pestpp-opt

April 29, 2019

1 Run PESTPP-OPT

In [1]:
```
import os
import shutil
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import flopy
import pyemu
```

```
flopy is installed in /Users/jeremyw/Dev/gw1876/activities_2day_mfm/notebooks/flopy
```

In [2]:
```
t_d = "template"
m_d = "master_opt"
```

In [3]:
```
pst = pyemu.Pst(os.path.join(t_d,"freyberg.pst"))
pst.write_par_summary_table(filename="none").sort_index()
```

Out[3]:

	type	transform	count	initial value	\
drncond_k00	drncond_k00	log	10	0	
flow	flow	log	1	0	
grhk3	grhk3	log	705	0	
grhk4	grhk4	log	705	0	
grhk5	grhk5	log	705	0	
grrech2	grrech2	log	705	0	
grrech3	grrech3	log	705	0	
grss3	grss3	log	705	0	
grss4	grss4	log	705	0	
grss5	grss5	log	705	0	
grstrt3	grstrt3	log	705	0	
grstrt4	grstrt4	log	705	0	
grstrt5	grstrt5	log	705	0	
grsy3	grsy3	log	705	0	
grsy4	grsy4	log	705	0	
grsy5	grsy5	log	705	0	
grvka3	grvka3	log	705	0	

grvka4	grvka4	log	705	0
grvka5	grvka5	log	705	0
hk6_cn	hk6_cn	log	1	0
hk7_cn	hk7_cn	log	1	0
hk8_cn	hk8_cn	log	1	0
pp_hk0	pp_hk0	log	32	0
pp_hk1	pp_hk1	log	32	0
pp_hk2	pp_hk2	log	32	0
pp_rech0	pp_rech0	log	32	0
pp_rech1	pp_rech1	log	32	0
pp_ss0	pp_ss0	log	32	0
pp_ss1	pp_ss1	log	32	0
pp_ss2	pp_ss2	log	32	0
pp_strt0	pp_strt0	log	32	0
pp_strt1	pp_strt1	log	32	0
pp_strt2	pp_strt2	log	32	0
pp_sy0	pp_sy0	log	32	0
pp_sy1	pp_sy1	log	32	0
pp_sy2	pp_sy2	log	32	0
pp_vka0	pp_vka0	log	32	0
pp_vka1	pp_vka1	log	32	0
pp_vka2	pp_vka2	log	32	0
rech4_cn	rech4_cn	log	1	0
rech5_cn	rech5_cn	log	1	-0.39794
ss6_cn	ss6_cn	log	1	0
ss7_cn	ss7_cn	log	1	0
ss8_cn	ss8_cn	log	1	0
strk	strk	log	40	0
strt6_cn	strt6_cn	log	1	0
strt7_cn	strt7_cn	log	1	0
strt8_cn	strt8_cn	log	1	0
sy6_cn	sy6_cn	log	1	0
sy7_cn	sy7_cn	log	1	0
sy8_cn	sy8_cn	log	1	0
vka6_cn	vka6_cn	log	1	0
vka7_cn	vka7_cn	log	1	0
vka8_cn	vka8_cn	log	1	0
welflux	welflux	log	2	0 to 0.176091
welflux_k02	welflux_k02	log	6	0

	upper bound	lower bound	standard deviation
drncond_k00	1	-1	0.5
flow	0.09691	-0.124939	0.0554622
grhk3	1	-1	0.5
grhk4	1	-1	0.5
grhk5	1	-1	0.5
grrech2	0.0413927	-0.0457575	0.0217875
grrech3	0.0413927	-0.0457575	0.0217875

grss3	1	-1	0.5
grss4	1	-1	0.5
grss5	1	-1	0.5
grstrt3	0.0211893	-0.0222764	0.0108664
grstrt4	0.0211893	-0.0222764	0.0108664
grstrt5	0.0211893	-0.0222764	0.0108664
grsy3	0.243038	-0.60206	0.211275
grsy4	0.243038	-0.60206	0.211275
grsy5	0.243038	-0.60206	0.211275
grvka3	1	-1	0.5
grvka4	1	-1	0.5
grvka5	1	-1	0.5
hk6_cn	1	-1	0.5
hk7_cn	1	-1	0.5
hk8_cn	1	-1	0.5
pp_hk0	1	-1	0.5
pp_hk1	1	-1	0.5
pp_hk2	1	-1	0.5
pp_rech0	0.0413927	-0.0457575	0.0217875
pp_rech1	0.0413927	-0.0457575	0.0217875
pp_ss0	1	-1	0.5
pp_ss1	1	-1	0.5
pp_ss2	1	-1	0.5
pp_strt0	0.0211893	-0.0222764	0.0108664
pp_strt1	0.0211893	-0.0222764	0.0108664
pp_strt2	0.0211893	-0.0222764	0.0108664
pp_sy0	0.243038	-0.60206	0.211275
pp_sy1	0.243038	-0.60206	0.211275
pp_sy2	0.243038	-0.60206	0.211275
pp_vka0	1	-1	0.5
pp_vka1	1	-1	0.5
pp_vka2	1	-1	0.5
rech4_cn	0.0791812	-0.09691	0.0440228
rech5_cn	-0.09691	-1	0.225772
ss6_cn	1	-1	0.5
ss7_cn	1	-1	0.5
ss8_cn	1	-1	0.5
strk	2	-2	1
strt6_cn	0.0211893	-0.0222764	0.0108664
strt7_cn	0.0211893	-0.0222764	0.0108664
strt8_cn	0.0211893	-0.0222764	0.0108664
sy6_cn	0.243038	-0.60206	0.211275
sy7_cn	0.243038	-0.60206	0.211275
sy8_cn	0.243038	-0.60206	0.211275
vka6_cn	1	-1	0.5
vka7_cn	1	-1	0.5
vka8_cn	1	-1	0.5
welflux	0.176091 to 0.30103	-0.30103 to 0	0.0752575 to 0.11928

```
welflux_k02                    1                  -1                    0.5
```

In [4]:
```
pst.pestpp_options = {}
#dvg = ["welflux_k02","welflux"]
dvg = ["welflux_k02"]
pst.pestpp_options["opt_dec_var_groups"] = dvg
pst.pestpp_options["opt_direction"] = "max"
```

In [5]:
```
par = pst.parameter_data
par.loc[:,"partrans"] = "fixed"

#turn off pumping in the scenario
par.loc["welflux_001","parlbnd"] = 0.0
par.loc["welflux_001","parval1"] = 0.0
dvg_pars = par.loc[par.pargp.apply(lambda x: x in dvg),"parnme"]
par.loc[dvg_pars,"partrans"] = "none"
par.loc[dvg_pars,"parlbnd"] = 0.0
par.loc[dvg_pars,"parubnd"] = 2.0
par.loc[dvg_pars,"parval1"] = 1.0

pst.rectify_pgroups()
pst.parameter_groups.loc[dvg,"inctyp"] = "absolute"
pst.parameter_groups.loc[dvg,"inctyp"] = "absolute"
pst.parameter_groups.loc[dvg,"derinc"] = 0.25

pst.parameter_groups.loc[dvg,:]
```

Out[5]:
```
                 pargpnme    inctyp  derinc  derinclb  forcen  derincmul  \
pargpnme
welflux_k02  welflux_k02  absolute    0.25       0.0  switch        2.0

                dermthd  splitthresh  splitreldiff splitaction  extra
pargpnme
welflux_k02  parabolic      0.00001           0.5     smaller    NaN
```

1.0.1 constraints

In [6]:
```
obs = pst.observation_data
obs.loc[:,"weight"] = 0.0
swgw_hist = obs.loc[obs.obsnme.apply(lambda x: "fa" in x and( "hw" in x or "tw" in x))
obs.loc[swgw_hist,:]
```

Out[6]:
```
                          obsnme       obsval  weight obgnme  extra
obsnme
fa_hw_19791230  fa_hw_19791230 -1289.170400     0.0  flaqx    NaN
fa_hw_19801229  fa_hw_19801229  -690.416200     0.0  flaqx    NaN
fa_tw_19791230  fa_tw_19791230  -573.743640     0.0  flaqx    NaN
fa_tw_19801229  fa_tw_19801229  -160.726405     0.0  flaqx    NaN
```

We need to change the obs group (obgnme) so that pestpp-opt will recognize these two model outputs as constraints. lets also assume that the sw-gw flux needs to be at least -1000

```
In [7]: obs.loc[swgw_hist,"obgnme"] = "less_than"
        obs.loc[swgw_hist,"weight"] = 1.0

        obs.loc[swgw_hist,"obsval"] = -300

        tot_abs_rate = ["flx_wells_19791230"]#,"flx_wells_19801229"]
        obs.loc[tot_abs_rate,"obgnme"] = "less_than"
        obs.loc[tot_abs_rate,"weight"] = 1.0
        obs.loc[tot_abs_rate,"obsval"] = -600.0
        pst.less_than_obs_constraints
```

```
Out[7]: obsnme
        fa_hw_19791230            fa_hw_19791230
        fa_hw_19801229            fa_hw_19801229
        fa_tw_19791230            fa_tw_19791230
        fa_tw_19801229            fa_tw_19801229
        flx_wells_19791230    flx_wells_19791230
        Name: obsnme, dtype: object
```

```
In [8]: pst.control_data.noptmax = 1
        pst.write(os.path.join(t_d,"freyberg_opt.pst"))
```

```
In [9]: pyemu.os_utils.start_slaves(t_d,"pestpp-opt","freyberg_opt.pst",num_slaves=10,master_d
```

```
In [10]: jco = pyemu.Jco.from_binary(os.path.join(m_d,"freyberg_opt.1.jcb")).to_dataframe().lo
         jco
```

```
Out[10]:                     wf0200090016  wf0200110013  wf0200200014  wf0200260010  \
         fa_hw_19791230         137.57200     126.32400      46.30000      21.90800
         fa_hw_19801229          22.58400      28.65600      12.03600      12.29200
         fa_tw_19791230           6.50728      14.53516      93.28136      92.42320
         fa_tw_19801229           4.10836       7.60104      15.29948      30.88604
         flx_wells_19791230    -150.00000    -150.00000    -150.00000    -150.00000

                              wf0200290006  wf0200340012
         fa_hw_19791230           18.12000        4.8320
         fa_hw_19801229           13.12800        3.3560
         fa_tw_19791230           71.84608       82.9612
         fa_tw_19801229           34.79872       17.5232
         flx_wells_19791230    -150.00000     -150.0000
```

```
In [11]: par_df = pyemu.pst_utils.read_parfile(os.path.join(m_d,"freyberg_opt.1.par"))
         print(par_df.loc[dvg_pars,"parval1"].sum())
         par_df.loc[dvg_pars,:]
```

```
8.1332977617072
```

Out[11]:

	parnme	parval1	scale	offset
parnme				
wf0200090016	wf0200090016	2.000000	1.0	0.0
wf0200110013	wf0200110013	2.000000	1.0	0.0
wf0200200014	wf0200200014	2.000000	1.0	0.0
wf0200260010	wf0200260010	0.133298	1.0	0.0
wf0200290006	wf0200290006	0.000000	1.0	0.0
wf0200340012	wf0200340012	2.000000	1.0	0.0

In [12]:
```
pst = pyemu.Pst(os.path.join(m_d,"freyberg_opt.pst"),resfile=os.path.join(m_d,"freybe
pst.res.loc[pst.nnz_obs_names,:]
```

Out[12]:

	name	group	measured	modelled	\
name					
fa_hw_19791230	fa_hw_19791230	less_than	-300.0	-699.3735	
fa_hw_19801229	fa_hw_19801229	less_than	-300.0	-714.4580	
fa_tw_19791230	fa_tw_19791230	less_than	-300.0	-407.7249	
fa_tw_19801229	fa_tw_19801229	less_than	-300.0	-299.7868	
flx_wells_19791230	flx_wells_19791230	less_than	-600.0	-1219.9948	

	residual	weight
name		
fa_hw_19791230	399.3735	1.0
fa_hw_19801229	414.4580	1.0
fa_tw_19791230	107.7249	1.0
fa_tw_19801229	-0.2132	1.0
flx_wells_19791230	619.9948	1.0

In [13]:
```
#todo chance constraints (fosm and en-based), well pars and constraints in scen perio
```

1.0.2 Opt under uncertainty part 1: FOSM chance constraints

In [14]:
```
pst.pestpp_options["opt_risk"] = 0.4
```

In [15]:
```
cn_pars = par.loc[par.pargp.apply(lambda x: "cn" in x),"parnme"]
cn_pars
```

Out[15]:

parnme	
hk6_cn	hk6_cn
hk7_cn	hk7_cn
hk8_cn	hk8_cn
rech4_cn	rech4_cn
rech5_cn	rech5_cn
ss6_cn	ss6_cn
ss7_cn	ss7_cn
ss8_cn	ss8_cn
strt6_cn	strt6_cn
strt7_cn	strt7_cn
strt8_cn	strt8_cn

```
sy6_cn        sy6_cn
sy7_cn        sy7_cn
sy8_cn        sy8_cn
vka6_cn      vka6_cn
vka7_cn      vka7_cn
vka8_cn      vka8_cn
Name: parnme, dtype: object
```

In [16]:
```
par = pst.parameter_data
par.loc[cn_pars,"partrans"] = "log"
pst.control_data.noptmax = 1
pst.write(os.path.join(t_d,"freyberg_opt_uu1.pst"))
pst.npar_adj
```

Out[16]: 23

In [17]:
```
pyemu.os_utils.start_slaves(t_d,"pestpp-opt","freyberg_opt_uu1.pst",num_slaves=20,mas
```

In [18]:
```
pst = pyemu.Pst(os.path.join(m_d,"freyberg_opt_uu1.pst"),resfile=os.path.join(m_d,"fr
pst.res.loc[pst.nnz_obs_names,:]
```

Out[18]:

	name	group	measured	modelled	\
name					
fa_hw_19791230	fa_hw_19791230	less_than	-300.0	-666.13442	
fa_hw_19801229	fa_hw_19801229	less_than	-300.0	-682.60800	
fa_tw_19791230	fa_tw_19791230	less_than	-300.0	-223.47050	
fa_tw_19801229	fa_tw_19801229	less_than	-300.0	-208.37540	
flx_wells_19791230	flx_wells_19791230	less_than	-600.0	-1586.33800	

	residual	weight
name		
fa_hw_19791230	366.13442	1.0
fa_hw_19801229	382.60800	1.0
fa_tw_19791230	-76.52950	1.0
fa_tw_19801229	-91.62460	1.0
flx_wells_19791230	986.33800	1.0

In [19]:
```
par_df = pyemu.pst_utils.read_parfile(os.path.join(m_d,"freyberg_opt_uu1.1.par"))
print(par_df.loc[dvg_pars,"parval1"].sum())
par_df.loc[dvg_pars,:]
```

```
10.575587155980312
```

Out[19]:

	parnme	parval1	scale	offset
parnme				
wf0200090016	wf0200090016	2.000000	1.0	0.0
wf0200110013	wf0200110013	2.000000	1.0	0.0
wf0200200014	wf0200200014	1.481006	1.0	0.0
wf0200260010	wf0200260010	1.094581	1.0	0.0
wf0200290006	wf0200290006	2.000000	1.0	0.0
wf0200340012	wf0200340012	2.000000	1.0	0.0

1.0.3 Opt under uncertainty part 2: ensemble-based chance constraints

```
In [20]: obs_df = pd.read_csv(os.path.join("master_prior_sweep","sweep_out.csv"),index_col=0)
         obs_df = obs_df.loc[obs_df.failed_flag==0,:]

In [21]: std = obs_df.std().loc[pst.nnz_obs_names]
         std

Out[21]: fa_hw_19791230        352.846322
         fa_hw_19801229        460.423786
         fa_tw_19791230        469.183099
         fa_tw_19801229        548.767448
         flx_wells_19791230    705.753519
         dtype: float64

In [22]: pst.observation_data.loc[pst.nnz_obs_names,"weight"] = std.loc[pst.nnz_obs_names]
         pst.pestpp_options["opt_std_weights"] = True
         pst.write(os.path.join(t_d,"freyberg_opt_uu2.pst"))

In [23]: pyemu.os_utils.start_slaves(t_d,"pestpp-opt","freyberg_opt_uu2.pst",num_slaves=10,mast

In [24]: par_df = pyemu.pst_utils.read_parfile(os.path.join(m_d,"freyberg_opt_uu2.1.par"))
         print(par_df.loc[dvg_pars,"parval1"].sum())
         par_df.loc[dvg_pars,:]

11.038907743747297


Out[24]:                      parnme   parval1  scale  offset
         parnme
         wf0200090016  wf0200090016  2.000000    1.0     0.0
         wf0200110013  wf0200110013  2.000000    1.0     0.0
         wf0200200014  wf0200200014  1.038908    1.0     0.0
         wf0200260010  wf0200260010  2.000000    1.0     0.0
         wf0200290006  wf0200290006  2.000000    1.0     0.0
         wf0200340012  wf0200340012  2.000000    1.0     0.0
```