Heuristic MINLP for Solving Optimal Power Flow Problems

Carleton Coffrin - Hassan Hijazi
NICTA Optimization Research Group

March 20, 2014

1 Introduction

The call for participation to the *Application of Modern Heuristic Optimization Algorithms for Solving Optimal Power Flow Problems* opens with,

> *"Heuristic optimization has undergone significant developments in recent years. By using different novel mechanisms for improved search exploration and exploitation, modern heuristic optimization tools have demonstrated a great promise for solving some real world problems, whose mathematical complexity prevents thus far the use of classical optimization algorithms."*

As a research group studying all forms of optimization, we strive to develop a deep understanding of the optimization landscape and to articulate the strengths and weaknesses of different approaches and technologies. Recognizing the competition is bore from recent advances in modern heuristic optimization tools such as evolutionary algorithms (EA) and metaheuristics (MH), we sought to take a different approach and attempt to solve the competition problems with classic off-the-shelf mathematical programming technologies. Comparing these technologies to other state-of-the-art methods (such as EA, and MH) provides insight for improving these general purpose solvers. The competition results will provide invaluable information on the overall viability of our proposed off-the-shelf mathematical programming solutions. However, regardless of the competition outcome, we hope that our submission has intrinsic value as a baseline for what is possible with current mathematical programming tools.

The remainder of this report documents how we went about modeling the competition problems as Mixed Integer NonLinear Programs (MINLPs) and how we integrated those MINLPs into the test bed infrastructure. We wrap-up with some general observations we made while preparing this competition submission.

1.1 Building the Models

The competition is comprised of three related optimization problems. Line loss minimization at a wind farm (WPP), line loss minimization (ORPD), and fuel cost minimization (OARPD). Although these problems share a core set of common constraints, each is slightly different and requires its own MINLP. The precise specification of each problem can be found in the following models:

1. Wind Farm Line Loss Minimization (WPP) - model/ACModel_ll_wf_soft.mod
2. Line Loss Minimization (ORPD) - model/ACModel_ll_sc.mod
3. Fuel Cost Minimization (OARPD) - model/ACModel_opf_sc_active_bounds.mod

The models are implemented using the mathematical programming language AMPL [3]. In each of these optimization problems, we found that the supplied MatPower files (i.e. the ".m" files in input_data), where insufficient for capturing a complete mathematical model of each of the competition problems. In all cases, details relating to bus shunt and transformer steps were missing. Additionally, in the case of the WPP problem, the daily *q_ref* value seems to be inconsistent with the test bed evaluation.[1] To formulate our MINLPs, we extracted the remaining required input data from the *mpc* and *proc* runtime variables and appended the provided MatPower cases to include the necessary information. We then built a translation tool *input_data/mp_opf2dat.py* to covert the augmented MatPower files into a data format that AMPL can read. These AMPL input data files are the ".dat" files in the input_data directory. Extreme care and testing was taken to ensure that the test bed's evaluation function and our AMPL models were solving the same problem.

1.2 Solving Technology

The AMPL modeling language is solver agnostic and selecting an appropriate solver is critical. Our previous experience suggested that MINLP solver Bonmin [1], which utilizes IPOPT [6] for solving non-linear programs (NLPs), would be an appropriate solver for these MINLPs. Both of these solvers are freely available via the coin-or project [2], and their admissible use in the competition was confirmed by Dr. Rueda [5]. Based on the given problem sizes we set IPOPTs iteration limit to 300 and Bonmin's termination criteria to an optimality gap of 0.05%. The remaining solver parameters are left at their default values. Both Bonmin and IPOPT are deterministic algorithms, hence our submission returns the same solution on all 31 replicate runs with slight variants in runtime for operating system overhead. The experiments were conducted on a single thread in a Dell PowerEdge R415 with 64GB memory and two AMD Opteron 4226 2.7GHz processors.

It is important to note that this solution approach is in fact a heuristic method for solving these MINLPs. IPOPT converges to a local stationary point and provides no guarantees of optimality for non-convex problems (such as, AC power flows) and hence the branch and bound algorithm for solving MINLPs implemented in Bonmin cannot provide global optimality guarantees neither. Despite these limitations, our previous experience has indicated that Bonmin and IPOPT provide high quality solutions to MINLPs arising in power systems. We are looking forward to seeing if this experience continues to hold true on these competition problems.

1.3 Test Bed Integration

Our test bed implementation "nicta.m" consists of three core steps,

1. Selecting the appropriate MINPL model for the problem at hand.
2. Calling AMPL via the command line interface.
3. Parsing the AMPL results and evaluating the solution in the test bed.

The bulk of the optimization is conducted by the general-purpose tools Bonmin and IPOPT. Unfortunately, this external solution approach yields only the final solution, rather than a trace of solution improvements. To follow the spirit of the competition in measuring total function evaluations and iterative solution improvements, we delay the introduction of the the Bonmin solution by I test bed solution evaluations, where I is the number of NLP iterations Bonmin used in reaching the solution. This provides a rough estimation for what a complete trace of solution improvements would be.

This behavior is implemented by having the population evaluation loop evaluate a random initialized solution until the I^{th} iteration, at which point, the solution produced by Bonmin is introduced. This Bonmin solution remains unmodified for 100 additional evaluations before a short circuiting behavior jumps to the last evaluation step, in the interest of agile development and regression testing.[2]

[1] However, the illustration in the competition guidelines does are appear to be consistent with the test bed evaluation.

[2] We projected that running the evaluation function the designated 57.66 million times required for the complete test bed evaluation would require over 7 days of serial computation on our hardware.

2 Test Bed Observations

While preparing this submission for the competition we made several observations about the test bed and test problems. In this section we document those observations and discuss how they affected our final submission.

1. As documented, the test bed does not enforce the continuous variable bounds as specified in the *x_min* and *x_max* vectors. We observed that violating these bounds may improve the fitness value. Dr. Rueda [4] suggested that we should satisfy these bounds, and hence, at the expense of a better fitness value, we enforce the variable bounds whenever possible.

2. If the continuous variable bounds are ignored, we noticed in the OARPD problem, it can be advantageous to set some generators active power injection to negative values in systems 118 and 300. As noted above, we do not allow such solutions but the test bed accepts these solutions without complaint.

3. We noticed that there is some times a discrepancy between the variable bounds as specified by *x_min* and *x_max* vectors and the MatPower file. For example, the bus voltage in x is often 1.0 ± 0.05 while it is 1.0 ± 0.06 in the MatPower file. When there is ambiguity, we adopt the bounds provided in the MatPower file.

4. We conjecture that several of the contingency cases are infeasible in light of the problem and vairable constraints. This conflicts with the competition guidelines, which suggest all of the test cases are solvable. A detailed analysis is provided in Section 3.

5. The "main_comment.m" file states *"an intervention scheme ensures that tailing contingencies are bypassed after conducting constraint handling as soon as any violation has been detected for intermediate contingencies. This strongly benefits the computational efficiency and quality of solutions for a given number of function evaluations while ensuring comparability between different implementations."* Although this behavior was designed to increase computational efficiency, it also allows the algorithm to avoid solving the contingencies entirely, by introducing a very small constraint violation into the base case. This point was discussed at length with Mr. Wildenhues with specific example from the 118 network [7, 8]. The ultimate conclusion was, *"only the achieved final fitness value regardless of the feasibility status will be considered."* This led us to design an algorithm that would avoid solving the contingencies in the interest of achieving the lowest possible fitness value.

6. Relating to the previous point. Due to the large objective values in the OARPD problem, it is quite easy for small constraint violations penalties to hide in the least significant digits of the fitness value.

7. The runtime of the test bed evaluation function greatly over shadows that of Bonmin. The cumulative runtime for Bonmin to produce solutions to all of the 102 competition problems is less than 1.5 minutes. On the other hand, simulating the function evaluations inside of the test bed (as discussed in Section 1.3) takes around 10.5 minutes on the same hardware.

3 Instance Data Analysis

Before solving the Line Loss Minimization (ORPD) and Fuel Cost Minimization (OARPD) problems in their entirety, we considered some relaxations of the problems to get an impression for the range of possible fitness values. We employed three relaxation approaches: (1) considering each contingency as an isolated optimization problem; (2) relaxing the discrete variables to continuous ones, so the problem can be solved quickly by a numerical method (such as IPOPT); (3) relaxing the problem constraints (e.g. reactive injection, voltage bounds, line capacities, ...) using a penalty function [3].

Table 1 summarizes the results of solving the ORPD and OARPD problems with relaxations 1 and 2 (called Relaxed) and relaxations 1, 2, and 3 (called Relaxed-λ) using IPOPT. The dashed lines "—" indicate

[3] Introducing constraint violation variables and chaining the objective to minimize constraint violations.

that IPOPT had difficulty finding a feasible solution. The table shows that there is some difficulty solving contingencies in systems 57 and 300. Interestingly the Relaxed-λ always converges to a saddle point in IPOPT, however, in each case when Relaxed was infeasible, the objective value of Relaxed-λ is greater than 0, indicating that not all of the constraints could be satisfied. Table 2 preforms a similar analysis on the WPP problem and suggests that scenarios 49 through 56 may be infeasible. The results for Relaxed and Relaxed-λ do not prove that these problems are infeasible, but they provide us with a strong indication, which is consistent with observations made in the test bed.

4 Closing Remarks

In conclusion of this report, we would like to thank the organizers for preparing this competition and for answering Carleton's abundant questions about the test bed. Comparing power system optimization algorithms on a consistent collection of real world problems is a valuable exercise in power system research, and we hope this competition will continue on as an annual exercise maintaining a consistent collection of high quality solutions and algorithms for power system optimization. Finally, as discussed above, we have made several design choices in the implementation of our competition submission. If the organizers feel any of these choices were not in the *spirit* of the competition, we will be happy to make any requested modifications.

System	Contingency	Relaxed	Relaxed-λ
ORPD			
57	0	24.27	0.00
57	1	—	0.17
57	2	—	0.04
118	0	114.68	0.00
118	1	117.42	0.00
118	2	117.03	0.00
118	3	116.19	0.00
118	4	115.14	0.00
300	0	380.62	0.00
300	1	—	40.09
300	2	—	9.07
300	3	—	29.90
OARPD			
57	0	41677.34	0.00
57	1	42646.77	0.00
57	2	—	0.04
118	0	134953.44	0.00
118	1	135000.48	0.00
118	2	134981.64	0.00
118	3	134991.10	0.00
118	4	134956.31	0.00
300	0	720198.43	0.00
300	1	—	1.35
300	2	—	0.08
300	3	721851.51	0.00

Table 1: Feasibility investigation of ORPD and OARPD Problems.

System	Scenario	Relaxed	Relaxed-λ
WPP			
41	1	1.29	0.00
...			
41	48	1.45	0.00
41	49	—	0.01
41	50	—	0.04
41	51	—	0.05
41	52	—	0.06
41	53	—	0.06
41	54	—	0.06
41	55	—	0.06
41	56	—	0.04
41	57	1.09	0.00
...			
41	96	1.49	0.00

Table 2: Feasibility investigation of WPP Scenarios.

References

[1] P. Bonami, L. T. Biegler, A. R. Conn, G. Cornuejols, I. E. Grossmann, C. D. Laird, J. Lee, A. Lodi, F. Margot, N. Sawaya, and A. Wächter. An algorithmic framework for convex mixed integer nonlinear programs. *Discrete Optimization*, 5(2):186 – 204, 2008.

[2] Coin-or (common infrastructure for operations research). Available at `http://www.coin-or.org`.

[3] R. Fourer, D. M. Gay, and B. Kernighan. Algorithms and model formulations in mathematical programming. chapter AMPL: A Mathematical Programming Language, pages 150–151. Springer-Verlag New York, Inc., New York, NY, USA, 1989.

[4] Jose L. Rueda. Subject: Re: REMINDER: IEEE PES Competition on Modern Heuristic Optimization Algorithms for Solving OPF. e-mail communication, March 2014.

[5] Jose L. Rueda. Subject: WG: OPF Competition. e-mail communication, January 2014.

[6] A. Wächter and L. T. Biegler. On the implementation of a primal-dual interior point filter line search algorithm for large-scale nonlinear programming. *Mathematical Programming*, 106(1):25–57, 2006.

[7] Sebastian Wildenhues. Subject: Re: Another Interesting Case. e-mail communication, February 2014.

[8] Sebastian Wildenhues. Subject: Re: Test Bed - V14. e-mail communication, February 2014.