A VARIATION OF THE BUFFON'S NEEDLE AND COIN PROBLEMS

NG WEI EN

BENJAMIN TAN

JARED CHEANG

LAU HONG RUI

VICTORIA SCHOOL

Contents

Abstract

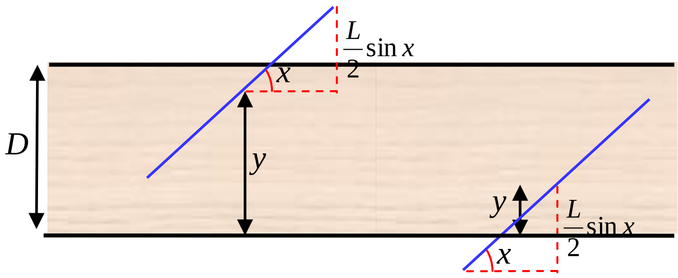
Buffon's Needle and Coin problems are classical geometric probability problems. In our project we apply a variation of the Buffon's Needle and Coin problems, using the principles of balancing, to the scenario of a needle or coin falling through a drain. We derived mathematical formulae for the probability of a needle or coin falling through two edges or a square grid and verified our formulae with computer simulations of our variations using the Chi-Square Goodness of Fit test. Finally, we apply our formulae to find the best combination of coins, worth a total of $1, to carry such that in the event that we drop the coins over a drain cover we are least likely to lose most of the $1. We found that the best combination would be one made up of one 50-cent, two 20-cent and one 10-cent coins.

1 Introduction

In this section, we present a brief introduction to the Buffon's Needle and Coin Problems as well as our variation of these two problems.

1.1 Buffon's Needle Problem

Buffon's Needle Problem is a problem of dropping a needle on a hardwood floor where the event of interest is when the needle crosses a crack between floorboards. The crack and the needle are assumed to have negligible width. Rezhdo (2011) defined variable x for the angle which the needle makes with the cracks and variable y for the perpendicular distance from the centre of the needle to one of the cracks. The range of values of x and y are $0 \le x < \pi$ and $0 \le y < D$, where D is the width of the floorboard or the distance between cracks. In the diagram below, the cracks are represented by the two horizontal lines.



Using trigonometry and calculus, Rezhdo (2011) obtained the following formula for the probability of a needle crossing a crack:

$$\text{P(needle crossing a crack)} = \begin{cases} \dfrac{2L}{\pi D}, & \text{if } L < D \\ \dfrac{2}{\pi}\left(-\sqrt{\left(\dfrac{L}{D}\right)^2 - 1} + \dfrac{L}{D} + \cos^{-1}\left(\dfrac{D}{L}\right)\right), & \text{otherwise} \end{cases}$$

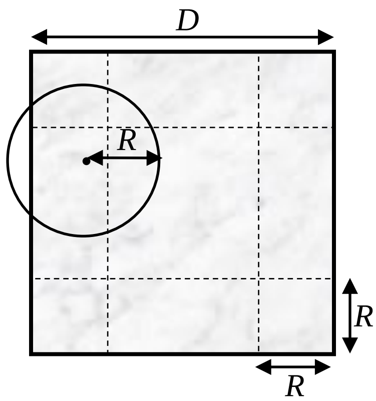
where

L is the length of needle,

D is the width of floorboard or distance between two cracks.

1.2 Buffon's Coin Problem

Buffon's Coin Problem is a problem of dropping a coin on a floor covered with identical square tiles, where the event of interest is when the coin crosses a crack between tiles. Similar to the needle problem, the cracks are assumed to have negligible width. Siergist (n.d.) modeled the Buffon's Coin Problem with square tiles of side length 1 unit using computer simulation and showed that probability of crossing a crack is given by $1-(1-2R)^2$ where *R* is the radius of coin and $R<\frac{1}{2}$.



Consider square tiles of length *D*. The coin crosses a crack when the distance between its centre and any one of the sides of the tile is at most *R*; the area of this region, of width *R*, along the sides of the tile is $D^2-(D-2R)^2$. Hence by considering ratio of areas, the probability of a coin crossing a crack is given by

$$\text{P(coin crossing a crack)} = \begin{cases} 1-\dfrac{(D-2R)^2}{D^2}, & \text{if } R<\dfrac{D}{2} \\ 1, & \text{otherwise} \end{cases}$$

where

R is the radius of the coin,

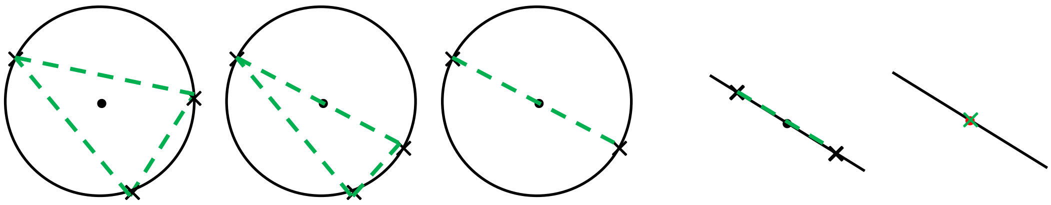
D is the width of square grid.

1.3 Centre of Gravity and Balancing

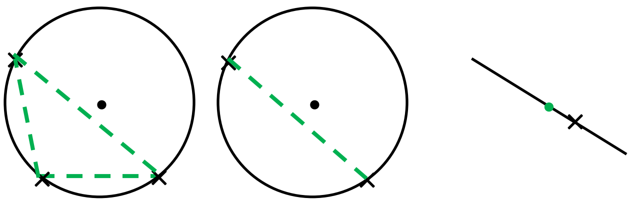
The line of gravity of an object is an imaginary straight line passing through the centre of gravity of the object and the centre of the Earth.

For the object to balance and not topple over, the line of gravity must intersect the base of support. The base of support is the area formed by a perimeter around the points of support for an object. If the centre of gravity of an object lies outside its base of support, it loses its balance (Brown, n.d.).

Examples of centre of gravity of coin or needle lying within the base of support.



Examples of centre of gravity of coin or needle lying outside the base of support.



× is a point of support

• is the centre of gravity of the object

1.4 A Variation of the Buffon's Needle and Coin Problems

In our project we attempted to apply the Buffon's Needle and Coin Problems to the scenario of a needle or coin falling through a drain cover; we imagine the floorboard and tile to be empty spaces while the cracks are part of the metal grilles.

The case of the needle or coin crossing a crack is equivalent to crossing a grille in our application. We assume that the needle or coin does not fall through the drain cover if it crosses a grille. As such, the probability of a needle or coin not falling through the drain cover is the probability of the needle or coin crossing a crack.

In our variation of the Buffon's Needle and Coin Problems we will consider the centre of gravity of the needle and the coin: the needle or coin does not fall through when it balances, that is, if its centre of gravity lies in its base of support.

Similar to the original Buffon's Needle and Coin problems, we assume that the grilles are of negligible width and in our variation we also assume that the masses of the needle and the coin are uniformly distributed and hence the centre of gravity lies in the middle of the needle and the coin.

1.5 Objectives

Our objectives are to:

- derive a set of formulae to calculate the probability of the needle or the coin balancing in our variation of the Buffon's Needle and Coin Problems,
- write computer programs to simulate the same scenarios and use the results to calculate the same probabilities for verification of our formulae,
- apply the formulae to find the best combination of coins worth $1 to carry.

2 Methodology

We derived mathematical formulae for the probabilities of the needle and coin balancing in our variation of the Buffon's Needle and Coin Problems, and created computer simulations for the scenarios of needle and coin “landing” between 2 edges and a square grid respectively. We then compared the probabilities calculated using our formulae with the results from the computer simulations for a range of values. In this section, we describe our methodology.

2.1 Mathematical Derivation

We adopted the variables as used in the original Buffon's Needle and Coin Problems. We derived different sets of formulae for our variation of the Buffon's Needle and Coin problems. The following sections describe the derivation of the formulae in detail.

2.1.1 Variation of the Buffon’s Needle Problem

From the original Buffon's Needle Problem,

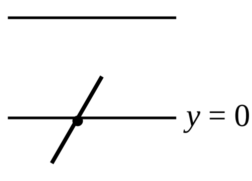
- L is the length of needle,
- D is the width of gap between two parallel lines,
- x is the angle of the needle in radians with respect to the parallel lines, with $0 \le x < \pi$,
- y is the *y*-coordinate of the centre of the needle with respect to the bottom line, with $0 \le y < D$.

In our variation of the needle problem, the needle balances when its centre of gravity lies within its base of support:

1. the centre of gravity of the needle lies on the bottom line, that is when $y = 0$, or
2. the needle crosses both lines.

Notice that $y < D$, hence we do not include the case of $y = D$.

Consider the cases of $L \le D$ and $L > D$.

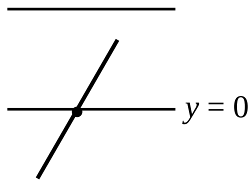


<u>Case 1</u>: $L \le D$

When the length of the needle is less than or equal to the distance between the two parallel lines, the needle balances only when $y = 0$.

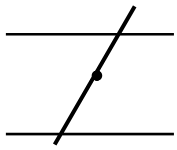
Case 2: $L > D$

When the length of the needle is greater than the distance between the two parallel lines, the needle balances when either of the following condition is satisfied



1. $y = 0$
2. $\left(y < \frac{L}{2}\sin x\right)$ AND $\left(y + \frac{L}{2}\sin x > D\right) \Rightarrow D - \frac{L}{2}\sin x < y < \frac{L}{2}\sin x$,

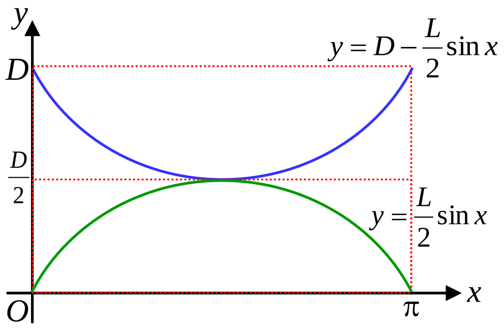
   where $y < \frac{L}{2}\sin x$ when the needle crosses the bottom line and $y + \frac{L}{2}\sin x > D$ when the needle crosses the top line.



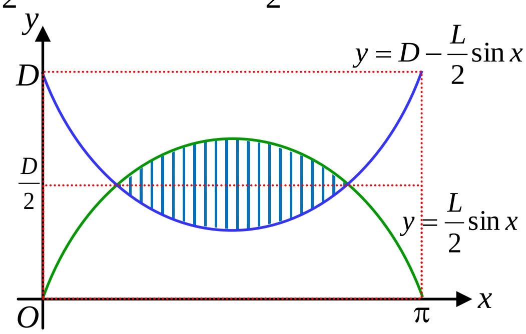
$D - \frac{L}{2}\sin x < y < \frac{L}{2}\sin x$

We will omit the case of $L \le D$ as we are not able to provide a theoretical probability for the occurrence of $y = 0$; a finite outcome out of an infinite number of possible outcomes which will give a probability that is almost negligible. Assuming the probability to be 0 would deem the outcome to be impossible, which intuitively is not the case. Hence we chose to exclude the case of $L \le D$.

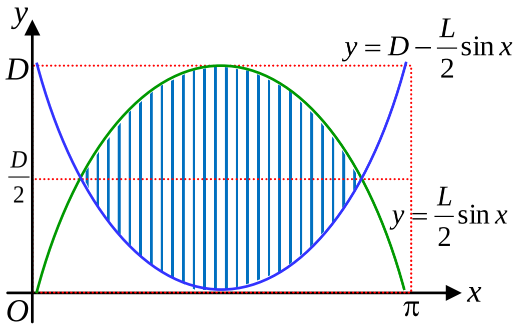
The following are sketches of graphs of $y = \frac{L}{2}\sin x$ and $y = D - \frac{L}{2}\sin x$:



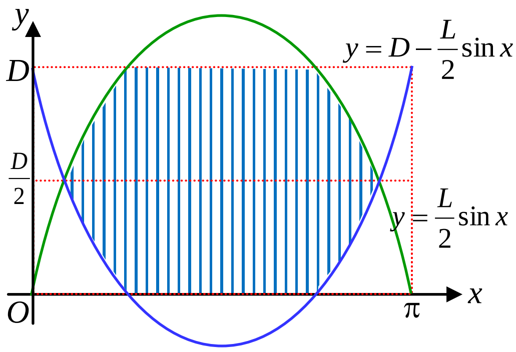
$L = D$



$D < L < 2D$



$L = 2D$



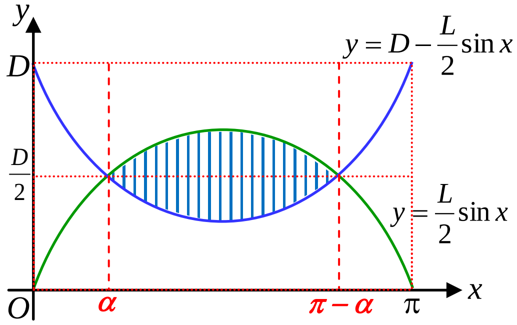
$L > 2D$

We further split case of $L > D$ into two cases:

2a. $D < L \le 2D$

2b. $L > 2D$

<u>Case 2a:</u> $D < L \le 2D$



<u>$D < L \le 2D$</u>

Area of shaded region

$$= \int_{\alpha}^{\pi-\alpha} \frac{L}{2}\sin x\,dx - \int_{\alpha}^{\pi-\alpha} D - \frac{L}{2}\sin x\,dx$$

$$= \int_{\alpha}^{\pi-\alpha} \frac{L}{2}\sin x - \left(D - \frac{L}{2}\sin x\right)dx$$

$$= \int_{\alpha}^{\pi-\alpha} L\sin x - D\,dx$$

$$= \left[-L\cos x - Dx\right]_{\alpha}^{\pi-\alpha}$$

$$= -L\cos(\pi-\alpha) - D(\pi-\alpha) - \left[-L\cos\alpha - D\alpha\right]$$

$$= L\cos\alpha - D\pi + D\alpha + L\cos\alpha + D\alpha$$

$$= 2L\cos\alpha + D(2\alpha - \pi)$$

$$= 2L\left(\frac{\sqrt{L^2-D^2}}{L}\right) + D(2\alpha-\pi)$$

$$= 2\sqrt{L^2-D^2} + D\left[2\sin^{-1}\left(\frac{D}{L}\right) - \pi\right]$$

P(needle balancing)

$$= \frac{2\sqrt{L^2-D^2} + D\left[2\sin^{-1}\left(\frac{D}{L}\right) - \pi\right]}{\pi D}$$

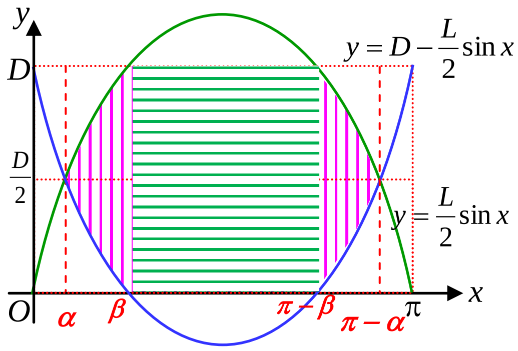
$$= \frac{2}{\pi}\left[\frac{\sqrt{L^2-D^2}}{D} + \sin^{-1}\left(\frac{D}{L}\right)\right] - 1$$

$$= \frac{2}{\pi}\left[\sqrt{\left(\frac{L}{D}\right)^2 - 1} + \sin^{-1}\left(\frac{D}{L}\right)\right] - 1$$

Case 2b: $L > 2D$

Area of green region

$$= D \times [(\pi - \beta) - \beta]$$
$$= D(\pi - 2\beta)$$



Area of pink region

$$= 2\int_{\alpha}^{\beta} \frac{L}{2}\sin x - \left(D - \frac{L}{2}\sin x\right) dx$$
$$= 2\int_{\alpha}^{\beta} L\sin x - D\,dx$$
$$= 2[-L\cos x - Dx]_{\alpha}^{\beta}$$
$$= 2[-L\cos\beta - D\beta - (-L\cos\alpha - D\alpha)]$$
$$= D(2\alpha - 2\beta) + 2L(\cos\alpha - \cos\beta)$$

Area of shaded region

$$= D(\pi - 2\beta) + D(2\alpha - 2\beta) + 2L(\cos\alpha - \cos\beta)$$
$$= D(\pi + 2\alpha - 4\beta) + 2L\left(\frac{\sqrt{L^2 - D^2}}{L} - \frac{\sqrt{L^2 - 4D^2}}{L}\right)$$
$$= D(\pi + 2\alpha - 4\beta) + 2\left(\sqrt{L^2 - D^2} - \sqrt{L^2 - 4D^2}\right)$$
$$= D\left[\pi + 2\sin^{-1}\left(\frac{D}{L}\right) - 4\sin^{-1}\left(\frac{2D}{L}\right)\right] + 2\left(\sqrt{L^2 - D^2} - \sqrt{L^2 - 4D^2}\right)$$

P(needle balancing)

$$= \frac{D\left[\pi + 2\sin^{-1}\left(\frac{D}{L}\right) - 4\sin^{-1}\left(\frac{2D}{L}\right)\right] + 2\left(\sqrt{L^2 - D^2} - \sqrt{L^2 - 4D^2}\right)}{\pi D}$$
$$= \frac{\pi + 2\sin^{-1}\left(\frac{D}{L}\right) - 4\sin^{-1}\left(\frac{2D}{L}\right)}{\pi} + \frac{2\left(\sqrt{L^2 - D^2} - \sqrt{L^2 - 4D^2}\right)}{\pi D}$$
$$= 1 + \frac{2}{\pi}\left[\sin^{-1}\left(\frac{D}{L}\right) - 2\sin^{-1}\left(\frac{2D}{L}\right) + \sqrt{\left(\frac{L}{D}\right)^2 - 1} - \sqrt{\left(\frac{L}{D}\right)^2 - 4}\right]$$

$$\boxed{\text{P(needle balancing)} = \begin{cases} \dfrac{2}{\pi}\left[\sqrt{\left(\dfrac{L}{D}\right)^2 - 1} + \sin^{-1}\left(\dfrac{D}{L}\right)\right] - 1, & \text{if } D < L \le 2D \\[2ex] 1 + \dfrac{2}{\pi}\left[\sin^{-1}\left(\dfrac{D}{L}\right) - 2\sin^{-1}\left(\dfrac{2D}{L}\right) + \sqrt{\left(\dfrac{L}{D}\right)^2 - 1} - \sqrt{\left(\dfrac{L}{D}\right)^2 - 4}\right], & \text{if } L > 2D \end{cases}}$$

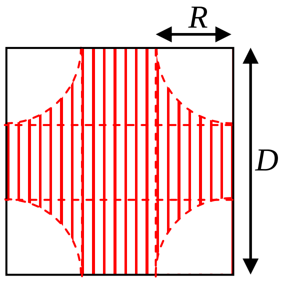
2.1.2 Variation of the Buffon's Coin Problem

From the original Buffon's Coin Problem,

R is the radius of the coin,

D is the width of square grid.

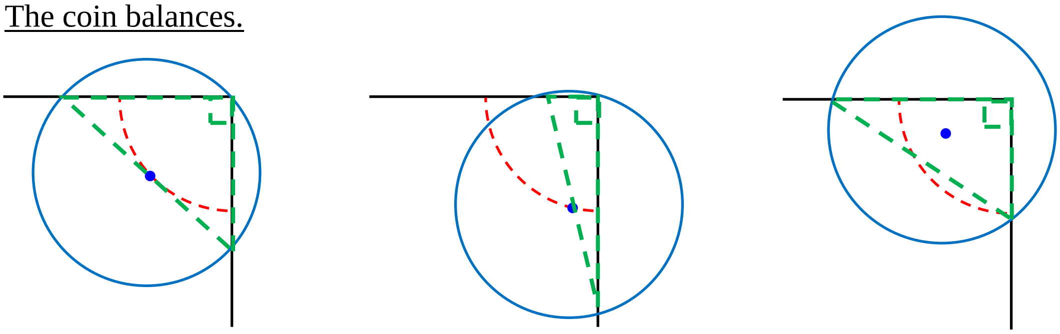
In our variation of the coin problem, the coin balances when its centre of gravity lies within its base of support. The coin balances when the centre of the coin lies (a) on the edge or corner, or (b) in any of the 4 quadrants with radius same as the coin as shown below.



If the centre of the coin lands in this red region, the coin will not balance.

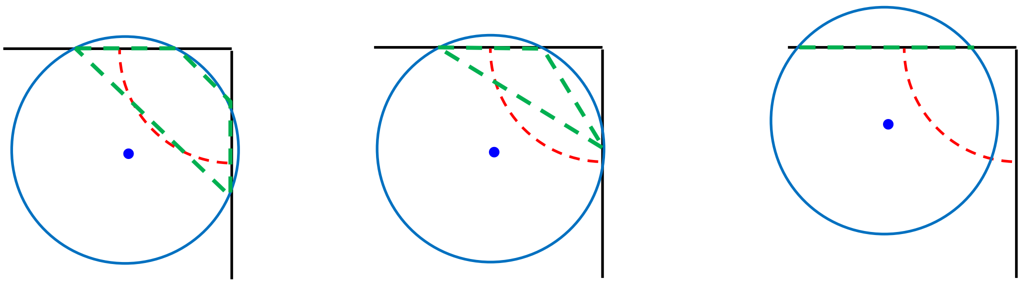
When the centre of the coin lies on the arc of the quadrant or inside the quadrant, the centre of the coin lies within the base of support as shown in the following diagrams:

The coin balances.

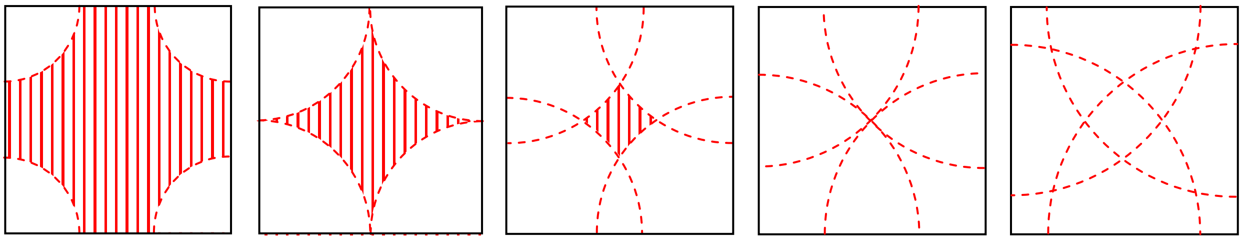


When the centre of the coin lies outside the quadrant, the centre of the coin does not lie within the base of support as shown below:

The coin does not balance.

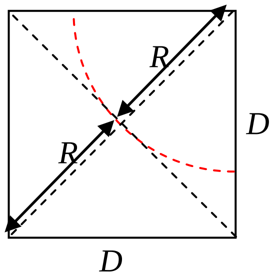


When the radius of the coin increases, the red region diminishes.



$D > 2R$ | $D = 2R$ | $R\sqrt{2} < D < 2R$ | $D = R\sqrt{2}$ | $D < R\sqrt{2}$

Consider the cases of $D \le R\sqrt{2}$, $R\sqrt{2} < D < 2R$ and $D \ge 2R$.

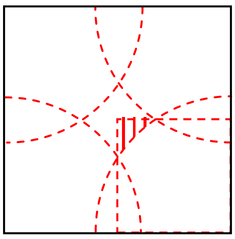


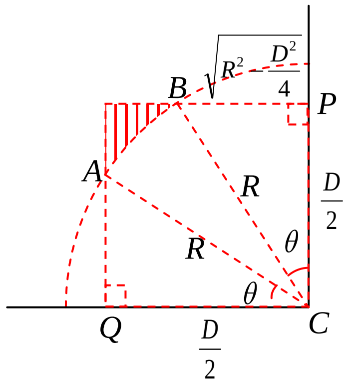
Case 1: $D \le R\sqrt{2}$ (that is $R \ge \dfrac{D\sqrt{2}}{2}$)

P(coin balancing) $= 1$.

By Pythagoras Theorem,
$4R^2 = 2D^2$
$D = R\sqrt{2}$

Case 2: $R\sqrt{2} < D < 2R$ (that is $\dfrac{D}{2} < R < \dfrac{D\sqrt{2}}{2}$)





By Pythagoras Theorem,

$$BP = AQ = \sqrt{R^2 - \left(\frac{D}{2}\right)^2} = \sqrt{R^2 - \frac{D^2}{4}}$$

$$\cos\theta = \frac{\frac{D}{2}}{R} = \frac{D}{2R}$$

$$\theta = \cos^{-1}\left(\frac{D}{2R}\right)$$

$$\angle ACB = \frac{\pi}{2} - 2\cos^{-1}\left(\frac{D}{2R}\right)$$

Area of sector ACB

$$= \frac{\angle ACB}{2\pi} \times \pi R^2$$

$$= \frac{\frac{\pi}{2} - 2\cos^{-1}\left(\frac{D}{2R}\right)}{2\pi} \times \pi R^2$$

$$= \frac{\pi R^2}{4} - R^2\cos^{-1}\left(\frac{D}{2R}\right)$$

Area of unshaded region *ABPCQ*

$$= 2\left[\frac{1}{2}\times\frac{D}{2}\times\sqrt{R^2-\frac{D^2}{4}}\right]+\frac{\pi R^2}{4}-R^2\cos^{-1}\left(\frac{D}{2R}\right)$$

$$=\frac{D}{2}\sqrt{R^2-\frac{D^2}{4}}+\frac{\pi R^2}{4}-R^2\cos^{-1}\left(\frac{D}{2R}\right)$$

P(coin balancing)

$$=\frac{4\left[\frac{D}{2}\sqrt{R^2-\frac{D^2}{4}}+\frac{\pi R^2}{4}-R^2\cos^{-1}\left(\frac{D}{2R}\right)\right]}{D^2}$$

$$=\frac{2}{D}\sqrt{R^2-\frac{D^2}{4}}+\frac{\pi R^2}{D^2}-\frac{4R^2}{D^2}\cos^{-1}\left(\frac{D}{2R}\right)$$

$$=\sqrt{\frac{4}{D^2}\left(R^2-\frac{D^2}{4}\right)}+\left(\frac{R}{D}\right)^2\left[\pi-4\cos^{-1}\left(\frac{D}{2R}\right)\right]$$

$$=\sqrt{4\left(\frac{R}{D}\right)^2-1}+\left(\frac{R}{D}\right)^2\left[\pi-4\cos^{-1}\left(\frac{D}{2R}\right)\right]$$

<u>Case 3</u>: $D \geq 2R$ (that is $R \leq \frac{D}{2}$)

P(coin balancing) $=\pi\left(\frac{R}{D}\right)^2$

P(coin balancing) =	$\begin{cases} \pi\left(\frac{R}{D}\right)^2, & \text{if } R \leq \frac{D}{2} \\ \sqrt{4\left(\frac{R}{D}\right)^2-1}+\left(\frac{R}{D}\right)^2\left[\pi-4\cos^{-1}\left(\frac{D}{2R}\right)\right], & \text{if } \frac{D}{2} < R < \frac{D\sqrt{2}}{2} \\ 1, & \text{if } R \geq \frac{D\sqrt{2}}{2} \end{cases}$

2.2 Simulations

Using Python, a programming language (Rossum, 1989), along with NumPy, a numerical computing library (Watt, Colbert, & Varoquaux, 2011), numexpr, a fast numerical expression evaluator (Cooke, n.d.), and matplotlib, a plotting library (Hunter, 2007), we wrote code that simulates the two variations of Buffon's Needle and Coin Problems. In this section, we will illustrate the logic of our programs. The code presented is a simplified version of our actual code.

2.2.1 Simulation of Variation of the Buffon's Needle Problem

We define the main function *needle_variation* which takes in the length of the needle (represented by *L*) and the width of the gap between the two lines (represented by *D*). This function returns the number of times the needle balances (out of the total number of trials).

```
def needle_variation(L, D, trials):
    hits = 0
    for _ in range(trials):
        x = random(0, pi)
        y = random(0, D)
        if y == 0:
            hits += 1
        elif y < L / 2 * sin(x) and D - y < L / 2 * sin(x):
            hits += 1
    return hits
```

For every trial, we generate a random number x and y in the range $[0, \pi)$ or $[0, D)$ respectively.

If y is equals to 0, the centre of the needle lies on the bottom line. Hence, the needle balances and we increment the *hits* variable. Otherwise, the following two cases are checked:

1. The needle crosses the bottom line: $y < \dfrac{L \sin x}{2}$.
2. The needle crosses the top line: $D - y < \dfrac{L \sin x}{2}$.

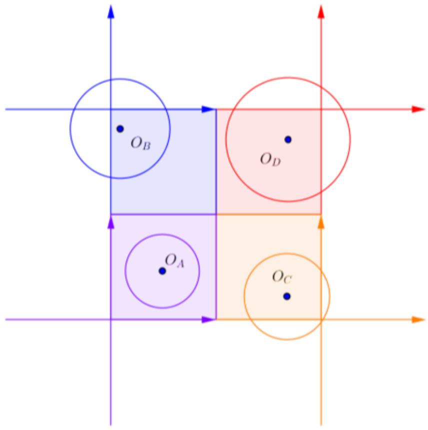
If both inequalities are satisfied, the needle balances and we increment the *hits* variable.

Finally, we return the *hits* variable, which contains the number of times the needle balances.

2.2.2 Simulation of Variation of the Buffon's Coin Problem

We first define *transform_centre* which takes in the width of the gap between the two lines (represented by D), and the coordinates of the centre of the coin (represented by x and y). This function maps the coordinates of the randomly generated centre to a new set of coordinates, with the closest corner as the origin.

```
def transform_centre(D, x, y):
    split = D / 2
    centre_x = x
    if x > split:
        centre_x = D - x
    centre_y = y
    if y > split:
        centre_y = D - y

    return (centre_x, centre_y)
```



For example, if the centre of the coin lies in the bottom right quadrant, represented by the point O_C, the bottom right corner is taken to be the origin. The new coordinates are calculated.

This mapping allows us to simplify the main function, *coin_variation*, which will be described later.

Afterwards, we define *get_pivots* that obtains the intersections of the circumference of the coin and the x and y axes by substituting $x = 0$ or $y = 0$ into the equation $R^2 = (x-h)^2 + (y-k)^2$, where (h, k) is the centre of the circle, to find the corresponding y and x values respectively. We check for the case when there is no corresponding x or y values, where there is no real solution for the equation in x or in y.

```
def get_pivots(R, x, y):
    pivots = []

    sqval = R**2 - y**2
    if sqval > 0: # no imaginary numbers
        sqrt = sqval**(0.5)
```

```
        pivots.append((x + sqrt, 0))
        pivots.append((x - sqrt, 0))
    elif sqval == 0: # tangent
        pivots.append((x, 0))

    sqval = R**2 - x**2
    if sqval > 0: # no imaginary numbers
        sqrt = sqval**(0.5)
        pivots.append((0, y + sqrt))
        pivots.append((0, y - sqrt))
    elif sqval == 0: # tangent
        pivots.append((0, y))

    return pivots
```

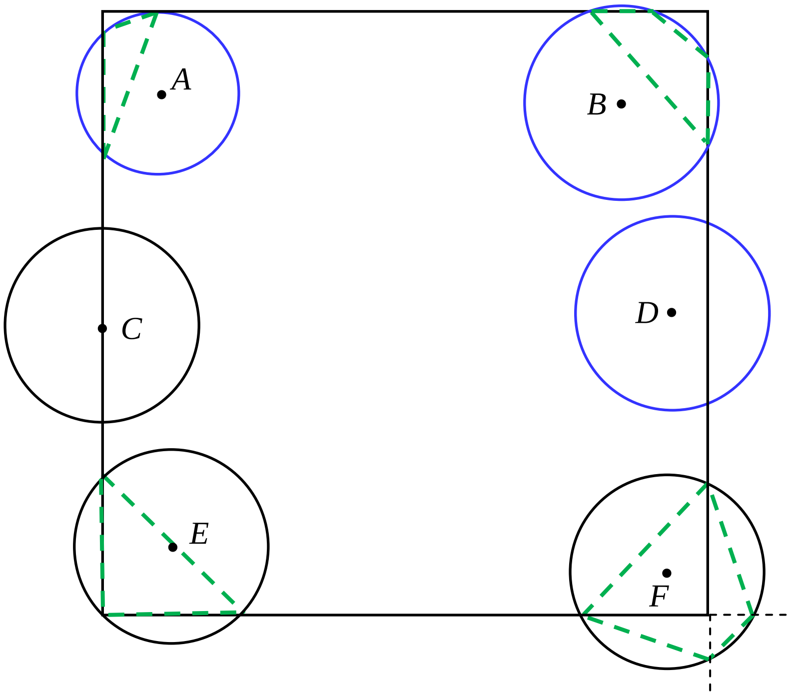
Finally, we define the main function *coin_variation*.

```
def coin_variation(R, D, trials):
    hits = 0
    for _ in range(trials):
        x = random(0, D)
        y = random(0, D)

        x, y = transform_centre(D, x, y)
        if x == 0 or y == 0:
            hits += 1
            continue

        pivots = get_pivots(R, x, y)
        if len(pivots) < 4:
            continue

        points = pivots + [(x, y)]
        hull_points = convex_hull(points)
        if not (x, y) in hull_points:
            hits += 1

    return hits
```

For every trial, we generate random numbers *x* and *y* for the centre of the coin in the range [0, *D*). We transform the coordinates of the centre of the coin. We check if the centre of the coin lies on any of the edges. If yes, we increment the *hits* variable and proceed to the next trial. Otherwise, we check if the centre of the coin lies within the base of support: we find the supporting points or pivots of the coin and check if the center lies in the base of support by calculating the convex hull of the supporting points and the center.

The convex hull of a set of points in a Euclidean plane is defined as the smallest convex set (which is a set of points where every point on the straight line segment joining the pair of points is within the set) of that set of points (Lauritzen, 2009). We compute the convex hull using the Andrew's monotone chain algorithm (Andrew, 1979).

By calculating the convex hull of the supporting points and the centre of the coin, we obtain the vertices that make up the hull. If the centre is one of these vertices, the centre of the coin would lie outside the base of support. Otherwise, the centre of the coin lies inside the base of support. The following diagram illustrates the different cases of whether the coin balances or does not balance.



Coins C, E and F balance as their centre of gravity lies within their base of support while coins A, B and D will not balance as their centre of gravity lies outside their base of support.

Coin	Balance?	Remarks
A	No	Tangent to an axis and centre of coin does not lie on edge
B	No	Centre of coin does not lie in base of support
C	Yes	Centre of coin lies on edge
D	No	Centre of coin does not lie in base of support
E	Yes	Centre of coin lies in base of support. The corner is counted twice as x-intercept and y-intercept, thus the code still works for this case.
F	Yes	The base of support only extends to the edges of the grid. Our code computes the base of support to extend over the edges of the grid, as indicated in the diagram. This does not affect the accuracy of the simulations.

2.3 Chi-Square Goodness of Fit Test

The Chi-Square goodness of fit test (χ^2) is a statistical test applied to sets of categorical data to evaluate the likeliness of any observed difference between the sets arose by chance and not due to an error in the hypothesis. This can be used to test the validity of a distribution assumed for a random phenomenon (Yale University, 1997-1998). Using the Chi-Square goodness of fit test, we compared our theoretical results with our experimental results.

We have two categories for our variations of the Buffon's Needle and Coin Problems: whether the needle or coin balances or does not balance. Hence, the degree of freedom is 1; taking *p*-value to be 0.05, the critical value is 3.841. We calculated the following statistic:

$$\chi^2 = \frac{(S - Pt)^2}{Pt} + \frac{[(t - S) - (t - Pt)]^2}{t - Pt}$$

where t is the total number of trials to be executed, S is the total number of times the simulated needle or coin balances and P is the theoretical probability of the needle or coin balancing.

Our null hypothesis, H_0, states that there is no significant difference between our theoretical results and our experimental results: the probability of needle or coin balancing obtained using our formulae and our computer simulation.

3 Analysis and Results

This section compares the results of our simulations and formulae, and also compares our formulae with the formulae of the original Buffon's Needle and Coin Problems.

3.1 Comparing Formulae with Results of Simulations

We compare the mathematical formulae with results of our simulations of our variations of the Buffon's Needle and Coin Problems for a range of values. The following tables show some examples of the Chi-Square goodness of fit test statistics.

Running t = 1 000 000 trials for needle variation:

L	D	Simulated (S)	Expected (Pt)	χ^2 Statistic	Formula applied to find P
1.1	1	18026	18180.525926	1.337718	$\frac{2}{\pi}\left[\sqrt{\left(\frac{L}{D}\right)^2-1}+\sin^{-1}\left(\frac{D}{L}\right)\right]-1$
1.4	1	130680	130254.012129	1.601810	
2	1	435983	435991.124177	0.000268	
2.9	1	650948	651234.929677	0.362477	$1+\frac{2}{\pi}\left[\sin^{-1}\left(\frac{D}{L}\right)-2\sin^{-1}\left(\frac{2D}{L}\right)+\sqrt{\left(\frac{L}{D}\right)^2-1}-\sqrt{\left(\frac{L}{D}\right)^2-4}\right]$
3.3	1	698043	698006.543286	0.006305	
5	1	805800	805657.821115	0.129108	

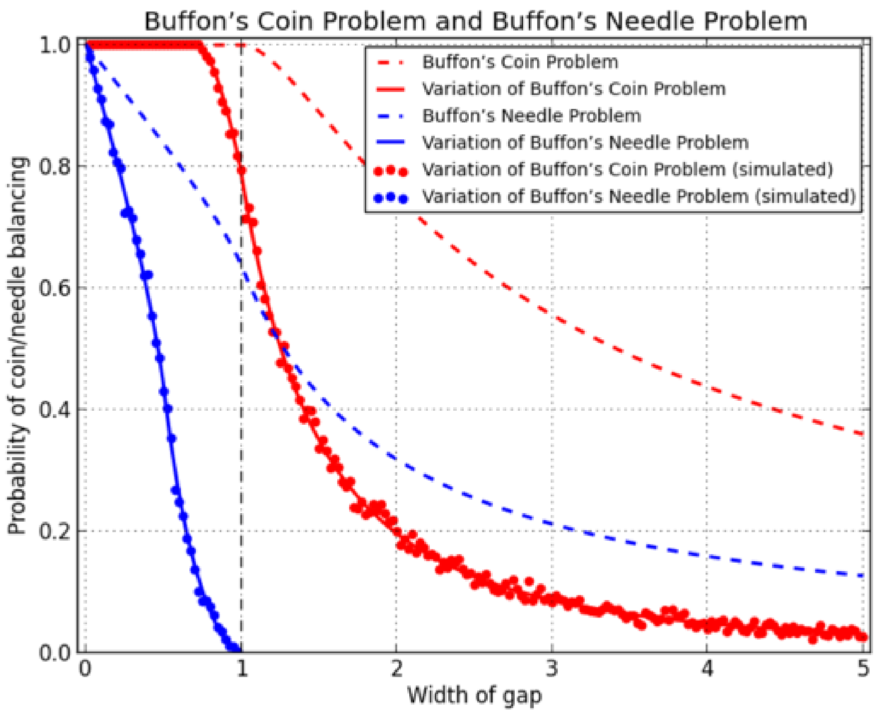
Running t = 100 000 trials for coin variation:

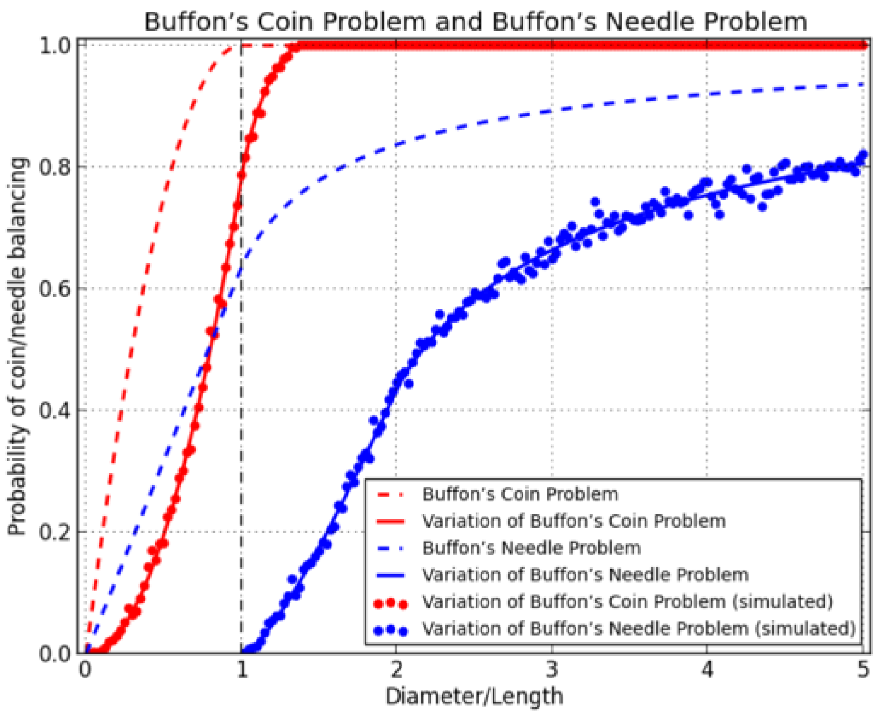
R	D	Simulated (S)	Expected (Pt)	χ^2 Statistic	Formula applied to find P
1	1	100000	100000	-	1
1	1.2	100000	100000	-	
1	$\sqrt{2}$	100000	100000	-	
1	1.5	99327	99331.962028	0.037105	$\sqrt{4\left(\frac{R}{D}\right)^2-1}+\left(\frac{R}{D}\right)^2\left[\pi-4\cos^{-1}\left(\frac{D}{2R}\right)\right]$
1	1.8	89815	89712.624262	1.135628	
1	1.9	84730	84706.3865961	0.043042	
1	2	78645	78539.816340	0.656407	$\pi\left(\frac{R}{D}\right)^2$
1	3	34873	34906.585040	0.049642	
1	5	12610	12566.370614	0.173249	

Our calculated χ^2 statistic was consistently less than 3.841 giving $p > 0.05$, hence we do not reject H_0. This means that there is no significant difference between using our formulae to calculate the probability of needle or coin balancing and using our computer program to simulate the needle and coin falling and keep count of balancing and compute the probability.

3.2 Comparing Formulae of Original and Variation of the Buffon's Needle and Coin Problems

The following are graphs for the original and variation of the Buffon's Needle and Coin Problems drawn on the same axes.





The red graphs show Buffon's Coin Problems while the blue graphs show Buffon's Needle Problems. Dashed lines indicate the probability of the original problem and solid lines indicate the probability of our variations. The dots are the probabilities we obtained through simulations; each dot corresponds to a thousand simulated trials.

The first graph shows the probability when the length of needle/diameter of coin is kept at a constant of one and the width of gap is varied, and vice versa for the second graph.

In both figures, our mathematical formulae and simulations for our variations of the Buffon's Needle and Coin Problems match very closely, as the Chi-Square goodness of fit test has shown that there is no significant difference between the results obtained from the formulae and the simulations.

The probabilities for our variations of the Buffon's Needle and Coin Problems are much lower than the original problems. This is expected because our definition of balance is more stringent in the variations than in the original problems.

The probability for the Buffon's Coin Problem and our variation is generally higher than the probability for the Buffon's Needle Problem and our variation. This is also within our expectation as there is a higher chance of being supported in the coin problem considering a bigger coin surface compared to a needle of negligible diameter and the use of a grid with four sides instead of two parallel lines.

4 Discussion

In this section, we discuss the limitations of our formulae and simulations, and how the formulae can be applied.

4.1 Limitations

The limitations of our formulae and simulations are due to the oversimplification of an object falling and the assumptions we made. Firstly, we assumed that edges are represented by two parallel lines or a single square grid. Secondly, we assumed that objects always fall flat instead of at various angles. Thirdly, we assumed that the mass of objects are uniformly distributed.

On the other hand, we can view our variation of the Buffon's Needle Problem to be a coin always landing vertically, with the length of the needle being the diameter of the coin.

4.2 Finding the Best Combination of Coins to Carry

Consider an authentic scenario of dropping some of your coins when pulling out your smartphone from your pocket while you are standing over a drain. You watch as your coins fall into the drain. To address this problem, we shall use the formulae derived for our variation of the Buffon's Coin Problem to consider the best combination of coins worth $1 to carry.

Supposed we limit the denominations to $1, 50-cent, 20-cent and 10-cent, where the diameters are 24.65 mm, 23.00 mm, 21.00 mm and 18.50 mm respectively (Monetary Authority of Singapore, 2013), and take the average length of gap between the grilles of drain covers in our school to be 30 mm. Using the formula

$$\text{P(coin balancing)} = \pi\left(\frac{R}{D}\right)^2 \quad \text{if } R \leq \frac{D}{2}$$

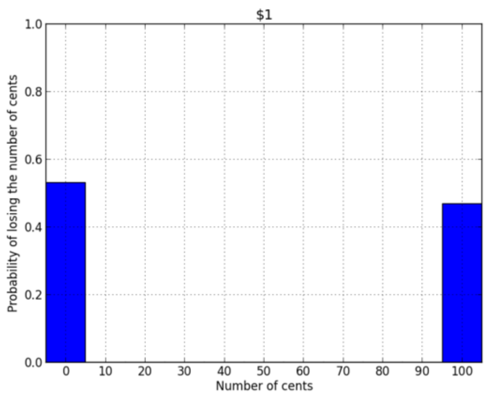
the probabilities of not losing a coin, in the event that the coin balances, are calculated:

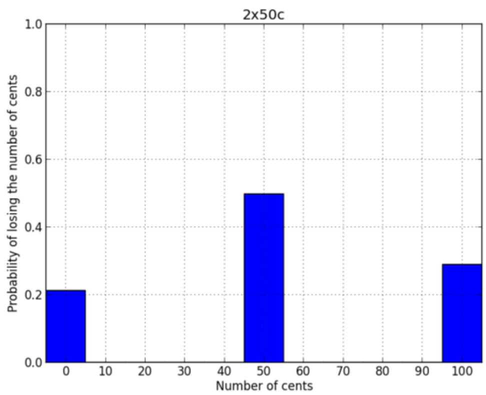
Probability \ Denominations	$1	50c	20c	10c
P(not losing)	0.5302507	0.4616396	0.3848451	0.2986695
P(losing)	0.4697493	0.5383604	0.6151549	0.7013305

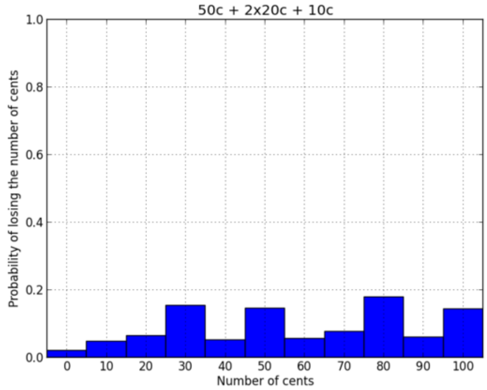
With the probability of losing individual coins of denomination \$1, 50-cent, 20-cent and 10-cent, the probability of losing a certain amount of money is calculated for each combination:

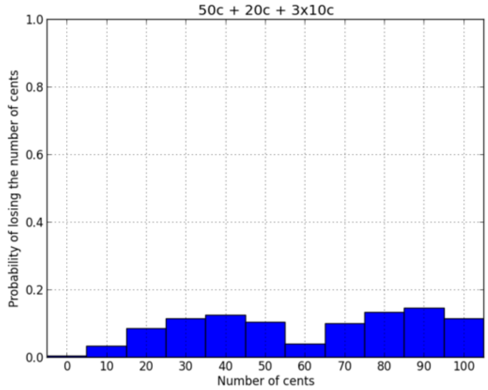
Losing / Denominations	0c	10c	20c	30c	40c	50c	60c	70c	80c	90c	\$1	Sum of Probabilities
\$1	0.53025										0.46975	1
2×**50c**	0.21311					0.49706					0.28983	1
50c + 2×**20c** + **10c**	0.02042	0.04795	0.06528	0.15329	0.05218	0.14633	0.05592	0.07613	0.17877	0.06085	0.14288	1
50c + **20c** + 3×**10c**	0.00473	0.03334	0.08586	0.11458	0.12515	0.10348	0.03889	0.10013	0.13363	0.14595	0.11424	1
50c + 5×**10c**	0.00110	0.01288	0.06050	0.14205	0.16678	0.07961	0.01502	0.07055	0.16566	0.19450	0.09135	1
5×**20c**	0.00844		0.06747		0.21569		0.34477		0.27555		0.08809	1
4×**20c** + 2×**10c**	0.00196	0.00919	0.02330	0.05876	0.09898	0.14088	0.19737	0.15012	0.18903	0.05999	0.07043	1
3×**20c** + 4×**10c**	0.00045	0.00426	0.01718	0.04392	0.08922	0.14529	0.18299	0.19745	0.16698	0.09593	0.05632	1
2×**20c** + 6×**10c**	0.00011	0.00148	0.00903	0.03196	0.07601	0.13585	0.19311	0.21352	0.17884	0.11506	0.04503	1
20c + 8×**10c**	0.00002	0.00046	0.00380	0.01840	0.05787	0.12566	0.19728	0.23247	0.20536	0.12267	0.03601	1
10×**10c**	0.00001	0.00013	0.00140	0.00878	0.03606	0.10162	0.19885	0.26682	0.23495	0.12260	0.02879	1

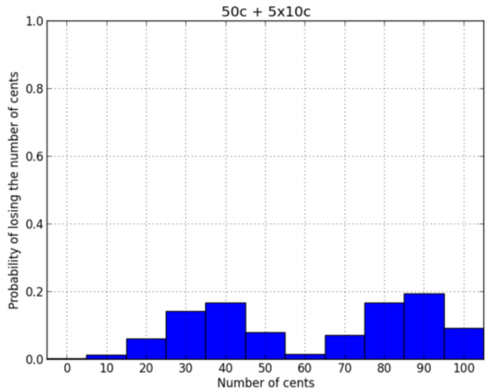
As seen from the above table, the probability of losing, say, 50 cents worth of money out of \$1 depends on the denominations that made up the \$1. We represented the probabilities in each row using bar graphs:

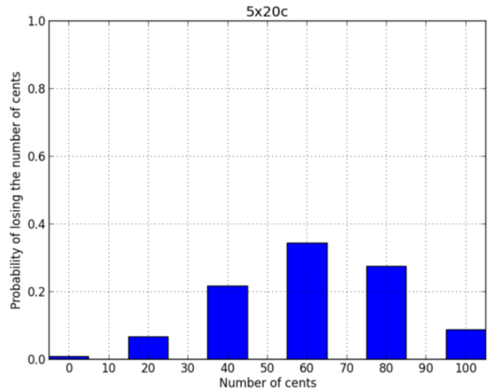


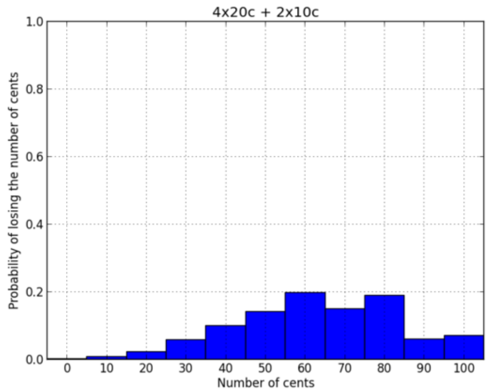


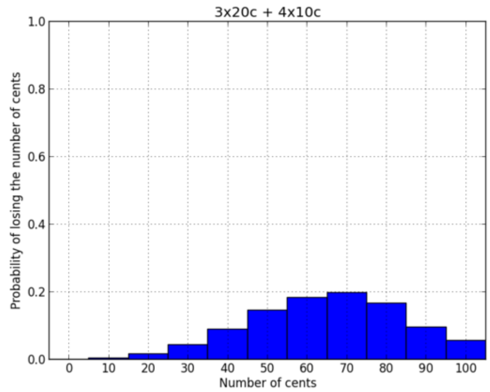

50c + 2x20c + 10c
Probability of losing the number of cents
Number of cents

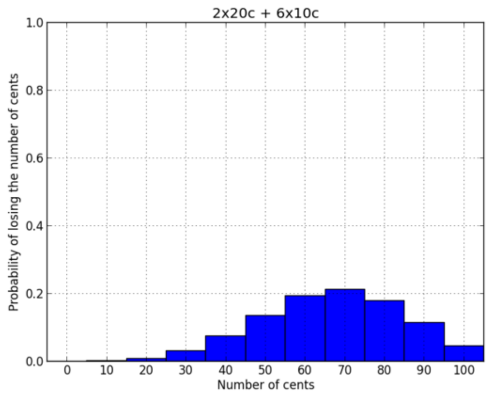

50c + 20c + 3x10c
Probability of losing the number of cents
Number of cents

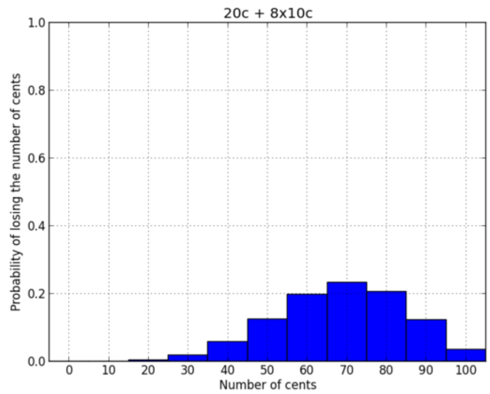

50c + 5x10c
Probability of losing the number of cents
Number of cents

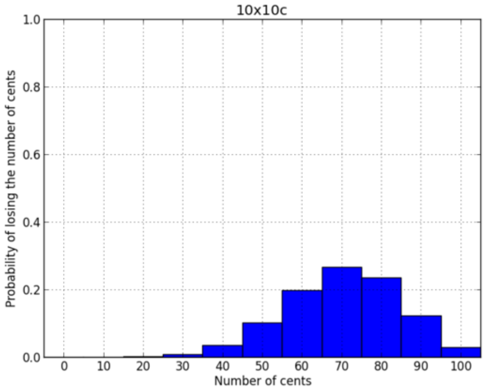

5x20c
Probability of losing the number of cents
Number of cents


4x20c + 2x10c
Probability of losing the number of cents
Number of cents


3x20c + 4x10c
Probability of losing the number of cents
Number of cents


2x20c + 6x10c
Probability of losing the number of cents
Number of cents


20c + 8x10c
Probability of losing the number of cents
Number of cents



To find the best combination of coins, the following criteria were used:

- Lower probability of losing all the $1 worth of money
- Lower probability of losing more than half the money

By comparing the graphs, “50c + 2×20c + 10c” and “50c + 20c + 3×10c” appear to be the best combinations; these are the two graphs that have probability spread out over all amounts and the probability of losing any particular amount stays below 0.2, while the other graphs generally have their probabilities concentrated more to the right indicating that the probability of losing coins is inclined towards losing a higher value than of a lower value.

Excluding the combination of a single $1 coin and the combination of two 50 cents coin which have a noticeably higher probability of losing all the $1 worth of money than the other combinations, the combinations of “50c + 2×20c + 10c” and “50c + 20c + 3×10c” are also found to have the lowest probability of losing 60 cents or more, 0.51455 and 0.53284 respectively.

To avoid losing more money in the event that we drop $1 worth of coins, we would recommend carrying $1 worth of coins in a combination of one 50-cent, two 20-cent and one 10-cent coins.

5 Conclusion

We made an attempt to model a needle or coin falling through gaps using the original Buffon's Needle and Coin Problems with consideration of the centre of gravity. We obtained the following formulas which were checked against our computer simulations:

$$\text{P(needle balancing)} = \begin{cases} \frac{2}{\pi}\left[\sqrt{\left(\frac{L}{D}\right)^2 - 1} + \sin^{-1}\left(\frac{D}{L}\right)\right] - 1, & \text{if } D < L \leq 2D \\ 1 + \frac{2}{\pi}\left[\sin^{-1}\left(\frac{D}{L}\right) - 2\sin^{-1}\left(\frac{2D}{L}\right) + \sqrt{\left(\frac{L}{D}\right)^2 - 1} - \sqrt{\left(\frac{L}{D}\right)^2 - 4}\right], & \text{if } L > 2D \end{cases}$$

where L is the length of the needle and D is the distance between the 2 parallel edges.

$$\text{P(coin balancing)} = \begin{cases} \pi\left(\frac{R}{D}\right)^2, & \text{if } R \leq \frac{D}{2} \\ \sqrt{4\left(\frac{R}{D}\right)^2 - 1} + \left(\frac{R}{D}\right)^2\left[\pi - 4\cos^{-1}\left(\frac{D}{2R}\right)\right], & \text{if } \frac{D}{2} < R < \frac{D\sqrt{2}}{2} \\ 1, & \text{if } R \geq \frac{D\sqrt{2}}{2} \end{cases}$$

where R is the radius of coin and D is the length of side of the square grid.

If we imagine the needle to be a coin landing vertically, where the length of the needle is the diameter of the coin, the formulae for the probability of needle balancing can be used to approximate the probability of the coin balancing when the coin lands vertically instead of landing flat, like in the case of the Buffon’s Coin Problem.

An area of improvement would be to compare the results of our formulae and simulations with real-life experiments to see how well they model a needle or a coin falling onto metal grilles, balancing or falling through. Nonetheless we applied our formulae to find the combination of coins worth $1 that has least probability of losing more than half of the money in the event that all the coins fall onto the drain cover. Based on our calculation, one should carry a combination of one 50-cent, two 20-cent and one 10-cent coins.

In our project, we chose to derive the formulae and at the same time write programs to simulate our problems and compare the two results. Although computer simulations cannot be used to prove conclusively that our formulae are correctly derived, they served as a means to verify our formulae and we found that useful; in the event that the formulae and the simulated results differ greatly, it implied that either or both were wrong.

In the future we may extend our project to include grids of different shapes such as other regular polygons and see how the probability varies with the shapes of the grid.

References

Andrew, A. M. (1979). Another efficient algorithm for convex hulls in two dimensions. *Information Processing Letters , 9(5),* 216-219.

Brown, E. (n.d.). *Equilibrium, Balance and Stability*. Retrieved 30 December, 2014, from https://www.msu.edu/course/kin/830/equilib_bal_stab.doc

Cooker, D. (n.d.). *numexpr*. Retrieved 26 December, 2014, from https://github.com/pydata/numexpr

Hunter, J. D. (2007). Matplotlib: A 2D graphics environment. *Computing In Science & Engineering, 9 (3),* 90-95.

Lauritzen, N. (2009, March). *Lecture on Convex Sets.* Retrieved 30 December, 2014, from http://www.fmf.uni-lj.si/~lavric/lauritzen.pdf

Monetary Authority of Singapore. (2013). *Coin Details*. Retrieved 26 December, 2014, from Monetary Authority of Singapore: http://www.mas.gov.sg/newcoins

Rezhdo, A. (2011). *Buffon's Needle Problem*. Retrieved 17 December, 2014, from https://math.dartmouth.edu/archive/m20f11/public_html/RezhdoProj.pdf

Rossum, G. (1989). *Python Programming Language*. Retrieved 26 December, 2014, from http://www.python.org

Siergist, K. (n.d.). *Buffon's Problems*. Retrieved 30 December, 2014, from Virtual Laboratories in Probability and Statistics: http://www.math.uah.edu/stat/buffon/Buffon.html

Van Der Walt, S., Colbert, S. C., & Varoquaux, G., (2011). The NumPy Array: A Structure for Efficient Numerical Computation. *Computing in Science & Engineering, 13(2),* 22-30.

Yale University. (1997-1998). *Chi-Square Goodness of Fit Test*. Retrieved 26 December, 2014, from http://www.stat.yale.edu/Courses/1997-98/101/chigf.htm

---

Our computer programs are accessible at http://tiny.cc/bce and our source codes are found at https://github.com/wei2912/bce-simulation