Suggested activities:

1. Solve the system (4.37) by using a multivariate root finder in Python. There are multiple options in scipy optimize, including root, fsolve, and newton_krylov. Find the receiver position (x, y, z) near earth and time correction d for known, simultaneous satellite positions

$$(15\,600,\ 7540,\ 20\,140),\ (18\,760,\ 2750,\ 18\,610),$$
$$(17\,610,\ 14\,630,\ 13\,480),\ (19\,170,\ 610,\ 18\,390)$$

in km, and measured time intervals 0.07074, 0.07220, 0.07690, 0.07242 in seconds, respectively. Set the initial vector to be (x0, y0, z0, d0) = (0, 0, 6370, 0). As a check, the answers are approximately

$$(x, y, z) = (-41.77271,\ -16.78919,\ 6370.0596)$$

and $d = -3.201566 \times 10^{-3}$ seconds.

2. Write a Python program to carry out the solution via the quadratic formula. Hint: Subtracting the last three equations of (4.37) from the first yields three linear equations in the four unknowns $x\,\overline{u}_x + y\,\overline{u}_y + z\,\overline{u}_z + d\,\overline{u}_d + \overline{w} = 0$, expressed in vector form. A formula for x in terms of d can be obtained from

$$0 = \det\left[\overline{u}_y \,\middle|\, \overline{u}_z \,\middle|\, x\,\overline{u}_x + y\,\overline{u}_y + z\,\overline{u}_z + d\,\overline{u}_d + \overline{w}\right]$$

noting that the determinant is linear in its columns and that a matrix with a repeated column has determinant zero. Similarly, we can arrive at formulas for y and z, respectively, in terms of d, that can be substituted in the first quadratic equation of (4.37), to make it an equation in one variable.

3. Skip

4. Now set up a test of the conditioning of the GPS problem. Define satellite positions (A_i, B_i, C_i) from spherical coordinates (ρ, ϕ_i, θ_i) as

$$A_i = \rho\,\mathrm{Cos}(\phi_i)\,\mathrm{Cos}(\theta_i)$$
$$B_i = \rho\,\mathrm{Cos}(\phi_i)\,\mathrm{Sin}(\theta_i)$$
$$C_i = \rho\,\mathrm{Sin}(\phi_i)$$

where $\rho = 26\,570$ km is fixed, while $0 \le \phi_i \le \pi/2$ and $0 \le \theta_i \le 2\pi$ for $i = 1, \ldots, 4$ are chosen arbitrarily. The ϕ coordinate is restricted so that the four satellites are in the upper hemisphere. Set $x = 0$, $y = 0$, $z = 6370$, $d = 0.0001$, and calculate the corresponding satellite ranges

$$R_i = \sqrt{A_i^2 + B_i^2 + (C_i - 6370)^2}$$

and travel times $t_i = d + R_i / c$.

We will define an error magnification factor specially tailored to the situation. The atomic clocks aboard the satellites are correct up to about 10 nanoseconds, or 10^{-8} second. Therefore, it is important to study the effect of changes in the transmission time of this magnitude. Let the backward, or input error be the input change in meters. At the speed of light, $\Delta\ t_i = 10^{-8}$ second corresponds to $10^{-8}\ s \approx 3$ meters. Let the forward, or output error be the change in position $\|\Delta x,\ \Delta y,\ \Delta z\|_\infty$, caused by such a change in t_i , also in meters. Then we can define the dimensionless error magnification factor (EMF)

$$\text{EMF} = \frac{\|\Delta x, \Delta y, \Delta z\|_\infty}{c\,\|\Delta t_1, \ldots, \Delta t_m\|_\infty}$$

and the condition number of the problem to be the maximum error magnification factor for all small $\Delta\ t_i$ (say, 10^{-8} or less). Change each t_i defined in the foregoing by $\Delta t_i = \pm 10^{-8}$, not all the same. Denote the new solution of the equations (4.37) by $(x,\ y,\ z,\ d)$ and compute the difference in position $\|\Delta x, \Delta y, \Delta z\|_\infty$ and the error magnification factor. Try different variations of the $\Delta t_i\,'\,s$. What is the maximum position error found, in meters? Estimate the condition number of the problem, on the basis of the error magnification factors you have computed.

5. Now repeat Step 4 with a more tightly grouped set of satellites. Choose all ϕ_i within 5 percent of one another and all θ_i within 5 percent of one another. Solve with and without the same input error as in Step 4. Find the maximum position error and error magnification factor. Compare the conditioning of the GPS problem when the satellites are tightly or loosely bunched.