Determination of Terminal Landing Footprint for On-Board Terrain Assessment and Intelligent Hazard Avoidance.

Scott R. Ploen*, Charles E. Kinney†, and Homayoun Seraji
Jet Propulsion Laboratory
California Institute of Technology
4800 Oak Grove Drive
Pasadena, CA 91009-8099
Email: sploen@grover.jpl.nasa.gov

Abstract

In this paper we develop a methodology based on ballistics to estimate the landing footprint associated with the powered terminal descent phase of a Mars soft landing. The analysis is based on an idealized two-impulse thrust maneuver and leads to an analytical expression for the elliptical boundary of the landing footprint. Our goal was to develop a method (with low computational overhead) to estimate the landing footprint for use in an on-board fuzzy-logic based inference engine for real-time hazard avoidance. The inference engine is utilized to combine an estimate of the landing footprint with information about the safeness of the Martian terrain to construct an overall landing site quality index. The landing site quality index is a critical parameter that will enable the spacecraft to make intelligent real-time decisions about landing safely on unknown and hazardous terrains. The footprint generated from the ballistic analysis is then compared to the footprint resulting from numerically integrating a representative guidance law. Our study shows that under certain conditions the ballistic footprint provides an excellent estimate of the guidance computed footprint. However, other cases were also observed where the ballistic approach overpredicted the landing footprint associated with the guidance law. It was observed that the guidance footprint was sensitive to the time of guided flight T, the spacecraft initial state relative to the nominal target site, and the structure/order of the guidance law.

*Corresponding Author
†Graduate Student, University of Idaho, Moscow, Idaho, 83844

1 Introduction

In this paper we focus on a proposed Mars landing scenario beginning at an altitude of 10 km with a lander attached to a subsonic parachute. The subsonic chute is jettisoned at an altitude between 500-1000 m when the spacecraft has reached a terminal vertical velocity of approximately -50 m/s. At this altitude the powered terminal descent phase is initiated during which the spacecraft is actively controlled to a safe landing site on the surface of Mars. During both the parachute and the powered terminal descent phase, on-board sensors such as LIDAR[1] and phase-array radar will construct a real time terrain map of the approaching Martian surface. This topographic map will provide on-board information regarding the safeness[2] of potential landing sites.

The ability to compute an estimate of the landing footprint (i.e, the area on the surface of Mars the spacecraft can reach during powered descent given fuel and environmental constraints) is critical as any proposed safe landing site must be commensurate with the control authority of the spacecraft. In other words, a landing site deemed safe by on-board analysis of the surface terrain map will become a candidate (feasible) landing site only if it lies within the reachable landing footprint of the spacecraft. The landing footprint is primarily dictated by the amount of fuel necessary to realize a soft landing. The ability to provide an on-board estimate of the landing footprint is an integral part of an autonomous terminal hazard avoidance system. Moreover, once knowledge of the reachable landing area has been obtained it is desirable to quantify the "quality" of each candidate landing site. For example, given two candidate landing sites deemed equally safe by on-board analysis of the terrain map it is reasonable to expect that the "best" landing site is the one that can be attained with minimum effort/fuel.

It is also critical to have the ability to quantify the safeness of each feasible landing site in a computationally efficient manner before a decision can be made about where the vehicle should land. To this end, a fuzzy-logic based autonomous reasoning engine has been proposed to process and fuse sensor data and knowledge of the landing footprint to enable the spacecraft to make an on-board autonomous choice of landing site [2]. This on-board capability is critical as it will enable the spacecraft to autonomously distinguish between a set of candidate landing sites that lie within close range of one another. For example, if the "best" landing site is isolated, it may make more sense from a hazard avoidance perspective to choose a slightly sub-optimal site that lies in proximity to other candidate landing sites.

In the Mars landing scenario discussed here the lander has no control authority while on the parachute; however, we assume that the landing footprint can be predicted during the parachute phase by integrating an on-board spacecraft/parachute model forward in time until the parachute is jettisoned. This simple scheme allows us to predict where the lander will be at parachute separation and allows us to predict

[1]The LIDAR sensor provides range data that can be converted into an elevation map for extracting terrain characteristics such as slope and roughness.

[2]The metric used for quantifying safeness of a landing area involves local surface roughness, slope, density of large rocks, etc.

the landing footprint at the initiaition of the controlled terrminal descent phase. Information about the predicted landing area during the parachute phase is particularly valuable as it allows on-board sensors to concentrate on the region where the vehicle will have the highest probability of landing. During this period, on-board algorithms will characterize the safeness of feasible landing sites and produce a safeness-ranked listing of feasible landing sites. Once the parachute is jettisoned, the spacecraft is steered to the best landing site as determined during the parachute phase. During terminal descent, the spacecraft will continue to process sensor information regarding the local terrain and it will have the capability to autonomously retarget to a safer location if deemed necessary.

In this paper we discuss some methods of estimating the reachable landing area available to a spacecraft during the terminal descent phase of a Mars soft landing. Our goal is to provide an estimate of the landing footprint for use in an on-board hazard detection and avoidance system. Further, in order to provide some insight into the more complicated 6 degree-of-freedom problem we first develop our landing area algorithms for a point mass spacecraft. Our landing area prediction algorithm is first developed based on the physics of elementary ballistics. We then recompute the landing footprint by directly integrating a representative guidance law [1],[4] and discuss the relationship between each approach. Next, we describe the on-board fuzzy-logic based inference engine and its relationship to the landing area estimate. Finally, we conclude and discuss directions for future work.

In order to make our analysis tractable we make the following assumptions:

A1 : During the terminal descent phase, the spacecraft is modeled as an unconstrained point mass with a full three translational degrees-of-freedom.

A2 : The curvature of Mars is neglected. As a result, the local gravity vector is constant in both magnitude and direction.

A3 : The rotation of Mars is ignored over the time scale in which the terminal descent phase occurs (20-30 seconds). As a result, the Mars surface-fixed frame (MSF) serves as an approximate inertial frame for this analysis.

A4 : The spacecraft thrusters can exert an ideal impulse in any direction thus providing an instantaneous change of the spacecraft velocity vector (i.e., instantaneous ΔV).

A5 : Only a finite amount of ΔV is available for the terminal descent phase. The total amount of ΔV allocated (denoted by ΔV_a) provides a direct measure of the mass of propellant required to realize a soft landing via the *rocket equation* [6]

$$\Delta V_a = -gI_{sp}\ln(\frac{m_0}{m}) \tag{1}$$

Here g denotes the magnitude of the local gravity vector, I_{sp} denotes the *specific impulse* of the thrusters, m_0 is the mass of the vehicle prior to the burn, and m is the mass of the spacecraft after the burn. If multiple impulsive burns are

required the rocket equation is used in succession to determine the total ΔV penalty.

A6 : Expect gravity, all environmental effects including atmospheric drag and wind disturbances are neglected during the terminal descent phase.

A7 : The mass of the spacecraft does not significantly vary over the terminal descent.

A8 : Variations (local slope/roughness) in the Martian surface terrain are neglected for the landing area calculation. As a result, all points on the Martian surface reside in the z=0 plane.

We will re-examine the above assumptions in greater detail in the sequel.

2 Determination of the Landing Footprint

In this section our goal is to compute an estimate of the reachable landing area associated with a spacecraft during the terminal (post-parachute) landing phase of a Mars soft landing. As discussed above, the algorithms developed here can also be invoked during the subsonic parachute phase (altitudes > 1km) to estimate the landing footprint before the parachute is jettisoned at 1km and control authority is regained.

2.1 Determination of the Landing Footprint via Ballistics

In this section we apply the theory of elementary ballistics to estimate the landing footprint of the lander. In a later section we will recompute the footprint by integrating a representative set of the guidance equations. Specifically, we assume that the spacecraft's nominal trajectory is a ballistic (free-fall) trajectory as shown in Figure 1. The terminal point of the nominal ballistic trajectory, denoted 0_N, is the nominal landing site.[3] This idealized terminal descent scenario is justified under the assumption that the descent thrusters can provide instantaneous changes in the spacecraft velocity. As a result, we assume that the lander remains in free-fall until it is directly over the desired landing site. At the instant before impact the total accumulated velocity vector is taken out by commanding the appropriate thrust vector required to realize a soft landing[4]. This situation is a useful *approximation* of the actual scenario (i.e., where the vehicle follows a controlled guidance generated trajectory) that allows an analytical solution to be obtained for the landing footprint. In the sequel we will compare the size of the landing area obtained via ballistic arguments to the exact landing area predicted by the guidance equations.

[3]As discussed above, the nominal landing site is determined by an on-board autonomous reasoning engine during the subsonic parachute phase. A correction must be performed immediately after the parachute phase to set up the proper initial conditions needed to reach the target.

[4]Here we assume that the nominal ballistic trajectory has been initialized so that sufficient fuel is available to realize a soft landing at the nominal target.

Figure 1 provides a detailed illustration of the terminal landing scenario. In order to describe the motion of the spacecraft relative to the surface of the planet, we introduce a Mars surface-fixed frame $\mathcal{F}_{MSF} = [\vec{i}, \vec{j}, \vec{k}]$ located at a specified point on the surface. It follows from neglecting the rotation of Mars (see A3), that the surface-fixed frame $\mathcal{F}_{MSF}$ provides an *approximate* inertial frame of reference.

As discussed above the nominal trajectory of the lander is assumed to be ballistic; as a result, the motion of the system is described by the standard equations

$$m\vec{a} = m\vec{g} \tag{2}$$

$$= -mg\vec{k} \tag{3}$$

where g denotes the magnitude of the local gravity vector in the vicinity of the planet's surface. By assumption A6 we have neglected atmospheric drag and other perturbations in our analysis. As a result, gravity is the only external force assumed to act on the lander between impulsive thrustings. Integrating (3), we find

$$\vec{v}(t) = \vec{v}_0 - gt\vec{k} \tag{4}$$

where $\vec{v}_0$ denotes the initial velocity of the spacecraft at the beginning of the terminal descent phase. Resolving (4) in $\mathcal{F}_{MSF}$, we find

$$\vec{v}(t) = \begin{bmatrix} v_{x0} \\ v_{y0} \\ v_{z0} - gt \end{bmatrix} \tag{5}$$

It can be seen from Figure 1 that the nominal motion of the lander takes place in a single plane determined by its resultant horizontal velocity vector $\vec{v}_H$ and the vertical velocity vector $\vec{v}_z$. As a result, we decompose the velocity vector (4) in the following manner

$$\vec{v}(t) = \vec{v}_H + \vec{v}_z(t) \tag{6}$$

where the *constant* horizontal velocity vector $\vec{v}_H$ resolved in $\mathcal{F}_{MSF}$ is given by

$$\vec{v}_H = \begin{bmatrix} v_{x0} \\ v_{y0} \\ 0 \end{bmatrix} \tag{7}$$

and the vertical velocity vector (resolved in $\mathcal{F}_{MSF}$) is given by

$$\vec{v}_z(t) = \begin{bmatrix} 0 \\ 0 \\ v_{z0} - gt \end{bmatrix} \tag{8}$$

The geometry of the nominal trajectory is shown in Figure 1. The direction determined by the horizontal velocity vector $\vec{v}_H$ is called the (nominal) *downrange* and is denoted by the unit vector $\vec{e}_1$. Note that the downrange direction makes an angle of

$$\phi = \arctan(\frac{v_{y0}}{v_{x0}}) \tag{9}$$

relative to the x-axis of $\mathcal{F}_{MSF}$. It is easily shown that the horizontal velocity vector can be expressed in terms of the downrange as follows

$$\begin{aligned}
\vec{v}_H &= v_{x0}\vec{i} + v_{y0}\vec{j} && (10)\\
&= v_H\left(\frac{v_{x0}}{v_H}\vec{i} + \frac{v_{y0}}{v_H}\vec{j}\right) && (11)\\
&= v_H(\cos\phi\vec{i} + \sin\phi\vec{j}) && (12)\\
&= v_H\vec{e}_1 && (13)
\end{aligned}$$

where $v_H = \sqrt{v_{x0}^2 + v_{y0}^2}$ is the magnitude of $\vec{v}_H$. The direction orthogonal to the (nominal) downrange is the (nominal) *crossrange* and is given by the unit vector $\vec{e}_2$ (also shown in Figure 1). Here $\vec{e}_2$ is chosen to complete the right handed triad associated with $\vec{e}_1$ and $\vec{e}_3 = \vec{k}$.

Integrating (3) a second time yields

$$\vec{r} = \vec{r}_0 + \vec{v}_0 t - \frac{1}{2}gt^2\vec{k} \tag{14}$$

where $\vec{r}_0$ denotes the initial position of the lander (relative to the origin of $\mathcal{F}_{MSF}$) at the beginning of the terminal descent phase. Resolving (14) in $\mathcal{F}_{MSF}$ results in

$$\vec{r}(t) = \begin{bmatrix} x(t) \\ y(t) \\ z(t) \end{bmatrix} = \begin{bmatrix} x_0 + v_{x0}t \\ y_0 + v_{y0}t \\ z_0 + v_{z0}t - \frac{1}{2}gt^2 \end{bmatrix} \tag{15}$$

The time taken for the spacecraft to impact the surface along its nominal ballistic trajectory is found by setting $z = 0$ in (15) and solving for the positive root

$$t_{go} = \frac{v_{z0} + \sqrt{v_{z0}^2 + 2gz_0}}{g} \tag{16}$$

Note that in $\mathcal{F}_{MSF}$ the nominal landing site $\vec{R}_N$ is given by

$$\vec{R}_N = \begin{bmatrix} x_N \\ y_N \\ z_N \end{bmatrix} = \begin{bmatrix} x_0 + v_{x0}t_{go} \\ y_0 + v_{y0}t_{go} \\ 0 \end{bmatrix} \tag{17}$$

Another useful expression for the nominal landing site relative to the origin of $\mathcal{F}_{MSF}$ is given by (see Figure 1)

$$\vec{R}_N = \vec{R}_P(t) + (v_H\Delta t)\vec{e}_1 \tag{18}$$

where $\vec{R}_P(t)$ denotes the instantaneous projection of the position vector on the surface of Mars and $\Delta t = t_{go} - t$. Note that multiplying the magnitude of the horizontal velocity by the time remaining until impact results in the distance to go along the downrange direction. A useful result follows from evaluating (18) at $t = 0$;

$$\begin{aligned}
\vec{R}_N &= \vec{R}_{P0} + (v_H t_{go})\vec{e}_1 && (19)\\
&= \vec{R}_{P0} + (\sqrt{v_{x0}^2 + v_{y0}^2}\,t_{go})\vec{e}_1 && (20)
\end{aligned}$$

where $\vec{R}_{P0}$ is the projection of $\vec{r}_0$ on the surface of Mars at the start of the terminal descent phase (i.e., $\vec{R}_{P0} = \vec{R}_P(0)$)

We now explain the main idea underlying our ballistics-based algorithm for determining the landing footprint. Figures 2 and 3 provide a detailed pictorial representation of the method. The key is to characterize the set of ballistic trajectories at each time (centered about the nominal ballistic trajectory) that the vehicle *could* be made to follow while still maintaining the overall ΔV constraint. Specifically, we ask how much ΔV, say $||\Delta\vec{V}_1||$, can be added to the current nominal horizontal velocity vector $\vec{V}_H$ such that the vehicle still conserves enough ΔV to realize a soft-landing. (Here we denote the necessary ΔV to ensure a soft-landing by $||\Delta\vec{V}_2||$.) Within the ballistic framework a potential landing site is considered feasible only if there exists a ballistic trajectory connecting the current lander position to the candidate target point that can be generated by a **two impulse maneuver** satisfying the inequality constraint

$$\Delta V_1 + \Delta V_2 \leq \Delta V_a \tag{21}$$

Here we have introduced the notation $\Delta V_1 := ||\Delta\vec{V}_1||$, $\Delta V_2 := ||\Delta\vec{V}_2||$, and ΔV_a denotes the allocated ΔV as dictated by the amount of available propellant. The boundary of the landing footprint (i.e., the set of all feasible target sites resulting in total fuel depletion) can be determined by studying the solutions of

$$\Delta V_1 + \Delta V_2 = \Delta V_a \tag{22}$$

As an aside, it is important to remember that the two-impulse maneuver used to determine the landing footprint occurs in software (not in reality) at each time during the nominal descent. Only if on-board intelligence deems the nominal landing site unsafe will an actual retargeting (and hence an actual two-impulse maneuver) occur.

We now develop explicit expressions for each term in (21). Recall from (13) that $\vec{v}_H(t^-) = v_H\vec{e}_1$ where the notation t^- denotes the time immediately before the first impulse is applied. We assume that the first impulsive maneuver imparts a vectorial ΔV of the following form (see Figure 2)

$$\begin{aligned} \Delta\vec{V}_1 &= \Delta V_1\vec{e} && (23)\\ &= \Delta V_1\cos\theta\vec{e}_1 + \Delta V_1\sin\theta\vec{e}_2 && (24) \end{aligned}$$

where $\vec{e}$ denotes the direction associated with $\Delta\vec{V}_1$. Note that the first impulsive maneuver in the horizontal plane *does not* change the vertical velocity vector. Furthermore, (see A4) we assume that the thrust vector can be applied in any direction in the horizontal plane.

At the instant after the first impulsive maneuver is realized, denoted t^+, the horizontal velocity becomes

$$\begin{aligned} \vec{v}_H(t^+) &= \vec{v}_H(t^-) + \Delta\vec{V}_1 && (25)\\ &= (v_H + \Delta V_1\cos\theta)\vec{e}_1 + \Delta V_1\sin\theta\vec{e}_2 && (26) \end{aligned}$$

The perturbed downrange motion of the vehicle (if the first impulse were actually realized in hardware) would occur in the direction determined by the unit vector $\vec{e}_1^+$.

(See Figures 2 and 3.) Similarly, the total spacecraft velocity after the first impulse is realized is given by

$$\begin{aligned} \vec{v}(t^+) &= \vec{v}_H(t^+) + \vec{v}_z(t) && (27) \\ &= (v_H + \Delta V_1 \cos\theta)\vec{e}_1 + \Delta V_1 \sin\theta \vec{e}_2 - v_z \vec{e}_3 && (28) \end{aligned}$$

Once the resultant spacecraft velocity is known after the first impulsive maneuver has been delivered, we can immediately determine the magnitude of the second impulse required to ensure a soft landing via energy methods. Specifically, it follows from conservation of total mechanical energy that

$$\frac{1}{2}m||\vec{v}(t^+)||^2 + mgz(t) = \frac{1}{2}m||\vec{v}(t_{go}^-)||^2 \tag{29}$$

where t_{go}^- denotes the time immediately before impact, m is the mass of the lander, and $z(t)$ denotes the altitude of the lander at the current time[5]. Note that we have assumed the potential energy at touchdown is zero by A8. However, if the spacecraft lands on a hill or in a crater the potential energy is $U(t_{go}^-) = \pm mgz_{t_{go}}$ and the above analysis must be modified. The objective is to determine the magnitude of the velocity $\vec{v}(t_{go}^-)$ of the spacecraft immediately before impact. This velocity must be cancelled by the second impulsive firing to ensure a soft landing. To this end, it follows from rearranging (29) that the velocity immediately before impact is

$$||\vec{v}(t_{go}^-)|| = \sqrt{||\vec{v}(t^+)||^2 + 2gz} \tag{30}$$

Substituting (28) into (30) and recalling that $v_H^2 = v_x^2 + v_y^2$ we find

$$||\vec{v}(t_{go}^-)|| = \sqrt{\Delta V_1^2 + 2\Delta V_1 v_H \cos\theta + v_x^2 + v_y^2 + v_z^2 + 2gz} \tag{31}$$

The energy associated with the nominal ballistic trajectory is

$$E = \frac{1}{2}m(v_x^2 + v_y^2 + v_z^2) + mgz \tag{32}$$

and it follows that (31) can be expressed

$$||\vec{v}(t_{go}^-)|| = \sqrt{\Delta V_1^2 + 2\Delta V_1 v_H \cos\theta + 2E/m} \tag{33}$$

As discussed above, the magnitude of the velocity that the propulsion system must take out before impact is $||\vec{v}(t_{go}^-)||$ (i.e., we require $\vec{v}(t_{go}^+) = \vec{0}$ for a soft landing). As a result, it follows that ΔV_2 in equation (21) is given by

$$\Delta V_2 = ||\vec{v}(t_{go}^-)|| \tag{34}$$

Upon substituting (24) and (33) into the inequality constraint (21) we obtain the important relationship

$$\Delta V_1 + \sqrt{\Delta V_1^2 + 2\Delta V_1 v_H \cos\theta + \frac{2E}{m}} \leq \Delta V_a \tag{35}$$

[5]Note that since the vertical velocity is not affected by the first maneuver we do not distinguish between t^- and t^+ in the vertical direction.

where $\Delta V_1 \geq 0$ since it represents the magnitude of a vector. In order to determine the boundary of the reachable landing area we determine the maximum allowable ΔV_1 (call this ΔV_1^*) consistent with the overall fuel constraint. In other words, we seek the solution ΔV_1^* of the nonlinear equation

$$\Delta V_1^* + \sqrt{\Delta V_1^{*2} + 2\Delta V_1^* v_H \cos\theta + \frac{2E}{m}} = \Delta V_a \tag{36}$$

where we must enforce $\Delta V_1^* \geq 0$. After some manipulation it follows from (36) that

$$\Delta V_1^* = \frac{\Delta V_a^2 - \frac{2E}{m}}{2\Delta V_a + 2v_H \cos\theta} \tag{37}$$

Note that ΔV_1^* is a function of θ. Recall that velocity is converted to range information by multiplying by the time remaining until impact $\Delta t = t_{go} - t$. As a result, the distance to the boundary of the reachable area relative to 0_N is given by

$$\rho = \Delta V_1^* \Delta t \tag{38}$$

$$= \frac{(\Delta V_a^2 - \frac{2E}{m})\Delta t}{2\Delta V_a + 2v_H \cos\theta} \tag{39}$$

It is convenient to express (39) as follows

$$\rho = \frac{\frac{(\Delta V_a^2 - \frac{2E}{m})\Delta t}{2\Delta V_a}}{1 + \frac{v_H}{\Delta V_a}\cos\theta} \tag{40}$$

Equation (40) represents a conic section expressed in polar coordinates [8]

$$\rho = \frac{p}{1 + e\cos\theta} \tag{41}$$

where the *eccentricity* is given by

$$e = \frac{v_H}{\Delta V_a} \tag{42}$$

and the *semilatus rectum* is

$$p = \frac{(\Delta V_a^2 - \frac{2E}{m})\Delta t}{2\Delta V_a} \tag{43}$$

Here θ denotes the polar angle. By the definition of a conic section in polar coordinates the point from which ρ is measured (in this case 0_N) is a focus of the conic. The numerical value of the eccentricity e characterizes the type of conic section represented by (41). To this end, recall that $\Delta V_1^* \geq 0$. From (40) this implies that

$$\Delta V_a \geq \sqrt{\frac{2E}{m}} \tag{44}$$

Recalling that

$$E = \frac{1}{2}m(v_z^2 + v_H^2) + mgz \tag{45}$$

we find

$$\Delta V_a \geq \sqrt{v_H^2 + v_z^2 + 2gz} \tag{46}$$

Note that (46) implies that $\Delta V_a > v_H$ (this also enforces that the denominator of (40) is positive). As a result

$$e = \frac{v_H}{\Delta V_a} < 1 \tag{47}$$

Hence in our ballistic based terminal landing scenario $0 \leq e < 1$ and it follows that the conic (41) represents an *ellipse* with focus centered at the nominal landing site 0_N. The landing ellipse can also be expressed in Cartesian coordinates relative to the center C of the ellipse (See Figure 4) as follows

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} = 1 \tag{48}$$

where the *semi-major axis* is determined by

$$a = \frac{p}{1 - e^2} \tag{49}$$

and the *semi-minor axis* is

$$b = a\sqrt{1 - e^2} \tag{50}$$

It is apparent from Figure 4 that most of the landing footprint lies *uprange* of the nominal crossrange. This can be explained by noting the θ dependence of ΔV^* as exhibited in (37). We note that the magnitude of the component of $\Delta\vec{V}_1$ that can be added to $\vec{v}_H$ in the downrange direction (i.e., positive $\vec{e}_1$) is limited. (See Figure 3.) For example, any component of $\Delta\vec{V}_1$ in the downrange direction ($-\frac{\pi}{2} < \theta < \frac{\pi}{2}$) increases the horizontal speed of the spacecraft. As a result, a larger terminal velocity develops and a larger fuel penalty is incurred to realize a soft landing. Conversely, if $\Delta\vec{V}_1$ has a component in the -$\vec{e}_1$ direction (i.e., $\frac{\pi}{2} < \theta < \frac{3\pi}{2}$) the horizontal speed is reduced and a lesser fuel penalty is incurred to realize a soft landing.

A MATLAB based simulation environment was developed to study a representative terminal landing scenario. The first simulation (Case 1) was initialized at $x_0 = 0$, $y_0 = 0$ and $z_0 = 500m$ with $v_{x0} = 50m/s$, $v_{y0} = 0$, and $v_{z0} = -50m/s$[6]. The value of g for Mars was taken as $3.69m/s^2$. The resulting ballistic trajectory, along with the time evolution of the landing footprint, is shown in Figure 5. The footprint at t=0 has a semi-major axis of $a = 300m$ and semi-minor axis of $b = 150m$.

An important special case occurs when $e = 0$ (i.e., $v_H = 0$). Here the spacecraft nominal trajectory is a vertical descent directly over the target and it follows from (41) that $\rho = p$. Recall that this represents the polar equation of a circle. As a result, the reachable landing area at any time is a circle centered at 0_N with radius

$$\rho = \frac{(\Delta V_a^2 - \frac{2E}{m})\Delta t}{2\Delta V_a} \tag{51}$$

[6]Here the initial conditions are commensurate with those anticipated for a future Mars lander after the sub-sonic parachute has been jettisoned.

The next simulation case (Case 2) was initialized at $x_0 = 0$, $y_0 = 0$ and $z_0 = 500[m]$ with zero horizontal velocity (i.e., $v_{x0} = v_{y0} = 0$) and $v_{z0} = -50[m/s]$. The trajectory is a vertical descent over the desired target and the resulting landing footprint is circular as shown in Figure 6. The footprint has a maximum radius of $200[m]$ at $t = 0$.

The distance to the boundary of the landing footprint (41) has been given relative to the nominal landing site 0_N. From Figure 2 (see also equation (18)) it follows that the boundary of the landing footprint relative to the origin of $\mathcal{F}_{MSF}$ is given by

$$\begin{aligned} \vec{R} &= \vec{R}_N + \rho\vec{e} && (52) \\ &= \vec{R}_P(t) + (v_H \Delta t)\vec{e}_1 + \rho\vec{e} && (53) \\ &= \vec{R}_{P0} + (v_H t_{go})\vec{e}_1 + \rho\vec{e} && (54) \end{aligned}$$

where $\vec{e} = cos\theta\vec{e}_1 + sin\theta\vec{e}_2$.

The ballistic-based methodology discussed above is used to construct an on-board estimate of the landing footprint at any given time. The landing area calculation is performed in *software* during the powered terminal descent phase; only in the case when the on-board hazard detection system detects a risk with the nominal landing site is an actual (i.e., in *hardware*) two-impulse retargeting maneuver performed. At the time when the nominal landing site has been deemed unsafe, the guidance system is provided with an updated target site from the on-board autonomous reasoning engine[7]. The thrust vector must then be re-oriented in the horizontal plane to realize the desired retargeting. The new landing site $\vec{R}_N^*$ is given relative to the origin of $\mathcal{F}_{MSF}$ as

$$\vec{R}_N^* = \vec{R}_N + \vec{r}_T \tag{55}$$

where (relative to the MSF frame) $\vec{r}_T = \vec{R}_N^* - \vec{R}_N = [x_T,\ y_T]^T$. The thrust angle θ_T required for the retargeting is

$$\theta_T = \arctan(\frac{y_T}{x_T}) \tag{56}$$

The geometry of retargting is shown in Figure 7. The magnitude of the first impulse required, $||\Delta\vec{V}_1||$, must be determined. Recalling that the radial distance of the new target from the nominal is related to ΔV_1 as follows

$$r_T = \Delta V_1 \Delta t \tag{57}$$

we find

$$\begin{aligned} \Delta V_1 &= \frac{r_T}{\Delta t} && (58) \\ &= \frac{\sqrt{x_T^2 + y_T^2}}{\Delta t} && (59) \end{aligned}$$

[7]Recall that the hazard detection system will only choose feasible targets; i.e., targets that are both "safe" and lie within the current landing footprint. As the new target lies at an interior point of the landing footprint, it can be reached from the current state with fuel to spare.

Combining (56) and (59) results in

$$\Delta\vec{V}_1 = \frac{\sqrt{x_T^2 + y_T^2}}{\Delta t}(\cos\theta_T\vec{e}_1 + \sin\theta_T\vec{e}_2) \tag{60}$$

$$= \frac{\sqrt{x_T^2 + y_T^2}}{\Delta t}\vec{e}_1^{+} \tag{61}$$

In order to avoid a singularity as $t \rightarrow t_{go}$ (i.e., $\Delta t \rightarrow 0$) we require that all retargetings occur a fixed amount of time before impact. This is not a restrictive assumption as the control authority of the vehicle becomes significantly diminished as the lander approaches the surface.

The third simulation case (Case 3) demonstrates a maneuver involving a retargeting. The lander was initialized at $x_0 = 0$, $y_0 = 0$ and $z_0 = 500[m]$ with $v_{x0} = 10[m/s]$, $v_{y0} = 0$, and $v_{z0} = -50[m/s]$. At approximately one-quarter way through the nominal landing maneuver the on-board hazard detection system determined that the nominal target was unsafe and chose an alternate landing site based on both a safety and fuel metric. (See the next section for more details.) The resulting trajectory and the time evolution of the landing footprint are shown in Figure 8. Note the abrupt change in the trajectory (and hence the landing footprint) after the retargeting occurs.

The above simulation results (Case1 - Case 3) are based on the assumption of negligible atmospheric drag. If a drag law of the form $\vec{F}_a = -b\vec{v}$ is included in the equations of motion it can be shown (see [7]) that the resulting motion falls off faster than the ballistic (parabolic) trajectory and the spacecraft z-velocity component tends toward a terminal value of $v_{z\infty} = \frac{mg}{b}$ after sufficient time-of-flight.

2.2 Determination of the Landing Footprint via Guidance

In a typical planetary landing scenario, the spacecraft will follow a trajectory generated by an on-board guidance law during terminal descent rather than a ballistic trajectory. In this section we determine the size of the reachable landing area obtained by integrating a representative guidance law. Our goal is to ascertain the feasibility of using the ballistic landing area calculation over the more computationally intensive calculation required to determine the footprint by direct integration of the guidance equations in the on-board inference engine.

We will assume that the terminal descent guidance law is generated by specifying a polynomial trajectory connecting the given initial state of the lander at the start of the terminal descent phase (taken as t= 0) to the desired terminal state on (or near) the surface of Mars at time t = T. The degree of the polynomial is chosen large enough to satisfy as many terminal constraints as deemed necessary to ensure a safe landing. Here we will assume that terminal constraints are imposed on the positions and velocities[8] of the lander at touchdown $\vec{r}_T = \vec{r}(T)$ and $\vec{v}_T = \vec{v}(T)$. In MSF relative coordinates the terminal constraints are represented as $r_T = [x_T, y_T, z_T]^T$ and

[8]The acceleration of the lander as well as higher order derivatives such as jerk and snap can also be specified. In this case, a higher order acceleration profile is required.

$v_T = [v_{x_T}, v_{y_T}, v_{z_T}]^T$. As a result, a polynomial trajectory consisting of six unknown coefficients is required. To meet this constraint a *linear* (MSF relative) acceleration profile is specified

$$\vec{a}(t) = \vec{c}_0 + \vec{c}_1 t \tag{62}$$

where $(\vec{c}_0, \vec{c}_1)$ are unknown vector coefficients to be determined. It is important to note that the acceleration profile given above represents the total acceleration (thrust + gravity + other forces) of the spacecraft. Integrating (62) we obtain

$$\vec{v}(t) = \vec{v}_0 + \vec{c}_0 t + \frac{1}{2}\vec{c}_1 t^2 \tag{63}$$

$$\vec{r}(t) = \vec{r}_0 + \vec{v}_0 t + \frac{1}{2}\vec{c}_0 t^2 + \frac{1}{6}\vec{c}_1 t^3 \tag{64}$$

Recalling that $\vec{r}(T) = \vec{r}_T$ and $\vec{v}(T) = \vec{v}_T$ we find

$$T\vec{c}_0 + \frac{T^2}{2}\vec{c}_1 = \vec{v}_T - \vec{v}_0 \tag{65}$$

$$\frac{T^2}{2}\vec{c}_0 + \frac{T^3}{6}\vec{c}_1 = \vec{r}_T - \vec{r}_0 - \vec{v}_0 T \tag{66}$$

Solving the above linear system of equation for the unknown coefficients results in

$$\vec{c}_0 = -\frac{2}{T}(\vec{v}_T - \vec{v}_0) + \frac{6}{T^2}(\vec{r}_T - \vec{r}_0 - \vec{v}_0 T) \tag{67}$$

$$\vec{c}_1 = \frac{6}{T^2}(\vec{v}_T - \vec{v}_0) - \frac{12}{T^3}(\vec{r}_T - \vec{r}_0 - \vec{v}_0 T) \tag{68}$$

To recapitulate, the commanded acceleration profile (62) with coefficients given by (67)-(68) transfers the vehicle from $(\vec{r}_0, \vec{v}_0)$ to $(\vec{r}_T, \vec{v}_T)$ in time T. [9]

We now assume that a nominal landing site (x_n, y_n) on the surface has been chosen so that $\vec{r}_T = \vec{r}_{nT} = (x_n, y_n, 0)$ (See Figure 9). Further we assume that $\vec{v}_T = \vec{v}_{nT} = \vec{0}$ and that the initial conditions of the spacecraft at parachute deploy $(\vec{r}_0, \vec{v}_0)$ are fixed. The nominal trajectory is then given by

$$\vec{a}_n(t) = \vec{c}_{0n} + \vec{c}_{1n} t \tag{69}$$

$$\vec{v}_n(t) = \vec{v}_0 + \vec{c}_{0n} t + \frac{1}{2}\vec{c}_{1n} t^2 \tag{70}$$

$$\vec{r}_n(t) = \vec{r}_0 + \vec{v}_0 t + \frac{1}{2}\vec{c}_{0n} t^2 + \frac{1}{6}\vec{c}_{1n} t^3 \tag{71}$$

where $(\vec{c}_{0n}, \vec{c}_{1n})$ denote $(\vec{c}_0, \vec{c}_1)$ in (67)-(68) evaluated at $\vec{r}_0, \vec{v}_0, \vec{r}_T = \vec{r}_{nT}, \vec{v}_T = \vec{0}$. The acceleration profile $\vec{a}_n$ represents total acceleration of the vehicle; i.e.,

$$\vec{a}_n = \vec{a}_{pn} + \vec{g} \tag{72}$$

[9] In reality the coefficients $(\vec{c}_0, \vec{c}_1)$ in the guidance law must be recomputed several times during the terminal descent to account for the build up of trajectory errors due to winds and other disturbances. However, in this study we assume that the nominal values of the guidance coefficients do not require updating to reach the desired target.

where $\vec{a}_{pn}$ is the thrust acceleration delivered by the propulsion system and $\vec{g}$ is the gravity vector. Here we have neglected all other disturbance forces acting on the lander. The amount of fuel expended to reach the nominal landing site, denoted ΔV_n, is then proportional to the following integral

$$\Delta V_n = \int_0^T \sqrt{\vec{a}_{pn}(t) \cdot \vec{a}_{pn}(t)} dt$$

where $\cdot$ denotes the standard inner product. The fuel capacity of the lander is assumed to be ΔV_a, where $\Delta V_a > \Delta V_n$ so that enough excess fuel is available to retarget during terminal descent if necessary.

We now discuss the computation of the guidance-based landing footprint. To this end consider the geometry of the nominal target shown in Figure 9. The position of the nominal landing site relative to the origin of MSF is denoted by $\vec{r}_{nT}$. In Figure 9 we have introduced the unit vector $\vec{e}_n$ pointing in the direction of $\vec{r}_{nT}$; i.e.,

$$\vec{e}_n = \cos(\psi)\vec{i} + \sin(\psi)\vec{j}$$

where $\psi = \arctan(\frac{y_n}{x_n})$ and $\vec{e}_t = -\sin(\psi)\vec{i} + \cos(\psi)\vec{j}$.

The boundary of the landing footprint can be determined by moving the target outward from the nominal until all current excess fuel is depleted. To this end, we parameterize the target position relative to the nominal in terms of the polar coordinates (ρ, γ) centered at the nominal (See Figure 9)

$$\vec{r}_T = \vec{r}_{nT} + \rho\vec{e}_\rho \tag{73}$$

where

$$\vec{e}_\rho = \cos(\gamma)\vec{e}_n + \sin(\gamma)\vec{e}_t \tag{74}$$

After some manipulation it follows

$$\vec{r}_T = \vec{r}_{nT} + \rho\cos(\psi + \gamma)\vec{i} + \rho\sin(\psi + \gamma)\vec{j} \tag{75}$$

The algorithm used to determine the landing footprint is as follows:

1. Input the current time $t = t^*$

2. Calculate the current amount of fuel remaining

$$\Delta V_a(t^*) = \Delta V_a(t = 0) - \int_0^{t^*} \sqrt{\vec{a}_{pn}(t) \cdot \vec{a}_{pn}(t)} dt$$

3. Evaluate $\vec{c}_0$ and $\vec{c}_1$ in (67)-(68) at $\vec{r}_0 = \vec{r}_n(t^*), \vec{v}_0 = \vec{v}_n(t^*), \vec{r}_T = \vec{r}_{nT} + \rho\cos(\psi + \gamma)\vec{i} + \rho\sin(\psi+\gamma)\vec{j}, \vec{v}_T = \vec{0}$. Denote the resulting guidance coefficients as $\vec{c}_{0r}(\rho, \gamma)$ and $\vec{c}_{1r}(\rho, \gamma)$.

4. Construct the total acceleration profile commensurate with the current spacecraft state and the translated target

$$\vec{a}_r(t) = \vec{c}_{0r}(\rho, \gamma) + \vec{c}_{1r}(\rho, \gamma)t \tag{76}$$

5. Construct the thrust acceleration required to reach the translated target

$$\vec{a}_{pr} = \vec{a}_r(t) - \vec{g} \tag{77}$$

6. Define

$$\Delta V(\rho, \gamma) = \int_{t^*}^{T-t^*} \sqrt{\vec{a}_{pr}(t) \cdot \vec{a}_{pr}(t)} dt \tag{78}$$

7. For each γ between 0 and π, [10] search for the value of ρ, denoted ρ^*, that renders

$$\Delta V(\rho^*, \gamma) = \Delta V_a(t^*) \tag{79}$$

8. The boundary of the landing footprint relative to the origin of MSF is then given at the current instant by

$$\vec{r}_T = \vec{r}_{nT} + \rho^* \cos(\psi + \gamma)\vec{i} + \rho^* \sin(\psi + \gamma)\vec{j} \tag{80}$$

   where $\rho^* = \rho^*(\gamma)$.

9. Update $t \rightarrow t^* + \Delta t_g$ where Δt_g is the guidance update interval and repeat.

The time T allocated to reach the nominal landing site is taken as fixed in the above algorithm. It is important to remember that the spacecraft follows the nominal trajectory $\vec{r}_n(t)$ and the landing footprint calculation is done exclusively in software; only if a retargeting is required are the nominal guidance coefficients $(\vec{c}_{0n}, \vec{c}_{1n})$ and the time to complete the landing T re-initialized to reflect the new initial conditions and modified landing site.

A MATLAB based simulation environment was developed to calculate the landing footprint associated with the guidance equations. In the first case the vehicle initial conditions were taken (relative to MSF) as $r_0 = (0, 0, 500)\ m$, $v_0 = (10, 10, -50)\ m/s$. The time of guided flight was chosen as $T = 12sec$ and $\Delta V_a = 150m/s$. The nominal target was chosen as the intersection of the surface with a ballistic trajectory initialized at (r_0, v_0); i.e., $r_{Tn} = (x_n, y_n, 0)$ where $x_n = x_0 + v_{x0}t_{go}$ and $y_n = y_0 + v_{y0}t_{go}$. The terminal velocity of the lander is assumed to be zero $v_{nT} = (0, 0, 0)m/s$. The landing footprint associated with the guidance equations at the initial time (t=0) is shown in Figure 10 along with the ballistic footprint (also at t=0) based on the same initial conditions and nominal target. Note that the ballistic footprint is in close agreement with the guidance based landing area although it slightly overpredicts the landing area in the region lying aft of the nominal target.

For Case 2 the initial conditions were taken (relative to MSF) as $r_0 = (0, 0, 700)\ m$, $v_0 = (20, 20, -60)\ m/s$. Again the time of guided flight was chosen as $T = 12sec$ with $\Delta V_a = 150m/s$. The nominal target was chosen as the intersection of the surface with a ballistic trajectory initialized at (r_0, v_0) and the terminal velocity of the lander is again taken to be zero $v_{nT} = (0, 0, 0)m/s$. The landing footprint associated with

[10] The landing footprint is symmetric about the $\gamma = 0$ line; as a result, only values of γ between 0 and π need be considered.

the guidance equations at the initial time (t=0) is shown in Figure 11 along with the ballistic footprint (also at t=0) based on the same initial conditions and nominal target. Note that the ballistic landing footprint overpredicts the guidance footprint in this case on the order of 100 m in most directions.

A number of other cases were studied and it was found that the size of the guidance generated landing footprint (relative to the ballistic) was sensitive to the time of guided flight T, the initial state when the parachute is jettisoned, and the selection of the nominal target. In most cases it was observed that the ballistic based landing footprint overpredicted the guidance footprint. Case 2 above is a representative example of this behavior. In the future, it is recommended that any candidate guidance law be carefully studied to ascertain if the ballistic based landing footprint provides an acceptable estimate of the landing footprint for the guidance law at hand. It may also be possible to constrain the form of the terminal guidance law in such a way that the ballistic footprint provides a reasonable estimate. In any case, it may still be desirable to utilize the ballistic algorithm during the parachute phase to provide a computationally efficient method of predicting the landing footprint expected during powered terminal descent. Another goal of future study is to produce a hybrid algorithm that generates the vehicle trajectory via the appropriate guidance logic but uses the more computationally efficient ballistic-based algorithm to compute the landing footprint.

3 Fuzzy Logic Based Quality Index of the Reachable Landing Area

In this section we discuss a fuzzy logic based architecture for characterizing the quality of each potential landing site within the landing footprint[11]. This work is part of an integrated effort at JPL to develop a fuzzy-logic based *autonomous reasoning engine* that enables the spacecraft to continuously asses and evaluate in real-time the terrain of the approaching Martian surface. For the reader unfamiliar with fuzzy logic we provide a brief high-level overview of some basic concepts and ideas; a more detailed treatment can be found in [5].

For our purposes we follow the definition of fuzzy logic given in [5] stating that fuzzy logic provides a framework for representing and implementing a human's heuristic knowledge regarding the properties of a system of interest. A typical fuzzy-logic based analysis is composed of the following four elements:

- A *rule-base* containing a set of if-then type rules containing a fuzzy logic quantification of a human expert's knowledge regarding the system of interest.
- An *inference mechanism* which mimics the expert's decision making process.
- A *fuzzification interface* that converts the inputs into information that the inference mechanism can use to activate and apply rules.

[11]The methodology discussed in this section can be applied to either the guidance or ballistic based footprints. For simplicity we assume that the ballistic footprint has been used.

- A *defuzzification interface* that converts the conclusions of the inference mechanism into actual numerical values.

Figure 12 provides a graphical overview of the above elements.

We now discuss each of the above elements in greater detail. For the landing footprint calculation we are interested in the overall "quality" of feasible landing sites. One issue that we are faced with immediately is to define precisely what is meant by "quality." To this end, we assume that the two variables determining overall landing site quality are (1) distance d of the proposed site from the nominal target (d is a simple parameterization of ΔV), and (2) the overall safeness of the proposed site as measured by a safety index s. Our analysis begins with numerical values (also known as *crisp values*) of each input variable d and s associated with each grid point in the landing footprint. Specifically, the value of distance d is an input that is generated during the landing ellipse calculation as follows:

$$d = \frac{r_T}{\rho} \tag{81}$$

where $r_T = \sqrt{x_T^2 + y_T^2}$ is the distance from the point in question to 0_N and ρ is the distance to the landing footprint boundary from O_N along the target angle θ_T (See Figure 7). As a result, a distance value $d \in [0, 1]$ is obtained for all feasible landing sites. The safeness index s is also an input normalized between 0 and 1 that classifies each landing site based on terrain characteristics such as slope, roughness, density of rocks, density of craters, etc. The output variable of interest is the "quality" of landing site to be discussed further below. The goal here is to develop a quantitative measure of the overall quality q of the landing site from the independent variables d and s .

Next, each input variable is translated from a numerical variable to a *linguistic* variable. A linguistic variable assumes "linguistic values" specified by user-defined linguistic terms. For example, let the linguistic variables DISTANCE and SAFENESS be associated with their numerical counterparts distance and safeness index respectively. The linguistic terms associated with each linguistic variable are then chosen as follows: DISTANCE = "Near", "Close", "Far", or "Distant" while SAFENESS = "Safe", "Moderately Safe", "Moderately Unsafe", or "Unsafe". The quality of each feasible landing site is represented by the linguistic variable QUALITY which is described by the linguistic terms "High","Medium","Low", and "Poor". The goal of the on-board fuzzy-logic based reasoning engine is then to use fuzzy inference to assign a linguistic value to (landing site) QUALITY for each grid point in the landing footprint.

The actual numerical values of the input arguments (d and s) are quantified within the fuzzy logic architecture by so-called *membership functions* [5]. For each input and output linguistic variable a membership function is constructed that represents the degree to which each linguistic variable is described by a particular linguistic term[12].

[12]The membership function serves to define exactly what is meant by the term *fuzzy set*, i.e., a set S that allows *grades* of membership.

For example, four membership functions are required for the four linguistic variables associated with the distance input d. The membership functions used in this study were taken to be trapezoids as they are versatile and easy to implement. However, other choices are possible [5]. The membership functions allow statements such as "The distance is near" or "The distance is close" to be readily quantified. Moreover, as the membership functions for adjacent linguistic values overlap it is possible for a given numerical value of d to be considered both "near" and "close" (with different degrees-of-membership) simultaneously.

We now imagine that an expert has constructed a *rule base* governing the linguistic variables describing the process of interest. Typically, the rules are given in *modus ponens* form, i.e.; **If** *premise* **Then** *consequent*. In the landing area scenario an example rule is: **If** SAFENESS is "Safe" **and** DISTANCE is "Near" **then** QUALITY is "High". The complete rule base is shown in the table below. The linguistic value of QUALITY is found by intersecting the appropriate row of DISTANCE with the appropriate column of SAFENESS.

DISTANCE \ SAFENESS	Safe	Mod-Safe	Mod-Unsafe	Unsafe
Near	High	Medium	Low	Poor
Close	High	Medium	Low	Poor
Far	Medium	Low	Low	Poor
Distant	Poor	Poor	Poor	Poor

A fuzzy inference system is then required to determine which particular set of rules from the above table are *active* during the analysis of each potential landing site[13]. Here the *minimum method* [5] was used to perform premise quantification to decide which conclusions are applicable for the current set of active rules. Essentially, the inference system takes into account the recommendations from each active rule as to what linguistic value should be assigned to the quality of the site in question. The membership functions associated with the landing site quality are then weighted according to the results of the premise quantification. A defuzzification algorithm is then applied to the weighted output membership functions to produce a crisp (i.e., numerical) value for the quality q of the landing site. Typically, defuzzification simply involves the computation of some measure of the area under the weighted output membership functions. Here we we applied the *center of area* method [5] resulting in

$$q = \frac{\int y\mu_a(y)dy}{\int \mu_a(y)dy} \tag{82}$$

where $\mu_a(\cdot)$ denotes the functional form of the entire (weighted) output membership function. To further simplify matters we assume that q has been normalized so that $0 \leq q \leq 1$.

Ultimately, the fuzzy-logic based architecture is equivalent to a nonlinear mapping $f(\cdot,\cdot)$ relating the output q to the inputs d and s; i.e.,

$$q(r_T) = f(d(r_T), s(r_T)) \tag{83}$$

[13]Recall that multiple sets of rules can be active at the same time as the linguistic variables DISTANCE and SAFENESS are permitted to take multiple linguistic values simultaneously.

Here we emphasize that the value for q varies as a function of the proposed landing site r_T within the reachable ellipse. The input-output mapping (83) is called the *control surface* and is shown in Figure 13 [14]. For example, it follows from Figure 13 that if the SAFENESS of a landing site is 0.8 and the (normalized) DISTANCE is 0.6 then the overall numerical value for quality is 0.466.

4 Conclusions

In this report we have developed an algorithm based on ballistics to estimate the landing footprint during the powered terminal descent phase of a Mars soft landing. Our algorithm leads to an analytical expression for the boundary of the landing footprint under the assumptions that the vehicle trajectory can be approximated by a ballistic trajectory and that on-board thrusters can realize ideal impulsive maneuvers. In order to ascertain the validity of these modeling assumptions on a realistic terminal landing scenario we compared the landing footprint generated from the ballistic analysis to the landing footprint resulting from directly integrating a representative guidance law. Our study suggests that the ballistic approach overpredicted the landing footprint associated with a guidance generated spacecraft trajectory. However, there were cases where the ballistic assumption provided a landing footprint in very close agreement to the guidance generated footprint. It was observed that the size of the guidance footprint relative to the ballistic footprint was sensitive to the time of guided flight T, and the initial state at parachute separation relative to the nominal target.

One goal of future study is to produce a hybrid algorithm that generates the vehicle trajectory via the appropriate guidance logic but uses the explicit ballistic-based algorithm to compute the landing footprint. In other words, a ballistic based analysis centered about the current guidance-generated spacecraft state is performed periodically along the nominal vehicle trajectory. Another potential application of the ballistic algorithm is to use it to provide a computationally efficient prediction of the terminal landing footprint while the lander is on the parachute. It is also crucial to study the effects of wind and other environmental disturbances in a future analysis. Further, a 6DOF study including lander attitude dynamics should also be undertaken.

In the last section of this report we described an architecture based on fuzzy-logic to combine information about the landing footprint with information about the safeness of the Martian terrain to construct a landing site quality index. The landing site quality index is a critical parameter that will enable the vehicle to make intelligent real-time decisions about landing safely on unknown and possibly hazardous planetary terrains. It can be argued that the ballistic-generated reachable area calculation is sufficient for designing a fuzzy-logic based on-board inference engine for determining landing site quality. For example, a more conservative estimate of the landing area size can be obtained directly from the ballistic-generated footprint by proper tuning of the linguistic values/membership functions (associated with DISTANCE) as well as

[14] Here the MATLAB fuzzy logic toolbox was used to construct the control surface.

the rule base underlying the fuzzy inference engine. Further, any a priori knowledge (e.g., from off-line simulation) regarding the size of the guidance generated landing footprint relative to the ballistic footprint should be utilized to weight (reduce) an on-board ballistic estimate of the landing area before it is provided to the inference engine.

5 Acknowledgments

This research was performed at the Jet Propulsion Laboratory, California Institute of Technology, under contract with the National Aeronautics and Space Administration. The authors would like to thank Dr. Daniel P. Scharf of JPL for carefully proofreading this report and providing valuable feedback.

References

[1] G.W. Cherry, "E-Guidance - A general explicit, optimizing guidance law for rocket-propelled spacecraft", *MIT Instrumentation Laboratory, R-456, 1964.*

[2] A. Howard, and H. Seraji, "A fuzzy rule-based safety index for landing site risk assessment", *Proc. WAC* 9th *Int. Sump. on Robotics with Applications.* Orlando, Florida, June 9-13, 2002.

[3] P.C. Hughes, *Spacecraft Attitude Dynamics.* John Wiley, New York, 1986.

[4] A.R. Klumpp, "Apollo lunar-descent guidance", *MIT Charles Stark Draper Laboratory, R-695, 1971.*

[5] K.M. Passino, and S. Yurkovich, *Fuzzy Control.* Addison-Wesley, Reading; Mass., 1998.

[6] J.E. Prussing, and B.A. Conway, *Orbital Mechanics.* Oxford University Press, New York, 1993.

[7] K.R. Symon, *Mechanics (3rd Edition).* Addison-Wesley, Reading; Mass., 1971.

[8] W.E. Wiesel, *Spaceflight Dynamics (2nd Edition).* McGraw-Hill, New York, 1997.

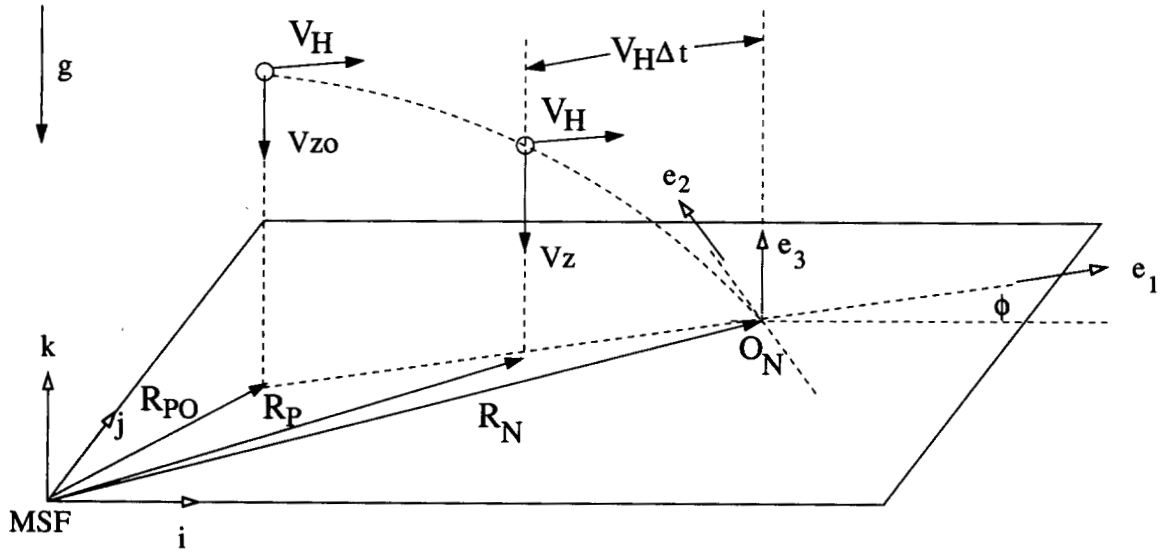


Figure 1: Nominal Ballistic Trajectory

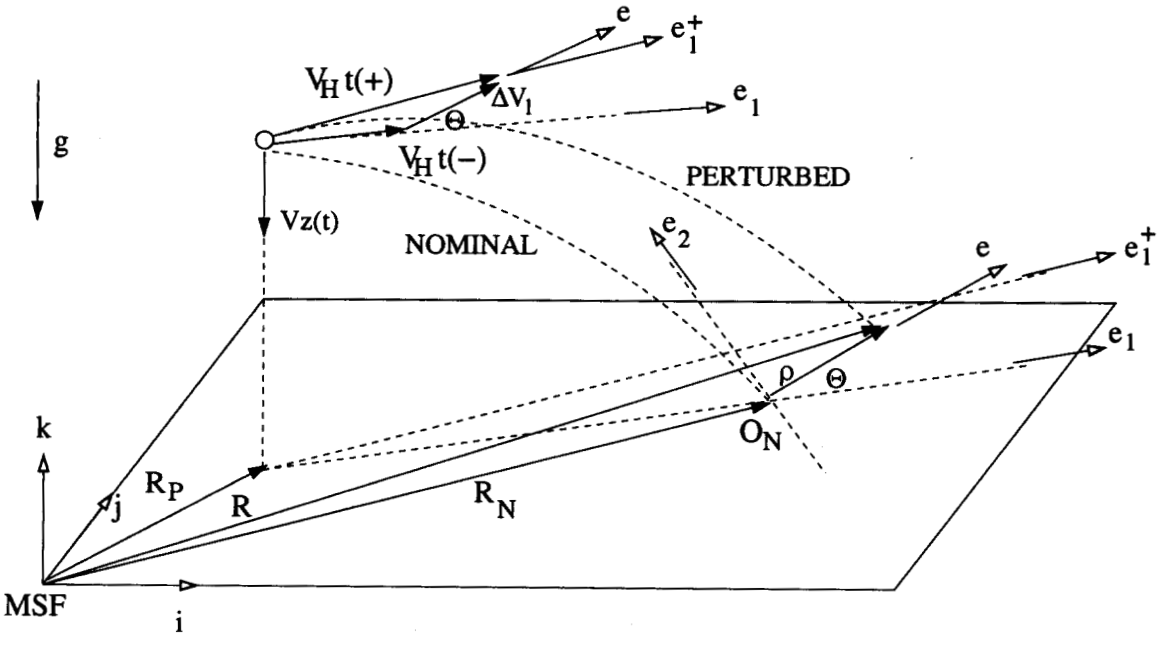


Figure 2: Ballistic Two-Impulse Maneuver

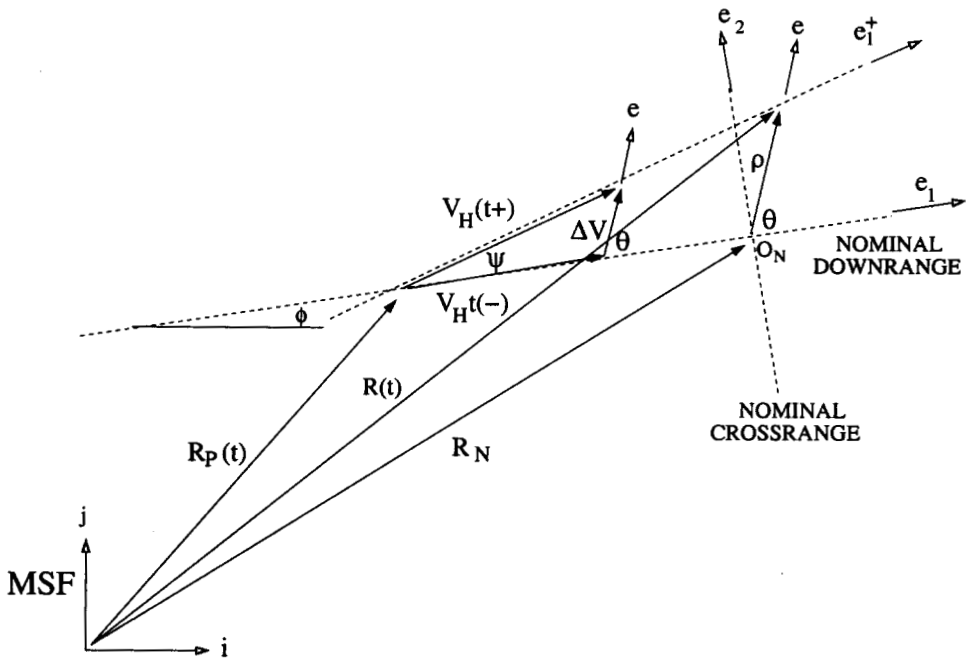


Figure 3: Ballistic Two-Impulse Maneuver

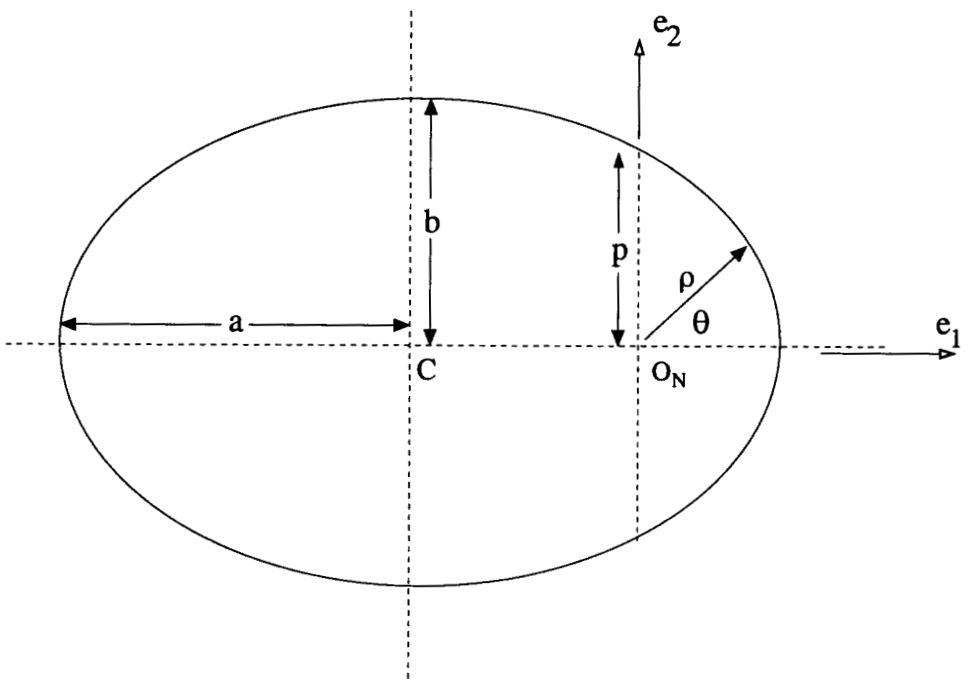


Figure 4: Geometry of Ballistic Landing Footprint

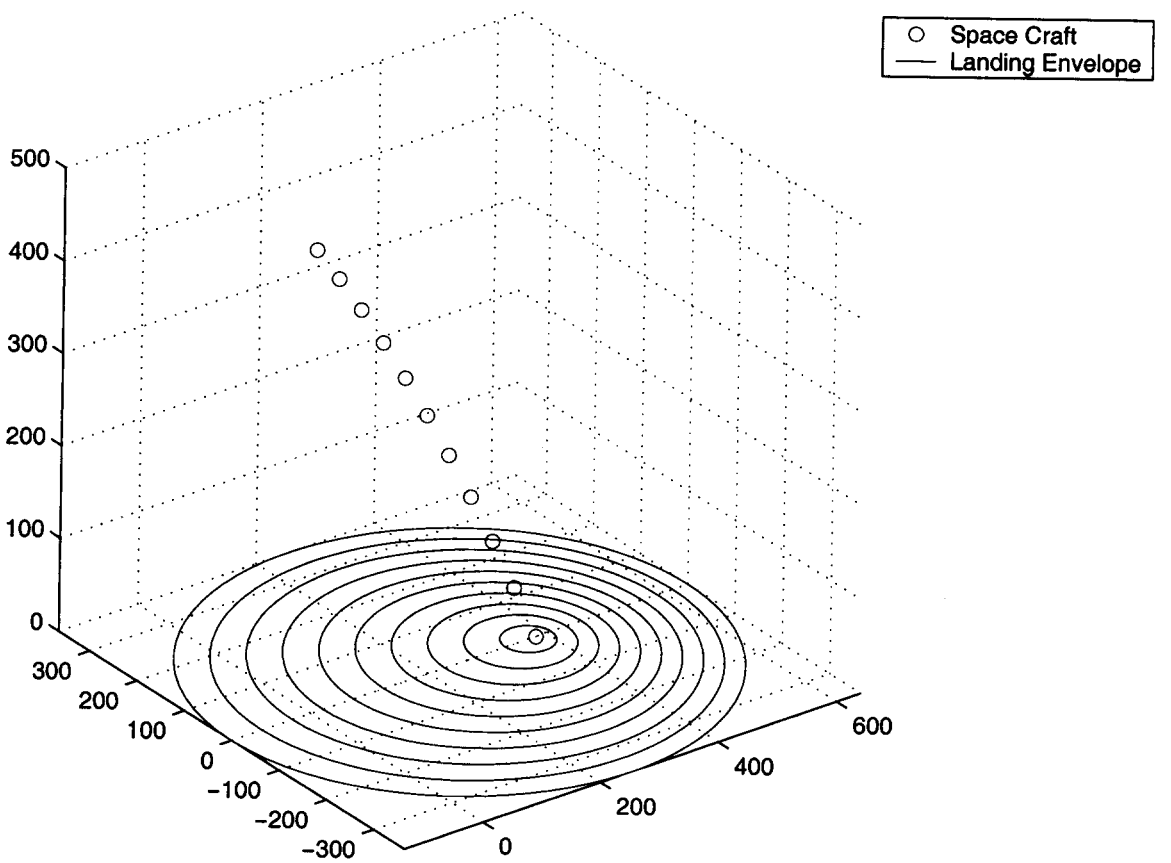


Figure 5: Case 1 - Ballistic w/ Initial Horizontal Velocity

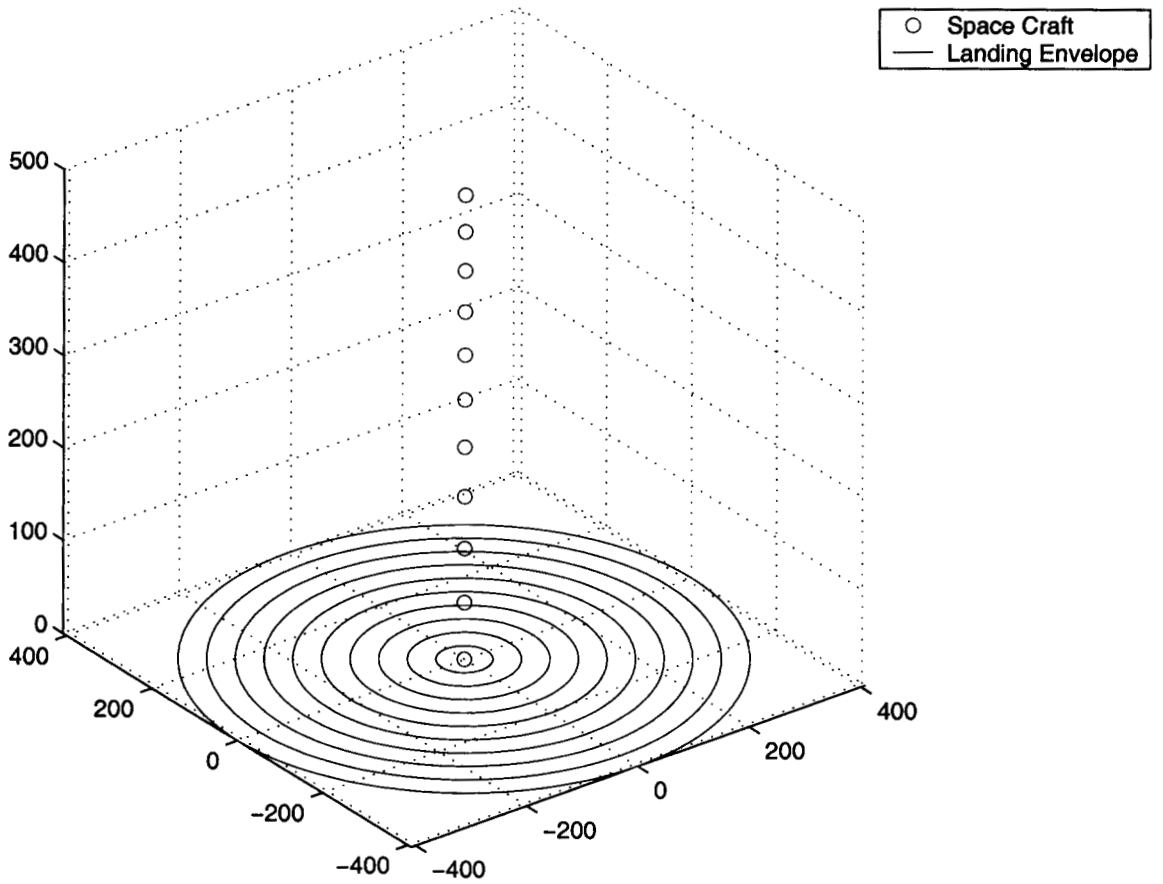


Figure 6: Case 2 - Ballistic w/ No Horizontal Velocity

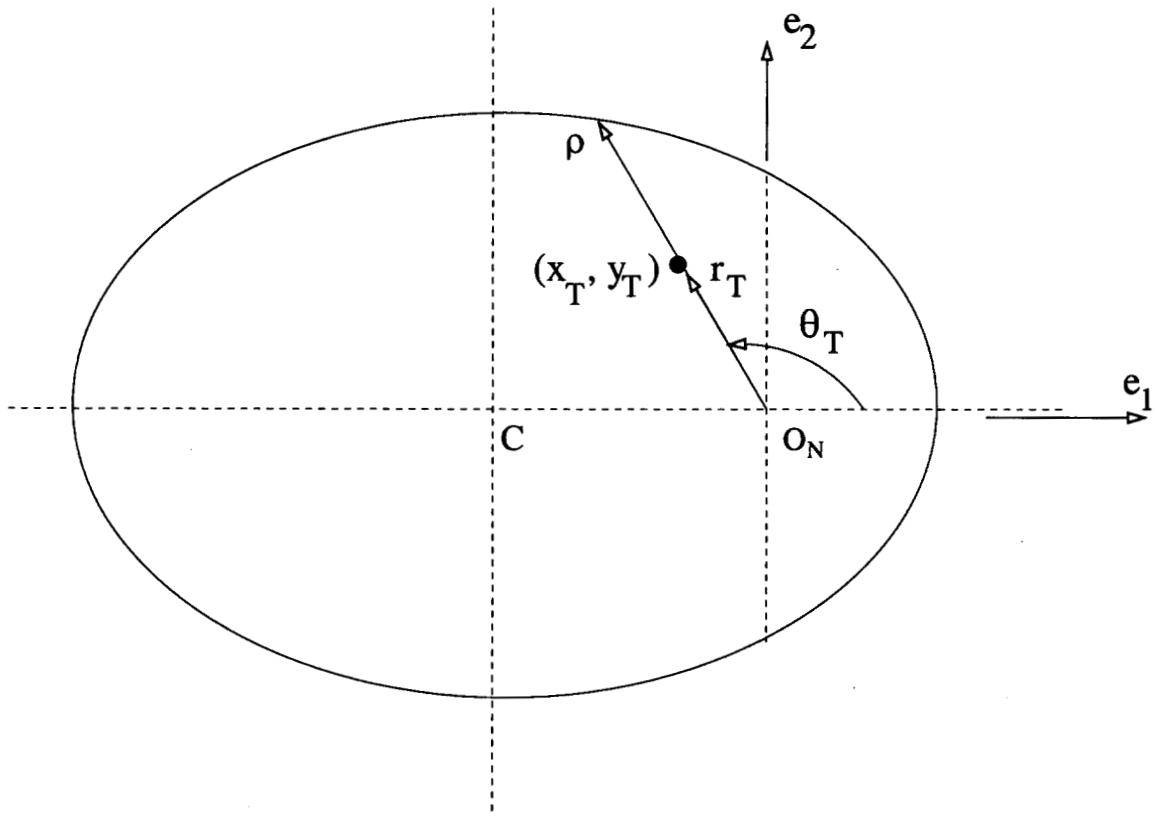


Figure 7: Geometry of Retargeting for Ballistic

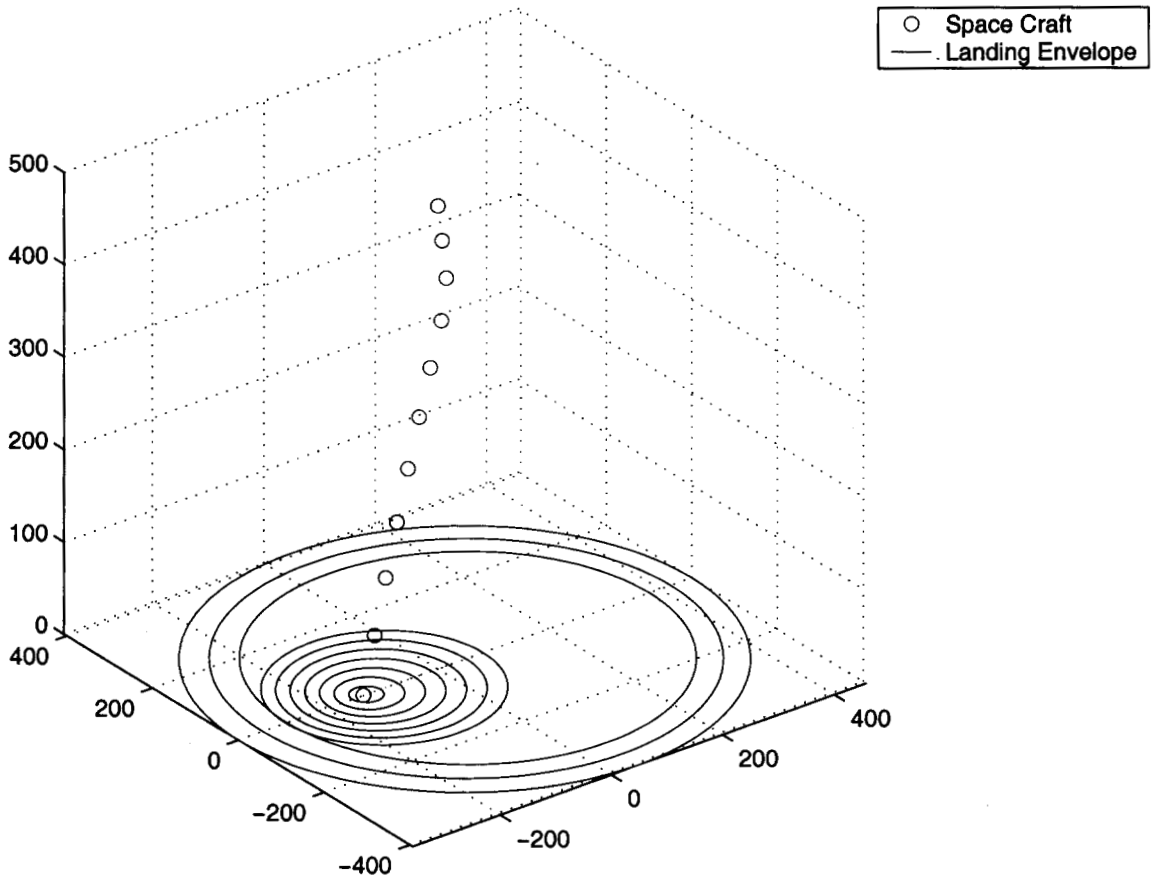


Figure 8: Case 3 - Ballistic w/ Initial Horizontal Velocity w/ Retargeting

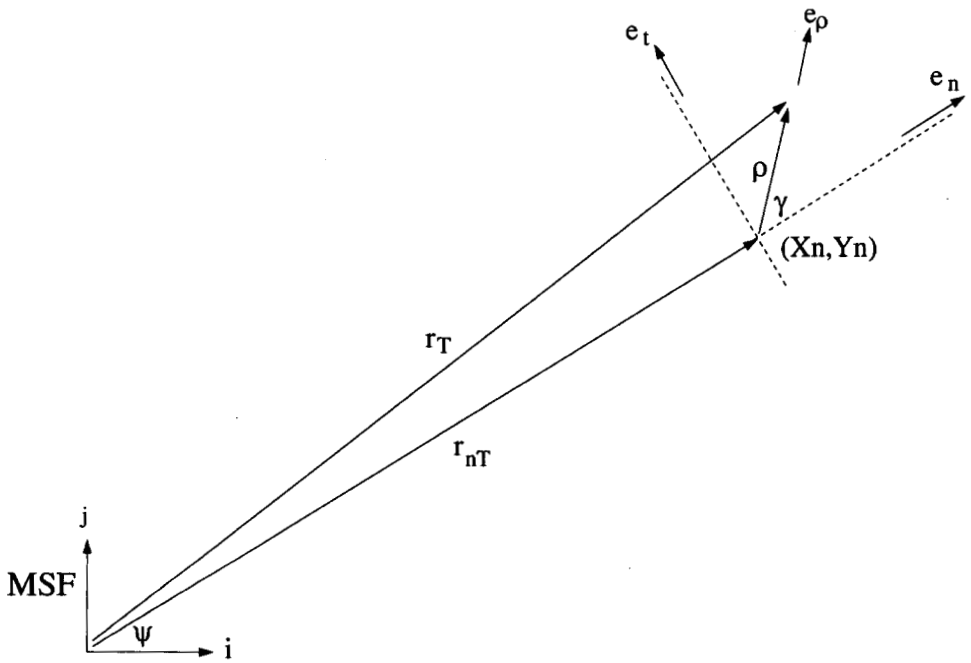


Figure 9: Geometry of the Guidance Based Landing Footprint

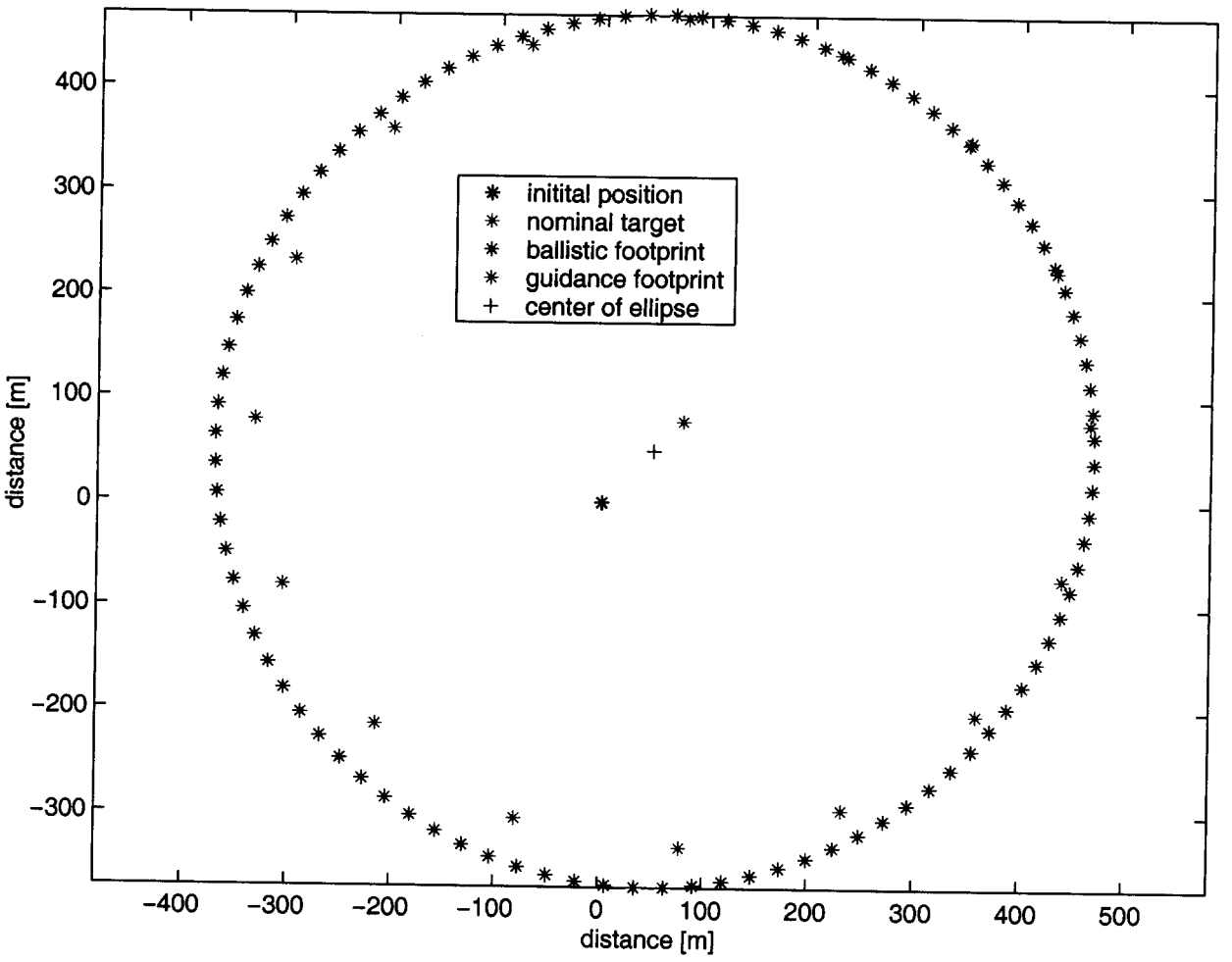


Figure 10: Comparison of Landing Footprints at t=0 - Case 1

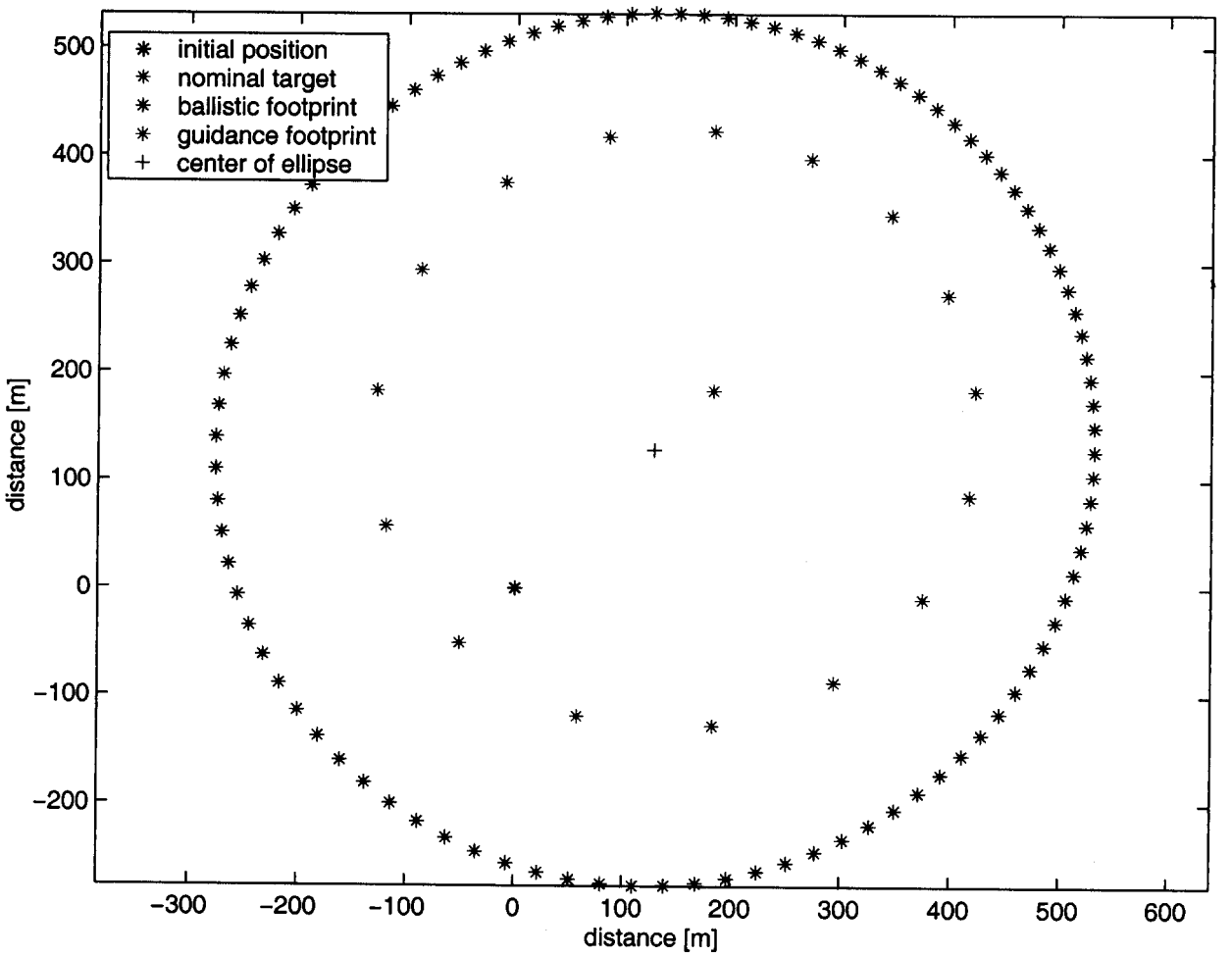
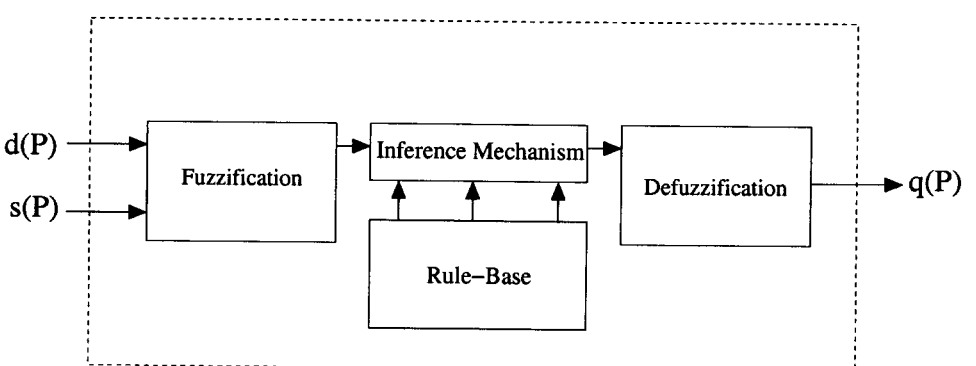


Figure 11: Comparison of Landing Footprints at t=0 - Case 2



$$q(P) = f(d(P),s(P))$$

Figure 12: Fuzzy Logic Overview

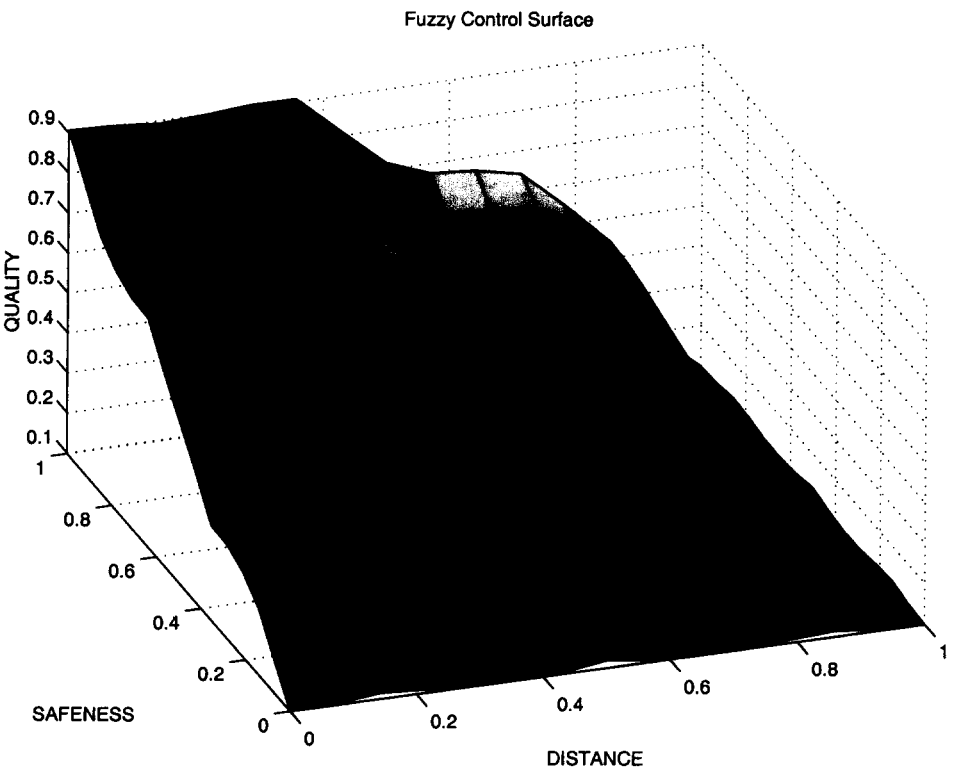


Figure 13: Input-Output Control Surface Based on Ballistics