Qualitative Transitions in Object Reorienting Behaviour, Part 1: The Effects of Varying Friction

Graham E. Deacon, Mark Wright, and Chris Malcolm

DAI Research Paper No. 863

Appears in the Proceedings of IEEE International Conference on Robotics and Automation, pp.2697–2704, Albuquerque, New Mexico, USA, April 1997.

Qualitative Transitions in Object Reorienting Behaviour, Part 1: The Effects of Varying Friction.

Graham E. Deacon, Mark Wright[1] and Chris Malcolm

Department of Artificial Intelligence, University of Edinburgh,
5 Forrest Hill, Edinburgh, EH1 2QL, Scotland.

{ged, markwr, cam}@dai.ed.ac.uk

Abstract

In this and a companion paper [7] we investigate the topology of the space of possible pushing behaviour for 2.5D objects of arbitrary outline. We do this by studying the Push-Stability Diagram (PSD) derived by Brost, which is a generalisation of a result of Mason's. In this paper, we show how to derive a PSD for arbitrarily curved objects, and investigate how changes in the friction between pushing and pushed objects affect the topology of the PSD for any given object. This leads us to include an extra axis on the PSD, resulting in an Extended Push-Stability Diagram. This representation makes explicit the topological changes in the operation space of possible reorientation behaviours for any given 2.5D object, leading to the possibility of using it in the design of feeders.

1 Introduction

In recent years a number of researchers have studied the sliding behaviour of objects. Much of this work has been precipitated by Mason's Ph.D. [10] (see also [12]), where he derived a simple rule to determine the sense of reorientation of an object being pushed on a horizontal surface that is independent of the contact pressure distribution. Peshkin [15] considered all possible contact pressure distributions in order to derive conservative estimates on the rate of reorientation of an object. This, in conjunction with Mason's results, allowed him to deduce sequences of fences to be suspended over a conveyor belt so that objects fed at one end would leave at the other in a known orientation to within an interval of uncertainty. This interval of uncertainty was removed in [3] by curving the ends of the fences appropriately. Brost [4], [5] employed Mason's results in order to generate robust grasping strategies for polygonal objects. Goldberg [8] built on Brost's analysis and developed an algorithm that can generate a sequence of squeezing actions that can reorient a polygon from an initially unknown orientation to a known one (with a 180^o ambiguity) by exploiting a parallel jaw gripper with one jaw that is free to slide laterally. Rao and Goldberg [16] extended this algorithm to apply to parts containing algebraic curves. Recently [17] Goldberg's algorithm has also been adapted to synthesise orienting strategies for polygonal objects on conveyor belts fitted with the curved fences reported in [3]. Part of Brost's [4], [5] analysis included the derivation of the Push-Stability Diagram (PSD), which was exploited in [6] to derive optimal length plans to orient a polygonal object for a given value of friction between pushing and pushed objects. In what follows we show how to extend the PSD to arbitrarily shaped objects and how the coefficient of friction affects the topology of a PSD thereby influencing the length of pushing plans.

[1] Current address: Edinburgh Virtual Environment Centre, James Clerk Maxwell Building, The University of Edinburgh, Mayfield Road, Edinburgh, EH9 3JZ, Scotland. Email: Mark.Wright@ed.ac.uk

2 Background

2.1 Mason's Rule

The result of Mason's that precipitated this work we shall just refer to as Mason's rule. It presupposes a quasi-static model of manipulation where Coulomb's law of dry friction applies. A geometric representation of Coulomb friction is the friction cone [13]. If μ is the coefficient of friction, then the friction cone at a point contact is formed by the set of vectors that make an angle θ with the normal at the point of contact, where $\tan\theta = \mu$. Since we will be considering essentially planar tasks, the friction cone will be represented by its projection onto a plane parallel to the surface of sliding.

Mason's rule can be stated as follows: Let R_l and R_r be the left and right rays delimiting the edges of the friction cone, and R_p the ray denoting the direction of pushing. These rays vote relative to the CoM to determine which way the object will rotate; that is to say, if two or more of these rays lie to the left of the

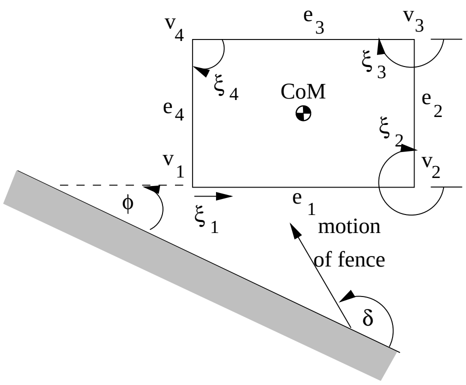


Figure 1: Labelling of a polygon and definition of the angles of orientation, ϕ, and pushing, δ.

line joining the point of contact to the CoM, then the object will rotate clockwise, if two or more of these rays lie to the right of this line, then the object will rotate anticlockwise. If one ray lies on the line from the point of contact to the CoM, and the other two are distributed one on each side of this line, pure translation occurs.

2.2 Constructing the PSD for Polygons

Brost's push-stability diagram [4], [5] represents, for a given value of the coefficient of friction between a fence and object, the outcome of a particular pushing operation given the initial orientation of the object, and the direction of pushing[2]. In essence it evaluates Mason's rule for all possible object orientations and pushing directions. In order to represent this graphically Brost constructs what he calls the *operation space* for the task by plotting pushing direction against object orientation.

In addition to Mason's assumption of quasi-static mechanics Brost assumes that the length of the pushing fence can be considered to be infinitely long. This simplifies the analysis since the behaviour of objects that possess concavities becomes equivalent to that of their convex hulls.

Figure 1 illustrates the labelling of a polygon to be pushed, and the senses in which the angle of pushing, δ, and angle of orientation, ϕ, are measured. The edges of the polygon are numbered e_1, e_2, $\ldots e_n$ (where n is the number of edges of the object's convex hull) in an anticlockwise direction starting from the longest edge. The vertices are numbered in a similar manner with vertex v_i on the anticlockwise end of edge e_i. In addition, each

[2]We ought to mention that the Edge Stability Map of Mani and Wilsons' [9], which was derived independently, is a version of the PSD and can be used to give the same information.

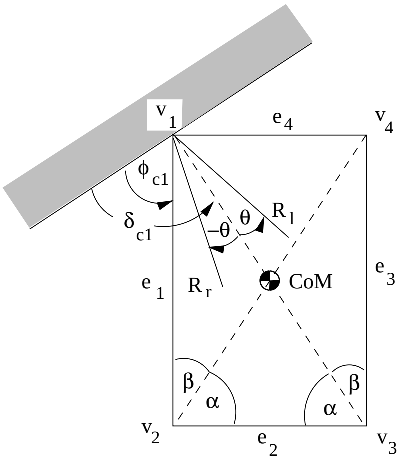


Figure 2: Geometry of a vertex in contact with a fence.

edge e_i has an edge-angle, ξ_i, which is the angle measured clockwise from e_1 to e_i. Note that ϕ is defined with respect to the edge e_1 of the polygon, and that δ is measured anticlockwise relative to the plane of the fence.

Brost's algorithm for constructing a PSD for a polygon is as follows. First the operation space is partitioned by the orientations at which an edge of the object's convex hull comes into contact with the pushing fence (these are just the values of the ξ_i). In between these orientations lie the ranges of orientations that some vertex is in contact with the fence. The result is further partitioned by the orientations at which the friction cone rays change their vote. If we imagine the object pivoting at the vertex of contact this occurs at the orientations when a friction cone ray lies along the line from the vertex to the CoM. As can be seen from Figure 2, this is a function of the interior angles of the object. Lastly the condition under which the pushing ray changes its vote is determined for each vertex contact configuration. This again is a function of the interior angles of the object. For the contact configuration shown in Figure 2 this condition is $\delta - \phi = \beta$.

The resulting partitioning is shown in Figure 3. The sense of the votes for each region is tallied resulting in the PSD shown in Figure 4.

The usefulness of the PSD stems from the fact that it encodes the "manipulation funnels" [11] or behavioural equivalence classes that are present in an object pushing task, that is to say that it makes explicit the range of initial conditions that will converge on the same outcome, that of a particular stable resting state. Stable resting states occur along the lines that have anticlockwise reorientation regions 'below' them, and clockwise reorientation regions 'above' them, since initial conditions on

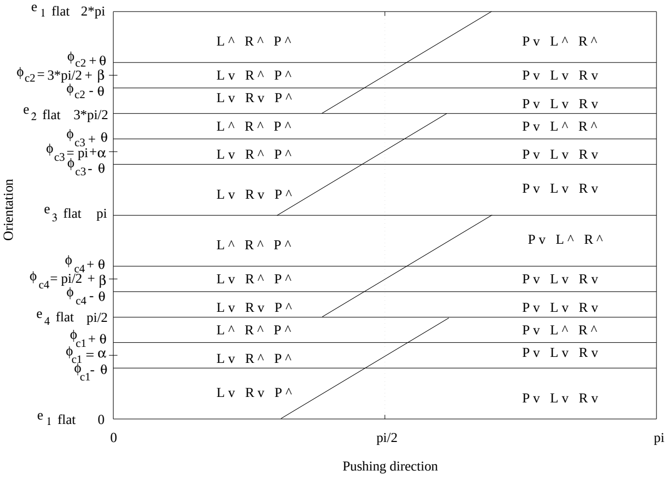


Figure 3: The partitioned operation space for reorienting a block by pushing.

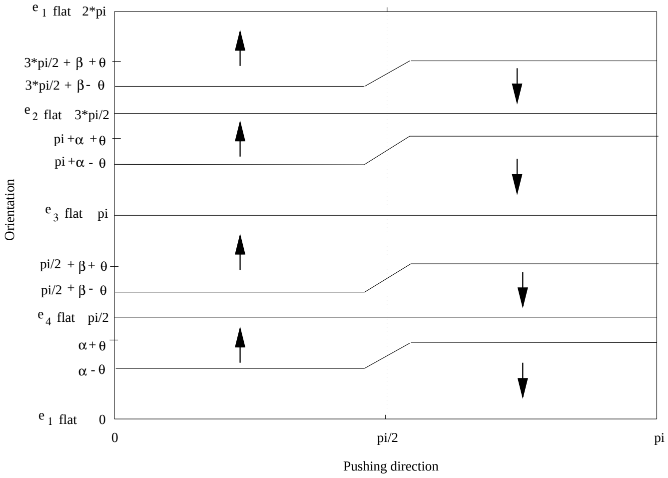


Figure 4: The Push-Stability Diagram for a rectangular block ($\theta < \beta$). Arrows indicate the sense of reorientation.

either side of these lines converge on the same orientation. Correspondingly these are known as convergent boundaries. Divergent boundaries are those that have an anticlockwise reorientation above them and a clockwise reorientation region below.

3 Generalising the PSD

3.1 Objects of Arbitrary Outline

It is actually straightforward to generalise the PSD to objects of arbitrary outline. For a curved object with no vertices we can imagine traversing the object's outline and constructing the tangent at each point on its boundary. This represents the fence making contact at that point. We can also construct the normal at each point, and the friction cone rays would make angles of $\pm\theta$ to it. Finally, if we construct a line from the point of contact to the CoM we have defined the decision boundary about which the rays involved in Mason's rule must vote. We will call this the *vote transition boundary* and its orientation is measured with respect to the fence in an anticlockwise sense by the angle ψ. See for example Figure 5. With this construction it is straightforward to determine which reorientation sense the friction cone rays vote for by comparing the values of $90^o + \theta$ (for R_l) and $90^o - \theta$ (for R_r) to ψ. In addition, the angle ψ is the critical value at which the pushing direction changes its vote.

For objects that include vertices in their perimeter we can imagine traversing the curve until a vertex is reached, and then the tangent pivoting through a range

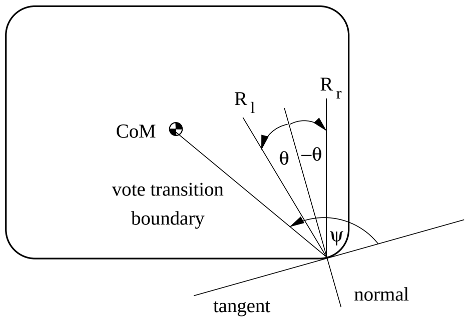


Figure 5: Reorientation sense voting for a curved object.

of orientations until it glances the next section of curve. We sample this range of orientations as the pivoting proceeds, and, as for curved sections of the perimeter, at each point determine (i) how the friction cone rays vote relative to the vote transition boundary, and (ii) the critical direction for the pushing ray.

For objects that include concavities we first find their convex hull using the algorithm described in [20][3], and then proceed as above.

At this point it will be useful to introduce what we will call the *vote transition curve*: If we plot on an ob-

[3]This algorithm, which uses a variation on the Hough transform called a pedal curve, implicitly delivers an object's radius function. (The radius function can be though of as the locus of the CoM as the object is rolled over the fence and appears as the basis of a number of parts orienting strategies [10], [8], [18], [17].) It is also straightforward with this technique to recover the diameter function used in [8], [16]. See [19].

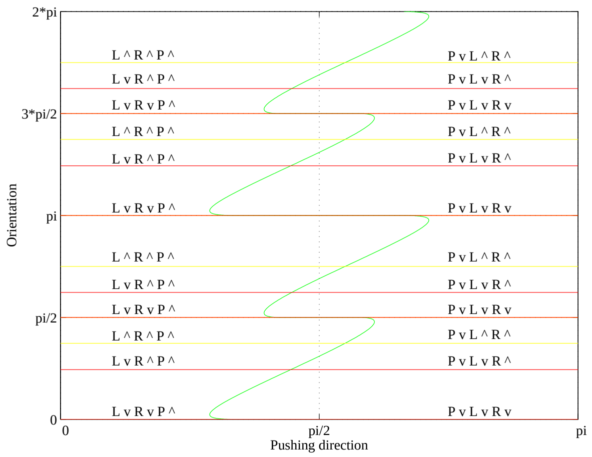


Figure 6: Decision boundaries on the PSD of a rounded rectangle.

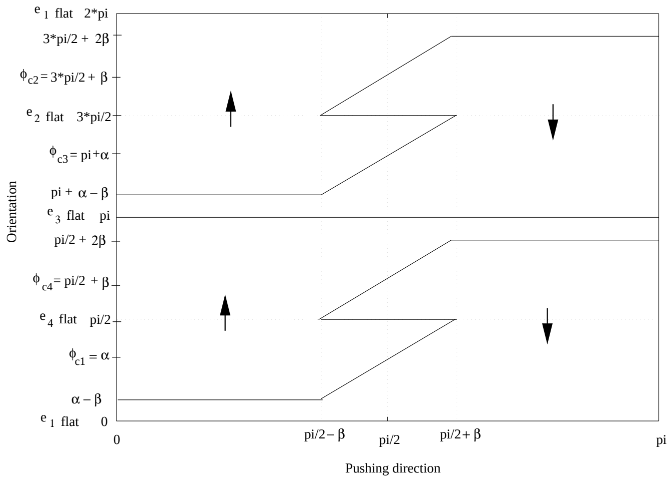


Figure 7: The PSD for a rectangular block with θ just greater than β. Arrows indicate the sense of reorientation.

ject's operation space the critical pushing directions that correspond to the pushing ray going through the CoM for each orientation of the object, *i.e.* we plot ψ against ϕ, we get a curve which is the locus of all the vote transition boundary directions. This is the vote transition curve. Figure 6 shows the vote partitioning of the operation space for the rounded rectangle of Figure 5. The vote transition curve appears as the curvy zig-zag running vertically through the middle. To the right of this curve the pushing ray votes for clockwise reorientation; to the left, anticlockwise. The vote transition curve also arises for polygonal objects, the difference being that for polygons it is piecewise linear; compare Figure 6 with Figure 3.

3.2 Variation in the Coefficient of Friction

3.2.1 Polygons

Figure 4 shows the PSD for a rectangular block, with $\theta < \beta$, where α and β are the angles between a line joining a vertex with the CoM and an edge of the block as shown in Figure 2. Note that for this example $\beta < \alpha$ and $\beta + \alpha = 90^o$. If we allow the friction cone half angle to increase until it is just greater than β we get the situation shown in Figure 7 where a change in the topology of the PSD occurs. If we allow θ to increase further until $\theta > \alpha$ then another change in the topology of the PSD occurs (Figure 8). Since these qualitative, topological changes affect the feeding strategies that can be synthesised from a PSD it is worth investigating why they occur.

Consider the situation shown in Figure 2. If we imagine the block rotating anticlockwise about v_1 then the condition for R_l to change its vote is when $\phi_{c1} = 90^o - \beta + \theta = \alpha + \theta$. But note that if $\phi_{c1} = \alpha + \theta = 90^o$ then R_l is about to change the sense of its vote when side e_4 is flat against the fence. So if $\alpha + \theta > 90^o$ then R_l can't change the sense of its vote until the block has rotated *into* the fence. Obviously this is physically inadmissible. What this means is that with $\alpha + \theta > 90^o$, (or equivalently $\theta > \beta$, since for this example $\alpha + \beta = 90^o$), R_l is 'unable' to change its vote, since the vote transition boundary always stays on the same side of R_l.

A similar situation occurs if we imagine the block pivoting clockwise about v_1. In this case the critical value of ϕ_{c1} for which R_r is about to change its vote is given by: $\phi_{c1} = 90^o - \beta - \theta = \alpha - \theta$. Now, if $\theta > \alpha$, R_r cannot change its vote until ϕ_{c1} is negative, *i.e.* the block has rotated into the fence. Again this is physically inadmissible, so if $\theta > \alpha$ then R_r is unable to change its vote.

From the parameterisation of Figure 4 we can also trace what happens to the PSD as the friction cone half-angle is allowed to increase. The v_1 contact interval corresponds to orientations between 0^o and 90^o on the PSD. Figure 4 shows that the horizontal segments of the divergent boundary in this region correspond to orientations of $\alpha + \theta$ and $\alpha - \theta$. As θ gets bigger we can imagine these horizontal lines getting further apart until $\alpha + \theta = 90^o$ (or $\theta = \beta$). At this point the anticlockwise turning region in the interval $\delta =]90^o + \theta, 180^o]$ has disappeared. This corresponds to the situation where R_l

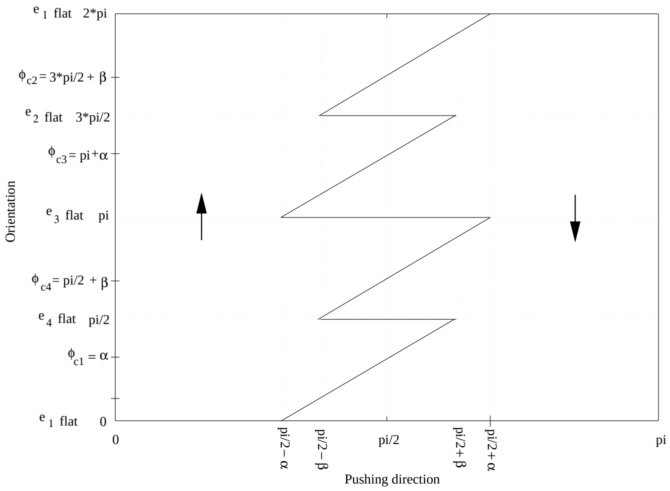


Figure 8: The PSD for a rectangular block with $\theta > \alpha$. Arrows indicate the sense of reorientation.

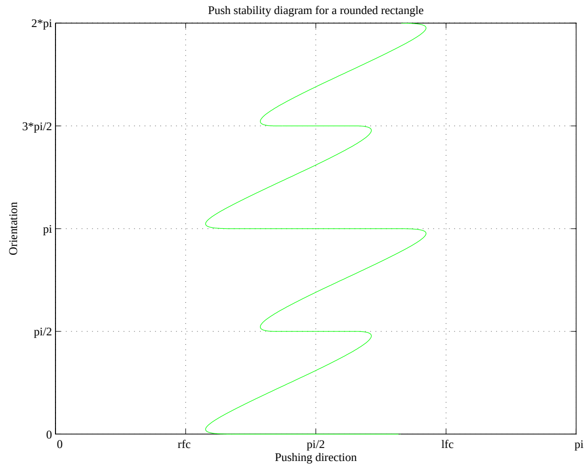
is no longer able to change its vote from clockwise to anticlockwise. If θ gets any larger part of the convergent boundary corresponding to e_4 being flat is 'annihilated'. Under these conditions, if the orientation of the block is just greater than ξ_4 and a pushing direction in the interval $\delta =]90^o + \beta, 180^o]$ is chosen, the block just rolls clockwise 'through' edge e_4 being flat gainst the fence, *i.e.* this interval no longer forms part of a convergent boundary.

If θ continues to increase then when $\theta > \alpha$ part of the convergent boundary corresponding to e_1 being flat is annihilated. This is the situation where R_r is unable to change its vote.

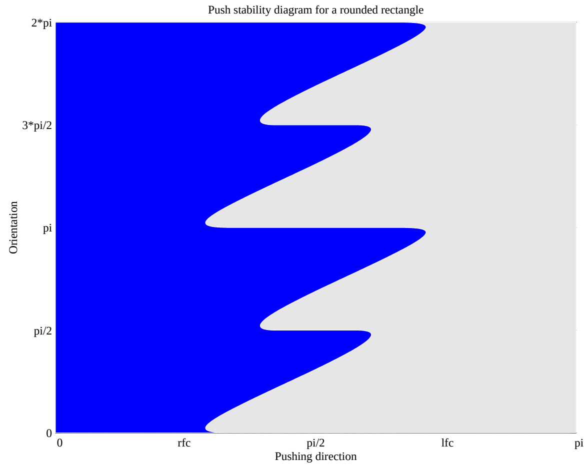
The reason why these 'annihilations' are symmetric about $\delta = 90^o$ can be seen by considering the behaviour of the object as it pivots about the vertices neighbouring v_1 on the PSD, *viz.* v_2 and v_4. In general these annihilations will not be symmetric but dependent on the interior angles of an object.

3.2.2 Curved Objects

Similar phenomena to that described above arise with curved objects. If we imagine traversing the perimeter of a curved object, just as before, the friction cone rays cannot change their respective votes if they always remain on the same side of the vote transition boundary. For R_l this would mean that it was always to the 'left' of the vote transition boundary, *i.e.* $90^o + \theta > \psi, \forall\phi$. For R_r this means always lying to the 'right' of the vote transition boundary, or $90^o - \theta < \psi, \forall\phi$. These conditions have a simple graphical representation on the PSD: We



(a)



(b)

Figure 9: PSD topology for a rounded rectangle with 'high' friction. Black shading denotes anticlockwise reorientation and grey clockwise.

can represent the right friction cone (*rfc*) ray on the PSD as the vertical line $\delta_{rfc} = 90^o - \theta$, and the left friction cone (*lfc*) ray as the line $\delta_{lfc} = 90^o + \theta$. (That is, δ_{rfc} is the pushing direction that coincides with the direction of the right friction cone ray. Similarly for the left friction cone ray and δ_{lfc}). If the ray R_l never changes its vote, then by definition δ_{lfc} never crosses the vote transition curve; similarly for R_r and δ_{rfc}. This situation is illustrated in Figure 9.

If we reduce the friction cone half angle, θ, from the situation shown in Figure 9 we can imagine the interval $\delta = [90^o - \theta, 90^o + \theta]$ getting thinner. Topological transitions in the PSD occur whenever δ_{lfc} or δ_{rfc} are tangent to the vote transition curve on the PSD, since this corre-

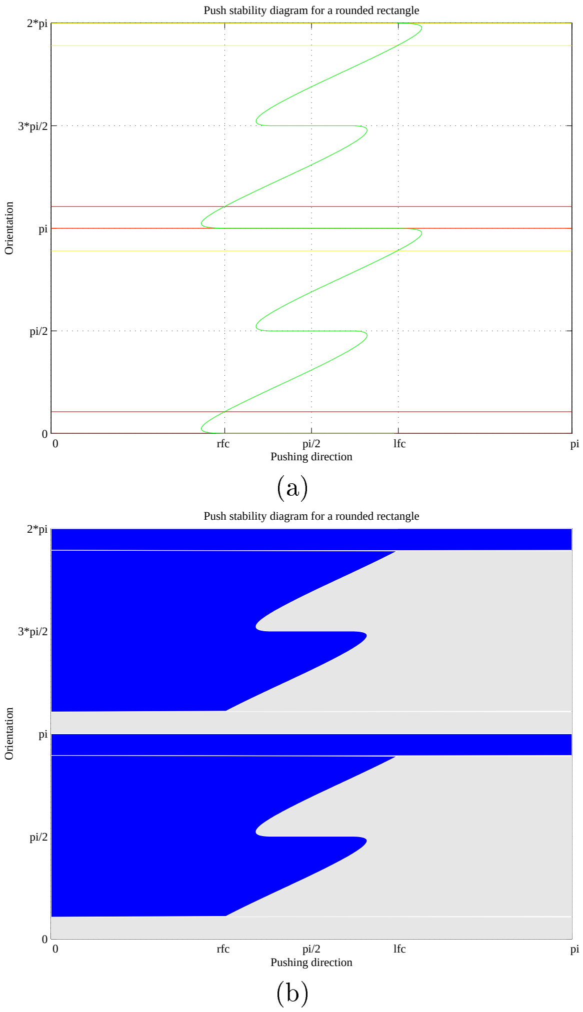


Figure 10: PSD topology for a rounded rectangle with 'medium' friction. Black shading denotes anticlockwise reorientation and grey clockwise.

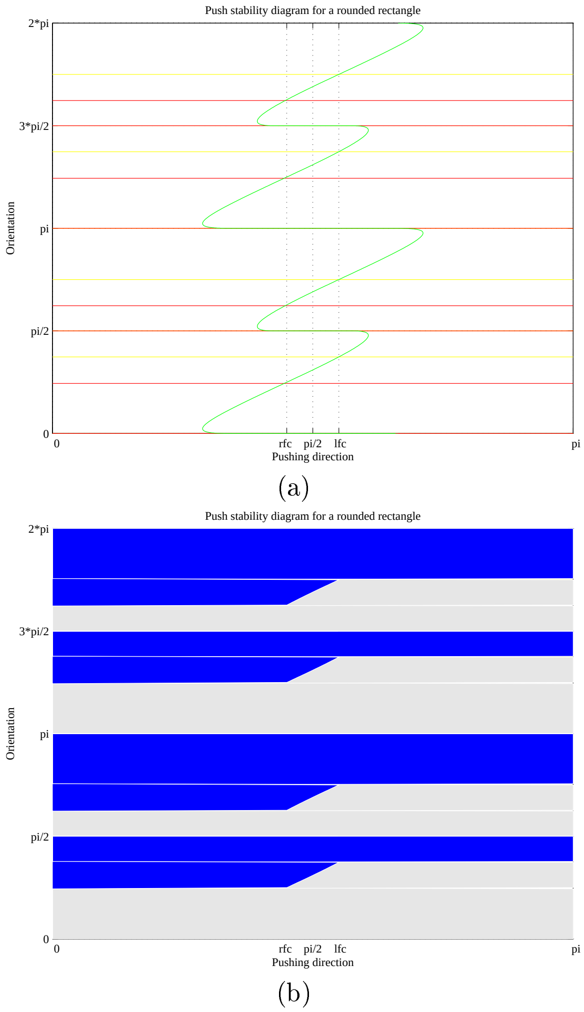


Figure 11: PSD topology for a rounded rectangle with 'low' friction. Black shading denotes anticlockwise reorientation and grey clockwise.

sponds with a situation where the relevant friction cone ray passes through the CoM. This occurs at the object orientation corresponding to the point of tangency. If the friction is reduced any further this allows the possibility of the relevant friction cone ray changing its vote. This phenomenon can be observed in Figures 10 and 11.

3.2.3 Polygons Revisited

We can use the vote transition curve to perform a similar analysis of the polygonal case, starting with a 'large' θ and gradually reducing it. If we consider the rectangular block of Figure 2, we see from Figure 8 that δ_{lfc} and δ_{rfc} would just touch the vote transition curve at $\delta = \frac{\pi}{2} \pm \alpha$. Also from Figure 8 we can see that this corresponds to the situations where either e_1 or e_3 is flat against the fence. If we imagine e_1 flat against the fence and a friction cone half-angle of $\theta = \alpha$, then for the friction cone at v_1, R_r passes through the CoM, and for the friction cone at v_2, R_l passes through the CoM, *i.e.* the critical point between being able to, and not being able to, change the voting sense. Reducing the friction further reveals a similar situation for edges e_2 and e_4 flat against the fence when $\theta = \beta$. This is consistent with our discussion in Section 3.2.1.

3.3 The Extended PSD

If we include a third axis to the PSD representing the magnitude of the coefficient of friction about the nor-

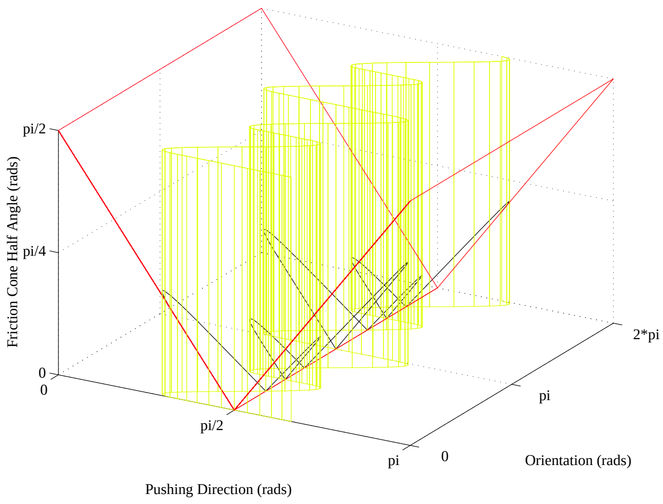


Figure 12: The Extended Push-Stability Diagram for a rounded rectangle showing the intersection of the vote transition curve with the friction cone planes.

mal to the fence, and extend the vote transition curve to exist over this whole range, we get a diagram that encodes all the possible distinct topologies of the PSD for a particular object for the whole range of possibilities of frictional coefficients. We will call this representation the Extended Push-Stability Diagram (EPSD). The EPSD for a rounded rectangle is shown in Figure 12. Horizontal slices across this diagram correspond to PSDs with particular friction coefficients. The turning points on the intersection of the extended vote transition curve and the friction cone planes occur at the values of friction where a topological transition in the PSD occurs due to a change in the value of the coefficient of friction. The critical values of friction are a function of the curvature of an object's perimeter and can be calculated from a differential geometry construction called the evolute [18]. This will be elaborated on in a companion paper [7].

4 Deriving PSDs for Objects of Arbitrary Outline

The foregoing discussion allows us to deduce a simple algorithm for constructing the PSD for arbitrary shaped objects.

First plot the vote transition curve on the operation space. To the left of this curve the pushing direction votes anticlockwise, and to the right clockwise, so colour to the left black (say), and to the right grey (say). Now include the horizontal lines corresponding to δ_{rfc} and δ_{lfc}. Where δ_{rfc} crosses the vote transition curve draw horizontal lines to the left and fill these bands with grey, since for all these orientations both friction cone rays lie to the right of the vote transition curve thus determining the sense of the vote as clockwise. Where δ_{lfc} crosses the vote transition curve draw horizontal lines to the right and colour these bands black, for similar reasons. The resulting black regions represent anticlockwise orientation, and the grey clockwise. (Compare Figures 6 and 11.)

5 Discussion

We have shown in this paper how variation in the coefficient of friction between the pushing and pushed objects affects the topology of the Push-Stability Diagram. This topology is also affected by the location of the CoM [7]. This is easy to see since the vote transition curve is defined with respect to the CoM. In addition, for certain locations of the CoM it is possible to lose extrema in the vote transition curve, corresponding to a change in the topologies possible in the PSD [18].

We previously derived an algorithm that takes a polygon from an initially random orientation to a known final orientation [6]. As it stands this algorithm is not applicable to PSDs with curved decision boundaries. To get round this we are considering representing PSDs as bitmaps in MATLAB and shifting more than one copy relative to each other in order to deduce orienting plans for arbitrarily shaped objects (*cf.* [9]).

Even if we are successful in deriving orienting plans for arbitrarily shaped objects this is still only half of the story: the position of the object remains unknown—after the last push in a plan it could lie anywhere along the fence. Mottaez and Goldberg [14] have proposed an iterative method to overcome this limitation to the degree of accuracy required. Alternatively a feeding mechanism that constrains two degrees of freedom could be used to remove the uncertainty in a further degree of freedom (*cf.* [1], [2]). We intend to look into this further in the future.

6 Summary

In this paper we have re-examined the construction of the Push-Stability Diagram, a graphical representation of an object's reorienting behaviour when pushed on a horizontal surface. We have shown how to construct this representation for objects of arbitrary outline, and by including an axis that represents the magnitude of the friction between the pushing and pushed objects, have shown how the effects of friction influence the possible reorientation behaviour of an object when pushed.

The fact that this variation in behaviour is made explicit means that it can be taken into account and used as a control parameter in the design of feeder mechanisms.

Acknowledgements

This paper was prepared whilst GED was employed on EPSRC grant GR/K 68271.

References

[1] Z. Balorda. Reducing uncertainty of objects by robot pushing. In *IEEE Int. Conf. on Robotics and Automation*, pages 1051–1056, Cincinnati, OH, May 1990.

[2] Z. Balorda. Automatic planning of robot pushing operations. In *IEEE Int. Conf. on Robotics and Automation*, pages 732–737, Atlanta, GA, May 1993.

[3] M. Brokowski, M. Peshkin, and K. Goldberg. Optimal curved fences for part alignment on a belt. *ASME J. of Mechanical Design*, 117:27–35, 1995.

[4] R.C. Brost. Planning robot grasping motions in the presence of uncertainty. Technical Report CMU-RI-TR-85-12, The Robotics Institute, Carnegie-Mellon University, Pittsburgh, PA, July 1985.

[5] R.C. Brost. Automatic grasp planning in the prescence of uncertainty. *Int. J. of Robotics Research*, 7(1):3–17, 1988.

[6] G.E. Deacon, P.L. Low, and C. Malcolm. Orienting objects in a minimum number of robot sweeping motions. In P.W.H. Chung, G. Lovegrove, and M. Ali, editors, *Proc. 6^{th} Int. Conf. on Industrial and Engineering Applications of Artificial Intelligence and Expert Systems*, pages 494–503, Edinburgh, Scotland, June 1993.

[7] G.E. Deacon, M. Wright, and C. Malcolm. Qualitative transitions in object reorienting behaviour, Part 2: The effects of varying the centre of mass. In *IEEE/RSJ 1997 Int. Conf. on Intelligent Robots and Systems*, Grenoble, France, September 1997.

[8] K.Y. Goldberg. *Stochastic Plans for Robotic Manipulation*. PhD thesis, Department of Computer Science, Carnegie-Mellon University, Pittsburgh, PA, August 1990.

[9] M. Mani and W.R.D. Wilson. A programmable orienting system for flat parts. In *Proc. North American Manufacturing Research Institute Conference XIII*, pages 427–432, Berkeley, CA, May 1985.

[10] M.T. Mason. *Manipulator Grasping and Pushing Operations*. PhD thesis, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, MA, June 1982.

[11] M.T. Mason. The mechanics of manipulation. In *IEEE Int. Conf. on Robotics and Automation*, pages 544–548, St. Louis, MO, March 1985.

[12] M.T. Mason. Mechanics and planning of manipulator pushing operations. *Int. J. of Robotics Research*, 5(3):53–71, 1986.

[13] J.L. Meriam. *Statics, 2^{nd} edition*. John Wiley and Sons, Inc., New York, NY, 1975.

[14] A.H. Mottaez and K.Y. Goldberg. Locating polygonal parts without sensors. In *SPIE Conf. on Sensors and Controls for Automated Manufacturing Systems*, Boston, MA, September 1993.

[15] M.A. Peshkin. *Planning Robotic Manipulation Strategies for Sliding Objects*. PhD thesis, Department of Physics and the Robotics Institute, Carnegie-Mellon University, Pittsburgh, PA, November 1986.

[16] A.S. Rao and K.Y. Goldberg. Manipulating algebraic parts in the plane. *IEEE Transactions on Robotics and Automation*, 11(4):598–602, 1995.

[17] J. Wiegley, K. Goldberg, M. Peshkin, and M. Brokowski. A complete algorithm for designing passive fences to orient parts. In *IEEE Int. Conf. on Robotics and Automation*, pages 1133–1139, Minneapolis, MN, April 1996.

[18] M. Wright and G.E. Deacon. Orienting 2.5D objects of arbitrary shape: A computational theory of planar orientation. (Unpublished) Working Paper, Department of Artificial Intelligence, University of Edinburgh, Edinburgh, Scotland, 1996.

[19] M. Wright, A. Fitzgibbon, P.J. Giblin, and R.B. Fisher. Beyond the Hough transform: Further properties of the $R\theta$ mapping and their applications. In J. Ponce, A. Zisserman, and M. Hebert, editors, *Object Representation in Computer Vision II*, pages 361–380. Springer Lecture Notes in Computer Science Vol. 1144, 1996.

[20] M. Wright, A. Fitzgibbon, P.J. Giblin, and R.B. Fisher. Convex hulls, occluding contours, aspect graphs and the Hough transform. *Image and Vision Computing*, 14(8):627–634, 1996.