A Successful "Applications Oriented" Undergraduate EE Course Elective: Electronic Navigation Systems

Prof. Richard J. Hartnett P.E., U.S. Coast Guard Academy

Richard J. Hartnett is a professor of electrical engineering at the U.S. Coast Guard Academy in New London, CT. He received his B.S.E.E. degree from the U.S. Coast Guard Academy, his M.S.E.E. degree from Purdue University, and his Ph.D. in EE from the University of Rhode Island. He is a registered Professional Engineer in the State of Connecticut, and his research interests include efficient digital filtering methods, improved receiver signal processing techniques for electronic navigation systems, and autonomous vehicle design.

Cmdr. Kelly Charles Seals P.E., U.S. Coast Guard Academy

Commander Kelly Charles Seals is Program Chair for Electrical Engineering at the U.S. Coast Guard Academy. He has a Ph.D. in Electrical and Computer Engineering from Worcester Polytechnic Institute, a M.S. in Electrical Engineering from Northeastern University, and a B.S. in Electrical Engineering from the U.S. Coast Guard Academy from which he graduated in 1998. He also received a Certificate in College Teaching from the Colleges of Worcester Consortium.

Dr. Paul Benjamin Crilly, U.S. Coast Guard Academy

Paul Crilly is an Associate Professor of Electrical Engineering at the United States Coast Guard Academy. He received his Ph.D. from New Mexico State University, his M. S. and B.S. degrees at Rensselaer Polytechnic Institute, all in Electrical Engineering. He was previously an Associate Professor of Electrical and Computer Engineering at the University of Tennessee and was a Development Engineer at the Hewlett Packard Company. His areas of interest include laboratory development, antennas, wireless communications, signal processing, and instrumentation.

A Successful "Applications Oriented" Undergraduate EE Course Elective: Electronic Navigation Systems

Abstract

In typical undergraduate Electrical Engineering programs, students take courses in circuit analysis, electronics, programming, signals and systems, digital design, physics, calculus, differential equations, plus some advanced courses such as linear algebra, discrete mathematics, communications systems, digital signal processing, antennas, and control systems. These courses are important, however students sometimes find themselves wondering about specific "real-world" applications for all the theory and information learned in these courses. Certainly as faculty members we make every attempt to discuss applications in class, but the unsettling fact is that some students really want to learn more about "real-world" systems! The purpose of this paper is to present how we connect theory to practice using navigation systems as the particular application.

Several "real-world" systems discussed in our junior/senior elective course called Electronic Navigation Systems are highlighted in this paper. One system we learn is the NAVSTAR Global Positioning System (GPS). Students find that shift registers are used to generate a unique 1023 bit pseudo-random binary sequence for each satellite, and they write Matlab code to generate sequences for all satellites (digital design application). After telling our students about Gold codes, they then use their functions to study the auto and cross correlation properties of the sequences, and we highlight how a receiver might use those methods to estimate signal time of arrival for this Code Division Multiplexed Access (CDMA) system employing Binary Phase-Shift Keying (BPSK) signals (communication system application). Using the ranging information from N satellites, students learn that GPS receivers typically solve an over-determined system of N equations (where N>4) for 3-d position (x,y,z), plus receiver time offset (t) relative to GPS system time (linear algebra application).

Given actual satellite locations and ranging information, students then write an iterative algorithm to solve for receiver position plus time offset (linear algebra and programming application). Then we show our students that GPS accuracy is a function of satellite position (azimuth and elevation) relative to the user, and they are asked to write a Matlab function which estimates GPS accuracy as a function of visible satellite azimuth/elevations. Finally, we ask our students to think about using a GPS receiver to try to find a restaurant in a large city such as New York, NY, where the only satellites visible would be the ones directly overhead. They quickly realize that they might have to ask for directions rather than depend on their GPS receiver inside their cellular telephone for guidance (linear algebra application)! Other systems discussed include aircraft navigation systems such as VOR (VHF Omni-Directional Radio Range) (AM and FM communications application), and Instrument Landing System (ILS) (antenna, phased array application).

Finally, we present multi-year data from end-of-course surveys as the assessment tool that suggest our students see this course as highly relevant, applicable, and motivating.

Studies on GPS: GPS P/N Sequence Generator Mini-Project, Gold Code Properties, and Signal Acquisition Using Cross-Correlation

In our sophomore level course called Digital Circuits and Computer Systems (DCCS), students first learn about shift registers. In the junior/senior level Electronic Navigation course, students see a more in-depth, direct application for shift registers, tied to "Pseudo-random Noise" (PN) sequence generation. More specifically, students are tasked with writing a Matlab[TM] function that uses two 10 bit shift registers in order to calculate each GPS satellite's unique 1023 bit L1 "Coarse Acquisition" (C/A) Gold Code (Figure 1)[1]. Validation of each satellite's correct pseudo-random noise (PN) sequence (of 1's and 0's) is accomplished by comparing the first 10 values (chips) of that satellite's PN sequence with an octal number provided in the GPS signal specification[1,2]. One solution for a function that produces the PN sequences for satellites 1-32 can be found in Appendix A, under the heading "GPS P/N Sequence Generator Mini-Project Solution."

Once each student completes the "PN code generator" assignment, we have them unravel the mysteries of Code Division Multiple Access (CDMA) or Direct Sequence Spread Spectrum (DSSS), as used in C/A code GPS. After converting their sequence of 1's and 0's to 1's and -1's, students study the autocorrelation and cross-correlation properties of each satellite's baseband sequence (as generated from their Matlab function) using Matlab's "xcorr" function. Ignoring Doppler for now, students learn that all satellites transmit continuously on roughly the same L1 band frequency, so GPS receivers must use the cross correlation techniques in order to extract pseudo-range (range plus clock offset) measurements from each satellite. They learn that each satellite's 1023 bit Gold Code sequence exhibits remarkable autocorrelation properties (i.e. single peak at zero lag, and small values for other lags), and remarkable cross-correlation properties (i.e. cross-correlation of PN sequences from different satellites produce only small values at all lags) (Figures 2 and 3). Most importantly, they learn WHY those properties are, in fact, remarkable.

While generating autocorrelation and cross correlation functions, students learn that GPS receivers typically use "running correlators" for signal acquisition (and signal time-of-arrival estimation), hence apriori knowledge of each satellite's PN sequence is required. In order to "find" Satellite #12, the student simply correlates an incoming baseband signal with the known PN sequence for Satellite #12. The timing of the cross correlation peak establishes signal time of arrival relative to the local receiver clock.

After learning about the autocorrelation and cross-correlation of GPS Gold Codes, students then do some detective work! We provide our students a file containing baseband satellite information, buried in noise. More specifically, we add the PN sequences of three different satellites together, at different delays, and add Gaussian noise (with σ=4), so our students cannot even see that a signal is present (Figure 4). We then ask them to figure out which satellites are present, and the signal times of arrival.

Students first load this data file into Matlab, and begin to "search the sky" using their PN Generator function and Matlab's "xcorr" function, in a loop that searches for satellite #1 through satellite #32. We tell the students to declare that a satellite is present if any correlation value over 500 is detected, and given the data in Figure 4, students find 3 satellites are visible (#5, #19, #25) at varying delays. They complete their detective work by plotting cross correlation results for those satellites detected (e.g. Figure 5). Our students are typically amazed how powerful the cross correlation methods are, and usually make comments like " … but how can this work? … you can't even SEE any of the signals in the original noisy signal!" We call this the "aha moment," where our students see results that they have produced, and their eyes are opened to the concept of processing gain from the de-spreading (cross-correlation) operation. At this point, we ask students if they had ever learned about auto and cross correlation functions, or DS spread spectrum systems, and they usually offer comments like " … yeah, but not like this … we didn't really see much application, and didn't realize why we should care much about those concepts."
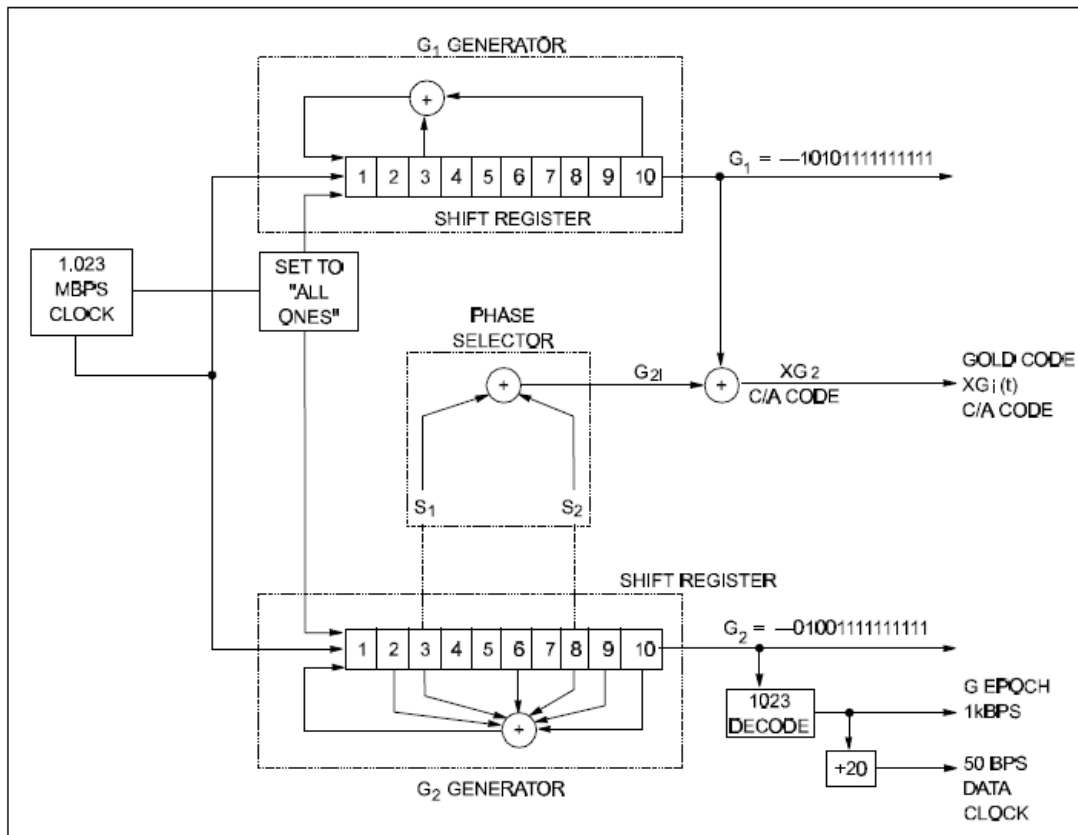

Figure 1. Shift register diagram for generating GPS C/A code PN sequence[1].
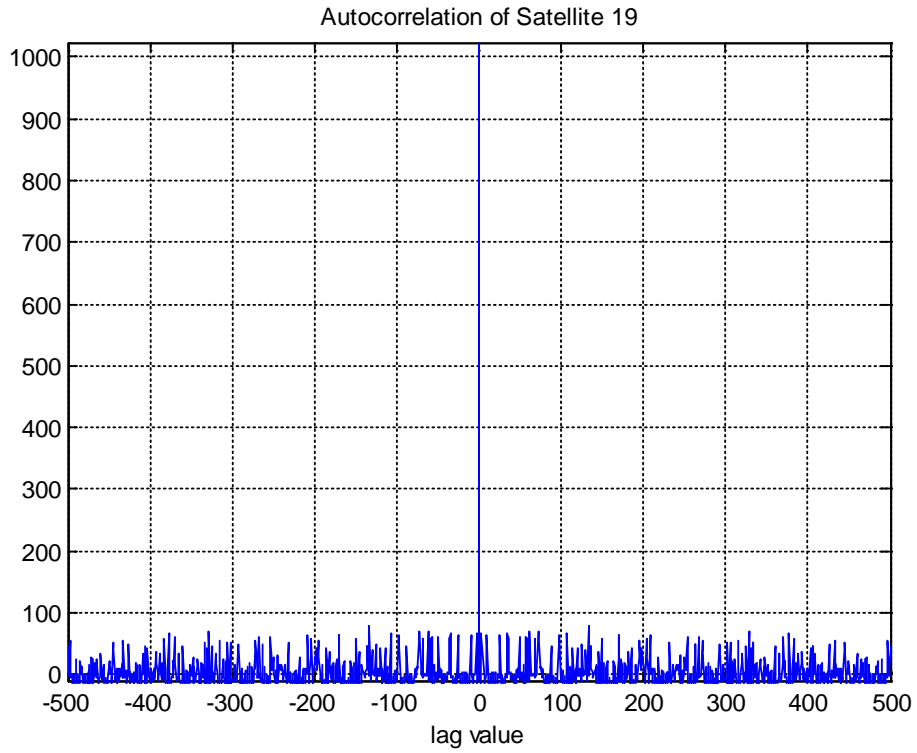
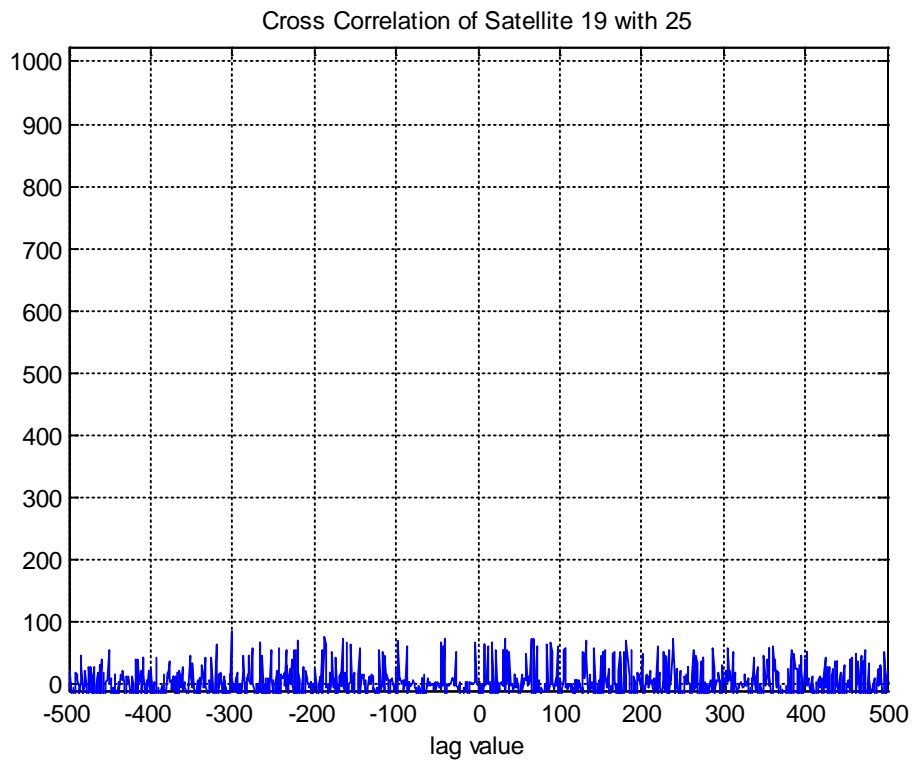Figure 2. Autocorrelation of PN sequence for Satellite #19.

Figure 3. Cross correlation of PN sequences for Satellite #19 and Satellite #25.
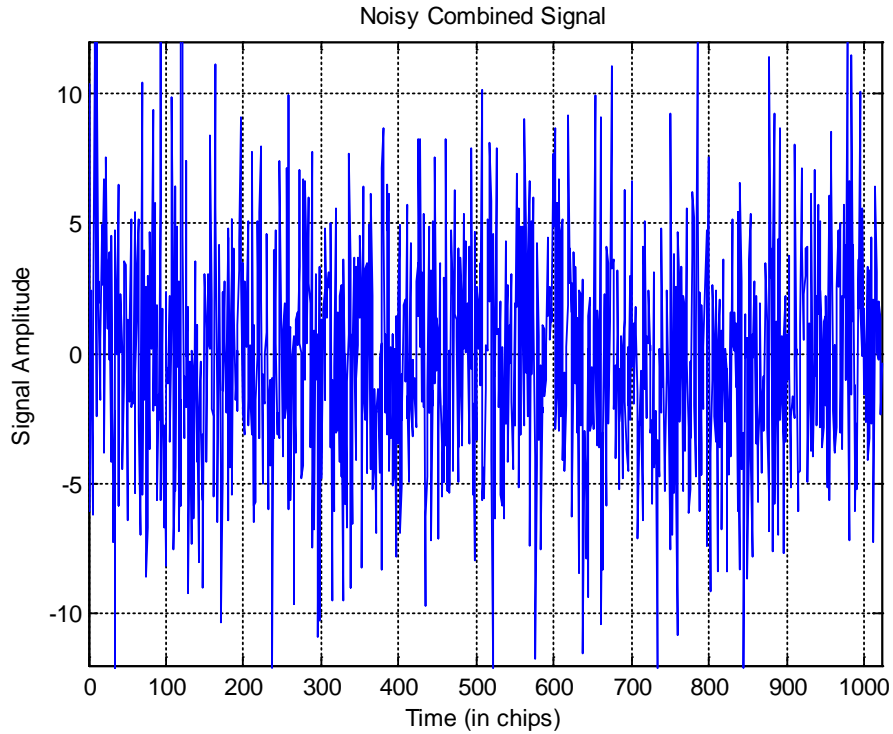
Figure 4. Noisy baseband signal, consisting of 3 satellite PN sequences, at different times of arrival, plus Gaussian noise (σ=4). Note that no deterministic signal is visible.
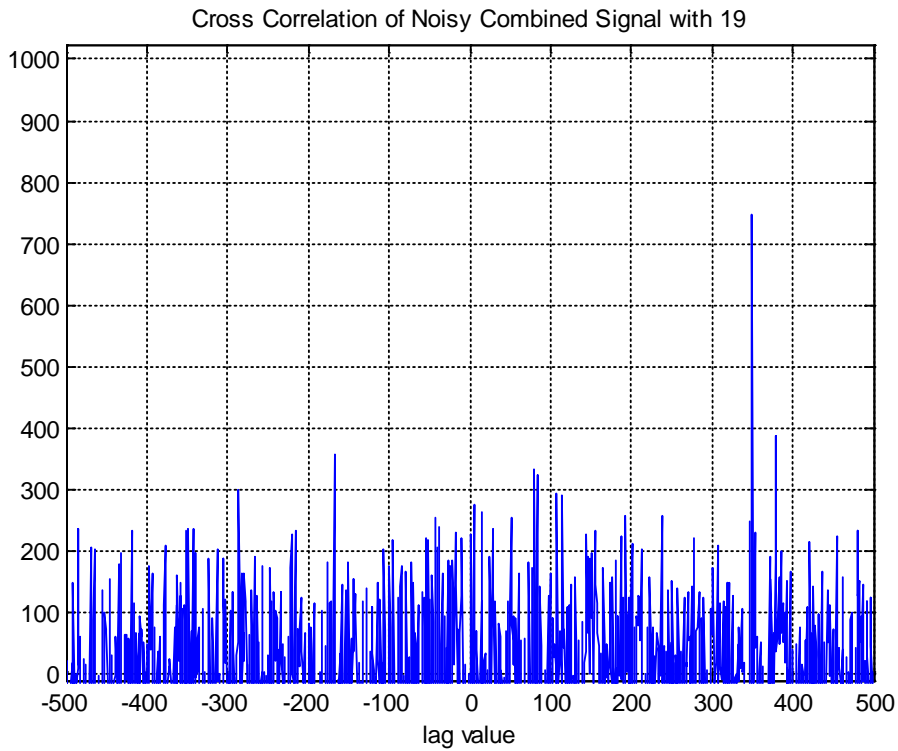

Figure 5. Cross correlation of noisy baseband signal from Figure 4, with PN sequence of Satellite 19, showing Satellite 19 is visible, with a correlation peak at lag of 350 chips.

Studies on GPS: Position Solution

Students in our major are required to take Probability Theory, and are highly encouraged to take Linear Algebra as a math-science elective. While these courses offer tremendous value for our EE's, there is little time to cover discipline-specific application. The calculation of Dilution of Precision (DOP) and position (x, y, z, and time) in GPS offer a wonderful opportunity to cover discipline-specific application of these fields in our Electronic Navigation course.

As part of a study on the mathematics of positioning using ranging-type systems, one is typically interested in the accuracy of a position estimate. There are two important factors that determine the accuracy of a position estimate: (1) the accuracy of the estimate of time-of-arrival (TOA) or estimate of range, and (2) the geometry of the satellites (or transmitters) relative to the user position. To motivate a linear algebraic investigation into GPS accuracy and position calculation, we pose these questions to our students: (1) "How exactly do GPS receivers solve for position, (2) How accurate is GPS?, and (3) How is accuracy related to satellite geometry?

Given N satellite positions $\{x_i, y_i, z_i\}$, i=1, N (in Earth-Centered, Earth Fixed, or ECEF coordinates), and N pseudo-range measurements $\{R_i\}$, i=1, N, we are interested in solving for the user position $\{x, y, z\}$ and receiver clock offset relative to GPS system time (b). Note that the pseudo-ranges $\{R_i\}$ represent the sum of true range from user position to each satellite, and a range offset term (Cb), which is the propagation speed (speed of light) multiplied by the clock offset.

$$R_i = \sqrt{\left(x - x_i\right)^2 + \left(y - y_i\right)^2 + \left(z - z_i\right)^2} + Cb \quad , i=1, \dots, N \qquad (1)$$

Solving for $\{x, y, z, \text{ and } b\}$ presents some challenge, since (1) represents a set of nonlinear equation in those variables. We then introduce classic linearization methods, and show our students how to solve for $\{x, y, z, \text{ and } b\}$ using familiar least-squares techniques. The iterative solution for $\{x, y, z, \text{ and } b\}$ usually involves assuming a position (called AP), and an arbitrary range offset b (usually 0). We then linearize these equations about point "AP", and form the over-determined system of equations in (2), where $\{X_k, Y_k, Z_k\}$ is the position of Satellite #k, $R_{k,AP}$ is the range from "AP" to Satellite #k, and $R_{k_{obs}}$ is the observed pseudo-range to Satellite #k.

$$
\begin{bmatrix} R_{1_{obs}} - R_{1,AP} \\ R_{2_{obs}} - R_{2,AP} \\ R_{3_{obs}} - R_{3,AP} \\ \vdots \\ R_{N_{obs}} - R_{N,AP} \end{bmatrix} = \begin{bmatrix} \dfrac{X_1 - X_{AP}}{R_{1,AP}} & \dfrac{Y_1 - Y_{AP}}{R_{1,AP}} & \dfrac{Z_1 - Z_{AP}}{R_{1,AP}} & 1 \\[2ex] \dfrac{X_2 - X_{AP}}{R_{2,AP}} & \dfrac{Y_2 - Y_{AP}}{R_{2,AP}} & \dfrac{Z_2 - Z_{AP}}{R_{2,AP}} & 1 \\[2ex] \dfrac{X_3 - X_{AP}}{R_{3,AP}} & \dfrac{Y_3 - Y_{AP}}{R_{3,AP}} & \dfrac{Z_3 - Z_{AP}}{R_{3,AP}} & 1 \\[2ex] \vdots & \vdots & \vdots & \vdots \\[1ex] \dfrac{X_N - X_{AP}}{R_{N,AP}} & \dfrac{Y_N - Y_{AP}}{R_{N,AP}} & \dfrac{Z_N - Z_{AP}}{R_{N,AP}} & 1 \end{bmatrix} \begin{bmatrix} \Delta X \\ \Delta Y \\ \Delta Z \\ C(\Delta b) \end{bmatrix} + \begin{bmatrix} e_1 \\ e_2 \\ e_3 \\ \vdots \\ e_N \end{bmatrix} \qquad (2)
$$

Equation (2) can be written in more compressed notation as

$$
\begin{bmatrix} \Delta R_1 \\ \Delta R_2 \\ \Delta R_3 \\ \vdots \\ \Delta R_N \end{bmatrix} = \underset{\sim}{A} \begin{bmatrix} \Delta X \\ \Delta Y \\ \Delta Z \\ C(\Delta b) \end{bmatrix} + \begin{bmatrix} e_1 \\ e_2 \\ e_3 \\ \vdots \\ e_N \end{bmatrix}. \qquad (3)
$$

At this point we have our students examine matrix $\underset{\sim}{A}$ more closely, and we have them note that this is called a "direction cosine" matrix, and can be written as

$$
\underset{\sim}{A} = \begin{bmatrix} \cos(AZ_{SV_1})\cos(EL_{SV_1}) & \sin(Az_{SV_1})\cos(EL_{SV_1}) & \sin(EL_{SV_1}) & 1 \\ \cos(AZ_{SV_2})\cos(EL_{SV_2}) & \sin(AZ_{SV_2})\cos(EL_{SV_2}) & \sin(EL_{SV_2}) & 1 \\ \cos(AZ_{SV_3})\cos(EL_{SV_3}) & \sin(AZ_{SV_3})\cos(EL_{SV_3}) & \sin(EL_{SV_3}) & 1 \\ \vdots & \vdots & \vdots & \vdots \\ \cos(AZ_{SV_N})\cos(EL_{SV_N}) & \sin(AZ_{SV_N})\cos(EL_{SV_N}) & \sin(EL_{SV_N}) & 1 \end{bmatrix}, \qquad (4)
$$

where AZ_{SV_k} and EL_{SV_k} represent azimuth and elevation angles from the assumed position (AP) to Space Vehicle (or Satellite) #k.

Recognizing that equation (3) represents an over-determined set of equations, and assuming a full-rank "A" matrix (rank 4), they can solve for $\begin{bmatrix} \Delta X \\ \Delta Y \\ \Delta Z \\ C(\Delta b) \end{bmatrix}$ as

$$
\begin{bmatrix} \Delta X \\ \Delta Y \\ \Delta Z \\ C(\Delta b) \end{bmatrix} = \left(\underset{\sim}{A}^T \underset{\sim}{A} \right)^{-1} \underset{\sim}{A}^T \begin{bmatrix} \Delta R_1 \\ \Delta R_2 \\ \Delta R_3 \\ \vdots \\ \Delta R_N \end{bmatrix} \tag{5}
$$

In the next iteration they subtract $\begin{bmatrix} \Delta X \\ \Delta Y \\ \Delta Z \\ C(\Delta b) \end{bmatrix}$ from the original assumed position and clock

offset to form a revised "AP," and they recalculate $\begin{bmatrix} \Delta R_1 \\ \Delta R_2 \\ \Delta R_3 \\ \vdots \\ \Delta R_N \end{bmatrix}$ and $\underset{\sim}{A}$ from equation (3), and

solve equation (5) for a new position correction. Usually 4 to 5 iterations are more than sufficient for convergence.

In order to convert latitude/longitude/altitude to "earth-centered, earth-fixed" (ECEF) coordinates, they make use of Matlab^{TM} functions ecef2lla, and lla2ecef.

Example#1

A typical problem we assign our students is the following: Assume we have downloaded an almanac and ephemeris, and we know satellite positions (in Earth-Centered, Earth-Fixed, or ECEF coordinates, in meters) at time of signal transmission are as listed below. Also, assume the pseudo-range measurements (in meters) for each SV are listed below. You can assume your GPS receiver was last used just south of Detroit, MI (at N42 degrees, W83 degrees), hence your initial "assumed position" for your position solution can begin there.

	(x coordinate)	(y coordinate)	(z coordinate)	(pseudorange)
SV2	7433442.899	-22926895.92	11096844.746	22784055.7
SV4	-4930906.599	-25650220.33	4268436.452	23772239.8
SV7	348603.197	-15668323.327	21765664.376	23313223.8
SV8	-11440481.794	-23040016.991	6063975.734	24292916.5
SV13	20929924.126	-7467688.968	14549441.864	25312443.4

Solve for my location, and my receiver time offset from GPS time. What city is this?

Answer: The student should find that the position N36.0654, W80.2804, altitude 0 meters, and clock offset of 8mS relative to GPS system time, provides the least mean squared estimate. The city is Winston-Salem, North Carolina. A solution to this problem can be found in Appendix A, under the heading "Solution to GPS Positioning Mini-Project."

Example#2: The designers of the Global Positioning System were smart enough to set the satellite orbits such that all satellites would never appear at the same elevation angle. Why might this be an important design consideration?

Answer: The matrix $\underset{\sim}{A}$ from equations (2)-(4) collapses to rank 3, hence one could not obtain a unique solution for 3-d position and time. Said another way, if all satellites are at the same elevation angle, and the user receiver traverses in the vertical direction, the pseudo-ranges to all satellites change the same amount, and you cannot tell if the receiver clock offset has shifted, or if the vertical position has shifted.

Studies on GPS: Dilution of Precision (DOP) and Accuracy

If we make a simplifying assumption that pseudo-range measurements from our satellite constellation are uncorrelated, and the pseudo-range measurements are equally noisy (σ_{TOA}'s are the same), then equation (5) helps us relate uncertainties in position and clock offset estimates to uncertainties in pseudo-range measurements. Said another way, we show our students that the covariance matrix for position and clock offset is related to the covariance of the pseudo-range measurements by the matrix

$$\underset{\sim}{G} = \left(\underset{\sim}{A}^T \underset{\sim}{A} \right)^{-1}, \tag{6}$$

and the square root of the trace of matrix G is called GDOP, or "Geometric Dilution of Precision." GDOP is a dimensionless multiplier that relates errors in the pseudo-range domain to error in the position domain (including clock offset).

$$GDOP = \sqrt{trace\left(\underset{\sim}{G} \right)} = \sqrt{\underset{\sim}{G}_{1,1} + \underset{\sim}{G}_{2,2} + \underset{\sim}{G}_{3,3} + \underset{\sim}{G}_{4,4}} \tag{7}$$

Other common "Dilution of Precision" (DOP) terms are defined as:

$$HDOP = \sqrt{\underset{\sim}{G}_{1,1} + \underset{\sim}{G}_{2,2}} \text{ (horizontal dilution of precision)}$$

$$PDOP = \sqrt{\underset{\sim}{G}_{1,1} + \underset{\sim}{G}_{2,2} + \underset{\sim}{G}_{3,3}} \text{ (position dilution of precision)}$$

$$TDOP = \sqrt{\underset{\sim}{G}_{4,4}} \text{ (time dilution of precision)}$$

$$VDOP = \sqrt{\underset{\sim}{G}_{3,3}} \text{ (vertical dilution of precision)}$$

This means that if we have some knowledge of the standard deviation of our pseudo-range measurements, and we know the azimuth and elevation angles of all satellites in view, we can translate that information into a calculation of accuracy! For example, an estimate of horizontal accuracy can be formulated as

$$R_{2DRMS} = 2\sqrt{\sigma_x^2 + \sigma_y^2} \cong 2 \times C \times \sigma_{TOA} \times HDOP , \qquad (8)$$

where R_{2DRMS} is the radius of a circle, inside which you will be between 95% and 98% of the time. Assuming good satellite visibility (and good geometry) throughout the day, typical values of HDOP are between 1 and 2, and assuming $\sigma_{TOA} = 10nS$ (a reasonable estimate), 2DRMS error circles of between 6 and 12 meters are produced.

Using m-files our students have produced for their GPS position solutions, we ask them to think about using a GPS receiver to try to find a restaurant in a large city such as New York, NY, where the only satellites visible would be perhaps only the ones directly overhead. At that point they begin calculating HDOP for satellites whose elevation angles are above 85 degrees. They quickly realize that even small errors (due to noise, or perhaps multipath reflections off buildings), translate to relatively large horizontal distances (e.g. 500-700 meters), so they might actually be better off asking for directions rather than depending on their GPS receiver inside their cellular telephone for guidance!

Studies on Instrument Landing System (ILS) and VHF Omni-Directional Radio Range: Electronic Navigation in the Cockpit

Other systems discussed in our Electronic Navigation Systems class include aircraft navigation systems such as VOR (VHF Omni-Directional Radio Range) (AM and FM communications application), and Instrument Landing System (ILS) (antenna, phased array application).

With respect to phased array antenna systems, our students typically have learned a little bit about such systems, but often have not heard about specific applications where highly directive antennas are required, such as those needed for aircraft precision approach systems. In the case of an aircraft ILS, this system consists of two separate antennas: one for the "localizer" (Figure 6), (providing left-right guidance on an approach), and one for the "glide slope" (providing up-down guidance on an approach). The general idea is that an aircraft flies down a very narrow "electronic corridor," where azimuth and elevation guidance is provided to the pilot. Students are asked to plot a localizer antenna gain pattern assuming (10 elements with λ/2 element separation), and typical results are shown in Figure 7. Once they have their code working, they typically play with the number of elements, and see how they can control antenna directivity. We also ask students to think about how large metal structures on the ground (e.g. aircraft taxiing nearby) might affect the transmitted signal, and we explain that in restricted visibility,

aircraft must hold short of a "ILS Hold Short" line, so as to avoid shifting the transmitted signal in space (since the aircraft becomes a parasitic re-radiator at VHF).

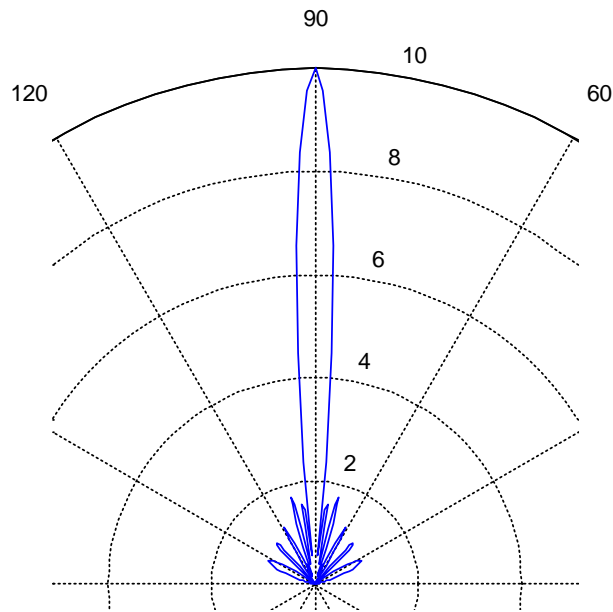Figure 6. Typical localizer antenna installed at runway end.

Figure 7. Localizer antenna pattern (assuming 10 elements with λ/2 element separation).

Another system used for aircraft navigation is the VOR system. A VOR transmitted signal originates from a ground-based transmitter, and it provides bearing information to the pilot. Essentially the signal consists of two major parts: (1) An omni-directional FM signal with 30Hz sinusoidal modulation, and (2) a highly directional signal that rotates 30 times/sec (thereby generating an AM signal at the receiver). The phase difference

between the demodulated omni-directional 30 Hz tone and the directional (AM) 30Hz tone is the bearing to or from the VOR transmitter. The general equation for a received VOR signal is given as

$$v_r(t) = \cos(w_c t)[1 + a\cos(w_m t - \alpha) + b\cos(w_u t + \beta\cos w_m t) + f(t)\cos w_i t], \qquad (9)$$

where w_c is the carrier frequency, w_m represents the AM modulating frequency, α represents the bearing to or from the station, w_u represents a frequency offset (subcarrier) for the FM, β represents the FM modulation index, and $f(t)\cos w_i t$ represents a Morse Identifier (another subcarrier). Students are generally amazed how simple the signal structure is, and how simple the receiver is for estimating phase difference.

Course Feedback

We have taught this course for a number of years now, and without exception, our students genuinely feel that they gain a lot of insight into the real-world application of their linear algebra, antennas, communications, digital design, and signals/systems courses. While space does not permit us to cover all student comments, the theme of the comments can be appreciated through these examples:

> *"This is one of those classes I will look back into the binder in years to come to solve problems or answer questions of others. I thought this had to be one of the best courses to take … I feel that I have a much greater understanding of the theory …"*

> *" … it pulled together a lot of ideas introduced in previous EE courses, and applied those ideas."*

> *"This will be incredibly applicable in the fleet, and I will really be able to take this information and awareness with me. I feel like I have a much better understanding of something I couldn't even scratch the surface of before, since it was so out of my range.*

> *"I was excited and fired up about the contemporary events that the instructor presented, and really have developed a big interest in this subject as a result of this class … best class ever!"*

> *"The material was complex, and it called on a lot of knowledge from previous courses. I am an EE that that had not taken Communications Systems, so I did not enter with a background in a lot of that information. However, by paying attention in class and asking questions, I found that I could achieve a very high understanding of the material, because in the end, it really was all applied math, and I understood all of it."*

The only negative student feedback received over the last 8 years has been that some students wanted a textbook that would support the electronic navigation systems course. Although course materials (presentations, notes, articles) were posted on-line (www.desire2learn.com), some students wanted a reference text to cover both theory and operation of electronic navigation systems, including topics on the mathematics of positioning, GPS (and augmentations such as DGPS and WAAS), eLoran, Inertial Navigation Systems, plus aircraft navigation systems such as VHF-Omnidirectional Range (VOR), Distance Measuring Equipment (DME), and Instrument Landing Systems (ILS), plus introductory Kalman Filtering. Although we know of no current undergraduate text book that covers this material in sufficient depth, we did allow students to borrow a copy of the text that covers GPS and its augmentations[3]. This interim solution worked well for that segment of the course, but was only feasible because we have a relatively small EE program and typical semester enrollment of fewer than 20 students in the course. We continue in our search for a solid text for this course.

Ongoing Challenges

While the class has been offered for almost a decade, and while we see this course as highly successful, we do have ongoing challenges. More specifically:

1) While this (3 credit/3 contact hour) course is an elective typically taken by EE majors in the spring of their junior or senior years, we have allowed exceptional students from other engineering majors (or from our Operations Research major) to take the course. In general this has been a great success, however it can be challenging when some students have limited background in linear systems theory, communication systems, linear algebra, or even Matlab programming. We have solved these challenges by offering the course with very small section sizes (sometimes as small as 8 students/section), and by "flipping the classroom" on occasion, so students are doing reading outside the classroom, and completing mini-project assignments in the classroom setting when needed. By completing some of this work in the classroom setting, other students and/or the professor were then able to assist those who lacked background in certain areas (e.g. Matlab syntax), with very positive outcomes.

2) While several schools offer more focused courses in specific selected navigation systems (e.g. "GPS," - AA272C from Stanford University), and several schools offer "Electronic Navigation" courses focused on one application (e.g. Ohio University's EE4853, avionics focus), our focus here is a bit broader in that our students learn the common principles of time-of-arrival (TOA) or phase estimation, and the fundamentals of mathematics of positioning, while discussing the unique challenges of a wide variety of applications (e.g. indoor navigation, underwater navigation, marine navigation, aircraft navigation). We continue to adjust the breadth/depth of the course each year, usually by changing the mini-projects assigned.

Students very much enjoy the "hands-on" mini-projects (typically Matlab-based), however adjusting these mini-projects does require significant effort, and we do expend significant effort to explain each project in detail, in order to keep all mini-projects and homework at the undergraduate level.

Concluding Remarks

We are continually encouraged that students (and former students!) e-mail us relevant articles they have read, or ask us questions about electronic navigation systems they come across. We continue to be amazed at the student interest in this course, and are surprised that we have had several students from other majors (engineering or mathematics) come to take this EE course. Course enrollments have ranged from 11 to 23 students each year for the past 8 years, including several students from outside the EE major. In total, more than 75% of our students in EE have taken this course over the last 8 years. We think the biggest reasons for the success of this offering lie in the practical nature of the course, as students SEE the reason they might care about a rank deficient matrix, and SEE how straightforward it is for a GPS receiver to solve for position. We believe that teaching the theory through these practical systems allows students the opportunity to gain additional insight, and we believe this contributes directly to their long-term learning and general satisfaction with their major and program of study.

We are continuing in our development of innovative hands-on problems, and are actively pursuing use of additional hardware (e.g. Ettus Research N210 Universal Software Radio Peripheral) for future class projects and demonstrations.

Appendix A.

GPS P/N Sequence Generator Mini-Project Solution

```matlab
function OutputSequence=PRN_Generator(Which_Satellite)
%
%Function name:  PRN_Generator
%Sample function call:  Y=PRN_Generator(3)
%generates the 1023 bit sequence for satellite PRN#3
%
%The most useful reference for this is the Global Positioning System
(GPS)
%Standard Positioning Service Signal Specification - Specifically, the
%table on page 15 shows what taps on the G2 register are used to
generate
%which satellite codes.
%

Initial=ones(1,10);  %all ones in a 10 bit shift register
G1_Register=Initial; %all ones in a 10 bit shift register
G2_Register=Initial;

G1_Taps=[3 10]; %corresponding to shift register taps 3 10;

Fixed_G2_Taps=[2 3 6 8 9 10];  %corresponding to taps  2,3,6,8,9,10

switch Which_Satellite
    case 1,
        G2_Taps=[2 6];
    case 2,
        G2_Taps=[3 7];
    case 3,
        G2_Taps=[4 8];
    case 4,
        G2_Taps=[5 9];
    case 5,
        G2_Taps=[1 9];
    case 6,
        G2_Taps=[2 10];
    case 7,
        G2_Taps=[1 8];
    case 8,
        G2_Taps=[2 9];
    case 9,
        G2_Taps=[3 10];
    case 10,
        G2_Taps=[2 3];
    case 11,
        G2_Taps=[3 4];
    case 12,
        G2_Taps=[5 6];
    case 13,
        G2_Taps=[6 7];
    case 14,
        G2_Taps=[7 8];
```

```matlab
    case 15,
        G2_Taps=[8 9];
    case 16,
        G2_Taps=[9 10];
    case 17,
        G2_Taps=[1 4];
    case 18,
        G2_Taps=[2 5];
    case 19,
        G2_Taps=[3 6];
    case 20,
        G2_Taps=[4 7];
    case 21,
        G2_Taps=[5 8];
    case 22,
        G2_Taps=[6 9];
    case 23,
        G2_Taps=[1 3];
    case 24,
        G2_Taps=[4 6];
    case 25,
        G2_Taps=[5 7];
    case 26,
        G2_Taps=[6 8];
    case 27,
        G2_Taps=[7 9];
    case 28,
        G2_Taps=[8 10];
    case 29,
        G2_Taps=[1 6];
    case 30,
        G2_Taps=[2 7];
    case 31,
        G2_Taps=[3 8];
    case 32,
        G2_Taps=[4 9];
    otherwise
        disp('You do not have a valid satellite number -- must be 1-
32');
end;

OutputSequence=[];
for Shift=1:1023,

    G2i_Bit=xor(G2_Register(G2_Taps(1)),G2_Register(G2_Taps(2)));
    G1_Bit=G1_Register(10);
    C_A_Bit=xor(G2i_Bit,G1_Bit);

    OutputSequence=[OutputSequence C_A_Bit];

    New_Bit_G1=xor(G1_Register(G1_Taps(1)),G1_Register(G1_Taps(2)));

    % note that T1, T2, T3, and T4
    %are just temporary variables that allow
    % us to do xor on 6 bits
    %New_Bit_G2 is xor of bits 2,3,6,8,9, and 10 in Register
```

```
T1=xor(G2_Register(Fixed_G2_Taps(1)),G2_Register(Fixed_G2_Taps(2)));
    T2=xor(T1,G2_Register(Fixed_G2_Taps(3)));
    T3=xor(T2,G2_Register(Fixed_G2_Taps(4)));
    T4=xor(T3,G2_Register(Fixed_G2_Taps(5)));
    New_Bit_G2=xor(T4,G2_Register(Fixed_G2_Taps(6)));

    G1_Register=[New_Bit_G1 G1_Register(1:9)];
    G2_Register=[New_Bit_G2 G2_Register(1:9)];

end;
    OutputSequence=OutputSequence(1:1023);

    %at this point we have a sequence of 1's and 0's ... and we want to
    %convert to 1's and -1's ...

    OutputSequence=2*OutputSequence-1;


return;
```

Solution to GPS Positioning Mini-Project

```
% First you must load satellite locations in matrix Sat_Locations
% where Sat_Locations(1,:) is a row vector that represents the
% x, y, and z coordinate of the first satellite
%
% Similarly, you must load measured pseudoranges as a
% row vector called Measured_Pseudoranges.  These are measurements
% and do not change through the iterations.

Sat_Locations=
[7.433442898999999e+06 -2.292689592000000e+07 1.109684474600000e+07;
-4.930906599000000e+06 -2.565022033000000e+07 4.268436452000000e+06;
 3.486031970000000e+05 -1.566832332700000e+07 2.176566437600000e+07;
-1.144048179400000e+07 -2.304001699100000e+07 6.063975733999999e+06;
 2.092992412600000e+07 -7.467688968000000e+06 1.454944186400000e+07];

Measured_Pseudoranges = [2.278405574299973e+07;
                         2.377223982668087e+07;
                         2.331322380629639e+07;
                         2.429291649273822e+07;
                         2.531244341334572e+07].';


AP=lla2ecef([42 -83 0] , 'WGS84' );  %just an initial position

for iteration=1:6,

    for m=1:length(Measured_Pseudoranges),
        %%calculate range to satellites from the Assumed Position
        AP_Range(m)=MyDistance(AP,Sat_Locations(m,:));
    end;

     A=[(Sat_Locations(1,:)-AP)/AP_Range(1) 1;
```

```
        (Sat_Locations(2,:)-AP)/AP_Range(2) 1;
        (Sat_Locations(3,:)-AP)/AP_Range(3) 1;
        (Sat_Locations(4,:)-AP)/AP_Range(4) 1;
        (Sat_Locations(5,:)-AP)/AP_Range(5) 1];

    Del_PR=[Measured_Pseudoranges-AP_Range].';
    DelX=A\Del_PR;
    AP=AP-DelX(1:3).';

end;


Final_Position=ecef2lla(AP,'WGS84')
Final_Time_Offset_from_GPS_Time=DelX(4)/3e8



%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function dist=MyDistance(PosA,PosB);
%Calculates distance between PosA and PosB in 3-space

dist=sqrt((PosA(1)-PosB(1))^2 + (PosA(2)-PosB(2))^2 + …
          (PosA(3)-PosB(3))^2);
return;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

REFERENCES

[1]"Global Positioning System Standard Positioning Service Signal Specification," http://www.gps.gov/technical/ps/1995-SPS-signal-specification.pdf, 2nd edition, 2 June,1995.

[2]M. Dunn, "GPS Interface Specification IS-GPS-200," http://www.gps.gov/technical/icwg/IS-GPS-200H.pdf, 24 September 2013.

[3]P. Misra, P. Enge, (2011). *Global Positioning System: Signals, Measurements, and Performance (Revised Second Edition*. Lincoln, MA., Ganga-Jamuna Press.