N90-29069

Satellite-Map Position Estimation for the Mars Rover

Akira Hayashi
Department of Computer Science
University of Texas at Austin
Austin, Texas 78712

Thomas Dean[1]
Department of Computer Science
Brown University
Box 1910, Providence, RI 02912

Abstract

This paper describes a method for locating the Mars rover using an elevation map generated from satellite data. In exploring its environment, the rover is assumed to generate a local rover-centered elevation map that can be used to extract information about the relative position and orientation of landmarks corresponding to local maxima. These landmarks are integrated into a stochastic map which is then matched with the satellite map to obtain an estimate of the robot's current location. The landmarks are not explicitly represented in the satellite map. The results of our matching algorithm correspond to a probabilistic assessment of whether or not the robot is located within a given region of the satellite map. By assigning a probabilistic interpretation to the information stored in the satellite map, we are able to provide a precise characterisation of the results computed by our matching algorithm.

1 Introduction

In the current projections for the Mars Rover project, a satellite is placed in Mars orbit prior to the rover's arrival in order to collect stereo images of the Martian surface with approximately 1 meter resolution. These images are relayed to earth and used to generate a high-resolution elevation map of the regions that the rover is expected to explore. Once the rover has landed on Mars, this elevation map will be used to keep track of the position of the rover and plan out paths for it to follow in performing its exploration of the planet's surface. As the rover moves about, it will use passive-stereo imaging and a laser range finder to construct a depth map of its immediate area. This depth map is then converted into an elevation map which is merged with the map generated from satellite data to provide greater detail.

We will refer to the map generated from satellite data simply as the *satellite map*, and the map generated from local observations made by the rover as the *rover map*. In this paper, we describe a technique for merging these two maps by using the rover map to locate the rover's position in the satellite map. Our method requires that the rover be able to extract the location of *landmarks* from the rover map, where a landmark corresponds to a local maximum (or *peak*) in the surrounding terrain. We assume that the measurements made by the rover are subject to known errors. The relative locations of the landmarks with respect to the rover's current location are stored in *stochastic* map [Smith *et al.*, 1985] that is maintained using the *approximate transform* method of [Smith and Cheeseman, 1986] (see also [Durrant-Whyte, 1988]). We describe an algorithm that provides an estimate of the rover's current location in terms of a probability assignment to fixed regions in the satellite map. For our methods to work, the terrain must have such locally observable features that can be differentiated given the resolution and accuracy of the information stored in the satellite map. The work described in this paper represents a specific application of a general technique first described in [Hayashi and Dean, 1988].

2 Problem Definition

2.1 Satellite Map

In the satellite map, the area of interest is divided into small square regions of the same size referred to as *sectors*. For each sector, the map contains both upper (H^+) and lower (H^-) bounds on the elevation

[1] This work was supported in part by the National Science Foundation under grant IRI-8612644 and by the Advanced Research Projects Agency of the Department of Defense and was monitored by the Air Force Office of Scientific Research under Contract No. F49620-88-C-0132.

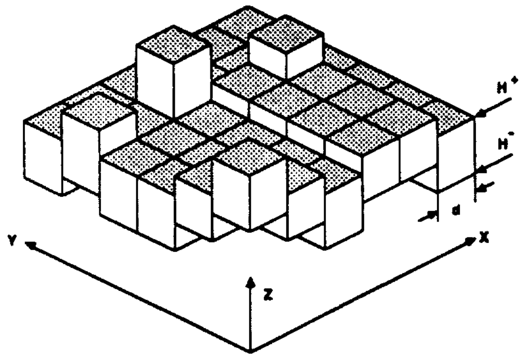

Figure 1: Satellite map

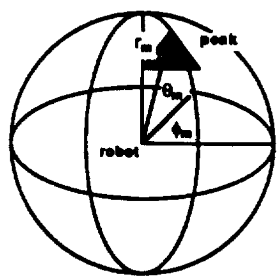

Figure 2: Polar coordinates for vision

within that sector (see Figure 1).

$$H^-_{i,j} \leq Z(X,Y) \leq H^+_{i,j}$$
$$X_i \leq X_{i+1}; X_i = i * d, X_{i+1} = (i+1) * d, 0 \leq i \leq N-1$$
$$Y_j \leq Y_{j+1}; Y_i = j * d, Y_{j+1} = (j+1) * d, 0 \leq j \leq N-1$$

where $Z(X,Y)$ is the actual height at location (X,Y), d is the length of the side of a sector (also referred to as the map's *resolution*), and there are N^2 sectors. $H^-_{i,j}$ and $H^+_{i,j}$ are the only information that the satellite map contains. There are no explicitly specified landmarks. The origin of the satellite map is chosen arbitrarily.

2.2 The Vision System

We assume that the rover has a vision system that can recognize the peaks of hills. The peaks of hills should be the most distinguishable features of a scene. We further assume that the vision system always succeeds in identifying unoccluded peaks in scenes, and that it is capable, with some statistical regularity, of identifying a peak as one that it has seen before. The vision system is not perfect, but the rover has a good approximation of its errors. The values that the vision system returns are the mean (v_m) and the standard deviation (σ_v) of the peak's location in the rover-centered polar coordinate system shown in Figure 2.

$$v_m = \begin{pmatrix} r_m \\ \phi_m \\ \theta_m \end{pmatrix} \qquad \sigma_v = \begin{pmatrix} \sigma_r \\ \sigma_\phi \\ \sigma_\theta \end{pmatrix}$$

where r is the distance to the peak, ϕ is the azimuth angle to the peak in radians, and θ is the elevation angle to the peak in radians. Azimuth angle is measured anti-clockwise from the East (*e.g.*, 0 for East and $\frac{1}{2}\pi$ for North), and the elevation angle is measured anti-clockwise from the horizontal direction. Obviously, a compass and gyro are necessary to make these measurements possible. We assume that each variable, r, ϕ, and θ, forms a mutually independent normal distribution, $N(r|r_m, \sigma_r^2)$, $N(\phi|\phi_m, \sigma_\phi^2)$, $N(\theta|\theta_m, \sigma_\theta^2)$. The notation $N(x|x_m, \sigma_x^2)$ is used as shorthand for a normal distribution of variable x with mean x_m and variance σ_x^2. The mean vector $\mu_V(x,y,z)$ and the covariance matrix $C_V(x,y,z)$ in Cartesian coordinate system will become necessary later and are derived from those in the polar coordinate system as:

$$\mu_V(x,y,z) = \begin{pmatrix} x_m \\ y_m \\ z_m \end{pmatrix} = \begin{pmatrix} r_m * \cos\theta_m * \cos\phi_m \\ r_m * \cos\theta_m * \sin\phi_m \\ r_m * \sin\theta_m \end{pmatrix} \tag{1}$$

$$C_V(x,y,z) = R_3 \cdot C_V(r,\phi,\theta) \cdot R_3^t \tag{2}$$

where

$$C_V(r,\phi,\theta) = \begin{pmatrix} \sigma_r^2 & 0 & 0 \\ 0 & \sigma_\phi^2 & 0 \\ 0 & 0 & \sigma_\theta^2 \end{pmatrix}$$

$$R_3 = \begin{pmatrix} x_m/r_m & -y_m & -z_m * \cos\phi_m \\ y_m/r_m & x_m & -z_m * \sin\phi_m \\ \sin\theta_m & 0 & r_m * \cos\theta_m \end{pmatrix}$$

R_3 is the value of J at the mean (μ_V), where J is the Jacobian of the transformation between the two coordinate systems. See Figure 3 for a visual characterisation of the mean and covariance.

In the figures, we use *certainty ellipsoids* to represent the mean vectors and covariance matrices of our spatial variables. A certainty ellipsoid is the region within which its corresponding spatial variable lies given some probability (say 90%). The center of the ellipsoid is the mean vector. The relationship between a certainty ellipsoid and a covariance matrix is explained in [Smith and Cheeseman, 1986].

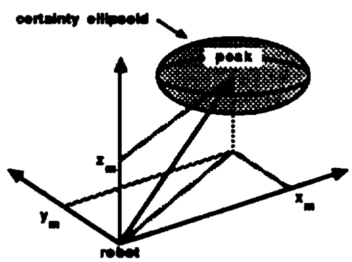

Figure 3: Returned values from the vision system

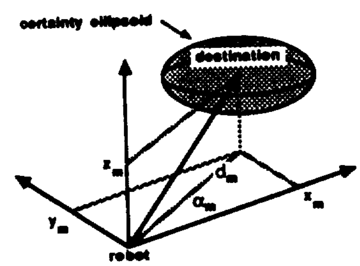

Figure 4: Movement coordinates

2.3 Rover's Movement

The format of the rover's movement command is a pair (d_m, α_m) where d_m is the horisontal distance to a destination (m), and α_m is the asimuth angle to a destination in radians.

To consider movement errors, we assume actual movement (d, α) is obtained using normal distributions, $N(d|d_m, \sigma_d^2)$ and $N(\alpha|\alpha_m, \sigma_\alpha^2)$ respectively. The means of the distributions (d_m and α_m) are the values specified in the movement command. The standard deviations (σ_d and σ_α) are computed from the accuracy of movements. The mean vector $\mu_M(x, y, z)$ and the covariance matrix $C_M(x, y, z)$ in Cartesian coordinate system are:

$$\mu_M(x,y,z) = \begin{pmatrix} x_m \\ y_m \\ z_m \end{pmatrix} = \begin{pmatrix} d_m * \cos\alpha_m \\ d_m * \sin\alpha_m \\ z_m \end{pmatrix} \quad (3)$$

$$C_M(x,y,z) = \begin{pmatrix} C_{xx} & C_{xy} & 0 \\ C_{yx} & C_{yy} & 0 \\ 0 & 0 & C_{zz} \end{pmatrix} \quad (4)$$

where

$$\begin{pmatrix} C_{xx} & C_{xy} \\ C_{yx} & C_{yy} \end{pmatrix} = R_2 \cdot C(d_m, \alpha_m) \cdot R_2^t$$

$$C(d_m, \alpha_m) = \begin{pmatrix} \sigma_d^2 & 0 \\ 0 & \sigma_\alpha^2 \end{pmatrix} \qquad R_2 = \begin{pmatrix} x_m/d_m & -y_m \\ y_m/d_m & x_m \end{pmatrix}$$

$$c_{zz} = \sigma_z^2$$

See the illustration in Figure 4. Note that elevation angle to a destination is not specified in the movement command, since the rover can only move along the surface of the ground no matter whether it is uphill or downhill. Consequently the corresponding mean (z_m) and the standard deviation (σ_z) must be computed separately. They represent the expectation and the variance of the difference of height between the current location and the destination, and therefore depend on the terrain along the movement path (which is not well known).

3 Building an Internal Map

When the rover explores an area, it builds an *internal map* so that it can match the internal map later with the satellite map. The internal map that we use is based on the *stochastic map* representation described in [Smith *et al.*, 1985]. A stochastic map consists of a mean vector and a covariance matrix of spatial variables, and gives us estimates of the spatial relationships of these variables, their uncertainties, and their interdependencies. In addition, it provides us with a very elegant way of propagating constraints from various observations.

3.1 Rover's Internal Map

The internal map is a stochastic map which consists of a mean vector ($\hat{\mathbf{u}}$) and a covariance matrix ($C(\mathbf{u}, \mathbf{u})$) of the vector ($\mathbf{u}$) of the *spatial variables.*

$$\mathbf{u} = \begin{pmatrix} \mathbf{u}_1 \\ \mathbf{u}_2 \\ \vdots \\ \mathbf{u}_n \end{pmatrix} \qquad \hat{\mathbf{u}} = \begin{pmatrix} \hat{\mathbf{u}}_1 \\ \hat{\mathbf{u}}_2 \\ \vdots \\ \hat{\mathbf{u}}_n \end{pmatrix}$$

$$C(\mathbf{u},\mathbf{u}) = \begin{pmatrix} C(\mathbf{u}_1,\mathbf{u}_1) & C(\mathbf{u}_1,\mathbf{u}_2) & \cdots & C(\mathbf{u}_1,\mathbf{u}_n) \\ C(\mathbf{u}_2,\mathbf{u}_1) & C(\mathbf{u}_2,\mathbf{u}_2) & \cdots & C(\mathbf{u}_2,\mathbf{u}_n) \\ \vdots & \vdots & \vdots & \vdots \\ C(\mathbf{u}_n,\mathbf{u}_1) & C(\mathbf{u}_n,\mathbf{u}_2) & \cdots & C(\mathbf{u}_n,\mathbf{u}_n) \end{pmatrix}$$

where

$$\begin{aligned} \mathbf{u}_i &= (x_i, y_i, z_i)^t \\ \hat{\mathbf{u}}_i &= E(\mathbf{u}_i) \\ C(\mathbf{u}_i, \mathbf{u}_j) &= E((\mathbf{u}_i - \hat{\mathbf{u}}_i) \cdot (\mathbf{u}_j - \hat{\mathbf{u}}_j)) \end{aligned}$$

Our spatial variables ($\mathbf{u}_i$'s) are

- locations of sensed peaks
- the rover's previous locations[2]
- the rover's current location ($\mathbf{u}_R$)

The coordinate system used is a Cartesian coordinate system whose origin is the initial location of the rover ($\mathbf{u}_1$) and whose x, y, z axes are parallel to those of a satellite map. It is referred to as the *global coordinate system*.

3.2 Information in the Internal Map

The estimation of a spatial variable $\mathbf{u}_i$ (i.e. the estimation of the ith variable's coordinates in the global coordinate system) is $\hat{\mathbf{u}}_i$ and its uncertainty is obtained from $C(\mathbf{u}_i, \mathbf{u}_i)$. As a special case, for the initial location which was chosen as the origin of the global coordinate system, $\hat{\mathbf{u}}_1 = \mathbf{0}$, and $C(\mathbf{u}_1, \mathbf{u}_1)$ is a $\mathbf{0}$ matrix, since there is no uncertainty of $\mathbf{u}_1$ with respect to itself. The estimation of the spatial relationship between $\mathbf{u}_i$ and $\mathbf{u}_j$ is $\hat{\mathbf{u}}_j - \hat{\mathbf{u}}_i$, and its interdependency is obtained from $C(\mathbf{u}_j, \mathbf{u}_i)$.

3.3 Moving

When the rover makes a movement of $\mathbf{u}_{RR'}$ from its current location $\mathbf{u}_R$, new location $\mathbf{u}_{R'}$ in the global coordinate system is

$$\mathbf{u}_{R'} = \mathbf{u}_R + \mathbf{u}_{RR'}$$

The mean vector and the covariance matrix of $\mathbf{u}_{RR'}$ are

$$\hat{\mathbf{u}}_{RR'} = \mu_M(x, y, z)$$
$$C(\mathbf{u}_{RR'}, \mathbf{u}_{RR'}) = C_M(x, y, z)$$

$\mu_M(x, y, z)$ and $C_M(c, y, z)$ are defined in (3) and (4).

The rover's internal map is expanded from $(\hat{\mathbf{u}}, C(\mathbf{u}, \mathbf{u}))$ to $(\hat{\mathbf{u}}', C(\mathbf{u}', \mathbf{u}'))$ as follows (also see Figure 5):

$$\begin{aligned} \hat{\mathbf{u}}_{R'} &= \hat{\mathbf{u}}_R + \hat{\mathbf{u}}_{RR'} \\ \hat{\mathbf{u}}' &= \begin{pmatrix} \hat{\mathbf{u}} \\ \hat{\mathbf{u}}_{R'} \end{pmatrix} \end{aligned} \tag{5}$$

$$\begin{aligned} C(\mathbf{u}_{R'}, \mathbf{u}_{R'}) &= C(\mathbf{u}_R, \mathbf{u}_R) + C(\mathbf{u}_{RR'}, \mathbf{u}_{RR'}) \\ C(\mathbf{u}, \mathbf{u}_{R'}) &= C(\mathbf{u}, \mathbf{u}_R) \\ C(\mathbf{u}', \mathbf{u}') &= \begin{pmatrix} C(\mathbf{u}, \mathbf{u}) & C(\mathbf{u}, \mathbf{u}_{R'})^t \\ C(\mathbf{u}, \mathbf{u}_{R'}) & C(\mathbf{u}_{R'}, \mathbf{u}_{R'}) \end{pmatrix} \end{aligned} \tag{6}$$

[2] In [Smith *et al.*, 1985], the stochastic map does not contain the rover's previous locations. They are not necessay for navigation purposes. But the previous locations are useful for our matching purpose, since they give us more clues. From the view point of matching, there is no difference between peaks of hills and the rover's locations.

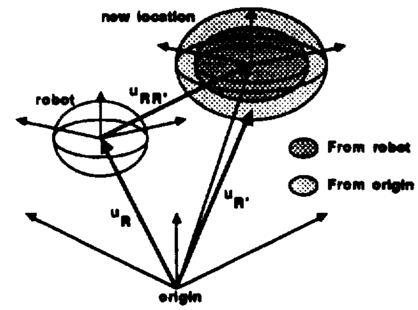

Figure 5: Rover movement

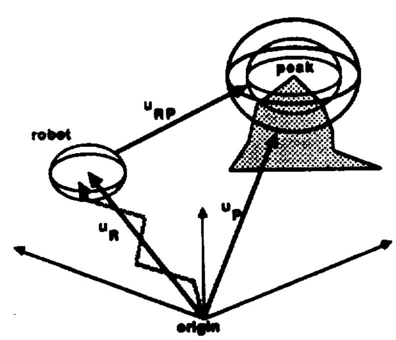

Figure 6: Seeing a new peak

3.4 Seeing a New Peak

When the rover finds a new peak at $\mathbf{u}_{RP}$ from its current location $\mathbf{u}_R$, then the location of the peak $\mathbf{u}_P$ (in the global coordinate system) is

$$\mathbf{u}_P = \mathbf{u}_R + \mathbf{u}_{RP}$$

The mean vector and the covariance matrix of $\mathbf{u}_{RP}$ are

$$\hat{\mathbf{u}}_{RP} = \mu_V(x, y, z)$$
$$C(\mathbf{u}_{RP}, \mathbf{u}_{RP}) = C_V(x, y, z)$$

$\mu_V(x, y, z)$ and $C_V(c, y, z)$ are defined in (1) and (2). The rover's internal map is expanded from $(\hat{\mathbf{u}}, C(\mathbf{u}, \mathbf{u}))$ to $(\hat{\mathbf{u}}', C(\mathbf{u}', \mathbf{u}'))$ in the same way as moving (see Figure 6).

3.5 Seeing the Same Peak Again

When the rover sees a peak from $\mathbf{u}_R$ and identifies it as $\mathbf{u}_P$ which it has seen before, the internal map is updated to get a better estimate of the spatial variables (Figure 7). In this case, the size of the map does not change, since no new spatial variable is added.

First, we define a sensor model as follows.

$$\mathbf{u}_{RP'} = \mathbf{f}(\mathbf{u}) + \mathbf{v} = \mathbf{u}_P - \mathbf{u}_R + \mathbf{v}$$

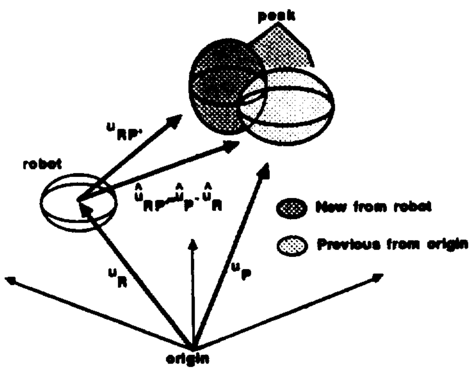

Figure 7: Seeing the same peak again

where $\mathbf{u}_{RP'}$ is the sensor measurement values, defined as μ_V in (1), $\mathbf{u}$ is the vector of spatial variables, and $\mathbf{v}$ is the sensor noise with 0 mean.

Next, we compute the conditional estimates of the above sensor values ($\hat{\mathbf{u}}_{RP'}$), and their uncertainties ($C(\mathbf{u}_{RP'}, \mathbf{u}_{RP'})$).

$$\hat{\mathbf{u}}_{RP'} = \mathbf{f}(\hat{\mathbf{u}}) = \mathbf{F_u} \cdot \hat{\mathbf{u}} = \hat{\mathbf{u}}_P - \hat{\mathbf{u}}_R$$

$$C(\mathbf{u}_{RP'}, \mathbf{u}_{RP'}) = \mathbf{F_u} \cdot C(\mathbf{u}^-, \mathbf{u}^-) \cdot \mathbf{F_u^t} + C(\mathbf{v}, \mathbf{v})$$

$$\mathbf{F_u} = \delta\mathbf{f}/\delta\mathbf{u} = \begin{pmatrix} 0 & \cdots & 0 & \underset{\mathbf{u}_P}{1} & 0 & \cdots & 0 & \underset{\mathbf{u}_R}{-1} & 0 & \cdots & 0 \end{pmatrix}$$

where $C(\mathbf{v}, \mathbf{v})$ is defined as C_V in (2).

Then, we update the map $(\hat{\mathbf{u}}^-, C(\mathbf{u}^-, \mathbf{u}^-))$ to $(\hat{\mathbf{u}}^+, C(\mathbf{u}^+, \mathbf{u}^+))$ using Kalman Filter equations [Smith and Cheeseman, 1986].

$$\begin{aligned} K &= C(\mathbf{u}^-, \mathbf{u}^-) \cdot \mathbf{F_u^t} \cdot [C(\mathbf{u}_{RP'}, \mathbf{u}_{RP'})]^{-1} \\ \hat{\mathbf{u}}^+ &= \hat{\mathbf{u}}^- + K \cdot (\mathbf{u}_{RP'} - \hat{\mathbf{u}}_{RP'}) \\ C(\mathbf{u}^+, \mathbf{u}^+) &= C(\mathbf{u}^-, \mathbf{u}^-) - K \cdot \mathbf{F_u} \cdot C(\mathbf{u}^-, \mathbf{u}^-) \end{aligned}$$

Through the above equations we can get better estimates and certainties of not only $\mathbf{u}_P$ and $\mathbf{u}_R$, but also all spatial variables that are correlated with $\mathbf{u}_P$ and $\mathbf{u}_R$.

4 Matching the Two Maps

We have described both the satellite map and the rover's internal map. Note that the rover's internal map is built from observations independent of the satellite map. In this section we will explain the algorithm used to match the two maps in order to locate the rover with respect to the satellite map. The basic idea is the following. If we know the location of any of the spatial variables with respect to the satellite map, we can transform all spatial constraints between the two maps. It is then easy to compare the two maps. Since we don't know (with certainty) the location of any spatial variable with respect to the satellite map, we attempt to rule out those locations that are implausible returning the likely locations as an estimate of the rover's location.

4.1 Sector Assignment

We start by assuming that the rover is located within certain sectors[3] of the satellite map.

$$\begin{aligned} X_i \leq &\ x_R < X_{i+1} \\ Y_i \leq &\ y_R < Y_{i+1} \\ H^-_{i,j} \leq &\ z_R \leq H^+_{i,j} \end{aligned} \tag{7}$$

where (x_R, y_R, z_R) is the rover's current location in the internal map, (X_i, Y_j) is the vertex of (i,j)th sector in the satellite map, (X_{i+1}, Y_{j+1}) is the vertex of $(i+1, j+1)$th sector in the satellite map, and $H^-_{i,j}, H^+_{i,j}$ are the lower and upper bounds of heights in the (i,j)th sector in the satellite map.

4.2 Assignment as a Sensor Measurement

We try to express the assignment (7) as a kind of sensor measurement so that we can incorporate it into the internal map. From (7) we get

$$\begin{aligned} X_O - X_{i+1} \leq &\ X_O - x_R < X_O - X_i \\ Y_O - Y_{j+1} \leq &\ Y_O - y_R < Y_O - Y_j \\ Z_O - H^+_{i,j} \leq &\ Z_O - z_R \leq Z_O - H^-_{i,j} \end{aligned} \tag{8}$$

where (X_O, Y_O, Z_O) is the origin of the satellite map.

We choose $(X_0, Y_0, H^-_{0,0})$ as the origin. Note that the middle terms in (8) correspond to $\mathbf{u}_{RO}$, the relative position of the satellite map's origin from the current location (Figure 8).

$$\mathbf{u}_{RO} = \mathbf{u}_O - \mathbf{u}_R \tag{9}$$

where $\mathbf{u}_O$ is the origin of the satellite map in the *global* coordinate system.

In our model, the vision system is supposed to return the mean vector and the covariance matrix for a sensed object. In other words, the vision system returns its certainty ellipsoid which corresponds to some confidence threshold. Hence we approximate the cuboid region (8) with a circumscribed ellipsoid (Figure 9), and then convert it to a mean vector and a covariance matrix. We can obtain a certainty ellipsoid from a covariance matrix [Smith and Cheeseman, 1986]. Here we follow the derivation in the opposite direction, and we get

$$\hat{\mathbf{u}}_{RO} = \begin{pmatrix} x_m \\ y_m \\ z_m \end{pmatrix} = \begin{pmatrix} 0.0 - (i + 0.5)d \\ 0.0 - (j + 0.5)d \\ H^-_{0,0} - (H^-_{i,j} + H^+_{i,j}) * 0.5 \end{pmatrix} \tag{10}$$

[3] Sectors are the square regions that make up satellite maps.

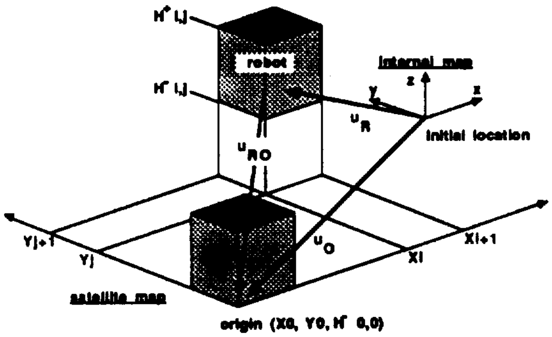

Figure 8: Assigning the rover a sector

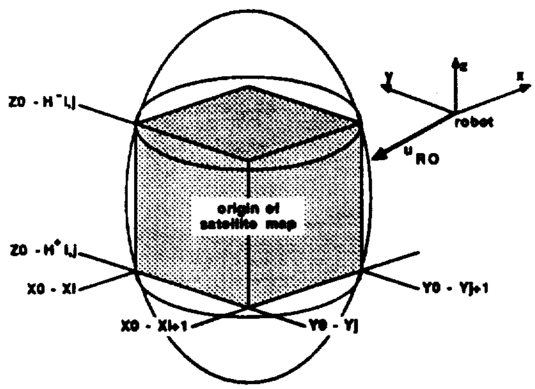

Figure 9: Circumscribed ellipsoid for $\mathbf{u}_{RO}$

$$C(\mathbf{u}_{RO}, \mathbf{u}_{RO}) = \begin{pmatrix} 3/4d^2 & 0 & 0 \\ 0 & 3/4d^2 & 0 \\ 0 & 0 & 3/4h^2 \end{pmatrix} \div \gamma^2 \quad (11)$$

where d is the resolution of the satellite map, h is $(H^{+}{}_{i,j} - H^{-}{}_{i,j})$, and γ in (11) is a constant chosen for a particular confidence threshold ($prob$[4]), and satisfies the following formula,

$$prob = 2 * \Phi(\gamma) - 1 - \sqrt{2/\pi} * \gamma * e^{-\gamma^2/2}$$

where $\Phi(\gamma)$ is the cumulative density function of the unit normal distribution.

4.3 Merging the Two Maps

We incorporate $\mathbf{u}_O$ in (9) (the origin of the satellite map) into the rover's internal map as a virtual landmark. The internal map is expanded from $(\hat{\mathbf{u}}, C(\mathbf{u}, \mathbf{u}))$ to $(\hat{\mathbf{u}}', C(\mathbf{u}', \mathbf{u}'))$ as follows, using (10) and (11).

$$\begin{aligned} \hat{\mathbf{u}}_O &= \hat{\mathbf{u}}_R + \hat{\mathbf{u}}_{RO} \\ \hat{\mathbf{u}}' &= \begin{pmatrix} \hat{\mathbf{u}} \\ \hat{\mathbf{u}}_O \end{pmatrix} \quad (12) \\ C(\mathbf{u}_O, \mathbf{u}_O) &= C(\mathbf{u}_R, \mathbf{u}_R) + C(\mathbf{u}_{RO}, \mathbf{u}_{RO}) \\ C(\mathbf{u}, \mathbf{u}_O) &= C(\mathbf{u}, \mathbf{u}_R) \\ C(\mathbf{u}', \mathbf{u}') &= \begin{pmatrix} C(\mathbf{u}, \mathbf{u}) & C(\mathbf{u}, \mathbf{u}_O)^t \\ C(\mathbf{u}, \mathbf{u}_O) & C(\mathbf{u}_O, \mathbf{u}_O) \end{pmatrix} \quad (13) \end{aligned}$$

Although we have incorporated only the origin of the satellite map, we actually have merged the two maps. Via $\mathbf{u}_O$, we can transform any constraint in the internal map to the constraint in the satellite map and vice versa.

4.4 Checking Assignment Consistency

The assignment made in (7) is arbitrary. We have to check whether it is consistent with both the given sastellite map and the rover's observations. For any peak the rover has seen (and also for any previous location of the rover), its coordinates in the *satellite map's* coordinate system are given as

$$\mathbf{u}'_P = \mathbf{u}_P - \mathbf{u}_O \quad (14)$$

If the initial assignment (7) is correct, then $\mathbf{u}'_P$ should be contained in some $CUBOID(k, l)$ of the satellite map, for $\mathbf{u}'_P = (x'_P, y'_P, z'_P)^t$ is the coordinates of a peak in the satellite map's coordinate system. For some k and l, we have

$$\begin{aligned} X_k \leq \; & x'_P < X_{k+1} \\ Y_l \leq \; & y'_P < Y_{l+1} \\ H^{-}{}_{k,l} \leq \; & z'_P \leq H^{+}{}_{k,l} \end{aligned} \quad (15)$$

In order to check the inequalities (15), we need the actual distribution of $\mathbf{u}'_P$. We use a certainty ellipsoid for that purpose. Given a confidence level ($prob$), the certainty ellipsoid $ELLPS(prob)$ for $\mathbf{u}'_P$ can be computed from its mean and covariance. These are obtained from the expanded internal map (12) and (13),

$$\begin{aligned} \hat{\mathbf{u}}'_P &= \hat{\mathbf{u}}_P - \hat{\mathbf{u}}_O \\ C(\mathbf{u}'_P, \mathbf{u}'_P) &= C(\mathbf{u}_P, \mathbf{u}_P) + C(\mathbf{u}_O, \mathbf{u}_O) - 2 \cdot C(\mathbf{u}_P, \mathbf{u}_O) \end{aligned}$$

When we set the confidence level ($prob$) sufficiently large, we should expect

$$\mathbf{u}'_P \in ELLPS(prob) \quad (16)$$

From (15) and (16), it follows that for any $\mathbf{u}_P$ in the internal map, there is some (k, l) s.t.

$$CUBOID(k, l) \textit{ intersects } ELLPS(prob) \textit{ of } \mathbf{u}'_P. \quad (17)$$

[4]The probability that (x, y, z) falls within the circumscribed ellipsoid. As we want the same sized ellipsoid in the next checking step, $prob$ should be the same as the value which is used in the checking step.

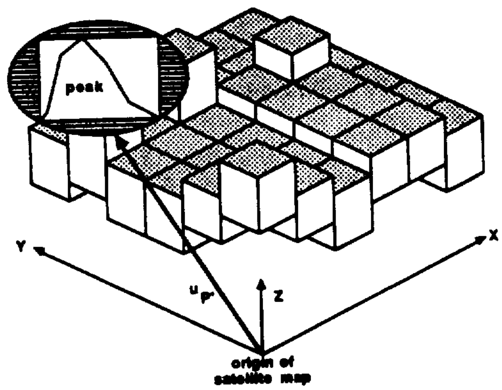

Figure 10: Checking consistency

Figure 10 illustrates this property.

To reiterate, statement (17) is only a necessary condition for a sector assignment (7) to be correct[5]. It is not a sufficient condition. In general, several sector assignments may satisfy (17). The total number of assignments which satisfy (17) depends on the satellite map's fuzziness and how many useful observations the rover has obtained so far. To find all sectors which are consistent, we repeat the above assignment and check steps for every sector in the satellite map.

5 Simulation

We tested our method using a simulation, and the initial results are encouraging. The simulation was carried out using a small terrain model to expedite the experiments. Realistic terrain data was obtained using techniques derived from fractal geometry. We assume that the rover has a vision system that can sense the peaks of hills.

The results of matching were weak at first. Too many sectors were left as possible locations of the rover. This was because sector assignments correspond to a much bigger certainty ellipsoid than those of sensor measurements; even an accurate sensor measurement became a vague one when it was compounded with a sector assignment in the matching phase. By making the covariance matrix of the sector assignment smaller, we were able to obtain satisfactory narrowing of the possible locations of the rover.

6 Conclusions

A method has been developed to locate the rover using local observations and a global satellite map. The method provides answers to questions of the form: "Is it consistent to assume that the rover is located within a fixed area?" Its theoretical foundations are firm in the sense that the matching algorithm checks mathematically necessary conditions for a location assumption to be correct; the algorithm does not rely on any heuristics.

It should be noted that our problem cannot be handled using methods adapted from work on cruise missile guidance systems [Kober *et al.*, 1979]. Since cruise missiles are equipped with a highly accurate inertial guidance system, there is little uncertainty about their positions and orientations. More importantly, the missile sensors provide measurements from roughly the same perspective used in constructing the navigation map (the analog of our satellite map). We have also considered the possibility of using template matching techniques for our problem [Thorpe, 1981], but the low resolution of the satellite map makes landmark identification difficult and would appear to severely reduce the accuracy of the matching method.

7 Future Work

The simulations carried out so far are not sufficient. In order to make our method more effective and robust:

- We need guidelines on good threshold values for consistency checking.
- We need a better way of modeling of sector assignments than pseudo-sensor readings.
- We need to provide some way of tuning our method to suit the requirements of particular satellite maps and sensors.

So far, our main concern has been finding the current location of the rover with respect to the satellite map. But our matching method can be applied to the rover's navigation problem as well. Our work attacks the same problem as [Levitt *et al.*, 1987], but with the emphasis on metric rather than qualitative matching. In navigation problems, global maps usually contain some distinctive places (landmarks) and the paths are specified in terms of landmarks. Being able to determine the current location of the rover, identify landmarks, and determine the rover's location relative to a particular landmark are important problems that have to be solved for successful path planning and path following. Our matching algorithm should be useful in solving these problems since it can check the consistency of assumptions on the location of any of the observed peaks in the global (satellite) map, or on the relative location of the rover to some landmark.

[5] To be more precise, it is not even a necessary condition because there is a slight chance that the actual position of a peak falls outside of the ellipsoid.

References

[Durrant-Whyte, 1988] Hugh F. Durrant-Whyte. *Integration, Coordination and Control of Multi-Sensor Robot Systems.* Kluwer Academic Publishers, 1988.

[Hayashi and Dean, 1988] Akira Hayashi and Thomas Dean. Locating a mobile robot using local observations and a global satellite map. In *Proceedings of Third IEEE International Symposium on Intelligent Control*, 1988.

[Kober *et al.*, 1979] W. Kober, C. Grosch, and T. Rosenthal. Three dimensional surface shell correlator. *SPIE*, 186, 1979.

[Levitt *et al.*, 1987] Tod S. Levitt, Daryl T. Lawton, David M. Chelberg, and Philip C. Nelson. Qualitative landmark-based path planning and following. In *Proceedings AAAI-87*, pages 689–694. AAAI, 1987.

[Smith and Cheeseman, 1986]
Randall Smith and Peter Cheeseman. On the representation of and estimation of spatial uncertainty. *The International Journal of Robotics Research*, 5:56–68, 1986.

[Smith *et al.*, 1985] Randall Smith, Mathew Self, and Peter Cheeseman. Estimating uncertain spatial relationships in robotics, 1985.

[Thorpe, 1981] C. Thorpe. Sonar image processing – an application of template matching through relaxation. Technical Report CMU-RI-TR-81-6, Carnegie-Mellon University Robotics Institute, 1981.