

ONLINE PREDICTION OF POWER SYSTEM TRAJECTORIES FROM PHASOR-MEASUREMENT-UNIT (PMU) DATA

JUAN A. BAZERQUE*, PABLO MONZÓN*, ALVARO GIUSTO*

**Facultad de Ingeniería, Universidad de la República
Uruguay*

Emails: jbzazerque@fing.edu.uy, monzon@fing.edu.uy, alvarog@fing.edu.uy

Abstract— This paper proposes an online method to integrate the trajectories of differential-algebraic equations in the context of networked power systems. We predict these trajectories from a real-time data stream generated by phasor measurement units, combining newly incoming data with past information. For this purpose, we propose a stochastic optimization technique that incorporates new data as they become available, trading-off measurement errors for modeling uncertainty. The method is tested on a benchmark system of 39 buses and 10 generators being able to anticipate the system recovery from a three-phase-to-ground short-circuit.

Keywords— Power system dynamics, synchrophasors, estimation.

1 INTRODUCTION

Recent advances on phasor measurement units (PMUs) pose a broad interest on their use for monitoring, protection, and dynamical security assessment of electrical power systems [1, 2]. The availability of synchronized phasor measurements of electrical variables across the power system calls for new methods to assess real-time transient stability, based on the prediction of system trajectories. A real-time prediction has to be fast enough to handle the relevant transient phenomena and accurate enough for the system safety. There exists a trade-off between accuracy and speed that can be relaxed with the help of model reduction techniques [3], clustering techniques and parallel computing schemes [4, 5]. One approach consists of fitting the power system trajectories using a set of simple basis functions (e.g. polynomials) and then use this approximation to predict the trajectories for a future horizon, see [6]. Alternatively, the power system model can be used to predict future trajectories based on a Taylor series expansion of the state variables [7]. These methods demand the estimation of an initial condition from the available PMU data.

In this context, the present paper builds on advances on online prediction of the system trajectories [2], poses the prediction as an optimization problem that takes into account all the system information available at real time, that is, current and past PMU measurements, nodal equations and machine dynamics. This approach allows us to weight each source of information, considering measurement noise and modeling errors. It also enables the incorporation of incoming PMU measurements into the predictor iteratively, in real time, without discarding the data collected in the past. As a byproduct, our method yields an estimate of the current state, along with the predicted trajectory, that can be incorporated to the methods reported above.

The article is organized as follows. Section 2 presents the model and the main ideas for online prediction. Section 3 describes the implementation de-

tails of the algorithm, that is tested and discussed with the help of an example in Section 4. Section 5 wraps up the paper with concluding remarks.

2 SYSTEM MODEL AND ESTIMATION SETUP

Consider a power grid connecting N_B buses and N_M synchronous generators, with each generator feeding each of the first N_M buses. The power system is described with the standard (second order) differential-algebraic model

$$\begin{cases} \dot{x} &= f_t(x, v) \\ 0 &= g_t(x, v) \end{cases} . \quad (1)$$

Vector $x \in \mathbb{R}^{2N_M}$ denotes the system state (rotor angle and speed of each machine), while $v \in \mathbb{R}^{2N_B}$ represents the link variables; i.e., the bus voltage at every bus in Cartesian coordinates. Function $f_t : \mathbb{R}^{2N_M} \times \mathbb{R}^{2N_B} \rightarrow \mathbb{R}^{2N_M}$ in the first equation gathers the whole system dynamics combining the swing equations for all synchronous generators [8]. The explicit form of the swing equation, which associates the rotational speed and acceleration of a generator with the system electrical variables is given in (11) in the Appendix. The subindex t in $f_t(\cdot)$ reflects the fact that the system dynamics are time-varying, as it could result due to a fault, load shedding event, etc. The second function $g_t : \mathbb{R}^{2N_M} \times \mathbb{R}^{2N_B} \rightarrow \mathbb{R}^{2N_B}$ collects the Kirchoff equations across the network. More precisely, $g_t(x, v) = i_m(x, v) - i_{NET}(x, v) = 0$ force the current flow balance at all buses, with vectors $i_m(x, v)$ and $i_{NET}(x, v)$ representing the (complex-valued) current flows injected by machines or flowing into the network, respectively. Particularly, an entry of i_m is null if the corresponding bus is not connected to a machine. Both $i_m(x, v)$ and $i_{NET}(x, v)$ belong to \mathbb{R}^{2N_B} since the complex value of the current flow at each of the N_B nodes is decomposed into its real and imaginary parts. Explicit forms of $i_m(x, v)$ and $i_{NET}(x, v)$ in terms of the system state x and voltages v are also given in the Appendix. As with f_t ,

the time-varying notation of $g_t(\cdot)$, reflects that the network can change.

Our algorithm will be based on PMU measurements collected across the network. PMUs measure a subset $z_V = S_V v \in \mathbb{R}^p$ of the voltage variables in v , where the selection matrix $S_V \in \mathbb{R}^{p \times 2(N_B)}$ collects the corresponding rows of the $2N_B \times 2N_B$ identity matrix (see Section 3.2). The PMU data are assumed to be noisy versions of the selected z_V corrupted by additive noise $\eta_V \in \mathbb{R}^p$ and denoted by

$$\bar{z}_V = S_V v + \eta_V. \quad (2)$$

Direct measurements $\bar{z}_X = S_X x + \eta_x$ of the angles and speeds in x can also be incorporated if available.

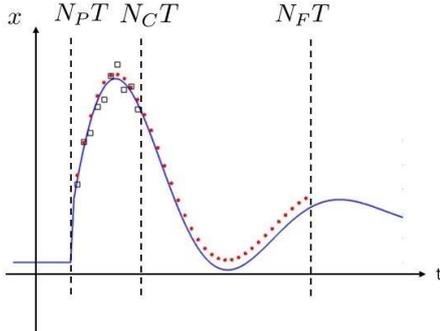


Figure 1: Online predictor estimates the trajectory up to a future time TN_F from measurements taken in a time window (TN_P, TN_C) .

Fig. 1 depicts the time-line of the online prediction problem, where the actual trajectory is represented with a continuous blue line. Let T_C denotes the current time and $[T_P, T_C]$ the past interval where data is collected. The future prediction horizon is represented by T_F . Such a trajectory is sampled, with sampling period T , so that past, current and future time instants are expressed in time periods $N_P = T_P/T$, $N_C = T_C/T$ and $N_F = T_F/T$, respectively. Noisy PMU measurements $\bar{z}_i, i \in \{N_P, \dots, N_C\}$ are depicted as black squares, and are available from time T_P to the current time T_C in which the prediction is carried out. These PMU data are utilized along with the system model to estimate the trajectory for both the past and future horizons. Red dots in Fig. 1 represents the prediction result. One tic T later the estimation is updated and the window shifted, taking into account the next PMU data and forgetting the oldest one. Indeed, each time new PMU data becomes available we solve the following trajectory prediction problem

Online trajectory prediction

$$\min_{x,v} r_{ss}(x,v) + ode(x,e) + alg(x,v) \quad (3)$$

The cost to be optimized consists of the sum of three terms. The first term measures the deviation of the estimation from the PMU measurements and will

be referred as *residual sum of squares*

$$r_{ss}(x,v) := \sum_{i=N_P}^{N_C} \alpha_i (\|\bar{z}_{V_i} - S_V v_i\|^2 + \|\bar{z}_{X_i} - S_X x_i\|^2) \quad (4)$$

Time varying coefficients α_i allow us to weight data uncertainty. The second and third terms capture the differential-algebraic model of the system, respectively.

$$ode(x,v) := \int_{T_P}^{T_F} \|\dot{x} - f_t(x,v)\|^2 \beta(t) dt \quad (5)$$

$$alg(x,v) := \int_{T_P}^{T_F} \|g_t(x,v)\|^2 \gamma(t) dt \quad (6)$$

At an actual trajectory both should be zero as long as the model is exactly known. However, the model may not be assumed exact since it includes several parameters that are, e.g., suddenly affected by system faults. Inclusion of these three terms in the cost allows to handle model uncertainties, as opposed to set the model equations as constraints. Functions f_t and g_t depend on machine parameters and the admittance matrix Y , which can be reasonably estimated with the help of protection devices in real time upon a given delay. Weight functions $\beta(t)$ and $\gamma(t)$ normalize the magnitude of dynamic and algebraic variables, and reflect our confidence in the model. They need tuning, using for instance cross validation.

Remark Observe that we have a case of *functional optimization*, where the optimization variable is the entire solution on the interval $[T_P, T_F]$ and the initial condition is the estimation for the previous window [9]. The estimation problem has two separate time intervals: a first one where measurements have great influence, $[TN_P, TN_C]$ and a second one where only the system's dynamics matters, $(TN_C, TN_F]$. The first part imposes a trade-off between the system's dynamics and the deviation from the measurements. For the second interval, the r_{ss} term is not involved and the solution of (3) can be solved by standard numerical integration methods (e.g. the trapezoidal method for differential-algebraic equations [8, p.859]) with initial condition the interpolation obtained from the past. However, the formulation of (3) as an optimization problem allows one to incorporate constraints based on field-experience, and to pursue convex methods based on recent reformulations of the optimal flow problem [12], although this is a direction for future research and will not be addressed here.

The algorithm developed in this paper requires the cost in (3) to be discretized, and that is the goal of next section.

3 THE COST WITH MATRIX VARIABLES

In this section we rewrite the proposed predictor in terms of matrix valued variables X and V . Such a matrix representation not only reflects on a compact

form of the discretized cost in (3), but also reduces the numerical complexity of its derivatives. As it is detailed in the next subsection, the optimization variables X and V collect all system-wide machine angles and speeds as well as complex voltages at a set of discrete time instants in the interval $[T_P, T_F]$.

For such a construction, we will use the Kronecker product and the Hadamard (entrywise) product, represented respectively by \otimes and \circ . Let Id_n be the identity matrix of size n , $\mathbf{0}_{n \times m}$ denote the zero matrix of dimensions $n \times m$ and $\mathbf{1}_n$ be the vector whose n elements are equal to one. A^T stands for the transpose matrix of A , and $\|A\|_F = \sqrt{\text{trace}(A^T A)}$ is its Frobenius norm.

3.1 Matrix variables

Let $\delta_m(t)$ denote the angle trajectory of the m -th machine. Consider a constant time step h chosen as a sub-multiple of the PMUs' period and discretize $\delta_m(t)$, collecting samples $\delta_{mn} := \delta_m((n-1)h + T_P)$ for $n = 1, \dots, N$. The number of samples is selected as $N = (T/h)(N_F - N_P) + 1$ so that yields $\delta_{m1} = \delta_m(T_P)$ and $\delta_{mN} = \delta_m(T_F)$, covering the time interval of interest. The same procedure is carried out to obtain samples of the machine speed trajectory $w_m(t)$. Then, samples δ_{mn} and w_{mn} are collected in the columns of the $N \times 2N_M$ matrix.

$$X = \begin{bmatrix} \delta_{11} & \omega_{11} & \dots & \delta_{1,N_M} & \omega_{1,N_M} \\ \delta_{21} & \omega_{21} & \dots & \delta_{2,N_M} & \omega_{2,N_M} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \delta_{N1} & \omega_{N1} & \dots & \delta_{N,N_M} & \omega_{N,N_M} \end{bmatrix}.$$

Recall that by construction the time indexes $1, \dots, N$ are relative to the current time interval $[T_P, T_F]$. In the same fashion, define V_{nm}^R and V_{nm}^I , $m = 1, \dots, N_B$ the real and imaginary parts of the Cartesian description of voltage phasors. Then build the $N \times 2N_B$ matrix

$$V = \begin{bmatrix} V_{11}^R & V_{11}^I & \dots & V_{1,N_B}^R & V_{1,N_B}^I \\ V_{21}^R & V_{21}^I & \dots & V_{2,N_B}^R & V_{2,N_B}^I \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ V_{N1}^R & V_{N1}^I & \dots & V_{N,N_B}^R & V_{N,N_B}^I \end{bmatrix}.$$

Besides, α , β and γ are vectors that collect the weight values at discretized times. Henceforth, we will use capital letters in $RSS(X, V)$, $ODE(X, V)$, and $ALG(X, V)$, for the functions that substitute (4), (5), and (6), respectively, after replacing continuous-time variables (x, v) for their matrix-valued counterparts.

3.2 Residual sum of squares

In the new variables, the first term of (3) can be rewritten as

$$RSS(X, V) = \frac{1}{2} \| \text{Diag}(\sqrt{\alpha})(\bar{Z}_X - S_T X S_X) \|_F^2 \quad (7) \\ + \frac{1}{2} \| \text{Diag}(\sqrt{\alpha})(\bar{Z}_V - S_T V S_V) \|_F^2.$$

with \bar{Z}_X , \bar{Z}_V , S_T , S_X and S_V defined as follows. S_T is the sub-sampling matrix that selects samples for those instants in which a PMU measurement is taken, and it is obtained by sampling the rows of the identity matrix Id_N . Matrix S_V selects the buses with PMUs and S_X selects the measured angles or speeds, if any. Matrices \bar{Z}_X and \bar{Z}_V contain the measurements.

3.3 Dynamic term

By discretizing the derivatives and the integral, the second term in (3) can be rewritten as

$$ODE(X, V) = \frac{h}{2} \| \text{Diag}(\sqrt{\beta})(DX - CF) \|_F^2 \quad (8)$$

Function $F := F(X, V) \in \mathbb{R}^{N \times 2N_M}$ is a matrix valued version of the swing equation, containing the dynamic information for every machine and for every time instant. We approximate the derivative in (3) with the trapezoidal rule (see e.g [8], page 842), and the integrals with quadrature methods. Differential matrix D implements the forward incremental quotient while matrix C aggregates the rows of F corresponding to subsequent time instants, according to the quadrature method of choice (e.g., Euler or trapezoidal rule). Details of function F and operators D and C are given in the Appendix.

3.4 Algebraic term

The algebraic term of (3) can be approximated by

$$ALG(X, V) = \frac{h}{2} \| \text{Diag}(\sqrt{\gamma})(I_m - I_{NET}) \|_F^2 \quad (9)$$

Matrix-valued functions $I_m = I_m(X, V)$ and $I_{NET} = I_{NET}(X, V)$ represent the current injected by machines and flowing to the network, respectively. Expressions for these functions may be found in the Appendix.

3.5 Algorithm

So far, we have expressed the cost in (3) in terms of structured matrices X and V . For each PMU cycle N_C , new data become available and we solve (3) in its discretized matrix form

$$[\hat{X}_{N_C}, \hat{V}_{N_C}] = \arg \min_{X, V} U(X, V) \quad (10)$$

with $U(X, V) := RSS(X, V) + ODE(X, V) + ALG(X, V)$, using a standard gradient-descent algorithm. In the next PMU cycle $N_C + 1$, the previous

solution $[\hat{X}_{NC}, \hat{V}_{NC}]$ is set as the initial seed for the next optimization iteration in (10), in order to speed convergence. The gradient-descent algorithm uses the expressions for the derivatives $\frac{\partial U}{\partial X}$ and $\frac{\partial U}{\partial V}$ with respect to matrices X and V that can be found in the Appendix.

4 SIMULATED EXAMPLE

In [10] we applied a preliminary version of the algorithm to a one-machine-infinite-bus (OMIB) as a proof of concept. In this Section, we move forward to a networked setup. The classical IEEE New England 39-bus system is adopted as the testbed for the numerical example here, with generators modeled with second order dynamics, see Fig. 2. The model includes 10 generators, 19 loads, and 36 transmission lines. The system trajectories were obtained with the DSAT software and integrating (1) by the standard methods in [8, p.859].

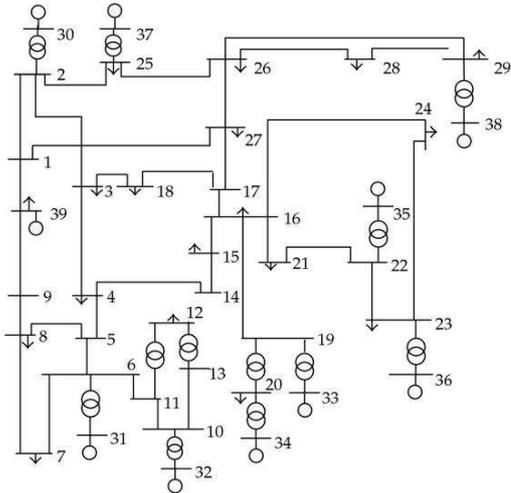


Figure 2: New England 39-bus system.

We simulate a three phase fault in the middle point of the line connecting buses 26 and 29. The DSAT trajectories will serve as the ground truth for the prediction algorithm. Fig. 3 represents such ground-truth trajectories for the generator at bus 38, which is the closest one to the fault. Starting from equilibrium at $t = 0$ it remains invariant until $t = 300$ ms, when the fault occurs. From then until the line is tripped at $t = 500$ ms the voltage drops.

At $t = 700$ ms the line is reclosed and the system transits back toward equilibrium, up to a time horizon of $t = 1$ s. The protection system reports the location and time of the fault, with a delay after the short circuit, (dashed line at $t = 400$ ms in Fig. 3) and the tripping event at time $t = 500$ ms.

Data for the predictor are collected by $N_M = 10$ PMUs, placed at the generator buses, which sample voltage phasors every 20ms. Readings from these

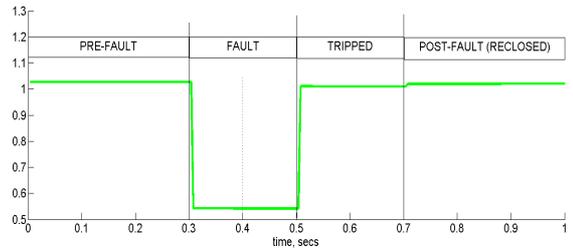


Figure 3: Ground-truth voltage at bus 38 and fault time-line.

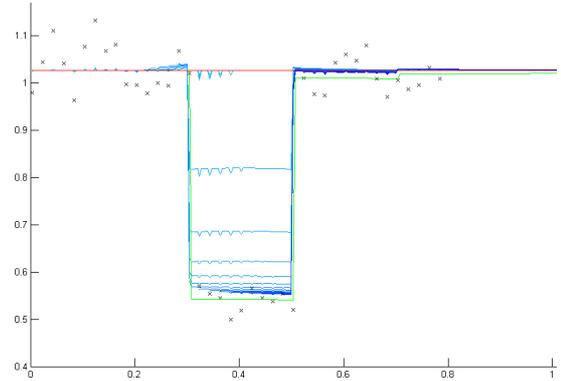


Figure 4: Voltage of busbar 38.

PMUs were simulated by adding 5 percent of white noise to samples of the ground truth trajectory. In applications, the optimal location of PMUs is depends heavily on the fault to be considered. This factor and the number of PMUs will affect the error of the algorithm.

Figure 4 shows together the ground-truth trajectory (green line), the initial guess for the algorithm (red line) all the trajectories successively estimated along the transient (blue lines). The speed of generator at bus 38 is shown in a similar way in Fig. 5. Both the machine acceleration due to the fault (green line) and the corresponding predictions (blue lines) can be appreciated.

The top plot in Fig. 6 depicts the PMU data (black crosses), the ground-truth trajectory (green line), and the prediction carried out at time $t = 380$ ms (blue

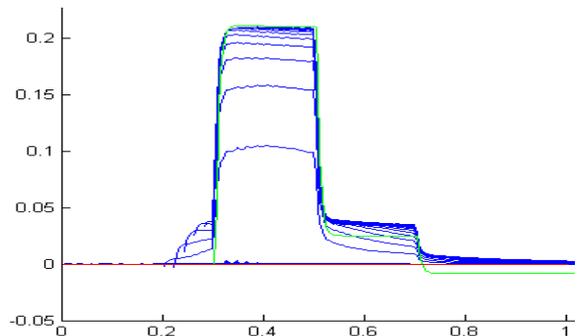


Figure 5: Speed of machine at busbar 38.

line). The PMU data at fault (300 – 380ms) are not enough yet to force the prediction to follow the incoming voltage dip. The second plot in Fig. 6 shows the prediction at time $t = 400$ ms. The fault location was reported at $t = 400$ ms and the system model updated accordingly for the prediction. The estimation has improved because more PMU data on fault are considered and the correct model has been already updated.

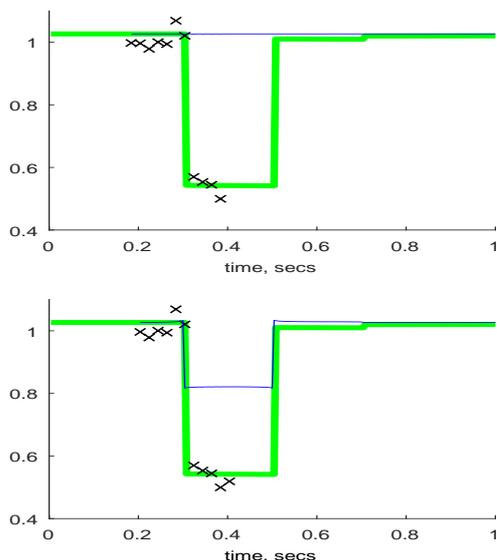


Figure 6: Progression of the voltage estimation of bus-bar 38. Predictions at 380, 400 ms.

The upper plot in Fig. 7 shows the situation right before the tripping ($t = 480$ ms). New data has been introduced and the voltage deep is now more accurately described. The programmed tripping time is known by the algorithm so that the trajectory prediction after tripping approximates the actual voltage. The bottom plot shows the situation after the line tripping ($t = 620$ ms). Again, the extra data collected renders the prediction more accurate. These figures corroborate the ability of (3) to smooth the data by using the dynamic model to average multiple data samples and hence reducing their noise effect.

5 CONCLUSIONS AND FUTURE WORK

A novel algorithm was presented for the prediction of trajectories in power systems. The technique leverages the full information available at each time instant, including the PMU data and the differential algebraic system model. Two distinctive properties of the proposed predictor with respect to available alternatives are that it can incorporate real-time data adaptively, while accommodating model errors and uncertainties. A matrix formulation of the prediction problem facilitates the algorithm construction for large power networks, and paves the way for the use of state-of-the-art optimization methods. Numerical experiments corroborate the ability of the proposed estimator to

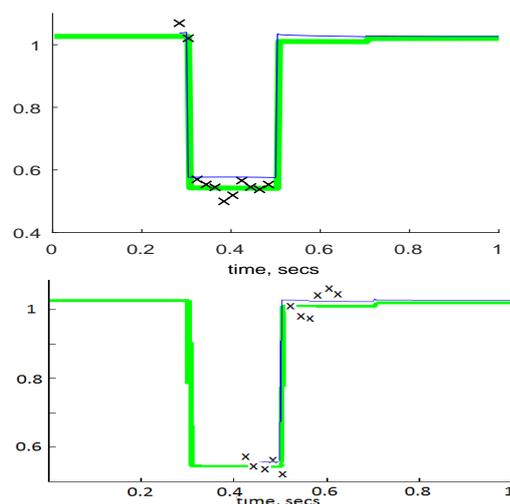


Figure 7: Progression of the voltage estimation of bus-bar 38. Predictions at 480, 620 ms.

smooth the data, to capture the networked dynamics of the grid, and to incorporate newly available phasor measurements to improve the results adaptively.

Next research steps will explore new optimization algorithms and dimensionality reduction techniques, as well as decentralized versions via in-network consensus. We will pursue convex methods to solve (3) based on recent reformulations of the optimal flow problem [12]. These approaches, along with migration of the algorithm code to a compiled language, will allow us to investigate the tradeoff between error performance and execution time.

References

- [1] Gómez-Expósito, A. and Abur, A. "Foreword for the Special Section on Synchrophasors Applications in Power systems", *IEEE Transactions on Power systems*, May 2013.
- [2] X. Wu and J. Zhao and A. Xu and H. Deng and P. Xu, "Review on Transient stability Prediction Methods based on Real Time Wide -area Phasor Measurements", *4th Intl. Conf. on Electric Utility Deregulation and Restructuring and Power Tech. (DRPT)*, 2011.
- [3] J. Cepeda, J. Rueda, D. Colomé, D. Echeverría, "Real-time transient stability assessment based on centre-of-inertia estimation from phasor measurement unit records," *IET Generation, Transmission and Distribution*, vol.8, no.8, pp.1363-1376, 2014.
- [4] C.-W. Liu, J.S. Thorp, " New Methods for Computing Power System Dynamic Response for Real-Time Transient Stability Prediction", *IEEE Trans. on Circuits and Systems-I:Fund. Theory and Appl.*, v. 47, n.3, March 2000.

- [5] S. Smith, C. Woodward, M. Liang, J. Chaoyang, A. del Rosso, "On-line transient stability analysis using high performance computing," *IEEE PES ISGT*, 2014.
- [6] Anjia Mao, "Ball Catching: the Inspiration to Power System Stability Control, A Fast Algorithm for the Generator's Disturbed Trajectory Prediction," *IEEE Power Engineering Society General Meeting*, vol., no., pp.1,6, 24-28 June 2007
- [7] Z. Wang and Z. Guo, "A Fast Method for Transient Stability Assessment Based on Taylor Series Expansion," *IEEE/PES T&D Conference and Exhibition: Asia and Pacific*, 2005.
- [8] P. Kundur. "Power System Stability and Control", Mc. Graw-Hill, 1994.
- [9] R. García and V. Pérez-García, "Solving Functional and Differential Equations With Constraints Via an Optimization Approach," *International Conference on Mathematical and Statistical Modeling*, 2006.
- [10] XX, YYY, ZZZ, WWWW, "Online prediction of power systems trajectories from noisy data by penalized least-squares minimization", *IEEE ISGT-LA*, Oct. 2015.
- [11] K.B. Petersen and M.S. Petersen. "The Matrix Cookbook", 2012 (<http://matrixcookbook.com>).
- [12] S. H. Low Convex relaxation of optimal power flow Part I: Formulations and equivalence, *IEEE Transactions on Control of Network Systems*, vol 1, no.1 pp. 15-27, 2014.

A Derivation of the ODF and ALG parts of the cost

Function $F(X, V) : \mathbb{R}^{(N \times 2N_M) \times (N \times 2N_B)} \rightarrow \mathbb{R}^{N \times 2N_M}$ encapsulates the system dynamics. It is obtained by looking at the equations for a single machine connected to a busbar, as it is shown next. In what follows, we will express complex numbers as vectors in \mathbb{R}^2 . For this case, the link variable is $\bar{V} = [V^R, V^I]^T$, the complex voltage phasor at terminal bus, and the state vector is $[\delta, \omega]^T$. Denote by \bar{E}_m and $B_m = \frac{1}{X_d}$ the internal fem and the transient susceptance of the machine, respectively. The internal fem is written as $\bar{E}_m = E_m[\cos(\delta), \sin(\delta)]^T$, and the current \bar{I} injected by the machine to the network is $\bar{I} = \frac{1}{jX_d}(\bar{E}_m - \bar{V}) = -jB_m(\bar{E}_m - \bar{V})$. This complex quantity $I = [I^R, I^I]^T$ can be written as an \mathbb{R}^2 vector

$$\bar{I} = -B_m J(\bar{E}_m - \bar{V}) = -B_m J \begin{bmatrix} \bar{E}_m^R - \bar{V}^R \\ \bar{E}_m^I - \bar{V}^I \end{bmatrix}$$

with the help of the auxiliary matrix $J = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}$. The electrical power injected by the machine turns out

to be $P_e = \bar{V}^T \bar{I}$:

$$P_e = \bar{V}^T(-B_m J(\bar{E}_m - \bar{V})) = -B_m \bar{V}^T J(\bar{E}_m - \bar{V}).$$

Thus, the swing equation of a given machine p can be expressed as

$$\begin{bmatrix} \dot{\delta}_p \\ \dot{\omega}_p \end{bmatrix} = \begin{bmatrix} \omega_p \\ \frac{\omega_0}{2H_p} [P_{mp} - K_{Dp}\omega_p + B_{mp}\bar{V}^T J(\bar{E}_{mp} - \bar{V})] \end{bmatrix}. \quad (11)$$

Then, we construct the function F , putting

$$[F(X, V)_{n,2p-1} \ F(X, V)_{n,2p}] = [\dot{\delta}_p(nh) \ \dot{\omega}_p(nh)]$$

for $n = 1, \dots, N$ and $p = 1, \dots, N_M$. For each machine p , and considering their own parameters, we define the scalar constants $\bar{K}_{Dp} = \frac{\omega_0}{2H_p} K_{Dp}$, $\rho_p = \frac{\omega_0}{2H_p} P_{mp}$ and construct the respective $N_M \times 1$ vectors K_D and ρ . We consider the numbers $K_p = \frac{\omega_0}{2H_p}$ and B_{mp} , and we form the $N_B \times N_B$ diagonal matrices K and B_m , padding with zeros for non machine buses. Finally, putting all together, we may write

$$\begin{aligned} F(X, V) &= X \left(Id_{N_M} \otimes \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix} \right) \\ &+ X \left(Diag(K_D) \otimes \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} \right) + \mathbf{1}_N (\rho \otimes [0 \ 1]) \\ &+ [V \circ I_m] \left(K \otimes \begin{bmatrix} 0 & 1 \\ 0 & 1 \end{bmatrix} \right) \left[\frac{Id_{2N_M}}{\mathbf{0}_{2(N_B - N_M) \times 2N_M}} \right] \end{aligned}$$

where

$$I_m = (E_m - V) \cdot [Id_{N_B} \otimes J^T] \cdot [B_m \otimes Id_2]$$

is the current injected by the machines (i.e., it is 0 for buses without generators). Vector E_m contains the internal voltage of the machines, padded with zeros for non generator buses.

For the algebraic term ALG , the current I_{NET} must be derived. For a given time instant n , the network (complex) equation is $I_{NET,n} = V_n Y$, where Y is the admittance matrix of the electrical network, with dimensions $(N_B \times N_B)$, $I_{NET,n}$ is a $(1 \times N_B)$ vector of net currents entering the power system and V_n is the $1 \times N_B$ vector of complex bus voltages. At a fixed instant, for a given bus q , and considering the real and imaginary parts of Y , we have that

$$I_q = \sum_{k=1}^{N_B} Y_{qk} V_k \Rightarrow$$

$$I_q^R + jI_q^I = \sum_{k=1}^{N_B} (Y_{qk}^R + jY_{qk}^I)(V_k^R + jV_k^I),$$

so that $I_q^R = \sum_{k=1}^{N_B} Y_{qk}^R V_k^R - Y_{qk}^I V_k^I$ and $I_q^I = \sum_{k=1}^{N_B} Y_{qk}^I V_k^R + Y_{qk}^R V_k^I$. We put the real and imaginary parts in a 1×2 row vector, obtaining

$$\bar{I}_q = [I_q^R \ I_q^I] = \sum_{k=1}^{N_B} [V_k^R \ V_k^I] (Y_{qk}^R Id_2 + Y_{qk}^I J^T).$$

These are collected in an $N \times 2N_B$ matrix

$$I_{NET} = V (Y^R \otimes Id_2 + Y^I \otimes J^T).$$

The previous expression can be easily extended to incorporate time changes in the network admittance matrix. As an example, suppose the power system faces a fault. Then, after the protection system informs the event and trigger the respective contingency actions, will know that there will be three different admittance matrices: pre-fault (Y_P), fault (Y_F) and post-fault (Y_{PF}). Defining $\Delta Y_F = Y_F - Y_P$ and $\Delta Y_{PF} = Y_{PF} - Y_P$, we obtain a unique admittance matrix

$$Y_n = Y_P + \mu_n \Delta Y_F + \nu_n \Delta Y_{PF}$$

where μ_n and ν_n indicate the active matrix at every instant. Then, we may write

$$\begin{aligned} I_{NET} = & V (Y_{PR} \otimes Id_2 + Y_{PI} \otimes J^T) + \\ & Diag(\mu)V (\Delta Y_{FR} \otimes Id_2 + \Delta Y_{FI} \otimes J^T) + \\ & Diag(\nu)V (\Delta Y_{PFR} \otimes Id_2 + \Delta Y_{PFI} \otimes J^T) \end{aligned}$$

Observe that we can accommodate any finite number of distinct admittance matrices.

B Derivatives required for the optimization algorithm

In Section III, a matrix formulation of the optimization problem was introduced, with particularly structured variables X and V , involving Kronecker and Hadamard products. In order to apply an iterative algorithm, the derivatives of the cost function with respect to X and V are needed. We could not find explicit expressions for the required derivatives in the literature, so we present here a brief derivation of one of them. The others can be obtained in a similar way. We start from [11]

$$\frac{d}{dX} tr(A^T X) = A \quad (12)$$

which is straightforward to prove from the definition of the trace. Similarly, for Hadamard product $\frac{d}{dX} tr(A \circ X) = Diag(A)$. A basic result for the derivatives we are interested in states that

$$\frac{d}{dX} tr[A^T (B \circ X)] = A \circ B$$

In order to prove it, we observe that

$$\begin{aligned} tr[A^T (B \circ X)] &= \sum_j [A^T (B \circ X)]_{jj} = \sum_j \sum_i a_{ij} x_{ij} b_{ij} \\ &= tr[(A \circ B)^T X] \end{aligned}$$

The result follows from (12). In a similar way, since

$$tr[A^T (B \circ (XC))] = tr[(A \circ B)^T XC] = tr[C(A \circ B)^T X]$$

we have that

$$\frac{d}{dX} tr[A^T (B \circ XC)] = (A \circ B)C^T \quad (13)$$

which will be our building block from which we obtain all the derivatives we need. As an example, we present the derivative of $t(X) = tr[A^T (XE \circ B)^T D(XF \circ C)]$, a kind of term that appears in ODF and ALG. Let us write down the difference $t(X + dX) - t(X)$ where dX is a small deviation from X . From it, we will obtain the sought derivative. Then $t(X + dX) =$

$$tr[A^T (((X + dX)E) \circ B)^T D(((X + dX)F) \circ C)]$$

We will keep first order terms, dropping higher order terms. It follows that

$$\begin{aligned} t(X + dX) \approx & t(X) + tr[A^T (XE \circ B)^T D(dXF \circ C)] \\ & + tr[A^T (dXE \circ B)^T D(XF \circ C)] + o(dX^2) \end{aligned}$$

which is equal to

$$\begin{aligned} t(X + dX) \approx & t(X) + tr[A^T (XE \circ B)^T D(dXF \circ C)] + \\ & tr[A(XF \circ C)^T D^T(dXE \circ B)] + o(dX^2) \end{aligned}$$

Then, using (13),

$$\begin{aligned} \frac{\partial}{\partial X} tr[A^T (XE \circ B)^T D(XF \circ C)] = & \\ & [(D^T((XE) \circ B)A) \circ C] F^T + \\ & + [(D((XF) \circ C)A^T) \circ B] E^T. \end{aligned}$$

Table I summarizes all the formulae needed by the optimization algorithm.

Since the internal fem of a given machine depends on the rotor angle, matrix E_m depends on variable X . Then, to complete the task, we need an expression for the derivative of a function $h(E_m)$ with respect to X . We focus on a general function $h(E(X))$, $h : \mathbb{R}^{N \times 2P} \rightarrow \mathbb{R}$. In our particular case, we have a special dependence of E on X . In fact, the elements $E_{m_{n,2p-1}}$ and $E_{m_{n,2p}}$, which are actually $E_{m_{np}}^R$ and $E_{m_{np}}^I$, depend only on δ_{np} , the angle of the machine p at time n . We have that

$$\frac{E_{m_{np}}^R}{\partial \delta_{np}} = -E_{m_{np}}^I, \quad \frac{E_{m_{np}}^I}{\partial \delta_{np}} = E_{m_{np}}^R$$

or, in a matricial form,

$$\frac{\partial}{\partial \delta_{np}} \begin{bmatrix} E_{m_{np}}^R \\ E_{m_{np}}^I \end{bmatrix} = \begin{bmatrix} -E_{m_{np}}^I \\ E_{m_{np}}^R \end{bmatrix} = J \begin{bmatrix} E_{m_{np}}^R \\ E_{m_{np}}^I \end{bmatrix}$$

Then, discarding zero terms,

$$\frac{\partial h}{\partial \delta_{np}} = \left[\frac{\partial h}{\partial E_{m_{np}}^R} \quad \frac{\partial h}{\partial E_{m_{np}}^I} \right] \cdot \begin{bmatrix} \frac{\partial E_{m_{np}}^R}{\partial \delta_{np}} \\ \frac{\partial E_{m_{np}}^I}{\partial \delta_{np}} \end{bmatrix} =$$

Table 1: Summary of the derivative formulae

Scalar function of X	Derivative
$tr [(A^T(B \circ X C))]$	$(A \circ B)C^T$
$tr [A^T(XE \circ B)^T D(XF \circ C)]$	$[(D^T((XE) \circ B)A) \circ C] F^T + [(D((XF) \circ C)A^T) \circ B] E^T$
$\frac{1}{2} \ D(XE \circ B) A\ _F^2$	$A \circ (D^T D(X \circ A) B B^T) E^T$
$tr [A(XE \circ B)^T D(XF \circ C)]$	$[(D^T(XE \circ B)A) \circ C] F^T + [(D^T(XE \circ B)A) \circ XF] C^T + [(D(XE \circ XE)A^T) \circ B] E^T$
$\frac{1}{2} \ D(X \circ (XA))B\ _F^2$	$[(D^T D(X \circ XA) B B^T) \circ X A] + [(D^T D(X \circ (XA)) B B^T) \circ X] A^T$

$$= \frac{\partial h}{\partial E_{mnp}^R} \cdot (-E_{mnp}^I) + \frac{\partial h}{\partial E_{mnp}^I} \cdot E_{mnp}^R.$$

So, in order to get the $N \times 2P$ matrix $\frac{\partial h}{\partial X}$, we may proceed as follows. First, we introduce the auxiliary matrix \tilde{E} which reorders the entries of E :

$$[\tilde{E}_{n,2p-1}, \tilde{E}_{n,2p}] = [-E_{n,2p}, E_{n,2p-1}].$$

Thus,

$$\tilde{E} = E \cdot (Id_P \otimes J^T).$$

Then, we have to construct the dependence of h on δ and add 0 for the dependence of E on ω .

$$\frac{\partial h}{\partial X} = \left\{ \frac{\partial h}{\partial E} \circ \tilde{E} \right\} \cdot \left(Id_P \otimes \begin{bmatrix} 1 & 0 \\ 1 & 0 \end{bmatrix} \right)$$

or

$$\begin{aligned} \frac{\partial h}{\partial X} &= \left\{ \frac{\partial h}{\partial E} \circ [E \cdot (Id_P \otimes J^T)] \right\} \cdot \left(Id_P \otimes \begin{bmatrix} 1 & 0 \\ 1 & 0 \end{bmatrix} \right) \\ &\triangleq \left(\frac{\partial h}{\partial E} \circ K_{E_1} \right) \cdot K_{E_2} \end{aligned}$$

As an application of the previous result, we may now calculate the derivative of $h(X) = tr [X^T A (V \circ (E(X)B)) C]$. We know that $\partial h = \partial tr [X^T A (V \circ (E(X)B)) C]$, so

$$\begin{aligned} \partial h &= tr [\partial (X^T A (V \circ (E(X)B)) C)] \\ &= tr [\partial X^T A (V \circ (E(X)B)) C] + \\ &\quad tr [X^T A \partial \{(V \circ (E(X)B)) C\}] \\ &= tr [\partial X^T A (V \circ (E(X)B)) C] + \\ &\quad tr [X^T A \partial \{(V \circ (E(X)B)) C\}] \end{aligned}$$

The derivative of the first term comes from Table I. For the second term, we may apply the *chain-rule-kind* previous result:

$$\begin{aligned} \frac{\partial h}{\partial X} &= A (V \circ (EB)) C \\ &+ \left\{ \frac{\partial}{\partial E} tr [X^T A (V \circ (EB)) C] \circ K_{E_1} \right\} \cdot K_{E_2} \end{aligned}$$

Finally,

$$\begin{aligned} \frac{\partial h}{\partial X} &= A (V \circ (EB)) C + \\ &\quad \{ [(A^T X C) \circ V] B^T \} \circ K_{E_1} \cdot K_{E_2} \end{aligned}$$