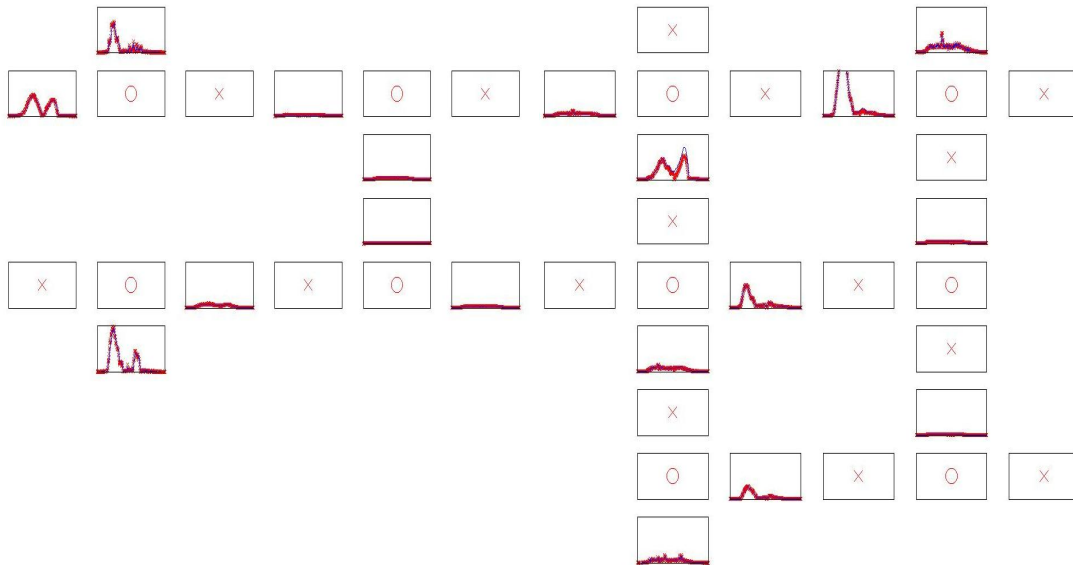
Na obrázku je vidět, že délky kolon ve vodorovném i svislém směru se srovnaly, což bylo cílem řízení.

Hodnota koeficientu PSK klesla na 217.872, tedy celkově se kolony dosti zkrátily.

Pro ukázkou ještě uvádíme průběh zelené, generované lokálním regulátorem pro levou horní křižovatku



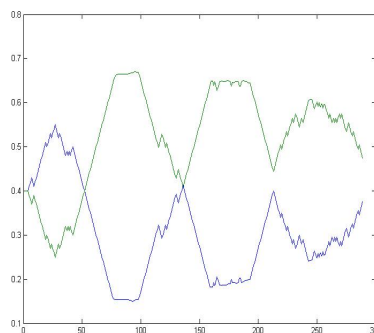
Experiment č. 4 Při posledním experimentu bylo ještě přidáno nadřazené řízení.



Obrázek 5: Experiment č. 4

Prostým srovnáním Obr. 4 a 5 příliš nepoznáme. Hodnota koeficientu průměrné délky front $PSK = 193.331$ však ukazuje další zlepšení - i když ne tak výrazné, jako při aktivaci lokálního řízení. Důvod je jasný. Lokální řízení zajišťuje minimální kolony. Nadřazené řízení má vyšší cíle, proto by mohlo kritérium délek kolon dokonce i trochu zhoršit.

Zde jsou ještě průběhy zelených na levé horní křižovatce generované lokálním regulátorem, který respektoval omezení, zadaná nadřazeným regulátorem.



4 Závěr

Vybraná strategie nadřazeného řízení je účinná a může se stát základem pro konstrukci horní vrstvy hierarchického regulátoru dopravy. V případě potřeby mohou být vzaty v úvahu i další strategie, popsané ve zprávě č. 2131.

Dodatek

V tomto dodatku jsou uvedeny hlavní programy systému MATLAB, které realizují nadřazené řízení. Ostatní programy byly uvedeny již v předchozích zprávách.

Hlavní program

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%                               J. Homolová, I. Nagy                               %
%      Simulace dvou mikrooblastí s lokálním řízením                          %
%                               a nadřazeným regulátorem                          %
%                               --- 29.10.2005 ---                               %
% ----- %
% Model je založen na bilanci příjezdů a odjezdů                               %
% a vztahu délka kolony - obsazenost.                                         %
% Odhad se provádí pomocí lineárního Kalmanova filtru.                         %
% Pro řízení je použito lineární programování.                                 %
% Nadřazený regulátor zadává okno pro lokální regulátory. %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

clc, clear all

% Základní klíče pro nastavení běhu
control=1; % s lokálním řízením: 1 = ano; 0 = ne
kalfilt=1; % s Kalmanovým filtrem 1 = ano (ep=yt-yp); 0 = ne (ep=
nadcont=0; % s nadřazeným regulátorem: 1 = ano; 0 = ne

% Základní parametry pro nastavení
n1=901; % začátek reálných dat
k=1; % počet dní pro simulaci
% počáteční zelené
z1{1}=[.4 .4]'; z1{2}=[.4 .4]'; z1{3}=[.4 .4]'; z1{4}=[.4 .4]';
z2{1}=[.4 .4]'; z2{2}=[.4 .4]'; z2{3}=[.4 .4]';
z2{4}=[.4 .4]'; z2{5}=[.4 .4]'; z2{6}=[.4 .4]';

cyc=90; % doba cyklu
mz=5; % minimální zelené (společné)
mp=mz/cyc;

% Počítané veličiny
d1{1}=[mp mp]'; d1{2}=[mp mp]'; d1{3}=[mp mp]'; d1{4}=[mp mp]';
d2{1}=[mp mp]'; d2{2}=[mp mp]'; d2{3}=[mp mp]';
d2{4}=[mp mp]'; d2{5}=[mp mp]'; d2{6}=[mp mp]';
n2=n1+k*288; % konec reálných dat
nd=n2-n1+1; % počet dat

% základní nastavení vah pro LP
cc1=zeros(1,24); cc1(1:2:15)=1;
```



```

cc2=zeros(1,32); cc2(1:2:19)=1;

% Základní nastavení nadřazeného regulátoru
zns=ones(20,1)*.4;           % poč. setpoint pro zelenou
znt=zns;
dz=.1;                       % poloměr okna setpointu (lokální)
dnz=.3;                       % poloměr okna setpointu (nadraz)
zi=.01;                       % inkrement zelené
zni=.001;                     % inkrement setpointu (nadraz)

% Natažení a amplitudová úprava reálných intenzit
load dat34k40
ki=1.2;                       % celkové zesílení/zeslabení vstupních intenzit
ai=[.12 .8 .75 .5 .8];       % zesílení/zeslabení jednotlivých vstupních intenzit
Id(1,:)=ki*Dat(1,n1:n2)*ai(1);
Id(2,:)=ki*Dat(3,n1:n2)*ai(2);
Id(3,:)=ki*Dat(5,n1:n2)*ai(3);
Id(4,:)=ki*Dat(7,n1:n2)*ai(4);
Id(5,:)=ki*Dat(9,n1:n2)*ai(5);

Ip=[0 0];                     % počáteční intenzity v řezech cyklů oblasti

% směrové vztahy v křižovatkách
a{1}=[0 .8 .2;
      0 0 0;
      0 1 0];
a{2}=[0 0 0;
      1 0 0;
      .7 .3 0];
a{3}=[0 0 0;
      .7 0 .3;
      1 0 0];
a{4}=[0 .1 .9;
      0 0 1;
      0 0 0];
a{5}=[0 0 .7 .3;
      0 0 .7 .3;
      0 0 0 0;
      0 0 0 0];
a{6}=[0 0 0 0;
      .6 0 0 .4;
      .7 0 0 .3;
      0 0 0 0];
a{7}=[0 0 1;
      0 0 1;
      0 0 0];
a{8}=[0 0 0;
      0 0 0];

```

```

    .5 .5 0];
a{9}=[0 0 0;
      0 0 0;
      .5 .5 0];
a{10}=[0 .4 .6 0;
       0 0 0 0;
       0 0 0 0;
       0 .5 .5 0];

% saturované toky (stejně pořadí křižovatek)
as=.8; % celkové zvětšení saturovaných toků
S{1}=[[20 0]; % sloupec: ramena křižovatky
      [ 0 0]; % řádek: fáze
      [ 0 20]]*as; % Př.: první křižovatka má v 1. rameni
S{2}=[[ 0 0]; % ve fázi 1 kapacitu 20, v druhé fázi 0
      [ 0 30];
      [70 0]]*as;
S{3}=[[ 0 0];
      [ 0 60];
      [30 0]]*as;
S{4}=[[30 0];
      [ 0 70];
      [ 0 0]]*as;
S{5}=[[42 0];
      [ 0 50];
      [ 0 0];
      [ 0 0]]*as;
S{6}=[[ 0 0];
      [70 0];
      [ 0 40];
      [ 0 0]]*as;
S{7}=[[50 0];
      [ 0 20];
      [ 0 0]]*as;
S{8}=[[ 0 0];
      [ 0 0];
      [ 0 60]]*as;
S{9}=[[ 0 0];
      [ 0 0];
      [ 0 85]]*as;
S{10}=[[55 0];
       [ 0 0];
       [ 0 0];
       [ 0 140]]*as;

% Konstrukce stavové struktury pro simulaci
zer2=zeros(4,1); zer1=zeros(3,1);
x={zer1,zer1,zer1,zer1,zer2,zer2,zer1,zer1,zer1,zer2}; % počáteční délky kolon

```

```

for i=1:4
    X{i}=mScons(x{i},S{i},z1{i},a{i}); % stavová struktura první mikrooblasti
end
for i=5:10
    X{i}=mScons(x{i},S{i},z2{i-4},a{i}); % stavová struktura druhé mikrooblasti
end
ic1=[1 3 5 6 8 9 10 11]; % čísla ramen s kolonami m-obl 1
ic2=[1 2 6 7 9 10 14 17 18 21]; % čísla ramen s kolonami m-obl 2

opt=optimset('Display','off'); % volby pro LP

% kovariance stavového modelu (kolony, obsazenosti, výst.int.)
rc=100; ro=10; ry=100;

% Popis mikrooblastí pro modelování (pro konstrukci stavového modelu)
[jx1,jy1,K1,Idat1,Iold1,ord1,zz1,xt1,Rw1,Rv1,Rx1,dd1]=iniJun1(S,Id,a,z1,d1,rc,ro,ry);
[jx2,jy2,K2,Idat2,Iold2,ord2,zz2,xt2,Rw2,Rv2,Rx2,dd2]=iniJun2(S,Id,a,z2,d2,rc,ro,ry);

% definice pomocných veličin
dd1=zeros(size(dd1)); dd2=zeros(size(dd2));
hh1=1-dd1; hh2=1-dd2;
zt1=zz1; zt2=zz2;
zs1=zz1; zs2=zz2;

% XXXXXXXXXXXXXXXXXXXXXXXX SIMULACE XXXXXXXXXXXXXXXXXXXXXXXX
% -----
xx1=[]; yy1=[]; oo1=[]; xx2=[]; yy2=[]; oo2=[]; zzc=[];
zzt1=[]; zzt2=[]; xtc1=[]; xtc2=[]; ee1=[]; ee2=[];XXT=[];
K1c=[]; K2c=[]; zA11=[];
if nadcont==1
    md1=[]; md2=[]; mh1=[]; mh2=[];
else
    md1=zeros(length(zz1),nd);
    md2=zeros(length(zz2),nd);
    mh1=ones(length(zz1),nd);
    mh2=ones(length(zz2),nd);
end

for t=1:nd, % <<< --- začátek dynamického cyklu ---

    % simulace obou mikrooblastí
    zc=[zt1; zt2]; zzc=[zzc zc];
    [Y,I,X,Ip]=microSim(Id,X,Ip,zc,t);

    % uchování proměnných
    xp1=[]; xp2=[]; ypa1=[]; ypa2=[];

```

```

[nj1,nf1,nia1]=narm(K1);
for i=1:nj1
    xp1=[xp1; X{i}.x]; ypa1=[ypa1; Y{i}];
end
op1=.4*xp1(jx1)+20+randn(size(xp1(jx1)))*0;
yp1=ypa1(jy1); % měřené intenzity
yt1=[yp1; op1]; % měřené výstupy

[nj2,nf2,nia2]=narm(K2);
for i=nj1+1:nj1+nj2
    xp2=[xp2; X{i}.x]; ypa2=[ypa2; Y{i}];
end
% porucha narušením linearitu vztahu "kolona-obsazenost"
kolobs=0; % nastavení velikosti poruchy
op2=.4*xp2(jx2-12)+20+randn(size(xp2(jx2-12)))*kolobs;
yp2=ypa2(jy2-12); % měřené intenzity z kř. 2
yt2=[yp2; op2]; % měřené výstupy z kř. 2

xx1=[xx1 xp1]; yy1=[yy1 yp1]; oo1=[oo1 op1];
xx2=[xx2 xp2]; yy2=[yy2 yp2]; oo2=[oo2 op2];

% porucha nulováním stavu
timerr=150; % okamžik poruchy
lenerr=1; % trvání poruchy
amperr=0; % velikost stavu v poruše
if abs(t-timerr)<=lenerr,
    xt1=amperr*ones(length(xt1),1);
    xt2=amperr*ones(length(xt2),1);
end

% ----- 1. m-oblast -----
Idat1{3}(2,t)=Y{6}(1); % měřená int. mezi oblastmi

% matice stavového modelu
[A1,B1,F1,C1,D1,Iold1,del1]=makeMat(t,Idat1,K1,xt1,zt1,Iold1,ord1);

% Kalmanuv filtr
[xt1,ey1,Rx1]=Kalman(xt1,yt1,zt1,A1,B1,F1,C1,D1,Rw1,Rv1,Rx1,kalfilt);

xtc1=[xtc1 xt1([1:2:2*nia1-1])]; % délky kolon
ee1=[ee1 ey1]; % chyby predikce

% optimalizace LP
if control==1
    [AA1,bb1,lb1,ub1]=condLP(K1,A1,B1,F1,xt1); % omezení pro LP
    cc1(2*2-1)=1.5; % přestvení váhy pro LP (n-tá váha: n*2-1)

    [ch1,fv1,ex1] = linprog(cc1,[],[],AA1,bb1,lb1,ub1,[],opt); % LP

```

```

if ex1<1,
    fprintf(' --> exit %i in LP1 at time %i\n',ex1,t);
else
    zn1=ch1((2*nia1+1):1:2*nia1+nf1); % načtení zelených
    [dw1,hw1,zs1]=restr(zn1,zt1,zs1,zi,dz,0,dd1,hh1); % omezení na zelené
    zt1=zs1; % optimální poměry zelené
    zzt1=[zzt1 zt1];
end
end

% ----- 2. m-oblast -----
Idat2{1}(1,t)=Y{4}(3); % měřená int. mezi m-obl.

% matice stavového modelu
[A2,B2,F2,C2,D2,Iold2,del2]=makeMat(t,Idat2,K2,xt2,zt2,Iold2,ord2);

% Kalmanuv filtr
[xt2,ey2,Rx2]=Kalman(xt2,yt2,zt2,A2,B2,F2,C2,D2,Rw2,Rv2,Rx2,kalfilt);

xtc2=[xtc2 xt2([1:2:2*nia2-1])]; % délky kolon
ee2 =[ee2 ey2]; % chyby predikce

% optimalizace LP
if control==1
    [AA2,bb2,lb2,ub2]=condLP(K2,A2,B2,F2,xt2);
    cc2(9*2-1)=1;
    cc2(6*2-1)=1.2;
    cc2(1*2-1)=1.2;
    [ch2,fv2,ex2] = linprog(cc2, [], [], AA2,bb2,lb2,ub2, [], opt);
    if ex2<1,
        fprintf(' --> exit %i in LP2 at time %i\n',ex2,t);
    else
        zn2=ch2((2*nia2+1):1:2*nia2+nf2);
        [dw2,hw2,zs2]=restr(zn2,zt2,zs2,zi,dz,0,dd2,hh2);
        zt2=zs2; % optimální poměry zelené
        zzt2=[zzt2 zt2];
    end
end

% ----- nadraz -----
if control==1 & nadcont==1
    It1=[];It2=[];

    for h=1:length(Idat1),It1=[It1;Idat1{h}(:,t)]; end
    for h=1:length(Idat2),It2=[It2;Idat2{h}(:,t)]; end

    [Ac,bc,lc,uc,zob1,zob2,cc,A,B,F,xt,w]=nadraz1(It1,It2,K1,K2,del1,del2,xt1,xt2,zt1,zt2);

```

```

dnd=ones(length(zob1)+length(zob2),1)*mp; % dolni meze zelenych pro nadraz
hnh=ones(length(zob1)+length(zob2),1)*(0.8-mp); % horni meze zelenych pro nadraz

[chc, fvc, exc] = linprog(cc, [], [], Ac, bc, lc, uc, [], opt); % LP

if exc>0 & control==1 % drift zelené
    zn=chc(3:end); % zelené z LP; 2=počet oblastí
    zob=[zob1, length(zt1)+zob2];
    [dw, hw, zns(zob)]=restr(zn, znt(zob), zns(zob), zni, dnz, dnz/1000, dnd, hnh);

    dd1(zob1)=dw(1:length(zob1));
    hh1(zob1)=hw(1:length(zob1));
    dd2(zob2)=dw((1+length(zob1)):end);
    hh2(zob2)=hw((1+length(zob1)):end);
else
    fprintf(' No solution at NADRAZ at time %d\n', t);
end
md1(:, t)=dd1; md2(:, t)=dd2;
mh1(:, t)=hh1; mh2(:, t)=hh2;
end

end % for t XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

% Zobrazení výsledků
clc
% Průměrná kolona
xx=[xx1(ic1,:); xx2(ic2,:)]; xtt=sum(xx); xts=mean(xtt);
fprintf(' Prumer ze souctu vsech delek kolon je: %g\n', xts);

% Plot výsledků
nh=150; % vertikální měřítko pro grafy

figure(1), clf, figure(2), clf
set(figure(1), 'Position', [1 10 670 850], 'MenuBar', 'none')
subplot(9,6, 7), plot(1:nd, xtc1(1,1:nd), 'xr', 1:nd, xx1(1,1:nd), 'b', 'markersize', 3), axis([0, nd, 0
set(get(gcf, 'CurrentAxes'), 'xtick', [], 'ytick', [])
subplot(9,6, 2), plot(1:nd, xtc1(2,1:nd), 'xr', 1:nd, xx1(3,1:nd), 'b', 'markersize', 3), axis([0, nd, 0
set(get(gcf, 'CurrentAxes'), 'xtick', [], 'ytick', [])
subplot(9,6,32), plot(1:nd, xtc1(3,1:nd), 'xr', 1:nd, xx1(5,1:nd), 'b', 'markersize', 3), axis([0, nd, 0
set(get(gcf, 'CurrentAxes'), 'xtick', [], 'ytick', [])
subplot(9,6,27), plot(1:nd, xtc1(4,1:nd), 'xr', 1:nd, xx1(6,1:nd), 'b', 'markersize', 3), axis([0, nd, 0
set(get(gcf, 'CurrentAxes'), 'xtick', [], 'ytick', [])
subplot(9,6,30), plot(1:nd, xtc1(5,1:nd), 'xr', 1:nd, xx1(8,1:nd), 'b', 'markersize', 3), axis([0, nd, 0
set(get(gcf, 'CurrentAxes'), 'xtick', [], 'ytick', [])
subplot(9,6,23), plot(1:nd, xtc1(6,1:nd), 'xr', 1:nd, xx1(9,1:nd), 'b', 'markersize', 3), axis([0, nd, 0
set(get(gcf, 'CurrentAxes'), 'xtick', [], 'ytick', [])
subplot(9,6,10), plot(1:nd, xtc1(7,1:nd), 'xr', 1:nd, xx1(10,1:nd), 'b', 'markersize', 3), axis([0, nd, 0
set(get(gcf, 'CurrentAxes'), 'xtick', [], 'ytick', [])

```

```
subplot(9,6,17),plot(1:nd,xtc1(8,1:nd),'xr',1:nd,xx1(11,1:nd),'b','markersize',3),axis([0,nd,0,nd])
set(get(gcf,'CurrentAxes'),'xtick',[],'ytick',[])
```

```
subplot(9,6,9),plot(nd/2,nh/2,'rx','markersize',10),axis([0,nd,0,nh])
set(get(gcf,'CurrentAxes'),'xtick',[],'ytick',[])
subplot(9,6,12),plot(nd/2,nh/2,'rx','markersize',10),axis([0,nd,0,nh])
set(get(gcf,'CurrentAxes'),'xtick',[],'ytick',[])
subplot(9,6,25),plot(nd/2,nh/2,'rx','markersize',10),axis([0,nd,0,nh])
set(get(gcf,'CurrentAxes'),'xtick',[],'ytick',[])
subplot(9,6,28),plot(nd/2,nh/2,'rx','markersize',10),axis([0,nd,0,nh])
set(get(gcf,'CurrentAxes'),'xtick',[],'ytick',[])
```

```
subplot(9,6,8),plot(nd/2,nh/2,'ro','markersize',10),axis([0,nd,0,nh])
set(get(gcf,'CurrentAxes'),'xtick',[],'ytick',[])
subplot(9,6,11),plot(nd/2,nh/2,'ro','markersize',10),axis([0,nd,0,nh])
set(get(gcf,'CurrentAxes'),'xtick',[],'ytick',[])
subplot(9,6,26),plot(nd/2,nh/2,'ro','markersize',10),axis([0,nd,0,nh])
set(get(gcf,'CurrentAxes'),'xtick',[],'ytick',[])
subplot(9,6,29),plot(nd/2,nh/2,'ro','markersize',10),axis([0,nd,0,nh])
set(get(gcf,'CurrentAxes'),'xtick',[],'ytick',[])
```

```
print -djpeg fig1_111
```

```
set(figure(2),'Position',[680 10 650 850],'MenuBar','none')
set(get(gcf,'CurrentAxes'),'xtick',[],'ytick',[])
subplot(9,6,7),plot(1:nd,xtc2(1,1:nd),'xr',1:nd,xx2(1,1:nd),'b','markersize',3),axis([0,nd,0,nd])
set(get(gcf,'CurrentAxes'),'xtick',[],'ytick',[])
subplot(9,6,14),plot(1:nd,xtc2(2,1:nd),'xr',1:nd,xx2(2,1:nd),'b','markersize',3),axis([0,nd,0,nd])
set(get(gcf,'CurrentAxes'),'xtick',[],'ytick',[])
subplot(9,6,32),plot(1:nd,xtc2(3,1:nd),'xr',1:nd,xx2(6,1:nd),'b','markersize',3),axis([0,nd,0,nd])
set(get(gcf,'CurrentAxes'),'xtick',[],'ytick',[])
subplot(9,6,27),plot(1:nd,xtc2(4,1:nd),'xr',1:nd,xx2(7,1:nd),'b','markersize',3),axis([0,nd,0,nd])
set(get(gcf,'CurrentAxes'),'xtick',[],'ytick',[])
subplot(9,6,50),plot(1:nd,xtc2(5,1:nd),'xr',1:nd,xx2(9,1:nd),'b','markersize',3),axis([0,nd,0,nd])
set(get(gcf,'CurrentAxes'),'xtick',[],'ytick',[])
subplot(9,6,45),plot(1:nd,xtc2(6,1:nd),'xr',1:nd,xx2(10,1:nd),'b','markersize',3),axis([0,nd,0,nd])
set(get(gcf,'CurrentAxes'),'xtick',[],'ytick',[])
subplot(9,6,41),plot(1:nd,xtc2(7,1:nd),'xr',1:nd,xx2(14,1:nd),'b','markersize',3),axis([0,nd,0,nd])
set(get(gcf,'CurrentAxes'),'xtick',[],'ytick',[])
subplot(9,6,23),plot(1:nd,xtc2(8,1:nd),'xr',1:nd,xx2(17,1:nd),'b','markersize',3),axis([0,nd,0,nd])
set(get(gcf,'CurrentAxes'),'xtick',[],'ytick',[])
subplot(9,6,10),plot(1:nd,xtc2(9,1:nd),'xr',1:nd,xx2(18,1:nd),'b','markersize',3),axis([0,nd,0,nd])
set(get(gcf,'CurrentAxes'),'xtick',[],'ytick',[])
subplot(9,6,5),plot(1:nd,xtc2(10,1:nd),'xr',1:nd,xx2(21,1:nd),'b','markersize',3),axis([0,nd,0,nd])
set(get(gcf,'CurrentAxes'),'xtick',[],'ytick',[])
```

```
subplot(9,6,2),plot(nd/2,nh/2,'rx','markersize',10),axis([0,nd,0,nh])
set(get(gcf,'CurrentAxes'),'xtick',[],'ytick',[])
```

```

subplot(9,6,9),plot(nd/2,nh/2,'rx','markersize',10),axis([0,nd,0,nh])
set(get(gcf,'CurrentAxes'),'xtick',[],'ytick',[])
subplot(9,6,25),plot(nd/2,nh/2,'rx','markersize',10),axis([0,nd,0,nh])
set(get(gcf,'CurrentAxes'),'xtick',[],'ytick',[])
subplot(9,6,20),plot(nd/2,nh/2,'rx','markersize',10),axis([0,nd,0,nh])
set(get(gcf,'CurrentAxes'),'xtick',[],'ytick',[])
set(get(gcf,'CurrentAxes'),'xtick',[],'ytick',[])
subplot(9,6,38),plot(nd/2,nh/2,'rx','markersize',10),axis([0,nd,0,nh])
set(get(gcf,'CurrentAxes'),'xtick',[],'ytick',[])
subplot(9,6,46),plot(nd/2,nh/2,'rx','markersize',10),axis([0,nd,0,nh])
set(get(gcf,'CurrentAxes'),'xtick',[],'ytick',[])
set(get(gcf,'CurrentAxes'),'xtick',[],'ytick',[])
subplot(9,6,48),plot(nd/2,nh/2,'rx','markersize',10),axis([0,nd,0,nh])
set(get(gcf,'CurrentAxes'),'xtick',[],'ytick',[])
subplot(9,6,35),plot(nd/2,nh/2,'rx','markersize',10),axis([0,nd,0,nh])
set(get(gcf,'CurrentAxes'),'xtick',[],'ytick',[])
subplot(9,6,28),plot(nd/2,nh/2,'rx','markersize',10),axis([0,nd,0,nh])
set(get(gcf,'CurrentAxes'),'xtick',[],'ytick',[])
subplot(9,6,17),plot(nd/2,nh/2,'rx','markersize',10),axis([0,nd,0,nh])
set(get(gcf,'CurrentAxes'),'xtick',[],'ytick',[])
subplot(9,6,12),plot(nd/2,nh/2,'rx','markersize',10),axis([0,nd,0,nh])
set(get(gcf,'CurrentAxes'),'xtick',[],'ytick',[])

```

```

subplot(9,6, 8),plot(nd/2,nh/2,'ro','markersize',10),axis([0,nd,0,nh])
set(get(gcf,'CurrentAxes'),'xtick',[],'ytick',[])
subplot(9,6,11),plot(nd/2,nh/2,'ro','markersize',10),axis([0,nd,0,nh])
set(get(gcf,'CurrentAxes'),'xtick',[],'ytick',[])
subplot(9,6,26),plot(nd/2,nh/2,'ro','markersize',10),axis([0,nd,0,nh])
set(get(gcf,'CurrentAxes'),'xtick',[],'ytick',[])
subplot(9,6,29),plot(nd/2,nh/2,'ro','markersize',10),axis([0,nd,0,nh])
set(get(gcf,'CurrentAxes'),'xtick',[],'ytick',[])
subplot(9,6,44),plot(nd/2,nh/2,'ro','markersize',10),axis([0,nd,0,nh])
set(get(gcf,'CurrentAxes'),'xtick',[],'ytick',[])
subplot(9,6,47),plot(nd/2,nh/2,'ro','markersize',10),axis([0,nd,0,nh])
set(get(gcf,'CurrentAxes'),'xtick',[],'ytick',[])

```

```
print -djpeg fig2_111
```

Nadřazené řízení

```

function [Ac,bc,lc,uc,porz1,porz2,cc,A,B,F,XT,w]=nadraz1(It1,It2,K1,K2,del1,del2,xt1,xt2,zt1,
% [zt1,zt2,exc]=nadraz(It1,It2,K1,K2,del1,del2,xt1,xt2,zt1,zt2,cx,zd,zh,zj,control,opt)
% senior controler for hierarchical control of traffic
% zt zelené [z1 z2 z3 z4]
% xt odhady délek kolon [c1 c2 ... c8]
% It aktuální intenzity [i1 i2 ... i6]
% S saturované toky (dvě fáze, jen rovně)
% cx věhy kolon v kritériu LP

```



```

% zd  minimální zelené
% zh  maximální zelené
% zi  inkrement pro aktuální zelenou
% control řízení: 0=ne, 1=ano
% opt  options pro LP: opt=optimset('Display','off');
%
% napsal Ivan a opičil se přitom po Jitce

sz1.kr=[1 2 4];    % [krizovatka]
sz1.ri{1}={ [1 3] };
sz1.ri{2}={ [2 1] [3 1;3 2] };
sz1.ri{3}={ [1 3] [2 3] };
sz2.kr=[1 2 4 6];
sz2.ri{1}={ [1 4] [2 4] };
sz2.ri{2}={ [2 1] [3 1] };
sz2.ri{3}={ [3 2] };
sz2.ri{4}={ [1 3] [4 3] };

[porz1,P1,PR2,IT1]=matP(K1,It1,xt1,sz1,del1,4,3);
[porz2,P2,PR1,IT2]=matP(K2,It2,xt2,sz2,del2,2,1);

okr1=[1,3,5];
okr2=[9,19];
XT1=sum(xt1(setdiff(1:2:length(xt1),okr1)));
XT2=sum(xt2(setdiff(1:2:length(xt2),okr2)));
XT=[XT1;XT2];          % stav pro NADRAZ

w1=1;    % pocatecni nastaveni vah
w2=1;
w3=20;
if 6*XT1>4*XT2+w3, w1=5; end
if 6*XT1+w3<4*XT2, w2=5; end
w=[w1 w2];

ZT1=zt1([porz1]);
ZT2=zt2([porz2]);
ZT=[ZT1;ZT2];          % rizeni pro NADRAZ

A=eye(2);              % A pro stav. model NADRAZ

B=[-P1 zeros(size(P2));
   zeros(size(P1)) -P2];
B(1,size(P1,2)+PR1.p)=-PR1.h;
B(2,PR2.p)=-PR2.h;

F=[IT1;IT2]; % F pro stav. model NADRAZ

```

```

zera=zeros(1,length(A)); c=0; % c=1:stejné, c=0: různé křižovatky
Ac=[eye(length(A)) -B; % A pro LP
    zera 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0;
    zera 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0;
    zera 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0;
    zera 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0;
    zera 0 0 0 0 0 0 0 0 1 1 0 0 0 0 0;
    zera 0 0 0 0 0 0 0 0 0 0 1 1 0 0 0;
    zera 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0;
    zera 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1;
];
bc=[A*XT+F; 1; 1; 1; 1; 1; 1; 1; 1]; % b pro LP
cc=[w1 w2 0 0 0 0 0 0 0 0 0 0 0 0 0 0]; % váhy kriteria pro LP
lc=[zera 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]; % dolní meze pro LP
uc=[ones(size(zera))*inf .8 .8 .8 .8 .8 .8 .8 .8 .8 .8 .8 .8 .8 .8]; % horní meze pro LP

```