

SIGNALS AND SYSTEMS LABORATORY 7: Feedback Control Systems and the Phase Plane

INTRODUCTION

Feedback systems have one defining characteristic: the input depends on the output of the system, or more generally on the state of the system. It may depend on other things as well, such as an input signal coming in externally, but there must be the *feedback path* from the output back to the input. The use of feedback changes the dynamics of the system, and can have many benefits. The classic work of the Bell laboratories engineer H. W. Bode (rhymes with Cody) showed that feedback can make a system much more precise in its response. It can also stabilize an otherwise unstable system. But it can also do the opposite. It can also destabilize an otherwise stable system. In this lab, we shall consider a classical feedback loop with variable loop gain, and then study the phase plane for two-dimensional linear state variable systems. This will be preparation for the next lab, which treats examples of state feedback control systems.

ROOT LOCUS AND STEP RESPONSE

All of the control systems that we are going to consider are *causal*, so the region of convergence for the Laplace transform will always be to the right of the imaginary axis of the complex plane. In practice, this is important since anti-causal systems tend to become unstable very easily.

Consider a continuous time linear system with transfer function $H(s) = B(s)/A(s)$. The numerator and denominator are polynomials in s . This system is *causal* and *stable* if and only if the roots of the denominator polynomial $A(s)$ are in the left half-plane, which is to say that they must all have *negative real parts*. Now suppose that this system is imbedded in the feedback system shown below in Figure One.

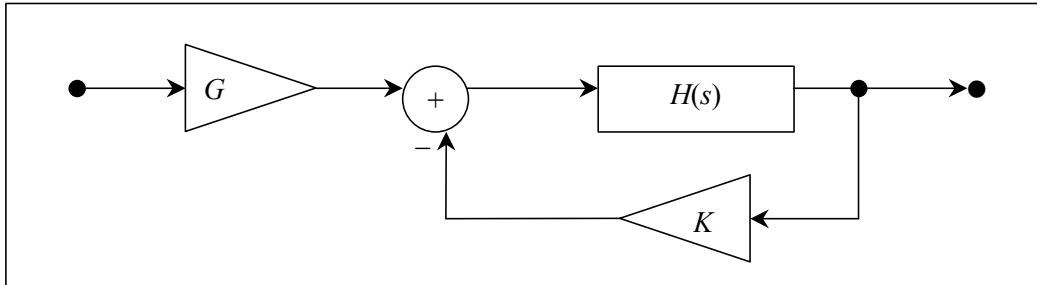


Figure One A Feedback System

The transfer function of the closed loop system is then

$$(1) \quad G(s) = \frac{GH(s)}{1 + KH(s)} = \left[\frac{1 + KH(0)}{H(0)} \right] \frac{H(s)}{1 + KH(s)}.$$

We have chosen the amplifier gain $G = [1 + KH(0)]/H(0)$ so that the DC gain of $G(s)$ is $G(0) = 1$. This is also the steady state value of the step response provided the system is stable. This is for convenience only, since we are going to compare the step responses for several choices of $G(s)$, as the feedback amplifier gain K is varied. Since $H(s)$ is the ratio of two polynomials, so is $G(s)$. A little algebra reduces equation (1) to

$$(2) \quad G(s) = \frac{GB(s)}{A(s) + KB(s)}$$

Whether or not the closed loop transfer function $G(s)$ is stable depends on the locations of the roots of the polynomial $A(s) + KB(s)$, and these roots will move as K is changed. The resulting graph of the set of roots of $A(s) + KB(s)$ is called a *root locus diagram*. These diagrams were once constructed with much labor, using a complicated set of rules together with a special device (a *spirule*), which was a combination of a protractor and a slide rule. These are now found only in museums. Using MATLAB, the root locus diagram is easily obtained. Consult the 'help' facility for the built-in MATLAB function 'rlocus', to see how. Experiment with some choices of $B(s)$ and $A(s)$. For example, try

```
»b=conv([1 1],[1 3]);a=conv([1 2 2],[1 3 9]);a=conv(a,[1 5]);
»rlocus(b,a)
```

You should see the following:

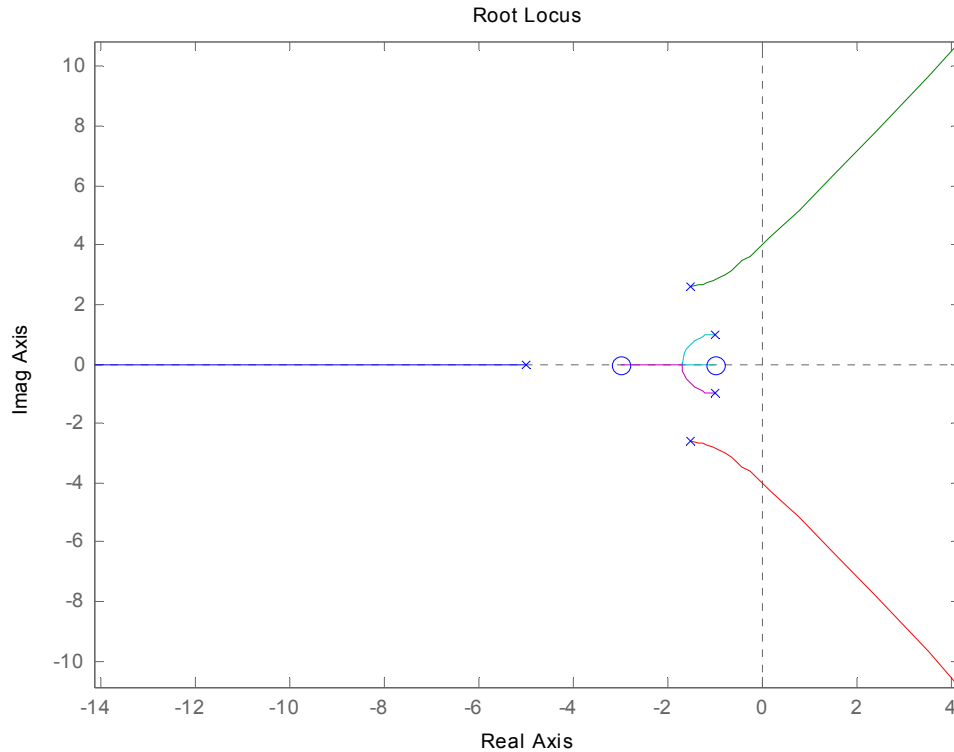


Figure Two Output of 'rlocus'

In this example, there are two zeros and five poles. The zeros attract two of the poles as K approaches infinity. The remaining three poles head off to infinity at angles asymptotically equal to multiples of $2\pi/3$. As K increases, the poles move along the path of the lines shown to either a finite zero or a zero at infinity asymptotically. This behavior follows the rules of root loci. It is quite common for the root locus diagram to cut across the imaginary axis. When this happens, the system $G(s)$ will become unstable. It is therefore important to know the range of values of K for which the system is stable.

ROUTH-HURWITZ STABILITY CRITERION

For a root locus plot such as that shown in Figure Two, there will be a range of K for which the closed-loop system is stable (all of the poles are in the left-half plane). A very effective way to determine this range of K is to use the Routh-Hurwitz stability criterion. This is done by taking the denominator of the closed loop transfer function in equation (2) and write it in the form:

$$(3) \quad D(s) = A(s) + K \cdot B(s) = a_n s^n + a_{n-1} s^{n-1} + \dots + a_1 s + a_0$$

Take these coefficients $a_n, a_{n-1}, \dots, a_1, a_0$ and put them into the Routh array according to the following pattern:

s^n	a_n	a_{n-2}	a_{n-4}	a_{n-6}	...
s^{n-1}	a_{n-1}	a_{n-3}	a_{n-5}	a_{n-7}	...
s^{n-2}	b_1	b_2	b_3	b_4	...
s^{n-3}	c_1	c_2	c_3	c_4	...
\vdots	\vdots	\vdots			
s^2	x_1	x_2			
s^1	y_1				
s^0	z_1				

Starting at the top and working your way down, the coefficients $b_1, b_2, b_3, \dots, c_1, c_2, c_3, \dots, x_1, x_2, y_1, z_1$, etc. may be calculated by taking the determinants of two-by-two matrices produced from the previous two rows. Each two-by-two matrix is made by taking the two elements in the leftmost column and the two elements in the column directly to right of the coefficient in question of the two rows directly above the coefficient in question. The determinant of the two-by-two matrix is then divided by the element in the leftmost column of the row directly above the element in question and finally the whole thing is negated. This is more easily seen in equation (4) below.

$$(4) \quad b_1 = -\frac{1}{a_{n-1}} \begin{vmatrix} a_n & a_{n-2} \\ a_{n-1} & a_{n-3} \end{vmatrix} \quad b_2 = -\frac{1}{a_{n-1}} \begin{vmatrix} a_n & a_{n-4} \\ a_{n-1} & a_{n-5} \end{vmatrix} \quad b_3 = -\frac{1}{a_{n-1}} \begin{vmatrix} a_n & a_{n-6} \\ a_{n-1} & a_{n-7} \end{vmatrix}$$

$$c_1 = -\frac{1}{b_1} \begin{vmatrix} a_{n-1} & a_{n-3} \\ b_1 & b_2 \end{vmatrix} \quad c_2 = -\frac{1}{b_1} \begin{vmatrix} a_{n-1} & a_{n-5} \\ b_1 & b_3 \end{vmatrix} \quad c_3 = -\frac{1}{b_1} \begin{vmatrix} a_{n-1} & a_{n-7} \\ b_1 & b_4 \end{vmatrix}$$

This pattern continues until all of the coefficients are found. When a row does not contain enough elements to compute these determinants, assume the undefined elements in that row are zero. The number of sign changes in the first column determines the number of unstable poles the system has. For stability, all of the coefficients must be positive. When a variable feedback gain such as K is present, the Routh array can be used to find the range of K over which the system is stable.

Assignment

1. Consider the open-loop transfer function $H(s) = \frac{s + 2}{(s^2 + 2s + 2)(s^2 + 2s + 10)}$.
 - a. Use a Routh array to find the maximum gain K_{max} for which closed-loop system $G(s)$ is stable.
 - b. Use the homebrew MATLAB function '[steps.m](#)' located on the web page under 'Functions for Lab 7' to see how the parameter K affects the unit step response of $G(s)$. Use a value of T so that the transients are suitably spread out. Include a print of the graph produced in your report. Comment on the nature of the step response as K is increased. In particular, compare *rise time* and *overshoot*.
 - c. Find the poles of $G(s)$ when $K = K_{max}$.

THE PHASE PLANE

The state variable equations for linear systems have the form $\dot{x} = Ax + Bu$, where x is the state vector. In this section, we shall consider 2-dimensional systems that are in *free motion*. This is to say that the input is $u = 0$. The homogeneous differential equation is then $\dot{x} = Ax$, or

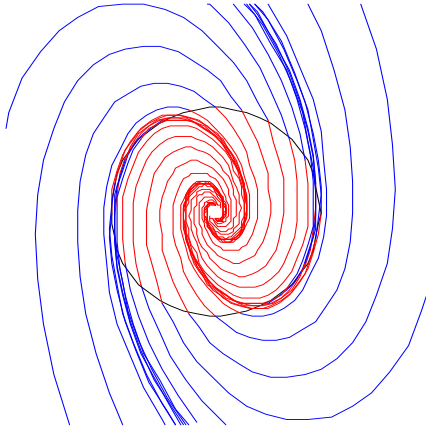
$$(5) \quad \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}.$$

The homogeneous solution is $x(t) = e^{At}x(0)$. When the system has dimension two, one can graph the trajectories as curves on the (x_1, x_2) plane. These pictures are often called *phase portraits*. The system need not be linear. In fact, phase plane portraits are a useful tool for two-dimensional non-linear differential equations as well. In Figure Three we see four examples of phase portraits. The horizontal axis is x_1 and the vertical axis is x_2 . Each curve represents a solution to the differential equation (5). (The circle is merely the unit circle and is there for reference only. It is not a trajectory.) One can classify these systems by the nature of the solutions in the vicinity of the *fixed point* $x = 0$. The classification depends on the *eigenvalues* of the matrix A , which are by definition the roots of the *characteristic polynomial*

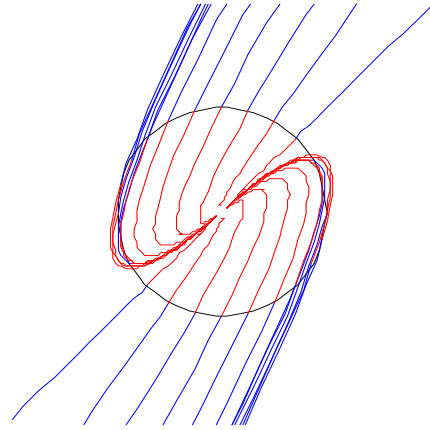
$$(6) \quad p(s) = \det(sI - A) = \det \begin{bmatrix} s - a_{11} & -a_{12} \\ -a_{21} & s - a_{22} \end{bmatrix} = s^2 - (a_{11} + a_{22})s + a_{11}a_{22} - a_{12}a_{21}.$$

Note here that I is the identity matrix: $I = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$.

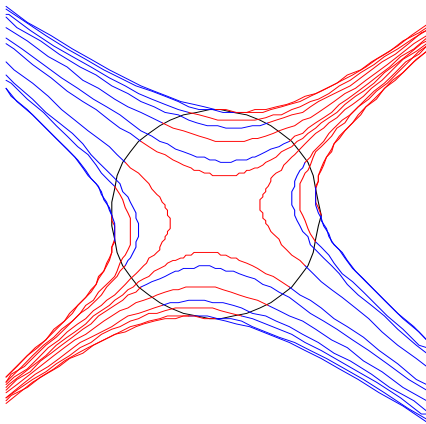
Spiral: 2 Complex Eigenvalues



Vortex: Repeated Eigenvalues



Saddle Point: 2 Real Eigenvalues
of Opposite Sign



Spider: 2 Negative Eigenvalues

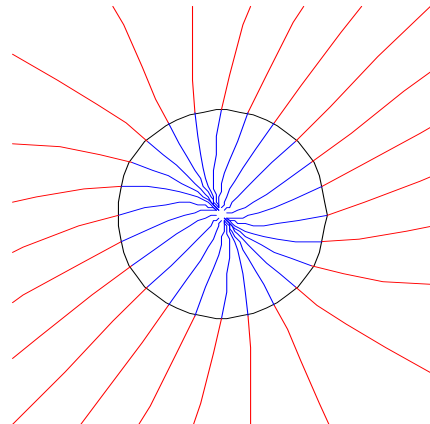


Figure Three Classification of fixed points in the phase plane

The behavior of the family of trajectories depends on the location of the eigenvalues. Complex eigenvalues produce a sinusoidal oscillation term in the matrix exponential and lead to spirals. Real eigenvalues lead to one or two solution *rays*. For every real eigenvalue λ , there is a real eigenvector ψ for which $A\psi = \lambda\psi$. The ray (or half line) which begins at zero and goes through the eigenvector ψ is a trajectory corresponding to the solution $x(t) = e^{\lambda(t-t_0)}\psi$ of the differential equation (5). Rays are trajectories only when they are aligned with an eigenvector. Consider the four types of rest points in Figure Three.

- i. *Spirals* occur when the eigenvalues of A are complex, i.e. have nonzero imaginary parts. The spiral will be towards the origin when the real part of λ is negative, and away from the origin when the real part is positive. When the real part is zero, the trajectories are closed, and become circles or ellipses. Since there are no real eigenvectors in this case, there will be no solution rays. In other words, no straight line is a trajectory.
- ii. *Saddle points* occur when both eigenvalues are real, but have opposite signs. There will be two solution rays: on one, trajectories head inward, and on the other, the trajectories head outward. Between these rays, trajectories head toward zero, turn, and then veer away.

- iii. *Nodes* occur when both eigenvalues are real, but have the same sign. If λ is negative, all trajectories will approach the origin, if λ is positive, all trajectories will head away from the origin. It may not be obvious in the lower right hand corner of Figure Three, but there are exactly two straight lines through the origin that contain trajectories. Can you see where they are?
- iv. *Repeated eigenvalues* are an odd in-between case. Here there will be exactly one straight line through the origin which contains trajectories. Such a case looks spiral-like off this ray. Trajectories bend towards the ray as they approach the origin.

The following MATLAB functions, 'randM.m' and 'traj.m', are located on the class web page under 'Functions for Lab 7'. We shall use them to provide some experience in classifying rest points.

```
% A=randM                                % traj(A)
% generates a random 2 by 2 matrix with    % plot trajectories of xdot = A*x,
% interesting phase portraits              % where A is 2x2
```

Assignment:

Learning to classify the phase portrait.

2. Open a graph window and resize and relocate it and the command window so that both can be seen. Then type in the following command:

```
»a=randM;traj(a);pause;eig(a)
```

The MATLAB function 'eig' will compute the two eigenvalues of the matrix. But the pause statement will delay this action until you hit a key. Before you hit the key, try to classify the eigenvalues from the picture. Then hit the return key to see if you are correct. You can use the up arrow to run another trial. The trajectories are color-coded. Each trajectory proceeds from red to blue. If the red part is near zero and the blue part is away from zero, the trajectory is unstable, and there will be at least one eigenvalue in the right halfplane. Classify according to the following list:

- i. Real, both positive but different (outward, two eigenvectors)
- ii. Real, repeated and positive (outward, but only one eigenvector)
- iii. Real, both negative but different (inward, two eigenvectors)
- iv. Real, repeated and negative (inward, but only one eigenvector)
- v. Real, one positive, one negative (saddle point)
- vi. Complex with negative real parts (inward spiral)
- vii. Complex with positive real parts (outward spiral)

When you feel you can do the classification without error, run this two more times and print out your results. Write your original guess and the eigenvalues on each print out.

Assignment:

3. Write a MATLAB function called 'vecfld.m' to draw a *vector field*. Actually, we want to draw a uniformly distributed set of little arrows on the *phase*, or (x_1, x_2) plane. The function has the following description:

```
% vecfld(A)
% draw a vector field, sampling
% uniformly in r,theta
```

The action of this function is shown in Figure Four. Once your program works, you should be able to get a similar picture by using the command

```
»a=randM;traj(a);vecfld(a)
```

The arrows should all have length $d = 0.15$. Each points in the direction that the state of the system is moving. At the top of your program, construct the 2 by 2 matrix $M = \begin{bmatrix} .5 & -.2 \\ .2 & .5 \end{bmatrix}$. Now suppose that we wish to plot a vector whose base is at x , a column vector of dimension two. The following statements will do it.

```
y=A*x;y=(d/sqrt(sum(y.*y)))*y;
X=[x,x+y,x+M*y,x+M'*y,x+y];
plot(X(1,:),X(2:,:),'b')
```

In this computation, the vector y starts off as the velocity vector $\dot{x} = Ax$, and is then normalized to have length d . The head of the vector is at $x+y$.

4. Consider the two matrices $\begin{bmatrix} 0.9e^{j\pi/4} & 0 \\ 0 & 0.9e^{-j\pi/4} \end{bmatrix}$ and $\begin{bmatrix} 0 & 1 \\ -0.81 & 1.8\cos(\pi/4) \end{bmatrix}$. Show the characteristic polynomial of each is $s^2 - 1.8\cos(\pi/4)s + 0.81$, whose roots are $s_1 = 0.9e^{j\pi/4} = s_2^*$. An infinite number of matrices share the same characteristic polynomial. Each of them may be written as $T^{-1}AT$, where T is nonsingular and A is, for example, $\begin{bmatrix} 0.9e^{j\pi/4} & 0 \\ 0 & 0.9e^{-j\pi/4} \end{bmatrix}$. Note that $T^{-1}AT$ is a coordinate transformation of the the matrix A corresponding to the coordinate transformation $y = Tx$. That is, if x satisfies the homogeneous equation $\dot{x} = Ax$, then $y = Tx$ satisfies the homogeneous equation $\dot{y} = TAT^{-1}y$.

Construct two 2-by-2 matrices whose characteristic polynomials are $p(s) = s^2 + 0.2s + 0.1$ and $q(s) = s^2 - 0.1s - 0.12$. There will be many solutions. Then, in the lab, for each of the two matrices make four phase portraits on the same page. If your matrix is called A , then generate the graphs as follows:

```
»for k=1:4;T=randn(2,2);subplot(2,2,k);traj(inv(T)*A*T);end
```

Note that the qualitative behavior of the phase portrait is independent of the coordinate system.

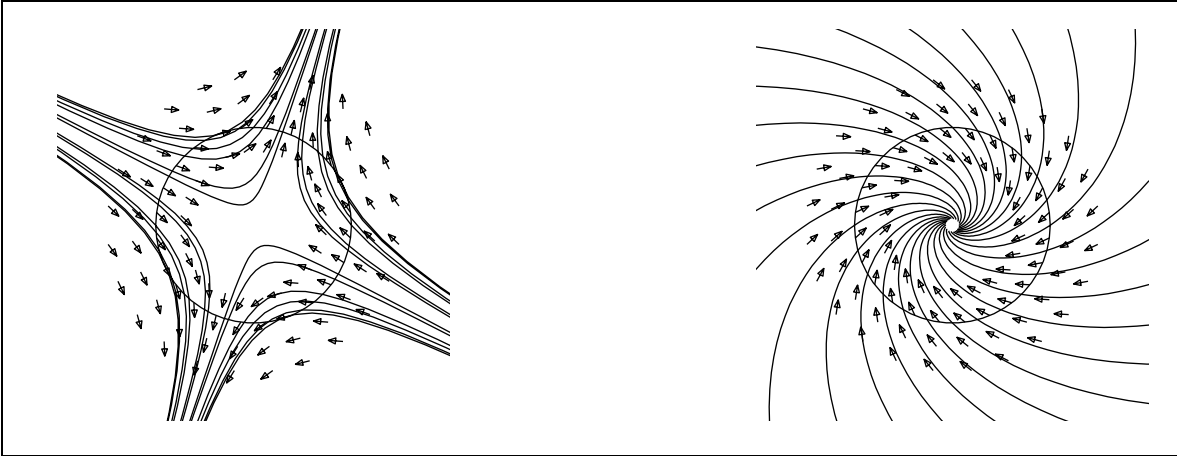


Figure Four Trajectories with vector field overlay

LOOP GAIN AND PHASE PORTRAITS

Consider a second order system with transfer function

$$(7) \quad H(s) = \frac{B(s)}{A(s)} = \frac{b_0s + b_1}{s^2 + a_1s + a_2} .$$

This system has several state variable realizations. A *canonical realization* is

$$(8) \quad \begin{cases} \dot{x} = Ax + Bu \\ y = Cx \end{cases}, \quad \text{where} \quad A = \begin{bmatrix} 0 & 1 \\ -a_2 & -a_1 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \quad C = [b_1 \quad b_0]$$

Suppose that this system is imbedded in the control loop of Figure One, and the input to the amplifier G is turned off. Then the input to the *open-loop* system $H(s)$ is

$$(9) \quad u = -Ky = -KCx .$$

If we substitute equation (9) into equation (8), we get the *closed-loop* state variable equation

$$(10) \quad \dot{x} = (A - BKC)x .$$

The use of output feedback has changed the open-loop matrix A to the closed-loop matrix $A - BKC$. The system poles have changed from the eigenvalues of A to the eigenvalues of $A - BKC$. Equivalently they have changed from the zeros of $A(s)$ to the zeros of $A(s) + KB(s)$. Thus we can study root locus diagrams and phase plane portraits together, at least for second order systems.

Assignment:

5. Show algebraically that the eigenvalues of $A - BKC$ are the zeros of $A(s) + KB(s)$, where $B(s)$ and $A(s)$ are the numerator and denominator polynomials of $H(s)$ in equation (7) and the matrices in $A - BKC$ are defined in equation (8).
6. Consider the open-loop system with transfer function

$$H(s) = \frac{s - 1}{s^2 + 2s + 17}.$$

This system is called *non-minimum-phase* because it has a zero in the right half plane. This zero will attract a pole in the root locus diagram, and therefore the closed-loop system will become unstable for some value of K . Solve for this value of K using a Routh array. Then using the tool '[steps.m](#)', plot some step responses for K ranging from 0 to the maximum value for stability. Then for the same values of K , plot phase portraits for $A - BKC$, using

```
»F=A-B*K*C;traj(F);eig(F)
```

Print the root locus, step responses, and phase portraits. Hand annotate the phase portraits, indicating what the closed-loop system poles (eigenvalues of $A - BKC$) are.