Proceedings of the *SPIE International Symposium on Intelligent Systems and Advanced Manufacturing: Modeling, Simulation, and Control Technologies for Manufacturing*, Philadelphia, October, 1995

Sensor configuration for error detection and recovery

Amy J. Briggs *

Department of Mathematics and Computer Science
Warner Hall
Middlebury College
Middlebury, VT 05753

ABSTRACT

Much of the early work in robotics focused on developing guaranteed plans for accomplishing tasks specified at a high level. Such task specifications might be of the form "mesh these two gears", or "place part A inside region B". It is not always possible, however, especially in the realm of assembly planning, to generate guaranteed plans. For example, errors in tolerancing of the parts might render an assembly infeasible. The Error Detection and Recovery (EDR) framework of Donald was developed to deal with these inadequacies of the guaranteed planning framework. EDR strategies will either achieve a goal if it is recognizably reachable, or signal failure. Given a geometrically-specified goal region G, an EDR strategy involves computing a failure region H and a motion plan that will terminate recognizably either in G or H.

The question addressed in this work is that of computing sensing strategies for distinguishing which of G and H have been attained. We propose a method with which we can strengthen the guarantee of reaching G or H into a guarantee of recognizability. In particular, we show how to configure a sensor or set of sensors so that a target object in G or in H can be distinguished. Our approach assumes a general sensor model, and builds on algorithms for computing partial visibility maps based on point-to-point visibility between objects in an environment. We characterize recognizability and confusability regions, that is, sensor placement regions from which an object in G or in H can be distinguished, and regions from which attainment of G or H could be confused.

Keywords: Error detection and recovery, sensor configuration, sensor planning, polygon visibility, uncertainty

1 INTRODUCTION

This paper examines the problem of sensor configuration planning within the Error Detection and Recovery framework developed by Donald.[10] An *Error Detection and Recovery* (EDR) strategy is one that is guaranteed to reach the goal when the goal is recognizably reachable, and signals failure otherwise. Given a start region and a motion strategy θ, an EDR planner finds a failure region H such that H is recognizably reachable, but such that

*Support for this work was provided in part by the National Science Foundation under grants No. IRI-8802390, IRI-9000532, IRI-9201699, and by a Presidential Young Investigator award to Bruce Donald, and in part by the Air Force Office of Sponsored Research, the Mathematical Sciences Institute, Intel Corporation, and AT&T Bell laboratories. The author was additionally supported by an AT&T Bell Laboratories Graduate Fellowship sponsored by the AT&T Foundation.

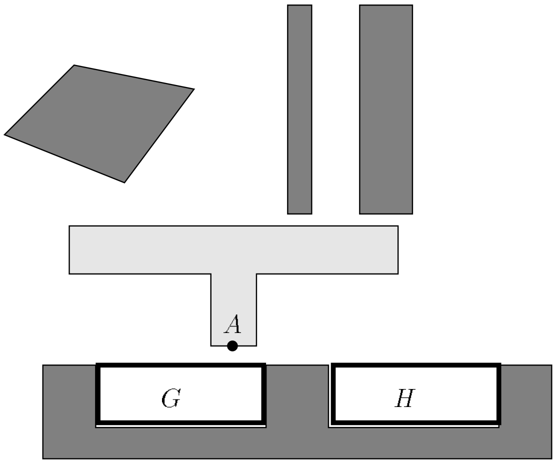

Figure 1: An example setup for the problem of sensor configuration in EDR. A represents the target; its reference point is indicated by the black dot. G is the goal region and H is the failure region. The darkly shaded polygons are obstacles. The problem is to find a sensor placement from which $A \in G$ and $A \in H$ can be distinguished.

it is impossible to enter G from H. The EDR framework guarantees that under generalized damper dynamics[1], the robot will eventually reach G or H. Furthermore, having entered G or H, it will never leave. Given this guarantee of reachability, we wish to strengthen it to a guarantee of recognizability: we want to know which of G and H has been attained. We will show how a sensor can be configured so that it can distinguish between the robot in G and the robot in H. Figure 1 gives an example of the problem we would like to solve.

The Error Detection and Recovery framework was developed to deal with the inherent *uncertainty* in robot tasks. Uncertainty in robot planning arises from errors in robot control and sensing, and in the geometric models of task environments. Our strategies for computing sensor configurations take into account errors in sensor placement and aim.

The remainder of the paper is organized as follows. In Section 3 we introduce our approach and define the notions of *recognizability* and *confusability*. Using point-to-point visibility as a model of detectability, we present in Section 4 our algorithms for computing recognizability regions for a target polygon at a known orientation in the plane. We show in Section 5 how we can restrict the set of sensor configurations so that readings that lead to confusion of G and H are avoided.

2 RELATED WORK

A different approach from the one taken in this paper to the incorporation of sensor planning in the EDR framework was first presented by Donald.[10] In that approach, an equivalence is established between sensing and motion in configuration space. *Active sensing* for a mobile robot is reduced to motion, by exploiting the similarity between visibility and generalized damper motions. In contrast, we present a framework that is closer to actual sensors. We rely on the visibility algorithms of Briggs,[2] which have been implemented and used with mobile robots in the lab (see Briggs[2,3]).

[1] The *generalized damper* is a dynamical model specified by the relationship $F = B(v - v_0)$ between forces and velocities, where F is the vector of forces and torques acting on a moving object, v_0 is the commanded velocity, v is the actual velocity, and B is a damping matrix. For more details, see Whitney,[30] Erdmann,[15] and Donald.[10]

Donald[11,12] has recently developed a theory of *information invariants* to characterize the information complexity of robot tasks. This work investigates the trade-offs between sensing, action, communication, and computation. Donald and Jennings[13,14] propose a framework of *constructive recognizability* to simplify robot programming in uncertain environments. In related work, Erdmann[16] proposes a method for automatically designing sensors based on the information requirements of a robot task. Jia and Erdmann[19] present algorithms and lower bounds for the problem of locating the minimum number of sensing points required to distinguish between a finite set of polygonal shapes.

The sensor placement problem has previously been addressed by Nelson and Khosla[26] and Kutulakos, Dyer, and Lumelsky[22] for visual tracking and vision-guided exploration. Several researchers have explored the problem of optimal sensor placement. Cameron and Durrant-Whyte[7] and Hager and Mintz[17] present a Bayesian approach to optimal sensor placement. Rimey and Brown[1,27] describe a task-oriented vision system for information gathering that uses Bayes nets.

Krotkov and Bajcsy[21] have addressed the problem of reliably measuring the three-dimensional position of objects in an unknown and cluttered scene. They use cooperative range recovery techniques to support mapping. Other researchers[25,20] have described methods for acquiring $3D$ information given range images obtained from a light stripe range finder.

Hutchinson[18] introduces the concept of a *visual constraint surface* to control motion. The idea is to combine position, force, and visual sensing in order to produce error-tolerant motion strategies. His work builds on that of preimage planners by adding visual feedback to compensate for uncertainty. Details on the implementation of vision-based control are described by Hutchinson and Castaño.[8]

Sharma and Hutchinson[28] define a measure of *robot motion observability* based on the relationship between differential changes in the position of the robot to the corresponding differential changes in the observed visual features. Lacroix, Grandjean, and Ghallab[23] describe a method for selecting view points and sensing tasks to confirm an identification hypothesis.

Cowan and Kovesi[9] study the problem of automatic camera placement for vision tasks. They consider the constraints on camera location imposed by resolution and focus requirements, visibility and view angle, and forbidden regions depending on the task. Given values bounding these constraints, they compute the set of camera locations affording complete visibility of a surface in $3D$. Zhang[31] considers the problem of optimally placing multiple sensors.

3 RECOGNIZABILITY AND CONFUSABILITY

By definition, an EDR plan achieves either G or H, and must be able to report which of G or H has been reached. We develop a method here of determining how a sensor should be configured so that this goal recognizability can be achieved. The basic idea is that the sensor should be positioned in such a way that attainment of the goal G can be recognized, attainment of the failure region H can be recognized, and attainment of G and H cannot be confused.[2] That is, given that we know target A is in $G \cup H$, we can determine which of G or H contains A.

The sensor we employ is an idealized but physically realizable model of a point-and-shoot sensor, such as a laser range finder. A sensor configuration is specified by a *placement* and a viewing direction, or *aim*. When in a particular configuration, the sensor returns a distance and normal reading to the nearest object, which is accurate to within some known bound. Such a ranging device has been developed in our robotics laboratory at Cornell,

[2] This is similar to the notion introduced by Buckley[6] of a *confusable set* in the context of motion planning. There, two contact points x and y are said to be *confusable* if they are capable of generating the same position and force measurements.

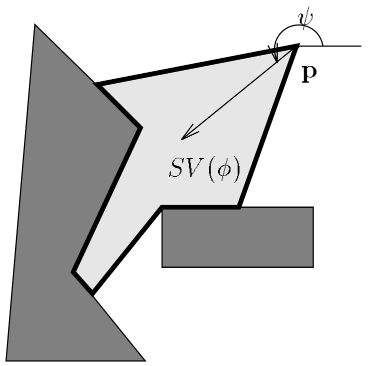

Figure 2: An example sensitive volume for a general sensor at configuration $\phi = (\mathbf{p}, \psi)$. The sensor is placed at position $\mathbf{p}$ and pointed in direction ψ. The sensitive volume is the lightly shaded region $SV(\phi)$. It is partly bounded by obstacles (darkly shaded).

and has been used for both map-making and robot localization. Details of how the sensor operates can be found in the work of Brown, Chew, and Donald.[4,5]

A translating target with two degrees of freedom and a fixed orientation has configuration space $\mathbb{R}^2$. We denote the space of sensor placements $C_{s_p} = \mathbb{R}^2$ and the space of sensor aims $C_{s_c} = S^1$. Our sensor configuration space is $C_s = C_{s_p} \times C_{s_c} = \mathbb{R}^2 \times S^1$. For a given sensor configuration $(\mathbf{p}, \psi)$, the sensor returns distance and normal readings for a subset of $\mathbb{R}^2$. We call this subset the *sensitive volume* of the sensor, and denote it by $SV(\mathbf{p}, \psi)$. Figure 2 illustrates an example of the sensitive volume for a general sensor at position $\mathbf{p} \in C_{s_p}$ and pointed in direction $\psi \in C_{s_c}$. In what follows, we restrict our attention to questions of visibility within the sensitive volume.

For a region X in the target object's configuration space, let $R(X)$ denote its *recognizability* region, that is, the set of all sensor placements from which the sensor can detect an object A in region X. Let $C(X,Y)$ denote the *confusability* region, that is, the set of all sensor placements from which the sensor cannot tell $A \in X$ and $A \in Y$ apart.

To guarantee goal recognizability for an EDR strategy, we wish to find a sensor placement $\mathbf{p} \in C_{s_p}$ such that $\mathbf{p} \in R(G) \cap R(H) - C(G,H)$. Figure 3 illustrates a case in which $A \in G$ and $A \in H$ may be confused.

4 VISIBILITY REGIONS

The main algorithmic paradigm we use in the computation of recognizability regions is point-to-point visibility. The notion we employ here is that of *partial visibility.*

DEFINITION 4.1. *A target polygon A is* partially visible *from a point p if at least one point in the closure of A is visible from p.*

The recognizability region of a stationary target polygon can be computed as the union of a set of visibility triangles. The approach is similar to that given by Suri and O'Rourke[29] for computing the region weakly visible from an edge. Our algorithm is as follows:

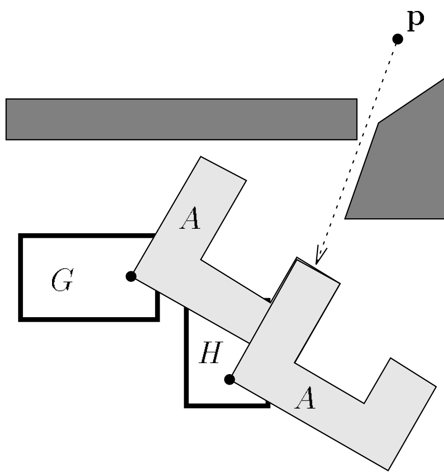

Figure 3: An example setup for the problem of confusability. A represents the target; its reference point is indicated by the black dot. G is the goal region and H is the failure region. The darkly shaded polygons are obstacles. $A \in G$ and $A \in H$ are confusable from sensor placement $\mathbf{p}$.

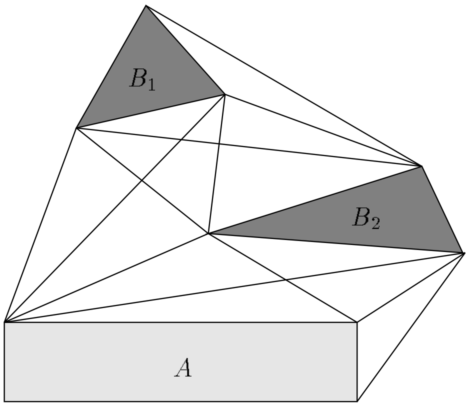

Figure 4: The visibility graph between target A and obstacles B_1 and B_2.

Partial visibility algorithm for a stationary target

1. Construct the visibility graph for the entire environment, consisting of target A and obstacles B. See Figure 4 for an illustration.
2. Extend each edge of the visibility graph maximally until both ends touch an edge of the environment. If neither of the endpoints of the extended visibility edge lie on the polygon A, discard the visibility edge. Otherwise, clip the edge at its intersection with A and call this piece a *visibility ray*. See Figure 5 for an illustration.
3. For each vertex v in the environment, perform an angular sweep of the visibility rays incident to v. If A remains visible to v throughout the swept angle between two adjacent visibility rays anchored at v, then the triangular swept region is output as a *visibility triangle.* Figure 6 gives an illustration.

The set of triangles can be computed in time $O(n(n+m))$ where n is the number of vertices in the environment and m is the number of vertices of A. To account for errors in sensor aim, we disallow narrow wedges of visibility.

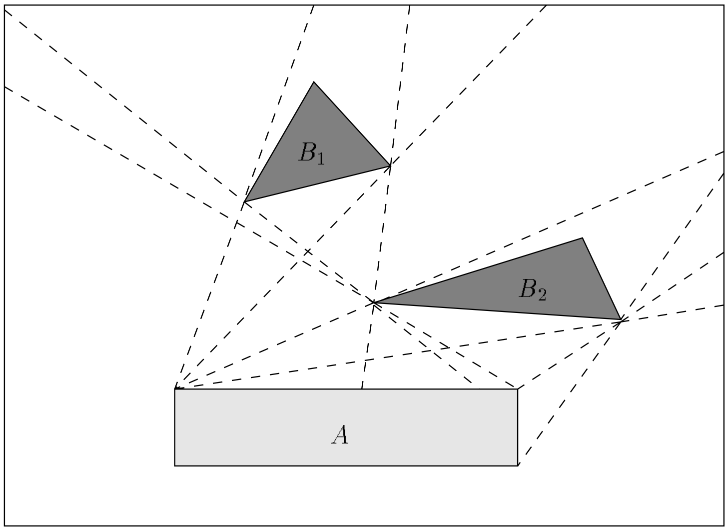

Figure 5: The visibility rays between target A and obstacles B_1 and B_2 are shown dashed.

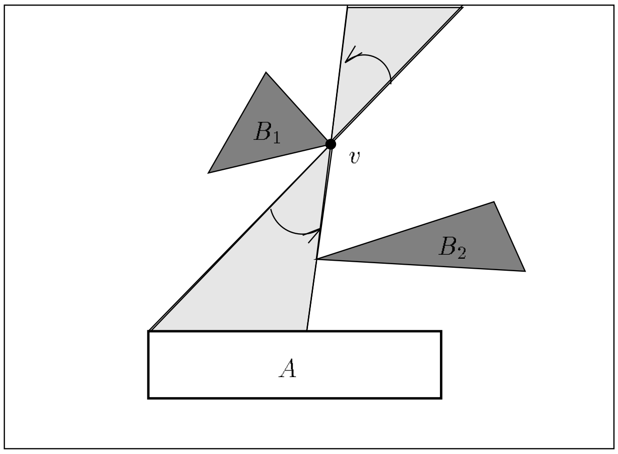

Figure 6: An angular sweep between two visibility rays at vertex v. The lightly shaded regions are visibility triangles.

This is accomplished by combining adjacent visibility triangles that end on the same polygon. The maximal union of adjacent visibility triangles anchored on a single vertex v and ending on the same polygon is called a *visibility polygon*. By construction, visibility polygons are simple. The *core triangle* of a visibility polygon anchored at v is the maximal inscribed triangle whose apex is v. If the angle at the apex of such a maximal visibility triangle is less than our angular uncertainty bound ϵ, we discard the polygon. Otherwise, we classify the maximal visibility triangle as an *ϵ-fat triangle*. After this processing, we now have $O(n(n+m))$ fat visibility triangles.

The boundary of the visibility region is formed by the union of the triangles, which can be shown to have complexity $O(n(n+m))$ using a result of Matoušek *et al.*[24] on the union of fat triangles. In the absence of directional uncertainty, the union can be computed in time $O(n^2(n+m)^2)$.

To account for errors in sensor placement, we shrink the visibility region by an uncertainty ball of radius ϵ_{pos} where ϵ_{pos} is the worst-case positional uncertainty of the sensor.

To detect a target that could be at any position within a goal polygon, we compute the set of all placements from which the target can be seen at any position within the goal. The computation of the visibility region in this case is an incremental one. The visibility region is formed by sweeping the target object through the goal and maintaining the visibility region as the visibility triangles change.

In order to detect the target as it enters the goal, we must compute the shadows swept for all positions of the target in the goal. We define a *swept shadow* of the goal in the weak visibility model to be a maximal connected region of C_{s_p} such that for each point $\mathbf{p}$ in the region, there exists a configuration of the target in the goal from which the target is totally occluded.

We compute swept shadows for the target at a fixed orientation anywhere in the goal by translating the target polygon along the edges of the goal. The boundary of a swept shadow is composed of obstacle segments and the curves (lines and conics) traced by free vertices. Discontinuities in the boundary of a swept shadow occur at *critical events*. We characterize the critical events as follows:

1. A moving visibility ray becomes aligned with a fixed edge of the visibility graph.
2. A free vertex of a shadow intersects an obstacle edge or the bounding polygon.
3. Two moving visibility rays bounding a shadow become parallel.

Partial visibility algorithm for a translating target

1. Let e be any edge of goal G. Consider A to be placed on one of the endpoints of e. Call this configuration $\mathbf{q}$. Construct the partial visibility region of target A at configuration $\mathbf{q}$.
2. Translate A along e. As the shadows cast by the obstacles change, call the area swept out by a shadow a *swept shadow*. Between critical events, the vertices of each shadow move along lines or conics. The equations of these curves can be computed algebraically given the positions of the obstacles in the environment and the visibility rays. Update the boundary of the swept shadows at critical events.
3. Translate A along all other edges e_i, $1 \leq i \leq k$, of G, repeating step 2 for each edge.
4. Compute each swept shadow independently as described in the above steps. The complement of the union of all the swept shadows, the target, and the obstacles is the partial visibility region.

The output of the algorithm is the set of swept shadows. Note that the boundary of a swept shadow is piecewise linear and conic.

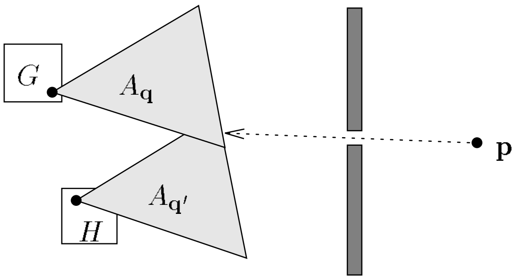

Figure 7: A sensor reading that confuses $A_{\mathbf{q}} \in G$ and $A_{\mathbf{q}'} \in H$ is due to an edge of $A_{\mathbf{q}}$ being colinear with an edge of $A_{\mathbf{q}'}$. The darkly shaded rectangles are obstacles.

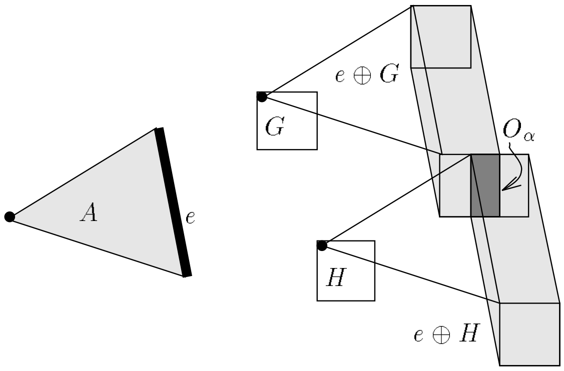

Figure 8: Edge e at orientation α of target A is convolved with G and H. The darkly shaded region is the overlap O_α. Sensor readings in O_α can lead to confusion of G and H.

The total running time of the algorithm for a goal with k vertices is $O(kmn^3(n+m))$ (see Briggs[2] for a detailed analysis).

5 CONFUSABLE REGIONS

The set $C(G, H)$ is the set of all sensor placements that could lead to confusion of G and H. A placement $\mathbf{p}$ is in the confusable region if the only visible portion of the target polygon could be due to an edge of A in G or an edge of A in H.

Note that a sensor reading that confuses a target $A_{\mathbf{q}}$ in G with a target $A_{\mathbf{q}'}$ in H is due to an edge of $A_{\mathbf{q}}$ being colinear with an edge of $A_{\mathbf{q}'}$. See Figure 7 for an example.

For each pair of edges (e_i, e_j) having the same orientation, we compute the *overlap* region $O(e_i, e_j) = (e_i \oplus G) \cap (e_j \oplus H)$. We define O_α to be the union of all $O(e_i, e_j)$ for all pairs of edges (e_i, e_j) having orientation α. See Figure 8.

The confusable region is defined as

$$C(G, H) = \{\mathbf{p} \mid \exists \mathbf{q} \in G \cup H, \forall \psi : (SV(\mathbf{p}, \psi) \cap A_{\mathbf{q}}) \subseteq O(e_i, e_j) \text{ for some } (e_i, e_j).\}.$$

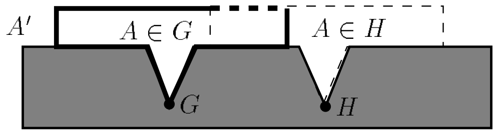

Figure 9: The set of thick solid edges comprises A' for $A \in G$. The dashed line outlines the target polygon in the failure region H. The thick dashed line is the overlap region.

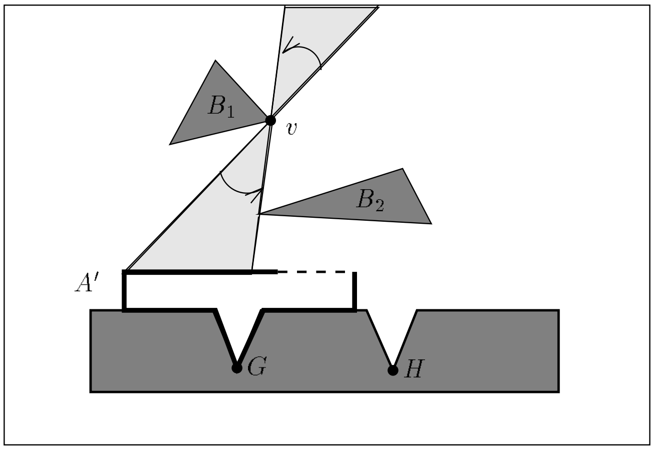

Figure 10: An angular sweep between two visibility rays at vertex v. The lightly shaded regions are visibility triangles. The thick solid edges comprise A', and the dashed line is the overlap region.

5.1 Discrete goal and failure regions

Before turning to the problem of handling polygonal goal and failure regions, we first consider the case in which the goal and failure regions are discrete points. Our technique for computing the set of good sensor placements is to first compute the set of overlap regions, and then compute the recognizability regions for the non-overlap portion of A in G and the non-overlap portion of A in H. The algorithm is as follows:

1. Compute all overlap regions $O(e_i, e_j)$ for all pairs of edges (e_i, e_j) having the same orientation. Note that in the case of point-sized goal and failure regions, the overlap regions consist of edge segments.
2. Perform the following steps for A in G and A in H:
   (a) Construct a new target A' by deleting the overlap segments from A. Figure 9 illustrates the overlap for a point-sized goal and failure region. The new target consists of a set of edge segments, where each edge segment has an associated outward-facing normal, so it is visible only from one side.
   (b) Compute the set of visibility triangles for target A' using the *partial visibility algorithm for a stationary target* as described in Section 4. Figure 10 gives an illustration of some visibility triangles.
   (c) Compute the union of the visibility triangles formed above. This is the partial visibility region for the non-overlap portion of A at this configuration.
3. Compute the intersection of the two visibility regions computed for A in G and A in H in steps 2(a)–2(c) above. This gives the set of all sensor placements from which both A in G and A in H can be recognized, but not confused.

5.2 Polygonal goal and failure regions

In the case of polygonal goal and failure regions, the computation of $R(G) \cap R(H) - C(G, H)$ is an incremental one. Recall that each overlap region is due to an edge of A in G being colinear with an edge of A in H. In this case, the overlap region $(e_i \oplus G) \cap (e_j \oplus H)$ for parallel edges e_i and e_j is formed by a line sweeping through a region determined by G and H.

To determine the set of placements from which G and H can be distinguished but not confused, we do the following:

1. Compute the set of overlap regions O_α for all orientations α of the edges of A.
2. Place A at a vertex of G. Let $A'(\mathbf{q}) = A_\mathbf{q} - (A_\mathbf{q} \cap (\cup O_\alpha)$ be the set of edge segments of A at configuration $\mathbf{q}$ not intersecting any O_α.
3. Compute the partial visibility region of $A'(\mathbf{q})$ as it sweeps through G, as described in Section 4. Note that the endpoints of the edges of $A'(\mathbf{q})$ are not fixed, but vary during the sweep.
4. Repeat steps 2. and 3. for A sweeping through H.
5. Take the intersection of the regions computed for A sweeping through G and A sweeping through H.

The resulting region is the set of all placements from which A at any position in $G \cup H$ can be detected, but $A \in G$ and $A \in H$ can not be confused.

6 CONCLUSION

The *Error Detection and Recovery (EDR)* system of Donald[10] provides a framework for constructing manipulation strategies when guaranteed plans cannot be found or do not exist. An EDR strategy attains the goal when the goal is recognizably reachable, and signals failure otherwise. The algorithms presented in this paper extend the guarantee of reachability to a guarantee of recognizability for the case of a polygon translating in the plane. We have presented algorithms for computing the set of sensor placements affording partial visibility of a stationary target, and partial visibility of a target translating through a goal at a known orientation. The algorithms account for uncertainty in sensor placement and aim. Our approach does not yet address the problem of sensor measurement error. In future work we plan to address the problem of planning sensing strategies when the target polygon may translate and rotate, resulting in unknown orientations of the target in G and H.

Acknowledgements

Thanks to Bruce Donald, Mike Erdmann, Tomás Lozano-Pérez, and Matt Mason for introducing me to the problems addressed in this paper and sharing their ideas on them. Thanks especially to Bruce Donald and Daniel Scharstein for many discussions and insightful comments on this work.

7 REFERENCES

[1] A. Blake and A. Yuille, editors. *Active Vision*. The MIT Press, 1992.

[2] A. J. Briggs. *Efficient Geometric Algorithms for Robot Sensing and Control*. PhD thesis, Department of Computer Science, Cornell University, Ithaca, NY, January 1995.

[3] A. J. Briggs. A robot surveillance system using automatic camera control. Submitted to 1996 IEEE International Conference on Robotics and Automation, September 1995.

[4] R. G. Brown. *Algorithms for Mobile Robot Localization and Building Flexible, Robust, Easy to Use Mobile Robots*. PhD thesis, Department of Computer Science, Cornell University, Ithaca, NY, May 1995.

[5] R. G. Brown, L. P. Chew, and B. R. Donald. Localization and map-making algorithms for mobile robots. In *Proceedings of the International Association of Science and Technology for Development (IASTED) International Conference on Robotics and Manufacturing, Oxford, England*, pages 185–190, September 1993.

[6] S. J. Buckley. *Planning and Teaching Compliant Motion Strategies*. PhD thesis, MIT Artificial Intelligence Laboratory, 1987.

[7] A. J. Cameron and H. Durrant-Whyte. A Bayesian approach to optimal sensor placement. *International Journal of Robotics Research*, 9(5):70–88, 1990.

[8] A. Castaño and S. Hutchinson. Hybrid vision/position servo control of a robotic manipulator. In *Proceedings of the IEEE International Conference on Robotics and Automation, Nice, France*, pages 1264–1269, May 1992.

[9] C. K. Cowan and P. D. Kovesi. Automatic sensor placement from vision task requirements. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10(3):407–416, May 1988.

[10] B. R. Donald. *Error detection and recovery in Robotics*, volume 336 of *Lecture Notes in Computer Science*. Springer-Verlag, Berlin, 1989.

[11] B. R. Donald. Information invariants in robotics, parts I and II. In *Proceedings of the 1993 IEEE International Conference on Robotics and Automation, Atlanta, GA*, 1993.

[12] B. R. Donald. On information invariants in robotics. Technical Report TR 93–1341, Department of Computer Science, Cornell University, 1993.

[13] B. R. Donald and J. Jennings. Constructive recognizability for task-directed robot programming. *Journal of Robotics and Autonomous Systems*, 9(1):41–74, 1992.

[14] B. R. Donald and J. Jennings. Constructive recognizability for task-directed robot programming. In *Proceedings of the IEEE International Conference on Robotics and Automation, Nice, France*, May 1992.

[15] M. A. Erdmann. On motion planning with uncertainty. Technical Report MIT–AI–TR 810, MIT Artificial Intelligence Laboratory, 1984.

[16] M. A. Erdmann. Understanding action and sensing by designing action-based sensors. In *Proceedings of the Sixth International Symposium on Robotics Research, Hidden Valley, PA*, October 1993.

[17] G. Hager and M. Mintz. Computational methods for task-directed sensor data fusion and sensor planning. *International Journal of Robotics Research*, 10(4):285–313, August 1991.

[18] S. Hutchinson. Exploiting visual constraints in robot motion planning. In *Proceedings of the IEEE International Conference on Robotics and Automation, Sacramento*, pages 1722–1727, April 1991.

[19] Y.-B. Jia and M. Erdmann. The complexity of sensing by point sampling. In *Workshop on the Algorithmic Foundations of Robotics, San Francisco*, February 1994.

[20] K. Kemmotsu and T. Kanade. Sensor placement design for object pose determination with three light-stripe range finders. In *Proceedings of the 1994 IEEE International Conference on Robotics and Automation, San Diego, CA*, pages 1357–1364, May 1994.

[21] E. Krotkov and R. Bajcsy. Active vision for reliable ranging: Cooperating focus, stereo, and vergence. *International Journal of Computer Vision*, 11(2):187–203, October 1993.

[22] K. N. Kutulakos, C. R. Dyer, and V. J. Lumelsky. Provable strategies for vision-guided exploration in three dimensions. In *Proceedings of the 1994 IEEE International Conference on Robotics and Automation, San Diego, CA*, pages 1365–1372, May 1994.

[23] S. Lacroix, P. Grandjean, and M. Ghallab. Perception planning for a multi-sensory interpretation machine. In *Proceedings of the IEEE International Conference on Robotics and Automation, Nice, France*, pages 1818–1824, May 1992.

[24] J. Matoušek, N. Miller, J. Pach, M. Sharir, S. Sifrony, and E. Welzl. Fat triangles determine linearly many holes. In *Proceedings of the 32nd Annual IEEE Symposium on Foundations of Computer Science*, pages 49–58, 1991.

[25] J. Maver and R. Bajcsy. Occlusions and the next view planning. In *Proceedings of the IEEE International Conference on Robotics and Automation, Nice, France*, pages 1806–1811, May 1992.

[26] B. Nelson and P. K. Khosla. Integrating sensor placement and visual tracking strategies. In *Proceedings of the 1994 IEEE International Conference on Robotics and Automation, San Diego, CA*, pages 1351–1356, May 1994.

[27] R. Rimey and C. Brown. Task-oriented vision with multiple Bayes nets. In A. Blake and A. Yuille, editors, *Active Vision*, chapter 14, pages 217–236. The MIT Press, 1992.

[28] R. Sharma and S. Hutchinson. On the observability of robot motion under active camera control. In *Proceedings of the 1994 IEEE International Conference on Robotics and Automation, San Diego, CA*, pages 162–167, May 1994.

[29] S. Suri and J. O'Rourke. Worst-case optimal algorithms for constructing visibility polygons with holes. In *Proceedings of the Second Annual ACM Symposium on Computational Geometry*, pages 14–23, 1986.

[30] D. Whitney. Force feedback control of manipulator fine motions. *Journal of Dynamic Systems, Measurement, and Control*, 99(2):91–97, June 1977.

[31] H. Zhang. Optimal sensor placement. In *Proceedings of the IEEE International Conference on Robotics and Automation, Nice, France*, pages 1825–1830, May 1992.