Modular Robot Connector Area of Acceptance from Configuration Space Obstacles

Nick Eckenstein, Mark Yim

***Abstract*— Physical connectors which have geometry to passively guide mating increases robustness in attachments. This is a key design area for self-reconfigurable modular robots, which frequently make and break connections. By repurposing the interpretation of a well-known motion planning tool in configuration space obstacles for its encoding of contact geometry, we present a method for determining a metric of error tolerance (area of acceptance) in multiple dimensions by construction. Watershed analysis is run on this configuration-space model to determine the full shape and area of the capture region for the connector pair. We show the results of this process for several connector types.**

I. Introduction

Docking of two rigid objects occurs in many situations from the assembly of a peg in a hole to the docking of two satellites in orbit. In mid-air refueling, a jet receives fuel from a tanker plane via a probe on the front of the jet mating with a trailing drogue from the tanker [1]. In self-reconfigurable modular robots, the (dis)connecting capability differentiates the robot system [2].

Criteria for a good connector design depends on the application but can include: strength, information/power transfer capabilities, and alignment error tolerance. For self-reconfigurable modular robots, errors in joint positioning can cause the docking faces to be positioned unpredictably. Alignment errors are often corrected using active systems such as actuated latches [3].However, this actuation increases system complexity. To counteract this, the sloped geometry of mating faces such as chamfers can be designed to help passively guide the docking process. For example, in mid-air docking/refueling applications turbulence can make the docking positioning difficult. Larger mating features can accommodate larger errors and allow for lower precision control often leading to lower cost and more robust systems. Error tolerance must then be weighed against factors such as size and shape.

In previous work we introduced the concept of **area of acceptance**. Area of acceptance (AA) is defined as 'the range of possible starting conditions for which mating will be successful' [4]. In other words, AA is the full set of possible errors (for translation, rotation, and the two in combination) that the connector can accommodate and still mate successfully.

A. Related Work

Our previous work on maximizing area of acceptance focused on planar connectors [4]. Two geometries called V-Face and X-Face respectively and their maximum offsets are shown in Fig. 1. The V-face is a simple example of large chamfered mating surface. Note the X-Face is actually a '2.5D' connector with two layers, giving it a wider allowable offset.

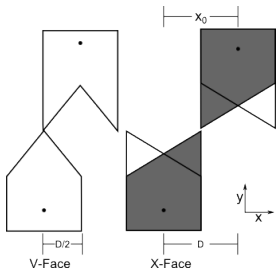


Fig. 1: Two-dimensional connector geometries.

In this work we propose a configuration-space approach applied to contact analysis. Configuration-space methods are common in robotics applications, often used to simplify collision checking in path planning or inform kinematic reasoning for a system. Accurate computation of this configuration space in these cases is important, but potentially computationally intensive. Techniques to quickly compute the bitmap representation of the configuration space have been developed using the FFT [5].Polygonal boundary representation is somewhat more common - it is more accurate at the small scale, and allows for analysis of contacts based on the shapes of the space formed. It is useful in situations where the space and robot do not lend themselves to discretization. An early but detailed survey of configuration space methods can be found in [6]. Of note from this survey is that out of the 27 papers surveyed, only one [7] successfully represented the full six dimensional C-space for a 3D polygonal robot.

In the contact space analysis arena Rimon et al. [8]use configuration space representations to form a first and second order mobility theory suited for robotic grasps. In a separate work from the same authors [9], planar objects are examined under a potential field for stability. However, these tests determine the stability rather than the capture region of the stable configurations. Capture regions in this context are the regions in configuration space in which a 3D object will fall to the same stable pose on a horizontal surface. This is similar to AA, but applied to one object on a plane rather than two docking objects. Kriegman [10] examines maximal capture regions with an assumption of dissipative dynamics.

In our analysis of 3D shapes, watershed algorithms can be used to identify maximal AA. Watershed algorithms are used in computer vision and image processing applications, in

Authors are with the GRASP Lab. and Dept. of Mechanical Engineering and Applied Mechanics, Univ. of Pennsylvania, Philadelphia, PA, USA. {neck, yim}@seas.upenn.edu

particular morphological segmentation. Often these methods are applied to pixel grids for edge detection in images or 3D meshes for contour/feature recognition [11]. A common strategy is to work from the minima up, 'flooding' the capture regions until the two sources meet, and then constructing a barrier or dam there. Other methods follow individual points along a path of steepest descent to determine which minimum they reach [12].

II. Background

The docking problem is the process of creating a physically rigid connection between two objects (e.g. robot modules). The problem can be broken into three parts: 1) *approach* from an arbitrary distance, 2) *contact*, sliding alignment and mating and 3) *attachment*. Without loss of generality we can consider one of the objects to be fixed and the other docking object moving relative to it. In the first two phases, the moving object moves in a constrained fashion where one degree of freedom (DOF) (e.g. the approach direction) has fixed velocity with the other DOFs compliant. For example, if we consider the six DOF of a robot module docking with a space station, one translational dimension is constrained to move such that the robot gets closer while the other translation and orientation DOF are free to move. Here, we will focus on the contact phase only.

In addition to this motion constraint we make further simplifying assumptions to aid analysis. We assume the directions which are free to move have motion that is quasi-static, fully damped, with no friction, no restitution, nor dynamic/inertial effects. This allows us to focus on the geometric features and just the initial offset conditions under which docking will succeed. These initial conditions are defined as an offset in the relevant DOF (rotation and translation) from the perfectly aligned state.

A. Configuration Space Representation

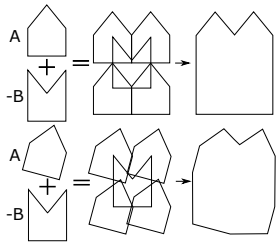


Fig. 2: C-obstacle construction by Minkowski difference, represented as A+(-B), for two different slices. The middle image shows a partial step. Top: θ=0, bottom: θ=$\frac{\pi}{12}$

In motion planning, the **configuration space** is the collision-free space of all possible states of a robot's DOF relative to obstacles in the environment. For a mobile robot moving on a plane amongst obstacles, the robot can be shrunk to a point and the obstacles concomitantly grown so that collision-free motion can be examined as a point moving through the space [13].

The configuration space representation is generated from the geometry of the robot and obstacle. For the docking case we will treat the two mating connectors as robot and obstacle. For example in the V-face connector the lower object (call it A) in Fig. 1 can be the robot and its mating connector (call it object B) can be the obstacle, resulting in Fig. 2.

A configuration space obstacle (C-obstacle) for a robot in the plane capable of only translation is generated by the Minkowski difference of the robot and obstacle. However, we wish to analyze the configurations including rotations. To do this, we can rotate our robot A successively through the range of possible rotations to generate 'slices' at each discrete rotation. In this case, we must rotate the objects about a particular point. The center of rotation is often unimportant in many analysis. As will be seen, the choice of rotation point cannot be ignored for our case.

The slices are then placed in a 3D space with the z-axis corresponding to the rotation angle theta. Finally, neighboring slices are joined into 3D layers to create an approximation of the C-obstacle. For efficiency we consider rotations bounded to regions where mating might be possible, i.e. $\frac{-\pi}{2}$ to $\frac{\pi}{2}$.

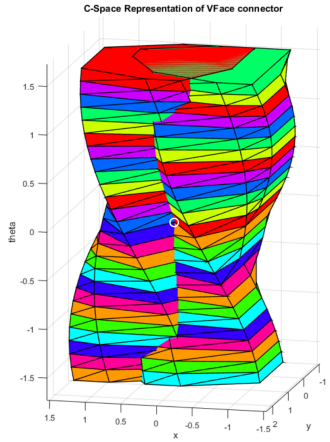


Fig. 3: Configuration space representation. Mating configuration indicated by a white circle.

III. Method Description

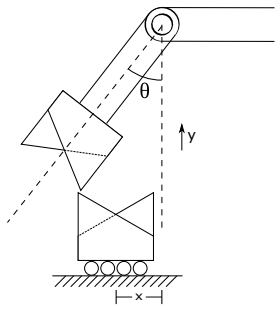


Fig. 4: Robot arm and connector setup

For our analysis, we will assume a planar connector attached to a robot arm and the mating connector on a movable base below it as in Fig. 4. The arm moves down in Y and the base is free to move in X. The arm is considered out of plane. Collisions are possible only between the two

connectors. We will define the approach direction to be in the negative Y direction, with translations in X and rotations in θ being unconstrained.

Fig. 5 shows an overview of the four steps to finding the area of acceptance. The first step examines the connector geometry. The second step finds the C-obstacle using the method described above. In the contact phase of the docking process, the alignment motion can be interpreted as motions of a point on the C-obstacle surface corresponding to the relative positions of the two objects in contact. Since the motions are constrained to have a negative Y component with no inertial effects, the direction of motions on *any* surface of the C-obstacle will be the projection of a line parallel to the Y axis going through the point of interest on the surface. In other words, if the negative Y direction was downward gravity, and the C-obstacle surface was a physically rendered surface, the path of a water droplet on the surface would be the path of the two objects in contact.

We seek to find the set of all points whose path ends at the target docked configuration. If the system has a non-trivial AA, the target docked configuration will be at a local minimum and there will be a neighborhood of connected areas that flow to this minimum. For example, in Fig. 3 a C-obstacle is shown for the V-Face connector. The target docked configuration indicated with a white circle is at a minimum in Y. The set of points to which water droplets would flow is termed a watershed [14].

In the final step, the area of acceptance is the projection of this watershed onto a plane perpendicular to Y. Computational geometric toolboxes aid our analysis. The

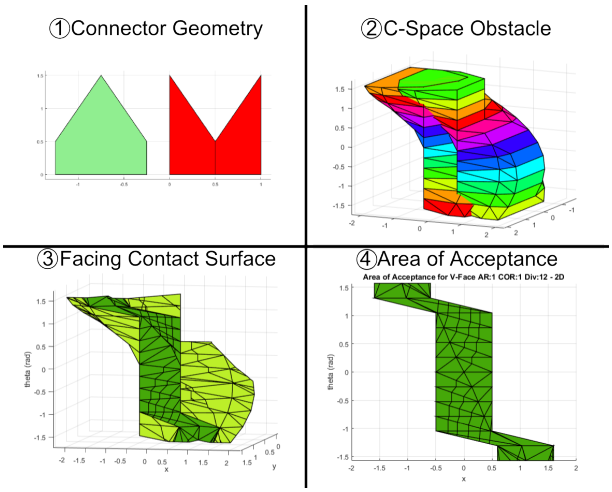


Fig. 5: Four of the steps to finding the Area of Acceptance

Multi-Parametric Toolbox contains functionality for representation, intersection, and Minkowski sum operations [15].

A. Design Parameterization

The area of acceptance domains we seek are offsets from the single straight-line direction of approach for mating. In the 2D connector case, the approach direction is y with offsets in the x and θ directions. The centers of rotation advance directly towards one another in the approach direction with offset values x and θ held constant until contact is made. Eventually the two connectors reach a stable configuration or fall away from each other completely. Stable configurations in the C-space are local minima, and we call the desired stable configuration the 'target docking configuration'.

In standard motion planning, θ axis geometry of the C-obstacles is made by rotating the robot and taking Minkowsi difference with the obstacle X, Y position fixed. In our case, the only reference that does not have a DOF arbitrarily free to move is the arm attached to the top mating connector (fixed in X and controlled in Y), thus it makes sense to use the center of rotation as the fixed reference frame and not a point on the obstacle. Unlike in motion planning, the center of rotation is now important to the C-obstacle shape and does impact watershed.

On the other hand, changes to reference in the X direction does not impact the watershed. There are two relative frames of reference for displaying the area of acceptance - the 'rotation frame' and the 'face frame'. Depictions of the two frames are shown in Fig. 6. Changing from 'rotation' to 'face' frame only shifts the X direction of each slice for better visualization of the AA. It does not change the Y values. Likewise it does not effect the total area of the AA.

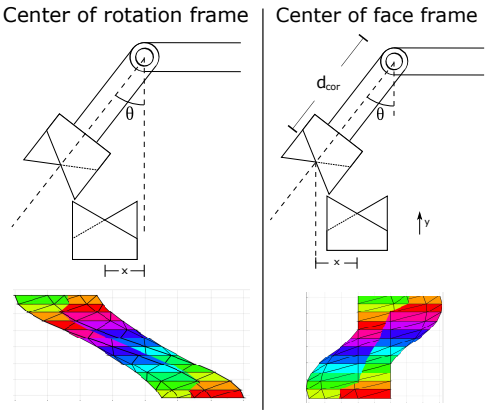


Fig. 6: 'Rotation frame' vs. 'Face frame'.

Two critical design parameters for planar faces are:

- *Aspect ratio of features* (AR), defined as $\frac{H}{D}$, where H is vertical difference between maximum and minimum points on the face, and D is width of the connector
- *Center of rotation distance* (COR), with magnitude of the distance from the center of the connector to center of rotation, normalized by the H: $COR = \frac{d_{cor}}{H}$.

COR can be positive or negative. Center of rotation points 'behind' the face of the connector are given negative values, whereas points 'in front' of the connector are given positive values. Changing the COR has a strong effect on the AA - more of the slices are uphill of the target for positive COR, which can create a larger watershed. In terms of contact, this means the robots with these COR far in front of the connector are more prone to rotate towards the target configuration. The opposite is true for COR far behind the connector.

IV. Watershed Algorithm

Once we have generated the C-obstacle as described in Section II-A, we reduce the C-obstacle to its outer boundary,

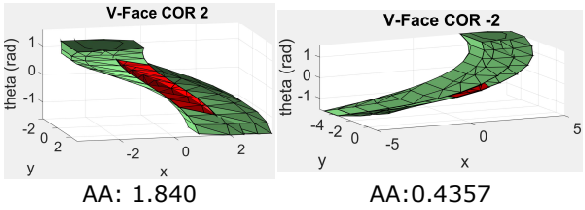
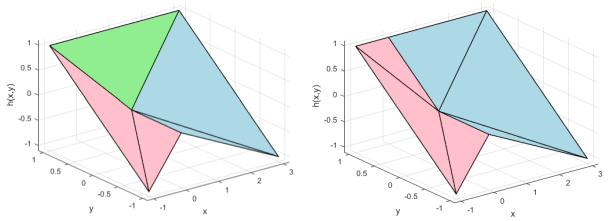


Fig. 7: Two configuration space obstacles for the V-Face connectors generated with same AR and different COR (+2/-2). AA is shown in red.

found by intersection of the obstacle and the closure of its complement. Since we know our target docking configuration, we can remove all parts below it, which cannot be in the watershed.

A. Pre-partitioning

One way to compute watersheds on grid data sets is by using flooding algorithms. Our representation is different from typical sets of vertices analyzed for watersheds in two ways: vertices are set in arbitrary convex polyhedra rather than on a grid, and the representation is likely to be sparse. This can lead to some incorrect watershed assignment with traditional flooding algorithms without the appropriate rules for construction of dams (Fig. 8).



(a) Naive assignment of vertices to watersheds by flooding

(b) Assignment using pre-partitioning

Fig. 8: Watershed assignment for a simple 'ridge' shape using flooding and pre-partition methods. The central green polyhedron in the first figure is indeterminate, as the vertices are assigned to multiple watersheds.

We present a new method for finding the watershed on convex polyhedra. First we pre-partition the set of polyhedra on the boundary such that each polyhedron has no more than one 'downhill' polyhedron. Each polyhedron is bounded and has a direction of steepest descent, and thus a (possibly empty) set of neighboring polyhedra to which points strictly descending along the polyhedra in this direction will go. To partition, we split polyhedra until each polyhedron has no more than one downhill polyhedron for all points inside it. Once partitioned, we can analyze each polyhedron as a whole rather than having to look at points independently.

To perform the splits, we find a polyhedron's shared edges with neighbors by intersection, and compare these to the set of 'downhill' edges. The downhill edges can be found by shifting any point on the edge some small ϵ along the direction of steepest descent. If this point is outside the polyhedron, the corresponding edge is a downhill edge. Downhill edges are divided according to which neighbor they intersect. The original polyhedron is then split into multiple polyhedra upwards along the direction of steepest *ascent*. Since this procedure can generate more polyhedra downhill of a previously split one, we repeat until no polyhedra remain which need be split. This method converges provided the polyhedra are a function on the downhill direction (each (x,y) corresponds to no more than one z).

B. Graph Traversal Algorithm for Watersheds

As each polyhedron leads to no more than one other, the set now resembles a graph structure known as a *directed pseudoforest*. Directed pseudoforests are defined as graphs in which each vertex has no more than one outgoing edge. For each polyhedron, we can follow the paths on this graph structure to find a watershed, with a few extra rules specific to the geometric conditions.

1) Ravine Condition: When two polyhedra point to one another, they have some line connecting them that must be 'downhill' of both. The points on these polyhedra then would normally flow down along this 'ravine' line until they reach another polyhedron. In the case where two polyhedra point to one another, we must go back to the geometry and find the new polyhedron they then point to. We call this a **ravine condition**. In order to determine the result of the ravine condition, we use the lowest point on the 'ravine' line $r_{\_}$. If this point is contained in one other polyhedron, the two polyhedra then have their corresponding outward graph edges reassigned to this new polyhedron.

It is possible that this point will be contained in more than one polyhedron and one must be chosen to proceed. We choose the polyhedron with the steepest downhill slope, as it represents the most likely direction for the motion to proceed in should some instability be introduced (as it often is in real cases).

2) Base Watersheds: After applying ravine conditions, certain geometric cases present **base watersheds** by having no further polyhedra to which they will flow. There are two cases in which we can declare a base watershed reached. The first case is one in which the polyhedron points to nothing, or equivalently the vertex on the graph has no outgoing edge. Polyhedra which satisfy this condition are assigned to a single 'outside' watershed. The second occurs when polyhedra point to each other in a loop containing more than two polyhedra. In this case we have reached a set around a single watershed point, and in this case these polyhedra are sides of a single minima.

Once the base watersheds and ravine conditions have been integrated into the graph, each polyhedron is assigned to a watershed as follows.

For each polyhedron, we traverse down the graph along corresponding nodes until we reach a base watershed, or a polyhedron (node) that is already assigned. We assign this polyhedron and each polyhedron along its path to the corresponding watershed.

Once all polyhedra have been assigned, we can determine which polyhedra make up the watershed corresponding to our target docking configuration. The set of polyhedra represents the 3D surface corresponding to the area of acceptance. To

find the final AA in the correct dimensions, we project this down into the free DOF - in the 2D case, x and θ.

V. Testing and Results

Two connectors - the V-Face and X-Face (Fig. 1) are used as examples. As a tool to evaluate connector shapes, the method is broadly applicable to any connector shape. We examine the two parameters mentioned earlier - (AR) and (COR).

These results are quantified as an area value in Tables I and II, with the plots showing the final shapes in Figures 9. Comparing to the results from the dynamic simulations performed in [4], we can see that most of the areas of acceptance are larger by a factor of up to two, as expected. Contrary to expectations, four values in the upper right section of the table are larger for V-Face than X-Face.

This may be due to the change from dynamic to quasi-static analysis. Several large patches reach critical points which in the dynamic case would pass into the area of acceptance, but in the quasi-static case do not. The results show that this method is capable of determining the area of acceptance within a certain level of accuracy.

Several trends are observable in terms of the design parameters. More remote (positive) COR tends to correspond to larger AA, as does smaller aspect ratio. We desire lower aspect ratio in order to keep connector size small for a given module width, so we find this result encouraging.

Some limitations to the method exist. Complexity scales up approximately with the number of slices we take in the rotational DOF (θ), and the number of connector edges. Geometrically complex connectors with lots of edges will take longer to analyze. The lower the number of slices taken, the less accurate the final representation of the configuration space and therefore the area of acceptance. The watershed algorithm however has the advantage of being complete on the polyhedral representation and preserves watershed regions that might be missed by flooding algorithms for lack of a vertex.

TABLE I: AA computed for V-Face

	COR:-1	COR:-1/2	COR:0	COR:1/2	COR:1
AR:1/4	0.38279	0.36586	0.85202	3.01069	3.14159
AR:1/2	0.40842	0.41129	0.86089	1.70012	2.77135
AR:1	0.43884	0.41643	0.87031	1.84022	3.14159
AR:2	0.45084	0.88724	0.84957	1.34941	3.14159
AR:4	0.46520	0.93465	0.88223	1.24182	3.14159

TABLE II: AA computed for X-Face

	COR:-1	COR:-1/2	COR:0	COR:1/2	COR:1
AR:1/4	1.09399	1.13635	1.28541	1.48352	1.84839
AR:1/2	0.97134	1.71965	1.99431	2.09141	2.27377
AR:1	0.92277	0.92203	1.79100	2.90494	3.13473
AR:2	0.88932	1.74696	1.66098	2.79853	4.09213
AR:4	0.89578	1.76955	1.65310	2.29481	4.65092

VI. Conclusions and Future Work

We have introduced a new method of analysis for the area of acceptance of mechanical connectors under alignment. The method makes use of the C-obstacle representation of the connector pair and a new watershed determination method to find the region of attraction also known as the **area of acceptance**. From the C-obstacle representation we present a pre-partitioning method that divides the polyhedra making up the boundary such that each one has one 'downhill' polyhedra. We can then represent the set of polyhedra as a traversable graph.Following the graph path leads each polyhedra to its watershed assignment. We performed this method on two different connectors with a variety of design parameters. The result is a metric we can use to compare the connectors. This information allows us to make informed decisions about connector design.

We have already used this method to generate multiple C-obstacles for pairs of 3D connectors. In future, we seek to expand the method to 3D connectors, and find the AA metric from the corresponding 6D C-space obstacles. We may also seek to extend this work to generating geometries with maximized areas of acceptance for use as across platforms.

References

[1] W. R. Williamson, E. Reed, G. J. Glenn, S. M. Stecko, J. Musgrave, and J. M. Takacs, "Controllable drogue for automated aerial refueling," *Journal of Aircraft*, vol. 47, no. 2, pp. 515–527, 2010.

[2] M. Yim, Y. Zhang, and D. Duff, "Modular robots," *IEEE Spectrum*, vol. 39, no. 2, pp. 30–34, 2002.

[3] E. H. Østergaard, K. Kassow, R. Beck, and H. H. Lund, "Design of the atron lattice-based self-reconfigurable robot," *Autonomous Robots*, vol. 21, no. 2, pp. 165–183, 2006.

[4] N. Eckenstein and M. Yim, "The x-face: An improved planar passive mechanical connector for modular self-reconfigurable robots," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2012, pp. 3073–3078.

[5] L. E. Kavraki, "Computation of configuration-space obstacles using the fast fourier transform," *IEEE Transactions on Robotics and Automation*, vol. 11, no. 3, pp. 408–413, 1995.

[6] K. D. Wise and A. Bowyer, "A survey of global configuration-space mapping techniques for a single robot in a static environment," *The International Journal of Robotics Research*, vol. 19, no. 8, pp. 762–779, 2000.

[7] B. Donald, "On motion planning with six degrees of freedom: Solving the intersection problems in configuration space," in *Robotics and Automation. Proceedings. 1985 IEEE International Conference on*, vol. 2. IEEE, 1985, pp. 536–541.

[8] E. Rimon and J. W. Burdick, "A configuration space analysis of bodies in contact i. 1st order mobility," *Mechanism and Machine Theory*, vol. 30, no. 6, pp. 897–912, 1995.

[9] R. Mason, E. Rimon, and J. Burdick, "A general stability test for multiply contacted objects in a potential field, with example applications to planar objects under gravity," *IEEE Transactions on Robotics and Automation*, 1996.

[10] D. J. Kriegman, "Let them fall where they may: Capture regions of curved objects and polyhedra," *The International Journal of Robotics Research*, vol. 16, no. 4, pp. 448–472, 1997.

[11] E. Zuckerberger, A. Tal, and S. Shlafman, "Polyhedral surface decomposition with applications," *Computers & Graphics*, vol. 26, no. 5, pp. 733–743, 2002.

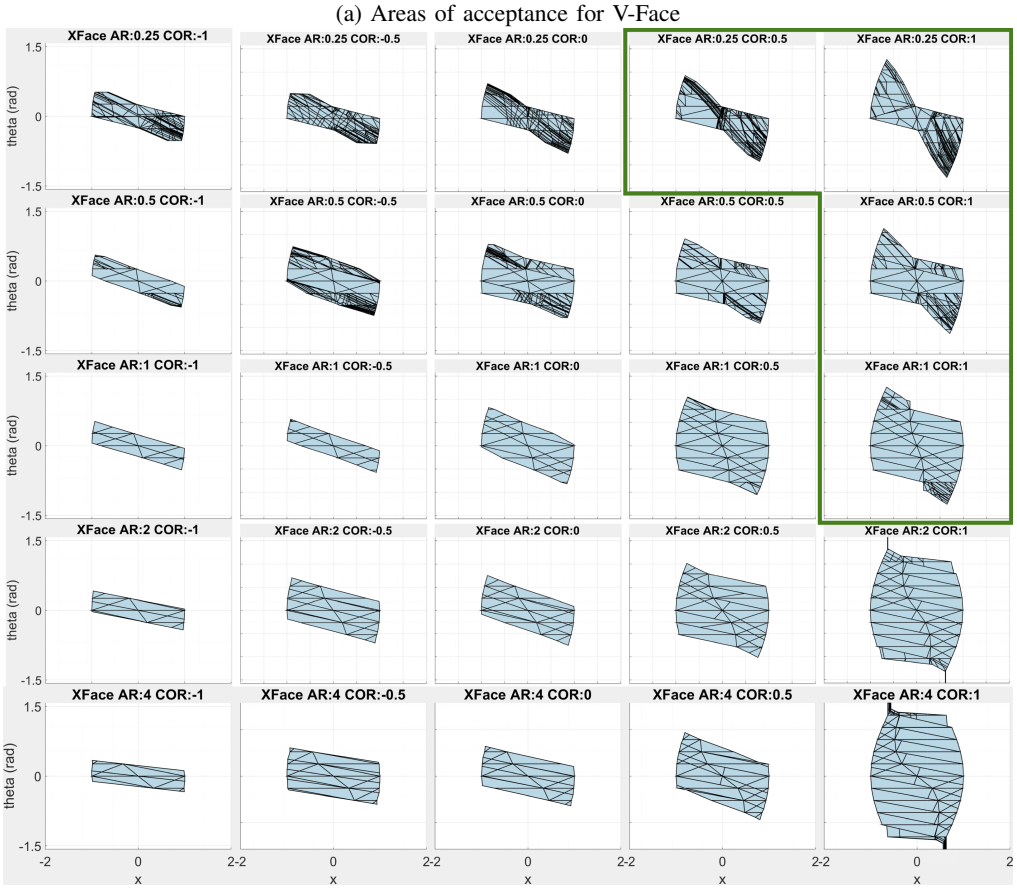
[12] A. P. Mangan and R. T. Whitaker, "Partitioning 3d surface meshes using watershed segmentation," *IEEE Transactions on Visualization and Computer Graphics*, vol. 5, no. 4, pp. 308–321, 1999.

[13] T. Lozano-Perez, "Spatial planning: A configuration space approach," in *Autonomous robot vehicles*. Springer, 1990, pp. 259–271.

[14] J. Cousty, G. Bertrand, L. Najman, and M. Couprie, "Watershed cuts: Minimum spanning forests and the drop of water principle," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 8, pp. 1362–1374, 2009.

[15] M. Herceg, M. Kvasnica, C. Jones, and M. Morari, "Multi-parametric toolbox 3.0," in *Proceedings of the European control conference*, no. EPFL-CONF-186265, 2013.

(a) Areas of acceptance for V-Face



(b) Areas of acceptance for X-Face

Fig. 9: Results of the method. All AAs are shown with 'center of face' frame of reference for clarity.