The Chase Problem (Part 1)

David C. Arney

> *We build systems like the Wright brothers built airplanes—build the whole thing, push it off a cliff, let it crash, and start all over again.*
>
> --- R. M. Graham [1970]

Introduction

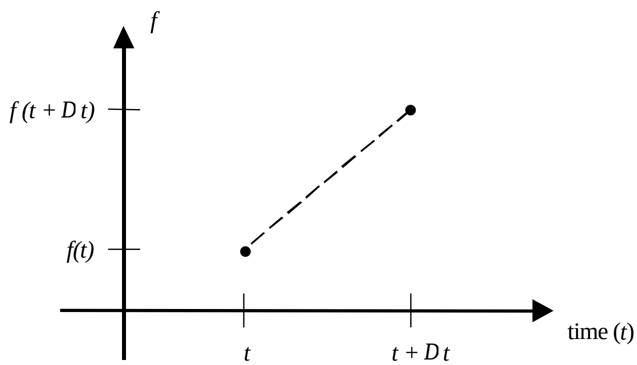
There are many situations where one thing, person, animal, or machine, chases another. Some of the applications of this kind of chasing in the military are: missiles intercepting planes (or other missiles), smart munitions seeking targets (i.e. anti-tank rounds seeking a tank), a unit or soldier pursuing and closing with an enemy unit or soldier, ships closing in on other ships, and torpedoes tracking and exploding on enemy ships. Of course, there are many non-military applications of chasing as well. Some of these are dogs running after cats, tacklers chasing and tackling ball carriers in football, hunters after their quarry (predators after prey), and children playing tag. All of these applications are three-dimensional (they occur in our three-dimensional world), but some are more easily, and possibly better, modeled in two dimensions because one dimension, like height, is not very significant. In this section, we will model these kinds of chase problems. We'll start in two dimensions, then refine our model to handle three dimensions.

Our problem is to determine the movement path for the chaser, given we know the location of the target. We will start with the assumption that the chaser has complete vision of the target and knows the target's position exactly. The chaser's position will be represented in two-dimensional Cartesian coordinates by $(x_0(t), y_0(t))$. Let's also start with assumptions that the chaser moves at a constant speed (given by s) and the target's position in two dimensions is given by the parametric relationship $(x_1(t), y_1(t))$. One technique to use for the chase model is to have the chaser move directly towards the target. This means that the chaser receives information as to the exact location of the target and heads in that direction. As the location of the target changes, the chaser adjusts its path to continue to move directly toward the target.

We can model this procedure through a discrete event model of time periods (intervals or timesteps) of length Δt. We use n to indicate the number of the time step in the model. Our generic modeling process for a time changing event is to set up a relationship that expresses the future state as the present state plus the change that occurs during the time interval. We will call this change our hypothesis for the change since we usually don't know before-hand exactly what will happen during the time interval. Figure 1 shows a schematic of this process using a generic function *f(t)* as the variable of interest.

FUTURE = PRESENT + CHANGE

$$f(t+\Delta t) = f(t) + \Delta t \text{ (hypothesis)}$$



difference equation: $f(n+1) = f(n) + D$ *per period*

Figure 1: Modeling change of a discrete time event simulation.

The last step in the diagram of Figure 1, shows a difference equation of the form

$$y(n+1) = y(n) + \Delta \text{ per time period} \qquad (1)$$

We will try to get our model for the movement of the chaser in this form to produce a simulation of the chase. Let's draw a diagram of what happens during a time interval Δt. Figure 2 shows the locations of the chaser and the target at some time t. The movement made by the chaser over that interval is indicated with the bold arrow. The final position of the chaser is given by $(x_0(t+\Delta t), y_0(t+\Delta t))$.

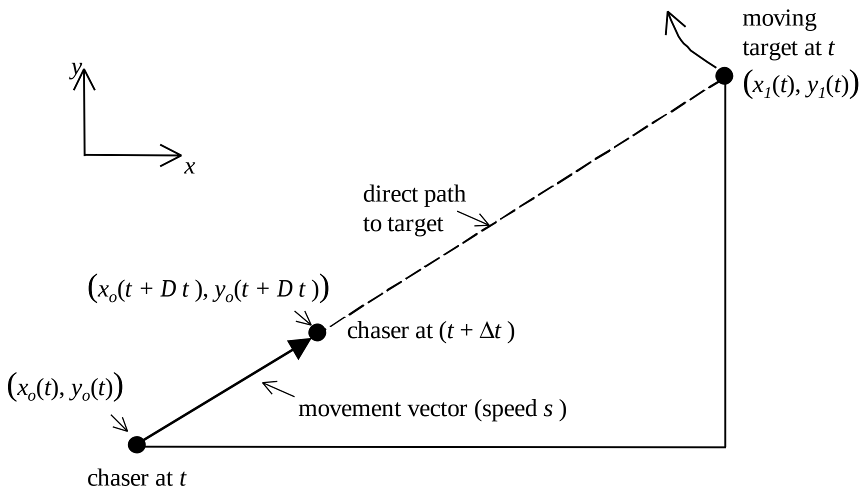


Figure 2: Movement by the chaser during the time interval t to Δt .

Our model needs to be a bit more sophisticated than the generic one shown in Figure 1. We need to keep track of two variables of interest, since we have a two-dimensional model. Our two changing variables of interest are the position components $x_0(t)$ and $y_0(t)$. Let's convert these two functions of the continuous variable t to discrete functions of our discrete time interval n. If we use the generic relation that $t = n\Delta t$, then, without explicitly showing the Δt in the discrete functions, we can represent $x_0(t)$ by $x_0(n)$, $y_0(t)$ by $y_0(n)$, $x_0(t+\Delta t)$ by $x_0(n+1)$, and $y_0(t+\Delta t)$ by $y_0(n+1)$. Next we'll try to relate these variables in the form of Equation (1) to obtain our mathematical model for this chasing process.

Let's visualize our relationships again. This time we carefully label the critical parts of our diagram, the points $(x_0(n), y_0(n))$, $(x_0(n+1), y_0(n+1))$, $(x_1(n), y_1(n))$ and the change in location of the chaser in each direction $\Delta x_0 =(x_0(n+1)- (x_0(n))$, and $\Delta y_0 .= (y_0(n+1)- (y_0(n))$. This new visual model is given in Figure 3.

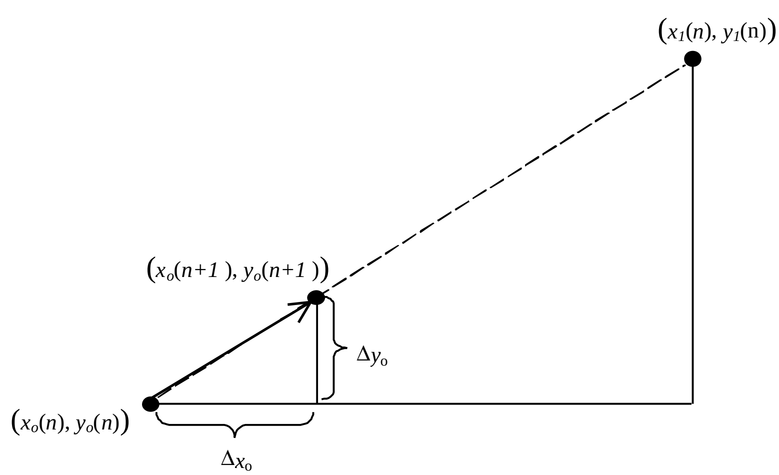


Figure 3: Movement by the chaser during the discrete time interval n to $n+1$.

We find two similar triangles in Figure 3. It's the relation between these similar right triangles that will enable us to write our model. Recall that we can set up proportional equations relating the sides of the triangles with the hypotenuse of the triangles. First let's determine formulas for each of the sides of our two triangles. The larger triangle in Figure 3 has its horizontal side of length $(x_1(n)-x_0(n))$. The vertical side has length $(y_1(n)-y_0(n))$. Therefore, by the Pythagorean Theorem the hypotenuse has length $\sqrt{(x_1(n)-x_0(n))^2+(y_1(n)-y_0(n))^2}$. The smaller triangle has sides Δx_0 and Δy_0. We need to determine the length of the hypotenuse in terms of values other than Δx_0 and Δy_0. We also know that the chaser moves at speed s over the time of the interval Δt. Therefore, the length of the hypotenuse represents the distance moved over the interval $s\Delta t$. Now, we can display our results geometrically, as we do in Figure 4.

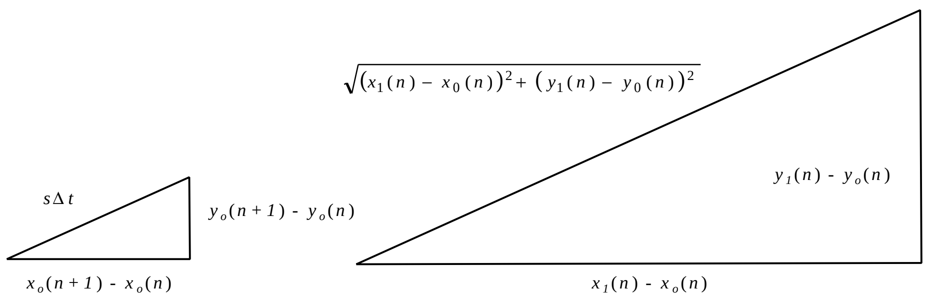


Figure 4: Similar triangles from Figure 3 with sides labeled with distances.

Our next step is to write out the equations relating the sides of the triangles with the hypotenuse of the triangles. First the horizontal side and hypotenuse of both triangles produce the relationship:

$$\frac{x_0(n+1)-x_0(n)}{s\Delta t}=\frac{x_1(n)-x_0(n)}{\sqrt{(x_1(n)-x_0(n))^2+(y_1(n)-y_0(n))^2}} \tag{2}$$

The vertical sides and hypotenuse produce:

$$\frac{y_0(n+1)-y_0(n)}{s\Delta t}=\frac{y_1(n)-y_0(n)}{\sqrt{(x_1(n)-x_0(n))^2+(y_1(n)-y_0(n))^2}} \tag{3}$$

We can clean up Equations (2) and (3) to create a system of two nonlinear difference equations for the unknowns $x_0(n+1)$ and $y_0(n+1)$, where both of these equations of our model are in the form of Equation (1):

$$x_0(n+1) = x_0(n) + \frac{s\Delta t(x_1(n) - x_0(n))}{\sqrt{(x_1(n) - x_0(n))^2 + (y_1(n) - y_0(n))^2}} \tag{4}$$

$$y_0(n+1) = y_0(n) + \frac{s\Delta t(y_1(n) - y_0(n))}{\sqrt{(x_1(n) - x_0(n))^2 + (y_1(n) - y_0(n))^2}} \tag{5}$$

This is our model, which will provide a means of determining the movement of the chaser, when we know the movement of the target. This system of difference equations is nonlinear and must be solved numerically by iteration. However, for any reasonable chase, we will need a computational tool, computer or calculator, to perform the iterations to determine the path of the chase. Remember our assumptions: the chaser moves at a constant speed, the chaser moves in a set direction over the time interval, and the chaser always sees the target. Let's look at an example.

Example 1: Target moving in a straight line.

In this example, where the target moves in a straight line, we need to have the starting coordinates and speed for the chaser and the parametric equations for the movement of the target. Let's start the chaser at the point *(-3,0).* We will use the following parametric equations for the target's movement:

$$x_1(t) = 3 + 3t \quad \text{and} \quad y_1(t) = 4t\,. \tag{6}$$

Therefore, at *t=0*, the target is located at the point *(3,0).* The target's speed is determined by the magnitude formula for its velocity. For Equation (6), the calculations for the speed use derivatives and simplify to $\sqrt{3^2 + 4^2} = 5$. We model our chaser with speed 7, so we should eventually catch the target, and our time interval set to $\Delta t = 0.1$. We convert our continuous parametric equations of Equation (6) to the following discrete equations:

$$x_1(n) = 3 + 3n\Delta t \quad \text{and} \quad y_1(n) = 4n\Delta t\,. \tag{7}$$

Now we can use our model in Equations (4) and (5) to solve for the movement of the chaser. Substituting the proper values and functions into Equations (4) and (5) we obtain:

$$x_0(n+1) = x_0(n) + \frac{7(0.1)(3 + 3(0.1)n - x_0(n))}{\sqrt{(3 + 3(0.1)n - x_0(n))^2 + (4(0.1)n - y_0(n))^2}}$$

$$y_0(n+1) = y_0(n) + \frac{7(0.1)(4(0.1)n - y_0(n))}{\sqrt{(3+3(0.1)n - x_0(n))^2 + (4(0.1)n - y_0(n))^2}}$$

A little algebraic clean up gives us the following equations:

$$x_0(n+1) = x_0(n) + \frac{0.7(3+0.3n - x_0(n))}{\sqrt{(3+0.3n - x_0(n))^2 + (0.4n - y_0(n))^2}} \qquad (8)$$

$$y_0(n+1) = y_0(n) + \frac{0.7(0.4n - y_0(n))}{\sqrt{(3+0.3n - x_0(n))^2 + (0.4n - y_0(n))^2}} \qquad (9)$$

with the initial condition that $x_0(0) = -3$ and $y_0(0) = 0$.

Now we can iterate Equations (8) and (9) to determine the path of the chaser. We'll do one by hand here, but continuing this work is very tedious and time consuming. We'll need to get more computational help to perform more iteration. We begin by substituting 0 for *n* in Equation (8) to get

$$x_0(0+1) = x_0(0) + \frac{0.7(3+0.3(0) - x_0(0))}{\sqrt{(3+0.3(0) - x_0(0))^2 + (0.4(0) - y_0(0))^2}}$$

Then we simplify, substitute the initial conditions, and perform more simplifying to obtain:

$$x_0(1) = -3 + \frac{0.7(3-(-3))}{\sqrt{(3-(-3))^2}} = -3 + \frac{0.7(6)}{\sqrt{6^2}} = -3 + \frac{4.2}{6} = -2.3$$

We do the same for Equation (9), where many parts of the equation evaluate to 0 and disappear in the calculations to obtain:

$$y_0(1) = y_0(0) + \frac{0.7(0-0)}{\sqrt{(3-(-3))^2}} = 0 + 0 = 0$$

Therefore, the chaser moves to the point *(-2.3,0)* during the first timestep of the chase. We continue this procedure for *n = 1,2,3,...,10* to produce the iterates given in Table 1. We determined these values using a computer to perform all the tedious, but necessary computations.

n	$x_0(n)$	$y_0(n$
0	-3	0
1	-2.3	0
2	-1.6	0.05
3	-0.91	0.15
4	-0.23	0.30
5	0.45	0.50
6	1.10	0.74
7	1.74	1.03
8	2.36	1.35
9	2.96	1.72
10	3.54	2.11

Table 1. Iterates for Equations (8) and (9), providing the path for the chaser.

When do we stop our iteration? There is no need to continue after the chaser has caught the target. We need to refine our model to include a stopping criteria for the iteration that reflects "catching" the target. This does not mean that the location of the chaser and the target have to be exactly the same at the end of a time interval. This would be extremely difficult or impossible to achieve. Since we don't have a continuous function for location, we don't have an easy mechanism to check locations during the time interval. Therefore, we will assume that "catching" the target means just being "close enough" at the end of a timestep. First, we need to determine what is "close enough." If the chaser is an explosive munition with a large "kill radius," then we might use that radius to determine "close enough." If we need an impact of the chaser and the target, we may say "close enough" is a very small radius. We usually call this "close enough" distance or the radius of kill, the tolerance of the stopping criteria and denote it by e.

We have numerous choices for determining this tolerance value. It could be a fixed value, like the radius of kill. It could be a function of the speed s and time interval Δt. We know that in our discrete model the chaser moves a distance $s\Delta t$ over each timestep. Even more sophisticated models combining these two criteria and others are possible. For our example, we will use the distance $e = 0.5(s\Delta t)$. This mean that our iteration will stop when the chaser and the target are within half the distance traveled by the chaser in a timestep. Let's return to our problem to complete the calculations we started in Example 1.

Example 2: Revisit of Example 1 (Target moving in a straight line).

We implement stopping criteria in our model by determining the distance, denoted *d(n)*, between the chaser and target after each iteration. The value of *d(n)* is determined by the distance formula between two points,

$$d(n) = \sqrt{(x_0(n) - x_1(n))^2 + (y_0(n) - y_1(n))^2} \qquad (10)$$

In this example, we've decided to stop the iteration when $d(n) \le e = 0.5(s\Delta t)$. Since we are using $s=7$ and $\Delta t=0.1$, we have $e=0.35$. We redo our iteration, now showing $d(n)$, $x_1(n)$, and $y_1(n)$, in addition to n, $x_0(n)$, and $y_0(n)$. These new data are given in Table 2 with two decimal points of accuracy. As we see in Table 2, when $n = 24$, we achieve our stopping criteria since $d(24) = 0.26 < 0.3 = e$. The chaser has "caught" the target near the point *(10, 9.5).* The graphs of the actual paths of the chaser and the target are given in Figure 5.

n	$x_0(n)$	$y_0(n)$	$x_1(n)$	$y_1(n)$	$d(n)$
0	-3	0	3	0	6
1	-2.3	0	3.3	0.4	5.61
2	-1.60	0.05	3.6	0.8	5.26
3	-0.91	0.15	3.9	1.2	4.92
4	-0.23	0.30	4.2	1.6	4.61
5	0.45	0.50	4.5	2	4.32
6	1.10	0.74	4.8	2.4	4.05
7	1.74	1.03	5.1	2.8	3.80
8	2.36	1.35	5.4	3.2	3.56
9	2.96	1.72	5.7	3.6	3.33
10	3.54	2.11	6.0	4.0	3.10
11	4.09	2.54	6.3	4.4	2.89
12	4.63	2.99	6.6	4.8	2.68
13	5.14	3.46	6.9	5.2	2.47
14	5.64	3.96	7.2	5.6	2.27
15	6.12	4.46	7.5	6.0	2.06
16	6.59	4.99	7.8	6.4	1.86
17	7.04	5.52	8.1	6.8	1.66
18	7.49	6.06	8.4	7.2	1.46
19	7.93	6.60	8.7	7.6	1.26
20	8.36	7.16	9.0	8.0	1.06
21	8.78	7.71	9.3	8.4	0.86
22	9.20	8.27	9.6	8.8	0.66
23	9.62	8.83	9.9	9.2	0.46
24	10.04	9.39	10.2	9.6	0.26

Table 2. Iterates and distances between the paths for the chaser and target.

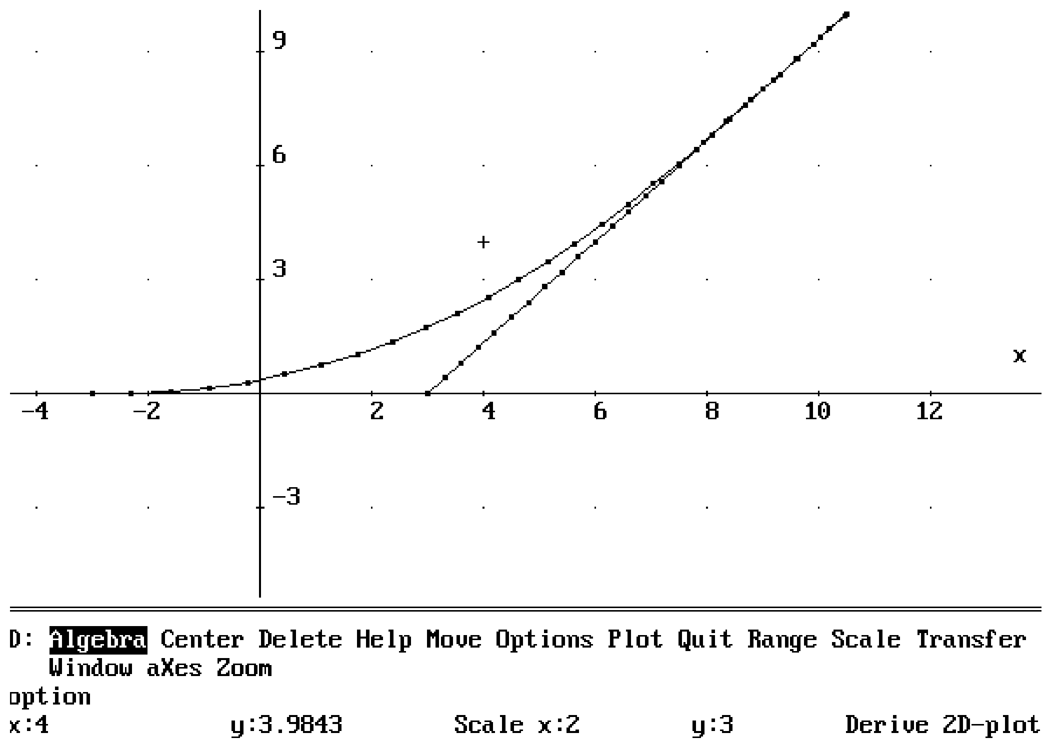


Figure 5. Graphs of the paths of the chaser and target, from the start of the chase to the "catch" at approximately the point *(10, 9.5).*

Does our solution make sense? Does the chaser move in an efficient path toward the target? Does the chaser stop when the stopping criteria is achieved? In general the answers to these questions are "yes". It appears we have a good model, but it may not be the best. More work will need to be done to determine better models.

Let's review our application of the modeling process to this chase problem. We started by understanding and analyzing our situation and needs of the problem. We defined our problem to be the determination of a path for the chaser, given a prescribed path for the target. We made assumptions and used the assumptions and our understanding of the process to build a model (the system of difference equations given by (8) and (9) and the stopping criteria established in (10)). Finally, we solved our model in Example 2 by iterating and plotting graphs of the paths of the chaser and target.

In our discussion of the model, we briefly mentioned the concepts of discrete and continuous models. These two concepts represent an important dichotomy in mathematics. We will view our modeling process with these two concepts in mind. Our behavior of interest, the movement of the chaser, is continuous in nature. At any value of time, the chaser's movement can be determined and the chaser moves smoothly, with no jumps or discontinuities. However, we modeled this movement as being discrete. We only determine the chaser's location at discrete values of time, Δt apart. Our solution, the sequence of location points given in Table 2, is also discrete. We finally convert the discrete sequence of the path to a continuous path by connecting the points in the graphs of Figure 5. We show this interplay between discrete and continuous representations in our modeling process in Figure 6.

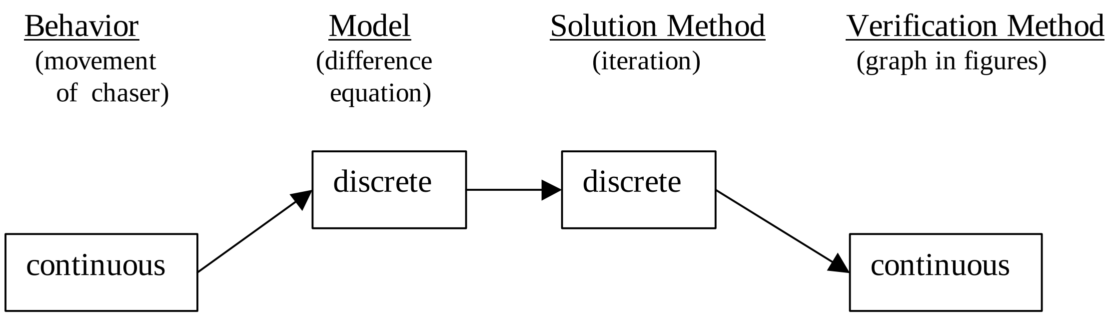


Figure 6. Interplay between discrete and continuous in the modeling process of the chase algorithm.

In general, this kind of interplay between discrete and continuous can occur in any phase of the modeling process. A schematic diagram of possible paths through the modeling process is given in Figure 7.

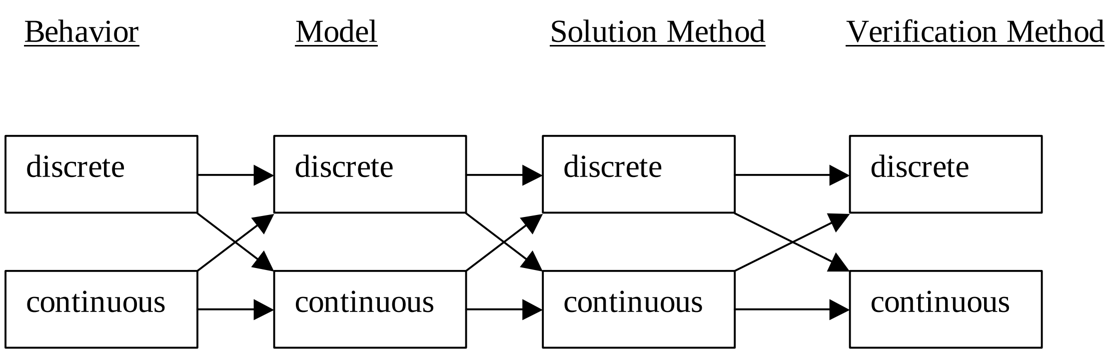


Figure 7. Interplay between discrete and continuous.

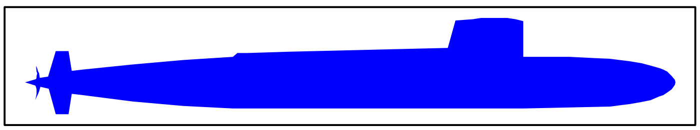
Let's do another example which shows more sophistication in the movements of the target and chaser. Let's see what happens when a tracking torpedo is launched at a ship.

Example 3: Torpedo attack.

An attack submarine located at point *(0,-10)* launches a tracking torpedo with speed 6 at a target ship at time $t = 0$ following a circular course given by the following parametric equations:

$$x_1(t) = -8\cos(0.5t) \quad \text{and} \quad y_1(t) = 8\sin(0.5t) \qquad (11)$$

We will determine the torpedo's path based on its seeker using our tracking model of moving directly toward the target ship over each time interval of length $\Delta t=0.2$. We assume the ship maintains its given course until the torpedo is detected at a distance of 3 units from the ship. Therefore, our stopping criteria for the first phase of the torpedo movement is $e=3$.



First, let's determine the ship's speed. This is calculated by finding the magnitude of the velocity vector in the two component directions, *x* and *y*. The formula for this calculation uses the derivatives of our two parametric equations in (11). The general formula for speed, denoted by $s_1(t)$, is $s_1(t) = \sqrt{(x_1')^2 + (y_1')^2}$. Taking the derivatives in Equation (11), we get

$$x_1'(t) = 4\sin(0.5t) \quad \text{and} \quad y_1'(t) = 4\cos(0.5t) \ . \qquad (12)$$

Substituting into our speed formula gives the ship's speed as

$$s(t) = \sqrt{(4\sin(0.5t))^2 + (4\cos(0.5t))^2} = \sqrt{16(\cos^2(0.5t) + \sin^2(0.5t))} = \sqrt{16} = 4 \,.$$

This means that the ship is traveling in a circular arc at a constant speed of 4, which is slower than the torpedo traveling at a speed of 6. If this situation continues, the torpedo will definitely be able to catch the ship and impact on its hull. Substituting the know values and functions into our model, Equations (4) and (5), produces

$$x_0(n+1) = x_0(n) + \frac{6(0.2)(-8\cos(0.5(0.2)n) - x_0(n))}{\sqrt{(-8\cos(0.5(0.2)n) - x_0(n))^2 + (8\sin(0.5(0.2)n) - y_0(n))^2}} \qquad (13)$$

$$y_0(n+1) = y_0(n) + \frac{6(0.2)(8\sin(0.5(0.2)n - y_0(n))}{\sqrt{(-8\cos(0.5(0.2)n) - x_0(n))^2 + (8\sin(0.5(0.2)n) - y_0(n))^2}} \qquad (14)$$

Starting with our initial condition, $x_0(0) = 0$ *and* $y_0(0) = -10$, we iterate (13) and (14) until we achieve our stopping criteria $d(n) < e = 3$. This produces the solution in the form of the sequence of points given in Table 3. We also show the target location $(x_1(n), y_1(n))$ and $d(n)$ in Table 3.

n	$x_0(n)$	$y_0(n)$	$x_1(n)$	$y_1(n)$	$d(n)$
0	0	-10	-8	0	12.81
1	-0.75	-9.06	-7.96	0.80	12.22
2	-1.46	-8.09	-7.84	1.59	11.60
3	-2.19	-7.09	-7.64	2.36	10.95
4	-2.72	-6.05	-7.37	3.12	10.28
5	-3.27	-4.99	-7.02	3.84	9.59
6	-3.74	-3.88	-6.60	4.52	8.87
7	-4.12	-2.75	-6.12	5.15	8.15
8	-4.42	-1.58	-5.57	5.74	7.41
9	-4.60	-0.40	-4.97	6.27	6.67
10	-4.67	0.80	-4.32	6.73	5.94
11	-4.60	2.00	-3.63	7.12	5.22
12	-4.38	3.18	-2.90	7.46	4.53
13	-3.99	4.31	-2.14	7.71	3.87
14	-3.41	5.37	-1.36	7.88	3.25
15	-2.65	6.30	-0.56	7.98	2.68

Table 3. Iterates of the paths of the torpedo and the ship and the distance between them, $d(n)$.

The graphs of the paths of the torpedo and ship for the first 15 timesteps are given in Figure 8. After 15 timesteps of length $\Delta t=0.2$ $(t=3)$, the torpedo is only 2.68 units from the ship.

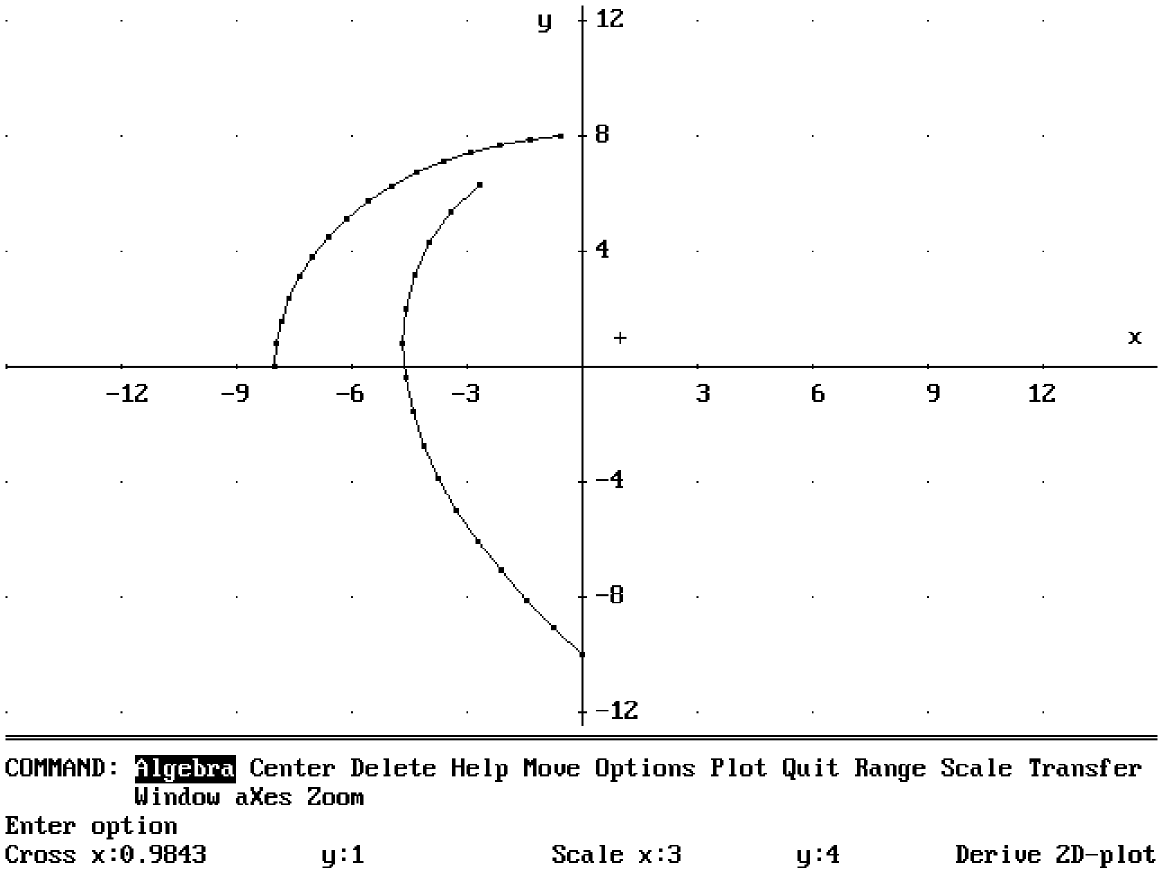


Figure 8. Graph of the paths of the torpedo and the ship, from launch (*t=0*) to detection of the torpedo at *t=3*.

In this case, the ship's evasive maneuver when it detects a torpedo, is to move directly away from the torpedo at the ship's maximum speed of 5. If we assume this maneuver is possible during the next timestep, we need to determine the parametric equations for the ship's new path, starting at *n=15* and $t=n\Delta t = 15(0.2)=3$. The data in Table 3 show the ship *at (-0.57, 7.98)* and the torpedo at *(-2.65, 6.30).* The direction vector between the two is *(-0.57-(-2.65), 7.98-6.30) = (2.08, 1.68).* The unit vector in that direction is $\dfrac{(2.08,1.68)}{\sqrt{2.08^2+1.68^2}} = \dfrac{(2.08,1.68)}{2.6737} = (0.778,0.628)$. Using the point (-0.57,7.98) as the start point when $t = 3$, we can use the point-direction parametric form of a straight line to represent the evasive path of the ship. Therefore, the new parametric equations for the ship's movement are written as

$$x_1(t) = -0.57 + 5(0.778)(t-3) \quad \text{and} \quad y_1(t) = 7.98 + 5(0.628)(t-3). \qquad (15)$$

Now writing the equations in (15) in terms of the discrete independent variable *n*, we obtain the new formulas:

$x_1(n) = -0.57 + 5(0.778)(0.2n - 3)$ and $y_1(n) = 7.98 + 5(0.628)(0.2n - 3)$.

These can be simplified to the forms

$$x_1(n) = 0.778n - 12.24 \quad \text{and} \quad y_1(n) = 0.628n - 1.44.$$

Assuming a kill radius for the torpedo of 0.1, we now establish a new stopping criteria and new target path for the second phase of the chase and continue our iterating at *n = 15*. The results of these iterations for the evasive movement of the ship and pursuit by the torpedo are given in Table 4. According to our calculations, the ship would be hit by the torpedo at *n=28* or *t=5.6,* unless this distance exceeds he torpedo's effective range.

n	$x_0(n)$	$y_0(n)$	$x_1(n)$	$y_1(n)$	$d(n)$
15	-2.65	6.30	-0.56	7.98	2.68
16	-1.72	7.05	0.21	8.61	2.47
17	-0.78	7.81	0.99	9.23	2.27
18	0.15	8.56	1.76	9.86	2.07
19	1.08	9.31	2.54	10.49	1.87
20	2.01	10.06	3.32	11.12	1.67
21	2.95	10.82	4.09	11.75	1.47
22	3.88	11.57	4.87	12.37	1.27
23	4.82	12.33	5.65	13.00	1.07
24	5.75	13.08	6.43	13.63	0.87
25	6.69	13.84	7.21	14.26	0.67
26	7.62	14.59	7.99	14.89	0.47
27	8.55	15.35	8.77	15.51	0.27
28	9.49	16.10	9.54	16.14	0.07

Table 4. Iterates of the paths of torpedo and ship during the evasive movement phase of the ship.

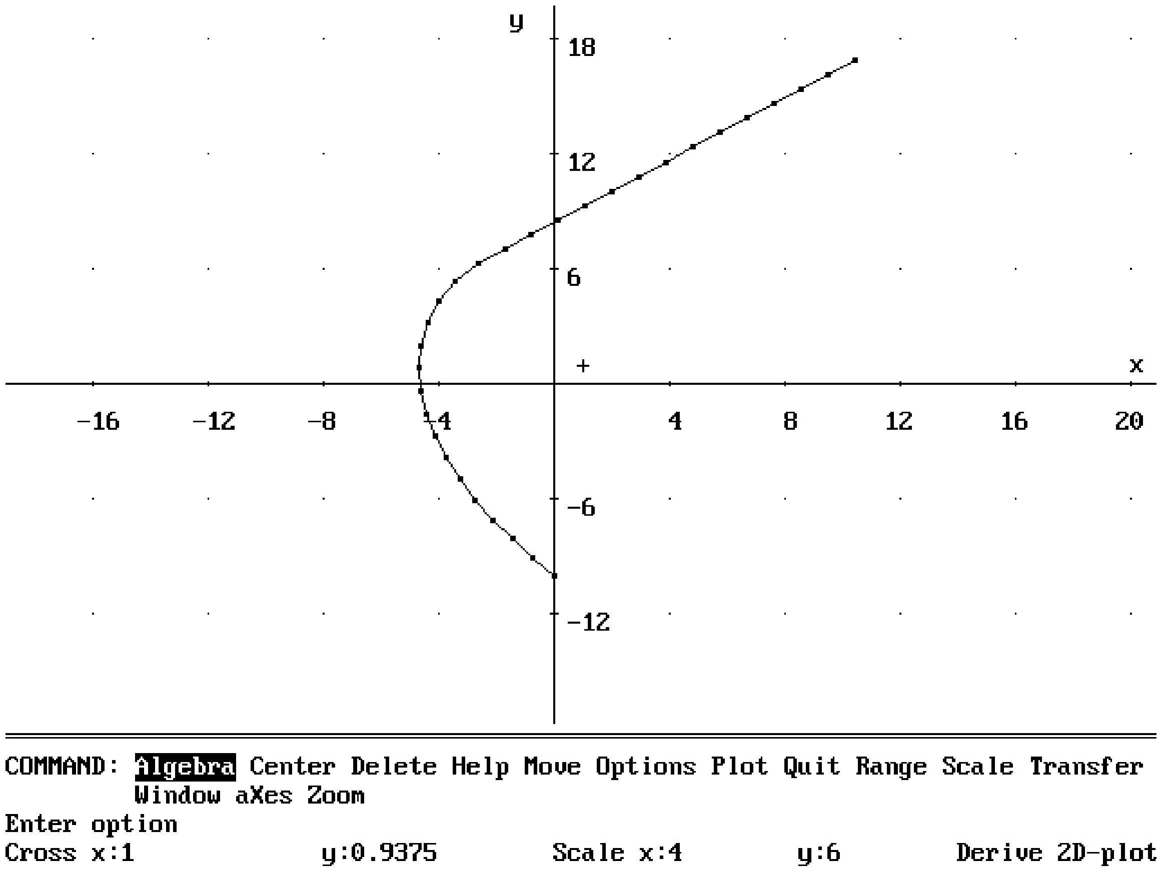


Figure 9. Graph of the path of the torpedo from launch (*t=0*) to impact with the ship at *t=5.6.*

3-Dimensional Model

We now extend our model to three dimensions so we can use it in situations where all three dimensions are significant, like the aerial pursuit of missiles after planes and other missiles. The extension is quite simple. The hypotenuse of the triangle now lies in three dimensions and can be projected onto the three coordinate planes. If we use z for our third coordinate (height above the xy-plane), the chaser's coordinates in the discrete variable become ($x_0(n)$, $y_0(n,)$,$z_0(n)$) and the target's coordinates are ($x_1(n)$, $y_1(n)$,$z_1(n)$). The hypotenuse of the triangle has length

$$\sqrt{(x_1(n)-x_0(n))^2+(y_1(n)-y_0(n))^2+(z_1(n)-z_0(n))^2}\ .$$

We make the necessary changes to Equations (4) and (5) and add our third equation to get the following 3-dimensional model for movement of the chaser:

$$x_0(n+1) = x_0(n) + \frac{s\Delta t(x_1(n) - x_0(n))}{\sqrt{(x_1(n) - x_0(n))^2 + (y_1(n) - y_0(n))^2 + (z_1(n) - z_0(n))^2}} \qquad (16)$$

$$y_0(n+1) = y_0(n) + \frac{s\Delta t(y_1(n) - y_0(n))}{\sqrt{(x_1(n) - x_0(n))^2 + (y_1(n) - y_0(n))^2 + (z_1(n) - z_0(n))^2}}$$

(17)

$$z_0(n+1) = z_0(n) + \frac{s\Delta t(z_1(n) - z_0(n))}{\sqrt{(x_1(n) - x_0(n))^2 + (y_1(n) - y_0(n))^2 + (z_1(n) - z_0(n))^2}} \qquad (18)$$

Our stopping criteria is achieved by iterating until

$$d(n) = \sqrt{(x_0(n) - x_1(n))^2 + (y_0(n) - y_1(n))^2 + (z_1(n) - z_0(n))^2} \leq e \qquad (19)$$

Example 4. Patriot intercepts Scud

A Scud missile is launched from location (0,0,0) at its intended target at point (150,200,0), with a trajectory given by $x_1(t) = 4t$, $y_1(t) = 3t$, and $z_1(t) = 10t - 0.2t^2$.

A Patriot missile with constant speed 15 is launched from coordinates *(100,0,*0) at *t=15* (after the launch of the Scud) and tracks the Scud using our chase model with a timestep of $\Delta t = 0.5$. It takes the Patriot tracking radars this much time to identify and lock in on the Scud. The Patriot has a kill radius of 2, but only safely destroys the target if the impact occurs at least 20 units high above the ground *(z>20).* Substituting the values and functions into Equations (16-19) produces the following system of difference equations:

$$x_0(n+1) = x_0(n) + \frac{7.5(2n - x_0(n))}{\sqrt{(2n - x_0(n))^2 + (1.5n - y_0(n))^2 + (5n - 0.05n^2 - z_0(n))^2}} \qquad (20)$$

$$y_0(n+1) = y_0(n) + \frac{7.5(1.5n - y_0(n))}{\sqrt{(2n - x_0(n))^2 + (1.5n - y_0(n))^2 + (5n - 0.05n^2 - z_0(n))^2}} \qquad (21)$$

$$z_0(n+1) = z_0(n) + \frac{7.5(5n - 0.05n^2 - z_0(n))}{\sqrt{(2n - x_0(n))^2 + (1.5n - y_0(n))^2 + (5n - 0.05n^2 - z_0(n))^2}} \qquad (22)$$

Iterating Equations (20-22), starting with *n=30 (t=15),* produces the paths for the Patriot and Scud given in Table 5. The distances between the Patriot and Scud are determined by Equation (19) and are also given in Table 5. The Scud is destroyed at *t=25.5 (n=51)* at location *(102,77,125),* when the Patriot closes to within 1.6 units of the Scud. The intercept is much higher than 20 units (*z=125>20*), so it is a safe and effective intercept.

n	$x_0(n)$	$y_0(n)$	$z_0(n)$	$x_1(n)$	$y_1(n)$	$z_1(n)$	$d(n)$
30	100	0	0	60	45	105	121.0
31	97.5	2.8	6.5	62	46.5	107	115.2
32	95.2	5.6	13.0	64	48	108.8	109.3
33	93.1	8.5	19.6	66	49.5	110.6	103.3
34	91.1	11.5	26.2	68	51	112.2	97.4
35	89.3	14.6	32.8	70	52.5	113.8	91.4
36	87.7	17.6	39.5	72	54	115.2	85.4
37	86.4	20.9	46.1	74	55.5	116.6	79.5
38	85.2	24.1	52.8	76	57	117.8	73.4
39	84.2	27.5	59.4	78	58.5	119.0	67.4
40	83.5	30.9	66.0	80	60	120	61.4
41	83.1	34.5	72.6	82	61.5	121	55.4
42	82.9	38.1	79.2	84	63	121.8	49.4
43	83.1	34.5	72.6	86	64.5	122.6	43.4
44	83.6	45.8	92.0	88	66	123.2	37.4
45	84.5	49.8	98.3	90	67.5	123.8	31.5
46	85.8	54.1	104.4	92	69	124.2	25.6
47	87.6	58.4	110.0	94	70.5	124.6	19.8
48	90.0	63.0	115.6	96	72	124.8	14.2
49	93.2	67.8	120.5	98	73.5	125.0	8.7
50	97.3	72.7	124.3	100	75	125.0	3.6
51	102.9	77.5	125.7	102	76.5	125.0	1.6

Table 5. Iterates for the paths of the Patriot and the Scud, from the time of the launch of the Patriot (*n=30*).

The space curves showing the paths of the Patriot and Scud are given in Figure 10.

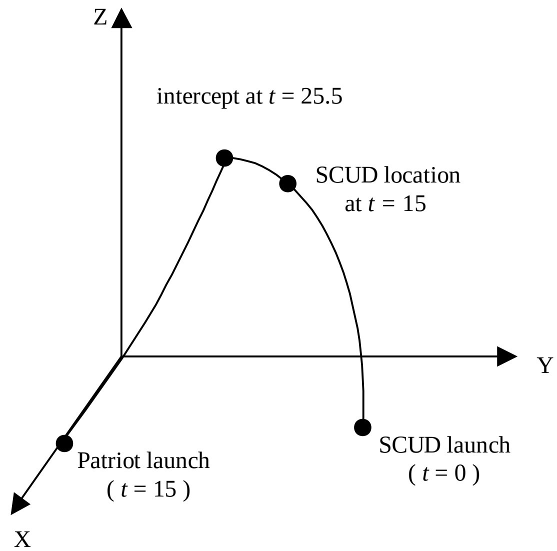
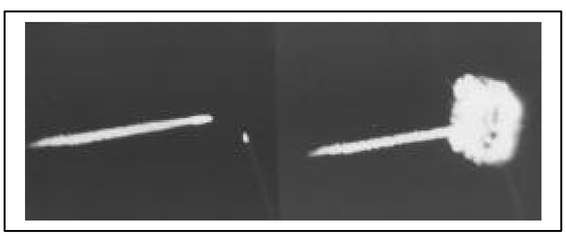


Figure 10. Paths of Patriot and Scud from launch to impact.

In this section, we have studied and solved a challenging problem with many applications. Our model and its solutions have performed well in the examples we have solved. There are several obvious questions we have not addressed. Some of these are: Will the chaser always catch the target? What happens when the target is faster than the chaser? What are the effects of changing parameters like Δt, e, and s. We could go on-and-on asking more-and-more questions. That's what makes this such a challenging and interesting problem. There are other challenges in our model. We always need to question our assumptions. What would happen if we lead our target, instead of moving directly toward it? What about the maneuverability of the chaser? Can it always turn fast enough to make the necessary move in the next timestep Δt? How should the target move to better evade the chaser? In later sections we refine our model to address some of these considerations.

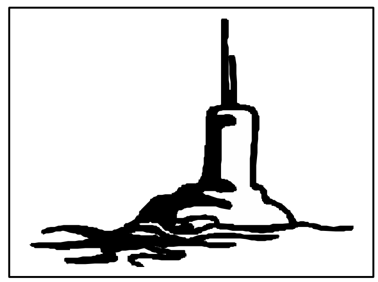


<u>Exercises</u>

1. An enemy tank, currently at location (10,0), is moving in a zigzag pattern away from your location with parametric equations: $x_1(t) = 12 + 3t$ and $y_1(t) = 2\sin(2t)$. You launch a tank tracking round moving at a speed of 12 from your location at (0,0). The guidance system of the radar-controlled round uses a timestep of $\Delta t = 0.25$ and always moves directly toward the tank target.

a) Write a system of difference equations in terms of the discrete variable *n* that models the movement of the round towards the target.
b) Iterate the equations to find the first 4 positions of the tank round and determine the location of the round at $t = 0$.
c) What is the distance between the tank and the round at $t = 1$? Is the round closer at $t = 1$ or $t = 0$?
d) Based on the findings in part (c), do you think the round will catch and impact with the tank? Conjecture an impact time and location based on the relative speed of the round and tank (do not perform the iterations).
e) If you have adequate computing resources, iterate your model to determine impact time and location.

2. A ship located at (25,10) detects a torpedo at (15,6) and begins the evasive maneuver of moving directly away from the torpedo at a constant speed of 8.
a) What are the parametric equations, using time t as the parameter, for the path of the ship with t=0 representing the start time of the path?
b) If the torpedo follows the ship with a speed of 10, how long will it take for the torpedo to catch the ship?



3. Write the parametric equations for the motion of a reconnaissance plane that flies in the path of a circular helix as shown in Figure 11. Use the variable representation's shown for x, y, and z. The circles have a radius of 5 and the plane has an upward speed (vertical or z direction) of 2. The start point (t=0) is the point (5,0,0). Is your solution the only possible solution or are other solutions possible?

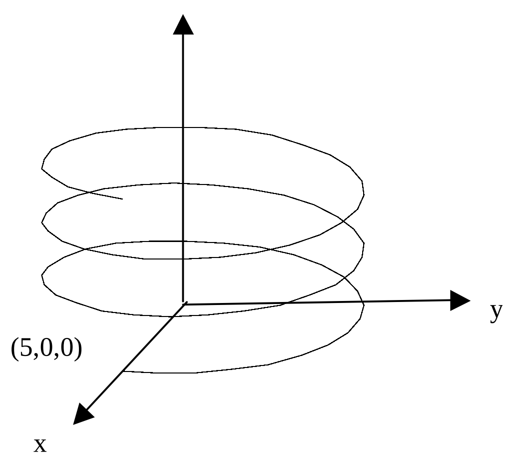


Figure 11. Circular helix for Question 3.

4. Is it ever possible for a slow chaser to “catch” a faster target? What is the difference in the chase when the target also has its own goal to achieve (i.e. Scud heading for its own target, football ball carrier trying to make a touchdown or firstdown) and when the target’s only goal is to evade the chaser (i.e. ship running from a torpedo, cat running from a dog)?

5. Explain the steps of the mathematical modeling process in your own words. What step is the most important to solve a problem successfully?

6. Discuss the dichotomy of discrete and continuous mathematics. Include in your discussion examples of behaviors and functions that are naturally discrete and behaviors and functions that are naturally continuous.