MOTION PLANNING AND CONTROL OF AUTONOMOUS VEHICLES
USING COLLISION AND RENDEZVOUS CONES

A Dissertation by
Vishwamithra Reddy Sunkara
Master of Science, Wichita State University, 2013

Submitted to the Department of Electrical Engineering and Computer Science
and the faculty of the Graduate School of
Wichita State University
in the partial fulfillment of
the requirements for the degree of
Doctor of Philosophy

July 2018
MOTION PLANNING AND CONTROL OF AUTONOMOUS VEHICLES
USING COLLISION AND RENDEZVOUS CONES

The following faculty members have examined the final copy of this dissertation for form
and content, and recommend that it be accepted in partial fulfillment of the requirement for
the degree of Doctor of Philosophy, with a major in Electrical Engineering and Computer
Science.

_________________________________________
Animesh Chakravarthy, Committee Chair

_________________________________________
John Watkins, Committee Member

_________________________________________
Edwin Sawan, Committee Member

_________________________________________
Vinod Namboodiri, Committee Member

_________________________________________
James Steck, Committee Member

_________________________________________
Zheng Chen, Committee Member

Accepted for the College of Engineering

_________________________________________
Steven Skinner, Interim Dean

Accepted for the Graduate School

_________________________________________
Dennis Livesay, Dean
ACKNOWLEDGMENTS

The satisfaction that accomplishes the successful completion of any task would be incomplete without the mention of the people who make it possible and whose constant guidance and encouragement crown all the efforts with success.

It is my privilege and pleasure to express my profound sense of respect, gratitude and indebtedness to my advisor Dr. Animesh Chakravarthy, Associate Professor, Wichita State University for his constant guidance, and constant encouragement throughout this research work.

I wish to express my deep gratitude to all my dissertation committee members for their cooperation and support.

I sincerely extend my thanks to National Science Foundation (NSF) for providing funding and support to the research work.

I am immensely indebted to my parents and my wife for their love and especially the unshakable belief in me.

Finally, I thank all those who directly and indirectly helped me in this regard.
This dissertation uses the notion of collision cones and rendezvous cones to address several motion planning problems for autonomous vehicles. Collision avoidance is fundamental to the problem of planning safe trajectories in dynamic environments. This problem appears in several diverse fields including robotics, air vehicles, underwater vehicles and computer animation. In the rendezvous problem, generating appropriate trajectories to achieve overlap of footprints of unmanned aerial vehicles is important in problems related to search and surveillance, and for establishing communication between a network of UAVs, or between a user and a base station in remote areas.

In the collision avoidance problem, much of the collision avoidance literature assumes shapes of the objects as circles. However, when objects are operating in closer proximity, or when objects are elongated and/or have non-convex shapes, a less conservative approach, that considers the exact shapes of the objects, is more desirable. This dissertation presents analytical collision avoidance laws in cooperative and non-cooperative dynamic environments. The collision avoidance laws are simulated on Ionic Polymer-Metal Composite (IPMC) actuated robotic fish. Collision cones are also used to analyze pursuit evasion games between two objects of arbitrary shapes. Collision avoidance of objects that can deform by changing their shape as a function of time is also presented.

The rendezvous problem requires communication/sensing footprints of vehicles to overlap. The need of the footprints to overlap is dictated by the requirement that no part of the sensed area is left uncovered in a search and surveillance operation; or by the need to position a relay UAV in the overlap region of two distant UAVs in order to enable them to communicate with each other. The concept of a rendezvous cone is used as the basis for the development of nonlinear analytical guidance laws that enable the overlap of footprints to the requisite depth.
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Chapter</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>1. INTRODUCTION</strong></td>
<td>1</td>
</tr>
<tr>
<td>1.1 Research Motivation</td>
<td>1</td>
</tr>
<tr>
<td>1.2 Research Objective</td>
<td>4</td>
</tr>
<tr>
<td>1.3 Literature Review</td>
<td>4</td>
</tr>
<tr>
<td>1.3.1 Collision Avoidance Laws</td>
<td>4</td>
</tr>
<tr>
<td>1.3.2 Cooperative Collision Avoidance and Formation Control</td>
<td>6</td>
</tr>
<tr>
<td>1.3.3 Ionic Polymer Metal Composite (IPMC) Actuated Robotic Fish</td>
<td>6</td>
</tr>
<tr>
<td>1.3.4 Pursuit Evasion Games Analysis</td>
<td>7</td>
</tr>
<tr>
<td>1.3.5 Collision Cones for Deforming Objects</td>
<td>8</td>
</tr>
<tr>
<td>1.3.6 Achieving Overlap of Footprints</td>
<td>9</td>
</tr>
<tr>
<td>1.4 Dissertation Organization</td>
<td>10</td>
</tr>
<tr>
<td><strong>2. COLLISION AVOIDANCE LAWS FOR OBJECTS WITH ARBITRARY SHAPES</strong></td>
<td>12</td>
</tr>
<tr>
<td>2.1 Collision Cone between Two Arbitrary Objects</td>
<td>12</td>
</tr>
<tr>
<td>2.2 Laws for Collision Avoidance between Two Arbitrary Objects</td>
<td>16</td>
</tr>
<tr>
<td>2.2.1 Collision avoidance law with $a_A$ as the primary input</td>
<td>19</td>
</tr>
<tr>
<td>2.2.2 Collision avoidance law with $\delta_A$ as the primary input</td>
<td>20</td>
</tr>
<tr>
<td>2.3 Analysis of Collision Avoidance Laws</td>
<td>22</td>
</tr>
<tr>
<td>2.3.1 Singularities</td>
<td>23</td>
</tr>
<tr>
<td>2.3.2 Effects of Time Delays</td>
<td>24</td>
</tr>
<tr>
<td>2.4 Simulation Results</td>
<td>26</td>
</tr>
<tr>
<td><strong>3. COOPERATIVE COLLISION AVOIDANCE AND FORMATION CONTROL FOR OBJECTS WITH HETEROGENOUS SHAPES</strong></td>
<td>32</td>
</tr>
<tr>
<td>3.1 Cooperative Collision Avoidance Laws</td>
<td>32</td>
</tr>
<tr>
<td>3.1.1 Cooperative Collision Avoidance Law with Acceleration Magnitudes as Primary Inputs</td>
<td>33</td>
</tr>
<tr>
<td>3.1.2 Cooperative Collision Avoidance Law with Acceleration Directions $\delta_A$ and $\delta_B$ as Primary Inputs</td>
<td>35</td>
</tr>
<tr>
<td>3.2 Combining Collision Avoidance with Formation Control</td>
<td>36</td>
</tr>
<tr>
<td>3.2.1 Formation Control</td>
<td>36</td>
</tr>
<tr>
<td>3.2.2 Integrating Collision Avoidance Laws with Formation Control</td>
<td>37</td>
</tr>
<tr>
<td>3.3 Simulation Results</td>
<td>39</td>
</tr>
<tr>
<td><strong>4. COLLISION AVOIDANCE BY IPMC ACTUATED ROBOTIC FISH USING COLLISION CONE APPROACH</strong></td>
<td>48</td>
</tr>
<tr>
<td>Chapter</td>
<td>Title</td>
</tr>
<tr>
<td>---------</td>
<td>-------</td>
</tr>
<tr>
<td>4.1</td>
<td>Robotic Fish Dynamic Model</td>
</tr>
<tr>
<td>4.1.1</td>
<td>Fish Body Dynamics</td>
</tr>
<tr>
<td>4.1.2</td>
<td>IPMC Fin Dynamics</td>
</tr>
<tr>
<td>4.1.3</td>
<td>Description of the Robotic Fish</td>
</tr>
<tr>
<td>4.2</td>
<td>Collision Cone</td>
</tr>
<tr>
<td>4.2.1</td>
<td>Guidance Law for Collision Avoidance</td>
</tr>
<tr>
<td>4.2.2</td>
<td>Conversion of commanded acceleration into fin thrusts</td>
</tr>
<tr>
<td>4.2.3</td>
<td>Integrated system architecture</td>
</tr>
<tr>
<td>4.3</td>
<td>Simulation Results</td>
</tr>
<tr>
<td>5.</td>
<td>Cooperative and Optimal Collision avoidance laws for IPMC/Servo-actuated robotic fish</td>
</tr>
<tr>
<td>5.1</td>
<td>Collision Avoidance Laws</td>
</tr>
<tr>
<td>5.1.1</td>
<td>Cooperative Collision Avoidance Laws</td>
</tr>
<tr>
<td>5.1.2</td>
<td>Optimal Cooperative Collision Avoidance Laws</td>
</tr>
<tr>
<td>5.1.3</td>
<td>Optimal Cooperative Collision Avoidance: Heading and Speed Change</td>
</tr>
<tr>
<td>5.1.4</td>
<td>Effects of Imperfect Information</td>
</tr>
<tr>
<td>5.2</td>
<td>2D Maneuverable Robotic Fish</td>
</tr>
<tr>
<td>5.2.1</td>
<td>2D Maneuvering Robotic Fish</td>
</tr>
<tr>
<td>5.2.2</td>
<td>State Space Model</td>
</tr>
<tr>
<td>5.3</td>
<td>Conversion of Collision Avoidance Commands into Actuator inputs for the Hybrid Tail</td>
</tr>
<tr>
<td>5.4</td>
<td>Simulation Results</td>
</tr>
<tr>
<td>6.</td>
<td>Pursuit Evasion Games using Collision Cones</td>
</tr>
<tr>
<td>6.1</td>
<td>Problem Formulation</td>
</tr>
<tr>
<td>6.2</td>
<td>Pursuit Evasion Game 1</td>
</tr>
<tr>
<td>6.3</td>
<td>Pursuit Evasion Game 2</td>
</tr>
<tr>
<td>7.</td>
<td>Collision Cones for Deforming Objects</td>
</tr>
<tr>
<td>7.1</td>
<td>Collision between a point object and a deforming object</td>
</tr>
<tr>
<td>7.1.1</td>
<td>Collision Conditions</td>
</tr>
<tr>
<td>7.1.2</td>
<td>Guidance Laws for Collision Avoidance</td>
</tr>
<tr>
<td>7.2</td>
<td>Collision between a circular object and a deforming object</td>
</tr>
<tr>
<td>7.2.1</td>
<td>Collision Conditions</td>
</tr>
<tr>
<td>Chapter</td>
<td>Page</td>
</tr>
<tr>
<td>----------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>7.2.2 Guidance Laws for Collision Avoidance</td>
<td>115</td>
</tr>
<tr>
<td>7.3 Collision between an arbitrarily shaped object and a deforming object</td>
<td>116</td>
</tr>
<tr>
<td>7.3.1 Collision Conditions</td>
<td>116</td>
</tr>
<tr>
<td>7.3.2 Guidance Laws for Collision Avoidance</td>
<td>118</td>
</tr>
<tr>
<td>7.4 Simulation Results</td>
<td>118</td>
</tr>
<tr>
<td>8. Achieving Overlap of Multiple Arbitrarily Shaped Footprints using Rendezvous Cones</td>
<td>123</td>
</tr>
<tr>
<td>8.1 Hyperbolic Geometry</td>
<td>123</td>
</tr>
<tr>
<td>8.2 Rendezvous cone for arbitrary shapes</td>
<td>124</td>
</tr>
<tr>
<td>8.2.1 Problem Formulation</td>
<td>124</td>
</tr>
<tr>
<td>8.2.2 Conditions under which $A'$ and $B'$ will make contact, while avoiding collision between $A$ and $B$</td>
<td>127</td>
</tr>
<tr>
<td>8.2.3 Conditions for overlap between $A'$ and $B'$</td>
<td>131</td>
</tr>
<tr>
<td>8.3 Guidance Law for Rendezvous of Arbitrary Shapes</td>
<td>138</td>
</tr>
<tr>
<td>8.3.1 Requirements for Rendezvous and Collision Avoidance</td>
<td>138</td>
</tr>
<tr>
<td>8.3.2 Partitioning of the $(r, \theta)$ space into sectors</td>
<td>140</td>
</tr>
<tr>
<td>8.3.3 Guidance Law Derivation</td>
<td>144</td>
</tr>
<tr>
<td>8.4 Simulation Results</td>
<td>148</td>
</tr>
<tr>
<td>8.4.1 Example 1: Stationary Leader</td>
<td>149</td>
</tr>
<tr>
<td>8.4.2 Example 2: Moving Leader</td>
<td>151</td>
</tr>
<tr>
<td>8.4.3 Example 3: Sweep Pattern</td>
<td>152</td>
</tr>
<tr>
<td>8.5 Further Extensions</td>
<td>156</td>
</tr>
<tr>
<td>9. Conclusion</td>
<td>157</td>
</tr>
<tr>
<td>Bibliography</td>
<td>159</td>
</tr>
</tbody>
</table>
## LIST OF TABLES

<table>
<thead>
<tr>
<th>Table</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1</td>
<td>Kinematic states at $t = 0$</td>
</tr>
<tr>
<td>4.1</td>
<td>Size information of IPMC and fish body</td>
</tr>
<tr>
<td>6.1</td>
<td>Results for Matrix Game 1: Game of Degree</td>
</tr>
<tr>
<td>6.2</td>
<td>Results for Matrix Game 1: Game of Kind</td>
</tr>
<tr>
<td>6.3</td>
<td>Results for Matrix Game 2: Game of Degree</td>
</tr>
<tr>
<td>6.4</td>
<td>Results for Matrix Game 2: Game of Kind</td>
</tr>
</tbody>
</table>
# LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>Circular Shape Approximation</td>
<td>2</td>
</tr>
<tr>
<td>1.2</td>
<td>Motion Planning in a Cluttered Environment</td>
<td>2</td>
</tr>
<tr>
<td>1.3</td>
<td>Examples of scenarios requiring overlaps of vehicle footprints: (a) Sweep operation (b) Maintenance of communication in mobile networks (c) Maintenance of communication between ground stations</td>
<td>3</td>
</tr>
<tr>
<td>2.1</td>
<td>Collision geometry between two circular objects</td>
<td>12</td>
</tr>
<tr>
<td>2.2</td>
<td>Collision geometry between two arbitrary objects [98]</td>
<td>15</td>
</tr>
<tr>
<td>2.3</td>
<td>Collision geometry between two arbitrary objects</td>
<td>17</td>
</tr>
<tr>
<td>2.4</td>
<td>Block diagram for collision avoidance law in $a_A$</td>
<td>20</td>
</tr>
<tr>
<td>2.5</td>
<td>Block diagram representation for collision avoidance law in $\delta_A$</td>
<td>23</td>
</tr>
<tr>
<td>2.6</td>
<td>Shape of the robot and obstacles considered in this simulation</td>
<td>27</td>
</tr>
<tr>
<td>2.7</td>
<td>Robot and Obstacle trajectories</td>
<td>28</td>
</tr>
<tr>
<td>2.8</td>
<td>Time histories of outputs $y_1, y_2, y_3$ to the three obstacles $B_1, B_2, B_3$ respectively</td>
<td>29</td>
</tr>
<tr>
<td>2.9</td>
<td>$\dot{V}_r$ (m/sec) vs Time (sec) for the three obstacles</td>
<td>30</td>
</tr>
<tr>
<td>2.10</td>
<td>$\psi$ (degrees) vs Time (sec) for the three obstacles</td>
<td>30</td>
</tr>
<tr>
<td>2.11</td>
<td>Time histories of (a) Acceleration $a_A$, (b) Heading angle $\alpha$</td>
<td>31</td>
</tr>
<tr>
<td>3.1</td>
<td>Acceleration, output and heading angle time histories obtained using a cooperative collision avoidance law</td>
<td>34</td>
</tr>
<tr>
<td>3.2</td>
<td>$XY$ trajectories of Objects $A$ and $B$</td>
<td>35</td>
</tr>
<tr>
<td>3.3</td>
<td>Shapes of objects $A, B, C$ and $D$</td>
<td>39</td>
</tr>
<tr>
<td>3.4</td>
<td>$XY$ trajectories of objects $A, B, C$ and $D$ when the network topology is a path graph with edges $AB, BC, CA$ and $AD$</td>
<td>40</td>
</tr>
<tr>
<td>3.5</td>
<td>Instances where the circular approximations to the objects would have collided, even though the actual objects did not collide</td>
<td>41</td>
</tr>
<tr>
<td>Figure</td>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>--------</td>
<td>-------------</td>
<td></td>
</tr>
<tr>
<td>3.6</td>
<td>Output $y$ corresponding to the six relative pairs.</td>
<td>42</td>
</tr>
<tr>
<td>3.7</td>
<td>Kinematic state trajectories of each relative pair of objects: (a) $\hat{r}$, (b) $\hat{\theta}$, (c) $V_\theta$ and (d) $V_r$, for each relative pair.</td>
<td>43</td>
</tr>
<tr>
<td>3.8</td>
<td>Accelerations of each object: (a) $A$, (b) $B$, (c) $C$ and (d) $D$.</td>
<td>44</td>
</tr>
<tr>
<td>3.9</td>
<td>Heading angles of objects $A, B, C$ and $D$.</td>
<td>45</td>
</tr>
<tr>
<td>3.10</td>
<td>Time histories of $\psi$ for all object pairs.</td>
<td>45</td>
</tr>
<tr>
<td>3.11</td>
<td>Positions and velocities of objects $A, B, C$ and $D$.</td>
<td>46</td>
</tr>
<tr>
<td>3.12</td>
<td>$XY$ trajectories of objects $A, B, C$ and $D$ when the network topology is a complete graph.</td>
<td>47</td>
</tr>
<tr>
<td>4.1</td>
<td>Robotic fish in body-fixed coordinate axes.</td>
<td>49</td>
</tr>
<tr>
<td>4.2</td>
<td>Illustration of an IPMC beam with passive fin [33].</td>
<td>51</td>
</tr>
<tr>
<td>4.3</td>
<td>Assembled robotic fish [96].</td>
<td>52</td>
</tr>
<tr>
<td>4.4</td>
<td>Block diagram depicting the integration of the guidance law with the IPMC actuator and fish dynamics.</td>
<td>58</td>
</tr>
<tr>
<td>4.5</td>
<td>Kinematic state trajectories of engagement between the robotic fish $A$ and obstacle fish $B &amp; C$: (a) $\hat{r}$, (b) $\hat{\theta}$, (c) $V_\theta$, (d) $V_r$.</td>
<td>60</td>
</tr>
<tr>
<td>4.6</td>
<td>(a) Output $y$ of robotic fish $A$ to fish $B &amp; C$, (b) Commanded acceleration $a_A$ by collision avoidance law, (c) $\delta_A$.</td>
<td>61</td>
</tr>
<tr>
<td>4.7</td>
<td>(a) Caudal Thrust, (b) Voltage Amplitude, (c) Tip deflection amplitude, (d) Tip slope amplitude of IPMC Actuator of caudal fin.</td>
<td>61</td>
</tr>
<tr>
<td>4.8</td>
<td>(a) Left Pectoral Thrust, (b) Voltage Amplitude, (c) Tip deflection amplitude, (d) Tip slope amplitude of IPMC Actuator of left pectoral fin.</td>
<td>62</td>
</tr>
<tr>
<td>4.9</td>
<td>(a) Right Pectoral Thrust, (b) Voltage Amplitude, (c) Tip deflection amplitude, (d) Tip slope amplitude of IPMC Actuator of right pectoral fin.</td>
<td>62</td>
</tr>
</tbody>
</table>
## LIST OF FIGURES (continued)

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.10(a)</td>
<td>(a)-(b) Body axis velocity components, $u$ and $v$ of robotic fish, (c) Time history of angle $\psi$ between robotic fish $A$ and $B$ &amp; $C$, (d) Time history of $\dot{\psi}$ between robotic fish $A$ and $B$ &amp; $C$.</td>
<td>63</td>
</tr>
<tr>
<td>4.11</td>
<td>$XY$ trajectory of robotic fish and obstacle.</td>
<td>63</td>
</tr>
<tr>
<td>5.1</td>
<td>Prototype of fabricated robotic fish</td>
<td>71</td>
</tr>
<tr>
<td>5.2</td>
<td>Schematic of robotic fish in planar motion</td>
<td>71</td>
</tr>
<tr>
<td>5.3</td>
<td>Model structure</td>
<td>74</td>
</tr>
<tr>
<td>5.4</td>
<td>Integrated System Architecture</td>
<td>79</td>
</tr>
<tr>
<td>5.5</td>
<td>Shape of bounding surface that encloses the robotic fish over its full range of flapping angles</td>
<td>79</td>
</tr>
<tr>
<td>5.6</td>
<td>Example 1: $XY$ Trajectories of robotic fish $A$ and $B$</td>
<td>80</td>
</tr>
<tr>
<td>5.7</td>
<td>Example 1: Time histories of accelerations $a_A$ and $a_B$ applied by fishes $A$ and $B$ respectively</td>
<td>81</td>
</tr>
<tr>
<td>5.8</td>
<td>Example 1: Kinematic state trajectories</td>
<td>82</td>
</tr>
<tr>
<td>5.9</td>
<td>Example 1: (a)-(d) Body axis velocity components $u_A$, $v_A$ and $r_A$ and applied moment $M_{aA}$ of robotic fish $A$</td>
<td>82</td>
</tr>
<tr>
<td>5.10</td>
<td>Example 1: (a)-(d) Body axis velocity components $u_B$, $v_B$ and $r_B$ and applied moment $M_{aB}$ of robotic fish $B$</td>
<td>83</td>
</tr>
<tr>
<td>5.11</td>
<td>Example 1: (a)-(b) Heading angle trajectories of robotic fish $A$ and $B$</td>
<td>83</td>
</tr>
<tr>
<td>5.12</td>
<td>Example 2: Kinematic state trajectories</td>
<td>84</td>
</tr>
<tr>
<td>5.13</td>
<td>Example 2: Time histories of accelerations $\bar{a}<em>A$ and $\bar{a}<em>B$ and corresponding amplitudes $A</em>{mA}$ and $A</em>{mB}$</td>
<td>85</td>
</tr>
<tr>
<td>5.14</td>
<td>Example 2: (a)-(d) Body axis velocity components $u_A$ and $v_A$ of robotic fish $A$</td>
<td>86</td>
</tr>
<tr>
<td>5.15</td>
<td>Example 2: (a)-(d) Angular velocity $r_A$ and heading angle $\phi_A$ of robotic fish $A$</td>
<td>86</td>
</tr>
<tr>
<td>5.16</td>
<td>Example 2: (a)-(d) Body axis velocity components $u_B$ and $v_B$ of robotic fish $B$</td>
<td>87</td>
</tr>
<tr>
<td>Figure</td>
<td>Description</td>
<td>Page</td>
</tr>
<tr>
<td>--------</td>
<td>-------------</td>
<td>------</td>
</tr>
<tr>
<td>5.17</td>
<td>Example 2: (a)-(d) Angular velocity $r_B$ and heading angle $\phi_B$ of robotic fish $B$</td>
<td>87</td>
</tr>
<tr>
<td>5.18</td>
<td>Example 2: XY Trajectories of $A$ and $B$</td>
<td>88</td>
</tr>
<tr>
<td>5.19</td>
<td>Example 3: (a) Applied acceleration $\bar{a}_A$ by collision avoidance law, (b) Voltage applied to IPMC of fish $A$</td>
<td>89</td>
</tr>
<tr>
<td>5.20</td>
<td>Example 3: XY Trajectory of fish $A$ as it performs non-cooperative collision avoidance maneuvers with obstacle fish $B$, $C$ and $D$</td>
<td>90</td>
</tr>
<tr>
<td>6.1</td>
<td>State Time Histories for Game 1, with accelerations $a_A = 1$ m/sec$^2$, $a_B = -3$ m/sec$^2$</td>
<td>94</td>
</tr>
<tr>
<td>6.2</td>
<td>Time Histories of $\phi$, $y$, acceleration inputs, and velocity heading angles for Game 1, with $a_A = 1$ m/sec$^2$, $a_B = -3$ m/sec$^2$</td>
<td>95</td>
</tr>
<tr>
<td>6.3</td>
<td>XY Trajectories of the Pursuer and Evader for Game 1, with $a_A = 1$ m/sec$^2$, $a_B = -3$ m/sec$^2$</td>
<td>95</td>
</tr>
<tr>
<td>6.4</td>
<td>State Time Histories for Game 1, with accelerations $a_A = -1$ m/sec$^2$, $a_B = -3$ m/sec$^2$</td>
<td>96</td>
</tr>
<tr>
<td>6.5</td>
<td>Time Histories of $\phi$, $y$, acceleration inputs, and velocity heading angles for Game 1, with $a_A = -1$ m/sec$^2$, $a_B = -3$ m/sec$^2$</td>
<td>96</td>
</tr>
<tr>
<td>6.6</td>
<td>XY Trajectories of the Pursuer and Evader for Game 1, with $a_A = -1$ m/sec$^2$, $a_B = -3$ m/sec$^2$</td>
<td>97</td>
</tr>
<tr>
<td>6.7</td>
<td>XY Trajectories of the Pursuer and Evader for Game 1, with $a_A = 3$ m/sec$^2$, $a_B = -2$ m/sec$^2$</td>
<td>97</td>
</tr>
<tr>
<td>6.8</td>
<td>State Time Histories for Game 2, with $\eta_A = 0$, $\eta_B = \frac{5\pi}{12}$</td>
<td>100</td>
</tr>
<tr>
<td>6.9</td>
<td>Time Histories of $\phi$, $y$, acceleration inputs, and velocity heading angles for Game 2, with $\eta_A = 0$, $\eta_B = \frac{5\pi}{12}$</td>
<td>101</td>
</tr>
<tr>
<td>6.10</td>
<td>XY Trajectories of the Pursuer and Evader for Game 2, with $\eta_A = 0$, $\eta_B = \frac{5\pi}{12}$</td>
<td>101</td>
</tr>
<tr>
<td>6.11</td>
<td>State Time Histories for Game 2, with $\eta_A = \frac{5\pi}{18}$, $\eta_B = 0$</td>
<td>102</td>
</tr>
</tbody>
</table>
LIST OF FIGURES (continued)

Figure                                      Page

6.12  Time Histories of $\phi, y$, acceleration inputs, and velocity heading angles for Game 2, with $\eta_A = \frac{5\pi}{18}, \eta_B = 0$ ....................................... 102

6.13  XY Trajectories of the Pursuer and Evader for Game 2, with $\eta_A = \frac{5\pi}{18}, \eta_B = 0$ .......................... 103

6.14  State Time Histories for Game 2, with $\eta_A = \frac{5\pi}{18}, \eta_B = 0$ .................................................. 103

6.15  Time Histories of $\phi, y$, acceleration inputs, and velocity heading angles for Game 2, with $\eta_A = \frac{\pi}{2}, \eta_B = 0$ .......................................................... 104

6.16  XY Trajectories of the Pursuer and Evader for Game 2, with $\eta_A = \frac{\pi}{2}, \eta_B = 0$ ................................. 104

7.1    Engagement geometry between a point object and a deforming object ......................................................... 107

7.2    Engagement geometry between a point object and (a),(b): an open chain, (c),(d): a closed chain .......................... 109

7.3    Geometric interpretation of the collision cone between a point object and a finite-sized deforming object .................. 110

7.4    Engagement geometry between a circular object and a deforming object ........................................................ 114

7.5    Engagement geometry between an arbitrary object and a deforming object ....................................................... 117

7.6    Spatial trajectories of object $A$ and deforming object $B$ .............................................................................. 119

7.7    Velocity distribution of $B(s)$ at several instants in time .................................................................................. 120

7.8    Relative velocity functions $V_r(s)$ and $V_A(s)$ at several instants in time ............................................................ 120

7.9    Angle function $\psi(s)$ at several instants in time ............................................................................................... 121

7.10   Collision cone function $y(s)$ at several instants in time .................................................................................... 121

7.11   Time histories of applied acceleration and heading angle of object $A$ .......................................................... 122

8.1    Illustration of hyperbolic sector and hyperbolic angle ......................................................................................... 124

8.2    Vehicles and their footprints: (a) $\psi < \pi$, (b) $\psi > \pi$ .................................................................................... 125

8.3    Representation of the rendezvous and collision cones on the $(V_\theta, V_r)$ plane, for (a) $\psi < \pi$, (b) $\psi > \pi$ . 130
<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.4</td>
<td>Definition of $\psi$ when $A'$ is a point and $B'$ is a circle. (a) $A'$ lies outside $B'$, (b) $A'$ lies inside $B'$</td>
</tr>
<tr>
<td>8.5</td>
<td>Definition of $\psi$ when $A'$ is a point and $B'$ is an arbitrary object. (a) $A'$ lies outside $B'$, (b) $A'$ lies inside $B'$</td>
</tr>
<tr>
<td>8.6</td>
<td>Definition of $\psi$ when $A'$ and $B'$ are both circles. (a) $A'$ lies outside $B'$ (b) $A'$ lies inside $B'$</td>
</tr>
<tr>
<td>8.7</td>
<td>Definition of $\psi$ when $A'$ and $B'$ are both arbitrary objects. (a) $A'$ lies outside $B'$, (b) $A'$ lies inside $B'$</td>
</tr>
<tr>
<td>8.8</td>
<td>(a) $b_m$ as a function of $c$, for $c &lt; 0$, (b) $\psi_m$ as a function of $c$, for $c &gt; 0$</td>
</tr>
<tr>
<td>8.9</td>
<td>A typical plot of (a) $y_1$, (b) $\dot{V}<em>r$, (c) $\dot{V}</em>\theta$ as a function of $\alpha$</td>
</tr>
<tr>
<td>8.10</td>
<td>Partitioning of 2-D space based on the relative signs of $y_1$, $y_2$, $V_{rb}$</td>
</tr>
<tr>
<td>8.11</td>
<td>Qualitative trajectory of $y_1(t)$ for an initial condition corresponding to $y_1 &lt; 0$, $V_r &gt; 0$</td>
</tr>
<tr>
<td>8.12</td>
<td>Footprint shape</td>
</tr>
<tr>
<td>8.13</td>
<td>Rendezvous of three footprints with a stationary leader footprint (Example 1)</td>
</tr>
<tr>
<td>8.14</td>
<td>Time history of angle $\psi$ made by $F_1$, $F_2$, and $F_3$ to the stationary leader</td>
</tr>
<tr>
<td>8.15</td>
<td>Time history of (a) the quantity $y_1$ of $F_1$, $F_2$, and $F_3$ to the stationary leader, (b) angle $\alpha$ made by $F_1$, $F_2$, and $F_3$</td>
</tr>
<tr>
<td>8.16</td>
<td>Rendezvous of two footprints with a moving leader footprint (Example 2)</td>
</tr>
<tr>
<td>8.17</td>
<td>Rendezvous of two footprints with a moving leader footprint, followed by a sweep (Example 3)</td>
</tr>
<tr>
<td>8.18</td>
<td>Time history of (a), $\psi$, (b) $y$, and (c) $\alpha$ between $F_2$ and $L$</td>
</tr>
<tr>
<td>8.19</td>
<td>Time history of $V_\theta$ and $V_r$ between $F_2$ and $L$</td>
</tr>
</tbody>
</table>
CHAPTER 1
INTRODUCTION

1.1 Research Motivation

An important component of robot path planning is the collision avoidance problem, that is, determining a safe trajectory of a robotic vehicle so that it circumvents various stationary as well as dynamic obstacles in its path. When the robot and obstacles are operating in close proximity, the relative shapes of the robots and the obstacles play an important role in the determination of collision avoidance trajectories. One common practice is to use polygonal approximations as bounding boxes for the shapes of the robots and obstacles. However, the polygonal approximation can lead to increased computational complexity (measured in terms of obstacle complexity, or the amount of information used to store a computer model of the obstacle [1], where obstacle complexity is measured in terms of the number of obstacle edges [2]). To overcome this, a common practice then is to use circular approximations (in 2-D environments) or spherical approximations (in 3-D environments) for the robots and the obstacles, because of the analytical convenience such approximations provide, along with the reduced information required to store a computational model of the obstacle. The obstacle avoidance conditions are then computed for the circle/sphere as a whole. These approximations become overly conservative in cases when an object is more elongated along one dimension compared to another. In such cases, nonspherical objects such as ellipsoids have been used to serve as better approximations for the object shapes [3]-[5]. For non-convex objects however, even ellipsoidal approximations can become overly conservative, because such approximations reduce the amount of available free space within which the robot trajectories can lie. This can be better understood from figures 1.2 and 1.1. In figure 1.1 an arbitrary shaped object (shown in red) is approximated by a circle (shown in blue), thereby wasting the space within which the safe trajectories can lie. In the
cluttered environment shown in figure 1.2, a safe trajectory for the point object would not be possible due to the over-conservative shape approximations shown.

![Circular Shape Approximation](image1.png)

Figure 1.1: Circular Shape Approximation

![Motion Planning in a Cluttered Environment](image2.png)

Figure 1.2: Motion Planning in a Cluttered Environment

This dissertation employs a collision cone based approach to determine analytical expressions of collision avoidance laws for moving objects of arbitrary shapes, without taking recourse to approximating the object shapes. The great benefit of using analytical expressions of collision conditions is that these then serve as a basis for determining analytical expressions of collision avoidance laws. Such analytical expressions can lead to tremendous computational savings, especially in multi-obstacle environments.
Figure 1.3: Examples of scenarios requiring overlaps of vehicle footprints: (a) Sweep operation (b) Maintenance of communication in mobile networks (c) Maintenance of communication between ground stations

Another important path planning problem is the rendezvous problem. The rendezvous problem is seen in many applications where overlap of vehicle footprints is crucial to the mission objective. The notion of what constitutes a vehicle’s footprint depends on the type of mission being performed. For example, when a group of UAVs are conducting a cooperative sweep of a region, the target should not be able to escape through any gaps between the surveillance footprints of the UAVs, while in a crop spraying operation by a
group of UAVs, no crop should be left unsprayed, thereby needing overlap of the spraying footprints. The rendezvous problem is also relevant in search and rescue missions looking for survivors after a disaster, since all the area needs to be swept so that no survivor is left undetected. Similarly, in cloud seeding and forest fire dousing, the need for achieving overlap of footprints can be seen. Also, to establish a wide communication network in remote areas, an overlap of communication footprints of a swarm of UAVs or those of a base station and a user is necessary. Such scenarios are schematically depicted in Figure 1.3. To address the rendezvous problem, the notion of rendezvous cones is used to achieve overlap of footprints to required depth.

1.2 Research Objective

The objective of this research is to demonstrate the applicability of the collision cone approach of [99] extensively for objects with arbitrarily shaped under different dynamic environments. The collision cone approach is used to determine collision avoidance laws between arbitrarily shaped objects. The usage of these avoidance laws by arbitrarily shaped objects in the context of formation control, pursuit evasion games, overlap of footprints, and deforming objects is achieved. The nonlinear analytical guidance laws proposed are simulated on dynamical systems like Ionic Polymer-Metal Composite (IPMC) actuated robotic fish.

1.3 Literature Review

1.3.1 Collision Avoidance Laws

Obstacle avoidance is a primary requirement in motion planning of autonomous vehicles. Several papers in the literature [6], [7], [8], [9], [10] address this problem. Motion planning can be classified [6] as static or dynamic, depending on whether the obstacles are stationary or moving, changing shape and size in the environment. The environments under which autonomous vehicles are operating could be completely known or partially known. The environment is completely known when the trajectory of the obstacles is known a priori and is partially known when obstacle trajectory is unknown or information about it is incomplete. This categorization is not universal and an alternative categorization is available.
in [7]. A significant research on motion planning in completely known as well as dynamic environments has been attempted in literature [9]. The fundamental objective of any motion planning algorithm is to generate a path of continuous sequence of positions from a starting point to a goal point given initial and final positions and orientations while optimizing some performance criterion, and report failure if no such path exists. Configuration space approach, Voronoi diagrams, retraction methods, potential functions, visibility graphs, accessibility graphs, tangent graphs, velocity obstacles etc. ([6], [7], [11], [12]) are some of the techniques which have been reasonably successful in achieving this objective. While these approaches are valid for a completely known environment, a dynamic unpredictable environment requires a different approach. Dynamic motion planning is more challenging than static motion planning even when complete information about the environment is available. This is shown by several available complexity results for motion planning [13].

The collision cone approach, originally introduced in [99], has some similarities with the velocity obstacle approach [12] in that both approaches determine the set of velocities of the robots that will place them on a collision course with one or more obstacles. However while the velocity obstacle approach, and its many extensions [14], has been restricted to circles/spheres, the fact that the collision cone approach has its roots in missile guidance, enables it to determine closed form collision conditions for a large class of object shapes. In [99], conditions that are both necessary and sufficient for collision of two arbitrarily shaped objects moving on a plane were determined, while in [98], [103] conditions that are necessary and sufficient for collision of quadric surfaces moving in 3-D space were determined. The collision cone approach of [99] has been extensively employed in the literature, for example, in driver assistance systems [15] - [17], safe trajectories for aircraft [18] - [20], multi-vehicle collision avoidance problem [16], and investigation of collision avoidance in biological systems such as fish schools [21].
1.3.2 Cooperative Collision Avoidance and Formation Control

It is commonly acknowledged that autonomous vehicles that can work cooperatively have a higher success rate and better operational capability, when compared to vehicles that perform a task individually. Some of the potential real world applications for multi-vehicle systems are search and rescue systems, space-based interferometers, combat systems, surveillance and reconnaissance systems, hazardous material handling, and distributed reconfigurable sensor networks.

Formation control with collision avoidance has been studied in the literature, and the literature largely focuses on using the potential field approach for collision avoidance ([22], [24], [23]). However, since the potential field approach ([25]) is a gradient-based approach that attempts to minimize a potential function, it has the drawback that it may get stuck at local minima. Also, the developed potential functions rely only on the instantaneous positions of the obstacles. The collision cone approach on the other hand is not a gradient-based approach, and is a one-step look-ahead approach that takes both the shape of the obstacles, as well as their instantaneous velocities into account, in the determination of collision avoidance trajectories.

1.3.3 Ionic Polymer Metal Composite (IPMC) Actuated Robotic Fish

The development of autonomous robotic fish that can perform effectively in underwater environments is an active area of research. Development of autonomous robotic fish can pave the way for a wide spectrum of underwater applications such as pollutant source seeking, oil spill monitoring, water quality monitoring, providing tsunami and seaquake warning, surveillance of leakage in underwater oil and gas pipelines, mine reconnaissance, fish survey and behavior study, and underwater search and rescue. For exploration of underwater environments using swarms of such robotic fish, the ability of the robotic fish to achieve collision avoidance is essential for safe deployment.

IPMCs are a class of electroactive polymers (EAPs). EAPs are emerging smart materials that can generate large deformations under electrical stimuli [26]. Due to their
similarities to biological muscles, EAPs are often called artificial muscles, and they have different configurations, which can be divided into two categories: dielectric EAPs and ionic EAPs. Dielectric EAPs can generate large forces with large deformations \cite{27, 28, 29}; however, they require high actuation voltages (typically higher than 1 kV), which limits their applications in bio-inspired robotic fish. Ionic polymer-metal composites (IPMCs) are an important category of ionic EAPs. Since IPMCs are soft, lightweight, low-power consuming, and are capable of generating flapping motion, they are ideal artificial muscles for small-scale underwater bio-robots.

To date, several efforts have been devoted toward IPMC-powered underwater robots \cite{30, 31, 32}. For example, Tan et al. developed a robotic fish propelled by an IPMC caudal fin \cite{32}, and then Chen et al. developed a speed model for the robotic fish \cite{33}. An IPMC-powered robotic manta ray and cow-nose ray have also been developed \cite{34, 35, 36}. Yang and Chen developed a 2D maneuverable robotic fish actuated by multiple IPMC fins \cite{96}. A dynamic model of robotic fish can be used to have a better sense of system behavior and is also required for control design purposes. Several works have focused on obtaining such dynamic models \cite{37, 38, 39, 40, 41, 42}. However, an ionic EAP has some limitations in force output and time response, and cannot generate high frequency flapping to achieve high speed. When compared to soft actuators, traditional servo motors can generate a large enough torque to drive the fish segments to flap at a desirable frequency. However, in order to achieve two-dimensional (2D) maneuvering capability, a shape change on the caudal fin is needed to direct the vortex for 2D thrust generation. For this reason, combining an EAP soft actuator and servo motor in a hybrid tail is beneficial since the tail structure has more power and is simpler than just using two servomotors or EAPs only.

1.3.4 Pursuit Evasion Games Analysis

In a pursuit evasion game, one or more pursuers try to capture one or more evaders, who try to avoid being captured \cite{43}. In some pursuit evasion games, capture is said to occur when the distance between the pursuer and the evader becomes less than some pre-specified
threshold, while in other games capture is said to occur when the pursuers see or surround the evader. Pursuit evasion games have applications in warfare, motion planning problems in robotics, network security, modeling animal behavior, etc. These games are also used to study collision avoidance in dynamic environments, in order to determine the worst-case behavior of the obstacle(s) and then check if the collision avoidance algorithm can guarantee collision avoidance even in the presence of this worst-case behavior. Examples of pursuit evasion differential games include the homicidal chauffeur game [44], [45], the game of two cars [46], the lion and the man, the lady in the lake [47], etc.

In [51], a visibility based pursuit evasion is analyzed using randomized strategy in a polygonal environment assuming point objects. In [50], a new class of searcher called φ searcher is presented to deal with visibility based pursuit evasion problems in polygonal environment. [43] is an excellent resource surveying autonomous search and pursuit-evasion in mobile robotics discussing the fundamentals of pursuit evasion games on graphs and polygonal environments with detailed theory and relevant applications. In [48], [49], a pursuer missile carrying a selectively aimable warhead, which is known to have a conical directed blast zone is considered. In this case, the target may be modelled as an evader trying to avoid this non-circular zone of the pursuer. In [111], multiple pursuers vs a fast evader game is analyzed using apollonius techniques while assuming pursuer and evader as point objects. While the literature on pursuit evasion games largely assumes the shapes of the objects to be circles, or in some cases ellipses, there can be many scenarios wherein analysis of pursuit evasion games for a larger class of shapes can be of interest.

1.3.5 Collision Cones for Deforming Objects

It is important to be able to perform collision avoidance in environments that include deforming objects. Examples of deforming objects include oil spills moving on the ocean surface, shape changing robots, snake like robots, and boundaries of vehicle swarms. There is very little work in the literature where avoidance of such deforming obstacles has been considered. Exceptions are [52], where the problem of a moving and deforming obstacle
is addressed using sliding mode control, [53]-[54], and [55] where path planning through deforming obstacles has been proposed using the notion of collision cones. The paper [55] addresses an application of avoiding a deforming obstacle using radar measurement and the proposed collision cone based algorithm is very specific to the application. However, a general theory of avoiding deforming obstacles is not available in the literature.

1.3.6 Achieving Overlap of Footprints

In recent years, UAVs equipped with sensors have demonstrated the potential to be used for communication [56]-[57], search [58]-[63] and surveillance applications [64]-[68]. Many of these applications require that the footprints of the UAVs need to overlap. Depending on the specific application, the term “footprint” could mean a sensor footprint, the region of effectiveness of equipment carried by a UAV, or a communication footprint.

There exists a class of sensor coverage problems in the literature, in which a given area is to be covered - for surveillance or search operations - by a group of mobile sensors. The broad issues in these applications are the requirement of distributing the sensor locations such that the maximum possible area is covered (static problem), or the need to make a group of mobile sensors follow a path through the region in such a way that over time the maximum area is covered by the sensors (dynamic problem) [70]-[75]. Several techniques have been proposed in the literature to solve these problems. In general, a majority of them are based on the probability distribution of some random events of interest in a given search area, and the solution is proposed in terms of providing the most effective coverage by positioning sensors in the search area in an optimal manner.

Overlap of sensor footprints is not difficult when they are circular or their effectiveness reduces uniformly in all directions (that is, they are isotropic), as in such cases all that is required is an algorithm that drives the sensors to within a certain specified distance of each other (this is basically just the standard formation problem). However, the problem is more difficult when the sensors are anisotropic, thereby making their footprints arbitrarily shaped. Such arbitrarily shaped footprints can occur in several ways. They can occur in
communication problems, where one vehicle is trying to communicate with another using a beam-forming antenna, which can have non-circular footprints, and can be made direction dependent \[70\], \[71\]. Arbitrary footprints can also occur in photography applications, wherein the video sensors have a directional sensing model with a sector-shaped field-of-view and also a limited depth of field \[78\], \[79\]. In fact, the papers in the literature that consider anisotropic footprints almost invariably consider sector shaped footprints (for example, see \[80\], \[81\]). However, even within this sector shaped region, if an obstacle occludes a portion of the field of view, then the effective visual footprint of the camera is the sector less the obstacle, and this can be arbitrary. Even for a sensor with a circular sensing zone, when there are obstacles within the circular footprint that impede the sensor from taking measurements along some directions, the effective footprint of the sensor becomes non-circular and arbitrary. Finally, a swarm of UAVs carrying their individual sensors can create a composite sensor footprint that is not necessarily isotropic \[82\], \[83\]. Several papers addressing sweep coverage (see for example \[80\], \[81\], \[84\]–\[88\], of which \[84\]–\[86\] relate to intruder detection), communication-aware surveillance \[89\]–\[92\] and rendezvous of agents \[93\]–\[95\] assumes circular sensing/communication zones, and thus circular footprints.

1.4 Dissertation Organization

The dissertation is organized as follows: in Chapter 2, two classes of collision avoidance laws between arbitrarily shaped objects, along with effects of singularities and time delays on performance of guidance laws are presented. Simulation results demonstrating working of these laws are also presented. In Chapter 3, cooperative collision avoidance laws are integrated with formation control laws. In Chapters 4 and 5, two integrated models comprising the nonlinear guidance laws for collision avoidance, the nonlinear dynamic models of robotic fish \[96\], \[97\] and IPMC fin dynamics are analyzed. In Chapter 6 we illustrate how a pursuit evasion game can be connected to collision cone concepts. We formulate two pursuit evasion games based on the collision cone, and also present numerical results for these two games. In Chapter 7, a general theory and avoidance guidance law for deforming
obstacles is presented. In Chapter 8, we present analytical guidance laws that enable a group of autonomous vehicles, such as UAVs, to achieve overlap of their footprints with that of a leader vehicle or with each other using the notion of rendezvous cones. Chapter 9 presents conclusions.
CHAPTER 2

COLLISION AVOIDANCE LAWS FOR OBJECTS WITH ARBITRARY SHAPES

In this chapter, analytical guidance laws for objects with arbitrary shapes are presented. Two collision avoidance laws are discussed. One considers the magnitude of the acceleration as the primary input, while the other considers the direction of acceleration as the primary input. Section 2.1 provides a review of the collision avoidance results for arbitrary objects moving on a plane, as determined in [99]. Section 2.2 uses these results to determine two classes of collision avoidance laws for arbitrarily shaped objects. Section 2.3 analyzes the effect of singularities and time delays on the performance of the guidance laws. Section 2.4 presents simulations that demonstrate the working of these collision avoidance laws.

2.1 Collision Cone between Two Arbitrary Objects

In this section, we review the result derived in [99] that determined the collision conditions between two arbitrary objects moving on a plane.

Figure 2.1: Collision geometry between two circular objects
First, consider two circles $F_1$ and $F_2$ moving with speeds $V_A$ and $V_B$ acting at angles $\alpha$ and $\beta$, respectively, as shown in Figure 2.1. The Line-of-Sight (LOS) is defined as the line joining the centers of the two circles, and its behavior is characterized by the following kinematic equations:

\begin{align}
V_r &= \dot{r} = V_B \cos(\beta - \theta) - V_A \cos(\alpha - \theta) \tag{2.1} \\
V_\theta &= r \dot{\theta} = V_B \sin(\beta - \theta) - V_A \sin(\alpha - \theta) \tag{2.2}
\end{align}

where $V_r$ and $V_\theta$ are, respectively, the relative velocity components along, and perpendicular to, the LOS. Taking the velocities of $A$ and $B$ to be constants, we can differentiate (2.1) and (2.2) to obtain

\begin{align}
\dot{V}_r &= \dot{\theta} V_\theta, \quad \dot{V}_\theta = -\dot{\theta} V_r, \tag{2.3}
\end{align}

which, after eliminating $\dot{\theta}$ can be written as:

\begin{align}
V_\theta^2 + V_r^2 = V_{\theta 0}^2 + V_{r 0}^2 \tag{2.4}
\end{align}

which on integration yields,

\begin{align}
V_{\theta m}^2 = V_{r 0}^2 + V_{\theta 0}^2 \tag{2.5}
\end{align}

where, the subscript '0' indicates the values of the relative velocity components at some initial time $t = t_0$. Assume the initial engagement commences with $V_{r 0} < 0$. Then, using basic results on missile guidance dealing with point objects, we can determine that the instant when the centers $P_1$ and $P_2$ of the two circles achieve their closest approach, occurs when $V_r = 0$. Let $V_r = 0$ occur at some time $t = t_m$, and let $r(t_m) = r_m$. Then, from (2.5), we can infer that

\begin{align}
V_{\theta m}^2 = V_{r 0}^2 + V_{\theta 0}^2 \tag{2.6}
\end{align}
where, $V_{\theta m}$ represents $V_{\theta}(t_m)$. Multiplying both sides of the second equation in (2.3) by $r$, and rearranging the resulting equation, we obtain,

$$\frac{V_{\theta} \dot{V}_{\theta}}{V_{\theta}^2} = -\frac{\dot{r}}{r} \Rightarrow \frac{V_{\theta}^2}{V_{\theta 0}^2} = - \left( \frac{r_0}{r} \right)^2$$

(2.7)

Putting $r = r_m$ at $t = t_m$ and substituting from (2.6):

$$r_m^2 = r_0^2 \left( \frac{V_{\theta 0}^2}{V_{\theta 0}^2 + V_{r 0}^2} \right)$$

(2.8)

Equation (2.8) thus gives the expression for the distance between the points $P_1$ and $P_2$ at the instant of closest approach as a function of the initial values of the relative velocity components and the initial relative separation. (Note that the expression assumes that the engagement commences at an initial condition $V_{r 0} < 0$. If $V_{r 0} > 0$, then the same expression gives the distance at closest approach if the trajectories are projected backwards in time.)

When $r_m \leq R$, (where $R = R_1 + R_2$ is the sum of the radii of the two circles), then the two circles would have encountered a collision. The collision condition between the two circles can then be written as:

$$r_0^2 \left( \frac{V_{\theta 0}^2}{V_{\theta 0}^2 + V_{r 0}^2} \right) \leq R^2, \quad V_{r 0} < 0$$

(2.9)

The above equations assume constant velocities throughout the engagement. When the objects move with varying velocities, we write the *instantaneous* collision condition between the two circles as:

$$r^2 \left( \frac{V_{\theta}^2}{V_{\theta}^2 + V_{r}^2} \right) \leq R^2, \quad V_{r} < 0$$

(2.10)

Now, consider the engagement between two arbitrary objects $A$ and $B$ as shown in Figure 2.2. We can construct a cone $\epsilon_3$ (generated by the lines $PQ$ and $RS$), which is the smallest cone such that it contains $A$ and $B$ on opposite sides of its vertex. (Note that $\epsilon_3$ is distinct from the collision cone). Furthermore, we can construct two *imaginary* circles
$F_1$ and $F_2$ that lie on opposite sides of the vertex $O$ of $\epsilon_3$, and are tangential to $\epsilon_3$. Then, assuming that $F_1$ and $F_2$ move with velocities identical to those of $A$ and $B$, respectively, we can infer that $A$ and $B$ lie on a collision course if and only if, $F_1$ lies on a collision course with $F_2$. This can be seen as follows: Assume $F_1$ is on a collision course with $F_2$, with point $C_1$ on $F_1$ lying on a collision course with point $C_2$ on $F_2$. Then, from basic conditions on collision between two point objects, we have that $V_{\theta(C_1C_2)} = 0, V_r(C_1C_2) < 0$. Now we can always construct a ray $D_1D_2$ that is parallel to $C_1C_2$, and such that $D_1$ lies on $A$ and $D_2$ lies on $B$. Since parallel rays have identical $V_r, V_\theta$ components, we therefore have that $V_{\theta(D_1D_2)} = V_{\theta(C_1C_2)}, V_r(D_1D_2) = V_r(C_1C_2)$. Therefore $V_{\theta(D_1D_2)} = 0, V_r(D_1D_2) < 0$, which implies that points $D_1$ and $D_2$ lie on a collision course, and thus $A$ lies on a collision course with $B$. The converse can be proved similarly.

We can then define an angle $\psi = 2\sin^{-1}\left(\frac{R_1+R_2}{r}\right)$ as shown in Figure 2.2 and use (2.10) to write the collision condition between two arbitrary objects $A$ and $B$ as:

$$\left(\frac{V_\theta^2}{V_\theta^2 + V_r^2}\right) \leq \sin^2\left(\frac{\psi}{2}\right), \ V_r < 0$$

where $V_\theta$ and $V_r$ represent the relative velocity components of the angular bisector of the cone $\epsilon_3$. Note that the above equation does not use any range information, and the angle $\psi$ captures the effects of the shapes of the two objects, for a given relative orientation.
2.2 Laws for Collision Avoidance between Two Arbitrary Objects

In this section, we demonstrate collision avoidance laws between two arbitrarily shaped objects, by making use of (2.11). We first use (2.11) to define a quantity \( y \) as follows:

\[
y = \frac{V_\theta^2}{V_\theta^2 + V_r^2} - \sin^2\left(\frac{\psi}{2}\right)
\]

(2.12)

The collision cone is then defined as the region in \((V_\theta, V_r)\) space for which \( y < 0 \) (for a given value of \( \psi \)). Using \( y \), we define three lemmas as follows:

Lemma 2.1: If \( A \) and \( B \) move with constant velocities, then \( y < 0, V_r < 0 \) are both necessary and sufficient conditions for collision to occur.

Lemma 2.2: If \( A \) and/or \( B \) move with varying velocities, then a necessary condition for collision to occur is that at the instant of closest approach, \( y < 0 \) occurs.

Lemma 2.3: The conditions (a) \( y < 0, V_r \geq 0 \) for all future time, and (b) \( y \geq 0 \) for all future time, are each sufficient conditions for collision avoidance.

The condition \( y < 0, V_r < 0 \) physically means that the relative velocity vector lies inside the collision cone, while the condition \( y = 0, V_r < 0 \) physically means that the relative velocity vector is aligned with the boundary of the collision cone.

We point out that in the computation of \( y(t) \) using (2.12), measurements of \( V_\theta(t) \), \( V_r(t) \) and \( \psi(t) \) are required. The angle \( \psi \) can be computed as follows. Let \( F(\bar{x}, \bar{y}) \) and \( G(\bar{x}, \bar{y}) \) represent the equations of the shapes of a pair of objects in an inertial reference frame \((\bar{x}, \bar{y})\). Compute the common tangents to both \( F \) and \( G \). Note that the number of such common tangents depends on the shapes \( F(\bar{x}, \bar{y}) \) and \( G(\bar{x}, \bar{y}) \), and there can be multiple common tangents. Let \( m_1, m_2, ..., m_n \) represent the slopes of these common tangents. We then have that \( \psi = \max_{i,j} |\tan^{-1} m_i - \tan^{-1} m_j| \).

In structured environments where the shapes of all the objects sharing the environment are known a priori, one can use the known \( F \) and \( G \) to directly compute \( \psi \). In unstructured environments where the shapes of all the objects are not known a priori, we
can reasonably assume that each robot knows its own shape and then use sensors such as ceiling-mounted cameras to measure the shape $G(\bar{x}, \bar{y})$ of the obstacle. In scenarios where ceiling-mounted cameras are not possible, one can use small radars/lidars mounted on the robot, and use the time-of-return of the reflected signals from the obstacle, to determine that portion of the obstacle shape $\partial G$ that is visible to the robot’s sensor. Using $F$ and $\partial G$, we compute the common tangents and determine $\psi$ as above. The determination of a good update rate depends on how rapidly the vehicles are maneuvering in the specific environment under consideration. In this dissertation, we present collision avoidance laws in a general framework that does not restrict itself to a specific single application or sensor type.

Consider the engagement geometry between two arbitrary objects $A$ and $B$, as shown in Figure 2.3. The state equation, $\dot{x} = f(x) + g(x)u$ corresponds to the kinematics of the line $P_1P_2$ and takes the form

![Figure 2.3: Collision geometry between two arbitrary objects](image)
\[
\begin{bmatrix}
\dot{r} \\
\dot{\theta} \\
\dot{V}_\theta \\
\dot{V}_r
\end{bmatrix}
= \begin{bmatrix}
0 \\
0 \\
-\sin(\delta_A - \hat{\theta}) \\
-\cos(\delta_A - \hat{\theta})
\end{bmatrix}
\begin{bmatrix}
a_A \\
a_B
\end{bmatrix}
+ \begin{bmatrix}
0 \\
0 \\
\sin(\delta_B - \hat{\theta}) \\
\cos(\delta_B - \hat{\theta})
\end{bmatrix}
\begin{bmatrix}
0 \\
0 \\
\sin(\delta_A - \hat{\theta}) \\
-\cos(\delta_A - \hat{\theta})
\end{bmatrix}
\begin{bmatrix}
\dot{V}_r \\
\dot{V}_\theta
\end{bmatrix}
\] (2.13)

Here, \( \hat{r} \) and \( \hat{\theta} \) represent, respectively, the range and the bearing angle of \( P_1P_2 \). \( \dot{V}_\theta \) and \( \dot{V}_r \) represent, respectively, the relative velocity components (of \( B \) with respect to \( A \)) that are normal to, and along, \( P_1P_2 \). \( a_A \) and \( a_B \) represent, respectively, the magnitudes of the accelerations of \( A \) and \( B \), while \( \delta_A \) and \( \delta_B \) represent the angles at which these accelerations are applied. In general, the above equation has two inputs \( a_A \) and \( \delta_A \) considering object \( B \) is not applying any acceleration (\( a_B = 0 \)). Note that, in general, \( \hat{\theta} \) is distinct from \( \theta \), since the line \( P_1P_2 \) joining the center of masses is not necessarily the same as the angular bisector line \( XOX' \). Because of this, \( \dot{V}_\theta \) and \( \dot{V}_r \) are distinct from \( V_\theta \) and \( V_r \). The quantities \( V_r, V_\theta \) are related to \( \dot{V}_r, \dot{V}_\theta \) by the equation:

\[
\begin{bmatrix}
\dot{V}_r \\
\dot{V}_\theta
\end{bmatrix}
= \begin{bmatrix}
\cos(\hat{\theta} - \theta) & -\sin(\hat{\theta} - \theta) \\
\sin(\hat{\theta} - \theta) & \cos(\hat{\theta} - \theta)
\end{bmatrix}
\begin{bmatrix}
V_r \\
V_\theta
\end{bmatrix}
\] (2.14)

In the special case when \( A \) and \( B \) are both circular objects, then the angular bisector of \( \epsilon_3 \) coincides with \( P_1P_2 \), that is, \( \hat{\theta} = \theta \), and thus, \( V_r = \dot{V}_r, V_\theta = \dot{V}_\theta \). The output equation \( y = h(x) \) is as written in (2.12). We now seek to determine a collision avoidance law that will drive \( y(t) \) from an initial negative value, to a non-negative value. This would be equivalent to determining \( a_A \) and/or \( \delta_A \) that will drive the velocity vector of \( A \) from the interior of the collision cone to the boundary of, or outside, the collision cone. We have that,

\[
\dot{y} = \left[ \frac{\partial y}{\partial r} \frac{\partial y}{\partial \theta} \frac{\partial y}{\partial V_\theta} \frac{\partial y}{\partial V_r} \frac{\partial y}{\partial \psi} \right] X \left[ \dot{r} \ \dot{\theta} \ \dot{V}_\theta \ \dot{V}_r \ \dot{\psi} \right]^T
\] (2.15)
and the partial derivatives of $y$ are the following:

\[
\begin{align*}
\frac{\partial y}{\partial r} &= 0; \quad \frac{\partial y}{\partial \theta} = 0; \quad \frac{\partial y}{\partial V_r} = \frac{2V_\theta V_r^2}{V_\theta^2 + V_r^2}; \\
\frac{\partial y}{\partial V_\theta} &= \frac{-2V_\theta^2 V_r \cdot \frac{\partial y}{\partial V_r} = -\sin \psi}{V_\theta^2 + V_r^2}; \quad \frac{\partial y}{\partial \psi} = -\sin \psi
\end{align*}
\]

(2.16)

2.2.1 Collision avoidance law with $a_A$ as the primary input

In this subsection, we determine a guidance law that treats the acceleration magnitude $a_A$ as the primary input, and assumes that $\delta_A$ is arbitrarily chosen. Note that the state equation (2.13) is affine in $a_A$, and we can use dynamic inversion to determine an analytical representation of a guidance law for $a_A$ (applied at a pre-defined angle $\delta_A$), that will drive $y(t)$ to a non-negative quantity.

Define an error signal $e(t) = w(t) - y(t)$, where $w(t)$ is the reference input indicated in Figure 2.4. Thus, $\dot{y} = \dot{w} - \dot{e}$. Taking $w(t) = \text{constant} \ \forall \ t$ and enforcing the error dynamics to be $\dot{e} = -K e$, (with $K > 0$), we get $\dot{y} = K(w - y)$. Using dynamic inversion techniques, we eventually obtain an equation of form:

\[
a_A N_{a,A} + a_B N_{a,B} = K(w - y)(V_\theta^2 + V_r^2)^2 + Y_a
\]

(2.17)

where, $N_{a,A}$, $N_{a,B}$ and $Y_a$ are given by the expressions:

\[
\begin{align*}
N_{a,A} &= 2V_r V_\theta (V_\theta \cos(\delta_A - \theta) - V_r \sin(\delta_A - \theta)) \\
N_{a,B} &= 2V_r V_\theta (V_\theta \sin(\delta_B - \theta) - V_\theta \cos(\delta_B - \theta)) \\
Y_a &= 2V_r V_\theta \dot{\theta}(V_\theta^2 + V_r^2) + 0.5 \dot{\psi} \sin \psi (V_\theta^2 + V_r^2)^2
\end{align*}
\]

(2.18)  (2.19)  (2.20)
Here, object $B$ is considered to be moving with zero acceleration ($a_B = 0$). Thus, acceleration of object $A$ is obtained from equation (2.17) as below:

$$a_A = \frac{K(w - y)(V_o^2 + V_r^2)^2 + Y_{a,A}}{N_{a,A}}$$  \hspace{1cm} (2.21)

Equation (2.21) is a guidance law that will drive $y(t)$ to $w \geq 0$, and is schematically represented in the block diagram of Figure 2.4. Note that since the error dynamics $e(t)$ are of first order, therefore any positive value of $K$ will guarantee that $e(t)$ decays exponentially to zero. Furthermore, when $y(t)$ becomes zero, this corresponds to the physical scenario of the velocity vector of $A$ being aligned with the boundary of the collision cone.

![Figure 2.4: Block diagram for collision avoidance law in $a_A$](image)

We however need to choose $K$ to be at least as large as to ensure that $y(t)$ decays to zero before the time to collision $t_m$. The error dynamics equation has a solution of the form: $e(t) = e(0)e^{-Kt}$, where $e(0)$ represents the initial error. We require that this error decay to 0 within a time less than $t_m$. It can be seen that this would be ensured by choosing a $K$ that satisfies

$$K > \frac{1}{t_m}ln\left(\frac{e(0)}{\varepsilon}\right)$$  \hspace{1cm} (2.22)

(wherein it is understood that $\varepsilon < e(0)$).

### 2.2.2 Collision avoidance law with $\delta_A$ as the primary input

In this subsection, we determine a guidance law for object $A$ taking $\delta_A$ (which is the angle at which the acceleration is applied), as the primary input, and assuming that the
acceleration magnitude $a_A$ is 1 when the velocity vector of $A$ lies inside the collision cone, (that is, $y < 0, V_r < 0$ are satisfied) and 0 when the velocity vector of $A$ lies outside the collision cone (that is, either (a) $y > 0$, or (b) $y < 0, V_r > 0$ are satisfied). To understand the value of using $\delta_A$ as an input, we can consider two extremes: one case wherein $\delta_A = \alpha$ and the second case wherein $\delta_A = \alpha + \pi/2$. When $\delta_A = \alpha$, the acceleration acts along the velocity vector of $A$, and its effect is to purely increase or decrease the speed of $A$ in order to bring the velocity vector of $A$ out of the collision cone. When $\delta_A = \alpha + \pi/2$, the acceleration vector acts orthogonal to the velocity vector of $A$ and its effect is to purely rotate the velocity vector of $A$ (while keeping speed constant) in order to bring the velocity vector of $A$ out of the collision cone. A variable $\delta_A$ generated by a collision avoidance law has the effect of modulating the influence of the applied acceleration between these two extremes, so as to result in an appropriate combination of simultaneous speed and heading change. Similar argument can be made for object $B$. We note that the kinematics equation (2.13) is not affine in $\delta_A$ and $\delta_B$, and because of this we cannot use a dynamic inversion technique in the direct form. We therefore define two new inputs $v_A$ and $v_B$, with $\dot{\delta}_A = v_A$ and $\dot{\delta}_B = v_B$, and append the same to the original state equation which is rewritten as follows:

$$
\begin{bmatrix}
\dot{x} \\
\dot{\delta}_A \\
\dot{\delta}_B
\end{bmatrix} = \begin{bmatrix}
f(x, \delta_A, \delta_B, a_A, a_B) \\
0 \\
0
\end{bmatrix} + \begin{bmatrix}
0 \\
1 \\
0
\end{bmatrix} v_A + \begin{bmatrix}
0 \\
0 \\
1
\end{bmatrix} v_B 
$$

(2.23)

where $\dot{x} = f(x, \delta_A, \delta_B, a_A, a_B)$ represents the equation (2.13).

We now employ the dynamic inversion technique on (2.23) treating $v_A$ and $v_B$ as the inputs, and $y$ as defined in (2.12) as the output. Note that the inputs $v_A$ and $v_B$ do not appear in the first derivative of $y$, but appear in the second derivative instead. We define the error as $e(t) = w(t) - y(t)$ where $w(t)$ is the reference input. Thus, $\dot{y} = \dot{w} - \dot{e}$. Taking $w(t) = \text{constant \forall t}$ and enforcing the error dynamics to be $\ddot{e} = -K_1 e - K_2 \dot{e}$, (with
$K_1, K_2 > 0$ for stability), using dynamic inversion we get:

$$\dot{\delta}_A N_{d,A} + \dot{\delta}_B N_{d,B} = (K_1(w - y) - K_2 \dot{y})(V_\theta^2 + V_r^2)^2 + Y_d$$

(2.24)

where,

$$N_{d,A} = 2a_A c_1 V_r^2 V_\theta + 2a_A s_1 V_r V_\theta^2$$

(2.25)

$$N_{d,B} = 2a_B c_2 V_r^2 V_\theta + 2a_B s_2 V_r V_\theta^2$$

(2.26)

$$Y_d = 2(s_1^2 V_r^2 - c_1^2 V_\theta^2) a_A^2 + 2(s_1^2 V_r^2 - c_2^2 V_\theta^2) a_B^2 + 4\dot{\delta}(s_1 V_r V_\theta^2 + s_1 V_r^3 + c_1 V_\theta^3 + c_1 V_r^2 V_\theta)a_A$$

$$-4\dot{\delta}(s_2 V_r V_\theta^2 + s_2 V_r^3 + c_2 V_\theta^3 + c_2 V_r^2 V_\theta)a_B - 4(s_1 s_2 V_r^2 - c_1 c_2 V_\theta^2) a_A a_B$$

$$-0.5(V_r^2 + V_\theta^2)^2(\ddot{\psi}^2 \cos \psi + \dot{\psi} \sin \psi) + 2\ddot{\theta}^2(V_r^4 - V_\theta^4) - 2\ddot{\theta} V_r V_\theta(V_r^2 + V_\theta^2)$$

(2.27)

and, for the purposes of brevity, we use the shorthand notation

$$s_1 = \sin(\delta_A - \theta); s_2 = \sin(\delta_B - \theta); c_1 = \cos(\delta_A - \theta); c_2 = \cos(\delta_B - \theta)$$

As object $B$ is assumed to be an obstacle moving with zero inputs we have $a_B = 0$ and $v_B = 0$. Thus, we obtain equation for $\delta_A$ as below:

$$\dot{\delta}_A = \frac{(K_1(w - y) - K_2 \dot{y})(V_\theta^2 + V_r^2)^2 + Y_d}{N_{d,A}}$$

(2.28)

The $\dot{\delta}_A$ from (2.28) is integrated in time to obtain $\delta_A(t)$, which then controls the direction of the acceleration vector appropriately, so as to drive $y(t)$ to the reference value $w > 0$. This is schematically shown in the blockdiagram of Fig 2.5.

2.3 Analysis of Collision Avoidance Laws

In this section, we perform an analysis of singularities and the influence of time delays for the collision avoidance law (2.21). A corresponding analysis can be performed for (2.28), and is omitted due to space considerations.
2.3.1 Singularities

An inspection of (2.21) shows that the guidance law becomes singular in any of the following cases:

(a) $V_r \sin(\delta_A - \theta) = V_\theta \cos(\delta_A - \theta)$: This corresponds to the physical scenario when the angle at which the acceleration is applied, coincides with that of the relative velocity vector, that is, $\delta_A = \theta + \tan^{-1} \frac{V_\theta}{V_r}$. When this happens, one can employ a different value of $\delta_A$ to overcome the singularity.

(b) $V_r = 0$: Assuming $V_r(0) < 0$, then the instant at which $V_r = 0$ occurs, corresponds to the instant of closest approach of the two objects. If the gain $K$ is chosen higher than the threshold (2.22), then at the instant of closest approach, the velocity vector of $A$ will already be outside the collision cone (that is, $y > 0$ will already have occurred), thus satisfying collision avoidance. The collision avoidance law will subsequently not be active when $V_r = 0$, and this singularity will thus never be encountered in practice.

(c) $\csc(\frac{\psi}{2}) = 0$: This can never be encountered in practice.

(d) $V_\theta = 0$: This corresponds to the physical scenario wherein $A$ and $B$ are moving such that the angular bisector of the cone $\epsilon_3$ (shown in Fig 2.3) does not rotate. When this happens, $A$ can apply its maximum acceleration and, as long as this acceleration has a component normal to the angular bisector, then (regardless of the numerical value of this acceleration), this will drive $V_\theta$ away from zero.
Whenever (a) or (d) occurs, performing the indicated operation will ensure that the singularity remains isolated in time.

2.3.2 Effects of Time Delays

In this section, the effect of time-delays on collision avoidance performance is analyzed. The time delays could be a consequence of sensor measurement delays, actuation delays and/or computation delays. If we consider that the effect of these delays is to cause the applied acceleration use $\psi(t - \tau), V_r(t - \tau), V_\theta(t - \tau)$ and $\delta_A(t - \tau)$ in lieu of $\psi, V_r, V_\theta$ and $\delta_A$ in (2.21), then the equation $\dot{y} = K(w - y)$ assumes the form:

$$\dot{y} = -Ky(t - \tau) \frac{Y(t)}{Y(t - \tau)} + \frac{X(t - \tau)Y(t) - X(t)Y(t - \tau)}{Y(t - \tau)} + Kw \frac{Y(t)}{Y(t - \tau)} \quad (2.29)$$

where $X(t)$ and $Y(t)$ are as follows:

$$X(t) = \frac{\partial y}{\partial V_\theta} V_r \frac{V_r}{r} - \frac{\partial y}{\partial V_r} \frac{V_\theta^2}{r} - \frac{\partial y}{\partial \psi} \dot{\psi} \quad (2.30)$$

$$Y(t) = -\frac{\partial y}{\partial V_r} \cos(\delta_A - \theta) - \frac{\partial y}{\partial V_\theta} \sin(\delta_A - \theta) \quad (2.31)$$

The partial derivatives in the above equations are provided in (2.16). It can be seen that if we put $\tau = 0$ in (2.29), then (2.29) becomes $\dot{y} = -Ky + Kw$. Now, defining

$$\Delta_m = \frac{Y(t)}{Y(t - \tau)}, \quad \Delta_a = \frac{X(t - \tau)Y(t) - X(t)Y(t - \tau)}{Y(t - \tau)} \quad (2.32)$$

we can rewrite (2.29) as:

$$\dot{y} = -K\Delta_my(t - \tau) + \Delta_a + K\Delta_mw \quad (2.33)$$

The effect of delay thus shows up in the quantities $\Delta_m$ and $\Delta_a$. Here, $\Delta_m$ and $\Delta_a$ both vary with time, and it can be seen that when $\tau$ is small, we have $\Delta_m \approx 1, \Delta_a \approx 0$. The effects of a time delay are thus the following:
(a) If $\Delta_m << 1$, it can result in $y$ taking too long to reach the reference value $w$, which will inhibit collision avoidance,

(b) If $\tau$ is larger than some threshold, then the system will become unstable causing $y$ to diverge away from 0, and

(c) Due to the presence of $\Delta_a$, the steady state value of $y$ will be different from the desired reference value $w$.

For successful collision avoidance, we require the following:

(a) $K\Delta_m$ to be greater than the threshold indicated in (2.22), which can be achieved if we employ a time-varying gain $K(t)$, such that $\dot{K} = K(t)\Delta_m$ is a constant greater than the threshold indicated in (2.22).

(b) $\tau$ to be less than a threshold, in order to ensure (2.33) is stable. An approximation to this threshold is determined as follows. With a constant $\hat{K}$ chosen as in (a) above, we can re-write (2.33) as:

$$\dot{y} = -\hat{K}y(t - \tau) + \Delta_a + \hat{K}w$$

(2.34)

In scenarios where $\Delta_a$ is slowly varying, one can employ a frozen-time LTI approximation on (2.34), with such a system having a constant value of $\Delta_a$.

$$sy(s) - y(0) = -\hat{K}e^{-\tau s}y(s) + \Delta_a(s) + \hat{K}w(s) \Rightarrow y(s) = \frac{y(0) + \Delta_a(s) + \hat{K}w(s)}{s + \hat{K}e^{-\tau s}}$$

(2.35)

The smallest $\tau$ for which (2.35) becomes unstable is found by substituting $s = j\omega$ in the denominator of (2.35):

$$j\omega + \hat{K}e^{-j\omega\tau} = 0 \Rightarrow \cos(\omega\tau) = 0, \ \omega - \hat{K}\sin(\omega\tau) = 0 \Rightarrow \tau_{max} = \frac{\pi}{2\hat{K}}$$

(2.36)

where $\tau_{max}$ represents the maximum value of $\tau$ for which the system remains stable. In other words, $\hat{K}$ needs to satisfy

$$\hat{K} < \frac{\pi}{2\tau_{max}}$$

(2.37)
Thus, while $\hat{K}$ needs to be larger than the threshold of (2.22) for successful collision avoidance, it also needs to be smaller than the threshold (2.37) to ensure robustness to time delays. There is thus an inherent trade-off between collision avoidance performance and robustness to time-delays.

(c) The reference value has to be adjusted to a new value $w_{\text{new}} = w + \frac{\Delta a}{K}$.

The statements (a)-(c) above pertain to a robustness analysis of the developed guidance law. In order to determine numerical values of $\hat{K}$ and $w_{\text{new}}$ on-line, we need to estimate $\Delta_m$ and $\Delta_a$ on-line, which in principle can be achieved using a Smith Predictor.

### 2.4 Simulation Results

The above collision avoidance laws are demonstrated through simulations. While these laws (2.21) and (2.28) hold for any arbitrary object shapes, the following non-convex shape is used for the purpose of these simulations:

$$X(\theta_R) = R_x \cos \theta_R, \quad Y(\theta_R) = R_y \sin \theta_R - K_r R_y \sin^3 \theta_R$$

(2.38)

where $(X, Y)$ are the coordinates of points on the boundary of the object, $R_x$ is the distance from the center of the object to the farthest point on its boundary along X and Y directions respectively and angle $\theta_R$ varies from $0^\circ$ to $360^\circ$. $K_r$ is the constant used to obtain different shapes. This shape is shown in Fig 2.6.

In the above equation, $(x_i, y_i)$ represent a local co-ordinate frame. When the heading angle of the vehicle is at an angle $\alpha_i$, as shown in Fig 2.6, the shape of each object assumes the following expression in a global $(X, Y)$ co-ordinate frame:

$$X_i = X_{ci} + x_i \cos \alpha_i - y_i \sin \alpha_i$$
$$Y_i = Y_{ci} + x_i \sin \alpha_i + y_i \cos \alpha_i$$

(2.39)

where, $(X_{ci}, Y_{ci})$ represent the coordinates of the center of the $i$th object in the $(X, Y)$ frame. Fig 2.6 shows that the shape of the object is not omni-directional, and furthermore, as the
heading angle of the object changes, the orientation of the object changes as well, and this influences the relative geometry between the objects $A$ and $B$.

![Figure 2.6: Shape of the robot and obstacles considered in this simulation](image)

Without loss of generality, it is assumed that the robots and obstacles all have the same shape indicated by (2.38). Consider a robot $A$ initially positioned with its center at the origin, and having to navigate to a goal point at $(11.5, 2.75)$. Three pop-up obstacles $B_1, B_2, B_3$, appear at the time instants $t = 0$ sec, $t = 5$ sec and $t = 10$ sec as shown in Fig 2.7.

The initial conditions of the robot to the first obstacle are given by $\hat{r}_1(0) = 2.3$ m, $\hat{\theta}_1(0) = 0.5 \text{ rad}$, $\hat{V}_{\theta 1}(0) = -0.03 \text{ m/sec}$ and $\hat{V}_{r 1}(0) = -0.7 \text{ m/sec}$. From (2.12), it is seen that this corresponds to $y_1(0) < 0$ which (in conjunction with $V_{r 1} < 0$) indicates that the velocity vector of $A$ lies inside the collision cone to $B_1$. The guidance law of (2.21) is applied, using a value of $\delta_A$ given by $\delta_A = \alpha + \pi/2$. A non-negative reference value $w = 0.1$ was chosen.

As seen in Fig 2.8, the initial negative value of $y_1$ is driven to $w = 0.1$ around $t = 1$ sec, by application of the acceleration, which in turn leads to a change in the angle of the velocity vector $\alpha$ (See Fig 2.11). When $y_1$ has become positive, the velocity vector of $A$ is now outside the collision cone to $B_1$. At around $t = 3$ sec, which is when $A$ passes $B$, $A$ switches to a conventional PN (Proportional Navigation) guidance law to guide it towards the original goal point. At $t = 5$ sec, the second obstacle $B_2$ appears and as seen in Fig 2.8, $y_2(5)$ is negative which (in conjunction with the negative value of $V_{r 2}(5)$) indicates that the velocity vector of $A$ is inside the collision cone to $B_2$. The collision avoidance law of (2.21) kicks in
again, and leads to the acceleration spike around the $t = 5$ sec mark, which drives $y_2(t)$ to 0.1 by causing a change in the heading $\alpha$. Shortly after $t = 8$ sec, $A$ has bypassed $B_2$ and switches back to the PN law to guide it towards the goal point leading to the acceleration $a_A$ and $\alpha$ profiles shown in Fig [2.11] A third obstacle $B_3$ appears at $t = 10$ sec and the switch to the collision avoidance law, followed by a switch back to the PN law (after collision avoidance) repeats.

Figure 2.7: Robot and Obstacle trajectories
Figure 2.8: Time histories of outputs $y_1$, $y_2$, $y_3$ to the three obstacles $B_1$, $B_2$, $B_3$ respectively.

Fig 2.7 shows the robot and obstacle spatial path trajectories. As can be seen, the collision avoidance laws cause the robot to take a curved path to the goal point, in order to avoid the obstacles. The time histories of $\dot{V}_r$ for each of the three obstacles are shown in Fig 2.9. Fig 2.10 demonstrates the angle $\psi$ from the robot $A$ to each of the obstacles $B_1$, $B_2$ and $B_3$, which is used in the collision avoidance law (2.21). The $a_A$ and $\alpha$ profiles for the entire simulation are shown in Fig 2.11.
Figure 2.9: $\hat{V}_r$ (m/sec) vs Time (sec) for the three obstacles

Figure 2.10: $\psi$ (degrees) vs Time (sec) for the three obstacles
Figure 2.11: Time histories of (a) Acceleration $a_A$, (b) Heading angle $\alpha$
CHAPTER 3
COOPERATIVE COLLISION AVOIDANCE AND FORMATION CONTROL
FOR OBJECTS WITH HETEROGENEOUS SHAPES

The collision avoidance laws in cooperative environments are presented in this chapter. Two objects \(A\) and \(B\) may share information of each other and utilize this to cooperatively avoid collision. In Section 3.1, we design two cooperative collision avoidance laws: one in which the robots change the magnitudes of their accelerations to generate trajectories that ensure collision avoidance, and another in which they change the angle at which they apply their accelerations (while keeping the magnitudes constant). In Section 3.2, we review a basic formation control law, and then demonstrate how the developed collision avoidance laws can be interspersed with the formation control laws. Section 3.3 presents simulations that demonstrate the working of the formation control law interspersed with the collision avoidance laws.

3.1 Cooperative Collision Avoidance Laws

We utilize the collision cone concept presented in [99] in designing two cooperative avoidance laws here. We consider the engagement geometry between two arbitrary shaped objects \(A\) and \(B\) as shown in Figure 2.3. The kinematics of this engagement governing the Line-of-Sight (LOS) are as given in 2.13.

The objects \(A\) and \(B\) are on a collision course if their relative velocities belong to a specific set. As was demonstrated in [99], this set of relative velocities can be encapsulated in a quantity \(y\), which is defined by equation 2.12.

The collision cone is then defined as the region in \((V_\theta, V_r)\) space for which \(y < 0\), and \(V_r < 0\) is satisfied. The lemmas (2.1 - 2.3) presented in Section 2.2 are applicable here as well.

In contrast to Sub-Sections 2.2.1 and 2.2.2 which dealt with non-cooperative collision avoidance, in the next two sub-sections, we determine cooperative collision avoidance laws.
that drive \( y(t) \) from an initial negative value to a non-negative value. This is equivalent to \( A \) and \( B \) cooperatively driving their relative velocity vector (initially lying inside the collision cone), to outside the collision cone.

### 3.1.1 Cooperative Collision Avoidance Law with Acceleration Magnitudes as Primary Inputs

In this section, the acceleration magnitudes \( a_A \) and \( a_B \) are treated as the primary inputs used to drive \( y(t) \) from an initial negative value to a non-negative value. This is equivalent to determining accelerations \( a_A \) and \( a_B \) that drive the velocity vectors of \( A \) and \( B \) from the interior of the collision cone to the boundary of, or outside, the collision cone.

Considering the equation \(2.17\) obtained by using dynamic inversion techniques, it can be seen that this equation in general has infinite number of solutions. While, in section 2.2 non-cooperative environment was assumed, here both objects \( A \) and \( B \) are moving with varying accelerations \( a_A \) and \( a_B \). Thus, any pair of accelerations \( a_A \) and \( a_B \) that satisfy \(2.17\) will cooperatively drive the output \( y \) to the chosen reference value \( w \geq 0 \). One such candidate solution is the following:

\[
\begin{align*}
  a_A &= \frac{K(w - y)(V^2_\theta + V^2_\tau)^2 + Y_a}{2N_{a,A}} \\
  a_B &= \frac{K(w - y)(V^2_\theta + V^2_\tau)^2 + Y_a}{2N_{a,B}}
\end{align*}
\]

The working of the above collision avoidance law is now illustrated by simulating a scenario comprising of two objects \( A \) and \( B \) of arbitrary shapes. These objects are initially inside the collision cone to one another, and would have collided in the absence of application of any collision avoidance accelerations.

In the simulation, \( A \) is moving at a constant speed of 3 m/s with an initial velocity heading angle \( \alpha_A = 45^\circ \), while \( B \) is moving at a constant speed of 4 m/s with an initial heading angle \( \alpha_B = -135^\circ \). These initial conditions put them on a collision course with
each other, and, if they continue to move forward with constant velocities, A and B will encounter a collision. The value of $y(0) = -1 < 0$, as shown in Figure 3.1(c).

Figure 3.1: Acceleration, output and heading angle time histories obtained using a cooperative collision avoidance law
Using the cooperative collision avoidance accelerations $a_A$ and $a_B$ of (3.1), (3.2), $A$ and $B$ are able to cooperatively drive $y(t)$ to zero. The acceleration magnitudes are shown in Fig 3.1(a),(b), and the time history of $y(t)$ is shown in Figure 3.1(c). These accelerations are applied at angles of $\delta_A = \alpha_A + \frac{\pi}{2}, \delta_B = \alpha_B + \frac{\pi}{2}$, that is, they are applied orthogonal to the respective velocity vectors. As a consequence of these accelerations, the heading angles $\alpha_A$ and $\alpha_B$ get modified, as shown in Figure 3.1(d),(e). A zero output value of $y$ means that the velocity vectors of both the objects align with the boundary of their respective collision cones to one another, thereby making the two objects just graze each other as can be seen in the $XY-$ plot of Fig 3.2, with the grazing occurring at $t = 0.8$ sec.

![Figure 3.2: XY trajectories of Objects A and B](image)

### 3.1.2 Cooperative Collision Avoidance Law with Acceleration Directions $\delta_A$ and $\delta_B$ as Primary Inputs

In this section, the acceleration magnitudes $a_A$ and $a_B$ are kept constant, while the directions of the acceleration vectors are treated as the primary inputs, and these inputs are used to drive $y(t)$ from an initial negative value to a non-negative value. In other words, $\delta_A$ and $\delta_B$ are modulated so as to drive the velocity vectors of $A$ and $B$ from the interior of the collision cone to the boundary of, or outside, the collision cone.
As in the case of equation (2.17) derived in Chapter 2, (2.24) also has an infinite number of solutions for $\dot{\delta}_A$ and $\dot{\delta}_B$. Depending on the requirements different solutions can be chosen. A candidate solution is as follows:

$$
\dot{\delta}_A = \frac{(K_1(w - y) - K_2\dot{y})(V_d^2 + V_f^2)^2 + Y_d}{2N_{d,A}} \tag{3.3}
$$

$$
\dot{\delta}_B = \frac{(K_1(w - y) - K_2\dot{y})(V_d^2 + V_f^2)^2 + Y_d}{2N_{d,B}} \tag{3.4}
$$

The $\dot{\delta}_A$ and $\dot{\delta}_B$ from the above equation are integrated with respect to time to obtain $\delta_A$ and $\delta_B$, which then control the direction of application of the acceleration vectors appropriately, so as to cooperatively drive $y(t)$ to reference value $w \geq 0$.

### 3.2 Combining Collision Avoidance with Formation Control

#### 3.2.1 Formation Control

In the formation control problem, a collection of robots exchange information with each other based on the communication topology of the network, in order to achieve a desired formation. We consider $N$ robots on a plane, each with double-integrator dynamics. Then, the state equations for the $i$th robot are: $\dot{x}_i = Ax_i + Bu_i$, where $i = [1, 2, 3, \ldots, N]$, $x_i \in \mathbb{R}^4$ is the state vector containing the (planar $XY$) position and velocity states of each vehicle, and $u_i \in \mathbb{R}^2$ contains the acceleration control inputs of each vehicle. The matrices $A$ and $B$ are as follows:

$$
A = \begin{bmatrix}
0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0
\end{bmatrix}, \quad B = \begin{bmatrix}
0 & 0 \\
1 & 0 \\
0 & 0 \\
0 & 1
\end{bmatrix} \tag{3.5}
$$

Using the Kronecker product, the state space equations for $N$ vehicles are as follows:

$$
\dot{x} = (I_N \otimes A)x + (I_N \otimes B)u \tag{3.6}
$$
Using $L$ and $\zeta$ to represent the Laplacian matrix of the interconnection topology and the relative coordinates of the robots in the desired formation, respectively, the following control law ensures the desired formation is achieved [24]:

$$u = a_F = -L(x + \zeta)$$  \hspace{1cm} (3.7)

### 3.2.2 Integrating Collision Avoidance Laws with Formation Control

In this section, we propose a switching mechanism to integrate the formation control law and collision avoidance law, thereby defining a hybrid control law. Here, we point out that the collision avoidance acceleration is presented in polar coordinates while the formation control acceleration is presented in cartesian coordinates. The reason the collision avoidance law is provided in polar co-ordinates is because it makes use of the $y$ function (2.12), which is similar to a “miss-distance” and is inherently a radial quantity. However, in integrating formation control with collision avoidance, the relevant coordinate transformation is made appropriately whenever switching is required. The integrated control acceleration is defined as below:

$$a_i = K_{F,i}a_{F,i} + K_{G,i}a_{G,i}$$  \hspace{1cm} (3.8)

where $K_{F,i}$ and $K_{G,i}$ are switching constants chosen appropriately to switch between formation control and collision avoidance as given below:

$$K_{G,i} = \begin{cases} 
1 & \text{if } y_{ij} < 0 \text{ and } V_{r,ij} < 0, \text{ for some } j \in N \\
0 & \text{Otherwise}
\end{cases}$$  \hspace{1cm} (3.9)

$$K_{F,i} = \begin{cases} 
0 & \text{if } y_{ij} < 0 \text{ and } V_{r,ij} < 0, \text{ for some } j \in N \\
1 & \text{Otherwise}
\end{cases}$$  \hspace{1cm} (3.10)
In the above equations, if a pair of robots \(i, j\) are on a collision course, we then have that 
\[
a_{G,i} = |a_{G,i}| \begin{bmatrix} \cos \delta_i, & \sin \delta_i \end{bmatrix}^T, \quad a_{G,j} = |a_{G,j}| \begin{bmatrix} \cos \delta_j, & \sin \delta_j \end{bmatrix}^T.
\]
If the first collision avoidance law is used, then the acceleration magnitudes \(|a_{G,i}|, |a_{G,j}|\) are obtained from (3.1), (3.2) and the acceleration angles are \(\delta_i = \alpha_i + \frac{\pi}{2}\). If the second collision avoidance law is used, then \(|a_{G,i}| = |a_{G,j}| = 1\) (or any other constant), while \(\delta_i, \delta_j\) are obtained from (3.3), (3.4).

We point out that in the above equations, while the formation control law is implemented in a distributed fashion (with each robot using the position and velocity information of all its neighbors on the communication graph), the collision avoidance law is implemented in a pairwise fashion (and these pairs are a subset of the robot pairs that have a connecting edge on the communication graph). This is done as follows. If a robot \(i\) finds that (among all its neighbors on the communication graph), its velocity vector is inside the collision cone to just one other robot \(j\), then the pair of robots \(i\) and \(j\) perform cooperative collision avoidance, as discussed earlier. On the other hand, if robot \(i\) finds that its velocity vector is inside the collision cone to more than one robot, say \(j\) and \(k\) (assuming again, that \(j\) and \(k\) are both neighbors of \(i\) on the communication graph), it prioritizes the collision cone which it should first come out of, by using a prediction of the times of closest approach \(t_{m,ij}\) and \(t_{m,ik}\). The time of closest approach \(t_m\) is itself a function of time \(t\), that is, \(t_m = t_m(t)\). The time to collision between \(i\) and \(j\) is given by the following equation:

\[
t_{m,ij} = \frac{-r_{ij} V_{r,ij}}{V_{r,ij}^2 + V_{\theta,ij}^2}
\]  

(3.11)

Note that (3.11) assumes \(V_{r,ij} < 0\), thereby ensuring positive values of time \(t_{m,ij}\). When \(V_{r,ij} > 0\), that means that \(i\) and \(j\) are moving away from each other, and during this time, the concept of time to collision obviously does not make sense, and therefore \(t_{m,ij}\) is not used in the computations. Thus, when robot \(i\) is inside the collision cones to robots \(j\) and \(k\), \(t_{m,ij}\) and \(t_{m,ik}\) are computed, following which \(\min(t_{m,ij}, t_{m,ik})\) is found. Robot \(i\) will then generate
the requisite acceleration to first come out of the collision cone to that robot which has the smaller $t_m$. Scenarios wherein robot $i$ is inside the collision cone to more than two robots are handled similarly.

3.3 Simulation Results

In this section, we present simulations demonstrating the integrated formation control with collision avoidance laws. We consider a scenario wherein four robots are initially located at the four corners of a square of 4 m length in an inertial $XY$ frame, with their velocity vectors all pointing towards the center. The four objects are tasked to make a square formation of 8 m in length, while utilizing the cooperative collision avoidance laws to determine their acceleration $a_{G,i}$ and the formation control laws to determine their accelerations $a_{F,i}$. The equations corresponding to the shapes of the objects are same as defined in 2.38. The values chosen for the four objects for the simulations are $R_{1,x} = -1.8, R_{1,y} = -1, K_{r1} = -0.5$, $R_{2,x} = -1.8, R_{2,y} = 1.8, K_{r2} = 0.75$, $R_{3,x} = -1.8, R_{3,y} = 1, K_{r3} = 0.1$, and $R_{4,x} = -0.7, R_{4,y} = 0.7, K_{r4} = -1$. The four objects are shown in Fig 3.3, and it can be seen that they are all of different shapes, thereby contributing to a heterogeneous environment.

![Figure 3.3: Shapes of objects A, B, C and D](image)

39
The initial velocity vectors of the objects are such that each lie inside the collision cone to the other. Table 3.1 shows the initial conditions of each of the six pairs of objects. As is evident from the table, $y(0) < 0$ and $V_r(0) < 0$ for all six pairs, implying the four objects are all on a collision course to each other. The topology of the network is a (connected) path graph, with edges $AB$, $BC$, $CD$, and $DA$.

As can be seen from the above table, each robot is inside the collision cone to every other robot resulting in multiple overlapping collision cones. Of the six robot pairs, we first exclude those that do not have an edge between them (for the path graph considered in this example, we would exclude the pairs $AC$ and $BD$). Among the remaining four pairs, each
Table 3.1: Kinematic states at \( t = 0 \)

<table>
<thead>
<tr>
<th>ICs</th>
<th>( r_0(m) )</th>
<th>( \theta_0(rad) )</th>
<th>( V_{\theta 0}(m/s) )</th>
<th>( V_{r 0}(m/s) )</th>
<th>( y_0 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( AB )</td>
<td>4</td>
<td>0</td>
<td>0</td>
<td>-0.2</td>
<td>-0.57</td>
</tr>
<tr>
<td>( AC )</td>
<td>5.6</td>
<td>0.78</td>
<td>0</td>
<td>-0.29</td>
<td>-0.22</td>
</tr>
<tr>
<td>( AD )</td>
<td>4</td>
<td>1.57</td>
<td>0</td>
<td>-0.2</td>
<td>-0.46</td>
</tr>
<tr>
<td>( BC )</td>
<td>4</td>
<td>1.57</td>
<td>0</td>
<td>-0.2</td>
<td>-0.45</td>
</tr>
<tr>
<td>( BD )</td>
<td>5.6</td>
<td>2.35</td>
<td>0</td>
<td>-0.29</td>
<td>-0.22</td>
</tr>
<tr>
<td>( CD )</td>
<td>4</td>
<td>3.1</td>
<td>0</td>
<td>-0.2</td>
<td>-0.40</td>
</tr>
</tbody>
</table>

robot is inside the collision cone to two other robots. We use the time of closest approach \( t_m \) to decide which collision cone each robot needs to first come out of. For each robot, the corresponding \( t_m \) to its two other (neighbor) robots is computed. The one with the smallest time to collision is chosen and the corresponding collision avoidance acceleration is applied to come out of that collision cone.

![Figure 3.5: Instances where the circular approximations to the objects would have collided, even though the actual objects did not collide](image)

The collision cone-based guidance law (3.1), (3.2) is applied, using \( \delta_{G,i} = \alpha_{G,i} + \pi/2 \). This value of \( \delta_{G,i} \) implies that the acceleration vector acts normal to the velocity vector of the robot. Figure 3.4 shows the robots’ spatial path trajectories for the entire time duration.
As can be seen from Figure 3.4, the objects achieve the desired formation while utilizing
the cooperative formation and guidance laws. Figure 3.5 shows two sample time instants
\((t = 1.9\ \text{sec} \text{ and } t = 5.3\ \text{sec})\) at which a pair of objects came quite close to each other, along with their circular approximations. As is clearly evident from Fig 3.5, had a circular approximation for the objects been employed, then the circles representing objects \(C\) and \(D\) would have collided at \(t = 5.3\ \text{sec}\), even though the actual objects did not. The same can be seen for objects \(B\) and \(C\) at \(t = 1.9\ \text{sec}\). Thus using the exact shape of the objects instead of approximating them with circles makes close maneuvers possible. This can provide an advantage in dense swarms, since it increases the amount of free space that the objects have, within which they can maneuver.

Figure 3.6: Output \(y\) corresponding to the six relative pairs.

The corresponding collision avoidance accelerations \(a_{G,i}\) generated can be seen in Figure 3.8 and the ensuing change in the robot’s heading angles is seen in Figure 3.9. At \(t = 0\), the collision avoidance accelerations are employed by all the four objects since they are all inside the collision cone to one another. When the \(ith\) robot’s velocity vector is not inside
any of the collision cones, it employs its formation control law. Application of the formation control law may get the robot’s velocity vector back into the collision cone, in which case, it switches back to the collision avoidance law. Figure 3.6 shows the plots of \( y \) for each robot pair, and it can be seen that the initial negative values of \( y \) are driven to become positive, by application of the collision avoidance law. The kinematic state trajectories are shown in Figure 3.7 while the applied accelerations are shown in Figure 3.8. The time intervals during which the collision avoidance acceleration is applied are shown in blue, while those during which the formation control acceleration is applied are shown in green. The heading angles of the four robots are shown in Figure 3.9 where again the time intervals during which the heading angles changes are generated by the collision avoidance acceleration are shown in blue, while those during which the heading angle changes are generated by the formation control acceleration are shown in green.

![Kinematic state trajectories and applied accelerations](image)

**Figure 3.7:** Kinematic state trajectories of each relative pair of objects: (a) \( \hat{r} \), (b) \( \hat{\theta} \), (c) \( V_\theta \) and (d) \( V_r \), for each relative pair
Figure 3.8: Accelerations of each object: (a) $A$, (b) $B$, (c) $C$ and (d) $D$

Figure 3.10 demonstrates the angle $\psi$ for all the six pairs of objects, as utilized in the collision avoidance laws. Figure 3.11 shows the consensus in the velocities of the four objects, as they achieve their desired formation. Finally, Figure 3.12 shows the trajectories of objects $A, B, C$ and $D$, when the communication topology is a complete graph, for the same initial conditions.
Figure 3.9: Heading angles of objects A, B, C and D

Figure 3.10: Time histories of $\psi$ for all object pairs
Figure 3.11: Positions and velocities of objects $A$, $B$, $C$ and $D$
Figure 3.12: XY trajectories of objects A, B, C and D when the network topology is a complete graph
CHAPTER 4

COLLISION AVOIDANCE BY IPMC ACTUATED ROBOTIC FISH USING COLLISION CONE APPROACH

This chapter analyzes an integrated model comprising the nonlinear guidance law for collision avoidance, the nonlinear fish dynamics of the 2D maneuverable robotic fish [96], and the IPMC fin dynamics [33] all integrated in a closed-loop system. Section 4.1 discusses the fish dynamic model, the IPMC fin dynamics and the 2D maneuverable robotic fish dynamics. Section 4.2 discusses the nonlinear guidance law, the conversion of the acceleration command generated by this guidance law into non-negative fin thrusts, and its integration with the fish dynamic and IPMC actuator model. Section 4.3 presents the simulations of this integrated system.

4.1 Robotic Fish Dynamic Model

4.1.1 Fish Body Dynamics

Refer Figure 4.1, which shows a schematic of the robotic fish. A body-fixed XY co-ordinate frame is defined such that the X-axis points out the nose of the robotic fish, and the Y-axis is positive towards its left (See Figure 4.1). The equations of motion of the robotic fish are [41]:

\[
\begin{align*}
\dot{u} &= \frac{m_b - m_y}{m_x} v r + \frac{f_x}{m_x} \\
\dot{v} &= \frac{m_b - m_x}{m_y} u r + \frac{f_y}{m_y} \\
\dot{r} &= \frac{\tau_z}{I_z} \\
\dot{\phi} &= r
\end{align*}
\]  

(4.1)  

(4.2)  

(4.3)  

(4.4)

Here, \( u \) and \( v \) represent the velocity components along the X and Y axes, while \( f_x, f_y \) represent the respective forces. \( m_x \) and \( m_y \) are the robot effective masses along the X and Y directions, respectively, and \( I_z \) is the effective inertia about the z axis. \( \phi \) represents the
heading angle, defined as the angle between the body-fixed \( XY \) coordinate frame and an inertial reference frame. \( r \) is the angular velocity of the fish body.

![Figure 4.1: Robotic fish in body-fixed coordinate axes](image)

When a body moves in a fluid, a portion of the surrounding fluid also moves with the body. This is called the added mass effect. We consider the added mass effect as an additional component \( m_a \) added to the body mass \( m_b \). Along similar lines, we also consider the effect of inertia of the surrounding fluid by means of an added-inertia effect.

The added mass \( m_{ay} \) along the \( y \) direction and added inertia along the \( z \)-direction can be obtained using slender body theory \[100\]:

\[
m_{ay} = \pi \rho R^2 (x_2 - x_1) \quad (4.5)
\]

\[
I_{az} = \pi \rho R^2 \frac{x_2^3 - x_1^3}{3} \quad (4.6)
\]

where \( x_1 \) and \( x_2 \) are the pectoral fin locations. For calculating the added mass \( m_{ax} \) along the \( x \)-direction, we use a procedure which considers the robot body as an ellipsoid of length of \( l_e \) and diameter \( d_e \) \[100\]:

\[
m_{ax} = \frac{2\pi l_e d_e^2}{6} \quad (4.7)
\]

Finally, \( m_x, m_y \) and \( I_z \) are obtained as follows:

\[
m_x = m_b + m_{ax}; \quad m_y = m_b + m_{ay}; \quad I_z = I_b + I_{az} \quad (4.8)
\]
The forces along $x$ and $y$ directions are computed from the thrust and drag forces as follows.

\[
\begin{align*}
    f_x &= T_c + T_r \cos(\theta_P) + T_l \cos(\theta_P) - F_D \cos(\beta) \\
    f_y &= T_r \sin(\theta_P) - T_l \sin(\theta_P) - F_D \sin(\beta) \\
    \tau_z &= M_h + M_D
\end{align*}
\] (4.9-4.11)

In the above equations, $T_c$ is the caudal actuator thrust, $T_r$ and $T_l$ are the right and left pectoral fin thrusts, $\theta_P$ is the angle between $x$ axis and right pectoral fin. The two pectoral fins are symmetrically installed. The angle $\beta$ is defined as $\tan^{-1}(\frac{u}{v})$. $M_h$ is the total hydrodynamic moment and is given by:

\[
M_h = \vec{r}_{Cr} \times \vec{T}_r + \vec{r}_{Cl} \times \vec{T}_l
\] (4.12)

Where $\vec{r}_{Cr}$ is the vector from the body center $C$ to the base of the right pectoral fin, and $\vec{r}_{Cl}$ is the corresponding vector to the base of the left pectoral fin. $F_D$ is the drag force, and $M_D$ is the moment due to drag, and these are obtained from the following equations:

\[
\begin{align*}
    F_D &= \frac{1}{2} \rho U^2 S_A C_D \\
    M_D &= -C_M r^2 \text{sgn}(r)
\end{align*}
\] (4.13-4.14)

where $S_A$ is the robotic fish wetted surface area, $C_D$ is the drag coefficient, $U$ is the speed of the robotic fish, and $C_M$ is the drag moment coefficient.

4.1.2 IPMC Fin Dynamics

An IPMC consists of an ion exchange membrane coated with two novel metal electrodes [101], such as gold or platinum. Application of a small voltage to the IPMC leads to ion transportation to the cathode side, which causes a swelling effect on that side and a shrinking effect on the anode side. Eventually, the IPMC bends to the anode side, thus
realizing the actuation effect. Chen et. al. obtained the equations governing the thrust generated by an IPMC actuator in water using Lighthill theory of elongated-body [33]. Based on this theory, the mean thrust $\overline{T}$ produced by an IPMC actuator is:

$$\overline{T} = \left[ \frac{m}{2} \left( \left( \frac{\partial w(z,t)}{\partial t} \right)^2 - U^2 \left( \frac{\partial w(z,t)}{\partial z} \right)^2 \right) \right]_{z=L_1}$$

(4.15)

where $w$ is the actuator bending displacement, $U$ is the fish velocity, $z = L_1$ is the length of the tail, $\langle \cdot \rangle$ denotes the mean value, and $m$ is the virtual mass density at $z = L_1$ which can be expressed as:

$$m = \frac{1}{4} \pi S_c^2 \rho_w B$$

(4.16)

where $S_c$ is the width of the tail at the end, $\rho_w$ is the water density, and $B$ is a non-dimensional parameter close to 1. In (4.15), $\frac{\partial w(z,t)}{\partial t}$ is the lateral velocity of the IPMC actuator, and $\frac{\partial w(z,t)}{\partial z}$ is the slope at the tip of the tail ($z = L_1$). Figure 4.2 shows a hybrid actuator which comprises the IPMC beam, with a passive fin attached to its tip.

![Figure 4.2: Illustration of an IPMC beam with passive fin](image)

When a sinusoidal input voltage is applied to the IPMC, the relevant equations governing the tip displacement and tip slope are [33]:

$$w(L_1,t) = A_m |H(j\omega)| \sin(\omega t + \angle H(j\omega))$$

(4.17)

$$\frac{\partial w(z,t)}{\partial z} \bigg|_{z=L_1} = A_m |H_d(j\omega)| \sin(\omega t + \angle H_d(j\omega))$$

(4.18)

where $A_m$ and $\omega$ are the amplitude and frequency of the applied voltage, respectively. $H(j\omega)$ and $H_d(j\omega)$ are the IPMC transfer functions that were derived in [102], and $\angle (\cdot)$ denotes
the phase angle. We note that $H(j\omega)$ and $H_d(j\omega)$ are both irrational transfer functions, since the flexible IPMC is modeled as an Euler-Bernoulli beam which is represented by a partial differential equation.

### 4.1.3 Description of the Robotic Fish

Fig. 4.3 shows an assembled robotic fish [96], whose dimensions and parameters were used for the model simulation and analysis in this paper. In the robotic fish, a microcontroller board (Nano, Arduino) was used to generate three voltage signals applied to the caudal fin and pectoral fins, respectively. A XBee communication device was used to transmit commands from a base station. Three H-Bridges (Gravitech, 2MOTOR-4NANO) were used to amplify the actuation driving current to propel the pectoral and caudal fins. A lithium ion polymer battery (Tenergy, 7.4V 6000 mAh) was used to provide electricity to the robotic fish. The total weight of the robot was 290 grams. Overall, the fish had slightly positive buoyancy. Table 4.1 displays the dimensions of the IPMC and the fish body.

![Assembled Robotic Fish](image)

**Figure 4.3:** Assembled robotic fish [96].
### Table 4.1: Size information of IPMC and fish body

<table>
<thead>
<tr>
<th></th>
<th>( W_0 )</th>
<th>( b_0 )</th>
<th>( b_1 )</th>
<th>( L_1 )</th>
<th>( L_0 )</th>
<th>( L_1 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( x_1 )</td>
<td>0.015 m</td>
<td>0.02 m</td>
<td>0.04 m</td>
<td>0.023 m</td>
<td>0.018 m</td>
<td>0.058 m</td>
</tr>
<tr>
<td>( x_2 )</td>
<td>0.075 m</td>
<td>0.157 m</td>
<td>0.05 m</td>
<td>0.23 m</td>
<td>0.08 m</td>
<td>75.3°</td>
</tr>
</tbody>
</table>

### 4.2 Collision Cone

#### 4.2.1 Guidance Law for Collision Avoidance

In this section, we utilize the guidance laws presented in Section 2.2 in determining collision avoidance laws for dynamic IPMC actuated robotic fish. Refer to Figure 2.3 in Chapter 2 which shows the engagement between two arbitrary shaped objects \( A \) and \( B \). Without loss in generality, object \( A \) in Figure 2.3 is considered as robotic fish and object \( B \) as an arbitrarily shaped obstacle in this chapter. The corresponding kinematic equations of the engagement between the robotic fish and the obstacle are as given in equation (2.13). The output here is defined as in equation (2.12), and the guidance law presented in 2.21 is used to drive the output \( y(t) \) to \( w \geq 0 \). This acceleration has to be converted to appropriate fin thrusts robotic fish dynamics can withstand. This is presented in below subsection.

#### 4.2.2 Conversion of commanded acceleration into fin thrusts

The acceleration generated by (2.21) now needs to be converted into thrusts \( T_c \) (of the caudal fin), and \( T_l, T_r \) (of the left and right pectoral fins). Collision avoidance of the fish with the obstacle can be achieved in one of three different ways:

(a) Generate an appropriate moment using the pectoral fin thrusts to create a turning trajectory for the fish to move around the obstacle. The value of this moment is given by \( M_h = K a_A \), where \( K \) is an appropriate proportional gain, \( a_A \) is the acceleration generated by (2.21), applied at an angle \( \delta_A = \alpha + \frac{\pi}{2} \). Depending on whether \( M_h \) needs to be clockwise or counter-clockwise, only one of \( T_l \) or \( T_r \) are non-zero. The corresponding \( T_l \) (or \( T_r \)) is determined using (4.12).

(b) Perform a pure translational acceleration using just the caudal fin alone, i.e., \( T_c > 0 \), while keeping the pectoral fin thrusts zero, i.e., \( T_l = 0, T_r = 0 \). This essentially involves a...
pure speed change maneuver performed by the fish, while keeping the direction of its velocity vector constant.

(c) Generate an appropriate combination of lateral and translational acceleration components, with $T_c > 0$, and either (i) $T_l > 0$, $T_r = 0$, or (ii) $T_l = 0$, $T_r > 0$. This essentially involves the fish generating a component of sideways motion in order to bypass the obstacle.

We point out that if we use (c) then, along with the sideways motion component generated by the corresponding pectoral fin thruster, a turning component is also generated. Similarly, if we use (a) then, along with the turning moment, a sideways motion component is also generated. The choice of which method to use depends on the location of the pectoral fins relative to the center of gravity of the fish. If the pectoral fins are so located that they generate relatively small turning moment, then one would prefer to use method (c), while if they are so located that they generate a large turning moment, then we would prefer to use (a).

Now, since $T_c$, $T_l$, and $T_r$ need to be non-negative at all times, this imposes a constraint on the $(a_A, \delta_A)$ pairs that can be used to achieve collision avoidance, at each instant in time. In other words, when $y < 0$, $V_r < 0$, $(a_A, \delta_A)$ need to satisfy not just the acceleration equation \[2.21\] for collision avoidance, but also the constraints $T_c \geq 0$, $T_l \geq 0$, $T_r \geq 0$. For case (a), the thrusts are always positive, while for case (b) one can ensure that the thrusts are positive simply by choosing the angle $\delta_A = \phi$.

For case (c), more analysis is required. From (4.9), (4.10), the thrust equations can be written as:

\[ T_c + T_l \cos \theta_P + T_r \cos \theta_P = y_1 \tag{4.19} \]
\[ T_r \sin \theta_P - T_l \sin \theta_P = y_2 \tag{4.20} \]
where,

\[ y_1 = m_x a_A \cos(\delta_A - \phi) + F_D \cos \beta \]  
\[ y_2 = m_y a_A \sin(\delta_A - \phi) + F_D \sin \beta \]  

From (4.20)-(4.22), we see that in order to obtain positive thrusts \( T_l > 0 \) and \( T_c > 0 \), we need to satisfy the following two inequalities:

\[ y_2 < 0 \]  
\[ y_1 + y_2 \cot \theta_P > 0 \]

We now look for a range of \((a_A, \delta_A)\) pairs that satisfy both inequalities. Defining \( q = \frac{-F_D \sin \beta}{a_A m_y} \), we see that (4.23) can be satisfied as follows:

Case (1): When \( a_A > 0 \), and

(a) \( \beta \in (\pi, 2\pi) \): We need that \( \delta_A \in R_{1a} \), where

\[ R_{1a} = \{ \delta_A : \delta_A \in (0, \phi + \sin^{-1} q) \cup (\pi - \sin^{-1} q + \phi, 2\pi) \} \]  

(b) \( \beta \in (0, \pi) \): We need that \( \delta_A \in R_{1b} \), where

\[ R_{1b} = \{ \delta_A : \delta_A \in (\sin^{-1} q + \phi, 2\pi - \sin^{-1} q + \phi) \} \]

Case (2): When \( a_A < 0 \), and

(a) \( \beta \in (0, \pi) \): We need that \( \delta_A \in R_{2a} \), where

\[ R_{2a} = \{ \delta_A : \delta_A \in (\phi + \sin^{-1} q, \pi + \phi - \sin^{-1} q) \} \]
(b) $\beta \in (\pi, 2\pi)$: We need that $\delta_A \in R_{2b}$, where

$$R_{2b} = \{\delta_A : \delta_A \in \left(\phi, \pi + \sin^{-1} q + \phi\right) \cup \left(2\pi - \sin^{-1} q + \phi, 2\pi + \phi\right)\} \quad (4.28)$$

Next, considering the inequality (4.24), this can be rewritten in the quadratic form $a_0 Z^2 + a_1 Z + a_2 > 0$, where $Z = \tan(\delta_A/2)$ and

$$a_0 = -a_A m_x \cos(\phi) + F_D \cos(\beta) + a_A m_y \cot(\theta) \sin(\phi) + F_D \sin(\beta) \quad (4.29)$$
$$a_1 = 2a_A m_x \sin(\phi) + 2a_A m_y \cot(\theta) \cos(\phi) \quad (4.30)$$
$$a_2 = F_D \cos(\beta) - a_A m_x \cos(\phi) - a_A m_y \cot(\theta) \sin(\phi) + F_D \sin(\beta) \cot(\theta) \quad (4.31)$$

Defining the roots of the quadratic equation as $Z_{1,2}$, where

$$Z_{1,2} = \frac{-a_1 \pm \sqrt{a_1^2 - 4a_0a_2}}{2a_0} \quad (4.32)$$

the possible cases then are:

Case (3): When $a_0 > 0$, (4.24) is satisfied when $\delta_A \in R_{3a} \cup R_{3b}$, where:

$$R_{3a} = \{\delta_A : \delta_A \in \left(2 \tan^{-1} Z_1, \pi\right)\} \quad (4.33)$$
$$R_{3b} = \{\delta_A : \delta_A \in \left(-\pi, 2 \tan^{-1} Z_2\right)\} \quad (4.34)$$

Case (4): When $a_0 < 0$, (4.24) is satisfied when $\delta_A \in R_{4a} \cup R_{4b}$, where:

$$R_{4a} = \{\delta_A : \delta_A \in \left(2 \tan^{-1} Z_2, \pi\right)\} \quad (4.35)$$
$$R_{4b} = \{\delta_A : \delta_A \in \left(-\pi, 2 \tan^{-1} Z_2\right)\} \quad (4.36)$$
Combining all of the above cases, we need the following in order to satisfy the two inequalities (4.23), (4.24) simultaneously:

<table>
<thead>
<tr>
<th>Condition</th>
<th>Notation</th>
</tr>
</thead>
<tbody>
<tr>
<td>$a_a &gt; 0$, $\beta \in (\pi, 2\pi)$, $a_0 &gt; 0$</td>
<td>$\delta_a \in R_{1a} \cap {R_{3a} \cup R_{3b}}$</td>
</tr>
<tr>
<td>$a_a &gt; 0$, $\beta \in (\pi, 2\pi)$, $a_0 &lt; 0$</td>
<td>$\delta_a \in R_{1a} \cap {R_{4a} \cup R_{4b}}$</td>
</tr>
<tr>
<td>$a_a &gt; 0$, $\beta \in (0, \pi)$, $a_0 &gt; 0$</td>
<td>$\delta_a \in R_{1b} \cap {R_{3a} \cup R_{3b}}$</td>
</tr>
<tr>
<td>$a_a &gt; 0$, $\beta \in (0, \pi)$, $a_0 &lt; 0$</td>
<td>$\delta_a \in R_{1b} \cap {R_{4a} \cup R_{4b}}$</td>
</tr>
<tr>
<td>$a_a &lt; 0$, $\beta \in (0, \pi)$, $a_0 &gt; 0$</td>
<td>$\delta_a \in R_{2a} \cap {R_{3a} \cup R_{3b}}$</td>
</tr>
<tr>
<td>$a_a &lt; 0$, $\beta \in (0, \pi)$, $a_0 &lt; 0$</td>
<td>$\delta_a \in R_{2a} \cap {R_{4a} \cup R_{4b}}$</td>
</tr>
<tr>
<td>$a_a &lt; 0$, $\beta \in (\pi, 2\pi)$, $a_0 &gt; 0$</td>
<td>$\delta_a \in R_{2b} \cap {R_{3a} \cup R_{3b}}$</td>
</tr>
<tr>
<td>$a_a &lt; 0$, $\beta \in (\pi, 2\pi)$, $a_0 &lt; 0$</td>
<td>$\delta_a \in R_{2b} \cap {R_{4a} \cup R_{4b}}$</td>
</tr>
</tbody>
</table>

At each instant in time, the appropriate $(a_A, \delta)$ pair that satisfies (2.21) and the relevant condition in the above table is chosen. A corresponding set of equations can be determined when the fish needs to pass the obstacle on the left of the obstacle, with $T_r > 0$, $T_l = 0$.

4.2.3 Integrated system architecture

A block diagram of the overall integrated system is shown in Fig 4.4. The guidance law acceleration $a_A$ generated from (2.21) is converted into reference thrust commands $T_{c,R}$, $T_{l,R}$, and $T_{r,R}$ of the caudal, left and right pectoral fins, while ensuring that all these thrusts are non-negative using the equations described above. These reference thrusts are then converted into amplitudes $A_{m,c}$, $A_{m,l}$ and $A_{m,r}$ of sinusoidal voltages of the three IPMC actuators, using:

$$A_{m,c} = \sqrt{\frac{4T_{c,R}}{m(\omega^2|H|^2 - U^2|H_d|)^2}}$$  \hspace{1cm} (4.37)

$$A_{m,l} = \sqrt{\frac{4T_{l,R}}{m(\omega^2|H|^2 - U^2|H_d|)^2}}$$  \hspace{1cm} (4.38)

$$A_{m,r} = \sqrt{\frac{4T_{r,R}}{m(\omega^2|H|^2 - U^2|H_d|)^2}}$$  \hspace{1cm} (4.39)
where \( \omega \) is the frequency of the sinusoidal voltage, and \( H, H_d \) are IPMC transfer functions, derived in [102]. The IPMC actuators then convert these voltage inputs into actual thrusts \( T_c, T_l, \) and \( T_r, \) using (4.15), where the tip deflection and tip slope are computed using (4.17) and (4.18), respectively. These thrusts influence the robot fish dynamics (4.1)-(4.4). The movement of the robotic fish, along with the movements of the obstacle, influence the relative velocity kinematic states \( \hat{r}, \hat{\theta}, \hat{V}_\theta, \) and \( \hat{V}_r, \) shown in (2.13). These in turn influence the quantity \( y \) used in the collision cone computations, and shown in (2.12). The value \( y \) along with the relative kinematic states is then fed back towards computation of the guidance law acceleration \( a_A. \)

![Figure 4.4: Block diagram depicting the integration of the guidance law with the IPMC actuator and fish dynamics.](image)

### 4.3 Simulation Results

In this section, we demonstrate simulation results corresponding to the integrated system depicted in the block diagram shown in Figure 4.4. The shape of the fish is defined by the equations in section 2.6, where \( R_x = 0.3 \, m, \, R_y = 0.1 \, m, \, K_R = -1, \) for the host
fish $A$, and $R_x = 0.2 \, m$, $R_y = 0.2 \, m$, $K_R = -0.5$ for the two obstacle fish $B$ & $C$, with $\theta_R$ varying from $0^\circ$ to $360^\circ$. Note that the equation of the fish shape corresponds to the fish depicted in Figure 4.3. The initial position of the fish is taken to be at the origin of an inertial frame, and its velocity is $2 \, \text{cm/sec}$, with a velocity heading angle of $45 \, \text{deg}$. Two obstacle fish $B$ and $C$, are assumed. The initial position of $B$ is $(3 \, \text{m}, 6.1 \, \text{m})$, and its velocity is $2 \, \text{cm/sec}$. The obstacle fish $C$ appears at 240 sec, at a location of $(6 \, \text{m}, 5 \, \text{m})$, moving with a velocity of $0.15 \, \text{cm/sec}$. We note that these speed values reflect numbers that are in line with what IPMC actuators are capable of producing. From Figures 4.5 and 4.6, it is evident that $y(0) < 0$ which in conjunction with $V_r(0) < 0$) indicates a situation wherein the initial velocity vector of $A$ lies inside the collision cone to $B$.

As can be seen from Figure 4.6, the initial negative value of $y$ is driven to $w = 0.2$ around $t = 140 \, \text{sec}$, by application of a turning moment that is proportional to the lateral acceleration. The relative velocity kinematics are shown in Figure 4.5, and the time history of the angle $\psi$ used to compute the lateral acceleration is shown in Figure 4.10(c). The evolution of the angle $\delta_A$ at which the acceleration is applied, is shown in Figure 4.6(c). The acceleration is converted into thrust commands of the caudal fin (shown in Figure 4.7(a)) and the left pectoral fin (shown in Figure 4.8(a)).

The thrust commands are then converted into amplitudes of the sinusoidal voltage applied to the IPMC actuator. These voltages are shown in Figure 4.7(b) (for the caudal fin), and Figure 4.8(b) (for the left pectoral fin). The ensuing tip deflections and tip slopes for each of the IPMC actuators are shown in Figure 4.7(c),(d) and 4.8(c),(d). Note that these plots show the amplitude of the underlying sine wave trajectories (shown as an inset in these figures) of the tip deflection and slope. The tip deflections are of the order of $10 \, \text{cm}$, while the tip slopes are of the order of $0.5 \, \text{rad}$, which are reasonable values corresponding to the IPMC actuator. During $t = [140, 240] \, \text{sec}$, the collision avoidance acceleration drops to zero, and accordingly so does the left pectoral thrust. The second fish $C$ appears at $t = 240 \, \text{sec}$, and this fish also lies in the collision cone of $A$ (as evident from Figures 4.5 and 4.6). To
avoid fish \( C \), fish \( A \) again executes a turning motion, this time using a moment generated by the right pectoral fin, while the thrust from the left pectoral fin is zero. The relevant plots of the thrust, voltage amplitude, and amplitudes of the tip deflection and tip slope of the right pectoral fin are shown in Figure 4.9. Figure 4.11 shows the spatial trajectories of the robotic fish \( A \) and the other two fish \( B \) and \( C \).

![Graphs showing kinematic state trajectories](image)

Figure 4.5: Kinematic state trajectories of engagement between the robotic fish \( A \) and obstacle fish \( B \) & \( C \): (a) \( \hat{r} \), (b) \( \hat{\theta} \), (c) \( \hat{V}_\theta \), (d) \( \hat{V}_r \).
Figure 4.6: (a) Output $y$ of robotic fish $A$ to fish $B$ & $C$, (b) Commanded acceleration $a_A$ by collision avoidance law, (c) $\delta_A$.

Figure 4.7: (a) Caudal Thrust, (b) Voltage Amplitude, (c) Tip deflection amplitude, (d) Tip slope amplitude of IPMC Actuator of caudal fin.
Figure 4.8: (a) Left Pectoral Thrust, (b) Voltage Amplitude, (c) Tip deflection amplitude, (d) Tip slope amplitude of IPMC Actuator of left pectoral fin.

Figure 4.9: (a) Right Pectoral Thrust, (b) Voltage Amplitude, (c) Tip deflection amplitude, (d) Tip slope amplitude of IPMC Actuator of right pectoral fin.
Figure 4.10: (a)-(b) Body axis velocity components, $u$ and $v$ of robotic fish, (c) Time history of angle $\psi$ between robotic fish $A$ and $B$ & $C$, (d) Time history of $\dot{\psi}$ between robotic fish $A$ and $B$ & $C$.

Figure 4.11: $XY$ trajectory of robotic fish and obstacle.
CHAPTER 5

COOPERATIVE AND OPTIMAL COLLISION AVOIDANCE LAWS FOR IPMC/SERVO-ACTUATED ROBOTIC FISH

This chapter analyzes an integrated model comprising the nonlinear guidance law for collision avoidance, the nonlinear fish dynamics of the 2D maneuverable robotic fish [97], and the IPMC fin dynamics [33] all integrated in a closed-loop system. In section 5.1, the nonlinear guidance laws used for cooperative collision avoidance are developed. In section 5.2, the fish dynamic model and the IPMC fin dynamics are discussed. In section 5.3, the developed collision avoidance laws are integrated with the IPMC fin dynamics and the fish dynamic model. Section 5.4 presents the simulations of this integrated system.

5.1 Collision Avoidance Laws

5.1.1 Cooperative Collision Avoidance Laws

In this subsection, we use a Lyapunov-based approach to determine analytical expressions for cooperative collision avoidance laws for two arbitrarily shaped objects \( A \) and \( B \) as given in engagement 2.3. The kinematic states are described in 2.13. The Lyapunov function used is as follows:

\[
Z = \frac{1}{2} (y - w)^2
\]

where, \( y \) is the output function defined in (2.12) and \( w \geq 0 \) is a reference input.

We intend to design feedback control laws that render the derivative of the Lyapunov function (7.14) negative definite, which is equivalent to driving \( y(t) \) to \( w \). Toward this end, we first compute the time derivative of \( y \) as \( \dot{y} = \frac{\partial y}{\partial x} \dot{x} \). where, the partial derivatives of \( y \) are as given in 2.16.

The acceleration magnitudes \( a_A \) and \( a_B \) are treated as the primary inputs used to drive \( y(t) \) from an initial negative value to the reference value \( w \geq 0 \). This is equivalent to
determining accelerations $a_A$ and $a_B$ that drive the velocity vectors of $A$ and $B$ from the interior of the collision cone to the boundary of, or outside, the collision cone, and these can be found by determining the acceleration values that will ensure that the derivative of Lyapunov function $\dot{Z} < 0$. The derivative of the Lyapunov function $Z$ is as follows:

$$
\dot{Z} = (y - w)(\dot{y} - \dot{w}) = \frac{(y - w)}{D^2} \times [(V_\theta \cos(\delta_A - \theta) - V_r \sin(\delta_A - \theta))a_A - (V_\theta \cos(\delta_B - \theta) - V_r \sin(\delta_B - \theta))a_B - 2V_r V_\theta \dot{\theta}(V_\theta^2 + V_r^2) + 0.5 \dot{\psi} \sin \psi (V_\theta^2 + V_r^2)]
$$

(5.2)

The appropriate values for $a_A$ and $a_B$ are chosen so that $\dot{Z} = -2KZ$ is satisfied, where $K > 0$. Doing so, we eventually obtain an equation of the form:

$$
a_A N_{a,A} + a_B N_{a,B} + K(y - w)D^2 - Y_a = 0 \quad (5.3)
$$

where, $N_{a,A}, N_{a,B}$ and $Y_a$ are given by the equations (2.18) - (2.20).

Thus, any pair of accelerations $a_A$ and $a_B$ that satisfy (5.3) will ensure $\dot{Z} < 0$ and these will cooperatively drive the quantity $y(t)$ to non-negative reference input $w$. As can be seen, there can be multiple $(a_A, a_B)$ combinations that will satisfy (5.3). A candidate solution is the following:

$$
a_A = \frac{-K(y - w)(V_\theta^2 + V_r^2)^2 + Y_a}{2N_{a,A}} \quad (5.4)
$$

$$
a_B = \frac{-K(y - w)(V_\theta^2 + V_r^2)^2 + Y_a}{2N_{a,B}} \quad (5.5)
$$

Since $\dot{Z} = -2KZ$ is a first order equation, therefore any positive value of $K$ will guarantee that $Z(t)$ decays exponentially to zero, and thus $y(t) \to w$. The gain $K$ needs to be chosen large enough to ensure that $Z$ comes to within an $\epsilon$ bound of zero, within time $t \leq t_m$. The larger the gain $K$, the smaller the achievable $\epsilon$.

We note that the quantities $N_{a,A}$ and $N_{a,B}$ used in the above guidance law (5.4)-(5.5) are functions of $\delta_A$ and $\delta_B$, respectively, which represent the angles at which the accelerations
$a_A$ and $a_B$ are applied (See Figure 2.3). From a purely kinematics standpoint, the following is readily evident. If $\delta_i$ is such that the acceleration is applied in the same direction as the velocity vector of the corresponding object, then $a_i$ is a pure translational acceleration and the object experiences a change in speed with no change in velocity heading angle. If $\delta_i$ is such that the acceleration is applied normal to the velocity vector, then $a_i$ results in a pure lateral acceleration, which causes the velocity vector of the object to rotate (that is, causes a change in heading angle) while the speed of the object remains constant. If $\delta_i$ is applied at any other angle, then the effect of $a_i$ is a combination of speed and heading angle change.

5.1.2 Optimal Cooperative Collision Avoidance Laws

In this section, we determine the optimal acceleration magnitudes $a_A$ and $a_B$, that minimize a given performance index, while also simultaneously satisfying (5.3). The performance index $J(a_A, a_B)$ is such that the sum of the squares of the control efforts of $A$ and $B$, at each time $t$, is minimized, that is, $J$ is defined as follows:

\[ J = \frac{1}{2}(a_A^2(t) + a_B^2(t)) \] (5.6)

The control problem is therefore defined as follows. At each time $t$, find acceleration magnitudes $a_A$ and $a_B$ so as to minimize the performance index (5.6), while satisfying the constraint equation (5.3). The Hamiltonian for this constrained optimal control problem is written as:

\[ H(x, a_A, a_B, \lambda) = \frac{1}{2}(a_A^2 + a_B^2) + \lambda(a_A N_{a,A} + a_B N_{a,B} + K(y - w)(V_\theta^2 + V_r^2)^2 - Y_a) \] (5.7)

where, $x = \begin{bmatrix} r & \theta & V_\theta & V_r & \psi \end{bmatrix}$ and $\lambda$ is a Lagrange multiplier. Then, the necessary conditions for minima of $J(a_A, a_B)$ that also satisfy the constraint (5.3) are found by solving the equations:

\[ \frac{\partial H}{\partial \lambda} = a_A^{*} N_{a,A} + a_B^{*} N_{a,B} + K(y - w)(V_\theta^2 + V_r^2)^2 - Y_a = 0 \] (5.8)
\[
\frac{\partial H}{\partial a_A} = a^*_A + \lambda N_{a,A} = 0 \tag{5.9}
\]
\[
\frac{\partial H}{\partial a_B} = a^*_B + \lambda N_{a,B} = 0 \tag{5.10}
\]

It can be seen that the Hessian matrix of the Hamiltonian $H$, defined as

\[
\begin{bmatrix}
\frac{\partial^2 H}{\partial a_A^2} & \frac{\partial^2 H}{\partial a_A \partial a_B} \\
\frac{\partial^2 H}{\partial a_A \partial a_B} & \frac{\partial^2 H}{\partial a_B^2}
\end{bmatrix}
\]

is strictly positive definite, thereby demonstrating that the Hamiltonian $H$ is convex in the quantities $a_A$ and $a_B$. Therefore, the above necessary conditions also represent sufficient conditions for a minima of $J(a_A, a_B)$, and furthermore the local minimum is also a global minimum. By solving (5.8)-(5.10) simultaneously, we determine the optimal values of $\lambda$, $a_A$, and $a_B$ are as follows:

\[
\begin{align*}
\lambda^* &= \frac{1}{N^2_{a,A} + N^2_{a,B}} (K(y - w)D^2 - Y_a) \tag{5.11} \\
A^*_A &= -\frac{N_{a,A}}{N^2_{a,A} + N^2_{a,B}} (K(y - w)D^2 - Y_a) \tag{5.12} \\
A^*_B &= -\frac{N_{a,B}}{N^2_{a,A} + N^2_{a,B}} (K(y - w)D^2 - Y_a) \tag{5.13}
\end{align*}
\]

Thus, the accelerations defined by (5.12), (5.13) represent the optimal acceleration magnitudes that will ensure minimum control effort as the objects $A$ and $B$ cooperatively drive $y(t)$ to $w \geq 0$.

5.1.3 Optimal Cooperative Collision Avoidance: Heading and Speed Change

In the above subsection, an implicit assumption was that the direction of application of the accelerations $\delta_A$ and $\delta_B$ was not an actively controlled input. In scenarios where the vehicles have the ability to actively control the values of $\delta_A$ and $\delta_B$, we can determine the optimal directions that will achieve collision avoidance, while simultaneously minimizing the performance index (5.6). Towards this end, the Hamiltonian remains the same as in (5.7). The necessary conditions for a minima of (5.6) that also satisfies the constraint (5.3) are as follows. In addition to (5.8), (5.9) and (5.10), the following two equations also need to be
satisfied:
\[
\frac{\partial H}{\partial \delta_A} = \lambda a_A (V_\theta s_1 + V_r c_1) = 0 \quad (5.14)
\]
\[
\frac{\partial H}{\partial \delta_B} = \lambda a_B (V_\theta s_2 + V_r c_2) = 0 \quad (5.15)
\]

Solving (5.8), (5.9), (5.10), (5.14), (5.15) simultaneously for the optimal acceleration magnitudes and directions, we see that the optimal acceleration magnitudes \(a^*_A, a^*_B\) are the same as defined in (5.12) and (5.13), and these need to be applied along directions \(\delta^*_A\) and \(\delta^*_B\) as follows:

\[
\delta^*_A = \hat{\theta} - \tan^{-1} \left( \frac{V_r}{V_\theta} \right) \quad (5.16)
\]
\[
\delta^*_B = \hat{\theta} - \tan^{-1} \left( \frac{V_r}{V_\theta} \right) \quad (5.17)
\]

The given necessary conditions are also sufficient conditions for minimizing (5.6), since the Hessian matrix corresponding to the second order partial derivatives of \(H\), with respect to \(a_A, a_B, \delta_A\) and \(\delta_B\) is positive semi-definite.

### 5.1.4 Effects of Imperfect Information

In this section, the effects of imperfect information on the performance of the collision avoidance laws are analyzed. Such imperfect information can result, for example, from sensor measurement errors. Consider the scenario wherein the applied accelerations \(a_A\) and \(a_B\) are based on imperfect information of the quantities \(V_\theta, V_r, \delta,\) and \(\psi\), and these imperfect measurements are represented by \(\tilde{V}_\theta, \tilde{V}_r, \tilde{\delta},\) and \(\tilde{\psi}\), respectively. When these imperfect quantities are substituted in (5.2), then the equation \(\dot{Z} = -2KZ\) assumes the following form:

\[
\dot{Z} = -2KZ \left( \frac{N_{a,A}}{2N_{a,A}} + \frac{N_{a,B}}{2N_{a,B}} \right) - K \Delta y \sqrt{2V} \left( \frac{N_{a,A}}{2N_{a,A}} + \frac{N_{a,B}}{2N_{a,B}} \right)
\]
\[
- \sqrt{2V} \left( \tilde{Y}_a \left( \frac{N_{a,A}}{2N_{a,A}} + \frac{N_{a,B}}{2N_{a,B}} \right) - Y_a \right) \quad (5.18)
\]
where, \( N_{a,A}, N_{a,B}, Y_A \) are as defined in (2.18,2.19,2.20), and \( \tilde{N}_{a,A}, \tilde{N}_{a,B}, \tilde{Y}_A \) are obtained by replacing \( V_r, V_\theta, \delta \) and \( \psi \) in (2.18,2.19,2.20) with \( \tilde{V}_r, \tilde{V}_\theta, \tilde{\delta} \) and \( \tilde{\psi} \), respectively. The quantity \( \Delta y \) represents an upper bound on the error in \( y \) (as a function of the errors in \( V_r, V_\theta, \theta \) and \( \psi \)) and is given by the expression:

\[
\Delta y = \frac{\partial y}{\partial V_r} \Delta V_r + \frac{\partial y}{\partial V_\theta} \Delta V_\theta + \frac{\partial y}{\partial \psi} \Delta \psi + \frac{\partial y}{\partial \theta} \Delta \theta \quad (5.19)
\]

where, \( \Delta V_r \equiv V_r - \tilde{V}_r, \Delta V_\theta \equiv V_\theta - \tilde{V}_\theta, \Delta \psi \equiv - \tilde{\psi}, \) and \( \Delta \theta \equiv \theta - \tilde{\theta} \). The quantities \( \frac{\partial y}{\partial V_r}, \frac{\partial y}{\partial V_\theta}, \) and \( \frac{\partial y}{\partial \psi} \) are as defined in (2.16). Now, defining two quantities \( \Delta_m \) and \( \Delta_a \) as follows:

\[
\Delta_m = \left( \frac{N_{a,A}}{2N_{a,A}} + \frac{N_{a,B}}{2N_{a,B}} \right) \quad (5.20)
\]

\[
\Delta_a = -K \Delta y \Delta_m - \left( \tilde{Y}_a \Delta_m - Y_a \right) \quad (5.21)
\]

we can rewrite (5.19) as:

\[
\dot{Z} = -2KZ \Delta_m + \sqrt{2}Z \Delta_a \quad (5.22)
\]

The effect of imperfections in sensor measurements are thus manifested in the appearance of the terms \( \Delta_m \) and \( \Delta_a \) in (5.22). Here, both \( \Delta_m \) and \( \Delta_a \) vary with time. Note that in the ideal case when there are no measurement errors, then \( \Delta_m = 1 \) and \( \Delta_a = 0 \), and (5.22) reduces to \( \dot{Z} = -2KZ \). We observe that the term \( \Delta_a \) in the system (5.22) can be treated as a vanishing perturbation since \( Z = 0 \) is an equilibrium point for not just the nominal system \( \dot{Z} = -2KZ \), but also for the perturbed system (5.22).

From the above, we can see the following. If \( \Delta_m > 0 \), and the magnitude of \( K \) is large enough to ensure that the RHS of (5.22) is strictly negative definite, we can be assured that even in the presence of imperfect measurements, \( Z(t) \) will (asymptotically, or otherwise) go to zero. The condition for the RHS of (5.22) to be strictly negative definite is found to be
as follows:

\[ Y_a < (K\sqrt{2V} + K\Delta y + \tilde{Y}_a)\Delta m \]  \hspace{1cm} (5.23)

Additional insight can be obtained as follows. By substituting (5.1) in (5.22), we can rewrite (5.22) in terms of \( y \) to obtain:

\[ \dot{y} = -Ky\Delta m + (Kw\Delta_m + \Delta_a) \]  \hspace{1cm} (5.24)

From (5.24), the following is apparent. For \( \Delta m > 0 \), as long as the gain \( K \) and the reference value \( w \) are (mutually) adjusted such that \( Kw\Delta_m + \Delta_a = \epsilon > 0 \), and \( K \) is large enough to ensure that the exponential decay of \( y \) occurs with a time constant that enables \( y \) to decay to \( \epsilon \) in time \( t < t_m \), collision avoidance can be guaranteed.

### 5.2 2D Maneuverable Robotic Fish

In this section, we discuss the robotic fish and its dynamic state space model that is subsequently used in the collision avoidance control.

#### 5.2.1 2D Maneuvering Robotic Fish

The robotic fish used for testing of the collision avoidance control laws consists of a rigid body and a two-joint hybrid tail. The first joint is actuated by a servo motor, while the second joint is actuated by a soft IPMC actuator [97]. See Fig 5.2. The servomotor connects the main body of the fish to the tail. The primary reason for using the servomotor in the first joint is to generate a sufficiently high forward thrust, which the servomotor achieves by driving a rectangular plate to flap at an optimal frequency. The purpose of using an IPMC in the second joint is to change the slope of the caudal fin, which can then direct the propelled fluid to generate a 2D thrust. Thus, this design enables the robotic fish to perform 2D maneuvers.

A lithium ion battery with a control circuit is zipped into a plastic bag to ensure waterproofing of the electronic components. Two gold-coated copper electrodes are placed on the rear side of the fish to provide actuation voltage signals to the IPMC. The entire
length of the fish is around 27 cm, the diameter is 8 cm, the total inside volume is around 120 cm$^3$, and the weight is about 180 g. A prototype of the fabricated robotic fish is shown in Fig 5.1.

![Prototype of fabricated robotic fish](image)

Figure 5.1: Prototype of fabricated robotic fish

![Schematic of robotic fish in planar motion](image)

Figure 5.2: Schematic of robotic fish in planar motion

5.2.2 State Space Model

A dynamic model, which captured the 2-dimensional movement of the robotic fish, was developed in [97]. This model captured the dynamics of the IPMC actuation, the two
joints, and the fish 2-D body dynamics. The variables in the model are shown in Fig 5.2, where \( u \) is the surge velocity and acts along the body-fixed \( x \)-axis, \( v \) is the sway velocity and acts along the body-fixed \( y \)-axis, and \( \omega_z \) is the yaw angular velocity about the \( z \)-direction. \( F_D \) is the drag force, \( F_L \) is the side force and \( M_D \) is the drag moment. \( \phi(t) \) is the heading angle, \( V_c \) is the velocity magnitude of the body, and \( \beta(t) \) is the sway angle. \( T_1 \) and \( T_2 \) are the hydrodynamic forces acting on the first and the second links, respectively, while \( \alpha_1 \) and \( \alpha_c \) are the angles of the first and the second joints, respectively.

In order to facilitate collision avoidance control design, we derived a control-oriented model in state space form. The overall state space model can be divided into two cascaded sub-models, namely the caudal fin sub-model and the fish body sub-model, with the sway angle \( \beta(t) \), velocity \( V_c(t) \), and angular velocity \( r(t) \) as (implicit) feedback quantities, as shown in Fig 5.3. The quantities \( F_x \) and \( F_y \) represent the net forces along the \( x \) and \( y \) directions, respectively, \( M_z(t) \) is the net turning moment about the \( z \)-axis, \( r(t) \) is the angular velocity of the body, \( X(t) \) and \( Y(t) \) denote the position coordinates of the fish in an inertial frame.

In Fig. 5.3, the caudal fin model outputs thrust forces \( F_x \) and \( F_y \), and turning moment \( M_z \), based on inputs of the first joint angle \( \alpha_1(t) \), the voltage \( V(t) \) applied to the IPMC joint, the swaying angle \( \beta \), and the velocity of the fish body \( V_c(t) \) from the output of the body dynamics. The caudal fin sub-model has three state variables, which are defined as:

\[
x_1 = \alpha_c, \ x_2 = \dot{\alpha}_c, \ x_3 = M_a
\] (5.25)

It has five inputs which are defined as:

\[
u_1 = \alpha_1, \ u_2 = V, \ u_3 = \beta, \ u_4 = V_c, \ u_5 = r
\] (5.26)

and three outputs which are:

\[
y_1 = F_x, \ y_2 = F_y, \ y_3 = M_z
\] (5.27)
Based on the dynamic model derived in [97], a state space model for the hybrid caudal fin can be written as follows:

\[ \dot{x}_1 = x_2 \]  
\[ \dot{x}_2 = \frac{-\xi x_2 - K_l x_1}{I_2 + \rho_w \frac{\pi}{4} \Gamma_2(\omega) \lambda_a} + \frac{x_3 - \rho_w \frac{\pi}{4} \Gamma_2(\omega) \chi \lambda_b}{I_2 + \rho_w \frac{\pi}{4} \Gamma_2(\omega) \lambda_a} \]  
\[ \dot{x}_3 = \frac{K x_2}{\gamma} + \frac{2 \alpha_0 K W k_e (\gamma - 1)}{\gamma} u_2 \]  
\[ F_x = T_1 \sin u_1 + T_2 \sin(u_1 + x_1) - F_D \cos u_3 + F_L \sin u_3 \]  
\[ F_y = T_1 \cos u_1 + T_2 \cos(x_1 + u_1) - F_D \sin u_3 - F_L \cos u_3 \]  
\[ M_z = T_1 a_1 \cos u_1 + T_2 a_1 \cos(x_1 + u_1) + M_D + M_1 \]

The variables used in the above equations are defined as follows:

\[ \chi(t) = L_1(\ddot{u}_1 \cos x_1 - \dot{u}_1 x_2 \sin x_1) \]  
\[ f_2(t) = \frac{-\xi x_2 - K_l x_1}{I_2 + \rho_w \frac{\pi}{4} \Gamma_2(\omega) \lambda_a} + \frac{x_3 - \rho_w \frac{\pi}{4} \Gamma_2(\omega) \chi \lambda_b}{I_2 + \rho_w \frac{\pi}{4} \Gamma_2(\omega) \lambda_a} \]  
\[ T_1 = -\ddot{u}_1 \rho_w \frac{\pi}{4} \Gamma_1(\lambda_f + L_0 \lambda_l) \]  
\[ T_2 = -\rho_0 \frac{\pi}{4} \Gamma_2(f_2 \lambda_b + \lambda_c \chi) \]  
\[ F_L = \frac{1}{2} \rho_w |u_2|^2 |S| u_3 C_L \]  
\[ F_D = \frac{1}{2} \rho_w |u_2|^2 |S| C_L \]  
\[ M_D = -C_M u_5 \text{sgn}(u_3) \]  
\[ M_1 = \rho_w \frac{\pi}{4} (\Gamma_1(\omega) u_1(t)(\lambda_c + 2 L_0 \lambda_f + L_0^2 \lambda_l) + \Gamma_2(\omega)(f_2(t) \lambda_a + \chi(t) \lambda_b) \]
\[ + f_2(t)(L_0 + L_1) \cos(x_1)(\lambda_b + \lambda_c \chi(t))) \]

where, \( C_D \) and \( C_M \) are the drag force coefficient and drag moment coefficient, respectively, and \( \rho_w \) is the density of water. Due to space limitations, the remaining parameters are not defined here, and can be found in [97].
For the body dynamics subsystem, the state variables are defined as:

\[
x_1 = X, \ x_2 = Y, \ x_3 = \phi, \ x_4 = u, \ x_5 = v, \ x_6 = r
\] (5.42)

The body dynamics subsystem’s inputs are:

\[
u_1 = F_x, \ u_2 = F_y, \ u_3 = M_z
\] (5.43)

and it’s outputs are:

\[
y_1 = X, \ y_2 = Y, \ y_3 = \beta, \ y_4 = V_c, \ y_5 = r
\] (5.44)

Figure 5.3: Model structure.

From the dynamic model derived in [97], the state space model of the body dynamics subsystem can be described as follows:

\[
\begin{align*}
\dot{x}_1 &= x_4 \cos x_3 - x_5 \sin x_3 \\
\dot{x}_2 &= x_5 \cos x_3 + x_4 \sin x_3 \\
\dot{x}_3 &= x_6 \\
\dot{x}_4 &= \frac{m_b - Y_v}{m_b - X_v} x_5 x_6 + \frac{u_1}{m_b - \dot{X}_u} \\
\dot{x}_5 &= \frac{m_b - X_u}{m_b - Y_v} x_4 x_6 + \frac{u_2}{m_b - \dot{Y}_v}
\end{align*}
\] (5.45-5.49)
\[
\dot{x}_6 = \frac{Y_\dot{u} - X_\dot{u}}{J_{bz} - N_\dot{r}} x_4 x_5 + \frac{u_3}{J_{bz} - N_\dot{r}}
\]  
(5.50)

where, \(X_\dot{u}, Y_\dot{u}\) and \(N_\dot{r}\) represent constant hydrodynamic derivatives describing the effects of the added mass of the surrounding fluid on the robotic fish, \(m_b\) is the body mass, and \(J_{bz}\) is the moment of inertia about the \(z\)-axis. The body dynamics sub-system’s outputs can then be readily written in terms of these states.

### 5.3 Conversion of Collision Avoidance Commands into Actuator inputs for the Hybrid Tail

The hybrid tail of the above robotic fish has two actuator inputs, namely, the joint angle \(\alpha_1(t)\) of the first joint, and the voltage \(V(t)\) applied to the IPMC. The accelerations generated from the collision avoidance laws presented in Section 5.1 have to be converted into appropriate values of these actuator inputs for the hybrid tail. We note that the collision avoidance laws derived in Section 5.1 generate accelerations that, in general, vary continuously with time. However, since the fish dynamics exhibit periodic motion, it is advantageous (from the viewpoint of the energy usage by the robotic fish) to convert these accelerations into piecewise-constant quantities, wherein the time interval over which each acceleration “piece” is a constant is equal to the flapping period \(T = \frac{2\pi}{\omega}\), where \(\omega\) is the (constant) flapping frequency of the robotic fish. It subsequently becomes possible to convert this piecewise-constant acceleration into a servomotor amplitude that is constant during each flapping cycle, and/or an IPMC voltage that is constant over each flapping cycle. Towards this end, we first define cycle-averaged quantities \(\bar{V}_\theta, \bar{V}_r, \bar{\psi}, \bar{\delta}_i\), as follows:

\[
\bar{V}_\theta = \frac{1}{T} \int_{t_k}^{t_k+T} V_\theta(t) dt
\]  
(5.51)

\[
\bar{V}_r = \frac{1}{T} \int_{t_k}^{t_k+T} V_r(t) dt
\]  
(5.52)

\[
\bar{\psi} = \frac{1}{T} \int_{t_k}^{t_k+T} (t) dt
\]  
(5.53)

\[
\bar{\delta}_i = \frac{1}{T} \int_{t_k}^{t_k+T} \delta_i(t) dt, \ i = A, B
\]  
(5.54)
where, $t_k$ represents the time corresponding to the start of the $k$th flapping cycle. Using these cycle-averaged quantities, a cycle-averaged quantity $\bar{y}$ is subsequently defined by replacing $V_\theta$, $V_r$, and $\psi$ in (2.12) with $\bar{V}_\theta$, $\bar{V}_r$, and $\bar{\psi}$, respectively. The cycle-averaged quantities $\bar{y}$, $\bar{V}_\theta$, $\bar{V}_r$, $\bar{\psi}$, and $\bar{\delta}_i$ then replace $y$, $V_\theta$, $V_r$, $\psi$, and $\delta_i$ in (5.4)-(5.5), and this leads to piecewise-constant values $\bar{a}_A$ and $\bar{a}_B$ of the commanded accelerations.

Collision avoidance of the fish with the obstacle (which could also be another fish) can be achieved in one of three different ways: (a) bending the IPMC, (b) changing the amplitude of the flapping angle, (c) a suitable blend of flapping amplitude change and IPMC bending. These are discussed below:

In (a), collision avoidance is achieved by applying an appropriate voltage $V(t)$ to the IPMC, so as to generate the requisite turning moment. This moment creates a turning trajectory for the robotic fish to move around an obstacle, or for a pair of robotic fish to move around each other. The value of this voltage is computed using (5.30), and is as follows:

$$V_i(t) = \frac{K_V \bar{a}_i}{2\alpha_0 W_k e(\gamma - 1)}, \quad i = A, B$$

where, $K_V$ is an appropriate proportional gain, $\bar{a}_A$ and $\bar{a}_B$ represent the cycle-averaged accelerations generated by the collision avoidance law, and these may be applied at angles $\delta_A = \alpha_A + \frac{\pi}{2}$ and $\delta_B = \alpha_B + \frac{\pi}{2}$.

In (b), collision avoidance is achieved as follows. The flapping angle $\alpha_1(t)$ is made to evolve according to the equation $\alpha_1(t) = A_m \sin \omega t$, where $A_m$ represents the amplitude of the flapping angle of the joint. The generated collision avoidance accelerations $\bar{a}_A$ and $\bar{a}_B$ are converted into flapping amplitudes $A_{m,A}$ and $A_{m,B}$, respectively, while keeping the voltage input to the IPMC $V_A(t) = V_B(t) = 0$. This essentially involves both a speed and heading change maneuver performed by both fish. From (5.31), the net hydrodynamic force $F_x$ including the effects of the collision avoidance acceleration $\bar{a}_A$ acting at angle $\delta_A$ is obtained.
(for fish \(A\)) as:

\[
F_x = T_1 \sin \alpha_1 + T_2 \sin(\alpha_1 + \alpha_c) - F_D \cos \beta + F_L \sin \beta - \bar{a}_A \cos \beta \tag{5.56}
\]

Eqn (5.56) is cycle-averaged over a time period \(T\). We note that \(T_1\) and \(T_2\) in (5.36) and (5.37) are functions of \(A_m\), since \(\ddot{u}_1 = -A_m \omega^2 \sin \omega t\) appears in the expression for \(T_1\), while \(\ddot{u}_1 = -A_m \omega \cos \omega t\) appear in the expression for \(\chi(t)\) in (5.34), which in turn influences \(T_2\) in (5.37). Substituting for \(T_1, T_2, F_L\) and \(F_D\) from (5.36)-(5.39) in (5.56) and averaging over a time period \(T\), the resulting equation is as follows:

\[
\int_{t_k}^{t_k+T} A_m \cos(\omega t) \sin(A_m \cos(\omega t)) dt = \int_{t_k}^{t_k+T} \frac{1}{K_c} (F_D \cos \beta + F_L \sin \beta - \bar{a}_A \cos \beta) dt
\]

where, \(K_c = \rho_w \frac{\pi}{4} \omega^2 (\Gamma_1 \lambda_f + \Gamma_1 L_0 \lambda_l + \Gamma_2 \lambda_c L_1)\).

We therefore need to solve (5.58) for the unknown quantity \(A_m\). We note that in (5.58), \(\bar{a}_A\) and \(A_m\) are piecewise-constant quantities (constant over each flapping cycle), while \(\omega\) is a constant quantity (constant over all time). It is evident from (5.58) that it is not straightforward to determine an explicit equation for \(A_m\). We therefore numerically solve (5.58), and use this solution as an input to the servo motor. Note that (5.58) is derived assuming ideal IPMC actuator dynamics for simplicity. Equations similar to (5.56) and (5.58) can be written to determine the flapping angle amplitude for fish \(B\).

In (c), collision avoidance is achieved by generating an appropriate combination of IPMC voltage \(V(t)\) and flapping amplitude \(A_m\). This essentially involves the robotic fish \(A\) and \(B\) generating a combination of turning moment, speed change and sideways motion in order to bypass each other, and may be achieved by a suitable combination of (5.55) and (5.58). A block diagram of the overall integrated system is presented in Fig 5.4. The guidance law accelerations \(\bar{a}_i\) generated from (5.3) are converted into appropriate voltages using equa-
tion (5.55), and/or flapping angle amplitudes using (5.58). These voltages when supplied to IPMC actuators generate applied moments which then influence the turning dynamics of $A$ and $B$, while the amplitudes commanded to the servomotors influence the translational dynamics of $A$ and $B$. The undulatory locomotion of $A$ and $B$ influence the relative velocity kinematic states (2.13). These in turn influence the quantity $\tilde{y}$ used in the collision cone computations. The value of $\tilde{y}$ along with the relative kinematic states is then fed back towards computation of the guidance law accelerations $\tilde{a}_i$ for the next flapping cycle.

5.4 Simulation Results

In this section, we present simulation results demonstrating the working of collision avoidance laws integrated with the fish dynamics, as shown in Fig 5.4. The shape of the robotic fish (as viewed from the top) is shown in Fig 5.5 for three different flapping angles - these being the zero flapping angle, and the two extreme flapping angles (on opposite sides of the fish body). A bounding surface $S$ is generated for the fish such that this surface always envelopes the fish, regardless of the flapping angle. This bounding surface is also shown in Fig 5.5 and is non-convex. This bounding surface provides an advantage in dense swarms, since it increases the amount of free space that the fishes have, within which they can maneuver. The equations of the bounding are as given in section (2.6). The values of the parameters in section (2.6) that correspond to the robotic fish shown in Fig 5.1 are $R_{r1,x} = 0.15, R_{r1,y} = -0.3, K_{r1} = 0.75, R_{r2,x} = 0.15, R_{r2,y} = -0.3, K_{r2} = 0.75$. These are the values chosen for the subsequent simulations.

We now present three simulations demonstrating the working of cooperative collision avoidance laws presented in Section 5.1 when integrated with the fish dynamic model presented in Section 5.2.

Example 1: In this example, cooperative collision avoidance is demonstrated while using the IPMC voltages $V_A(t)$ and $V_B(t)$ as the primary inputs for fish $A$ and $B$, respectively. The initial conditions are chosen such that $A$ and $B$ are on a collision course with one other and would collide in the absence of a collision avoidance law. The initial position of fish
$A$ is (0.25 m, 0.36 m) in an inertial frame, and moving with a velocity of 0.11 m/s at a heading angle of $-31.44^0$. Fish $B$ has an initial position (1.54 m, 1.23 m), and its velocity is 0.11 m/s with an initial heading angle of $-211.4^0$. These speed values are in line with the speed capable of being produced by the robotic fish in Fig 5.2. These initial positions and heading angles are shown in Fig 5.6.

Figure 5.4: Integrated System Architecture

Figure 5.5: Shape of bounding surface that encloses the robotic fish over its full range of flapping angles

Fig 5.8 shows the plots of $\dot{y}$, as well as $V_\theta(V_\theta)$ and $V_r(V_r)$, as functions of time. Due to the undulatory motion of the fish, it is seen that the relative velocity quantities $V_\theta$ and $V_r$,
both exhibit elements of oscillatory behavior. The cycle-averaged values $\bar{V}_\theta$ and $\bar{V}_r$ exhibit relatively smoother behavior and are more representative of the mean trajectories of the fish. As seen from Fig 5.8, for the chosen initial conditions, $\bar{y} < 0$ and $\bar{V}_r < 0$, thereby indicating that the relative velocity vector lies inside the collision cone, and $A$ and $B$ are on a collision course with each other.

Figure 5.6: Example 1: $XY$ Trajectories of robotic fish $A$ and $B$

The cooperative collision avoidance acceleration magnitudes computed from (5.4) and (5.5) are shown in Fig 5.7. These accelerations are applied at angles $\delta_A = \alpha_A + \frac{\pi}{2}$ and $\delta_B = \alpha_B + \frac{\pi}{2}$, to produce pure heading changes in both $A$ and $B$. These accelerations are converted into suitable voltages $V_A(t)$ and $V_B(t)$ using (5.55), and are shown in Fig 5.7. These voltage values adhere to the sustainable voltage range of IPMC used by robotic fish. Due to the ensuing turning moments, the negative value of $\bar{y}$ is driven to $w \geq 0$ as is evident from Fig 5.8. The applied moment for fish $A$ is shown in Fig 5.9 along with body axis velocity components $u_A$, $v_A$ and angular velocity $r_A$. The corresponding quantities for fish $B$ are shown in Fig 5.10. The heading angle changes produced as a result of applied moments
on $A$ and $B$ are shown in Fig 5.11. It is seen that as a consequence of the applied moments, the mean values of $\phi_A$ and $\phi_B$ have increased, and these correspond to $A$ and $B$ both turning to their left.

![Figure 5.7: Example 1: Time histories of accelerations $a_A$ and $a_B$ applied by fishes $A$ and $B$ respectively](image)

Fig 5.6 shows the spatial trajectories of $A$ and $B$ as they cooperatively avoid a collision. The undulatory motion of the fish is clearly evident from this figure. From Fig 5.7 it was seen that the computed collision avoidance accelerations generate negative voltages, and this is also evident in Fig 5.6, since both $A$ and $B$ turn their mean heading angles to their respective left directions. At $t = 6$ sec, $\bar{y}$ is driven to $w \geq 0$, and this implies that the velocity vectors of both fish are driven from inside the collision cone to outside the collision cone. Beyond $t = 6$sec, the accelerations $\bar{a}_A$ and $\bar{a}_B$ both drop down to zero, and so do the IPMC voltages.
Figure 5.8: Example 1: Kinematic state trajectories

Figure 5.9: Example 1: (a)-(d) Body axis velocity components $u_A$, $v_A$ and $r_A$ and applied moment $M_{aA}$ of robotic fish $A$
Figure 5.10: Example 1: (a)-(d) Body axis velocity components $u_B$, $v_B$ and $r_B$ and applied moment $M_{aB}$ of robotic fish $B$

Figure 5.11: Example 1: (a)-(b) Heading angle trajectories of robotic fish $A$ and $B$

83
Example 2: In this example, cooperative collision avoidance is demonstrated while considering the flapping amplitudes $A_{mA}$ and $A_{mB}$ as the primary inputs. Fish $A$ is initially located at $(0.26 \text{ m}, 0.13 \text{ m})$ and moves with initial velocity of $0.05 \text{ m/s}$ at a heading angle of $1^0$, while fish $B$ is at $(3.95 \text{ m}, 0.35 \text{ m})$ and moves with initial velocity of $0.05 \text{ m/s}$ at a heading angle of $120^0$. As seen in Fig 5.12, the initial conditions correspond to $\bar{y}(0) < 0$ and $\bar{V}_r(0) < 0$, which indicate that $A$ and $B$ are on a collision course.

![Kinematic state trajectories](image)

Figure 5.12: Example 2: Kinematic state trajectories

The cooperative collision avoidance accelerations are shown in Fig 5.13. These accelerations are applied at angles $\delta_A = \alpha_A$ and $\delta_B = \alpha_B$, and produce speed changes by converting these accelerations into appropriate amplitudes $A_{mA}$ and $A_{mB}$, where these amplitudes are computed from equation (5.58). As seen in Fig 5.13, the accelerations are opposite in sign and (since these accelerations are acting along the heading direction of the corresponding fish), indicate that $A$ is decelerating while $B$ is accelerating. A corresponding trend is seen for the flapping amplitudes (that is, decrease in flapping amplitude of $A$ and an increase in flapping amplitude of $B$).
As evident from Fig 5.12, the effects of these amplitude changes are to drive \( \bar{y} \) from its initial negative value to a reference \( w \geq 0 \) and thereby avoid collision with each other. The body axis velocity components \( u_A, v_A \), angular velocity \( r_A \) and heading angle \( \phi_A \) for robotic fish \( A \) are shown in Figs 5.14 and 5.15, along with zoomed in versions that show the corresponding values from 0 to 4 sec which demonstrate the oscillations characteristic of the undulatory fish dynamics. The corresponding figures for fish \( B \) are Figs 5.16 and 5.17.

Figure 5.13: Example 2: Time histories of accelerations \( \bar{a}_A \) and \( \bar{a}_B \) and corresponding amplitudes \( A_{mA} \) and \( A_{mB} \)

Fig 5.18 shows the spatial trajectories of \( A \) and \( B \). \( A \) and \( B \) attain their closest approach around \( t = 38 \) sec, and they avoid each other. After they have crossed their point of closest approach, the collision avoidance accelerations \( \bar{a}_A \) and \( \bar{a}_B \) drop to zero.
Figure 5.14: Example 2: (a)-(d) Body axis velocity components $u_A$ and $v_A$ of robotic fish $A$.

Figure 5.15: Example 2: (a)-(d) Angular velocity $r_A$ and heading angle $\psi_A$ of robotic fish $A$. 

"86"
Figure 5.16: Example 2: (a)-(d) Body axis velocity components $u_B$ and $v_B$ of robotic fish $B$

Figure 5.17: Example 2: (a)-(d) Angular velocity $r_B$ and heading angle $\phi_B$ of robotic fish $B$
Example 3: In this example, a non-cooperative scenario is considered wherein the robotic fish $A$ is on a collision course with several pop-up obstacle fish $B$, $C$ and $D$. The obstacle fish $B$ and $C$ are taken to be bigger in size than $A$, while the stationary obstacle fish $D$ is bigger in size and also has a different shape compared to $A$, $B$ and $C$. The robotic fish $A$ needs to move to a sequence of time-varying goal points, while avoiding collisions with $B$, $C$ and $D$.

Fig 5.20 shows the spatial trajectory taken by $A$ as it navigates to multiple goal-points. $A$ is initially located at $(0.35 \text{ m}, 0.47 \text{ m})$ and moves with an initial velocity of $0.12 \text{ m/s}$ and an initial heading angle of $-32^0$. $B$ is on a collision course with $A$. The collision avoidance laws generate an acceleration $\bar{a}_A$, which is converted to a corresponding IPMC voltage $V_A(t)$, which upon application, enables $A$ to avoid $B$. After passing $B$, fish $A$ heads toward a goal point defined by $(2 \text{ m}, 8 \text{ m})$. While $A$ is moving towards this goal point, obstacle fish $C$ pops-up at $t = 72 \text{ sec}$ and is on a collision course with $A$. $A$ generates the appropriate values of $\bar{a}_A$ and $V_A$, using which it can avoid colliding with $C$. After passing $C$, $A$ now
heads towards a new goal point (3 m, 12 m). Now, a stationary obstacle fish $D$ pops-up at $t = 152 \text{ sec}$. $A$ avoids $D$ with the closest approach occurring at $t = 180 \text{ sec}$. After passing $D$, fish $A$ heads toward goal point (3 m, 14 m). Figure 5.20 shows the curvy path taken by $A$ in avoiding collisions with $B$, $C$ and $D$ in reaching its eventual goal point. Figure 5.19 shows the time histories of acceleration and voltage (applied to the IPMC) as generated by the collision avoidance laws.

Figure 5.19: Example 3: (a) Applied acceleration $\ddot{a}_A$ by collision avoidance law, (b) Voltage applied to IPMC of fish $A$
Figure 5.20: Example 3: XY Trajectory of fish A as it performs non-cooperative collision avoidance maneuvers with obstacle fish B, C and D
In this chapter, collision avoidance laws using the collision cone approach presented in chapter 2 are used to analyze pursuit evasion games. In Section 6.1 we illustrate how a pursuit evasion game can be connected to the collision cone concepts. In Sections 6.2 and 6.3 we formulate two pursuit evasion games based on the collision cone, and also present numerical results for these two games.

### 6.1 Problem Formulation

We incorporate the collision cone approach in the context of a two player pursuit evasion game. The pursuer seeks to cause a collision by applying an acceleration so as to drive the relative velocity vector deep into the collision cone, while the evader seeks to avoid collision by applying an acceleration that will drive the relative velocity vector out of the collision cone. The game is played on a plane, and it is assumed that both the players have bounded acceleration, and these acceleration bounds are identical for both the players.

The state and output equations for the 2-player game are as follows:

\[
\begin{align*}
\dot{x} &= f(x) + g_A(x)a_A + g_B(x)a_B \\
y &= h(x)
\end{align*}
\]  

(6.1)  

(6.2)

where, the state vector \( x \) is defined as  

\[
x = \begin{bmatrix} \dot{r} & \dot{\theta} & \dot{V}_r & V_A & V_B & \alpha_A & \alpha_B \end{bmatrix}^T.
\]

The state equations for the first four states are as given in (2.13), while those for the remaining four states are as follows.

\[
\begin{align*}
\dot{V}_A &= a_A \cos (\delta - \alpha_A) \\
\dot{V}_B &= a_B \cos (\delta - \alpha_B) \\
\dot{\alpha}_A &= \frac{a_A}{V_A} \sin (\delta - \alpha_A) \\
\dot{\alpha}_B &= \frac{a_B}{V_B} \sin (\delta - \alpha_B)
\end{align*}
\]  

(6.3)
The output equation for $y$ is as given in (2.12).

Here, $a_A$ and $a_B$ are the accelerations of the two players, with $A$ being the pursuer and $B$ being the evader. These accelerations are applied at angles $\delta_A$ and $\delta_B$, respectively, with the angles being measured relative to an inertial horizontal reference. In the context of the collision cone, $B$ applies an acceleration so as to bring the relative velocity vector out of the collision cone, while $A$ applies an acceleration so as to bring the relative velocity vector inside the collision cone. From a mathematical perspective, $B$ attempts to get the quantity $y(t)$ to a positive value (which corresponds to the physical scenario of the relative velocity vector being outside the collision cone), while $A$ attempts to get the quantity $y(t)$ to a negative value. Let $t_m$ represent the time at which $V_r$ becomes zero, that is, $V_r(t_m) = 0$. Two games are considered.

At one level, the two games are qualitative (that is, games of kind) in that the game ends if the pursuer captures the evader within a specified time duration, or the evader avoids the pursuer for that time duration. Since the quantity $y(t)$ defines the instantaneous depth of the relative velocity vector inside (or outside) the collision cone, the games are also quantitative (that is, games of degree) in the following sense. (a) In the scenario that the pursuer captures the evader (that is, $y(t_m) < 0, V_r(t_m) < 0$), the larger the value of $|y(t_m)|$, the deeper the impact of the pursuer into the evader, while the smaller the value of $|y(t_m)|$, the shallower the impact. (b) In the scenario that the evader manages to avoid capture, then the smaller the value of $|y(t_m)|$, the smaller the miss-distance and thus the narrower the escape by the evader, while the larger the value of $|y(t_m)|$, the larger the miss-distance, and the more comfortable the evader’s escape. Game 1 is open-loop in the sense that neither the pursuer nor the evader use any feedback of the other agent’s states, while Game 2 is closed-loop in the sense that both the pursuer and the evader use continuous feedback of the value of $y(t)$, and incorporate this value in their respective guidance laws.
6.2 Pursuit Evasion Game 1

In the first game (Game 1), both the players $A$ and $B$ apply their individual accelerations $a_A$ and $a_B$ normal to the respective velocity vectors. Thus, the maneuver is one of pure heading change by both the players, while maintaining constant speed. This makes the trajectories of both the players circular, with the radius of each player’s trajectory depending on the magnitude of the applied acceleration. These acceleration magnitudes $a_A$ and $a_B$ are discretized to lie between $-2m/sec^2$ to $3.5m/sec^2$ for $A$, and between $-3m/sec^2$ and $3m/sec^2$ for $B$. A

In all the cases shown below, the initial conditions are identical. The initial coordinates of the center of the pursuer are $(0, 0)$, while that of the center of the pursuer are $(15, 0)$. The initial speed of the pursuer is $7m/sec$, while that of the evader is $6m/sec$. The initial heading of the pursuer is $45^0$, while that of the evader is $120^0$. For these initial conditions, the initial value of $y$, that is, $y(0)$ is negative, and $V_r(0)$ is also negative, thus indicating that the initial speeds and headings of the pursuer and evader are such that they are on a collision course. The shapes of the pursuer and the evader are represented by equations 2.38 presented in section 2.4, and the shape is shown in Figure 2.6.

In the results of Figures 6.1-6.3, the pursuer applies a constant acceleration of $1m/sec^2$, while the evader applies a constant acceleration of $-3m/sec^2$. Figure 6.3 shows the trajectories of the two objects, and as can be seen from the figure they come very close to each other, but do not collide. The instant at which the closest approach occurs, is when $V_r = 0$, and this is seen from Figure 6.1 to be at around $t = 2.1sec$. Figure 6.2 shows that $y$, which was initially negative, becomes positive at the instant of closest approach, which makes sense because the two objects did not collide in this case. Since the vehicles apply their accelerations normal to their respective velocity vectors, their speeds remain constant throughout the engagement. The plots of their velocity heading angles are shown in Figure 6.2.

In Figures 6.4,6.5,6.6, the pursuer applies a constant acceleration of $-1m/sec^2$, while the evader’s acceleration is $-3m/sec^2$. As can be seen from Figure 6.5, the value of $y$
decreases from its initial negative value, and the two agents eventually collide (as seen in Figure 6.6). In other words, this means that the net effect of their accelerations is to cause the relative velocity vector to go deeper inside the collision cone. The influence of the applied accelerations by the pursuer and evader on their velocity heading angles are seen in Figure 6.5.

In Figure 6.7, the evader applies a constant acceleration of $3 m/sec^2$, while the pursuer applies a constant acceleration of $-2 m/sec^2$. Accordingly, the evader traces a counterclockwise trajectory, while the pursuer’s trajectory is clockwise. At the instant of closest approach shown in the figure, the pursuer and the evader are well separated, and there is no collision. The effect of the applied accelerations thus has been to drive the relative velocity vector out of the collision cone (or, in other words, drive $y$ from its initial negative value to a positive value at time $t = t_m$).

![Figure 6.1: State Time Histories for Game 1, with accelerations $a_A = 1 m/sec^2, a_B = -3 m/sec^2$](image_url)
Figures 6.1-6.7 are thus the simulation results for a few specific cases of acceleration combinations. Defining the value of $y(t_m)$ (which is $y$ at the instant when $V_r = 0$ occurs) to be the payoff, the results of the matrix game for multiple $(a_A, a_B)$ combinations are shown in Table 6.1. All the entries correspond to the same initial condition discussed earlier.

![Time Histories](image)

Figure 6.2: Time Histories of $\phi$, $y$, acceleration inputs, and velocity heading angles for Game 1, with $a_A = 1\ m/sec^2$, $a_B = -3m/sec^2$

![Trajectories](image)

Figure 6.3: XY Trajectories of the Pursuer and Evader for Game 1, with $a_A = 1m/sec^2$, $a_B = -3m/sec^2$
Figure 6.4: State Time Histories for Game 1, with accelerations $a_A = -1\,m/sec^2, a_B = -3\,m/sec^2$

Figure 6.5: Time Histories of $\phi$, $y$, acceleration inputs, and velocity heading angles for Game 1, with $a_A = -1\,m/sec^2, a_B = -3\,m/sec^2$
Figure 6.6: XY Trajectories of the Pursuer and Evader for Game 1, with $a_A = -1\text{m/sec}^2, a_B = -3\text{m/sec}^2$

Figure 6.7: XY Trajectories of the Pursuer and Evader for Game 1, with $a_A = 3\text{m/sec}^2, a_B = -2\text{m/sec}^2$
If \( y(t_m) = 0 \) occurs, then this means that at the instant of closest approach, the relative velocity vector is aligned with the boundary of the collision cone, and the two objects just graze each other. While the entries in Table 6.1 correspond to those of a game of degree, we can extract a game of kind matrix from Table 1. We do so by replacing negative values of \( y(t_m) \) (which indicate collision) by \(-1\), positive values of \( y(t_m) \) (which indicate avoidance) by \(+1\), and values of \( y(t_m) \) that belong to a small interval of \([-0.01, 0.01]\) by 0. By doing so, Table 1, assumes the form shown in Table 6.2.

Table 6.1: Results for Matrix Game 1: Game of Degree

<table>
<thead>
<tr>
<th>( a_A \rightarrow a_B \downarrow )</th>
<th>-2</th>
<th>-1</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>3.5</th>
</tr>
</thead>
<tbody>
<tr>
<td>-3</td>
<td>-0.14</td>
<td>-0.88</td>
<td>0.45</td>
<td>0.01</td>
<td>0.15</td>
<td>0.03</td>
<td>0.51</td>
</tr>
<tr>
<td>-2</td>
<td>-0.75</td>
<td>-0.79</td>
<td>-0.79</td>
<td>-0.73</td>
<td>0.04</td>
<td>0.57</td>
<td>0.13</td>
</tr>
<tr>
<td>-1</td>
<td>-0.74</td>
<td>-0.82</td>
<td>-0.52</td>
<td>-0.56</td>
<td>-0.76</td>
<td>-0.15</td>
<td>0.08</td>
</tr>
<tr>
<td>0</td>
<td>-0.75</td>
<td>-0.68</td>
<td>-0.50</td>
<td>-0.81</td>
<td>-0.39</td>
<td>-0.73</td>
<td>-0.73</td>
</tr>
<tr>
<td>1</td>
<td>-0.54</td>
<td>-0.49</td>
<td>-0.48</td>
<td>-0.55</td>
<td>-0.63</td>
<td>-0.64</td>
<td>-0.68</td>
</tr>
<tr>
<td>2</td>
<td>-0.57</td>
<td>-0.48</td>
<td>-0.64</td>
<td>-0.65</td>
<td>-0.43</td>
<td>-0.58</td>
<td>-0.61</td>
</tr>
<tr>
<td>3</td>
<td>-0.46</td>
<td>-0.45</td>
<td>-0.65</td>
<td>-0.59</td>
<td>-0.44</td>
<td>-0.51</td>
<td>-0.53</td>
</tr>
</tbody>
</table>

Table 6.2: Results for Matrix Game 1: Game of Kind

<table>
<thead>
<tr>
<th>( a_A \rightarrow a_B \downarrow )</th>
<th>-2</th>
<th>-1</th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>3.5</th>
</tr>
</thead>
<tbody>
<tr>
<td>-3</td>
<td>-1</td>
<td>-1</td>
<td>+1</td>
<td>0</td>
<td>+1</td>
<td>+1</td>
<td>+1</td>
</tr>
<tr>
<td>-2</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
<td>1</td>
<td>+1</td>
<td>+1</td>
<td>+1</td>
</tr>
<tr>
<td>-1</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
<td>+1</td>
</tr>
<tr>
<td>0</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
</tr>
<tr>
<td>1</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
</tr>
<tr>
<td>2</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
</tr>
<tr>
<td>3</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
</tr>
</tbody>
</table>

If we now focus our attention exclusively on the set of columns that range from \( a_A = 0 \) to \( a_A = 3.5 m/sec^2 \), we see that the zero entry (corresponding to \( a_A = 1 m/sec^2 \), \( a_B = -3 m/sec^2 \)) corresponds to a local saddle point for this game.
6.3 Pursuit Evasion Game 2

In the second game (Game 2), again the pursuer $A$ tries to make $y < 0$, while the evader $B$ tries to make $y > 0$. In this game, both $A$ and $B$ use guidance laws for the magnitudes of $a_A$ and $a_B$ (based on the current values of $y$), to achieve the above objective. These guidance laws are obtained using a nonlinear dynamic inversion technique presented in chapter 2. For player $A$, the acceleration law is as given in equation 2.21, which seeks to make $y = w$, where $w < 0$ represents the negative reference value chosen by the pursuer $A$. A similar acceleration law exists for the evader $B$ with the difference that the reference value for the evader is positive (that is, $w > 0$). When $a_B$ is zero (that is, $B$ moves with constant velocity), the acceleration law (2.21) guarantees that $A$ will achieve $y = w < 0$, and the dynamics of $y$ will evolve according to the equation $\dot{y} + y = K_A w$, where $K_A$ represents the acceleration gain of $A$. Similarly, when $a_A$ is zero (that is, $A$ moves with constant velocity), the guidance law guarantees that $B$ will achieve $y = w > 0$, and the dynamics of $y$ will evolve according to $\dot{y} + y = K_B w$, where $K_B$ represents the acceleration gain of $B$.

An examination of (2.21) shows that the magnitude of the acceleration is also a function of the angle $\delta_A$, which represents the angle at which $a_A$ is applied. When $\delta_A = \alpha + \frac{\pi}{2}$, the above acceleration is applied normal to the velocity vector of $A$ (that is, the influence of the applied acceleration is to cause a pure heading change in the velocity vector of $A$), while when $\delta_A = \alpha_A$, the above acceleration is applied along the velocity vector of $A$ (that is, the influence of the applied acceleration is to cause a pure speed change in $A$). For other values of $\delta_A$, the effect of the applied acceleration is to cause a combination of speed and heading change in $A$. A corresponding set of statements can be made for the angle $\delta_B$.

This game is played between the angles $\delta_A$ and $\delta_B$ at which $a_A$ and $a_B$ are applied. These angles belong to the range $[0, \frac{\pi}{2}]$, relative to the line normal to the velocity vector of each vehicle. In other words, $\delta_A = \alpha_A + \frac{\pi}{2} + \eta_A$, and $\delta_B = \alpha_B + \frac{\pi}{2} + \eta_B$, where $\eta_A \in [0, \frac{\pi}{2}]$, and $\eta_B \in [0, \frac{\pi}{2}]$. The initial conditions are taken as identical to those of Game 1.
Figures 6.8-6.10 represent the scenario when $\eta_A = 0$, $\eta_B = \frac{5\pi}{12}$, which means that $\delta_A = \alpha_A + \frac{\pi}{2}$, and $\delta_B = \alpha_B + \frac{5\pi}{12} + \frac{\pi}{2}$. The time of closest approach occurs when $V_r = 0$ and from Figure 6.8 it is seen that this occurs at around 1.4 seconds. At this time, $y(0)$ is very nearly zero, indicating that the evader has just barely grazed the pursuer. The accelerations of both the pursuer and the evader hit their saturation limits ($\pm 10m/sec^2$) for a short while at the initial portion of the trajectory, but at the time when the two objects graze each other, the acceleration magnitudes are as obtained from the guidance law.

Figure 6.8: State Time Histories for Game 2, with $\eta_A = 0, \eta_B = \frac{5\pi}{12}$
Figure 6.9: Time Histories of $\psi$, $y$, acceleration inputs, and velocity heading angles for Game 2, with $\eta_A = 0, \eta_B = \frac{5\pi}{12}$

Figure 6.10: XY Trajectories of the Pursuer and Evader for Game 2, with $\eta_A = 0, \eta_B = \frac{5\pi}{12}$

In Figures 6.11-6.13 we have $\eta_A = \frac{5\pi}{18}$ and $\eta_B = \frac{\pi}{2}$. In this case, the evader managed to increase $y$ from its initial value (in other words, it could increase the miss-distance from what it originally would have been), and could successfully avoid the collision. At the instant when $V_r = 0$ (around 6 seconds), $y$ is around 0.3, which is a positive number, and thus represents a case of no collision.
Figure 6.11: State Time Histories for Game 2, with $\eta_A = \frac{5\pi}{18}, \eta_B = 0$

Figure 6.12: Time Histories of $\phi$, $y$, acceleration inputs, and velocity heading angles for Game 2, with $\eta_A = \frac{5\pi}{18}, \eta_B = 0$
Figure 6.13: XY Trajectories of the Pursuer and Evader for Game 2, with $\eta_A = \frac{5\pi}{18}, \eta_B = 0$

In Figures 6.14-6.16 we have $\eta_A = \frac{\pi}{2}$ and $\eta_B = 0$. In this case, the evader managed to increase $y$ from its initial value (in other words, it could reduce the miss-distance from its original value) but could not quite avoid the collision. At the instant when $V_r = 0$ (around 3.3 seconds), $y$ is around $-0.1$, which represents a collision, and this collision is evident from Figure 6.16.

Figure 6.14: State Time Histories for Game 2, with $\eta_A = \frac{\pi}{2}, \eta_B = 0$
Figure 6.15: Time Histories of $\phi$, $y$, acceleration inputs, and velocity heading angles for Game 2, with $\eta_A = \frac{\pi}{2}$, $\eta_B = 0$

![Graphs showing time histories of various parameters](image)

Figure 6.16: $XY$ Trajectories of the Pursuer and Evader for Game 2, with $\eta_A = \frac{\pi}{2}$, $\eta_B = 0$

![XY trajectories graph](image)

Table 6.3: Results for Matrix Game 2: Game of Degree

<table>
<thead>
<tr>
<th>$\eta_A \rightarrow$</th>
<th>$\frac{\pi}{2}$</th>
<th>$\frac{5\pi}{12}$</th>
<th>$\frac{5\pi}{18}$</th>
<th>$\frac{5\pi}{36}$</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\eta_B$ ↓</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$\frac{\pi}{2}$</td>
<td>-0.05</td>
<td>-0.06</td>
<td>-0.04</td>
<td>0.01</td>
<td>0.03</td>
</tr>
<tr>
<td>$\frac{5\pi}{12}$</td>
<td>-0.05</td>
<td>-0.06</td>
<td>-0.03</td>
<td>0.02</td>
<td>0.006</td>
</tr>
<tr>
<td>$\frac{5\pi}{18}$</td>
<td>-0.05</td>
<td>-0.05</td>
<td>-0.03</td>
<td>0.14</td>
<td>-0.03</td>
</tr>
<tr>
<td>$\frac{5\pi}{36}$</td>
<td>-0.05</td>
<td>-0.06</td>
<td>-0.02</td>
<td>-0.03</td>
<td>-0.06</td>
</tr>
<tr>
<td>0</td>
<td>-0.10</td>
<td>-0.06</td>
<td>0.3</td>
<td>-0.07</td>
<td>-0.09</td>
</tr>
</tbody>
</table>
Defining $y(t_m)$ to represent the payoff, the results of the matrix game for different combinations of $(\phi_A, \phi_B)$ chosen by the players are given in Table 6.3. Table 6.3 thus represents a game of degree. Similar to what was done in Game 1, we can extract a game of kind matrix from Table 6.3 and do so by replacing negative values of $y(t_m)$ (which indicate collision) by $-1$, positive values of $y(t_m)$ (which indicate avoidance) by $+1$, and values of $y(t_m)$ that belong to a small interval of $[-0.006, 0.006]$ by $0$. By doing so, Table 6.3 assumes the form shown in Table 6.4.

Table 6.4: Results for Matrix Game 2: Game of Kind

<table>
<thead>
<tr>
<th>$\eta_A \rightarrow \eta_B$</th>
<th>$\frac{\pi}{2}$</th>
<th>$\frac{5\pi}{12}$</th>
<th>$\frac{5\pi}{18}$</th>
<th>$\frac{5\pi}{36}$</th>
<th>$0$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\frac{\pi}{2}$</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
<td>+1</td>
<td>+1</td>
</tr>
<tr>
<td>$\frac{5\pi}{12}$</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
<td>+1</td>
<td>0</td>
</tr>
<tr>
<td>$\frac{5\pi}{18}$</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
<td>+1</td>
<td>-1</td>
</tr>
<tr>
<td>$\frac{5\pi}{36}$</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
</tr>
<tr>
<td>0</td>
<td>-1</td>
<td>-1</td>
<td>+1</td>
<td>-1</td>
<td>-1</td>
</tr>
</tbody>
</table>

If we now focus our attention exclusively on the last two columns and the bottom four rows of Table 6.4 (these correspond to $\eta_A = \frac{5\pi}{36}$ to $\eta_A = 0$, and $\eta_B = \frac{5\pi}{12}$ to $\eta_B = \frac{5\pi}{36}$), we see that the zero entry (corresponding to $\eta_A = 0$, $\eta_B = \frac{5\pi}{12}$) corresponds to a local saddle point for this game.
CHAPTER 7
COLLISION CONES FOR DEFORMING OBJECTS

In this chapter, we present analytical guidance laws for objects changing their shapes as a function of time. Sections 7.1, 7.2 and 7.3 present collision conditions and collision avoidance guidance laws for point object vs deforming object, circular object vs deforming object and arbitrarily shaped object vs deforming object, respectively. Section 7.4 presents simulations demonstrating the working of these guidance laws.

7.1 Collision between a point object and a deforming object

7.1.1 Collision Conditions

Consider an engagement between a point object \( A \) and a finite-sized object \( B \), which has the capability to deform by changing the shape of its boundary. \( B \) can represent an oil spill, or a swarm of vehicles, as schematically depicted in Fig 7.1. The velocity of each point on the boundary or perimeter of \( B \) is written as the vector sum of the individual deformation component of that point and an overall translational component. Let \( s \) represent a curvilinear distance along the boundary of \( B \), measured with respect to some reference point, and let the length of the perimeter of \( B \) be represented by \( L \). In Fig 7.1 \( C_1 \) is chosen as the reference point and a point at a distance \( s \) from \( C_1 \) is referred to as \( B(s) \). Points on the perimeter of \( B \) are parametrized as \( B(s), s \in [0, L] \). Let \( r(s) \) represent the distance between \( A \) and \( B(s) \), and \( \theta(s) \) represent the angle made by the line joining \( A \) and \( B(s) \). The points \( B(s_1), B(s_2), B(s_3), ... \) are denoted by \( C_1, C_2, C_3, ... \) in the figure. Let \( V_B(s) \) represent the magnitude of the velocity at each point \( B(s) \), and \( \beta(s) \) represent the corresponding direction of this velocity. Then, we can construct multiple lines \( AC_1, AC_2, ..., \) each of which have their distinct values of \( V_\theta \) and \( V_r \). We can write the relative velocity components \( V_r(s) \equiv \dot{r}(s) \) and \( V_\theta(s) = r(s)\dot{\theta}(s) \) as follows:

\[
V_r(s) = V_B(s) \cos(\beta(s) - \theta(s)) - V_A \cos(\alpha - \theta(s)) \quad (7.1)
\]
Figure 7.1: Engagement geometry between a point object and a deforming object

\[ V_\theta(s) = V_B(s) \sin(\beta(s) - \theta(s)) - V_A \sin(\alpha - \theta(s)) \] (7.2)

We then state the following lemma governing collision between \( A \) and \( B(s) \).

Lemma 7.1: Consider a point object \( A \) and a finite object \( B \) that is simultaneously moving and deforming such that the points \( B(s) \) are all moving with velocities that are constant in time. Then, \( A \) is on a collision course with \( B \) if and only if there exists at least one ray \( AB(s_i) \) passing through \( B \) that has the properties \( V_\theta(s_i) = 0 \), \( V_r(s_i) < 0 \).

Proof: Follows from Lemma presented in [99] between point object and non-deforming arbitrary object.

Assume that \( V_B(s) \) and \( \beta(s) \) both vary continuously from \( s = 0 \) to \( s = L \). Refer Fig 7.2(a), where the object \( B \) is an open chain, an example of which would be a snake robot moving on a plane. Furthermore, in Fig 7.2(a), any ray \( AC \) passing through \( B \) will intersect \( B \) at exactly one point. We then have the following lemma:

Lemma 7.2: Let \( B(s) \) be an open chain, and let the engagement geometry between \( A \) and \( B \) be such that any ray \( AC \) passing through \( B \) will intersect \( B \) at exactly one point.
Then, the condition:

\[ V_\theta(s = 0)V_\theta(s = L) \leq 0 \]  \hspace{1cm} (7.3)

is both necessary and sufficient for there to exist exactly one \( s_i \in [0, L] \) that satisfies \( V_\theta(s_i) = 0 \).

Proof: Since \( V_B(s) \) and \( \beta(s) \) are both continuous functions of \( s \), therefore from (7.1), (7.2), \( V_\theta(s) \) also varies continuously with \( s \). Hence, the condition \( V_\theta(s = 0)V_\theta(s = L) \leq 0 \) implies that the function \( V_\theta(s) \) has exactly one zero crossing for \( s \in [0, L] \). The converse is similarly true.

Next, consider the scenario in Fig 7.2(b)-(d). In Fig 7.2(b), \( B(s) \) is still an open chain. However, there can be some rays emanating from \( A \) that will intersect \( B(s) \) at more than one point. In Fig 7.2(c), (d), \( B(s) \) is a closed chain (which is also non-convex), examples of which could be the boundary of an oil spill or a swarm of vehicles. In (c), any ray \( AC \) passing through \( B(s) \) will intersect \( B(s) \) at two points \( C_1 \) and \( C_2 \), while in (d), there are some rays emanating from \( A \) that can intersect \( B(s) \) at four points. When \( B(s) \) is a closed chain, then \( V_\theta(s = 0) = V_\theta(s = L) \), and in such scenarios, we cannot rely on (7.3) to determine collision conditions. We instead use the following lemma:

Lemma 7.3: Let \( B(s) \) be a closed chain. Then, the condition

\[
\max_{s \in [0, L]} V_\theta(s) \min_{s \in [0, L]} V_\theta(s) \leq 0
\]  \hspace{1cm} (7.4)

is both necessary and sufficient for there to exist at least one \( s_i \in [0, L] \) that satisfies \( V_\theta(s_i) = 0 \).

Proof: Since \( V_\theta(s) \) is a continuous function of \( s \), therefore when (7.4) holds, this implies that the function \( V_\theta(s) \) has at least one zero crossing. The converse is similarly true.
When the conditions of either Lemma 3 or Lemma 4 are satisfied, we then define the collision cone as the set of heading angles of $A$ that will cause $A$ to collide with $B(s)$. A geometric representation of the collision cone is shown in Fig 7.3. In this figure, the blue curve shows the boundary of $B(s)$ at the current time $t$. Consider a few representative points $B(s_1), B(s_2), \ldots$ as shown. The vectors $V_B(s_1), V_B(s_2), \ldots$ represent the respective velocities of these points. Shift each of these velocity vectors so that they are positioned at $A$. For illustrative purposes, consider the velocity vector $V_B(s_1)$. Shift $V_B(s_1)$ to $A$. Then, draw a line emanating from the tip of this shifted $V_B(s_1)$ so that this line is parallel with $AB(s_1)$ and
is of the same length as $AB(s_1)$. This line is shown as $A\hat{B}(s_1)$ in the figure. Repeat the same for the other points $B(s_2), B(s_3), ...$ and thus obtain the points $\hat{B}(s_2), \hat{B}(s_3), ...$. With $A$ as center, draw an arc whose radius is equal to the speed $V_A$ of $A$. Then, $P$ and $Q$ represent the extremal points of intersection of this arc with the rays emanating from the tip of the shifted velocity vectors of $B(s)$. The cone $PAQ$ then represents an approximation of the collision cone. The exact collision cone would be one where the same procedure is applied to all the points on the perimeter of $B$. It can be seen from the figure that the extremities of the collision cone do not necessarily correspond to the extremities of the deforming object. For instance, the lower boundary of the collision cone passes through $Q$. If object $A$ has a heading angle along $AQ$, then $A$ will collide with $B(s_4)$ even though $B(s_3)$ lies to the right of $B(s_4)$ (from the vantage point of $A$).

![Figure 7.3: Geometric interpretation of the collision cone between a point object and a finite-sized deforming object](image)

From (7.2), the condition that $V_\theta(s) = 0$ is written as:

$$V_B(s) \sin(\beta(s) - \theta(s)) = V_A \sin(\alpha - \theta(s))$$

(7.5)
From (7.5), for each \(s \in [0, L]\), the corresponding heading angle \(\alpha\) that will ensure that \(V_\theta(s)\) is zero, is as follows:

\[
\alpha = \theta(s) + \sin^{-1} \left[ \frac{V_B(s)}{V_A} \sin(\beta(s) - \theta(s)) \right] \tag{7.6}
\]

The collision cone is then defined as the set of \(\alpha\) that satisfy (7.6) for each \(s\), as well as \(V_r(s) < 0\), that is,

\[
C_\alpha = \{ \alpha : V_\theta(s) = 0 \cap V_r(s) < 0, \ s \in [0, L] \} \tag{7.7}
\]

If the heading angle of \(A\) is such that it lies inside the collision cone, we can obtain an expression for the time to collision between \(A\) and \(B(s)\) as follows:

\[
t_m(s) = \frac{-r(s)V_r(s)}{V_r(s)^2 + V_\theta(s)^2} \tag{7.8}
\]

where, it is understood that \(t_m(s)\) always takes positive values, since the values of \(V_r(s)\) corresponding to the collision cone are all negative.

### 7.1.2 Guidance Laws for Collision Avoidance

In this section, we determine guidance laws that will ensure that \(A\) avoids a collision with \(B\). Let the acceleration magnitude of \(A\) be \(a_A\), applied at an angle of \(\delta_A\). Then, the kinematics of the engagement between \(A\) and \(B(s)\) are represented by the following equations, which are valid for all \(s \in [0, L]\):

\[
\begin{align*}
\dot{r}(s) &= V_r(s) \tag{7.9} \\
\dot{\theta}(s) &= \frac{V_\theta(s)}{r(s)} \tag{7.10} \\
\dot{V}_\theta(s) &= \frac{-V_\theta(s)V_r(s)}{r(s)} - a_A \sin(\delta_A - \theta(s)) \tag{7.11} \\
\dot{V}_r(s) &= \frac{V_\theta(s)^2}{r} - a_A \cos(\delta_A - \theta(s)) \tag{7.12}
\end{align*}
\]
Define a set $S$ as follows:

$$S = \{ s \in [0, L] : V_r(s) < 0 \} \quad (7.13)$$

To determine a guidance law for collision avoidance, define a candidate Lyapunov function as follows:

$$Z = \frac{1}{2} [\max_{s \in S} V_\theta(s)]^2 \quad (7.14)$$

The time derivative of the Lyapunov function along the trajectories of the system (7.9)-(7.12) is:

$$\dot{Z} = m[V_\theta(s)] \times \begin{bmatrix} \frac{\partial m[V_\theta(s)]}{\partial r(s)} & \frac{\partial m[V_\theta(s)]}{\partial \theta(s)} & \frac{\partial m[V_\theta(s)]}{\partial V_r(s)} & \frac{\partial m[V_\theta(s)]}{\partial V_\theta(s)} \end{bmatrix} \times \begin{bmatrix} \dot{r}(s) \\ \dot{\theta}(s) \\ \dot{V}_\theta(s) \\ \dot{V}_r(s) \end{bmatrix} \quad (7.15)$$

In the above equation, $m[V_\theta(s)] \equiv \max_{s \in S} V_\theta(s)$, and it is understood that $\frac{\partial(\cdot)}{\partial r(s)}$ is a vector of the form:

$$\frac{\partial(\cdot)}{\partial r(s)} = \begin{bmatrix} \frac{\partial(\cdot)}{\partial r(s_1)} & \frac{\partial(\cdot)}{\partial r(s_2)} & \cdots \end{bmatrix} \quad (7.16)$$

The expressions for the vector quantities $\frac{\partial(\cdot)}{\partial \theta(s)}$, $\frac{\partial(\cdot)}{\partial V_\theta(s)}$ and $\frac{\partial(\cdot)}{\partial V_r(s)}$ can be similarly written. Defining

$$\bar{s} \equiv \arg \max_{s \in S} V_\theta(s) \quad (7.17)$$

the time derivative $\dot{Z}$ can be written as:

$$\dot{Z} = V_\theta(\bar{s}) V_\theta(\bar{s}) = V_\theta(\bar{s}) \left[ -V_\theta(\bar{s}) V_r(\bar{s}) \frac{r(\bar{s})}{r(\bar{s})} - a_A \sin(\delta_A - \theta(\bar{s})) \right] \quad (7.18)$$
By choosing a guidance law of the form:

\[ a_A = V_\theta(\bar{s}) \frac{K - V_\tau(\bar{s})/r(\bar{s})}{\sin(\delta_A - \theta(\bar{s}))} \]  \hspace{1cm} (7.19)

where \( K > 0 \), it can be ensured that the Lyapunov function \( Z \) follows the dynamics \( \dot{Z} = -KZ \). Eqn \( (7.19) \) thus guarantees that collision avoidance is achieved, provided the gain \( K \) is sufficiently large to ensure that \( L \) decays to a value \( \epsilon \) before time \( t_m(\bar{s}) \).

From \( (7.19) \), it is seen that we can also choose a guidance law for \( A \) of the form:

\[ a_A = K_1 V_\theta(\bar{s}) \]  \hspace{1cm} (7.20)

where, \( K_1 \) satisfies \( K_1 > \frac{K - V_\tau(\bar{s})/r(\bar{s})}{\sin(\delta_A - \theta(\bar{s}))} \).

We note that depending on the shape of the object \( B \), there can be discontinuities in the value of \( \bar{s} \), as the engagement evolves in time. For the sake of simplicity, the analysis involving such discontinuities is omitted in this dissertation.

We note that alternative collision avoidance laws are also possible. For instance, we could use a guidance law that will attempt to drive the minimum value of \( V_\theta(s) \) over \( s \in S \) to zero, and this too will achieve collision avoidance. If we define the Lyapunov function as:

\[ Z = \frac{1}{2} [\min_{s \in S} V_\theta(s)]^2 \]  \hspace{1cm} (7.21)

then by following a similar set of steps as before, we arrive at a guidance law similar to \( (7.19) \), with \( \bar{s} \) now defined as \( \bar{s} \equiv \arg \min_{s \in S} V_\theta(s) \).

When the translational and/or deformation velocities of the object \( B(s) \) vary with time, that is, \( B(s) \) has an acceleration of magnitude \( a_B(s) \), acting at angles given by \( \delta_B(s) \), then the Lyapunov function follows the dynamics:

\[ \dot{Z} = -KZ + a_B(\bar{s})V_\theta(\bar{s}) \sin(\delta_B(\bar{s}) - \theta(\bar{s})) \]  \hspace{1cm} (7.22)
From (7.22), it is evident that as along as the acceleration law (7.19) uses a $K$ that is large enough to keep the right hand side of (7.22) negative definite, and furthermore, as $L \to 0$, the deformation of $B(s)$ is such that $a_B(s) \to 0$, then the guidance law (7.19) will guarantee that $A$ avoids collision with $B$.

7.2 Collision between a circular object and a deforming object

7.2.1 Collision Conditions

Consider an engagement between a circular object $A$ (which is non-deforming) and a deforming object $B$, as shown in Fig 7.4. Let $P$ represent the center of the circle $A$. For every possible line that can be drawn from $P$ to the boundary $B(s)$, let $r(s)$ represent the length and $\theta(s)$ represent the bearing angle. Similarly, let $V_r(s)$ and $V_\theta(s)$ represent the relative velocity components of each such line. We can then use (2.12) to define a function $y(s)$ as follows:

$$y(s) = \frac{r(s)^2 V_\theta(s)^2}{V_\theta(s)^2 + V_r(s)^2} - R^2$$

(7.23)

Figure 7.4: Engagement geometry between a circular object and a deforming object
We then have the following Lemma.

Lemma 7.4: Let \( A \) and \( B(s) \) move with constant velocity vectors \( \vec{V}_A \) and \( \vec{V}_B(s) \), respectively. Then, \( A \) is on a collision course with \( B(s) \) if and only if there exist some \( s \in [0, L] \) that satisfy \( y(s) < 0 \) and \( V_r(s) < 0 \).

Proof: Follows from the above discussion.

The function \( y(s) \) can then be used to define the collision cone between \( A \) and \( B(s) \) as follows:

\[
C_\alpha = \{ \alpha : y(s) < 0 \cap V_r(s) < 0, s \in [0, L] \}
\]  

(7.24)

7.2.2 Guidance Laws for Collision Avoidance

An avoidance acceleration for \( A \) is one that drives the minimum value of \( y(s) \) that lies in the set \( S \) to zero. Accordingly, define a candidate Lyapunov function as:

\[
Z = \frac{1}{2} \left[ \min_{s \in S} y(s) \right]^2
\]  

(7.25)

The time derivative of the Lyapunov function along the trajectories of the system (7.9)-(7.12) is:

\[
\dot{Z} = m[y(s)] \times \left[ \frac{\partial m[y(s)]}{\partial r(s)} \frac{\partial m[y(s)]}{\partial \theta(s)} \frac{\partial m[y(s)]}{\partial V_\theta(s)} \frac{\partial m[y(s)]}{\partial V_r(s)} \right] \times \begin{bmatrix} \dot{r}(s) \\ \dot{\theta}(s) \\ \dot{V}_\theta(s) \\ \dot{V}_r(s) \end{bmatrix}
\]  

(7.26)

In the above equation, for the sake of brevity, we have employed the notation

\[
m[y(s)] = \min_{s \in S} y(s)
\]  

(7.27)
Defining \( \bar{s} \equiv \arg \min_{s \in S} y(s) \), the time derivative \( \dot{L} \) can be written as:

\[
\dot{Z} = y(\bar{s})\dot{y}(\bar{s}) = y(\bar{s}) \left[ \frac{-V_\theta(\bar{s})V_r(\bar{s})}{r(\bar{s})} - a_A \sin(\delta_A - \theta(\bar{s})) \right]
\]  

(7.28)

By choosing a guidance law of the form:

\[
a_A = \left[ -0.5K(V_\theta(\bar{s})^2 + V_r(\bar{s})^2)/r(\bar{s})^2 \right] \left[ r(\bar{s})^2V_\theta(\bar{s})^2 - R^2(V_\theta(\bar{s})^2 + V_r(\bar{s})^2) \right] / \left[ -V_\theta(\bar{s})V_r(\bar{s})^2 \sin(\delta - \theta(\bar{s})) + V_r(\bar{s})V_\theta(\bar{s})^2 \cos(\delta - \theta(\bar{s})) \right]
\]

(7.29)

where \( K > 0 \), it can be ensured that the Lyapunov function \( Z \) follows the dynamics \( \dot{Z} = -KZ \). Eqn (7.29) thus guarantees that collision avoidance is achieved, provided the gain \( K \) is sufficiently large to ensure that \( Z \) decays to a value \( \epsilon \) before time \( t_m(\bar{s}) \).

7.3 Collision between an arbitrarily shaped object and a deforming object

7.3.1 Collision Conditions

Consider an engagement between an arbitrarily shaped object \( A \) (which is non-deforming) and a deforming object \( B \), as in Fig 7.5. From each point \( B(s) \) on the boundary of \( B \), we construct tangents to \( A \), and we consider the pair of tangents that subtend the largest angle at \( A \). For instance, in Fig 7.5, the tangents to \( A \) emanating from \( C_1 \) are the lines \( C_1P_{C1} \) and \( C_1Q_{C1} \) and the angle between these tangents is denoted as \( \psi(s_1) \). Similarly, the tangents to \( A \) emanating from \( C_3 \) are the lines \( C_3P_{C3} \) and \( C_3Q_{C3} \), and the angle between these tangents is \( \psi(s_3) \). In this way, we can define a function \( \psi(s) \).
Figure 7.5: Engagement geometry between an arbitrary object and a deforming object

The quantities \( V_r(s_1) \) and \( V_\theta(s_1) \) represent the relative velocity components of the angular bisector of the sector \( P_{C1}C_1Q_{C1} \). Similarly, \( V_r(s_3) \), \( V_\theta(s_3) \) represent the relative velocity components of the angular bisector of the sector \( P_{C3}C_3Q_{C3} \). In this way, we can define functions \( V_\theta(s) \) and \( V_r(s) \). Eventually, we can use the functions \( \psi(s) \), \( V_\theta(s) \) and \( V_r(s) \) to define a function \( y(s) \) as follows:

\[
y(s) = \frac{V_\theta(s)^2}{V_\theta(s)^2 + V_r(s)^2} - \sin^2 \frac{\psi(s)}{2} \quad (7.30)
\]

Using the above definitions of \( y(s) \) and \( V_r(s) \), we can state the following Lemma:

Lemma 7.5: Let \( A \) be an arbitrarily shaped object moving with a constant velocity \( \vec{V}_A \) and \( B \) be a deforming object whose boundary \( B(s) \) is moving with velocity \( \vec{V}_B(s) \), where \( \vec{V}_B(s) \) is constant in time, for each \( s \). Furthermore, assume the engagement is such that \( \psi(s) < \pi, \forall s \in [0, L] \). Then, \( A \) is on a collision course with \( B(s) \) if and only if there exist some \( s \in [0, L] \) that satisfy \( y(s) < 0 \) and \( V_r(s) < 0 \).
Proof: Follows from the above discussion.

The collision cone between $A$ and $B(s)$ is defined as in (7.24), where $y(s)$ is as defined in (7.30) and $V_r(s)$ is defined as discussed above.

### 7.3.2 Guidance Laws for Collision Avoidance

An avoidance acceleration for $A$ is one that drives the minimum value of $y(s)$ to zero. By defining a Lyapunov function as given in (7.25), with $y(s)$ defined as in (7.30), and performing a series of steps similar to (7.28), we can obtain a guidance law for collision avoidance between the arbitrary object $A$ and the deforming object $B(s)$ as follows:

$$a_A = \left[ -Ky(s)(V_r(s)^2 + V_\theta(s)^2)^2 + 2(V_r(s)^2 + V_\theta(s)^2)V_\theta(s)^2 \operatorname{cosec}^2 \left( \frac{\psi(s)}{2} \right) 
\left( V_r(s)\dot{\theta}(s)/V_\theta(s) + \cot \left( \frac{\psi(s)}{2} \right) \frac{\psi(s)}{2} \right) \right] \left[ 2V_r(s)V_\theta(s) \operatorname{cosec}^2 \left( \frac{\psi(s)}{2} \right) (-V_r(s) \sin(\delta - \theta(s)) + V_\theta(s) \cos(\delta - \theta(s))) \right] (7.31)$$

### 7.4 Simulation Results

In this section, we present simulations that demonstrate the working of the guidance laws developed above. We consider the case of an engagement between an arbitrarily shaped object $A$ and a deforming object $B$. $A$ is taken to be an oval shape, while $B$ is an object, such that its boundary $B(s)$ has seven points $s_1, ..., s_7$ moving with distinct velocities $V_B(s_1), ..., V_B(s_7)$. The velocities of all the points on the line $s_is_{i+1}$, are convex combinations of the velocities $V_B(s_i)$ and $V_B(s_{i+1})$. The speed distribution $V_B(s)$ and heading angle distribution $\beta(s)$ are given in Fig 7.7. The influence of this speed distribution is that $B$ follows a translation and deformation trajectory as shown in Fig 7.6. It is seen from Fig 7.6 that the net effect of the deformations is to cause the length $L$ of $B(s)$ to increase with time. Accordingly, the domain of the horizontal axis in Fig 7.7 is small at $t = 0$, and then becomes progressively larger with time.
A moves with a constant speed of 3 m/sec, and its initial heading angle is 50°. Its initial heading angle lies inside the collision cone, and if A continues to move with its initial velocity vector, it would collide with B. However, using the developed guidance law (7.31), A is able to drive its velocity vector out of the collision cone and avert the collision, as evident from Fig 7.6. The relative velocity functions $V_\theta(s)$ and $V_r(s)$ are shown in Fig 7.8, as time snapshots at several instants in time. The angle function $\psi(s)$ is shown at the same times in Fig 7.9. Using $V_\theta(s)$, $V_r(s)$ and $\psi(s)$, the resulting $y(s)$ is shown in Fig 7.10. As is seen from this figure, at time $t = 0$, the $y(s)$ function is negative for some $s$, and $V_r(s)$ is negative for almost all $s$ (Fig 7.8). Under the influence of the guidance laws, the function $y(s)$ is driven to be non-negative for all $s$. Fig 7.11 shows the acceleration profile. This acceleration was applied at an angle normal to the velocity vector of A, and therefore changed the heading angle of A with no change in speed. The change in the heading angle of A is also shown in Fig 7.11.

Figure 7.6: Spatial trajectories of object A and deforming object B
Figure 7.7: Velocity distribution of $B(s)$ at several instants in time

Figure 7.8: Relative velocity functions $V_r(s)$ and $V_\theta(s)$ at several instants in time
Figure 7.9: Angle function $\psi(s)$ at several instants in time

Figure 7.10: Collision cone function $y(s)$ at several instants in time
Figure 7.11: Time histories of applied acceleration and heading angle of object $A$
CHAPTER 8

ACHIEVING OVERLAP OF MULTIPLE ARBITRARILY SHAPED FOOTPRINTS USING RENDEZVOUS CONES

In this chapter, we discuss using the notion of rendezvous cones to achieve overlap of multiple arbitrarily shaped footprints. Section 8.2 develops the equations for the rendezvous cone associated with objects of arbitrary shapes. Section 8.3 develops the analytical guidance laws that enable a vehicle’s velocity vector to enter the rendezvous cone from any initial condition, and thereby achieve overlap of the vehicles footprints. Section 8.4 presents simulations that demonstrate the working of these guidance laws.

8.1 Hyperbolic Geometry

In this section, we review some fundamental definitions in hyperbolic geometry. A hyperbola $H$ is the locus of points such that the difference between the distance of a point to the two foci is a constant. Thus, if a point $P$ on a hyperbola has distances $r_1$ and $r_2$ to the two foci, then $|r_1 - r_2| = 2a$, where $a$ is the semi-major axis of the hyperbola. A rectangular hyperbola is a hyperbola whose two asymptotes are orthogonal to each other. A rectangular hyperbola that has the cartesian coordinate axes $OX$ and $OY$ as its asymptotes is represented by the equation $xy = 1$. A hyperbolic sector is a region of the Cartesian plane bounded by rays from the origin to two points $(a, 1/a)$ and $(b, 1/b)$ and by the rectangular hyperbola $xy = 1$. When this hyperbola is rescaled and its orientation altered by a rotation, the hyperbolic sector can be correspondingly defined. A hyperbolic sector in standard position has $a = 1$ and $b > 1$. When in standard position, a hyperbolic sector determines a hyperbolic triangle, which in this case is a right-angled triangle that has one vertex at the origin. The base of this triangle lies on the line $y = x$, and the third vertex lies on the hyperbola $xy = 1$, with the hypotenuse being the segment from the origin to the point $(x, y)$ on the hyperbola. The length of the base of this triangle is $\sqrt{2}\cosh u$, and its height is $\sqrt{2}\sinh u$, where $u$ is
the hyperbolic angle. The hyperbolic angle \( u \) determines a hyperbolic sector that has area \( u \), and this is shown as the shaded region in Fig 8.1.

![Figure 8.1: Illustration of hyperbolic sector and hyperbolic angle](image_url)

The above concepts will be seen to be critical to model the overlap of sensor footprints and to quantify the penetration of one footprint into another.

### 8.2 Rendezvous cone for arbitrary shapes

#### 8.2.1 Problem Formulation

Refer Figure 8.2(a),(b). In these figures, \( A \) and \( B \) represent two circular vehicles, of radii \( R_A \) and \( R_B \), respectively. Here, \( A' \) and \( B' \) represent their respective footprints. These footprints are arbitrarily shaped, and may be convex or non-convex. It is assumed that each vehicle can be located anywhere inside the footprint and not necessarily in the central region of its footprint. The quantities \( V_A \) and \( V_B \) represent the velocities of \( A \) and \( B \), and these velocity vectors make angles of \( \alpha \) and \( \beta \), respectively, with the horizontal. In Fig 8.2(a), the geometry is such that the sizes of the footprints \( A' \) and \( B' \) are of the same order of magnitude, while in Fig 8.2(b), the size of footprint \( A' \) is relatively small, and is such that it lies close to the larger, non-convex footprint \( B' \). While the majority of scenarios would be similar to that shown in Figure 8.2(a), the theory and guidance laws developed in this dissertation are applicable to the scenario of Figure 8.2(b) as well. We note that the shapes \( A' \) and \( B' \) could also represent boundaries of two vehicle swarms attempting to merge with one another.
Depending on the specific application, the objective may be either of the following:

(i) Generate trajectories that achieve an overlap of footprints $A'$ and $B'$, while ensuring that the vehicles $A$ and $B$ do not collide. This would be required in applications involving sweep operations such as those illustrated in Fig 1.3(a).

(ii) Generate trajectories that cause vehicle $A$ to lie within the footprint $B'$, while ensuring that $A$ does not collide with $B$. This would be required in applications involving communication operations such as those illustrated in Fig 1.3(b),(c).

Figure 8.2: Vehicles and their footprints: (a) $\psi < \pi$, (b) $\psi > \pi$
More precise statements for the problems corresponding to (i) and (ii) are as follows:

Problem (i): Given two moving vehicles $A$ and $B$, with footprints whose shapes are $A'$ and $B'$, respectively, determine acceleration laws for $B$ that will generate a trajectory to cause $B'$ to overlap with $A'$, to a specified depth of overlap $c$, while ensuring collision avoidance between $A$ and $B$.

Problem (ii): Given two moving vehicles $A$ and $B$, and the footprint of $A$ is $A'$, determine acceleration laws for $B$ that will generate a trajectory to cause $B$ to move inside $A'$, to a specified depth of overlap $c$, while ensuring collision avoidance between $A$ and $B$.

It is evident that by defining $B' \equiv B$, Problem (ii) can be considered as a special case of Problem (i).

In the context of this dissertation, two shapes are said to have performed a rendezvous when they make contact, and subsequently overlap with one another to some (possibly pre-defined) extent of overlap. The extent of overlap becomes an important metric in scenarios such as those in Fig 1.3(b), where it is required that S-UAV1 and S-UAV2 achieve not just an arbitrary overlap of their footprints, but also ensure that the extent of this overlap is sufficient to accommodate the physical dimensions of R-UAV1 (the relay UAV). In this dissertation, we quantify the extent of this overlap using concepts from hyperbolic geometry.

The rendezvous cone is defined as the cone of relative velocities that enable the above defined rendezvous. Thus, in (i) above, the rendezvous cone is the cone of relative velocities that enables $A'$ to rendezvous with $B'$, while in (ii) it is defined as the cone of relative velocities that enable $A$ to rendezvous with $B'$. We define $t_0$ as the initial time, $t_\psi$ as the time instant at which the two shapes just touch each other at their boundaries, and $t_f$ as the time instant at which they have achieved their desired extent of overlap. The overall rendezvous guidance then comprises two phases: the first phase is when $t \in [t_0, t_\psi)$ and during this phase the two shapes are separated, while the second phase is when $t \in [t_\psi, t_f]$ and during this phase the two shapes have overlapped.
We next determine the conditions under which $A'$ and $B'$ will make contact (that is, just touch each other at their boundaries), after which we determine the conditions under which $A'$ and $B'$ will overlap to a specified extent.

8.2.2 Conditions under which $A'$ and $B'$ will make contact, while avoiding collision between $A$ and $B$

In Figure 8.2, let $P_1$ and $P_2$ represent the centers of the circles $A$ and $B$. Then, the relative velocity components of the line $P_1P_2$ defined as $V_\theta$ and $V_r$ are given by equations (2.1), (2.2) where, $\theta$ denotes the angle made by $P_1P_2$ with respect to the horizontal. For any ray $C_1C_2$ passing through $A'B'$ (where $C_1$ is a point in $A'$ and $C_2$ is a point in $B'$), we can write the relative velocity components of line $C_1C_2$ in terms of $V_\theta$ and $V_r$ as given in equations 2.14 where, $\hat{\theta}$ represents the angle made by the line $C_1C_2$ with the horizontal. Since equations (2.14) are both continuous in $\hat{\theta}$, therefore as we sweep $\hat{\theta}$ from ray $Q_1Q_2$ to ray $R_1R_2$, the relative velocity components vary continuously.

Using the result given in Section 2.1, we can see that $A'$ and $B'$ are on a path to rendezvous if and only if there exists a ray $D_1D_2$, passing through $A'$ and $B'$ (with $D_1$ on $A'$ and $D_2$ on $B'$), that satisfies the property $V_{\theta(D_1D_2)} = 0, V_{r(D_1D_2)} < 0$. In Fig 8.2(a), there can be only one ray satisfying $V_{\theta(D_1D_2)} = 0$ (that is, one zero crossing in the quantity on the left hand side of (8.2), as we perform an angular sweep from $V_{\theta(Q_1Q_2)}$ to $V_{\theta(R_1R_2)}$. Since the relative velocity components vary continuously between the lines $Q_1Q_2$ and $R_1R_2$, we have that:

$$V_{\theta(D_1D_2)} = 0 \iff V_{\theta(Q_1Q_2)}V_{\theta(R_1R_2)} \leq 0$$

(8.1)

The relative velocity components $V_{\theta(Q_1Q_2)}$ and $V_{\theta(R_1R_2)}$ can be defined by substituting $\hat{\theta} = \theta + \psi_1, \hat{\theta} = \theta - \psi_2$, respectively, in (2.1 and 2.2). After making these substitutions, it can be seen that the condition $V_{\theta(Q_1Q_2)}V_{\theta(R_1R_2)} \leq 0$ is equivalent to the condition:

$$V_\theta^2 \cos \psi_1 \cos \psi_2 - V_\theta V_r \sin (\psi_1 - \psi_2) - V_r^2 \sin \psi_1 \sin \psi_2 \leq 0$$

(8.2)
The angles $\psi_1$ and $\psi_2$ are as shown in Fig 8.2(a). Using the above conditions we define the quantity $y_1$ as in equation 2.12.

Additionally, the relative velocity components along and normal to the angular bi-sector of the sector $Q_2OR_2$ is given by equations 2.14 where, $\hat{\theta} = \frac{1}{2}(\theta_1 + \theta_2)$. Defining the angle $\psi$ as $\psi = \psi_1 + \psi_2$, $\psi$ is then the angle of the smallest cone that can be constructed such that it completely contains $A'$ and $B'$, with $A'$ and $B'$ lying on opposite sides of the vertex of this cone. Eqn (2.12) can also be written in terms of the angles $\hat{\theta}$ and $\psi$. Noting that $\psi_1 - \psi_2 = 2(\hat{\theta} - \theta)$, we obtain $\psi_1 = \frac{\psi}{2} + (\hat{\theta} - \theta)$ and $\psi_2 = \frac{\psi}{2} - (\hat{\theta} - \theta)$, and substituting the same in (2.12), we get:

$$y_1 = \frac{V_\theta^2 \cos^2 (\hat{\theta} - \theta) - 2V_\theta V_r \cos (\hat{\theta} - \theta) \sin (\hat{\theta} - \theta) + V_r^2 \sin^2 (\hat{\theta} - \theta)}{V_\theta^2 + V_r^2} - \sin^2 \left(\frac{\psi}{2}\right), t \leq t_\psi$$

Fig 8.2(a) corresponds to the case when $\psi < \pi$. Thus, for the case $\psi < \pi$, when $A$ and $B$ move with constant velocities, the conditions $y_1 < 0$ and $\hat{V}_r < 0$ together form necessary and sufficient conditions for $A'$ to rendezvous with $B'$, at some future time. When $A$ and/or $B$ move with varying velocities, satisfaction of these inequalities at any instant $t$ means that, at that instant, $A'$ and $B'$ are on a path to rendezvous with each other. For $\psi < \pi$, the rendezvous cone $\mathcal{R}$ is defined in the relative velocity space as:

$$\mathcal{R}(t) = \{(V_\theta, V_r) : y_1(t) < 0 \text{ and } \hat{V}_r(t) < 0\}$$

(8.4)

The rendezvous cone can equivalently be defined in physical space as:

$$\mathcal{R}_\alpha(t) = \{\alpha : y_1(t) < 0 \text{ and } \hat{V}_r(t) < 0\}$$

(8.5)

When $\psi > \pi$ (see Fig 8.2(b)), there can be two rays satisfying $V_\theta(D_1D_2) = 0$ as we perform an angular sweep from $V_\theta(Q_1Q_2)$ to $V_\theta(R_1R_2)$. These two rays are $180^0$ apart. Thus, a ray satisfying $V_\theta(D_1D_2) = 0$ can exist even in the $y_1 > 0$ region. Because of this, the set
of velocities that will cause \( A' \) to overlap with \( B' \) becomes larger than that defined by the set \( y_1 < 0, \hat{V}_r < 0 \), and in fact contains even a portion of the \( y_1 > 0 \) set. For \( \psi > \pi \), the rendezvous cone is defined in the relative velocity space as:

\[
\mathcal{R}(t) = \{(V_\theta, V_r) : \hat{V}_r(t) < 0 \cup (y_1 > 0 \text{ and } \hat{V}_r(t) > 0)\}
\] (8.6)

and, equivalently, in physical space as:

\[
\mathcal{R}_\alpha(t) = \{\alpha : \hat{V}_r(t) < 0 \cup (y_1 > 0 \text{ and } \hat{V}_r(t) > 0)\}
\] (8.7)

We desire that \( A' \) and \( B' \) rendezvous with each other, while also ensuring that \( A \) and \( B \) do not collide with each other. This becomes particularly important when either of \( A \) or \( B \) lie close to the perimeter of their respective footprints. Since \( R_A \) and \( R_B \) denote the radii of the sensors \( A \) and \( B \), respectively, the collision condition between \( A \) and \( B \) can be found by substituting \( \psi_1 = \psi_2 = \psi_1 = \sin^{-1} \left( \frac{R_A + R_B}{r} \right) \) in (8.2). Additionally, we substitute \( \hat{\theta} = \theta \) in (2.14). Doing so, the conditions (8.2) and (2.14), respectively, become as follows:

\[
V_\theta^2 \left[ \frac{(R_A + R_B)^2 - r^2}{r^2} \right] - V_r^2 \left[ \frac{(R_A + R_B)^2}{r^2} \right] < 0 \quad (8.8)
\]

\[
V_r < 0 \quad (8.9)
\]

Dividing (8.8) by the total relative velocity \( V_\theta^2 + V_r^2 \), we can define a quantity \( y_2 \) as follows:

\[
y_2 = \frac{V_\theta^2 \left( \frac{(R_A + R_B)^2 - r^2}{r^2} \right) - V_r^2 \left( \frac{(R_A + R_B)^2}{r^2} \right)}{V_\theta^2 + V_r^2} \quad (8.10)
\]

The collision cone \( \mathcal{C} \) is then defined in the relative velocity space as:

\[
\mathcal{C}(t) = \{(V_\theta, V_r) : y_2(t) < 0 \text{ and } V_r(t) < 0\}
\] (8.11)
and, equivalently in physical space as:

\[ C_\alpha(t) = \{ \alpha : y_2(t) < 0 \text{ and } V_r(t) < 0 \} \]  

(8.12)

Combining (8.2), (2.14) and (8.8), (8.9) we see that the region in the \((V_\theta, V_r)\) space that corresponds to the set of relative velocities that cause \(A'\) and \(B'\) to lie on a rendezvous course, while simultaneously ensuring that \(A\) and \(B\) are not on a collision course, is as shown in Figure 8.3. The boundaries of the \(y_1 < 0\) and \(y_2 < 0\) regions are obtained by replacing the inequality signs in (8.2) and (8.8), respectively, with equality signs, and solving each of the resulting equations for \((V_\theta, V_r)\). Note that the boundaries of the rendezvous cone are (in general) not symmetric with respect to the negative \(V_r\) axis, while those of the collision cone are symmetric. In order to achieve rendezvous of \(A'\) and \(B'\) while at the same time ensuring collision avoidance between \(A\) and \(B\), the \((V_\theta, V_r)\) trajectory needs to lie in the shaded region.

Figure 8.3: Representation of the rendezvous and collision cones on the \((V_\theta, V_r)\) plane, for 
(a) \(\psi < \pi\), (b) \(\psi > \pi\)

From Fig 8.2(a),(b), we can see that \(\psi \equiv \psi_1 + \psi_2\) is equal to the angle subtended by the common tangents to \(A'\) and \(B'\). At the instant \(t_\psi\) when \(A'\) and \(B'\) just touch each other at their boundaries, \(\psi = \pi\). Beyond this instant, however, when \(A'\) and \(B'\) begin to overlap,
one cannot construct common tangents to $A'$ and $B'$ as before, and the quantity $\psi$, in its current form, becomes ill-defined. As a consequence, (2.12) (and (8.3)) make sense and are valid only for the portion of the trajectories of $A'$ and $B'$ up to the instant of rendezvous, that is, $t \leq t_\psi$. In the next subsection, we demonstrate how $\psi$ can be defined from the instant when the overlap has commenced. This is important because it helps us to quantify the extent of overlap between $A'$ and $B'$, as is discussed subsequently.

### 8.2.3 Conditions for overlap between $A'$ and $B'$

When $A'$ overlaps with $B'$, $\psi$ can no longer be defined by common tangents as in Fig 8.2(a),(b). To demonstrate how to define $\psi$ in such a situation, we proceed in a progressively incremental fashion as follows. We first define $\psi$ when $A'$ is a point, and $B'$ is a circle. We then show how this can be extended to the case when $A'$ is a point and $B'$ has an arbitrary shape. We then consider the case when both $A'$ and $B'$ are circles, and finally extend this to the case when both $A'$ and $B'$ have arbitrary shapes. A proper understanding of the definition of $\psi$ in each of these cases is crucial to the development of the subsequent guidance laws.

**Case 1. $A'$ is a point and $B'$ is a circle**

Let $R$ represent the radius of $B'$. When $A'$ is outside $B'$, we have that $\sin \frac{\psi}{2} = \frac{R}{r}$, and $A'O = r$, as shown in Figure 8.4(a).

---

**Figure 8.4:** Definition of $\psi$ when $A'$ is a point and $B'$ is a circle. (a) $A'$ lies outside $B'$, (b) $A'$ lies inside $B'$.
When \( A' \) is inside \( B' \) (as in Figure 8.4b), we have \( r < R \), and the equation \( \sin \frac{\psi}{2} = \frac{R}{r} \) then makes sense only when \( \psi \) is a complex valued quantity. We therefore write \( \psi = a + ib \), and it can be shown that \( a = \pi \) and therefore \( \sin \frac{\psi}{2} = \cosh \frac{b}{2} \), (where \( b \) is a real-valued quantity). A geometric interpretation for the quantity \( b \) is obtained from hyperbolic geometry. We construct an imaginary rectangular hyperbola \( xy = \frac{r^2}{2} \), such that the vertex of this hyperbola is located at \( A' \), and the axis of this hyperbola is \( OA' \). This hyperbola thus moves and reorients itself at each time instant, according to the relative motion of \( A' \) and \( B' \).

Now, draw a tangent to the circle \( B' \) at the point \( C \) where \( B' \) intersects the line \( OA' \). The points of intersection of this tangent line with the hyperbola are called \( P \) and \( P' \), as shown in Fig 8.4(b). Define a hyperbolic sector \( POP' \) as the sector bounded by the straight lines \( OP, OP' \) and the hyperbolic arc \( P'P \). Then, \( b \) is defined as the hyperbolic angle that corresponds to the hyperbolic sector \( P'OP \). The instantaneous value of the hyperbolic angle \( b \) is related to the area of the hyperbolic sector \( POP' \) as follows:

\[
b = \frac{2}{r^2} \times \text{Area of hyperbolic sector } POP' \tag{8.13}
\]

Furthermore, we have the relation \( \cosh \frac{b}{2} = \frac{R}{r} \). Since the \( \cosh(\cdot) \) function is a monotonically increasing value of its argument, we can see that larger values of \( b \) correspond to a smaller \( r \) and thus a deeper penetration of the point \( A' \) inside the circle \( B' \). Thus the instantaneous value of \( \cosh \frac{b}{2} \) is related to the depth of penetration of \( A' \) inside \( B' \).

**Case 2. \( A' \) is a point and \( B' \) has an arbitrary shape**

When \( A' \) lies outside \( B' \), the angle \( \psi \) is defined as shown in Figure 8.5(a). In this figure, \( \psi < \pi \), but a corresponding figure can be drawn for the case when the shape of \( B' \) is such that \( \psi > \pi \). Then, as \( A' \) approaches \( B' \), the value of \( \psi \) approaches \( \pi \), and when \( A' \) makes contact with \( B' \) at the boundary, \( \psi = \pi \). We now construct an imaginary circle \( \mathcal{X} \) such that \( A' \) is inside \( B' \) if and only if \( A' \) is inside \( \mathcal{X} \). To construct this circle, draw a line...
through $A'$ that is normal to $B'$ at the point where this line intersects $B'$. In Figure 8.5(b), this line is $A'C_1$ which is normal to $B'$ at the point $C_1$. Extend this line so that it passes through the opposite end of $B'$, which is $C_2$ in Figure 8.5(b). Then, $\mathcal{X}$ is the circle that has $C_1C_2$ as a diameter.

Thus, when $A'$ lies inside $B'$, it also lies inside the circle $\mathcal{X}$. During this phase, as in Case 1, $\psi = \pi + ib$, and $\sin \frac{\psi}{2} = \cosh \frac{b}{2} = \frac{R}{r}$, with $R = \frac{C_1C_2}{2}$. Note that due to the relative motion of $A'$ and $B'$, the normal can keep changing, and this, in turn, can change the value of $C_1C_2$, and thus, $R$. In other words, $R = R(t)$ in this case. Then $b$ is related to the area of the hyperbolic sector $POP'$, according to (8.13). Again, larger values of $b$ correspond to a deeper penetration of $A'$ inside $B'$, with this penetration being defined along the normal $C_1C_2$.

We point out that there could be some scenarios where there can be more than one line passing through $A$ that is normal to $B'$. In such cases, one can choose any of these normals to define the circle $\mathcal{X}$. For example, the chosen normal may be the same as the one when $A'$ first enters $B'$.

![Figure 8.5: Definition of $\psi$ when $A'$ is a point and $B'$ is an arbitrary object. (a) $A'$ lies outside $B'$, (b) $A'$ lies inside $B'$](image)

**Case 3. $A'$ and $B'$ are both circles**
Consider two circles $A'$ and $B'$ of radii $R_1$ and $R_2$, and with centers $O$ and $O'$, respectively. Let $r$ denote the distance between the centers. When $A'$ and $B'$ do not overlap, the angle $\psi$ is defined as shown in Figure 8.6(a). When $A'$ overlaps with $B'$, we again have $\psi = a + ib$, and $\sin \psi = \cosh \frac{b}{2}$. The geometric interpretation of $b$ is then obtained as follows. We find the point $C$ on the line $OO'$, such that the distance $OC = \frac{R_1 r}{R_1 + R_2}$. Then, $\frac{R_1}{OC} = \frac{R_2}{r - OC}$.

Construct two imaginary hyperbolas (one corresponding to $A'$ and another corresponding to $B'$), both with vertex at $C$, and with a common axis $OO'$. These imaginary hyperbolas move according to the relative motion of $A'$ and $B'$, while their vertices and axes remain coincident.

Draw two lines $PP'$ and $P_1P'$, where $PP'$ is normal to $OO'$, and tangent to $B'$, while $P_1P'$ is normal to $OO'$, and tangent to $A'$. Then, along the lines of (8.13), we see that $b$ is
related to the areas of the hyperbolic sectors $POP'$ and $P_1OP_1'$ as follows:

\[ b = \frac{2}{OC^2} \times \text{Area of hyperbolic sector } POP' = \frac{2}{O'C^2} \times \text{Area of hyperbolic sector } P_1O'P_1 \]

and $\cosh \frac{b}{2}$ satisfies the equations $\cosh \frac{b}{2} = \frac{R_1}{OC} = \frac{R_2}{r - OC}$. As before, larger values of $b$ correspond to smaller values of $OC$ and thus deeper penetration of $A'$ into $B'$.

**Case 4. $A'$ and $B'$ are both arbitrary shapes**

When $A'$ and $B'$ do not overlap, the angle $\psi$ is defined as shown in Figure 8.7(a). In this figure, $\psi < \pi$, but a corresponding figure can be drawn for the case when the shape of $B'$ is such that $\psi > \pi$. Then, as $A'$ approaches $B'$, the value of $\psi$ approaches $\pi$, and when $A'$ and $B'$ make contact at their boundaries, $\psi = \pi$. We now construct two imaginary circle $\mathcal{X}_1$ and $\mathcal{X}_2$ such that $A'$ overlaps with $B'$ if and only if $\mathcal{X}_1$ overlaps with $\mathcal{X}_2$. To construct these circles, we proceed as follows. Refer Fig 8.7(b). Assume that $A'$ and $B'$ intersect at two points. Let these points of intersection be $E_1$ and $E_2$. Draw normals to $A'$ at each of these intersection points, and then extend these normals till they meet at a point. Let $O$ represent the point of intersection of these normals. Then, construct the angular bisector of the lines $OE_1$ and $OE_2$, and let the point at which this angular bisector intersects $A'$ be $C$. Extend the line $OC$ in both directions. Let the other point of intersection of $OC$ with $A'$ be $C_1$, and let the points of intersection of $OC$ with $B'$ be $C_2$ and $C_3$. Then, construct two circles: one with $CC_1$ as diameter (this circle of radius $R_1 = CC_1/2$ is $\mathcal{X}_1$), and another with $C_2C_3$ as diameter (this circle of radius $R_2 = C_2C_3/2$ is $\mathcal{X}_2$). Then, when $A'$ overlaps with $B'$, we have that $\mathcal{X}_1$ overlaps with $\mathcal{X}_2$, and therefore, $\psi = \pi + ib$, and $\sin \frac{\psi}{2} = \cosh \frac{b}{2}$.

Since $\mathcal{X}_1$ and $\mathcal{X}_2$ have been constructed as above, the corresponding hyperbolas can be constructed as in Case 3, and the quantity $b$ similarly defined. As before, larger values of $b$ correspond to increased penetration of $A'$ inside $B'$.

We point out that the above construction will ensure that $\psi$ varies continuously as it changes from a real-valued quantity (when $A'$ is outside $B'$), to a value $\pi$ (when $A'$ just
touches $B'$), to a complex valued quantity (when $A'$ has begun to overlap with $B'$). Note that due to relative motion of $A'$ and $B'$, the radii of the two circles will, in general, change with time, that is $R_1 = R_1(t)$, and $R_2 = R_2(t)$.

Figure 8.7: Definition of $\psi$ when $A'$ and $B'$ are both arbitrary objects. (a) $A'$ lies outside $B'$, (b) $A'$ lies inside $B'$

When $A'$ and $B'$ have overlapped, the quantity $y_1$ cannot be computed using (8.3) since the quantities $\psi_1$ and $\psi_2$ are undefined after the overlap commences. However, $y_1$ can be computed using (8.3), after replacing $\sin \frac{\psi}{2}$ with $\cosh \frac{b}{2}$, and this leads to the equation:

$$y_1 = \frac{V_\theta^2 \cos^2 (\dot{\theta} - \theta) - 2V_\theta V_r \cos(\dot{\theta} - \theta) \sin (\dot{\theta} - \theta) + V_r^2 \sin^2 (\dot{\theta} - \theta)}{V_\theta^2 + V_r^2} - \cosh^2 \left( \frac{b}{2} \right), t \geq t_\psi$$

(8.15)
Recalling that $t_\psi$ represents the time at which $A'$ and $B'$ have made contact at their boundaries, (8.3) is used to compute $y_1$ for $t < t_\psi$, while (8.15) is used for $t > t_\psi$. At the instant when $t = t_\psi$, we have $\sin\left(\frac{\psi}{2}\right) = \cosh\left(\frac{b}{2}\right) = 1$, and (8.3) and (8.15) are both identical. Also, when $A'$ and $B'$ have overlapped, then $y_1$ is always negative, and can be used to quantify the depth of overlap between $A'$ and $B'$, as discussed subsequently.

When $A'$ is fully inside $B'$, we construct the circles $X_1$ and $X_2$ as follows. Consider the instant when $A'$ is just inside $B'$ and touches $B'$ at a single point. At that instant there is a single normal (say $\hat{n}$) passing through that point of intersection. Beyond this time instant, as $A'$ penetrates deeper inside $B'$, we continue to use $\hat{n}$ for constructing $X_1$ and $X_2$.

When $A'$ and $B'$ have overlapped, the rendezvous cone is defined in physical space as follows:

$$\mathcal{R}_\alpha(t) = \{ \alpha : C_2 < y_1(t) < C_1 \} \quad (8.16)$$

where, $C_1 \leq 0$, and $C_2 < 0$ represent (user-defined) upper and lower bounds on the desired depth of overlap. From a practical standpoint, defining such upper and lower bounds on the depth of overlap makes sense for the following reasons. The depth of overlap needs to be large enough to ensure that the overlap remains robust, that is, small wind gusts acting on the vehicles should not cause the overlap to be lost. When used in communication applications (Fig 1.3(b)), the depth of overlap needs to be large enough to ensure that there is enough space for a relay UAV to position itself inside the overlap region. At the same time, however, the depth of overlap should not be too excessive because that can reduce the geographical area that can be swept by the two footprints. Keeping these two objectives in mind, upper and lower bounds $C_1$ and $C_2$ for the depth of overlap can be appropriately defined.

Equations (8.3) and (8.15) can also be written in terms of the relative velocity components of the angular bisector, as follows: Substituting $\theta = \hat{\theta}$, $V_\theta = \hat{V}_\theta$, $V_r = \hat{V}_r$ in (8.3),
these equations become:

\[
y_1 = \begin{cases} 
\frac{V_r^2}{V_{\theta}^2 + V_r^2} - \sin^2(\frac{\psi}{2}), & t \leq t_\psi \\
\frac{V_r^2}{V_{\theta}^2 + V_r^2} - \cosh^2(\frac{b}{2}), & t > t_\psi 
\end{cases}
\]  

(8.17)

Representation of \(y_1\) in the form of (8.17) makes it amenable for an analysis of the rendezvous (as discussed in Sections 8.3.1 and 8.3.2, while the representation of \(y_1\) using (8.3) and (8.15) makes it amenable for the design of guidance laws that enable this rendezvous (as discussed in Section 8.3.3).

8.3 Guidance Law for Rendezvous of Arbitrary Shapes

In this section, we propose guidance laws that enable the footprint \(A'\) to rendezvous with the footprint \(B'\), while simultaneously ensuring collision avoidance between the vehicles \(A\) and \(B\).

8.3.1 Requirements for Rendezvous and Collision Avoidance

From the discussion in the previous section, we see that in order to achieve overlap of the arbitrary shapes \(A'\) and \(B'\), while also ensuring that \(A\) and \(B\) do not collide, we need the following:

(a) When \(\psi\) is real and \(\psi \leq \pi\): For rendezvous to occur, we require \(y_1 < 0\), \(\hat{V}_r < 0\), and for collision avoidance we require either (i) \(y_2 > 0\), or (ii) \(y_2 < 0\), \(V_r > 0\).

(b) When \(\psi\) is real and \(\psi > \pi\): For rendezvous to occur, we require either (i) \(\hat{V}_r < 0\), or (ii) \(y_1 > 0\), \(\hat{V}_r > 0\), and for collision avoidance we require either (i) \(y_2 > 0\), or (ii) \(y_2 < 0\), \(V_r > 0\).

(c) When \(\psi\) is complex: \(A'\) and \(B'\) have already made contact. We then control the depth of overlap by manipulating the (negative) value of \(y_1\), while satisfying the collision avoidance condition of either (i) \(y_2 > 0\), or (ii) \(y_2 < 0\), \(V_r > 0\).

In addition to \(t_\psi\), we define two more times \(t_{m1}\) and \(t_{m2}\), where \(t_{m1}\) is the time instant at which \(\hat{V}_r = 0\) (which is the time at which \(A'\) and \(B'\) are at their point of closest approach), while \(t_{m2}\) is the time instant at which \(V_r = 0\) (which is the time at which the centers of the two sensors are closest to each other).
At time $t = t_{m1}$, let the corresponding values of $\psi$ and $b$ be denoted by $\psi_{m1}$ and $b_{m1}$.

From (8.17), we can see that $y_1(t_{m1})$ is as follows:

$$y_1(t_{m1}) = \begin{cases} \cos^2 \left( \frac{\psi_{m1}}{2} \right), & t_{m1} \leq t_\psi, \\ - \sinh^2 \left( \frac{b_{m1}}{2} \right), & t_{m1} > t_\psi \end{cases} \quad (8.18)$$

Now, $y_1(t_{m1})$ can be interpreted as a “miss-angle” function. If the value of $y_1(t_{m1})$ is positive, then $A'$ and $B'$ have not overlapped, and the miss-angle $\psi_{m1}$ is a real-valued quantity, defined as the angle between the common tangents to $A'$ and $B'$ at $t = t_{m1}$. However, if $y_1(t_{m1})$ is negative, then $A'$ and $B'$ have overlapped, and $\psi_{m1}$ is a complex-valued quantity, which can be interpreted using the hyperbolic geometry construction discussed in Section 8.2.C.

Now, assume that we define a quantity $c$ such that:

$$c = \begin{cases} \cos^2 \left( \frac{\psi_{m1}}{2} \right), & t_{m1} \leq t_\psi, \\ - \sinh^2 \left( \frac{b_{m1}}{2} \right), & t_{m1} > t_\psi \end{cases} \quad (8.19)$$

Figure 8.8 shows the plot of $\psi_{m1}$ as a function of $c$, for $0 < c < 1$, and also $b_{m1}$ as a function of $c$, for $c < 0$. It indicates that for $0 < c < 1$ at the instant when $\hat{V}_r = 0$, the two footprints have not overlapped, while if $-\infty < c < 0$ at the instant when $\hat{V}_r = 0$, then the two footprints have overlapped.
Next, at time $t = t_{m2}$, let the corresponding value of $r$ be denoted by $r_{m2}$. Then, from (8.10), we can see that $y_2(t_{m2})$ is as follows:

$$y_2(t_{m2}) = \frac{(R_A + R_B)^2 - r_{m2}^2}{r_{m2}^2} \quad (8.20)$$

From (8.20), we see that $y_2(t_{m2})$ can be interpreted as another miss-angle function: If $y_2(t_{m2})$ is positive, then $A$ and $B$ have not collided, and the angle $\cos^{-1}\left(\frac{(R_A + R_B)^2 - r_{m2}^2}{r_{m2}^2}\right)$, defined as the miss-angle, is a real-valued quantity and represents the angle subtended by the common tangents to the two circles $A$ and $B$. On the other hand, if at the instant of closest approach, $y_2(t_{m2})$ is negative, then $A$ and $B$ have collided, and the corresponding miss-angle is a complex-valued quantity.

### 8.3.2 Partitioning of the $(r, \theta)$ space into sectors

In this subsection, we partition the physical 2-D space into sectors depending on the signs of $y_1$, $y_2$, and $\dot{V}_r$. This partitioning will facilitate design of guidance laws in the next subsection. To perform this partitioning, we first determine the extrema of $y_1$. From (8.17), we can determine $\frac{\partial y_1}{\partial \alpha}$ as follows:

$$\frac{\partial y_1}{\partial \alpha} = \frac{2\dot{V}_r \dot{V}_\theta (\dot{V}_r \frac{\partial \dot{V}_r}{\partial \alpha} - \dot{V}_\theta \frac{\partial \dot{V}_\theta}{\partial \alpha})}{(\dot{V}_\theta^2 + \dot{V}_r^2)^2} \quad (8.21)$$
and from the above equation, we see that candidates for the extrema of $y_1$ correspond to those values of $\alpha$ that correspond to $\hat{V}_r = 0$, $\hat{V}_\theta = 0$, and $\hat{\omega}_{V_r} = -\tan(\alpha - \theta)$. It can be further shown (from the sign of $\frac{\partial^2 y_1}{\partial \alpha^2}$ computed at each of these extrema candidates), that $y_1$ has a maxima at $\hat{V}_r = 0$, and a minima at $\hat{V}_\theta = 0$.

A typical plot of $y_1$, $\hat{V}_r$ and $\hat{V}_\theta$ vs. $\alpha$ is shown in Figure 8.9. As seen in this figure, $y_1$ attains two local minima at $\hat{V}_\theta = 0$, one of which lies in the $\hat{V}_r < 0$ region, while the other lies in the $\hat{V}_r > 0$ region. Similarly, $y_1$ attains two local maxima at $\hat{V}_r = 0$, one of which lies in the $\hat{V}_\theta < 0$ region, while the other lies in the $\hat{V}_\theta > 0$ region. The local minima physically correspond to the scenarios when the relative velocity vector is aligned with the angular bisector of the $y_1 < 0$ conical sector, while the local maxima correspond to the scenarios when the relative velocity vector lies on the $\hat{V}_r = 0$ line.

Based on the above, we can do a partitioning of the $(r, \theta)$ plane into different sectors based on the signs of $y_1$ and $\hat{V}_r$, as shown in Figure 8.10. The $\hat{V}_r = 0$ line partitions the physical space into two half-spaces. Each of these half-spaces, in turn, can be partitioned into $y_1 < 0$ and $y_1 > 0$. The physical $(r, \theta)$ space thus comprises four sectors as follows:
Sector 1: \( y_1 < 0, \dot{V}_r < 0, \)
Sector 2: \( y_1 < 0, \dot{V}_r \geq 0, \)
Sector 3: \( y_1 \geq 0, \dot{V}_r < 0, \)
Sector 4: \( y_1 \geq 0, \dot{V}_r \geq 0. \)

The above four sectors thus constitute a partitioning of the physical space based on the rendezvous cone. Each of these four sectors is further partitioned into two sub-sectors based on the collision cone, and these sub-sectors are appropriately obtained from the following: 
(i) \( y_2 < 0, V_r < 0, \) (ii) \( y_2 < 0, V_r \geq 0, \) (iii) \( y_2 \geq 0, V_r < 0, \) (iv) \( y_2 \geq 0, V_r \geq 0. \)

Figure 8.10: Partitioning of 2-D space based on the relative signs of \( y_1, y_2, V_{rb} \)

The objective is to apply the appropriate acceleration to maneuver the relative velocity vector from any initial condition on the plane, into the sector represented by \( y_1 < 0, \dot{V}_r < 0, \) while avoiding the sector \( y_2 < 0, V_r < 0. \) In other words, the heading angle of the velocity vector of \( A \) should be driven into the set defined by \( \{ \alpha : (y_1 < 0 \text{ and } \dot{V}_r < 0) / (y_2 < 0 \text{ and } V_r < 0) \}. \) After the relative velocity vector is driven into this set, it is then further maneuvered inside this set to make \( A' \) overlap with \( B' \) to a desired penetration depth. Thus, for example, if the initial relative velocity vector is such that it lies in the \( y_1 < 0, \dot{V}_r > 0 \) region, then, in order to drive it into the \( y_1 < 0, \dot{V}_r < 0 \) region as shown in Figure
The qualitative nature of \( y_1 \) will be as illustrated in Figure 8.11(b). A description of this qualitative nature of \( y_1 \) is provided below.

Note that the \( y_1 < 0 \) sector comprises two halves: one wherein \( \partial y_1 / \partial \alpha < 0 \) (that is, an increase in \( \alpha \) causes a decrease in \( y_1 \)) and another half where \( \partial y_1 / \partial \alpha > 0 \) (that is, an increase in \( \alpha \) causes an increase in \( y_1 \)). The dashed line \( ZZ' \) in Fig 8.11(a) represents the angular bisector that separates these two half-cones, and \( y_1 \) attains its minimum when the relative velocity vector aligns with \( ZZ' \). For time \( t \in [0, t_1] \), the relative velocity vector is first driven deeper into the \( y_1 < 0, \dot{V}_r > 0 \) region, till it aligns with the line \( OZ \) in Fig 8.11(a). Thus, for \( t \in [0, t_1] \), the value of \( y_1 \) decreases, till it reaches a minimum at \( t = t_1 \) as seen in Fig 8.11(b).

For \( t \in (t_1, t_2] \), the relative velocity vector is driven to align with \( OX \), and during this time, \( y_1 \) increases to zero, so that \( y_1(t_2) = 0 \). For \( t \in (t_2, t_3] \), the relative velocity vector is driven to align with \( OU' \) (which is the \( \dot{V}_r = 0 \) line), and during this time \( y_1 \) increases. At \( t = t_3 \), the relative velocity vector is aligned with \( OU' \), and at that instant, \( y_1 \) attains its maximum value. For \( t \in (t_3, t_4] \), the relative velocity vector is driven to align with \( OY' \), and \( y_1 \) is correspondingly driven towards zero, so that \( y_1(t_4) = 0 \). For \( t \in (t_4, t_5] \), the relative velocity vector is driven into the sector bounded by the lines \( OZ' \) and \( OY' \), during which time \( y_1 \) decreases to its negative reference value \( c_1 \), where \( c_1 \in [-1, 0) \). When \( y_1 = -c_1 \in [-1, 0) \) is attained, \( y_1 \) is kept constant at this reference till time \( t = t_\psi \), which is when \( A' \) and \( B' \) touch each other at their boundaries. For \( t > t_\psi \), \( y_1 \) is driven to a new reference \( c_1 < -1 \), with the appropriate value of \( c_1 \) being chosen so as to correspond to the desired depth of overlap between \( A' \) and \( B' \). In the next subsection, we determine analytical guidance laws that facilitate rendezvous through such a variation of \( y_1 \).
8.3.3 Guidance Law Derivation

In this section, we determine a guidance law that will achieve rendezvous of \( A' \) and \( B' \), while ensuring collision avoidance between \( A \) and \( B \). We define two reference values \( c_1 < 0 \) (where \( c_1 \) represents the desired, possibly time-varying, depth of penetration of \( A' \) and \( B' \)) and \( c_2 > 0 \) (where \( c_2 \) represents the desired miss angle function between \( A \) and \( B \)). Two options are possible:

(a) Use a guidance law to achieve \( y_1 = c_1 \), while ensuring that \( y_2(t_{m2}) > 0 \) is maintained,

(b) Use a guidance law to achieve \( y_2 = c_2 \), while ensuring that \( y_1(t_{m1}) < 0 \) is maintained.

In (a), it is required that the value of \( c_1 \) be chosen appropriately such that \( y_2(t_{m2}) > 0 \) is feasible, while in (b), it is required that the value of \( c_2 \) be chosen such that \( y_1(t_{m1}) < 0 \) is feasible.

The state equations governing the relative dynamics of \( A \) (\( A' \)) and \( B \) (\( B' \)) are as given in [2.13]. Here, \( r \) represents the distance between the centers of \( A \) and \( B \). Associated with these state equations, are the two output functions \( y_1 \) (as defined in [8.3] and [8.15]) and \( y_2 \), as defined in [8.10].
For the guidance law (a), choose a Lyapunov function $L_1 = \frac{1}{2}(y_1 - c_1)^2$. We then need a law for $a_A$ such that $\dot{L}_1$ is negative definite. We see that $\dot{L}_1 = (y_1 - c_1)(\dot{y}_1 - \dot{c}_1)$, where $\dot{y}_1$ is obtained by taking the derivative of (2.12) along the trajectories of the system (2.13), as follows:

$$\dot{y}_1 = \frac{F_1 a_A + F_2}{(V_r^2 + V_\theta^2)^2}$$  \hspace{1cm} (8.22)

and for $t \leq t_\psi$, $F_1$ and $F_2$ are given by:

$$F_1 = (V_r^2 + V_\theta^2)\{y_1 [2V_r \cos (\delta - \theta) + 2V_\theta \sin (\delta - \theta)]$$
$$+ \sin (\psi_1 - \psi_2) [V_\theta \cos (\delta - \theta) + V_r \sin (\delta - \theta)]$$
$$+ 2V_r \sin \psi_1 \sin \psi_2 \cos (\delta - \theta) - 2V_\theta \cos \psi_1 \cos \psi_2 \sin (\delta - \theta)\}$$  \hspace{1cm} (8.23)

$$F_2 = -(V_r^2 + V_\theta^2) \left[ \frac{V_\theta \sin (\psi_1 - \psi_2)(V_r^2 - V_\theta^2) + 2V_r V_\theta^2 \cos (\psi_1 - \psi_2)}{r} \right]$$
$$+ \dot{\psi}_1 [V_r^2 \cos \psi_1 \sin \psi_2 + V_\theta^2 \sin \psi_1 \cos \psi_2] + \dot{\psi}_2 [V_r^2 \sin \psi_1 \cos \psi_2 + V_\theta^2 \cos \psi_1 \sin \psi_2]$$
$$+ (\dot{\psi}_1 - \dot{\psi}_2)V_r V_\theta \cos (\psi_1 - \psi_2)$$ \hspace{1cm} (8.24)

If we choose

$$a_A = \frac{[-K_1(y_1 - c_1) + \dot{c}_1](V_r^2 + V_\theta^2)^2 - F_2}{F_1}$$ \hspace{1cm} (8.26)

with $K_1 > 0$, then this leads to $\dot{L}_1 = -K_1(y_1 - c_1)^2$, which is negative definite for $y_1 \neq c_1$. This ensures that $L_1$ goes to zero, and $y_1$ reaches a value of $c_1$.

The above law would enable $A'$ to be on a path to rendezvous with $B'$, until the time $t = t_\psi$, which is when $\psi = \pi$ occurs. For $t > t_\psi$, we cannot use (8.24) and (8.25) because $\psi_1$ and $\psi_2$ are undefined, and we recall that during this time, we use (8.15) to determine $y_1$. By taking the derivative of (8.15) along the trajectories of the system (2.13), it can be seen that $\dot{y}_1$ has a structure similar to that in (8.22) with $F_1$ and $F_2$ in (8.22) now replaced by the following equations, which use the hyperbolic angle $b$ in lieu of $\psi$:

$$F_1 = \frac{f_1 a_A + f_2}{(V_r^2 + V_\theta^2)^2}$$ \hspace{1cm} (8.25)
Thus, for \( t \geq t_\psi \), we continue to use (8.26) as the guidance law, but with \( F_1 \) and \( F_2 \) as in (8.27) and (8.28).

The guidance law of (8.26) works when the initial relative velocity vector is located inside the \( y_1 < 0, \hat{V}_r < 0 \) sector (see Fig. 8.10), and it drives \( y_1 \) monotonically to the value \( c_1 \). However, if the initial relative velocity vector is located in any other sector (such as in Fig. 8.11(a)), it needs to be first driven into the \( y_1 < 0, \hat{V}_r < 0 \) region, before (8.26) can be applied. To achieve this, we use the following law, similar to (8.26), but now with a switching gain \( K_1 \):

\[
a_A = \frac{-K_1 y_1 (V_r^2 + V_\theta^2) - F_2}{F_1}
\]

where, \( K_1 \) is obtained as follows:

\[
K_1 = \begin{cases} 
K_0 \operatorname{sgn} \left( \frac{\partial y_1}{\partial \alpha} \right), & \text{if } y_1(t) < 0, \hat{V}_r(t) > 0 \\
-K_0, & \text{if } y_1(t) > 0, \hat{V}_r(t) > 0 \\
K_0, & \text{if } y_1(t) > 0, \hat{V}_r(t) < 0
\end{cases}
\]

with \( K_0 > 0 \). Consider Figure 8.11(a),(b) for the purpose of illustration, with the relative velocity vector \( \vec{V} \) initially in the \( X'OZ \) sector. Following (8.30), \( K_1 \) is initially negative thus causing \( y_1(t) \) to become more negative as the relative velocity vector is driven deeper inside the \( y_1 < 0 \) region. When \( \vec{V} \) enters the \( ZOX \) sector, \( \frac{\partial y_1}{\partial \alpha} > 0 \), and from (8.30), \( K_1 \) becomes positive causing \( y_1(t) \) to be driven toward zero and \( \vec{V} \) aligns with the \( OX \) line.
When in the $XOU'$ sector, $K_1$ switches to a negative value causing $y_1(t)$ to become positive and this continues till $\vec{V}$ crosses into the $\hat{V}_r < 0$ region, at which time $K_1$ is made positive again, thereby driving $y_1(t)$ to zero and causing the velocity vector to align with the $y = 0$, $\hat{V}_r < 0$ line (which is $OY'$ in Fig 8.11(a)). Subsequently, (8.26) is used to drive $y_1$ to $c_1$, with the reference value $c_1$ chosen such that $c_1 \in [-1, 0)$. For $t > t_\psi$, the reference value $c_1$ is allowed to be less than $-1$, with the chosen value depending on the required depth of penetration between $A'$ and $B'$. Such a state-based switching guidance law will lead to the overall trajectory of $y_1(t)$ as shown in Figure 8.11(b), and described by the following equation:

$$y_1(t) = \begin{cases} 
  y(0)e^{-K_1t}, & 0 \leq t \leq t_2 \\
  \epsilon e^{K_0(t-t_2)}, & t_2 \leq t \leq t_3 \\
  y_1(t_3)e^{-K_0(t-t_3)}, & t_3 \leq t \leq t_4 \\
  -\epsilon e^{K_0(t-t_4)}, & t_4 < t \leq t_5 \\
  y_1(t_5), & t_5 < t \leq t_\psi \\
  y_1(t_5)e^{K_0(t-t_\psi)}, & t > t_\psi 
\end{cases}$$

(8.31)

In the above equation, $\epsilon > 0$ is an arbitrarily small number that is used to enable the relative velocity vector cross the $y_1 = 0$ line. Thus, if the complete rendezvous (including the overlap) needs to occur within a given time $t = T$, we can use (8.31) to determine a lower bound on $K_0$ so as to ensure that the overlap is indeed achieved in that time.

For the guidance law (b), choose a Lyapunov function $L_2 = \frac{1}{2}(y_2 - c_2)^2$. We then need a guidance law for $a_A$ such that $\dot{L}_2$ is negative definite. We can see that $\dot{L}_2 = (y_2 - c_2)\dot{y}_2$, and $\dot{y}_2$ has the following form:

$$\dot{y}_2 = \frac{F_3a_A}{r^2(V_r^2 + V_\theta^2)^2}$$

(8.32)
where, $F_3$ is given by:

$$F_3 = r^2[2V_r \sin (\delta - \theta) (R_A + R_B)^2 - 2V_\theta \cos (\delta - \theta) ((R_A + R_B)^2 - r^2)](V_r^2 + V_\theta^2)$$

$$+ 2r^2[V_\theta \cos (\delta - \theta) + V_r \sin (\delta - \theta)][V_\theta^2((R_A + R_B)^2 - r^2) - V_r^2(R_A + R_B)^2]$$

(8.33)

If we choose

$$a_A = \frac{-K_2(y_2 - c_2)r^2(V_r^2 + V_\theta^2)^2}{F_3}$$

(8.34)

with $K_2 > 0$, then this leads to $\dot{L}_2 = -K_2(y_2 - c_2)^2$, which is negative definite for $y_2 \neq c_2$. This ensures that $L_2$ goes to zero, and $y_2$ reaches a value of $c_2$. Appropriate choice of the gain $K_2$ will ensure that $y_2$ reaches $c_2$ before $V_r = 0$ occurs.

The rendezvous and collision avoidance guidance laws are then applied as follows:

If $(y_2(t) < 0$ and $V_r(t) < 0$ and $r(t) < threshold$)

Use collision avoidance law (8.34) to achieve $y_2 = c_2$,

else

while $\psi$ is real,

Use rendezvous guidance law (8.29), (8.26) with $F_1$, $F_2$ from (8.24), (8.25)

to achieve $y_1 = c_1$, with $c_1 \in [-1, 0]$

while $\psi$ is complex,

Use rendezvous guidance law (8.26) with $F_1$, $F_2$ from (8.27), (8.28)

to achieve $y_1 = c_1$, with $c_1 < 0$

8.4 Simulation Results

In this section, we present simulation results that demonstrate the working of the guidance laws discussed above. We consider a scenario wherein multiple sensors (denoted as $F_1$, $F_2$, and $F_3$) need to move in such a way as to each achieve overlaps between their individual footprints and that of another sensor, denoted as $L$. The footprints of each of the sensors are shown in Figure 8.12 from which it is evident that the footprints are all non-convex.
It is assumed that $F_1$, $F_2$ and $F_3$ all have footprints of the same size, while $L$ has a footprint of larger size. The variable $\alpha$ represents the heading angle of the sensor. Fig 8.12 shows that the shape of the footprint is not omni-directional, and furthermore, as the heading angle $\alpha$ of each sensor changes, its footprint changes orientation as well, and this changes the relative geometry between the sensor $F_i$ and sensor $L$. We consider three examples as follows.

8.4.1 Example 1: Stationary Leader

The initial locations of $F_1$, $F_2$, and $F_3$ are taken as (5, 8), (38, 5), and (22, 4), respectively. All coordinates are given in meters. Their initial speeds are 1 m/sec, while their initial heading angles are $135^0$, $50^0$, and $210^0$. The $L$ sensor is assumed to be stationary and is located at (20, 15). The initial locations and heading angles are shown in Fig 8.13 from which it is evident that if $F_1$, $F_2$ and $F_3$ simply followed straight line trajectories with their initial heading angles, rendezvous with $L$ would not have occurred for either of them. The spatial trajectories of each of $F_1$, $F_2$, and $F_3$ as the rendezvous guidance law causes them to rendezvous with $L$, and achieve an overlap with $L$, are also shown in Fig 8.13.

The trajectories of the different parameters in the guidance law are shown in Figs 8.14-8.15. Figure 8.14 shows the angle $\psi_i$ for each of the footprints $F_1$, $F_2$, and $F_3$ to footprint $L$. Fig 8.14(a) demonstrates the real part of $\psi$, which occurs during the initial portion of the engagement when $F_1$, $F_2$ and $F_3$ are each approaching $L$. When the respective footprint has just made contact with $L$, $\psi = \pi$, after which time, $\psi$ becomes complex and attains the
form $\psi = \pi + jb$. It is thus evident that $F_1$, $F_2$ and $F_3$ make their first contact with $L$ at times $t = 12$ sec, $t = 15$ sec, and $t = 10$ sec, respectively. From these times onward, the real part of $\psi_i$ remains a constant equal to $\pi$, while the imaginary part of $\psi_i$ changes with time. When $\psi_i$ is real, the guidance law (8.29) is used in conjunction with (8.24) and (8.25), and when $\psi_i$ becomes complex, the guidance law (8.29) is used in conjunction with (8.27) and (8.28).

Figure 8.15(a) shows the time histories of $y_1$ for each of the sensors $F_1$, $F_2$ and $F_3$ to $L$. The initial value of $y_1$ for each of the sensors is positive, thus indicating that the initial velocity vectors for each of the sensors are outside their individual rendezvous cones to $L$. Under the guidance law, $y_1$ is driven from the positive initial values to a negative value which, in this example, is taken as $c_1 = -0.4$. Finally, Fig 8.15(b) shows the time histories of the angle $\alpha_i$, which is obtained using the equation $\dot{\alpha}_i = \frac{a_i}{V_i}$. In this example, the angle $\delta_i$ at which the acceleration is applied, is chosen to be normal to the velocity vector, that is, $\delta_i = \alpha_i + \frac{\pi}{2}$.
8.4.2 Example 2: Moving Leader

In this example, we consider the rendezvous of two footprints $F_1$ and $F_2$, with a moving leader $L$. The initial locations, speeds, and heading angles of $F_1$ and $F_2$ are identical to those in Example 1. The leader $L$ is now moving with a speed of 0.3 m/sec and a heading angle of $90^\circ$. Fig 8.16 shows the trajectories of all 3 sensors, and it is evident that both
sensors $F_1$ and $F_2$ achieve rendezvous with the moving sensor $L$, to a requisite depth of penetration.

Figure 8.16: Rendezvous of two footprints with a moving leader footprint (Example 2)

8.4.3 Example 3: Sweep Pattern

In this example, we consider the rendezvous of $F_1$ and $F_2$ with $L$, as $L$ traces a sinusoidal sweep pattern on the plane. It is desired that as $L$ traces the sweep, $F_1$ and $F_2$ remain overlapped with $L$. To enable this, the guidance laws need to not just achieve an overlap between $F_i$ and $L$, but they also need to do it in such a way that by the time the overlap is achieved, the velocity of $F_i$ matches that of $L$. This is achieved by a blending of the rendezvous guidance law with a velocity consensus law. In other words, the acceleration of each sensor is a weighted combination of the rendezvous acceleration and the difference between the velocity of $F_i$ and $L$.

The initial locations of $F_1$ and $F_2$ are $(2, -5)$ and $(18, 5)$, respectively. Fig 8.17 shows that $F_1$ and $F_2$ first achieve rendezvous with $L$, following which, as $L$ traces a sinusoidal sweep, $F_i$ do the same as well, under the influence of $L$. Fig 8.18(a) shows the time history of $\psi$, from which it is evident that $F_2$ first made contact with $L$ at $t = 22$ sec, which is the time at which $\psi$ starts to become complex. Fig 8.18(b) shows the time history of $y_1$ for $F_2$.
to $L$. It is evident that $y_1$ changes from its initial positive value to a negative value at the time at which $F_2$ just made contact with $L$. However, at the time this initial contact was made, there was significant relative velocity between $F_i$ and $L$, as is evident from Fig 8.19 which shows that the relative velocity components $V_r$ and $V_\theta$ between $F_2$ and $L$ were both non-zero at $t = 22$ sec.

Fig 8.18(c) shows the trajectory of the heading angle $\alpha$ of $F_2$. It is clear that at $t = 22$ sec, when contact was first made, $\alpha$ was around $108^0$, which indicates that the heading angle of $F_2$ does not match the heading angle of $L$. By subsequently using a velocity consensus law in tandem with the rendezvous guidance law, $F_2$ achieves a heading angle $\alpha = 90^0$, as evident in Fig 8.18(c). As seen in Fig 8.19, $V_\theta = 0$, $V_r = 0$ is achieved from $t = 27$ sec onwards. Furthermore, while the velocity consensus was being achieved, $y_1$ remained negative as evident from Fig 8.18(b), and this ensured that $F_1$ and $F_2$ remained overlapped with $L$ during this phase. $L$ then performs a sinusoidal sweep from $t = 28$ sec onwards, and $F_1$ and $F_2$ remain overlapped with $L$ during this sweep pattern, as is clearly evident from Fig 8.17.
Figure 8.17: Rendezvous of two footprints with a moving leader footprint, followed by a sweep (Example 3)
Figure 8.18: Time history of (a), $\psi$, (b) $y$, and (c) $\alpha$ between $F_2$ and $L$

Figure 8.19: Time history of $V_\theta$ and $V_r$ between $F_2$ and $L
8.5 Further Extensions

This chapter provides a core solution to the problem of getting two or more footprints to overlap with the footprint of a leader vehicle. The multi-vehicle problem can be addressed by using this core solution in a hierarchical fashion. Assume that there are \( n \) vehicles (with initially disjoint footprints) that need their footprints to overlap in some specified topology. These \( n \) vehicles could be split into \( k \) clusters comprising of \( n_1, n_2, ..., n_k \) vehicles, respectively, and each cluster has a designated leader \( A_1, A_2, ..., A_k \). Then, \( n_1 \) follower sensors can move in trajectories obtained from the proposed guidance laws so that their footprints overlap with that of their leader \( A_1 \), while \( n_2 \) follower sensors move so that their footprints overlap with that of their leader \( A_2 \), and so on. At the end of this operation, \( k \) clusters \( X_1, X_2, ..., X_k \) will have formed, where all the followers in each \( X_i \) cluster have achieved an overlap of their footprint with their leader \( A_i \). A single vehicle from cluster \( X_i \) can then move to achieve overlap of its footprint with another vehicle in cluster \( X_j \). During this motion of \( X_i \), the leader of cluster \( X_i \) can move with the sole objective of getting its footprint to overlap with that of a designated leader of cluster \( X_j \). The individual vehicles in \( X_i \) will continue to maintain their overlap with their leader using the core solution presented in this chapter.

The above represents one of many possible approaches, all of which are centralized solutions in that there needs to be a central authority that computes which vehicle footprint is to overlap with which other vehicle’s footprint. A more general solution that involves fully decentralized/distributed control laws that achieve \( n \)–vehicle overlap can be constructed by embedding the proposed core solution in a graph theoretic framework, and is an avenue for future research.
CHAPTER 9
CONCLUSION

In this dissertation, we addressed the problems of collision avoidance and rendezvous of vehicle footprints. The collision avoidance problem is addressed using the collision cone approach [99] to develop collision avoidance guidance laws in dynamic environments. The collision cone approach does not put any constraints on the shapes of the objects, and the exact shape of the objects is considered in determining the collision avoidance guidance laws. Two collision avoidance laws are developed for arbitrarily shaped objects, one governing the magnitude of acceleration as primary input, and the other governing the direction of acceleration. The effects of time delays and singularities are also analyzed. We also develop cooperative collision avoidance laws for heterogeneous shaped objects that are integrated with formation control law and simulated on objects with double integrator dynamics. These guidance laws are then tested on two different types of IPMC-actuated robotic fish dynamic models. Using a Lyapunov-based approach, analytical expressions of nonlinear guidance laws for cooperative collision avoidance are determined, and furthermore, it is shown how these guidance laws can be made energy-optimal.

An investigation into the incorporation of the collision cone in a pursuit evasion game, is conducted. The shapes of the objects in the pursuit evasion game are arbitrary. The study considers two players with the pursuer applying an acceleration that attempts to bring the relative velocity vector into the interior of the collision cone, and the evader applying an acceleration that attempts to bring the relative velocity vector out of the collision cone. These investigations demonstrate several interesting facets illustrating the presence of (local) saddle points in the game, and call for further investigations using the tools of differential games and optimal control theory.

We also demonstrated the collision cone between a point object - a deforming object, a circular shaped object - a deforming object, and an arbitrarily shaped object and a deforming
object. For each of these cases, the collision cone functions are embedded in a Lyapunov-based framework, and these are used to develop analytical expressions for nonlinear guidance laws to enable collision avoidance. Treatment of scenarios that involve the presence of discontinuities at the boundaries of deforming objects is a promising avenue for further work.

There are many applications that require vehicles to follow trajectories so as to cause their sensing or communication footprints to make contact with, and subsequently overlap, with one another. This dissertation proposed the concept of a rendezvous cone towards devising such trajectories. By driving the relative velocity vector into the rendezvous cone, vehicles can be steered along trajectories that will ensure overlap of their footprints. The rendezvous cone is applicable for footprints of any arbitrary shape, convex as well as non-convex. Analytical expressions of the rendezvous cone are developed, and are then used as the basis for development of analytical guidance laws that enable the rendezvous to a desired depth of overlap.

The collision cone approach and notion of rendezvous cones presented in this dissertation assume availability of accurate sensor data in computing the collision and rendezvous cones. A future avenue to extend this research would be to consider these uncertainties in the sensor and measurement data. Analyzing the effects of delays on the collision cone and rendezvous cone computations is an avenue for further research.
BIBLIOGRAPHY


