Model Improvement

University of Wisconsin - Madison

May 20, 2015

1 Task Description

Based on the Task 4, the polar power-voltage formulation is selected as the most promise for quick and robust solutions to the ACOPF problems and to be investigated to improve the model. The task is implemented with the three solvers (KNITRO, CONOPT, IPOPTH) as chosen in the Task 4. This model analysis is undertaken with three data sets (IEEE 118 bus, Polish 3375 and Winter RTO data sets).

2 Model Analysis and Improvement

2.1 Notation

1. `ic=*`
   Initial condition for ACOPF problems and `*` refers to options to choose a different starting point.

2. `Time` : CPU time to solve ACOPF problems.

   - First : CPU time from generating starting points to a solution
   - Start : CPU time from starting points to a solution

3. `O.V` : Optimal objective value.

4. `M.S` : Model Status.

5. `S.S` : Solver Status.

2.2 Attempt for model improvement

1. Add different initial conditions for ACOPF problems

   - `ic=7`: [**DCLoss**] Real power and voltage angle values are initialized using a DCOPF model with line loss approximation (`--lineloss= 1.055`). Voltage magnitudes are initialized at 1.0. Reactive power is initialized using relevant equations from the AC transmission line model (applied to each line separately) and the initialized voltage magnitude and voltage angle values.

- `ic=8`: [**Matpower**] Use voltage magnitude, voltage angle, real power, and reactive power values given in Matpower solutions (if available), but it is not used for this task.
- `ic=9`: [**inputFile**] Use voltage magnitude, voltage angle, real power, and reactive power values given in the GDX file, but it is not used for this task.

2. Modify initial guesses

   - All initial conditions that are used for task 4 except for `ic=2` initialize line power flows using initialized voltage anlges and magnitudes as well.
   - `ic=6+` : Discard the objective function which minimzes the loss of the system for the Q-subproblem.
   - `ic=1+,3+` : Voltage angles are set to zero for `ic=1,3` where voltage angle variables are initialized using random draw between $-\pi$ and π.

2.3 Assessement of the ACOPF solution

Since IPOPTH is the only solver able to handle the D-curve constraint, CONOPT and KNTRO do not use the D-curve constraint for all test case. The Winter RTO is very large data set thus it is presented separately.

Solver: IPOPTH		POLAR				
		TIME		O.V	M.S	S.S
		First	Start			
case118	ic=0	1sec	1sec	129.66K$	Normal Completion	Locally Optimal
	ic=1+	1sec	1sec	129.66K$	Normal Completion	Locally Optimal
	ic=2	0.8sec	0.8sec	129.66K$	Normal Completion	Locally Optimal
	ic=3+	1sec	1sec	129.66K$	Normal Completion	Locally Optimal
	ic=4	2sec	1.5sec	129.66K$	Normal Completion	Locally Optimal
	ic=5	1.5sec	1sec	129.66K$	Normal Completion	Locally Optimal
	ic=6+	3sec	1sec	129.66K$	Normal Completion	Locally Optimal
	ic=7	1.5sec	1sec	129.66K$	Normal Completion	Locally Optimal

Solver: IPOPTH		POLAR				
		TIME		O.V	M.S	S.S
		First	Start			
case3375	ic=0	10sec	10sec	7.4099M$	Normal Completion	Locally Optimal
	ic=1+	1min20sec	1min20sec	7.4099M$	Normal Completion	Locally Optimal
	ic=2	9.5sec	9.5sec	7.4099M$	Normal Completion	Locally Optimal
	ic=3+	35sec	35sec	7.4099M$	Normal C ompletion	Locally Optimal
	ic=4	15sec	11.5sec	7.4099M$	Normal Completion	Locally Optimal
	ic=5	14sec	10.5sec	7.4099M$	Normal Completion	Locally Optimal
	ic=6+	infeas	infeas	infeas	Infeasibility in the decoupled ACOPF	
	ic=7	15sec	11sec	7.4099M$	Normal Completion	Locally Optimal

Solver: IPOPTH		POLAR		
		TIME		O.V
		First	Start	
case3375wp (with the D-curve)	ic=0	11sec	11sec	7.4110M$
	ic=1+	3min40sec	3min40sec	7.4110M$
	ic=2	35sec	35sec	7.4110M$
	ic=3+	17sec	17sec	7.4110M$
	ic=4	20sec	16sec	7.4110M$
	ic=5	22.5sec	18.5sec	7.4110M$
	ic=6+	infeas	infeas	infeas
	ic=7	19.5sec	11sec	7.4110M$

Table 1: ACOPF solution characteristic with IPOPTH solver

Solver: CONOPT		POLAR				
		TIME		O.V	M.S	S.S
		First	Start			
case118	ic=0	1sec	1sec	129.66K$	Normal Completion	Locally Optimal
	ic=1+	0.9sec	0.9sec	129.66K$	Normal Completion	Locally Optimal
	ic=2	1.5sec	1.5sec	129.66K$	Normal Completion	Locally Optimal
	ic=3+	1.4sec	1.4sec	129.66K$	Normal C ompletion	Locally Optimal
	ic=4	2sec	1.5sec	129.66K$	Normal Completion	Locally Optimal
	ic=5	2.3sec	1.8sec	129.66K$	Normal Completion	Locally Optimal
	ic=6+	3.5sec	1.5sec	129.66K$	Normal Completion	Locally Optimal
	ic=7	1.7sec	1sec	129.66K$	Normal Completion	Locally Optimal

Solver: CONOPT		POLAR				
		TIME		O.V	M.S	S.S
		First	Start			
case3375wp	ic=0	12min	12min	7.4099M$	Normal Completion	Locally Optimal
	ic=1+	6min30sec	6min30sec	7.4099M$	Normal Completion	Locally Optimal
	ic=2	12min	12min	7.4099M$	Normal Completion	Locally Optimal
	ic=3+	4min	4min	7.4099M$	Normal C ompletion	Locally Optimal
	ic=4	1min35sec	1min30sec	7.4099M$	Normal Completion	Locally Optimal
	ic=5	infeas	infeas	infeas	Normal Completion	Locally Infeasible
	ic=6+	infeas	infeas	7.4099M$	Infeasibility in the decoupled ACOPF	
	ic=7	1min51sec	1min47sec	7.4099M$	Normal Completion	Locally Optimal

Table 2: ACOPF solution characteristic with CONOPT solver

Solver: KNITRO		POLAR				
		TIME		O.V	M.S	S.S
		First	Start			
case118	ic=0	0.8sec	0.8sec	129.66K$	Normal Completion	Locally Optimal
	ic=1+	0.8sec	0.8sec	129.66K$	Normal Completion	Locally Optimal
	ic=2	0.7sec	0.7sec	129.66K$	Normal Completion	Locally Optimal
	ic=3+	1.5sec	1sec	129.66K$	Normal C ompletion	Locally Optimal
	ic=4	1.5sec	1sec	129.66K$	Normal Completion	Locally Optimal
	ic=5	1.7sec	1.2sec	129.66K$	Normal Completion	Locally Optimal
	ic=6+	3.2sec	1sec	129.66K$	Normal Completion	Locally Optimal
	ic=7	1.8sec	1.2sec	129.66K$	Normal Completion	Locally Optimal

Solver: KNITRO		POLAR				
		TIME		O.V	M.S	S.S
		First	Start			
case3375wp	ic=0	10sec	10sec	7.4099M$	Normal Completion	Locally Optimal
	ic=1+	3min10sec	3min10sec	7.4099M$	Normal Completion	Locally Infeasible
	ic=2	12sec	12sec	7.4099M$	Normal Completion	Locally Optimal
	ic=3+	1min50sec	1min50sec	7.4099M$	Normal C ompletion	Locally Optimal
	ic=4	18sec	13sec	7.4099M$	Normal Completion	Locally Optimal
	ic=5	17sec	12sec	7.4099M$	Normal Completion	Locally Optimal
	ic=6+	infeas	infeas	7.4099M$	Infeasibility in the decoupled ACOPF	
	ic=7	15sec	11sec	7.4099M$	Normal Completion	Locally Optimal

Table 3: ACOPF solution characteristic with KNITRO solver

- `Observation`
  - CONOPT solver fails to obtain a feasible point with the D-curve constraint for the `case3375wp` by saying that initial function value is too large and so does KNITRO.
  - IPOPTH solver is suited well for ACOPF problems by providing fast and robust convergence to an optimal solution even with the D-curve constraint.
  - With modified initial condition `ic=1+,3+`, the improved performance with more robust convergence is shown. However, the added initial condition `ic=7` shows no significant improvement.
  - By combining the result of the Task 4, It is difficult to judge which initial condition is the best for the ACOPF problem since they are dependent on both the system and solver. Among modified initial conditions, `ic=0,2` show quite great performances in terms of the CPU time including both `(First, Start)`, and robust convergence for all test case with IPOPTH and KNITRO.

2.4 Winter RTO data

Solver : IPOPTH, KNITRO		POLAR			
		TIME (First Start)	O.V	M.S	S.S
Winter RTO	ic=0	infeas	infeas	Normal Completion	Locally Infeasible
	ic=1+	infeas	infeas	Normal Completion	Locally Infeasible
	ic=2	infeas	infeas	Normal Completion	Locally Infeasible
	ic=3+	infeas	infeas	Normal Completion	Locally Infeasible
	ic=4	infeas	infeas	Normal Completion	Locally Infeasible
	ic=5	infeas	infeas	Normal Completion	Locally Infeasible
	ic=6+	infeas	infeas	Infeasibility in the decoupled ACOPF	
	ic=7	infeas	infeas	Normal Completion	Locally Infeasible

Table 4: ACOPF solution characteristic with Winter RTO

- `Observation`
  - For the Winter RTO case, none of the initial conditions happen to provide an optimal solution. This suggests that very large system is a hard problem and good initial conditions would be necessary to find a feasible point.
  - Further investigaton, such as data/equation modifications or improved ACOPF formulation, would be necessary to make this data set more suitable for the ACOPF (Future work).

3 Conclusion

- `Model`
  - Winter RTP data set is very large system and it is shown that the ACOPF model with the polar power-voltage formulation is difficult to solve it without given good starting points.
  - It would be worthwhile to take a detailed inspection for the rectangular current-voltage formulation that was able to find a different local solution(Future work).

- `Initial guess`
  - On the whole, `ic=0,2` provide the fastest convergence for the ACOPF with/without the D-curve constraint.
  - Better choice of the voltage angle is required for the `ic=1,3`. The more sophisticated way to choose voltage angles is to initialize the difference of the voltage between $-\frac{\pi}{3}$ and $\frac{\pi}{3}$ arbitrarily (Future work).
- `Solver`
  - Based on the results above, IPOPTH is the most promising solver for ACOPF problems among three solvers. It is faster and more robust to find an optimal solution than other solvers.
  - Modification to handle the D-curve constraint should be added for CONOPT and KNITRO.
  - It would be necessary for the Winter RTP data and improvement of the model to use different option files(Future work).

Appendix A: UC AC models

Similar to the SLP models, using gams option `nlp=ipopth` or `nlp=knitro` which changes the NLP subsolver from the default CONOPT solver is recommended it tends to reduce solution time. Table 5 compares solution times of the three UC AC models in the model archive for the 6 RTS96 files (all of which provide 24 hour demand data), while Table 6 displays the objective function from those models.

Case	UC Polar	UC Rect	UC IV
rts96_winter_wday	16.637s	36.347s	Locally infeasible
rts96_winter_wend	17.054s	35.363s	Locally infeasible
rts96_summer_wday	17.201s	36.672s	Locally infeasible
rts96_summer_wend	14.196s	38.787s	Locally infeasible
rts96_spring_wday	16.502s	39.645s	Locally infeasible
rts96_spring_wend	13.219s	55.203s	Locally infeasible

Table 5: UC AC Performance times

Case	UC Polar	UC Rect	UC IV
rts96_winter_wday	8027942.8084	8027442.1786	Locally infeasible
rts96_winter_wend	7348662.7043	7348163.7214	Locally infeasible
rts96_summer_wday	8161451.9444	8160920.1585	Locally infeasible
rts96_summer_wend	7514637.1255	7514138.2306	Locally infeasible
rts96_spring_wday	7850012.3767	7849510.2250	Locally infeasible
rts96_spring_wend	7332762.8914	7332294.0353	Locally infeasible

Table 6: UC AC Objective function values

While the polar formulation consistently performs better time-wise, the rectangular formulation occasionally returned a slightly improved solution 4 of 6 times. The IV formulation failed to return a solution and converged to a locally infeasible point. The IV formulation was tested using nlp subsolvers knitro and conopt with similar convergence problems.

Appendix B: Large scale solution finding

B.1 Improving initial conditions for large scale models

When considering large scale datasets in the AC models, regular solution practices may be insufficient in finding solutions. Large-scale AC models are much harder, if not impossible to solve without good initial conditions. This section discusses the methodology used in solution finding for the large scale Summer and Winter test cases. While the basic gist is outlined in Procedure 1, Sections B.1.1 and B.1.2 provide further details about the process.

Procedure 1: Feasibility methodology

$(\tilde{P}, \tilde{F}^P, \tilde{\theta}, U) \leftarrow$ Solve `UC_DC --lineloss=1.055`
Loop until convergence
   $(P, Q, F^P, F^Q, \theta, V) \leftarrow$ Solve `polar_acopf(`$\tilde{P}, \tilde{F}^P, \tilde{\theta}, U$`)`
end

B.1.1 24 hour UC_DC model

The first step is to first produce a reasonable generator commitment profile that would span across the 24 hour planning period. The solution provided by the unit commitment DC approximation model includes time dependent constraints such as ramping, minimum up and down time, and the option `--lineloss=1.055` approximates line loss by increasing demand by 5.5%, which is the average maximum estimate we see in small test cases. This avoids the gargantuan task of solving a unit commitment AC model while still solving an ACOPF model that realistically accounts for time dependency constraints from period to period. Using the lineloss option allows us to account for potential losses in the unit commitment model, and avoids overly "tight" initial conditions. This simple change proved to be rather effective by providing much better starting points for the active power generation variable P and unit commitment profile U.

Given the complexity of the dataset, there were issues with numerical inaccuracies in the 24 hour UC_DC model, whereby the "integer solution" was not within the tolerance of the "final solution". This was further confirmed by CPLEX options `(MIPKappStats=2, quality=1)` which provide information on the conditioning of the matrices and numerical accuracy. To fix this, a combination of CPLEX model options (`numbericalemphasis=1, scaind=1, mipemphasis=2`), and manual scaling of parameters and variables where appropriate, were found to be useful at times. Different datasets and model options required different combinations, a list of useful options is shown below.

Useful CPLEX options

- `threads=#`
- `names=no`
- `quality=1`
- `mipkappastats=2`

- `mipemphasis=2`
  Emphasis on optimality
- `bardisplay=2`
  Additional printouts
- `scaind=1`
  Aggressive scaling
- `numericalemphasis=1`
  Numerical precision emphasis

B.1.2 Feasible AC solutions

The solution of the 24 hour UCDC model provides a unit commitment profile that can be used in single time period ACOPF models, as well as starting points for bus angles and active power. Finding feasibility in large scale ACOPF models is very challenging, even with the improved DCOPF solutions that approximate line loss. As such, multiple iterations of the polar_acopf model may be needed in order to find AC solutions. Tests with different solvers resulted in the conclusion that a combination of different solvers was possibly needed for this task, and the final solutions for Winter and Summer were obtained using both KNITRO and CONOPT.

On the outset, KNITRO consistently performed better in finding feasible solutions from the starting points provided in Section B.1.1. A closer look at the solutions however revealed inconsistencies that could be attributed to numerical accuracy issues. Specifically, the solution was usually comprised of many small values, oftentimes $< 1e^{-8}$. CONOPT on the other hand did not fare well without good starting values for reactive power, but excelled over KNITRO in providing reliable and more numerically accurate solutions. In addition, CONOPT is able to take user provided starting points, so provided with good initial conditions, CONOPT was generally found to be a good choice for solution refinement.

Due to the general difficulty of the model however, there was no one size fits all when it came to using CONOPT for solution refinement. Eliminating noisy values ($< 1e^{-8}$) from the KNITRO solution sometimes helped, but most of the solutions were obtained using a combination of CONOPT options, as listed below. Like in the UCDC case, the need for options varied based on dataset and even timeperiod. The one consistent thread throughout however, was that cases with high active demand (i.e. greater stress on the system) were typically alot more difficult to solve, with the solvers encountering problems maintaining feasibility during the solution process.

Useful CONOPT options

- `LMMXSF=1`
  Method used to determine step in Phase 0. 1=method based on bending.

- `LMMXST=1`
  Method used to determine step length while tightening tolerances.

- `RVHESS=15`
  Memory factor for Hessian generation

- `lsanrm=t`
  Use steepest edge instead of steepest descent

- `lsusdf=0`
  Flag for forcing defined variables into basis

- `lstcrs=t`
  Triangular crashed turned on

- `lslack=t`
  Use triangular crash procedure and select initial basis as the crash variables and slacks

Also helpful in finding feasible points is to solve the AC model with a simplified objective function, that is using options `--obj=linear` or `--obj=0`, which solve the AC model with a simplified linear and 0 objective function respectively. This implicitly encourages the solver to concentrate more on feasibility than optimality. However, more research needs to be done to consider its effect on the solution space