Trilateration and Bilateration In 3D and 2D Space Using Active Tags

Article · October 2017

2 authors:

Peyman Askari
Prilyx Research and Development Ltd.

Fardin Barekat
Prilyx Research and Development Ltd.

PRILYX RESEARCH AND DEVELOPMENT LTD.

Trilateration and Bilateration In 3D and 2D Space Using Active Tags

Peyman Askari
Fardin Barekat

October 18, 2017

CONTENTS

1. INTRODUCTION

Given one known reference point, or anchor, in three dimensional space, and a single point, or tag, with a known distance to this anchor, it is possible to determine the sphere upon which this tag resides through a process here referred to as Unilateration. The problem with Unilateration is that it does not give exact coordinates in three dimensional space. The benefit of Unilateration, conversely, is that it is computationally trivial.

Given two known reference points, or anchors, in three dimensional space, and a single point, or tag, with a known distance to each of these two anchors, it is possible to determine the circle, representing the intersection of two spheres, upon which this tag resides through a process here referred to as Bilateration. The circle that represents the intersection of two spheres (see figure 1.1), will lie on a plane that is orthogonal to the vector between the two anchors. The problem with Bilateration is that it still does not give exact coordinates in three dimensional space. However, in two dimensional space, Bilateration will give two unique solutions.

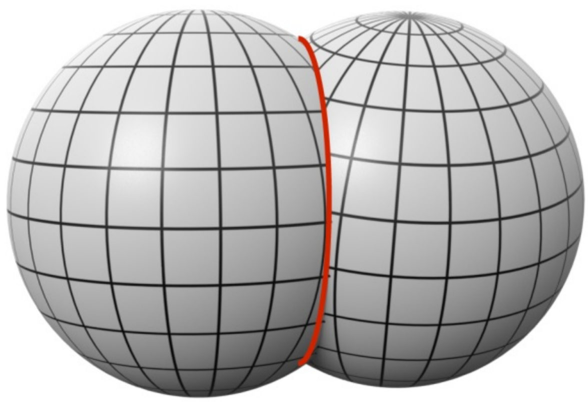

Figure 1.1: Intersection of two sphere represented by red circle

Given three known reference points, or anchors, in three dimensional space, and a single point, or tag, with a known distance to each of these three anchors, it is possible to determine a maximum of two point (see figure 1.2), representing the intersection of three spheres, out of which one will be the precise coordinates of this tag through a process known as Trilateration. In two dimensional space however, the solution to Trilateration equates to a single point. The main problem with Trilateration is that it is more computationally expensive than either Unilateration or Bilateration.

The table below will summarizes solutions of Unilateration, Bilateration and Trilateration in 3D and 2D space, along with their computational expense.

Method	**Computational Expense**	**Solution in 3D**	**Solution in 2D**
Unilateration	Low	Sphere	Circle
Bilateration	Medium	Circle	Two points
Trilateration	High	Two points	Single point

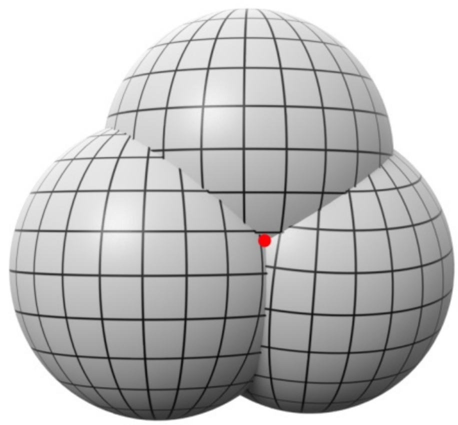

Figure 1.2: Intersection of three sphere represented by red dots (on each side)

Although Bilateration alone still fails to provide exact coordinates in three dimensional space, it does provide two exact coordinates in two dimensional space. Bilateration is, thus, more efficient than Trilateration, since it uses less mathematics and less computation to calculate the coordinates of a tag. This, in turn will reduce the computation resources required to solve the lateration problem. This document will describe a more computationally efficient version of Bilateration through the additional use of either simple logic or a third anchor to check for the correct coordinates.

2. Goal

Computationally, it might not make much difference for a computer to perform Trilateration versus Bilateration a few times per second. But the difference is huge, if a computer is performing this calculation hundreds or thousands of times per second.

The main goal of this report is to find a simpler method of Bilateration in order to reduce the computational workload of the computer performing the calculations.

3. Background

3.1. True Trilateration

As mentioned before, true Trilateration is the process of finding two intersection points of three spheres in three dimensional space. The following will describe the process for Trilateration.

Suppose there is three unique anchor points in three dimensional space, $A_1(x_1, y_1, z_1)$, $A_2(x_2, y_2, z_2)$, and $A_3(x_3, y_3, z_3)$ and we would like to find the coordinates of the tag $T(x, y, z)$. Also the distance from T to each of the anchor points are known:

$$d_1 = |T - A_1|$$
$$d_2 = |T - A_2|$$
$$d_3 = |T - A_3|$$

The derivation of Trilateration in three dimensional space with any three arbitrary points is very complex and it is described in appendix A. The MATLAB program for computing the solutions to the general Trilateration problem, is in appendix B.

For simplicity, we will assume that the three anchor points lie on the xy plane with $z = 0$. Here we rewrite the coordinates of the three anchors and the equation of the sphere associated with each anchor as follows:

$$A_1(0,0,0) \implies x^2 + y^2 + z^2 = d_1^2 \tag{3.1}$$
$$A_2(x_2,0,0) \implies (x - x_2)^2 + y^2 + z^2 = d_2^2 \tag{3.2}$$
$$A_3(x_3,y_3,0) \implies (x - x_3)^2 + (y - y_3)^2 + z^2 = d_3^2 \tag{3.3}$$

Now we will subtract equations 3.1 and 3.2 to solve for x:

$$\begin{array}{rr} & x^2 + x_2^2 - 2xx_2 + y_2 + z^2 = d_2^2 \\ \text{minus} & x^2 + y^2 + z^2 = d_1^2 \\ \hline \end{array}$$

$$x_2^2 - 2xx_2 = d_2^2 - d_1^2$$
$$2xx_2 = d_1^2 - d_2^2 + x_2^2$$
$$x = \frac{d_1^2 - d_2^2 + x_2^2}{2x_2} \tag{3.4}$$

Now we will subtract equations 3.1 and 3.3 to solve for y:

$$\begin{array}{rr} & x^2 + x_3^2 + y^2 + y_2^3 - 2xx_3 - 2yy_3 + z^2 = d_3^2 \\ \text{minus} & x^2 + y^2 + z^2 = d_1^2 \\ \hline \end{array}$$

$$x_3^2 + y_3^2 - 2xx_3 - 2yy_3 = d_3^2 - d_1^2$$
$$2yy_3 = d_1^2 - d_3^2 + x_3^2 + y_3^2 - \frac{x_3\left(d_1^2 - d_2^2 + x_2^2\right)}{x_2}$$
$$y = \frac{d_1^2 - d_3^2 + x_3^2 + y_3^2 - \frac{x_3\left(d_1^2 - d_2^2 + x_2^2\right)}{x_2}}{2y_3} \tag{3.5}$$

Since we have the expressions for x (3.4) and y (3.5), we can plug these into equation 3.1 and solve for z:

$$z = \pm\sqrt{d_1^2 - \left(\frac{d_1^2 - d_2^2 + x_2^2}{2x_2}\right)^2 - \left(\frac{d_1^2 - d_3^2 + x_3^2 + y_3^2 - \frac{x_3\left(d_1^2 - d_2^2 + x_2^2\right)}{x_2}}{2y_3}\right)^2} \tag{3.6}$$

There will two unique solutions to the Trilateration problem, since the result of three intersecting spheres are two points according to figure 1.2.

4. Methods

The Bilateration method outlined below, computes two possible solutions for a tag in two dimensional space, given the distance from this point to two anchors. From the two solutions, only one will be selected as the final solution based on one of the following:

- Simple logic based of physical location of anchors, or
- Distance from a third anchor point to only check for the correct solution, or
- Historical movement of the tag

5. Bilateration Derivation

Using the same simplified technique as we have used for Trilateration, we will derive the solutions for Bilateration in two dimension. For simplicity we will assume we have two anchor points,

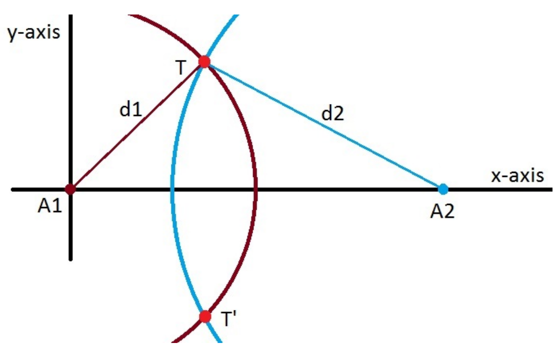

Figure 5.1: Simplified Bilateration

where one lies at the origin, $A_1(0,0)$, and the other lies on the x axis, $A_2(x_2,0)$. Since we know the distance from the tag $T(x, y)$ to the anchor points, A_1 and A_2, are d_1 and d_2 respectively, the following holds true:

$$A_1(0,0) \implies x^2 + y^2 = d_1^2 \tag{5.1}$$

$$A_2(x_2,0) \implies (x - x_2)^2 + y^2 = d_2^2 \tag{5.2}$$

where, 5.1 is the equation of a circle associated to A_1, and 5.2 is the equation of a circle associated to A_2.

To solve for x we can subtract equation 5.1 from 5.2:

$$\begin{array}{lr} & x^2 + x_2^2 - 2xx_2 + y^2 = d_2^2 \\ \text{minus} & x^2 + y^2 = d_1^2 \\ \hline & x_2^2 - 2xx_2 = d_2^2 - d_1^2 \\ & 2xx_2 = d_1^2 - d_2^2 + x_2^2 \\ & x = \dfrac{d_1^2 - d_2^2 + x_2^2}{2x_2} \end{array} \tag{5.3}$$

Then we substitute the expression for x (5.3) into equation 5.2 and solve for y:

$$y = \pm\sqrt{d_1^2 - \left(\frac{d_1^2 - d_2^2 + x_2^2}{2x_2}\right)^2} \tag{5.4}$$

There is two unique solutions here, since the intersection of two circles in a plane will result in a maximum of two points.

For a more general solution to Bilateration using any two arbitrary anchor points, see appendix C.

6. Bilateration Code

From a programming point of view the code for Bilateration will take the following pseudo code, which is based on the general derivation of Bilateration (Appendix C).

** Code is redacted **

where X_1 and Y_1 are the coordinates of the first solution and X_2 and Y_2 are the coordinates of the second solution.

7. Benefits of Bilateration

Suppose you have a room that is 10×10 meters and you are trying to find the coordinates of the tag T in the room.

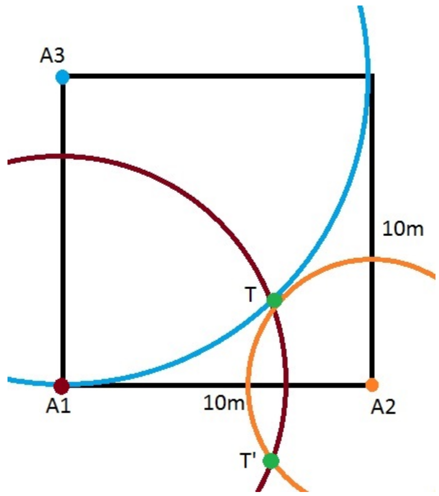

Figure 7.1: Room Example

7.1. Computation Benefits

Using the Trilateration method, we would require all three anchors points, A_1, A_2, and A_3, to compute the coordinate of the tag, given the distance from the tag to each of the anchors.

On the other hand, using the Bilateration method will give two solutions, given the distance from the tag to anchors A_1 and A_2. The solutions will be location of T and T'. Using simple logic, T will be chosen as the only solution and T' will be eliminated since it lies outside of the room.

We can also use a third anchor (A_3) to check for the correct solution, which is this case will be T.

The historical movement of the tag can also be used to chose the correct solution. This means, if we know where the tag has been in the past (a second ago), then the new position of the tag must be close to the last position (i.e. the tag cannot jump from T to T' is a short period of time).

7.2. Cost Benefits

Continuing the same room example, the trilateration method requires a minimum of three anchors having reliable communication with a tag to find its location. Depending on the hardware used, the Trilatration method required the anchors to be spaced close to each other, such that a tag in the field of coverage, is able to communicate to at least three anchors. In terms of implementation, this means more hardware, setup time, and cost.

On the other hand, using Bilateration, a tag can be located by only two anchors (A_1 and A_2). In terms of implementation, this means the hardware can be more spaced apart from each other, since a tag requires reliable communication with only two anchors and not three. The result of this is less hardware, setup time and cost.

7.3. ACCURACY

Using Trilateration in the same room example, gives only one way to get a solution from three anchors at any point in time. If more accuracy is needed, the Bilateration method gives three ways for three solution:

1. A1 and A2
2. A2 and A3
3. A1 and A3

Three chances to perform Bilateration mean we have three sets of solutions for the tag at any point in time. This provides the means for checking one solution against the other, which concludes in more accurate results.

8. BILATERATION LIMITATION

By using Bilateration with either simple logic or a third anchor, the solution provides the location of a tag in only a two dimensional environment. However, this limitation does not pose a problem in an office type environment where elevation can be neglected (i.e. no one flies in an office).

9. CONCLUSION

Trilateration is a computationally expensive math problem that needs to be solved to get the coordinates of a tag, and must use a minimum of three anchor points.

However, with Bilateration, only two anchors are used to solve for two possible solutions to the location of anchor. Then the correct solution can be chosen by using either simple logic based on the physical implementation of the anchor points, or using a third anchor point to only check for the correct solution, which is trivial.

Using Bilateration with simple logic to chose the correct solution, uses the least amount of computational resources, however during the installation of the tracking system consideration must be given to the location of anchor points relative to the structure or building so the logic can be determined.

Using Bilateration and an additional third anchor to check for the correct solution, achieves the same result and is computationally less expensive compared to Trilateration.

The table below summarizes the findings in this report:

Method	**Implementation**	**Computation Cost**
Bilateration with simple logic	Consideration of structure is required	Least expensive
Bilateration with third anchor	No consideration of structure needed	Moderately expensive
Trilateration	No consideration of structure needed	Most expensive

Appendices

A. Trilateration General Derivation

Suppose we have three known arbitrary anchor points $A_1(x_1, y_1, z_1)$, $A_2(x_2, y_2, z_2)$, and $A_3(x_3, y_3, z_3)$, in three dimensional space and we would like to find the coordinates of a tag $T(x, y, z)$, from which we know the distance d_1, d_2, and d_3 to the anchor points such that:

$$d_1 = |T - A_1| \implies d_1^2 = (x - x_1)^2 + (y - y_1)^2 + (z - z_1)^2 \tag{A.1}$$

$$d_2 = |T - A_2| \implies d_1^2 = (x - x_2)^2 + (y - y_2)^2 + (z - z_2)^2 \tag{A.2}$$

$$d_3 = |T - A_3| \implies d_3^2 = (x - x_3)^2 + (y - y_3)^2 + (z - z_3)^2 \tag{A.3}$$

Also the three anchor points must satisfy the following conditions:

$$|A_1 - A_2| < |A_1 - A_3| + |A_2 - A_3|$$
$$|A_1 - A_3| < |A_1 - A_2| + |A_2 - A_3|$$
$$|A_2 - A_3| < |A_1 - A_2| + |A_1 - A_3|$$

Which means that the three anchor points must form a triangle in space and must not be collinear (i.e. the three anchors do not lie in a straight line).

Solving this problem, is a very complex mathematical problem. We know that any three points in space will lie on a single plane, P, in three dimensional space. To make the mathematic simpler to work with we need to apply a few operations to change the reference frame, to achieve the following:

- Have an anchor at the origin
- Have a second anchor on the x axis
- Have the third anchor lie on the xy plane

To achieve this, the following steps need to be taken:

1. Translate the reference frame, such that one of the anchors lies at origin. In this case we would like A_1 to move to origin.
2. Align the normalized vector of plane P on the z axis. This results in plane P along with A_2 and A_3, lying on the xy plane.
3. Rotate the plane P about the z axis, such that a second anchor point, A_2, lies on the x axis. In this case we would like A_2 to be on x axis.
4. Perform simplified Trilateration according to section 3.1.
5. Rotate and translate the solution back to the original frame of reference.

Each of these steps will be explained below.

A.1. TRANSLATION OF REFERENCE FRAME

The first step requires us to translate the plane P, where the three anchor points lie, such that the first anchor point A_1 will lie at the origin point $(0,0,0)$.

Therefore, we use the vector $-\overrightarrow{A_1}$, which will get subtracted by all the anchor points to give the translated coordinates:

$$A_1 - \overrightarrow{A_1} \implies A_1(x_1 - x_1, y_1 - y_1, z_1 - z_1) \tag{A.4}$$

$$A_2 - \overrightarrow{A_1} \implies A_2(x_2 - x_1, y_2 - y_1, z_2 - z_1) \tag{A.5}$$

$$A_3 - \overrightarrow{A_1} \implies A_3(x_3 - x_1, y_3 - y_1, z_3 - z_1) \tag{A.6}$$

A.2. ROTATION OF NORMALIZED VECTOR OF P TO z AXIS

In this step, the goal is to align the P plane, in such a way that it lies on the xy plane, by aligning the normal unit vector (aka. normalized vector) of P that passes through origin, onto the z axis unit vector $\hat{k} = (0,0,1)$.

To do this, we first need to find the normalized vector of P passing through origin, which from the previous step is also the anchor point A_1 (see A.4).

To find the normalized vector $\mathbf{N}$, we need to have a parametric equation for plane P in the form of:

$$P = f\left(x, y, z\right) = ax + by + cz + d = 0 \tag{A.7}$$

from which the normal vector will be:

$$\mathbf{N} = \begin{bmatrix} a \\ b \\ c \end{bmatrix} \tag{A.8}$$

Note that the normal vector is *not* the normalized vector. The normalized vector of $\mathbf{N}$ is:

$$\hat{\mathbf{N}} = \frac{\mathbf{N}}{|\mathbf{N}|} \tag{A.9}$$

where $|\mathbf{N}|$ is the norm or magnitude of $\mathbf{N}$.

A.2.1. EQUATION FOR PLANE P

Now we need to find an equation of plane P that passes through the three anchor points $A_1(x_1, y_1, z_1)$, $A_2(x_2, y_2, z_2)$, and $A_3(x_3, y_3, z_3)$.

The equation of the plane P can be found by doing a cross product of two vectors that lie on P. In this case our two vectors are $\overrightarrow{A_1A_2}$ and $\overrightarrow{A_1A_3}$. Since anchor A_1 is at origin, the vector notation simplifies to $\overrightarrow{A_2}$ and $\overrightarrow{A_3}$.

The cross product of $\overrightarrow{A_2}$ and $\overrightarrow{A_3}$ is:

$$\overrightarrow{A_2} \times \overrightarrow{A_3} = \begin{vmatrix} \hat{i} & \hat{j} & \hat{k} \\ x_2 - x_1 & y_2 - y_1 & z_2 - z_1 \\ x_3 - x_1 & y_3 - y_1 & z_3 - z_1 \end{vmatrix} = \begin{vmatrix} y_2 - y_1 & z_2 - z_1 \\ y_3 - y_1 & z_3 - z_1 \end{vmatrix} \hat{i} - \begin{vmatrix} x_2 - x_1 & z_2 - z_1 \\ x_3 - x_1 & z_3 - z_1 \end{vmatrix} \hat{j} + \begin{vmatrix} x_2 - x_1 & y_2 - y_1 \\ x_3 - x_1 & y_3 - y_1 \end{vmatrix} \hat{k}$$

$$\begin{aligned} &= \left[\left(y_2 - y_1\right)(z_3 - z_1) - \left(y_3 - y_1\right)(z_2 - z_1) \right] \hat{i} \\ &\quad - \left[(x_2 - x_1)(z_3 - z_1) - (x_3 - x_1)(z_2 - z_1) \right] \hat{j} \\ &\quad + \left[(x_2 - x_1)\left(y_3 - y_1\right) - (x_3 - x_1)\left(y_2 - y_1\right) \right] \hat{k} \end{aligned}$$

$$\overrightarrow{A_2} \times \overrightarrow{A_3} = \mathbf{N} = \begin{bmatrix} \left(y_2 - y_1\right)(z_3 - z_1) - \left(y_3 - y_1\right)(z_2 - z_1) \\ (x_3 - x_1)(z_2 - z_1) - (x_2 - x_1)(z_3 - z_1) \\ (x_2 - x_1)\left(y_3 - y_1\right) - (x_3 - x_1)\left(y_2 - y_1\right) \end{bmatrix} \tag{A.10}$$

This cross product, is also a normal vector $\mathbf{N}$ to plane P, which we will use later in section A.2.2.

By substituting the values of A.10 into equation A.7, we get the following equation for plane P is:

$$\begin{aligned} P = &\left[\left(y_2 - y_1\right)(z_3 - z_1) - \left(y_3 - y_1\right)(z_2 - z_1) \right] x \\ &+ \left[(x_3 - x_1)(z_2 - z_1) - (x_2 - x_1)(z_3 - z_1) \right] y \\ &+ \left[(x_2 - x_1)\left(y_3 - y_1\right) - (x_3 - x_1)\left(y_2 - y_1\right) \right] z = 0 \end{aligned} \tag{A.11}$$

A.2.2. Rotation of $\hat{\mathbf{N}}$ to $\hat{k}$

Since we have the normal vector of plane P, we can find the normalized vector $\hat{\mathbf{N}}$ from equation A.9:

$$\hat{\mathbf{N}} = \frac{\mathbf{N}}{|\mathbf{N}|} = \begin{bmatrix} \dfrac{a}{\sqrt{a^2 + b^2 + c^2}} \\ \dfrac{b}{\sqrt{a^2 + b^2 + c^2}} \\ \dfrac{c}{\sqrt{a^2 + b^2 + c^2}} \end{bmatrix} \tag{A.12}$$

where:

$$\begin{aligned} a &= \left(y_2 - y_1\right)(z_3 - z_1) - \left(y_3 - y_1\right)(z_2 - z_1) \\ b &= (x_3 - x_1)(z_2 - z_1) - (x_2 - x_1)(z_3 - z_1) \\ c &= (x_2 - x_1)\left(y_3 - y_1\right) - (x_3 - x_1)\left(y_2 - y_1\right) \end{aligned}$$

To make things simpler we will denote the normalized vector $\hat{\mathbf{N}}$ as:

$$\hat{\mathbf{N}} = \begin{bmatrix} u \\ v \\ w \end{bmatrix} \tag{A.13}$$

where:

$$u = \frac{a}{\sqrt{a^2+b^2+c^2}}$$
$$v = \frac{b}{\sqrt{a^2+b^2+c^2}}$$
$$w = \frac{c}{\sqrt{a^2+b^2+c^2}}$$

Now we can rotate the normalized vector $\hat{\mathbf{N}}$ to go on the z axis. To do this, first we will rotate $\hat{\mathbf{N}}$ about z, such that $\hat{\mathbf{N}}$ lies on the xz plane. Then we will rotate $\hat{\mathbf{N}}$ about y axis again, such that it lies on the z axis.

ROTATION OF $\hat{\mathbf{N}}$ TO xz PLANE To rotate the vector $\hat{\mathbf{N}}$ about z axis in the clockwise direction, $\hat{\mathbf{N}}$ need to multiplied by a rotation matrix in the form of:

$$R_z(\theta) = \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{A.14}$$

But first we have to find expressions for $\cos\theta$ and $\sin\theta$ in terms of normalized vector $\hat{\mathbf{N}}$.

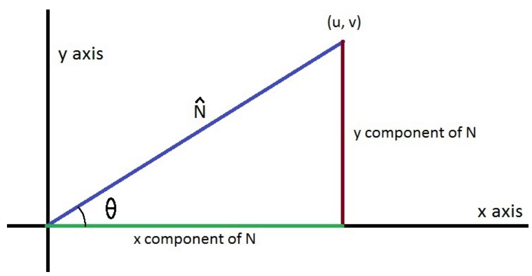

Figure A.1: Rotation of $\hat{\mathbf{N}}$ about z axis

By projecting the normalized vector $\hat{\mathbf{N}}$ onto the xy plane (i.e. neglecting the z component) (refer to figure A.1) we can clearly see that the expressions for $\cos\theta$ and $\sin\theta$ are:

$$\cos\theta = \frac{u}{\sqrt{u^2+v^2}} \tag{A.15}$$
$$\sin\theta = \frac{v}{\sqrt{u^2+v^2}} \tag{A.16}$$

By substituting expressions A.15 and A.16 into A.14, we get the following rotation matrix about z

axis:

$$R_z(\theta) = \begin{bmatrix} \dfrac{u}{\sqrt{u^2+v^2}} & \dfrac{v}{\sqrt{u^2+v^2}} & 0 \\ -\dfrac{v}{\sqrt{u^2+v^2}} & \dfrac{u}{\sqrt{u^2+v^2}} & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{A.17}$$

Now we can multiply the rotation matrix $R_z(\theta)$, by the normalized vector $\hat{\mathbf{N}}$:

$$R_z(\theta)\hat{\mathbf{N}} = \hat{\mathbf{N}}_{xz} = \begin{bmatrix} \frac{u}{\sqrt{u^2+v^2}} & \frac{v}{\sqrt{u^2+v^2}} & 0 \\ -\frac{v}{\sqrt{u^2+v^2}} & \frac{u}{\sqrt{u^2+v^2}} & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u \\ v \\ w \end{bmatrix} = \begin{bmatrix} \frac{u^2}{\sqrt{u^2+v^2}} + \frac{v^2}{\sqrt{u^2+v^2}} \\ -\frac{uv}{\sqrt{u^2+v^2}} + \frac{uv}{\sqrt{u^2+v^2}} \\ w \end{bmatrix} = \begin{bmatrix} \frac{u^2+v^2}{\sqrt{u^2+v^2}} \\ 0 \\ w \end{bmatrix} = \begin{bmatrix} \sqrt{u^2+v^2} \\ 0 \\ w \end{bmatrix} \tag{A.18}$$

where $\hat{\mathbf{N}}_{xz}$ is the rotated normalized vector $\hat{\mathbf{N}}$ onto xz plane.

ROTATION OF $\hat{\mathbf{N}}_{xz}$ TO z AXIS Once again we have to rotate the vector $\hat{\mathbf{N}}_{xz}$ from equation A.18 about the y axis, such that it lies on the z axis. The rotation matrix about y axis will take the following form:

$$R_y(\psi) = \begin{bmatrix} \cos\psi & 0 & -\sin\psi \\ 0 & 1 & 0 \\ \sin\psi & 0 & \cos\psi \end{bmatrix} \tag{A.19}$$

Looking at figure A.2 the expressions for $\cos\psi$ and $\sin\psi$ are:

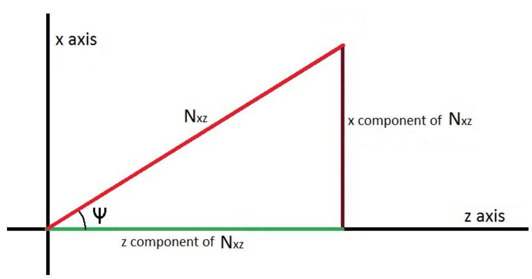

Figure A.2: Rotation of $\hat{\mathbf{N}}_{xz}$ about y axis

$$\cos\psi = \frac{w}{\sqrt{w^2 + \left(\sqrt{u^2+v^2}\right)^2}} = \frac{w}{\sqrt{u^2+v^2+w^2}} = w \tag{A.20}$$

$$\sin\psi = \frac{\sqrt{u^2+v^2}}{\sqrt{w^2 + \left(\sqrt{u^2+v^2}\right)^2}} = \frac{\sqrt{u^2+v^2}}{\sqrt{u^2+v^2+w^2}} = \sqrt{u^2+v^2} \tag{A.21}$$

By substituting the expressions A.20 and A.21 into A.19, the rotation matrix $R_y(\psi)$ is:

$$R_y(\psi) = \begin{bmatrix} w & 0 & -\sqrt{u^2+v^2} \\ 0 & 1 & 0 \\ \sqrt{u^2+v^2} & 0 & w \end{bmatrix} \tag{A.22}$$

Now we can multiply the rotation matrix $R_y(\psi)$, by the vector $\hat{\mathbf{N}}_{xz}$:

$$\begin{aligned} R_y(\psi)\hat{\mathbf{N}}_{xz} = \hat{\mathbf{N}}_z &= \begin{bmatrix} w & 0 & -\sqrt{u^2+v^2} \\ 0 & 1 & 0 \\ \sqrt{u^2+v^2} & 0 & w \end{bmatrix} \begin{bmatrix} \sqrt{u^2+v^2} \\ 0 \\ w \end{bmatrix} \\ &= \begin{bmatrix} w\sqrt{u^2+v^2} - w\sqrt{u^2+v^2} \\ 0 \\ u^2+v^2+w^2 \end{bmatrix} \\ &= \begin{bmatrix} 0 \\ 0 \\ u^2+v^2+w^2 \end{bmatrix} \end{aligned} \tag{A.23}$$

This perfectly makes sense, because by rotating $\hat{\mathbf{N}}_{xz}$ onto z axis, we should have it lie on the z axis.

A.2.3. Rotation of A_2 and A_3 to xy Plane

By now we should have A_1 at origin, and the matrix rotations for transferring Plane P to xy plane. We can use the same rotation matrices to transfer anchors A_2 and A_3 to xy plane. Note that rotation is not commutative. Therefore, each anchor will be first rotated around z axis using A.17 and then around y axis using A.22.

$$A_{2xy} = R_z(\theta)R_y(\psi)A_2$$
$$A_{3xy} = R_z(\theta)R_y(\psi)A_3$$

where A_{2xy} and A_{3xy} are the rotated versions of A_2 and A_3, on the xy plane.

We will use a 2 step process to find A_{2xy} and A_{3xy}.

1. $$A_{2\theta} = R_z(\theta)A_2$$
$$A_{3\theta} = R_z(\theta)A_3$$

2. $$A_{2xy} = R_y(\psi)A_{2\theta}$$
$$A_{3xy} = R_y(\psi)A_{3\theta}$$

where $A_{2\theta}$ and $A_{3\theta}$ are the rotated representations of A_2 and A_3 rotated about z axis by angle θ.

FINDING $A_{2\theta}$ AND $A_{3\theta}$

$$A_{2\theta} = \begin{bmatrix} \frac{u}{\sqrt{u^2+v^2}} & \frac{v}{\sqrt{u^2+v^2}} & 0 \\ -\frac{v}{\sqrt{u^2+v^2}} & \frac{u}{\sqrt{u^2+v^2}} & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_2 - x_1 \\ y_2 - y_1 \\ z_2 - z_1 \end{bmatrix} = \begin{bmatrix} \frac{u(x_2-x_1)+v(y_2-y_1)}{\sqrt{u^2+v^2}} \\ \frac{u(y_2-y_1)-v(x_2-x_1)}{\sqrt{u^2+v^2}} \\ z_2 - z_1 \end{bmatrix} \tag{A.24}$$

$$A_{3\theta} = \begin{bmatrix} \frac{u}{\sqrt{u^2+v^2}} & \frac{v}{\sqrt{u^2+v^2}} & 0 \\ -\frac{v}{\sqrt{u^2+v^2}} & \frac{u}{\sqrt{u^2+v^2}} & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_3 - x_1 \\ y_3 - y_1 \\ z_3 - z_1 \end{bmatrix} = \begin{bmatrix} \frac{u(x_3-x_1)+v(y_3-y_1)}{\sqrt{u^2+v^2}} \\ \frac{u(y_3-y_1)-v(x_3-x_1)}{\sqrt{u^2+v^2}} \\ z_3 - z_1 \end{bmatrix} \tag{A.25}$$

FINDING A_{2xy} AND A_{3xy}

$$A_{2xy} = \begin{bmatrix} w & 0 & -\sqrt{u^2+v^2} \\ 0 & 1 & 0 \\ \sqrt{u^2+v^2} & 0 & w \end{bmatrix} \begin{bmatrix} \frac{u(x_2-x_1)+v(y_2-y_1)}{\sqrt{u^2+v^2}} \\ \frac{u(y_2-y_1)-v(x_2-x_1)}{\sqrt{u^2+v^2}} \\ z_2 - z_1 \end{bmatrix} = \begin{bmatrix} \frac{w(u(x_2-x_1)+v(y_2-y_1))}{\sqrt{u^2+v^2}} - (z_2 - z_1)\sqrt{u^2+v^2} \\ \frac{u(y_2-y_1)-v(x_2-x_1)}{\sqrt{u^2+v^2}} \\ u(x_2-x_1) + v(y_2-y_1) + w(z_2-z_1) \end{bmatrix} \tag{A.26}$$

$$A_{3xy} = \begin{bmatrix} w & 0 & -\sqrt{u^2+v^2} \\ 0 & 1 & 0 \\ \sqrt{u^2+v^2} & 0 & w \end{bmatrix} \begin{bmatrix} \frac{u(x_3-x_1)+v(y_3-y_1)}{\sqrt{u^2+v^2}} \\ \frac{u(y_3-y_1)-v(x_3-x_1)}{\sqrt{u^2+v^2}} \\ z_3 - z_1 \end{bmatrix} = \begin{bmatrix} \frac{w(u(x_3-x_1)+v(y_3-y_1))}{\sqrt{u^2+v^2}} - (z_3 - z_1)\sqrt{u^2+v^2} \\ \frac{u(y_3-y_1)-v(x_3-x_1)}{\sqrt{u^2+v^2}} \\ u(x_3-x_1) + v(y_3-y_1) + w(z_3-z_1) \end{bmatrix} \tag{A.27}$$

By substituting the values of u, v, and w, the z component of the above expressions will become zero as expected, and we will get the following representation of A_2 and A_3, on the xy plane:

$$A_{2xy} = \begin{bmatrix} \frac{w(u(x_2-x_1)+v(y_2-y_1))}{\sqrt{u^2+v^2}} - (z_2 - z_1)\sqrt{u^2+v^2} \\ \frac{u(y_2-y_1)-v(x_2-x_1)}{\sqrt{u^2+v^2}} \\ 0 \end{bmatrix} \tag{A.28}$$

$$A_{3xy} = \begin{bmatrix} \frac{w(u(x_3-x_1)+v(y_3-y_1))}{\sqrt{u^2+v^2}} - (z_3 - z_1)\sqrt{u^2+v^2} \\ \frac{u(y_3-y_1)-v(x_3-x_1)}{\sqrt{u^2+v^2}} \\ 0 \end{bmatrix} \tag{A.29}$$

A.3. Rotation of A_{2xy} to x Axis

Now that we know the coordinates of A_{2xy} and A_{3xy} from A.28 and A.29, we can rotate A_{2xy} and A_{3xy}, such that A_{2xy} lands on x axis. This can be done by rotation about z axis (equation A.14). The rotation matrix to perform this rotation is:

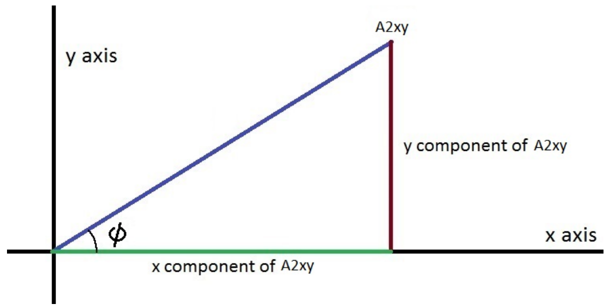

Figure A.3: Rotation matrix of A_{2xy} to go on x axis

$$R_z(\phi) = \begin{bmatrix} \cos\phi & \sin\phi & 0 \\ -\sin\phi & \cos\phi & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{A.30}$$

where:

$$\cos\phi = \frac{\dfrac{w\left(u\left(x_2-x_1\right)+v\left(y_2-y_1\right)\right)}{\sqrt{u^2+v^2}} - \left(z_2-z_1\right)\sqrt{u^2+v^2}}{\sqrt{\left(\dfrac{w\left(u\left(x_2-x_1\right)+v\left(y_2-y_1\right)\right)}{\sqrt{u^2+v^2}} - \left(z_2-z_1\right)\sqrt{u^2+v^2}\right)^2 + \left(\dfrac{u\left(y_2-y_1\right)-v\left(x_2-x_1\right)}{\sqrt{u^2+v^2}}\right)^2}}$$

$$\sin\phi = \frac{\dfrac{u\left(y_2-y_1\right)-v\left(x_2-x_1\right)}{\sqrt{u^2+v^2}}}{\sqrt{\left(\dfrac{w\left(u\left(x_2-x_1\right)+v\left(y_2-y_1\right)\right)}{\sqrt{u^2+v^2}} - \left(z_2-z_1\right)\sqrt{u^2+v^2}\right)^2 + \left(\dfrac{u\left(y_2-y_1\right)-v\left(x_2-x_1\right)}{\sqrt{u^2+v^2}}\right)^2}}$$

Now we can apply this rotation to A_{2xy} and A_{3xy}:

$$A_{2x} = R_z(\phi) A_{2xy}$$

$$= \begin{bmatrix} \cos\phi & \sin\phi & 0 \\ -\sin\phi & \cos\phi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \frac{w(u(x_2-x_1)+v(y_2-y_1))}{\sqrt{u^2+v^2}} - (z_2 - z_1)\sqrt{u^2+v^2} \\ \frac{u(y_2-y_1)-v(x_2-x_1)}{\sqrt{u^2+v^2}} \\ 0 \end{bmatrix}$$

$$A_{2x} = \begin{bmatrix} \left(\frac{w(u(x_2-x_1)+v(y_2-y_1))}{\sqrt{u^2+v^2}} - (z_2 - z_1)\sqrt{u^2+v^2}\right)\cos\phi + \left(\frac{u(y_2-y_1)-v(x_2-x_1)}{\sqrt{u^2+v^2}}\right)\sin\phi \\ -\left(\frac{w(u(x_2-x_1)+v(y_2-y_1))}{\sqrt{u^2+v^2}} - (z_2 - z_1)\sqrt{u^2+v^2}\right)\sin\phi + \left(\frac{u(y_2-y_1)-v(x_2-x_1)}{\sqrt{u^2+v^2}}\right)\cos\phi \\ 0 \end{bmatrix} \tag{A.31}$$

$$A_{3x} = R_z(\phi) A_{3xy}$$

$$= \begin{bmatrix} \cos\phi & \sin\phi & 0 \\ -\sin\phi & \cos\phi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \frac{w\left(u(x_3-x_1)+v\left(y_3-y_1\right)\right)}{\sqrt{u^2+v^2}} - (z_3 - z_1)\sqrt{u^2+v^2} \\ \frac{u\left(y_3-y_1\right)-v(x_3-x_1)}{\sqrt{u^2+v^2}} \\ 0 \end{bmatrix}$$

$$A_{3x} = \begin{bmatrix} \left(\frac{w(u(x_3-x_1)+v(y_3-y_1))}{\sqrt{u^2+v^2}} - (z_3 - z_1)\sqrt{u^2+v^2}\right)\cos\phi + \left(\frac{u(y_3-y_1)-v(x_3-x_1)}{\sqrt{u^2+v^2}}\right)\sin\phi \\ -\left(\frac{w(u(x_3-x_1)+v(y_3-y_1))}{\sqrt{u^2+v^2}} - (z_3 - z_1)\sqrt{u^2+v^2}\right)\sin\phi + \left(\frac{u(y_3-y_1)-v(x_3-x_1)}{\sqrt{u^2+v^2}}\right)\cos\phi \\ 0 \end{bmatrix} \tag{A.32}$$

By substituting the $\cos\phi$ and $\sin\phi$ expressions into A.31, the expression for A_{2x} simplifies to:

$$A_{2x} = \begin{bmatrix} \left(\frac{w(u(x_2-x_1)+v(y_2-y_1))}{\sqrt{u^2+v^2}} - (z_2 - z_1)\sqrt{u^2+v^2}\right)\cos\phi + \left(\frac{u(y_2-y_1)-v(x_2-x_1)}{\sqrt{u^2+v^2}}\right)\sin\phi \\ 0 \\ 0 \end{bmatrix} \tag{A.33}$$

This makes sense, because this representation of anchor A_2 is lying on the x axis and the y and z components are zero.

A.4. Simplified Trilateration

Now that we have a representation of all three anchors on the xy plane with A_1 at origin and A_2 on the x axis, we can perform the simplified Trilateration as described in section 3.1. In this case our input values are:

$$A_1 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \tag{A.34}$$

$$A_2 = \begin{bmatrix} \left(\frac{w(u(x_2-x_1)+v(y_2-y_1))}{\sqrt{u^2+v^2}} - (z_2 - z_1)\sqrt{u^2+v^2}\right)\cos\phi + \left(\frac{u(y_2-y_1)-v(x_2-x_1)}{\sqrt{u^2+v^2}}\right)\sin\phi \\ 0 \\ 0 \end{bmatrix} \tag{A.35}$$

$$A_3 = \begin{bmatrix} \left(\frac{w(u(x_3-x_1)+v(y_3-y_1))}{\sqrt{u^2+v^2}} - (z_3 - z_1)\sqrt{u^2+v^2}\right)\cos\phi + \left(\frac{u(y_3-y_1)-v(x_3-x_1)}{\sqrt{u^2+v^2}}\right)\sin\phi \\ -\left(\frac{w(u(x_3-x_1)+v(y_3-y_1))}{\sqrt{u^2+v^2}} - (z_3 - z_1)\sqrt{u^2+v^2}\right)\sin\phi + \left(\frac{u(y_3-y_1)-v(x_3-x_1)}{\sqrt{u^2+v^2}}\right)\cos\phi \\ 0 \end{bmatrix} \tag{A.36}$$

The solution to the simplified Trilateration will be in the form of:

$$T = \begin{bmatrix} x \\ y \\ z \end{bmatrix} \tag{A.37}$$

$$T' = \begin{bmatrix} x \\ y \\ -z \end{bmatrix} \tag{A.38}$$

A.5. Rotating and Transferring Back to Original Reference Frame

To transform the coordinates of the solution back to the original reference frame, we need to rotate and transfer the coordinates with the inverse of the same rotation and transfer matrices from section A.1 to section A.3, while keeping the order in place.

In the previous sections, we performed the following operations on the anchor coordinates:

1. Transfer the anchors by $-\overrightarrow{A_1}$
2. Rotate the anchors about z axis using $R_z(\theta)$ (matrix A.17)
3. Rotate the anchors about y axis using $R_y(\psi)$ (matrix A.22)
4. Rotate the anchors about z axis using $R_z(\phi)$ (matrix A.30)

To get the representations of solutions T and T' in the original reference frame, we need to perform these steps:

1. Rotate the anchors about z axis using $R_z(\phi)^{-1}$ (matrix A.30)
2. Rotate the anchors about y axis using $R_y(\psi)^{-1}$ (matrix A.22)

3. Rotate the anchors about z axis using $R_z(\theta)^{-1}$ (matrix A.17)

4. Transfer the anchors by $\overrightarrow{A_1}$

Finding the inverse of a 3×3 rotation matrix is easy:

$$R_z(\phi)^{-1} = R_z(-\phi) \tag{A.39}$$

$$R_y(\psi)^{-1} = R_y(-\psi) \tag{A.40}$$

$$R_z(\theta)^{-1} = R_z(-\theta) \tag{A.41}$$

Therefore, the following holds true:

$$T_o = R_z(\phi)^{-1} R_y(\psi)^{-1} R_z(\theta)^{-1} T + \overrightarrow{A_1} \tag{A.42}$$

$$T'_o = R_z(\phi)^{-1} R_y(\psi)^{-1} R_z(\theta)^{-1} T' + \overrightarrow{A_1} \tag{A.43}$$

where T_o and T'_o is the Trilateration solution to the original anchors $A_1(x_1, y_1, z_1)$, $A_2(x_2, y_2, z_2)$, and $A_3(x_3, y_3, z_3)$.

B. Trilateration Code

The following MATLAB code performs Trilateration in 3D with any three arbitrary anchor points and long as they satisfy the following conditions:

$$|A_1 - A_2| < |A_1 - A_3| + |A_2 - A_3|$$
$$|A_1 - A_3| < |A_1 - A_2| + |A_2 - A_3|$$
$$|A_2 - A_3| < |A_1 - A_2| + |A_1 - A_3|$$

and the distance d_1, d_2, and d_3 from the anchors A_1, A_2, and A_3 respectively must reach at least one point.

** Code is redacted **

C. Bilateration General Derivation

Let $A_1(x_1, y_1)$ and $A_2(x_2, y_2)$ be two known anchor points in 2D space and let $T(x, y)$ be the unknown tag that we would like the coordinates for. Also, we know that:

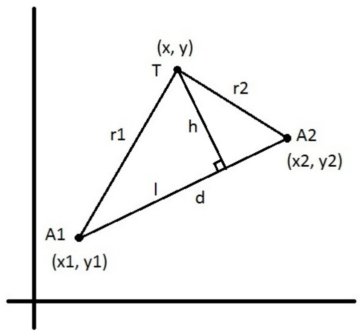

Figure C.1: Arbitrary anchors and tag location

$$r_1 = |A_1 - T|$$

$$r_2 = |A_2 - T|$$

$$d = |A_1 - A_2|$$

Then the following can be driven:

$$d = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

$$l = \frac{r_1^2 - r_2^2 + d^2}{2d}$$

$$h = \sqrt{r_1^2 - l^2}$$

And finally the two solutions to (x, y) are the following:

1.
$$x = \frac{l}{d}(x_2 - x_1) + \frac{h}{d}(y_2 - y_1) + x_1$$
$$y = \frac{l}{d}(y_2 - y_1) - \frac{h}{d}(x_2 - x_1) + y_1$$

2.
$$x = \frac{l}{d}(x_2 - x_1) - \frac{h}{d}(y_2 - y_1) + x_1$$
$$y = \frac{l}{d}(y_2 - y_1) + \frac{h}{d}(x_2 - x_1) + y_1$$