Motion Planning in Dynamic Environments: Obstacles Moving Along Arbitrary Trajectories

Zvi Shiller[1], Frederic Large[2], Sepanta Sekhavat[2] and Christian Laugier[2]

[1] Department of Mechanical Engineering-Mechatronics
College of Judea and Samaria
Ariel, Israel
shiller@yosh.ac.il

[2] INRIA Rhone-Alpes
38330 Grenoble, France
laugier@imag.fr

Abstract

This paper generalizes the concept of velocity obstacles [3] to obstacles moving along arbitrary trajectories. We introduce the non-linear velocity obstacle, which takes into account the shape, velocity and path curvature of the moving obstacle. The non-linear v-obstacle allows selecting a *single* avoidance maneuver (if one exists) that avoids any number of obstacles moving on any known trajectories. For unknown trajectories, the non-linear v-obstacles can be used to generate local avoidance maneuvers based on the current velocity and path curvature of the moving obstacle. This elevates the planning strategy to a second order method, compared to the first order avoidance using the linear v-obstacle, and zero order avoidance using only position information. Analytic expressions for the non-linear v-obstacle are derived for general trajectories in the plane. The non-linear v-obstacles are demonstrated in a complex traffic example.

1 Introduction

Dynamic environments represent an important and a growing segment of modern automation, in applications as diverse as, intelligent vehicles negotiating freeway traffic, air and sea traffic control, automated wheel chairs [2], automated assembly, and animation. Common to these applications is the need to quickly select avoidance maneuvers that avoid potential collisions with moving obstacles.

There has been a sizable body of work on this problem (see [3] for an extended survey). However, not until recently, the problem was addressed in what we consider to be a *zero order* approach because it relies explicitly on the positions of both robot and obstacles to determine potential collisions. A *first order*, velocity based, approach was presented in [3], introducing the concept of Velocity Obstacles (v-obstacles). The v-obstacles allow to efficiently select a *single* velocity by the robot that avoids any number of moving obstacles (if such a solution exists), given that they maintain their current velocities. This method is termed *first order* since it is based on the first order approximation (velocity) of the trajectories followed by the moving obstacles.

Although only first order, it was shown to successfully handle obstacles moving on arbitrary trajectories [3]. However, if the obstacles are moving along *known* trajectories, then deriving the v-obstacle to reflect their exact trajectories can result in fewer adjustments by the avoiding vehicle. In addition, the exact (non-linear) v-obstacle may be necessary in cases where the linear v-obstacle indicates a collision, when in fact there is none because of the curved motion of the other vehicle.

For example, a vehicle moving along a curved road, turning left, appears to be on a collision course with the vehicles on the opposite lane, using the linear v-obstacle, when in fact there is no imminent collision if the approaching vehicles stay on the curved lane. The v-obstacle that takes into account the road curvature should eliminate this confusion. Another example is the problem of merging with traffic that follows a turn-around (a popular intersection in European cities), where the vehicles moving around the curve appear to be on a collision course with the vehicle waiting to join, using the linear v-obstacle. Here too, the appropriate v-obstacle will allow the waiting vehicle to plan an efficient maneuver that safely merges with the traffic flow.

A similar first order approach for collision detection between moving obstacles of arbitrary shapes was developed separately in [1] based on results from missile guidance. A different approach for collision avoidance uses game theory in the context of conflict resolution in air traffic management [5]. It assumes a competitive game that ensures safety by computing the worst case strategies for the pursuer and evader. While based on a nice theoretical foundation, this approach seems difficult to practically extend to more than two players.

This paper extends the concept of linear v-obstacles [3] to obstacles moving along arbitrary trajectories. It focuses on the unified representation of static and moving obstacles, and not on the avoidance strategy problem. This representation assumes a single avoiding intelligent agent and any number of apathetic (neither competitive nor cooperative) obstacles.

The paper is organized as follows: we first review the *linear* velocity obstacle in Section 2. Section 3 extends the velocity obstacle to general trajectories, and a convenient approximation of its boundaries is presented in Section 4. This representation is associated with an inherent singularity as discussed in Section 5. Section 6 discusses the use of the velocity obstacle representation to compute avoidance maneuvers.

2 Review of the Linear V-Obstacle

A major advantage of the v-obstacle is that it is represented directly in the configuration space [3]. The v-obstacle represents the forbidden velocity "vectors" of a robot, or equivalently, the "velocity obstacle," at a given time, which are represented by a set of "points" in the configuration space. The geometry of this set of points can be precisely and easily defined, as discussed below. For simplicity, we consider circular robots and obstacles. Growing the obstacles by the radius of the robot transforms the problem to a point robot avoiding circular obstacles. It is also assumed that the instantaneous states (position, velocity, and acceleration) of obstacles moving along arbitrary trajectories are either known or measurable.

A few words about notation: henceforth, A denotes a point robot, and $\mathcal{B}$ denotes the set of points defining the geometry of and obstacle. Since the obstacle is solid, $\mathcal{B}$ does not depend on t. $B(t)$ denotes the set of points occupied by the obstacle B at time t. Thus, if $b \in \mathcal{B}$ is some representative point (usually the center) of $\mathcal{B}$, and it coincides with some point $c(t)$ at time t, then $B(t) = c(t) + \mathcal{B}$. a/b denotes a *ray* that consists of the half line that originates at a, passes by b, and does not include a. Similarly, a/v denotes a ray that originates at a and is parallel to v.

The linear v-obstacle is demonstrated for the scenario shown in Figure 1, where, at time t_0, obstacle $B(t_0)$, moving at some linear velocity v_b, is to be avoided by a point robot A. The linear v-obstacle at time t_0 is constructed by first generating the so called *relative velocity cone* (RVC) by sweeping a half line from A along ∂B, the boundary of B. RVC is thus defined as the union of all rays originating from A and passing through $\partial B(t_0)$:

$$RVC = \cup A/b, \quad b \in \partial B(t_0). \tag{1}$$

RVC is the set of all velocities, $v_{a/b} \neq 0$, of A *relative* to B that would result in collision at some time $t \in (0, \infty)$, assuming that the obstacle stays on its current course at its current speed. Consequently, relative velocities outside of RVC ensure avoidance of B at all times under the same assumptions. Translating RVC by v_b produces the *velocity obstacle*, VO [3]:

$$VO = v_b + RVC \tag{2}$$

where in this context, "+" denotes the Minkowski sum:

$$VO = \{x | x = y + v_b, y \in RVC\}. \tag{3}$$

Thus, VO represents a set of *absolute* velocities, v_a, of A that would result in collision at some time $t \in (0, \infty)$. Geometrically, each point $x \in VO$ represents a vector originating at A and terminating at x. In Figure 1, v_{a2} is a colliding velocity, whereas v_{a1} is not. Note that VO reflects the positions of A and B, and v_b, at $t = t_0$, and is independent of v_a.

We call VO a *linear* v-obstacle because it ensures avoidance using a *single* maneuver at time t_0 under the assumption that the obstacle maintains its current course and speed. For an obstacle moving on a curved path, its current velocity represents a first order approximation of its actual trajectory at time t_0. Similarly, the linear v-obstacle represents a first order approximation of the non-linear v-obstacle presented later.

By construction, RVC is a convex cone with extreme rays tangent to $B(t_0)$, as was shown in Figure 1. The extreme rays represent the set of relative velocities that would result in A grazing $\mathcal{B}$ at some future time. Denoting λ the ray $A/v_{a/b}$, the point of contact, p, between A and $\mathcal{B}$ is the first (closest to A) intersection between λ and $\partial B(t_0)$, as stated in the following Proposition. The time of contact is determined by the magnitude of $v_{a/b}$.

Proposition 1:
Let A and $\mathcal{B}$ be moving at constant velocities such that $v_{a/b} \in RVC$. Let λ be the ray $A/v_{a/b}$. Assuming $A \cap B(t_0) = \emptyset$ at $t = t_0$, A penetrates $\mathcal{B}$ at some point $p = \lambda \cap \partial B(t_0), (A, p) \cap B(t_0) = \emptyset$.

Proof: Let a and b move at constant velocities v_a and v_b on trajectories that intersect at point c at time $t = t_1$. It is easy to show that at any time $t_0 < t < t_1$, the points $a(t), b(t), c$ form a triangle with sides $v_a(t_1 - t), v_b(t_1 - t), v_{a/b}(t_1 - t)$. This triangle is similar to the "velocity triangle" formed by the vectors v_a, v_b, and $v_{a/b}$.

Referring to Figure 2, the line segment $A/v_{a/b} \cap B(t_0)$ represents points on $\mathcal{B}$ that would collide with A since the trajectories of these points form with the trajectory of A triangles that are similar to the velocity triangle. Of those points, $p(t_0) \in \partial B(t_0), (A, p) \cap B(t_0) = \emptyset$ is the closest to A, and is hence the first to collide with A. $\square$

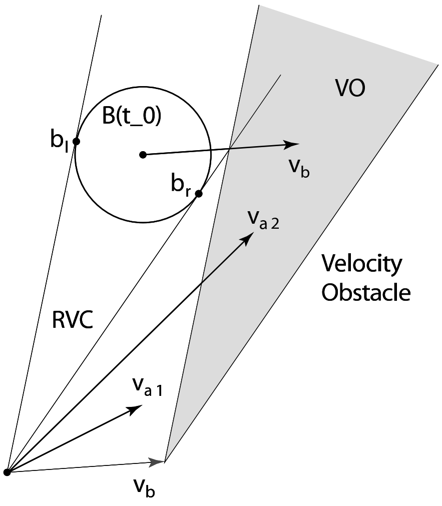

Figure 1: The Linear Velocity Obstacle.

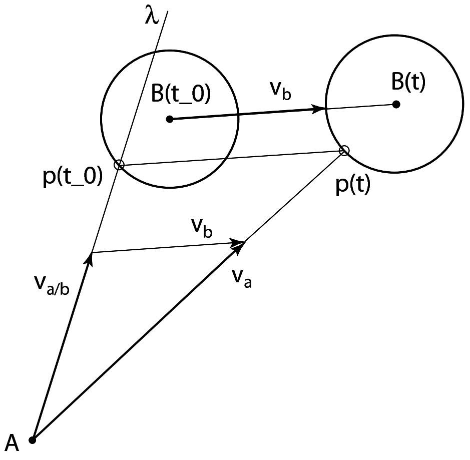

Figure 2: The Tangency Point.

Proposition 2:
Let $\mathcal{B}$ be a circular obstacle, moving at a constant velocity v_b, and A be a point robot. Also, let the right and left extreme rays of RVC be λ_r and λ_l, and their right and left tangency points with $B(t_0)$ be b_r and b_l, respectively. A will graze $\mathcal{B}$ at point b_r iff $v_{a/b}$ belongs to λ_r. The same holds for point b_l and ray λ_l.

Proof: This is a direct result of Proposition 1, where b_r is the only point (for a circular $\mathcal{B}$) at which the ray $A/v_{a/b}$, $v_{a/b} \in \lambda_r$, intersects $B(t_0)$. Contact at other points (resulting from $v_{a/b} \notin \lambda_r$ but $v_{a/b} \in RVC$) results in A penetrating $\mathcal{B}$. □

Observing that points b_r and b_l partition $\partial B(t_0)$ into two subsets: ∂U (farthest from A) and ∂L (closest to A), we state the following Proposition:

Proposition 3:
Let A and $\mathcal{B}$ be moving at constant velocities such that $v_{a/b}$ is interior to RVC. Assuming $A \cap B = \emptyset$ at $t = t_0$, A penetrates $\mathcal{B}$ through some point $p \in \partial L$.

Proof: This follows from the ray $A/v_{a/b}$ penetrating $B(t_0)$ first through ∂L. Alternatively, $v_{a/b}$ points into $B(t_0)$ along ∂L (p_1 in Figure 3), and not along ∂U (p_2 in Figure 3) at time of contact t. □

VO represents the velocities of A that would result in collision at any time $t = (0, \infty)$. It is useful to identify a subset of VO that would result in collision at a specific time. The time to collision, t_c, for any $v_{a/b} \in RVC$ is simply

$$t_c = t_0 + \frac{\|p\|}{\|v_{a/b}\|}, \;\; p = \partial L \cap A/v_{a/b}. \tag{4}$$

where $\|\cdot\|$ is the Euclidean norm, and p is a point (and vector) in a coordinate frame centered at A.

Using (4), we obtain the set $RVC(t)$ of all relative velocities, in a frame centered at A, that would result in collision with any point of $\mathcal{B}$ at time $t > t_0$:

$$RVC(t) = \frac{c(t_0) + \mathcal{B}}{t - t_0}, \;\; t > t_0. \tag{5}$$

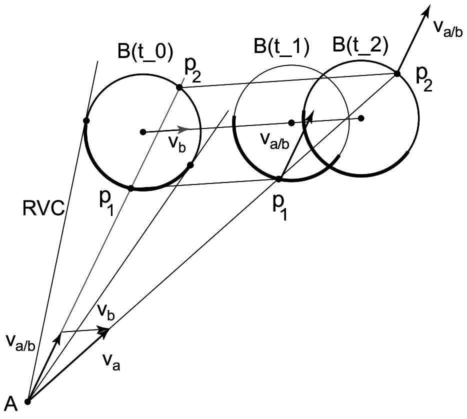

Figure 3: Collision points between A and B.

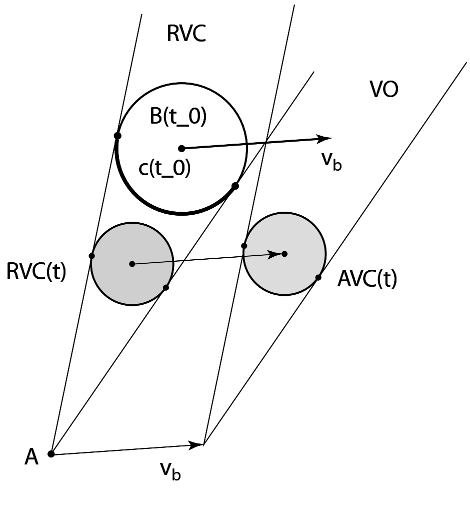

Figure 4: A Temporal Element of VO.

where $c(t_0)$ represents the position of $\mathcal{B}$ at time t_0.

The shape of $RVC(t)$ thus depends on $\mathcal{B}$ and its relative position to A at time t_0, as shown in Figure 4. Translating $VC(t)$ by v_b produces the set $AVC(t)$ of all *absolute* velocities that would result in collision with any point of $\mathcal{B}$ at time $t > t_0$:

$$AVC(t) = v_b + RVC(t). \tag{6}$$

This leads to the following formal statement of the linear v-obstacle, $VO = \bigcup_{t>t_0} AVC(t)$:

Theorem 1: Considering at time t_0 a point robot A, located at the origin, and an obstacle $\mathcal{B}$ centered at $c(t_0)$ with a constant velocity v_b. The linear v-obstacle, VO, represents the set of all linear velocities of A that would collide with $\mathcal{B}$:

$$VO = v_b + \bigcup_{t>t_0} \frac{c(t_0) + \mathcal{B}}{t - t_0}. \tag{7}$$

Clearly, the linear v-obstacle can be truncated to reflect collisions within a specified time interval $[t_1, t_2]$ simply by defining the time interval in (7):

$$VO(t_1, t_2) = v_b + \bigcup_{t_2 \geq t > t_1} \frac{c(t_0) + \mathcal{B}}{t - t_0}. \tag{8}$$

Truncating VO to reflect specific time limits allows to focus the motion planning problem on potential collisions occurring within a specified time interval, such as imminent collisions occurring within some given time horizon [3]. This permits avoidance maneuvers that are potentially risky, but at times that are practically insignificant at the decision time t_0. It also allows the consideration of large static obstacles, such as surrounding walls and highway barriers, whose v-obstacles cover the entire velocity space.

3 The Non-Linear V-Obstacle

The non-linear v-obstacle ($NLVO$) applies to the scenario shown in Figure 5, where, at time t_0, a point robot, A, attempts to avoid a circular obstacle, $\mathcal{B}$, that at time t_0 is located at $c(t_0)$), and is following a general known trajectory, $c(t)$. $NLVO$ thus consists of all velocities of A at t_0 that would result in collision with the obstacle at any time $t > t_0$. Selecting a *single* velocity, v_a, outside $NLVO$ should therefore avoid collision at all times, or

$$(A + v_a^{t_0} t) \notin (c(t) + \mathcal{B}) \quad \forall t > t_0 \tag{9}$$

where $v_a^{t_0}$ denotes the velocity of A at time t_0. Note that the emphasis here is on selecting a *single* avoidance maneuver at time t_0, for any trajectory followed by the obstacle.

The non-linear v-obstacle is constructed by first determining the absolute velocities of A, v_a, that would result in collision at a specific time t. Referring to Figure 5, $v_a^{t_0}(t, p)$ that would result in collision with point $p \in B(t)$ at time $t > t_0$ expressed in a frame centered at $A(t_0)$ is simply

$$v_a^{t_0}(t, p) = \frac{c(t) + r_p}{t - t_0}, \tag{10}$$

where r_p is the vector to point p in the obstacle's fixed frame.

Similarly, the set, $NLVO(t)$ of all absolute velocities of A that would result in collision with any point in $B(t)$ at time $t > t_0$ is:

$$NLVO(t) = \frac{c(t) + \mathcal{B}}{t - t_0} \tag{11}$$

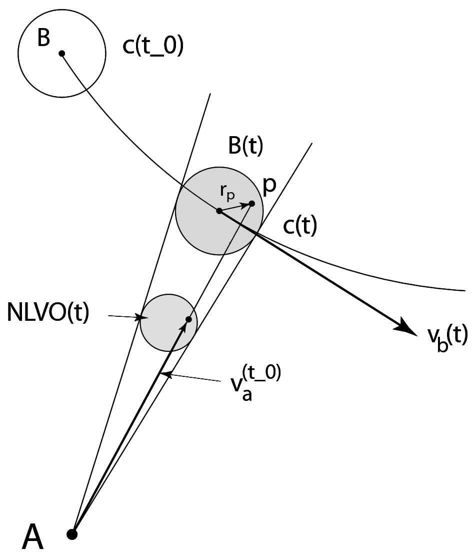

Figure 5: Construction of the non-linear v-obstacle.

Geometrically, $NLVO(t)$ is a scaled $\mathcal{B}$, bounded by the cone formed between A and $B(t)$. Note that the extreme rays of this cone are not necessarily the points where A grazes B, as discussed later. Note also that $NLVO(t)$ is independent of $v_b(t)$, since it is a function of $B(t)$ and not of its future positions (determined by $v_b(t)$). This leads to the construction of the linear v-obstacle, as stated in the following Theorem:

Theorem 2: Let A be a point robot, located at time $t = t_0$ at the origin, and B be an obstacle that is moving along a general trajectory $c(t), t = [t_0, \infty)$. The non-linear v-obstacle, $NLVO$, representing the set of all linear velocities of A that would collide with $B(t)$ at time $t = (t_0, \infty)$ is defined by

$$NLVO = \bigcup_{t > t_0} \frac{c(t) + \mathcal{B}}{t - t_0}, \tag{12}$$

The non-linear v-obstacle is a warped cone with apex at A. The boundaries of $NLVO$ represent velocities that would result in A grazing $\mathcal{B}$. The tangency points between A and $B(t)$ are determined by the equivalent linear v-obstacle, as stated in the following Lemma. We first define the equivalent linear v-obstacle:

Definition 1: Let $B(t)$ be at $c(t)$, and moving at $v_b(t)$ at time $t > t_0$. Its equivalent linear v-obstacle, $ELVO(B, t)$, is the linear v-obstacle of a virtual $\mathcal{B}$ that reaches $c(t)$ by moving at a constant $v_b(t)$ over $[t_0, t]$. $ELVO(B, t)$ is constructed at time t_0 of the linear trajectory.

Referring to Figure 6, $ELVO(B, t)$ is constructed by integrating $v_b(t)$ backwards in time to t_0, constructing $ERVC$ (the equivalent relative velocity cone) at point $c_0 = c(t) - v_b(t)(t - t_0)$, then translating $ERVC$ by $v_b(t)$.

Lemma 1:
Let A be a point robot and $\mathcal{B}$ be a circular obstacle, moving along a general trajectory, $c(t)$. A will

graze $B(t)$ iff

$$v_a^{t_0}(t) \in NLVO(t) \cap \partial ELVO(B,t)$$

with $\partial ELVO(B,t)$ being the extreme rays of $ELVO(B,t)$.

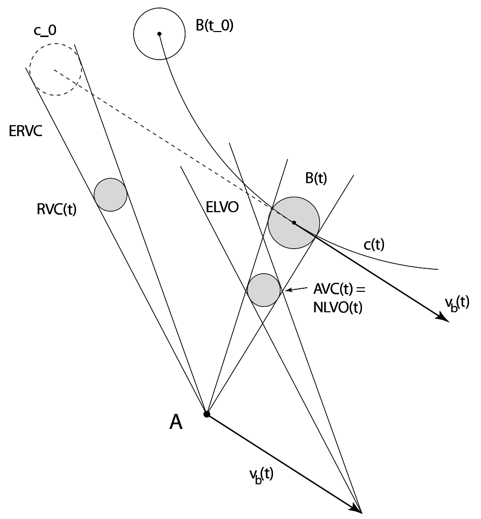

Figure 6: The Equivalent Linear V-Obstacle.

Proof: The proof is based primarily on the fact that A grazing $B(t)$ at t is a contact of first order and therefore depends only on the relative position and velocity at t. Hence we can equivalently consider the problem for a general curve $c(t)$ or for the tangent to $c(t)$ and solve the problem using $ELVO(B,t)$. We begin by proving that the direction (not necessarily the magnitude) of the relative velocity $v_{a/p}$ at the tangency point p at contact is not affected by the rotation of $B(t)$.

Referring to Figure 7, consider an arbitrary point, $p \in \partial B(t)$. If $v_b(t)$ is the velocity of $c(t)$, then the velocity of p, v_p is:

$$v_p = v_b + r \times \omega, \tag{13}$$

where ω satisfies $v_b = \rho \times \omega$, ρ is the instantaneous radius of curvature (here we use it as a vector), and r is the position vector of p on $B(t)$. The relative velocity $v_{a/p}$ is

$$v_{a/p} = v_a - v_p = v_a - v_b - r \times \omega. \tag{14}$$

For p to be a tangency point, it is necessary that $v_{a/p}$ be tangent to $B(t)$, or $v_{a/p}$ be perpendicular to r:

$$v_{a/p} \cdot r = (v_a - v_b - r \times \omega) \cdot r = 0. \tag{15}$$

where $(\cdot)$ denotes the inner product. It follows that:

$$v_{a/p} \cdot r = 0 \Longleftrightarrow v_{a/b} \cdot r = 0. \tag{16}$$

Thus, p is a tangency point at t iff A makes a contact with p at t, and $v_{a/b}$ is tangent to $B(t)$ at p. In other words, $p \in \partial B(t)$ and $(v_a^{t_0}(t,p) - v_p(t)) \perp r_p(t)$. From (16) this is equivalent to $p \in \partial B(t)$ and $(v_a^{t_0}(t,p) - v_b(t)) \perp r_p(t)$. Note that

$$v_a^{t_0}(t,p) \in NLVO(t) = AVC(t) \subset ELVO(B,t)$$

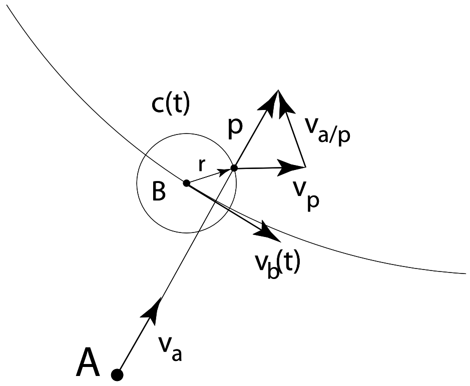

Figure 7: Tangency Points for a General Trajectory.

therefore (see figure 6):

$$(v_a^{t_0}(t,p) - v_b(t)) \in RVC(t) \subset ERVC(B,t)$$

Since c_0 is a translation of $c(t)$, $r_p(t)//r_p^0$ (the relative position of p on the virtual $\mathcal{B}$ at c_0). Therefore a velocity $v_a^{t_0}$ leads to A grazing B at time t iff $v_{a/b}^{t_0} \in \partial RVC(t)$ (leading to a collision with $p \in \partial B(t)$) and $v_{a/b}^{t_0} \perp r_p^0$. There are two and only two such relative velocities which are the elements of $RVC(t)$ belonging to the extremal rays of $ERVC(B,t)$. It follows that $v_a^{t_0} \in NLVO(t) \cap \partial ELVO(B,t)$, which proves the Lemma. □

The following Theorem regarding the boundary of $NLVO$ follows from the proof of Lemma 1:

Theorem 3: For a point robot A and a circular obstacle $\mathcal{B}$ moving along an arbitrary curve $c(t)$, given $b_l(t)$ and $b_r(t)$, the tangency points between $ERVC(B(t))$ and the virtual $\mathcal{B}$ at c_0, the boundary of $NLVO$ consists of the curves:

$$\frac{b_l(t)}{t-t_0} + v_b, \qquad \frac{b_r(t)}{t-t_0} + v_b$$

where $b_l(t), b_r(t)$ are the vectors to the respective points in a coordinate frame centered at A.

4 An Analytic Approximation of $NLVO$

A conservative representation of the boundary of $NLVO$ can be derived using complex numbers. The trajectory, $c(t)$, followed by the obstacle, $\mathcal{B}$, can be represented by

$$c(t) = d(t)e^{i\theta(t)}. \tag{17}$$

where $i = \sqrt{-1}$, and $d(t)$ and $\theta(t)$ are measured in a coordinate frame centered at A (see Figure 8). Differentiating (17) yields $v_b(t)$:

$$v_b(t) = \dot{d}(t)e^{i\theta(t)} + i\dot{\theta}(t)d(t)e^{i\theta(t)}. \tag{18}$$

Assuming $t_0 = 0$, we divide (17) by t to obtain the center, $c_v(t)$, of $NLVO$:

$$c_v(t) = \frac{d(t)}{t}e^{i\theta(t)}. \tag{19}$$

By Theorem 3, the boundary of $NLVO$ consists of the tangency points between $ELVO$ and $RVC(t)$. For simplicity, let $\mathcal{B}$ be circular, although the following applies, with minor modifications, to $\mathcal{B}$ of any shape. $ELVO$ is a cone with a center line passing through $c_v(t)$ and an apex at $v_b(t)$. The center line, $c_l(t)$, of this cone is therefore

$$\begin{aligned} c_l(t) &= c_v(t) - v_b(t) \\ &= [(\frac{1}{t} - i\dot{\theta}(t))z(t) - \dot{z}(t)]e^{i\theta(t)}. \end{aligned} \tag{20}$$

The corresponding unit vector is

$$\begin{aligned} \hat{c}_l(t) &= \frac{c_l(t)}{\|c_l(t)\|} \\ &= \frac{[(1 - it\dot{\theta}(t))d(t) - t\dot{d}(t)]e^{i\theta(t)}}{\sqrt{(1+\dot{\theta}^2(t))d^2(t)t^2 + \dot{d}^2(t)t^2 - 2d(t)\dot{d}(t)t}}. \end{aligned} \tag{21}$$

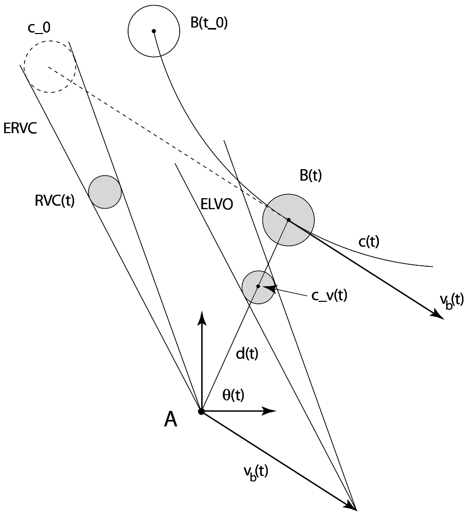

Figure 8: Construction of the approximate non-linear v-obstacle.

The exact boundary of $NLVO$ is traced by the tangency points between $ELVO$ and $RVC(t)$. For simplicity, we approximate the exact tangency points by a simpler procedure that computes instead the tangency points, $vo_r(t)$ and $vo_l(t)$ between $RVC(t)$ and the two tangents passing parallel to $\hat{c}_l(t)$:

$$vo_r(t) = c_v(t) + i\frac{r}{t}\hat{c}_l(t) \tag{22}$$

$$vo_l(t) = c_v(t) - i\frac{r}{t}\hat{c}_l(t), \tag{23}$$

where r is the radius of $\mathcal{B}$. The points $vo_r(t)$ and $vo_l(t)$ are exterior to the true $NLVO$ because $(b_r, b_l) \cap (vo_r(t), vo_l(t))$, where (a, b) represent the boundary arc between a and b. Thus, using $vo_r(t)$ and $vo_l(t)$ yields a conservative representation of $NLVO$, except near singularity points, where $A \in RVC(t)$, as discussed later. It is important to note that the computation of the exact tangency points is not terribly complicated. Never the less, we prefer this approximation since it avoids trigonometric calculations.

We now apply the preceding formula to a circular obstacle of radius r that moves along a circular trajectory of radius R, centered at point $O = De^{i\phi}$ in a coordinate frame centered at A, at angular speed ω, starting at $\theta(t_0) = \theta_0$:

$$c(t) = De^{i\phi} + Re^{i\theta(t)} \tag{24}$$

$$c_v(t) = \frac{1}{t}[De^{i\phi} + Re^{i\theta(t)}] \tag{25}$$

$$v_b(t) = i\omega Re^{i\theta(t)} \tag{26}$$

$$c_l(t) = \frac{1}{t}[De^{i\phi} + (1 - i\omega t)Re^{i\theta(t)}] \tag{27}$$

$$\hat{c}_l(t) = \frac{De^{i\phi} + (1 - i\omega t)Re^{i\theta(t)}}{\sqrt{D^2 + R^2 + (\omega tR)^2 + 2DR(cos\alpha + \omega t sin\beta)}} \tag{28}$$

$$\theta(t) = \omega t + \theta_0 \tag{29}$$

$$\alpha = \theta(t) - \phi \tag{30}$$

$$\beta = \theta(t) - \phi \tag{31}$$

$$\begin{aligned} vo_r(t) &= \frac{1}{t}[De^{i\phi} + (1 - i\omega t)Re^{i\theta(t)} \\ &+ \frac{r(iDe^{i\phi} + (i + \omega t)Re^{i\theta(t)})}{\sqrt{D^2 + (1 + \omega^2 t^2)R^2 + 2DR(cos\alpha + \omega t sin\beta)}}] \end{aligned} \tag{32}$$

$$\begin{aligned} vo_l(t) &= \frac{1}{t}[De^{i\phi} + (1 - i\omega t)Re^{i\theta(t)} \\ &- \frac{r(iDe^{i\phi} + (i + \omega t)Re^{i\theta(t)})}{\sqrt{D^2 + (1 + \omega^2 t^2)R^2 + 2DR(cos\alpha + \omega t sin\beta)}}]. \end{aligned} \tag{33}$$

5 Singularities

The boundary of $NLVO$ consists of the tangency points between $ERVC$ and $RVC(t)$ (also $ELVO$ and $AVC(t)$). This assumes that the apex of $ERVC$ (A) is outside of $RVC(t)$, for otherwise $ERVC$ is not defined. There are cases when $A \in RVC(t)$, as shown in Figure 9. This occurs when $B(t)$ moves away from A such that when translated to the virtual point c_0, it includes A. Although $ERVC$ is not defined at such singularities, $NLVO(t)$ is, and so is $RVC(t) = NLVO(t) - v_b$. The only consequence of $ERVC$ not being defined is that we cannot identify tangency points between A and $B(t)$. This implies that A cannot be tangent to $B(t)$. Therefore, any attempt to reach $B(t)$ at a singularity would result in A penetrating $B(t)$, for *all* $v_a^{t_0}(t) \in NLVO(t)$.

6 Avoidance Maneuvers

Given the efficient representation of the moving obstacles by NLVO, we now discuss their use for planning avoidance maneuvers, which in turn can be used to construct a local or global motion planner. Motion planning in dynamic environments deserves a separate treatment and is hence out of the scope of this paper. We comment here on the attainable cartesian velocities from which an avoidance maneuver (an avoiding velocity) can be selected, and the time horizon which is used to truncate the velocity obstacles to account for potential velocities within a specified time frame.

6.1 Attainable Cartesian Velocities

The set of attainable cartesian velocities (ACV) of the maneuvering robot represents the avoidance maneuvers that are dynamically feasible over a given time interval, Δt [3]. To formally define this set, we consider nonredundant dynamic systems of the form:

$$\dot{x}_1 = g(x_1, x_2) \tag{34}$$

$$\dot{x}_2 = f(x_1, x_2, u), \tag{35}$$

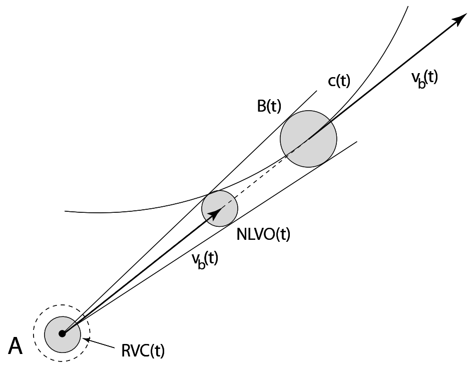

Figure 9: A singularity.

where x_1 represents the position state, x_2 the velocity states, and u the controls. The v-obstacles reflect the colliding *cartesian* velocities of the maneuvering robot at given cartesian positions relative to the moving obstacles, which may not necessarily be x_1 and x_2. Denoting x the cartesian position coordinates and v the cartesian velocities, we can map (under the assumption of nonredundancy) x_1 to x and x_2 to v:

$$x = G(x_1, x_2) \tag{36}$$

$$v = F(x_1, x_2). \tag{37}$$

The attainable *cartesian* velocities, $ACV(t+\Delta t)$, are integrated from the current state $(x_1(t), x_2(t))$ by applying all admissible controls $u(t) \in U$:

$$ACV(t + \Delta t) = \{v | v = v(t) + F(\Delta t \dot{x}_2(u)), u \in U\}. \tag{38}$$

The geometric shape of $ACV(t + \Delta t)$ depends on the specific system dynamics. The simplest is a point mass model, with system dynamics represented by:

$$\dot{x}_1 = x_2 \tag{39}$$

$$\dot{x}_2 = u, \tag{40}$$

where $x_1, u \in \mathcal{R}^2$. The set of attainable cartesian velocities is then:

$$ACV(t + \Delta t) = \{v | v = v(t) + \Delta t u, u \in U\}. \tag{41}$$

Assuming constant control constraints of the form

$$|u_1| \leq 1 \tag{42}$$

$$|u_2| \leq 1, \tag{43}$$

ACV consists of the rectangular set of admissible controls, scaled by Δt and translated by $v(t)$.

Note that the attainable velocities are computed for time $t + \Delta t$, and hence apply to the position $x_1(t + \Delta t)$ of the maneuvering robot. Thus, the attainable velocities, when intersected with v-obstacles that correspond to the incremented position, would indicate those velocities that are safe if selected at time $t = t + \Delta t$.

6.2 Time Horizon

The time horizon plays an important role in selecting feasible avoidance maneuvers. It allows considering only those maneuvers that would result in a collision within a specified time interval.

Not considering the time horizon would dictate the consideration of collisions occurring at any time, which would be too prohibitive and would preclude safe maneuvers that are only momentarily dangerous. The selection of a too large time horizon is safe, but it would preclude safe maneuvers and thus would conflict with completeness. The selection of a too small time horizon would allow maneuvers that are definitely dangerous, as they would bring the robot too close at too high a velocity to avoid the obstacle, but they would ensure that no safe maneuver was overlooked.

The minimal safe time horizon is the one that allows sufficient time for the robot to avoid the obstacle. This depends on the size of the obstacle, its velocity, and the attainable velocities of the robot. The absolute minimum time horizon is the time it would take for the robot to clear the way of the obstacle. For a static obstacle, this is simply the time of an optimal avoidance maneuver, which for a typical passenger car is approximately 1 s [4]. This implies that assuming a time horizon shorter than 1 s would leave no time for the approaching vehicle to avoid the static obstacle. Using a larger time horizon would be safe, but would preclude safe avoidance maneuvers.

7 Examples

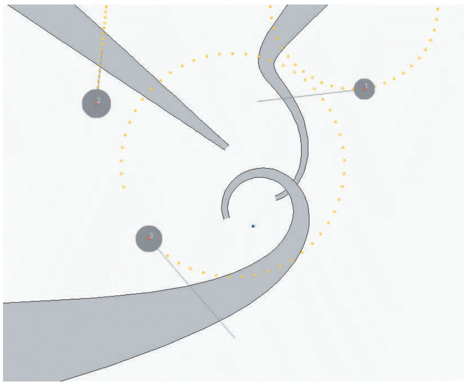

Figure 10: V-Obstacles for circular and straight line motions.

The NLVO were implemented in an interactive Java applet for real-time simulation of a robot avoiding moving circular obstacles. Figure 10 shows three obstacles and their NLVO as seen by the point robot A. The v-obstacle of obstacle B2 is a straight cone since it moves at a constant velocity.

Figure 11 shows the use of the NLVO to solve a difficult traffic merging problem. The robot A coming from the left wishes to merge tangentially with the traffic in the right lane of a curved road after crossing the left lane with opposing traffic, using a constant velocity. The vehicles on the curved lanes move at constant speeds.

In Figure 11 (a) represents the initial configuration with the trajectories and the velocities of the robots; (b) represents the same situation as (a), but with the NLVO drawn. The complexity of this situation is apparent from the many discontinuous sets of avoiding velocities. The choice of a velocity in the free space in (b) permits to perform the entire maneuver safely at a constant speed, as shown in the remaining snapshots.

8 Conclusions

The concept of velocity obstacles was generalized to consider obstacles moving on arbitrary trajectories. The nonlinear v-obstacle consists of a warped cone that is a time-scaled map of the obstacle along its trajectory. Selecting a single velocity vector outside the nonlinear v-obstacle guarantees avoidance of the obstacle during the time interval for which the v-obstacle has been generated. Analytic expressions of the non-linear v-obstacle for general planar obstacles were derived. They can

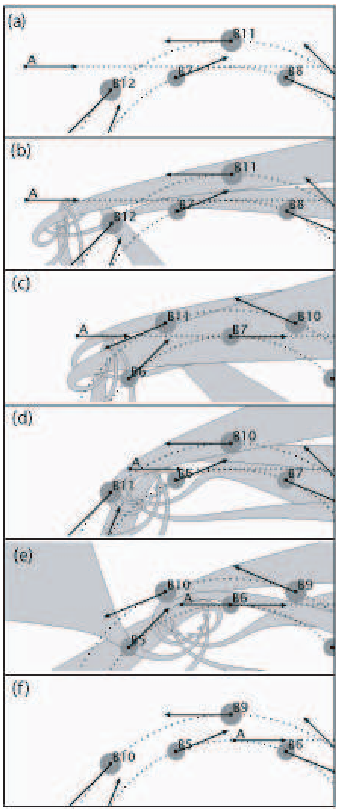

Figure 11: Merging with traffic along a curved road at a constant speed.

be used to approximate unknown trajectories by using the current velocity and path curvature of the moving obstacle. Such second order approximations yield more efficient avoidance maneuvers (fewer adjustments) than the first approximation offered by the linear v-obstacle.

9 Acknowledgments

This work was partly supported by the IMARA *(Informatique, Mathmatiques et Automatique pour la Route Automatise)* program of INRIA on intelligent road transportation. Dr. Christian Laugier's support for the first author's visit to INRIA Rhone Alpes, Grenoble, during July 2000, is gratefully acknowledged. The first author also thanks Professor Debasish Ghose from India Institute of Science, Bangalore, for stimulating discussions during his sabbatical at UCLA.

References

[1] A. Chakravarthy and D. Ghose. Obstacle avoidance in a dynamic environment: A collision cone approach. *IEEE Transactions on Systems, Man, and Cybernetics, Part A – Systems and Humans*, 28(5):562–574, September 1998.

[2] Prassler E., Scholz J., and P. Fiorini. Navigating a robotic wheelchair in a railway station during rush-hour. *International Journal of Robotics Research*, 18(5), May 1999.

[3] P. Fiorini and Z. Shiller. Motion planning in dynamic environments using velocity obstacles. *International Journal of Robotics Research*, 17(7):760–772, July 1998.

[4] Z. Shiller and S. Sundar. Emergency lane-change maneuvers of autonomous vehicles. *ASME Journal of Dynamic Systems, Measurement and Control*, 120(1):37–44, March 1998.

[5] C. Tomlin, G.J. Pappas, and S. Sastry. Conflict resolution for air traffic management: a study in multi-agent hybrid systems. *IEEE Transactions on Automatic Control*, 43(4):509–521, April 1998.