# Parallel Implementation of AC Optimal Power Flow and Time Constrained Optimal Power Flow using High Performance Computing


Alex Werner
*Dept of Mathematics*
*University of Wisconsin-Platteville*
Platteville, USA
werneral@uwplatt.edu

Kapil Duwadi
*Dept ElectricalEngineering and ComputerScience*
*South Dakota StateUniversity*
Brookings, USA
kapil.duwadi@jacks.sdstate.edu

Nicholas Stegmeier
*Dept of Mathematics*
*South Dakota StateUniversity*
Brookings, USA
nicholas.stegmeier@sdstate.edu

Timothy M. Hansen
*Dept of electrical engineering and computer science*
*South Dakota StateUniversity*
Brookings, USA
timothy.hansen@sdstate.edu

Jung-Han Kimn
*Dept of Mathematics*
*South Dakota State University*
Brookings, USA
jung-han.kimn@sdstate.edu



***Abstract*—One of the major challenge to maintain an economically efficient, secure, and reliable power system operation is to find the solution to the non-linear AC optimal power flow (ACOPF) and time-constrained optimal power flow problems (TCOPF) as fast and accurately as possible. This paper exploits a scalable parallel implementation of ACOPF and TCOPF that can be solved independently by multiple CPU cores with the aim of reducing computation time. Parallel performance of the ACOPF implementation is evaluated with strong scaling tests on the standard IEEE 118 bus, 300 bus and 1354 bus test cases. An initial parallel implementation of TCOPF is also presented, along with numerical results for the standard IEEE 5 bus test case. Initial results indicate that the parallel ACOPF implementation is scalable, with additional performance possible by optimizing the parallel preconditioner. The parallel TCOPF implementation provides realistic results, along with promising parallel performance which will be explored further for larger test cases in the future.**



***Index Terms*—ACOPF, DMNetwork, high performance computing, PETSc, scalability, primal-dual interior point method,time-constrained optimal power flow**


## I. Introduction

THE penetration of distributed energy resources, especially from renewable sources, is rising globally. For example, in the U.S., renewable energy sources provided 17% of total electricity generation in 2017 [1]. Renewable portfolio standards require independent power producers to have certain percentages of electricity generation from renewable sources, e.g., it is required to have 30% electricity generation from renewable sources in Hawaii and 33% in California by 2020 [2]. The stochastic nature of the generation profile from most of renewable sources pose a challenge to maintaining a reliable, secure, and economically efficient power system operation. In light of these circumstances, currently a major problem is to find the optimal operating state of all generating sources under varying load and generation conditions to maintain reliable, secure, and efficient power system operation. The problem of finding optimal dispatch of generators that satisfies the underlying network and other constraints is termed AC optimal power flow. The Federal Energy Regulatory Commission estimates that a 5% increase in ACOPF solution accuracy from a 2009 base case can result in up to $20 billion and $90 billion savings per year in the US and worldwide, respectively. ACOPF minimizes a cost function with respect to the vector of continuous optimization variables, such as nodal voltages, generator active power, generator reactive power, etc. — the details of which can be found in [3]. The solution to ACOPF also satisfies constraints such as line limits, generation (both active and reactive power) limits, and voltage limits, among other quantities. While ACOPF gives the best operating point for a given instant in time, an extended problem such as *time-constrained* optimal power flow incorporates constraints that are coupled through time to give optimal operating points during each specified time interval. Examples of such time-coupled constraints are generater ramping constraints, and minimum generator up-time and down-time. TCOPF has been formulated in [4]–[6]. Both ACOPF and TCOPF are large-scale, non-linear optimization problems used for real power system operation by independent system operators. Utilizing high-performance computing by splitting the problem into multiple cores can significantly reduce the solution time, which would improve efficiency of power system market operation. In this paper, the major contributions are,


This material is based upon work supported by the National Science Foundation under Grant Nos. 1559978 and ECCS-1608722.



978-1-7281-0554-3/19/$31.00 ©2019 IEEE

- parallel implementation of ACOPF using PETCs and DMNetwork tested in standard IEEE 118 bus, 300 bus and 1354 bus test cases; and
- parallel implementation of TCOPF tested on the standard IEEE 5 bus test case, but general enough to be used in larger sized systems.[1]

The remaining sections of the paper are organized as follows. Section II describes relevant works in this field. Section III describes the formulation of ACOPF and TCOPF as a non-linear optimization problem. Details about the parallel implementation of ACOPF and TCOPF using PETSc and DMNetwork are provided in Section IV. Section V discusses the different power system test cases that the parallel implementations are tested on, including optimization parameters. Section VI presents the simulation results and analysis on the parallel performance (e.g., execution time, speedup, and solution accuracy) of the developed algorithms for solving ACOPF and TCOPF problems. Finally, conclusions are drawn in Section VII, and future work is discussed in Section VIII.

## II. Related work

High performance computing is gaining popularity in power system applications, such as optimization, real-time control and simulation, and integrated transmission-distribution simulations [7]–[9]. In an approach investigated by [10], [11], a given power system network is decomposed into multiple overlapping regions, each having regional optimal power flow problems that can be solved by multiple cores. This would limit the number of overlapping regions depending upon the nature of constraints, especially in case of constraints that involve the core variables from both overlapping regions and could limit the number of cores that can be used. Moreover, forming distributed optimal power flow problems in the case of TCOPF may be even more challenging, and in such a case parallelizing over time could be more effective. Tools such as DMNetwork (implemented in PETCs) and PLASMO (Platform for Scalable Modeling and Optimization) that use graph based abstraction for model representation and interaction between submodels are also becoming popular for solving optimal power flow problems [12]. In this paper, scalable parallel implementations for ACOPF and TCOPF optimization problems are explored using the Portable Extensible Toolkit for Scientific Computation (PETSc), especially by leveraging PETSC's DMNetwork library focusing on large-scale power system networks. The underlying algorithm used to solve an optimization problem greatly affects the performance of a parallel implementation. Gradient line search, interior point method, augmented Lagrangian, and sequential quadratic programming are some of the industrially popular algorithms to solve OPF problems [13]. However, not all of these algorithms are equally suitable for parallel implementations using high performance computing. In this paper, the primal-dual interior point method (PDIPM) is chosen as the core algorithm to solve ACOPF and TCOPF. A major advantage of the PDIPM approach [14] compared to other algorithms is its insensitivity to problem size regarding the number of iterations required to reach to the optimal solution. Solving non-linear optimization problems using PDIPM essentially requires solving a linear system of equations, which is very suitable for parallel implementation in multiple cores using high performance computing.

[1]Source code available at https://github.com/alexWerner/SDStateHPCREU/tree/master/network

## III. Formulation of ACOPF and TCOPF

Consider a transmission network with $N_{\mathcal{B}}$ buses collected in set $\mathcal{B} := \{1, 2, \ldots, N_{\mathcal{B}}\}$ and $N_{\mathcal{L}}$ transmission lines collected in set $\mathcal{L} \subset \mathcal{B} \times \mathcal{B}$. Let $\boldsymbol{Y} \in \mathbb{C}^{N_{\mathcal{B}} \times N_{\mathcal{B}}}$ be the admittance matrix of the transmission network. Note, $Y_{ij} = 0, ij \in \mathcal{B}$, if there is no transmission line connecting node $i$ and $j$, and $Y_{ij} = Y_{ji}$. In a real power system network, the matrix $\boldsymbol{Y}$ is sparse because $N_{\mathcal{L}} << N_{\mathcal{B}} \times N_{\mathcal{B}}$.

Let $\boldsymbol{v}$ and $\boldsymbol{s}$ be the column matrices of complex voltage and net injected complex power at all buses[2]. The relation between $\boldsymbol{v}$ and $\boldsymbol{s}$ is given by (1). This can be derived from the definition of complex power in power system which is just the product of complex voltage with complex conjugate current. In this case, complex current simply replaced by product of admittance and voltage.

$$\boldsymbol{s} = \boldsymbol{p} + j\boldsymbol{q} = \boldsymbol{v}(\boldsymbol{Y}\boldsymbol{v})^* \tag{1}$$

where $\boldsymbol{p}$ and $\boldsymbol{q}$ represent the column vectors of real and imaginary parts of the complex power at all buses, respectively (also known as active power and reactive power). Note, the complex voltage at bus $k \in \mathcal{B}$, i.e., $v_k$, can be expressed in terms of its magnitude $u_k$ and angle $\theta_k$ by (2).

$$v_k = u_k \angle \theta_k \tag{2}$$

Let us denote the column vectors of voltage magnitudes and angles as $\boldsymbol{u}$ and $\boldsymbol{\theta} \in \mathbb{R}^{N_{\mathcal{B}} \times 1}$, respectively. The voltage angle at any bus is with respect to the angle of the single reference bus (typically index 1), and its voltage angle will be zero. Furthermore, active power generation sources are indexed by the set $\mathcal{P} := \{1, 2, \ldots, N_{\mathcal{P}}\}$, and reactive power generation sources indexed by the set $\mathcal{Q} := \{1, 2, \ldots, N_{\mathcal{Q}}\}$. Now an active power injection matrix, $\boldsymbol{C_P} \in \mathbb{R}^{N_{\mathcal{B}} \times N_{\mathcal{P}}}$, is defined in such a way that $(C_P)_{ki} = 1, \forall k \in \mathcal{B}, \forall i \in \mathcal{P}$, if active power generation source $i$ is connected to bus $k$, else 0. The reactive power injection matrix $\boldsymbol{C_Q}$ can be similarly defined. We also define $\boldsymbol{p_G}$, $\boldsymbol{p_D}$, $\boldsymbol{q_G}$, and $\boldsymbol{q_D} \in \mathbb{R}^{N_{\mathcal{P}} \times 1}$ as column vectors of active power generation, active power demand, reactive power generation, and reactive power demand, respectively.

The AC power flow equation (one of the network constraints that needs to satisfied) can then be written as (3) and (4) from [5]

$$\boldsymbol{C_P p_G} - \boldsymbol{p_D} - \Re(\boldsymbol{v}(\boldsymbol{Y}\boldsymbol{v})^*) = 0, \tag{3}$$

[2]Upper-case (lower-case) boldface letters will be used for matrices (column vectors);$(.)^T$ for matrix transposition; $\Re(.)$ and $\Im(.)$ denote the real and imaginary parts of a complex number, respectively; $j := \sqrt{-1}$ the imaginary unit; for a given column vector $\boldsymbol{x}$, $x_i$ represents $i^{th}$ entry.

$$\boldsymbol{C_Q q_G} - \boldsymbol{q_D} - \Im(\boldsymbol{v}(\boldsymbol{Y v})^*) = 0. \quad (4)$$

The transmission line is limited by its thermal capacity to carry power. Let $S_{ij}, \forall i \neq j \in \mathcal{B}, Y_{ij} \neq 0$ be the upper limit of power flow over a transmission line connecting buses $i$ and $j$. To formulate this as one of the constraints, we have to consider the power flow through the line from both directions (i.e., power flowing from $i$ to $j$, and from $j$ to $i$). If $s_i$ and $s_j$ are the complex power injected into line $ij$ (if this line exists) from buses $i$ and $j$, respectively, then (5) and (6) need to be satisfied.

$$s_i s_i^* - S_{ij}^2 \leq 0, \forall i \neq j \in \mathcal{B}, Y_{ij} \neq 0 \quad (5)$$

$$s_j s_j^* - S_{ij}^2 \leq 0, \forall i \neq j \in \mathcal{B}, Y_{ij} \neq 0 \quad (6)$$

Other constraints to be satisfied are zero angle for reference bus, the active and reactive power sources also have upper and lower bounds on their generation capacity, and the voltage magnitude should not violate overvoltage and undervoltage thresholds.

$$\min_{\boldsymbol{\theta}, \boldsymbol{u}, \boldsymbol{p_G}, \boldsymbol{q_G}} f(\boldsymbol{p_G}) \quad (7)$$

$$\text{s.t., (3), (4), (5), (6)}$$

$$\theta_1 = 0 \quad (8)$$

$$u_{min} \leq u_k \leq u_{max}, \forall k \in \mathcal{B} \quad (9)$$

$$p_{G,k,min} \leq p_{G,k} \leq p_{G,k,max}, \forall k \in \mathcal{P} \quad (10)$$

$$q_{G,k,min} \leq q_{G,k} \leq q_{G,k,max}, \forall k \in \mathcal{Q} \quad (11)$$

where $u_{min}, u_{max}$ represent lower and upper bound on voltage magnitudes, and lower and upper bounds on active power and reactive power at bus $k$ are denoted by $p_{G,k,min}, p_{G,k,max}, q_{G,k,min}$ and $q_{G,k,max}$, respectively. Equations (9), (10), (11) describe that voltage magnitude, active power dispatch and reactive power dispatch must be within lower and upper bounds.

ACOPF is thus a non-linear optimization problem that seeks to minimize the cost of generation in a given power system, defined by (7), subject to constraints (3) – (6) and (8) – (11). Note that in this standard formulation, the cost function is only a function of active (real) power output of the generators, and typical cost functions of generating units are quadratic.Time-coupled constraints, such as generator ramp rate limits, cannot be addressed by ACOPF alone, but rather a *time-constrained optimal power flow* (TCOPF) must be solved. ACOPF gives the optimal generator dispatch at a specific instant in time, while TCOPF will give the optimal solution over a given time period of interest $[0, T-1]$, such as in a day-ahead generator schedule, with $N_T$ equal time-steps $\Delta T$. Let $\boldsymbol{r}$, $\boldsymbol{d}$ be the vector of ramp up and ramp down limits for all generators. Ramp limit indicates the power that can be changed within one time step $\Delta T$. The change in active power output of the $k^{th}$ generator between time steps $\tau$ and $\tau + 1$ is limited by ramp rate constraints formulated in (12) and (13) for ramp up and ramp down limits, respectively.

$$p_{G,k}(\tau + 1) - p_{G,k}(\tau) \leq r_k, \forall k \in \mathcal{P} \quad (12)$$

$$p_{G,k}(\tau) - p_{G,k}(\tau + 1) \leq d_k, \forall k \in \mathcal{P} \quad (13)$$

The full TCOPF formulation is presented in (14), subject to (12), (13), and (15)- (22). It is important to mention at this point that in TCOPF, some of the constraints are time coupled and is not necessary to satisfy them in each time step rather they are satisfied in a period of time. Note, for every variable to indicate the value at particular time step, a superscript $\tau$ is used.

$$\min_{\boldsymbol{\theta(\tau)}, \boldsymbol{u(\tau)}, \boldsymbol{p_G(\tau)}, \boldsymbol{q_G(\tau)}} \sum_{\tau=1}^{N_T} f(\boldsymbol{P_G(\tau)}) \quad (14)$$

$$\text{s.t. (12), (13), } \tau = 1, .., N_T - 1$$

$$\boldsymbol{C_P p_G(\tau)} - \boldsymbol{p_D(\tau)} - \Re(\boldsymbol{v(\tau)}(\boldsymbol{Y v(\tau)})^*) = 0 \quad (15)$$

$$\boldsymbol{C_Q q_G(\tau)} - \boldsymbol{q_D(\tau)} - \Im(\boldsymbol{v(\tau)}(\boldsymbol{Y v(\tau)})^*) = 0 \quad (16)$$

$$\theta_1(\tau) = 0 \quad (17)$$

$$s_i(\tau) s_i(\tau)^* - S_{ij}^2 \leq 0, \forall i \neq j \in \mathcal{B}, Y_{ij} \neq 0 \quad (18)$$

$$s_j(\tau) s_j(\tau)^* - S_{ij}^2 \leq 0, \forall i \neq j \in \mathcal{B}, Y_{ij} \neq 0 \quad (19)$$

$$u_{min} \leq u_k(\tau) \leq u_{max}, \forall k \in \mathcal{B} \quad (20)$$

$$p_{G,k,min}(\tau) \leq p_{G,k}(\tau) \leq p_{G,k,max}(\tau), \forall k \in \mathcal{P} \quad (21)$$

$$q_{G,k,min}(\tau) \leq q_{G,k}(\tau) \leq q_{G,k,max}(\tau), \forall k \in \mathcal{Q} \quad (22)$$

Both ACOPF and TCOPF are large-scale non-linear optimization problems, and PDIPM is a common approach to solving such problems [5], [15]. The matrix to be solved using PDIPM approach in general is very ill-conditioned, and solving linear systems of equations involving ill-conditioned matrices is challenging [5], especially during parallel implementation. The next sections describes parallel implementation section to address these issues and reduce solution time. The goal is to improve both the solution time and solution accuracy of the parallel implementation of ACOPF and TCOPF problems using PDIPM.

## IV. Parallel Implementation

We use the Portable, Extensible Toolkit for Scientific Computation (PETSc) to parallelize the power flow data structures and computations. PETSc is a collection of data structures and high-level numerical routines designed for the parallel numerical solution of problems modeled by partial differential equations [16]. Specifically, PETScs DMNetwork data management object for distributed networks is used to represent the power grid of buses, connections, loads, resistances, and other pertinent features. The DMNetwork object combined with algebraic solvers within PETSc provide a greatly simplified interface for network problems over many MPI processes [17]. As described in Section III, the primary difference between the ACOPF and TCOPF formulations is the number of constraints. TCOPF has additional constraints for each time period, resulting in a larger overall matrix system to be solved. However, the primal dual interior point method is used in either case, and the topology of the underlying network is constant for a given problem. As such, essentially

the same parallel implementation can be used for both ACOPF and TCOPF. The flow chart for parallel implementation of solving ACOPF problem is shown in Fig. 1.The power grid network data is initially read in root MPI process. User-defined data structures are registered with a DMNetwork object using PETScs function DMNetworkRegisterComponent. After registering the generators, buses, loads, and branches, the network is distributed using DMNetworkDistribute, which passes a subset of the network to each process. Using the distributed network, the matrices and vectors needed in the primal dual interior point method are constructed in parallel. The Krylov subspace solvers (KSP) in PETSc is used to solve linear equations of the form Ax = b in each process.

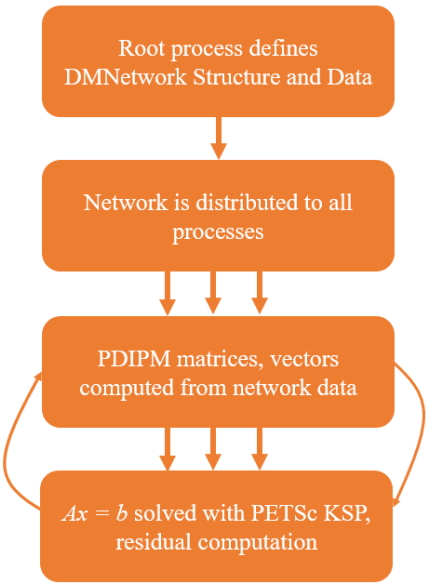


Fig. 1. Flowchart of parallel algorithm implementation

## V. Optimization Parameters and Test System

Three IEEE standard power system networks, namely 118 bus, 300 bus, and 1354 bus systems, are used to evaluate the parallel implementation of ACOPF. The generator information, line parameters, and load data for these systems were collected from MATPOWER (a free collection of MATLAB scripts used to solve steady state power flow and optimization problems), and the solution accuracy is compared with the result from the *runopf* function within MATPOWER [16]. The parallel algorithm is implemented in PETSc and run on Blackjack cluster of South Dakota State University. This particular cluster has 50 nodes, with each node having 12 cores/threads composed of two 2.93 GHz Intel Xeon X5670 processors (6 cores each) with 48 GB of RAM. The processors use a non-uniform memory access architecture. The cluster uses Mellanox InfiniBand low-latency interconnect for inter-node communication.

To test the parallel implementation of TCOPF, currently the IEEE 5 bus test case is used. The optimization horizon for TCOPF is one day, with 24 1-hour timesteps. A synthetic load profile is generated for each bus, and is provided along with the open source code[3]. Ramp rate limits, specified as 100 MW/h for both ramp up and down limits for each generator in the 5 bus system, are considered as the only time-coupling constraints in TCOPF.

## VI. Parallel Performance

The execution time with increasing number of CPUs for three test cases are plotted in Fig. 2. To mitigate the chance of bad runs, the simulation was executed three times with the average times for execution plotted for each case. Fig. 2 shows that the average execution time reduces for 300 bus and 1354 bus test case systems, however for the 118 bus case it reduces until 8 cores. One reason this occurs could be that the problem size for 118 bus is too small so that the communication time overhead is much larger than the computation time, making it unsuitable after a certain number of CPUs for parallelism. The reduced average execution time for higher test case systems is promising in that a lot of time can be saved by using high performance computing to solve ACOPF problems. Strong

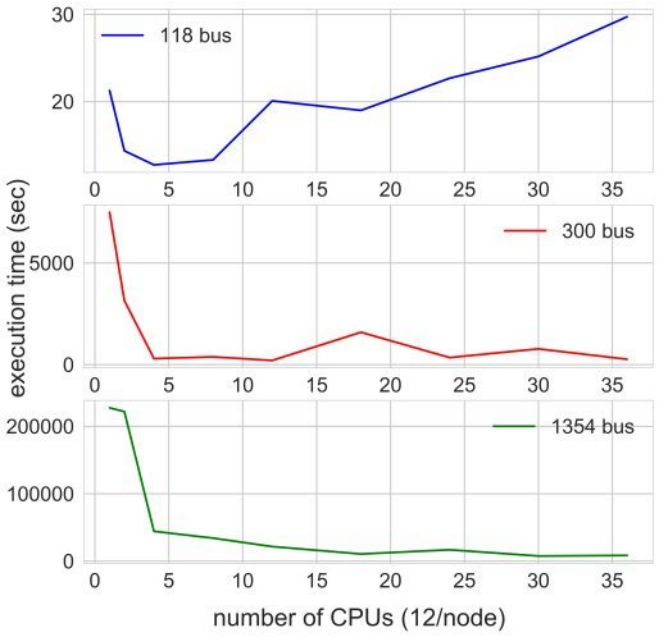


Fig. 2. Execution time versus number of CPU cores in three different case systems.

scaling results for the parallel ACOPF implementation with an Additive Schwarz (ASM) preconditioner are shown in Fig. 3. For networks large enough to provide a high computational intensity, e.g., the 1354 bus and 300 bus systems, the implementation shows significant speedup. However, variability is seen in performance, especially with the 300 bus system. Several factors can influence the number of iterations needed for convergence, but the primary factor in this case is the number of subdomains for the ASM preconditioner. The

[3]https://github.com/alexWerner/SDStateHPCREU/tree/master/tcopf

TABLE I
SYSTEM ACTIVE POWER GENERATION AFTER PARALLEL ACOPF (MW)

| | 118 bus | 300 bus | 1354 bus |
|---|---|---|---|
| 1 core | 4319.37 | 23832.0 | 74092.7 |
| 2 cores | 4319.37 | 23829.4 | 74139.5 |
| 4 cores | 4319.44 | 23831.4 | 73685.1 |
| 8 cores | 4319.39 | 23835.4 | 79168.6 |
| 12 cores | 4319.46 | 23835.5 | 77004.2 |
| 18 cores | 4319.48 | 23835.1 | 76247.1 |
| 24 cores | 4319.67 | 23835.5 | 74263.8 |
| 30 cores | 4319.63 | 23835.7 | 77803.8 |
| 36 cores | 4319.59 | 23836 | 76185.8 |

number of subdomains is set to the number of MPI processes, so increasing the number of CPUs can change the numerical convergence rate. The dependence on preconditioner is further illustrated by comparing against the performance of using PETSc's Jacobi preconditioner or no preconditioner at all – in both cases, the linear solver failed to converge within several hundred thousand iterations. In other words, the problem is extremely numerically stiff without an effective preconditioner. Table I shows the total MW generation from all generators in

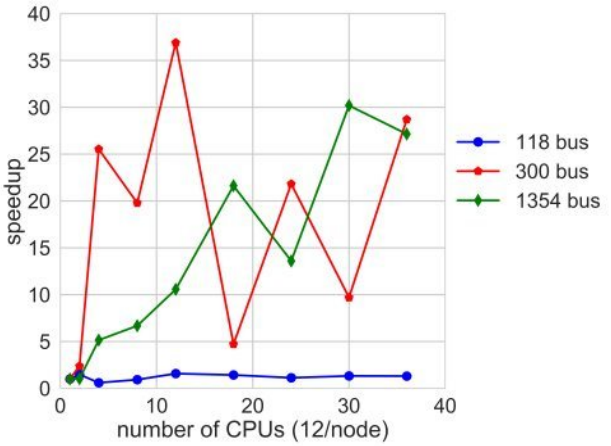


Fig. 3. Parallel speedup of ACOPF code in the 118 bus (blue), 300 bus (red), and 1354 bus (green) test systems. Significant performance variation is seen due to varying parallel decomposition and subdomain construction with the ASM preconditioner.

each of the three test case systems after implementing parallel ACOPF in different numbers of cores. The inaccuracies seem to increase with an increase in system size, though they are small and did not significantly impact the objective function value (which is the cost of generating power). In fact, a maximum difference of \$10/MWh was observed in the 118 test case system compared to different number of cores. This suggests a room for improvement to get more accurate results in future in the parallel test cases.

The total cost of active power generation in all three test case systems after parallel implementation of ACOPF is presented in Table II for different numbers of cores. The cost of producing power did not vary much for the 118 bus system, however the variation is higher for larger test case systems. A maximum difference of around \$300 is observed in the 300 bus

TABLE II
SYSTEM COST OF GENERATION AFTER PARALLEL ACOPF (\$/HR)

| | 118 bus | 300 bus | 1354 bus |
|---|---|---|---|
| 1 core | 129660.7 | 719725.2 | 74092.7 |
| 2 cores | 129660.7 | 719704.2 | 74139.5 |
| 4 cores | 129660.1 | 719850.9 | 73685.1 |
| 8 cores | 129660.1 | 720018.8 | 79168.6 |
| 12 cores | 129662.9 | 720022.4 | 77004.2 |
| 18 cores | 129661.9 | 720006.4 | 76247.1 |
| 24 cores | 129670.4 | 720022.4 | 74263.8 |
| 30 cores | 129668.5 | 720031.5 | 77803.8 |
| 36 cores | 129667 | 720043.6 | 76185.8 |

TABLE III
PERCENTAGE ACCURACY OF OBJECTIVE FUNCTION COMPARED TO KNOWN SOLUTION

| | 118 bus | 300 bus | 1354 bus |
|---|---|---|---|
| 1 core | 100.0 | 100.0 | 99.9685 |
| 2 cores | 100.0 | 99.9971 | 99.9053 |
| 4 cores | 99.9995 | 99.9825 | 99.4812 |
| 8 cores | 99.9995 | 99.9592 | 93.1156 |
| 12 cores | 99.9983 | 99.9587 | 96.0377 |
| 18 cores | 99.9991 | 99.9609 | 97.0598 |
| 24 cores | 99.9925 | 99.9587 | 99.7375 |
| 30 cores | 99.994 | 99.9574 | 94.9582 |
| 36 cores | 99.9951 | 99.9557 | 97.1426 |

and \$5483 in 1354 bus systems, while it is under \$10 for the 118 bus system. The deviation tends to be bigger for larger test case systems. The lower total cost in 1354 bus system compared to 118 and 300 bus systems is due to the smaller cost coefficients for generators specified in the IEEE standard test case.

The percentage accuracy of the objective function value (which is the total cost of active power generation) compared to the base optimal value obtained from MATPOWER is given in Table III. The minimum accuracy across all tests was 93% in 1354 bus system with 8 cores. The accuracy is more than 99% for the 118 and 300 bus systems.

Lastly, an initial implementation for parallel TCOPF is also tested in the IEEE 5 bus test case system, which results in a comparatively small matrix problem for each iteration of PDIPM. Nonetheless, some speedup is seen in the strong scaling data presented in Fig. 4. For larger processor counts, the communication cost begins to outweigh the computation cost for this network size. Future work will involve the development of an open source generalized computational framework for TCOPF that can process arbitrary networks of larger size than the current test case. For large test case in TCOPF, KKT matrix will increase rapidly and will be more sparse given to increase in number of constraints.

## VII. CONCLUSIONS

In this paper, a scalable implementation of ACOPF with effective preconditioning using PETSc's Additive Schwarz preconditioner has been presented. Parallelization of the algorithm was accomplished using the DMNetwork framework within PETSc, which decomposes the network and provides an interface to algebraic solvers. Initial results for a parallel

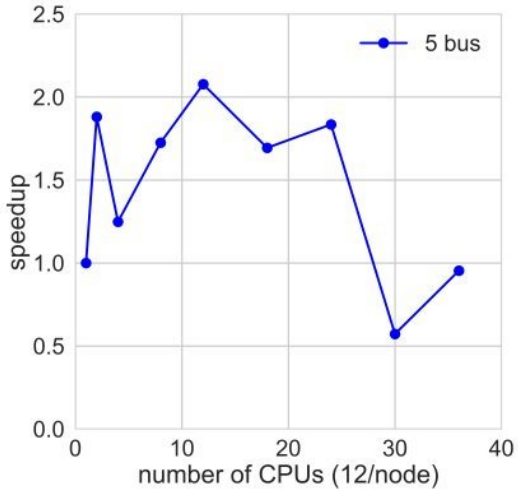


Fig. 4. Parallel speedup of TCOPF code on South Dakota State University's Blackjack cluster.

ACOPF implementation demonstrated significant speedup as a function of the number of CPUs for large test systems. The communication overhead may have caused reduced speedup with increasing CPUs, which is also dependent on whether the CPUs are within one node or distributed over multiple nodes. However, the number of subdomains for the ASM preconditioner was probably the root cause of the variable performance seen in the strong scaling data. Initial work on a TCOPF implementation has also been shown to improve speedup for 5 bus test case system for limited scaling of processors.

## VIII. Future Research

Implementing TCOPF in larger test case systems requires realistic load data and generator information, e.g., ramp rate limits, up-time, and down-time, which are often confidential and hard to obtain. In the future, a more generalized TCOPF formulation, including multiple time-coupling constraints, and its implementation in parallel will be explored. This brings new challenges, especially in terms of parallel decomposition of the network. Different methods of decomposing power system networks for effective parallel implementation might need more detail understanding of problem itself and also in depth knowledge of new tools such as PETSc and PLASMO. Additionally, synthetic generator information and load data which closely resemble the real load and generator data are necessary [17], [18]. Finally, the different approaches will be tested to achieve the best scalable parallel performance for both ACOPF and TCOPF. Given the overall dependence of the computation time and scalability on the preconditioner for the linear system, we will investigate the performance benefits of user-defined preconditioners based on domain decomposition methods that better consider the underlying characteristics of the network.

## Acknowledgment

The authors would like to thank the 2018 High Performance Computing Research Experience for Undergraduates (REU) program held at South Dakota State University for giving the research opportunity and tools needed to accomplish this project.

## References

[1] Energy Information Administration , "Electricity in the United States," 2018. [Online]. Available: https://www.eia.gov/energyexplained/index.php?page=electricity_in_the_united_states

[2] National Conference of State Legislatures , "State Renewable Portfolio Standards and Goals," 2018. [Online]. Available: http://www.ncsl.org/research/energy/renewable-portfolio-standards.aspx

[3] G. Torres and V. Quintana, "An Interior-Point method for Nonlinear Optimal Power Flow using Voltage Rectangular Coordinates," *IEEE Transactions on Power Systems*, vol. 13, no. 4, pp. 1211–1218, Nov. 1998.

[4] N. Meyer-Huebner, M. Suriyah, and T. Leibfried, "On Efficient Computation of Time Constrained Optimal Power Flow in Rectangular Form," *2015 IEEE Eindhoven PowerTech*, pp. 1–6, Jun. 2015.

[5] P. Gerstner, M. Schick, V. Heuveline, N. Meyer-h, T. Leibfried, V. Slednev, W. Fichtner, and V. Bertsch, "A Domain Decomposition Approach for Solving Dynamic Optimal Power Flow Problems in Parallel with Application to the German Transmission Grid," Heidelberg University, Heidelberg, Tech. Rep., Nov. 2016.

[6] P. Gerstner, V. Heuveline, M. Schick, and P. No, "A Multilevel Domain Decomposition approach for solving time constrained Optimal Power Flow problems," Heidelberg University, Heidelberg, Tech. Rep., Sep. 2015.

[7] D. M. Falcão, "High Performance Computing in Power System Applications," in *Second International Conference on Vector and Parallel Processing*, Sep. 1996, pp. 1–23.

[8] R. C. Green, L. Wang, and M. Alam, "Applications and Trends of High Performance Computing for Electric Power Systems: Focusing on Smart Grid," *IEEE Transactions on Smart Grid*, vol. 4, no. 2, pp. 922–931, Jun. 2013.

[9] B. Palmintier, E. Hale, T. M. Hansen, W. Jones, D. Biagioni, H. Sorensen, H. Wu, and B.-M. Hodge, "IGMS: An Integrated ISO-to-Appliance Scale Grid Modeling System," *IEEE Transactions on Smart Grid, special issue on High-Performance Computing Applications for a More Resilient and Efficient Power Grid*, vol. 8, no. 3, pp. 1525–1534, May 2017.

[10] B. H. Kim and R. Baldick, "Coarse-Grained Distributed Optimal Power Flow," *IEEE Transactions on Power Systems*, vol. 12, no. 2, pp. 932–939, May 1997.

[11] R. Baldick, B. H. Kim, C. Chase, and Y. Luo, "A Fast Distributed Implementation of Optimal Power Flow," *IEEE Transactions on Power Systems*, vol. 14, no. 3, pp. 858–864, Aug 1999.

[12] J. Jalving, S. Abhyankar, K. Kim, M. Hereld, V. M. Zavala, B. Engineering, E. S. Division, and C. S. Division, "A Graph-Based Computational Framework for Simulation and Optimization of Coupled Infrastructure Networks," *IET Generation, Transmission & Distribution*, vol. 11, no. 12, pp. 1–20, Aug. 2017.

[13] A. Castillo and R. P. O'Neill, "Computational Performance of Solution Techniques Applied to the ACOPF," Tech. Rep., Jan. 2013. [Online]. Available: http://www.ferc.gov/industries/electric/indus-act/market-planning/opf-papers/acopf-5-computational-testing.pdf

[14] R. Jabr, A. Coonick, and B. Cory, "A Primal-Dual Interior Point Method for Optimal Power Flow Dispatching," *IEEE Transactions on Power Systems*, vol. 17, no. 3, pp. 654–662, Aug. 2002.

[15] R. D. Zimmerman and C. E. Murillo-s, "MATPOWER Interior Point Solver MIPS 1.2.2 User's Manual," Tech. Rep., Dec 2016.

[16] R. D. Zimmerman, C. E. Murillo-Sanchez, and R. J. Thomas, "MATPOWER: Steady-State Operations, Planning, and Analysis Tools for Power Systems Research and Education," *IEEE Transactions on Power Systems*, vol. 26, no. 1, pp. 12–19, Feb. 2011.

[17] T. M. Hansen, E. K. P. Chong, S. Suryanarayanan, A. A. Maciejewski, and H. J. Siegel, "A partially observable Markov decision process approach to residential home energy management," *IEEE Transactions on Smart Grid*, vol. 9, no. 2, pp. 1271–1281, Mar. 2018.

[18] V. Durvasulu and T. M. Hansen, "Market-based generator cost functions for power system test cases," *IET Cyber-Physical Systems: Theory & Applications, special issue on Cyber Physical Power Systems: Advanced Intelligent Technologies and Applications*, vol. 3, no. 4, pp. 194–205, Dec. 2018.