4
Stability Analysis Tools
## TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>§4.1</td>
<td>A Multistep Difference Scheme</td>
<td>4–3</td>
</tr>
<tr>
<td>§4.2</td>
<td>Polynomial Stability Conditions</td>
<td>4–3</td>
</tr>
<tr>
<td>§4.2.1</td>
<td>Amplification Polynomial</td>
<td>4–3</td>
</tr>
<tr>
<td>§4.2.2</td>
<td>Hurwitz Polynomial</td>
<td>4–4</td>
</tr>
<tr>
<td>§4.2.3</td>
<td>Amplification-to-Hurwitz Transform Module</td>
<td>4–5</td>
</tr>
<tr>
<td>§4.2.4</td>
<td>Hurwitz-to-Amplification Transform Module</td>
<td>4–5</td>
</tr>
<tr>
<td>§4.3</td>
<td>The Routh-Hurwitz Criterion</td>
<td>4–6</td>
</tr>
<tr>
<td>§4.3.1</td>
<td>The Hurwitz Determinants</td>
<td>4–6</td>
</tr>
<tr>
<td>§4.3.2</td>
<td>Routh-Hurwitz Criterion Modules</td>
<td>4–7</td>
</tr>
<tr>
<td>§4.4</td>
<td>The Liénard-Chipart Stability Criterion</td>
<td>4–9</td>
</tr>
<tr>
<td>§4.</td>
<td>Notes and Bibliography</td>
<td>4–10</td>
</tr>
<tr>
<td>§4.</td>
<td>References</td>
<td>4–11</td>
</tr>
</tbody>
</table>
This Chapter discusses techniques for stability analysis of multilevel systems of difference equations. The key tools are stability polynomials in two flavors: Schur-Cohn and Routh-Hurwitz. The use of a computer algebra system to handle calculations with free parameters is emphasized.

§4.1. A Multistep Difference Scheme

To fix the ideas, suppose that discretization of a linear test equation with $k$ state variables leads to a system of difference equations written in the matrix form

$$A_0 u_{n+1} + A_1 u_n + A_2 u_{n-1} = f_{n+1}. \quad (4.1)$$

The following notation is consistently used in the sequel:

- $u_{n+1}, u_n, u_{n-1}$: The state vector with $k$ components, at time stations $t = t_{n+1}, t_n$ and $t_{n-1}$, respectively. Here $t = t_n$ is the current time station, at which the solution is known, whereas $t_{n+1} = t_n + h$ is the next time station. Previous state solutions, such as $u_{n-1}$ at time station $t_{n-1}$, are part of the historical data. Historical terms such as $A_2 u_{n-1}$ arise if one uses a multistep integration scheme spanning more than one timestep, or a multistep predictor.

- $A_0, A_1, A_2$: A set of $k \times k$ matrices whose entries depend on parameters of the test equation and on the time integration procedure. If state $u_{n-1}$ does not enter in the time discretization (4.1), $A_2 = 0$. Additional terms such as $A_3 u_{n-2}$ may appear if the time integrator uses more previous solutions.

- $f_{n+1}$: The discretized forcing function at $t = t_{n+1}$. In linear systems this term does not depend on the state $u$.

§4.2. Polynomial Stability Conditions

§4.2.1. Amplification Polynomial

To convert (4.1) to an amplification polynomial, set $u_{n-1} = u_n/z = u_{n+1}/z^2$ and $u_n = u_{n+1}/z$. Here $z$ is a variable that will play the role of amplification factor. Factoring out $u_{n+1}$ gives

$$(A_0 + z^{-1}A_1 + z^{-2}A_2)u_{n+1} = f_{n+1}. \quad (4.2)$$

For stability analysis the applied force term $f$ is set to zero. To avoid negative powers it is convenient to multiply both sides by $z^2$ to get

$$(z^2A_0 + zA_1 + A_2)u_{n+1} = Pu_{n+1} = 0. \quad (4.3)$$

This equation has a nontrivial solution if the determinant of the matrix $P = z^2A_0 + zA_1 + A_2$ vanishes. Expanding the determinant as a polynomial in $z$ yields

$$P_A(z) = \det P = \det(z^2A_0 + zA_1 + A_2) = a_0 + a_1z + a_2z^2 + \ldots + a_{n_A}z^{n_A}. \quad (4.4)$$

This $P_A$ receives the name amplification polynomial; it has order $n_A$. Denote the $n_A$ (generally complex) roots of $P_A(z) = 0$ by $z_i, i = 1, \ldots, n_A$. The polynomial $P_A$ is called stable if all roots lie on or inside the unit circle in the complex $z$ plane:

$$|z_1| \leq 1, \ |z_2| \leq 1, \ldots, |z_{n_A}| \leq 1. \quad (4.5)$$

1 This substitution will be connected to the so-called “discrete $z$ transform” in Chapter 8.
This can be expressed compactly by introducing the spectral radius:

\[
\rho = \max_{i=1}^{n_H} |z_i| \quad \Rightarrow \quad P_A \text{ stable if } \rho \leq 1. \tag{4.6}
\]

This condition can be tested directly by the so-called Schur-Cohn stability criterion, which is discussed in detail in [4.50]. We shall not use this condition directly because symbolic computations with free parameters tend to be messy. Instead we transform \( P_A(z) \) to a Hurwitz polynomial to apply a more traditional stability test.

§4.2.2. Hurwitz Polynomial

A polynomial \( P_H(s) \) is called Hurwitz\(^2\) if the location of its roots in the left-hand plane \( \Re(s) \leq 0 \) determines stability. More precisely, a Hurwitz polynomial of order \( n_H \) is called stable if all roots \( s_i \) of \( P_H(s) = 0 \) lie in the negative complex \( s \) plane:

\[
\Re(s_i) \leq 0, \quad i = 1, \ldots n_H. \tag{4.7}
\]

To pass from \( P_A(z) \) to \( P_H(s) \) or vice-versa one uses the conformal involutory transformations

\[
z = \frac{1 + s}{1 - s}, \quad s = \frac{z - 1}{z + 1}. \tag{4.8}
\]

To pass from amplification polynomial to Hurwitz, replace \( P_A(z) \) by \( P_A \left( \frac{1+s}{1-s} \right) \). On expanding in \( z \), this process in general produces the quotient of two polynomials: \( P_H(s)/Q_H(s) \). Upon suppressing any common factors, the zeros of \( P_H(s)/Q_H(s) \) are those of \( P_H(s) \). The Hurwitz polynomial is obtained by taking the numerator of the fraction.

To pass from Hurwitz to amplification polynomial, replace \( P_H(s) \) by \( P_A \left( \frac{(z-1)}{(z+1)} \right) \). On expanding in \( z \), this process in general produces the quotient of two polynomials: \( P_A(z)/Q_A(z) \). Upon suppressing any common factors, the zeros of \( P_A(z)/Q_A(z) \) are those of \( P_A(s) \). The amplification polynomial is obtained by taking the numerator of the fraction.

**Example 4.1.** Suppose \( P_A(z) = -1 - z + 2z^2 = 0 \). This has roots \( z_1 = 1 \) and \( z_2 = -1/2 \) so \( \rho = 1 \) and \( P_A \) is stable as per (4.6). To get the Hurwitz polynomial, transform using the first of (4.8):

\[
P_A = -1 - z + 2z^2 \rightarrow -1 - \frac{1 + s}{1 - s} + 2 \frac{(1 + s)^2}{(1 - s)^2} = \frac{3s + s^2}{1 - s^2} \rightarrow P_H(s) = 3s + s^2. \tag{4.9}
\]

This has roots \( s_1 = 0 \) and \( s_2 = -3 \), so it is also stable as per (4.7). To go back to the amplification form, transform using the second of (4.8):

\[
P_H = 3s + s^2 \rightarrow 3 \frac{z - 1}{z + 1} + \frac{(z - 1)^2}{(z + 1)^2} = \frac{-2 - 2z + 4z^2}{(1 + z)^2} \rightarrow P_A(z) = -2 - 2z + 4z^2. \tag{4.10}
\]

This reproduces the original \( P_A(z) \) except for a factor of 2.

\(^2\) A nomenclature used by many authors. To be historically correct it should be called a Hermite-Routh-Hurwitz polynomial as discussed in *Notes and Bibliography*, but that is too long. Hurwitz is credited with closing the problem in 1895 [4.48].
§4.2 POLYNOMIAL STABILITY CONDITIONS

Example 4.2. Take \( P_A(z) \) as the most general fourth order polynomial: 
\[
P_A(z) = a_0 + a_1 z + a_2 z^2 + a_3 z^3 + a_4 z^4.
\]
Transform to a Hurwitz polynomial with the Mathematica module described below, with the invocation
\[
b=\text{HurwitzPolynomialList}\{a_0,a_1,a_2,a_3,a_4\},1,\text{False}\].
\]
This gives
\[
P_H(s) = a_0 + a_1 + a_2 + a_3 + a_4 + (-4a_0 - 2a_1 + 2a_3 + 4a_4)s + (6a_0 - 2a_2 + 6a_4)s^2 + (-4a_0 + 2a_1 - 2a_3 + 4a_4)s^3 + (a_0 - a_1 + a_2 - a_3 + a_4)s^4.
\]
(4.11)

Transforming back with \( \text{AmplificationPolynomialList}[b,1,\text{False}] \) gives 
\[
P_A(z) = 16(a_0 + a_1 z + a_2 z^2 + a_3 z^3 + a_4 z^4),
\]
which is the original polynomial except for a factor of 16.

§4.2.3. Amplification-to-Hurwitz Transform Module

The module \( \text{HurwitzPolynomialList} \) show in Figure 4.1 performs the \( P_A(z) \rightarrow P_H(s) \) transformation automatically. It is particularly useful for polynomials of high order (over 4) and those with symbolic coefficients. The module is invoked as
\[
b=\text{HurwitzPolynomialList}[a,r,\text{norm}]
\]
(4.12)

The arguments are:

- \( a \) A list of the coefficients \( \{a_0,a_1,\ldots,a_n\} \) of the amplification polynomial.
- \( r \) Set to 1. (A non-unity value is used for root sensitivity studies, a topic not treated here.)
- \( \text{norm} \) A normalization flag. Set to True to request that the coefficient of the highest power in \( s \) of the Hurwitz polynomial be scaled to one. This may be an useful option in numerical work. It should not be used in symbolic work, as it may complicate subsequent processing.

The module returns:

- \( b \) A list of the coefficients \( \{b_0,b_1,\ldots,b_k\} \) of the Hurwitz polynomial. The list may be shorter than \( a \), that is, \( k < n \), if one or more trailing coefficients vanish. For example, to map the amplification polynomial \( P_A(z) = (1 - z)^4 = 1 - 4z + 6z^2 - 4z^3 + z^4 \) to Hurwitz, say
\[
b=\text{HurwitzPolynomialList}\{[1,-4,6,-4,1]\},1,\text{True}\].
\]
This returns \( \{0,0,0,0,1\} \) in \( b \), that is, \( P_H(s) = s^4 \). This is easy to check: the four roots of \( P_A(z) = 0 \) are 1, which map into four zero roots of \( P_H(s) = 0 \).
stations

\[ \text{AmplificationPolynomialList}[b_, r_, \text{norm}_\text{\_}] := \text{Module}[\{\text{PA}, \text{PH}, \text{a}, \text{k}, \text{n}=\text{Length}[\text{b}], \text{i}, \text{j}=0\}, \quad \text{If} \quad [\text{n}<0, \quad \text{Return}[\{\}]]; \quad \text{k}=\text{FindLastNonzero}[\text{b}]; \quad \text{If} \quad [\text{k}=0, \quad \text{Return}[\{0\}]]; \quad \text{PH}=\text{b}[1]+\text{Sum}[\text{b}[\text{i}+1] \cdot \text{s}^\text{i}, \{\text{i}, 1, \text{k}-1\}]; \quad \text{PA}=\text{Numerator}[\text{Together}[\text{Simplify}[\text{PH}/.\text{s} \rightarrow (\text{z}-\text{r})/(\text{z}+\text{r})]]]; \quad \text{a}=\text{Table}[0, \{\text{k}\}]; \quad \text{a}[1]=\text{PA}/.\text{z} \rightarrow 0; \quad \text{For} \quad [\text{i}=1, \text{i} \leq \text{k}-1, \text{i}++, \quad \text{a}[[\text{i}+1]]=\text{Coefficient}[\text{PA}, \text{z}^\text{i}]]; \quad \text{If} \quad [\text{norm}, \quad \text{j}=\text{FindLastNonzero}[\text{a}]; \quad \text{If} \quad [\text{j}>0, \quad \text{a}={\text{a}}/{\text{a}}[[\text{j}]]]]; \quad \text{Return}[\text{a}]]; \]

Figure 4.2. Module to produce amplification polynomial from Hurwitz polynomial.

§4.2.4. Hurwitz-to-Amplification Transform Module

Sometimes it is necessary to pass from a Hurwitz polynomial to the corresponding amplification form. The module \text{AmplificationPolynomialList} listed in Figure 4.2 performs the \( P_H(s) \rightarrow P_A(z) \) transformation automatically. The module is invoked as

\[ a=\text{AmplificationPolynomialList}[b, r, \text{norm}] \] (4.13)

The arguments are:

- \( b \): A list of the coefficients \{\( b_0, b_1, \ldots, b_n \)\} of the amplification polynomial.
- \( r \): Set to 1. (A non-unity value is used for root sensitivity studies, a topic not treated here.)
- \( \text{norm} \): A normalization flag. Set to True to request that the coefficient of the highest power in \( s \) of the amplification polynomial be scaled to one. This may occasionally be an useful option in numerical work. It should not be used in symbolic work, as it may complicate subsequent processing.

The module returns:

- \( a \): A list of the coefficients \{\( a_0, a_1, \ldots, a_k \)\} of the Hurwitz polynomial. The list may be shorter than \( b \), that is, \( k < n \), if one or more trailing coefficients vanish. For example, the call \( a=\text{AmplificationPolynomialList}[\{0,1,3,3,1\},1,\text{False}] \) returns \( \{0,0,0,-1,1\} \) in \( a \). This is the mapping of \( P_H(s) = s(s+1)^3 \) to \( P_A(z) = z^3(z-1) \).

§4.3. The Routh-Hurwitz Criterion

Consider the Hurwitz polynomial of order \( n \) with real coefficients

\[ P_H(s) = b_0 + b_1 s + b_2 s^2 + \ldots + b_n s^n, \quad \text{with} \quad b_i \text{ real}, \quad b_0 \geq 0. \] (4.14)

Note that \( b_0 \geq 0 \) is assumed; if \( b_0 < 0 \) the whole polynomial should be scaled by \(-1\), which does not change the zeros of \( P_H \).

§4.3.1. The Hurwitz Determinants

To determine if \( P_H(s) \) is stable according to (4.7), introduce the following determinants, known as
Hurwitz determinants:

\[
\Delta_i = \det \begin{bmatrix}
  b_1 & b_3 & b_5 & \ldots & b_{2i-1} \\
  b_0 & b_2 & b_4 & \ldots & b_{2i-2} \\
  0 & b_1 & b_3 & \ldots & b_{2i-3} \\
  0 & 0 & b_2 & \ldots & b_{2i-4} \\
  \vdots & \vdots & \vdots & \ddots & \vdots \\
  \vdots & \vdots & \vdots & \ddots & b_i 
\end{bmatrix}
\]

(4.15)

The indices in each row increase by two, whereas the indices in each column decrease by one. The term \( a_j \) is set to zero if \( j < 0 \) or \( j > n \). Note that \( \Delta_1 = b_1 \). The stability criterion was enunciated by Hurwitz [4.48] as: A necessary and sufficient condition that the polynomial (4.14) have roots with negative real parts is that

\[
\Delta_1(= b_1), \Delta_2, \ldots \Delta_n,
\]

(4.16)

be all positive. Readable proofs may be found in Henrici [4.41], Jury [4.50] or Uspenky [4.85]. The proofs of Gantmacher [4.27] and Hurwitz [4.48] are difficult to follow. Hurwitz made the following observations. First, expanding \( \Delta_n \) by the last column, it is easily shown that \( \Delta_n = b_n \Delta_{n-1} \). The condition that \( \Delta_{n-1} \) and \( \Delta_n \) be positive is equivalent to the requirement that \( \Delta_{n-1} \) and \( b_n \) be positive. Thus the theorem remains valid if \( \Delta_n \) is replaced by \( b_n \). Second, \( \Delta_{n+1}, \Delta_{n+2}, \text{etc.,} \) vanish identically since all entries of the last column vanish. The theorem can therefore be reformulated as: all non-identically vanishing terms of the infinite sequence \( \Delta_1, \Delta_2, \ldots \) must be positive.

Remark 4.1. A necessary condition for (4.14) to be stable is that all coefficients \( b_0 \) through \( b_n \) be positive. The proof is quite simple: if the real part of all roots is negative, every linear factor of \( P_H(s) \) is of the form \( s + p \) with \( p > 0 \), and every quadratic factor is of the form \( z^2 + p_1 z + p_2 \) with \( p_1 > 0 \) and \( p_2 > 0 \). Since the product of polynomials with positive coefficients likewise has positive coefficients, it follows that a stable (4.14) can have only positive coefficients. Consequently finding a negative coefficient is sufficient to mark the polynomial as unstable. But the condition is not sufficient for \( n > 2 \), as the following examples make clear.

Example 4.3. For the quadratic polynomial \( P_H(s) = b_0 + b_1 s + b_2 s^2 \), with \( b_0 > 0 \), the conditions given by (4.16), with \( \Delta_2 \) replaced by \( b_2 \), are

\[
\Delta_1 = b_1 > 0, \quad b_2 > 0.
\]

(4.17)

These are satisfied if the three coefficients: \( b_0, b_1, b_2 \) are positive.

Example 4.4. For the cubic polynomial \( P_H(s) = b_0 + b_1 s + b_2 s^2 + b_3 s^3 \), with \( b_0 > 0 \), and \( \Delta_3 \) replaced by \( b_3 \), the conditions are

\[
b_1 > 0, \quad \Delta_2 = \det \begin{bmatrix} b_1 & b_3 \\ b_0 & b_2 \end{bmatrix} = b_1 b_2 - b_0 b_3 > 0, \quad b_3 > 0.
\]

(4.18)

These are satisfied if the coefficients: \( b_0, b_1, b_3 \) are positive, and \( b_1 b_2 > b_0 b_3 \).

Example 4.5. For the quartic polynomial \( P_H(s) = b_0 + b_1 s + b_2 s^2 + b_3 s^3 + b_4 s^4 \), with \( b_0 > 0 \), and \( \Delta_4 \) replaced by \( b_4 \), the conditions are

\[
b_1 > 0, \quad \Delta_2 = \det \begin{bmatrix} b_1 & b_3 \\ b_0 & b_2 \end{bmatrix} = b_1 b_2 - b_0 b_3 > 0, \quad \Delta_3 = \det \begin{bmatrix} b_1 & b_3 & 0 \\ b_0 & b_2 & b_4 \\ 0 & b_1 & b_3 \end{bmatrix} = b_1 b_2 b_3 - b_0 b_3^2 - b_3^2 b_4 > 0, \quad b_4 > 0.
\]

(4.19)

These are satisfied if the coefficients: \( b_0, b_1, b_4 \) are positive, \( b_1 b_2 > b_0 b_3 \), and \( b_1 b_2 b_3 > b_0 b_3^2 + b_3^2 b_4 \).
HurwitzDeterminant[b_, k_] := Module[
{n = Length[b] - 1, i, j, m, A},
If [k < 1 || k > n || n <= 0, Return[0]]; If [k == n, Return[b[[n + 1]]]]; If [k == 1, Return[b[[2]]]]; A = Table[0, {k}, {k}];
For [i = 1, i <= k, i++,
For [j = 1, j <= k, j++, m = 2*j - i + 1;
If [m > 0 && m <= n + 1, A[[i, j]] = b[[m]] ];];
Return[Simplify[Det[A]]];
]

§4.3.2. Routh-Hurwitz Criterion Modules

Figure 4.3 lists a Mathematica module that computes the Hurwitz determinant \( \Delta_k \). The module is invoked as follows:

\[
\Delta_k = \text{HurwitzDeterminant}[b, k] \quad (4.20)
\]

The arguments are:

- \( b \) A list of the coefficients \( \{b_0, b_1, \ldots, b_n\} \) of the Hurwitz polynomial \( P_H(s) = b_0 + b_1 s + \ldots + b_n s^n \), with real coefficients and \( b_0 > 0 \).
- \( k \) Determinant index, \( k = 1, \ldots, n \).

HurwitzDeterminant returns the Hurwitz determinant \( \Delta_k \). If \( k \) is 1 or \( n \), the module returns \( b_1 \) or \( b_n \), respectively. If \( k \) is outside the range 1 through \( n \), it returns zero.

RouthHurwitzStability[b_] := Module[
{n = Length[b] - 1, i, c},
 n = FindLastNonzero[b] - 1; If [n < 3, Return[b]]; c = Table[0, {n + 1}]; c[[1]] = b[[1]]; For [i = 2, i <= n + 1, i++, c[[i]] = HurwitzDeterminant[b, i - 1]]; Return[Simplify[c]];]

Module RouthHurwitzStability, listed in Figure 4.4, uses HurwitzDeterminant to return the list of Hurwitz determinants (4.16), with \( \Delta_n \) replaced by \( b_n \). It is invoked as

\[
c = \text{RouthHurwitzStability}[b] \quad (4.21)
\]

where the only argument is

- \( b \) A list of the coefficients \( \{b_0, b_1, \ldots, b_n\} \) of the Hurwitz polynomial \( P_H(s) = b_0 + b_1 s + \ldots + b_n s^n \), with real coefficients and \( b_0 > 0 \). The module returns:
- \( c \) A list of Hurwitz determinants arranged as \( b_1, \Delta_2, \ldots, \Delta_{n-1}, b_n \).

**Example 4.6.** Investigate the stability of the polynomial \( P_H(s) = 1 + 3s + 6s^2 + 12s^3 + 11s^4 + 11s^5 + 6s^6 \) used by Henrici as example [4.41, p. 557]. A call \( c = \text{RouthHurwitzStability}[[1, 3, 6, 12, 11, 11, 6]] \) returns \( c = \{1, 3, 6, 6, 20, 4, 6\} \); consequently the polynomial is stable.
§4.4 The Liénard-Chipart Stability Criterion

Table 4.1. Liénard-Chipart Stability Conditions for \( n \leq 12 \)

<table>
<thead>
<tr>
<th>Order ( n )</th>
<th>Positivity conditions</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>( b_0, b_1 )</td>
</tr>
<tr>
<td>2</td>
<td>( b_0, b_1, b_2 )</td>
</tr>
<tr>
<td>3</td>
<td>( b_0, b_1, \Delta_2, b_3 )</td>
</tr>
<tr>
<td>4</td>
<td>( b_0, b_1, \Delta_3, b_3, b_4 )</td>
</tr>
<tr>
<td>5</td>
<td>( b_0, b_1, \Delta_2, b_3, \Delta_4, b_5 )</td>
</tr>
<tr>
<td>6</td>
<td>( b_0, b_1, \Delta_3, b_3, \Delta_5, b_5, b_6 )</td>
</tr>
<tr>
<td>7</td>
<td>( b_0, b_1, \Delta_2, b_3, \Delta_4, b_5, \Delta_6, b_7 )</td>
</tr>
<tr>
<td>8</td>
<td>( b_0, b_1, \Delta_3, b_3, \Delta_5, b_5, \Delta_6, b_7, b_8 )</td>
</tr>
<tr>
<td>9</td>
<td>( b_0, b_1, \Delta_2, b_3, \Delta_4, b_5, \Delta_6, b_7, \Delta_8, b_9 )</td>
</tr>
<tr>
<td>10</td>
<td>( b_0, b_1, \Delta_3, b_3, \Delta_5, b_5, \Delta_6, b_7, \Delta_8, b_9, b_{10} )</td>
</tr>
<tr>
<td>11</td>
<td>( b_0, b_1, \Delta_2, b_3, \Delta_4, b_5, \Delta_6, b_7, \Delta_8, b_9, \Delta_{10}, b_{11} )</td>
</tr>
<tr>
<td>12</td>
<td>( b_0, b_1, \Delta_3, b_3, \Delta_5, b_5, \Delta_6, b_7, \Delta_8, b_9, \Delta_{11}, b_{11}, b_{12} )</td>
</tr>
</tbody>
</table>

\[
\text{LienardChipartStability}[b_\_]:=\text{Module}[\{n=\text{Length}[b]-1, i, c\},
\text{If}[n<3, \text{Return}[b]];\]
\[
\text{If}[\text{EvenQ}[n], c=\text{Table}[\{b[[i]], \text{HurwitzDeterminant}[b, i]\},
\{i, 1, n, 2\}];\]
\[
\text{If}[\text{OddQ}[n], c=\text{Table}[\{b[[i+1]], \text{HurwitzDeterminant}[b, i+1]\},
\{i, 1, n-1, 2\}]; c=\{c[1], c\}];\]
\[
\text{AppendTo}[c, b[[n+1]]]; c=\text{Flatten}[c];\]
\[
\text{Return}[c];\]

Example 4.7. Take the last coefficient of the foregoing polynomial as variable: \( P_H(s) = 1 + 3s + 6s^2 + 12s^3 + 11s^4 + 11s^5 + gs^6 \). For which values of \( g \) is the polynomial stable? Now \( c=\text{RouthHurwitzStability}[\{1, 3, 6, 12, 11, 11, g\}] \) returns \( c=\{1, 3, 6, 6, -88+18g, -968+324g-27g^2, g\} \). A study of the sign of the last three entries shows that the polynomial is only stable for \( 5.615099820540244 \leq g \leq 6.384900179459756 \), which includes the value \( g = 6 \) used in the previous example.

Example 4.8. Another example of Henrici [4.41, p. 558] is: for which values of \( \tau \) is \( P_H(s) = 1 + 4\tau s + 6\tau s^2 + 4\tau s^3 + s^4 \) stable? The call \( c=\text{RouthHurwitzStability}[\{1, 4\tau, 6\tau, 4\tau, 1\}] \) returns \( c=\{1, 4\tau, 4\tau(-1+6\tau), 32\tau^2(-1+3\tau), 1\} \). A little study shows that \( P_H(s) \) is stable if \( \tau > 1/3 \).

§4.4. The Liénard-Chipart Stability Criterion

In 1914 Liénard and Chipart [4.57] streamlined the Routh-Hurwitz criterion, showing that only about half of the Hurwitz determinants are in fact needed, with the remaining conditions replaced by certain polynomial coefficients. The actual conditions for polynomials of order 1 through 12 are listed in Table 4.1, which clearly displays alternating patterns for coefficients and determinants.

The advantages of this reformulation are more important for polynomials with symbolic coefficients, because determinants tend to be substantially more complicated than original coefficients as the polynomial order gets large. For numerical coefficients the advantage is marginal as long as computations are done on computers.
Module LienardChipartStability, listed in Figure 4.5, returns the list of coefficients and Hurwitz determinants furnished by the Liénard-Chipart stability criterion. It is invoked as

\[ c = \text{LienardChipartStability} [b] \]  

(4.22)

The only argument is

- A list of the coefficients \( \{b_0, b_1, \ldots, b_n\} \) of the Hurwitz polynomial \( P_H(s) = b_0 + b_1 s + \ldots + b_n s^n \), with real coefficients and \( b_0 > 0 \). The module returns
- A list of positivity conditions arranged as illustrated for \( n \leq 12 \) by Table 4.1.

**Example 4.9.** Given the sixth order Hurwitz polynomial \( P_H(s) = 1 + 3s + 6s^2 + 12s^3 + 11s^4 + 11s^5 + 6s^6 \),

\[ c = \text{LienardChipartStability} [[1,3,6,12,11,11,6]] \]

returns \( c = \{1,3,6,6,11,4,6\} \), confirming stability.

**Example 4.10.** Given the sixth order Hurwitz polynomial \( P_H(s) = 1 + 3s + 6s^2 + 12s^3 + 11s^4 + 11s^5 + gs^6 \),

\[ c = \text{LienardChipartStability} [[1,3,6,12,11,11,g]] \]

returns \( c = \{1,3,6,6,11,-968+324g-27g^2,g\} \). The study of the sign of the last three entries is simpler than in Example 4.7, and again shows that \( P_H(s) \) is stable for \( 5.615099820540244 \leq g \leq 6.384900179459756 \).

**Example 4.11.** Given the Hurwitz polynomial \( P_H(s) = 1 + 4\tau s + 6\tau s^2 + 4\tau s^3 + s^4 \) of Example 4.8,

\[ c = \text{LienardChipartStability} [[1,4\tau,6\tau,4\tau,1]] \]

returns \( c = \{1,4\tau,6\tau,32\tau^2(-1+3\tau),1\} \). This immediately shows that \( P_H(s) \) is stable if \( \tau > 1/3 \).

**Notes and Bibliography**

The assessment of stability of mechanical control systems created by the nascent Industrial Revolution in England led, by the mid XIX Century, to the mathematical question of whether a given polynomial of order \( n \) and real coefficients has roots with negative real parts. Of course, what makes the problem interesting is to find a solution that can be expressed solely in terms of the coefficients, thus avoiding the explicit computations of the roots.

The question was raised by Maxwell in 1868 in connection with the design of steam engine regulating devices then known as governors. He was not aware that the problem had already been solved in 1856 by Hermite [4.42]. The applied mathematician Routh [4.77] gave an algorithm to find the number of roots \( n_U \) (the unstable roots) in the right-half plane, which for \( n_U = 0 \) solves Maxwell’s question. Independently of Routh in 1895 Hurwitz gave another solution in terms of determinants on the basis of Hermite’s paper. Modern proofs may be found in Henrici [4.41] or Uspenky [4.85]. The proof of Gantmacher [4.27] is unreadable.

Hurwitz’ criterion was further streamlined by Lienart and Chipart [4.57] in 1914. Despite it obvious benefits for high order symbolic polynomials the Liénard-Chipart criterion is described only in a few books. It is discussed in detail by Gantmacher [4.27, p. 220], but unfortunately the theorem statement is incorrect, and the purported proof is wrong. The criterion is briefly mentioned by Jury [4.50, p. 62]; however the sui generis notation used in this book makes the linkage to Routh-Hurwitz difficult to understand.

The related problem of requiring the roots of an amplification polynomial to be within the unit circle was solved separately in the early 1900s by Schur [4.79] and Cohn [4.8].

The general problem of finding whether the roots of a polynomial lie on a certain region of the complex plane, without actually finding the roots, lead to root clustering criteria. This is important in the design of filters and control devices of various kinds. The clustering problem is exhaustively studied in [4.50] for several regions. This book is hampered, however, by unfortunate notation and poor organization.

---

3 On p. 221 of [4.27] it is stated that the Liénard-Chipart criterion can be expressed in four equivalent forms, labelled 1 through 4. But form 1 is only valid for even \( n \) and form 4 for odd \( n \), as can be seen by comparing those statements with Table 4.1. Forms 2 and 3 are always wrong. Like most Russian books, Gantmacher shies away from numerical examples. Any such computation would have immediately displayed the mistake.


§4. References


