Periodic Trajectories of Mobile Robots

Alexandra Q. Nilles and Steven M. LaValle

***Abstract*— Differential drive robots, such as robotic vacuums, often have at least two motion primitives: 1) travelling in straight lines, and 2) rotating in place upon encountering an obstacle. In this paper, we analytically determine the location and stability of periodic trajectories of such robots in regular polygons. This analysis leads to simple, open-loop control schemes that allow either predictable and stable "patrolling" dynamics, or ergodic "exploratory" dynamics. The results are useful for controlling simple mobile robots with minimal sensing and actuation, in spaces with known geometry.**

I. INTRODUCTION

Consider the path that a differential-drive mobile robot takes as it navigates a room. This robot has two motion primitives that it can execute reliably: moving in a straight line, and turning in place. It also has sensors that allow the robot to determine whether it is in contact with an environmental boundary, and its heading relative to that boundary.

Can we guarantee that the robot **patrols** the space on a repeatable, periodic path? Robots with robust patrolling behavior have applications such as monitoring environmental conditions in labs, warehouses, or greenhouses, where a few fixed sensors may not give enough information.

One way to create these motion patterns for mobile robots is to augment the robots with sensors such as cameras, and use algorithms such as SLAM to compute a map of the space and estimate the robot's state. However, these robots are expensive, require a large amount of computational and electrical power, and their accuracy can be impacted by changing environmental conditions (such as low light).

The approach of this paper is to treat the robot as a dynamical system, defined by its motion primitives, independent from the specific hardware implementation. This dynamical system is closely related to mathematical billiards [1]. In billiards, an agent travels in a straight line until contacting a boundary of its environment, then bounces so that the incident angle is always related to the outgoing angle by $\theta_i = -\theta_o$ (*specular* bouncing, see Figure 1).

However, classical billiards is a *conservative* dynamical system - collisions are elastic, and the energy of the agent is conserved. But robots (with energy supplies) can have nonconservative dynamics. For example, *pinball billiards* is a model where the agent is deflected toward the normal with each bounce, $\gamma\theta_i = -\theta_o$ for $0 \leq \gamma \leq 1$. If the agent bounces at the normal vector each time, this is called *slap billiards*. Work in this area focuses on analyzing the existence and structure of attractors [2] [3].

A. Nilles and S. LaValle are with the Department of Computer Science, University of Illinois at Urbana-Champaign. Urbana, IL 61801, USA. {nilles2,lavalle}@illinois.edu

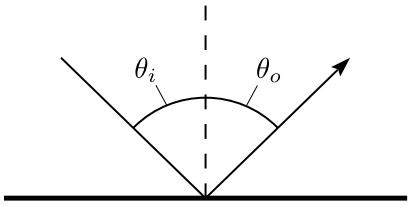


Fig. 1. Definitions of the incoming angle of the agent (θ_i) and the outgoing angle (θ_o), relative to the normal vector at the point of impact.

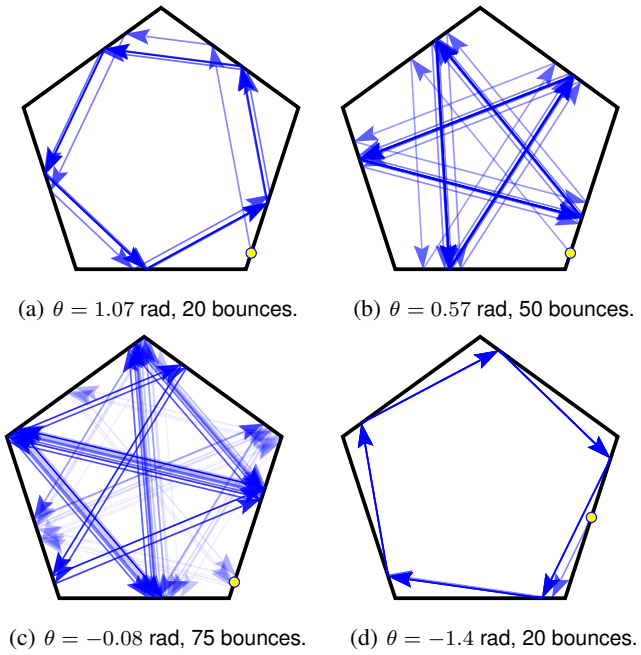
The dynamical system in this paper generalizes slap billiards: the agent travels in straight lines, and upon collision with a boundary, bounces off the boundary at a fixed angle relative to the boundary normal vector, independent of the incoming angle. The contributions of this paper are 1) proving the existence of periodic orbits in n-sided regular polygons, and showing the range of bounce angles that will produce periodic orbits; 2) an analysis of these orbits, showing their stability and robustness to modeling errors; and 3) the locations where the robot collides with the polygon while in these stable periodic orbits.

This paper is organized as follows: Section II outlines the results of the previous paper on this model, as well as closely related work. Section III presents the model definition and some useful concepts from dynamical systems. Section IV analyzes the specific case where the robot is constrained to bounce between sequential edges of the polygon, which illustrates a technique for using the fixed point of a trigonometric function to determine the existence and location of a periodic path. Section V generalizes this approach to find periodic paths of period k in regular n-gons, when k divides n. We then discuss possible extensions of this work, including to more generalized polygonal environments, and remaining open questions.

II. PRIOR WORK

Also related is work on the combinatorial complexity of the region touched by specular bouncing ("visibility with reflection") in simple polygons [4]. There, the authors show the combinatorial complexity of the region touched after k bounces, and provide a near-optimal algorithm for computing the region. Our work differs by looking at non-specular bouncing and focusing on the effects of environment symmetries on the structure of this region.

In robotics, our model is inspired by systems such as differential drive mobile robots with bump sensors and as few as one single-point infrared range sensors [5], which are able to execute the action of aligning to a specified angle to a wall, and travelling in straight lines. This dynamical



(a) $\theta = 1.07$ rad, 20 bounces. (b) $\theta = 0.57$ rad, 50 bounces.

(c) $\theta = -0.08$ rad, 75 bounces. (d) $\theta = -1.4$ rad, 20 bounces.

Fig. 2. Different behaviors of bounce trajectories in a regular pentagon. Older bounces become 2% more transparent with each new bounce. The circle on the boundary indicates the starting point of the trajectory.

analysis relies only on the ability to execute the two motion primitives (move straight forward and align to θ), and does not rely on any specific hardware. **todo:** elaborate on how this shows our model is feasible

In the previous paper [6], the authors characterized some of the long-term dynamics of this dynamical system. They showed that the robot will have an orbit of period two between parallel edges, and the robot will move monotonically "outward" from acute vertices, or "out" from edges that would meet in an acute vertex if extended to their intersection.

The previous paper then defined distance- and link-unbounded segments on arbitrary polygons, as regions of the boundary where the robot may travel an unbounded distance while bouncing in that region or bounce an unbounded number of times. The paper describes an algorithm for classifying the boundary of an arbitrary polygon into distance- and link-unbounded segments. However, the algorithm will not terminate when the robot's trajectory converges to a periodic orbit, since in this case the distance- and link-unbounded regions shrink to points on the polygon boundary, such as in Figures 2a and 2b.

The purpose of this paper is to begin to identify such cases - where the dynamical system's attractor is a set of separate points, not intervals, on δP. Then the algorithm in [6] can be used only for environments with attractors that are segments on δP, for which the algorithm is guaranteed to terminate.

III. MODEL DEFINITION

A point robot moves in a bounded subset of the plane P, defined by a continuous boundary δP, homeomorphic to $\mathbb{S}^1$. For most of this discussion, the environment will be a simple regular polygon, so δP is piecewise linear: an n-gon with n straight edges intersecting at n vertices $(v_0, v_1, \ldots, v_{n-1})$.

The robot drives in a straight line until encountering δP. It then rotates until its heading is at an angle θ clockwise of the inward-facing boundary normal, where $-\pi/2 < \theta < \pi/2$. Then the robot sets off in a straight line again. The mapping function that these actions define between points on δP is $B_\theta : \delta P \to \delta P$, defining our dynamical system. We will refer to this function as the *bounce map*, in the same spirit as the *logistic map* in discrete dynamical systems. When the bounce map is iterated k times, we write B_θ^k. B_θ is not well defined on vertices of δP, but the number of points where B_θ sends the robot to a vertex is finite, so we will not consider such trajectories.

A sequence of points $[p_0, \ldots, p_k]$ is a *flow* of B_θ if $p_i = B_\theta(p_{i-1})$ for $1 \le i \le k$. A flow is an *orbit* if $B_\theta(p_k) = p_0$, and the *period* of this orbit is $k+1$. A *limit cycle* is an orbit whose neighboring flows tend asymptotically toward or away from it [7].

IV. BOUNCING TO SEQUENTIAL EDGES

Proposition 1: In every regular n-sided polygon with side length l and interior angle ϕ, there exists a range for θ such that iterating $B_\theta(x)$ on any $x \in \delta P$ results in a stable limit cycle of period n, which strikes the boundary at points that are distance x_{FP} from the nearest clockwise vertex, with:

$$x_{FP} = \begin{cases} \frac{lc(\theta,\phi)}{1+c(\theta,\phi)} & \phi/2 < \theta < \pi/2 \\ \frac{l}{1+c(\theta,\phi)} & -\pi/2 < \theta < -\phi/2 \end{cases}$$

Where $c(\theta, \phi) = \cos(\theta)/\cos(\theta - \phi)$.

Proof: Take a regular n-gon with side length l, and boundary δP. Let the robot begin its trajectory at a point $p \in \delta P$ which is at a distance x from the nearest vertex in the clockwise direction. We will begin by constraining the robot to bounce counterclockwise, at an angle θ such that it strikes the nearest adjacent edge, such as in Figure 2a.

Define a map $f_\theta : (0, l) \to (0, l)$ that takes x, the robot's distance from vertex p_i, and maps it to $f(x)$, the resulting distance from vertex p_{i+1}, after application of this constrained bounce map.

Then, using the triangle formed by two adjacent edges and the robot's trajectory between them, we can solve for $f(x)$. Let $\phi = (n-2)\pi/n$ be the vertex angle of the regular polygon.

By the law of sines:

$$\frac{f_\theta(x)}{\sin(\pi/2 - \theta)} = \frac{l - x}{\sin(\pi - (\pi/2 - \theta) - \phi)}$$

$$f_\theta(x) = \frac{(l - x)\cos(\theta)}{\cos(\theta - \phi)} = c(l - x)$$

By iterating this map, we find that the fixed point is:

$$f_\theta^\infty = \sum_{i=1}^{\infty}(-l)(-c)^i = l + \sum_{i=0}^{\infty}(-l)(-c)^i$$

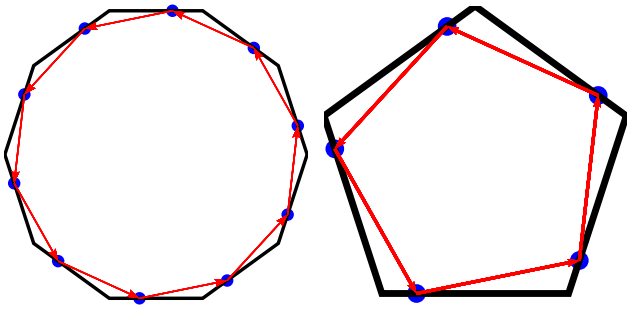


Fig. 3. Predicted (collisions indicated by blue dots) and simulated limit cycles when bouncing to adjacent edge in regular polygons.

The sum is geometric, and finite when $|c| < 1$. If this condition holds, then the fixed point becomes:

$$f_\theta^\infty = \frac{lc}{1+c}$$

So we would expect the trajectory of a robot with bounce angle θ satisfying $|c| < 1$ to converge to a limit cycle in the shape of an inscribed n-gon, with collision points at distance $(lc)/(1+c)$ from the nearest vertex in the clockwise direction.

When this calculation is redone for bounces with $-pi/2 < \theta < 0$, the resulting fixed point is $l/(1+c)$, with the same $c = \cos(\theta)/\cos(\theta - \phi)$. Since $c \in (0,1)$ for stable orbits, we find that the fixed point for bounces can take all values in $(0, l)$ except for $l/2$, which would require $c = 1$. ■

Also, note that calculating the Lyapunov exponent for the map $f_\theta(x)$ gives the same result: $|f'_\theta(x)| = c$, which implies that $|c| < 1$ gives a stable fixed point. **cite?**

This condition, $|c| < 1$, implies that a stable cycle will result for any θ within the bounds $\phi/2 < \theta < \pi/2$ or $-\pi/2 < \theta < -\phi/2$, which were confirmed through simulation, such as in Figure 3.

A. Implications for Errors in Implementations

For each stable orbit in a given environment, we can use the bounds on c to determine the range of angles that will result in that orbit. Thus, when designing a "patrolling" robot in an environment with regular polygonal geometry, a robot designed to bounce at an angle in the center of one of these ranges will be maximally robust to actuator or sensor errors. The resulting maximum allowable error, ϵ_{max}, will be $\pm|(\pi - \phi)/2|$. Bounces with error within this range will still result in stable orbits of the workspace.

However, these orbits will impact the boundary at a different location than expected. If there is a constant error in the bounce angle, so that the effective bounce angle is $\theta + \epsilon$, with $\epsilon < \epsilon_{max}$, the resulting difference in the location of the collision point on each edge will be $\Delta_x = f^\infty_{\theta+\epsilon} - f^\infty_\theta$.

It is likely that physical implementations of the required motion primitives will be imperfect - for example, differential drive robots can have assymetries in the motors powering each wheel, which would result in a slightly curved path through the interior of the environment, or a slight under- or over-turning while aligning to θ. These differences between the model and the implementation may be a constant offset, $\theta+\epsilon$ - or they may be time-varying. **todo bound error from independent random ϵ at each stage**

V. GENERALIZATION

Proposition 2: In every regular n-sided polygon, there exists a stable limit cycle with period k for all k such that $k > 1$ and $k|n$.

Proof: By induction: for all prime n, $n \geq 3$, the statement is true by Proposition 1, since $k = n$ and Proposition 1 guarantees a stable limit cycle that strikes each edge sequentially.

Assume the statement is true for all $n' < n$. Then Proposition 1 guarantees a stable limit cycle that strikes each edge of an n-sided polygon sequentially. For all k such that $k|n$, we can choose k edges of the n-gon such that the edges are equally spaced. We can then imagine extending these edges to their intersection points, forming a regular k-gon. By Proposition 1, the bounce map in this regular k-gon is guaranteed to induce a stable limit cycle, with collision points at parameter x on the edge for all x except the very center point of the edge. Thus θ can be chosen such that the bounce map sends the robot to every $(n/k) - th$ edge such that the trajectory will converge to a stable cycle with period k. Thus by induction, the statement is true for all $n \geq 3$. ■

Theorem 1: In every regular n-sided polygon with side length l, if $k|n$, there exists a stable periodic orbit with k bounces where each collision with the polygon boundary is at a distance x $(0 < x < l)$ from the nearest vertex in the clockwise direction. **todo: x as function of k**

Proof: Instead of bouncing between adjacent edges, we may ask what happens when the robot bounces between edge p_0p_1 and edge p_mp_{m+1}, "skipping" $m-1$ edges, such as in Figure 2b where the robot bounces off every other edge.

Let $m \leq \lfloor n/2 \rfloor$ (if this is not the case, reflect the polygon across the vertical center axis, solve, and reflect back).

Extend the line segments p_0p_1 and p_mp_{m+1} to their point of intersection q, forming the triangle $p_0p_{m+1}q$. Let $a = \angle qp_{m+1}p_0 = \angle qp_0p_{m+1}$, by symmetry. Let $b = \angle p_{m+1}qp_0$. Let $A = |qp_{m+1}| = |qp_0|$ and $B = |p_{m+1}p_0|$. See Figure 4. Each of the sides of the polygon has length l, and the robot begins its trajectory at a point which is distance x from p_0. We wish to find the resulting distance from point p_m, $f_{\theta,m}(x)$.

Then, by the law of sines, we have:

$$A = \frac{B\sin(a)}{\sin(b)}$$

We can then form the triangle from the points p_0, p_{m+1}, and the center of the regular n-gon. The distance from the center of a regular n-gon to any of its vertices is $\frac{l}{2\sin(\pi/n)}$. The angle subtended by the edges p_0p_1 through p_mp_{m+1} is $2\pi(m+1)/n$. Thus we can solve for B:

$$B = \frac{l\sin(\pi(m+1)/n)}{\sin(\pi/n)}$$

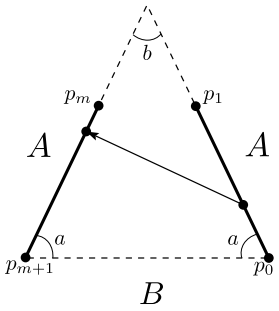


Fig. 4. A bounce from edge p_0p_1 to edge p_mp_{m+1}. The other edges of the polygon are not drawn.

The angle a can be found by considering the polygon formed by edges p_0p_1 through p_mp_{m+1}, closed by edge $p_{m+1}p_0$. This polygon has $m+2$ vertices, so its angle sum is $m\pi$. m of these vertices have the vertex angle of the regular n-gon, $(n-2)\pi/n$. The remaining two vertices have angle a. Therefore:

$2a + m(n-2)\pi/n = m\pi$

So $a = m\pi/n$.

And thus A:

$$A = \frac{l\sin(\frac{\pi(m+1)}{n})\sin(\frac{m\pi}{n})}{\sin(\frac{\pi}{n})\sin(\frac{\pi(n-2m)}{n})}$$

Then, using the triangle formed by the the bounce of the robot, and again the law of sines, we have:

$$\frac{A-x}{\sin(\theta-\pi/2+(2\pi m)/n)} = \frac{A-l+f_{\theta,m}(x)}{sin(\pi/2-\theta)}$$

Solved for $f_{\theta,m}$ and rewritten:

$$f_{\theta,m}(x) = \frac{(x-A)\cos(\theta)}{\cos(\theta+\pi(n-2m)/n)} + l - A$$

Observation: When $m = 1$ (agent skips no edges while bouncing around polygon), A reduces to l, and the expression for $f_{\theta,1}(x)$ reduces to $f_\theta(x)$ as previously derived.

Corollary: When $mk = n$, for some integer k, A becomes:

$$A = \frac{l\sin(\frac{\pi}{k}+\frac{\pi}{n})\sin(\frac{\pi}{k})}{\sin(\frac{\pi}{n})\sin(\pi-\frac{2\pi}{k})} = l\left(\frac{\tan(\frac{\pi}{k})}{\tan(\frac{\pi}{n})}+1\right)$$

VI. SIMULATION

The figures and experimental simulations for this paper were computed using a program written in Haskell and relying heavily on the excellent *Diagrams* library [8]. **todo: write up on numerical precision**

The simulator is also quite general, and could be of use to those studying classical billiards, or variants such as pinball billiards. It is also capable of simulating random bounces, or random noise on top of a deterministic bouncing law. Code is open source and on GitHub [1].

[1] `https://github.com/alexandroid000/bounce`

TABLE I. An Example of a Table

One	Two
Three	Four

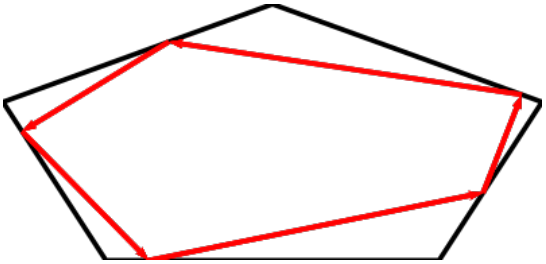


Fig. 5. Stable limit cycles exist in polygons with fewer symmetries than regular polygons.

VII. DISCUSSION

We are inspired by work on map dynamics in polygons such as [9] and [10], and it is possible that similar techniques from projective geometry could be applied to this dynamical system.

In non-regular polygons, we cannot solve for orbits as the fixed point of one mapping function. Yet, limit cycles still exist in polygons with enough symmetry, as seen in Figure 5.

APPENDIX

Appendixes should appear before the acknowledgment.

ACKNOWLEDGMENT

REFERENCES

[1] S. Tabachnikov, *Geometry and Billiards*. American Mathematical Society, 2005.

[2] R. Markarian, E. Pujals, and M. Sambarino, "Pinball billiards with dominated splitting," *Ergodic Theory and Dynamical Systems*, vol. 30, pp. 1757–1786, 2010.

[3] G. Del Magno, J. Lopes Dias, P. Duarte, J. P. Gaivão, and D. Pinheiro, "Srb measures for polygonal billiards with contracting reflection laws," *Communications in Mathematical Physics*, vol. 329, no. 2, pp. 687–723, 2014.

[4] B. Aronov, A. R. Davis, T. K. Dey, S. P. Pal, and D. C. Prasad, *Visibility with multiple reflections*. Berlin, Heidelberg: Springer Berlin Heidelberg, 1996, pp. 284–295. [Online]. Available: http://dx.doi.org/10.1007/3-540-61422-2_139

[5] J. S. Lewis and J. M. O'Kane, "Planning for provably reliable navigation using an unreliable, nearly sensorless robot," *International Journal of Robotics Research*, vol. 32, no. 11, pp. 1339–1354, September 2013.

[6] L. H. Erickson and S. M. LaValle, "Toward the design and analysis of blind, bouncing robots," in *IEEE International Conference on Robotics and Automation*, 2013.

[7] E. Jackson, *Perspectives of Nonlinear Dynamics:*. Cambridge University Press, 1992, vol. 1.

[8] B. Yorgey. (2017) Diagrams. Version 1.3. [Online]. Available: https://projects.haskell.org/diagrams

[9] W. P. Hooper and R. E. Schwartz, "Billiards in nearly isosceles triangles," *Journal of Modern Dynamics*, vol. 3, no. 2, pp. 159–231, June 2013.

[10] R. Schwartz, "The pentagram map," *Experiment. Math.*, vol. 1, no. 1, pp. 71–81, 1992. [Online]. Available: http://projecteuclid.org/euclid.em/1048709118