

AI-AUGMENTED ENGINEERING DYNAMICS: A CONVERSATIONAL APPROACH

Taner Onsay, Ph.D.
Author
August 29, 2026 · Revision 1.0

© 2026 AI First Engineering

Observe



Idealize



Model



Solve / Analyze



Validate



Report

The engineering workflow you will use throughout this book

by Taner Onsay

CHAPTER - 2

PART 3 · ≈ 80 MIN

Projectile Motion

The four balls return. Predict where they land, measure the real flight, track it on video, and find out how good the ideal model really is.

3

Observe

Idealize

Model

Solve / Analyze

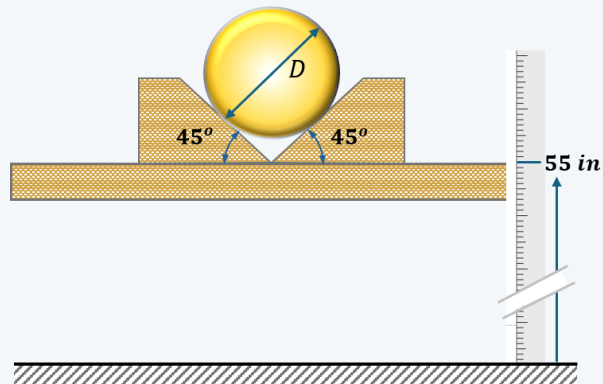
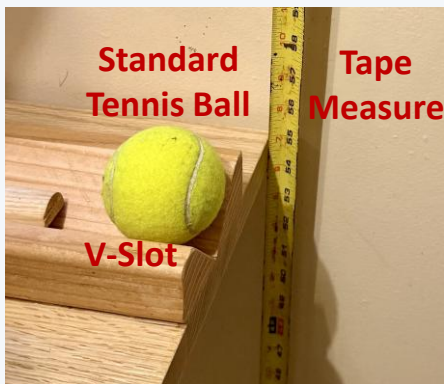
Validate

Report

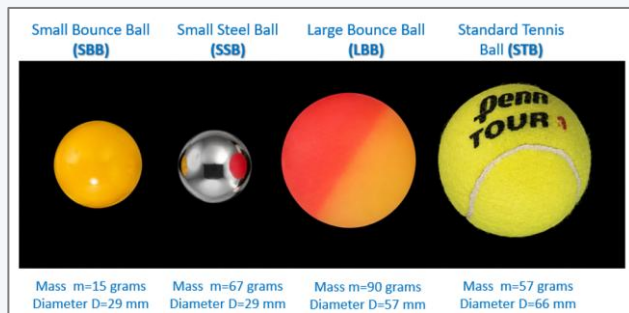


Same four balls as Chapter 1. New question: what happens when they leave the table with a horizontal velocity?

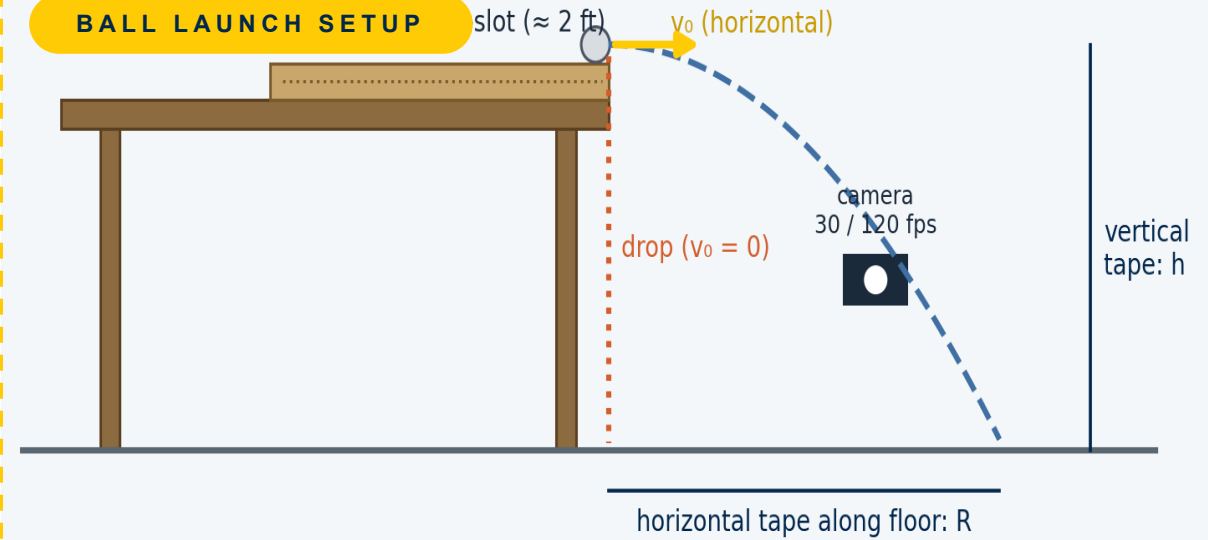
V-SLOT PROJECTILE APPARATUS



FOUR-BALL DIFFERENT BALLS



BALL LAUNCH SETUP



≈2 ft horizontal runway with 45°+45° V-slot on the table; slot exit at the table edge; vertical tape beside the table; horizontal tape on the floor along the flight direction; the four Chapter 1 balls.

Observe

Idealize

Model

Solve / Analyze

Validate

Report



Vertical drop · low-speed launch · higher-speed launch

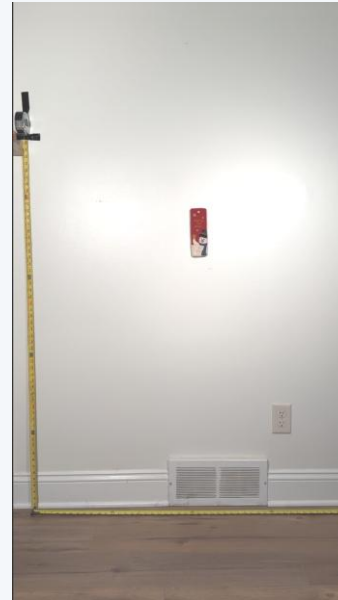
Vertical Drop

Condition A — ball released from runway-exit height, $v_0 \approx 0$.



Low Horizontal Speed

Condition B — rolled along the V-slot at a lower speed.



Higher Horizontal Speed

Condition C — rolled along the V-slot at a higher speed.



What is the same in all three?

Same ball, same exit height, same floor. Only the horizontal launch differs.



What would you need to measure to compare them fairly?

Time in the air. Where each lands. And — for the launches — how fast the ball left the runway.

Observe

Idealize

Model

Solve / Analyze

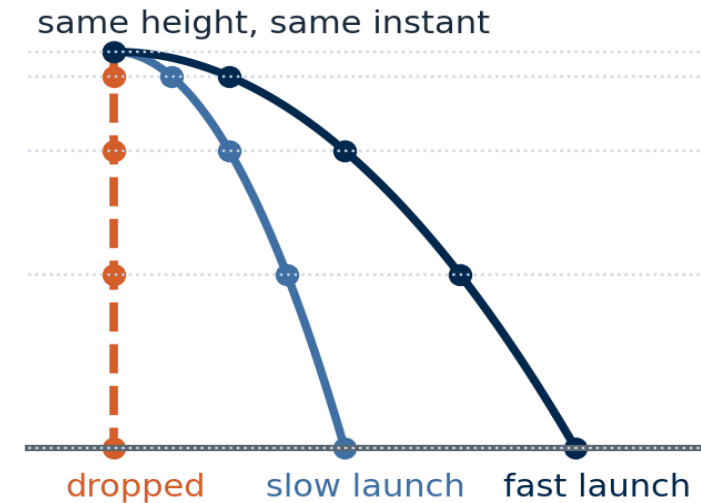
Validate

Report



PREDICT

Three identical balls leave the same height at the same instant: one dropped, two launched horizontally at different speeds. Which reaches the floor first?



Commit before the slow-motion reveal.

Vote: dropped · slow launch · fast launch · all together. Give a one-line reason.

Observe

Idealize

Model

Solve / Analyze

Validate

Report

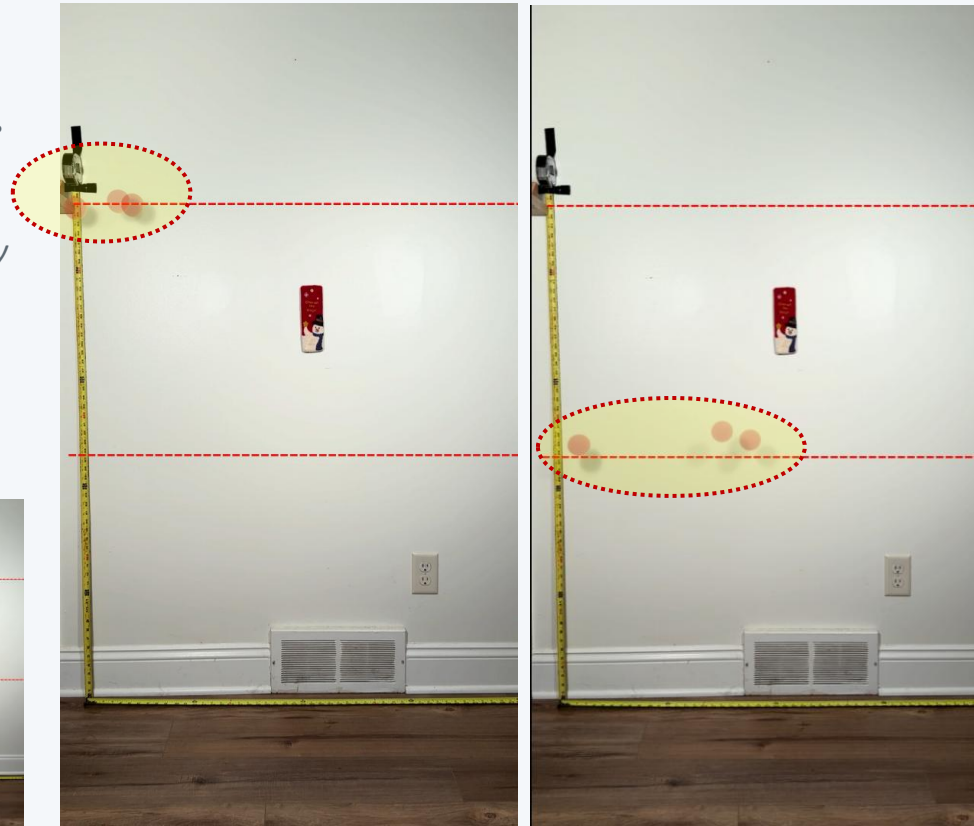


Watch the three balls' heights frame by frame

MEDIA PLACEHOLDER

Three clips overlaid at the release frame; horizontal guidelines across all three at two frames shows that vertical positions are approximately equal in this test and they **hit the ground almost at the same time**. The increments of frame (motion resolution) between the 3 clips made it difficult to align them at the release.

(VIDEO)



The observation is yours to make

Frame by frame, are the three balls at the same height? Say what you see — including any small difference.



Why it happens

The only acceleration is vertical and (nearly) constant. Vertical motion "does not know" about horizontal motion.



If they do not match exactly

That is data, not failure. Release timing, frame selection, and air are all candidates. We will rank them in VALIDATE.

Observe

Idealize

Model

Solve / Analyze

Validate

Report



Particle, uniform gravity, no drag, horizontal release at the table edge

THE MODEL

Particle	the ball's mass center (rolling and spin ignored after release)
Coordinates	fixed x (along the flight), y (up); $a_y = -g$
Origin	the runway exit, at height h above the floor — stated, not assumed
Assumptions	uniform gravitational field; constant $g = 9.81 \text{ m/s}^2$; negligible aerodynamic drag; non-rotating local frame; release velocity exactly horizontal, at the table edge
Knowns	h (vertical tape), g; for launches, the release frame
Unknowns	flight time t_f ; landing distance R; release speed v_{0x}



Model, not law

These equations are true for the model. Whether the model is good enough for these balls over this height is exactly what the experiment will tell us.

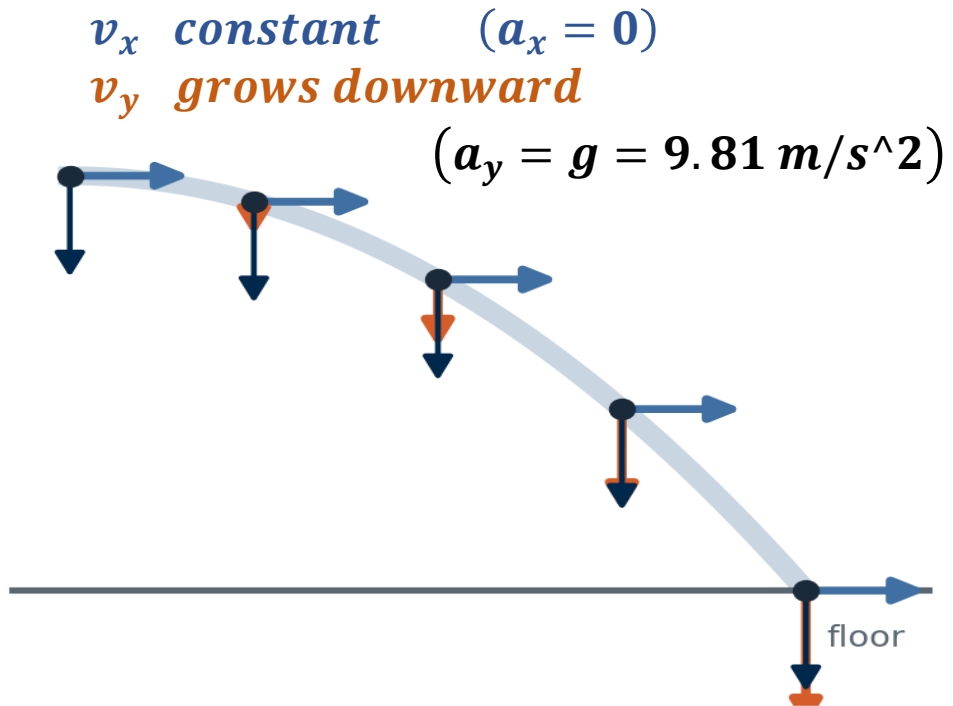


Which assumption would you doubt first?

Rank them now, before the data. Write your ranking down — you will compare it with the measured disagreement.



Horizontal: constant velocity. Vertical: constant acceleration. Nothing new — just applied to each component.



Why can we treat x and y separately? Because the basis vectors are fixed (Part 2) and the only acceleration is along y .

HORIZONTAL LAUNCH FROM HEIGHT h

$$a_x = 0: \quad v_x = v_{0x} = \text{constant} \rightarrow x = v_{0x} t$$

$$a_y = -g: \quad v_y = -g t \rightarrow v_y = \frac{dy}{dt} \rightarrow$$

$$y - h = \int_0^t v_y dt = -\frac{1}{2} g t^2$$

@ Landing ($t = t_f$, $y = 0$ and $x = R$: Range):

$$t_f = \sqrt{(2h/g)}, \quad R = v_{0x} t_f$$

t_f is the Chapter 1 drop time. The horizontal speed sets only how FAR the ball goes, never how LONG it is in the air.



Mass and diameter do not appear. Launch speed changes the range, not the time.

1

Same flight time for every ball

$t_f = \sqrt{2h/g}$ contains neither mass nor diameter. Ideal prediction: all four balls, all three conditions, land at the same time.

2

Range grows with launch speed

$R = v_{0x} t_f$. Double the launch speed, double the range — with the same flight time.

3

Launch speed can be inferred

Measure R and t_f ; then $v_{0x} \approx R / t_f$. A tape measure and a frame count are enough — no sensor on the ball.

ILLUSTRATIVE NUMBERS — REPLACE WITH THE MEASURED h

$$h = 0.75 \text{ m} \rightarrow t_f = \sqrt{(2 \cdot 0.75 / 9.81)} = 0.39$$

A ball leaving at 1.5 m/s lands at $R \approx 0.59 \text{ m}$; at 3.0 m/s, $R \approx 1.17 \text{ m}$. Same 0.39 s in the air.



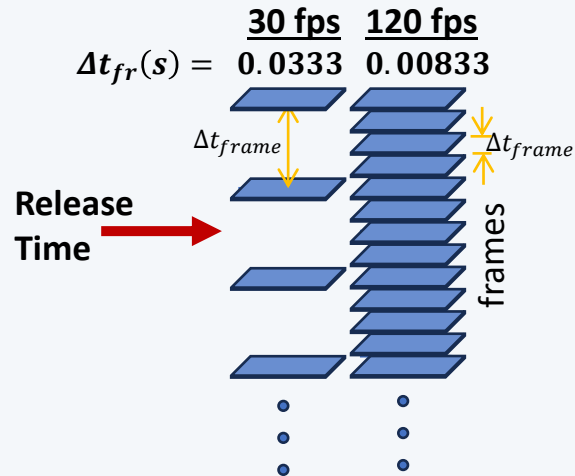
Predict before the data

Will the four real balls give exactly the same flight time? If not — is the theory wrong, or is something in the model list on the previous slide not quite true for one of them? Write your guess.



Flight time from video: frame rate sets the measurement resolution

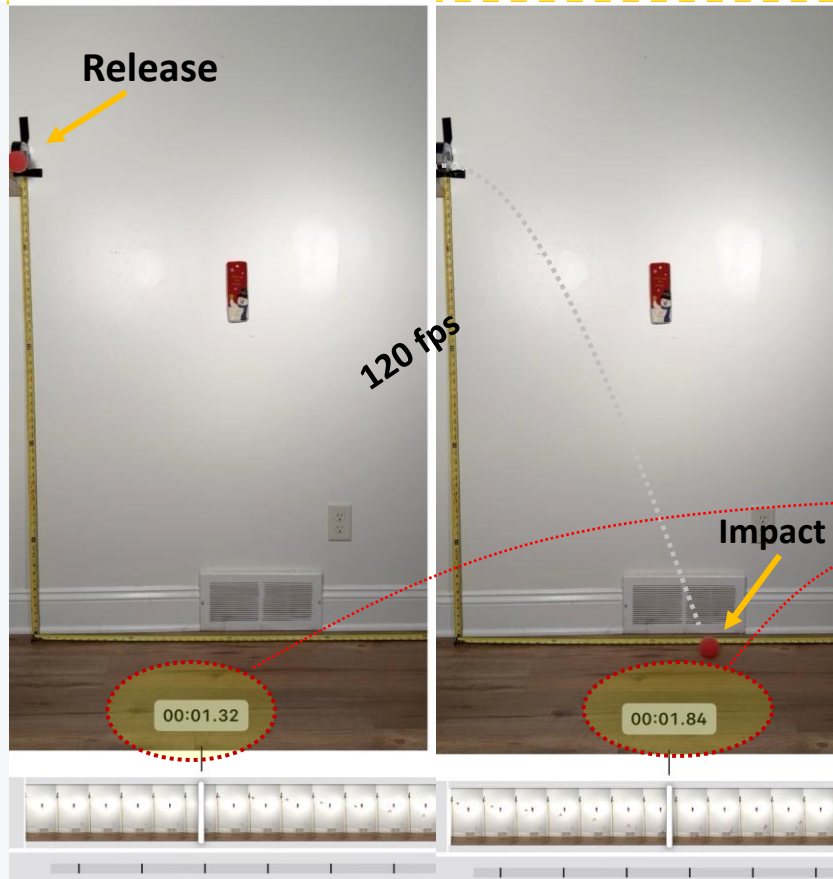
VIDEO TIME RESOLUTION



Δt_{fr} time between frames

$1/30 = 0.0333 \text{ s/frame}$

$1/120 = 0.00833 \text{ s/frame}$



Step 1 — find the release time

First frame in which the ball has clearly left the runway. Same ambiguity: the true release occurs somewhere between frames.



Step 2 — find the impact time

First frame in which the ball touches the floor. The exact instant of impact may occur between frames.

$$\Delta t = t_{\text{impact}} - t_{\text{release}}$$

$$= 00:01.84 - 00:01.32 = 0.52 \text{ s}$$



Step 3 — Estimate measurement uncertainty

% relative uncertainty

$$30 \text{ fps: } \pm 2 \text{ frames} \rightarrow \pm 0.067 \text{ s} \rightarrow \frac{0.067}{0.52} = 0.13 = 13\%$$

$$120 \text{ fps: } \pm 2 \text{ frames} \rightarrow \pm 0.017 \text{ s} \rightarrow \frac{0.017}{0.52} = 0.33 = 3.3\%$$

Higher frame rate does not change the physics. It improves our ability to measure it by reducing measurement uncertainty.



h from the vertical tape, R from the floor tape, t from the frame count →
 $v_{0x} = R / t$

MEDIA PLACEHOLDER

[INSERT FOUR-BALL COMPARISON TABLE]

Rows: four balls (same as Chapter 1: sizes/masses as recorded). Columns per condition A/B/C: t (120 fps), R, $v_{0x} = R/t$, $g_{\text{eff}} = 2h/t^2$, repeat-trial spread.



What you measure

h once (tape). R per trial (floor tape, landing mark or first-contact frame). t per trial (frame count).



What you compute

$v_{0x} = R/t$ for the launches. $g_{\text{eff}} = 2h/t^2$ for every trial — the drop trials give the cleanest g.



What you look for

Do the four balls agree? Do A, B, C give the same t? Is the repeat-trial spread bigger than the ball-to-ball difference?

Observe



Idealize



Model



Solve / Analyze



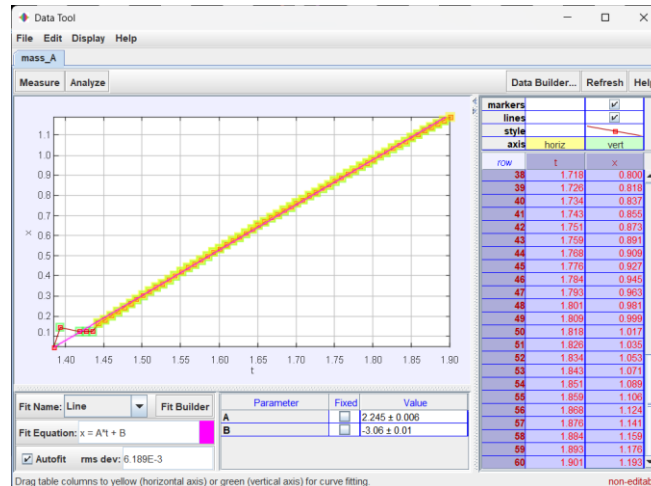
Validate



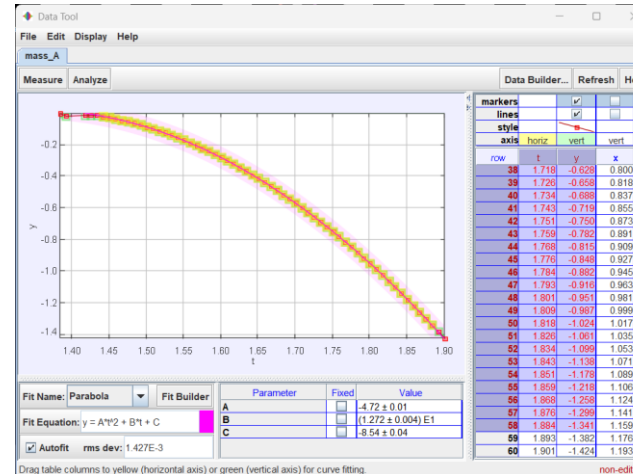
Report



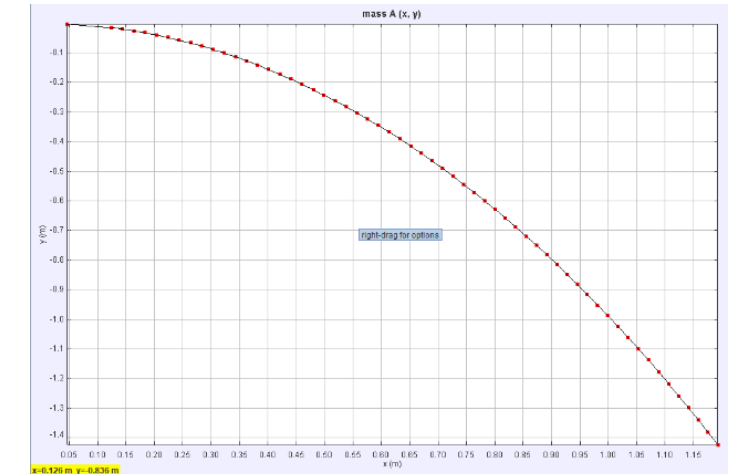
Track $x(t)$ and $y(t)$. Fit x linearly, y quadratically. Read v_{0x} , v_{0y} and an effective g from the fits.

SBB - TRACKER $x(t)$ 

Linear fit; slope = v_{0x} .

SBB - TRACKER $y(t)$ 

Quadratic fit; coefficients give v_{0y} and g_{eff}

SBB - TRACKER $y(t)$ 

Measured path with parabolic $y(x)$

WHAT THE FITS MEAN

$$x(t) \approx v_{0x} t \quad y(t) \approx h + v_{0y} t - \frac{1}{2} g_{eff} t^2$$

$v_{0y} \approx 0$ tests "horizontal release". $g_{eff} \approx 9.81$ tests "constant g , no drag". Curvature in $x(t)$ would test "no drag" a second way.



Not a black box

Tracker differentiates by finite differences and fits by least squares. Before trusting it: did the scale come from the tape in the plane of the flight? Is the origin where you said it was?



A prediction is a hypothesis until the floor weighs in



MODEL PREDICTS

$$t_f = \sqrt{(2h/g)}$$

same for every ball and launch speed; $R = v_{0x} t_f$; $g = 9.81 \text{ m/s}^2$

vs



EXPERIMENT MEASURES

$$t = ? \quad v_{0x} = ? \quad g_{\text{eff}} = ?$$

frame count · tape measure · Tracker fits — four balls, three conditions, two frame rates

MEDIA PLACEHOLDER

[INSERT IDEAL vs MEASURED TRAJECTORY]

Measured x-y points with the ideal parabola overlaid.

If they disagree — and they will, a little — who is wrong: the model, the measurement, the frame count... or the assumption you ranked last?

Observe



Idealize



Model



Solve / Analyze



Validate



Report



Sources of disagreement, and how each one would show up in the data



Frame selection (± 1 frame each end)

Fingerprint: scatter between repeat trials of about one frame; shrinks $\times 4$ at 120 fps.



Release consistency

Fingerprint: R varies between "identical" trials while t does not.



Runway alignment / non-horizontal exit

Fingerprint: Tracker's $v_{0y} \neq 0$; t differs between drop and launch.



Camera parallax and scale plane

Fingerprint: g_{eff} systematically off for all trials by the same factor.



Rolling / spin and launch variability

Fingerprint: light or small balls behave differently from heavy ones at launch.



Aerodynamic effects

Fingerprint: the lightest / largest ball lands short; $x(t)$ bends slightly; grows with speed.

The engineering question is not "was there error?" — there always is. It is: which errors are large enough to change the conclusion, and which measurement would settle it?



AI AS MODELING PARTNER

"My measured flight time differs from the ideal result by 3%. Give me possible causes, but do not decide which is correct. Tell me what additional measurement would distinguish them." — then YOU decide, with the data.

Observe



Idealize



Model



Solve / Analyze

**Validate**

Report



DISCUSS



The four balls did not give exactly identical results.

Does that mean the theory is wrong?

Talk to a neighbor for a minute. Separate three things: the model's assumptions, the measurement's resolution, and the ball-to-ball differences. Which one explains what you saw?

Observe



Idealize



Model



Solve / Analyze



Validate



Report



Same model, one more decomposition: $v_{0x} = v_0 \cos \theta$, $v_{0y} = v_0 \sin \theta$

LAUNCH SPEED v_0 AT ANGLE θ ABOVE HORIZONTAL

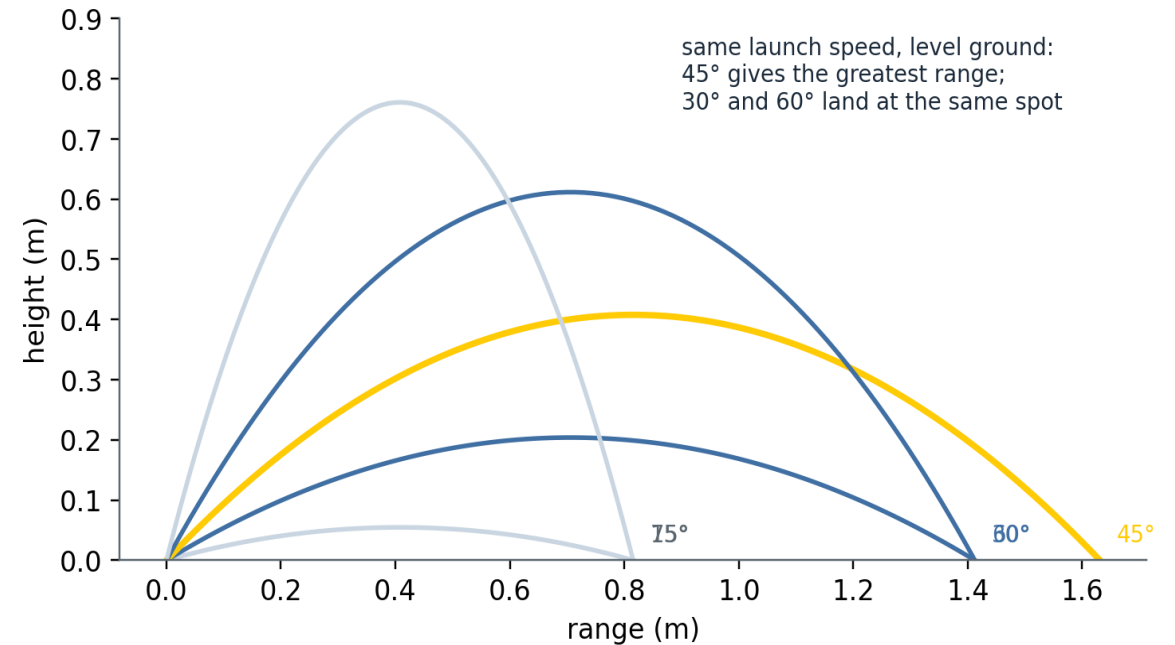
$$x = (v_0 \cos \theta) t$$

$$y = y_0 + (v_0 \sin \theta) t - \frac{1}{2} g t^2$$

$$v_y = v_0 \sin \theta - g t \quad (v_x = v_0 \cos \theta, \text{ always})$$

$$v_y^2 = (v_0 \sin \theta)^2 - 2g(y - y_0)$$

Only two of the three vertical equations are independent — the third is the first two with t eliminated. At the apex $v_y = 0$, not $v = 0$.

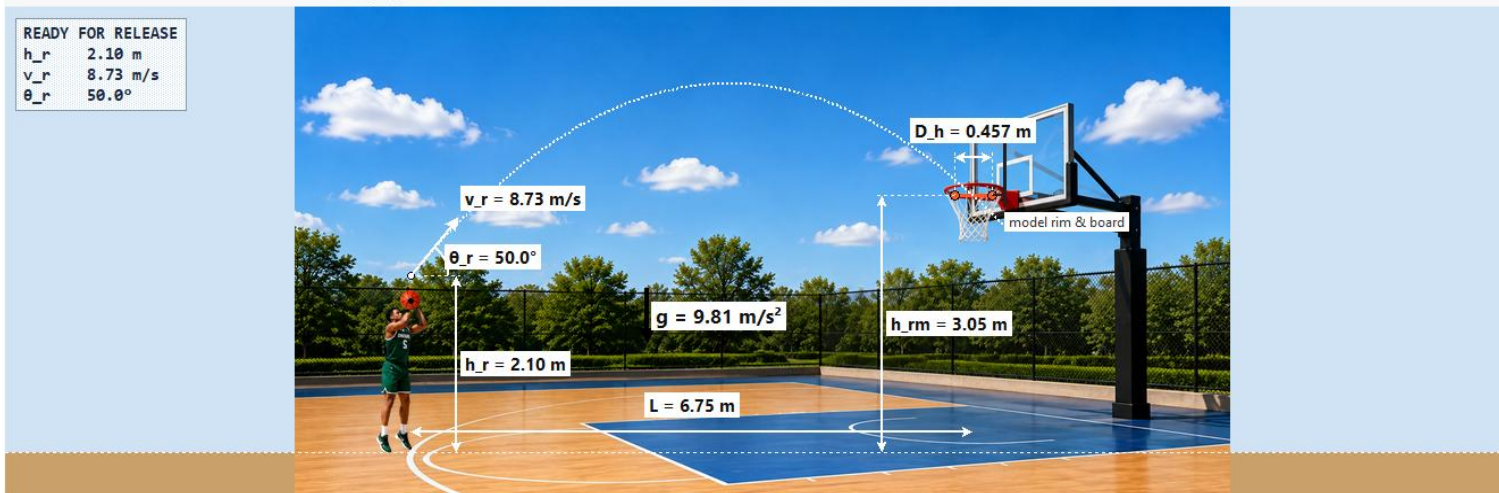


Shown, not derived: on level ground 45° gives the greatest range, and ϑ and $90^\circ - \vartheta$ land at the same spot. Try it in the notebook with a parameter sweep on ϑ .



The angled shot, live: define the release, predict, press PLAY

Shot — SIDE / TRAJECTORY view (the physics view)



The physics view: every dimension of this Part on one picture — h_r , v_r , ϑ_r , L , g , and the rim at $h_{rm} = 3.05$ m.



1 · Define the shot

v_r , θ_r , h_r , L — sliders or typed values; presets included.



2 · Predict, then PLAY

Commit — short, make, or long. PREDICTION MODE hides the graphs.



3 · Motion + graphs together

Ball, $x(t)$, $y(t)$, v , a side by side; slow motion; 0.05 s steps.

The simulator runs exactly the equations of the previous slide — $a_x = 0$, $a_y = -g$, applied to x and y . If you can predict it by hand, you can check the machine.



Same model, same fine print

Ideal planar projectile after release: no drag, no spin, no lateral error, no rim/backboard rebound (Version 1). The MODEL-card assumptions are the assumptions in the code.

Observe



Idealize



Model



Solve / Analyze



Validate



Report



$v_r = 8.73 \text{ m/s}$ at 50.0° , released at 2.10 m , 6.75 m from the hoop. Short, make, or long?

2. PREDICT, THEN PLAY

APPLY / GENERATE SHOT

▶ PLAY

■ PAUSE

↺ RESET

Shot generated — ready to PLAY

Time: 0.05 s 0.05 s ▶

Playback speed: 0.5× (slow motion)

VIEW: ☒ SIDE / TRAJECTORY ☐ TOP / RIM

GRAPH DISPLAY: ☐ None ☒ Position ☐ Velocity ☐ Acceleration

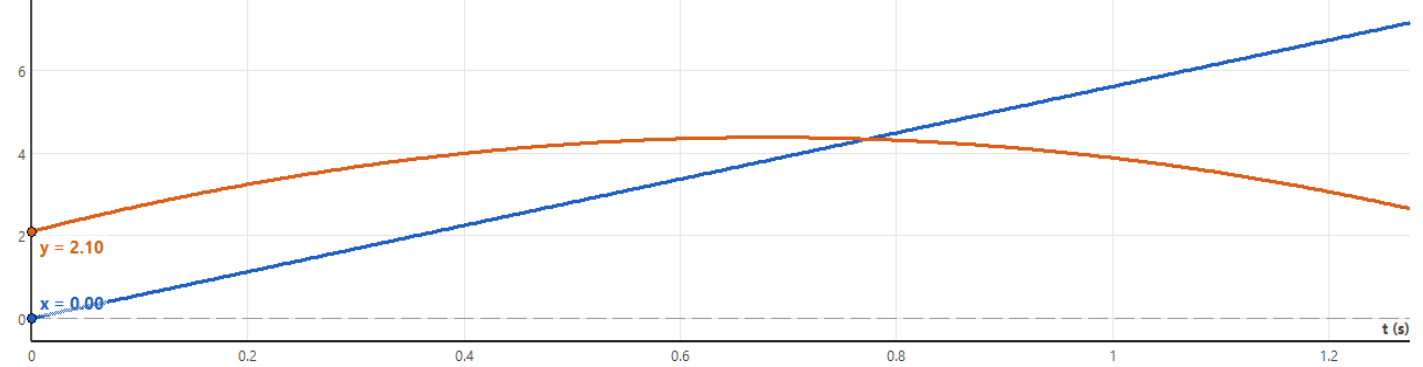
PREDICTION MODE on: the class commits before the graphs exist.

TWO WHAT-IFS — VOTE FIRST

1. Same v_r , angle down to 40° : make or miss?
2. Same angle, v_r down 2%: how far does the landing point move?

Ready. PLAY: the player rises to h_r , the ball leaves with v_r at θ_r , and ideal projectile motion begins

Position (m): x linear, y quadratic



The reveal: $x(t)$ linear, $y(t)$ quadratic — the same two shapes you fitted in Tracker. A basketball shot is the four-ball experiment with a vertical launch component.



Why the 2% question bites

Range scales with v_r^2 — a 2% speed error moves the ball centre about 4% of the range, $\approx 0.27 \text{ m}$ here, against a centre window of $\pm 0.11 \text{ m}$. Release speed is the shooter's hard problem.



Connect back

PREDICT here plays the role of the vote before the 120 fps reveal; the graphs play the role of the Tracker fits. Same rhythm: predict $\rightarrow$ run $\rightarrow$ compare.

Observe



Idealize



Model



Solve / Analyze



Validate



Report



EXPLORE WITH AI — and check the machine before you trust it

IDEAL PLANAR PROJECTILE MOTION AFTER RELEASE (Version 1 model)
no drag, no spin, no lateral error, no rim / backboard rebound

Release: $v_r = 8.73$ m/s, $\theta_r = 50.0^\circ$, $h_r = 2.10$ m, $L = 6.75$ m, $g = 9.81$ m/s²

GENERATED EQUATIONS:

$$\begin{aligned} x(t) &= v_r \cos \theta_r \cdot t = 5.61 \cdot t \quad [\text{m}] & v_x &= v_r \cos \theta_r = 5.61 \text{ m/s (constant)} & a_x &= 0 \\ y(t) &= h_r + v_r \sin \theta_r \cdot t - \frac{1}{2}gt^2 = 2.10 + 6.69 \cdot t - 4.905 \cdot t^2 \quad [\text{m}] & v_y &= 6.69 - 9.81 \cdot t \quad [\text{m/s}] \\ a_y &= -g = -9.81 \text{ m/s}^2 & \text{apex } t &= 0.68 \text{ s (} v_y = 0, v_x = 5.61 \neq 0 \text{)} & \text{rim plane reached at } t &= 1.20 \text{ s} \end{aligned}$$

DERIVED QUANTITIES:

apex: $t = 0.682$ s, $x = 3.825$ m, $y = 4.379$ m ($v_y = 0$, $v_x = 5.61$ m/s)
ball centre crosses the rim plane ($y = h_{rm} = 3.05$ m): $t = 1.202$ s, $x = 6.747$ m, $e_x = -0.003$ m
entry angle = 42.3° , clearance front +0.106 m / back +0.112 m
TOP view freeze: $t = 1.178$ s — ball bottom reaches rim height ($y = h_{rm} + r_b$)
flight solved until $t_{end} = 1.276$ s → RESULT: NOTHING BUT NET

At $t = 0.00$ s: $x = 0.000$ m, $y = 2.100$ m, $v_x = 5.612$ m/s, $v_y = 6.688$ m/s

Geometry: $h_{rm} = 3.05$ m, $D_h = 0.457$ m, $D_b = 0.239$ m, centre window $|e_x| < 0.109$ m

MATH / EQUATIONS pane: the model of the CURRENT shot — generated equations, apex, rim-plane crossing, entry angle, result.



AI EXPLORATION PROMPT — GO ONE STEP DEEPER

"I am studying projectile motion with a basketball-shot simulator (ideal model: no drag or spin). Propose ONE effect the model neglects that would matter most for a real 6.75 m three-point shot, estimate its size, and tell me what measurement from a filmed shot would confirm it. Ask me to predict before you reveal."

Same rule for every tool: PMA, Tracker, AI, your calculator — none is the authority; the model and its stated assumptions are. Verify, then trust.

ENGINEER'S CHECK BOX — VERIFY THE PANE BY HAND

- ☐ Rim-plane time: solve $2.10 + 6.69t - 4.905t^2 = 3.05$. $t = 1.20$ s? Which root did you keep, and why?
- ☐ Landing: $x(1.202 \text{ s}) = 5.61 \times 1.202 = 6.75 \text{ m} \rightarrow e_x \approx 0$ — "nothing but net", by hand.
- ☐ Entry angle: $v_y = 6.69 - 9.81(1.202) = -5.10 \text{ m/s} \rightarrow \tan^{-1}(5.10/5.61) = 42.3^\circ$. Matches the pane?
- ☐ Apex: $v_y = 0$ at $t = 0.68$ s — but $v = v_x = 5.61 \text{ m/s} \neq 0$. The apex-velocity misconception, closed twice.

Observe



Idealize



Model



Solve / Analyze



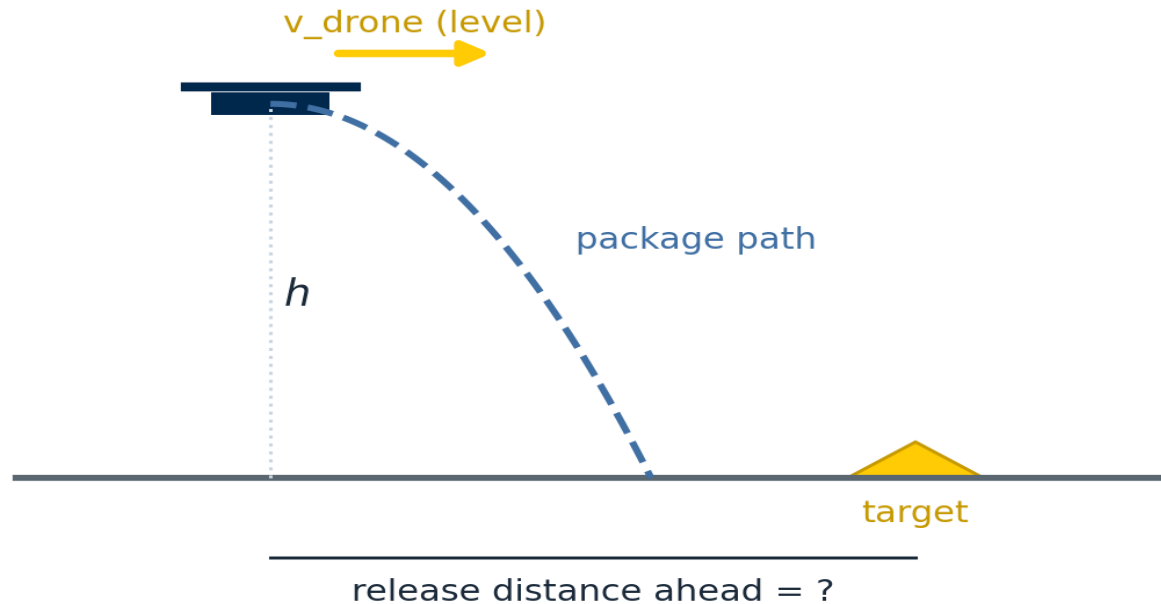
Validate



Report



A drone flying level must release a package to hit a ground target. When should it let go?



The trap: "the package starts from rest." It does not — it starts with the drone's velocity. Everything else follows from the four-ball model.

THE MODEL

Particle	the package (mass center)
Coordinates	x horizontal along the flight, y up; origin at the release point
Knowns	$h = 40 \text{ m}$; drone speed 12 m/s , level; $g = 9.81 \text{ m/s}^2$
Assumptions	no wind, negligible drag, package released cleanly (no push), level target
Modeling decision	release velocity of the package = velocity of the drone: $v_{0x} = 12 \text{ m/s}$, $v_{0y} = 0$
Unknowns	release distance d ahead of the target; impact speed and angle



Release 34 m before the target; the package lands at 30 m/s, 67° below horizontal

SOLVE

$$t_f = \sqrt{(2h/g)} = \sqrt{(80/9.81)} = 2.86 \text{ s}$$

$$d = v_{0_x} t_f = 12 \times 2.86 = 34.3 \text{ m}$$

$$v_y = -g t_f = -28.0 \text{ m/s}, \quad v_x = 12 \text{ m/s}$$

$$|v| = \sqrt{(12^2 + 28^2)} = 30.5 \text{ m/s} \quad \text{at } 66.8^\circ \text{ below horizontal}$$

The drone must release the package about 34 m before it is over the target — while the package still carries the drone's 12 m/s forward.



Interpret

Release early; the package "leads" the target. At impact it is moving almost three times faster than the drone — mostly downward. That number sizes the package's cushioning.



Extension — hillside target

If the target sits on a 15° upslope, the landing point satisfies $y = -x \tan 15^\circ$ instead of $y = -h$. Set that up (do not solve now): which equation changes?



Checks that cost thirty seconds and catch most errors



Limits. $h \rightarrow 0 \Rightarrow t_f \rightarrow 0, d \rightarrow 0. v_{\text{drone}} \rightarrow 0 \Rightarrow d \rightarrow 0$ (straight drop). Both sensible.



Independent route. $|v|^2 = v_o^2 + 2gh = 144 + 785 \Rightarrow 30.5 \text{ m/s}$. Matches (energy, Chapter 4).



Sign and direction. $v_y < 0$ at impact; angle steep and downward. Matches the sketch.



Misconception — "the package starts from rest"

Then $d = 0$ and the drone would release directly overhead — and miss by 34 m. Released objects inherit the carrier's velocity.



Misconception — "velocity is zero at the top"

For an angled launch, only v_y is zero at the apex; v_x is untouched. The velocity is horizontal there, not zero.



AI as Critic

"Here is my drone solution. I assumed no wind. Critique whether a 5 m/s headwind would change d by more than the 120 fps timing resolution would, and how I could estimate it."

MODEL and VALIDATE are engineering actions, not labels: an unstated assumption ($v_o = 0$) is the single most expensive error in this Part.

Observe



Idealize



Model



Solve / Analyze

**Validate**

Report



Before the synthesis

Q1 — At the apex of an angled launch, the velocity is...

- A) zero
- B) horizontal, equal to $v_0 \cos \theta$
- C) horizontal, equal to v_0
- D) vertical, momentarily

Q2 — A ball launched at v_0 lands on a ledge HIGHER than the launch point. Its landing speed is...

- A) greater than v_0
- B) equal to v_0
- C) less than v_0
- D) depends on the angle only

Answers in the notes. Defend your vote with an equation, then with a sentence.

Observe



Idealize



Model

**Solve / Analyze**

Validate



Report



Part 3 — what you can now explain and do



You can now EXPLAIN

- The projectile model's assumptions, from memory, and which experiment would test each one
- Why the horizontal launch speed changes the range but not the flight time
- Why the four balls should — ideally — land together, and what a real difference could mean
- What one video frame is worth at 30 fps versus 120 fps, and how that uncertainty propagates to t , v_{0x} and g



You can now DO

- Set up origin and axes, write $x(t)$, $y(t)$, and solve for a landing point, a flight time, or an unknown launch condition
- Infer a launch speed from a range and a frame count; compare it with a Tracker fit
- Fit $x(t)$ and $y(t)$ and read v_{0x} , v_{0y} , g_{eff} from the fits — and say what the software computed
- Rank sources of disagreement by their fingerprint in the data and propose the measurement that would settle it

Next — Part 4: when the PATH is known, another description exposes what Cartesian components hide. The bicycle wheel and the bent wire make speed-change and direction-change visible.



An inverse problem with only a tape measure

MEDIA PLACEHOLDER

[INSERT INSTRUCTOR COMPOSITE — STEEL BALL IN FREE FLIGHT FROM THE CURVED RAIL]

Existing multiple-exposure composite with origin on the floor below the launch point; label y_A , x_B , y_{max} .



Given only three lengths

launch height y_A , landing distance x_B , apex height y_{max} — no clock.



Find

(a) v_o at A, (b) v_B at landing, (c) time of flight. Hint: the apex gives v_{oy} from $v_y^2 = v_{oy}^2 - 2g(y_{max} - y_A)$; the landing then gives t and v_{ox} .



Why it is here

A launch at an ANGLE, solved backward from geometry — a good self-assessment challenge after the horizontal four-ball case.