GAMS CODE FOR DC- OPF

CONTRIBUTED BY DR. ALIREZA SOROUDI, UNIVERSITY COLLEGE DUBLIN, DUBLIN, IRELAND.

EMAIL: ALIREZA.SOROUDI@GMAIL.COM

ALIREZA.SOROUDI@UCD.IE

DESCRIPTION

The current DC optimal power flow (OPF) model finds the optimal operating schedules of generating units considering transmission line constraints.

The objective function is defined as the total operating costs of generating units.

The model type is LP and is applied to a 5 bus PJM test system as described in the following reference:

F. Li and R. Bo, "Small test systems for power system economic studies," IEEE PES General Meeting, Minneapolis, MN, 2010, pp. 1-4.

doi: 10.1109/PES.2010.5589973

The proposed model is able to handle large scale set of data for practical power system transmission networks.

INPUT-OUTPUT

Inputs:

- Generator costs + technical characteristics (min/max , connection point)
- Network characteristics (reactances, demands, line flow limits)

Outputs

- local marginal prices (LMP)
- Generators outputs
- Line flows
- Angle values

DATA

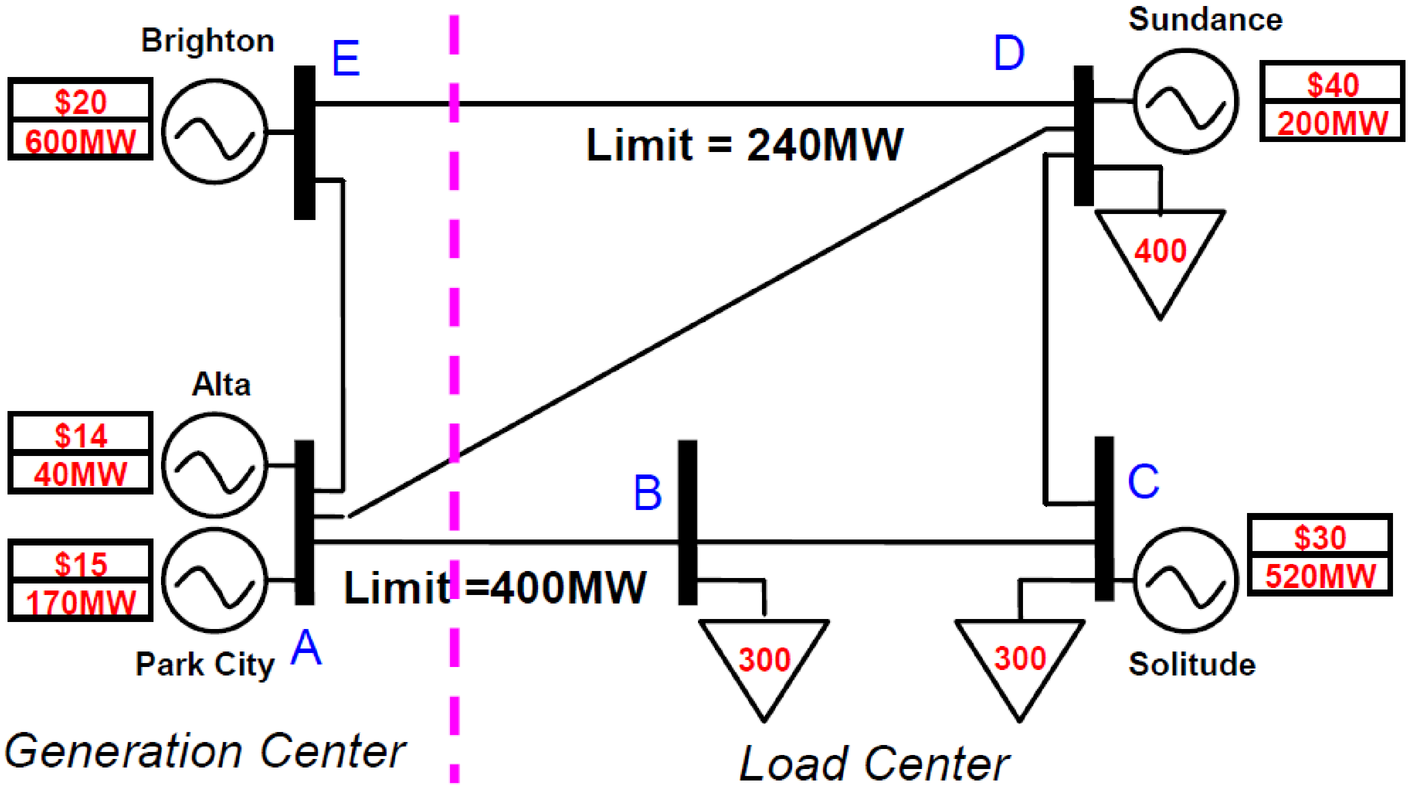

Fig. 1. The PJM 5-bus system.

F. Li and R. Bo, "Small test systems for power system economic studies," IEEE PES General Meeting, Minneapolis, MN, 2010, pp. 1-4.

```
$Title DC Optimal Power Flow (OPF) model

$ontext
-----------------------------------------------------------------------------------------
The current DC optimal power flow (OPF) model finds the optimal operating schedules of generating units considering transmission line constraints.
The objective function is defined as the total operating costs of generating units.
The model type is LP and is applied to a 5 bus PJM test system as described in the following reference:
F. Li and R. Bo, "Small test systems for power system economic studies," IEEE PES General Meeting, Minneapolis, MN, 2010, pp. 1-4.
doi: 10.1109/PES.2010.5589973
The proposed model is able to handle large scale set of data for practical power system transmission networks.

Inputs:
Generator costs + technical characteristics (min/max , connection point)
Network characteristics (reactances, demands, line flow limits)
Outputs:
local marginal prices (LMP)
Generators outputs
Line flows
Angle values
------------------------------------------------------------------------------------------

Contributed by Dr. Alireza Soroudi, University College Dublin, Dublin, Ireland.
email: alireza.soroudi@gmail.com
       alireza.soroudi@ucd.ie
$offtext

sets
bus  /1*5/
slack(bus) /1/
GenNo /Alta,ParkCity,Solitude,Sundance,Brighton/
scalars
Sbase /100/
;
alias(bus,node);

table GenData(GenNo,*)  Generating units characteristics
         b    pmin pmax
Alta     14   0    40
ParkCity 15   0    170
Solitude 30   0    520
Sundance 40   0    200
Brighton 10   0    600
;

* ---------------------------------------------------
set GBconect(bus,GenNo) connectivity index of each generating unit to each bus
/1    .    Alta
 1    .    ParkCity
 3    .    Solitude
 4    .    Sundance
```

```
 5    .    Brighton / ;

****************************************************
****************************************************
Table BusData(bus,*) Demands of each bus in MW
         Pd
2        300
3        300
4        400
;
****************************************************
set conex          Bus connectivity matrix
/
1    .    2
2    .    3
3    .    4
4    .    1
4    .    5
5    .    1
* -------------------------------------------------------
/;
conex(bus,node)$(conex(node,bus))=1;

table branch(bus,node,*)    Network technical characteristics
               x         Limit
1    .    2    0.0281    400
1    .    4    0.0304    1000
1    .    5    0.0064    1000
2    .    3    0.0108    1000
3    .    4    0.0297    1000
4    .    5    0.0297    240
* -------------------------------------------------------
 ;

branch(bus,node,'x')$(branch(bus,node,'x')=0)=branch(node,bus,'x');
branch(bus,node,'Limit')$(branch(bus,node,'Limit')=0)=branch(node,bus,'Limit');
branch(bus,node,'bij')$conex(bus,node) =1/branch(bus,node,'x');
*****************************************************
Variables
OF
Pij(bus,node)
Pg(GenNo)
delta(bus)
;

Equations
*********************************************
const1
const2
const3
;
*********************************************************************
const1(bus,node)$( conex(bus,node)) .. Pij(bus,node)=e= branch(bus,node,'bij')*(»
delta(bus)-delta(node));
const2(bus) .. +sum(GenNo$GBconect(bus,GenNo),Pg(GenNo))-BusData(bus,'pd')/Sbase»
=e=+sum(node$conex(node,bus),Pij(bus,node));
const3    .. OF=g=sum(GenNo,Pg(GenNo)*GenData(GenNo,'b')*Sbase);
```

```
model loadflow     /const1,const2,const3/;

Pg.lo(GenNo)=GenData(GenNo,'Pmin')/Sbase;
Pg.up(GenNo)=GenData(GenNo,'Pmax')/Sbase;
delta.up(bus)=pi;
delta.lo(bus)=-pi;
delta.fx(slack)=0;
Pij.up(bus,node)$((conex(bus,node)))=1* branch(bus,node,'Limit')/Sbase;
Pij.lo(bus,node)$((conex(bus,node)))=-1*branch(bus,node,'Limit')/Sbase;

solve loadflow minimizing OF using lp;
parameter report(bus,*);
report(bus,'Gen(MW)')= sum(GenNo$GBconect(bus,GenNo),Pg.l(GenNo))*sbase;
report(bus,'load(MW)')= BusData(bus,'pd');
report(bus,'LMP($/MWh)')=const2.m(bus)/sbase ;

display report,Pij.l;
```