# Chapter 1

# Introduction to Counterfeit Coins Problem

## 1.1 Introduction

Many real-world problems can be formulated as combinatorial problems. The combinatorics underlying a problem is not always evident, but once captured leads to a clean and elegant mathematical model. For that reason, the first approach too many problems consist of separating the practical aspects of the problem from its combinatorial characteristics. We may be, more or less, conscious of this mental process. A kid who plays chess is most probably unaware of the combinatorics of the game, and therefore, is not able to cleverly evaluate the consequences of his/her moves. A more expert player can predict in advance the possible configurations the game may evolve into and decide his/her move accordingly, or even anticipate who will be the winner. Combinatorics has received an extraordinary impulse during the past few decades. Due to the relevance of its practical applications, combinatorics has evolved from a small mathematical discipline practised by a very restricted number of scholars, into a well-structured theory with a complex hierarchy of notions and results. Computer science is perhaps the field where combinatorial methods have been mostly applied. This is not surprising since computers' world is discrete and classical mathematics must be "discretized" to be implemented on computers. In addition to classical methods, like the pigeonhole principle, the double-counting argument, the Ramsey theorem, etc., recently techniques like the probabilistic method and the linear algebra method have found striking applications in theory of computing. Combinatorial techniques have shown a dramatic power in solving a wide variety of computer science problems, ranging from computational complexity to operation research, computational biology, the design of algorithms, cryptography, zero-knowledge and so on.

A contribution of the present thesis is to provide evidence of the versatility of the combinatorial reasoning. The problems discussed in the thesis attain to two distinct areas of research, Search Theory and Algorithm, which share a deep connection with combinatorics. It has been this common aspect to solicit our interest and motivate our research in both areas. The Counterfeit Coins Problem is a well-known complex search problem in combinatorics as well as in computer science. It can be related to the Data Structure (such as a binary tree), Algorithm, Graph Theory, therefore researching this problem is meaningful. The counterfeit coins problem can be described as given a set of $n$ look-alike coins containing one counterfeit which is a bit heavier or lighter than the genuine coins. The aim is to find the counterfeit coins, with minimum numbers of weighing $w$ (trials) with a simple arm balance scale. While a balance scale provides information about the counterfeit coins by comparing the weights of the two subsets of coins, we can also detect counterfeit coins by a spring scale which provides information by weighing a subset of coins instead of talking about the balance scale and spring scale, we will use the more general term of the comparison-type device and test-type device. Simple arguments show that, this is achievable if and only if $n \leq 3^w$, for this problem if the quantity of coins is less, it is not difficult to be solved, but if generalizing the quantity of coins to $n$ (such as 500 or more) ask how to determine quickly the minimum number of comparison required and how to weight and needs look for general law and theoretical basis, and then it is not easy to solve. More interesting versions are obtained by varying the amount of information available to us that is whether we have access to some additional coins that are known to be genuine, and the amount of information we seek, that is whether we would like to know if the counterfeit coin is heavier or lighter than the genuine coins. At first, we will consider the versions obtained by changing the answers to the following four questions for the single counterfeit coins, two of which deals with the amount of information we have and the other deals with the amount of information we seek. Four questions are:

$Q_1$ **:** Do we know in advance if the counterfeit coin is heavier or lighter than the genuine ones?

$Q_2$ **:** Do we know if there is a counterfeit coin?

$Q_3$ **:** Do we have access to additional coins that are known to be genuine?

$Q_4$ **:** Do we want to know if the counterfeit coin is heavier or lighter than the genuine ones?

Since the answer to $\boldsymbol{Q_4}$ is definitely "yes" when the answer to $\boldsymbol{Q_1}$ is "yes", the combination of different answers to these questions generated the 12 different versions. A solution to one of these problems includes two pieces of information: (*i*) the weighing descriptions, that are the set of coins put on each side of the balance scale in each weighing, and (*ii*) a map from their results in the answer; that is the identity of the counterfeit coins and whether it is lighter or heavier than the genuine ones if such information is desired. In a sequential solution, the description of a trial can be affected by the results of the previous ones. In a non-sequential solution, the trials are independent.

**Table 1.1:** The optimal relationship between $n$ and $w$ for which different variants of the single counterfeit coins problem have solutions.

| Name | $\boldsymbol{Q_1}$ | $\boldsymbol{Q_2}$ | $\boldsymbol{Q_3}$ | $\boldsymbol{Q_4}$ | Number of Coins |
|---|---|---|---|---|---|
| $\boldsymbol{P_1}$ | Yes | Yes | Yes | Yes | $3^w$ |
| $\boldsymbol{P_2}$ | Yes | Yes | No | Yes | $3^w$ |
| $\boldsymbol{P_3}$ | Yes | No | Yes | Yes | $3^w - 1$ |
| $\boldsymbol{P_4}$ | Yes | No | No | Yes | $3^w - 1$ |
| $\boldsymbol{P_5}$ | No | Yes | Yes | Yes | $(3^w - 1)/2$ |
| $\boldsymbol{P_6}$ | No | Yes | Yes | No | $(3^w + 1)/2$ |
| $\boldsymbol{P_7}$ | No | Yes | No | Yes | $(3^w - 3)/2$ |
| $\boldsymbol{P_8}$ | No | Yes | No | No | $(3^w - 1)/2$ |
| $\boldsymbol{P_9}$ | No | No | Yes | Yes | $(3^w - 1)/2$ |
| $\boldsymbol{P_{10}}$ | No | No | Yes | No | $(3^w - 1)/2$ |
| $\boldsymbol{P_{11}}$ | No | No | No | Yes | $(3^w - 3)/2$ |
| $\boldsymbol{P_{12}}$ | No | No | No | No | $(3^w - 3)/2$ |

The generalized counterfeit coins problem [1, 4, 41] is that we are given $n$ coins where we know that a coin may be counterfeit. Along with it may be given one or many true (or genuine) coins. Our problem is a distinctive case of the counterfeit coins problem where we are given eight coins, and we know that one of the coins is false and distinguishable by weight. This algorithm is based on the existing classical solution of a more formal instance of this problem, known as the Eight Coins Problem, where there eight coins are given among which only one coin is false. The solution is shown in Figure 1, which is a *decision tree*. The tree in this figure represents a set of decisions by which we can get the solution to our problem. We use H or L as a suffix to represent the counterfeit (or false) coins as *heavier* or *lighter*, respectively. In the solution of the eight coins problem in the form of a decision tree in Figure 1, each internal vertex (other than leaf vertices) represents a comparison between a pair of sets of coins using an equal arm balance.

In the decision tree in Figure 1, we consider three coins on one side of the equal-arm balance, which is the root of the tree. If the weight-sums are equal, then surely each of these coins is a true coin, and the false coin is either G or H with their possibilities either heavier or lighter. In this situation, as A is a true coin, which we use in comparing separately with G and H after a comparison between G and H themselves. If the weight-sum containing coins A is greater than the weight-sum containing coins D (i.e. A+B+C > D+E+F), then we can say that the false coins belong to these six coins only, where either A is heavier, or B is heavier, or C is heavier, or D is lighter, or E is lighter, or F is lighter. At the same time in this situation, both G and H are true coins. The next comparison is highly important in order to make the height of the tree as small as possible; we do three things as follows: (*i*) keep a pair of coins A and E on their own sides, (*ii*) another pair of coins B and D interchange their sides, and (*iii*) the remaining pair of coins C and F are removed from the comparison. Therefore, subsequently, the weight-sum of A and D (i.e. A+D) are compared with the weight-sum of B and E (i.e. B+E). Three cases may arise.

**Case 1:** If weight-sums are equal, then either C or F is a false coin (as these coins are removed from this comparison), where either C is heavier, or F is lighter (following the root of the tree). Therefore, we compare C with A (a true coin). If the weight of C is more than the weight of A, then C is heavier; otherwise, these two coins must have the same weight resulting F as lighter.

**Case 2:** If A+D > B+E, then certainly either A or E is a false coin, as these coins are kept in their own sides, and the logical relation (following the root of the tree) is unchanged. Here either A is heavier, or E is lighter. Therefore, we compare B (a true coin) with A. If the weight of A is more than the weight of B, then A is heavier; otherwise, these two coins must have the same weight resulting E as lighter.

**Case3:** In a similar way, if A+D < B+E, then definitely either B or D is a false coin, as these coins have interchanged their sides and the logical relation (following the root of the tree) has also changed. Here, either D is lighter, or B is heavier. Therefore, we compare B with A (a true coin). If the weight of B is more than the weight of A, then B is heavier; otherwise, these two coins must have the same weight resulting D as lighter. Similarly, we may consider the case of weight-sums following the remaining branch of the root of the tree, where A+B+C < D+E+F. Here either A is lighter, or B is lighter, or C is lighter, or D is heavier, or E is heavier, or F is heavier. The remaining part of the subsequent comparisons is done in a similar way as it is explained above and shown in Figure 1.

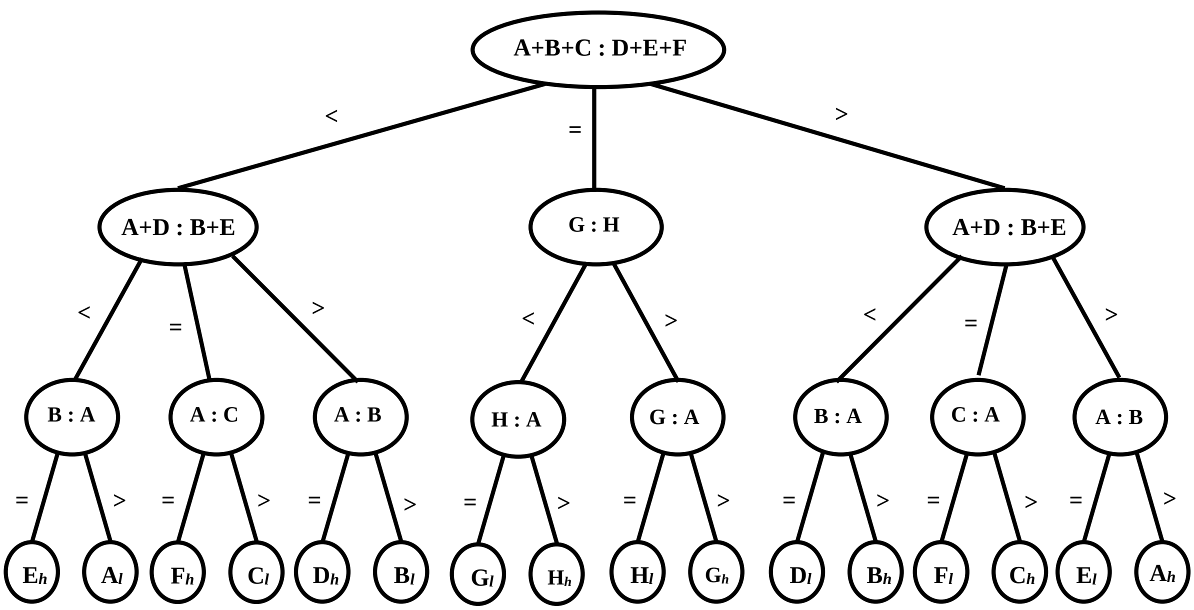


**Figure 1:** The existing solution of the eight coins problem in the form of a decision tree.

## 1.2 A Brief Background on Counterfeit Coins problem

In January of 1945, the following problem appeared in the *American Mathematical Monthly*, contributed by E. D. Schell: *You have eight similar coins and a beam balance. At most one coin is counterfeit and hence underweight. How can you detect whether there is* *an underweight coin and if so which one, using the balance only twice?* Since such counterfeit coins problem is today as much a part of Mathematics and Computer Science. The problem was popular on both sides of the Atlantic during World War II [1] it has even been suggested that it should be dropped over Germany in an attempt to sabotage their war effort in solving [11, 37, 40]. Kaplansky, Neugebauer, and Panel gave the following general solution to the problem of underweight counterfeit coins: If $3^{n-1} \leq N < 3^n$ then $n$ weighing sufficient to show if there is a counterfeit coin among $N$. If it is known that the counterfeit coins exist, then $n$ weighing will identify the coins among $3^{n-1} \leq N < 3^n$. Dyson [30] gave an elegant solution using ternary labels when it is not known if the counterfeit coins are heavy or light.

### 1.2.1 Eight Coins Problem

In this problem, eight coins are given of which only one is false, either heavier or lighter. The aim is to find out the false coins using a minimum number of comparisons amongst the coins and to determine the characteristic feature of the false coins whether it is heavier or lighter in comparison to each of the true coins. Now the question is what the minimum number of comparison is required since each coin has the two possibilities: heavier and lighter, so the total possibility is 16, if we consider $w$ is the total number of comparison then, $w = \log_3{}^{16} \cong 3$.

### 1.2.2 Twelve Coins Problem

Among twelve similar coins, there is one counterfeit. It is not known whether the coin is lighter or heavier than a genuine one all the genuine coins weigh the same. Using three weighings on a pan balance, how can the counterfeit be identified and, in the process, determined to be lighter or heavier than a genuine coin?

### 1.2.3 *n* Coins Problem

The $N$ coins problem describes as there is a set of $n$ coins, out of this n coins one coin is counterfeit with the possibility of heavier or lighter remaining ($n$–1) coins are genuine. We must identify the counterfeit coin with $w = \log_3 2n$ weighings; if one additional genuine coin is given then a minimum number of weighings is $w = \log_3(2n+1)$, if the counterfeit coin is lighter than the minimum number of weighings is $w = \log_3 n$.

## 1.3 Difficulty Level of the Counterfeit Coins Problem

A common phenomenon in combinatorial search theory is that while it is often straightforward to find an optimal procedure searching for one object among a small number of objects, it is immensely more difficult to search optimal procedure for any number of objects, because the number may be odd or even finding counterfeit coins among any numbers of given coins suing single arm balance can be represented as a ternary decision tree, since the balance can be top left, top right, or balance on any given measurement. A decision tree can be *adaptive* or *oblivious.* Adaptive algorithms can solve counterfeit coins problem in $O(\log n)$, but a simple change in the problem solve the problem in $O((\log n)^2)$ measurements. Bellman and Glass [12] studied the problem of identifying two counterfeit coins problem among $n$ coins with a balance scale. They wrote, “A small amount of analysis discloses the enormous difference in complexity between the one coin and two coins problem” solving two counterfeit coins problems is much more difficult because there are seven different versions of the problem such as both are heavier. Two heavier false coins may be equally or unequally heavier. If we denote them as $H_1$ and $H_2$ and their weights as $\omega(H_1)$ and $\omega(H_2)$, we can define these cases as $\omega(H_1) = \omega(H_2)$ (equally heavier) and $\omega(H_1) > \omega(H_2)$ (unequally heavier and we assume that $H_1$ is heavier than $H_2$). Two lighter false coins may be equally or unequally lighter. If we denote them as $L_1$ and $L_2$ and their weights as $\omega(L_1)$ and $\omega(L_2)$, we can define this case as $\omega(L_1) = \omega(L_2)$ (equally lighter) and $\omega(L_1) < \omega(L_2)$ (unequally lighter and we assume that $L_1$ is lighter than $L_2$). Let the true coins is denoted as T, and weight of the true coins is $\omega(T)$. Therefore, heavier and lighter coins are denoted as H and L, respectively. Then, $\omega(H) - \omega(T) = \Delta(\omega(H))$ and $\omega(T) - \omega(L) = \Delta(\omega(L))$ the following situations may arise $\Delta(\omega(H)) > \Delta(\omega(L))$, $\Delta(\omega(H)) < \Delta(\omega(L))$, $\Delta(\omega(H)) = \Delta(\omega(L))$.

## 1.4 Motivations behind the Present Work

The classic puzzle of the counterfeit coin has long served as a stiff test of one’s reasoning power and ingenuity. The problem of the counterfeit coin has circulated in many guises over the years. We have encountered versions involving 8, 10, 12, or 13 coins. In many cases, it is also to be determined whether the counterfeit coin is lighter or heavier than the rest. Finding an algorithm to solve the general problem of determining the minimum number of weighing given *n* coins, one of which is counterfeit, is a popular computer programming exercise. In its standard form, the problem concerns 12 coins identical in size, shape, and appearance. One coin, however, is counterfeit, having a slightly different weight than the other 11 coins. Using only a two-pan balance, what is the smallest number of weightings that would guarantee that we would find the counterfeit coin?

It is said that during the Second World War the English dropped leaflets containing this problem on the German troops with the goal of distracting and thus disorganizing them; supposedly, they wasted 40,000 man-hours on solving it [1]. The classical problem of false coins has recently found applications in the Theory of Coding and Information for detecting errors in code. For most of the twentieth century, Hong Kong had a significant circulation counterfeit coin problem. This can be glimpsed by reading the 1936 Report of the Government Analyst. The introduction of the relatively high value $10 coin towards the end of the century made this problem worse [20, 52]. Every effort was made to incorporate all the most modern anti-counterfeiting features into the coin. This included using a Bimetal Coin and an interrupted fine milled edge. As the problem worsened the material of the coin was changed, but this did not appear to stem the problem. By 2001 the number of counterfeit $10 coins withdrawn had quadrupled and the first quarter figures (of 160,000) for 2002 showed a threefold increase over that previous year [53].

The most frequently seen counterfeit or altered U.S. coins, according to PCGS's 2004 book *Coin Grading and Counterfeit Detection*, include:

- 1856 Flying Eagle Cent
- 1909-S VDB Lincoln Cent
- 1955 double-die Lincoln Cent
- 1916-D Mercury Dime

- Cincinnati commemorative half dollar
- 1804 Bust dollar (a million-dollar rarity)
- 1893-S Morgan dollar
- Saint-Gaudens high-relief double eagle

The above examples show how rampant is the counterfeit coin problem. As such it becomes necessary to find out ways to detect the counterfeit coins. There are various quantitative tests that can be performed to help with counterfeit detection. Often, any one test or several tests are not conclusive, but they can provide important information.

## 1.5 Objectives of the Present Work

In this thesis, we have exhaustively studied each existing counterfeit coins problem based algorithm that we have obtained along with their difficulty level. We have tried to identify and minimize the drawbacks of the existing algorithm in our present work. The foremost objectives of the proposed generalized algorithm for solving *n coins problems* are highlighted as follows —

(1) To develop an algorithm for counterfeit coins problem that is entirely for any value of *n* and the possibility of the counterfeit is heavier or lighter; it follows some distinct deterministic steps. Whereas the other algorithms, we have studied so far, first they declare the possibility of the counterfeit coin.

(2) All the different counterfeit coin related algorithms have been applied to the problem based on adaptive and non-adaptive search technique. It often does not lead to deterministic steps for finding the solution. As there is no standard definition of difficulty levels these methods are applied to self-addressed problem by changing the original essence of the problem, therefore, to develop a technique which will follow some deterministic steps to find out the valid solution of the problem without depending on various versions of a probable list of numbers and difficulty levels is another important aim of the present work.

(3) In the entire existing two counterfeit coins problems [10, 17, 31], they proposed algorithms for finding only heavier or lighter coins, but they did not give any clearer view about the different possibility of the counterfeit coins; however, we have proposed the crystal clearer view of the various natures of coin counterfeiting and their combinations, which is novel for combinatorial point of view.

(4) To develop a methodology, which can right away decide for solving $n$ coins (where $n = 2^P$ where $P$ is any positive integer ≥ 3) problem using decision tree where there is only one false coin.

(5) To develop an algorithm for counterfeit coins problem, this can straightway be applied for solving many input sizes.

(6) To cultivate an entire decision tree representation of the counterfeit coin problem, this is novel in the sense that nobody has ever tried and/or developed such a depiction earlier.

(7) Identifying some new spheres of application of solving a counterfeit coin problem.

(8) To have a brief study of different counterfeit coins problem generation algorithms.

## 1.6 Achievements out of the Work Done

In a modest way, the following contributions have been made in this thesis work:

(1) We have studied exhaustively all the different existing counterfeit coins problem related algorithms.

(2) We have modified the classical solution of eight coins problem. In our new solution we reduced the external path length by 4, total number of comparison by 3, maximum number of coin in a comparison by 2, and the average height of the tree by 0.25, which is novel in the context of data structure and analysis of algorithms.

(3) We have exhaustively studied different algorithms based on the single counterfeit coins problem, and we have overcome the problem to solve all the possibility of counterfeit coins heavier and lighter.

(4) We have developed an algorithm for solving *n* coins problem using decision tree, with the computational complexity O(*n*).

(5) We have identified different application domains of the counterfeit coin problem.

(6) We have developed a new algorithm for solving two counterfeit coins problem with the possibility of equally heavier and equally lighter.

(7) We propose a new algorithm for two counterfeit coins with the possibility un-equalled heavier and un-equally lighter.

(8) We have also developed a new algorithm where one coin is heavier and the other is lighter, but the difference between the heavy and the original coin is greater than the difference between the light and the original coin, that is $\Delta(\omega(H)) > \Delta(\omega(L))$. We have also proposed a new algorithm where one coin is heavier and the other is lighter, but the difference between the heavy and the original coin is less than the difference between the light and the original coin, that is $\Delta(\omega(H)) < \Delta(\omega(L))$.

(9) We have also developed a new algorithm in the form of a decision tree for solving two coins counterfeiting with the difference between the heavy and the original coin is equal to the difference between the true and the lighter coin, that is $\Delta\omega(H) = \Delta\omega(L)$.

## 1.7 Outline of the Thesis

The thesis consists of seven chapters. In Chapter 2, we have discussed the related background and the nature of the problem. We have also made an extensive study on different existing algorithms for solving the counterfeit coins problem, along with their comparative advantages and inadequacies.

In Chapter 3, we have proposed two new solutions of the eight coins problem, and the theoretical lower bound of the single counterfeit coin problem with its associated difficulty levels. In this chapter, we have also considered the general *n* coins problem, where only one coin is fake out of *n* identical given coins.

In Chapter 4, we have discussed the necessary and sufficient conditions for the two counterfeit coins problem, and we have developed algorithms for different combinations of heavier and lighter counterfeit coins, and calculated the computational complexity and related experimental results.

Chapter 5 considers the two counterfeit coins problem where the difference between the heavier and the original coin is unequal to the difference between the lighter and the original coin. This chapter also includes all allied experimental results.

In Chapter 6, we have developed a new algorithm for solving two coins counterfeiting, where the difference between the heavier and the original coin is equal to the difference between the true and the lighter coin, and calculated the computational complexity and corresponding experimental results.

The thesis is concluded in Chapter 7, where we have discussed some applications of counterfeit coins problem and the impact of the counterfeit coins problem on the economy. This chapter summarises the works included in this thesis, and also draws attention to some plausible open problems as further research scopes.