Spoofing GPS Receiver Clock Offset of Phasor Measurement Units[1]

A. D. Domínguez-García

Department of Electrical and Computer Engineering
University of Illinois at Urbana-Champaign

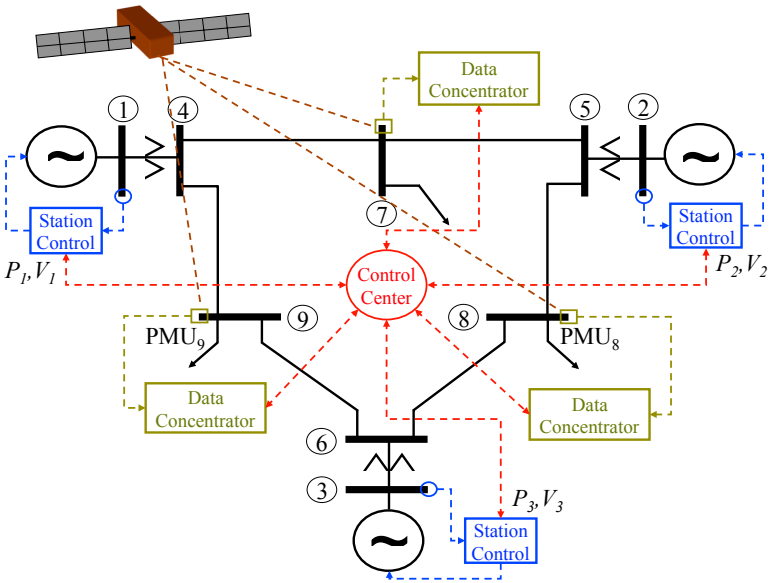
TCIPG Seminar Series on Technologies for a Resilient Power Grid
Urbana, IL
October 3, 2014

[1] X. Jiang, J. Zhang, B. Harding, J. Makela, and A. D. Domínguez-García, "Spoofing GPS Receiver Clock Offset of Phasor Measurement Units," IEEE Trans. on Power Systems, 2013.

Outline

Phasor Measurement Unit (PMU) Basics

- Loosely speaking, a PMU provides a phasor description of a physical quantity in a power network, e.g., voltage, under the assumption that
  - The quantity time dependence is accurately described by a sinusoid
- Let $v_i(t)$ denote the voltage in some bus of a power network, then
  - Time domain representation: $v_i(t) = \sqrt{2} V_i \cos(\omega t + \theta_i)$
  - Phasor representation: $\overline{V}_i = V_i \angle \theta_i$



- PMUs are equipped with a GPS receiver to derive a time stamp in Coordinated Universal Time (UTC); this provides
  - A common time reference to the phase angle measurements across a wide area

Motivation

- A GPS receiver acquires signals from satellites, decodes each satellite's navigation data, and estimates its position and UTC
- A spoofing attack on a GPS receiver can induce a faulty time stamp, which introduces errors in the PMU's phase measurements
- Such an attack can be implemented with a GPS simulator to generate rogue signals matching the genuine signals [Humphreys et al. '09]
- In this presentation:
  - We demonstrate the feasibility of a spoofing attack on the GPS receiver of a PMU
  - We show that such an attack can cause significant errors in the phase measurements provided by the PMUs

The Landscape Today[a]

[a] Preliminary Special Reliability Assessment Whitepaper: Extended Loss of GPS Impact on Reliability, NERC

- *PMUs are mostly used for monitoring*
- *The electric industry is testing the use of PMUs to support operations*
- *PMUs are* **not** *used in any real-time closed-loop control application, e.g., automatic generation control*

Outline

GPS Receiver Position and Time Synchronization Basics

- A GPS receiver determines its distance from a satellite by
  - estimating the signal travel time, and
  - multiplying that by the speed of signal propagation (speed of light)
- Given the satellites' positions and ranges from receiver, the receiver location could be computed through a process known as trilateration

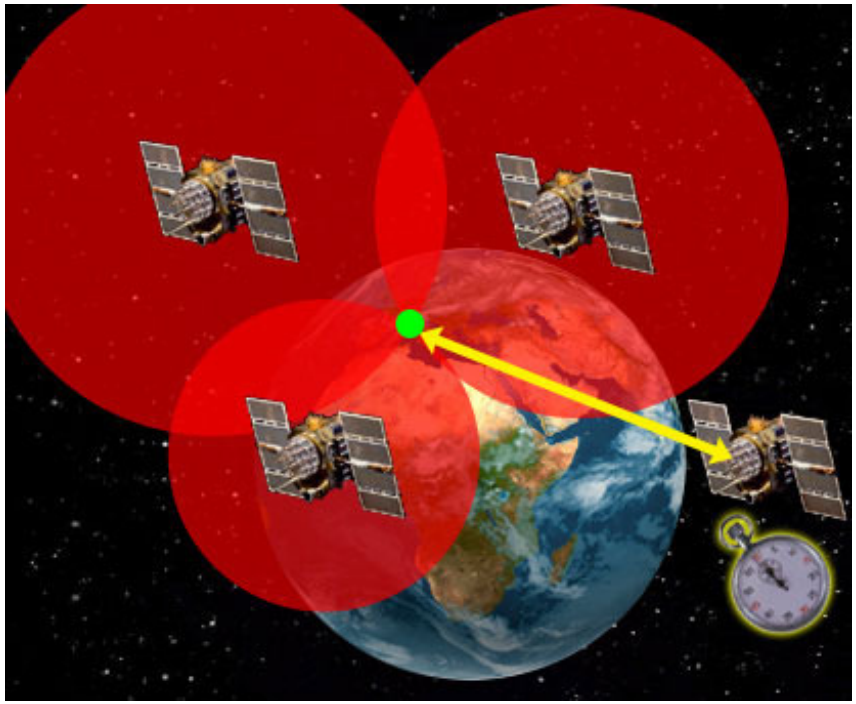


Figure: Trilateration

GPS Receiver Position and Time Synchronization Basics

- In theory, three satellites are sufficient to determine the receiver's exact location, assuming:
  - no noise in the measurements
  - time between satellites' clocks and receiver's are perfectly synchronized

- In reality, the receiver clock has an offset t_u from the GPS time t_E arising from internal hardware bias in the local clock oscillator

- We can express the clock offset as:

$$t_u = t_r - t_E, \tag{1}$$

where t_r denotes the receiver clock time

GPS Receiver Position and Time Synchronization Basics

- The t_{UTC} is offset from GPS time $t_E = t_r - t_u$ by an integer number of leap seconds $\Delta t_{UTC} = 16\ s$

- Therefore, t_{UTC}, which is used for PMU time synchronization, is computed as follows:

$$t_{UTC} = t_E - \Delta t_{UTC} \tag{2}$$

- A voltage phasor is measured at each bus and time-stamped using the reference time signal t_{UTC}

- This time-stamp is common to all buses and provides the synchronization of the PMUs' phasor measurements

Four Visible Satellites

- Let ρ_i and r_i be the i^{th} satellite's pseudorange and true range
- Let x_i, y_i, and z_i be the i^{th} satellite's ECEF coordinates
- Let x_u, y_u, z_u, be the receiver's ECEF coordinates
- Let t_u be the receiver clock offset. Then,

$$\rho_i = r_i - ct_u, \quad i = 1, 2, 3, 4 \tag{3}$$

$$r_i = \sqrt{(x_i - x_u)^2 + (y_i - y_u)^2 + (z_i - z_u)^2} \tag{4}$$

  where c denotes the speed of light

- The GPS receiver computes x_i, y_i, and z_i through a set of parameters contained in the GPS signal known as the ephemerides

More than Four Visible Satellites

- It is almost always the case that more than four satellites are visible at a particular instant of time (more equations than unknowns)

- The GPS receiver obtains x_u, y_u, z_u, and t_u by solving a Least Squares Errors Estimation (LSE) problem of the form:

$$\text{minimize} \quad f_0 = \sum_{i=1}^{n} (\rho_i - r_i + ct_u)^2 \tag{5}$$

  where $n > 4$ denotes the number of visible satellites

- We use GPS receivers mostly for locational purposes, i.e., we care about x_u, y_u, z_u

- PMUs care about the clock offset t_u

Keplerian Elements

- The accurate characterization of the GPS satellites' orbits is essential for determining the receiver's position
- In the absence of external perturbations, the trajectory of a satellite is solely governed by the gravitational force of Earth
- This trajectory is fully characterized by six parameters known as as *Keplerian elements*
- The Keplerian elements allow the receiver to compute the position and velocity vectors of the satellite at any point

Computation of Satellite's ECEF Coordinates

- In order to describe a satellite's orbit even more accurately, the additional forces acting on the satellite must be considered
- It is still possible to completely characterize the satellite's motion under full perturbation with the Keplerian elements
  - however, these are no longer constants
- In order to characterize how the Keplerian elements change over time, a set of parameters is added to the satellite's navigation signal
- This expanded parameter set which contains the Keplerian elements is known as the satellite's ephemerides
  - Up-to-date ephemerides are uploaded from the GPS control segment to the satellites once per day and then broadcast to the receiver
- The information contained in the ephemerides is used by the GPS receiver to compute the position of a satellite's ECEF coordinates

Outline

Attack Formulated as an Optimization Program

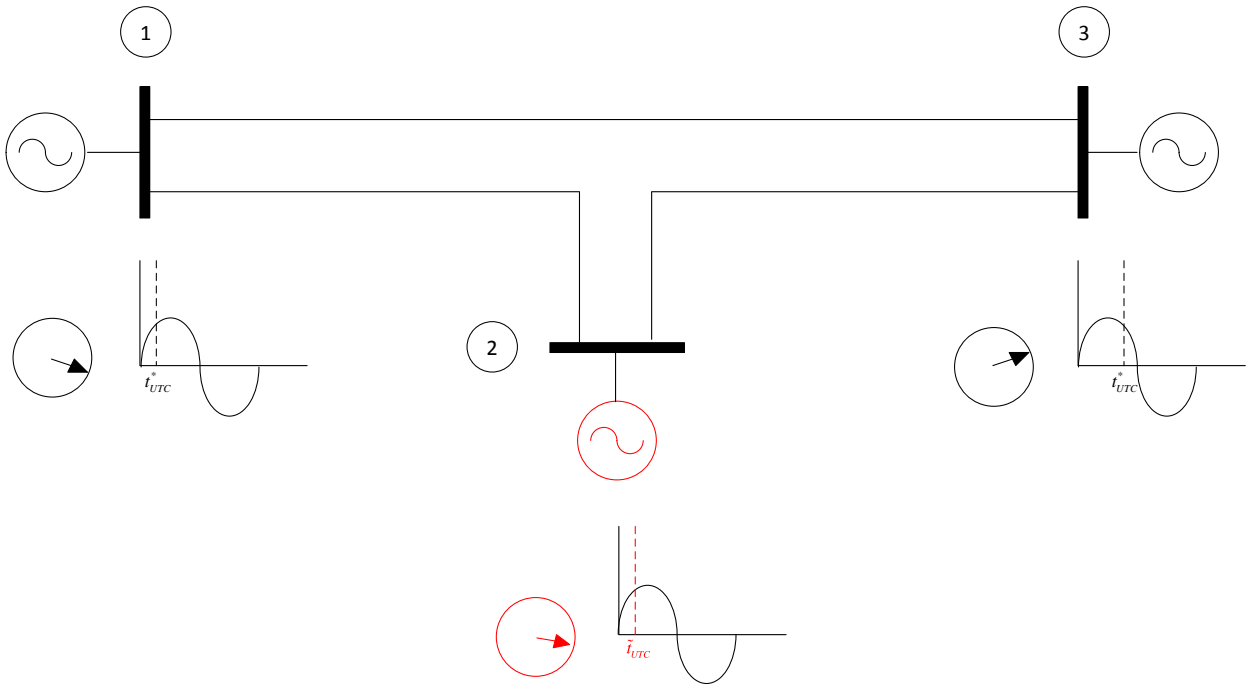
- The objective is to maximize the difference between the PMUs receiver clock offset before and after the attack
- The optimization is performed for a given instant in time, which is the time when the spoof is to be implemented
- The decision variables re the satellites' ephemerides, pseudoranges, and the receiver ECEF coordinates
- Additional constraints are added to capture the possibility that the GPS receiver may implement some form of spoofing detection scheme

Impact of Spoofing on PMU Phase Information

- Time synchronization across PMUs is crucial for maintaining an accurate measurement of phase angles

- For a 60-Hz signal, the PMU's phase measurement error ε_θ is related to the receiver clock offset error through the linear relationship

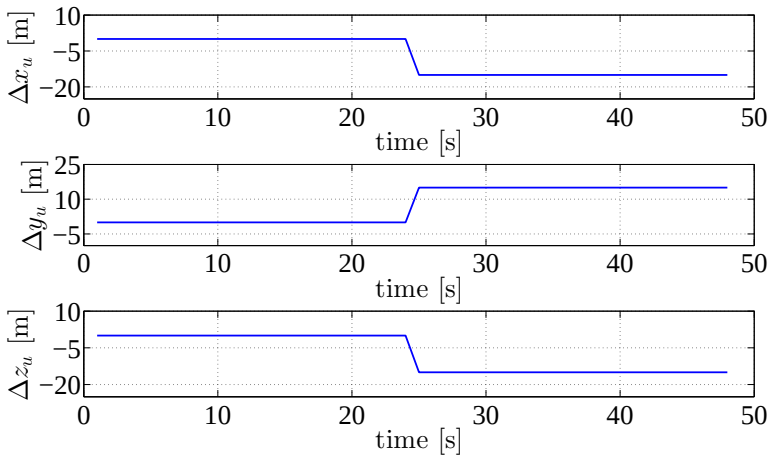
$$\varepsilon_\theta = [60 \times (t_u - t_u^*) \times 360^\circ] \text{ mod } 360^\circ \tag{6}$$



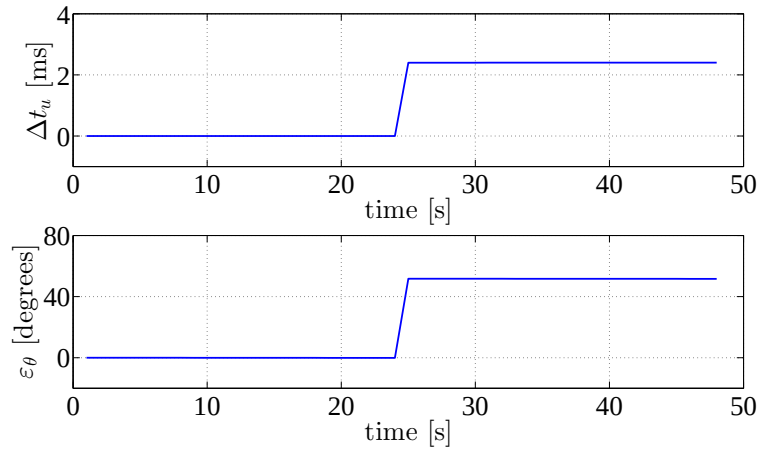
Spoofing Attack When Four Satellites are Visible

- Each of the satellites' ephemerides are limited to $\pm 2\%$ of their pre-attack values

- The GPS receiver location is also restricted to vary at most 15 m from its pre-attack position



(a) Receiver ECEF Coordinates.

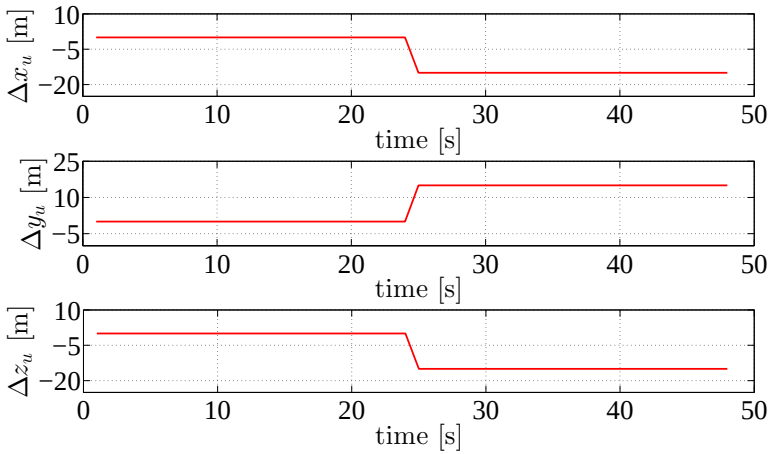


(b) Receiver clock offset and phase angle.

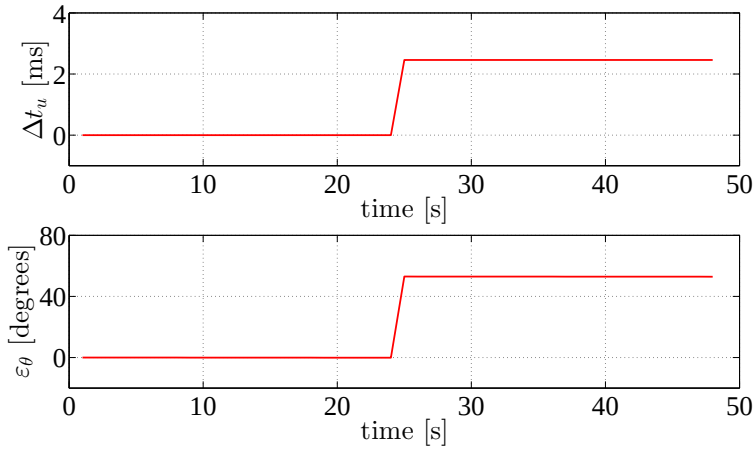
Figure: Receiver Position, Clock Offset, and PMU Phase Error

Spoofing Attack When Seven Satellites are Visible

- Each of the satellites' ephemerides are limited to $\pm 2\%$ of their pre-attack values
- The GPS receiver location is also restricted to vary at most 15 m from its pre-attack position



(a) Receiver ECEF Coordinates.



(b) Receiver clock offset and phase angle.

Figure: Receiver Position, Clock Offset, and PMU Phase Error for Spoofing Seven Satellites

Outline

Summary

- We formulated A GPS spoofing attack inflicting maximum time offset error in PMU receiver
- Attack can be implemented for arbitrary number of visible satellites
- However note that:
  - PMUs are mostly used for monitoring
  - The electric industry is testing the use of PMUs to support operations
  - PMUs are **not** used in any real-time closed-loop control application, e.g., automatic generation control