

# Ill-Conditioned Power Flow Analysis Using a Quantized State-Based Approach

Liya Huang, Federico Milano, and Georgios Tzounas

*School of Electrical and Electronic Engineering*

*University College Dublin*

Dublin, Ireland

liya.huang@ucdconnect.ie, federico.milano@ucd.ie, georgios.tzounas@ucd.ie

**Abstract**—This paper focuses on power flow analysis through the lens of the Newton flow, a continuous-time formulation of Newton’s method. Within this framework, we explore how quantized-state concepts, originally developed as an alternative to time discretization, can be incorporated to govern the evolution of the Newton flow toward the power flow solution. This approach provides a novel perspective on adaptive step-size control and shows how state quantization can enhance robustness in ill-conditioned cases. The performance of the proposed approach is discussed with the ACTIVSg70k synthetic test system.

**Index Terms**—Ill-conditioned power flow, continuous Newton approach, convergence analysis, quantized state system.

## I. INTRODUCTION

### A. Motivation

Power flow analysis is a fundamental task in power system studies, and several numerical methods are available for its solution, e.g., see [1]–[6]. Among them, methods based on the continuous Newton framework [1], [2] have shown promising robustness for ill-conditioned cases. Within this framework, Newton iterations are interpreted as the numerical integration of a dynamical system whose equilibrium corresponds to the power flow solution. Building on this interpretation, this paper explores how concepts from quantized-state methods – originally proposed as an alternative to time discretization – can be incorporated to enhance performance and robustness for challenging ill-conditioned and poorly initialized cases.

### B. Literature Review

From a numerical viewpoint, power flow problems can be classified into four categories: well-conditioned, ill-conditioned, bifurcation point, or unsolvable. Standard Newton’s method is effective in well-conditioned cases. Saddle-node bifurcations, on the other hand, are commonly detected using continuation methods [7], while maximum loadability optimal power flow formulations [8] offer an alternative approach.

This paper focuses on ill-conditioned cases, where a solution exists but standard approaches like Newton’s method and fast decoupled power flow (FDPF) [6] struggle to converge. Such cases are commonly addressed using robust Newton variants, e.g., see [9]. The continuous Newton framework generalizes

This work is supported by Research Ireland (SFI), under grant number 21/SPP/3756 through the NexSys Strategic Partnership Programme.

these approaches by reformulating the power flow equations into a continuous-time dynamical system, the *Newton flow* [10]. This allows the solution to be obtained through numerical integration using any explicit or implicit discretization scheme [1], [2]. For implicit schemes, each integration step requires the solution of a nonlinear algebraic system via Newton iterations, resulting in a double (inner/outer) loop structure. In this context, adaptive time stepping has also been explored as a means of improving efficiency, e.g., by updating the step size based on local truncation error estimates in explicit schemes or on the convergence of the inner loop in implicit ones.

It is natural to ask whether discretization paradigms beyond conventional time-stepping can be incorporated into this framework and provide enhanced performance and robustness. The main idea of this paper is that such an alternative paradigm can be provided by quantized state system (QSS) methods [11], [12]. These methods replace time discretization with state quantization, triggering updates only when a state changes by a prescribed amount, the *quantum*. Recent work has provided additional insight into QSS through a *duality* perspective, in which time is allowed to evolve as a state-dependent variable [13]. In this paper, we explore how the QSS logic can be embedded into the Newton flow to realize event-driven step-size adaptation for ill-conditioned power flow analysis.

### C. Contribution

The main contribution of this paper is to introduce QSS concepts into the continuous Newton framework for power flow analysis, enabling state-event-driven step-size adaptation for solving the associated Newton flow. In addition, we provide a matrix-pencil-based analysis of explicit and implicit discretizations of the Newton flow, offering insight into their local convergence properties. The proposed approach shows potential to achieve improved performance and robustness for ill-conditioned and poorly initialized cases.

### D. Paper Organization

The remainder of the paper is organized as follows. Section II reviews the continuous Newton framework and analyzes the local convergence of its explicit and implicit discretizations under varying step sizes. Section III introduces the QSS concept and its use for event-driven step-size adaptation.

Section IV presents the case study based on the ACTIVSg70k test system. Finally, conclusions are drawn in Section V.

## II. CONTINUOUS NEWTON APPROACH FOR POWER FLOW ANALYSIS

This section presents an overview of the continuous Newton approach applied to power flow analysis. We first outline its formulation and describe explicit and implicit variants. Then, we evaluate local convergence around the power flow solution.

### A. Formulation

Power flow analysis refers to solving the nonlinear system:

$$\mathbf{0} = \mathbf{g}(\mathbf{y}_o) \quad (1)$$

where  $\mathbf{y}_o \in \mathbb{R}^n$  is the column vector of system variables, including voltage magnitudes and angles at PQ buses, reactive power and voltage angles at PV buses, and active/reactive power at the slack bus. The functions  $\mathbf{g}(\mathbf{y}_o) \in \mathbb{R}^n$  represent the corresponding power flow constraints across the network.

The main idea of continuous Newton's methods is that the solution of (1) coincides with the steady state of the *Newton-flow*, i.e., of the continuous system:

$$\mathbf{g}_y \mathbf{y}' = -\mathbf{g}(\mathbf{y}) \quad (2)$$

where  $\mathbf{g}_y$  is the Jacobian matrix  $\mathbf{g}_y = \partial \mathbf{g} / \partial \mathbf{y}$ . In other words, power flow analysis is translated into the problem of numerically integrating (2) until its equilibrium  $\mathbf{y}_o$  is reached. Moreover, since only the steady state is of interest, the trajectory of  $\mathbf{y}$  itself is irrelevant – what matters is whether a discrete approximation of (2) converges and how quickly.

The simplest among all discretization schemes is the forward Euler method (FEM). Application to (2) gives:

$$\mathbf{0} = \mathbf{g}_{y_k}(\mathbf{y}_{k+1} - \mathbf{y}_k) + h_k \mathbf{g}(\mathbf{y}_k) \quad (3)$$

where  $h_k$  is the integration step size. Obtaining  $\mathbf{y}_{k+1}$  requires factorizing the matrix  $\mathbf{g}_{y_k}$  at each step.

Observe that (3) corresponds to the  $k$ -th iteration of Newton's method with damping factor  $h_k \in (0, 1]$ . In particular, setting  $h_k = 1$  recovers standard Newton's method, which is effective when  $\mathbf{g}_{y_k}$  is well-conditioned, whereas  $h_k < 1$  yields robust Newton's method, useful to improve convergence in cases where  $\mathbf{g}_{y_k}$  is ill-conditioned.

A limitation of (3) is that its numerical stability margin shrinks as  $\mathbf{g}_{y_k}$  becomes ill-conditioned. This forces small step sizes to achieve convergence. In this context, implicit schemes such as the backward Euler method (BEM) become attractive [2], as they remain stable even for arbitrarily large step sizes and overdamp fast dynamics, offering improved robustness. Applied to the Newton flow (2), BEM reads:

$$\mathbf{0} = \mathbf{g}_{y_{k+1}}(\mathbf{y}_{k+1} - \mathbf{y}_k) + h_k \mathbf{g}(\mathbf{y}_{k+1}) = \phi(\mathbf{y}_{k+1}) \quad (4)$$

System (4) is solved at each step using Newton iterations:

$$\mathbf{y}_{k+1}^{[i+1]} = \mathbf{y}_{k+1}^{[i]} - \phi_y^{-1}(\mathbf{y}_{k+1}^{[i]}) \phi(\mathbf{y}_{k+1}^{[i]}), \quad i \in \mathbb{N}. \quad (5)$$

Thus, BEM leads to a double-loop algorithm: an *inner* loop, where, for a given  $k$ , (5) is solved to compute  $\mathbf{y}_{k+1}$  from

$\mathbf{y}_k$ ; and an *outer* loop, where, once  $\mathbf{y}_{k+1}$  is obtained,  $k$  is advanced and the process is repeated until  $|\mathbf{y}_{k+1} - \mathbf{y}_k| < \varepsilon$ . When this condition is satisfied, (2) has reached steady state, i.e.,  $\mathbf{y}' = \mathbf{0}$ , and (1) is solved. Further implementation details of (5) are provided in the case study of Section IV.

### B. Local Convergence Analysis

We discuss the local convergence of FEM (i.e., robust Newton's method) and BEM in the neighborhood of the power flow solution. To this end, the discrete-time mappings (3), (4) are linearized around  $\mathbf{y}_o$ , and their convergence is assessed using linear stability theory [14], [15].

*FEM*: Linearizing (3) around  $\mathbf{y}_o$  yields:

$$\mathbf{0} = \mathbf{g}_{y_o}(\tilde{\mathbf{y}}_{k+1} - \tilde{\mathbf{y}}_k) + h_k \mathbf{g}_{y_o} \tilde{\mathbf{y}}_k \quad (6)$$

where  $\tilde{\mathbf{y}} = \mathbf{y} - \mathbf{y}_o$ . Equivalently:

$$\tilde{\mathbf{y}}_{k+1} = (1 - h_k) \tilde{\mathbf{y}}_k \quad (7)$$

where, in exact arithmetic,  $\mathbf{g}_{y_o}^{-1} \mathbf{g}_{y_o} = \mathbf{I}$  and this also holds numerically as long as  $\mathbf{g}_{y_o}$  is well-conditioned. The corresponding  $z$ -domain matrix pencil [16]  $z\mathbf{I} - (1 - h_k)\mathbf{I}$  yields a repeated eigenvalue  $z = 1 - h_k$  with multiplicity  $n$ . The stability condition  $|z| < 1$  gives the bound  $0 < h_k < 2$ .

*BEM*: Linearizing (4) around  $\mathbf{y}_o$  yields:

$$\mathbf{0} = \mathbf{g}_{y_o}(\tilde{\mathbf{y}}_{k+1} - \tilde{\mathbf{y}}_k) + h_k \mathbf{g}_{y_o} \tilde{\mathbf{y}}_{k+1} \quad (8)$$

Equivalently:

$$(1 + h_k) \tilde{\mathbf{y}}_{k+1} = \tilde{\mathbf{y}}_k \quad (9)$$

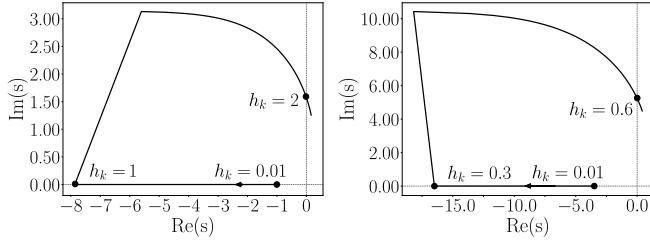
For well-conditioned cases, the corresponding  $z$ -domain matrix pencil  $(1 + h_k)z\mathbf{I} - \mathbf{I}$  has a repeated eigenvalue  $z = 1/(1 + h_k)$  with multiplicity  $n$ . Since  $|z| < 1$  holds  $\forall h_k > 0$ , the method is locally stable around  $\mathbf{y}_o$  for any positive step.

Figures 1 and 2 show the local convergence behavior of FEM and BEM, respectively, as the step size  $h_k$  varies. The results are obtained by first mapping from  $z$ - to  $s$ -domain:

$$s = \log(z)/h_k = \alpha + j\beta \quad (10)$$

Under well-conditioned cases, Fig. 1a shows that, as expected, convergence of FEM is monotonic for  $0 < h_k < 1$ , with the fastest decay rate obtained for  $h_k = 1$ ; as  $h_k$  increases beyond 1, behavior becomes oscillatory, and for  $h_k > 2$ , the method diverges. For BEM, Fig. 2 confirms that convergence is monotonic  $\forall h_k > 0$ , with the decay rate decreasing gradually as  $h_k$  increases. This implies that, near  $\mathbf{y}_o$ , a larger  $h_k$  leads to slower local convergence.

FEM is also known to be sensitive to ill conditioning. In particular, when  $\mathbf{g}_y$  is ill-conditioned, its inversion amplifies numerical errors, while residual inaccuracies accumulate across iterations, thus introducing a non-negligible distortion to the local mapping around  $\mathbf{y}_o$ . To qualitatively show this effect, we consider an illustrative model of the distorted pencil as  $z\mathbf{I} - (1 - h_k + \epsilon)(1 + \eta)\mathbf{I}$ , where  $\eta$  accounts for imprecision in the factorization of  $\mathbf{g}_{y_o}$ ; and  $\epsilon$  represents the accumulated residual error during Newton iterations. For example, with



(a) Well-conditioned case. (b) Ill-conditioned case.

Fig. 1: Local convergence region, FEM.

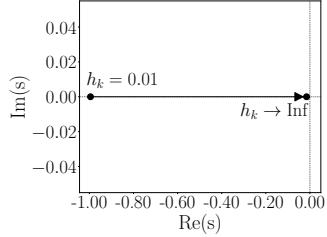


Fig. 2: Local convergence region, BEM.

$\eta = 2.33$  and  $\epsilon = -0.7$ , the local convergence region contracts to  $0 < h_k < 0.6$ , as shown in Fig. 1b. In contrast, the implicit nature and strong numerical damping properties of BEM greatly reduce the impact of numerical and residual errors, thereby offering enhanced robustness. In practice, robustness also depends on the step-size selection. In the following, BEM is equipped with a state-event-driven rule for adapting  $h_k$ .

### III. QSS-BASED STEP CONTROL

In this section, we integrate QSS concepts into the numerical solution of the power flow problem through the Newton flow formulation (2). In the family of QSS methods, differential equations are solved by quantizing the state variables rather than discretizing time. The simplest member of this family is the first-order scheme, QSS1 [11]. Consider the test equation:

$$y'(t) = f(y(t)) , \quad y \in \mathbb{R} \quad (11)$$

In QSS1, the derivative  $y'(t)$  is evaluated using a piecewise-constant input signal  $q(t)$  that approximates  $y(t)$ :

$$y'(t) = f(q(t)) \quad (12)$$

where  $q(0) = y(0)$ . The signal  $q(t)$  updates only when  $y(t)$  changes by a fixed *quantum*  $\Delta q$ :

$$q(t) = \begin{cases} y(t) & \text{if } |y(t) - q(t^-)| \geq \Delta q \\ q(t^-) & \text{otherwise} \end{cases}$$

Between two consecutive quantization events, the right-hand side of (12) remains constant, resulting in  $y(t)$  being piecewise linear. Let  $t_k$  denote the time of the  $k$ -th event, where  $q(t_k) = y(t_k)$ . For  $t \in [t_k, t_{k+1})$ , the state evolves as:

$$y(t) = y(t_k) + (t - t_k)f(y(t_k)) , \quad t \in [t_k, t_{k+1}) \quad (13)$$

The next event occurs at  $t = t_{k+1}$ , when the deviation between  $y(t)$  and  $y(t_k)$  reaches the quantum:

$$|y(t) - y(t_k)| = \Delta q \quad (14)$$

Substituting (13) in (14) at  $t = t_{k+1}$ , we get  $h_k|f(y(t_k))| = \Delta q$ . Equivalently, the step size  $h_k = t_{k+1} - t_k$  is:

$$h_k = \Delta q / |f(y(t_k))| \quad (15)$$

We now extend the same idea to the Newton flow (2). The quantized variables are updated according to:

$$q_j(t) = \begin{cases} y_j(t) & \text{if } |y_j(t) - q_j(t^-)| \geq \Delta q \\ q_j(t^-) & \text{otherwise} \end{cases}$$

where  $j$  is the index of vector  $\mathbf{y}$ ,  $j \in \{1, \dots, n\}$ . To compute the derivative of  $\mathbf{y}(t)$  at the time instant of each quantization event, the quantized signal  $\mathbf{q}(t)$  is substituted into (2), yielding:

$$\mathbf{g}_y(\mathbf{q}(t))\mathbf{y}' = -\mathbf{g}(\mathbf{q}(t)) \quad (16)$$

$\mathbf{y}'(t)$  is then approximated as:

$$\mathbf{y}' = \mathbf{f}(\mathbf{q}(t)) = -[\mathbf{g}_y(\mathbf{q}(t))]^{-1}\mathbf{g}(\mathbf{q}(t)) \quad (17)$$

From this, the step size that determines when each variable reaches its next quantization event can be estimated as:

$$h_{k,j} = \Delta q / |f_j(\mathbf{y}(t_k))| \quad (18)$$

where  $f_j$  is the  $j$ -th equation of (17). After computing  $h_{k,j}$  for each equation using (18), we then select the smallest value:

$$h_k = \min\{h_{k,j}\} , \quad j \in \{1, \dots, n\} \quad (19)$$

as the global step size. Combination of (18) and (19) produces a QSS-based rule for adaptive step-size selection that integrates naturally with the BEM discretization of the Newton flow discussed in Section II. Compared with classical adaptive strategies, such as truncation-error controllers or iteration-count heuristics, which typically require tuning multiple parameters, the proposed mechanism is simpler to configure, as it requires only the choice of  $\Delta q$ . Moreover, because the step size is explicitly tied to controlled state variations, the same principle could be leveraged to handle discrete events such as PV–PQ switching.

### IV. CASE STUDY

This section presents simulation results based on the ACTIVSg70k synthetic test system [17]. The system model includes 70,000 buses, 88,207 transmission lines and transformers, and 10,390 generators. All simulations are carried out in Dome [18], on a computer equipped with an Intel Xeon E3-1245 v5 processor, 16 GB of RAM, running a 64-bit Linux OS.

In the following, we first present base-case results for the original, properly initialized and well-conditioned system. We then examine performance and robustness under modified, poorly-initialized conditions. We consider three reference solvers, namely FEM, FDPF, Runge-Kutta 4 (RK4) [2], and four BEM-based variants, including one fixed-step implementation, one with heuristic step adaptation, and two incorporat-

ing the proposed QSS-based step control, as summarized in Table I.

TABLE I: Power flow solver configurations considered.

Notation	Step size	Description
FEM	Fixed	Robust Newton
FDPF	Fixed	Fast decoupled power flow method
RK4	Adap.	Runge Kutta 4
BEM-J <sub>1</sub>	Fixed	BEM with one inner loop iteration ( $i^{\max} = 1$ )
BEM-J	Adap.	BEM with multiple inner loop iterations ( $i^{\max} > 1$ )
BEM-J <sub>1</sub> -QSS	Adap.	BEM-J <sub>1</sub> with $h_k$ governed by (19)
BEM-J-QSS	Adap.	BEM with $i^{\max} > 1$ and $h_k$ governed by (19)

Unless otherwise stated, the initial step size is set equal to 1 in all configurations. For BEM-based solvers, the Hessian term arising in the computation of  $\phi_y^{-1}$  in (5) is neglected [2]. Moreover, for QSS-based schemes,  $\Delta q$  is set at 20, while a maximum  $h_k^{\max} = 8000$  is imposed.

### A. Base-Case Results

We begin by evaluating the performance of the solver configurations summarized in Table I in the base case, corresponding to the original, well-initialized ACTIVSg70k system. The results are reported in Table II and indicate that all methods converge successfully. Among the reference solvers, FEM exhibits the best overall performance. Furthermore, the QSS-based methods, BEM-J-QSS and BEM-J<sub>1</sub>-QSS, achieve a significant reduction in the number of iterations compared to their respective counterparts, BEM-J and BEM-J<sub>1</sub>, resulting in a modest speedup. Among all tested solvers, BEM-J<sub>1</sub>-QSS has the fastest overall convergence. For completeness, we note that RK4 adapts its step based on an estimate of the local truncation error, while BEM-J adjusts it heuristically according to the convergence rate of the inner Newton loop.

TABLE II: Base-case statistics.

Method	Main loop	Inner loop	CPU time [s]
FEM	25	–	5.89
FDPF	12	–	6.93
RK4	19	–	10.24
BEM-J <sub>1</sub>	25	1	4.61
BEM-J	24	139	18.69
BEM-J <sub>1</sub> -QSS	11	1	2.82
BEM-J-QSS	7	126	17.39

Figure 3 shows the iteration trajectory of the voltage phase angle at bus 2 obtained with FEM and BEM-J. Both methods are run under fixed steps in this figure to allow a direct comparison. For FEM, iterations converge fastest at  $h_k = 1$ , become slower and non-oscillatory for  $h_k < 1$ , oscillatory for  $1 < h_k < 2$ , and diverge when  $h_k > 2$ . In contrast, BEM-J converges to the power flow solution without oscillating for all tested step sizes. These results are consistent with the analysis of Section II-B. Figure 4 shows the trajectory and

step-size evolution of BEM-J-QSS. It can be seen that BEM-J-QSS requires fewer iterations than BEM-J, since  $h_k$  increases according to (19) as state-derivative magnitudes decrease and the system approaches the steady state.

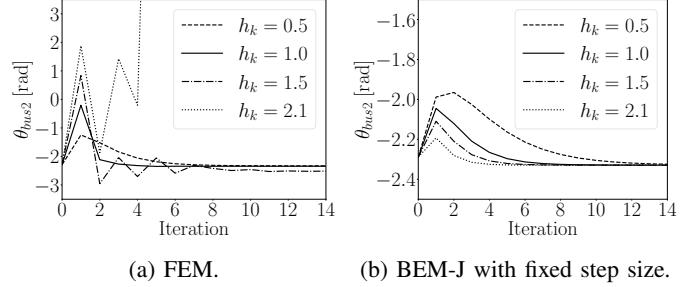


Fig. 3: Well-initialized case: FEM and BEM-J iterations.

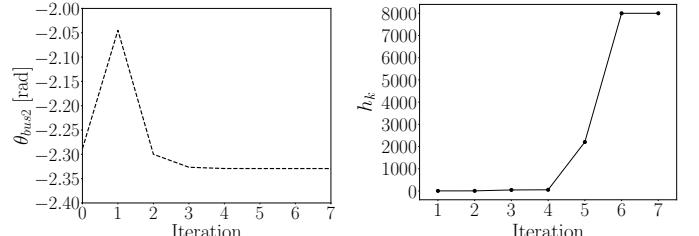


Fig. 4: Well-initialized case: BEM-J-QSS iterations.

### B. Performance for Poorly Initialized Conditions

We examine the performance of the methods in Table I under poorly initialized conditions. Starting from the base case, the initial voltage angles  $\theta_{\text{init}}$  are uniformly scaled by a factor  $\alpha$  to deteriorate the initialization.

The results obtained for a scaling factor of  $\alpha = 1.35$  are summarized in Table III. As shown, this scaling is sufficiently large to cause divergence of the reference solvers. In contrast, both BEM-J-QSS and BEM-J<sub>1</sub>-QSS converge faster than their respective counterparts, BEM-J and BEM-J<sub>1</sub>.

TABLE III: Poorly-initialized-case statistics,  $\alpha = 1.35$ .

Method	Main loop	Inner loop	CPU time [s]
FEM	Diverge	–	–
FDPF	Diverge	–	–
RK4	Diverge	–	–
BEM-J <sub>1</sub>	25	1	4.56
BEM-J	23	135	18.02
BEM-J <sub>1</sub> -QSS	12	1	3.67
BEM-J-QSS	7	102	16.54

The iteration trajectories of BEM-J<sub>1</sub> and BEM-J<sub>1</sub>-QSS for  $\alpha = 1.35$  are shown in Figure 5a. Further increasing the scaling factor provides additional insight into their robustness. In particular, for  $\alpha = 1.66$ , as shown in Fig. 6a, BEM-J<sub>1</sub> diverges at iteration 62 due to an ill-conditioned Jacobian matrix, whose condition number increases by approximately  $10^4$  times compared with its previous value. This issue does not occur in BEM-J<sub>1</sub>-QSS.

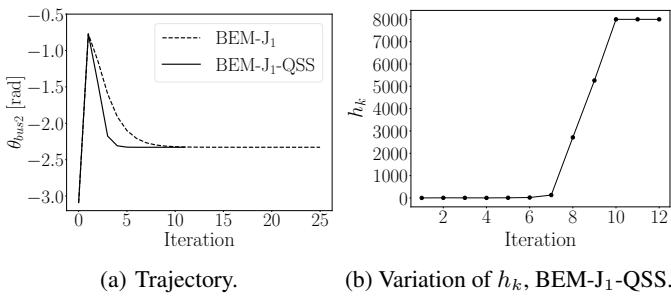


Fig. 5: Iterations of  $\theta_{bus2}$  with BEM-J<sub>1</sub> and BEM-J<sub>1</sub>-QSS,  $\alpha = 1.35$ .

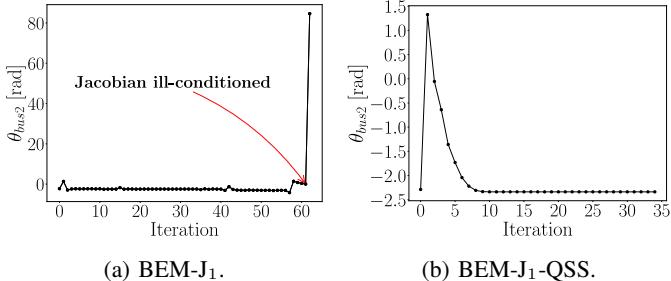
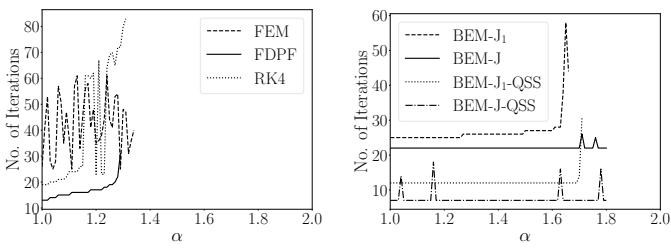


Fig. 6: Iterations of  $\theta_{bus2}$  with BEM-J<sub>1</sub> and BEM-J<sub>1</sub>-QSS,  $\alpha = 1.66$ .

To further assess robustness, each method is tested on a set of 500 cases obtained for scaling factors uniformly distributed in the range  $\alpha \in (1, 2]$ . The percentage of converging cases for all methods in Table I is reported in Table IV. As shown, the reference methods (FEM, FDPF, and RK4) exhibit low overall convergence rates under poorly initialized conditions. BEM-J and BEM-J-QSS achieve the highest convergence rates, while BEM-J<sub>1</sub> and BEM-J<sub>1</sub>-QSS show reduced robustness due to their inner loop approximation. The number of iterations required by each method as  $\alpha$  varies within  $(1, 2]$  is shown in Fig. 7. Each curve terminates at the largest value of  $\alpha$  for which the method maintains convergence.

TABLE IV: Converging cases under scaled initial voltage angles.

Method	Convergence [%]	Method	Convergence [%]
FEM	39.80	BEM-J	82.25
FDPF	32.78	BEM-J <sub>1</sub> -QSS	68.32
RK4	33.40	BEM-J-QSS	82.28
BEM-J <sub>1</sub>	65.93		



(a) Reference solvers.

(b) BEM-based solvers.

Fig. 7: Iterations of  $\theta_{bus2}$  for all methods,  $\alpha \in (1, 2]$ .

## V. CONCLUSION

This paper integrates QSS concepts into the continuous Newton framework for power flow analysis. By leveraging QSS-based state-update logic, the proposed approach enables event-driven adaptive step-size control, improving both convergence speed and robustness in ill-conditioned cases. Simulation results show that it accelerates convergence relative to fixed-step and heuristically tuned BEM variants and enhances robustness when combined with BEM-J<sub>1</sub>. Future work will examine the use of QSS-based event handling within the same framework to manage switching events during iterations triggered by operational limit violations.

## REFERENCES

- [1] F. Milano, “Continuous Newton’s method for power flow analysis,” *IEEE Trans. on Power Systems*, vol. 24, no. 1, pp. 50–57, 2008.
- [2] ———, “Implicit continuous Newton method for power flow analysis,” *IEEE Trans. on Power Systems*, vol. 34, no. 4, pp. 3309–3311, 2019.
- [3] W. Huang, X. Pan, M. Chen, and S. H. Low, “DeepOPF-V: Solving AC-OPF problems efficiently,” *IEEE Trans. on Power Systems*, vol. 37, no. 1, pp. 800–803, 2021.
- [4] L. N. de Oliveira, F. D. Freitas, and N. Martins, “A modal-based initial estimate for the Newton solution of ill-conditioned large-scale power flow problems,” *IEEE Trans. on Power Systems*, vol. 38, no. 5, pp. 4962–4965, 2023.
- [5] S. Iwamoto and Y. Tamura, “A load flow calculation method for ill-conditioned power systems,” *IEEE Trans. on Power Apparatus and Systems*, vol. PAS-100, no. 4, pp. 1736–1743, 1981.
- [6] R. A. Van Amerongen, “A general-purpose version of the fast decoupled load flow,” *IEEE Trans. on Power Systems*, vol. 4, no. 2, pp. 760–770, 2002.
- [7] S.-H. Li and H.-D. Chiang, “Continuation power flow with nonlinear power injection variations: a piecewise linear approximation,” *IEEE Trans. on Power Systems*, vol. 23, no. 4, pp. 1637–1643, 2008.
- [8] W. A. Bukhsh, A. Grothey, K. I. McKinnon, and P. A. Trodden, “Local solutions of the optimal power flow problem,” *IEEE Trans. on Power Systems*, vol. 28, no. 4, pp. 4780–4788, 2013.
- [9] T. J. Overbye, “A power flow measure for unsolvable cases,” *IEEE Trans. on Power Systems*, vol. 9, no. 3, pp. 1359–1365, 2002.
- [10] R. Hauser and J. Nedic, “The continuous Newton–Raphson method can look ahead,” *SIAM J. on Optimization*, vol. 15, no. 3, pp. 915–925, 2005.
- [11] E. Kofman and S. Junco, “Quantized-state systems: A DEVS approach for continuous system simulation,” *Trans. of The Society for Modeling and Simulation International*, vol. 18, no. 3, pp. 123–132, 2001.
- [12] E. Kofman, “A second-order approximation for DEVS simulation of continuous systems,” *Simulation*, vol. 78, no. 2, pp. 76–89, 2002.
- [13] L. Huang and G. Tzounas, “On the duality between quantized time and states in dynamic simulation,” *arXiv preprint arXiv:2510.03785*, 2025.
- [14] G. Tzounas, I. Dassios, and F. Milano, “Small-signal stability analysis of numerical integration methods,” *IEEE Trans. on Power Systems*, vol. 37, no. 6, pp. 4796–4806, 2022.
- [15] G. Tzounas and G. Hug, “Unified numerical stability and accuracy analysis of the partitioned-solution approach,” *IEEE Trans. on Power Systems*, vol. 39, no. 2, pp. 3141–3152, 2023.
- [16] I. Dassios, G. Tzounas, and F. Milano, “Robust stability criterion for perturbed singular systems of linearized differential equations,” *Journal of Computational and Applied Mathematics*, vol. 381, p. 113032, 2021.
- [17] A. B. Birchfield *et al.*, “Grid structural characteristics as validation criteria for synthetic networks,” *IEEE Trans. on Power Systems*, vol. 32, no. 4, pp. 3258–3265, 2016.
- [18] F. Milano, “A Python-based software tool for power system analysis,” in *IEEE Power & Energy Society General Meeting*, 2013, pp. 1–5.