DYNAMICS AND CONTROL OF MICRO-ROBOT SWARMS: PLANAR MOTION AND STATE ESTIMATION

TECHNICAL REPORT

Hanspeter Schaub* **and John L. Junkins†**

The problem of controlling Micro-Robotic Vehicles (MRVs) is studied in this report. Two types of state estimators are presented for the cases of full and partial state measurement. A potential gradient type control law is then developed to drive the MRVs to the target location without hitting one another.

INTRODUCTION

Microrobotic systems are being studied for future use on land and in space. To perform a specific task, the philosophy is to replace one large, expensive and complex robot with a multitude of small, inexpensive and simple robots. This report studies miniature land based robotic vehicles(MRVs). They are assumed to have sensors aboard which are to be transported to a specific target area. As long as "most" MRVs make it to the target area, the mission is considered to be a success.

Each MRV has some limited navigational sensing capability such as heading or range to target. Also, it is assumed that the MRVs are able to sense each other in some manner. The idea is to control the MRVs in such a manner that they do no collide and are capable of some simple cooperation. As an example, it the MRVs only have a limited sensor range, then not all will see the target initially. Here it would be beneficial for MRVs that do see the target to be able to communicate and guide the MRVs that do not see it.

This report will present two types of state estimators. The first one assumes that the MRVs have the full state measured. The second one assumes that the MRVs only can measure their relative distance to the target and their compass heading.

*Graduate Research Assistant, Aerospace Engineering Department, Texas A&M University, College Station TX 77843.

†George Eppright Chair Professor of Aerospace Engineering, Aerospace Engineering Department, Texas A&M University, College Station TX 77843.

In this case the relative orientation to the target will be obtained through the state estimator. These estimators will be useful in running simulations since they allow one to incorporate realistic effects of measurement noise and limited sensing capabilities. Of particular interest will be to generate some stochastic model of how many MRVs actually reach the target given different conditions and measurement assumptions.

PROBLEM STATEMENT

Each miniature robot is equipped with two tracks to propel it forward and rotate it as illustrated in Figure 1. The origin of the coordinate system is assumed to be the target. The MRV heading is given by the angle θ. The range to target is denoted by r and the relative heading is ϕ.

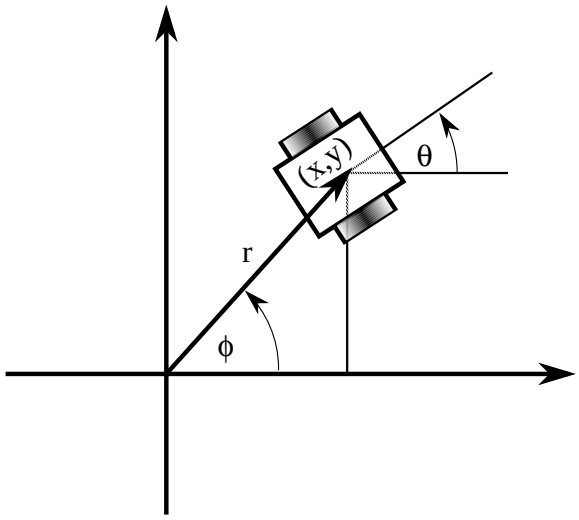

Figure 1 Illustration of the Miniature Robot Crawler

Let $p_i = (x_i, y_i, \theta_i)^T$ be the state vector of the i-th MRV, then the equations of motion are given by[1]

$$\dot{p}_i = B(p_i)\omega_i = \frac{1}{2}\begin{bmatrix} R_r\cos\theta_i & R_l\cos\theta_i \\ R_r\sin\theta_i & R_l\sin\theta_i \\ 2\frac{R_r}{R_w} & -2\frac{R_l}{R_w} \end{bmatrix}\begin{bmatrix} \omega_{r_i} \\ \omega_{l_i} \end{bmatrix} = f(p_i, \omega_i) \tag{1}$$

where $\omega_i = (\omega_{r_i}, \omega_{l_i})^T$, R_r and R_l are the right and left track radii and R_w is the distance between the two tracks. The MRV can rotate by having each track rotate at different speeds. Track slippage or sticking effects are not modeled here.

STATE ESTIMATION

Estimation Algorithm Outline

In the following development the state estimation process for a particular MRV is studied. Therefore the index counter i for each MRV state p_i is dropped in this section. Since the position sensor updates typically occur at a low frequency, it is necessary to estimate the states between these updates to calculate the current control input. Between these sensor updates it is assumed that the track angular velocities can be sampled at a high frequency. The measured track angular velocities ω_m will contain some white noise w which is assumed to have a standard deviation of σ_ω. The true track angular velocity is given as

$$\omega(t) = \omega_m(t) - w(t) \tag{2}$$

The following extended Kalman estimator provides current state estimates and is able to filter out some measurement noise by optimal weighting based upon forward propagating the state covariance matrix.[2] With the following formulas, a subscript indicates the number of data sets that were used to find its estimate. A superscript indicates the time step of the state estimate. Let $\hat{p}_k^k$ be the current state estimate at time step k. Until the next sensor update is available at time $k+1$, the state estimates are then forward integrated using the nonlinear equations of motion in Eq. (1), this process is indicated formally as

$$\hat{p}_k^{k+} = \hat{p}_k^k + \int_k^{k+} \dot{p} dt \tag{3}$$

The notation p^{k+} indicates the state p at time $k+$, where $k < k+ < k+1$. To be able to forward propagate the state covariance estimate derived from measurement error covariances, the linearized dynamics about some reference states p_{ref} and ω_{ref} are required.

$$\dot{p}_{ref} = f(p_{ref}, \omega_{ref}) \tag{4}$$

Taking the first Taylor expansion of Eq. (1) about the reference motion we obtain

$$\dot{p} \approx f(p_{ref}, \omega_{ref}) + \left.\frac{\partial f(p,\omega)}{\partial p}\right|_{ref} (p - p_{ref}) + \left.\frac{\partial f(p,\omega)}{\partial \omega}\right|_{ref} (\omega - \omega_{ref}) + \cdots \tag{5}$$

After subtracting Eq. (4) from (5) we obtain

$$\delta\dot{p} = \left.\frac{\partial f(p,\omega)}{\partial p}\right|_{ref} \delta p + \left.\frac{\partial f(p,\omega)}{\partial \omega}\right|_{ref} \delta\omega \tag{6}$$

where $\delta p = p - p_{ref}$ and $\delta\omega = \omega - \omega_{ref}$. For linearization of the dynamics, the reference states are set equal to the best present estimates. Therefore $p_{ref} = p_k^{k+}$ and $\omega_{ref} = \omega_m$. Using Eq. (2) the vector $\delta\omega$ is clearly $-w$ and is the driving process noise of the linearized system. The linearized system can now be written as

$$\delta\dot{p} = F(p_k^{k+}, \omega_m)\delta p + G(p_k^{k+})\delta\omega \tag{7}$$

where the matrices F and G are defined as

$$F = F(p_k^{k+}, \omega_m) = \left.\frac{\partial f(p,\omega)}{\partial p}\right|_{(p_k^{k+},\omega_m)} = \frac{1}{2}\begin{bmatrix} 0 & 0 & -(R_r\omega_r + R_l\omega_l)\sin\theta \\ 0 & 0 & (R_r\omega_r + R_l\omega_l)\cos\theta \\ 0 & 0 & 0 \end{bmatrix} \tag{8}$$

$$G = G(p_k^{k+}, \omega_m) = \left.\frac{\partial f(p,\omega)}{\partial \omega}\right|_{(p_k^{k+},\omega_m)} = \frac{1}{2}\begin{bmatrix} R_r\cos(\theta) & R_l\cos(\theta) \\ R_r\sin(\theta) & R_l\sin(\theta) \\ 2R_r/R_w & -2R_l/R_w \end{bmatrix} \tag{9}$$

From here on we will only use the short hand notation F and G where their dependence will be implicitly understood.

Let P_k^k be the state covariance matrix at time step k. This covariance matrix provides a measure of how uncertain the current state estimates are. A high entry in P indicates a high uncertainty of the current state estimate. Let Q be the covariance matrix associated with the driving process noise and the 2x1 vector σ_w be the standard deviation of the process noise. The matrix Q is then defined as

$$Q = \begin{bmatrix} \sigma_{w_1}^2 & 0 \\ 0 & \sigma_{w_2}^2 \end{bmatrix} \tag{10}$$

Without including the effect of process noise the covariance matrix P would eventually tend to zero. This means that only the previous measurements will be trusted and future updates ignored. Including the covariance matrix Q allows the estimator to be tuned such that P never will go to zero. Past measurements are never perfectly trusted. In between sensor updates, this covariance matrix P is updated using

$$P_k^{k+} = P_k^k + \int_k^{k+} \dot{P} dt \tag{11}$$

where the covariance matrix derivative is given as the inhomogeneous Lyapunov equation[3]

$$\dot{P} = FP + PF^T + GQG^T \tag{12}$$

Standard literature on continuous covariance propagation includes an extra "learning" term in the above equation, resulting in the Riccati equation.

$$\dot{P} = FP + PF^T - PH^T\Lambda_{vv}^{-1}HP + GQG^T$$

This term would "decrease" the covariance matrix if the sensor output were sampled continuously. However, the sensor output is only sampled at discrete times and not continuously like the track angular velocities. Therefore this term is dropped here since no sensor based learning occurs between times k and $k+1$. Once a new sensor measurement is available the covariance matrix P will also be updated discretely along with the state vector.

Since the F and G matrices for the MRVs contain large blocks of zeros, the calculation of $\dot{P}$ can be simplified. Let the P matrix be partitioned as

$$P = \begin{bmatrix} P_{11} & P_{12} \\ P_{12}^T & P_{22} \end{bmatrix} \tag{13}$$

where P_{11} is a 2x2 matrix, P_{12} is a 2x1 matrix and P_{22} is a scalar. The matrix F is partitioned as

$$F = F(p, \omega_m) = \begin{bmatrix} 0 & F_{12} \\ 0 & 0 \end{bmatrix} \tag{14}$$

where the 2x1 matrix F_{12} is defined as

$$F_{12} = \frac{1}{2} \begin{bmatrix} -(R_r\omega_{m_r} + R_l\omega_{m_l}) \sin\theta \\ (R_r\omega_{m_r} + R_l\omega_{m_l}) \cos\theta \end{bmatrix} \tag{15}$$

The matrix G is written as

$$G = G(p) = \begin{bmatrix} G_1 \\ G_2 \end{bmatrix} \tag{16}$$

with the 2x2 matrix G_1 being

$$G_1 = \frac{1}{2} \begin{bmatrix} R_r \cos(\theta) & R_l \cos(\theta) \\ R_r \sin(\theta) & R_l \sin(\theta) \end{bmatrix} \tag{17}$$

and the 2x1 matrix G_2 being

$$G_2 = \begin{bmatrix} \frac{R_r}{R_w} & -\frac{R_l}{R_w} \end{bmatrix} \tag{18}$$

Using these definitions, the time derivatives of the P matrix partitions are expressed as

$$\dot{P}_{11} = P_{12}F_{12}^T + F_{12}P_{12}^T + G_1QG_1^T \tag{19a}$$

$$\dot{P}_{12} = F_{12}P_{22} + G_1QG_2^T \tag{19b}$$

$$\dot{P}_{22} = G_2QG_2^T \tag{19c}$$

Let $\hat{Y}^{k+1}$ be the estimated output vector of the sensors at time $k+1$. It is defined as

$$\hat{Y}^{k+1} = h(\hat{p}_k^{k+1}) \tag{20}$$

The generally nonlinear term $h(p_k^{k+1})$ maps the current state estimate into a best prediction of the observation vector. The measured sensor output vector $\tilde{Y}^{k+1}$ at time $k+1$ given in terms of the true state vector p^{k+1} is

$$\tilde{Y}^{k+1} = h(p^{k+1}) + v \tag{21}$$

where the vector v is the gaussian measurement noise with standard deviation σ_v. The covariance matrix associated with the measurement noise, assuming no correlation of measurement errors, is

$$\Lambda_{vv} = \begin{bmatrix} \sigma_{v_1}^2 & 0 & \cdots \\ 0 & \sigma_{v_2}^2 & \cdots \\ \vdots & \vdots & \ddots \end{bmatrix} \tag{22}$$

Assume that at time $k+1$ a new sensor update $\tilde{Y}^{k+1}$ is available. The current state estimate $\hat{p}_k^{k+1}$ is updated to incorporate the new sensor measurement through the extended Kalman filter recursion[2]

$$\hat{p}_{k+1}^{k+1} = \hat{p}_k^{k+1} + K^{k+1}\left(\tilde{Y}^{k+1} - \hat{Y}^{k+1}\right) \tag{23}$$

The matrix K^{k+1} is the optimal kalman gain matrix which is found through

$$K^{k+1} = P_k^{k+1} H^T \left(\Lambda_{vv} + H P_k^{k+1} H^T \right)^{-1} \tag{24}$$

where H is defined as

$$H = \frac{\partial h}{\partial p} \left(p_k^{k+1} \right) \tag{25}$$

The state covariance matrix P is updated to reflect the presence of a new sensor measurement through

$$P_{k+1}^{k+1} = \left(I - K^{k+1} H \right) P_k^{k+1} \tag{26}$$

Full State Measurement Case

If the MRV sensor are capable of feeding back the full state vector, then the estimated observation vector $\hat{Y}^{k+1}$ can be written as

$$\hat{Y}^{k+1} = p_k^{k+1} \tag{27}$$

For this case the mapping $h(p_k^{k+1})$ is the identity mapping and the H matrix is the identity matrix. This estimator was numerically simulated where the track angular velocity vector is prescribed to be

$$\omega(t) = \begin{bmatrix} 1 \\ 1 \end{bmatrix} + \begin{bmatrix} .7 \\ .5 \end{bmatrix} t \tag{28}$$

The true initial states are

$$p(0) = \begin{bmatrix} -10 \\ 9 \\ .3 \end{bmatrix} \tag{29}$$

All angle measurements are given in radians. The initial state estimate was simply set to zero. To reflect this *large* initial ignorance, the state covariance matrix P is initialized as a diagonal matrix with diagonal entries 100. The standard deviations vectors of the state or angular velocity measurements are

$$\sigma_v = \begin{bmatrix} 0.1 \\ 0.1 \\ 0.01 \end{bmatrix} \tag{30}$$

$$\sigma_\omega = \begin{bmatrix} 0.1 \\ 0.1 \end{bmatrix} \tag{31}$$

The simulation used a simple Euler integration method to solve the differential equations. The estimation errors are shown as a solid black line in Figure 2 below. As a comparison, the actual sensor measurement noise levels were included in the background.

Since the initial covariance matrix is very large, the estimator almost ignores existing state estimates and uses the new sensor data to quick converge close to the true states. The estimators can also act as a noise filter. The amount of filtering achieved is controlled by the entries of the two vectors σ_v and σ_ω.

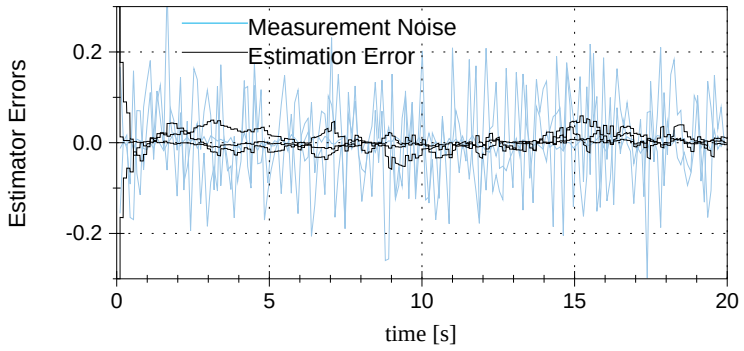

Figure 2 State Estimator Noise Filtering Capability

Partial State Measurement case

If the MRV sensors only can feed back the range to target and the compass heading θ, then the relative heading to the target will have to be learned by the estimator. Let the range r be

$$r = \sqrt{x^2 + y^2} \tag{32}$$

The estimated observation vector is then

$$\hat{Y}^{k+1} = H(p_k^{k+1})p_k^{k+1} = \begin{bmatrix} x/r & y/r & 0 \\ 0 & 0 & 1 \end{bmatrix} p_k^{k+1} \tag{33}$$

A numerical simulation was run to show how well the relative heading ϕ can be estimated. The track angular velocity was prescribed to be

$$\omega(t) = \begin{bmatrix} 1 \\ 1.1 \end{bmatrix} + \begin{bmatrix} .7 \\ .69 \end{bmatrix} t \tag{34}$$

The initial state and state estimate were the same as in the previous simulation. The standard variation vector σ_v of the measurement noise is set to

$$\sigma_v = \begin{bmatrix} 0.5 \\ 0.01 \end{bmatrix} \tag{35}$$

and σ_ω is kept the same as in the previous simulation.

The state estimation errors are shown in Fig. 3. The range r and relative heading ϕ can be estimated. The estimation errors in θ are orders of magnitude smaller than the ones for r and ϕ since the angle θ is measured directly. The noise level on the range measurement is plotted in the background. For this simulation, after about 40 seconds the range estimate error is smaller than the range noise level.

This simulation makes the MRV move large distances as is shown in Fig. 4. The larger the maneuver, the better the estimator will filter out the range and heading information. A second factor affecting how fast the estimator will estimate the states properly is how accurately the range itself is measured.

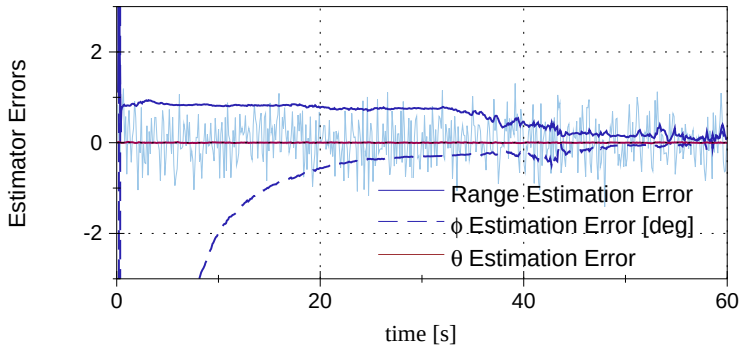

Figure 3 Partial State Measurement Case

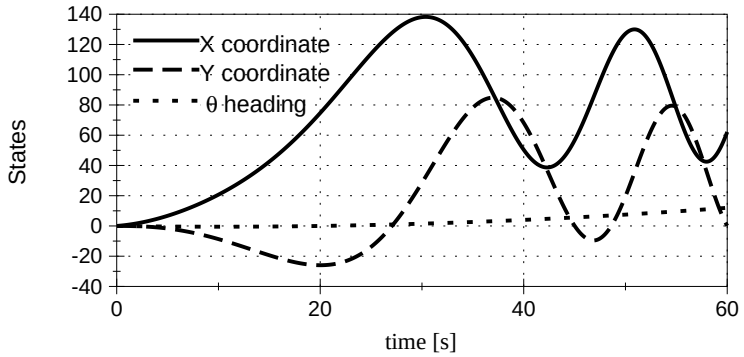

Figure 4 MRV States During Maneuver

POTENTIAL GRADIENT CONTROL LAW

The objective of each MRV is to reach a certain target area without colliding with other MRVs. Further, the computational effort of the control law should be kept as low as possible. To achieve this a simple Potential gradient type control law is presented here. The underlying principle involves generating an attractive potential around the target state and projecting the MRV kinematics onto the gradient of the potential function as illustrated in Fig. 5. The MRV will move until it reaches a local potential function minimum.

Given a potential function $V_i(x_i, y_i, \theta_i)$, the target MRV kinematics are then chosen to move down the local potential function gradient as[4]

$$\frac{dp_i}{dt} = -\gamma \frac{\nabla V_i}{||\nabla V_i||} = B(p_i)\omega_i \tag{36}$$

where the scalar γ is a scaling parameter which controls the overall speed at which the MRV will move down the gradient direction. The control input ω_i is then found to be

$$\omega_i = -\frac{\gamma}{||\nabla V_i||}(B^T B)^{-1} B^T \nabla V_i \tag{37}$$

We adopt the least squares inverse solution, because with only two kinematic controls, we cannot achieve dp_i/dt exactly. Since B is a 3x2 matrix, the calculation of ω_i only

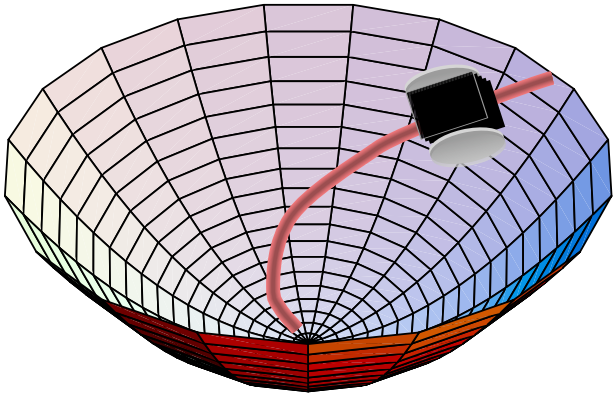

Figure 5 Gradient Type Control Law Illustration

requires the inverse of a 2x2 matrix $B^T B$.

$$B^T B = \frac{1}{4}\begin{bmatrix} R_r^2\left(1+\frac{4}{R_w^2}\right) & R_r R_l\left(1-\frac{4}{R_w^2}\right) \\ R_r R_l\left(1-\frac{4}{R_w^2}\right) & R_l^2\left(1+\frac{4}{R_w^2}\right) \end{bmatrix} \tag{38}$$

Note that $B^T B$ is independent of the state vector p. The inverse of $B^T B$ is given as

$$(B^T B)^{-1} = \frac{1}{4}\begin{bmatrix} \frac{4+R_w^2}{R_r^2} & \frac{4-R_w^2}{R_r R_l} \\ \frac{4-R_w^2}{R_r R_l} & \frac{4+R_w^2}{R_l^2} \end{bmatrix} \tag{39}$$

and only needs to be calculated once since it is constant. The control input ω_i can now be written explicitly as

$$\omega_i = -\frac{\gamma}{||\nabla V_i||}\begin{bmatrix} \frac{\cos\theta}{R_r} & \frac{\sin\theta}{R_r} & \frac{R_w}{2R_r} \\ \frac{\cos\theta}{R_l} & \frac{\sin\theta}{R_l} & \frac{-R_w}{2R_l} \end{bmatrix} \nabla V_i \tag{40}$$

Assume that the MRV is attracted to the point (x_0, y_0), then the following attractive potential function V_i^a is adopted

$$V_i^a(x_i, y_i, \theta_i) = \frac{1}{2}k_1[\Delta x^2 + \Delta y^2] + \frac{1}{2}k_2\alpha^2 \tag{41}$$

where $\Delta x = x_0 - x_i$, $\Delta y = y_0 - y_i$ and $\alpha = \phi - \theta$ as illustrated in Fig. 6 and ϕ is defined as

$$\phi = \tan^{-1}\left(\frac{\Delta y}{\Delta x}\right) \tag{42}$$

Note that the range of the angle α must be limited to be within $-\pi < \alpha < \pi$. Otherwise the MRV might try to reorient itself by rotating the "long" way around (i.e. $+200°$ instead of $-160°$). This development assumes the MRVs have a "front" and a "'back" side. Each MRV is driven such that it drives "forward" towards the

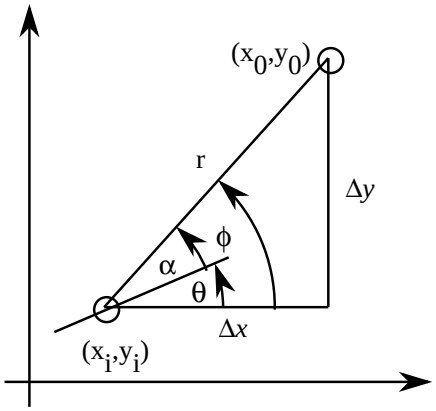

Figure 6 MRV Coordinate Illustration

target. An alternative development can accommodate the case of no preference for driving "forwards" or "backwards."

The gradient vector of V_i^a is

$$\nabla V_i^a(x_i, y_i, \theta_i) = \begin{bmatrix} -k_1\Delta x + k_2\alpha\frac{\Delta y}{\Delta x^2+\Delta y^2} \\ -k_1\Delta y - k_2\alpha\frac{\Delta x}{\Delta x^2+\Delta y^2} \\ -k_2\alpha \end{bmatrix} \tag{43}$$

This potential function V_i^a will attract the MRV to the point (x_0, y_0) and orient it such that it faces this point. Without any information about the heading angle θ in V_i^a the MRV would not be able to reach (x_0, y_0). The term $\partial V_i^a/\partial\theta$ would be zero, which would lead to $\dot{\theta} = 0$ in Eq. (37). The MRVs would simply move in a straight line until a local minimum is found. Including the heading information is important since the MRVs are unable to move sideways. For them to move to the left or right they have to change their heading θ.

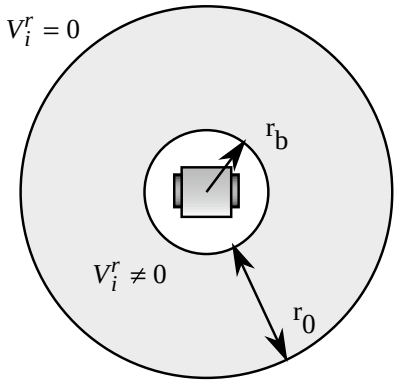

Figure 7 Repulsive Potential Zone Description

To keep one MRV from running into a second MRV, a repulsive potential function V_i^r is added around the second MRV whenever the distance $r = \sqrt{\Delta x^2 + \Delta y^2}$ between

them is too small. Let r_b be the outer most radius of the MRV and r_0 be the outer distance from the MRV at which the repulsive potential function is non zero as shown in Fig. 7. Then the repulsive potential will be non zero whenever the distance between the two MRVs is less than $r_b + r_0$. By activating the repulsive potentials only over a finite range, the computational burden of each MRV is greatly reduced. The repulsive gradient only needs to be calculated when another MRV is within the specified stand off distance.

As with the attractive potential function, the repulsive potential function has to include information about the MRV heading angle θ. However, instead of trying to point a given MRV towards another MRV, here we try to make the MRVs face away from each other. Let $\hat{r} = r - r_b$ and $\beta = \phi - \theta - \pi$, then the repulsive potential is defined as

$$V_i^r(x_i, y_i, \theta_i) = \frac{1}{2}k_3\left(\frac{1}{\hat{r}} - \frac{1}{r_0}\right)^2 + \frac{1}{2}k_4\beta^2] \qquad if \quad r < r_b + r_0 \tag{44}$$

The gradient of V_i^r is

$$\nabla V_i^r(x_i, y_i, \theta_i) = \begin{bmatrix} k_3\left(\frac{1}{\hat{r}} - \frac{1}{r_0}\right)\frac{\Delta x}{\hat{r}^2 r} + k_4\beta\frac{\Delta y}{r^2} \\ k_3\left(\frac{1}{\hat{r}} - \frac{1}{r_0}\right)\frac{\Delta y}{\hat{r}^2 r} - k_4\beta\frac{\Delta x}{r^2} \\ -k_4\beta \end{bmatrix} \qquad if \quad r < r_b + r_0 \tag{45}$$

Again care must be taken to ensure that the range of the angle β lies within $-\pi \leq \beta \leq \pi$ to avoid unnecessarily large rotations.

To illustrate the attractive and repulsive potentials, the motion of a single MRV is simulated. The initial MRV state is (20,20,.2 rad) and the target coordinates are $(0, 0)$. A stationary MRV is placed at $(10, 10)$ to test the avoidance skills of this control law. The result is shown in Fig. 8.

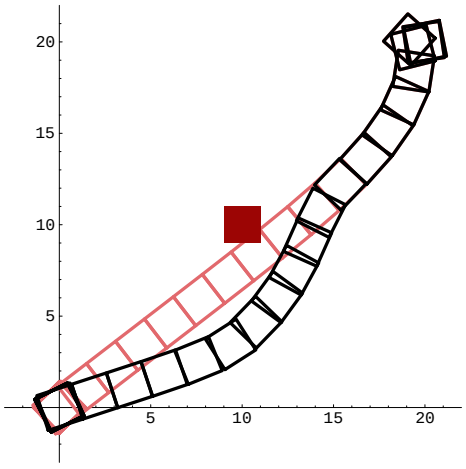

Figure 8 Sample Target Convergence and Obstacle Avoidance

The light colored path shows how the MRV would approach the target without any repulsive potentials present. It's path would take it right through the other

MRV. With the repulsive potential active the MRV is able to maneuver around the stationary MRV and then resume it's path to the target. How quickly the MRV will turn towards the target is controlled by the gain k_2. The larger k_2 is relative to the gain k_1, the faster the MRV will line up with the target.

Now that an attractive/repulsive potential approach has been established that lets the MRVs converge to a target and avoid hitting each other, the next step is to try to make them cooperate finding the target. Typically, each MRVs sensors will only have a limited range. They will only be able to detect each other through some means over some finite distance. At the same time there will be some upper range limit on detecting the target. The general idea is to make the MRVs that do see the target help the MRVs that do not see target.

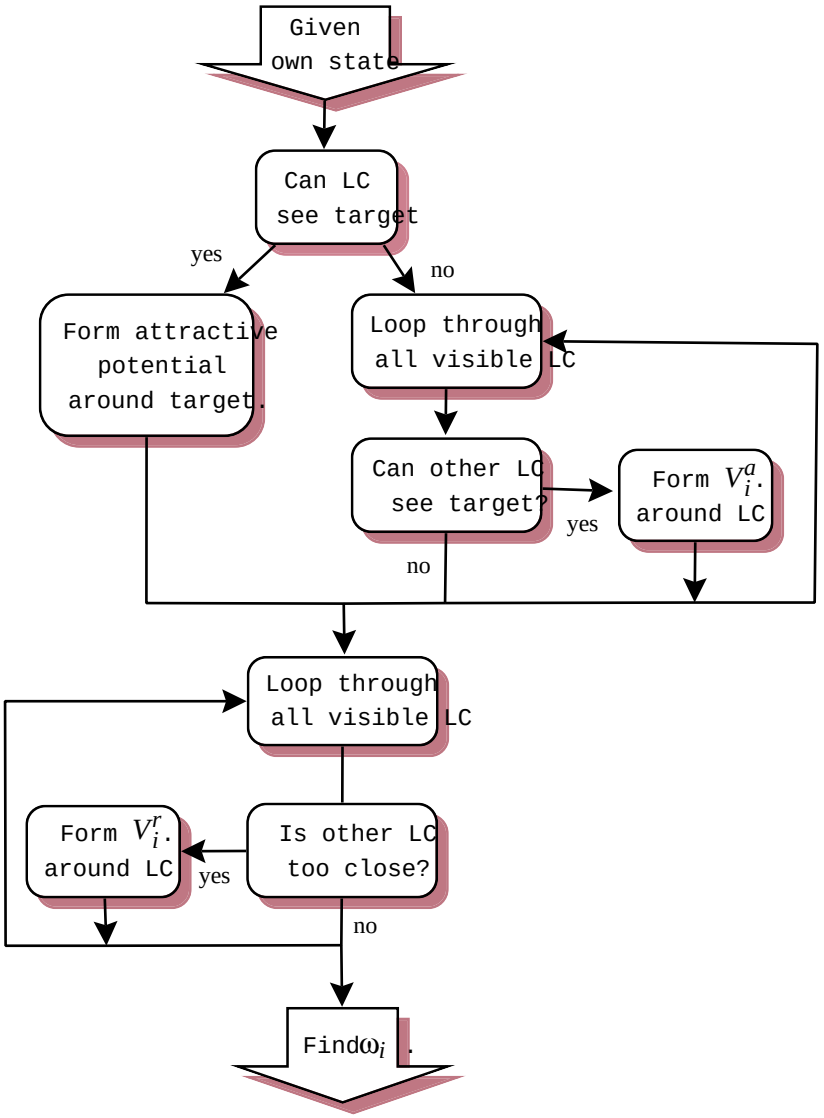

Figure 9 Flow Chart of Potential Gradient Construction

What is proposed in this study is to modulate the signal that the MRVs send out to detect each other between two states. One state would signal to others that this MRV cannot see the target. In this case the other MRVs would simply ignore it unless it is too close. However, if the MRV can see the target, then the signal would be modified to indicate this information to other MRVs. If any of the other MRVs within sensor range cannot see the target themselves, then they will form attractive potentials around the MRVs that can see the target. With this simply *can see / can't see* signal modulation, one can greatly expand the area around the target where the MRVs will converge upon it. The flow diagram in Fig. 9 illustrates this logic process.

The same attractive potential used around the target state can also be used to make other MRVs attractive. If the distance between two MRVs starts to get too small, then the repulsive potential is activated to deflect the gradient and thereby avoid collisions. As soon as the MRV has the target within its own sensor range, then it will ignore all other MRVs, except those within standoff range, and proceed straight to the target.

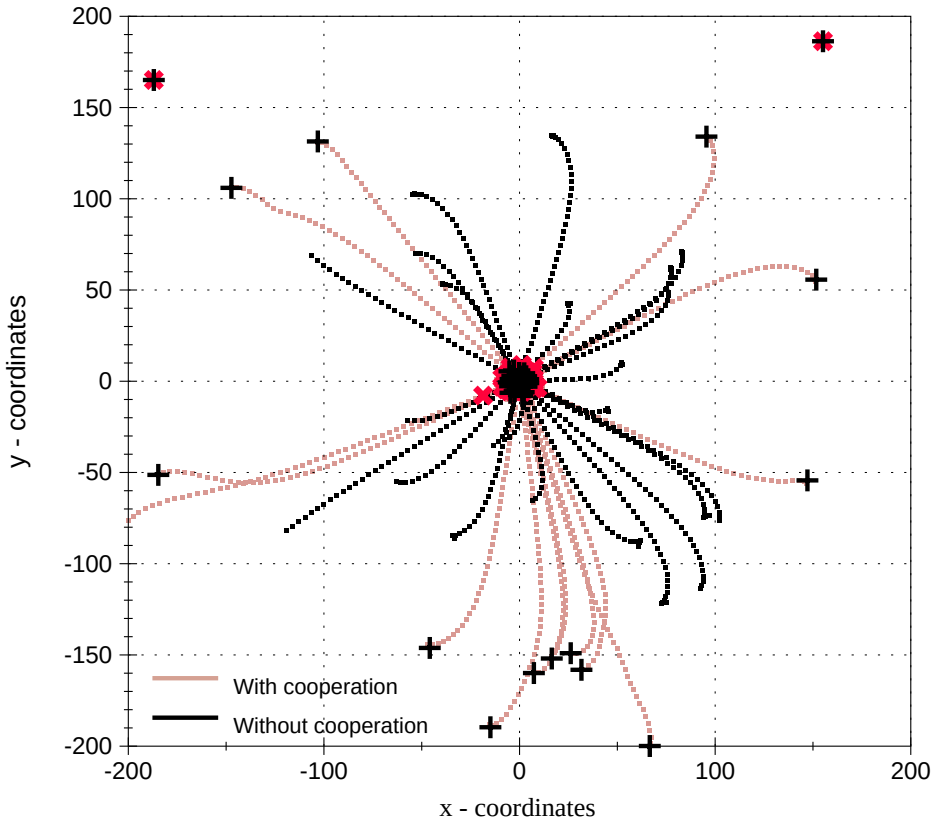

Figure 10 Illustration of MRV Cooperation

This concept is illustrated in the following simulation. The target is the coordinate origin. Forty MRVs are scattered around the target with a standard deviation in position of 100. The integration was performed using the Euler method with a step size of 0.4 seconds and let run for 250 seconds. Both track radii are 0.5 and the wheel

base R_w is 2. The gains for the potential functions where $k_1 = 3$, $k_2 = 20$, $k_3 = 6$ and $k_4 = 1$. The potential function gradient gain γ is set to 1. To generate the repulsive potential, the MRV radius r_b is set to 1 and the function radius r_0 is set to 5. The MRV sensor range to detect the target is 150 and the sensor range to detect other MRVs is 75. The simulation is run once when the cooperation is activated and once when it is inactive. Both cases are shown in Fig. 10.

The black paths are the MRVs that made it to the target without any cooperation from other MRVs. The light colored paths are the ones that only made it to the target when they received cooperation. In this case MRVs that see the target act as "attractive beacons" until the unseeing MRV has the target in sensor range. Clearly this simple signal modification is able to greatly extend the range of convergence for the MRVs. In this simulation the MRVs that could not see the target or other MRV who can see the target simply remain stationary. It is possible to have them perform some "standard search pattern" with the motto "some search is better than no search."

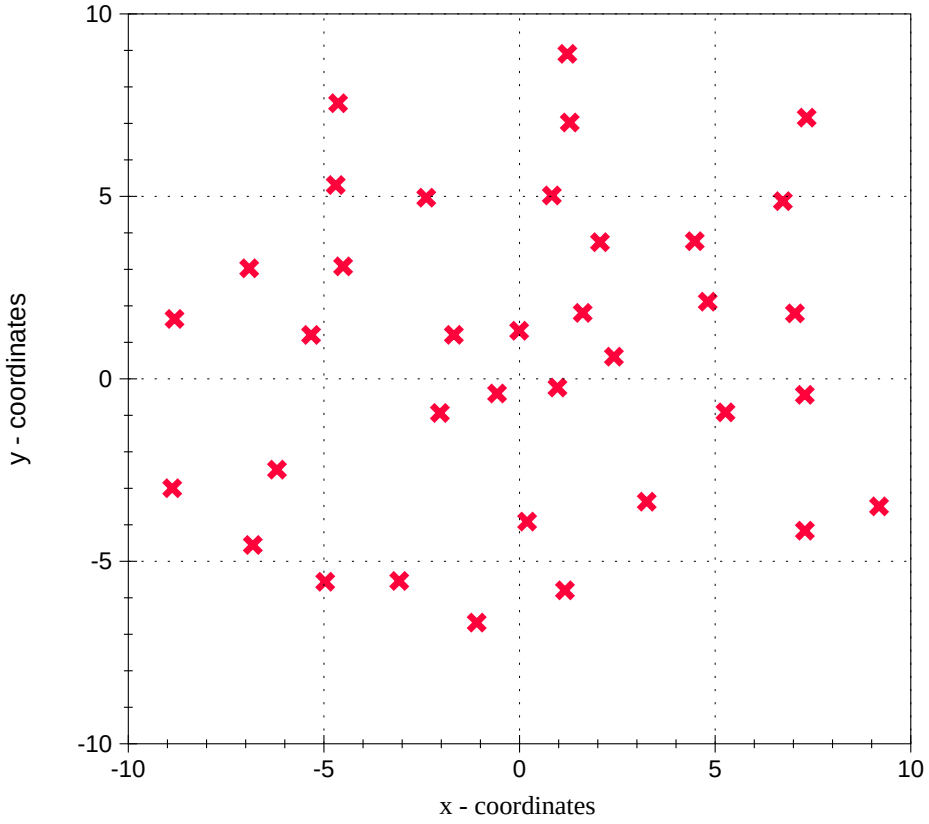

Figure 11 Final Positions of MRVs around Target

The final positions of all the MRVs that make it close to the target within the simulation time are shown in Fig. 11. As expected they cluster around the target (coordinate origin) and maintain some standoff distance between them. This standoff distance can be controlled with the parameters r_0 and r_b, along with the repulsive

potential gain k_3.

C-CODE OVERVIEW

The source code file name is **grad8.c**. All simulation data is read in from the support file called *data*. The following output files are generated:

1. The file *arrived* lists how many MRVs have arrived at the target at a given time.
2. The file *est_txy* provides the estimated (x, y) MRV positions for a given time step.
3. The file *diff_txy* provides the difference between the estimated and the actual MRV (x, y) locations at a given time step.
4. The file *xy* stores the (x, y) positions of each MRV without time information. This file is used to generate MRV tracks of their motion.
5. The file *txy* stores the actual MRV (x, y) locations at a given time step.

REFERENCES

1. Feddema, John T., Kwan S. Kwok, Brian J. Driessen, Spletzer, Barry L., and Weber, Thomas M. "Miniature Autonomous Robotic Vehicle (MARV)," *IEEE International Conference on Robotics and Automation*, Albuquerque, NM, April 1997. submitted for conference.
2. Junkins, John L. *An Introduction to Optimal Estimation of Dynamical Systems*. Sijthoff & Noordhoff International Publishers, Alphen aan den Rijn, Netherlands, 1978.
3. Strikwerda, T. E. and Junkins, John L. "Star pattern recognition and Spacecraft Attitude Determination," Technical report, U. S. Army Corps of Engineers, Engineering Topographic Laboratories, 1981.
4. Webb, Glenn V. and Junkins, John L. "A Path Planning Approach for the SSRMS in Operational Sapce: A Local Approach with Global Perspective," Technical report, Aerospace Engineering Department, Texas A&M University, May 1996.