Global Positioning System

A Thesis Presented to

The Faculty of the Mathematics Program

California State University Channel Islands

In (Partial) Fulfillment

of the Requirements for the Degree

Masters of Science

by

Jacquelyne Ta

April, 2011

© 2011

Jacquelyne Ta

ALL RIGHTS RESERVED

Jacquelyne Ta
MS Math
SP 11

APPROVED FOR THE MATHEMATICS PROGRAM

Cynthia J. Wyels 5/11/11

Dr. Cynthia J. Wyels, Committee Chairperson Date

Geoffrey Buhl 5/11/11

Dr. Geoffrey Buhl, Committee Member Date

APPROVED FOR THE UNIVERSITY

Dr. Gary Berg, Dean of Extended University Date

To my parents, Linda Tran and Peter Ta, to my two sisters (Jeannette and Diana) and four brothers (Joseph, Duke, David, and Paul), to my in-laws Emily Cosby and Dave Dunlap, and especially to my loving husband Eli Dunlap, in gratitude for their encouragement and support.

Acknowledgements

I have always been fascinated in science and technology. I was fortunate to be a part of the NASA CSUN/JPL Pair Program during my undergraduate at CSU Northridge. The program exposed me to a variety of research projects such as analyzing solar physics data, determining satellite orbits, and modeling the Northridge earthquake. One subject that stuck with me was the Global Positioning System. There are so many fascinating mathematical concepts and physical components that made this technology function the way it is today. I knew this would be a subject I can keep writing about.

My time at CSU Channel Islands has been filled with caring and knowledgeable staff. The graduate courses in applied math are always intriguing such as neural networks and pattern recognition. Dr. Cynthia Wyels, my advisor, has been very supportive of my thesis topic. Dr. Wyels has walked me through the mathematical concepts I did not understand. I have much gratitude and respect for her professional guidance and expertise. Also, I would like to thank Dr. Buhl for his time reading my thesis and all his feedbacks.

Those closest to me often get the least recognition, but they deserve the most. I would like to thank my parents because they have instilled the value of education from an early age, and I would not have made it this far without them pushing me along. It is a joy to grow up in a family of seven and I thank my sisters and brothers for influencing me diversely. Emily and Dave, my in-laws, constantly reminded me to finish this dissertation every time we talked over the phone. They have given me the extra boost and confidence I needed to finish this dissertation. Most of all, I would like to thank my husband, Eli, for his love, patience, and support. He has provided much needed balance in my life, and I am far happier and productive as a result.

Abstract

Global Positioning System

by Jacquelyne Ta

The Global Positioning System (GPS) is a satellite-based navigation system made up of a network of 24 satellites placed into orbit by the United States Department of Defense (DOD) and managed by the United States Air Force 50th Space Wing. GPS was originally intended for military applications. It was designed to assist soldiers and military vehicles, planes, and ships in accurately determining their locations world-wide. However, in the 1980s, the government made the system available for civilian use.

Civilian applications are evolving and expanding constantly. Today, the uses of GPS have extended to include both the commercial and scientific worlds. Commercially, GPS is used as a navigation and positioning tool in airplanes, boats, cars, and most outdoor recreational activities such as hiking, fishing, and kayaking. In the scientific community, GPS plays an important role in the earth sciences. Meteorologists use GPS for weather

forecasting and global climate studies. Geologists use GPS for surveying and earthquake studies to measure tectonic motions during and between earthquakes.

The GPS is vast, expensive and involves a lot of technical ingenuity, but the fundamental concepts at work are quite simple and intuitive. The objective of this thesis paper is to give the readers a working familiarity with both the basic theoretical and practical aspects of how the GPS works. Chapter 1 introduces a condensed GPS program history that involves three competing concepts from the Transit, Timation, and Project 621B programs. Chapter 2 examines the GPS system consisting of three segments: space segment, control segment, and user segment. The three segments contribute to overall accuracy, reliability, and functionality. Chapter 3 outlines the basic mathematical methods used to calculate user's position based on a system with no errors. Finally, Chapter 4 and 5 covers two important math theories needed to handle satellite and receiver clock errors: Newton-Raphson Method and the Least Squares Method.

Contents

1 Global Navigation **1**

1.1 Transit Satellite System 2

1.2 Naval Research Laboratory (NRL) Timation 5

1.3 Air Force Project 621B 6

1.4 Joint Program Office (JPO) 8

2 System Segmentation **11**

2.1 Space Segment 12

2.2 Operational Control Segment 16

2.3 User Segment 19

3 Determining User's Position **23**

3.1 GPS Single Point Positioning 24

3.2 Examples: Linear System of GPS with no Errors 30

4 Pseudorange **38**

4.1 Satellite and Receiver Clock Offset 39

4.2 Introduction to Newton-Raphson Method 42

4.3 Multivariate Newton-Raphson Method 46

4.4 Conclusion 48

4.5 Examples 49

4.6 Exercises 54

5 Least Squares **56**

5.1 Determining User's Position with Least Squares 58

5.2 Exercises 60

Appendices **62**

1 Global Navigation

"Nature is an infinite sphere of which the center is everywhere and the circumference nowhere."

Blaise Pascal

Where am I? The question seems simple; the answer, historically, has proven not to be. For centuries, navigators and explorers have searched the heavens for a system that would enable them to locate their position on the globe. The ancient Polynesians used a fundamental technique known as the angular measurements of the natural stars to locate their whereabouts. The natural stars are the constellations we know today. As technology evolved over many years through science, engineering and mathematical research, the constellation was replaced with a highly sophisticated satellite system known as the Navigation System for Timing and Ranging (NAVSTAR), but commonly referred to as the Global Positioning System (GPS).

The GPS is a satellite-based navigation system that was developed by the U.S. Department of Defense (DOD) in the early 1970s under its NAVSTAR satellite program. During the first half of the 20th century, researchers at various organizations were separately developing the technology that would eventually be used to create GPS. These organizations include the Transit, Timation, and 621B program. Initially, GPS was developed as a military system to fulfill U.S. military needs. However, it was later made available to civilians, and is now a dual-use system that can be accessed by both military and civilian users.

The GPS are being used in aircrafts, ships, and cars. There are many ways to benefit from GPS. For example, the GPS has helped saving lives and property across the nation. One incredible story was a rescue of a twenty-seven year old man who was bitten by a snake while camping in the woods. Once the snake struck, he knew the proper course of action was to remain as still as possible, but he was alone in the middle of nowhere with poisonous venom coursing through his veins. With a GPS system in his camping gear, he was able to contact emergency services, giving them his exact location. With a GPS receiver for as low as $100, anyone, anywhere in the world, can instantaneously determines his or her location and never be lost again.

1.1 Transit Satellite System

Radio waves are invisible and completely undetectable to humans. All of today's modern technology uses radio waves to communicate whether it is a cell phone, a baby monitor, or a GPS receiver. Radio was the first modern technology applied to position finding. Much of the early pioneering in radio waves was done by Reginald Fessenden, a Canadian inventor interested in increasing his skills in the electrical field, who moved to New York hoping to gain employment with the famous inventor, Thomas Edison. As early as 1912, Reginald Fessenden began conducting experiments on the coast of Massachusetts and devised a simple system of using radio waves to help ships determine their positions. The system was extremely inaccurate and geographically limited. Despite his failures, his idea of devising a simple system to locate a ship's position inspired other scientists to pursue further research.

On October 4th, 1957, Sputnik was launched into space by Russia and it became known that "artificial stars" could be used for navigation. Sputnik was a little more than an orbiting radio transmitter. The fact that a satellite's position could be tracked from the ground captured the attention of scientists across the globe. It was the first step in recognizing that a subject's whereabouts on the ground could be determined using radio signals from a satellite.

Two scientists at the Johns Hopkins University Applied Physics Laboratory (APL), George Wieffenbach and William Guier, realized that they could determine Sputnik's orbit from the Doppler frequency shift as it passed overhead. Their measurements were subsequently confirmed by findings from other tracking sites. From this observation, Frank T. McClure at APL reasoned that, conversely, if the orbit of a satellite were known, then the Doppler-shift measurements could also be used to determine any ground position on Earth.

Pursuing this concept further, the US Navy initiated studies for its first satellite navigation system: the Transit. The Transit satellite system was developed by the APL of John Hopkins University for the Navy, in 1958. The goal of this program was to provide accurate positioning for U.S submarines using the Doppler shift measurement technique. A prototype satellite, Transit 1A, was launched in September 1959 but failed to reach orbit. A second satellite, Transit 1B, was successfully launched in April 1960. A constellation of five satellites were placed in orbit to provide reasonable global coverage.

The satellite system was placed in low polar orbits, at an altitude of about 600 nautical miles or 373 miles with an orbital period of about 106 minutes. The distance of 373 miles is about the distance from Camarillo, CA to San Francisco, CA. The system also has ten spare satellites (one spare for each) kept in orbit just in case one or more fail. The Transit satellite systems made a remarkable starting point to global navigation. A submarine was able to locate its position within 6 to 10 minutes and was accurate to within 25 meters or 82 feet.

An error of 25 meters is clearly intolerable. The GPS receiver could be anywhere on a circle with a radius of 25 meters. The Transit's intolerable accuracy raised other concerns regarding its ability to provide high-speed, real-time position measurements. For example, the Transit system would not be useful for high-speed aircrafts. Nonetheless, the program demonstrated the feasibility of a satellite-based navigational guidance system. The Transit program development was important and a success.

1.2 Naval Research Laboratory (NRL) Timation

The Transit satellite systems made a remarkable starting point to global navigation. Due to its lack of ability to provide high-speed positioning for the U.S aircrafts, the Naval Research Laboratory's (NRL's) Naval Center for Space Technology (NCST) was in search for a better navigation system. Roger L. Easton, an American scientist, conceived the idea of using synchronized clocks in the satellites while conducting a ranging and velocity experiment. One of Roger Easton's key patents, filed in 1963, provided the basis for GPS passive ranging and the simultaneous synchronization of the navigator's clock with the satellite clock.

In 1964, NRL conceived the Timation (TIMe/navigATION) program. The goal for Timation is to examine the means of speeding up and simplifying the positioning capability. The NRL believed that the passive ranging technique required high-precision clocks to provide accurate position and precise time measurements to ground observers. The Timation program was also designed to re-examine the most effective satellite constellation for providing worldwide coverage and to improve the prediction of satellite's orbit.

The first launch of the Timation I satellite was in 1967; the Timation II satellite followed in 1969. Each satellite bore a quartz clock. Timation proved that a system using a passive ranging technique, combined with highly accurate clocks, could provide the basis for a new and revolutionary navigation system.

1.3 Air Force Project 621B

Each of the systems, Transit and Timation, has its drawbacks. The Transit system was too slow and too intermittent to keep up with the high speeds of airplanes. The Timation system was easy to jam and could only provide two-dimensional positioning. In 1963, the Space Division of the Air Force began supporting a study known as Project 621B. The Air Force requested Aerospace to lead Project 621B. The Air Force placed a high priority on finding a better three-dimensional positioning system for its aircraft; they hoped to obtain an accuracy of 15 meters or 50 feet.

Project 621B focused on developing a new type of satellite-ranging signal based on pseudorandom noise (PRN) codes. The signal modulation is a repeated digital sequence of fairly random bits, ones or zeros. Although the codes are random, in reality, the codes are generated using a mathematical algorithm. These codes have a predictable pattern, which is periodic and can be replicated by a suitably equipped receiver (Figure 1). The advantage of using PRN is having the ability to reject noise, jamming or deliberate interference. The code modulation is different for each satellite, which significantly minimizes the signal interference.

The procedure of the GPS range determination can be described as follows. Assume for a moment that both the satellite and the receiver clocks are synchronized with each other. When the PRN code is transmitted from the satellite to the receiver, the receiver generates an exact replica of that code. By comparing the transmitted code and its replica, the receiver can compute the signal travel time.

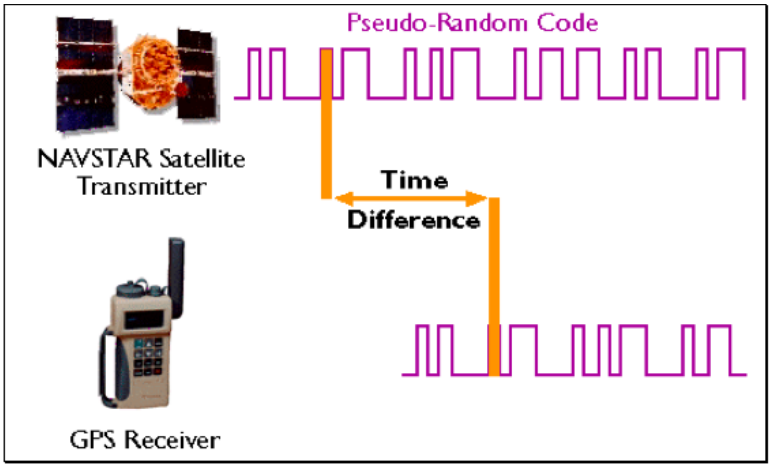


Figure 1: A simple example of a short PRN code sequence.

1.4 Joint Program Office (JPO)

In the 1970s, there were three competing projects: John Hopkins University Applied Physics Laboratory's (APL) Transit, Naval Research Laboratorys (NRL) Timation, and Air Force's Program 621B. Each of these three projects demonstrated great potential for global navigation. However, none of them could locate a user's location to a tolerable accuracy as a stand alone. Ivan Getting, the first president of Aerospace Corporation, knew that any full-scale GPS effort would need support from DOD. It was becoming increasingly clear that some type of coordination was needed among the three competing ideas to fully develop a satellite-based navigation system.

To increase efficiency and reduce interservice bickering, the Joint Program Office (JPO) was formed in 1973. The program was located at the the Air Force's Space and Missile Organization, with the Air Force as the lead service. DOD appointed an Air Force Colonel, Bradford Parkinson, who was in charge of the Program 621B in the Air Force Space Division, to lead the possibility of merging the three competing concepts. Parkinson believed that the three concepts must work together in order to determine user's position, velocity, and time (PVT). A meeting with twelve military officers was scheduled at the Pentagon in 1973, over Labor Day weekend to discuss and finalize the multi-service system. After a long, tireless brainstorming session, they came up with a blueprint for NAVSTAR/Global Positioning System, bringing precise positioning ability for the military and civilians.

The 1973 satellite orbital configuration plan was a total of 24 satellites, eight in each of three orbits with inclinations of 63 degrees (Figure 2). The orbits were equally spaced around the equator and the orbital altitudes were 10,980 nautical miles. The three satellite orbits were initially selected because it would be easier to have orbital spares. There are three spare satellites in orbit that could easily replace any single failure satellite. If a GPS satellite fails, the rocket booster in a spare satellite is released to enter its designated orbit. The satellite configuration provided a minimum of six satellites in view at any time, with a maximum of eleven.

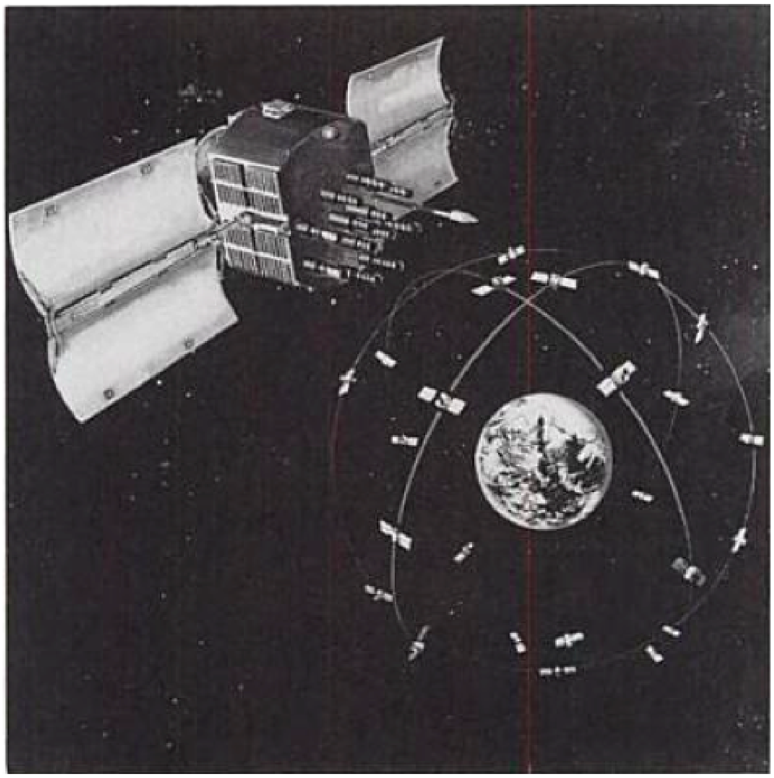


Figure 2: GPS orbital configuration of 8 satellites in each 3 orbits.

2 System Segmentation

"Genius is one percent inspiration and ninety-nine percent perspiration."

Thomas A. Edison

The blueprint for NAVSTAR/GPS was approved by the DOD Research and Engineering in 1973. Basically the blueprint approved in 1973 is still the current system in use today. The system used atomic clocks in its satellites and orbits similar to those used for the Timation system, but with higher altitudes to provide a 12-hour period (one complete orbit revolution). The structure and frequencies of the digital signals was essentially the same as those used in Project 621B.

The GPS is divided into three main segments; the space segment, the control segment, and the user segment (Figure 3). The space segment consists of a constellation of 24 NAVSTAR satellites. Each satellite broadcasts radio frequency ranging codes and a navigation data message. The control segment consists of a network of monitoring and controlling facilities. It is used to manage the satellite constellation and update the satellite data messages. The user segment consists of a variety of radio navigation receivers specifically designed to receive, decode, and process the GPS satellites ranging codes and navigation data messages. The three segments contribute to overall accuracy, reliability, and functionality.

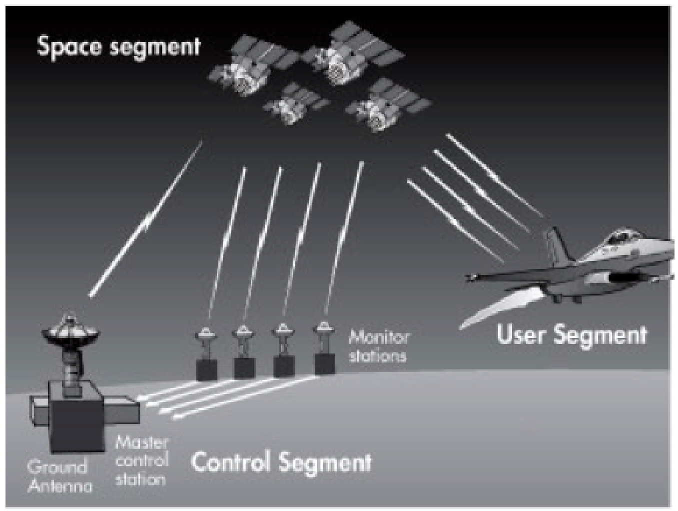


Figure 3: NAVSTAR/GPS major segments.

2.1 Space Segment

The configuration of the GPS satellites has always represented a compromise between user needs, budgetary constraints, and technical feasibility. Deciding where to put the satellites was not an easy task. Researchers and data analysts were under extreme pressure to answer the fundamental questions of constellation management: how many, how high, how close, and how long. Despite the tremendous pressure and budgetary constraints, the constellation management goal has never changed: to provide the most functional system for the broadest class of users, given a limited amount of resources.

Early GPS models focused on uniform constellations because they provide the most satellite visibility on a global scale. The initial proposal in the 1970s was for 24 satellites circling 3 orbital planes. Each orbit contained 8 satellites spaced uniformly 45 degrees apart. The inclination angle was set to 63 degrees with orbital period at 11 hours and 58 minutes. The three orbital planes would be perpendicular to one another and equally spaced around the equator.

Most of the early testing and launching was done in Yuma, Arizona. Keep in mind that the goal is to optimize the best satellite configuration under certain criteria. These criteria include: cost, number of satellites, total coverage, and geometric configuration. To test and study the GPS models, several obstacles had to be overcome. For example, the methods used to evaluate coverage over the whole Earth throughout the course of a day relied on point-by-point evaluation over an extensive space-time grid. In addition, GPS receivers only locked onto four satellites at a time, so every combination of four satellites had to be examined individually. This method was cumbersome and slow. Luckily, a breakthrough in computer software made the computations much more feasible.

After much research in optimizing performance through constellation management, the current GPS model today includes 24 satellites; each satellite completes one orbit in 11 hours and 58 minutes. Though the model initially proposed seems perfect, there were some modifications that needed to be made. The differences between the initial model and today's model are that the satellites are arranged in 6 orbital planes at 60 degrees apart with 4 satellites in each plane (Figure 4). Theoretically, three or more GPS satellites will always be visible from most points on the earth's surface, and four or more GPS satellites can use be used to determine an observer's position anywhere on the earth's surface 24 hour per day.

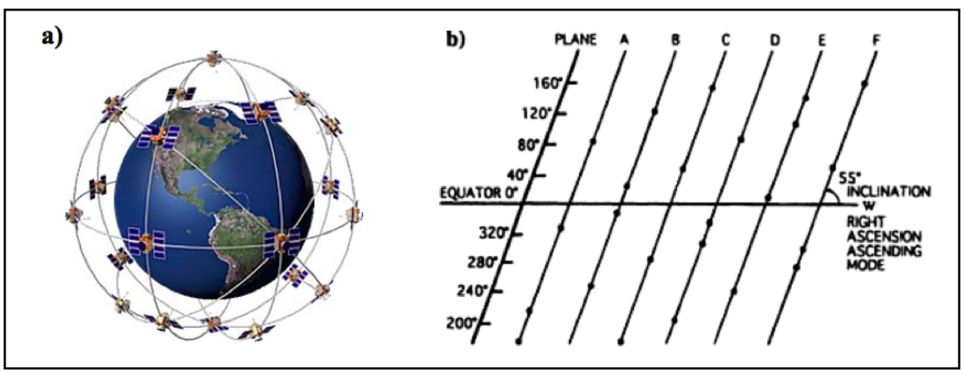


Figure 4: GPS satellite constellation: a) the six equally orbit planes; b) satellite positions on each of the six orbit planes.

In addition to finding the best optimal GPS satellite configuration, the mechanics of each satellite are equally important. Under the same constraints, scientists and researchers must decide on the quality and reliability of the parts in a satellite (Figure 5). A GPS satellite weighs approximately 2,000 pounds and is about 17 feet across with the solar panels extended (about the same length as a car). Transmitter power is only 50 watts or less. Satellites orbit around the earth at about 12,500 miles above the earth's surface. They are constantly moving, making two complete orbits in less than 24 hours. They travel at a speed of roughly 7,000 miles an hour. GPS satellites are powered by solar energy. They have backup batteries onboard to keep them running in the event of a solar eclipse, when there's no solar power. Small rocket boosters on each satellite keep them flying in the correct path. Each satellite is built to last about 10 years. Replacements are constantly being built and launched into orbit.

Figure 5: Navstar GPS IIF.

One of the most important parts of a satellite is the atomic clock. Each GPS satellite has three or four onboard caesium or rubidium atomic clocks. An atomic clock is a clock that uses an electronic transition frequency in the microwave region of the electromagnetic spectrum of atoms as a frequency standard for its timekeeping element. Atomic clocks are among the most accurate time and frequency standards known. The cost for an atomic clock ranges from a few thousand dollars to about $\$20,000$ depending on the type.

Although the atomic clock is highly accurate, it is not perfect. This means that the satellite clock error is about 8.64 to 17.28 nanoseconds per day. The corresponding range error is 2.59 to 5.18 meters. GPS receivers, in contrast, use inexpensive crystal clocks, which are less accurate than the satellite clocks. As such, the receiver clock error is much larger than that of the GPS satellite clock. A more detailed discussion on time error and its effects on calculating a user's position are given in chapter 4.

2.2 Operational Control Segment

The operational control segment (OCS) has the responsibility of maintaining the satellites and their proper functioning. It tracks the GPS satellites in order to determine and predict satellite location. The OCS monitors the satellites' subsystem health and status, atomic clocks, and atmospheric data. The OCS also updates each satellite with new and corrected navigation messages once per day or as needed. The message includes information about each satellite's positions as a function of time, a satellite's clock parameters, atmospheric data and satellite's almanac. Almanac is a data file that contains the approximate orbit information of all satellites.

To accomplish these functions, the control segment of the GPS system consists of three different physical components (Figure 6): five monitor stations, the ground antennas, and the master control station (MCS). The monitoring stations are located in Hawaii, Kwajalein, Colorado Springs, Ascension Island, and Diego Garcia. The monitor stations passively track the GPS satellites as they pass overhead by making pseudorange and delta range measurements. Each monitor station also contains two cesium atomic clocks referenced to GPS system time. These facilities are unmanned and provide approximately 92 percent tracking coverage of the GPS satellites. That is, the satellites are not continuously visible to the monitor stations. All collected data are transmitted to the MCS for processing.

The ground antenna facilities are collocated with monitor stations at Kwajalein, Ascension Island, Diego Garcia, and Cape Canaveral. These locations have been selected to maximize satellite coverage. These facilities provide the means of commanding and controlling the satellites and

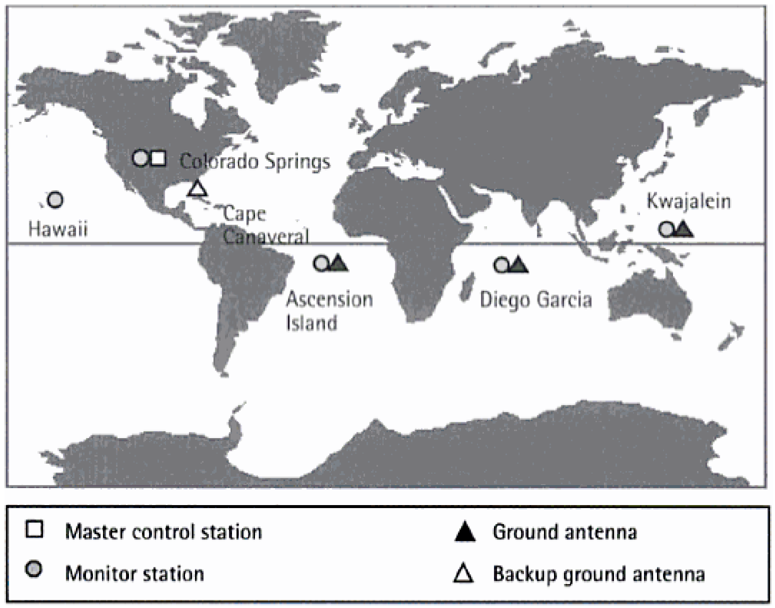


Figure 6: GPS master control center and monitor stations.

uploading the navigation messages and other data.

The MCS is located at Schriever Air Force Base (AFT) in Colorado Springs, CO. The MCS performs a multitude of functions to support the operation of GPS as a system. One principal activity is the processing of the data collected at the remote monitor stations to calculate estimates of the GPS satellite clock, ephemeris, and almanac data. Many steps are involved in this processing; the algorithms used are continually evolving based on control segment experience. The control segment must take meticulous care to ensure that all clock and ephemeris data uploads and other signal transmissions are correct.

2.3 User Segment

The user segment consists of receivers specifically designed to receive, decode, and process the GPS satellite signals. GPS receivers must observe the signal from at least four satellites to obtain three-dimensional position and velocity estimates. There was only one GPS receiver available in 1980, but ten years later, more than 100 different makes and models have become available (Figure 7). Although these receivers have differences in their design, construction, and capabilities, they share a number of basic principles in their operation. Each receiver consists of a number of basic building blocks: an antenna and associated preamplifier, a radio frequency (RF) front-end section, a signal tracker block, a command entry and display, and a power supply.

The important components of the GPS receiver are the antenna, radio frequency, and the signal tracker. The job of the antenna is to convert the energy in the electromagnetic waves arriving from the satellites into an electric current that can be handled by the electronics in the receiver. The size and shape of the antenna are very important, as these characteristics govern in part the ability of the antenna to pick up the GPS signals. The RF section of the GPS receiver translates the frequency of the signals arriving at the antenna. The RF section contains a precision quartz crystal

Figure 7: Types of GPS receivers.

oscillator, an enhanced version commonly found in wristwatches. The signal tracker block in the GPS receiver must be able to isolate the signals from each particular satellite in order to measure the code pseudorange and the phase of the carrier. This isolation is achieved through the use of a number of signal channels in the receiver. The channels in a GPS receiver may be implemented in one of two basic ways. A receiver may have dedicated channels, each continuously tracking a particular satellite. A continuous tracking receiver has five or more hardware channels to track four satellites simultaneously plus other channels to acquire new satellites. Modern military GPS receivers predominantly use a continuous satellite tracking architecture.

The other channelization concept uses one or more sequencing channels. A sequencing channel "listens" to a particular satellite for a period of time, makes measurements on that satellite's signal, and then switches to another satellite. A single-channel must sequence through four satellites to obtain a three-dimensional position. The time-to-first-fix (TTFF) is a measure of the time required for a GPS receiver to aquire satellite signals and navigation data. The time to first fix for a single-channel is very slow, but can be reduced by having a pair of sequencing channels.

A variation of the sequencing channel is the multiplexing channel. With this design, a receiver sequences through the satellites at a fast rate, essentially acquiring all the broadcasts messages from the individual satellites at the same time. With a multiplexing receiver, the time to first fix is 30 seconds or less, the same as a receiver with dedicated multiple channels.

The GPS receivers are rendered out-of-date at a fast pace due to the never-ending technology discoveries. Newer receivers with Wide Area Augmentation System (WAAS) capability can improve accuracy to less than three meters on average. Users can also get better accuracy with Differential GPS (DGPS), which corrects to within an average of three to five meters. The technology is changing so quickly that we will see further advances in the next decade as even higher-speed, lower-power integrated circuits come out of the research labs.

3 Determining User's Position

"Pure mathematics is, in its way, the poetry of logical ideas."

Albert Einstein

Like the internet, the GPS technology has established itself as an indispensable component in our daily life. This technology now boasts more civilian than military users. What makes the GPS so attractive to civilians is its capability to provide accurate three-dimensional navigation anywhere in the world and at any time. Any civilian can use the GPS by simply purchasing a low cost GPS receiver. However, most of its users know relatively little about the mathematical methods behind the GPS. There is a notion that it is too complicated. The GPS is complicated, but the basic idea of how the system works is quite intuitive.

There are a wide variety of GPS positioning techniques that are used in today's GPS receiver. GPS positioning techniques may be categorized as being predominantly based on code or carrier measurements. For the purpose of this chapter, we will only discuss code measurements (also referred to as pseudorange measurements) with single point positioning since it is generally simple. The only disadvantage to this technique is low accuracies.

3.1 GPS Single Point Positioning

Single point positioning (Figure 8) is achieved by using the measurements from four or more satellites at a single receiver on the earth's surface. Each satellite transmits signal of its location at a given time. Solutions may be attained almost instantaneously, using an inexpensive hand-held GPS receiver. Almost all of the GPS receivers currently available on the market are capable of displaying their point-positioning coordinates. The success of this technique depends greatly on having good satellite geometry. The ideal configuration is one satellite directly above the antenna of the GPS receiver and the other three satellites spread apart at 120 degrees with an inclination of 20 degrees. The geometry of satellites is quantified by the geometrical dilution of precision (GDOP). Satellite configurations exemplifying poor or good GDOP are illustrated in Figure 9.

The basic mathematical methods behind GPS single point positioning will be explained using a simplified geometric model. The single point positioning technique requires the satellites to have good geometry, four or more satellite ranges, and the following assumptions:

1. *Receiver and satellite clock are synchronized.*

2. *Cause of signal delay (ionosphere and troposphere) does not exist.*

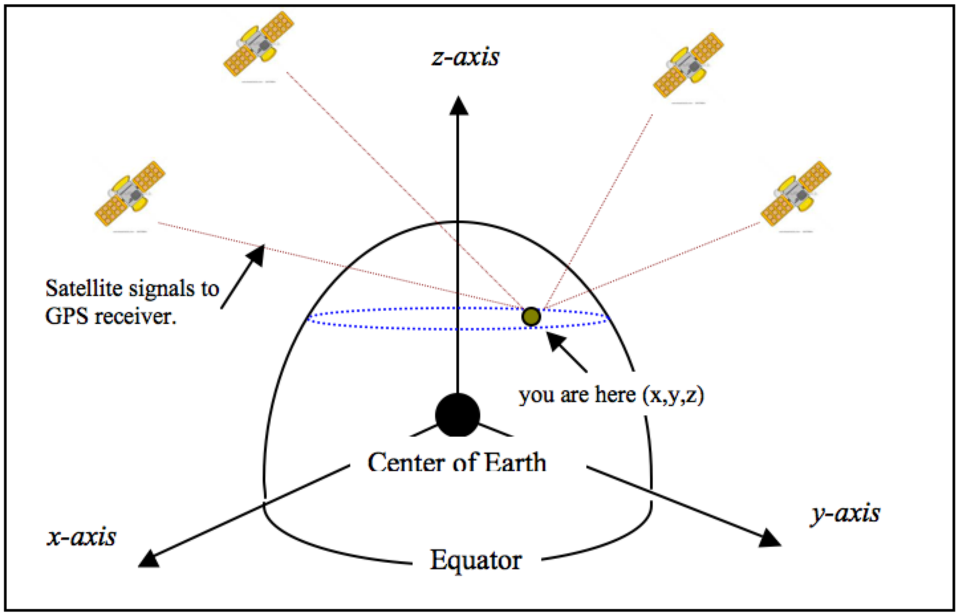


Figure 8: Single Point Positioning.

3. *Satellite orbit errors, receiver noise, and multipath ambiguities do not exist.*

4. *Satellite and receiver position are in the earth center, earth fixed (ECEF) coordinate system.*

5. *Since the satellites are in motion, the GPS receivers must then take into account Doppler data, which represents the relative speeds between the satellites and the receiver. However, for the purposes of this explanation, we will consider the system to be stationary.*

Under these assumptions, the basic measurement made by a GPS re-

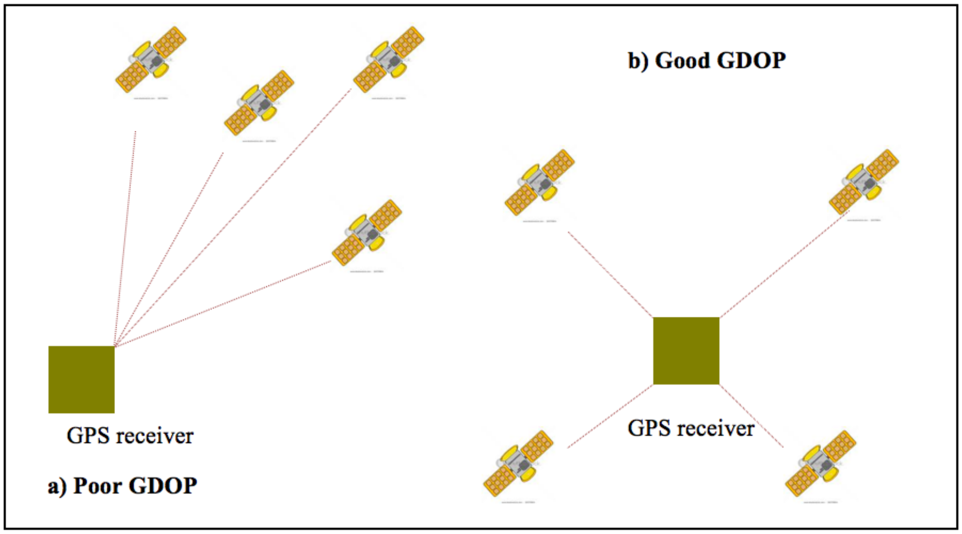


Figure 9: Poor and Good GDOP.

ceiver is the time required for a signal to propagate from a GPS satellite to the receiver. Whenever a problem involves how fast, how far, or for how long, the distance formula should come to mind. The distance or range to a satellite is the difference in the time the code is transmitted from the satellite and the time the code is received at a GPS receiver, multiplied by the speed of light. That is,

$$Range = (reception \quad time - emission \quad time) \times speed \quad of \quad light$$

or

$$R = (t_r - t_e)c \tag{1}$$

where c = speed of light (299,792,458 or 3 × 10^8 meters per second), t_r=

GPS signal reception time of the receiver, and t_e= GPS signal emission time of the satellite.

With a single such measurement, one can determine something about the position of the receiver; it must lie somewhere on a sphere centered on the satellite with a radius equal to the measured range (Figure 10). In other words, the satellite acts like a protractor and uses the known range as a radius and draws an imaginary sphere around it. Let's denote this range R_1.

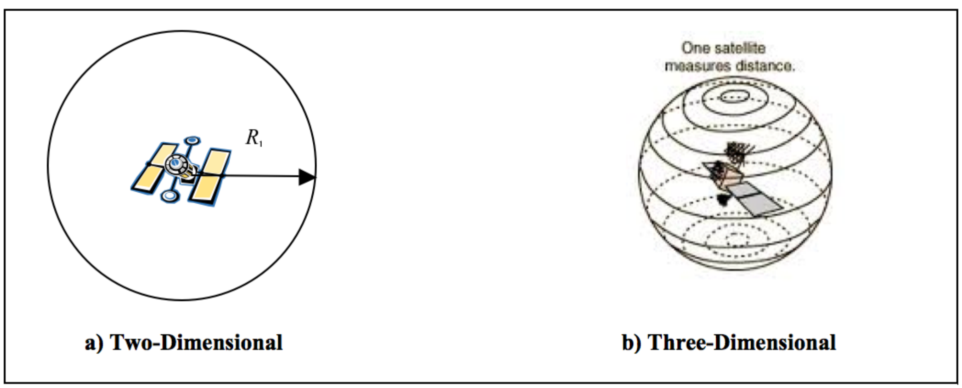


Figure 10: Range determination from a single source.

Next, if the second satellite simultaneously makes a range measurement of radius R_2, then the receiver must also lie anywhere on the sphere. The two spheres will intersect with the locus of intersection points forming a circle (Figure 11). It is important to understand and visualize the intersection of these two spheres. In 2 dimensions, the intersection of the two circles consists of at most two points, A and B. The user's position could be at either one of these two points. In 3 dimensions, the intersection of two spheres is a circle. The user's position could be anywhere on the circle. Intuitively, more information is needed to determine the user's position.

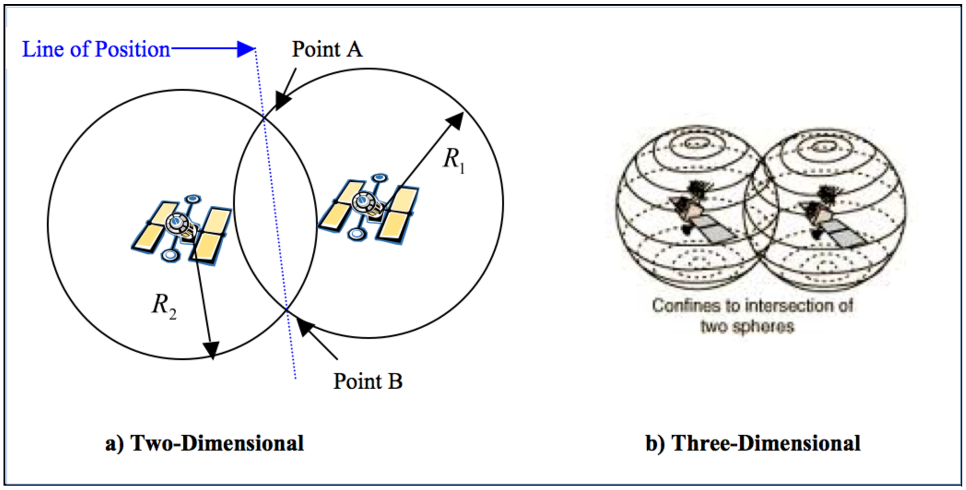


Figure 11: Ambiguity resulting from measurements to two sources.

A third simultaneous range measurement to a third satellite, radius of R_3, gives a third sphere that intersects the other two spheres at only two points (Figure 12). One of these points can be dismissed immediately as being the location of our receiver because it will lie far out in space. In 2 dimensions, three ranges are enough to give exact user's position.

Theoretically, only three ranges to three simultaneously tracked satellites are needed to locate a user's position because one point can be elimi-

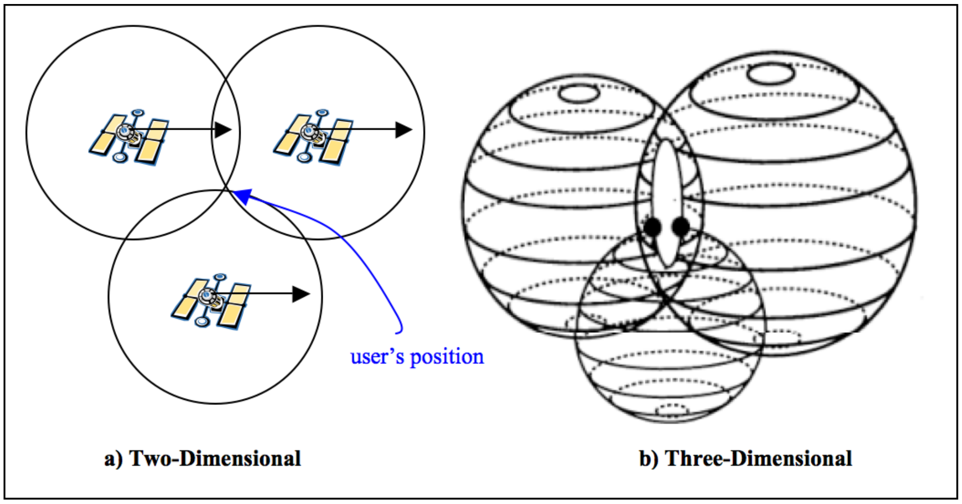


Figure 12: Position ambiguity removal by additional measurements.

nated. However, recall that a single point positioning method needs at least four satellites. The fourth satellite is needed to account for the receiver and satellite clock offset. More details about the receiver and satellite clock error are in Chapter 4. To summarize, the chart below (Figure 13) illustrates the geometry for each of the cases discussed earlier under our simplifying assumptions.

Intersection	Equivalency	Result
Intersection of 2 spheres	Circle	Circle
Intersection of 3 spheres	Circle ∩ Sphere	Two points
Intersection of 4 spheres	Two points ∩ Sphere	One point

Figure 13: Results from sphere intersection.

The position of the receiver can be represented by the following three mathematical equations:

$$\begin{aligned} R_1 &= \sqrt{(x-x_1)^2+(y-y_1)^2+(z-z_1)^2} \\ R_2 &= \sqrt{(x-x_1)^2+(y-y_1)^2+(z-z_1)^2} \\ R_3 &= \sqrt{(x-x_1)^2+(y-y_1)^2+(z-z_1)^2} \end{aligned} \tag{2}$$

where $(x, y, z)=$ unknown receiver position, (x_1, y_1, z_1) , (x_2, y_2, z_2) and (x_3, y_3, z_3) are satellite locations in space, and R_1, R_2 and R_3 are the range measurements solved through $R = (t_r - t_e)c$. This system of three equations can be solved through systems of linear algebra.

3.2 Examples: Linear System of GPS with no Errors

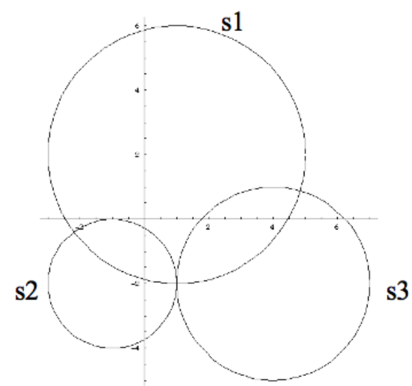
There are two examples: one in 2 dimensions and the other in 3 dimensions. Satellite clock bias and the relativistic effects have been estimated and accounted for. The examples given are based on the same assumptions discussed early in section 3.1. The purpose of the examples is to indicate to the reader how one might solve the relevant linear systems of equations. Take note that the GPS receivers do not operate along the lines of the examples. Nevertheless, they serve as a practical conceptual tool for analyzing GPS performance. For instance, the examples will give insight about the process of triangulation and the general framework for GPS. *(Hint: use matrices).*

Example 1: Two-Dimensional User Position

The position and ranges of three satellites are given in Table 1. Calculate the user's position (x, y).

Table 1. Satellite data.

Satellite	*Position*	*Range*
1	(1,2)	4
2	(−1,−2)	2
3	(4,−2)	3



Graph of three satellites.

Since the range of each satellite is given, the formula $R = (t_r - t_e)c$ is not needed. To calculate the user's position, use the distance formula. Set up the three equations and solve for (x, y) using algebraic methods (Figure 14).

Satellite	*Position*	*Range*	*Distance Formula*
1	(1,2)	4	$4 = \sqrt{(x-1)^2 + (y-2)^2}$
2	(−1,−2)	2	$2 = \sqrt{(x-(-1))^2 + (y-(-2))^2}$
3	(4,−2)	3	$3 = \sqrt{(x-4)^2 + (y-(-2))^2}$

Figure 14: The position and range of each satellite is represented in the distance formula.

Simplify the three equations,

$$(x-1)^2 + (y-2)^2 = 16$$

$$(x+1)^2 + (y+2)^2 = 4$$

$$(x-4)^2 + (y+2)^2 = 9.$$

Expand all the squares and subtract the first equation from each of the other two. The results: two linear systems of equations with two unknowns,

$$4x + 8y = -12$$

$$-6x + 8y = -22.$$

This system can be solved using basic elimination or substitution method. However, it is best to solve it using an augmented matrix because this will help build the foundation for more challenging linear systems of equations.

The Gauss-Jordan elimination method is used so that the matrix is in reduced row echelon form. The same solution could be found using either method. The user's position is at $(1, -2)$.

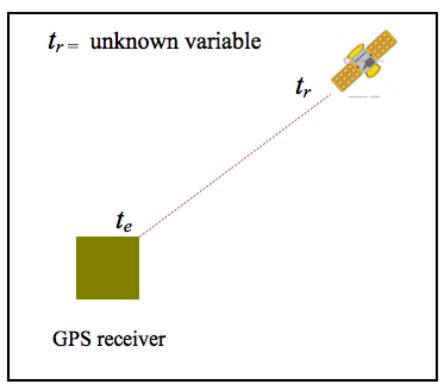
See Appendices for iteration codes (3.2 Example 1).

Example 2: Three-Dimensional User Position

Positions and times for four satellites are given in Table 2. Calculate the user's position . Keep in mind that there is a $xyz - coordinate$ system with the earth centered at the origin. The point at sea level will have $x^2 + y^2 + z^2 = 1$ in this system and time will be measured in units of milliseconds. The GPS system finds distances by knowing how long it takes a radio signal to get from one point to another. For this example, the speed of light is 0.47 (in units of earth radii per millisecond).

Table 2. Satellite data.

Satellite	*Position*	t_e
1	(1,2,0)	19.9
2	(2,0,2)	2.4
3	(1,1,1)	32.6
4	(2,1,0)	19.9



To locate the user's position using the satellite data in Table 2, two formulas are needed;

$$R = (t_r - t_e)c$$

$$R = \sqrt{(x - x_1)^2 + (y - y_1)^2 + (z - z_1)^2}.$$

For satellite 1, the signal was sent at time 19.9 and arrived at time t_r. Traveling at a speed of light, that makes the range $R = 0.047(t_r - 19.9)$. The satellite's position at $(1, 2, 0)$ can also be expressed as

$$R = \sqrt{(x - 1)^2 + (y - 2)^2 + (z - 0)^2}.$$

Combining these results leads to the equation,

$$(x - 1)^2 + (y - 2)^2 + (z - 0)^2 = 0.047^2(t_r - 19.9)^2.$$

The positions of the other three satellites can be derived using the same logic (Figure 15). These are not linear equations, but can be linearized algebraically.

1. Simplify the four equations.

   i) $2x + 4y + 0z - 2(0.047^2)(19.9)t_r = 1^2 + 2^2 + 0^2 - 0.047^2(19.9^2) + x^2 + y^2 + z^2 - 0.047^2 t_r^2;$

   ii) $4x + 0y + 4z - 2(0.047^2)(2.4)t_r = 2^2 + 0^2 + 2^2 - 0.047^2(2.4^2) + x^2 + y^2 + z^2 - 0.047^2 t_r^2;$

Satellite	Position	t_e	Equations
1	(1,2,0)	19.9	$(x-1)^2+(y-2)^2+(z-0)^2=0.047^2(t_r-19.9)^2$
2	(2,0,2)	2.4	$(x-2)^2+(y-0)^2+(z-2)^2=0.047^2(t_r-2.4)^2$
3	(1,1,1)	32.6	$(x-1)^2+(y-1)^2+(z-1)^2=0.047^2(t_r-32.6)^2$
4	(2,1,0)	19.9	$(x-2)^2+(y-1)^2+(z-0)^2=0.047^2(t_r-19.9)^2$

Figure 15: The position and range of each satellite is represented in the distance formula.

iii) $2x + 2y + 2z - 2(0.047^2)(32.6)t_r = 1^2 + 1^2 + 1^2 - 0.047^2(32.6^2) + x^2 + y^2 + z^2 - 0.047^2 t_r^2$;

iv) $4x + 2y + 0z - 2(0.047^2)(19.9)t_r = 2^2 + 1^2 + 0^2 - 0.047^2(19.9^2) + x^2 + y^2 + z^2 - 0.047^2 t_r^2$

2. Subtract the first equation from each of the other three.

   i) $2x - 4y + 4z + 2(0.047^2)(17.5)t_r = 8 - 5 + 0.047^2(19.9^2 - 2.4^2)$

   ii) $0x - 2y + 2z + 2(0.047^2)(12.7)t_r = 3 - 5 + 0.047^2(19.9^2 - 32.6^2)$

   iii) $2x - 2y + 0z + 2(0.047^2)(0)t_r = 5 - 5 + 0.047^2(19.9^2 - 19.9^2)$

3. End results.

   i) $2x - 4y + 4z + 0.077t_r = 3.86$

   ii) $-2y + 2z - 0.056t_r = -3.47$

iii) $2x - 2y = 0$

The resulting three equations are linear with three unknowns due to the coefficients of the quadratic terms cancelling out. Next, formulate the linear system as an augmented matrix and solve:

$$\left[\begin{array}{cccc|c} 2 & -4 & 4 & 0.077 & 3.86 \\ 0 & -2 & 2 & -0.056 & -3.47 \\ 2 & -2 & 0 & 0 & 0 \end{array}\right] \sim \left[\begin{array}{cccc|c} 1 & 0 & 0 & 0.095 & 5.41 \\ 0 & 1 & 0 & 0.095 & 5.41 \\ 0 & 0 & 1 & 0.067 & 3.67 \end{array}\right].$$

Looking at the augmented matrix, the system cannot have a unique solution. However, if the satellite data are accurate, then there must be a solution to the original system of quadratic equations and this linear system must be consistent.

By deriving the general solution, it will be possible to express three of the unknowns in terms of the fourth. Thus, the general solutions are:

$$x = 5.41 - 0.095t_r$$

$$y = 5.41 - 0.095t_r$$

$$z = 3.67 - 0.067t_r$$

$$t = free.$$

Return to the first equation from satellite 1 and substitute the above expressions for x, y, and z. The quadratic equation in one variable should look like this,

$(5.41 - 0.095t_r - 1)^2 + (5.41 - 0.095t_r - 2)^2 + (3.67 - 0.067t_r)^2 = 0.047^2(t_r - 19.9)^2$

or

$$0.02t_r^2 - 1.88t_r + 43.56 = 0.$$

Solving the quadratic equations, the solutions to are 43.1 and 50.0. We can eliminate 43.1 since it will give us a point that lies 4,000 miles above the earths surface.

The user's position (x, y, z) is $(0.667, 0.667, 0.332)$ with length 0.9997. That places the point on the surface of the earth.

4 Pseudorange

"Perfect numbers like perfect men are very rare."

Rene Descartes (1596-1650)

To determine the user's position, the GPS makes use of the concept of time-of-arrival ranging. This concept entails measuring the time it takes for a signal transmitted by a satellite at a known location to reach the user's receiver. The first time a receiver is operated, it must perform an iterative search for the first satellite signal unless the receiver was loaded with a recent constellation almanac. The almanac gives the approximate orbit for each satellite and is used to estimate the range to a satellite. In chapter 3, the distance and rate formulas play a very important role in GPS calculations. Examples given assumed no measurement errors, so that all satellite ranges would intersect at one point. These assumptions, however, are fallacious. Realistically, our system of equations is almost certain to be inconsistent.

This chapter will introduce how GPS receivers modify the mathematical range computations to accommodate the time errors and deal with inconsistent system of equations. Recall that the first assumption was that the clock in the GPS receiver was synchronized with the clocks in the satellites. Realistically, when a GPS receiver is switched on, its clock will, in general, not be synchronized with the satellite clocks. This is due to the fact that GPS receivers use inexpensive crystal clocks while GPS satellites use atomic clocks. The range measurements the GPS receiver makes are biased due to the satellite and receiver clock errors. Therefore, the ranges are referred to as pseudoranges.

4.1 Satellite and Receiver Clock Offset

Satellite and receiver clock offset causes errors to the GPS range measurements. These errors are common to all users observing the same satellite. The GPS satellite clocks, although highly accurate, are not perfect. The satellite clock error is about 8.64 to 17.28 nanoseconds per day. The corresponding range error is about 2.59 to 5.18 meters. One nanosecond error is equivalent to a range error of about 30 centimeters. GPS receivers, in contrast, use inexpensive crystal clocks that produce a much larger error than the satellite clocks.

The error caused by the satellite and receiver clock causes the three spheres with radii equal to the measured pseudoranges to not have a common point of intersection (Figure 16). Recall that the range is the difference in time received and transmitted, multiplied by the speed of light: $R = (t_r - t_e)c$. Taking the satellite and receiver clock errors into account, the pseudorange can be represented as

$$R = (t_r - t_e)c - (\delta t_r - \delta t_s)c \qquad (3)$$

where δt_r and δt_s denote the clock errors of the receiver and satellite, respectively. The GPS satellite clock error term, δt_s, is known through GPS satellite orbit determination. If the receiver clock error can be determined, then the pseudoranges can be corrected and the position of the receiver determined. So, we actually have four unknown quantities or parameters that we must determine: the three coordinates of our position (x, y, z) and the receiver clock offset.

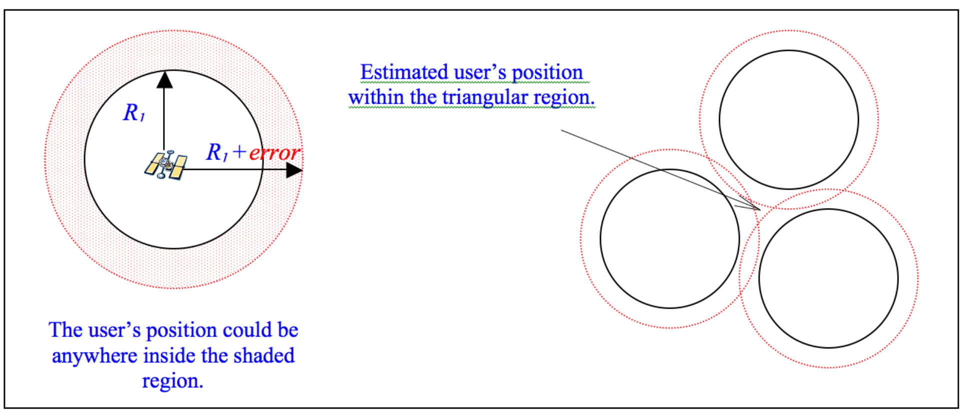


Figure 16: Effects of time errors on user's position in two-dimensional.

It is mathematically impossible to uniquely determine the values of four parameters given only three measurements. The obvious solution is to simultaneously measure an additional pseudorange to a fourth satellite. The geometric ranges can be represented as follows,

$$\begin{aligned} R_1 &= \sqrt{(x-x_1)^2+(y-y_1)^2+(z-z_1)^2}-(\Delta T)c \\ R_2 &= \sqrt{(x-x_2)^2+(y-y_2)^2+(z-z_2)^2}-(\Delta T)c \\ R_3 &= \sqrt{(x-x_3)^2+(y-y_3)^2+(z-z_3)^2}-(\Delta T)c \\ R_4 &= \sqrt{(x-x_4)^2+(y-y_4)^2+(z-z_4)^2}-(\Delta T)c. \end{aligned} \tag{4}$$

The set of four equations must be solved simultaneously to obtain the values for (x, y, z) together with the clock offset, ΔT. Keep in mind that if the clocks of the GPS receiver and the satellite were perfectly synchronized, the time offset would be zero.

The system of equations for four satellite ranges is multivariate and non-linear. The pseudorange measurements are dependent on the receiver coordinates in a nonlinear way because of the squares and square roots. Consequently, these non-linear equations can be solved for the unknowns by employing iterative techniques based on linearization. The most famous and commonly used today is the Newton-Raphson method.

4.2 Introduction to Newton-Raphson Method

In numerical analysis, Newton's method (also known as Newton-Raphson method), named after Isaac Newton and Joseph Raphson, is perhaps the best known method for finding successively better approximations to the zeroes (or roots) of a real-valued function. This iterative method requires only one initial guess and can converge quickly, especially when the initial guess is near the desired root.

The Newton-Raphson method is based on the principle that if the initial guess of the root of $f(x) = 0$ is at x_i where $i = 0$, then if one draws the tangent to the curve at $f(x_i)$, the point x_{i+1} where the tangent crosses the $x - axis$ is an improved estimate of the root (Figure 17). The definition of the slope of a function at $x = x_i$ is

$$f'(x_i) \approx \frac{f(x_i) - 0}{x_i - x_{i+1}}, \tag{5}$$

which gives the iterative formula

$$x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)}. \tag{6}$$

The iterative formula above is used for solving nonlinear equations of the form $f(x) = 0$. So starting with an initial guess, x_i, one can find the next guess, x_{i+1}. To find the approximate error $|E_a|$, use

$$|E_a| = |\frac{x_{i+1} - x_i}{x_{i+1}}| \times 100. \tag{7}$$

Depending on the error tolerance E_t, if $|E_a| > E_t$, then repeat the algorithm, else stop the algorithm. The process is repeated until a sufficiently accurate value is reached.

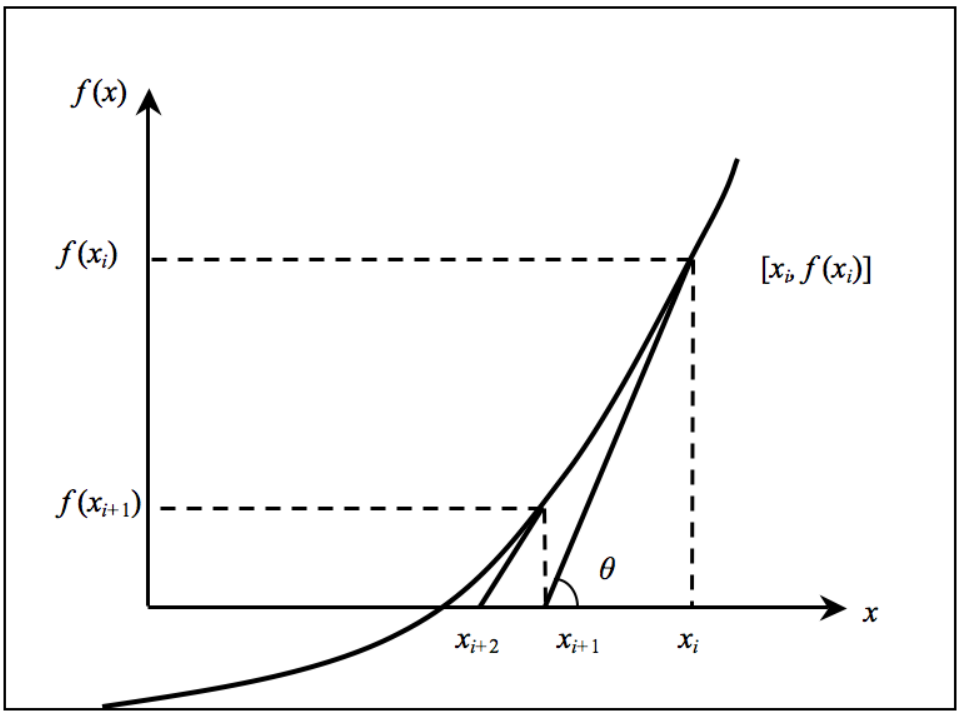


Figure 17: Geometrical illustration of the Newton-Raphson method.

There are advantages and disadvantages to Newton-Raphson method. Two advantages are: (1) it converges quickly, if it converges, and (2) it requires only one guess. However, there are four disadvantages to this method: (1) the initial guess can't be zero (dividing by zero), (2) root jumping, (3) oscillations near local maxima or minima, and (4) inflection points that are close to the roots.

The first disadvantage is the initial guess can not be an x-value at which the derivative is zero. The iteration equation is $x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)}$. If $f'(x_i)$ is equal to zero, then x_{i+1} cannot be defined. The second disadvantage is root jumping: a perfect example is the function $f(x) = \sin x$. The sine function has a periodic behavior and many roots that satisfy $f(x) = \sin x = 0$. Therefore, when finding a particular root not close to zero, the method ends up giving the root closest to zero and skip the other ones (Figure 18).

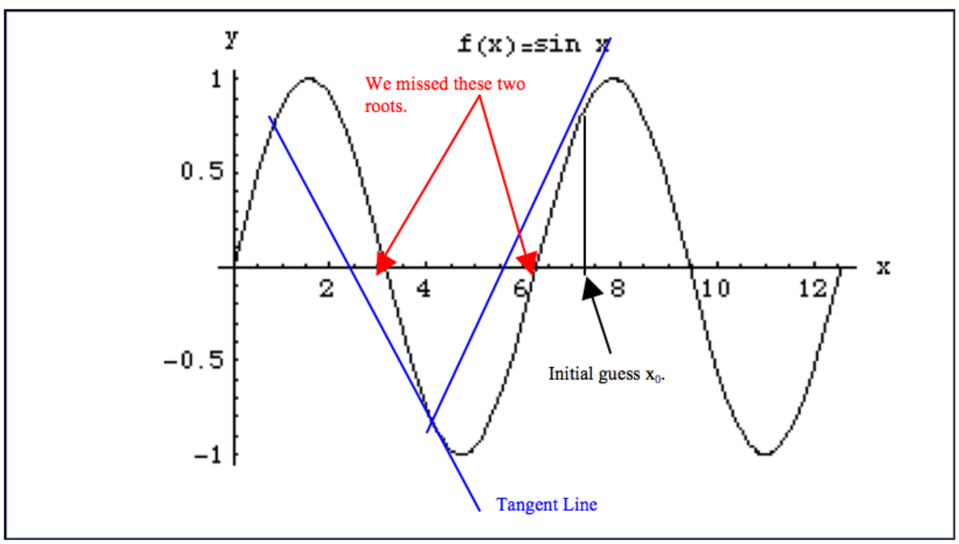


Figure 18: Geometrical illustration of the sine function using Newton-Raphson method.

The third disadvantage consists of the oscillations near local maxima or minima. A great example is a function of a parabola (Figure 19). When finding the roots closest to zero, the iterative solution will start bouncing around and oscillates closer or around $x = 0$. This happens because the parabolic function does not have real roots. The last disadvantage consists

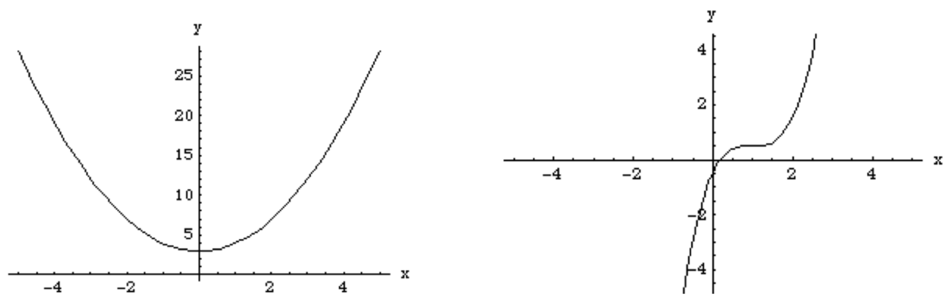


Figure 19: *(Right)* Plot of $f(x) = x^2 + 3$. *(Left)* Plot of $f(x) = (x-1)^3 + 0.512$.

of inflection points that are close to the roots. An inflection point is a change in concavity. The best function that can be used to demonstrate this effect is the cubic function $f(x) = (x-1)^3 + 0.512$ (Figure 19). The inflection point of this function occurs at $x = 1$ and the solution is at $x = 0.200$. If the initial guess is at $x = 5$, then the fifth and sixth iteration solution is $x = 0.9259$ and $x = -30.12$, respectively. The solution starts to converge towards the solution, but at the sixth iteration, it diverges. Then it starts to converge again until the eighteenth iteration where $x = 0.200$.

The Newton-Raphson Method does have drawbacks, however, these do not affect the GPS mathematical computations. The equations used to calculate user's position are not sine, parabolic, or cubic functions. The initial guess is always going to be somewhere on the earth's surface. All GPS receivers have the capability to store all previous locations of the user's whereabouts.

4.3 Multivariate Newton-Raphson Method

The Newton-Raphson method for finding the root of a nonlinear function can be extended to solve a system of nonlinear equations. A general nonlinear system

$$\begin{aligned} f_1(x_1, x_2, x_3, \ldots, x_i) &= 0 \\ f_2(x_1, x_2, x_3, \ldots, x_i) &= 0 \\ f_3(x_1, x_2, x_3, \ldots, x_i) &= 0 \\ &\vdots \\ f_j(x_1, x_2, x_3, \ldots, x_i) &= 0 \end{aligned} \tag{8}$$

can be written in a more compact vector form as $F(\overrightarrow{x}) = 0$, where

$$\overrightarrow{x}^{(k)} = [x_1, x_2, x_3, \ldots, x_i]$$

$$F(\overrightarrow{x}) = \begin{bmatrix} f_1(\overrightarrow{x}) \\ f_2(\overrightarrow{x}) \\ f_3(\overrightarrow{x}) \\ \vdots \\ f_j(\overrightarrow{x}) \end{bmatrix}, 0 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}.$$

$$i = \text{index over variables}$$

$$j = \text{index over functions}$$

$$k = 1, 2, \ldots \text{number of iterations}$$

Here the tangent plane to the surfaces given by the equations to solve is the analogy to the tangent line in the single variable case. The tangent plane to the system of non-linear equations can be evaluated using the Jacobian matrix and the Residual vector at the k^{th} iteration. An example for solving a single-variable problem using Jacobian and Residual vectors is in section 4.5, example 1. The Jacobian matrix consists of partial derivatives of f and the Residual vector consists of values of f evaluated at the initial estimated values.

$$J^{(k)}(\overrightarrow{x}^k) = \begin{bmatrix} \frac{\partial f_1}{\partial x_1}(\overrightarrow{x}^k) & \frac{\partial f_1}{\partial x_2}(\overrightarrow{x}^k) & \cdots & \frac{\partial f_1}{\partial x_i}(\overrightarrow{x}^k) \\ \frac{\partial f_2}{\partial x_1}(\overrightarrow{x}^k) & \frac{\partial f_2}{\partial x_2}(\overrightarrow{x}^k) & \cdots & \frac{\partial f_2}{\partial x_i}(\overrightarrow{x}^k) \\ \frac{\partial f_3}{\partial x_1}(\overrightarrow{x}^k) & \frac{\partial f_3}{\partial x_2}(\overrightarrow{x}^k) & \cdots & \frac{\partial f_3}{\partial x_i}(\overrightarrow{x}^k) \\ \vdots & \vdots & \cdots & \vdots \\ \frac{\partial f_j}{\partial x_1}(\overrightarrow{x}^k) & \frac{\partial f_j}{\partial x_2}(\overrightarrow{x}^k) & \cdots & \frac{\partial f_j}{\partial x_i}(\overrightarrow{x}^k) \end{bmatrix}, R^{(k)}(\overrightarrow{x}^k) = \begin{bmatrix} f_1(\overrightarrow{x}^k) \\ f_2(\overrightarrow{x}^k) \\ f_3(\overrightarrow{x}^k) \\ \vdots \\ f_j(\overrightarrow{x}^k) \end{bmatrix}$$

The next approximation can be obtained by solving the linear system $J^{(k)}\delta^{(k)} = -R^{(k)}$ where $\delta^{(k)} = \overrightarrow{x}^{(k+1)} - \overrightarrow{x}^{(k)}$. Then, use the values $\delta^{(k)}$ to compute the new estimated values using $\overrightarrow{x}^{(k+1)} = \overrightarrow{x}^{(k)} + \delta\overrightarrow{x}^{(k)}$. Repeat the iterations until the difference between two consecutive iteration results is below an acceptable threshold. In general, 2% or lower is acceptable.

4.4 Conclusion

In a realistic situation, the systems of equations generated by satellite ranging measurements will most likely be inconsistent. Due to this reason, we have to use the Newton-Raphson method to estimate the user's position. The Newton-Raphson Method is used in GPS receivers when ranging from four satellites with four unknowns; (X, Y, Z) and ΔT. The summary below illustrates the steps to compute the iterations to get the next best estimate of a user's position with one and two variable(s).

Steps to Newton-Raphson iterations.

1. Set all functions equal to zero.
2. Start with an initial guess.
3. Calculate the Jacobian and Residual Matrix.
4. Find the next approximation to the desired solution by solving the linear system.
5. Calculate the new initial values.
6. Calculate the approximate error. If error is sufficiently small, stop the iteration, else repeat step 3 using the new initial values.

4.5 Examples

Example 1 illustrates linear equations in one variable, while example 2 illustrates non-linear equations in two variables.

Example 1

Given the equation $f(x) = 3x - 2$. Follow the steps of Newton-Raphson Method to approximate where the equation crosses the x-axis.

1. Set all functions equal to zero:

   $f(x) = 3x - 2 = 0$

2. Start with an initial guess:

   $x^{(1)} = [5]$

3. Calculate the Jacobian and Residual Matrix:

   $J^{(1)} = [f'(x^{(1)})] = [3]$

   $R^{(1)} = [f(x^{(1)})] = [13]$

4. Find the next approximation to the desired solution by solving the linear system:

   $J^{(k)}(x^{(k+1)}) - x^{(k)}) = -R(x^{(k)})$ or $3(x^{(2)} - 5) = -13$

5. Calculate the new initial values:

   $x^{(2)} = \frac{2}{3}$

6. Calculate the approximate error. If error is sufficiently small, stop the iteration, else repeat step 3 using the new initial values:

   In this case, it took one iteration to find that the equation crosses the x-axis at $x = \frac{2}{3}$.

Example 2

Given the equation $x^3 = 20$. Follow the steps of Newton-Raphson Method to approximate where the equation crosses the x-axis.

1. Set all functions equal to zero:

   $f(x) = x^3 - 20 = 0$

2. Start with an initial guess:

   $x^{(1)} = [10]$

3. Calculate the Jacobian and Residual Matrix:

   $J^{(1)} = [f'(x^{(1)})] = [300]$

   $R^{(1)} = [f(x^{(1)})] = [980]$

4. Find the next approximation to the desired solution by solving the linear system:

   $J^{(k)}(x^{(k+1)}) - x^{(k)}) = -R(x^{(k)})$ or $300(x^{(2)} - 10) = -980$

5. Calculate the new initial values:

   $x^{(2)} = 6.7333$

6. Calculate the approximate error. If error is sufficiently small (less than 2%), stop the iteration, else repeat step 3 using the new initial

values:

$|E_a| = |\frac{x_{i+1}-x_i}{x_{i+1}}| \times 100$

$|E_a| = |\frac{6.7333-10}{6.7333}| \times 100 = 48.52\%$

Since the approximate error is greater than 2%, repeat step 3. See iteration table below.

Iteration (k)	x**-value**
1	10
2	6.7333
3	4.6359
4	3.4008
5	2.8436
6	2.7202
7	2.7144

In this case, it took seven iterations to find that the equation crosses the x-axis at $x = 2.7144$.

See Appendices for iteration codes (4.5 Example 2).

Example 3

Given the equation $f_1(x_1, x_2) = x_1^2 + x_2^2 - 4$ and $f_2(x_1, x_2) = x_1^2 - x_2 + 1$. Follow the steps of Newton-Raphson Method to approximate the intersection.

1. Set all functions equal to zero:

   $f_1(x_1, x_2) = x_1^2 + x_2^2 - 4 = 0$

   $f_2(x_1, x_2) = x_1^2 - x_2 + 1 = 0$

2. Start with an initial guess:

   $x^{(1)} = [x_1, x_2] = [1, 2]$

3. Calculate the Jacobian and Residual Matrix:

$$J^{(1)}(\overrightarrow{x}^1) = \begin{bmatrix} 2 & 4 \\ 2 & -1 \end{bmatrix}, R^{(1)}(\overrightarrow{x}^1) = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

4. Find the next approximation to the desired solution by solving the linear system:

$$\begin{bmatrix} 2 & 4 \\ 2 & -1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = - \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \delta x^{(1)} = \begin{bmatrix} -0.1 \\ -0.2 \end{bmatrix}$$

5. Calculate the new initial values:

   $x_1^{(2)} = x_1^{(1)} + \delta x_1^{(1)} = 1 + (-0.1) = 0.9$

   $x_2^{(2)} = x_2^{(1)} + \delta x_2^{(1)} = 2 + (-0.2) = 1.8$

6. Calculate the approximate error. If error is sufficiently small, stop the iteration, else repeat step 3 using the new initial values:

   It only took one iteration to find out that the intersection is at $(1.5, 2.5)$.

See Appendices for iteration codes (4.5 Example 3).

4.6 Exercises

Problem 1: Systems of linear equations in two variables.

Graph the equations $f_1(x_1, x_2)$ and $f_2(x_1, x_2)$ given below. Then use the multivariate NewtonRaphson method of linear approximations to find the intersection.

$$f_1(x_1, x_2) = 2x_1 + 2x_2 - 4$$

$$f_2(x_1, x_2) = x_1 - 2x_2 + 5$$

Solution: Linear equations tend to converge very quickly. It only took two iterations to get our true answer with an initial guess of $x = -1$ and $y = -2$.

Even when the initial guess is far off at $x = -100$ and $y = -300$, it still only took two iterations. The intersection point is at $(-0.3333, 2.3333)$.

See Appendices for iteration codes (4.6 Problem 1).

Problem 2: Systems of non-linear equations in two variables.

Graph the equations $f_1(x_1, x_2)$ and $f_2(x_1, x_2)$ given below. Then use multivariate NewtonRaphson method of linear approximations to find the intersection for the circle and parabola.

$$f_1(x_1, x_2) = x_1^2 + x_2^2 - 9$$

$$f_2(x_1, x_2) = x_1^2 - x_2 + 1$$

Solution: With an initial guess that is close to the true answer, it took only 4 iterations. Taking this problem a step further, we ask what happens when initial guess is way far off from the true answer. How many iterations will it take to get as close as we can to the answer? Say the initial guess is $x = 100$ and $y = 200$. It took about 10 iterations. The intersection point is at $(1.3044, 2.7016)$.

See Appendices for iteration codes (4.6 Problem 2).

5 Least Squares

"Life is complex. It has real and imaginary components."

Unknown

When the GPS receiver acquires four satellites, the Newton-Raphson Method can be used to approximate the user's position. However, what happens when more than four satellites are above the GPS user's horizon? At any one point in time, when the GPS receiver is turned on, twelve of the twenty-four satellites are available (the other twelve are on the other side of the earth). When the receiver tracks five or more satellites simultaneously, we have a situation in which we have more measurements than unknowns. That is, there could be at least five equations with only four unknowns. In such system, the system of equations is over-determined.

What can be done in this situation? One solution is to discard the extra observations, but although expedient, it is a sin for mathematicians to waste the data given. The best approach is to apply the method of Least Squares. The method of Least Squares was devised in the early 1800s by the great German mathematician and father of modern geodesy, Karl Friedrich Gauss. In this method, a unique best-fit solution (i.e, the user's position) can be obtained for the unknown parameters.

Consider a system of n equations in k unknowns that has no solution: $A\overrightarrow{x} = \overrightarrow{b}$ where A is an $n \times k$ matrix, $\overrightarrow{x} \in \Re^k$, $\overrightarrow{b} \in \Re^n$

or

$$\begin{bmatrix} \overrightarrow{a}_1 & \overrightarrow{a}_2 \cdots & \overrightarrow{a}_k \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_k \end{bmatrix} = \overrightarrow{b} \implies x_1\overrightarrow{a}_1 + x_2\overrightarrow{a}_2 + \cdots + x_k\overrightarrow{a}_k = \overrightarrow{b}.$$

We look for the vector $\overrightarrow{x}^*$ that makes the error $\|\overrightarrow{b} - A\overrightarrow{x}^*\|$ as small as possible. It so happens that the closest vector will be the projection of $\overrightarrow{b}$ onto the column space of A, $C(A)$. In other words, the least squares estimate of the data is the orthogonal projection of the data vector onto the independent variable subspace. Therefore, $A\overrightarrow{x}* = proj_{C(A)}\overrightarrow{b}$.

Definition 1. *A least squares solution to the system of equations* $A\overrightarrow{x} = \overrightarrow{b}$ *is a vector* $\overrightarrow{x}^*$ *that minimizes* $\|A\overrightarrow{x}^* - \overrightarrow{b}\|$*, using the standard inner product on* $\Re^n$*.*

A well known theorem appearing in textbooks discussing the least squares method follows.

Theorem 1. *Every least squares solution to $A\overrightarrow{x} = \overrightarrow{b}$ is an actual solution to $A^T A\overrightarrow{x}^* = A^T\overrightarrow{b}$.*

5.1 Determining User's Position with Least Squares

How does the GPS receiver apply the Least Squares method to locate user's position? The solution to a least squares problem is the vector $\overrightarrow{x}*$ given by $\overrightarrow{x}* = (A^T A)^{-1} A^T \overrightarrow{b}$. The process of recursive least squares is similar to Newton-Raphson Method. The Jocobian matrix is A and the residual is $\overrightarrow{b}$. To be consistent with notation, the solution to a least squares problem is $\overrightarrow{x}* = (J^T J)^{-1} J^T \overrightarrow{R}$. For n satellites' positions, the distance equations are set up as follows,

$$r_1 = \sqrt{(X - x_1)^2 + (Y - y_1)^2 + (Z - z_1)^2} - (\Delta T)c$$
$$r_2 = \sqrt{(X - x_2)^2 + (Y - y_2)^2 + (Z - z_2)^2} - (\Delta T)c$$
$$\cdots$$
$$r_n = \sqrt{(X - x_n)^2 + (Y - y_n)^2 + (Z - z_n)^2} - (\Delta T)c$$

where (X, Y, Z) is the user's location, r is the distance from a known satellite to the user, ΔT is the time offset, and (x_n, y_n, z_n) are satellites' positions.

The recursive least squares method works like this:

1. Set each equation equal to zero:

$$f_1 = r_1 - \sqrt{(X - x_1)^2 + (Y - y_1)^2 + (Z - z_1)^2} - (\Delta T)c = 0$$

$$f_2 = r_2 - \sqrt{(X - x_2)^2 + (Y - y_2)^2 + (Z - z_2)^2} - (\Delta T)c = 0$$

$$\ldots$$

$$f_n = r_n - \sqrt{(X - x_n)^2 + (Y - y_n)^2 + (Z - z_n)^2} - (\Delta T)c = 0$$

2. Guess user's position $U^{(k)} = (X, Y, Z, T)$ where $k = 1$ is the number of iterations.

3. Calculate the Jocobian and Residual matrix. Differentiate with respect to each variable (X, Y, Z, T) to construct the $n \times 4$ matrix $J^{(k)}$. Then evaluate the $J^{(k)}$ matrix at the initial guess.

$$J^{(k)} = \begin{bmatrix} \frac{\partial f_1}{\partial X}(U^{(k)}) & \frac{\partial f_1}{\partial Y}(U^{(k)}) & \frac{\partial f_1}{\partial Z}(U^{(k)}) & \frac{\partial f_1}{\partial T}(U^{(k)}) \\ \frac{\partial f_2}{\partial X}(U^{(k)}) & \frac{\partial f_2}{\partial Y}(U^{(k)}) & \frac{\partial f_2}{\partial Z}(U^{(k)}) & \frac{\partial f_2}{\partial T}(U^{(k)}) \\ \vdots & \vdots & \vdots & \vdots \\ \frac{\partial f_n}{\partial X}(U^{(k)}) & \frac{\partial f_n}{\partial Y}(U^{(k)}) & \frac{\partial f_n}{\partial Z}(U^{(k)}) & \frac{\partial f_n}{\partial T}(U^{(k)}) \end{bmatrix}$$

Construct $n \times 1$ matrix $\overrightarrow{R}^{(k)}$ and evaluate at the initial guess.

$$\overrightarrow{R}^{(k)} = \begin{bmatrix} f_1(U^{(k)}) \\ f_2(U^{(k)}) \\ \vdots \\ f_n(U^{(k)}) \end{bmatrix}$$

4. Calculate $\overrightarrow{x}* = (J^T J)^{-1} J^T \overrightarrow{R}$ where $\overrightarrow{x}* = (\Delta X, \Delta Y, \Delta Z, \Delta T)$.

5. Calculate the next best estimate $U^{(k+1)} = (X+\Delta X, Y+\Delta Y, Z+\Delta Z)$.

6. Replace the initial position vector $U^{(k)}$ in step 1 with the updated position vector $U^{(k+1)}$. Repeat all the steps until the difference between the previous and current answer are sufficiently small.

5.2 Exercises

Problem 1: System of Linear Equations

Given the system of linear equations below, solve for $\overrightarrow{x}*$ and $\|A\overrightarrow{x}* - \overrightarrow{b}\|$. Interpret the results.

$$2x_1 - 2x_2 = 2$$

$$x_1 - 2x_2 = 1$$

$$x_1 + x_2 = 4$$

Solution: $\overrightarrow{x}* = [\frac{71}{29}, \frac{33}{29}]$. The minimum distance $\|A\overrightarrow{x}* - \overrightarrow{b}\| = \frac{6}{\sqrt{29}}$. No other vector $\overrightarrow{x}^*$ will produce a distance between $A\overrightarrow{x}$ and $\overrightarrow{b}$ smaller than $\frac{6}{\sqrt{29}}$.

See Appendix for codes (5.2 Problem 1).

Satellite No.	x 1.0e+007	y 1.0e+007	z 1.0e+007	*Range* (m)
3	-1.14435581932368	2.18537228998174	0.92840515634504	21364414.7719640
13	0.88498653721608	1.52115049991917	1.98379922835602	21133235.1936572
16	-1.28799462471086	0.84279115293681	2.17295977908060	23653202.7278275
19	-0.62238562333828	2.55024173739922	-0.38396284978272	22023593.4330683
20	1.04260459627803	2.18281560737286	-1.10756652807472	23223303.5998171
23	0.15114376130666	2.36981504953570	1.16498729017268	20094101.9563987
27	1.98311575365209	0.65606228041700	1.72062794938024	23840149.6080318

Figure 20: Data from International Journal of Applied Engineering Research.

Problem 2: System of Non-Linear Equations

A typical GPS receiver receives the following seven satellite positions given below (Figure 20). The data given had time adjusted so the time offset is zero. Use the recursive least squares method to calculate user's position.

Solution: The initial guess was (91750,56995,26919). It took 5 iterations to get the user's position. The user's position is at $x = 917,589$, $y = 5,699,510$, and $z = 2,691,980$.

See Appendix for iteration codes (5.2 Problem 2).

Appendices

3.2 Example 1

Example

```
f1[x1_, x2_] := (x1 - 0)^2 + (x2 - 2)^2 - 16
f2[x1_, x2_] := (x1 + 2)^2 + (x2 + 1)^2 - 4
f3[x1_, x2_] := (x1 - 3)^2 + (x2 - 0)^2 - 9
```

```
f1[x1_, x2_] := (x1 - 1)^2 + (x2 - 2)^2 - 16
f2[x1_, x2_] := (x1 + 1)^2 + (x2 + 2)^2 - 4
f3[x1_, x2_] := (x1 - 4)^2 + (x2 + 2)^2 - 9
```

```
f1sol = Solve[f1[x1, x2] == 0, x2]
f2sol = Solve[f2[x1, x2] == 0, x2]
f3sol = Solve[f3[x1, x2] == 0, x2]
```

```
f1a = x2 /. f1sol[[1]]
f1b = x2 /. f1sol[[2]]
f2a = x2 /. f2sol[[1]]
f2b = x2 /. f2sol[[2]]
f3a = x2 /. f3sol[[1]]
f3b = x2 /. f3sol[[2]]
```

$\left\{\left\{x2 \to 2 - \sqrt{15 + 2\,x1 - x1^2}\right\}, \left\{x2 \to 2 + \sqrt{15 + 2\,x1 - x1^2}\right\}\right\}$

$\left\{\left\{x2 \to -2 - \sqrt{3 - 2\,x1 - x1^2}\right\}, \left\{x2 \to -2 + \sqrt{3 - 2\,x1 - x1^2}\right\}\right\}$

$\left\{\left\{x2 \to -2 - \sqrt{-7 + 8\,x1 - x1^2}\right\}, \left\{x2 \to -2 + \sqrt{-7 + 8\,x1 - x1^2}\right\}\right\}$

$2 - \sqrt{15 + 2\,x1 - x1^2}$

$2 + \sqrt{15 + 2\,x1 - x1^2}$

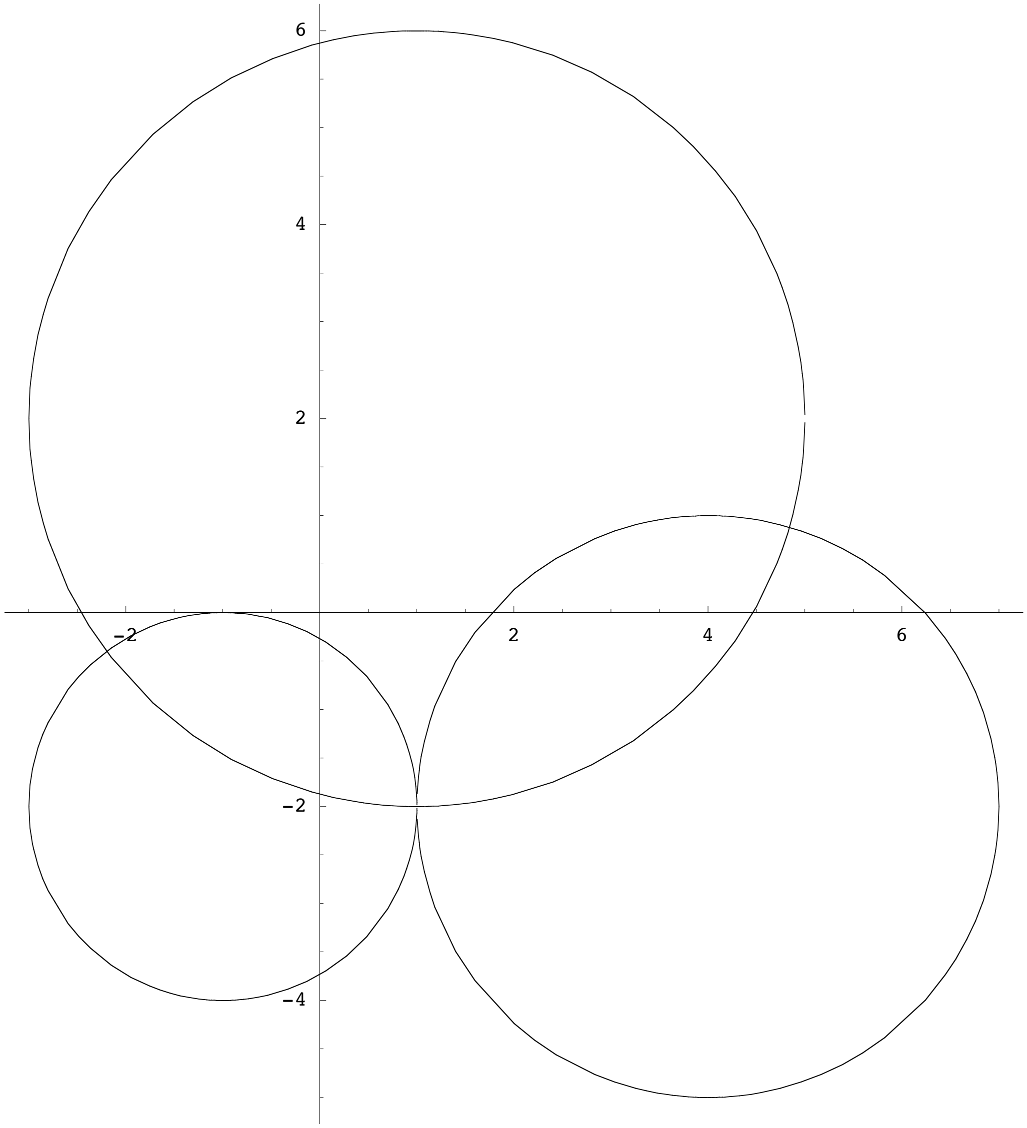
$-2 - \sqrt{3 - 2\,x1 - x1^2}$

$-2 + \sqrt{3 - 2\,x1 - x1^2}$

$-2 - \sqrt{-7 + 8\,x1 - x1^2}$

$-2 + \sqrt{-7 + 8\,x1 - x1^2}$

```
Off[Plot::plnr];
Plot[{f1a, f1b, f2a, f2b, f3a, f3b}, {x1, -3, 7}, AspectRatio → Automatic];
```



```
J = ( ∂x1 f1[x1, x2]  ∂x2 f1[x1, x2] );
    ( ∂x1 f2[x1, x2]  ∂x2 f2[x1, x2] )
% // MatrixForm
R = ( f1[x1, x2] );
    ( f2[x1, x2] )
% // MatrixForm
δx = ( x1n );
     ( x2n )
% // MatrixForm
```

$$\begin{pmatrix} 2\,(-1+x1) & 2\,(-2+x2) \\ 2\,(1+x1) & 2\,(2+x2) \end{pmatrix}$$

$$\begin{pmatrix} -16+(-1+x1)^2+(-2+x2)^2 \\ -4+(1+x1)^2+(2+x2)^2 \end{pmatrix}$$

$$\begin{pmatrix} x1n \\ x2n \end{pmatrix}$$

```
J /. {x1 → 1, x2 → 2};
% // MatrixForm
R /. {x1 → 1, x2 → 2};
% // MatrixForm
```

$$\begin{pmatrix} 0 & 0 \\ 4 & 8 \end{pmatrix}$$

$$\begin{pmatrix} -16 \\ 16 \end{pmatrix}$$

```
MySol = Solve[J.δx == -R, {x1n, x2n}][[1]]
```

$$\left\{x1n \to -\frac{-5+x1^2-3\,x2-x1\,x2-x2^2}{2\,x1-x2},\ x2n \to -\frac{5+6\,x1+x1^2+4\,x1\,x2-x2^2}{2\,(2\,x1-x2)}\right\}$$

```
x1L = x1n /. MySol[[1]]
x2L = x2n /. MySol[[2]]
```

$$-\frac{-5+x1^2-3\,x2-x1\,x2-x2^2}{2\,x1-x2}$$

$$-\frac{5+6\,x1+x1^2+4\,x1\,x2-x2^2}{2\,(2\,x1-x2)}$$

```
IterationResults = Table[0, {7}, {3}];
```

```
x1i = 2;
x2i = 3;
Do[
 IterationResults[[i, 1]] = i;
 IterationResults[[i, 2]] = x1i;
 IterationResults[[i, 3]] = x2i;
 x1i = x1i + x1L /. {x1 → x1i, x2 → x2i};
 x2i = x2i + x2L /. {x1 → x1i, x2 → x2i}
 , {i, 7}]
```

```
IterationResults // TableForm // N
```

1.	2.	3.
2.	27.	-8.87255
3.	12.5031	-4.64995
4.	5.69866	-2.90668
5.	2.60091	-2.20873
6.	1.35173	-2.01703
7.	1.02627	-2.00012

```
x1i = 2;
x2i = 3;
i = 1
IterationResults = Table[0, {20}, {3}];
While[Abs[(x1L /. {x1 → x1i, x2 → x2i})] > 0.0001,
 IterationResults[[i, 1]] = i;
 IterationResults[[i, 2]] = x1i;
 IterationResults[[i, 3]] = x2i;
 x1i = x1i + x1L /. {x1 → x1i, x2 → x2i};
 x2i = x2i + x2L /. {x1 → x1i, x2 → x2i};
 i = i + 1
]
```

```
IterationResults // TableForm // N
```

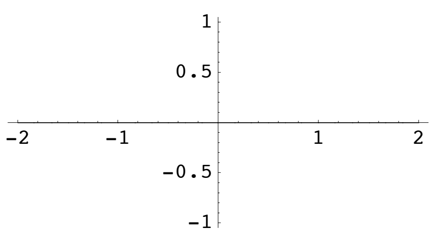
1

1.	2.	3.
2.	27.	-8.87255
3.	12.5031	-4.64995
4.	5.69866	-2.90668
5.	2.60091	-2.20873
6.	1.35173	-2.01703
7.	1.02627	-2.00012
8.	1.00017	-2.
0.	0.	0.
0.	0.	0.
0.	0.	0.
0.	0.	0.
0.	0.	0.
0.	0.	0.
0.	0.	0.
0.	0.	0.
0.	0.	0.
0.	0.	0.
0.	0.	0.
0.	0.	0.

```
(x1L /. {x1 → x1i, x2 → x2i}) // N
```

-7.24783×10^{-9}

```
In[1]:= (*4.5 Example 2*)

In[1]:= f1[x1_, x2_] := x1^3 - 20
        f2[x1_, x2_] := x2

In[3]:= f1sol = Solve[f1[x1, x2] == 0, x2]
        f2sol = Solve[f2[x1, x2] == 0, x2]
        f1a = x2 /. f1sol[[1]]
        (*f1b=x2/.f1sol[[2]]*)
        f2a = x2 /. f2sol[[1]]
        (*f2b=x2/.f2sol[[2]]*)
        Off[Plot::plnr];
        Plot[{f1a, f2a}, {x1, -2, 2}, AspectRatio → Automatic];

Out[3]= {{}}

Out[4]= {{x2 → 0}}

Out[5]= x2

Out[6]= 0
```



In[9]:= $J = \begin{pmatrix} \partial_{x1}\ \text{f1[x1, x2]} & \partial_{x2}\ \text{f1[x1, x2]} \\ \partial_{x1}\ \text{f2[x1, x2]} & \partial_{x2}\ \text{f2[x1, x2]} \end{pmatrix};$

`% // MatrixForm`

$R = \begin{pmatrix} \text{f1[x1, x2]} \\ \text{f2[x1, x2]} \end{pmatrix};$

`% // MatrixForm`

$\delta x = \begin{pmatrix} \text{x1n} \\ \text{x2n} \end{pmatrix};$

`% // MatrixForm`

Out[10]//MatrixForm=

$$\begin{pmatrix} 3\,x1^2 & 0 \\ 0 & 1 \end{pmatrix}$$

Out[12]//MatrixForm=

$$\begin{pmatrix} -20 + x1^3 \\ x2 \end{pmatrix}$$

Out[14]//MatrixForm=

$$\begin{pmatrix} \text{x1n} \\ \text{x2n} \end{pmatrix}$$

```
In[15]:= J /. {x1 → 2, x2 → 2};
         % // MatrixForm
         R /. {x1 → 2, x2 → 2};
         % // MatrixForm
```

Out[16]//MatrixForm=

$$\begin{pmatrix} 12 & 0 \\ 0 & 1 \end{pmatrix}$$

Out[18]//MatrixForm=

$$\begin{pmatrix} -12 \\ 2 \end{pmatrix}$$

```
In[19]:= Solve[J.δx == -R, {x1n, x2n}]
```

Out[19]= $\left\{\left\{x1n \to \frac{20 - x1^3}{3\,x1^2},\ x2n \to -x2\right\}\right\}$

```
In[20]:= MySol = Solve[J.δx == -R, {x1n, x2n}][[1]]
         x1L = x1n /. MySol[[1]]
         x2L = x2n /. MySol[[2]]
```

Out[20]= $\left\{x1n \to \frac{20 - x1^3}{3\,x1^2},\ x2n \to -x2\right\}$

Out[21]= $\frac{20 - x1^3}{3\,x1^2}$

Out[22]= $-x2$

```
In[23]:= IterationResults = Table[0, {5}, {3}];
```

In[62]:=
```
x1i = 10;
x2i = 0;
i = 1
IterationResults = Table[0, {20}, {3}];
While[Abs[(x1L /. {x1 → x1i, x2 → x2i})] > 0.0001,
 IterationResults[[i, 1]] = i;
 IterationResults[[i, 2]] = x1i;
 IterationResults[[i, 3]] = x2i;
 x1i = x1i + x1L /. {x1 → x1i, x2 → x2i};
 x2i = x2i + x2L /. {x1 → x1i, x2 → x2i};
 i = i + 1
]

IterationResults[[i, 1]] = i;
IterationResults[[i, 2]] = x1i;
IterationResults[[i, 3]] = x2i;
IterationResults // TableForm // N
```

Out[64]= 1

Out[70]//TableForm=

1.	10.	0.
2.	6.73333	0.
3.	4.63593	0.
4.	3.40082	0.
5.	2.84364	0.
6.	2.7202	0.
7.	2.71443	0.
0.	0.	0.
0.	0.	0.
0.	0.	0.
0.	0.	0.
0.	0.	0.
0.	0.	0.
0.	0.	0.
0.	0.	0.
0.	0.	0.
0.	0.	0.
0.	0.	0.
0.	0.	0.
0.	0.	0.

In[52]:=
```
2.7202 - 2.7144299016428564`
```

Out[52]= 0.0057701

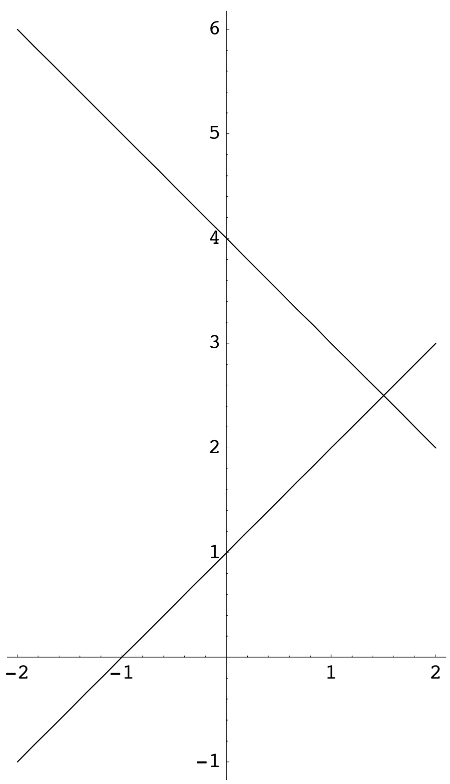
```
(*4.5 Example 3*)

f1[x1_, x2_] := x1 + x2 - 4
f2[x1_, x2_] := x1 - x2 + 1

f1sol = Solve[f1[x1, x2] == 0, x2]
f2sol = Solve[f2[x1, x2] == 0, x2]
f1a = x2 /. f1sol[[1]]
(*f1b=x2/.f1sol[[2]]*)
f2a = x2 /. f2sol[[1]]
(*f2b=x2/.f2sol[[2]]*)
Off[Plot::plnr];
Plot[{f1a, f2a}, {x1, -2, 2}, AspectRatio → Automatic];
```

{{x2 → 4 - x1}}

{{x2 → 1 + x1}}

4 - x1

1 + x1



```
J = ( ∂x1 f1[x1, x2]  ∂x2 f1[x1, x2]
      ∂x1 f2[x1, x2]  ∂x2 f2[x1, x2] );
% // MatrixForm
R = ( f1[x1, x2]
      f2[x1, x2] );
% // MatrixForm
δx = ( x1n
       x2n );
% // MatrixForm
```

$$\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$$

$$\begin{pmatrix} -4 + x1 + x2 \\ 1 + x1 - x2 \end{pmatrix}$$

$$\begin{pmatrix} x1n \\ x2n \end{pmatrix}$$

```
J /. {x1 → 2, x2 → 2};
% // MatrixForm
R /. {x1 → 2, x2 → 2};
% // MatrixForm
```

$$\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$$

$$\begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

```
Solve[J.δx == -R, {x1n, x2n}]
```

$$\left\{\left\{x1n \to \frac{3}{2} - x1,\ x2n \to \frac{5}{2} - x2\right\}\right\}$$

```
MySol = Solve[J.δx == -R, {x1n, x2n}][[1]]
x1L = x1n /. MySol[[1]]
x2L = x2n /. MySol[[2]]
```

$$\left\{x1n \to \frac{3}{2} - x1,\ x2n \to \frac{5}{2} - x2\right\}$$

$$\frac{3}{2} - x1$$

$$\frac{5}{2} - x2$$

```
IterationResults = Table[0, {5}, {3}];
```

```
x1i = -1;
x2i = -2;
Do[
 IterationResults[[i, 1]] = i;
 IterationResults[[i, 2]] = x1i;
 IterationResults[[i, 3]] = x2i;
 x1i = x1i + x1L /. {x1 → x1i, x2 → x2i};
 x2i = x2i + x2L /. {x1 → x1i, x2 → x2i}
 , {i, 5}]
```

```
IterationResults // TableForm // N
```

1.	-1.	-2.
2.	1.5	2.5
3.	1.5	2.5
4.	1.5	2.5
5.	1.5	2.5

```
x1i = 1000;
x2i = -3000;
i = 1
IterationResults = Table[0, {20}, {3}];
While[Abs[(x1L /. {x1 → x1i, x2 → x2i})] > 0.0001,
 IterationResults[[i, 1]] = i;
 IterationResults[[i, 2]] = x1i;
 IterationResults[[i, 3]] = x2i;
 x1i = x1i + x1L /. {x1 → x1i, x2 → x2i};
 x2i = x2i + x2L /. {x1 → x1i, x2 → x2i};
 i = i + 1
]

IterationResults[[i, 1]] = i;
IterationResults[[i, 2]] = x1i;
IterationResults[[i, 3]] = x2i;
IterationResults // TableForm // N
```

```
1
```

1.	1000.	-3000.
2.	1.5	2.5
0.	0.	0.
0.	0.	0.
0.	0.	0.
0.	0.	0.
0.	0.	0.
0.	0.	0.
0.	0.	0.
0.	0.	0.
0.	0.	0.
0.	0.	0.
0.	0.	0.
0.	0.	0.
0.	0.	0.
0.	0.	0.
0.	0.	0.
0.	0.	0.
0.	0.	0.
0.	0.	0.

```
x1i = 1;
x2i = 3;
(x1L /. {x1 → x1i, x2 → x2i}) // N
x1i = x1i + x1L /. {x1 → x1i, x2 → x2i} // N
x2i = x2i + x2L /. {x1 → x1i, x2 → x2i} // N
```

```
0.5

1.5

2.5
```

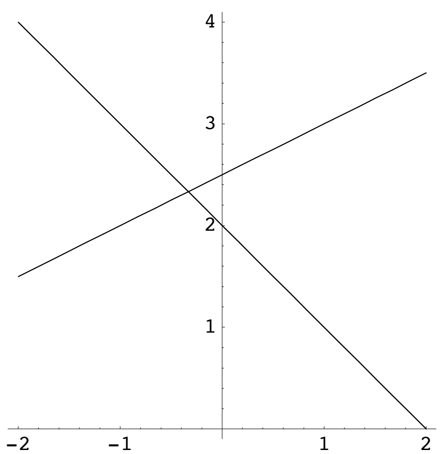
```
(*4.6 Problem 1*)

f1[x1_, x2_] := 2 x1 + 2 x2 - 4
f2[x1_, x2_] := x1 - 2 x2 + 5

f1sol = Solve[f1[x1, x2] == 0, x2]
f2sol = Solve[f2[x1, x2] == 0, x2]
f1a = x2 /. f1sol[[1]]
(*f1b=x2/.f1sol[[2]]*)
f2a = x2 /. f2sol[[1]]
(*f2b=x2/.f2sol[[2]]*)
Off[Plot::plnr];
Plot[{f1a, f2a}, {x1, -2, 2}, AspectRatio → Automatic];
```

$\{\{x2 \to 2 - x1\}\}$

$\left\{\left\{x2 \to \frac{5 + x1}{2}\right\}\right\}$

$2 - x1$

$\frac{5 + x1}{2}$



$J = \begin{pmatrix} \partial_{x1}\ f1[x1,\ x2] & \partial_{x2}\ f1[x1,\ x2] \\ \partial_{x1}\ f2[x1,\ x2] & \partial_{x2}\ f2[x1,\ x2] \end{pmatrix};$

`% // MatrixForm`

$R = \begin{pmatrix} f1[x1,\ x2] \\ f2[x1,\ x2] \end{pmatrix};$

`% // MatrixForm`

$\delta x = \begin{pmatrix} x1n \\ x2n \end{pmatrix};$

`% // MatrixForm`

$\begin{pmatrix} 2 & 2 \\ 1 & -2 \end{pmatrix}$

$\begin{pmatrix} -4 + 2\ x1 + 2\ x2 \\ 5 + x1 - 2\ x2 \end{pmatrix}$

$\begin{pmatrix} x1n \\ x2n \end{pmatrix}$

```
J /. {x1 → 2, x2 → 2};
% // MatrixForm
R /. {x1 → 2, x2 → 2};
% // MatrixForm
```

$$\begin{pmatrix} 2 & 2 \\ 1 & -2 \end{pmatrix}$$

$$\begin{pmatrix} 4 \\ 3 \end{pmatrix}$$

```
Solve[J.δx == -R, {x1n, x2n}]
```

$$\left\{\left\{x1n \to -\frac{1}{3} - x1,\ x2n \to \frac{7}{3} - x2\right\}\right\}$$

```
MySol = Solve[J.δx == -R, {x1n, x2n}][[1]]
x1L = x1n /. MySol[[1]]
x2L = x2n /. MySol[[2]]
```

$$\left\{x1n \to -\frac{1}{3} - x1,\ x2n \to \frac{7}{3} - x2\right\}$$

$$-\frac{1}{3} - x1$$

$$\frac{7}{3} - x2$$

```
IterationResults = Table[0, {5}, {3}];
```

```
x1i = -1;
x2i = -2;
Do[
 IterationResults[[i, 1]] = i;
 IterationResults[[i, 2]] = x1i;
 IterationResults[[i, 3]] = x2i;
 x1i = x1i + x1L /. {x1 → x1i, x2 → x2i};
 x2i = x2i + x2L /. {x1 → x1i, x2 → x2i}
 , {i, 5}]
```

```
IterationResults // TableForm // N
```

1.	-1.	-2.
2.	-0.333333	2.33333
3.	-0.333333	2.33333
4.	-0.333333	2.33333
5.	-0.333333	2.33333

```
x1i = 1000;
x2i = -3000;
i = 1
IterationResults = Table[0, {20}, {3}];
While[Abs[(x1L /. {x1 → x1i, x2 → x2i})] > 0.0001,
 IterationResults[[i, 1]] = i;
 IterationResults[[i, 2]] = x1i;
 IterationResults[[i, 3]] = x2i;
 x1i = x1i + x1L /. {x1 → x1i, x2 → x2i};
 x2i = x2i + x2L /. {x1 → x1i, x2 → x2i};
 i = i + 1
]

IterationResults[[i, 1]] = i;
IterationResults[[i, 2]] = x1i;
IterationResults[[i, 3]] = x2i;
IterationResults // TableForm // N
```

1

1.	1000.	-3000.
2.	-0.333333	2.33333
0.	0.	0.
0.	0.	0.
0.	0.	0.
0.	0.	0.
0.	0.	0.
0.	0.	0.
0.	0.	0.
0.	0.	0.
0.	0.	0.
0.	0.	0.
0.	0.	0.
0.	0.	0.
0.	0.	0.
0.	0.	0.
0.	0.	0.
0.	0.	0.
0.	0.	0.
0.	0.	0.

```
x1i = 1;
x2i = 3;
(x1L /. {x1 → x1i, x2 → x2i}) // N
x1i = x1i + x1L /. {x1 → x1i, x2 → x2i} // N
x2i = x2i + x2L /. {x1 → x1i, x2 → x2i} // N
```

-1.33333

-0.333333

2.33333

4.6 Problem 2

Example

```
f1[x1_, x2_] := x1^2 + x2^2 - 9
f2[x1_, x2_] := x1^2 - x2 + 1

f1sol = Solve[f1[x1, x2] == 0, x2]
f2sol = Solve[f2[x1, x2] == 0, x2]
f1a = x2 /. f1sol[[1]]
f1b = x2 /. f1sol[[2]]
f2a = x2 /. f2sol[[1]]
(*f2b=x2/.f2sol[[2]]*)
Off[Plot::plnr];
Plot[{f1a, f1b, f2a}, {x1, -3, 3}, AspectRatio → Automatic];
```

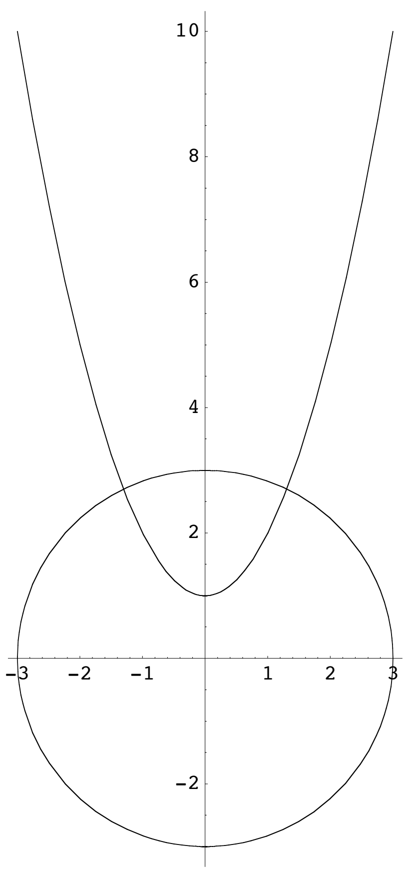
$\{\{x2 \to -\sqrt{9 - x1^2}\}, \{x2 \to \overline{9 - x1^2}\}\}$

$\{\{x2 \to 1 + x1^2\}\}$

$- \overline{9 - x1^2}$

$\overline{9 - x1^2}$

$1 + x1^2$



```
J = ( ∂x1 f1[x1, x2]  ∂x2 f1[x1, x2] );
    ( ∂x1 f2[x1, x2]  ∂x2 f2[x1, x2] )
% // MatrixForm
R = ( f1[x1, x2] );
    ( f2[x1, x2] )
% // MatrixForm
δx = ( x1n );
     ( x2n )
% // MatrixForm
```

$$\begin{pmatrix} 2\,x1 & 2\,x2 \\ 2\,x1 & -1 \end{pmatrix}$$

$$\begin{pmatrix} -9 + x1^2 + x2^2 \\ 1 + x1^2 - x2 \end{pmatrix}$$

$$\begin{pmatrix} x1n \\ x2n \end{pmatrix}$$

```
J /. {x1 → 1, x2 → 2};
% // MatrixForm
R /. {x1 → 1, x2 → 2};
% // MatrixForm
```

$$\begin{pmatrix} 2 & 4 \\ 2 & -1 \end{pmatrix}$$

$$\begin{pmatrix} -4 \\ 0 \end{pmatrix}$$

```
Solve[J.δx == -R, {x1n, x2n}]
```

$$\left\{\left\{x1n \to -\frac{-9 + x1^2 + 2\,x2 + 2\,x1^2\,x2 - x2^2}{2\,x1\,(1 + 2\,x2)},\ x2n \to -\frac{-10 + x2 + x2^2}{1 + 2\,x2}\right\}\right\}$$

```
MySol = Solve[J.δx == -R, {x1n, x2n}][[1]]
x1L = x1n /. MySol[[1]]
x2L = x2n /. MySol[[2]]
```

$$\left\{x1n \to -\frac{-9 + x1^2 + 2\,x2 + 2\,x1^2\,x2 - x2^2}{2\,x1\,(1 + 2\,x2)},\ x2n \to -\frac{-10 + x2 + x2^2}{1 + 2\,x2}\right\}$$

$$-\frac{-9 + x1^2 + 2\,x2 + 2\,x1^2\,x2 - x2^2}{2\,x1\,(1 + 2\,x2)}$$

$$-\frac{-10 + x2 + x2^2}{1 + 2\,x2}$$

```
IterationResults = Table[0, {5}, {3}];
```

```
x1i = 1;
x2i = 2;
Do[
 IterationResults[[i, 1]] = i;
 IterationResults[[i, 2]] = x1i;
 IterationResults[[i, 3]] = x2i;
 x1i = x1i + x1L /. {x1 → x1i, x2 → x2i};
 x2i = x2i + x2L /. {x1 → x1i, x2 → x2i}
 , {i, 5}]

IterationResults // TableForm // N
```

1.	1.	2.
2.	1.4	2.8
3.	1.30823	2.70303
4.	1.30444	2.70156
5.	1.30444	2.70156

```
x1i = 100;
x2i = 200;
i = 1
IterationResults = Table[0, {20}, {3}];
While[Abs[(x1L /. {x1 → x1i, x2 → x2i})] > 0.0001,
 IterationResults[[i, 1]] = i;
 IterationResults[[i, 2]] = x1i;
 IterationResults[[i, 3]] = x2i;
 x1i = x1i + x1L /. {x1 → x1i, x2 → x2i};
 x2i = x2i + x2L /. {x1 → x1i, x2 → x2i};
 i = i + 1
]

IterationResults // TableForm // N
```

1

1.	100.	200.
2.	50.4939	99.7756
3.	25.7291	49.6889
4.	13.325	24.6966
5.	7.08659	12.3017
6.	3.91733	6.30118
7.	2.29743	3.65414
8.	1.5428	2.81078
9.	1.32344	2.70336
10.	1.30458	2.70156
0.	0.	0.
0.	0.	0.
0.	0.	0.
0.	0.	0.
0.	0.	0.
0.	0.	0.
0.	0.	0.
0.	0.	0.
0.	0.	0.
0.	0.	0.

```
(x1L /. {x1 → x1i, x2 → x2i}) // N
```

-7.14558×10^{-9}

5.2 Problem 1

```
A = (2  -2
     1  -2
     1   1);

b = (2
     1
     4);

At = Transpose[A];
LS = Inverse[At.A].At.b;
% // MatrixForm
Null
```

$$\begin{pmatrix} \frac{71}{29} \\ \frac{33}{29} \end{pmatrix}$$

```
MD = Norm[A.LS - b];
% // MatrixForm
```

$$\frac{6}{\sqrt{29}}$$

5.2 Problem 2

```
Off[General::spell1]

sx1 = -1.14435581932368 * 10^7;
sy1 = 2.18537228998174 * 10^7;
sz1 = 0.92840515634504 * 10^7;

sx2 = 0.88498653721608 * 10^7;
sy2 = 1.52115049991917 * 10^7;
sz2 = 1.98379922835602 * 10^7;

sx3 = -1.28799462471086 * 10^7;
sy3 = 0.84279115293681 * 10^7;
sz3 = 2.17295977908060 * 10^7;

sx4 = -0.62238562333828 * 10^7;
sy4 = 2.55024173739922 * 10^7;
sz4 = -0.38396284978272 * 10^7;

sx5 = 1.04260459627803 * 10^7;
sy5 = 2.18281560737286 * 10^7;
sz5 = -1.10756652807472 * 10^7;

sx6 = 0.15114376130666 * 10^7;
sy6 = 2.36981504953570 * 10^7;
sz6 = 1.16498729017268 * 10^7;

sx7 = 1.98311575365209 * 10^7;
sy7 = 0.65606228041700 * 10^7;
sz7 = 1.72062794938024 * 10^7;

r1 = 21364414.7719640;
r2 = 21133235.1936572;
r3 = 23653202.7278275;
r4 = 22023593.4330683;
r5 = 23223303.5998171;
r6 = 20094101.9563987;
r7 = 23840149.6080318;
```

$$f1[x1\_, x2\_, x3\_] := \sqrt{(x1 - sx1)^2 + (x2 - sy1)^2 + (x3 - sz1)^2}$$

$$f2[x1\_, x2\_, x3\_] := \sqrt{(x1 - sx2)^2 + (x2 - sy2)^2 + (x3 - sz2)^2}$$

$$f3[x1\_, x2\_, x3\_] := \sqrt{(x1 - sx3)^2 + (x2 - sy3)^2 + (x3 - sz3)^2}$$

$$f4[x1\_, x2\_, x3\_] := \sqrt{(x1 - sx4)^2 + (x2 - sy4)^2 + (x3 - sz4)^2}$$

$$f5[x1\_, x2\_, x3\_] := \sqrt{(x1 - sx5)^2 + (x2 - sy5)^2 + (x3 - sz5)^2}$$

$$f6[x1\_, x2\_, x3\_] := \sqrt{(x1 - sx6)^2 + (x2 - sy6)^2 + (x3 - sz6)^2}$$

$$f7[x1\_, x2\_, x3\_] := \sqrt{(x1 - sx7)^2 + (x2 - sy7)^2 + (x3 - sz7)^2}$$

```
Eq1 = f1[x, y, z];
Eq2 = f2[x, y, z];
Eq3 = f3[x, y, z];
```

```
Eq4 = f4[x, y, z];
Eq5 = f5[x, y, z];
Eq6 = f6[x, y, z];
Eq7 = f7[x, y, z];

Ao =
  {{∂x Eq1, ∂y Eq1, ∂z Eq1},
   {∂x Eq2, ∂y Eq2, ∂z Eq2},
   {∂x Eq3, ∂y Eq3, ∂z Eq3},
   {∂x Eq4, ∂y Eq4, ∂z Eq4},
   {∂x Eq5, ∂y Eq5, ∂z Eq5},
   {∂x Eq6, ∂y Eq6, ∂z Eq6},
   {∂x Eq7, ∂y Eq7, ∂z Eq7}
  };
bo =
  {{r1 - Eq1},
   {r2 - Eq2},
   {r3 - Eq3},
   {r4 - Eq4},
   {r5 - Eq5},
   {r6 - Eq6},
   {r7 - Eq7}
  };

x0 = 91750;
y0 = 5.699509248624045*^4;
z0 = 2.6919786033788896*^4;

x0 = Table[x0, {5}];
y0 = Table[y0, {5}];
z0 = Table[z0, {5}];
For[i = 1, i < 5,
 A = Ao /. {x → x0[[i]], y → y0[[i]], z → z0[[i]]};
 At = Transpose[A];
 b = bo /. {x → x0[[i]], y → y0[[i]], z → z0[[i]]};
 LS = Inverse[At.A].At.b;
 x0[[i + 1]] = (x0 [[i]] + LS[[1, 1]]);
 y0[[i + 1]] = (y0 [[i]] + LS[[2, 1]]);
 z0[[i + 1]] = (z0 [[i]] + LS[[3, 1]]);
 i ++]
x0
y0
z0
```

$$\frac{x0[[5]] - 918074}{918074} * 100$$

$$\frac{y0[[5]] - 5703774}{5703774} * 100$$

$$\frac{z0[[5]] - 2693919}{2693919} * 100$$

{91750, 775634., 922341., 922166., 922166.}

$\{56995.1,\ 5.3613\times10^{6},\ 5.71941\times10^{6},\ 5.72225\times10^{6},\ 5.72225\times10^{6}\}$

$\{26919.8,\ 2.50087\times10^{6},\ 2.70227\times10^{6},\ 2.70323\times10^{6},\ 2.70323\times10^{6}\}$

```
0.445673

0.323896

0.34545
```

References

[1] Kalman, D. *An Undertermined Linear System for GPS*, The Mathematical Association of America, Vol. 33, No.5, 2002.

[2] Langley, R. *The Mathematics of GPS*, University of New Brunswick, GPS World, 1991.

[3] Sadun, L. *Applied Linear Algebra, The Decoupling Principle 2nd Edition*, American Mathematical Society, 2008.

[4] The Department of Energy Mines and Resources (EMR) *GPS Positiong Guide*, Natural Resources Canada, Ottawa, Ontario, 1994.

[5] Grewal, M. *Global Positioning Systems, Inertial Navigation, and Integration*, John Wiley Sons, Inc., Canada, 2001.

[6] El-Rabbany, A. *Introduction to GPS, The Global Positioning System*, Artech House, Inc., 2002.

[7] Kaplan, E. *Understanding GPS: Principles and Applications*, Artech House, Boston, 1996.

[8] KUMAR, B. *Determination of GPS receiver position using Multivariate Newton-Raphson Technique for over specified cases*, International Journal of Applied Engineering Research, Volume 3, Number 11, pp. 1457-1460, 2008.

[9] STROM, R. *Charting a Course Toward Global Navigation*, The Aerospace Corporation, Crosslink, pp. 6-11, 2002.

[10] MASSATT, P. *Optimizing Performances through Constellation Management*, The Aerospace Corporation, Crosslink, pp. 17-21, 2002.

[11] FAUSETT, L. *Numerical Methods: Algorithms and Applications*, Prentice Hall, New Jersey, pp. 241-254, 2003.

[12] KEFFER, D. *ChE 301 Lecture Notes*, pp. 1-10, 1998.

[13] http://www.globalsecurity.org/space/systems/timation.htm

[14] http://www.math.tamu.edu/ dallen/physics/gps/gps.htm