We have established the critical role of the roots of the characteristic equation in analyzing the behavior of the homogeneous solution of an LTI ODE. We will soon see that these roots are also critical in the particular solutions.

The roots of the characteristic equation are the most important information in dynamic systems.

For a CE with known numerical coefficients, we can find roots numerically in MATLAB, on our calculators, etc.

Example

Suppose we know that the characteristic equation for a system is

$$\lambda^4 + 10\lambda^3 + 35\lambda^2 + 50\lambda + 24 = 0$$

Using a root-finding function such as **ROOTS**, we can easily find the roots:

$$\lambda_{1,2,3,4} = -1, -2, -3, -4$$

and from these roots, we can easily determine that the system is stable, has a time constant of 1 second, a settling time of 4 seconds, and has purely exponential behavior (no sinusoidal behavior since the roots are not complex).

But now suppose that the characteristic equation comes from a system still under design, with a design parameter C that has not yet been selected, and the characteristic equation is still in the form of, say,

$$\lambda^4 + 10\lambda^3 + C\lambda^2 + 50\lambda + 24 = 0$$

We cannot calculate the roots of the CE unless we first assign a number to C. Now let's assume that the design problem is such that we need to find a value of C to ensure that the system is stable and has a settling time of no more than 2 seconds. Let's add the additional criterion that C is expensive, so if multiple C's exist that satisfy the settling time criterion, we would like to find the minimum C possible. How should we proceed?

We could guess numerical values of C, then calculate the corresponding roots using MATLAB **ROOTS** or similar, and (maybe) (someday) we might actually find a value of C that satisfies the design specifications. This search can even be automated using a loop structure in a program such as MATLAB. But this approach has many potential drawbacks:

1. We don't actually know if any such a C even exists, so
   (a) We could search for a long time, yet never find one that works, then finally conclude that it's hopeless, so we give up, even though there may be a value that works just fine
   (b) Or, we could spend forever on a hopeless search for a solution that really doesn't exist
2. Or, there may be multiple values of C that could work (entire ranges of values, perhaps)
   (a) We might be able to gain a huge design advantage (cost, in this example) by choosing wisely from among these values, except that we have no way of knowing whether or not they even exist, so we can never know whether or not we found them all
   (b) In most cases, we may simply stop looking after we find the first value that works, because, after all, we have no way to know whether or not additional values exist

Routh-Hurwitz

- Routh-Hurwitz refers to a powerful method used to analyze the *real* parts of the roots of characteristic equations. It does not provide information about the imaginary parts.
- In its basic form, Routh-Hurwitz can be used to determine whether or not a system is stable, **without actually calculating any of its roots**. To be precise, it can actually tell us exactly how many of the system's roots (if any) are unstable or marginally stable.
- This capability may be utilized in selecting any unknown parameters that appear in a characteristic equation. **Routh-Hurwitz enables us to prove whether or not any values for the unknown parameters exist that can produce a stable set of roots**, thereby saving us from hopeless design searches; and moreover, if such values do exist, Routh-Hurwitz provides the necessary and sufficient conditions on the unknown parameters that would ensure system stability, saving us from potentially unlimited time and effort spent searching for such values in a trial-and-error fashion. It's that good.
- Routh-Hurwitz may also be used to evaluate the settling time (speed of response) of a system whose characteristic equation is known, *without* actually calculating the roots.
- And, this latter capability may be utilized to determine whether or not values exist for any unknown parameters, such that the resulting system will have a specified minimum speed of response; and, if such values exist, Routh-Hurwitz provides the necessary and sufficient conditions that the parameters must satisfy.

In the example given previously, where the CE is

$$\lambda^4 + 10\lambda^3 + C\lambda^2 + 50\lambda + 24 = 0$$

we were asked to find the minimum C that will give us a stable system with a settling time no greater than 2 seconds. We cannot find the actual roots of this equation until we assign a specific numerical value of C, but using Routh-Hurwitz, we can do all of the following:

- Determine whether or not any C's exist that make the system stable; and if so, we can also derive algebraic equations that can be used to find such values (or, often, entire ranges of values). If no such values exist, we can prove it, and thus not waste additional time trying to finalize a design that cannot work.
- Determine whether or not any C's exist that make the settling time less than 2 seconds; and if so, we can also derive algebraic equations that can be used to find such values (or, often, entire ranges of values). If no such values exist, we can prove it, and thus not waste additional time trying to finalize a design that cannot work.

Necessary Condition: Suppose that we have an n^{th}-order characteristic equation in the form

$$a_n\lambda^n + a_{n-1}\lambda^{n-1} + ... + a_1\lambda + a_0 = 0 \tag{1}$$

where all a_i's are real.

As we know, an LTI ODE system is stable if, and only if, all of the roots of the characteristic equation have negative real parts.

Theorem: If the coefficients a_i in Eq. (1) do not *all* have the same sign (i.e., if they are not either all positive or all negative), then the roots of Eq. (1) cannot all have negative real parts.

We do not prove this theorem, but it is true and provable.

Consider, for example, our standard form

$$\lambda^2 + 2\zeta\omega_n\lambda + \omega_n^2 = 0 \qquad \Rightarrow \lambda_{1,2} = -\zeta\omega_n \pm \omega_n\sqrt{\zeta^2 - 1}$$

First, note that

$$|\zeta\omega_n| > |\omega_n\sqrt{\zeta^2 - 1}|$$

Therefore, the sign of both roots is determined by the sign of $(-\zeta\omega_n)$.

If ζ and ω_n have opposite signs, then $-\zeta\omega_n$ is positive, so both roots are positive. However, if ζ and ω_n have the same sign, whether it's positive or negative, then both roots are negative.

The theorem applies to the coefficients in the characteristic equation. The first one is $(+)$ (equal to one); the second one is $(+)$ if ζ and ω_n have the same sign; and the third one is $(+)$ (ω_n^2).

Therefore, the coefficients of the polynomial are all $(+)$ if ζ and ω_n have the same sign; and the roots are both negative. If ζ and ω_n have opposite signs, the second coefficient is $(-)$ and the roots are positive.

As a practical matter, we want all of our roots to have negative real parts; therefore, we reject any system whose characteristic equation does NOT have coefficients whose signs are all the same. If any of the coefficients in Eq. (1) contain unknown parameters that need to be selected as part of a design process, this theorem provides the first condition for their selection. Note that a coefficient of zero (i.e., a missing power in the characteristic equation) has no sign, but the theorem requires that all coefficients have the same sign. Therefore, if we have a missing power term in the characteristic equation, the system cannot be stable.

This theorem should always be utilized as the first condition to be satisfied in any design, but it is only a *necessary* condition; it is not sufficient to ensure that all roots have negative real parts.

Sufficient Conditions: Routh-Hurwitz

Assuming that all a_i have the same sign, we may use Routh-Hurwitz to see whether or not all roots actually do have negative real parts. The first step in this process is to construct the so-called *Routh Table* as follows:

$$\begin{pmatrix}
a_n & a_{n-2} & a_{n-4} & ... & \leftarrow \text{last entry is } a_1 \text{ if } n \text{ is odd , or } a_0 \text{ if } n \text{ is even} \\
a_{n-1} & a_{n-3} & a_{n-5} & ... & \leftarrow \text{last entry is } a_0 \text{ if } n \text{ is odd , or } 0 \text{ if } n \text{ is even} \\
b_1 \equiv \frac{a_{n-1}a_{n-2}-a_na_{n-3}}{a_{n-1}} & b_2 \equiv \frac{a_{n-1}a_{n-4}-a_na_{n-5}}{a_{n-1}} & b_3 = ... & ... & \leftarrow \text{row continues until all future entries} = 0 \\
c_1 \equiv \frac{b_1a_{n-3}-a_{n-1}b_2}{b_1} & c_2 \equiv \frac{b_1a_{n-5}-a_{n-1}b_3}{b_1} & c_3 = ... & ... & \leftarrow \text{row continues until all future entries} = 0 \\
d_1 \equiv \frac{c_1b_2-b_1*c_2}{c_1} & d_2 \equiv \frac{c_1b_3-b_1c_3}{a_{n-1}} & d_3 = ... & ... & \leftarrow \text{row continues until all future entries} = 0 \\
(n+1 \text{ total rows}) & & & &
\end{pmatrix}$$

Table Construction Notes:

1. The Routh table has a total of $n+1$ rows.
2. The first row begins with the coefficient of the highest power in the polynomial, and continues with the third coefficient, fifth coefficient, etc., until the end of the polynomial is reached. Therefore, the last entry in the first row is a_1 if n is odd, and a_0 if n is even.
3. The second row begins with the coefficient of the second-highest power in the polynomial, and continues with the fourth coefficient, sixth coefficient, etc., until a_0 if n is odd, or 0 if n is even. The reason for adding the 0 at the end of the second row, in the case of n even, is to make the second row have the same number of elements as the first row.
4. All subsequent rows are called *derived* rows because their entries are calculated from the rows above. In each case, the next row is calculated from the two above it using the pattern shown in the diagram.
5. As rows are added to the table, they become progressively shorter.
6. The last row will be the $(n+1)^{th}$-row. It will always have only a single nonzero entry. The next-to-last row may have either one or two nonzero entries.
7. If an entry in the first column of a derived row is zero, we do not write zero in that position, but rather, a symbol such as ϵ, which should be interpreted as a very small number having the same sign as the first entry in the row above it. The significance of the *sign* is given in the interpretation discussion below.
8. Any row in a Routh table may be multiplied by a positive number without changing the interpretation given below. Thus, if the first entry in the row above any derived row is positive, we may skip the division step shown in the formulas for the entries for that derived row (i.e., we may multiply the derived row by the first entry of the row above, which effectively eliminates the denominators).

Interpretation of the Routh Table

The number of *sign changes* in the first column of the Routh table is equal to the number of roots with positive real parts. Thus, in order for a system to be stable, the entire first column of the Routh table must have the same sign.

A *sign change* occurs whenever the first entry in a row has the opposite sign of the first entry in the row above it. For counting sign changes, only the first entries in each row matter.

If the table includes any derived zeroes in the first column (replaced by ϵ terms as defined in step (vii) above), these do not count as a sign change. However, the entry below them would count as a sign change if it has a different sign from the entry above them. If the entries above and below them have the same sign, a derived zero in the first column indicates roots with zero real parts. Since such roots indicate marginally stable behavior, we typically won't accept a system with zero entries in the first column, even if there are no actual sign changes above or below. This is discussed in more detail later.

Example

Find the number of roots with non-negative real parts for the following polynomial:

$$\lambda^3 + 6\lambda^2 + 12\lambda + 8 = 0 \tag{2}$$

Construct the Routh table:

$$\begin{pmatrix} 1 & 12 \\ 6 & 8 \\ \frac{64}{6} & \\ 8 & \end{pmatrix}$$

Interpretation: Since there are no sign changes in the first column, all roots of the polynomial have negative real parts.

The actual roots may be found by a root-finder: $\lambda_{1,2,3} = -2, \ -2, \ -2$, so we see that Routh-Hurwitz is correct.

Recall from note (8) that any row in the Routh table may be multiplied by a positive number without changing the resulting interpretation. In the third row above, we may multiply by 6 to obtain:

$$\begin{pmatrix} 1 & 12 \\ 6 & 8 \\ 64 & \\ 8 & \end{pmatrix}$$

Clearly, there are still no sign changes in the first column, so the result does not change. Why would we choose to multiply a row? Purely for convenience! This leaves all table entries as integers, which is simply a bit easier to construct.

Example

Find the number of roots with non-negative real parts for the following polynomial:

$$\lambda^3 - 6\lambda^2 + 12\lambda - 8 = 0 \tag{3}$$

First, note that since the coefficients are not all the same sign, this CE fails the necessary condition. Without going further, we already know that not all of the roots have negative real parts.

For completeness, let's construct the Routh table anyway:

$$\begin{pmatrix} 1 & 12 \\ -6 & -8 \\ \frac{-64}{-6} = +10.667 & \\ -8 & \end{pmatrix}$$

This table has 3 sign changes in the first column, indicating that three roots have positive real parts. How do we count the sign changes? Going down the first column of the table, we see that the signs are, in order, (+), (-), (+), and (-). Thus, there is a sign change from row 1 to row 2; and from row 2 to row 3; and from row 3 to row 4. Thus, a total of 3 sign changes.

Using a root-finder, we can verify the Routh-Hurwitz result by calculating the roots: $\lambda_{1,2,3} = 2,\ 2,\ 2$. As shown by Routh-Hurwitz, we see that there are indeed exactly three roots with positive real parts!

As in the previous example, we may multiply the third row by 6 to obtain:

$$\begin{pmatrix} 1 & 12 \\ -6 & -8 \\ 64 & \\ -8 & \end{pmatrix}$$

This leaves all entries as integers, which is a convenience when constructing Routh tables. The number of sign changes in the first column is not affected. But remember: you can only multiply a row by a positive number.

Example

Find the number of roots with non-negative real parts for the following polynomial:

$$\lambda^5 + 8\lambda^4 + 59\lambda^3 + 224\lambda^2 + 598\lambda + 676 = 0 \tag{4}$$

Construct the Routh table:

$$\begin{pmatrix} 1 & 59 & 598 \\ 8 & 224 & 676 \\ 248 & 4,108 & \\ 32,864 & 112,096 & \\ 27,799,808 & & \\ 112,096 & & \end{pmatrix}$$

Note that each derived row (i.e., rows 3-6) has been multiplied by the first entry in the row above it to keep all entries as integers (for convenience). Again, we can only multiply by positive numbers, but each first entry is positive.

Interpretation: There are no sign changes in the first column of the table, and therefore, the polynomial has no roots with non-negative real parts. The fifth-order system is stable.

We can verify this result by calculating the roots: $\lambda_{1,2,3,4,5} = -2, \; -2 \pm i3, \; -1 \pm i5$. Clearly, there are indeed no roots with positive real parts.

Example: Design for stability

Find conditions on the variable k to ensure that the system is stable, whose characteristic equation is given by

$$\lambda^4 + 8\lambda^3 + 5\lambda^2 + 4\lambda + k = 0 \tag{5}$$

First, we enforce the *necessary condition* for stability, namely, that all coefficients must have the same sign. Thus, the first condition is simply

$$k > 0 \tag{6}$$

To find any additional conditions, we construct the Routh table:

$$\begin{pmatrix} 1 & 5 & k \\ 8 & 4 & 0 \\ 36 & 8k & \\ 144-64k & & \\ 8k & & \end{pmatrix}$$

where each derived row (i.e., rows 3-5) has been multiplied by the first entry in the row above it to keep all entries as integers (for convenience). Note that this is only allowed if the first entry in the row above is positive. Beginning in row 4, we don't know the actual value of the first entry in the row until we assign a number to the variable k. But we intend to enforce conditions on k such that all entries in the first column will in fact be positive, in order for the system to be stable. Thus, if we are able to enforce such conditions, the multiplication is allowed. If we are not able to enforce such conditions, we will discover this independently of any such multiplication.

Interpretation: For stability, we want no sign changes in the first column. Thus, we require

$$144 - 64k > 0 \Rightarrow k < \frac{144}{64} = 2.25 \tag{7}$$

$$8k > 0 \Rightarrow k > 0 \tag{8}$$

Combining these two conditions, we conclude that the system is stable if and only if

$$0 < k < 2.25$$

Students may utilize a root-finding algorithm (such as **roots** in MATLAB) to verify that values of k within this range lead to a stable system, and outside of this range lead to an unstable system.

For example, if $k = 1$, the roots are $\lambda_{1,2,3,4} = -7.3945, -0.1459 + 0.6402i, -0.1459 - 0.6402i, -0.3137$

If $k = 10$, the roots are $\lambda_{1,2,3,4} = -7.3703, -1.2323, 0.3013 + 1.0051i, 0.3013 - 1.0051i$, i.e., there are 2 roots with positive real parts. If we substitute $k = 10$ into the Routh Table, we see that row 4 becomes negative, but rows 3 and 5 remain positive. Thus we have a sign change from row 3 to 4, and another from row 4 to 5 - two roots with positive real parts!

Example: Design for stability

Find conditions on the variable k to ensure that the system is stable, whose characteristic equation is given by

$$5\lambda^4 + 8\lambda^3 + 1\lambda^2 + 4\lambda + k = 0 \tag{9}$$

First, we enforce the *necessary condition* for stability, namely, that all coefficients must have the same sign. Thus, the first condition is simply

$$k > 0$$

To find any additional conditions, we construct the Routh table:

$$\begin{pmatrix} 5 & 1 & k \\ 8 & 4 & 0 \\ -12 & 8k & \\ 48 + 64k & & \\ 8k & & \end{pmatrix}$$

For stability, we require no sign changes in the first column. But there is a sign change between rows 2 and 3 that is not affected by the value of k, so we cannot make this system stable no matter what value of k we select. Students may utilize a root-finding algorithm (such as **roots** in MATLAB) to illustrate this result.

There are two very important observations to make about the previous two examples.

1. The necessary condition, requiring all coefficients in the characteristic equation to have the same sign (in both examples, this condition was satisfied by $k > 0$), is redundant. The Routh-Hurwitz criteria are sufficient conditions, so if they are satisfied, the necessary condition is also satisfied. However, this doesn't mean the necessary condition should be ignored, because unless it can be satisfied, there is no need to proceed to construct the Routh table.

2. When there are unknown parameters, Routh-Hurwitz enables us to prove whether or not a stable system is even possible, and if so, what conditions must be satisfied (which serve as design constraints). In the first example, we found that a stable system was possible, and that we must have $0 < k < 2.25$. In the second example, we proved that no value of k could possibly make the system stable. Although it *might* be possible to find these results through trial-and-error, it's also possible that we would never find the first solution, and we could never prove the second solutio. But we can utilize Routh-Hurwitz to prove both results!!

Example: Design for stability with two unknown parameters

Find conditions on the variables c and k to ensure that the system is stable:

$$\lambda^4 + 8\lambda^3 + 5\lambda^2 + c\lambda + k = 0 \tag{10}$$

First, we know that all coefficients must have the same sign. Thus, we must have

$$k > 0 \quad , \quad c > 0 \tag{11}$$

These are only necessary conditions, not sufficient to ensure stability. To find the sufficient conditions, we construct the Routh table:

$$\begin{pmatrix} 1 & 5 & k \\ 8 & c & 0 \\ 40 - c & 8k & \\ 40c - c^2 - 64k & & \\ 8k & & \end{pmatrix}$$

where, as before, each derived row (i.e., rows 3-5) has been multiplied by the first entry in the row above it to keep all entries as integers (for convenience). Note that this is only allowed if the first entry in the row above is positive.

Interpretation: For stability, we want no sign changes in the first column. Thus, the sufficient conditions for stability are

$$40 - c > 0 \Rightarrow c < 40 \tag{12}$$

$$40c - c^2 - 64k > 0 \tag{13}$$

$$8k > 0 \Rightarrow k > 0 \tag{14}$$

Note that the presence of two unknowns does not change the method for constructing or interpreting the Routh table. Routh-Hurwitz readily handles multiple unknowns. The interpretation of the sufficient conditions when multiple unknowns are present is generally more complicated than it is with a single unknown. Nevertheless, we are still far better off than if we blindly search numerically for (in this case) exact combinations of c and k that would produce stability.

We can use Eq. (13) to determine the maximum value of k:

$$k < \frac{40c - c^2}{64} \quad \Rightarrow \quad \text{maximum when } \frac{d}{dc}(40c - c^2) = 40 - 2c = 0 \quad \Rightarrow \quad c = 20 \quad \Rightarrow k < 6.25$$

Summarizing, in order for the system of Eq. (10) to be stable, we must have

$$0 < k < 6.25 \tag{15}$$

$$40c - c^2 > 64k \tag{16}$$

What are the chances of learning this via trial-and-error?

Routh-Hurwitz may even be used in cases where none of the numerical values of the coefficients in the characteristic polynomial are known in advance. The resulting algebraic conditions for stability will become more complicated as the order of the characteristic polynomial increases, but the basic procedure is the same.

Example: Design for stability with no prior coefficients

Find conditions on all coefficients a_i to ensure that the system is stable, whose characteristic equation is given by

$$\lambda^3 + a_2\lambda^2 + a_1\lambda + a_0 = 0 \tag{17}$$

Note that we have divided by whatever coefficient may have existed on the highest-order term prior to constructing the Routh table, for convenience.

First, we enforce the *necessary condition* for stability, namely, that all coefficients must have the same sign:

$$a_0 > 0 \quad , \quad a_1 > 0 \quad , \quad a_2 > 0$$

However, as we have previously seen, these conditions will always be enforced by the sufficient conditions that arise out of the Routh table, so it is not necessary to state them explicitly. To find sufficient conditions for stability, we construct the Routh table:

$$\begin{pmatrix} 1 & a_1 \\ a_2 & a_0 \\ a_2a_1 - a_0 & \\ a_0 & \end{pmatrix}$$

from which we find the three algebraic conditions $a_2 > 0$, $a_0 > 0$, and $a_1 > a_0/a_2$.

We now repeat this problem for the 4^{th}-order polynomial

$$\lambda^4 + a_3\lambda^3 + a_2\lambda^2 + a_1\lambda + a_0 = 0 \tag{18}$$

Constructing the Routh table:

$$\begin{pmatrix} 1 & a_2 & a_0 \\ a_3 & a_1 & \\ a_3a_2 - a_1 & a_3a_0 & \\ a_1(a_3a_2 - a_1) - a_0a_3^2 & & \\ a_3a_0 & & \end{pmatrix}$$

The necessary conditions for stability are defined by

$$a_3 > 0 \tag{19}$$
$$a_3a_2 - a_1 > 0 \tag{20}$$
$$a_1(a_3a_2 - a_1) - a_0a_3^2 > 0 \tag{21}$$
$$a_3a_0 > 0 \tag{22}$$

Axis-shifting

Routh-Hurwitz can be used to find how many roots have real parts greater than any number, say, s, where $s \neq 0$. When we use Routh-Hurwitz for $s \neq 0$, we call it *axis shifting*.

The settling time of a stable, large-order system is determined by its slowest-settling root, which is the root whose real part is the least negative. Thus, it's often important to know just how negative the least-negative root really is. Using axis-shifting, we can see how many roots have real parts greater than any number.

For example, suppose we want to know if a system has a settling time of, say, 4 seconds or less. A settling time of 4 seconds corresponds to a time constant of 1 second, which corresponds to a real part of a root equal to (-1). Thus, we need to know that all of the system's roots lie to the left of the $s = -1$ line in the complex plane. Therefore, we utilize Routh-Hurwitz with an axis-shift of $s = -1$.

If we have unknown parameter values to be selected in a design, we can use axis-shifting to find sufficient conditions to meet a specified settling time - or, as the case may be, we can prove that no such values exist.

Problem Construction

To perform axis-shifting, we substitute $\lambda + s$ for λ in the characteristic equation, then collect terms on λ to obtain a modified characteristic equation. For example, suppose the initial characteristic polynomial is

$$\lambda^3 + 6\lambda^2 + 12\lambda + 8 = 0$$

Suppose we want to know how many roots have real parts greater than $s = -1$. We axis-shift to $s = -1$, then perform a Routh-Hurwitz analysis on the shifted polynomial. First, substitute $\lambda - 1$ for λ:

$$(\lambda - 1)^3 + 6(\lambda - 1)^2 + 12(\lambda - 1) + 8 = \lambda^3 + 3\lambda^2 + 3\lambda + 1 = 0$$

An axis-shifted equation has the same order as the original, but different coefficients, some of which may be negative or zero. When that happens, we know immediately that at least one root has a real part larger than s. In this case, the shifted polynomial does not have any negative coefficients, so we proceed to construct the Routh table:

$$\begin{pmatrix} 1 & 3 \\ 3 & 1 \\ 8 & \\ 1 & \end{pmatrix}$$

There are no sign changes in the first column; therefore, we know that none of the roots of $\lambda^3 + 6\lambda^2 + 12\lambda + 8 = 0$ have real parts greater than $s = -1$. In fact, we have previously found that all three roots are $\lambda = -2$, so we see that this result is correct. Now let's try axis-shifting to $s = -3$:

$$(\lambda - 3)^3 + 6(\lambda - 3)^2 + 12(\lambda - 3) + 8 = \lambda^3 - 3\lambda^2 + 3\lambda - 1 = 0$$

Since not all coefficients are positive, we know that not all roots have real parts smaller than $s = -3$. To see how many roots are larger than $s = -3$, we can construct the Routh table:

$$\begin{pmatrix} 1 & 3 \\ -3 & -1 \\ 8 & \\ -1 & \end{pmatrix}$$

There are three sign changes, indicating that three roots of the original polynomial are greater than $s = -3$. This is correct; all three roots are $\lambda = -2$.

Example: Design for speed

Find conditions on the variable k to ensure that the system settling time is less than four seconds, where the characteristic equation is given by

$$\lambda^4 + 8\lambda^3 + 5\lambda^2 + 4\lambda + k = 0 \tag{23}$$

For a settling time of 4 seconds, we require that the system time constant be no greater than $\tau = 1$. This, in turn, means that all roots should have real parts that are less than (i.e., more negative than) $s = -1$. Therefore, we perform an axis-shifting of the polynomial, by substituting $\lambda - 1$ for λ:

$$(\lambda - 1)^4 + 8(\lambda - 1)^3 + 5(\lambda - 1)^2 + 4(\lambda - 1) + k = \lambda^4 - 4\lambda^3 - 13\lambda^2 + 14\lambda + (k - 6) = 0 \tag{24}$$

Since the coefficients in the axis-shifted polynomial do not all have the same sign, regardless of what value we might select for k, we cannot create a system in which all of the roots have real parts less than $s = -1$, as required.

This simple exercise proves that our design goal is impossible to achieve. Alternatively, we may have sought our design by trial-and-error substitution of numerical values for k, followed by use of a numerical root-finding routine to see what the corresponding roots are for that value of k. Since no value of k will produce the roots that we seek, the numerical search could never succeed. How much time/effort is saved by the Routh-Hurwitz approach in this case?!?

3. **Zero in the first column**

In constructing a Routh table, we may calculate zero for one or more entries in the first column. In order to interpret this result, we replace the zero in the first column with a variable, say, ϵ, then proceed to construct the remainder of the table interpreting ϵ as a *positive number* for purposes of calculating entries.

Repeat: If a zero appears in the first column of a Routh table, replace it by a variable ϵ, and construct the remainder of the table assuming that $\epsilon > 0$.

To **interpret** the Routh table, we consider $\epsilon = 0$. Note that $\epsilon = 0$ does not, by itself, represent an actual sign change in the first column, because 0 has no sign and thus cannot be the *opposite* sign of the entry above or below it.

To see if a sign change has actually occurred, we must consider the signs of the entries in the rows just above and just below ϵ. If those two rows have the same sign, then the interpretation of $\epsilon = 0$ is that we have two roots with a *real part equal to zero* - i.e., a complex conjugate pair of roots with real part zero. However, if the entries in the rows just above and just below the row containing ϵ have different signs, then there is a sign change, and thus we have a root with a *positive real part.*

A zero in the first column of a Routh table always means that there are roots with non-negative real parts. When we are evaluating system stability, even a root with zero real part is normally considered unacceptable (we don't accept marginal stability). Thus, for stability, we often don't bother to complete and interpret the Routh table if a zero appears in the first column. However, if we are using axis-shifting to evaluate system speed of response, then we may accept a *zero* root (since the zero root, in this case, actually corresponds to a root at the specified axis).

We may summarize the presence of a derived zero in the first column as follows:

(a) Replace the zero with a variable interpreted as $\epsilon > 0$

(b) Construct the remainder of the Routh table

(c) Count sign changes down the first column as usual, noting that $\epsilon = 0$ by itself does not constitute a sign change

(d) If the number of sign changes is zero, then there exist roots with real parts equal to zero

(e) If the number of sign changes is greater than zero, there exist roots with positive real parts.

(f) For stability purposes, we probably won't accept any zero in the first column. For speed of response purposes, we may accept a zero in the first column.

Example: Derived zero in the first column

Consider the the characteristic equation given by

$$\lambda^3 + 2\lambda^2 + \lambda + 2 = 0 \tag{25}$$

We construct the Routh table to determine whether or not any roots have positive real parts:

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

For the purpose of constructing the table, we consider $\epsilon > 0$. Thus, from row 2 to 3 to 4, we have signs of positive, (no sign), and then positive, i.e., we have no sign changes. Therefore, the polynomial has roots with zero (but not positive) real parts.

In fact, using a root-finder, we see that the roots of this polynomial are $\lambda_{1,2,3} = -2\ ,\ 0 \pm i$; Routh-Hurwitz correctly identified that there are no roots to the right of zero (no sign changes), and there are two roots at zero.

Example: Derived zero in the first column

Consider the the characteristic equation given by

$$\lambda^3 + 2\lambda^2 - 4\lambda - 8 = 0 \tag{26}$$

We construct the Routh table to determine whether or not any roots have positive real parts:

$$\begin{pmatrix} 1 & -4 \\ 2 & -8 \\ \epsilon & \\ -8 & \end{pmatrix}$$

For the purpose of constructing the table, we consider $\epsilon > 0$. Thus, from row 2 to 3 to 4, we have signs of positive, (no sign), and then negative, i.e., we have one sign change. Therefore, the polynomial has one root with a positive real part.

In fact, the roots of this polynomial are $\lambda_{1,2,3} = -2\ ,\ -2\ ,\ 2$