S.T. Yau High School Science Award

Research Report

The Team

Name of team member: Berkeley Chau
School: Palos Verdes Peninsula High School
City, Country: Rancho Palos Verdes, CA 90275, United States

Name of supervising teacher: Tim Swartz
Job Title: Professor
School/Institution: Simon Fraser University
City, Country: Burnaby, BC V5A 1S6, Canada

Title of Research Report

Advancing Putting Strategy Through Physical Modeling and Simulation of Golf Ball Rolling Dynamics and Hole-Capture Mechanics

Date

September 11, 2026

Advancing Putting Strategy Through Physical Modeling and Simulation of Golf Ball Rolling Dynamics and Hole-Capture Mechanics

Berkeley Chau

Abstract

We implement a simulation-based approach to determine and evaluate putting strategies depending on the green's parameters and golfers' skill levels. For various greens and multiple skill profiles ranging from an amateur to professionals, we generate and rank targeting strategies by comparing their expected strokes to hole out. Monte Carlo simulations are used to generate random putting trajectories that account for execution errors associated with a player's skill level. Our Python-based computer program combines existing physical models for a golf ball's rolling trajectory and hole capture conditions, and is the first complete putting-strategy simulation that fully incorporates a golf ball's rolling trajectory and rim-interaction dynamics, which includes a ball's lip-out behavior and flyover state. To reduce computation time and resources deriving from the stochastic nature of a golfer's performance, a non-recursive method was developed to determine a strategy's expected strokes to hole out. Our simulator is calibrated and validated with extensive data in the literature. Accounting for winner's curse selection bias and false discovery rate across multiple strategies, we are able to identify winning strategies compared to Dave Pelz's baseline at different putt lengths, green slopes, and golfers' skill. Amateurs benefit particularly from aiming higher and softer on longer putts with higher break.

Keywords: golf putting, Monte Carlo simulation, strokes gained, hole capture model, execution error, winner's curse, false discovery rate

Acknowledgement

1. Berkeley Chau would like to thank Professor Tim Swartz for his guidance and support throughout the research process. His expertise and feedback were invaluable in shaping the direction of this project. Tim Swartz's mentorship ensured that the research was conducted scientifically, and he provided valuable insights that helped revise the paper.

2. Berkeley Chau would also like to acknowledge the golf community. When Berkeley asked for help to find PGA Tour expected strokes to hole out data, Scott Fawcett (the CEO of the Decade golf course management system) assisted in finding the PGA Tour data, which is the basis of the continuation putts function.

3. Berkeley Chau is indebted to his parents for the support and encouragement they provided throughout the research process. Their unwavering belief in his abilities helped Berkeley push through the scientific process.

4. Contributions: Berkeley Chau conceived the research idea, developed the simulator, designed and executed the experiments, analyzed the results, and wrote the manuscript. Professor Tim Swartz offered feedback on the robustness of the research design, as well as guidance on the interpretation of the results and the writing of the manuscript.

Declaration of Academic Integrity

The participating team declares that the paper submitted is comprised of original research and results obtained under the guidance of the instructor. To the team's best knowledge, the paper does not contain research results, published or not, from a person who is not a team member, except for the content listed in the references and the acknowledgment. If there is any misinformation, we are willing to take all the related responsibilities.

Names of team members – Berkeley Chau

Signatures of team members B Chau

Name of the instructor – Tim Swartz

Signature of the instructor - TS

Date – Sept 11/26

Table of Contents

1 Introduction **1**

2 Analytical Modeling of Golf Ball Rolling Dynamics and Hole Capture **2**
2.1 Symbols and conventions . . . 2
2.2 Green speed . . . 3
2.3 Rolling dynamics . . . 4
2.4 Hole-capture model . . . 4

3 Simulation of Golf Ball–Green Interaction and Hole Capture Dynamics **8**
3.1 Simulator scope, inputs, and outputs . . . 9
3.2 Numerical integration . . . 9
3.3 Green-to-hole transition . . . 10
3.4 Stopping conditions . . . 10

4 Study of Putting Strategy Improvements **11**
4.1 Physical parameterization of putting scenarios . . . 11
4.1.1 Scenario generation . . . 11
4.2 Computing the Pelz baseline starting conditions . . . 13
4.3 Strategy construction . . . 14
4.4 Models for execution error and green-reading error . . . 14
4.5 Scoring: expected strokes and the continuation model $N(d)$ . . . 16
4.6 Monte Carlo evaluation and validation . . . 18
4.6.1 Initial Monte Carlo evaluation . . . 18
4.6.2 Fresh Monte Carlo revalidation and determination of winning strategies . . . 20

5 Validation of Putting Simulator **21**
5.1 Verification against source models . . . 21
5.2 External validation . . . 21

6 Results and Discussion **24**
6.1 Strategy gains for short putts . . . 24
6.2 Strategy gains for putts of longer length . . . 25
6.3 Robustness and sensitivity analysis . . . 30
6.3.1 Physical and numerical assumptions . . . 30
6.3.2 Error-model sensitivity . . . 33

7 Conclusion **35**

References **37**

1 Introduction

Ever since golf's conception, putting has often been treated as a "sport within a sport." Following his missed cut at the 1957 Masters, legendary golfer Ben Hogan once remarked, "I have always contended golf is one game and putting is another. One game is played in the air, the other is played on the ground" (Grimsley, 1957). However, putting is a component of golf that has lacked thorough statistical analysis, with golfers often supporting their putting "strategies" with "feel" and anecdotal advice. Dave Pelz was one of the first to change this. With physical experimentation, Pelz stated that a putt's optimal trajectory passes through the center of the cup on a covered hole and stops 17 inches past the hole (Pelz and Frank, 2000, p. 190). However, Pelz's targeting strategy does not account for variations in putt slope, length, and speed. Additionally, Pelz's threshold does not take into account a golfer's skill level. While Yousefi & Swartz model a putt's difficulty as a function of its length and orientation relative to the hole (Yousefi and Swartz, 2013), they do not provide a method to determine a better targeting strategy than Pelz's default. So, we determine the physical green and skill conditions where alternative targeting strategies matter most. Broadie & Shin used a simulator-based approach to find an optimal putting strategy by using PGA ShotLink data to estimate touring professionals' errors in putting (Broadie and Shin, 2014). However, they only find an optimal strategy for touring professionals and assume the optimal strategy must pass through the hole's center. Furthermore, their simulation model for a golf ball does not account for a ball's lip-out. For hole-interaction models, Broadie & Shin strictly applied Penner's capture threshold, which is a quadratic fit based on Holmes' capture threshold (Penner, 2002; Holmes, 1991). There is one flaw in using this threshold: at the hole, a ball is either captured or is allowed to continue rolling at an unchanged pace, which does not account for a ball's motion on the rim.

To address these limitations, we implement Hogan & Antali's capture model to account for a ball's rim-motion around the hole (Hogan and Antali, 2025). Incoming balls can either fall in after rotating with the rim, "lip out" after rotating a set angle around the rim, or even separate from the rim's surface, causing a ballistic flyover. Hogan & Antali's model does not explicitly model a ballistic flyover. Therefore, for balls that separate from the rim's surface, we still use Penner's capture threshold (Penner, 2002) to determine whether the ball is captured or not, making a hybrid hole-capture model. We use this more refined hole-capture model so we can accurately assess where the ball stops on the putting surface, which we need to determine an optimal targeting strategy (the closer a ball stops to the hole, the fewer strokes a golfer is expected to take to hole out from that position).

We also use player error models from Broadie & Shin and Bansal & Broadie, which are calibrated from ShotLink data and account for a player's skill (Broadie and Shin, 2014; Bansal and Broadie, 2008). By adding a skill component to the simulation, it allows for the incorporation of a player's individual "error" in putting. To validate our simulator, we reproduce Fearing et al.'s published make-rate probability curve, a logistic regression based on empirical ShotLink data (Fearing et al., 2011). Additionally, we reproduce the capture-rate regions shown in Penner's figures (Penner, 2002).

Using the validated simulator, we develop a method to find strategies better than the Pelz

baseline for different scenarios consisting of varying putt lengths, green speeds, and slopes. For each scenario, alternative strategies are generated that differ in the starting line and speed from the Pelz targeting strategy. To account for a player's skill errors, Monte Carlo simulations are used to generate all strategy's expected strokes to hole out. To determine winning strategies, we compare the alternative strategy's expected strokes to hole out to the Pelz baseline, finding strategies' "strokes gained" on the Pelz baseline (Broadie, 2012). Next, we redetermine a strategy's strokes gained with new Monte Carlo simulations to account for the winner's curse (Capen et al., 1971; Smith and Winkler, 2006) arising from the random Monte Carlo simulations. Due to the large number of scenarios run, we control multiple strategies' false discovery rate of beating the Pelz baseline with Benjamini–Hochberg correction (Benjamini and Hochberg, 1995).

For short-range putts, neither pros nor amateurs gain significantly from alternative strategies. On average from alternative strategies, all skill profiles only gain -0.0010–0.0147 strokes per putt, which translates to at least about seventy putts on that length to gain a single stroke. Our findings demonstrate that putting strategy affects amateurs' scoring for putts of longer length, and suggest that a conservative strategy (hitting putts slower, aiming for more break) is the best strategy for amateurs on longer putts with high break. Professionals also benefited from a conservative strategy on the longest putts with the most break. However, after averaging strokes gained across the longest putts and the shorter putts in the lag putt dataset, their strokes gained from alternate strategies are negligible compared to amateurs (≈ 0.037 strokes per putt for amateurs on average compared to touring professionals losing strokes with alternate strategies).

2 Analytical Modeling of Golf Ball Rolling Dynamics and Hole Capture

2.1 Symbols and conventions

Table 1 collects the symbols and conventions used throughout this section. Our putting surface is a planar surface with a single slope, which is measured from two components per Penner's conventions: the slope along the line from the ball to the hole (uphill/downhill slope, ϕ), and the slope perpendicular to that line (sidehill slope, θ) (Penner, 2002). A positive ϕ indicates an uphill slope, and a positive θ indicates a putt moving to the left by gravity. For our putting axes, the positive y-axis is defined as the line from the golf ball's initial point to the hole. From the bird's-eye perspective of the green, the positive x-axis is defined as perpendicular and clockwise to the y-axis. Initially, the ball starts at $(0, 0)$. So, the green's "hole" coordinates are $(0, D)$, where D is the distance to the hole center from the initial putt. We model a ball's starting conditions as a pure-rolling golf ball with no initial spin, so the ball's initial velocity is defined by its startline angle and initial speed. We define the startline angle, ψ, as the angle relative to the line intersecting the ball's initial start position and the hole's center. Positive ψ would aim the ball towards the positive x-axis, so if $\psi = 90°$, the ball's initial startline would be along the positive x-axis. Similarly, we define v_0 as the initial speed at which the ball is rolled. Therefore, our initial velocity would be with respect to

Symbol	Meaning	Units	Convention / value
x, y	putting-plane axes	m	$+y$: ball $\rightarrow$ hole; $+x$: perpendicular, clockwise
θ	sidehill slope angle	deg	positive: ball breaks left
ϕ	uphill/downhill slope angle	deg	positive: uphill toward the hole
ψ	startline (aim) angle	deg	from the ball–hole line; positive toward $+x$
D	distance to hole	ft	from the ball's starting point
d	leave distance	ft	resting position to hole center; scoring model, Section 4.5
v_0	initial ball speed	m/s	pure-rolling launch speed
v	instantaneous speed of the golf ball	m/s	—
β	instantaneous velocity direction of the golf ball	deg	from $+y$; equals ψ at launch
m, r	ball mass, radius	kg, m	0.04593, 0.02135
R	hole radius	m	0.054
g	gravitational acceleration	m/s^2	9.80665
S	Stimpmeter reading (green speed)	ft	—
ρ_g	rolling-resistance coefficient	—	$\approx 0.784/S$
I_{b}	inertia ratio I/mr^2	—	2/5 (assuming a uniform solid sphere)
δ	impact distance at the rim	m	hole-interaction model
γ	elevation angle of ball's center relative to hole's rim	rad	Hogan–Antali's rim angle, renamed; hole-interaction model

Table 1: Symbols and conventions for the simulator.

our coordinate system: $\mathbf{v}(0) = v_0\,(\sin\psi,\ \cos\psi)$.

2.2 Green speed

Green speed is defined as the Stimpmeter reading of the green, a measure of the green's frictional resistance to a rolling golf ball. The stimp S of a putting green is the distance in feet a golf ball rolls on a flat green when released from a standard Stimpmeter device (Penner, 2002). So if the putting green is of 10 stimp, it would run out 10 feet after being released by the Stimpmeter. Holmes, as reported by Penner (2002), calculates the initial speed a ball rolls down a Stimpmeter to be $v_S = 1.83$ m/s. Penner's equations of motion for a golf ball use a different measurement of green speed, ρ_g. We find Penner's friction coefficient from a green's S by utilizing Penner's given deceleration coefficient for a golf ball rolling on a flat green, $a_y = -\frac{5}{7}\,\rho_g\,g$ (Penner, 2002, Eq. (5)). By setting Penner's flat-green rollout equal to the Stimp in meters S_{m} of a green, we can solve for ρ_g in terms of S_{m} and v_S:

$$\rho_g = \frac{7v_S^2}{10\,g\,S_{\mathrm{m}}}, \tag{1}$$

Stimp in meters being $S_{\mathrm{m}} = 0.3048\,S$. This equation matches Penner's own conversion from stimp to ρ_g, with Penner's printed Equation (6) values being reproduced within an error of 7×10^{-4} in ρ_g units.

2.3 Rolling dynamics

We use Penner's derived differential equations of motion for a rolling golf ball (Penner, 2002), which assume the ball is in pure rolling motion, with "sliding" effects being negligible. The equations take into account not only the rolling of the rigid golf ball with a fixed inertia, but also the "retarding" frictional forces during motion due to the softness of the greens. The differential equations are a combination of two components: gravity and friction. Gravity is defined as g; β represents the angle the ball is heading from the positive y-axis, and I_{b} represents the dimensionless ratio I/mr^2, where m is the mass of the golf ball and r is the radius of the ball; I_{b} is a constant at $2/5$. The magnitude of the friction component is f, and λ represents the friction's direction (from the $+y$ axis):

$$m\ddot{x} = -mg\sin\theta - f\sin\lambda, \tag{2}$$

$$m\ddot{y} = -mg\cos\theta\sin\phi - f\cos\lambda, \tag{3}$$

$$\tan\lambda = \frac{\rho_g\cos\theta\cos\phi\,\sin\beta - I_{\mathrm{b}}\,\sin\theta}{\rho_g\cos\theta\cos\phi\,\cos\beta - I_{\mathrm{b}}\,\cos\theta\sin\phi}, \tag{4}$$

$$f = \frac{\rho_g\cos\theta\cos\phi\,\cos\beta - I_{\mathrm{b}}\,\cos\theta\sin\phi}{(1+I_{\mathrm{b}})\cos\lambda}\,mg, \tag{5}$$

$$\beta = \mathrm{atan2}(\dot{x},\,\dot{y})\,. \tag{6}$$

2.4 Hole-capture model

We implement a hybrid hole-interaction model by combining Hogan & Antali's rim motion model, and Penner's capture threshold for a flat hole (Hogan and Antali, 2025; Penner, 2002). Penner's capture condition alone (as Broadie & Shin use it, Broadie and Shin, 2014) strictly decides capture versus no capture, but does not give an exit position after rim motion (when the ball "rolls" across the lip of the hole) and does not distinguish capture from lip-out like Hogan & Antali's model does (when the ball rolls on the rim and returns to the green's surface). We account for a ball's exit position and velocity after a lip-out, which is important for determining the ball's next trajectory after a lip-out by using the hybrid model. Table 2 briefly summarizes the origin of each component for the hole-interaction model. Because Penner assumes pure rolling motion, we do not use Hogan & Antali's hole model with spin motion. Therefore, the model's rim-entry state can be described as two quantities: entry speed v and impact distance δ (the perpendicular distance of the velocity's direction from the hole center). We can calculate the impact distance, δ, using our equation:

$$\delta = \frac{|\,\Delta x\,v_y - \Delta y\,v_x\,|}{v}, \tag{7}$$

where Δx and Δy are the x and y positions of the ball relative to the hole's center on the rim, and v_x and v_y are the x and y components of the velocity vector. After the interaction with the hole model, the ball can be in three states: captured in the cup, "lipping" out after rotating a set angle around the hole, or ballistically exiting, "flying" over the rim and returning in the same direction. Hogan & Antali define the possible states of the ball's interaction with the hole with four regions in

the (v_0, δ) space. The first region is when the ball is moving fast enough to "separate" from the rim in a "flyover". Hogan & Antali did not make a capture condition or describe the motion of a golf ball in the flyover region, stating it is out of their scope. Therefore, we use Penner's calibrated capture threshold for region 1, where the ball can either be captured or leave the hole in a ballistic exit with unchanged velocity. In region 2, the ball briefly interacts with the rim before losing contact and dropping into the cup. In regions 3 and 4, the ball rotates around the rim and does not separate, where it can either fall in the hole (region 4) or lip out after rotating a set angle (region 3). All four region tests are applied sequentially in numerical order; each result's outcome ensures the next test has a valid input domain.

Component	Origin
Four-region hole-capture structure; separation curve c_1, fall-in curve c_2, rim energy function	Source: Hogan & Antali's spin-free flat-hole model (Hogan and Antali, 2025)
Region-1 capture rule, Eq. (9)	Source: Penner's fit to Holmes' capture map (Penner, 2002, Eq. (23)), flat-hole form
Impact distance δ, Eq. (7)	Reformulation: standard geometry to convert between the trajectory model and hole capture model in our coordinates
Rim elevation angle γ	Reformulation: Hogan & Antali's rim angle, renamed
Flat (horizontal) hole slope	Approximation: kept for consistency with the spin-free rim model that assumes a flat slope
Lip-out exit state calculation	Reformulation of Hogan & Antali's conservation of speed results (Hogan and Antali, 2025, §3.4 and §4.2)
Root finding and roll-around angle formulation	Implementation choice: numerical implementation of stated formulas only (Brent's method, bounded minimization, adaptive quadrature)
Green-Hole Transition	Implementation choice: switching states between trajectory and hole models by interpolating the ball's state between timesteps; timestep sensitivity in Section 6.3
Ballistic-exit position handling at a distance $2\sqrt{R^2 - \delta^2}$ from the initial point	Implementation choice: exit geometry only, no role in capture decisions
Unchanged exit speed after a ballistic exit	Assumption: No speed loss transition; energy-loss investigated in Section 6.3
Zero-time hole episodes	Implementation choice: elapsed time underestimated, on the order of a second per encounter; does not affect results (Hogan and Antali, 2025, p. 18)
Omission of a golf ball's spin	Scope boundary: omits spin-mediated escapes from inside the hole, bounded to a tiny corner of region 4 (Hogan and Antali, 2025, §7 and Fig. 12); worst-case investigation in Section 6.3

Table 2: Sourcing for all the components of the hole model: methods derived directly from source papers, reformulations of source papers, implementation choices the sources do not specify explicitly, and physical assumptions replacing unmodeled physics.

Region 1: rim separation capture and exit conditions. Hogan & Antali's first region curve, which they attributed to Holmes' condition, is defined as (Hogan and Antali, 2025, Eq. (4.17)):

$$v^2 > c_1^2(\delta) = g\,r\,/\big(1-\delta^2/R^2\big)\,. \tag{8}$$

Hogan & Antali's equation defines when the ball separates from the rim without rolling around the rim. When the ball separates from the rim and does not get captured (returns to the green's surface), it is considered a "flyover" putt. Hogan & Antali do not explore a capture condition for this region, noting that Holmes found seven distinct outcomes there (Hogan and Antali, 2025, endnote 17).

Holmes' free-fall-only model (a subset of his full capture model) captures if the ball falls one ball radius before its edge strikes the back wall (Holmes, 1991). However, this free-fall model has a capture threshold of 1.31 m/s, too conservative for his full 1.626 m/s capture boundary (Hogan and Antali, 2025, §2). Therefore, for all rim-separation region 1 scenarios, we use Penner's capture threshold that is calibrated to Holmes' full capture map, which includes capture modes beyond strict free-fall (Penner, 2002, Eq. (23)). Penner's capture threshold is as follows: the ball is captured if and only if

$$v \;\le\; v_c(\delta) = (1.63\ \text{m/s})\left(1-\frac{\delta^2}{R^2}\right). \tag{9}$$

We define a ball exiting from a flyover state as a ballistic exit, with the exit position being the far-rim point on the other side of the hole the ball was crossing. The distance from the ball's original entry state to the ball's exit state across the hole can be found with the equation $2\sqrt{R^2-\delta^2}$. For the ball's velocity after the ballistic exit, we intentionally do not model any loss or change of velocity due to the hole interaction because of a lack of explicit data modeling the energy loss of the golf ball after interacting with the hole. However, we probe the potential effects caused by modeling a simple energy loss of the ball's velocity after a ballistic exit on our final results (Section 6.3).

Region 2: fall-in from lost rim contact. The second region of hole capture per Hogan & Antali's spin-free model is when the ball is entering slowly enough to fall into the hole after brief interaction with rim motion (Hogan and Antali, 2025, §4.3 and Eq. (4.20)). More specifically, Hogan & Antali derive the region 2 threshold from when the normal contact force from the hole rim onto the golf ball reaches zero during rim motion; below region 2, the rim would actually have to pull on the ball to keep it in contact with the rim, so the ball detaches and free-falls in. Unlike region 1 flyovers, the ball in this case actually enters rim motion. The threshold to be in region 2 motion, per Hogan & Antali's equations, is as follows:

$$v^2 < c_2^2(\delta) = \frac{2\,g\,r/(1+I_\mathrm{b})}{R\,\delta^2/(R-r)^3-1}. \tag{10}$$

Regions 3 and 4: lip-out versus rim capture. To find the boundary between regions 3 and 4, we do not use a generic boundary curve because the region 3/4 boundary exists only in parametric

form (Hogan and Antali, 2025, Eqs. (4.23) and (4.24)). Hogan & Antali's third and fourth capture regions from their spin-free capture motion are both derived from the same energy function, which describes the squared descent rate of the ball along the rim (proportional to the energy of the ball's descent into the hole), with γ describing the elevation angle of the ball's center relative to the rim line ($\pi/2$ at rim arrival, 0 at the hole entry). Hogan & Antali call this angle β; to avoid confusion with our velocity direction angle, we rename the elevation angle γ. Hogan & Antali's energy function is as follows (Hogan and Antali, 2025, Eq. (4.11)):

$$\dot{\gamma}^2(\gamma) = \frac{v^2}{r^2}\left(1 - \frac{\delta^2}{(R - r\cos\gamma)^2}\right) + \frac{2g}{r\,(1 + I_{\mathrm{b}})}\,(1 - \sin\gamma)\,. \tag{11}$$

If this energy function has a "root" where the change in descent angle is zero, then the ball has stopped descending into the hole and will ascend out of the hole. Per Hogan & Antali's separatrix figure and quotations, the elevation angle associated with the energy function's root is the "minimum" elevation angle relative to the hole (Hogan and Antali, 2025, p. 11 and Fig. 5). Because the energy function could potentially have multiple roots, we take the first zero of $\dot{\gamma}^2$ descending from $\gamma = \pi/2$. The energy of the ball's descent after the first zero (for elevation angles below the zero) is actually initially negative, proving that the roots with elevation angles closer to zero are not physically accessible. $\dot{\gamma}^2(\gamma)$ is therefore first checked to see if it has a positive value at an elevation angle of 0. If it does not, a clear lip-out has occurred (negative energy at capture), and we solve for the elevation angle with Brent's method over $[0, \pi/2]$ directly. If the energy at the 0 elevation angle is positive, there are two possible scenarios. The first is when there is just no root at all, and the ball's descent energy remains positive for all elevation angles, entering hole motion (region 4) and being captured. However, there could have been multiple roots (implying a lip-out) that cause the ball's energy to appear positive at both endpoints. To check for both possibilities, we check for a sign change (the change from positive to negative energy) with a bounded minimization, checking for a minimum energy value below zero. If the minimum value is positive, the ball enters region 4 and is holed. Otherwise, the minimum energy is negative. Since the energy function is continuous on $[0, \pi/2]$ and the energies at both endpoints are positive, a negative minimum energy implies multiple roots. To find the physically plausible root (the solution with the highest elevation angle), we solve for the minimum elevation angle with Brent's method with a search bound defined between the global minimum energy's γ and $\gamma = \pi/2$. The values of (δ, v) that have a root of the energy function make up region 3. Due to our omission of spin, our model omits spin-mediated escapes when the ball is inside the hole. However, Hogan & Antali attribute hole lip-outs to a tiny corner of region 4 (Hogan and Antali, 2025, §7 and Fig. 12).

Lip-out exit status. Next, we have to calculate the exit position in the case of a region 3 lip-out. Our exit position is defined by our entry rim point, rotated by the "lip-out" angle and direction of the lip-out. This angle is found by the roll-around angle found in Hogan & Antali's paper (Hogan and Antali, 2025, Eqs. (4.13) and (4.14)). Eq. (4.13) only accounts for the roll-around angle from

the ball's entry point to the place where the ball is lowest relative to the rim. It does not account for the ball ascending out of the rim, and because of the motion's symmetry, Eq. (4.14)'s integral doubles this component, calculating the total lip-out angle. We use this equation to calculate our lip-out angle. We can determine the lip-out's direction (clockwise or counterclockwise, viewed from the bird's-eye perspective of the green) by the sign of $\Delta x\, v_y - \Delta y\, v_x$, the signed numerator of Eq. (7) evaluated at the rim-entry point. If the expression is positive, it is a counterclockwise lip-out; otherwise, it is in a clockwise direction. Hogan & Antali show that during the rim motion, the energy of the ball is conserved (Hogan and Antali, 2025, §3.4), and thereby the speed of exit of the ball is equivalent to its entry. Furthermore, it is stated in the paper that the angular velocity (and thus the tangential velocity) around the hole is the same at the start and end of rim motion (Hogan and Antali, 2025, §4.2). When the ball enters the hole, its velocity can be thought of as two components: a velocity tangent to the rim's surface, combined with a "radial" velocity component seeking inwards to the hole. At rim entry, the ball sits at distance R from the hole center. The component of velocity perpendicular to the radius vector can be found as

$$v_t = \frac{|\,\Delta x\, v_y - \Delta y\, v_x\,|}{R} = \frac{v\,\delta}{R}. \tag{12}$$

The conservation of speed throughout the hole motion, as well as the preserved tangential component, leaves the radial component fixed. If we use the Pythagorean theorem, we find the radial component's magnitude:

$$v_r = \sqrt{v^2 - (v\delta/R)^2}. \tag{13}$$

When the ball exits the hole, we simply reverse the radial component so it seeks outwards, keep the same tangential component, and rotate it per the "lip-out" angle.

Multiple hole interactions. Upon a lip-out or a ballistic exit, the ball is placed on the rim, but moving outward radially. Therefore, the simulator never instantaneously retriggers hole modeling. However, if the ball rolls back and crosses the rim again, the hole model will trigger again, allowing for a realistic simulation with multiple hole interactions. Finally, a hole episode consumes no simulated time; a continuation will keep the same rim-entry time. This is because for all region 1 and 3 exit scenarios, we do not calculate the time elapsed for each hole episode. Per region encounter, the underestimated time elapsed could be on the order of a second (Hogan and Antali, 2025, p. 18). Nonetheless, we do not use our timestep count for anything but terminating our simulator based on a pre-determined max simulation time threshold.

3 Simulation of Golf Ball–Green Interaction and Hole Capture Dynamics

Our simulator is a deterministic model, combining Penner's rolling dynamics (Penner, 2002) on a planar green with a custom hole model combining Hogan & Antali's lip-out condition (Hogan and

Antali, 2025) with Penner's capture threshold for putts that do not circle the rim. The simulator's inputs consist of a green's "scenario" and start conditions. A scenario is the green's physical parameters: the green's horizontal and vertical slopes, green speed, and distance to the hole. Start conditions are the ball's initial speed and direction. The outcome of our simulator is where the ball ends up on the green. We verify our implemented equations by reproducing source outcomes (Section 5.1).

3.1 Simulator scope, inputs, and outputs

The simulator's putting surface is a constant-slope putting green that is completely unbounded; nothing stops or reflects the ball. We do not model a green's "grain", change in slope, or a ball's initial "skid" and "spin". Because we do not include spin in our simulator, we use Hogan & Antali's spin-free capture model, which excludes certain lip-out conditions (Hogan and Antali, 2025). Additionally, we do not account for energy loss during a hole interaction; different holes have unique energy losses. So, we do not assume a constant energy loss. A ball's state while moving consists of its planar position and velocity; we numerically integrate Penner's differential equations (Penner, 2002) until the ball stops or is captured in the hole. After a putt's simulation, the ball is either deemed captured or is stopped at a certain resting position. We define the "leave" of the golf ball as the distance from the resting position to the hole center. We have two simulation modes: hole-transparent and hole-interactive putts. Hole-transparent putts treat the hole as a pass-through point, so no putts can be holed, and the only outcome is a ball stopping. We use this to find the initial starting conditions that satisfy Dave Pelz's empirical targeting strategy (Section 4.2). Hole-interactive simulation includes the hole-interaction model, making it fit to simulate a golfer's actual putts to determine a strategy's expected strokes to hole out. For both modes, the only difference is in the handling of a hole event. Otherwise, all stopping and trajectory physics are identical. The simulator is completely deterministic; the only randomness comes from variance in the putt's starting conditions by incorporating a golfer's "skill" profile (Section 4.4).

3.2 Numerical integration

To numerically integrate the equations of motion, we followed Broadie & Shin's method of second-order Runge-Kutta numerical integration (Broadie and Shin, 2014). They justified the use of a second-order Runge-Kutta numerical method by overlaying the stopping error of the final position of first-order, second-order, and fourth-order Runge-Kutta methods (Broadie and Shin, 2014, Figure 13). They found "a noticeable improvement" from using a second-order method compared to using a first-order method, but no significant improvement from a fourth-order Runge-Kutta method compared to a second order method. For the main datasets, we used a numerical integration timestep of 0.02 seconds, and a timestep of 0.01 seconds for everything in our literature validation/comparison section. We justify the numerical integration timestep of 0.02 seconds by comparing our dataset results from using a timestep of 0.02 seconds to 0.01 seconds (Section 6.3).

The transition from a ball rolling on the green to entering the hole, as well as the stopping conditions, are all resolved by interpolating velocity and position between timesteps.

3.3 Green-to-hole transition

We determine the transition for the golf ball changing "state" from the green to the hole. Between integration timesteps, the ball moves a certain distance. The movement can situate the golf ball from outside the hole on the initial timestep to inside the hole on the next timestep. If we put these parameters as our initial parameters to the hole model, the ball is not sitting on the edge of the hole, leading to false outcomes. With a sufficiently large integration timestep, the ball could actually "skip" over the hole between timesteps. Therefore, we determine if the ball has entered hole motion by determining whether the segment between the ball's two timesteps crosses the hole rim, eliminating "skipping" the hole for practical timesteps. Furthermore, a segment crossing into the hole only enters hole motion if the entering velocity has a "radial" component that points towards the hole. After a ball exits the hole and returns to the green, it is placed on the hole lip. However, the ball's velocity has no inward radial component, meaning that no part of the ball would go into the hole. We then find where the ball intersects with the rim surface by determining the position where the segment intersects the hole. To ensure an accurate ball velocity entering the hole, we linearly interpolate the ball's entering velocity between the timesteps to the ball's position on the rim.

3.4 Stopping conditions

Finally, we determine our stopping criteria. Our numerical integration with fixed timesteps would actually never land on exactly zero velocity, because of a lack of infinite precision. Therefore, our thresholds for stopping are a rule for declaring a ball at rest, with the added test of whether that ball would stay at rest. To implement a satisfactory stopping condition, we first check whether the ball is at a sufficiently slow speed to consider stopping (0.005 m/s). If the ball is at an extremely low speed (below 0.005 m/s), the ball is considered holed if its center is inside the rim radius. Otherwise, we check if the slope is severe enough to re-accelerate a ball at rest here. We generalized Penner's one-sloped rollback threshold to include combinations of both slopes with a small angle approximation, $\sqrt{\tan^2\theta + \tan^2\phi} > \rho_g$ (Penner, 2002). If the slope is not sufficiently large, we stop the ball at this position. If the slope is sufficiently high to reaccelerate the ball, we do not immediately stop it, instead checking the ball's acceleration at a hypothetical zero velocity (so there is no friction force). If this acceleration is below our minimum acceleration threshold (below 0.01 m/s^2), we stop the ball. Otherwise, we probe with a unit velocity vector pointed straight downhill; now the acceleration function includes rolling friction opposing the motion. If the net downhill acceleration is still above our minimum acceleration threshold, then gravity is sufficiently strong to counteract the friction force, and we continue the simulation. Else, we mark the stopped position. It is important to note that because our minimum speed and acceleration thresholds are arbitrary constants, we do run a retrial of our results with different stopping thresholds, yielding

near identical results (Section 6.3). To avoid indefinite rolling, we set a maximum simulation time to 25 seconds. We found that this maximum simulation time was sufficient to capture all putts; no putts in our dataset were still rolling after 25 seconds for all simulations.

4 Study of Putting Strategy Improvements

We determine which putting strategies have lower expected strokes than the Pelz baseline. For an individual putt on a green (a scenario) with a unique distance, slope, and speed, we determine the starting conditions Dave Pelz would recommend. Then, we determine a grid of alternative strategies that are made by modifying his starting conditions. For every strategy (including the Pelz strategy), we simulate a golfer hitting the putt imperfectly many times by adding execution error to the strategy's intended starting conditions. We then average each strategy's expected strokes to hole out over each imperfect putt, and compare its strokes to hole out to the Pelz default strategy. We resimulate the best alternative strategy and the Pelz baseline to ensure lucky execution draws did not determine a winning strategy. We do this for two datasets, a short dataset (3.3–11.5 feet) and a long dataset (11.5–50 feet), each consisting of 32 unique scenarios. For all scenarios, we use four skill levels, consisting of one amateur skill profile (a 90-shooting amateur) and three professional skill profiles for the best, average, and worst tour pros.

4.1 Physical parameterization of putting scenarios

The experiment's inputs are determined by the number of scenarios s (unique greens) and the number of strategies per green compared to the Pelz baseline strategy. A targeting strategy T is defined as a golfer's intended starting conditions (v_0, ψ). We split the experiment into two datasets with 32 scenarios, each generated by a Sobol sequence (Sobol', 1967) around slope, stimp, and distance boundaries. The first dataset is for short putts ranging from 3.3–11.5 feet, and the other dataset is for lag putts ranging from 11.5–50 feet. For each scenario, we base our alternative strategies as set deviations from the Pelz baseline targeting strategy. We use our hole-transparent simulation model to find the starting conditions (v_0, ψ) of Pelz's baseline targeting strategy T_0. All strategies for a scenario are in the targeting pool $\mathcal{T} = \{T_0, T_1, \ldots, T_{152}\}$, where each alternative targeting strategy T_n $(n \neq 0)$ is a fixed speed and line offset from T_0's starting conditions (v_0, ψ). In total, for each scenario we determine $|\mathcal{T}| = 153$ targeting strategies (including 1 for T_0). Despite a golfer's attempted T, the actual simulated putt's starting conditions are scattered around T per a golfer's skill level (Section 4.4).

4.1.1 Scenario generation

A scenario s is defined as the putt's physical parameters, or could be thought of as a ball's green with a distance, slope, and green speed: (D, θ, ϕ, S). Every other environmental parameter (hole radius, ball mass, gravity) is fixed at the values in Table 1. We made two datasets, each a set of 32 scenarios. Each dataset varies specifically in the range of putting distance. The first

dataset we made was designed specifically for short putts, putts from 3.3–11.5 feet. The second dataset was designed for lag putts, with a distance range of 11.5–50 feet (3.5–15.24 m). Slopes (both θ and ϕ) were bounded between $\pm 2.5°$. Implementation of higher slopes than $\pm 2.5°$ for both directions, combined with a fast green, would never stop the ball, which would be unplayable (the ball would always roll off the green) and unrealistic. Green speeds ranged with a stimp from 9 to 13, effectively spanning everyday to championship greens. We generated the scenarios themselves using a Sobol sequence (Sobol', 1967; Joe and Kuo, 2008) over these slope, distance, and stimp boundaries (Table 3).

Short box				Lag			
Distance (ft)	θ (°)	ϕ (°)	Stimp	Distance (ft)	θ (°)	ϕ (°)	Stimp
6.40	1.77	1.29	12.78	38.00	-1.92	0.58	9.43
9.89	-1.08	-0.74	10.82	28.79	2.29	-0.40	12.67
7.97	0.28	0.35	11.39	19.33	-0.81	2.24	10.30
5.24	-1.94	-2.31	9.46	48.70	1.22	-2.41	11.54
3.92	0.89	-0.58	9.58	42.77	-0.12	-0.89	11.32
9.33	-1.25	2.39	11.52	13.39	0.37	0.71	10.59
10.48	2.36	-1.53	10.22	22.85	-1.53	-1.30	12.44
5.83	-0.40	0.82	12.13	32.06	1.75	1.47	9.71
5.97	0.44	-1.86	11.77	33.28	-0.97	-1.72	10.78
11.38	-2.41	1.11	9.81	24.06	0.75	1.60	11.01
8.43	1.30	-0.25	12.38	14.60	-2.40	-1.08	9.89
3.78	-0.93	2.10	10.45	43.98	2.15	1.23	12.13
4.35	1.91	0.01	10.57	47.51	-1.69	2.03	12.98
7.84	-0.25	-2.01	12.52	18.13	1.28	-1.91	9.25
10.02	1.05	1.63	9.21	27.60	-0.57	0.14	11.84
7.29	-1.74	-1.03	11.13	36.81	0.20	-0.28	10.12
6.96	1.12	1.05	10.01	35.68	-0.46	1.81	12.31
10.38	-1.65	-1.61	12.09	26.46	0.01	-1.67	9.54
7.44	2.14	1.99	9.65	17.01	-1.80	1.01	11.43
4.77	0.00	-0.03	11.70	46.38	1.47	-1.13	10.66
3.30	1.52	-2.08	11.34	45.11	-2.20	-2.12	10.45
8.89	-0.69	0.27	9.27	15.73	2.02	1.98	11.71
11.08	0.51	-1.12	12.95	25.19	-1.17	-0.08	9.32
6.23	-2.33	1.84	10.90	34.41	0.88	0.20	12.58
5.34	2.30	-0.80	9.00	30.79	-1.35	-0.49	11.90
10.94	-0.48	1.55	11.09	21.57	1.64	0.33	10.14
9.03	0.66	-2.41	10.64	12.11	-0.30	-2.31	12.77
4.18	-1.50	0.56	12.69	41.49	0.48	2.49	9.00
4.91	0.05	2.33	12.33	49.98	-0.68	0.80	9.86
8.33	-2.19	-0.33	10.25	20.61	1.02	-0.64	12.12
9.50	1.70	0.72	11.94	30.07	-2.04	1.38	10.97
6.82	-1.16	-1.31	9.90	39.28	2.49	-1.56	11.24

Table 3: All environmental parameters for the 32 scenarios of the short-putt and lag datasets. The two datasets are both results of independent Sobol draws. All four execution profiles (3 touring profiles, one amateur profile) use the same scenarios in the table. θ is the green's horizontal slope angle (a positive value indicating a right-to-left putt), ϕ is the green's vertical slope (a positive value indicating an uphill putt), and Stimp represents the speed of the green.

4.2 Computing the Pelz baseline starting conditions

For each scenario we determine the starting (v_0, ψ) that satisfies the Pelz targeting strategy T_0. We define a hole-transparent green with identical slope, stimp, and length to the scenario s. To find T_0 we made an objective function that calculates how "well" a proposed starting condition (v_0, ψ)'s trajectory matches Pelz's requirements. Pelz's strategy is based on two conditions: on a hole-transparent green, the ball's optimal trajectory first must pass through the center of the hole and then stop 17 inches past the hole (Pelz and Frank, 2000, p. 190). Then, we minimized the objective function to find Pelz's optimal (v_0, ψ).

The objective function is made from two squared components, each associated with one of Pelz's two requirements. The first component ensures the putt's trajectory goes through the center of the hole. Across the ball's entire trajectory, we compute $d_{\min}$, the closest approach to the hole center:

$$d_{\min} = \min_{0 \le t \le t_{\text{end}}} \sqrt{x(t)^2 + \big(y(t) - D\big)^2}. \tag{14}$$

For our simulation, $(x(t), y(t))$ is the ball's position in the hole-transparent trajectory, the transparent hole center is at $(0, D)$, and t_{end} is the ball's total simulation time. Because we numerically integrate $(x(t), y(t))$ at discrete timesteps, the minimum is computed over the straight segments between integration points to ensure the closest approach is captured even when the discrete timesteps "skip" the hole.

The objective function penalizes both a miss of the hole's center and a miss of the target distance past the hole. Our "optimal" roll-out distance is when the ball's final y coordinate $y_{\text{end}} = y(t_{\text{end}})$ is exactly D plus the rollout target $\Delta = 0.4318$ m (= 17 inches exactly) past the hole. The full objective is as follows:

$$\mathcal{L}(v_0, \psi) = \left(\frac{d_{\min}}{2R}\right)^2 + \left(\frac{y_{\text{end}} - D - \Delta}{\Delta}\right)^2. \tag{15}$$

We normalize each error so $d_{\min}$ and stopping position are optimized equally by the objective function. If we did not normalize our objective function, a 3-inch discrepancy in closest distance to the hole would look equal to a 3-inch discrepancy in stopping point. However, a 3-inch $d_{\min}$ would mean the ball never touched the hole, while a 3-inch difference between y_{end} and Δ would only be a minor trajectory difference. So we normalize our first term by the diameter of the hole $2R$, and the second in units of the roll-out target Δ. We minimize the objective function to find T_0 by optimizing the (v_0, ψ) with an L-BFGS-B search initialized from a simple heuristic warm start (v_0, ψ):

$$T_0 = \arg\min_{(v_0, \psi)} \mathcal{L}(v_0, \psi). \tag{16}$$

We analyzed the effects of using an elaborate Gaussian Process Regression model to determine initial (v_0, ψ) for the L-BFGS-B algorithm on our final (v_0, ψ), to no change. Therefore, we use our

simple heuristic instead. To ensure the convergence of our search algorithm, we accept the starting condition when $\mathcal{L}(T_0) < 10^{-5}$. All scenarios for every dataset converged on the first run of the L-BFGS-B algorithm. Pelz's baseline targeting strategy does not depend on skill, so the single T_0 strategy is used for each scenario, regardless of skill profile.

4.3 Strategy construction

After the initial computation of the Pelz baseline targeting strategy, we produce alternate strategies as fixed speed and line offsets around the Pelz baseline. All strategies are direct offsets of the Pelz baseline; for example, a sample strategy would be: $T_n = T_0 + (\Delta v_n, \Delta\psi_n)$. A negative speed offset would result in a more conservative targeting strategy, and a negative line offset would result in aiming more "left". Likewise, a positive speed offset would be a more aggressive strategy, and a positive line offset would be equivalent to aiming "right". The alternative strategies form a single uniform lattice of offsets around the Pelz baseline; we take the baseline (v_0, ψ) and apply every combination of 9 speed offsets $\mathcal{T}_v = \{0, \pm 0.05, \pm 0.10, \pm 0.15, \pm 0.20\}$ m/s and 17 line offsets $\mathcal{T}_\psi = \{0, \pm 0.5, \pm 1.0, \pm 1.5, \pm 2.0, \pm 2.5, \pm 3.0, \pm 3.5, \pm 4.0\}^\circ$ (uniform 0.5° steps). Therefore, all strategies are in the set $\mathcal{T} := \left\{T_0 + (\Delta v, \Delta\psi) : \Delta v \in \mathcal{T}_v,\ \Delta\psi \in \mathcal{T}_\psi\right\}$, making for $|\mathcal{T}| = |\mathcal{T}_v| \times |\mathcal{T}_\psi| = 9 \times 17 = 153$ strategies. The Pelz baseline is equivalent to the offset $(0, 0)$ in $\mathcal{T}$, so there are a total of 152 alternate strategies, $\mathcal{T} \setminus \{T_0\}$. We include the Pelz baseline T_0 in the strategy set as a baseline for other strategies to compare against. The strategy "grid" around the Pelz baseline T_0 is a fixed, symmetric design decided before any strategy was evaluated, and the grid is used for every scenario, dataset, and skill profile.

4.4 Models for execution error and green-reading error

A golfer does not perfectly execute a strategy's starting conditions every time. So given an input targeting strategy $T = (v_0, \psi)$ and an execution profile (a golfer's "skill" level), we find the executed launch conditions $\tilde{T} = (\tilde{v}_0, \tilde{\psi})$ that a simulator would actually run for a single simulated putt. We implement Broadie & Shin's execution profile for three types of touring professionals (best tour pros, average tour pros, and worst tour pros) (Broadie and Shin, 2014, Fig. 7) and Bansal & Broadie's execution profiles for an ordinary 90-shooting amateur golfer (Table 4) (Bansal and Broadie, 2008, §4). For all skill profiles, errors in line are modeled as a Gaussian distribution about the strategy's starting ψ. A golfer's simulated starting line $\tilde{\psi}$ is defined as

$$\tilde{\psi} = \psi + \sigma_\psi Z_\psi, \qquad Z_\psi \sim \mathcal{N}(0, 1), \tag{17}$$

σ_ψ is a golfer's standard deviation in starting line, which for all four execution profiles is a fixed constant independent of v_0. Per both Broadie & Shin's and Bansal & Broadie's execution models, $\tilde{v}_0^2$ is drawn from a normal distribution about the intended v_0^2, with the standard deviation of the

squared speed being a piecewise function $\sigma_{v^2}(v_0^2)$. Therefore, $\tilde{v}_0$ is determined as follows:

$$\tilde{v}_0^2 = v_0^2 + \sigma_{v^2}(v_0^2)\, Z_v, \qquad Z_v \sim \mathcal{N}(0,1), \qquad \tilde{v}_0 = \sqrt{\max(\tilde{v}_0^2, 0)}. \tag{18}$$

For the best, average and worst tour pros' speed execution profile, we use Broadie & Shin's $\sigma_{v^2}(v_0^2)$ piecewise-linear function with 3 breakpoints. Broadie & Shin publish velocity error values ($r_k = \sigma_{v^2}(v_k^2)/v_k^2$) associated with each breakpoint $v_0^2 = (1, 4, 9)$ m^2/s^2. To find the standard deviation (y_1, y_2, y_3) associated with each breakpoint, we multiply Broadie & Shin's error values for each execution profile by $v_0^2 = (1, 4, 9)$ m^2/s^2. Broadie & Shin's piecewise function for a starting speed's standard deviation is therefore as follows:

$$\sigma_{v^2}(v_0^2) = \max\big(0,\ \ell(v_0^2)\big), \qquad \ell(v_0^2) = \begin{cases} y_1 + \dfrac{y_2 - y_1}{4 - 1}\left(v_0^2 - 1\right), & v_0^2 < 4 \\ y_2 + \dfrac{y_3 - y_2}{9 - 4}\left(v_0^2 - 4\right), & v_0^2 \geq 4 \end{cases} \tag{19}$$

with v_0^2 in m^2/s^2, anchor points (v_k^2, y_k) at $v_k = 1, 2, 3$ m/s (so $v_k^2 = 1, 4, 9$), and $y_k = r_k v_k^2$, where r_k is Broadie & Shin's printed error value for a breakpoint. Broadie & Shin extrapolate $\sigma_{v^2}(v_0^2)$ for $v_0^2 < 1$ and $v_0^2 > 9$ by using the same linear segment as for $1 \leq v_0^2 < 4$ and $4 \leq v_0^2 \leq 9$, respectively. As a result, the piecewise function has two branches, even with three distinct breakpoints.

Execution profile	σ_ψ (deg)	$\sigma_{v^2}(v_0^2)$ breakpoints	σ_g (deg)	Source
Tour, best	0.626	$r_{1,2,3} = 0.019, 0.035, 0.044$	0.379	Broadie and Shin, 2014
Tour, average	0.791	$r_{1,2,3} = 0.034, 0.041, 0.043$	0.400	Broadie and Shin, 2014
Tour, worst	0.903	$r_{1,2,3} = 0.046, 0.058, 0.055$	0.435	Broadie and Shin, 2014
Amateur (90-shooter)	1.5	$\beta_{0,1,2} = 0.060, 0.078, 0.085$; $v_\beta = 3.052$ m/s	0.25	Bansal and Broadie, 2008

Table 4: Constants used for a given skill level's execution profile. For generation of main datasets, we do not incorporate green reading error, σ_g. The $\sigma_{v^2}(v_0^2)$ breakpoints for the tour profiles are Broadie & Shin's relative speed error values r_k at $v_k = 1, 2, 3$ m/s, and the $\sigma_{v^2}(v_0^2)$ breakpoints for the amateur profile are Bansal & Broadie's speed error values at $v_\beta = 3.0519$ m/s.

Broadie & Shin's speed standard deviation function (Broadie and Shin, 2014) produces negative standard deviation for sufficiently low values of v_0^2 ($v_0 < 0.508$ m/s for the worst tour pros, $v_0 < 0.464$ m/s for the average, and $v_0 < 0.727$ m/s for the best); below that value of v_0^2 we floor standard deviation to 0, so tour pace is deterministic for low paces.

For the 90-shooting amateur, we use Bansal & Broadie's piecewise function (Bansal and Broadie, 2008) for standard deviation in v_0^2, which relies on a single breakpoint velocity v_β instead of three like Broadie & Shin:

$$\sigma_{v^2}(v_0^2) = \begin{cases} \beta_2 v_\beta^2 - \beta_0(v_\beta^2 - v_0^2), & v_0^2 \leq v_\beta^2 \\ \beta_2 v_\beta^2 + \beta_1(v_0^2 - v_\beta^2), & v_0^2 > v_\beta^2, \end{cases} \tag{20}$$

with $0 \leq \beta_0 \leq \beta_1 \leq \beta_2$. Bansal & Broadie describe their breakpoint v_β as being the starting velocity that rolls out 25 feet on a flat, 9 stimp green. We calculate $v_\beta = 3.0519$ m/s using our flat-green rollout relation (Eq. (1)). The standard deviation $\sigma_{v^2}(v_0^2)$ never goes to zero, even for

extremely slow starting speeds. Therefore, for soft putts $\tilde{v}_0^2$ can become a negative value. So, we clamp negative $\tilde{v}_0^2$ to zero in Eq. (18).

We use Broadie & Shin's touring professionals model and Bansal & Broadie's 90-shooting amateur model to account for a golfer's misread of the green (Broadie and Shin, 2014; Bansal and Broadie, 2008). The green reading model accounts for variations in the targeting strategy's chosen (v_0, ψ) because of a golfer's misread of slope. $T = (v_0, \psi)$ is initially planned on a golfer's initial perceived slope $(\hat{\theta}, \hat{\phi})$ (true slope (θ, ϕ) plus reading error). However, the actual trajectory model runs on the true green, with the initial (v_0, ψ) the golfer derives from the misread green $(\hat{\theta}, \hat{\phi})$. Both Broadie & Shin and Bansal & Broadie model the golfer's perceived slope based on the scenario's true slope (θ, ϕ), and the standard deviation of a golfer's misread, σ_g:

$$(\hat{\theta}, \hat{\phi}) = (\theta, \phi) + \varepsilon\,(\cos\omega,\ \sin\omega), \qquad \varepsilon \sim \mathcal{N}(0, \sigma_g^2), \quad \omega \sim \mathcal{U}(0, 2\pi). \tag{21}$$

with σ_g being a golfer's constant direction error, used by all skill profiles. The slope error model for touring pros is based on the first putt σ_g values from Broadie & Shin. For amateurs, the σ_g values are derived from Bansal & Broadie's amateur golfer misread model (Bansal and Broadie, 2008, Eq. (4)).

All datasets in Section 6 are purely generated on a correct read (perceived green is set equal to the true green, $(\hat{\theta}, \hat{\phi}) = (\theta, \phi)$) as a design choice; the incorporation of green reading error into our main dataset would raise our variance in expected strokes sharply. Therefore, every difference between targeting strategies is attributable to execution error alone, conditional on a correct read. Nonetheless, reading error is introduced for winning targeting strategies as a separate, measurable effect. So, we sample all winning strategies and the Pelz baseline with green reading error and measure the results on expected strokes (Section 6.3).

4.5 Scoring: expected strokes and the continuation model $N(d)$

After the simulation of a putt, the ball either is captured in the hole or stops at a distance d away from the hole. We convert these physical values into the expected number of putts taken in order to rank strategies; the higher a strategy's expected number of putts, the worse it is. The expected putts taken for a given simulated putt is:

$$\text{strokes} = \begin{cases} 1, & \text{holed,} \\ 1 + N(d), & \text{stopped, leave distance } d. \end{cases} \tag{22}$$

If the putt is marked as holed, then the expected number of putts taken is trivially 1. However, if the ball misses the hole, we need to develop a method to accurately estimate the number of putts taken from that leave point. If the putt missed, the expected strokes formula would therefore be one (one stroke for the initial putt) plus the expected number of strokes to hole out from that leave position, $N(d)$ (Broadie, 2012). One way we could calculate $N(d)$ is to recursively use our execution model from our putt's final position to calculate the expected number of putts from there. In that

case, we would have to assume a targeting strategy T, and then simulate a golfer's "skill" by using our execution noise profile. After this, we would have to evaluate this miss's expected strokes with a large number of inner execution error samples, averaging the expected strokes to hole out from all of those simulated putts (coming from the execution error modeling a golfer's skill noise in their initial (v_0, ψ)). If a simulation in the second putt misses the hole, we would simulate this process recursively until every single putt is holed, costing an exceptionally large amount of compute power.

Instead, we approximate the expected "continuation" putts for the ball to be holed out from the leave distance from real-world data, using a different model for PGA Tour (Tour worst, best, and average profiles) and amateur putters. From our hole's coordinates $(0, D)$ and a ball's resting position $(x(t_{\text{end}}), y(t_{\text{end}}))$, a putt's leave distance is defined as:

$$d = \sqrt{x(t_{\text{end}})^2 + \big(y(t_{\text{end}}) - D\big)^2}. \tag{23}$$

For the three PGA Tour player execution profiles, we use the PGA Tour's table that has published expected strokes to hole out data as a function of putt distance from 1–100 feet (PGA Tour, 2011), as well as the trivial datapoint $N(0 \text{ feet}) = 1.00$ (if the putt's starting position is 0 feet or in the hole, it always counts as 1 stroke from that position). We fit their discrete datapoints to a smooth, continuous $N(d)$. We fit a PCHIP (piecewise cubic Hermite interpolating polynomial; Fritsch and Carlson, 1980; Fritsch and Butland, 1984) log-distance continuation model from PGA Tour expected strokes data: For leave distances in between 0 and 1 feet, we linearly interpolate $N(d)$ between the trivial datapoint $N(0 \text{ feet}) = 1.00$ and the PGA Tour's datapoint $N(1 \text{ feet}) = 1.001$. The full piecewise continuation model is shown:

$$N(d) = \begin{cases} 1.00 + d\,(1.001 - 1.00), & 0 \le d < 1 \text{ ft}, \\ \text{PCHIP}\big(\log d;\ \{(\log d_i, N_i)\}_{i=1}^{100}\big), & 1 \text{ ft} \le d \le 100 \text{ ft}, \\ 2.382, & d > 100 \text{ ft}, \end{cases} \tag{24}$$

where $\{(d_i, N_i)\}_{i=1}^{100}$ are the PGA Tour's published expected-strokes values at $d_i = 1, 2, \ldots, 100$ feet, and $2.382 = N(100 \text{ ft})$ is held constant for $d > 100$ feet.

For the amateur model, we fit our continuation function directly based on the simulated expected putts versus distance chart for a 90-shooter (standard amateur) given in Bansal and Broadie (2008) that covers 2–50 feet. Like the professional continuation model, we also use a PCHIP log-distance continuation model, with putts in between 0 and 2 feet being linearly interpolated to the $N(0 \text{ feet}) = 1.00$ anchor and the datapoint $N(2 \text{ feet}) = 1.08$:

$$N(d) = \begin{cases} 1.00 + \dfrac{d}{2}\,(1.08 - 1.00), & 0 \le d < 2 \text{ ft}, \\ \text{PCHIP}\big(\log d;\ \{(\log d_i, N_i)\}_{i=1}^{15}\big), & 2 \text{ ft} \le d \le 50 \text{ ft}, \\ 2.39, & d > 50 \text{ ft}, \end{cases} \tag{25}$$

where $\{(d_i, N_i)\}_{i=1}^{15}$ are the amateur (90-shooter), standard-hole $N^*(d)$ values from Bansal and Broadie (2008, Table 1), and $2.39 = N(50 \text{ ft})$ is held constant for $d > 50$ feet.

We do not account for additional factors like stimp and slope, which are factors in determining expected strokes to hole out. For example, uphill putts from the same distance are generally considered "easier" than downhill putts from the same distance, because downhill putts have a smaller margin for error. Additionally, the smooth function $N(d)$ for the amateur model is based on a table with sparse data points. Because we store every simulation's unique physical outcome, we do not resimulate and simply re-evaluate the expected strokes for all putts' physical outcomes with continuation models based on varying fit metrics, and entirely different continuation models to check if our results stand (Section 6.3).

4.6 Monte Carlo evaluation and validation

For both the lag (11.5–50 feet) and the short (3.3–11.5 feet) dataset, we have a total of 32 scenarios s, with 153 targeting strategies T per scenario. Scenarios and targeting strategies are equivalent across all four execution profiles. For all s, we estimate each of the 153 targeting strategies' $T \in \mathcal{T}$'s expected strokes to hole out, and compare them to the Pelz baseline to determine if a targeting strategy is "better" than the Pelz baseline. We draw $K = 1000$ execution starting conditions $\tilde{T} = (\tilde{v}_0, \tilde{\psi})$, and we simulate and determine the expected strokes to hole out for each. We average the expected strokes to hole out for all K execution draws to determine expected putts $\hat{E}(s, T)$ to hole out for each targeting strategy. For all targeting strategies and execution profiles, we use identical z-scores to determine $(\tilde{v}_0, \tilde{\psi})$. Because we use "paired" execution draws, we determine the targeting strategy's expected strokes gained on the Pelz baseline for each execution draw k, $\text{SG}_k(s, T)$. Then, we average $\text{SG}_k(s, T)$ over all K execution draws to determine the strokes gained from that scenario on the Pelz baseline, $\widehat{\text{SG}}(s, T)$. Per scenario, we select the alternate targeting strategy that gains the most strokes on the Pelz baseline, $T^*(s) = \arg\max_{T \in \mathcal{T} \setminus \{T_0\}} \widehat{\text{SG}}(s, T)$, and compare it to the Pelz baseline again under 4000 execution draws to ensure a targeting strategy doesn't beat Pelz baseline purely on "favorable" execution draws (Kim and Nelson, 2006). Finally, we ensure the 95% confidence interval of expected strokes gained from the alternate targeting strategy beats the Pelz baseline, and run a Benjamini–Hochberg false discovery rate correction (Benjamini and Hochberg, 1995), set at 2.5%. We repeat this process across all four execution profiles (best, worst, and average touring professionals, and a 90-shooting amateur), keeping the same scenarios and targeting strategies (the only difference is the execution profile itself) to determine how the best targeting strategy changes in importance per skill level.

4.6.1 Initial Monte Carlo evaluation

To account for the player's physical execution error, we simulate each targeting strategy with 1000 unique execution draws over the strategy's (v_0, ψ), for both the lag (11.5–50 feet) and the short (3.3–11.5 feet) dataset. Using the hole-interactive simulator, we simulate the execution draws for all four execution profiles (best tour pros, average tour pros, worst tour pros, and a 90-shooting

amateur) and determine average expected strokes $\hat{E}(s,T)$ for each targeting strategy. For every execution draw, we define a player's executed startline conditions $\tilde{T} = (\tilde{v}_0, \tilde{\psi})$ as the intended strategy $T = (v_0, \psi)$ with an execution "offset" (Eqs. (17) and (18)), whose magnitude depends on the z-scores (Z_ψ, Z_v) drawn from a normal distribution $\mathcal{N}(0,1)$. Therefore, we generate 1000 z-score offsets from the normal distribution:

$$\mathcal{Z} = \left\{(Z_{\psi,k},\, Z_{v,k})\right\}_{k=1}^{K}, \qquad Z_{\psi,k},\, Z_{v,k} \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0,1). \tag{26}$$

We use the same $\mathcal{Z}$ for all targeting strategies in the scenario, making each noise draw equivalent in deviation from the mean, regardless of the targeting strategy (so strategies have "paired" offsets). Therefore, instead of comparing the mean expected strokes $\hat{E}(s,T)$ to the Pelz baseline ($\hat{E}(s,T) = \frac{1}{K}\sum_{k=1}^{K} E_k(s,T)$), we compare the strokes gained by the alternate targeting strategy each individual execution draw:

$$\mathrm{SG}_k(s,T) = E_k(s,T_0) - E_k(s,T). \tag{27}$$

Averaging $\mathrm{SG}_k(s,T)$ determines a targeting strategy's strokes gained on the Pelz baseline:

$$\widehat{\mathrm{SG}}(s,T) = \frac{1}{K}\sum_{k=1}^{K} \mathrm{SG}_k(s,T) = \hat{E}(s,T_0) - \hat{E}(s,T). \tag{28}$$

A positive value would indicate that a targeting strategy "gained" strokes on the Pelz baseline, making a golfer take fewer strokes on average. For a fixed execution profile, $E_k(s,T)$ is the expected strokes taken (Eq. (22)) on execution draw k of targeting strategy T in scenario s. By comparing paired execution draws' expected strokes $E_k(s,T)$ instead of the mean expected strokes $\hat{E}(s,T)$, we reduce the standard error of $\widehat{\mathrm{SG}}(s,T)$. Finally, the best "alternative" targeting strategy maximizes strokes gained on the Pelz baseline:

$$T^*(s) = \arg\max_{T \in \mathcal{T}\setminus\{T_0\}} \widehat{\mathrm{SG}}(s,T). \tag{29}$$

The initially chosen best targeting strategy $T^*(s)$ does not have to necessarily beat the Pelz baseline. We define the null hypothesis for a targeting strategy beating the Pelz baseline as an alternate targeting strategy T not having lower expected strokes than the Pelz baseline. To evaluate the hypothesis, we construct the 95% confidence interval for $\widehat{\mathrm{SG}}(s,T)$ by using a paired z-test (the large number of inner samples makes it near identical to a t-test). An initial winning targeting strategy is one whose $\widehat{\mathrm{SG}}(s,T) > 0$ for $\widehat{\mathrm{SG}}(s,T)$'s entire 95% confidence interval. Because one "tail" of the 95% confidence interval would always beat the Pelz baseline, the 95% confidence interval is equivalent to a p-value of less than 2.5%. Targeting strategies "win" not only because of their optimized strategy, but because the execution draws (even if identical across all targeting strategies) favored them. To mitigate this "winner's" curse (Capen et al., 1971; Smith and Winkler, 2006), we use fresh Monte Carlo revalidation for winning targeting strategies.

4.6.2 Fresh Monte Carlo revalidation and determination of winning strategies

Currently, we run each of the 153 targeting strategies per scenario with 1000 execution profile sample offsets. The targeting strategies beating the Pelz baseline are biased by favorable execution draws (winner's curse; Capen et al., 1971; Smith and Winkler, 2006).

By making every targeting strategy use the same z-score offsets, we remove the possibility that one targeting strategy could have gotten "more favorable" draws compared to other targeting strategies, because all draws are equivalent. However, the initial winning targeting strategies still benefit from the "equal" draws more than others. The initial winning targeting strategy still has favorable execution draws relative to its own targeting strategy. Take a scenario where a more aggressive targeting strategy (a targeting strategy with a high initial velocity compared to the Pelz baseline) is actually best. If the execution noise suggests numerous high-velocity offsets, conservative strategies could look better than they actually are; the high-velocity offsets would account for the conservative initial strategy. Likewise, the optimal aggressive strategy could be overlooked, because with the high-velocity offset execution draws, the strategy would get "too" aggressive, reducing the strategy's perceived effectiveness.

For all scenarios, the best initial targeting strategy $T^*(s)$ and the Pelz baseline gets resimulated with $K_{\text{fresh}} = 4000$ new execution draws generated from a fresh seed (reusing the same execution draws means that the draws would still account for making the targeting strategy favorable) to reduce selection optimism. After resimulation, the winning targeting strategy's strokes gained on the Pelz baseline, $\widehat{\text{SG}}(s, T^*(s))$ is re-evaluated using the same paired execution draws method used initially to determine our final strokes gained on the Pelz baseline with fresh execution draws, $\widehat{\text{SG}}^{\text{F}}(s, T^*(s))$. We determine alternate targeting strategies' mean strokes gained for the entire dataset by averaging $\widehat{\text{SG}}^{\text{F}}(s, T^*(s))$ across all 32 proposed alternate targeting strategies, even those performing worse than the Pelz baseline. We re-evaluate if the best initial targeting strategy $T^*(s)$ "beats" the Pelz baseline with the same evaluation metric used for determining initial winning targeting strategies: determining if $\widehat{\text{SG}}^{\text{F}}(s, T^*(s)) > 0$ for $\widehat{\text{SG}}^{\text{F}}(s, T^*(s))$'s entire 95% confidence interval by using a paired z-test to calculate the p-value associated with the targeting strategy. Therefore, we tested the null hypothesis for all scenarios in two stages: In the first stage, we run the null hypothesis against all scenarios, determining which targeting strategies initially beat the Pelz baseline. For the second stage, we resample the initial best strategies for every scenario with fresh-sample revalidation, checking whether the best strategies now beat the Pelz baseline throughout the entire confidence interval and recalculating the p-values for each strategy. If a scenario's initial winning strategy does not beat the Pelz baseline with the fresh execution draws, we do not try the runner-up or any subsequent initial strategy on the larger K_{fresh} draws. If we decided to run different strategies until we find a strategy that passes the 95% confidence interval test, our 2.5% ratio on faulty strategies passing would be inflated. In a scenario where the Pelz baseline is truly the best strategy, each candidate has at worst a 2.5% chance of beating the Pelz baseline because of the 95% confidence interval test we chose. If we keep iterating through n strategies, the chance that something passes is $1 - (1 - 0.025)^n$ — roughly 22% by the tenth strategy. If there are enough

strategies, the probability a certain strategy passes on a scenario where the Pelz baseline is the best strategy nears 100%. The other alternative is to resample all strategies with the fresh execution draws. Not only does this introduce the first problem of falsely passing a scenario where the Pelz baseline is best, but it also reintroduces winner's curse, as we are reranking strategies based on noisy execution draws, and the best "reranked" strategy will always be partially influenced by the execution draws. So, we only test the initial best strategy on fresh draws.

Because we are testing if a large number of scenarios beat the Pelz baseline by calculating their 95% intervals (and thereby their p-values), each strategy's entire 95% confidence interval beating the Pelz baseline is not enough; we have to control for false discovery rate across all tests. We also apply a Benjamini–Hochberg false discovery rate correction (Benjamini and Hochberg, 1995) at 2.5% to account for multiple hypothesis testing with the large number of scenarios. The scenarios whose strategies survive the Benjamini–Hochberg false discovery rate correction and also have $\widehat{\mathrm{SG}}^{\mathrm{F}}(s, T^*(s))$'s 95% confidence interval entirely beating the Pelz baseline are considered winning scenarios (and their associated strategies are also called winning strategies).

5 Validation of Putting Simulator

Our simulator implements a "hybrid" between different models (Penner for numerical differential equations, Hogan & Antali hole capture) that has not been implemented before (Penner, 2002; Hogan and Antali, 2025). Therefore, we split our validation into two parts: verifying our model successfully replicates the source equations we used to make the custom simulator, and external validation of our full custom simulator against published literature figures and empirical data.

5.1 Verification against source models

By reproducing printed values from source papers, we show the faithful implementation of their source equations. For strict tests, we use their printed equations and values. To test our simulator's accuracy for our trajectory models, we recreated Penner's launch speeds for 10 foot putts (2.37/3.05/1.36 m/s) at various slopes (level/uphill/downhill), reproducing Penner's stated final values to within 2 cm of his final position (Penner, 2002, Fig. 3). For our capture model, we first recreated Hogan & Antali's printed trajectories, as well as the region map including the separatrix, with their strict turnaround angles and their start conditions. Furthermore, we recreated their boundary intersections for their capture boundaries within 0.002 of their stated impact parameter ratio and entry velocity (Hogan and Antali, 2025, Figs. 5 and 7). The verification tests so far verify components in isolation by testing for hole and rolling behavior apart from each other. The entire composed simulator is tested with external validation over make-rate regions and empirical data.

5.2 External validation

We first reimplemented Broadie & Shin's green-reading experiment (Broadie and Shin, 2014). Their experiment measured the make percentage for downhill, sidehill, and uphill putts of the same

length and slope magnitude with green-reading error switched on and off. We replicate the fact that when the golfer perfectly reads the green, downhill putts are made more than uphill putts of similar length. We also reproduce that when Broadie & Shin accounted for the fact that a golfer does misread the green, the narrative shifts and uphill putts have a higher one-putt percentage compared to downhill putts. We used their paper-stated values: their short-putt distance, slope, and execution noise values. Our target strategy to replicate these figures used the stated past-the-hole baseline strategy (initial parameters passing through the hole and stopping 1.5 feet past the hole). The three categories for uphill, sidehill, and downhill putts are approach angles relative to the fall line (the straight line on a sloped green where if a ball is putted along the line, the ball will not break), which we replicate per their conventions. Per read, we draw one perceived green (one reading error), and rotate the ball around the hole using their fall-line angle conventions (for uphill, sidehill, and downhill putts), making a reading execution draw hit all three putts at the same time. We then solve the aim/speed target on the perceived green, running our putts on the real green with execution noise. We show our simulated probabilities of holing out plotted against putt type (uphill, downhill, sidehill) with and without green-reading error, compared to Broadie & Shin's green-reading experiment in Fig. 1.

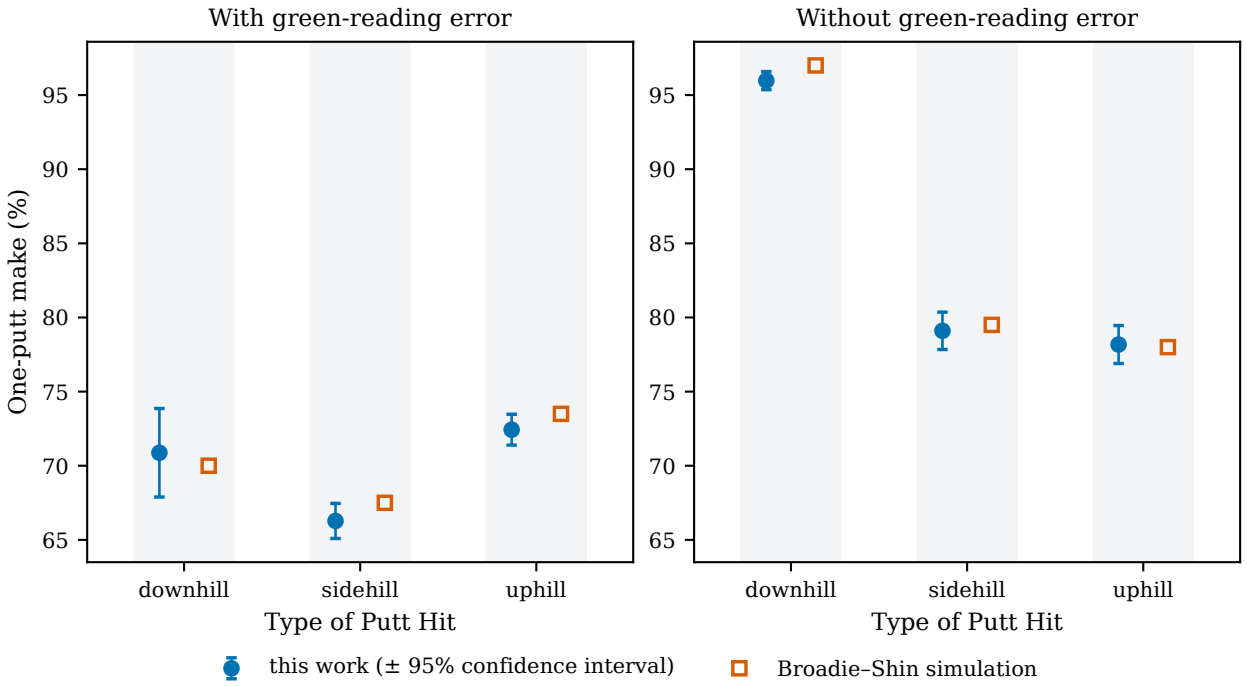


Figure 1: Simulated probabilities of holing out plotted for different putt types (uphill, downhill, sidehill) with and without green-reading error, compared to Broadie & Shin's green-reading experiment (Broadie and Shin, 2014). The putt was simulated on a green with a 2.3 degree slope and a stimp of 11, with the ball starting 5 feet away from the hole. The bars represent the 95% confidence intervals of our simulated values.

Our second re-implementation was against Fearing's empirical model of PGA putting (Fearing

et al., 2011). Built from empirical PGA ShotLink data (the tour's shot-tracking system), Fearing et al. fit a curve which gives the probability of holing a putt based on distance. However, real empirical make percentages are based on many unique greens and players. With ShotLink data unavailable to the public, we used the same green-reading and execution error parameters from Broadie & Shin (Broadie and Shin, 2014) to simulate putts on a single green. The green had slope and speed parameters typical of an average PGA Tour green: We used a slope of 1.5 degrees and a stimp of 11 (Bansal and Broadie, 2008). The ball was placed from 3-20 feet in foot increments (making for 18 unique distances) covering the short-to-medium putts range. For each distance, we used 5 approach directions relative to the fall line (0°, 45°, 90°, 135°, and 180°). We used the tour-average execution error and green-reading error from Broadie & Shin with the initial parameters passing through the hole and stopping 1.5 feet past the hole as the aiming strategy. The make rate at each distance would be the average over each of the different approach directions (Fig. 2).

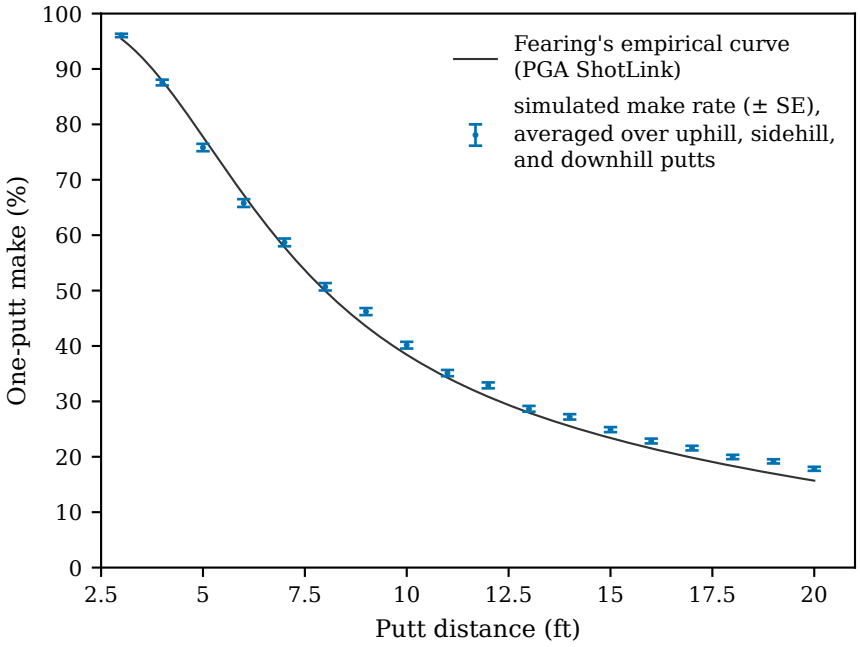


Figure 2: Make-rate percentage versus distance, comparing Fearing's empirically-based make-rate curve (Fearing et al., 2011) and our simulator-generated points. To generate our points, we choose one moderately-sloped green for all distances, simulating many putts around that hole with tour-average green-reading and execution error, and finding the average make-rate.

Finally, we reimplemented Penner's hole-out regions for putts (Penner, 2002). Using their stated green conditions, we swept over a grid of initial launch speeds and launch angles, checking if the putts are captured by the hole. That capture region is called the "capture window". We compare our capture windows (Fig. 3) against the ones they publish (Penner, 2002, Fig. 13).

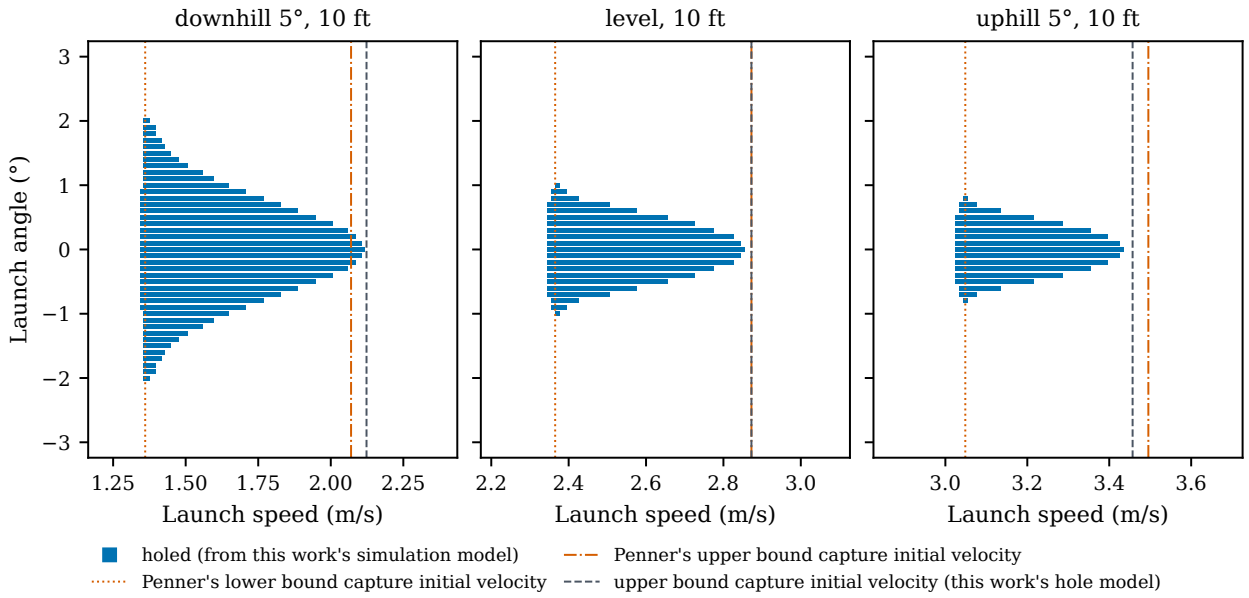


Figure 3: Simulated hole-out starting conditions for putts, with launch direction (how much break is played) on the y-axis and initial velocity on the x-axis. Each graph reproduces Penner's capture-regions for that specific putt, and are from Penner's "The physics of putting" Fig. 13 (Penner, 2002). Blue regions represent our hole-out regions, while white regions are swept and represent missed putts. The dotted orange line represents Penner's minimum hole-out velocity, and the dashed-dotted orange line represents their maximum hole-out velocity for the given putt. Our maximum hole-out velocity is defined as a vertical dashed gray line.

6 Results and Discussion

6.1 Strategy gains for short putts

We study different putting strategies for short-range putts. The "short-putt" dataset consists of 32 scenarios. A scenario s is defined as a putting situation consisting of the distance the ball is from the hole (D), the green's slope (uphill slope, ϕ and sidehill slope, θ), and the green's speed (S). In these 32 scenarios, the slopes, speeds, and distances vary between $\pm 2.5°$, 9–13 stimp, and 3.3–11.5 feet, respectively. Individual scenarios are determined from a Sobol sequence (Sobol', 1967) about these boundaries. We run 153 unique player strategies (including Pelz's baseline targeting strategy T_0) for each scenario. Player execution error in starting line and speed is included in this study. We use four execution profiles: three tour pros' execution profiles and one for an average 90-shooting amateur. We run 1000 execution draws associated with a specific execution profile to determine the average putts to hole out. In summary, repeated across all four execution profiles, there are 32 scenarios $\times$ 153 strategies $\times$ 4 execution profiles $\times$ 1000 execution draws = 19,584,000 total putts simulated for the "short-putt" dataset (Table 5).

Parameters	Value/Count	Description/Details
Scenarios (s)	32	Combinations of putting distance (D), green slope (θ, ϕ), and green speed (S)
Player strategies	153	Fixed speed and line offsets based on the Pelz baseline (v_0, ψ)
Execution profiles	4	Skill profiles evaluated: Tour pros (best, average, worst) and an average 90-shooting amateur
Execution draws	1000	Random deviations in strategy's speed and line based on the execution profile
Total simulated putts	19,584,000	Total number of putts simulated

Table 5: Parameters for the short-putt and lag-putt datasets. The total simulated putts is the product of the number of scenarios, strategies, execution profiles, and execution draws. The only difference in the short and lag putt datasets is the distance range for putts.

While we found strategies that beat the Pelz baseline, their strokes gained on the Pelz baseline are insignificant to an 18-hole score (Table 6). On average across all execution profiles, -0.0010 to 0.0147 strokes gained were found from using the best alternate strategy $T^*(s)$ from each scenario (3 of the 4 execution profiles show a positive mean gain; the worst tour profile shows none), similar to Broadie & Shin's findings (Broadie and Shin, 2014). Even for amateur golfers who benefited the most from winning strategies, it will still take a golfer about seventy putts to shave a single stroke from their total score. Therefore, for short putts, we find that the Pelz baseline targeting strategy is a good approximate strategy for all skill levels.

Dataset	Metric	Tour (best)	Tour (average)	Tour (worst)	Amateur (90-shooter)
Short putt	Average strokes gained for alternate strategies	0.0020	0.0049	-0.0010	0.0147
	Winning scenarios after 4000 execution draws, of 32	14	17	10	15
	Winning scenarios after FDR correction	12	16	8	14

Table 6: Short-putt dataset outcome metrics for chosen strategies. Winning scenarios after 4000 draws beat the Pelz baseline over its entire 95% confidence intervals. Winning scenarios after false discovery rate (FDR) correction means that the scenario survives the Benjamini–Hochberg false discovery rate correction set at 2.5% and beats the Pelz baseline over its 95% confidence interval with the 4000 execution draws. Our winning scenarios are those that pass FDR correction. Average strokes gained are calculated from all alternate strategies, even those that do not beat the Pelz baseline.

6.2 Strategy gains for putts of longer length

For the simulation of putts of longer length, we use the same experimental design as the short-putt dataset, but with a larger distance range of 11.5–50 feet (Table 5). The "lag" dataset consisted of 32 scenarios, with the same slope and speed ranges as the short-putt dataset. The incorporation of a larger distance range induced greater strategy gains for amateurs: on average for all scenarios, amateurs gained 0.037 strokes on the Pelz baseline, with 24 winning scenarios. There are 9.30 initial lag putts per round of golf from 15–50 feet for amateurs, accounting for approximately 52% of all

initial putts (Bansal and Broadie, 2008). Over an 18 hole round, the alternate strategies result in a cumulative gain of $0.037 \times 9.30 \approx 0.35$ strokes per round for amateurs. Over a 72-hole tournament, this corresponds to $0.35 \times 4 \approx 1.39$ expected strokes gained. This is a significant amount of strokes saved for amateurs, as many tournaments are won by a single stroke.

However, a larger distance range for all scenarios did not improve strategy gains for professional profiles. For professionals the winning scenarios were exclusively for putts with greater length than 20 feet, but the strokes gained from these strategies were modest compared to the amateur model (Fig. 4) and didn't arise often enough to offset the lack of winning scenarios for other distances. The worst tour pros only had 6 scenarios where a strategy beat the Pelz baseline after fresh execution draws revalidation and false discovery rate control (Table 7), and the average and best tour pros only had 7 winning scenarios each. None of the tour profiles gained strokes on average from choosing the best alternative strategy $T^*(s)$: the worst, average, and best tour pros actually lost 0.0012, 0.0047, and 0.0057 strokes, respectively, after deviating from the Pelz baseline (Table 7).

Dataset	Metric	Tour (best)	Tour (average)	Tour (worst)	Amateur (90-shooter)
Lag	Average strokes gained for alternate strategies	−0.0057	−0.0047	−0.0012	0.0373
	Winning scenarios after 4000 execution draws, of 32	8	8	7	25
	Winning scenarios after FDR correction	7	7	6	24

Table 7: Lag dataset outcome metrics for chosen strategies over 32 scenarios (11.5–50 feet) per execution profile.

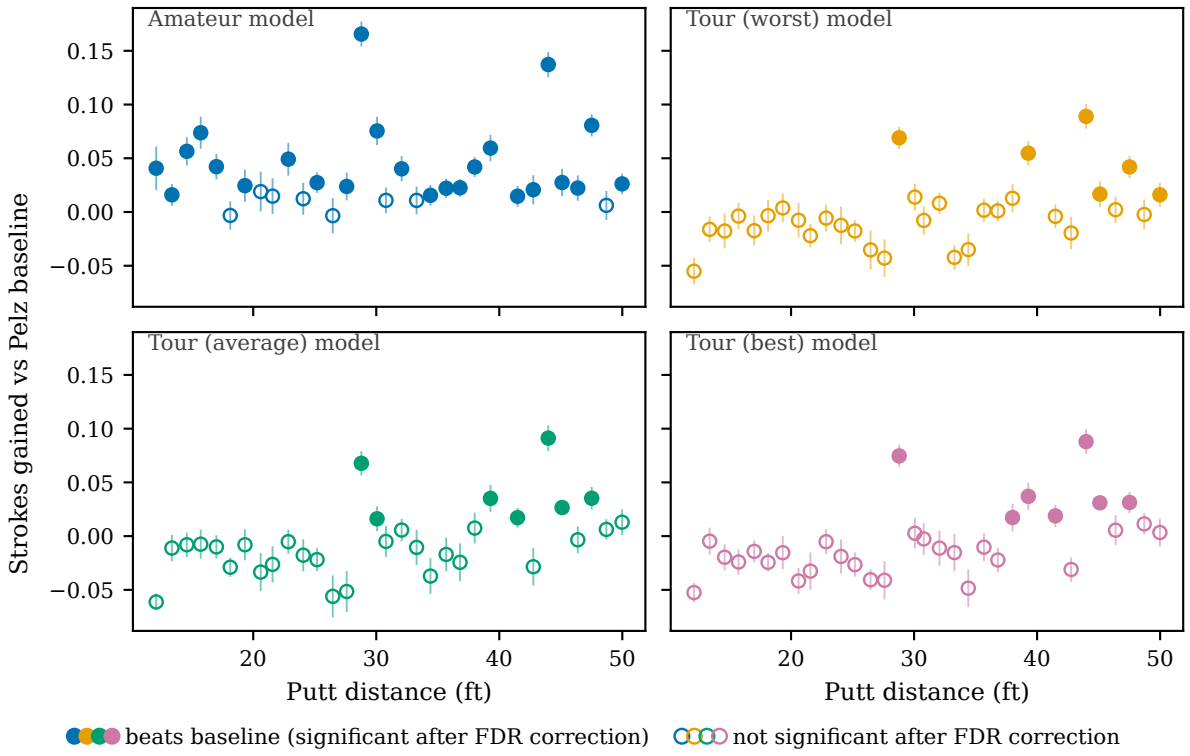


Figure 4: Lag Putt strokes gained for each alternate strategy versus distance for all execution profiles. Filled markers represent winning strategies accounting for fresh execution draws and false discovery rate correction. The vertical lines represent 95% confidence intervals, and open markers mean that the scenario's chosen strategy did not survive false discovery rate correction.

For all execution profiles, high break scenarios (scenarios with a high sidehill slope magnitude $|\theta|$) had more winning strategies, and their winning strategies had a higher strokes gained on the Pelz baseline (Fig. 5). After running a Spearman test (Spearman, 1904) ($\rho(|\theta|, \text{fresh strokes gained})$), we found an upward trend between $|\theta|$ and strokes gained on the Pelz baseline for all execution profiles: ρ was 0.64, 0.53, 0.66, 0.52 for amateurs, tour-worst, tour-average and tour-best profiles respectively. A slope's magnitude is a good indicator of a scenario's expected strokes gained for lag putts; for amateurs, the Spearman coefficient gives 0.01 between a putt's distance and strokes gained.

For most of the winners' strategies, a trend emerged: compared to the Pelz baseline, winners generally had a more conservative approach, aiming for a higher start-line and lower-pace strategy (Fig. 6). For professionals, all winning strategies either aimed higher than the Pelz baseline or hit putts at a slower pace. For amateurs, all except one winning strategy aimed higher or hit putts at a slower pace than the Pelz baseline. Additionally, no winning strategies sat at the bounds of our strategy grid ($\Delta|\psi| = \pm 4.0°$) or the outermost speed rung ($\Delta v_0 = \pm 0.20$ m/s), showing the winning strategies were not results of the search space we defined. Compared to the Pelz baseline, winning strategies selected had a similar make-percentage (Table 8). However, the main difference was how close the missed putts were. The winning strategies had a more equal proportion of putts missed

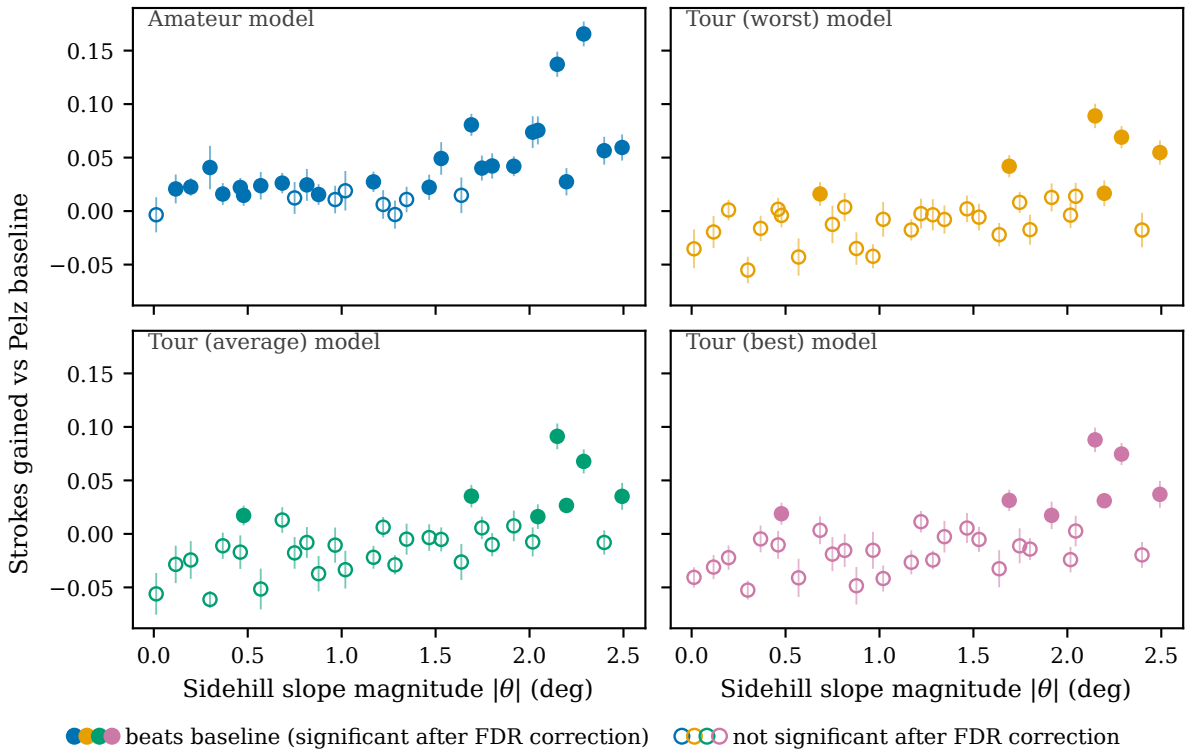


Figure 5: Fresh-sample strokes gained versus sidehill slope magnitude $|\theta|$ for each execution profile. Filled markers are winners after fresh execution draws and false discovery rate correction; vertical lines are 95% confidence intervals.

short and long (47.1% missed short, 45.2% missed long) compared to the Pelz baseline, which had a majority of putts missing "past" the hole (30.3% missed short, 61.1% missed long). A roughly equal proportion of putts were being missed short and long of the hole actually meant the winning strategies had missed slightly closer to the hole. Combined with near equal make percentage to the Pelz baseline, winning strategies would therefore gain strokes by reducing their chances of 3-putting, leaving their second putts closer to the hole. Professional winning strategies follow this same trend; aim higher and hit putts softer in order to make sure the putts stop closer to the hole than the Pelz baseline (minimize 3-putt percentage), at a slightly lower one-putt percentage. The reason professionals gain a negligible amount of strokes from these alternative strategies is because even though the winning strategies are stopping closer to the hole, professional golfers already have an extremely high percentage to hole-out (1.05 expected putts from 2.99 feet) from the stopping distance associated with the Pelz baseline. So, the reduction in mean stopping distance would not significantly affect professional golfers' scoring as much as amateurs. On the other hand, not only do amateurs have a lower probability of holing out from the Pelz baseline's mean leave distance (1.31 expected putts from 3.25 feet), their mean leaves are also farther away from the hole due to their relative lack of skill. Therefore, the conservative strategy, leaving the putts closer to the hole (1.22 expected putts from the winning strategy's mean leave distance of 2.73 feet) than the Pelz

baseline at a slightly lower one-putt percentage, would benefit amateurs the most.

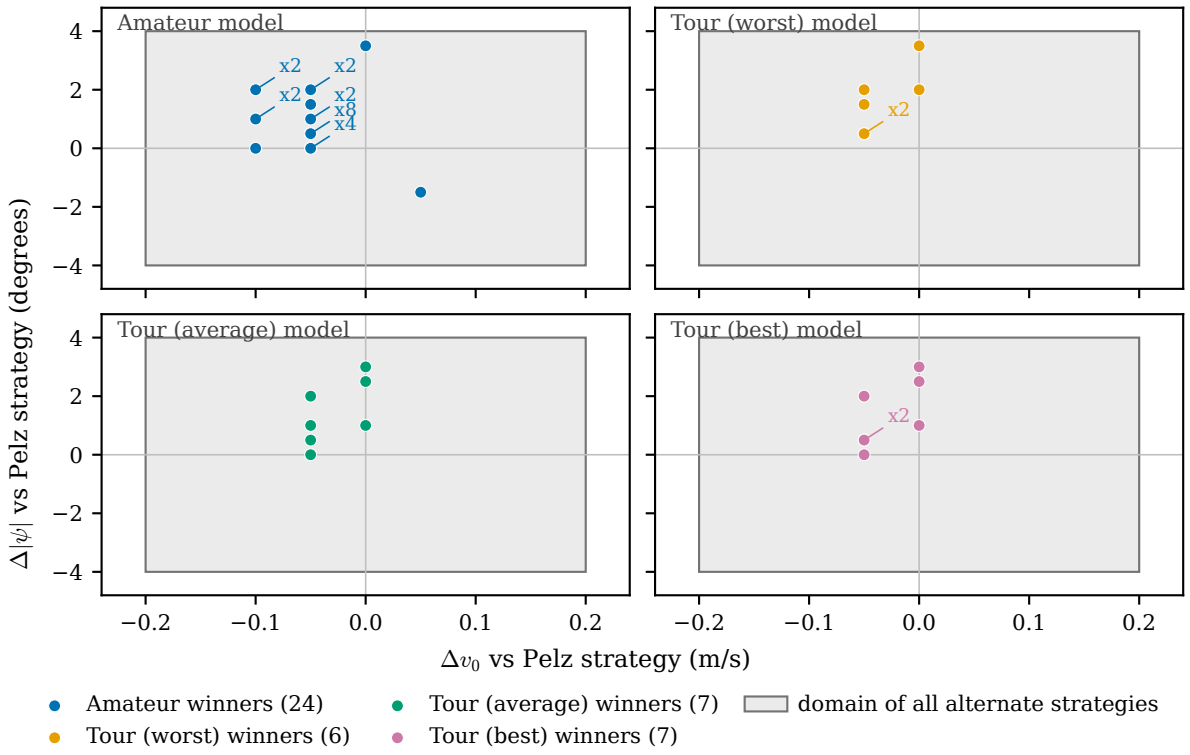


Figure 6: All winning strategy deviations of the starting line and pace from the Pelz baseline. The axes represent the speed and line offsets of the winning strategies versus the Pelz baseline. Positive $\Delta|\psi|$ values indicating higher aim compared to Pelz's suggested $|\psi|$. Negative Δv_0 values indicating lower speed compared to Pelz's suggested v_0. The shaded box is the range of our searched strategies: every scenario's strategy pool is Pelz's baseline plus a uniform lattice of line offsets from 0 to $\pm 4.0°$ in $0.5°$ steps and speed offsets from 0 to ± 0.20 m/s in 0.05 m/s steps, making for 153 strategies per scenario.

	Direction (of winners)		One-putt %		Long-miss %		Short-miss %		Mean leave of misses (ft)	
Execution profile	Slower	Aim higher	Pelz	Winner	Pelz	Winner	Pelz	Winner	Pelz	Winner
Tour (best)	4/7	6/7	9.6	8.9	67.3	45.1	23.1	46.0	3.01	1.97
Tour (average)	4/7	6/7	9.7	8.8	68.8	45.9	21.4	45.2	2.99	1.92
Tour (worst)	4/6	6/6	7.2	5.9	65.3	43.1	27.5	51.0	3.40	2.28
Amateur (90-shooter)	22/24	18/24	8.6	7.7	61.1	45.2	30.3	47.1	3.25	2.73

Table 8: Winning strategies putt outcome metrics and strategies compared to the Pelz baseline strategy. Winning strategies have around the same one-putt percentage as the Pelz baseline; however the more conservative approach allows for a closer leave distance upon misses, making a lower expected putts to hole out overall.

6.3 Robustness and sensitivity analysis

Our stated results rely on set environmental parameters. We first modify our physical assumptions, and then resimulate our winning strategies $T^*(s)$ under the perturbed conditions, analyzing the discrepancies in results. Furthermore, we use our existing physical model to externally validate our simulator, replicating make-rate and hole-out data from published figures. Next, we modify our experimental procedure to account for green-reading error, and measure the effect of including green reading error on the winning strategy's strokes gained on the Pelz baseline. To address our continuation model $N(d)$ (Section 4.5) we rescore all winning strategies' putts' physical outcomes under different continuation models, quantifying the effect of continuation model choice on our results.

6.3.1 Physical and numerical assumptions

Every physical result so far depends on assumptions we use in the simulator. We test whether any single assumption is influencing our calculation of expected strokes to hole out. We resimulated over five physical categories: energy loss from a flyover, the capture speed for a slow putt, our stopping criteria, spin-induced captures/lip-outs, and our integration timestep itself. We resimulate our perturbed models to verify the lag dataset's findings (our main findings arise from the lag dataset). For the resimulations over the model's different flyover energy loss, topple in speed, stopping criteria, and spin-induced lip-outs, we resimulate over scenarios that survived the fresh Monte Carlo confidence interval test (they did not have to pass the False Discovery Rate test), making for a total of 48 scenarios resimulated (8 tour best, 8 tour average, 7 tour worst, and 25 amateur) over the 4 execution profiles with 4000 execution draws per scenario. For the resimulations over the model's integration timestep, we check whether the strokes gained from all alternate strategies vs Pelz baseline (the mean of $\widehat{\mathrm{SG}}^{\mathrm{F}}(s, T^*(s))$ across scenarios), converges on a higher timestep. Therefore, we resimulate all (not just winning) scenarios for all execution profiles with different timesteps. In every perturbed re-evaluation reported below, every winning strategy stayed ahead of the Pelz baseline on the point estimate; the tables report how many winning scenarios kept their 95% confidence interval and how far their strokes gained moved.

Flyover energy loss. The first parameter we resimulated was flyover energy loss. When the ball's speed exceeds a certain threshold when entering the hole, it does not have enough time to "drop" into the cup. Instead, the ball passes right through by "ski jumping" over the cup and briefly interacting with the far edge of the hole before returning to the surface. This phenomenon is called a "flyover" putt. In a flyover putt, the ball bounces off the rim wall and loses kinetic energy. The model we used (Hogan & Antali's lip-out model, Hogan and Antali, 2025) does not account for flyover energy loss; the exact speed loss is an experimental constant that varies from hole to hole, so we assume no energy loss for our main simulations. To study the effects of flyover energy loss, we resimulated all scenarios whose 95% confidence intervals for strokes gained beat the Pelz baseline with a flyover speed loss of 30, 50, and 70 percent (Table 9). For the majority of the scenarios,

the flyover speed loss had a small effect on the winning strategies' strokes gained, with the average change in strokes gained being less than 15% of the scenario's own strokes gained.

	Alternate strategies whose 95% CI still beats Pelz (kept / total)				Average shift in strokes gained
Flyover speed loss	Amateur	Tour worst	Tour average	Tour best	% of scenario's own strokes gained
30%	23/25	6/7	7/8	8/8	9.8%
50%	22/25	6/7	7/8	8/8	13.3%
70%	21/25	6/7	7/8	8/8	14.2%

Table 9: Effects of flyover speed loss on putt outcomes for scenarios that beat the Pelz baseline for the entire 95% confidence interval. Note that these are not necessarily winning scenarios; winning scenarios must also survive the Benjamini–Hochberg correction. The average shift in strokes gained is the average percentage change in $\widehat{\mathrm{SG}}^{\mathrm{F}}(s, T^*(s))$ after the change in flyover speed loss, averaged over all 48 alternate strategies.

Topple-in speed. The second parameter we resimulated over was the topple-in speed slow putts require to be captured by the hole, the c_2 capture threshold of Eq. (10). A ball that arrives crawling actually reaches the rim and starts rolling around the edge of the cup; whether it goes in or enters region 3/4 capture motion goes to a certain speed. Because our conservative strategy induces slow-moving putts at the hole boundary, the speed boundary is the door the winners' softer pace uses. So, we scaled the topple-in speed that slow putts go in by ±20%, narrowing and widening that window and analyzing its effects (Table 10). All scenarios still beat the Pelz baseline despite changes in topple-speed, with the change in topple-in speed on average affecting less than 1% of a scenario's own strokes gained.

	Alternate strategies whose 95% CI still beats Pelz (kept / total)				Average shift in strokes gained (over all 48 alternate strategies)
Topple-in speed scale	Amateur	Tour worst	Tour average	Tour best	% of scenario's own strokes gained
×0.8	25/25	7/7	8/8	8/8	0.6%
×1.2	25/25	7/7	8/8	8/8	0.8%

Table 10: Effects of topple-in speed scale on strokes gained for scenarios that beat the Pelz baseline.

Rim spin band. The third parameter was resimulating over the rim spin band. Under certain conditions, a ball lips out even after it enters the hole temporarily due to an excess of spin (Hogan and Antali, 2025, §7). Our model currently omits spin, because Penner models a golf ball's trajectory as pure rolling without spin (Penner, 2002). The model has a precise bound between when the ball catches the rim and drops, or lips out instead, depending on the initial velocity and impact parameter (how far off-center the ball enters the hole). At that edge is where the spin physics would be the most important. We first resimulate so every putt the model captured when arriving with less than 0.05 m/s of the boundary curve is a lip-out. Then, we have a harsher version where every putt

within 0.10 m/s of the borderline counts as a lip-out (Table 11). Even with the worst-case analysis of spin's effect, our results still hold, with all winning strategies still beating the Pelz baseline. Even reversing every capture within 0.10 m/s within the boundary covers a tiny portion of the arrival space. So, effects on winning strategies are small.

	Alternate strategies whose 95% CI still beats Pelz (kept / total)				Average shift in strokes gained (over all 48 alternate strategies)
Rim spin band	Amateur	Tour worst	Tour average	Tour best	% of scenario's own strokes gained
0.05 m/s	24/25	7/7	8/8	8/8	2.7%
0.10 m/s	25/25	7/7	8/8	8/8	3.7%

Table 11: Effects of rim spin band on strokes gained for scenarios that beat the Pelz baseline.

Stopping criteria. Our fourth parameter was the minimum speed and acceleration requirement a ball needed for the simulator to continue motion. It could be argued that our minimum speed and acceleration thresholds for our putts were too large, making the ball stop too early. So, we resimulated all winning scenarios with a halved minimum speed and acceleration stopping criteria. Halving both thresholds moved no winning scenario's strokes gained by more than 0.000010 strokes, and every winning scenario kept its 95% confidence interval.

Integration timestep. Our last condition was the timestep of the experiment itself. If our integration timestep is too large, the simulator would not be an accurate representation of a ball's real trajectory, affecting the initial state of the hole capture model's parameters (the entry speed v and impact distance δ, Eq. (7)). We partially account for this by interpolating a hole interaction initial parameters (v, δ) between two timesteps. However, within that one timestep, the true path would still account for gravity and friction acting continuously, and our within-step reconstruction is a straight line trajectory. So, we still decided to resimulate all scenarios, not just winning strategies, with timesteps ranging from a halved timestep (0.01 seconds) to Broadie & Shin's original timestep (Broadie and Shin, 2014) in 0.01 second increments, to see how results would change if we chose a different timestep. For all resimulations, we use 4000 inner execution draws per scenario to determine if our mean strokes gained per scenario gets changed. We also resimulate all scenarios with Broadie & Shin's integrated timestep, to see how our results would change.

Our timestep choice did not significantly affect our results. With the halved timestep resimulations, the maximum discrepancy between our chosen timestep, 0.02, and the halved timestep's $\widehat{\mathrm{SG}}^{\mathrm{F}}(s, T^*(s))$ was only 0.00053 strokes (Fig. 7). All changes in $\widehat{\mathrm{SG}}^{\mathrm{F}}(s, T^*(s))$ associated with the timestep change stayed under a quarter of the original $\widehat{\mathrm{SG}}^{\mathrm{F}}(s, T^*(s))$ per scenario or 2 times the standard error of $\widehat{\mathrm{SG}}^{\mathrm{F}}(s, T^*(s))$, ensuring that our strategies did not pass under a new timestep only because of statistical noise. For timesteps smaller than around 0.04 seconds, the strokes gained on the Pelz baseline converge. The winner's magnitude of strokes gained on the baseline is inflated by Broadie & Shin's integration timestep (Broadie and Shin, 2014), which makes our apparent strokes

gained metric higher.

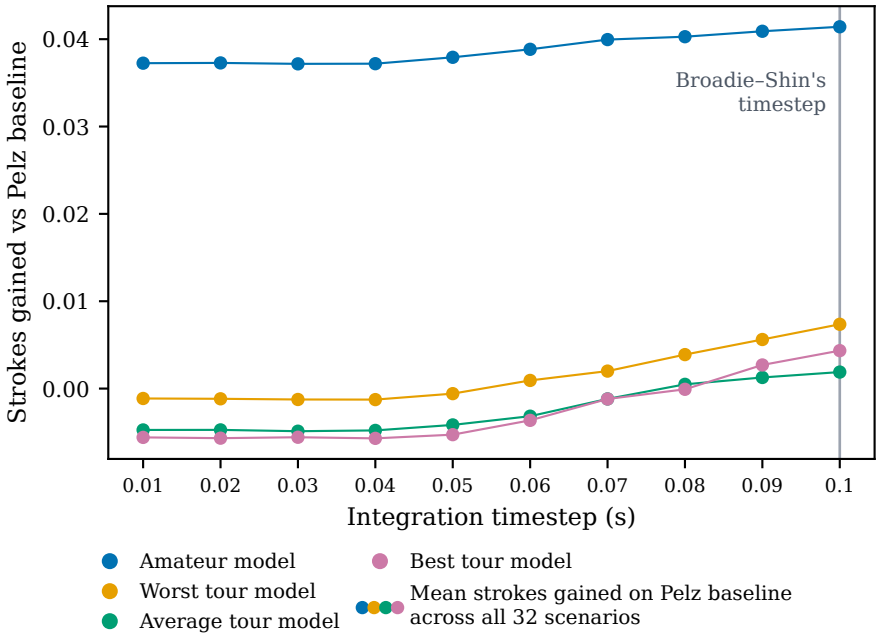


Figure 7: Winning strategies' strokes gained on the Pelz baseline versus a simulation's integration timestep for amateur and pro skill levels.

6.3.2 Error-model sensitivity

Green misread model For our main experiment, we do not model a golfer's misread. As a result, we develop a misread model that takes into account a golfer's "perceived" slope, which is the slope the golfer would read after their associated green reading uncertainty. We then resimulate all 48 strategies that beat the Pelz baseline (including the four strategies that failed with fresh FDR validation) and their associated Pelz baseline (25 amateur scenarios, 7 tour-worst scenarios, 8 tour-average scenarios, 8 tour-best scenarios), with the green-reading model and analyze the differences between including and not having green-reading error.

The misread model models the slope that a golfer reads, which is based on the green's true slope and a golfer's skill level(Section 4.4). The reading noise constants for both amateur and the worst tour professional profiles were calibrated per player profiles, with standard deviation being 0.25 degrees for amateurs (Bansal and Broadie, 2008), and 0.435, 0.379, and 0.400 degrees for the worst, best, and average touring professionals respectively (Broadie and Shin, 2014). Each reading noise constant (worst tour professional and amateur) was the result of a different calibration method from the two different sources (Bansal & Broadie and Broadie & Shin). Therefore, the higher standard deviation for touring professionals is not evidence that pros read worse than amateurs. The player then chooses the strategy's start-line (ψ) and start-speed (v_0) based on the incorrect read. Since the player is actually putting on the real green with the true slope, the incorrect read is executed on the true green. Both the Pelz baseline strategy and the winning strategy for all 48 scenarios

are re-sampled with 32 independent green reads (a player's perceived slope of the green, accounting for error in green reading) in order to make the comparison fair; the implementation of the green reading error would affect both the winning and the Pelz baseline strategy. For each green read, both strategies are re-evaluated on the true, physical green with the same 1000 execution draws, accounting for the execution starting line and speed error. We then find the mean expected strokes gain of the strategy over the baseline for all 32 reads.

Under calibrated misreading, the winning strategies still beat the Pelz baseline on average (Table 12). Averaged across all 25 amateur scenarios, the winning strategies beat the Pelz baseline by 0.0333 strokes (95% confidence interval [0.0211, 0.0454]). For each scenario, two versions of the strokes gained on Pelz baseline exist: the strokes gained under a correct read assumption, and the strokes gained under a misread. Both versions are measured on the same execution offsets, so only the effect of the misread affects the final result. We define how much strokes gained is affected by the misread as the strokes gained with misread divided by the strokes gained without the misread across all 48 scenarios. After accounting for a golfer's misread, this fraction's median is 0.73, showing that for the median strategy 73% of the initial strokes gained on the Pelz baseline is maintained with green misreading.

Alternate strategies whose mean strokes gained still beats Pelz (kept / total)				Median strokes gained with misread / strokes gained without misread
Amateur	Tour worst	Tour average	Tour best	All player profiles
24/25	7/7	7/8	7/8	0.73

Table 12: Effects of green misreading on strategy performance. We measure what fraction of strokes gained on the Pelz baseline was maintained under misreading conditions by dividing strokes gained with misread by strokes gained without misread for all scenarios, reporting its median value.

Continuation model $N(d)$ robustness Every single putt a golfer takes in the simulation is scored as follows: If the ball is holed, it counts as one stroke; otherwise, the total expected strokes is $1 + N(d)$. Expected putts do not just depend on distance; other factors like slope of the greens and stimp (green speed) play a significant role in determining expected strokes. Because of this variability in the accuracy of a distance-based continuation model, it could be argued that our winning strategies are a direct result of which scoring model we used, not of an actual "better" strategy. As a result, we took the previously stored physical outcomes from all 32 scenarios (no re-simulation) in the 4 lag datasets (1 dataset per skill level; 3 for the tour-level profiles, 1 for the amateur profile). We then rescored all physical outcomes under 3 continuation models: the PCHIP interpolated PGA Tour expected strokes table (PGA Tour, 2011), the PCHIP interpolated Bansal & Broadie amateur expected strokes table (Bansal and Broadie, 2008) (designed to represent amateur golfers, having higher continuation putts per distance), and a custom saturating exponential model, defined as:

$$N(d) = 1 + a\,(1 - e^{-bd}) \tag{30}$$

with $a = 1.35$ and $b = 0.1804$. The formula is designed so that the expected putts flatten at 2.35 putts, modeling that the difference between a 40 and 50 footer's expected putts is small relative to the difference in a 3 and 13 footer's expected putts. This model is completely synthetic, with hand-set parameters that are not calibrated with real data. This is important because the purpose of this test is to evaluate whether the shape of the continuation putts function drives winning strategies. While swapping PGA Tour continuation models for amateur continuation models makes short putts more taxing, both models are calibrated on real data and follow a natural shape. The saturating exponential model differs in that the overall shape is different, so we can evaluate whether the winning strategies are better just because of an arbitrary, modeled shape. Because each dataset is rescored under all models, the continuation models in the sweep include the original scoring model used for the original results (amateur continuation model for the amateurs, the PGA Tour PCHIP model for all tour profiles). For reproducibility, we checked and verified that all results were identical when rescoring under the same model. As a result, for the 4 skill profiles there were 2 new continuation models, making for 8 new dataset-continuation model pairings.

For the dataset-continuation model pairings that affected the results most, 20 out of 32 scenarios still have the strategy initially chosen $T^*(s)$ beating the Pelz baseline. For the dataset-continuation model pairing that affected results least, 30 out of 32 scenarios still keep the initial strategy beating the Pelz baseline. Furthermore, the initially chosen strategy still gains on average 0.012 to 0.043 strokes under all different continuation models, showing that the strategies still beat the Pelz baseline, even under different continuation models.

7 Conclusion

In this work, we implement a physics-based simulation that combines a golf ball's rolling trajectory and rim-interaction dynamics to evaluate alternate putting strategies across varying green topographies and skill levels. We effectively integrate multiple physical models to simulate the complete motion of a golf ball on a putting surface, combining the Penner capture model with Hogan & Antali's rim-interaction model (Penner, 2002; Hogan and Antali, 2025) to account for a ball's lip-out behavior and flyover state, as well as determining the physics to change state between the different models. Our simulator includes Monte Carlo simulation to capture a golfer's execution variances, and we develop a non-recursive method to determine a strategy's expected strokes to hole out that reduces computation time and resources. By comparing a strategy's expected strokes to the Pelz baseline targeting strategy, we are able to determine alternate strategies that gain strokes on the Pelz baseline. To account for statistical side efects such as winner's curse (Capen et al., 1971; Smith and Winkler, 2006) and the false discovery rate associated with multiple strategies, we use a multi-stage evaluation process to determine winning strategies that beat the Pelz baseline. First, we use paired offset execution draws to determine the best alternate strategy for each scenario. Then we use fresh execution draws to evaluate the winning strategy's performance against the Pelz baseline and account for winner's curse associated with the initial evaluation. Finally,

we use the Benjamini–Hochberg procedure to control the false discovery rate associated with the multiple winning strategies (Benjamini and Hochberg, 1995).

Our study reveals that for short putts from 3.3 to 11.5 ft, all four player profiles (90-shooting amateur, worst, average, and best tour pro) only gain -0.0010–0.0147 strokes per putt. However, for lag putts from 11.5 to 50 ft, only the amateur gained significant strokes from alternative strategies, gaining 0.037 strokes per putt. This may seem like a small amount; however, the effects of the strategy are cumulative to ≈ 1.39 strokes over the course of a 72-hole tournament, where winning a tournament can be determined by a single stroke. Touring pros find negligible benefit from using alternative strategies in lag putts. However, touring pros gain from alternate strategy usage exclusively in long lag putts over 20 feet. Additionally, both amateurs and touring professionals gain significantly more strokes from alternate strategies in high-breaking scenarios. For a majority of winning scenarios, a conservative strategy is favored: one that aims higher and hits putts softer than the Pelz baseline. For both amateurs and professionals, it is suggested for long and high-breaking putts to aim higher and hit putts softer to minimize the distance from the hole after the first putt. However, for short putts, the Pelz baseline targeting strategy is sufficient for both amateurs and professionals.

References

Matulya Bansal and Mark Broadie. A simulation model to analyze the impact of hole size on putting in golf. In S. J. Mason, R. R. Hill, L. Mönch, O. Rose, T. Jefferson, and J. W. Fowler, editors, *Proceedings of the 2008 Winter Simulation Conference*, pages 2826–2834. IEEE, 2008.

Yoav Benjamini and Yosef Hochberg. Controlling the false discovery rate: A practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society. Series B (Methodological)*, 57(1):289–300, 1995. doi: 10.1111/j.2517-6161.1995.tb02031.x.

Mark Broadie. Assessing golfer performance on the PGA TOUR. *Interfaces*, 42(2):146–165, 2012. doi: 10.1287/inte.1120.0626.

Mark Broadie and Dongwook Shin. Golf analytics: A random putting model and its applications to optimal targeting strategy and attribution analysis. Working paper, Graduate School of Business, Columbia University, 2014.

E. C. Capen, R. V. Clapp, and W. M. Campbell. Competitive bidding in high-risk situations. *Journal of Petroleum Technology*, 23(6):641–653, 1971. doi: 10.2118/2993-PA.

Douglas Fearing, Jason Acimovic, and Stephen C. Graves. How to catch a tiger: Understanding putting performance on the PGA TOUR. *Journal of Quantitative Analysis in Sports*, 7(1), 2011. doi: 10.2202/1559-0410.1268.

F. N. Fritsch and J. Butland. A method for constructing local monotone piecewise cubic interpolants. *SIAM Journal on Scientific and Statistical Computing*, 5(2):300–304, 1984. doi: 10.1137/0905021.

F. N. Fritsch and R. E. Carlson. Monotone piecewise cubic interpolation. *SIAM Journal on Numerical Analysis*, 17(2):238–246, 1980. doi: 10.1137/0717021.

Will Grimsley. Hogan, Middlecoff fail to qualify. *Mason City Globe-Gazette*, page 13, 1957. Associated Press wire story, April 6, 1957.

S. John Hogan and Mate Antali. Mechanics of the golf lip out. *Royal Society Open Science*, 12: 250907, 2025. doi: 10.1098/rsos.250907.

Brian W. Holmes. Putting: How a golf ball and hole interact. *American Journal of Physics*, 59(2): 129–136, 1991.

Stephen Joe and Frances Y. Kuo. Constructing Sobol sequences with better two-dimensional projections. *SIAM Journal on Scientific Computing*, 30(5):2635–2654, 2008. doi: 10.1137/070709359.

Seong-Hee Kim and Barry L. Nelson. Selecting the best system. In Shane G. Henderson and Barry L. Nelson, editors, *Simulation*, volume 13 of *Handbooks in Operations Research and Management Science*, pages 501–534. Elsevier, 2006. doi: 10.1016/S0927-0507(06)13017-0.

Dave Pelz and James A. Frank. *Dave Pelz's Putting Bible: The Complete Guide to Mastering the Green*. Doubleday, 2000. ISBN 9780385500241.

A. R. Penner. The physics of putting. *Canadian Journal of Physics*, 80(2):83–96, 2002. doi: 10.1139/p01-137.

PGA Tour. Strokes gained – putting baseline (2010 season). PGATOUR.COM, 2011. URL https://web.archive.org/web/20110505093031/http://www.pgatour.com/r/strokes-gained-putting-baseline/index.html. Archived by the Internet Archive Wayback Machine, May 5, 2011.

James E. Smith and Robert L. Winkler. The optimizer's curse: Skepticism and postdecision surprise in decision analysis. *Management Science*, 52(3):311–322, 2006. doi: 10.1287/mnsc.1050.0451.

I. M. Sobol'. On the distribution of points in a cube and the approximate evaluation of integrals. *USSR Computational Mathematics and Mathematical Physics*, 7(4):86–112, 1967. doi: 10.1016/0041-5553(67)90144-9. English translation of Zh. Vychisl. Mat. Mat. Fiz. 7(4):784–802 (1967).

C. Spearman. The proof and measurement of association between two things. *The American Journal of Psychology*, 15(1):72–101, 1904. doi: 10.2307/1412159.

Kasra Yousefi and Tim B. Swartz. Advanced putting metrics in golf. *Journal of Quantitative Analysis in Sports*, 9(3):239–248, 2013. doi: 10.1515/jqas-2013-0010.