Discrete Mathematics

Session XIV

# Recurrence Relations

Mehran S. Fallah

July 2020

# Introduction

In earlier sessions, we saw some ***recursive*** (***inductive***) definitions and constructions.

In this session, we shall investigate ***sequences*** $\{a_n\}_{n=0}^{\infty}$ where the $n$th term $a_n$ depends on some of the prior terms $a_{n-1}, a_{n-2}, \dots, a_0$.

This study of what are called either ***recurrence relations*** or ***difference equations*** is the discrete counterpart to ideas applied in ordinary ***differential equations***.

Our development, however, will not employ any ideas from differential equations.

As further ideas are developed, we shall see some of the many applications that make this topic so important.

We first introduce recurrence relations. Then, it is studied how one may formulate a (combinatorial) problem as a recurrence relation.

A feel of how one may solve a recurrence relation is developed, although the systematic method for solving certain classes of recurrence relations is the topic of our next session.

# Sequences and Recurrence Relations

A ***sequence*** $a$ is a function from the set $\mathbb{Z}^{\geq 0}$ of nonnegative integers to some set $\mathcal{A}$. A sequence $a: \mathbb{Z}^{\geq 0} \longrightarrow \mathcal{A}$ is usually denoted by $\{a_n\}_{n=0}^{\infty}$, $(a_n)_{n \in \mathbb{Z}^{\geq 0}}$, or by its individual ***elements*** (***terms***) $a_0, a_1, a_2, \ldots$. Here, $a_n$ means $a(n)$, which is called the $n$th element of the sequence.

For example, the sequence $\{1 + 2n^2\}_{n=0}^{\infty}$ can also be represented by its elements as

$$1, 3, 9, 19, 33, \ldots$$

Sometimes, a sequence can be defined ***recursively*** by a so-called ***recurrence relation***, which relates any element of the sequence (except for some ***initial*** elements whose values are given separately as ***boundary***, or ***initial***, ***conditions***) to the ***previous*** elements. This is in contrast to defining the elements of a sequence as a function of their ***indices*** (***positions***.)

As an instance, the Fibonacci sequence $\{F_n\}_{n=0}^{\infty}$ can be defined as

$$F_0 = 0, \qquad F_1 = 1, \qquad \text{and} \qquad F_n = F_{n-1} + F_{n-2} \ \text{ for } n \in \mathbb{Z}^{\geq 2}.$$

Given this definition, one can obtain the elements of the sequence as follows:

$$\begin{aligned} F_0 &= 0, \\ F_1 &= 1, \\ F_2 &= F_1 + F_0 = 1 + 0 = 1, \\ F_3 &= F_2 + F_1 = 1 + 1 = 2, \\ F_4 &= F_3 + F_2 = 2 + 1 = 3, \\ &\cdots \end{aligned}$$

$$\{F_n\}_{n=0}^{\infty} = 0, 1, 1, 2, 3, 5, 8, 13 \ldots$$

# Sequences and Recurrence Relations (Ctd.)

Assume that the sequence $\{a_n\}_{n=0}^{\infty}$ is defined by $a_0 = 3$ and $a_n = 5a_{n-1}$ for $n \geq 1$. For any index $n \geq 1$, we have

$$\begin{aligned} a_n &= 5a_{n-1}, \\ a_{n-1} &= 5a_{n-2}, \\ a_{n-2} &= 5a_{n-3}, \\ &\dots \\ a_2 &= 5a_1, \text{ and} \\ a_1 &= 5a_0. \end{aligned}$$

It follows that $a_n = 5^n \cdot a_0 = 3 \cdot 5^n$ for every $n \geq 0$. That is, $\{a_n\}_{n=0}^{\infty} = \{3 \cdot 5^n\}_{n=0}^{\infty}$. The sequence $\{3 \cdot 5^n\}_{n=0}^{\infty}$ is said to be a ***solution*** to the following ***recurrence relation*** where $a_0 = 3$.

$$a_n = 5a_{n-1} \text{ for } n \geq 1.$$

A ***solution to a recurrence relation*** is a sequence whose elements satisfy the relation. By ***solving*** a recurrence relation, we mean finding a solution to the recurrence relation whose $n$th element is defined in terms of $n$, that is, the elements of the solution are defined as a function of their indices.

# Sequences and Recurrence Relations (Ctd.)

In general, every complex sequence that solves the recurrence relation $a_n = da_{n-1}$ where $n \geq 1$ and $d$ is a constant is of the form $\{cd^n\}_{n=0}^{\infty}$ in which $c$ is an arbitrary complex number. That is, the relation has ***infinitely many*** solutions. Given the initial condition $a_0 = c_0$, the ***unique*** solution to the recurrence relation would be $\{c_0 d^n\}_{n=0}^{\infty}$ (Use mathematical induction to prove this assertion.)

For example, every solution to the relation $a_n = 2.1a_{n-1}$ where $n \geq 1$ is of the form

$$\{c(2.1)^n\}_{n=0}^{\infty}.$$

If the sequence also satisfies the initial condition $a_0 = 5$, its unique solution is the sequence $\{5 \cdot (2.1)^n\}_{n=0}^{\infty}$.

Now, consider the recurrence relation $a_n^2 a_{n-1} = 1$ where $n \geq 1$ and $a_0 = 2$. For every solution $\{a_n\}_{n=0}^{\infty}$ to this equation (if any,) we define a corresponding sequence $\{b_n\}_{n=0}^{\infty}$ by $b_n = \log_2 a_n$ for $n \geq 0$. Thus, the given relation is reduced to $2b_n + b_{n-1} = 0$ where $n \geq 1$ and $b_0 = 1$. The unique solution to this relation is

$$\left\{\left(\frac{-1}{2}\right)^n\right\}_{n=0}^{\infty}.$$

Therefore, the unique solution to the given equation is

$$\left\{2^{\left(\frac{-1}{2}\right)^n}\right\}_{n=0}^{\infty} = 2, \frac{1}{\sqrt{2}}, \sqrt[4]{2}, \frac{1}{\sqrt[8]{2}}, \cdots$$

# Sequences and Recurrence Relations (Ctd.)

Let $\{a_n\}_{n=0}^{\infty}$ be the sequence with $a_0 = 0$, $a_1 = 1$, and $a_n - 2a_{n-1} + 2a_{n-2} = 0$ for $n \geq 2$. We have,

$$a_0 = 0, a_1 = 1, a_2 = 2, a_3 = 2, a_4 = 0, a_5 = -4, a_6 = -8, \dots$$

One may, surprisingly, guess the general form

$$a_n = 2^{\left(\frac{n}{2}\right)} \sin\left(\frac{n\pi}{4}\right).$$

Which of the following can you solve?

$$a_n^2 - 2a_{n-1} = 1 \qquad (n \geq 1)$$

$$a_n - n\sqrt{a_{n-1}} = 0 \qquad (n \geq 1)$$

$$a_n - 2a_{n-1} + 3a_{n-2} = 0 \qquad (n \geq 2)$$

$$a_n - a_{n-2} + a_{n-4} = 0 \qquad (n \geq 4)$$

$$\log a_n - \sin a_{n-1} = \frac{1}{10}n \qquad (n \geq 1)$$

Is there a general method for solving all recurrence relations?

The answer is No, but fortunately there exists a systematic approach to solving certain classes of recurrence relations.

Before we get to solving recurrence relations, we give some examples illustrating how one may solve a (combinatorial) problem by formulating it as a recurrence relation.

# Formulating a Problem as a Recurrence Relation

**Example 1.** A bank pays $6\%$ (annual) interest on savings, compounding the interest monthly. If Bonnie deposits $\$1000$ on the first day of May, how much will this deposit be worth a year later?

**Solution.** The annual interest rate is $6\%$, so the monthly rate is

$$6\%/12 = 0.5\% = 0.005.$$

For $0 \leq n \leq 12$, let $p_n$ denote the value of Bonnie's deposit at the end of $n$ months.

Then $p_n = p_{n-1} + 0.005p_{n-1}$, where $0.005p_{n-1}$ is the interest earned on $p_{n-1}$ during month $n$, for $1 \leq n \leq 12$, and $p_0 = \$1000$.

The recurrence relation

$$p_n = 1.005p_{n-1},$$

where $n \geq 1$ and $p_0 = \$1000$, has the solution $p_n = p_0(1.005)^n$. Consequently, at the end of one year, Bonnie's deposit is worth

$$\boxed{\$1000(1.005)^{12} = \$1061.68.}$$

# Formulating a Problem as a Recurrence Relation (Ctd.)

**Example 2.** In how many ways can one arrange $n$ symbols from the set $\{+,-\}$ in a row so that no consecutive $+'s$ occur in the arrangement?

**Solution.** Let $a_n$ denote the answer. That is, $a_n$ counts the number of ways one can make a row of $n$ symbols $+$ and $-$ such that no successive $+$'s occur in the row. Two cases are possible: the row either begins with $-$ or with $+$.

For the former case, we put $-$ in the first position, and then, we must arrange $n-1$ symbols after the first sign so that no successive $+$'s occur in the arrangement. This can be done in $a_{n-1}$ ways.

$$- \ \overset{a_{n-1}\text{ ways}}{\boxed{+ - + \quad \cdots \quad - +}}$$

If the row begins with $+$, we must put $-$ in the second position, and then, arrange $n-2$ symbols so that no successive $+$'s occur in the arrangement. To do so, we have $a_{n-2}$ ways.

$$+ \ - \overset{a_{n-2}\text{ ways}}{\boxed{+ - \quad \cdots \quad - -}}$$

Hence,

$$a_n = a_{n-1} + a_{n-2} \qquad (n \geq 2.)$$

Moreover, the initial (boundary) conditions are $a_0 = 1$ and $a_1 = 2$. Thus, we have

$$\boxed{\{a_n\}_{n=0}^{\infty} = 1, 2, 3, 5, 8, 13, 21, \ldots = \{F_{n+2}\}_{n=0}^{\infty}.}$$

# Formulating a Problem as a Recurrence Relation (Ctd.)

**Example 3.** A method for sorting numeric data is a technique called the ***bubble sort***. Here, the input is a positive integer $n$ and an array $x[1..n]$ of real numbers that are to be sorted in ascending order.

```
procedure BubbleSort (n:positive integer; array x[1..n]:real)
begin
   for i := 1 to n − 1 do
      for j := n downto i + 1 do
        if x[j] < x[j-1] then
          begin {interchange}
               temp := x[j-1]
               x[j-1] := x[j]
               x[j] := temp
          end
end
```

If $a_n$ denotes the number of comparisons needed to sort $n$ numbers in this way, we get the following recurrence relation.

$$a_n = a_{n-1} + (n - 1), \qquad n \geq 2, \qquad a_1 = 0.$$

Thus,

$$\begin{aligned} a_n &= a_{n-1} + (n - 1) = a_{n-2} + (n-2) + (n-1) = \cdots \\ &= a_1 + 1 + \cdots + (n-1) \\ &= \frac{n(n-1)}{2} \end{aligned}$$

# Formulating a Problem as a Recurrence Relation (Ctd.)

**Example 4 (The towers of Hanoi.)** It is said that after creating the world, God set on Earth three rods made of diamond and 64 rings of gold. These rings are all different in size. At the creation they were threaded on one of the rods in order of size, the largest at the bottom and the smallest at the top. God also created a monastery close by the rods. The monks' task in life is to transfer all the rings onto another rod. The only operation permitted consists of moving a single ring from one rod to another, in such a way that no ring is ever placed on top of another smaller one. When the monks have finished their task, according to the legend, the world will come to an end. This is probably the most reassuring prophecy ever made concerning the end of the world, for if the monks manage to move one ring per second, working night and day without ever resting nor ever making a mistake, their work will still not be finished ***500,000 million years after they began***!

The problem can obviously be generalized to an arbitrary number of rings. To solve the general problem, we need only realize that to transfer the $n$ smallest rings from rod $i$ to rod $j$ (where $1 \le i \le 3,\ 1 \le j \le 3,\ i \ne j,$ and $n \ge 1$), we can first transfer the smallest $n-1$ rings from rod $i$ to rod $6-i-j,$ next transfer the $n$th ring from rod $i$ to rod $j$, and finally retransfer the $n-1$ smallest rings from rod $6-i-j$ to rod $j$.

```
procedure Hanoi(n, i, j)
if n = 1  then  write i "->" j
          else
                {Hanoi(n-1, i, 6-i-j)
                 write i "->" j
                 Hanoi(n-1, 6-i-j, j)}
```

# Formulating a Problem as a Recurrence Relation (Ctd.)

```
procedure Hanoi(n, i, j)
if n = 1  then  write i "->" j
          else
                {Hanoi(n-1, i, 6-i-j)
                 write i "->" j
                 Hanoi(n-1, 6-i-j, j)}
```

For example, with $n = 3$, $i = 1$, and $j = 2$, we obtain the solution given in the following figure.

```
Hanoi(3, 1, 2) ⟶ Hanoi(2, 1, 3) ⟶ Hanoi(1, 1, 2) ⟶ write 1 "->" 2  [1]
                 write 1 "->" 2 [4] write 1 "->" 3 [2]
                 Hanoi(2, 3, 2)   Hanoi(1, 2, 3) ⟶ write 2 "->" 3  [3]
                               ↘
                                  Hanoi(1, 3, 1) ⟶ write 3 "->" 1  [5]
                                  write 3 "->" 2 [6]
                                  Hanoi(1, 1, 2) ⟶ write 1 "->" 2  [7]
```

# Formulating a Problem as a Recurrence Relation (Ctd.)

What about the number of moves?

```
procedure Hanoi(n, i, j)
if n = 1  then  write i "->" j
          else
                {Hanoi(n-1, i, 6-i-j)
                 write i "->" j
                 Hanoi(n-1, 6-i-j, j)}
```

Let $a_n$ denote the number of moves. It is immediate that $a_1 = 1$. For $n \geq 2$, we have

$$a_n = a_{n-1} + 1 + a_{n-1} = 2a_{n-1} + 1.$$

For $n = 64$, the number of moves is $2^{64} - 1$. If the monks manage to move one ring per second, working night and day without ever resting nor ever making a mistake, their work will still not be finished

$$2^{64} - 1 = 18{,}446{,}744{,}073{,}709{,}551{,}616$$

seconds or

$$584{,}942{,}417{,}355$$

years after they began!

Thus, for $n \geq 1$,

$$a_n = 1 + 2 + 2^2 + \cdots + 2^{n-2} + 2^{n-1} = 2^n - 1.$$

# Formulating a Problem as a Recurrence Relation (Ctd.)

**Example 5.** The number of compositions of a positive integer is the number of ways it can be written as a sum of positive integers where the order of summands is relevant. For example, the number of compositions of 4 is 8:

$$4 = 3 + 1 = 2 + 2 = 2 + 1 + 1 = 1 + 3 = 1 + 2 + 1 = 1 + 1 + 2 = 1 + 1 + 1 + 1.$$

Find the number of compositions for a given positive integer $n$.

**Solution.** As a first solution, the number of compositions of $n$ with $k$, $k \geq 1$, summands equals the number of integer solutions to the equation

$$x_1 + x_2 + \cdots + x_k = n,$$

Where $x_i \geq 1$ for $1 \leq i \leq k$. The number of solutions to this equation equals the number of combinations with repetitions of $n - k$ of $k$, that is,

$$H(k, n-k) = \binom{k + (n-k) - 1}{n-k} = \binom{n-1}{n-k} = \binom{n-1}{k-1}.$$

Thus, the number of compositions of a positive integer $n$ is

$$\sum_{k=1}^{n} \binom{n-1}{k-1} = 2^{n-1}.$$

Note that this is the number of ways one can decide whether or not to place a bar (|) in each of the places marked red in the following figurer, where the number of stars equals $n$.

$$* \,\_\, * \,\_\, * \,\_\, \;\; . \; . \; . \;\; \_\, * \,\_\, * \,\_\, * \,\_\, *$$

# Formulating a Problem as a Recurrence Relation (Ctd.)

The second solution is based on deriving a recurrence relation for the number $a_n$ of compositions of a positive integer $n$.

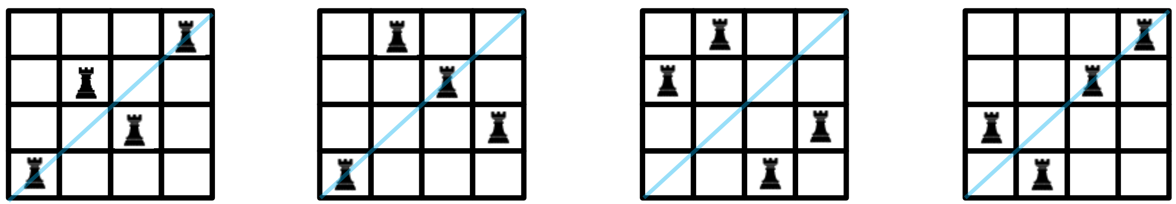
It is immediate that $a_1 = 1$. In every composition of $n \geq 2$, the last summand is either $1$ or a positive integer greater than $1$. If the last summand is $1$, one can obtain a composition of $n-1$ by omitting the last summand. If the last summand is greater than $1$, subtracting $1$ from the last summand yields a composition of $n-1$. Every composition of $n-1$ is also converted to that of $n$ by reversing each of the procedures described above. For example,

| | | |
|---|---|---|
| $4+1$ | $4$ | $5$ |
| $3+1+1$ | $3+1$ | $3+2$ |
| $2+2+1$ | $2+2$ | $2+3$ |
| $2+1+1+1$ | $2+1+1$ | $2+1+2$ |
| $1+3+1$ | $1+3$ | $1+4$ |
| $1+2+1+1$ | $1+2+1$ | $1+2+2$ |
| $1+1+2+1$ | $1+1+2$ | $1+1+3$ |
| $1+1+1+1+1$ | $1+1+1+1$ | $1+1+1+2$ |

Thus, $a_n = a_{n-1} + a_{n-1} = 2a_{n-1}$ where $n \geq 2$ and $a_1 = 1$. Hence, $a_n = 2^{n-1}$ for all $n \geq 1$.

# Formulating a Problem as a Recurrence Relation (Ctd.)

**Example 6.** In chess a piece called a rook, or castle, is allowed at one turn to be moved horizontally or vertically over as many unoccupied spaces as one wishes. In how many ways can one place $n$ rooks on an $n \times n$ chessboard so that 1) no two of them can take each other, and 2) they are placed symmetric with respect to the diagonal of the chessboard?



**Solution.** Let $a_n$ be the number of ways that one can place $n$ rooks on an $n \times n$ chessboard as above. We know that exactly one rook must be placed in each column. The rook of the first column can either be placed on the diagonal or in any of the other squares of the column. For the former case, the number of arrangements is $a_{n-1}$. It is $(n-1)a_{n-2}$ for the latter case.

It follows that $a_n = a_{n-1} + (n-1)a_{n-2}$ for $n \geq 3$. Moreover, $a_1 = 1$ and $a_2 = 2$. For example, $a_4 = a_3 + 3a_2 = (a_2 + 2a_1) + 3a_2 = 2a_1 + 4a_2 = 10$.

**Textbook: Ralph P. Grimaldi, Discrete and Combinatorial Mathematics**

**Please consult with Chapter 10 of your textbook.**