How does the spin of the cue ball and the angle at which it hits the object ball affect the trajectory of the object ball?

Holly Jackson
Old Orchard School
7th Grade

SYNOPSYS CHAMPIONSHIP 2013 PROJECT ABSTRACT

THE ABSTRACT IS A REQUIRED PART OF YOUR PROJECT

Bring your abstract with you to project check-in on Tuesday, March 12, 2013.

PROJECT NUMBER: 071-880-2 STUDENT NAME(s): Holly Marie Jackson

You should bring at least ten copies of your abstract with you when you come to the Championship. One copy should remain on display with your project during the Championship. You will want others to give to the judges. Your abstract should be written after you finish your research and experimentation and should include:

- Your project title, the full name(s) of all team members, and your school (all centered).
- The purpose of your project
- Your hypothesis or evaluation criteria
- A brief statement about the procedures and equipment you used
- Your results (analysis of data)
- Your conclusions

Type or print neatly using 10- or 12- point black type. Single space throughout.
Center your project title, your name(s), and school.
You must use this form. Your abstract should be less than 500 words, and it should fit within the lines on this form.

Billiard Bots

Holly Marie Jackson, Old Orchard School

Purpose:
I wanted to find how the spin of a cue ball and its aiming angle (angle it travels at relative to straight) affect the trajectory of the object ball.

Procedures and Equipment:
In my experiment, I used a robot with the cue ball attached to it for repeatability. I then used this "bot" to hit the object ball. I hit the object ball at 7 different angles (0°, 3˚, 5˚, 7˚, 8˚, 9˚, and 10˚) and with 4 different spins (no spin, right English which is counterclockwise spin, left English which is clockwise spin, and rolling), repeating each test 5 times. To find the coordinates of the final resting position of the object ball, I used a tape measure. Then, I used inverse tangent ($\tan^{-1}$) to find the cut angle. Because the first tests only showed a small and often inconclusive angle change, I also did my experiment with a rubber band on the cue ball to see if the angles became more definite.

Hypothesis:
I had multiple hypotheses for all of the different scenarios (e.g. right English & 5˚ angle, etc.). I created diagrams that represented the ideal situation for the measurements when the cue ball was hit straight on with no spin and at angles with no spin. I created a table with all of my predictions for the other scenarios as well. My belief was that the friction between the balls would create specific changes in the object ball's direction.

Results:
I had two control tests with no spin. One was when I hit the object ball straight on and the other was when I hit it at angles. In both cases, the object ball's path was slightly left of the ideal modeled scenarios ranging from -1.66˚ to -3.73˚ straight on and from 6.6˚ to 45.1˚ with aiming angles. I preformed numerous tests to compare to these controls but there are too many to present all of them here. My most surprising results were with left English and right English. For the left English shots hit on angle, the throw (change in direction) of the object ball increased as the aiming angle increased ranging from about 10˚ at a 3˚ aiming angle to 29˚ at a 10˚ aiming angle. In the right English shots hit on angle, the throw did not seem to have a specific pattern. I predicted that for both English shots on angle, the throw would decrease as the aiming angle increased, which did not concur with my results.

Conclusions:
In my conclusions, I compared the effects of cut-induced throw, spin-induced throw, and total induced throw. I found that a left aiming angle produces left cut-induced throw, and the addition of right English increases the total amount of left throw. Left English produces right spin-induced throw that counteracts the left cut-induced throw. My full write-up contains an analysis of many other interesting effects as well, including changes in friction and the effect of rolling.

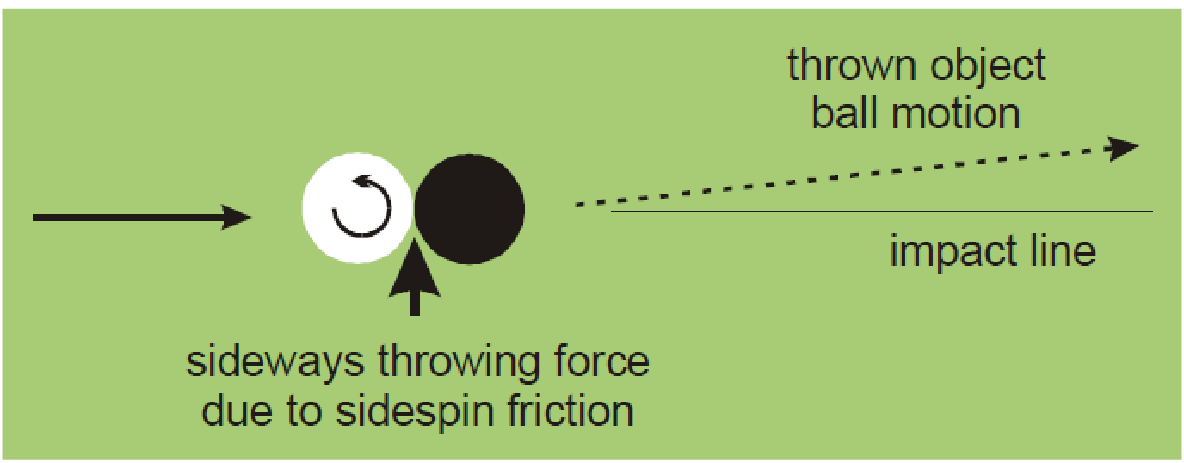
Question:

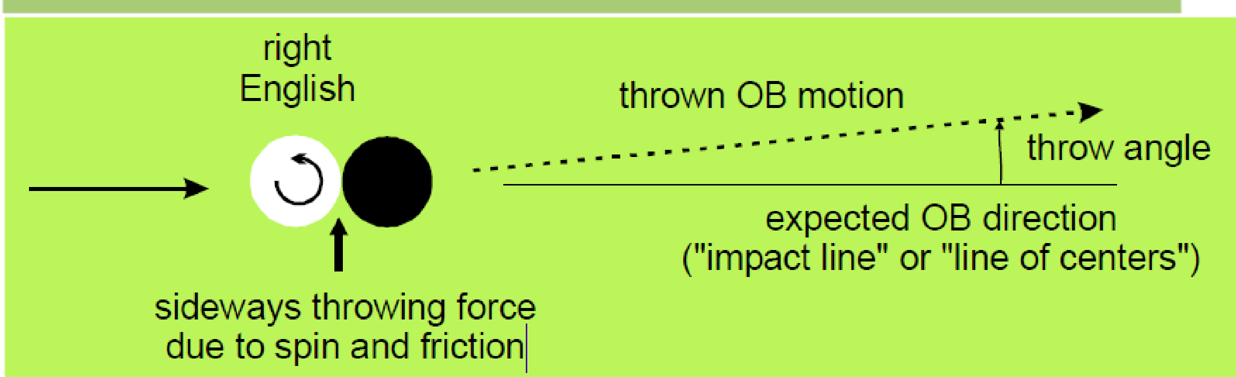
How does the spin of the cue ball and the aiming angle affect the trajectory of the object ball?

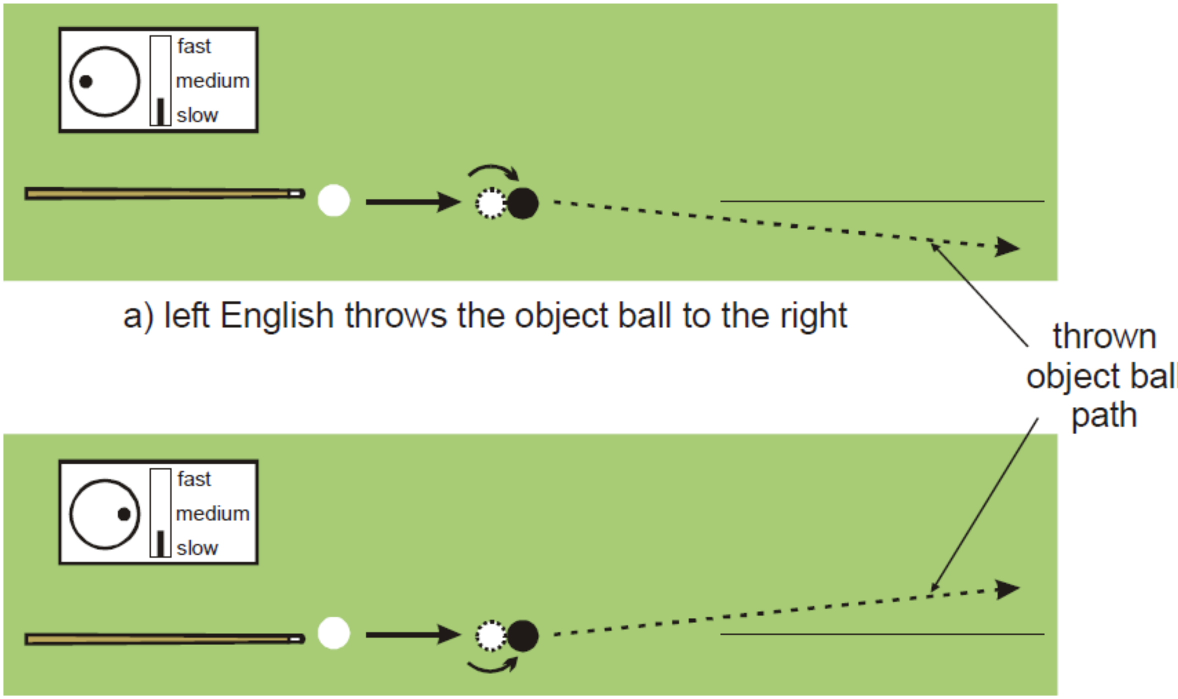
Background Research:

Throw is the change in the object ball's direction due to friction between the cue ball and the object ball during impact.

When you hit the cue ball with a right or left English, you inflict spin-induced throw on the object ball. This means that even though you are hitting the object ball in the center, the spin of the cue ball gives the object ball a throw angle. Shots using spin-induced throw are called right and left Englishes. A left English is when you hit the cue ball on the left side which causes the ball to spin clockwise. This causes the cue ball to spin to the left while continuing to go straight. When it hits the object ball, the object ball will move to the right. A right English is the exact opposite. It inflicts counterclockwise spin on the cue ball, causing the object ball to go to the left. The diagrams below show the effects of spin-induced throw.



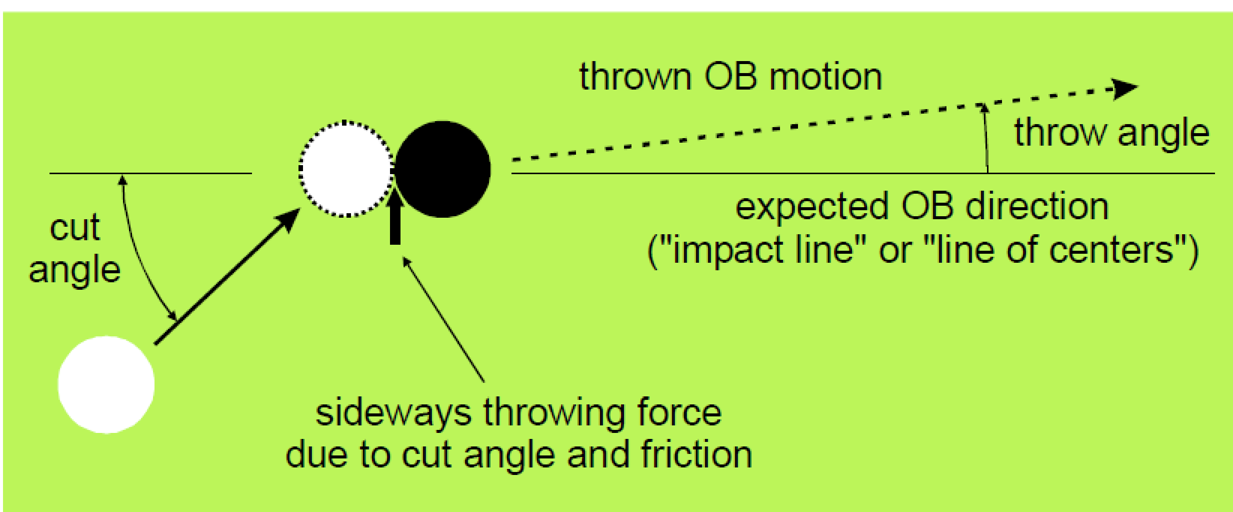




a) left English throws the object ball to the right

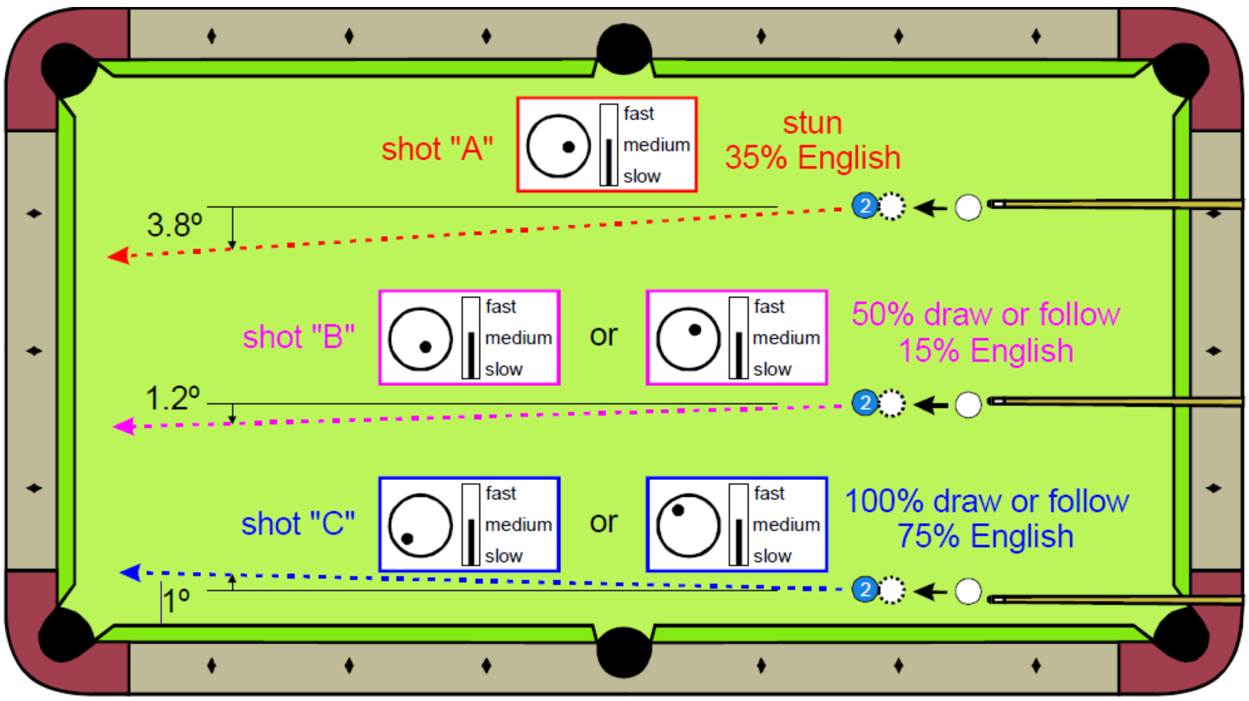
b) right English throws the object ball to the left

Cut-induced throw is when you hit the cue ball in the center. Though, the cue ball hits the object ball on the side, causing it to go in another direction. In summary, you hit the cue ball so it travels at an angle and it hits the

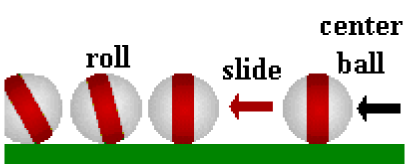


Another thing that affects the throw of the pool ball is cling. Cling is friction caused by chalk, dirt, or roughness on the balls.

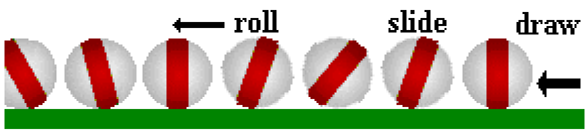
The higher the percentage of English there is, the farther to the side (left or right) the cue ball is hit.



A **center ball (or stun)** is when you hit the cue ball in the center. The ball slides for a while and then rolls. If you hit the ball hard, then there will be a long period of sliding. If you hit the ball lightly, then there will be a short period of sliding.

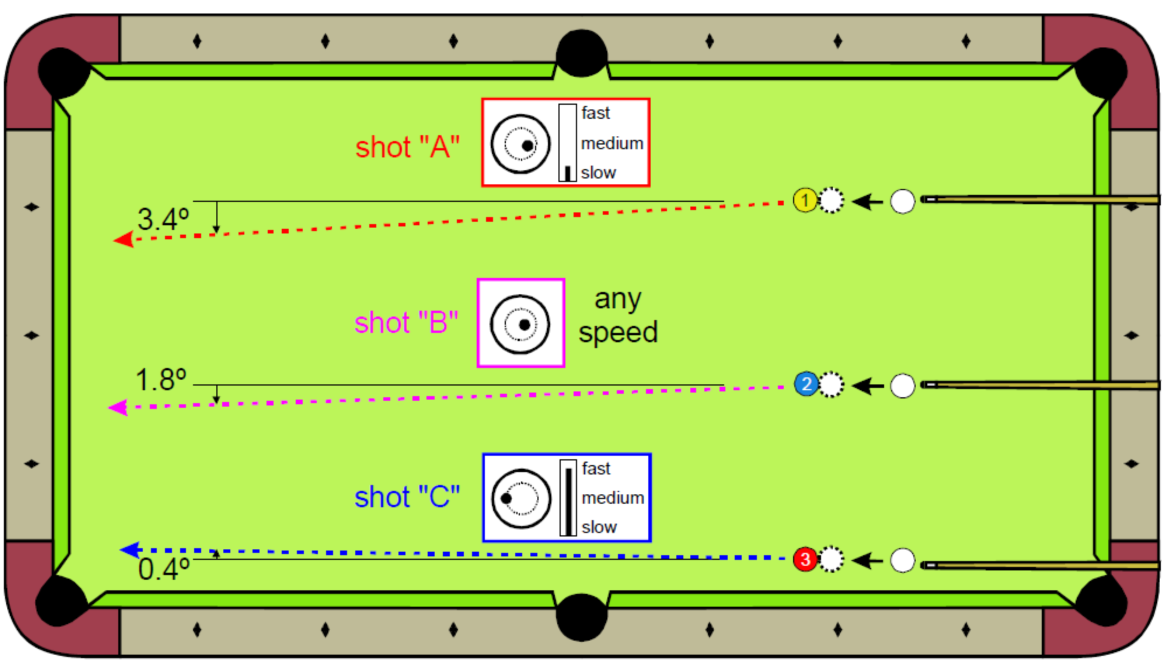


A **draw** is when you hit the cue ball lower than the center. The ball spins backwards and then slides. After, it rolls forwards. If you hit the ball hard, then there will be a long period of spinning backwards. If you hit the ball lightly, then there will be a short period of spinning backwards. If the cue ball hits another ball when it is still in a backwards spin, it will come back to you.

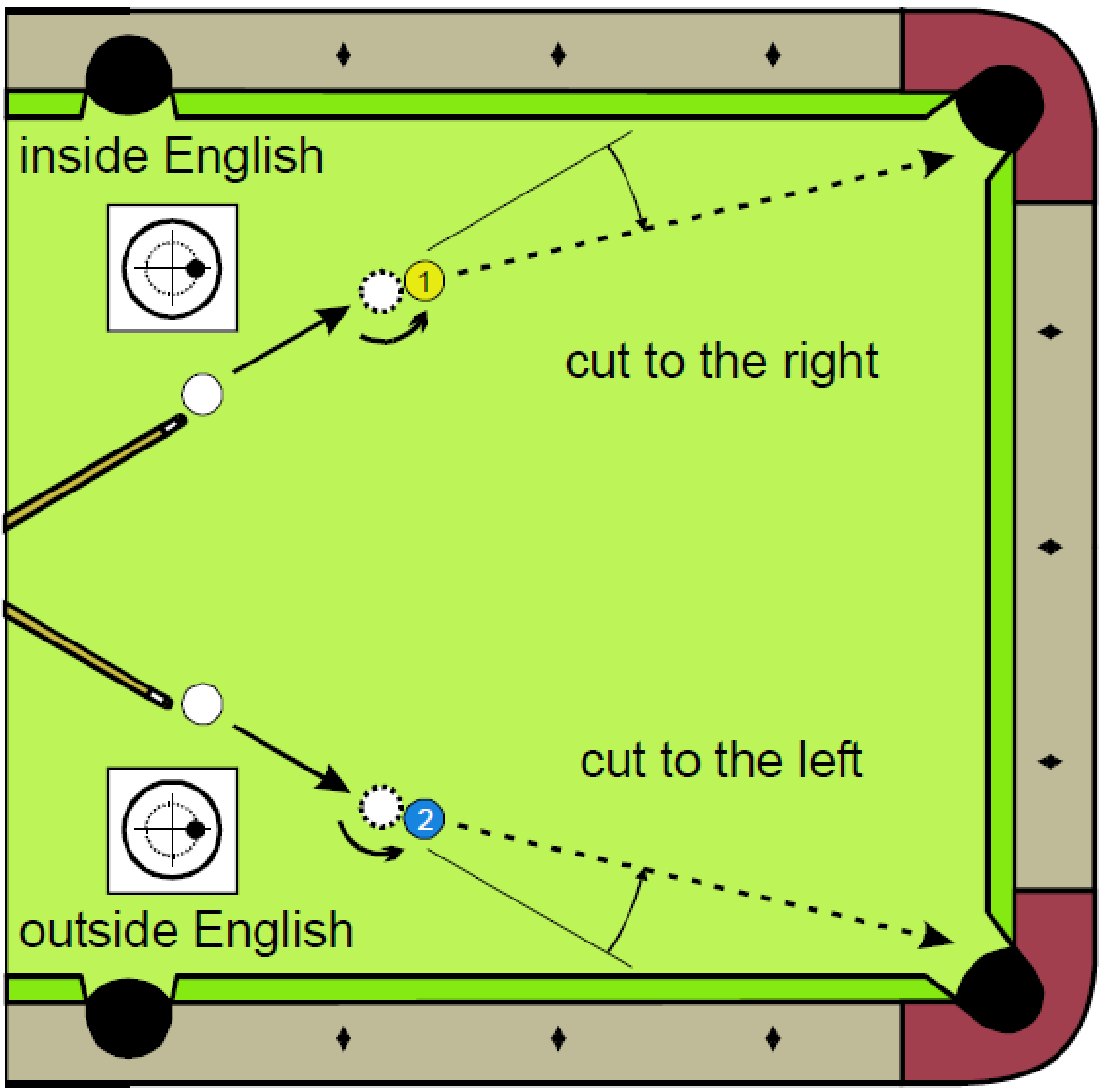


A **follow** is when you hit the cue ball higher than the center. The ball rotates forward. If it hits another ball, it will keep rolling forward. Though, this will happen in a center ball and a draw if the cue ball starts rolling before it hits the other ball.

Shot "A" is a slow shot with about 50% right English. Shot "B" is a 25% right English shot. Shot "C" is a 100% (maximum) left English shot. You can tell this from the position of the cue stick.



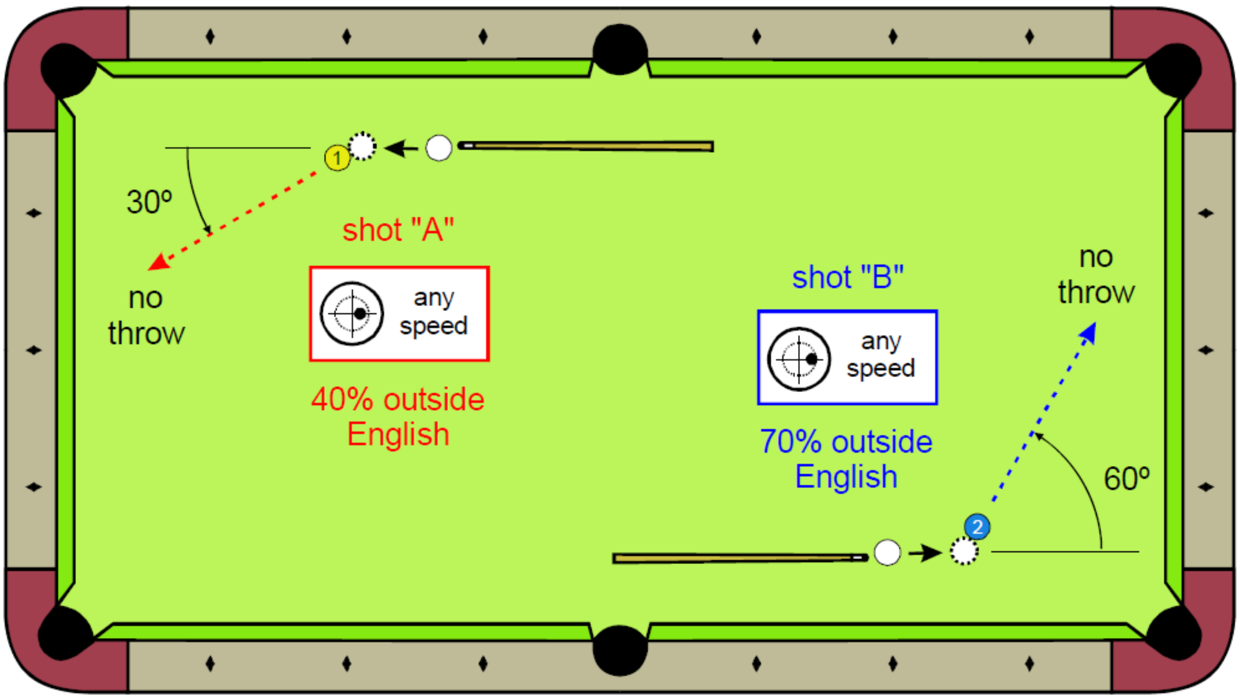
Sometimes a hit has both cut-induced and spin-induced throw. This means that the cue ball is being hit with a right or left English, but hits the object ball on the side. When this happens in a right English, it is called an inside English. When this happens in a left English, it is called an outside English.



At the inside English in the diagram, the ball is being hit on the right side (right English). Though, the ball is hitting "the cut side, or 'in' side," of the object ball. On the other one, a right

English is being used, but the cue ball is hitting the "out" side of the ball, or the side away from the cut side. So, it is an outside English.

If an outside or inside English has a higher percent, the cue ball is being hit farther to the side, like in the normal Englishes. But, the cue ball hits the side of the object ball, so it goes off at an extreme angle.



There is usually less throw with an inside English.

In my experiment, I will measure the effect of both cut-induced, spin-induced throw, and a mix.

Sources:

Alciatore, David G. The Illustrated Principles of Pool and Billiards. New York, NY: Sterling Pub., 2004.

Alciatore, David G. "Throw – Part I: Introduction." Billiards and Pool Principles, Techniques, Resources. Aug. 2006. 30 Jan. 2013 <http://billiards.colostate.edu/bd_articles/2006/aug06.pdf>.

Alciatore, David G. "Throw – Part IV: Spin-induced throw." Billiards and Pool Principles, Techniques, Resources. Nov. 2006. 30 Jan. 2013 <http://billiards.colostate.edu/bd_articles/2006/nov06.pdf>.

Alciatore, David G. "Throw – Part V: SIT speed effects." Billiards and Pool Principles, Techniques, Resources. Dec. 2006. 30 Jan. 2013 <http://billiards.colostate.edu/bd_articles/2006/dec06.pdf>.

Alciatore, David G. "Throw – Part VI: Inside/outside English." Billiards and Pool Principles, Techniques, Resources. Jan. 2007. 30 Jan. 2013 <http://billiards.colostate.edu/bd_articles/2007/jan07.pdf>.

Loy, Jim. "The Physics of Billiards." The Physics of Billiards. 2002. 21 Feb. 2013 <http://www.jimloy.com/billiard/phys.htm>.

Materials:

1. Adept Technology® Cobra® Robot
2. Pool Table
3. 2 Cue Balls
4. Object Ball
5. Computer with Adept ACE® Program
6. 2 Screws
7. Tape Measure
8. Pencil or Pen

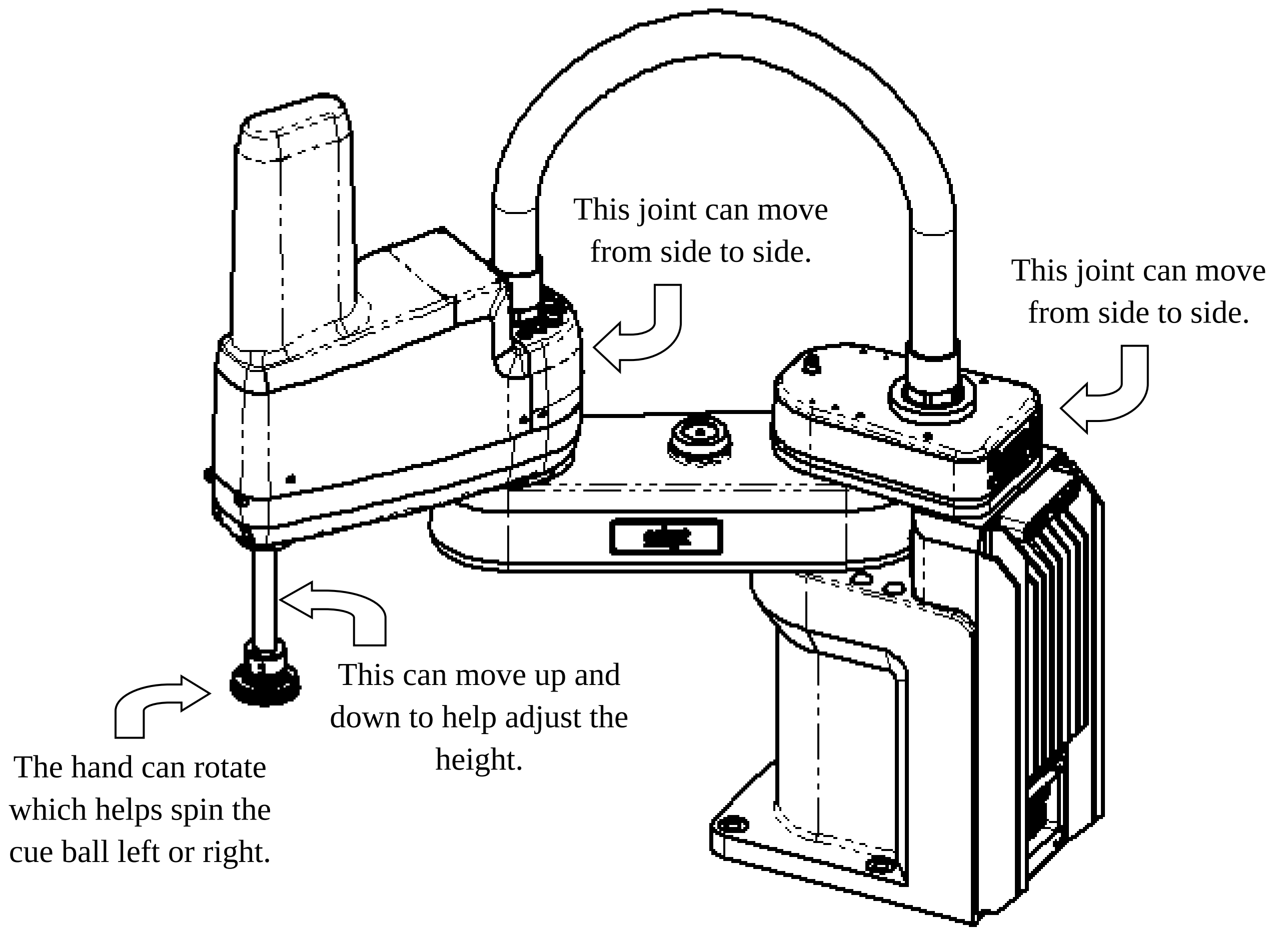
Adept ACE

Adept Technology® Cobra® Robot

Adept Technology® Cobra® Robot i800:

The Cobra has a built-in amplifier and motion controller.

It has 4 joints.

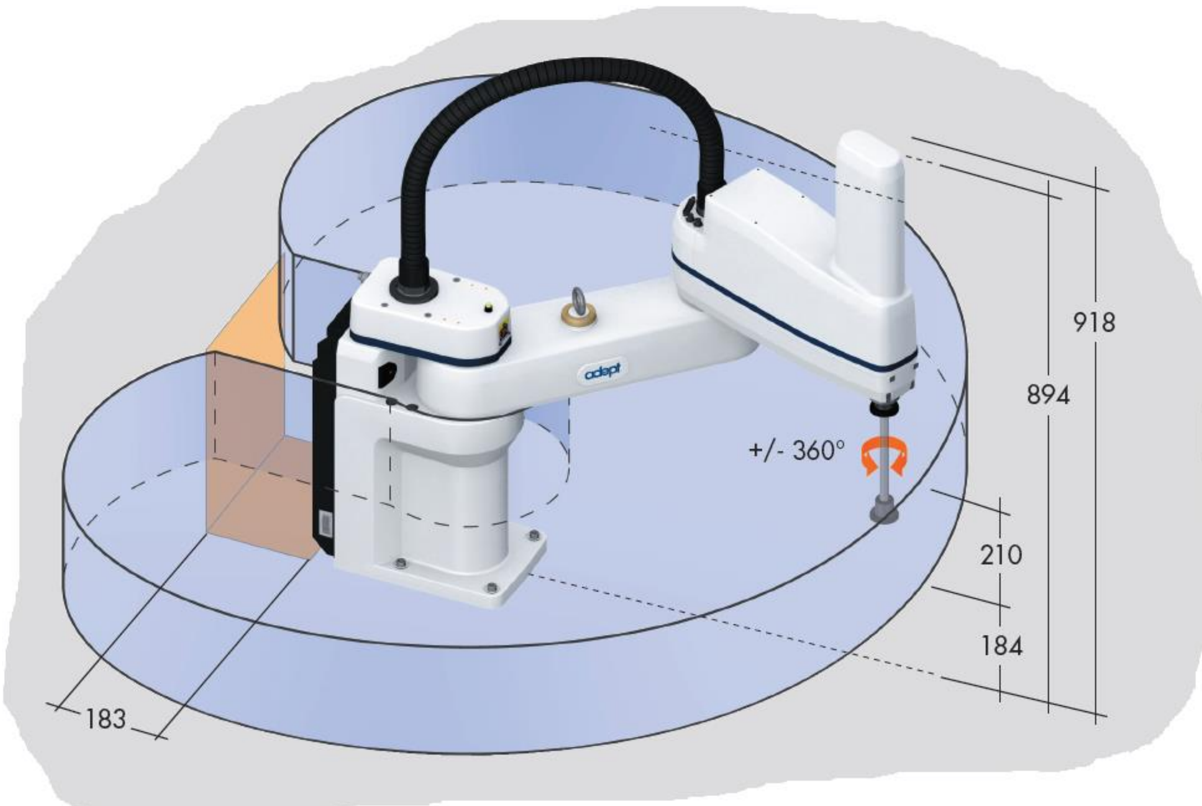


There are not many cables on the Cobra robot.

The Cobra has high accuracy, superior slow-speed following, and is easy to calibrate.

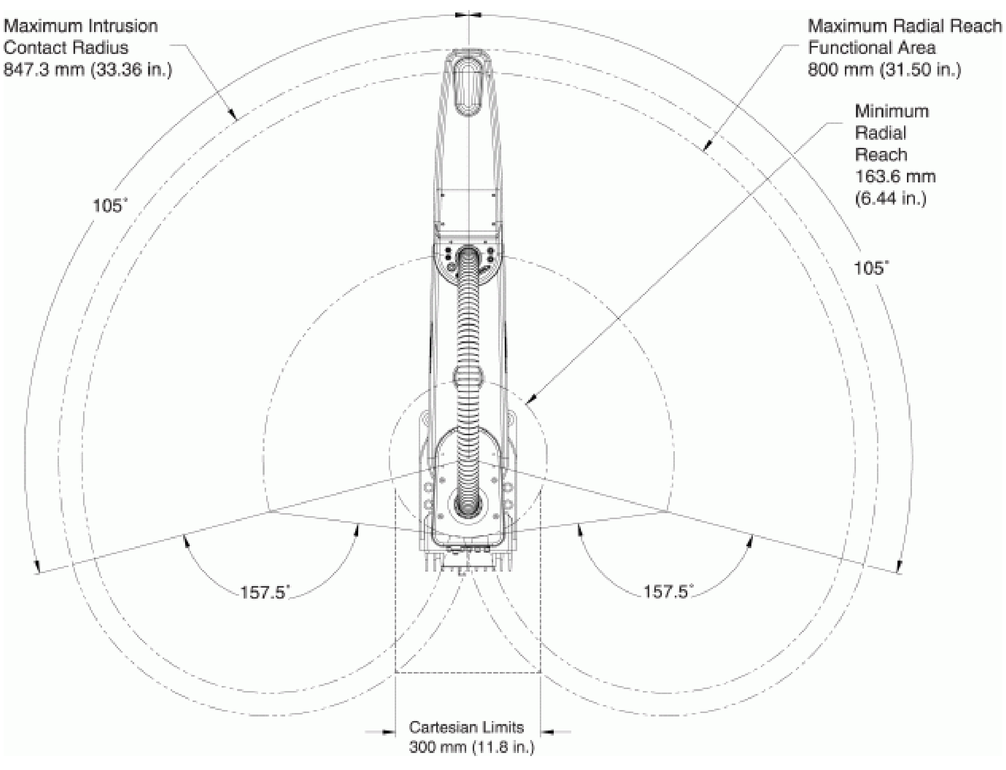
Its lightweight arm lets it deliver the maximum acceleration it can give.

It also can be troubleshoot very fast.



The Cobra had and 800 mm work envelope, or reach.

It can change up to 210 mm in height.



The tests a Cobra can perform are very repeatable.

On average, the robot is only 0.017 mm off when moving side to side

On average, the robot is only 0.003 mm off when moving up and down.

On average, the robot is only 0.019 degrees off when moving at angles.

Procedure:

1. Collect all of the materials.
2. Connect the cue ball to the robots hand for the control test.
3. Program and set up the robot to move the cue ball along the desired path. (See Programming Sheet)
4. Start the program so the robot can calibrate.
5. Place the reference triangle where the cue ball stops.
6. Continue the program.
7. Place the object ball in the reference triangle and carefully remove it.
8. Continue the program.
9. Have the robot move the cue ball forward 13 inches into the object ball which is 12 inches.
10. Find the final resting position of the object ball.
11. Record the x and y coordinates of the object ball positions on your data table.
12. Reset both balls.
13. Repeat steps 4-12 but move the cue ball on an angled path to the object ball (see figure). These angles are 3, 5, 7, 8, 9, and 10 degrees.
14. Connect the cue ball to the robot's hand and program the robot to have it spin clockwise while traveling for the right English spin-induced throw test.
15. Repeat steps 5-13 under these circumstances.
16. Connect the cue ball to the robot's hand and program the robot to have it spin counterclockwise while traveling for the right spin English spin-induced throw test.
17. Repeat steps 5-13 under these circumstances.

18. Program and set up the robot with two screws pushing the cue ball so it rolls for the rolling test.
19. Repeat steps 5-13 under these circumstances.
20. Compute the angles at which the object ball moved off the straight course from for each test on Microsoft Excel.
21. Average the angles for each test and compare.

Simplified Procedure:

Step 1

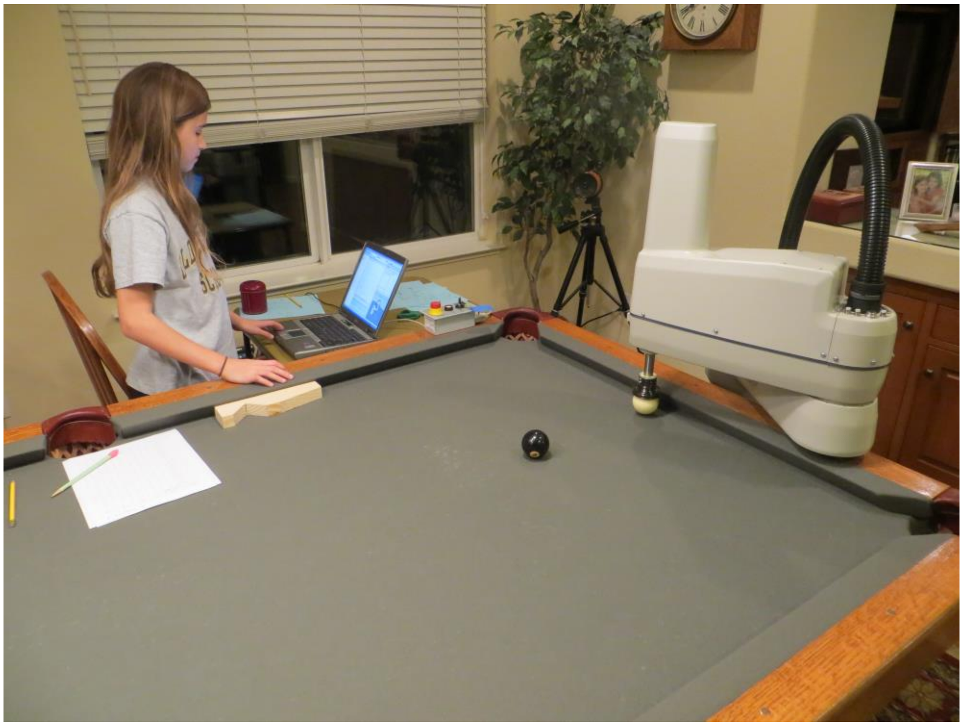
Start the program so the robot can calibrate. When it stops, push the reference triangle up against the cue ball attached to it.

Step 2

Press continue on the program. The cue ball should move out of the reference triangle. Place the object ball in the reference triangle being careful not to move it.

Step 3

Carefully remove the reference triangle making sure that the object ball does not move.



Step 4

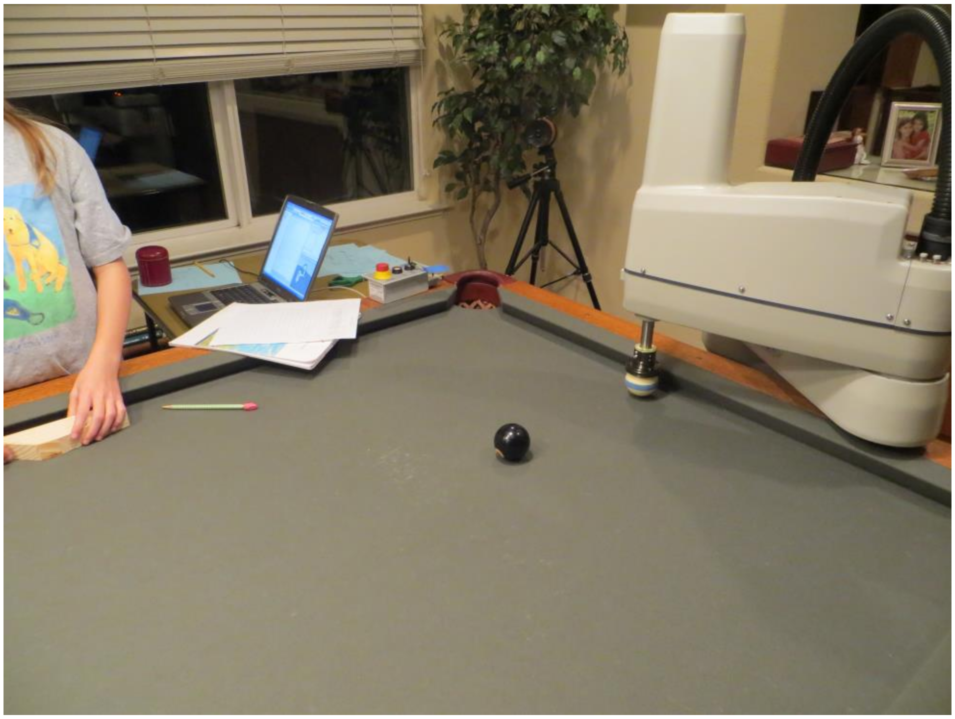
Continue the program and watch the cue ball hit the object ball.

Step 5

Wait for the object ball to come to a halt. Then, measure its x and y coordinates with a tape measure.

Step 6

Repeat the test with right and left spin and at different angles. Also, repeat the test with a rubber band.



Key

(green) = comment

(blue) = command

```
.PROGRAM cobra_pool()
;
        SPEED 10, 10 ALWAYS
        ACCEL 300, 300
        DISABLE SCALE.ACCEL[1]
        DISABLE SCALE.ACCEL.ROT[1]


        distance = 350          ; Distance to move when striking the ball
(350mm = 13.8")
        angle = 10              ; Angle to move at when striking the ball
(0 = straight, Neg = right, Pos = Left)
        spin = 375            ;left  ; Amount of spin during strike (0 =
no spin, Neg = CCW, Pos = CW)
;        spin = -375           ;right
;        spin = 0

        hspeed = 22
        rspeed = 1000

        startx = 350
        targetx = startx+304.8
        zvalue = 330
        startr = -100

        SET start_loc.t = TRANS(startx,0,zvalue,0,180,startr)
        SET target_loc.t = TRANS(targetx,0,zvalue,0,180,startr)
        SET mid_loc.t =
TRANS(startx+distance/2*COS(angle),distance/2*SIN(angle),zvalue,0,180,sta
rtr)
        SET end_loc.t =
TRANS(startx+distance*COS(angle),distance*SIN(angle),zvalue,0,180,startr)

        SET #reset_loc.j = #PPOINT(56.892,-128.574,62,0)

        ; Set slow speed and move to each location as a test.
        ; Record the joint angle precision points for each location
        SPEED 10, rspeed ALWAYS
        MOVE #reset_loc.j
        BREAK
        MOVES start_loc.t
        BREAK
        HERE #start_loc.j
        MOVES end_loc.t
        BREAK
        HERE #end_loc.j
        MOVES target_loc.t
        BREAK
        HERE #target_loc.j
        MOVES mid_loc.t
        BREAK
        HERE #mid_loc.j

        DECOMPOSE jts[] = #end_loc.j
```

```
        SET #end_loc_mod.j = #PPOINT(jts[0],jts[1],jts[2],jts[3]+spin)

        ; Set slow speed and move to start position then straight to
target position.
        MOVE #start_loc.j
        BREAK
        MOVES #target_loc.j
        BREAK
        ; Pause for operator to align reference traingle and press
continue
        PAUSE

        ; Move back to start location
        MOVES #start_loc.j
        BREAK
        ; Pause for operator to press continue
        PAUSE

        ; Clear the timer, set speed, and move to end location to strike
the object ball
        TIMER 1 = 0
        SPEED hspeed, rspeed ALWAYS
        MOVES #mid_loc.j
        MOVES #end_loc_mod.j
    ;    MOVES end_loc.t
        BREAK
        TYPE "Elapsed Time =", TIMER(1), " seconds"

        ; Wait 1.6 seconds then move back to start location
        FOR i = 1 TO 100
            WAIT
        END
        SPEED 10, rspeed ALWAYS
        MOVES #start_loc.j
        BREAK

.END
```

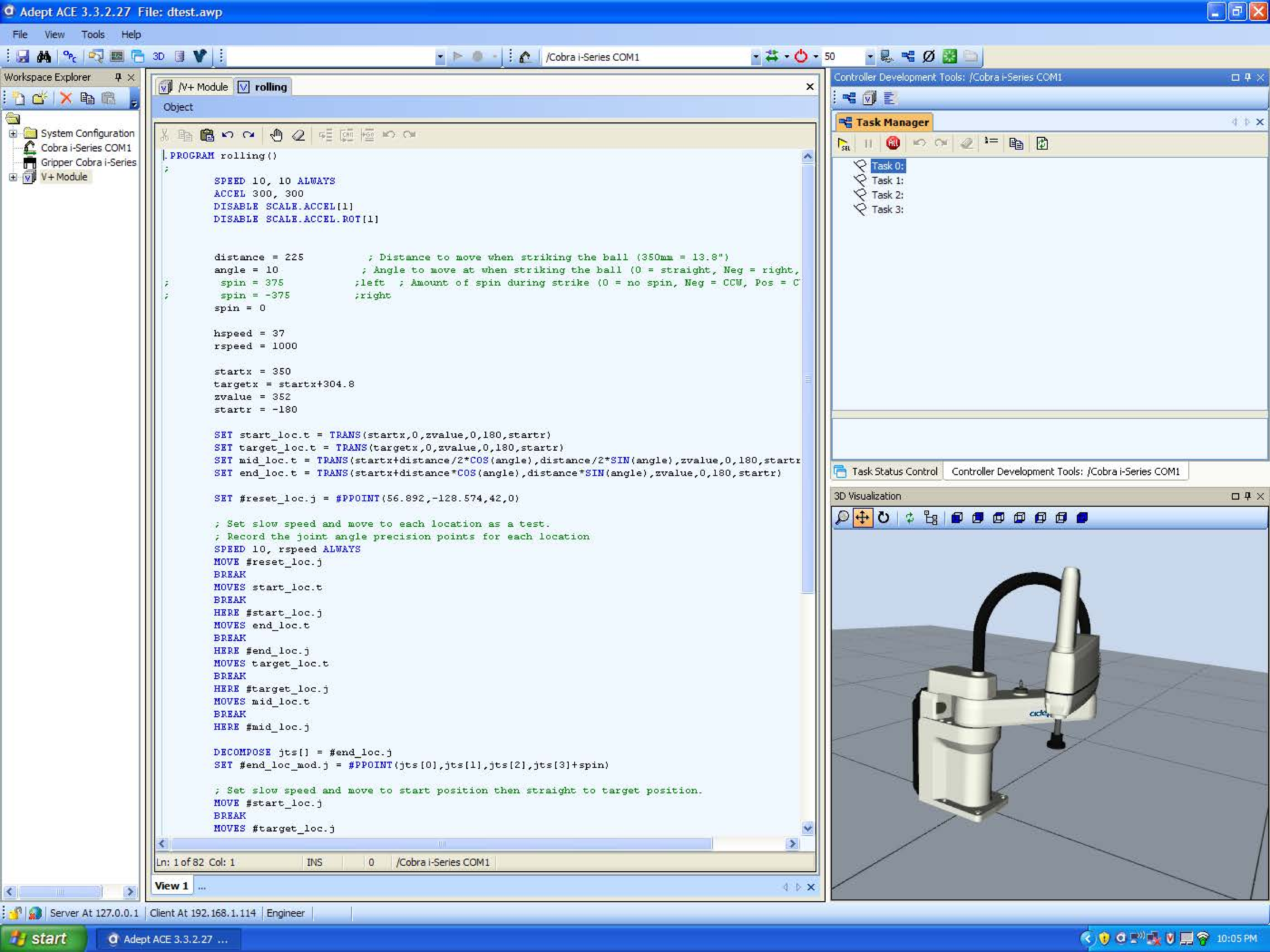
Key

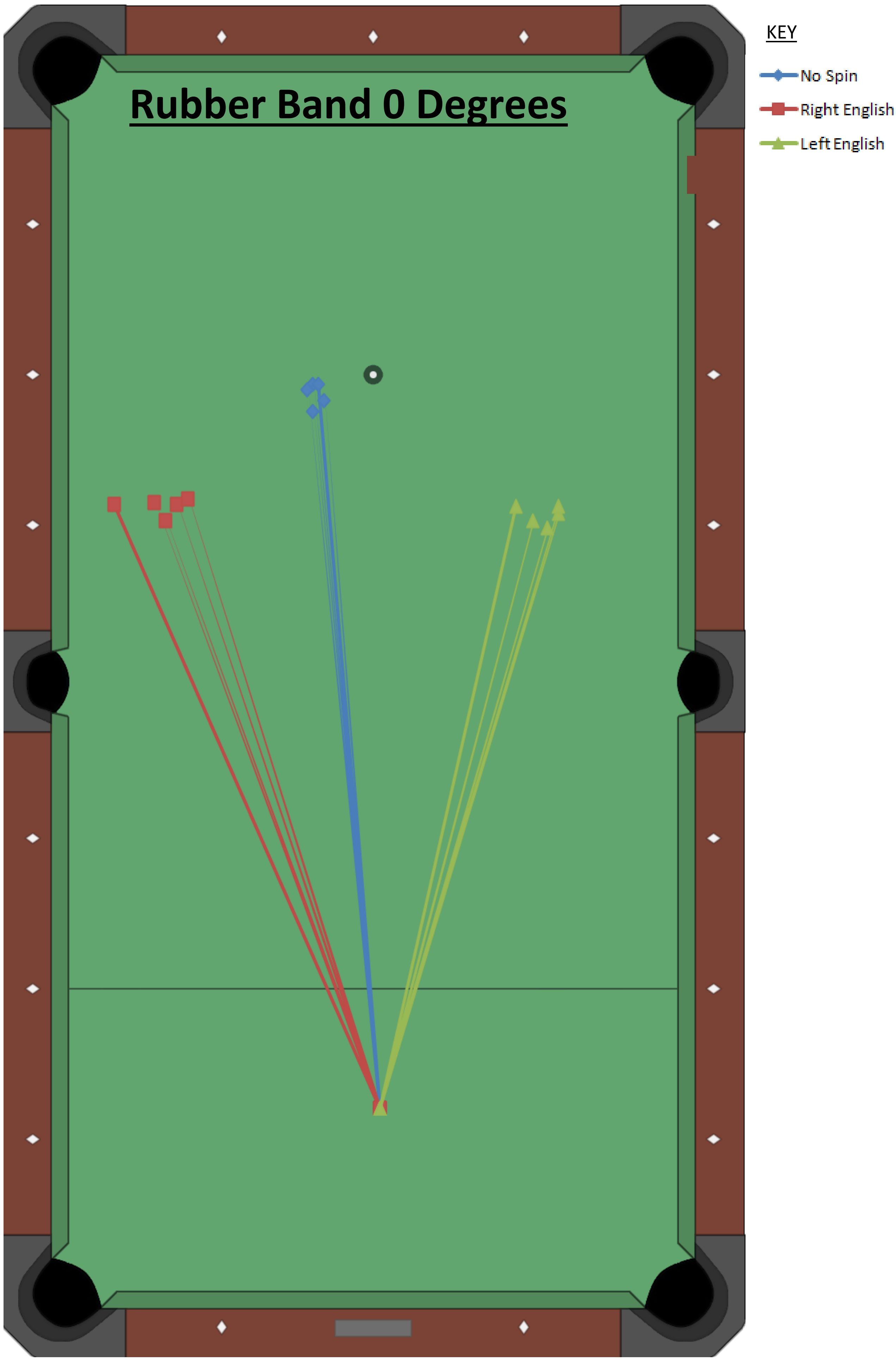
(green) = comment

(blue) = command

```
.PROGRAM rolling()
;
        SPEED 10, 10 ALWAYS
        ACCEL 300, 300
        DISABLE SCALE.ACCEL[1]
        DISABLE SCALE.ACCEL.ROT[1]


        distance = 225          ; Distance to move when striking the ball
(350mm = 13.8")
        angle = 10              ; Angle to move at when striking the ball
(0 = straight, Neg = right, Pos = Left)
;        spin = 375           ;left  ; Amount of spin during strike (0 =
no spin, Neg = CCW, Pos = CW)
;        spin = -375          ;right
        spin = 0

        hspeed = 37
        rspeed = 1000

        startx = 350
        targetx = startx+304.8
        zvalue = 352
        startr = -180

        SET start_loc.t = TRANS(startx,0,zvalue,0,180,startr)
        SET target_loc.t = TRANS(targetx,0,zvalue,0,180,startr)
        SET mid_loc.t =
TRANS(startx+distance/2*COS(angle),distance/2*SIN(angle),zvalue,0,180,sta
rtr)
        SET end_loc.t =
TRANS(startx+distance*COS(angle),distance*SIN(angle),zvalue,0,180,startr)

        SET #reset_loc.j = #PPOINT(56.892,-128.574,42,0)

        ; Set slow speed and move to each location as a test.
        ; Record the joint angle precision points for each location
        SPEED 10, rspeed ALWAYS
        MOVE #reset_loc.j
        BREAK
        MOVES start_loc.t
        BREAK
        HERE #start_loc.j
        MOVES end_loc.t
        BREAK
        HERE #end_loc.j
        MOVES target_loc.t
        BREAK
        HERE #target_loc.j
        MOVES mid_loc.t
        BREAK
        HERE #mid_loc.j

        DECOMPOSE jts[] = #end_loc.j
```

```
        SET #end_loc_mod.j = #PPOINT(jts[0],jts[1],jts[2],jts[3]+spin)

        ; Set slow speed and move to start position then straight to
target position.
        MOVE #start_loc.j
        BREAK
        MOVES #target_loc.j
        BREAK
        ; Pause for operator to align reference traingle and press
continue
        PAUSE

        ; Move back to start location
        MOVES #start_loc.j
        BREAK
        ; Pause for operator to press continue
        PAUSE

        ; Clear the timer, set speed, and move to end location to strike
the object ball
        TIMER 1 = 0
        SPEED hspeed, rspeed ALWAYS
        MOVES #mid_loc.j
        MOVES #end_loc_mod.j
    ;    MOVES end_loc.t
        BREAK
        TYPE "Elapsed Time =", TIMER(1), " seconds"

        ; Wait 1.6 seconds then move back to start location
        FOR i = 1 TO 100
            WAIT
        END
        SPEED 10, rspeed ALWAYS
        MOVES #start_loc.j
        BREAK

.END
```


Adept ACE 3.3.2.27 File: dtest.awp
File View Tools Help
/Cobra i-Series COM1
50
Workspace Explorer
System Configuration
Cobra i-Series COM1
Gripper Cobra i-Series
V+ Module
/V+ Module
rolling
Object
.PROGRAM rolling()
;
SPEED 10, 10 ALWAYS
ACCEL 300, 300
DISABLE SCALE.ACCEL[1]
DISABLE SCALE.ACCEL.ROT[1]
distance = 225 ; Distance to move when striking the ball (350mm = 13.8")
angle = 10 ; Angle to move at when striking the ball (0 = straight, Neg = right,
; spin = 375 ;left ; Amount of spin during strike (0 = no spin, Neg = CCW, Pos = C
; spin = -375 ;right
spin = 0
hspeed = 37
rspeed = 1000
startx = 350
targetx = startx+304.8
zvalue = 352
startr = -180
SET start_loc.t = TRANS(startx,0,zvalue,0,180,startr)
SET target_loc.t = TRANS(targetx,0,zvalue,0,180,startr)
SET mid_loc.t = TRANS(startx+distance/2*COS(angle),distance/2*SIN(angle),zvalue,0,180,startr
SET end_loc.t = TRANS(startx+distance*COS(angle),distance*SIN(angle),zvalue,0,180,startr)
SET #reset_loc.j = #PPOINT(56.892,-128.574,42,0)
; Set slow speed and move to each location as a test.
; Record the joint angle precision points for each location
SPEED 10, rspeed ALWAYS
MOVE #reset_loc.j
BREAK
MOVES start_loc.t
BREAK
HERE #start_loc.j
MOVES end_loc.t
BREAK
HERE #end_loc.j
MOVES target_loc.t
BREAK
HERE #target_loc.j
MOVES mid_loc.t
BREAK
HERE #mid_loc.j
DECOMPOSE jts[] = #end_loc.j
SET #end_loc_mod.j = #PPOINT(jts[0],jts[1],jts[2],jts[3]+spin)
; Set slow speed and move to start position then straight to target position.
MOVE #start_loc.j
BREAK
MOVES #target_loc.j
Ln: 1 of 82 Col: 1
INS
0
/Cobra i-Series COM1
View 1
Controller Development Tools: /Cobra i-Series COM1
Task Manager
Task 0:
Task 1:
Task 2:
Task 3:
Task Status Control
Controller Development Tools: /Cobra i-Series COM1
3D Visualization
Server At 127.0.0.1
Client At 192.168.1.114
Engineer
start
Adept ACE 3.3.2.27 ...
10:05 PM


Rubber Band 0 Degrees
KEY
No Spin
Right English
Left English

Error Analysis:

In my experiment, I took five different measurements for each of the cut shots (0˚, 3˚, 5˚, 7˚, 8˚, 9˚, and 10˚) with the three different types of spin (right, left, and none). I compared the average angles of the three shot types to determine if the spin shots travelled at a different angle than the no spin shots. However, because my results varied, I had to be careful in drawing a conclusion. While the average measurement is the most likely angle, the correct angle could be anywhere in between the maximum and minimum of my five samples. I drew error bars on each graph to show the range of my results. If the top of the error bar of one spin measurement (one that is smaller) is greater than the bottom of the error bar of another spin measurement (one that is larger), then you can't conclude with certainty which has a greater angle. In other words, when the error bars overlap, the comparison of two angles is inconclusive.

Type of Hit	Result	Conclusion
No Spin − Straight (Control)	The results from the rubber band and no rubber band tests range from -3.73˚ to -1.66˚ which is slightly off straight.	My hypothesis was not correct. I suspect that my pool table might not be perfectly level so it affected my results causing the object ball to drift to the left side of the table.
No Spin − Angles (Control)	When no rubber band was used, the results were in most cases within a few degrees of my hypothesis. When a rubber band was used with larger aiming angles, the cut angle became significantly smaller than hypothesized.	With no rubber band, my hypothesis was close to being correct especially if you consider the drift added from the pool table not being level. With no rubber band, there is not much throw induced on the object ball. That is why it does not move much off of its predicted path. With a rubber band, my hypothesis was incorrect. At the higher aiming angles, the cut angles were much smaller than hypothesized. This may be because at the larger angles, there is throw on the object ball without a rubber band, and less with one causing the rubber band tests to look smaller.
Rolling − Straight	The results from the rolling test range from -4.22˚ to -1.9˚ which is slightly off straight.	My hypothesis was not correct. However the results are nearly identical to the no spin – straight control test. So again my assumption that my pool table might not be level is the cause.
Rolling − Angles	The cut angle for rolling was consistently larger than the no spin – angles control test by 1.8˚ to 5.8˚.	My hypothesis was correct. The rolling of the cue ball caused the object ball to throw to the right from the path of the no spin – angles control test.
Left English− Straight	In the no rubber band test, the object ball went nominally -1.3˚ to the left. In the rubber band test, the ball went almost 10˚ to the right.	My hypothesis was half correct. The object ball's final location was in one test to the left and in the other to the right. I suspect that my possibly tilted pool table caused the object ball to sway towards the left in the no rubber band test which offset the throw to the right.

Left English− Angles	The object ball moved at a range from 13.5˚ (at a 3˚ aiming angle) to 57.6˚ (at a 10˚ aiming angle) to the right. Also, the results were very different from the no spin – angles control test results, showing that the left English had an effect on the trajectory of the object ball.	My hypothesis was half correct. I hypothesized correctly that the ball would be thrown to the right. Though, my hypothesis that there would be less throw as the aiming angle increased was incorrect. There was actually on average more throw as the aiming angle increased.
Right English− Straight	The object ball went to the left.	My hypothesis was correct. The object ball travelled to the left of the pool table. Even if you take into account the possible bias of the pool table and subtract it, the object ball still was thrown to the left.
Right English − Angles	The object ball did not always go to the left of the no spin – angles control test for the tests without a rubber band. Sometimes the results overlapped. There was too much variation in the results for it to be conclusive. With the rubber band, all of the results conclusively went to the left.	My hypothesis was correct for seven out of the twelve right English – angles tests. All of the rubber band tests proved to be conclusive and sent the object ball to the left of the no spin – angles results. But, only one no rubber band tests proved my hypothesis correct. For the rest, there was too much variation, so I could not conclude that the object ball was sent to the left of the no spin – angles test.