從醉漢走路到輸不起的莊家

陳光澄

致理科技大學 資訊管理系

小岢是個高二的學生，某日在數學課裡聽老師介紹「醉漢走路」(drunkard's walk)的例子，感到非常有趣，於是放學回家後，迫不及待地想與剛升上國中的妹妹小晴分享：

【醉漢走路碰壁版】有位醉漢喝完酒，左搖右晃在一條窄巷中徘徊。巷子非常窄，乃至於醉漢只要向左踏一步就會碰到左邊牆壁，向右踏兩步就會碰到右邊牆壁(請參考附圖 1)。假設醉漢所踏出的每一步，向左和向右的機會相同，那麼最終他會撞到左邊牆壁的機率是多少呢？

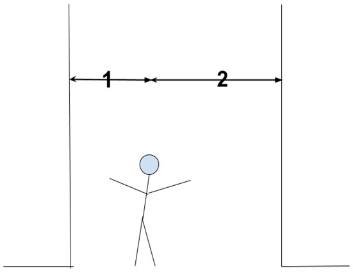

圖 1

聽完哥哥的描述，小晴陷入了短暫的沉思，不久後眼中便閃爍著慧婕的光芒，斷言醉漢會撞到左邊牆壁的機會應該是 2/3！小岢豎起大拇指，誇讚還沒有正式學過機率的妹妹具有不錯的直覺，並進一步解釋的機率是如何求出來的：醉漢最終會撞到左邊牆壁的情況，跨出腳步的方向與過程可記錄為 {左}、{右左左}、{右左右左左}、{右左右左右左左}、...將所有可能的機率加總，(利用「無窮等比級數和公式」)可求得整體機率為

$$\frac{1}{2}+\left(\frac{1}{4}\right)\frac{1}{2}+\left(\frac{1}{4}\right)^2\frac{1}{2}+\left(\frac{1}{4}\right)^3\frac{1}{2}\cdots=\frac{\frac{1}{2}}{1-\frac{1}{4}}=\frac{2}{3}$$

小晴雖然不很清楚哥哥計算機率所用的公式，卻也進一步提出了額外的看法：既然是醉漢「走路」，按理說最好也要能夠結合「向前移動」的成分，而不單單只是「左搖右晃」像螃蟹一樣只能向左或向右「平移」。於是花了點時間和哥哥討論，一起「開發」出一個醉漢走路的「新版本」。

【醉漢走路前進版】

來到公元二十五世紀的未來世界，城市的街道已經演進為智慧型自動移運系統。不過基於環保考量，智慧道路僅在感應到民眾移動時才會提供前進的動力。未來世界雖然先進，然而還是少不了貪杯的醉漢。為了解決「酒醉上路」的問題與精簡警察人力，智慧道路僅有左右各兩個步幅的寬度，不設安全欄杆，左右三步處則各有一排水溝等候著醉漢的「醉架光臨」(因踩空而掉落的醉漢，將隨自動傳輸水道系統移運至「茫然收容中心」集中管理)。設想某個月黑風高的夜裡，一位醉得厲害的酒鬼，手裡拿著酒瓶左搖右晃擺盪在智慧道路的正中央，每次的左搖或右晃的踏步，智慧道路將自動提供醉漢前進一個步幅的動力，且不算太遠的前方（100 步？ 500 步？）恰好是醉漢溫暖的家（請參考附圖 2)。假設醉漢所踏出的每一步，向左搖或向右晃的機會相同。請問，最終醉漢能夠安全地返抵家門嗎？

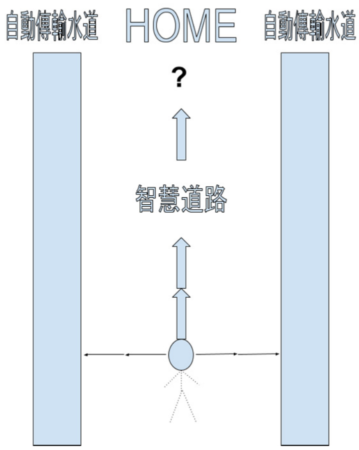

圖 2

開發出新版本的醉漢走路以後，小晴斬釘截鐵地認定醉漢一定回不了家，理由是醉漢離家總有一段距離，儘管智慧道路會提供前進的動力，但如此頻繁地左搖右晃，結果一定會在返抵家門前跌落水溝。小岢在數學課裡學過一點機率，所以堅持醉漢還是有那麼一點點機會可以順利回到家。當兩人為此爭執不休之際，爸爸剛好也下班了，因為正值 COVID-19 肺炎疫期，於是拔下口罩摺疊丟入垃圾桶後，立即加入了兩兄妹的討論戰局。

你們知道嗎？「醉漢走路」的例子，是數學裡的一種統計模型的應用，它有個正式名稱叫做「隨機漫步」(random walk)。在機率論的發展史裡，它是「賭徒破產問題」(Gambler's Ruin Problem） 的一個特例（請參考附圖 3)。

【賭徒破產問題】

假設賭徒口袋裡有 r 元，賭徒與莊家的總資產是 n 元，（n、r 皆為正整數，通常莊家的財力 $n-r$ 元遠大於賭徒口袋所擁有的 r 元）。賭徒和莊家玩遊戲，每次遊戲結束，賭徒有 p（可先假設為 $1/2$）的機會贏得 1 元，$1-p$ 的機會輸掉 1 元，下圖的圓圈（狀態）內顯示的數字即為賭徒口袋中的錢數。當 $n<\infty$，隨著遊戲持續進行，賭徒有可能面臨破產（抵達狀態 0）或贏得系統之總資產（抵達狀態 n）的結局。不過若 $n\to\infty$，$p\leq 1/2$，賭徒最終將避免不了會回到最左邊的狀態 0，亦即口袋空空的破產命運。

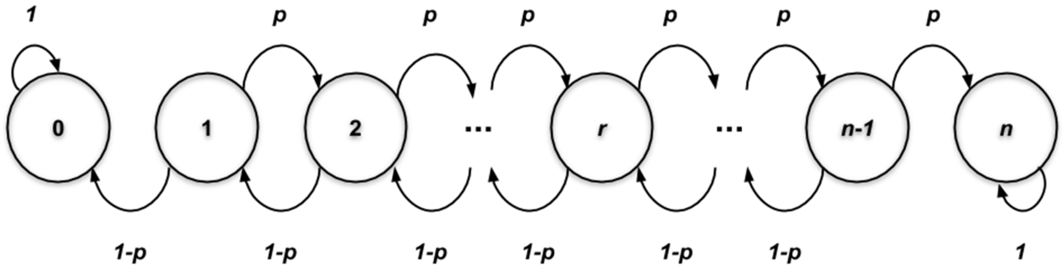

圖 3

小晴瞪大了眼睛，完全想像不到「醉漢」和「賭徒」竟然會在數學問題裡扯上關連。爸爸再進一步解釋：如果把狀態圖作如下的調整（請參考附圖 4），左右邊的狀態稱為「吸收態」，畢竟醉漢不論是撞到窄巷左右兩邊的牆壁，亦或跌入智慧道路左右兩側的水溝裡，在滿身醉意的情況下，恐怕一時之間也爬不起來。如此【賭徒破產問題】就搖身一變成為【醉漢走路碰壁版】或【醉漢走路前進版】。這可說是數學裡「一般化」思維的實際應用，也是數學模式之所以吸引人的地方。

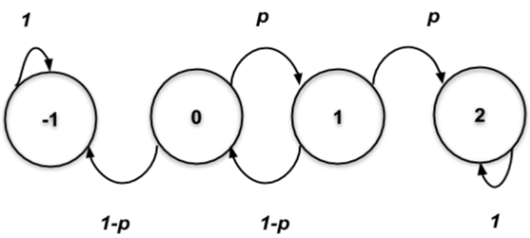

圖 4

至於你們所開發的【醉漢走路前進版】，究竟醉漢最終能否安全地返抵家門？為了更充分瞭解這個問題的可能發展，我們可以連同前面的一些例子，透過電腦程式來進行「隨機模擬」，也就是說，將數學模式的特徵寫入電腦程式，透過電腦高速運算的特性，針對一些在現實裡無法真正執行或難以重複的工作與試驗（如賭博的過程、醉漢的踏步）進行擬真操作。在正式進行電腦模擬之前，爸爸補充了一個重要的數學性質：

【馬可夫性質】

考慮一系列的隨機試驗，每次試驗的結果以 $X_i,\ i=1,2,3,\dots,n$ 表示，若一系列的試驗結果 $\{X_0,X_1,X_2,\cdots\}$（X_0代表初始狀態），滿足

$P(X_{n+1}=j|X_n=i_n,X_{n-1}=i_{n-1},X_{n-2}=i_{n-2},\cdots,X_0=i_0)=P(X_{n+1}=j|X_n=i)=P_{ij}$

（第 $n+1$ 次實驗結果只與前一次，亦即第 n 次實驗結果相關）

其中 X_n 代表系統在時間 n 的狀態，P_{ij} 代表將由目前的狀態 i 轉移至狀態 j 的機率。則稱 $\{X_0,X_1,X_2,\cdots\}$具有馬可夫性質（Markov Property）。（請參考文獻 5，p.137）

有沒有發現我們前面所討論的三個例子，無論是左搖右晃的醉漢、左右前行的醉漢、進行賭局的賭徒，他們在某個時間點的狀態（醉漢所在位置或賭徒輸贏的錢數），都只和前一個狀態有關，所以醉漢不會從 0 的位置直接跳到 2，賭徒輸贏的錢數也不可能從 +3 突然變成 -5，因為在我們的條件設定下，醉漢或賭徒的狀態轉移都服從馬可夫性質。

以前述【醉漢走路碰壁版】為例，因為醉漢在 $\{-1,0,1,2\}$ 四個狀態間的轉移服從馬可夫性質，因此當左搖右晃的機率相同時（請參考附圖 4 在 $p=1/2$ 的情況），我們可以將他在四個狀態間的轉移機率利用「一步轉移機率矩陣」（one-step transition probability matrix）M 描述。轉移機率矩陣 M 的列代表醉漢目前所在的狀態，行則代表醉漢下一步可能前往的狀態：（亦即，矩陣的元素用來描述從上一個狀態轉移至下一個狀態的轉移機率）

$$\boldsymbol{M}=\begin{array}{c c} & \begin{array}{cccc}\mathbf{-1} & \mathbf{0} & \mathbf{1} & \mathbf{2}\end{array} \\ \begin{array}{c}\mathbf{-1}\\ \mathbf{0}\\ \mathbf{1}\\ \mathbf{2}\end{array} & \begin{bmatrix}1 & 0 & 0 & 0\\ .5 & 0 & .5 & 0\\ 0 & .5 & 0 & .5\\ 0 & 0 & 0 & 1\end{bmatrix}\end{array}$$

其中 $\{-1,2\}$ 這兩個狀態稱為「吸收態」，因為醉漢碰到牆壁就停止左搖右晃的腳步了。由於醉漢在狀態間的轉移滿足馬可夫性質，若以 A 代表矩陣 M之所有狀態所構成的集合，可將 n 步轉移機率描述為：$p_{ij}^{(n)}=\sum_{k\in A}p_{ik}\cdot p_{kj}^{(n-1)}$，其中 $p_{ij}^{(0)}=1$，若 $i=j,\ p_{ij}^{(0)}=0$，若 $i\neq j$，並透過 $M^{(\boldsymbol{n})}=\underbrace{M*M*\cdots M}_{n\ times}$， 計算各狀態間 n 步轉移機率所構成之「n 步轉移機率矩陣」。舉例來說，$M^{(\mathbf{2})}=M*M=\begin{array}{c c} & \begin{array}{cccc}\mathbf{-1} & \mathbf{0} & \mathbf{1} & \mathbf{2}\end{array} \\ \begin{array}{c}\mathbf{-1}\\ \mathbf{0}\\ \mathbf{1}\\ \mathbf{2}\end{array} & \begin{bmatrix}1 & 0 & 0 & 0\\ .5 & .25 & 0 & .25\\ .25 & 0 & .25 & .5\\ 0 & 0 & 0 & 1\end{bmatrix}\end{array}$ 是描述醉漢接下來兩步的狀態轉移機率之「兩步轉移機率矩陣」。設想若醉漢的初始位置在狀態 0 ，則第一步有 0.5 的機率向左搖（撞到左邊牆壁停止，-1 為吸收態，即 $p_{0,-1}=0.5$），也有 0.5 的機率向右晃（來到狀態 1，即 $p_{0,1}=0.5$）。若是後者發生，才有可能跨出第二步，因此

接下來各有 $0.5 * 0.5 = 0.25$ 的機率向左搖（由狀態 0 出發，兩步後回到狀態 0）或向右晃（由狀態 0 出發，兩步後來到吸收態 2），這兩種可能發生的兩步轉移，可分別透過上述 $M^{(2)}$ 矩陣之第二列第二行元素 $p_{0,0}^{(2)} = 0.25$ 與第二列第四行元素 $p_{0,2}^{(2)} = 0.25$ 來描述其轉移機率。當醉漢的初始位置在狀態 1 時，兩步轉移機率的考慮方式亦相同（$p_{1,-1}^{(2)} = p_{1,1}^{(2)} = 0.25$）。

長期而言，經過若干次的左搖右晃後，醉漢終將從 $\{0,1\}$ 這兩個狀態（稱為「過渡態」)撞擊到左側或右側的牆壁(「吸收態」)，轉移機率亦將趨於穩定（稱為「穩態機率」)。若以 $p_{i,k}$ 代表由狀態 i 轉移至狀態 k 的一步轉移機率，$\pi_{i,j}$ 代表由狀態 i 轉移至吸收態 j 的穩態機率，則穩態機率必滿足 $\pi_{i,j} = \sum_{k\in A} p_{i,k} \cdot \pi_{k,j}$（請參考文獻 6，p.110-115）。可利用上述一步轉移機率矩陣 P，建立以下聯立方程式組求解：

$$\begin{cases}\pi_{-1,-1} = p_{-1,-1} \cdot \pi_{-1,-1} + p_{-1,0} \cdot \pi_{0,-1} + p_{-1,1} \cdot \pi_{1,-1} + p_{-1,2} \cdot \pi_{2,-1} \\ \pi_{0,-1} = p_{0,-1} \cdot \pi_{-1,-1} + p_{0,0} \cdot \pi_{0,-1} + p_{0,1} \cdot \pi_{1,-1} + p_{0,2} \cdot \pi_{2,-1} \\ \pi_{1,-1} = p_{1,-1} \cdot \pi_{-1,-1} + p_{1,0} \cdot \pi_{0,-1} + p_{1,1} \cdot \pi_{1,-1} + p_{1,2} \cdot \pi_{2,-1} \\ \pi_{2,-1} = p_{2,-1} \cdot \pi_{-1,-1} + p_{2,0} \cdot \pi_{0,-1} + p_{2,1} \cdot \pi_{1,-1} + p_{2,2} \cdot \pi_{2,-1}\end{cases} \Rightarrow$$

$$\begin{cases}\pi_{-1,-1} = \pi_{-1,-1} + 0 + 0 + 0 = 1 \\ \pi_{0,-1} = 0.5 + 0.5 \cdot \pi_{1,-1} \\ \pi_{1,-1} = 0.5 \cdot \pi_{0,-1} \\ \pi_{2,-1} = 0 + 0 + 0 + \pi_{2,-1} = 0\end{cases} \Rightarrow \pi_{0,-1} = 2/3,\ \pi_{1,-1} = 1/3$$

$$\begin{cases}\pi_{-1,2} = p_{-1,-1} \cdot \pi_{-1,2} + p_{-1,0} \cdot \pi_{0,2} + p_{-1,1} \cdot \pi_{1,2} + p_{-1,2} \cdot \pi_{2,2} \\ \pi_{0,2} = p_{0,-1} \cdot \pi_{-1,2} + p_{0,0} \cdot \pi_{0,2} + p_{0,1} \cdot \pi_{1,2} + p_{0,2} \cdot \pi_{2,2} \\ \pi_{1,2} = p_{1,-1} \cdot \pi_{-1,2} + p_{1,0} \cdot \pi_{0,2} + p_{1,1} \cdot \pi_{1,2} + p_{1,2} \cdot \pi_{2,2} \\ \pi_{2,2} = p_{2,-1} \cdot \pi_{2,2} + p_{2,0} \cdot \pi_{0,2} + p_{2,1} \cdot \pi_{1,2} + p_{2,2} \cdot \pi_{2,2}\end{cases} \Rightarrow$$

$$\begin{cases}\pi_{-1,2} = \pi_{-1,2} + 0 + 0 + 0 = 0 \\ \pi_{0,2} = 0.5 \cdot \pi_{1,2} \\ \pi_{1,2} = 0.5 \cdot \pi_{0,2} + 0.5 \\ \pi_{2,2} = 0 + 0 + 0 + \pi_{2,2} = 1\end{cases} \Rightarrow \pi_{0,2} = 1/3,\ \pi_{1,2} = 2/3$$

因 $\{-1,2\}$ 為吸收態，故 $\pi_{-1,-1} = \pi_{2,2} = 1$，$\pi_{-1,2} = \pi_{2,-1} = 0$

故得知當 $n \to \infty$ 時，n 步轉移機率矩陣為 $M^{(\infty)} = \lim_{n\to\infty} M^{(n)} = \begin{array}{c c} & \begin{array}{cccc} \mathbf{-1} & \mathbf{0} & \mathbf{1} & \mathbf{2} \end{array} \\ \begin{array}{c} \mathbf{-1} \\ \mathbf{0} \\ \mathbf{1} \\ \mathbf{2} \end{array} & \begin{bmatrix} 1 & 0 & 0 & 0 \\ 2/3 & 0 & 0 & 1/3 \\ 1/3 & 0 & 0 & 2/3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{array}$，

其中第二列第一行所呈現的穩態機率 $\pi_{0,-1} = 2/3$，就是小晴當初所直覺判斷，初始位置在 0 的醉漢最終會撞到左邊牆壁的機率，也是我們接下來希望透過電腦來檢驗的結果！

因為小豈與小晴兩兄妹尚未正式接觸過程式語言，所以爸爸透過虛擬碼(pseudo code)的描述方式，解釋希望藉由程式讓電腦執行的工作與步驟：

【模擬-醉漢走路碰壁版】

```
設定 TIMES=0 ，用來紀錄醉漢碰壁時移動的總步數
設定 POSITION=0，代表醉漢所在的原始位置
重複執行
        RAND=電腦隨機由{-1,1}集合中選出的一個數字
        POSITION=POSITION+RAND，PRINT(POSITION)
        TIMES=TIMES+1
        IF POSITION=-1 OR POSITION=2 ，PRINT(TIMES) ，執行終止
```

（正式程式碼請參考文獻 7）

上述虛擬碼可以模擬並記錄醉漢左搖右晃的過程（位置），並列印出醉漢撞壁時跨出的總步數（單次）。為了透過模擬檢驗<u>小晴</u>的直覺猜測、<u>小岦</u>的公式推導與前述 n 步轉移機率矩陣在 $n \to \infty$ 時的計算結果，亦即：醉漢最終會撞到左邊牆壁的機會為 2/3，我們可以將程式略做修改，加上一個外迴圈執行 100,000（或更多）次的模擬，並紀錄其中醉漢撞到左邊牆壁的次數，藉由多次模擬的過程與結果，估算醉漢撞到左邊牆壁的機率是否接近 2/3。

```
設定 INDEX=0，用來紀錄醉漢撞到左邊牆壁的次數
執行迴圈 100,000 次
        設定 TIMES=0 ，用來紀錄醉漢碰壁時移動的總步數
        設定 POSITION=0，代表醉漢所在的原始位置
        重複執行
                RAND=電腦隨機由 {-1,1} 集合中選出的一個數字
                POSITION=POSITION+RAND
                TIMES=TIMES+1
                IF POSITION=-1 OR POSITION=2 ，重複執行終止
IF POSITION=-1，INDEX=INDEX+1
ELSE ，INDEX=INDEX
結束迴圈
PRINT(INDEX/100000)
```

（正式程式碼請參考文獻 7）

下表是透過電腦模擬 10 回醉漢碰壁 100,000 次的結果（請參考附表 1），可以看出醉漢撞到左邊牆壁的估算機率的確都非常接近理論值 $\pi_{0,-1} = 2/3$。現實生活裡要找到真正的

醉漢來「試驗」100,000 次，簡直就是「不可能的任務」，藉助電腦來模擬的好處就是能夠做到快速、擬真。

附表 1

模擬回別	1	2	3	4	5	6	7	8	9	10
估算機率	0.6669	0.6673	0.6693	0.6684	0.6671	0.6665	0.6658	0.6686	0.6660	0.6636

【模擬-醉漢走路前進版】

爸爸進一步提示小呰與小晴兩兄妹：你們有沒有發現，在醉漢能夠透過智慧道路前進的情況下，他能否平安回到家的關鍵不就是最終左搖與右晃所踏出的「總步數」嗎? 若醉漢的家在前方 100 步之處，因為左搖或右晃都算一步，所以若是最後踏出的總步數不小於 100，就代表他最終能平安回到家（醉漢已藉由智慧道路提供的 100 步前進動力返抵家門）。我們只需將前述虛擬碼略作修改，即可透過模擬估算在經歷了一系列左搖右晃的前進過程後，醉漢能夠安然返抵家門的機率。

```
設定 INDEX=0，用來紀錄醉漢可以安然返抵家門的次數
執行迴圈 100,000 次
    設定 TIMES=0 ，用來紀錄醉漢移動的總步數
    設定 POSITION=0，代表醉漢所在的原始位置
    重複執行
        RAND=電腦隨機由 {-1,1} 集合中選出的一個數字
        POSITION=POSITION+RAND
        TIMES=TIMES+1
        IF POSITION=-3 OR POSITION=3 OR TIMES=100，重複執行終止
IF TIMES=100，INDEX=INDEX+1
ELSE ，INDEX=INDEX
結束迴圈
PRINT(INDEX/100000)
```

（正式程式碼請參考文獻 7）

「怎麼電腦計算出來的機率都等於 0！難道妹妹的直覺有可能是對的嗎？」看到小呰露出一副無法置信的表情，爸爸不禁笑著繼續解釋：對於無法確定的問題，數學裡有種思

維叫做「特殊化」，往往有助於我們在探索問題答案的過程中，能夠更深刻地瞭解問題的本質。舉例來說，我們可以先設定醉漢的家就在前方不遠四步之處，並以「左、右」分別代表醉漢左搖或右晃的踏步，這樣就很容易運用你們所學過的數學裡的「窮舉法」，將所有可能的狀況列示如下表（請參考附表 2，加框字代表醉漢踩空）。

附表 2

可能路徑	結果	可能路徑	結果
左左右左	抵達家門	右右左右	抵達家門
左右左左	抵達家門	右左右右	抵達家門
左左右右	抵達家門	右右左左	抵達家門
左右右左	抵達家門	右左左右	抵達家門
左右左右	抵達家門	右左右左	抵達家門
左右右右	抵達家門	右左左左	抵達家門
右右$\boxed{右}$	跌入右邊水溝	左左$\boxed{左}$	跌入左邊水溝

計算求得醉漢可以安然返抵家門的機率為 $12\left(\frac{1}{2}\right)^4=\frac{12}{16}=0.75$，不幸跌落水溝的機率為 $2\left(\frac{1}{2}\right)^3=\frac{2}{8}=0.25$，只需將上面的虛擬碼中，IF 的判斷改為 TIMES=4，模擬 10 回，即可得以下模擬結果（請參考附表 3）。

附表 3

模擬回別	1	2	3	4	5	6	7	8	9	10
估算機率	0.7489	0.7513	0.7500	0.7505	0.7515	0.7506	0.7502	0.7504	0.7498	0.7530

「真的耶，結果都很接近理論值 0.75！」接下來小岢和小晴又在爸爸的指導下，分別完成了醉漢的家在前方 10 步與 50 步之處的 10 回 100,000 次模擬過程的虛擬碼，結果估算出醉漢可以安然返抵家門的兩列機率值（請參考附表 4）。

附表 4

醉漢離家	1	2	3	4	5	6	7	8	9	10
10 步	0.3179	0.3163	0.3159	0.3182	0.3167	0.3142	0.3159	0.3176	0.3171	0.3176
50 步	0.0009	0.0009	0.0011	0.0009	0.0008	0.0010	0.0010	0.0010	0.0009	0.0009

「有沒有辦法透過數學方法，計算出醉漢的家在前方 10 步的情況下，他能夠安抵家門的確實機率呢？」小蚩的眼裡，閃爍著一種想要追根究底的炯炯有神目光。

這個問題顯然比醉漢離家四步的情況要複雜一點。或許我們可以發揮數學裡的逆向思維工具「排除法」，並運用「組合方法」的技巧來解決這個問題。爸爸接著解釋：醉漢能夠安抵家門的機率確實不容易計算，因為當他的家在前方 10 步的情況下，考慮左搖或右晃的步數，共有 $\binom{10}{0}+\binom{10}{1}+\binom{10}{2}+\cdots+\binom{10}{10}=2^{10}=1024$ 種可能組合，不過其中有超過一半的組合是無法實現的（亦即，醉漢會在十步之內跌落排水溝），我們可以將這些無法實現的狀況排除，便能求得確實的機率。

首先，由於左搖右晃的特性，當醉漢的家在前方 10 步的情況下，跌落水溝僅可能發生在醉漢的第 3、5、7、9 步。同樣以「左、右」分別代表醉漢左搖或右晃的踏步，考慮如下機率：

- 醉漢於第 3 步跌落：路徑包含{左左左}或{右右右}，共占 $\frac{2}{8}$ 的機率。
- 醉漢於第 5 步跌落：路徑包含{xx 左左左}或{xx 右右右}，其中 xx 須為左右或右左，共占 $2\cdot C_1^2\cdot\frac{1}{32}=\frac{4}{32}$ 的機率，亦或路徑包含{左左右左左}或{右右左右右}，共占 $\frac{2}{32}$ 的機率。
- 醉漢於第 7 步跌落：路徑包含{xxxx 左左左}或{xxxx 右右右}，其中 xxxx 須為 2 左 2 右，共占$2\cdot C_2^4\cdot\frac{1}{2^7}=\frac{12}{128}$ 的機率，亦或路徑包含{xxxx 右左左}或{xxxx 左右右}，其中前後 xxxx 分別須為 3 左 1 右與 3 右 1 左，但須排除左左左右或右右右左的情況，共占 $2\cdot(C_1^4-1)\cdot\frac{1}{2^7}=\frac{6}{128}$ 的機率。
- 醉漢於第 9 步跌落：路徑包含{xxxxxx 左左左}或{xxxxxx 右右右}，其中 xxxxxx 須為 3 左 3 右，但須排除左左左右右右或右右右左左左的情況，共占 $2\cdot(C_3^6-2)\cdot\frac{1}{2^9}=\frac{36}{512}$ 的機率，亦或路徑包含{xxxxxx 右左左}或{xxxxxx 左右右}，其中前後 xxxxxx 分別須為 4 左 2 右與 4 右 2 左，但須排除左左左左右右、左左左右左右、左左左右右左、左左右左左右、左右左左左右、右左左左左右 等六種情況（以 4 左 2 右為例，4 右 2 左的組合可透過左右交換求得，故不贅述），共占 $2\cdot(C_2^6-6)\cdot\frac{1}{2^9}=\frac{18}{512}$ 的機率。

因此，醉漢的家在前方 10 步的情況下，他能夠安抵家門的確實機率為 $1-\frac{2}{8}-\frac{4+2}{32}-\frac{12+6}{128}-\frac{36+18}{512}=\frac{162}{512}=0.31640625$，與電腦模擬結果（附表 4 的第一列）接近。顯然，醉漢的家在前方 50 步的情況會複雜許多，大概很難再用類似方法歸納吧？畢竟 2^{50} 已是千兆等級的龐大數字呢！爸爸帶著略為詭異的表情笑著說。

「真的是這樣嗎？怎麼我總覺得計算結果似乎有規律可循呢？」小尚對爸爸的說法表示懷疑，並提出自己的發現：前面的計算式裡，整體的機率 1 所扣掉的每一項機率，似乎都滿足某種規律呢！與爸爸討論過後，小尚寫出了下列猜想：

〈 小尚猜想 〉【醉漢走路前進版】中所描述的醉漢，家在前方 n 步時能夠安抵家門的機率為 $P(n)=1-\sum_{k=0}^{\left\lfloor\frac{n-3}{2}\right\rfloor}\frac{2\cdot 3^k}{2^{2\cdot k+3}}=1-\sum_{k=0}^{\left\lfloor\frac{n-3}{2}\right\rfloor}\left(\frac{1}{4}\right)\left(\frac{3}{4}\right)^k$，$n\geq 3$（$n\leq 2$ 時必能安抵家門）

雖然這只是經過觀察歸納得到的猜想算式，小尚還是拿它來計算了當醉漢的家分別在前方 50 與 100 步的情況下，他能夠安然返抵家門的機率：

$$P(50)=1-\sum_{k=0}^{23}\frac{2\cdot 3^k}{2^{2\cdot k+3}}\approx 0.001003391，P(100)=1-\sum_{k=0}^{48}\frac{2\cdot 3^k}{2^{2\cdot k+3}}\approx 7.550955\cdot 10^{-7}$$

「看來猜想所計算的機率 $P(50)$ 與模擬的結果（附表 4 的第二列）大致相同，醉漢的家在前方 50 步的情況下，他能夠平安回家的機率就只剩下千分之一左右，難怪一開始設定離家 100 步遠會估算出 0 的機率（100,000 次當中成功不到一次），把實驗次數增加到 1,000,000 次試試看吧！」趨於一致的計算結果讓小尚多了點信心，於是他打算增加電腦模擬的次數，看看醉漢的家在前方 100 步的情況下，是否也能得到與猜想算式一致的結果。

這次電腦就沒那麼快顯示答案了，畢竟百萬次的醉漢走路對於電腦而言，需要更大規模計算能耐（CPU & RAM） 與運算時間的投入，於是爸爸說，我們先來談談和醉漢走路師出同門的賭徒破產問題吧！（後續結果：與小尚猜想計算的結果大致相同，透過 10 回 1,000,000 次的模擬，可以估算出醉漢能安然返抵家門的機率應該小於 1/1,000,000，但並非全無可能，所以小尚的直覺沒錯，醉漢還是有機會回家的！不過爸爸也提醒小尚，模擬歸模擬，猜想歸猜想，都不是數學上的嚴格證明，前述的猜想需要更進一步的證明才「算數」，僅可視作是對於研究主題更深入探討的重要里程碑。至於這個「小尚猜想」的證明其實不難，主要的關鍵在於醉漢從狀態 0 開始左搖右晃三步若仍未跌落水溝，則其位置必處於於狀態 -1 或 1 ，以位於狀態 1為例，醉漢接下來兩步的可能走法與相對應之機率如附圖 5 所示。至於前述猜想的詳細證明過程，就留待有興趣的讀者做進一步探討囉！）

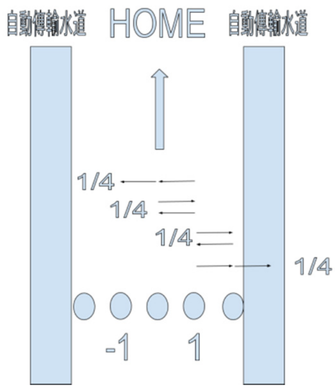

圖 5

稍後爸爸又補充了另外一種求解的方法：建立狀態集 $A=\{-3,\ -2,\ -1,0,1,2,3\}$ 的「一步轉移機率矩陣」M，並進一步計算 $M^{(10)}$、$M^{(50)}$ 與 $M^{(100)}$，則亦可獲得與前述猜想一致之結果（請參考文獻 7 的計算結果）。儘管不同求解方法之思考角度可能有很大的差異，然而只要方法正確，終將獲致「殊途同歸」的結論，這也是數學問題引人入勝的特徵之一。

【模擬-賭徒破產問題】

你們不妨想想看，為什麼從古至今，天下的賭徒，只要是拿金錢當作賭注的，往往都難逃破產的命運呢？」爸爸提出了這個問題當作開場白，想順便聽聽兩兄妹的想法..

「因為莊家可以操控賭徒獲勝的機率。」

「因為莊家的資本通常都很大，以小搏大是很難勝出的。」

「很好！你們提出來的重點剛好都是問題的關鍵。」爸爸誇讚了兄妹倆後，繼續說明：為了避免破產，有些較理性的賭客，會設定虧損上限作為停損點（損失的金錢達到上限就停止），或是避免贏錢後受到誘惑所設定的贏利上限（賺得的金錢達到上限就停止）。然而，對於賭性堅強的玩家，設定的標準往往抵不上賭局的誘惑。而他們又是如何一步步走上破產的道路呢？」

雖然賭博不是好事，不過為了舉例方便，假設你們倆的口袋各有三塊錢，玩猜拳遊戲，輸的人要給贏的人一塊錢，遊戲持續進行到其中一個人的口袋清空（輸光）為止，如果你們的猜拳技術旗鼓相當，那麼遊戲的過程就會和前述的【醉漢走路前進版】有異曲同工之妙。可以設定醉漢向左搖代表小豈（可視為莊家）獲勝，醉漢向右晃代表小晴（可視為賭徒）獲勝，醉漢跌落左側或右側的自動傳輸水道則代表遊戲結束（小晴或小豈輸光口袋所有的錢）。而醉漢藉由系統提供之動力所前進的步數，即為分出勝負的決勝局之前，遊戲已持續進行的累積次數。

小岢與小晴不約而同地微笑點頭表示贊同。爸爸接著提到：其實遊戲的過程也和前面所提到的【醉漢走路碰壁版】非常類似，只是這次輸與贏的對決，等同於醉漢左搖右晃在一個窄巷的正中央，而其左右三步處各有一面牆壁等著他的「醉架光臨」，你們說是不是？

「對耶！ 醉漢撞到右邊三步的牆，就相當於妹妹贏走了我所有的錢，醉漢撞到左邊三步的牆，就相當於我贏走了妹妹所有的錢！」聽完爸爸的描述，小岢彷彿發現了新大陸一般。

那麼如果是爸爸和小晴玩，但是爸爸財力比較雄厚（莊家），口袋裡有十塊錢，在猜拳技術勢均力敵的情況下，小晴可以成功翻盤（贏得兩人擁有的金錢總數十三塊錢）的機會是多少呢？過去的學者對於破產問題的研究，多以賭徒的角度出發，透過設定系統的總資產 n（即賭徒與莊家擁有的金錢總數）為上限，討論口袋擁有 r 元的賭徒，在經歷一連串的賭局過後，如何破產或取得系統所擁有的總資產（達狀態 0 或 n）。接著爸爸適時補充了以下理論作說明：

【古典破產問題（轉移機率）】

考慮一個具有兩個吸收狀態作為邊界的隨機漫步系統，狀態空間設為 $\{0,1,2,3,\dots,n\}$ 代表賭徒口袋所擁有的金錢數，其中 0、n 為代表賭局結束的兩個吸收狀態，分別描述賭徒破產（狀態 0）與賭徒贏得系統之總資產（狀態 n），故賭局結束，而 $1,2,3,\dots,n-1$ 則為過渡狀態。令 $u_i=\pi_{i,0}$， $v_i=\pi_{i,n}$ 分別代表賭徒由過渡狀態 i 出發最後進入狀態 0 與狀態 n 的機率，p、q 分別代表賭徒單次賭局贏與輸的機率，則可透過以下差分聯立方程組，計算 u_i 與 v_i（即 $1-u_i$）的機率：

$$\begin{cases} u_1=q+pu_2 \\ u_i=qu_{i-1}+pu_{i+1}，2\le i\le n-2，而\ u_0=1，u_n=0\ 為兩邊界條件，可求得 \\ u_{n-1}=qu_{n-2} \end{cases}$$

$$u_r=\begin{cases} \frac{n-r}{n}， & r=1,2,\cdots,n-1，若\ p=q \\ \frac{(q/p)^n-(q/p)^r}{(q/p)^n-1}， & r=1,2,\cdots,n-1，若\ p\ne q \end{cases}，且當\ n=\infty\ 時\quad u_r=\begin{cases} 1， & 若\ p\le q \\ (q/p)^r， & 若\ p>q \end{cases}$$

（詳細證明過程請參考文獻 1，p.114-p.116。或參考文獻 4，p.344-p.347）

換句話說，若是小晴能夠翻盤，贏走爸爸的十塊錢，口袋裡的錢數就來到吸收態 $\{13\}$。因為猜拳技術勢均力敵所以 $p=q$ ，故小晴獲勝機率的理論值便是 $v_3=1-u_3=1-\frac{13-3}{13}\approx 0.2307692$。我們用虛擬碼來規劃一下模擬的步驟，並且看看計算的結果是否接近理論值吧！

```
設定 INDEX=0，用來紀錄小晴在 100000 場賭局裡贏的局數
執行迴圈 100000 次
        設定 TIMES=0 ，用來紀錄一場遊戲執行的總步數
        設定 POSITION=0，代表小晴賺取的金錢數（亦可設定為小晴口袋擁有的
        金錢數為 3）
        重複執行
                RAND=電腦隨機由 {-1,1} 集合中選出的一個數字
                POSITION=POSITION+RAND
                TIMES=TIMES+1
                IF POSITION=-3 OR POSITION=10 ，重複執行終止（亦即擁有金
                錢數為 0 or 13）
IF POSITION=10，INDEX=INDEX+1
ELSE ，INDEX=INDEX
結束迴圈
PRINT(INDEX/100000)                                    （正式程式碼請參考文獻 7）
```

模擬 10 回 100000 場賭局，可得以下模擬結果（請參考附表 5），顯示小晴約有兩成多的翻盤機會，與前述理論值 $v_3 \approx 0.2307692$ 的結果頗為接近。

附表 5

模擬回別	1	2	3	4	5	6	7	8	9	10
估算機率	0.2298	0.2297	0.2309	0.2308	0.2313	0.2313	0.2288	0.2316	0.2305	0.2311

前面曾經提及「醉漢走路」在機率論的發展史裡，可視為「賭徒破產問題」 的一個特例，所以【醉漢走路碰壁版】之中，醉漢轉移至吸收態（撞牆）的機率，自然也可以使用 $u_r = \frac{n-r}{n}$ 來計算。將狀態空間重新調整成 $\{0,1,2,3\}$ ，果然也能求得醉漢開始位於狀態 1，最終會撞到左側牆壁的機率，即 $u_1 = \frac{3-1}{3} = \frac{2}{3}$，最終會撞到右側牆壁的機率，即 $v_1 = 1 - u_1 = \frac{1}{3}$。

如果換成某位賭性堅強的爸爸到賭場下注，口袋裡準備了十美元，一次下注一美元，在賭場的資本額為爸爸賭金 10,000 倍（這算是相當保守的估計）的情況下，爸爸有機會贏下整個賭場嗎？這次爸爸獲勝機率的理論值僅有 $v_{10} = 1 - u_{10} = 1 - \frac{100010-10}{100010} \approx 9.9999 \cdot 10^{-5}$ （還不到 $\frac{1}{10000}$）。

設定 **INDEX=0**，用來紀錄爸爸在 **100** 場賭局裡能夠獲勝的局數
執行迴圈 **100** 次
設定 **TIMES=0** ，用來紀錄一場遊戲執行的總步數
設定 **POSITION=0**，代表爸爸賺取的金錢數（亦可設定為爸爸口袋擁有的金錢數 **10**）
重複執行
RAND=電腦隨機由 **{-1,1}** 集合中選出的一個數字
POSITION=POSITION+RAND
TIMES=TIMES+1
IF POSITION=-10 OR POSITION=100000 ，重複執行終止（金錢數為 **0 or 100010**）
IF POSITION=100000，**INDEX=INDEX+1**
ELSE ，**INDEX=INDEX**
結束迴圈
PRINT(INDEX/100) （正式程式碼請參考文獻 7）

模擬 10 回，得以下模擬結果（請參考附表 6），可以看出安排的十回電腦模擬中，每回包含 100 個賭性堅強的爸爸，其中並沒有任何一位能夠贏得走賭場的所有資產。從前述理論值的計算，可知大約每 10000 位爸爸之中才有一位有機會「翻盤」。至於某位爸爸一路過關斬將，最終以「屢戰屢勝」之姿順利奪下賭場的機率僅僅只有 $\left(\frac{1}{2}\right)^{100000}$ ，其值顯然趨近於 0。

附表 6

模擬回別	1	2	3	4	5	6	7	8	9	10
估算機率	0	0	0	0	0	0	0	0	0	0

「看來絕大部分的賭徒都無法翻盤，那麼他們的錢到底是怎麼輸光的呢？總共又要經歷多少次賭局才會破產呢？」為了解決小晴心中的疑惑，於是爸爸又適時補充了以下理論作說明：

【古典破產問題（賭局次數期望值）】
考慮前述具有兩個吸收狀態作為邊界的隨機漫步系統，狀態空間設為 $\{0,1,2,3,\dots,n\}$ 代表賭徒口袋所擁有的金錢數，其中 0、n 為代表賭局結束的兩個吸收狀態，分別描述賭徒破

產（狀態 0）與賭徒贏得系統之總資產（狀態 n），故賭局結束，而 $1,2,3,\ldots,n-1$ 則為過渡狀態。令 T_r 代表賭徒一開始時擁有 r 元的情況下，直至此賽局結束所需之推移次數，則 $D_r = E(T_r)$ 為有限值，並滿足下述非齊次差分方程式：

$D_r = q(D_{r-1}+1) + p(D_r+1) = qD_{r-1} + pD_{r+1} + 1$，$1 \le r \le n-1$，且有邊界值 $D_0 = D_n = 0$，可求解得

$$D_r = \begin{cases} r(n-r) & ，\text{若 } p = q \\ \frac{r}{q-p} - \frac{n}{q-p}\frac{(q/p)^r-1}{(q/p)^n-1} & ，\text{若 } p \neq q \end{cases}，\text{當莊家資產無限時，} \lim_{n\to\infty} D_r = \begin{cases} \frac{r}{q-p}，& \text{若 } p < q \\ \infty，& \text{若 } p \ge q \end{cases}$$

（詳細證明過程請參考文獻 4，p.348-p.349）

換句話說，前例中爸爸的口袋裡有十美元，賭場的資本額為爸爸賭本的 10,000 倍，在這樣的情況下，平均而言，爸爸期望會在經歷 $D_{10} = 10\cdot(100{,}010-10) = 1{,}000{,}000$ 場賭局後，輸光口袋所有的錢或是贏走賭場的所有資產，這個賭局次數的期望值大到超乎我們的想像！

不過電腦模擬的結果又是如何呢？若是我們在前述虛擬碼的「重複執行終止」前加入「PRINT(TIMES)」，並模擬十次，可以藉此瞭解這十位賭性堅強的爸爸到底在歷經多少場賭局過後，才輸光了口袋裡的 10 元美金或贏走賭場的所有資產。（請參考附表 7）。

附表 7

模擬次別	1	2	3	4	5	6	7	8	9	10
賭局數	506	15,128	924	190	292	26	3,408	192	3,479,820	214

從結果可以看出這幾位賭性堅強的爸爸最終都輸光了口袋裡的十美元，並在輸光前經歷了數十次到數百萬次不等的賭局！為什麼透過電腦模擬他們最終所經歷之賭局次數，竟有如此大的差異，似乎難以用來描述理論的期望賭局次數呢？別忘了電腦所模擬的這幾次對局，結果都只顯示了賭徒「破產」的情況，然而理論上賭徒還是有非常非常小微小的機率，在經歷了數百億甚至更多次的賭局過後，終於得以「翻盤」！由於可能相當冗長的對局過程，其隨機變異會變得相當難控制，因此有限次的電腦模擬結果，無法真實反映出理論上賭局將會執行之總次數的期望值。這是電腦模擬的局限之處，也是為何前項模擬僅針對 100 場賭局進行的緣故。因為光是要模擬記錄這些對象的冗長對戰過程所牽涉的繁雜計算，也會讓一般電腦的運作變得非常吃力。

接下來，我們透過記錄狀態（輸贏金額，即前述虛擬碼的 POSITION）的變化資料所繪製的折線圖，觀察某四位好賭的爸爸在賭局裡金額輸贏變化的詳細過程（請參考附圖 6）。

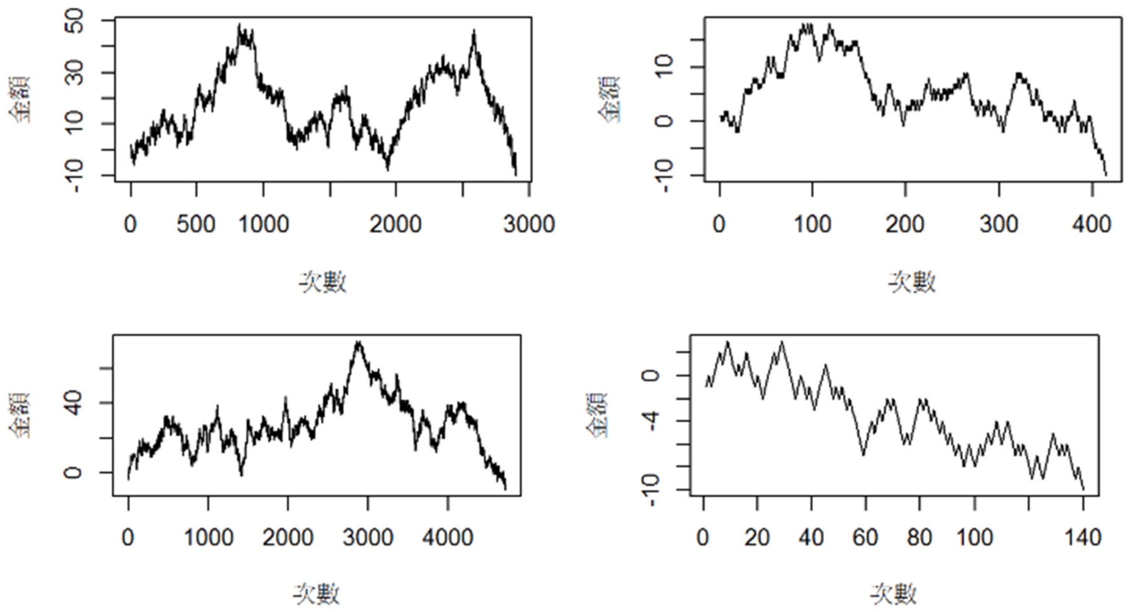

圖 6

在上述四組模擬資料中，第一位好賭的爸爸最輝煌的成績是贏到 50 元左右（即口袋裡擁有接近 60 元），也曾在接近第 2,000 次對決前，幾乎輸光了口袋裡的錢，然後在第 2,500 次對決後，迎向第二個獲利的高峰期，當然最終還是在第 2,904 次對決時把口袋裡所有的錢輸光，這就是絕大部分賭徒的宿命（其它三位好賭的爸爸皆然，只是過程有快有慢，其變化趨勢圖有點像在紀錄買貴了的股票，其股價終將在賭盤的「盤整」過後，跌至當初所購買的水準之下）。不過，當莊家的財力不夠雄厚時（例如玩家與莊家的財力比是 1:10 而非 1:10000 時），莊家還是很有可能慘遭玩家「滅頂」 的，這也就是為什麼在電視新聞或報章雜誌裡，偶爾會看到一些民間「組頭」在賭局（例如選舉地下賭盤）尚未完全結束前就提早「跑路」的新聞。

當然，莊家可也不是省油的燈。全天下所有贏不了的玩家，面對的自然是一群輸不起的莊家。賭博遊戲的「勝率」 是莊家可以輕易控制的，只要稍微調整一下前述虛擬碼的一些參數，例如，讓 RAND=電腦隨機由 {-1,1,-1,1,-1} 集合中選出的一個數字，玩家的勝率馬上降為 2/5，因此能讓莊家破產的機率就更低了！以下結果是設定玩家金錢 10 元，莊家金錢 100 元，在玩家勝率分別為 499/1000、49/100、2/5 的情況下，以電腦模擬 10,000 次的對決共 10 回，計算玩家能夠成功翻盤（讓莊家破產）的機率（請參考附表 8）。相對應之機率的理論值分別為

$$v_r = 1 - u_r = 1 - \frac{(501/499)^{110} - (501/499)^{10}}{(501/499)^{110} - 1} \approx 0.07383794，當\ \ p = 499/1000$$

$v_r = 1 - u_r = 1 - \frac{(51/49)^{110} - (51/49)^{10}}{(51/49)^{110} - 1} \approx 0.006110713$，當 $p = 49/100$

$v_r = 1 - u_r = 1 - \frac{(3/2)^{110} - (3/2)^{10}}{(3/2)^{110} - 1} \approx 0$，當 $p = 2/5$

附表 8

玩家勝率	1	2	3	4	5	6	7	8	9	10
499/1000	.0774	.0731	.0741	.0709	.0724	.0731	.0679	.0772	.0735	.0718
49/100	.0060	.0059	.0063	.0057	.0046	.0062	.0063	.0073	.0054	.0065
2/5	0	0	0	0	0	0	0	0	0	0

理論與模擬的結果皆顯示玩家能成功翻盤這件事,會隨著勝率降低,變得越來越困難。換言之，玩家與莊家勝率接近時，或許玩家還有翻盤的機會。不過當勝率的差距拉大，縱使莊家的資本只有玩家的十倍，也導致玩家就算再努力，也很難翻轉頹勢。上表的最後一列顯示了在 2/5 的勝率下，模擬共 10 回玩家與莊家的 10,000 次對決，結果顯示機率掛零，代表在電腦模擬的過程裡，玩家沒有一次成功翻盤過，與根據理論計算所得的結果一致（有興趣的讀者可以對照理論值，修改程式碼裡的參數，估算與驗證不同狀況下，玩家翻盤的微小機率）。畢竟無論是否合法，莊家就是得靠開設賭局來謀求生計。然而賭博這種「遊戲」是不會主動去迷戀賭客的，所以「酒不醉人人自醉，賭不迷人人自迷」對於嗜賭者而言，應該也是頗為貼切的說法。面對理所當然輸不起的莊家，你還甘心做個到處碰壁的爛醉賭徒嗎？

參考文獻

1. 黃文璋 (1995)：**隨機過程**。台北市：華泰書局。
2. 胡守仁翻譯 (2009)：**醉漢走路**。台北市：天下遠見出版股份有限公司。
3. 凡異出版社編譯 (1984)：**概率萬花筒**(1984)。新竹市：凡異出版社。
4. Feller, W. (1968), *An Introduction to Probability Theory and Its Applications, Vol.1*, (3rd ed.)，John Wiley & Sons, New York.
5. Sheldon M. Ross (1993), *Introduction to Probability Models* (5th ed.)，ACADEMIC PRESS, INC.
6. Howard M. Taylor & Samuel Karlin (1994), *An Introduction to Stochastic Modeling* (revised ed.)，ACADEMIC PRESS, INC.
7. 陳光澄 (2020)：drunkard's walk（本文章之 R 語言程式實作），取自 Rpubs 網頁超連結 https://rpubs.com/ggcchen/602648