Optimální ustálený chod
Optima Power Flow -OPF

$C_i(P_i)$ **cena výroby i-tého zdroje**

Cílové funkce:

1. **minimalizace přenosových ztrát**
2. **minimum ceny vyráběné energie**
3. **alokace kompenzačních výkonů**
4. **minimalizace ekologických vlivů**
5. **optimalizace výkonových rezerv**

omez. podmínky typu rovnosti, nerovnosti

$$\overline{\mathrm{e}}(\overline{x}) = \overline{0},\ \overline{\mathrm{N}}(\overline{x}) \leq \overline{0}$$

Omezení veličin a funkčních hodnot

$$\overline{x}^L \leq \overline{x} \leq \overline{x}^U,\ \ \overline{\Phi}^L \leq \overline{\Phi} \leq \Phi^U$$

Algoritmus separátního řešení

1) **iniciace startovacího bodu**

$\overline{\rho}^0, \overline{\lambda}^0, \overline{\nu}^0$ **,k=0**

2) **řešení ustáleného stavu**

$$\overline{e}(\overline{\sigma}, \overline{\rho}^0) = \overline{0}$$

(BPF-výpočet $\overline{\sigma}$)

3) **Je-li režim optimální pak konec jinak pokračuj**

4) **aproximace (LP,QP) cílové funkce a omezujících podmínek k:=k+1;**

5) **optimalizace $\overline{\xi}^{(k)}$, návrat do bodu 2**

Algoritmus Integrovaného řešení

Lagrangián:

$$L(\bar{\xi}) = f(\bar{x}) + \bar{\mu}_b^T \bar{B} + \bar{\mu}_p^T . \bar{P} + \bar{\lambda}^T \bar{e} + \bar{\nu}^T \bar{N}$$

$$\bar{\xi} = \left[\bar{x}, \bar{\lambda}, \bar{\nu} \right]$$

Algoritmus řešení:

1. Iniciace. k=0,

$$\bar{\xi} = \bar{\xi}^0, \bar{\mu}_b = \bar{\mu}_b^0, \bar{\mu}_p = \bar{\mu}_p^0$$

2. Výpočet $\nabla_{\bar{\xi}} L^{(k)}$, určení akt. Γ_a

1. **Je-li režim optimální pak konec jinak pokračuj**
2. **generace rovnic (stanovení** $\left[J^{(k)} \right], \left[H^{(k)} \right], \left[\nabla^2 L(\xi^{(k)} \right]$
3. **řešení rovnic** $\left[\nabla^2 L(\xi^{(k)}) \right] \bullet \Delta \bar{\xi}^{(k)} = -\nabla \bar{L}^{(k)}$,
4. **aktualizace** $\bar{\xi}, \bar{\mu}_p, \bar{\mu}_b$

   k=k+1, návrat na 2.

Příklad uspořádání:

$$\xi^T = \begin{bmatrix} \bar{\delta}^T & \bar{U}^T & \bar{t}^T & \bar{\lambda}_P^T & \lambda_Q^T \\ PQPU & NU & Nt & PQPU & PQ \end{bmatrix}$$

$$L = P_0 + \bar{\lambda}_P^T \left(\bar{P}_{act} - \bar{P}^W \right) + \lambda_Q^T \left(\bar{Q}_{act} - \bar{Q}^W \right)$$

$$L = P_0 + \sum_{k \in PQPU} \lambda_p(k) . \left(P_{act}(k) - P^W(k) \right) +$$

$$+ \sum_{k \in PQ} \lambda_q(k) . \left(Q_{act}(k) - Q^W(k) \right) +$$

$$+ \sum_{\forall j} \frac{\eta_j^U}{2} \left(x - x_j^U \right)^2 l_j^U + \frac{\eta_j^L}{2} \left(x_j - x_j^L \right)^2 l_j^L$$

$l_j^{U,L}$ = logická veličina indikující aktivnost omezení

$\eta_j^{U,L}$ = penalizační koeficient aktivního omezení

Newtonova metoda optimalizace (postupné kvadratické programování)
ustálený chod: def. S $\Rightarrow$ 0
OPF : $\nabla L \Rightarrow$ 0

$$L = f\left(\bar{\rho},\bar{s}\right) + \bar{\lambda}^T . \overline{\mathrm{e}}\left(\bar{\rho},\bar{s}\right) + \sum_{\forall a} \mu.\Gamma_a \; ..Lagrangeova \;\; funkce$$

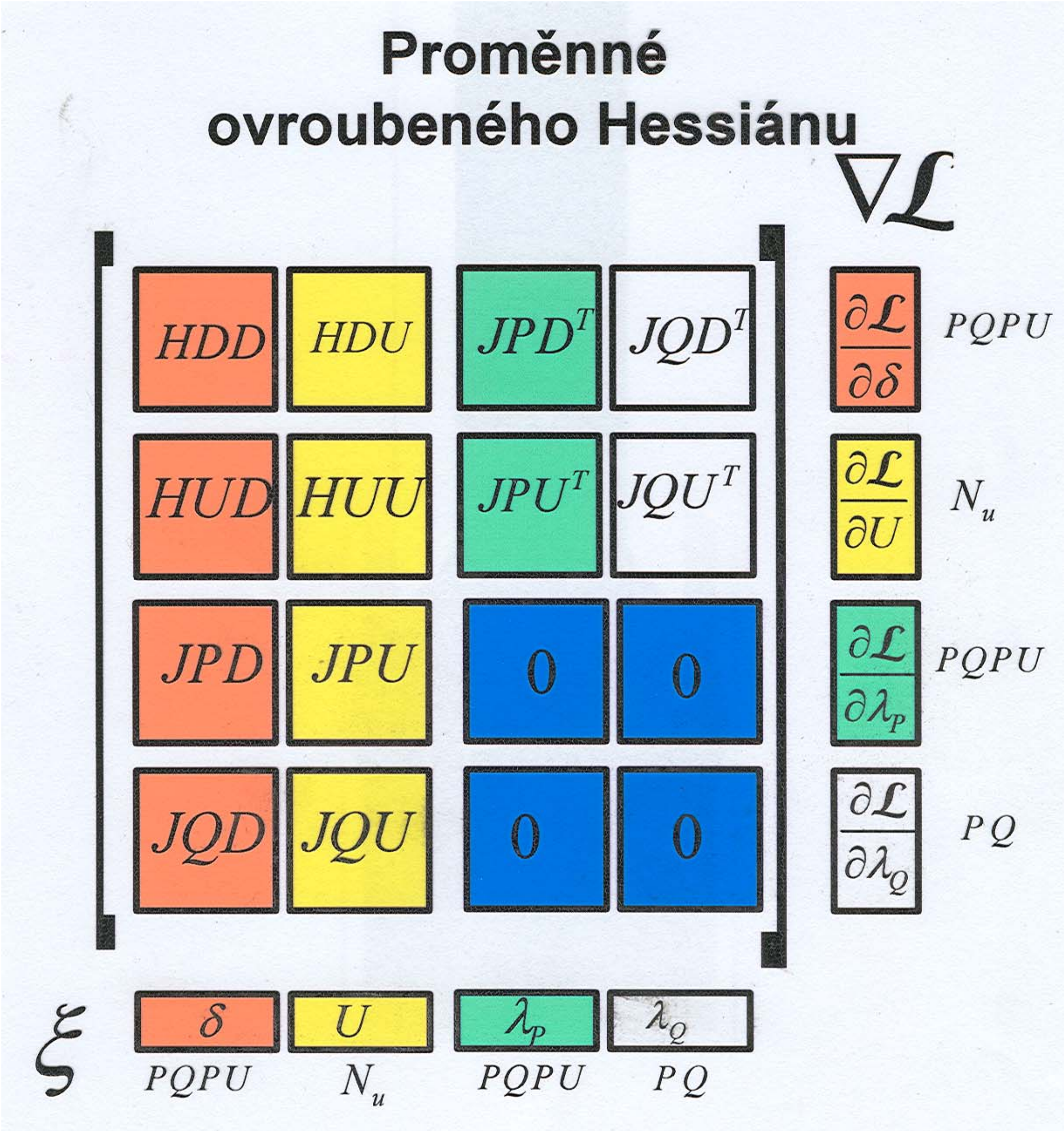
a…množina aktivních omezení

$$\bar{\xi}^T = \left[\bar{\rho}^T, \bar{\sigma}^T, \lambda^T\right] ...vektor \; proměných$$

$$\begin{bmatrix} \Delta\bar{\rho} \\ \Delta\bar{\sigma} \\ \Delta\bar{\lambda} \end{bmatrix} = \left[\begin{array}{cc} \boxed{\begin{matrix} \partial\nabla_{\bar{\rho}}L/\partial\bar{\rho} & \mathrm{H} & \partial\nabla_{\bar{\rho}}L/\partial\bar{\sigma} \\ \partial\nabla_{\bar{\sigma}}L/\partial\bar{\rho} & & \partial\nabla_{\bar{\sigma}}L/\partial\bar{\sigma} \end{matrix}} & \boxed{\begin{matrix} \mathrm{J}^{\mathrm{T}} & \partial\nabla_{\bar{\rho}}L/\partial\bar{\lambda} \\ & \partial\nabla_{\bar{\sigma}}L/\partial\bar{\lambda} \end{matrix}} \\ \boxed{\begin{matrix} \partial\nabla_{\bar{\lambda}}L/\partial\bar{\rho} & \mathrm{J} & \partial\nabla_{\bar{\lambda}}L/\partial\bar{\sigma} \end{matrix}} & \boxed{\begin{matrix} \mathrm{0} & \partial\nabla_{\bar{\lambda}}L/\partial\bar{\lambda} \end{matrix}} \end{array}\right] \begin{bmatrix} \nabla_{\bar{\rho}}L \\ \nabla_{\bar{\sigma}}L \\ \nabla_{\bar{\lambda}}L \end{bmatrix}$$

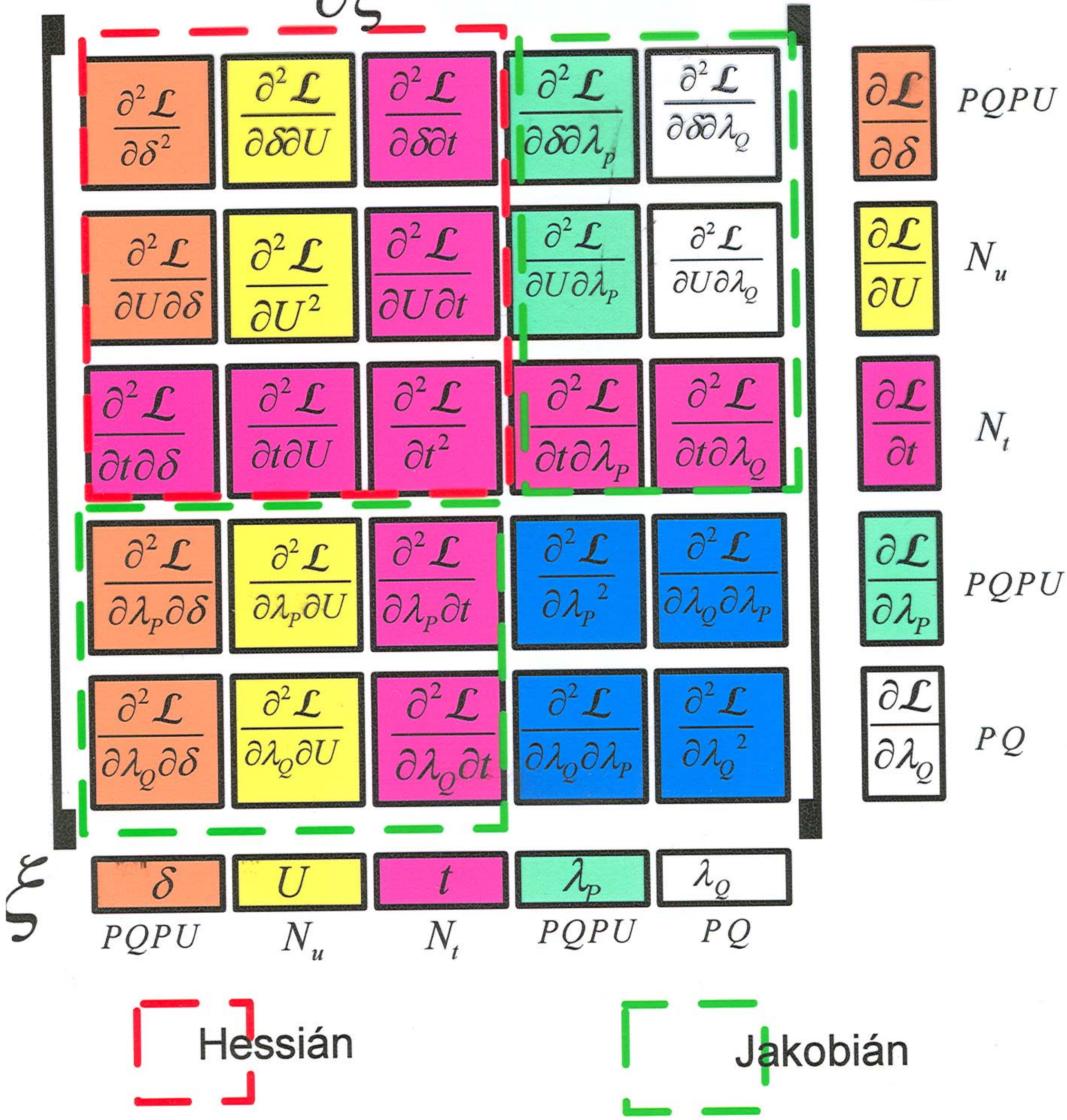
$$\nabla L = \frac{\partial L}{\partial\bar{\xi}} ....gradient \;\; Lagrangiánu$$

$$\left[W\right] = \frac{\partial\left\{\nabla_{\bar{\xi}} L\right\}}{\partial\bar{\xi}} = \frac{\partial^2 L}{\partial\bar{\xi}^2} ....ovroubený \; Hessián$$


Proměnné
ovroubeného Hessiánu
$\nabla\mathcal{L}$
HDD
HDU
JPD^T
JQD^T
HUD
HUU
JPU^T
JQU^T
JPD
JPU
0
0
JQD
JQU
0
0
$\frac{\partial\mathcal{L}}{\partial\delta}$
PQPU
$\frac{\partial\mathcal{L}}{\partial U}$
N_u
$\frac{\partial\mathcal{L}}{\partial\lambda_P}$
PQPU
$\frac{\partial\mathcal{L}}{\partial\lambda_Q}$
PQ
ξ
δ
U
λ_P
λ_Q
PQPU
N_u
PQPU
PQ

Struktura ovroubeného Hessiánu

$$[W] = \frac{\partial}{\partial \overline{\xi}} \{\nabla \mathcal{L}\}$$



$$\frac{\partial L}{\partial x_k} = \frac{\partial P_0}{\partial x_k} + \overline{\lambda}_p^T \frac{\partial \overline{P}_{act}}{\partial x_k} + \lambda_q^T \frac{\partial \overline{Q}_{act}}{\partial x_k} +$$

$$\frac{\partial L}{\partial x_k} = \frac{\partial P_0}{\partial x_k} + \sum_{k \in PQPU} \frac{\partial}{\partial x_k}\left\{\lambda_p(k).P_{act}(k)\right\} +$$

$$+ \sum_{k \in PQ} \frac{\partial}{\partial x_k}\left\{\lambda_q(k).Q_{act}(k)\right\} + \sum_{\forall j} \eta_j^U x_j l_j^U + \eta_j^L x_j l_j^L$$

$$\frac{\partial^2 L}{\partial x_i.\partial v_j} = \frac{\partial^2 P_0}{\partial x_i.\partial v_j} + \overline{\lambda}_p^T \frac{\partial^2 \overline{P}_{act}}{\partial x_i.\partial x_j} + \lambda_q^T \frac{\partial^2 \overline{Q}_{act}}{\partial x_i.\partial x_j}$$

$$\frac{\partial^2 L}{\partial x_i.\partial x_j} = \frac{\partial^2 P_0}{\partial x_i.\partial x_j} + \sum_{k \in PQPU} \frac{\partial^2}{\partial x_i.\partial x_j}\left\{\lambda_p(k).P_{act}(k)\right\} +$$

$$+ \sum_{k \in PQ} \frac{\partial^2}{\partial x_i.\partial x_j}\left\{\lambda_q(k).Q_{act}(k)\right\} + \sum_{\forall j} \eta_j^U x_j l_j^U + \eta_j^L x_j l_j^L$$

OPF-IP

$$\underset{\overline{x} \subseteq \mathrm{D}_x}{opt} \left\{ f\left(\overline{x}\right) \right\}$$

$$\mathrm{D}_x = \left\{ \begin{array}{l} \overline{x} : \overline{\mathrm{e}}\left(\overline{x}\right) = \overline{0}, \\ \overline{\mathrm{N}}^{\mathrm{L}} \leq \overline{\mathrm{N}}\left(\overline{x}\right) \geq \overline{\mathrm{N}}^{\mathrm{U}}, \\ \overline{x}^{L} \leq \overline{x} \leq \overline{x}^{U} \end{array} \right\}$$

$$\overline{\mathrm{N}}\left(\overline{x}\right) - \overline{\mathrm{N}}' - \overline{\mathrm{N}}^{\mathrm{L}} = \overline{0}$$

$$1)\ \ \overline{\mathrm{N}}\left(\overline{x}\right) + \overline{\mathrm{N}}'' - \overline{\mathrm{N}}^{\mathrm{U}} = \overline{0}$$

$$2)\ \ \overline{\mathrm{N}}'' + \overline{\mathrm{N}}' - \overline{\mathrm{N}}^{\mathrm{U}} + \overline{\mathrm{N}}^{\mathrm{L}} = \overline{0}$$

$$\overline{x} + \overline{x}'' = \overline{x}^{U}$$

$$\overline{x} - \overline{x}' = \overline{x}^{L}$$

$$\overline{x}'' + \overline{x}' = \overline{x}^{U} - \overline{x}^{L}$$

$$nez\acute{a}pornost : \overline{x}', \overline{x}'', \overline{\mathrm{N}}', \overline{\mathrm{N}}''$$

Rovnice pro řešení:

$$\nabla_{\bar{x}} L = \bar{0} = \nabla_{\bar{x}} f\left(\bar{x}\right) + \left\{\nabla_{\bar{x}} \bar{e}\left(\bar{x}\right)\right\} \bar{\lambda}$$

$$+ \left\{\nabla_{\bar{x}} \bar{N}^{\mathrm{T}}\left(\bar{x}\right)\right\} \bar{\lambda}_N + \bar{\lambda}_x - \mu\left(\bar{x} - \bar{x}'\right)^{-1} \overline{\text{❶}}$$

$$\nabla_{\bar{N}''} L = \bar{0} = -\bar{\lambda}_{\bar{N}} - \bar{\lambda}_{UL} + \mu\left[\bar{N}''_{\%}\right]^{-1} \overline{\text{❶}}$$

$$\nabla_{\bar{N}'} L = \bar{0} = -\bar{\lambda}_{UL} + \mu\left[\bar{N}'_{\%}\right]^{-1} \overline{\text{❶}}$$

$$\nabla_{\bar{x}''} L = \bar{0} = -\bar{\lambda}_{\bar{x}} + \mu\left[\bar{\mathrm{x}}''_{\%}\right]^{-1} \overline{\text{❶}}$$

$$\nabla_{\bar{\lambda}} L = \bar{0} = \bar{e}\left(\bar{x}\right)$$

$$\nabla_{\bar{\lambda}_{\bar{N}}} L = \bar{0} = \bar{N}^{\mathrm{U}} - \bar{N}\left(\bar{x}\right) - \bar{N}''$$

$$\nabla_{\bar{\lambda}_{\bar{x}}} L = \bar{0} = \bar{x}^{U} - \bar{x} - \bar{x}''$$

$$\nabla_{\bar{\lambda}_{UL}} L = \bar{0} = \bar{N}^{\mathrm{U}} - \bar{N}^{\mathrm{L}} - \bar{N}'' - \bar{N}'$$

% konstrukce obroubeného hessianu.

```
radkuJ=size(J,1);nulaW=zeros(radkuJ,radkuJ);    % pravy spodni roh matice W
HUU=sparse(NU,NU);HUD=HUU;HDD=HUU; %  iniciace H
for Ip=1:size(II,1),       % pro nenulový prvek matice Y
    k=II(Ip); m=JJ(Ip); % indexy v matici Y (spojené uzly)
     ya=Ym(Ip);Dkm=Deltakm(Ip);        % modul a uhel
   if k~=m                               % nediagonální prvky
       Sk_Uk_Um =ya*exp(i*Dkm);
       Sk_Uk_Dm =-i*Ua(m)*ya*exp(i*Dkm);
        Sk_Um_Dk =i*Ua(k)*ya*exp(i*Dkm);
        Sk_Um_Dm =-Sk_Um_Dk ;
        Sk_Dk_Dm =Ua(m)*Ua(k)*ya*exp(i*Dkm);
        Sk_Dm_Dm =-Sk_Dk_Dm ;
        Sk_Um_Um =0;
     if TU(k)==3                     %pro uzel typu Slack
             HUU(k,m)=HUU(k,m)+real(Sk_Uk_Um);
             HUD(k,m)=HUD(k,m)+real(Sk_Uk_Dm);
             HDD(k,m)=HDD(k,m)+real(Sk_Dk_Dm);
             HUU(m,k)=HUU(m,k)+real(Sk_Uk_Um);
             HUD(m,k)=HUD(m,k)+real(Sk_Um_Dk);
             HDD(m,k)=HDD(m,k)+real(Sk_Dk_Dm);
              HUU(m,m)=HUU(m,m)+real(Sk_Um_Um);
              HUD(m,m)=HUD(m,m)+real(Sk_Um_Dm);
              HDD(m,m)=HDD(m,m)+real(Sk_Dm_Dm);
      end %if TU(k)==3
      if TU(k)~=3                     %pro uzly typu PQPU
       HUU(m,k)=HUU(m,k)-real(Sk_Uk_Um)*LP(k);
            HUD(m,k)=HUD(m,k)-real(Sk_Um_Dk)*LP(k);
                  HDD(m,k)=HDD(m,k)-
real(Sk_Dk_Dm)*LP(k);
                  HUU(k,m)=HUU(k,m)-
real(Sk_Uk_Um)*LP(k);
                 HUD(k,m)=HUD(k,m)-real(Sk_Uk_Dm)*LP(k);
```

```
                    HDD(k,m)=HDD(k,m)-
real(Sk_Dk_Dm)*LP(k);
          HUU(m,m)=HUU(m,m)-real(Sk_Um_Um)*LP(k);
      HUD(m,m)=HUD(m,m)-real(Sk_Um_Dm)*LP(k);
            HDD(m,m)=HDD(m,m)-real(Sk_Dm_Dm)*LP(k);
      end %if TU(k)~=3


      if TU(k)==1                    %pro uzly typu PQ
                  HUU(m,k)=HUU(m,k)-
imag(Sk_Uk_Um)*LQ(k);
                  HUD(m,k)=HUD(m,k)-
imag(Sk_Um_Dk)*LQ(k);
                  HDD(m,k)=HDD(m,k)-
imag(Sk_Dk_Dm)*LQ(k);
            HUU(k,m)=HUU(k,m)-imag(Sk_Uk_Um)*LQ(k);
                  HUD(k,m)=HUD(k,m)-
imag(Sk_Uk_Dm)*LQ(k);
                  HDD(k,m)=HDD(k,m)-
imag(Sk_Dk_Dm)*LQ(k);
        HUU(m,m)=HUU(m,m)-imag(Sk_Um_Um)*LQ(k);
                  HUD(m,m)=HUD(m,m)-
imag(Sk_Um_Dm)*LQ(k);
                  HDD(m,m)=HDD(m,m)-
imag(Sk_Dm_Dm)*LQ(k);
      end %if TU(k)==1

% diagonální prvky k=m
     Sk_Uk_Uk=2*ya*exp(i*Dkm);
     Sk_Dk_Dk=-Sact(k)+Ua(k)^2*ya*exp(i*Dkm);
     Sk_Uk_Dk=i*(Sact(k)/Ua(k)-Ua(k)*ya*exp(i*Dkm));

     if TU(k)==3                      %pro uzel typu Slack
                  HUU(k,k)=HUU(k,k)+real(Sk_Uk_Uk);
                  HUD(k,k)=HUD(k,k)+real(Sk_Uk_Dk);
                  HDD(k,k)=HDD(k,k)+real(Sk_Dk_Dk);
```

```
        end %      if TU(k)==3
        if TU(k)~=3                          %pro uzly typu PQPU
            HUU(k,k)=HUU(k,k)-real(Sk_Uk_Uk)*LP(k);
            HUD(k,k)=HUD(k,k)-real(Sk_Uk_Dk)*LP(k);
            HDD(k,k)=HDD(k,k)-real(Sk_Dk_Dk)*LP(k);
        end %if TU(k)~=3

  if TU(k)==1                                %pro uzly typu PQ
            HUU(k,k)=HUU(k,k)-imag(Sk_Uk_Uk)*LQ(k);
            HUD(k,k)=HUD(k,k)-imag(Sk_Uk_Dk)*LQ(k);
            HDD(k,k)=HDD(k,k)-imag(Sk_Dk_Dk)*LQ(k);
        end %if TU(k)==1
  end %if k~=m
 end %for Ip


for k=vse
    HUU(k,k)=HUU(k,k)+LUH(k)*WUH
+LUD(k)*WUD+WUS;
end

% sestavení výsledné matice H
H=[HDD(PQPU,PQPU)  HUD(vse,PQPU)';
    HUD(vse,PQPU)  HUU(vse,vse)];

W=[H  -J'; -J  nulaW];% sestavení výsledné matice W
```