CHAPTER 4

TERMINATION OF SIMPLE PROGRAMS

4.1 INTRODUCTION

In the previous chapter the method of developing a set of recurrence equations for a program was detailed. In this chapter the use of Z transformation technique to analyse the recurrence equations is explained. Using the properties of Z transformation the program termination property is studied.

4.2 TRANSFORMATION OF RECURRENCE EQUATIONS

The variables appearing in the program loop block of a single loop can be expressed by a set of recurrence equations with number of loop iterations as a parameter (equation 3.15). These equations represent a set of linear discrete equations. Z transform is a rule by which a sequence of numbers is converted into a function of complex variable, z. It possesses a number of elegant properties which can be used for studying discrete system.

In general Z transform of a causal sequence of
numbers $g(k)$ is defined by

$$Z\{g(k)\} = g(0) + g(1)z + g(2)z^2 + g(3)z^3 + \ldots$$

(4.1)

where $z$ is an arbitrary complex number [37-40]

In the recurrence equations developed for a program, $k$ represents the number of loop iterations and it takes positive integer values only. In Z transform theory $k$ represents discrete time and it also takes only positive integer values. Hence, for transforming recurrence equations into Z transform equations number of loop iterations is treated equivalent to the discrete time.

The left shifting property of Z transform is used to transform the set of recurrence equations of the program into Z transform equations. Let $g(k+1)$ represents the sequence of value of the function $g$ at $(k+1)\text{th}$ discrete time, where $k = 0, 1, 2, \ldots$.

Let $Z\{g(k+1)\}$ denotes the Z transformation of the function. The left shifting property is given by

$$Z\{g(k+1)\} = Z[(G(z) - G(0))]$$

(4.2)

Applying this property to equation (3.15) the following equations are obtained.
\[ Z[v] = Z[v(z) - v(0)] = \]
\[ \sum_{l,k+1}^{1,l} Z[f(v, v, \ldots, v, v, \ldots, v)] \]
\[ l,k \quad q-l,k \quad q \quad m \]

\[ Z[v] = Z[v(z) - v(0)] = \]
\[ \sum_{2,k+1}^{2,2} Z[f(v, v, \ldots, v)] \]
\[ 2,2 \quad l,k \quad 2,2 \quad q-l,k \quad q \quad m \]

\[ Z[v] = Z[v(z) - v(0)] = \]
\[ \sum_{d,k+1}^{d,d} Z[f(v, v, v, \ldots, v, v, \ldots, v)] \]
\[ d,k+1 \quad 2,k+1 \quad d-1,k+1 \quad d,k \quad \ldots \quad 2,k+1 \quad \ldots \quad d-1,k+1 \quad \ldots \quad d,k \quad \ldots \quad q-l,k \quad q \quad m \]

\[ Z[v] = Z[v(z) - v(0)] = \]
\[ \sum_{q-l,k+1}^{q-1,k+1} Z[f(v, v, v, \ldots, v)] \]
\[ q-1,k+1 \quad q \quad m \]

(4.3)

Simplifying equation (4.3),
\[ v(z) = F(v(z), v(z), \ldots, v(z), v(0)) \]
$\ldots, v(0), v(0)$

\begin{equation}
\begin{align*}
v(z) &= F(v(z), v(z), \ldots, v(z), v(0), \ldots, v(0), v(0), \ldots, v(0)) \\
v(0), \ldots, v(0), v(0), v(0), \ldots, v(0) \\
\end{align*}
\end{equation}

where (4.4)

$V \{v(0), v(0), \ldots, v(0)\}$ is the initial state vector.

The recurrence equation of the example 3.1 is given by:

$$v_{1,k+1} = v_{1,k}^{\text{\textendash}1}$$

(4.5)

Applying Z transform to equation (4.5)

$$Z[v_{1,k+1}] = Z[v_{1,k}] - Z[1]$$
Simplifying,

\[ z \frac{v(z)}{z - 0} = \frac{z}{z - 1} \quad (4.6) \]

The recurrence equations of example 3.2 are given by:

\[
\begin{align*}
  v_{1,k+1} &= v_{1,k} + v_{2,k} \\
  v_{2,k+1} &= v_{1,k+1} + 1.0 \quad (4.7)
\end{align*}
\]

Applying Z transform to equation (4.7) and simplifying

\[
\begin{align*}
  z \frac{v(0)}{z - 1} &= \frac{z}{z - 1} + \frac{1}{z - 1} \\
  v(z) &= \frac{v(z) - v(0) + 1}{z - 1} \quad (4.8)
\end{align*}
\]

4.3 TRANSFORMATION OF SLACK VARIABLE

The slack variable \( S(k) = f(\bar{V}(k)) \) for a single loop program can be transformed into Z transform equation by applying Z transform techniques as

\[
S(z) = Z \{ f(\bar{V}(k)) \} \quad (4.9)
\]
Substituting the values for \( v(z) \) where \( 1 \leq i \leq m \) from equation (4.4) the slack variable can be expressed in terms of \( V(z) \).

The slack variable for the example 3.1 is

\[ s(k) = v(k) \]

Taking Z transform for \( s(k) \) and using equation (4.6), the Z transform of the slack variable is given by:

\[ s(z) = \frac{z v(0) - v(0) + v(0) - v(0) - 1}{z - 1} \]

The slack variable for example 3.2 is

\[ s(k) = v(k) - v(2) \]

Taking Z transform and substituting the values of \( v(z) \) and \( v(z) \) from equation (4.8) the following equation is obtained

\[ s(z) = \frac{z(v(0) - v(0) + v(0) - v(0) - 1)}{z - 1} \]

\[ = \frac{z}{z - 1} \]
4.4 DOMAIN OF TERMINATION

The predicate term contains variables and a relational operator from the set \{ >, \geq, <, \leq, =, \neq \}. Depending on the relational operator the domain of termination will vary. The domain of termination is the domain in which the value of slack variable is such that the predicate is true. The domain of termination for the different relational operators appearing in the predicate term is shown in Fig.4.1.

4.5 APPLICATION OF INITIAL AND FINAL VALUE THEOREMS

It is possible to determine, the initial value of the term \( g(k) \) of the sequence at \( k = 0 \) directly from its \( z \) transformation and is given by the initial value theorem, which may be expressed as:

\[
\lim_{k \to 0} \lim_{z \to p^+} g(k) = g(z) \quad (4.12)
\]

The final value theorem gives the final value of \( G(k) \) as \( k \to \infty \) directly from its \( Z \) transformation, which may be expressed as:

\[
\lim_{k \to \infty} \lim_{z \to 1} g(k) = (z-1) G(z) \quad (4.13)
\]
FIG. 4.1.a  DOMAIN OF TERMINATION FOR RELATIONAL OPERATOR $>$

FIG. 4.1.b  DOMAIN OF TERMINATION FOR RELATIONAL OPERATOR $\geq$
Fig. 4.1-c Domain of termination for relational operator $<$

Fig. 4.1-d Domain of termination for relational operator $\leq$
FIG. 4.1 e DOMAIN OF TERMINATION FOR RELATIONAL OPERATOR

\[ 0^* \leq \epsilon < 0^* \]
However, if \( G(z) \) has a pole outside the unit circle, the final value is unbounded.

In equation (4.9) it is shown that the slack variable of a program can be expressed in terms of \( Z \) transformed functions. From the nature of the solution, the termination of the program may be determined. The initial value theorem and final value theorem are applied to the slack variable in equation (4.9) and the initial value and final value of the slack variable are found. From these values the termination of the program is decided. Considering example (3.1), the \( Z \) transform of the slack variable is given by equation (4.10). Applying initial value theorem, the initial value \( S(0) \) is given by:

\[
\lim_{z \to \infty} S(0) = S(z)
\]

\[
\lim_{z \to \infty} \frac{z \cdot v(0)}{z-1} = \frac{v(0)}{1} = v(0)
\]

(4.14)

Applying final value theorem, the final value \( S(\infty) \) is given by
\[
\lim_{z \to 1} S(t) = (z-1) s(z)
\]

\[
\lim_{z \to 1} z v(0) = (z-1) \left\{ \frac{v(0)}{z-1} \right\}
\]

\[
= -\infty \quad \text{(4.15)}
\]

(since there is a pole at \( z = 1 \), the value is unbounded).

The slack variable of the program in example 3.1, varies from \( v(0) \) to \(-\infty\) and for the program to terminate, the value must be equal to zero at some finite iteration. If the value of \( v(0) \) is negative, the slack value goes on decreasing to \(-\infty\) and it never reach the value 0. However, if the value of \( v(0) \) is positive, the slack value of the variable will decrease from that value and crosses the zero value and goes to \(-\infty\). The exact value of the variable taking 0 value is not known. Hence, it is difficult to prove the termination condition using initial and final value theorems.

Considering the program in the example 3.2, for equation (4.11) the initial and final value theorem are
applied and the following results are obtained.

\[ S(0) = -1 \]
\[ S(\infty) = -1 \]  \hspace{1cm} (4.16)

The program will terminate if the slack variable value is greater than or equal to zero. But its value remains constant and equals to -1 for all the iterations. Hence the program will not terminate.

4.6 APPLICATION OF INVERSE Z TRANSFORM TO PROGRAMS

The process of generating the sequence, whose Z transform is given in called the inverse Z transform and is denoted by

\[ G(k) = \mathcal{Z}^{-1}[G(z)] \]  \hspace{1cm} (4.17)

Two methods are available for finding the inverse Z transform. They are

1. Partial fraction expansion; and
2. Direct division

Partial fraction expansion method is the simplest procedure for obtaining the sequence of numbers which generates the given Z transform. It is a simple procedure which requires basically two operations of factoring the denominator polynomial of the function being inverse Z
transformed and then making the appropriate partial fraction expansion.

Direct division method of inverting a Z transform is a very elementary process. It is based on the observation that the Z transform is a power series in the variable $z$. For the case when the function $G(z)$ to be inverted is a rational function of $z$, the denominator polynomial is divided in such a way so as to yield a power series in terms of the variable $z^\text{-1}$.

The inverse Z transform technique can be used to expand the sequence of numbers of the slack variable. The nature of the sequence will be used to gather the information about the termination of programs.

Applying the direct method of inverse Z transform for the equation (4.10), the following equation is obtained.

$$S(k) = v(0) + \frac{[v(0)-1]}{1} z + \frac{[v(0)-2]}{1} z^\text{-2} + \ldots + \frac{[v(0)-k]}{1} z^\text{-k}$$

(4.18)

For the program to terminate at finite iteration $k$, 

$$S(k) = 0 \text{ at that iteration}$$
\[ v(0) - k = 0 \]
\[ v(0) = k \]  
\[ (4.19) \]

\( k \) is a positive integer. Hence the program will terminate if \( S(0) \) is positive integer and will not terminate for all other values. Partial fraction expansion method is applied to equation (4.10) and the following equation is obtained

\[ S(k) = v(0) - k \]
\[ 1 \]

Applying \( S(k) = 0 \) to the above equation the program termination conditions are obtained. Fig.4.2 shows the domain of termination and the values of \( s(k) \) for two set of values of initial conditions.

**EXAMPLE 3.2**

The Z transform of the slack variable of the example 3.2 is given in equation (4.11) inverse Z transform is directly applied and the result is given by

\[ S(k) = -1 \]

The program will terminate if \( S(k) \) is greater than or equal to Zero. But \( S(k) = -1 \) is for all iterations. Hence the program will not terminate. Fig.4.3
CASE 1
$V_{K(0)} = 5$

CASE 2
$V_{K(0)} = -1$
(will not terminate)

FIG 4.2 DOMAIN OF TERMINATION FOR EXAMPLE 3.1

CASE 1: Program will terminate at $k = 5$

CASE 2: Program will not terminate
FIG. 4.3  DOMAIN OF TERMINATION OF EXAMPLE 3.2
shows the graphical representation of the domain of termination.

4.7 GENERALISED THEORY OF PROGRAM TERMINATION

In Section 4.3, it is shown that the slack variable can be represented by Z transformed function \( S(z) \) using Z transform technique (equation 4.9).

In general equation (4.9) can be written as

\[
s(z) = \frac{a_0 + a_1 z^{-1} + \ldots + a_m z^{-m}}{1 + b_1 z^{-1} + \ldots + b_n z^{-n}}
\]  

\( a_0, a_1, \ldots, a_m, b_1, b_2 \ldots b_n \) are real constants.

The numerator and denominator are multiplied by \( z^n \) and then numerator and denominator are factored to obtain.

\[
S(z) = \frac{b (z - z_1)(z - z_2) \ldots (z - z_r)}{(z - p_1)(z - p_2) \ldots (z - p_n)}
\]  

where \( z_i \) and \( p_i \) are the zeros and poles of \( S(z) \) respectively and \( b \) is some constant. The response of the above system depends on the location of the poles of the system.
4.7.1 simple pole at \( z = a \)

\( z = a \) be a simple pole and the corresponding function is of the form.

\[
\frac{b}{z-a} \quad (4.22)
\]

This will generate a sequence

\[
k \quad b(a) \quad \text{for } k = 0, 1, 2, \ldots \quad (4.23)
\]

Depending on the value of \( a \), the response will be different. Fig.4.4 shows the response of the slack variable for simple real poles.

**Case 1**

The value of \( a \) in equation (4.23) is greater than one and the response is a diverging sequence.

**Case 2**

The value of \( a \) is equal to one and the response is a constant sequence and equal to the value of \( b \).

**Case 3**

The value of \( a \) is less than one but greater than
FIG. 4.4 PROGRAM RESPONSE DUE TO SIMPLE POLE
or equal to zero. The response is monotonically decreasing from the value of b.

Case 4

The value of a is less than or equal to zero but less than -1. The response is a sequence of numbers alternating in sign and with decreasing in magnitude.

Case 5

The value of a is equal to -1. The response is an alternating sequence, alternating in sign and with constant magnitude b.

Case 6

The value of a is greater than -1 but less than 0. The response is a sequence of numbers alternating in sign and with increasing in magnitude.

All the six cases are illustrated with suitable examples.

Example 4.1
Consider the program given below:

```
r = 5
10 If (r=0) go to 20
```
Using the z transform techniques the Z transform of the slack variable is:

\[ S(z) = \frac{5z}{z-2} \]  \hspace{1cm} (4.24)

The response is a diverging sequence and increasing in magnitude (case 1) from the initial value 5. The domain of termination is given in Fig.4.1.e. The response will never enter the domain of termination for all iterations. The program will not terminate for the given initial value. The program termination for all other relational operators for the given initial value is given below.

<table>
<thead>
<tr>
<th>Relational operator in the predicate</th>
<th>Program termination</th>
</tr>
</thead>
<tbody>
<tr>
<td>&gt;</td>
<td>Terminating</td>
</tr>
<tr>
<td>&gt;</td>
<td>Terminating</td>
</tr>
<tr>
<td>&lt;</td>
<td>Non terminating</td>
</tr>
<tr>
<td>&lt;</td>
<td>Non terminating</td>
</tr>
<tr>
<td>=</td>
<td>Terminating</td>
</tr>
</tbody>
</table>
Example 4.2

Consider the program given below:

```
r = 5
10 If (r=0) goto 20
   r = 5
   goto 10
20 stop
end
```

The Z transform of the slack variable is

\[ S(z) = \frac{5z}{z-1} \]  

(4.25)

The response is constant in magnitude (case 2). The domain of termination is given in Fig.4.1.e. The response will never enter the domain of termination for all iterations. The program will not terminate for the given initial value.

The program termination for other relational operators for the given initial value is given below.

<table>
<thead>
<tr>
<th>Relational operator in the predicate</th>
<th>Program Termination</th>
</tr>
</thead>
<tbody>
<tr>
<td>&gt;</td>
<td>Terminating</td>
</tr>
<tr>
<td>&gt;</td>
<td>Terminating</td>
</tr>
<tr>
<td>≤</td>
<td>Non terminating</td>
</tr>
<tr>
<td>&lt;</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Example 4.3

Consider the program given below:

\[ r = 5 \]

\[ 10 \quad \text{If } (r=0) \quad \text{goto 20} \]

\[ r = 0.5r \]

\[ \text{goto 10} \]

\[ 20 \quad \text{stop} \]

\[ \text{end} \]

The Z transform of the slack variable is

\[ S(z) = \frac{5z}{z-0.5} \quad (4.26) \]

The response is monotonically decreasing from the initial value (case 3). The domain of termination is given in Fig.4.1.e. The response will enter domain of termination after few iterations. The program will terminate for the given initial condition. The program termination for other relational operators for the given initial value is given below:
Example 4.4

Consider the program given below:

\[
\begin{align*}
   & r = 5 \\
   & 10 \quad \text{If } (r=0) \text{ goto 20} \\
   & \quad r = 0.5r \\
   & \quad \text{goto 20} \\
   & 20 \quad \text{stop} \\
   & \text{end}
\end{align*}
\]

The Z transform of the slack variable is

\[
S(z) = \frac{5z}{z + 0.5} \quad (4.27)
\]

The response is alternating in sign and decreasing in magnitude (case 4). The domain of termination is given in Fig.4.1.e. The program will terminate after few
iterations. The program termination for other relational operators is given below.

<table>
<thead>
<tr>
<th>Relational operator in the predicate</th>
<th>Program termination</th>
</tr>
</thead>
<tbody>
<tr>
<td>&gt;</td>
<td>Terminating</td>
</tr>
<tr>
<td>&gt;</td>
<td>Terminating</td>
</tr>
<tr>
<td>=</td>
<td>Terminating</td>
</tr>
<tr>
<td>&lt;</td>
<td>Terminating</td>
</tr>
<tr>
<td>&lt;</td>
<td>Terminating</td>
</tr>
<tr>
<td>≠</td>
<td>Terminating</td>
</tr>
</tbody>
</table>

**Example 4.5**

Consider the program given below:

\[
\begin{align*}
r &= 5 \\
10 &\quad \text{If } (r=0) \text{ goto } 20 \\
&\quad r = -r \\
&\quad \text{goto } 10 \\
20 &\quad \text{stop} \\
&\quad \text{end}
\end{align*}
\]

The Z transform of the slack variable is

\[
S(z) = \frac{5z}{z+1} \quad (4.28)
\]
The response is alternating in sign and constant magnitude (case 5). The domain of termination is given in Fig. 4.1.e. The program will not terminate. The program termination for other relational operators is given below:

<table>
<thead>
<tr>
<th>Relational operator in the predicate</th>
<th>Program termination</th>
</tr>
</thead>
<tbody>
<tr>
<td>&gt;</td>
<td>Terminating</td>
</tr>
<tr>
<td>&gt;</td>
<td>Terminating</td>
</tr>
<tr>
<td>&lt;</td>
<td>Terminating</td>
</tr>
<tr>
<td>&lt;</td>
<td>Terminating</td>
</tr>
<tr>
<td>!</td>
<td>Terminating</td>
</tr>
<tr>
<td>=</td>
<td>Terminating</td>
</tr>
</tbody>
</table>

Example 4.6

Consider the program given below:

```
10 If (r=0) goto 20
r = -2r
goto 20
20 stop
end
```

The Z transform of the slack variable is
The response is increasing in magnitude and alternating in sign (case 6). The domain of termination is given in Fig. 4.1.e. The program will not terminate. The program termination for other relational operators is given below.

<table>
<thead>
<tr>
<th>Relational operator in the predicate</th>
<th>Program termination</th>
</tr>
</thead>
<tbody>
<tr>
<td>&gt;</td>
<td>Terminating</td>
</tr>
<tr>
<td>&gt;</td>
<td>Terminating</td>
</tr>
<tr>
<td>&lt;</td>
<td>Terminating</td>
</tr>
<tr>
<td>&lt;</td>
<td>Terminating</td>
</tr>
<tr>
<td>=</td>
<td>Terminating</td>
</tr>
</tbody>
</table>

4.7.2 Pair of Complex Conjugate Poles

Consider a slack variable, which has a pair of complex conjugate poles given by

\[
P = a + jb \quad \text{and} \quad P = a - jb.
\]

The response of the pole is given by
\[ A r \cos (k \theta + \phi) \quad \text{for} \quad k = 0, 1, 2, \ldots \quad (4.30) \]

where \( A \) and \( \phi \) are constants obtained from the partial fraction expansion of \( S(z) \) and \( r = \sqrt{\frac{2}{a + b}}, \)
\[ \tan \theta = \frac{b}{a}. \]

Depending on the location of the poles the response will be different and the response for complex pole is given in Fig. 4.5.

**Case 1**

The poles are located outside the unit circle. The response is a sinusoidal oscillation increasing in magnitude.

**Case 2**

The poles are located on the unit circle. The response is a sinusoidal like oscillation of constant magnitude.

**Case 3**

The poles are located inside the unit circle. The response is a sinusoidal like oscillation decreasing in magnitude.
FIG. 4-5 PROGRAM RESPONSE DUE TO COMPLEX POLE
In general the termination of program can be determined by knowing the location of the poles of the slack variable and domain of termination.

4.8. ILLUSTRATIVE EXAMPLE

The concepts discussed in this chapter are applied for a practical computer program and the termination of the program is analysed.

Example 4.7

Consider the square root program given below:

Read n
a = 1.0
b = 1.0
c = 0.0

2  If ( a > n ) go to 3
   c = c + 1
   b = b + 2
   a = a + b
   go to 2

3  Write c
   stop

The program is a single loop program. It consists $\bar{X} = \{n\}$, input variables, $\bar{Y} = \{a,b,c\}$ program variables and $\bar{V} = \{b,a,n\}$ with initial state vector
\( V(0) = \{ b(0), a(0), n(0) \} \). Using the procedure explained in the chapter 3, the recurrence equations are given by:

\[
\begin{align*}
  v_{1,k+1} &= v_{1,k} + 2 \\
  v_{2,k+1} &= v_{1,k+1} + v_{2,k}
\end{align*}
\]

The slack variable \( s(k) \) is given by:

\[
s(k) = v_{2,k} - n
\]

Taking \( Z \) transform for the recurrence equations and the slack variable, the following equation is obtained.

\[
s(z) = \frac{3}{z - 3z + 3z - 1} \frac{2}{z (1-n) + z (1+2n) - nz}
\]

Applying initial value theorem,

\[
\lim_{z \to \infty} s(0) = s(z) = 1 - n
\]

The \( s(z) \) has poles at the \( z=1 \) value of \( s(\infty) \) is unbounded. The program will terminate when the value of \( s \)
becomes positive. If the value of \( n \) is negative, then the program terminates at zeroth iteration. If the value of \( n \) is positive then the program terminates after finite iterations.

Partial fraction expansion for \( s(z) \) is applied and the resultant equation is given by

\[
\frac{(1-N)z}{z(z+1)} + \frac{2z}{z-1} + \frac{z(z+1)}{(z-1)^2} (4.33)
\]

Application of inverse Z transformation to equation (4.31) yields.

\[
s(k) = (1-n) + 2k + k^2 (4.34)
\]

The value of \( s(k) \) is plotted for various values of \( n \) and is shown in Fig.4.6. The program will terminate when the value of variable \( s \) is in the domain of termination (shaded area). Hence for all values of \( n \) (positive, negative, zero) the program will terminate.

4.8 CONCLUSION

In this chapter the recurrence equations of the programs are converted into Z transform equations and the properties of Z transform theory are used and the termination of simple programs are studied.
FIG. 4.6.a RESPONSE OF S(k) FOR N<0

FIG. 4.6.b RESPONSE OF S(k) FOR N=1
Fig 4.6: Response of $S(k)$ for $N > 1$