

出手角度及速度對籃球投籃 準確性影響之探討

王仁宗／中山大學
戴玉林／中山大學
許秀桃／中山大學
曾淑平／臺南女子技術學院

壹、前言

投籃是一個精密的動作，從接球，起身，舉球，到出手投籃，每一個細節都影響著進球與否。過去的研究，顯示出影響投籃準確度的因素，除了本身技巧之外，投籃距離也是影響因素之一，梁家音等人（1998）以一位甲組女籃選手為對象，探討其投籃動作型式是否會在某一關鍵性距離，產生階段性的表現，經過二階段的實驗後，結果發現：即使是甲組選手，其籃球投籃的動作模式明顯會受投籃距離的影響。另一研究則探討肌力訓練對籃球基本投籃動作技巧的效果（周禹廷，19987），以 50 位大學生為對象，實施十二週肌力訓練後，結果發現：投籃命中率進步了 3.40 次（7.36 次至 10.76 次）。除此之外，影響投籃準確度的另一因素是拇指，麥財振等人（1995）在其研究探討手指對投籃成功率的影響，16 名甲組籃球選手，分別以運動貼布包紮各個手指，一一測驗罰球線投籃成功率，結果顯示：拇指的確是影響投籃成功的關鍵。游正忠等人（1997），為了提昇投籃動作的技術水平，以力學原理建立籃球投籃動作的分析模式，研究中以優秀籃球選手黃致豪為對象，拍攝其投籃動作加以分析和觀察，最後提出了投籃動作的建議：1、持球時雙手均勻分配控球於穩定狀態下；2、強調出手的尾隨動作，務必將動作完成；3、上臂肌力需強化；如此才能訓練出投籃的準確度。儘管以往在籃球投籃準確度的研究不少，不過在本論文中，主要目的在探討球員出手投籃

時，球離手的角度及速度，對於進球的影響。每個人的出手角度都不盡相同，也因此發展出不同的投籃風格，大致上來說，越高的投籃角度須要越大的出手速度。本論文將從運動力學的角度出發，當給定了球的出手角度及初速後，便決定了球的軌跡路線，也決定了進球與否。我們將求出在同一角度下，出手速度所允許的範圍，再求出在這樣的速度範圍內，所允許的距離誤差。這樣子求出來的，是一個「範圍」，當一個球員在那樣「大概」的距離，以「大概」的角度，「大概」的出手速度，便可以進球，這就是籃球運動員所稱「出手的感覺」，我們用運動力學的角度把這樣的感覺「量化」了。以下我們先討論對於將球投進籃框的運動力學。

貳、籃球軌跡及進球的力學模型

一、球的軌跡

我們知道，將球拋出後，球會因為重力的影響，其軌跡會是拋物線。從圖一中，我們假設球離手的初速是 V ，球與水平面的夾角是 θ ，那麼便可以知道球在水平方向與垂直地面方向的初速。

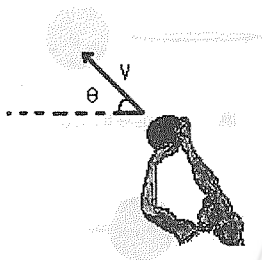
$$\text{水平方向：} V_x = V \cdot \cos \theta$$

$$\text{垂直方向：} V_y = V \cdot \sin \theta$$

接下來，由牛頓運動力學便可求得，球的水平進程及垂直進程為：

$$S_x = V_x t = V \cos \theta \cdot t$$

$$S_y = V_y t = V \sin \theta \cdot t - \frac{1}{2} g t^2$$



圖一：投籃示意圖

在這裡我們將球離手的點為原點，也就是高度及水平座標定為零的地方， g 是重力加速度。算出了每個時刻的水平進程與垂直進程，就等於算出了每個時刻的位置，也就是求出了球的軌跡，如此我們便能對進球與否作判斷。

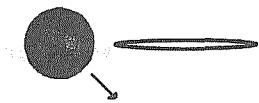
二、籃球進球的模型

對於進球的判斷，我們先說明不計籃板反彈進球，光以籃球與籃框之間作判斷。接下來我們大概將情形分為兩種，一種是球在籃框中心之前，一種是球在籃框中心之後。如果球在下降時到了籃框的高度，其水平距離是在籃框前緣之前，或籃框後緣之後，毫無疑問的這球是在籃框之外，根本不會進球，請看圖二。

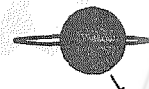
如果球的中心點到了籃框同高的高度，且水平位置是在籃框之內，也沒有碰到籃框，那就是標準的進球，請看圖三。但請注意，雖然球在籃框高度沒有碰到籃框，並不一定是空心球，也許球會在繼續下降一點高度時碰到籃框，這樣的球就是平常看到的彈框進球，而非空心球。

如果球的中心點在還沒下降到籃框同高的高度時，便碰到了籃框，那麼球會被籃框反彈，基本上這樣的球我們視為不進球，請看圖四。

事實上，即使籃球被籃框所反彈，還是有一點進球的可能。也就是說，碰到籃框，也不見得就彈出。請看圖五，當球碰到籃框邊緣時，根據牛頓第三運動定



圖二：不進球的情況



圖三：進球的情況

律，作用力與反作用力的原理，籃框對球必產生一個方向相反的作用力，也就是沿原路路線回去的方向。這作用力來自於球進來和反彈的速度差與籃框與球的接觸時間，也就是：

$$F = ma = m \frac{dv}{dt} = m \frac{V_{\perp} - (-r \cdot V_{\perp})}{dt} = m \frac{(1+r) \cdot V_{\perp}}{dt}$$

其中 m 是籃球的質量， r 是反彈的係數， dt 是籃框與球的接觸時間，是進球時的垂直速度分量。如果反彈係數 r 越大則速度差越大， r 最大為 1。

當這個力 F 的垂直分量大於球的重量時，其合力是向上的，球便會彈出，所以反彈的力量必須小於球的重力：

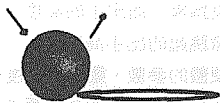
$$F \sin \theta_2 \leq mg$$

實際上此力如果只是稍大於球的重力，那麼此球雖會有點彈起，但還是能在籃框範圍內再掉回籃框，我們在投籃時，有時球會在籃框內來回彈幾次才進球就是這個原理，在這裡為簡化模型，我們限制反彈力要小於球的重力。

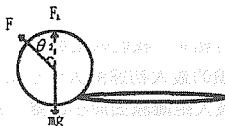
當我們把 F 代入此式則可求得進球的最小角度：

$$\theta_2 \leq \sin^{-1} \frac{mgdt}{m(1+r)V_{\perp}} = \sin^{-1} \frac{gdt}{(1+r)V_{\perp}}$$

從以上式子我們可以知道，進球與否實際上跟球的重量無關，而且當籃框給予球的反彈係數越大，則等號右邊越小，所能允許的角度範圍越小。這也就是為什麼硬的籃框比較不好投進球的原因，因為硬的籃框會反彈較大的力量和速度。



圖四：球反彈籃框而出



圖五：球反彈籃框的力圖

如果球是彈到籃框前緣內側，基本上在垂直反彈方向仍是相反的，仍可以用相同的條件作判斷。

不過實際上在計算時，因為要彈框且進球，必須要反彈係數很小，計算出的角度及速度限制範圍會很小，對於進球範圍的加大影響很小，在本文的程式結果我們便不討論此點。

參、程式說明與參數設定

本文程式以 C 語言寫成，在 Windows 上編譯後執行。程式是以牛頓運動力學與上述模型為基礎，當時間前進時，算出每個時刻的水平與垂直座標。時間前進的快慢要看間隔時間的給定，間隔時間給的越短，則程式所須時間較長，但結果會較準確，我們取時間間隔為 0.005 秒。

程式的目地，是在計算同一距離下，球在不同角度下欲進球所須的初速，這初速不是只有一個答案，而是一個範圍，所以我們會求出這個範圍。投籃角度上，我們取的投籃範圍，從 30 度起，每隔一度跑一次結果，算到 80 度。一般來說，球員投籃不會低於 30 度，那樣的角度太平坦而難以進球；球員也通常不會以超過 80 度接近直角的角度出手，那樣須要很高的初速，也就是很大的力道才能投出，不是正常的投籃方式。

接下來是參數的設定，首先，我們的研究設定是一個固定的中距離，取 4.5 公尺。球員出手的水平位置設為零，而出手的高度設定在 2.5 公尺，那大概是一個身高約 180 公分的球員稍微跳起的出手高度。

接下來是一些籃球相關硬體的參數，籃框的高度是三呎，換算成公制單位為 3.048 公尺，籃球的半徑為 12 公分，籃框的半徑為 22.5 公分。

肆、程式結果與討論

下表一我們列出程式執行結果，我們列出從出手角度 30 度到 80 度的結果，其中初速範圍是指要進球所須的最大初速減去最小初速，距離範圍是指在進球時的初速範圍內，可以容許的最大距離減去最小距離。跟距離有關的單位為公尺，跟速度有關的單位是公尺每秒。

這些數據乍看之下，似乎隨著投籃角度的增大，其最大初速以及最小初速也

表一：程式執行結果。(速度單位為公尺/秒，距離單位為公尺)

出手角度	初速範圍	最大初速	最小初速	距離範圍	最遠距離	最近距離
30	0.076	8.392	8.316	0.4	4.9	4.5
31	0.036	8.016	7.98	0.05	4.51	4.46
32	0.016	7.894	7.878	0	4.5	4.5
35	0.006	7.596	7.59	0	4.5	4.5
36	0.022	7.528	7.506	0.02	4.5	4.48
37	0.018	7.45	7.432	0.03	4.5	4.47
38	0.032	7.396	7.364	0.06	4.51	4.45
39	0.044	7.348	7.304	0.08	4.52	4.44
40	0.04	7.29	7.25	0.09	4.52	4.43
41	0.06	7.262	7.202	0.14	4.55	4.41
42	0.056	7.218	7.162	0.12	4.54	4.42
43	0.07	7.196	7.126	0.15	4.56	4.41
44	0.066	7.164	7.098	0.17	4.56	4.39
45	0.084	7.156	7.072	0.18	4.58	4.4
46	0.074	7.128	7.054	0.19	4.57	4.38
47	0.084	7.124	7.04	0.19	4.58	4.39
48	0.094	7.126	7.032	0.23	4.6	4.37
49	0.098	7.126	7.028	0.23	4.6	4.37
50	0.098	7.128	7.03	0.23	4.6	4.37
51	0.102	7.138	7.036	0.25	4.6	4.35
52	0.108	7.156	7.048	0.26	4.61	4.35
53	0.112	7.178	7.066	0.28	4.62	4.34
54	0.108	7.194	7.086	0.27	4.61	4.34
55	0.116	7.232	7.116	0.27	4.62	4.35
56	0.114	7.262	7.148	0.27	4.61	4.34
57	0.124	7.312	7.188	0.29	4.63	4.34
58	0.132	7.364	7.232	0.3	4.63	4.33
59	0.134	7.418	7.284	0.31	4.64	4.33
60	0.132	7.472	7.34	0.31	4.63	4.33
61	0.134	7.54	7.406	0.3	4.63	4.32
62	0.138	7.614	7.476	0.31	4.63	4.32
63	0.142	7.698	7.556	0.32	4.64	4.32
64	0.142	7.786	7.644	0.31	4.63	4.32
65	0.156	7.896	7.74	0.33	4.65	4.32
66	0.156	8.002	7.846	0.32	4.64	4.32
67	0.164	8.128	7.964	0.33	4.65	4.32
68	0.162	8.254	8.092	0.34	4.65	4.31
69	0.168	8.4	8.232	0.34	4.65	4.31
70	0.17	8.558	8.388	0.34	4.65	4.31
71	0.178	8.736	8.558	0.34	4.65	4.31
72	0.184	8.93	8.746	0.34	4.65	4.31
73	0.19	9.146	8.956	0.34	4.65	4.31
74	0.2	9.386	9.186	0.36	4.66	4.3
75	0.204	9.65	9.446	0.36	4.66	4.3
76	0.212	9.948	9.736	0.36	4.66	4.3
77	0.224	10.286	10.062	0.36	4.66	4.3
78	0.232	10.666	10.434	0.36	4.66	4.3
79	0.242	11.102	10.86	0.36	4.66	4.3
80	0.256	11.606	11.35	0.36	4.66	4.3

跟著成長，初速範圍也跟著變大；同樣地，最遠距離、最近距離及距離範圍也有大致增大的區勢。但仔細推敲其內容，我們可得到一些不同於直觀的結果，當我們將這些數據畫成圖，便可以得到一些結論。

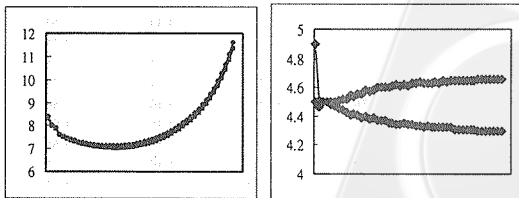
首先看下圖六，我們可以看到圖六左，關於最大初速與最小初速的曲線圖，這個圖明顯指出所須最大及最小初速並不絕對隨著角度增加而增加，而是先降後升，呈一個倒過來的山坡。而速度範圍的增加趨勢，請看圖七，則是也是先下降，然後隨然角度越大範圍越大，但其成長趨勢並不陡，也就是說雖然投籃角度越大，所增加的初速範圍並不一定符合效益。

一開始初速必須大，是因為在角度很平時，其速度垂直分量並不大，因此在空中停留的時間很短，所以必須加快速度才能加大水平速度分量，才能在下降期間將球投進籃框。在這段期間內，也由於投球的角度太平，水平速度太大，因此在下降時的角度也平，球碰到籃框的機會也大，自然進球的可能性較小，所以初速範圍並不大。

而隨著角度變大，速度垂直分量變大，在空間停留的時間夠久，籃球在下降時的角度也比較大，進球的機會較大，因此對所須的初速要求並不那麼嚴格，所以初速範圍便漸漸的擴大。值得注意的是，因為角度越大，在空中停留時間想必不短，所以垂直初速及水平初速都必須越大，增加區勢越加陡峭，但是初速範圍的增加區勢並沒有跟著陡峭，角度如果太大，進球的機會並不會跟著成正比成長，同時由於後來所須的初速過大，力量也得加大，不僅很難使力，力道也變得不好控制。

再來看到最大與最小距離的數據圖(圖六右)以及圖七，基本上距離範圍是隨著角度而增加的，但是增加趨勢是隨著角度越大而趨緩。

在圖右右的最左邊，一開始我們看到最大距離有一個值很大的點，這是由於在這角度有很大的初速範圍，使得在這樣大的範圍之內，出現了可以在比較遠的距離以同樣初速和角度投進這球，不過很少人會在這樣的角出手，所以我們不



圖六：(左)最大初速與最小初速(右)最大距離與最小距離

表二：程式執行結果。速度單位為公尺/秒，距離單位為公尺

出手角度	初速範圍	最大初速	最小初速	距離範圍	最遠距離	最近距離
46	0.074	7.128	7.054	0.19	4.57	4.38
47	0.084	7.124	7.04	0.19	4.58	4.39
48	0.094	7.126	7.032	0.23	4.6	4.37
49	0.098	7.126	7.028	0.23	4.6	4.37
50	0.098	7.128	7.03	0.23	4.6	4.37
51	0.102	7.138	7.036	0.25	4.6	4.35

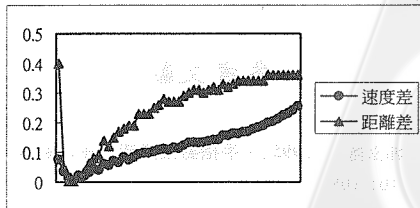
討論這點。過了一開始較怪異的點，再來是一段幾乎重疊的線段，原因還是出自角度太平，速度範圍在這段變的很小，當然能允許的距離差也小的可憐。

過了這段很平的區域，最遠距離開始變大，最近距離也開始變小，也就是說距離範圍也開始變大。不過在後來最遠及最近距離的增加趨勢變緩，允許的距離範圍也變的趨緩。這是因為在角度大時，球的初速變的很大，如果距離移動太多，很可能球就因為一小點時間的變化而跑出籃框之外。

如果我們試圖找出一個比較符合效益的出手角度，大概可以找到兩個區段，這些考量到出手初速不能太大、容許的初速範圍與距離範圍不能太小。

第一個區段出現在 46 到 51 度之間，請看表二(截取了表一的其中一段)，這段期間的初速範圍與距離範圍，其成長區勢及大小都比之前的大許多，最重要的是其最大及最小初速都比較小，也就是出手的力道不必太強，而出手力道如果不強，則比較好控制出手姿勢。這一區間大概是一般球員最常用的出手角度，也符合了數據計算及人體工學。

第二個區間出現在 57 到 64 度之間，請看表三(截取了表一的其中一段)，這段區間是初速範圍開始成長較陡的地方，其最大最小初速也還在能接受範圍之內，而其最大優勢在於出手距離，因為在這段區間內，所能容許的初速範圍大，而且最遠及最近距離幾乎一樣，因此可以在很大的距離差內用更大不同的初速出



圖七：速度範圍與距離範圍

表三：程式執行結果。速度單位為公尺/秒，距離單位為公尺

出手角度	初速範圍	最大初速	最小初速	距離範圍	最遠距離	最近距離
57	0.124	7.312	7.188	0.29	4.63	4.34
58	0.132	7.364	7.232	0.3	4.63	4.33
59	0.134	7.418	7.284	0.31	4.64	4.33
60	0.132	7.472	7.34	0.31	4.63	4.32
61	0.134	7.54	7.406	0.3	4.63	4.33
62	0.138	7.614	7.476	0.31	4.63	4.32
63	0.142	7.698	7.556	0.32	4.64	4.32
64	0.142	7.786	7.644	0.31	4.63	4.32

手，這是蠻令人驚訝的結果。

也就是說在這同樣的距離範圍內的容許的速度、角度範圍都很大！但是這樣的出手角度須要的初速比較大，須要比較大的力道，有些球員會採取這樣的出手角度。

伍、結論

從容許的速度與距離範圍來探討出手角度，是比較不同以往研究討論正確出手姿勢的方式。正確的投籃姿勢雖然是基本的，但它無法告訴你採取怎樣的出手角度較好，因為這些出手角度沒有絕對的好壞之分，只能靠感覺來決定出手力道跟速度。雖是如此，但球員們仍會傾向於採取容許誤差較大的出手方式，而從我們的數據中，也可以得到證明。而我們找出了兩段符合效益的出手角度區間，也比較了其優缺點，這兩段區間也確實是表較多球員採取的出手角度。

出手的感覺其實也跟這誤差範圍有關，當我們抓到那樣大概的出手點、力道、和遠近感，也就是角度、初速和距離，只要在大概範圍內，便可不斷的進球，那就是出手的感覺。

參考文獻

- 麥財振、許壬榮、郭玄隆。(1995)。手指對投籃成功率之影響。國立臺灣體育學報，6卷，101-109。
- 游正忠、洪得明。(1997)。籃球中鋒球員遠距離投籃定性分析。國立體育學院論叢，8卷第1期，175-186。

梁嘉音、卓俊伶、簡耀輝。(1998)。距離對籃球投籃動作型式的影響：個案研究，體育學報，25卷，199-208。

同禹廷。(1998)。肌力訓練對籃球基本投籃技巧學習及得分之影響。大專體育，38卷，40-47。

附錄

程式

```
#include <math.h>
#include <stdio.h>
#include <stdlib.h>

#define PI 3.141592
#define BASKET_HIGH 3.048
#define BALL_R 0.12
#define BASKET_RING 0.225
#define CONTACT_T 0.3
#define REBOUND 0.3
#define GRAVITY -9.8

double X_position();
double Y_position();
double X_velocity();
double Y_velocity();
int caculate();
int judge();

void main() {
    //定義初始值
    double init_x=0.,init_y=2.5;
    double init_v=3.,init_vstart=3.,init_vend=15.,dt=0.005;
    double theta=40.0,delta_v=0.002;
    double basket_dis=4.5;
    double start_a=0.,start_b=0.;
    double end_a=0.,end_b=0.;
    double distance=0.,dis_start,dis_end,dis_far,dis_near;
    int ja=0,jb=0;
    int stop=0;

    FILE *input_file_ans;
    FILE *input_file_ans2;
    input_file_ans = fopen("ans.dat", "w");
    input_file_ans2 = fopen("ans2.dat", "w");

    for(theta=30.; theta <= 80.; theta += 1.) {
        start_a=0.;
        start_b=0.;
        end_a=0.;
```

```

end_b=0.;
ja=0;
jb=0;
dis_far=basket_dis;
dis_near=basket_dis;

//=====本迴圈在計算相同角度下所能允許的速度範圍=====
for(init_v=init_vstart; init_v<init_vend; init_v += delta_v) {
    stop = caculate(init_x,init_y,init_v,(theta*PI)/180.,dt,
                    basket_dis);

    if(stop>=2 && ja==0) {
        start_a=init_v;
        ja=1;
    }
    if(stop<=1 && ja==1) {
        end_a=init_v-delta_v;
        ja=0;
    }

    if(stop==2 && jb==0) {
        start_b=init_v;
        jb=1;
    }
    if(stop=2 && jb==1) {
        end_b=init_v-delta_v;
        jb=0;
    }
}

//=====本迴圈在計算相同角度所允許的速度下,其允許的距離誤差=====
for(init_v=start_a; init_v<=end_a; init_v += delta_v) {
    ja=0;
    dis_start=basket_dis;
    dis_end=basket_dis;
    for(distance = basket_dis-0.5 ; distance<= basket_dis+0.5 ;
        distance += 0.01) {

        stop = caculate(init_x,init_y,init_v,(theta*PI)/180.,dt,distance);
        if (stop>1 && ja==0) {dis_start = distance;
            ja=1;}
        if (stop<=1 && ja==1) {dis_end = distance-0.05;
            ja=0;}
    }

    dis_far = (dis_far > dis_end) ? dis_far : dis_end;
    dis_near = (dis_near < dis_start) ? dis_near : dis_start;
}

//=====將結果計錄至檔案中=====
fprintf(input_file_ans,
        "theta = %lf  va_range = %lf (%lf-%lf)  distance_range = %lf
        (%lf-%lf)  Area=%lf\n",theta,end_a-start_a,end_a,start_a,
        dis_far-dis_near,dis_far,dis_near,(end_a-start_a)*(dis_far-dis_near));

fprintf(input_file_ans2,

```

```

"%lf %lf %lf %lf %lf %lf %lf %lf %lf\n",
theta,end_a-start_a,end_a,start_a,dis_far-dis_near,dis_far,dis_near,
(end_a-start_a)*(dis_far-dis_near));
}
fclose(input_file_ans);
fclose(input_file_ans2);
}

//=====輸入球的初始速度位置後,計算每個時刻的位置,最後判斷進球與否====
int caculate(double init_x,double init_y,double init_v,double theta,
double dt,double basket_dis) {
double nowt=0.;
double tempx=0.,tempy=2.0,tempvy=0.,tempvx=0.;
int loops=0;
int stop=0;

while ( tempy>0 && loops<1000) {
tempx = X_position(init_x,init_v,theta,nowt);
tempy = Y_position(init_y,init_v,theta,nowt);
loops++;
nowt = nowt + dt;
tempvy = Y_velocity(init_y,init_v,theta,nowt);
tempvx = X_velocity(init_x,init_v,theta,nowt);
if ( tempvy<0. && (stop = judge(tempx,tempy,tempvx,tempvy,basket_dis))>=1 )
break;
}
return stop;
}

//=====輸入球的速度及位置後,判斷進球與否=====
int judge(double tempx,double tempy,double tempvx,double tempvy,
double basket_dis) {
int stop=0; //stop=1 表示沒進,stop=2 表示空心球,stop=3 表示碰框進球
double x_ball_to_ring_a = tempx-(basket_dis-BASKET_RING);
double x_ball_to_ring_b = tempx-(basket_dis+BASKET_RING);
double y_ball_to_ring = tempy-BASKET_HIGH;
double bata=0.;

if (tempx <= basket_dis) {
if ( x_ball_to_ring_a*x_ball_to_ring_a+y_ball_to_ring*y_ball_to_ring
<= BALL_R*BALL_R ) {
if(x_ball_to_ring_a<=0) stop=1;
else {
if (fabs(tempvy)*(1.+REBOUND)/CONTACT_T <= fabs(GRAVITY))
stop=3;
else stop=1;}
} else {
if (y_ball_to_ring<=0) {
if(x_ball_to_ring_a<=0) stop=1;
else stop=2;}
}
} else {
if (x_ball_to_ring_b*x_ball_to_ring_b+y_ball_to_ring*y_ball_to_ring
<= BALL_R*BALL_R ) {
if(x_ball_to_ring_b>=0) stop=1;
}
}
}

```

```
        else {
            if (fabs(tempvy)*(1.+REBOUND)/CONTACT_T <= fabs(GRAVITY))
                stop=3;
            else stop=1;
        }
    }else {
        if (y_ball_to_ring<=0) {
            if(x_ball_to_ring_b>=0) stop=1;
            else stop=2;
        }
    }
    return stop;
}

//=====計算球的水平位置=====
double X_position(double x,double v,double theta,double dt){
    return (x+v*cos(theta)*dt);
}

//=====計算球的垂直位置=====
double Y_position(double y,double v,double theta,double dt){
    return (y+v*sin(theta)*dt+0.5*GRAVITY*dt*dt);
}

//=====計算球的水平速度=====
double X_velocity(double x,double v,double theta,double dt){
    return v*cos(theta);
}

//=====計算球的垂直速度=====
double Y_velocity(double y,double v,double theta,double dt){
    return (v*sin(theta)+GRAVITY*dt);
}
```

