Proceedings of the 1991 IEEE
International Conference on Robotics and Automation
Sacramento, California - April 1991

Potential Field Methods and Their Inherent Limitations for Mobile Robot Navigation

Yoram Koren
Johann Borenstein

Department of Mechanical Engineering and Applied Mechanics
The University of Michigan, Ann Arbor

ABSTRACT

Potential field methods are rapidly gaining popularity in obstacle avoidance applications for mobile robots and manipulators. While the potential field principle is particularly attractive because of its *elegance* and *simplicity*, substantial shortcomings have been identified as problems that are *inherent* to this principle. Based upon mathematical analysis, this paper presents a *systematic* criticism of the inherent problems. The heart of this analysis is a differential equation that combines the robot and the environment into a unified system. The identified problems are discussed in qualitative and theoretical terms and documented with experimental results from actual mobile robot runs.

1. Introduction

During the past few years, potential field methods (PFM) for *obstacle avoidance* have gained increased popularity among researchers in the field of robots and mobile robots. The idea of imaginary forces acting on a robot has been suggested by Andrews and Hogan [1] and Khatib [7]. In these approaches obstacles exert repulsive forces onto the robot, while the target applies an attractive force to the robot. The sum of all forces, the resultant force $\boldsymbol{R}$, determines the subsequent direction and speed of travel. One of the reasons for the popularity of this method is its *simplicity* and *elegance*. Simple PFMs can be implemented quickly and initially provide acceptable results without requiring many refinements. Krogh and Thorpe [8] suggest a *generalized potential field* method that combines *global* and *local path planning*.

Potential field methods have been implemented on mobile robots with real sensory data by Brooks [5], and by Arkin [2]. However, the robot in Arkin's work was very slow; it traversed an obstacle course at 0.12 cm/sec (0.4 feet/sec).

In our own previous research we have developed a PFM, called the *virtual force field* (VFF) method [3]. Through extensive experimental work with the VFF method, implemented on our mobile robot CARMEL, we have gained much insight in the strengths and weaknesses of this method. Among the weaknesses of the VFF method we identified problems that are inherent to PFMs in general. This experience prompted us to write this paper – to stimulate discussion on possible remedies and to caution of over-optimism in view of the simplicity and elegance of PFMs.

2. The Virtual Force Field (VFF) Method

The *virtual force field* (VFF) method is especially designed for real-time obstacle avoidance with fast mobile robots [3]. The VFF method allows fast, continuous, and smooth motion of the controlled vehicle among unexpected obstacles. We briefly explain the VFF method below.

The VFF method uses a two-dimensional Cartesian grid, called the *histogram grid* $\boldsymbol{C}$, for obstacle representation. Each cell (i,j) in the *histogram grid* holds a *certainty value* (CV) $c_{i,j}$ that represents the confidence of the algorithm in the existence of an obstacle at that location. This representation was derived from the *certainty grid* concept that was originally developed by Moravec and Elfes, [9]. In the *histogram grid*, CVs are incremented when the range reading from an ultrasonic sensor indicated the presence of an object at that cell.

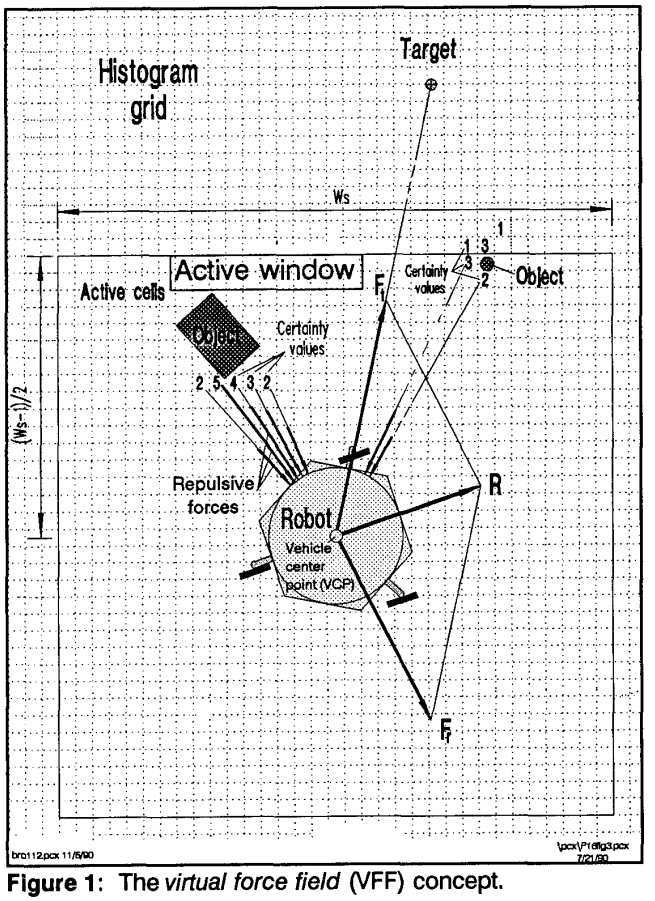


Figure 1: The *virtual force field* (VFF) concept.

Simultaneously, the potential field concept is applied to the *histogram grid* as shown in Fig. 1. It works as follows: As the vehicle moves, a window of $w_s \times w_s$ cells accompanies it, overlying a square region of C. We call this region the "*active region*" (denoted as C^*), and cells that momentarily belong to the *active region* are called "*active cells*" (denoted as $c^*_{i,j}$). In our current implementation, the size of the window is 33×33 cells (with a cell size of 10cm×10cm), and the window is always centered about the robot's position.

Each *active cell* exerts a virtual repulsive force $F_{i,j}$ toward the robot. The magnitude of this force is proportional to $c^*_{i,j}$ and inversely proportional to d^n, where d is the distance between the cell and the center of the vehicle, and n is a positive number.

$$F_{i,j} = \frac{F_{cr} W^n C_{i,j}}{d^n(i,j)} \left(\frac{x_i - x_0}{d(i,j)} \hat{x} + \frac{y_j - y_0}{d(i,j)} \hat{y} \right) \quad (1)$$

where

F_{cr} Repulsive force constant.

$d(i,j)$ Distance between *active cell* (i,j) and the robot.

$C_{i,j}$ Certainty value of *active cell* (i,j).

W The width of the mobile robot.

x_0, y_0 Robot's present coordinates.

x_i, y_j Coordinates of *active cell* (i,j).

In our implementation as well as in the following discussion we assume $n=2$. All virtual repulsive forces add up to yield the resultant repulsive force F_r

$$F_r = \sum_{i,j} F_{i,j} \quad (2)$$

Simultaneously, a virtual attractive force F_t of constant magnitude is applied to the vehicle, "pulling" it toward the target.

$$F_t = F_{ct} \left(\frac{x_t - x_0}{d_t} \hat{x} + \frac{y_t - y_0}{d_t} \hat{y} \right) \quad (3)$$

where F_{ct} is the target (attraction) force constant; d_t is the distance between the target and the robot; and x_t, y_t are the target coordinates.

Summation of F_r and F_t yields the *resultant force vector* R

$$R = F_r + F_t \quad (4)$$

The direction of R, (denoted δ and given in degrees), is used as the reference for the robot's steering-rate command Ω:

$$\Omega = k(\delta - \theta) \quad (5)$$

where k is the proportional constant for steering (sec^{-1}), θ is the current direction of travel (in degrees), and δ is the commanded direction of travel (in degrees). This equation will be utilized in the mathematical analysis.

3. Robot-Environment Mathematical Analysis

To analyze the behavior of the mobile robot in the presence of obstacles, first we introduce a differential equation that describes the robot's motions as affected by the environment. This equation is based on combining the steering system model with environmental parameters. The basic model for the robot's steering motor is a first-order differential equation given by

$$\tau\dot{\omega} + \omega = \Omega \quad (6)$$

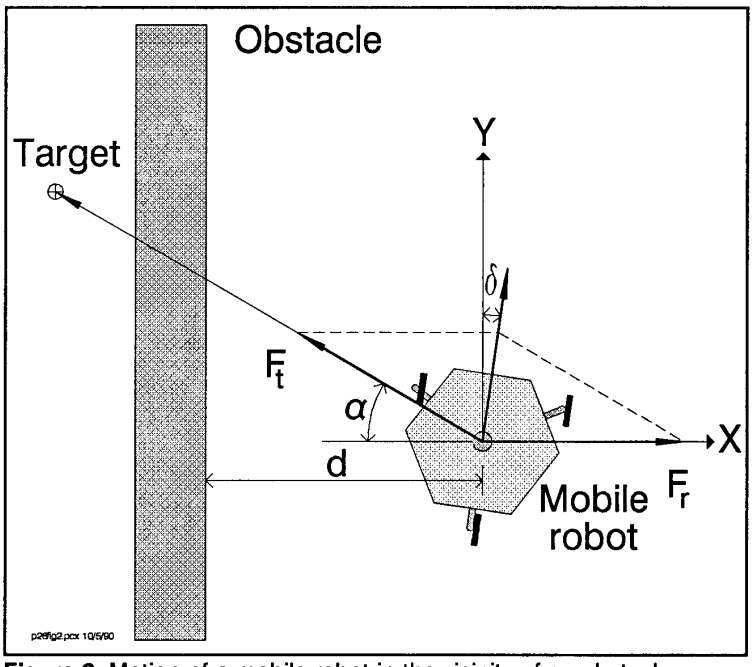


Figure 2: Motion of a mobile robot in the vicinity of an obstacle.

where Ω is the steering-rate command and ω is the actual steering-rate; τ is the time-constant of the steering motor and is dependent on the robot's mass. Note that the actual relationship is more complicated than the one represented in Eq. (6), since our mobile robot also includes a corrective network in the steering controller.

Substituting Eq. (5) and $\dot{\theta} = \omega$ into Eq. (6) yields the steering equation

$$\tau\ddot{\theta} + \dot{\theta} + k\theta = k\delta \tag{7}$$

For the development of the *robot-environment* model, we assume that the robot travels on one side of a long obstacle, with the objective to reach a target located on the other side of the obstacle, as depicted in Fig. 2. Since the object is long, the repulsive force has only a lateral component in the direction of X

$$F_r = \frac{F_{rx} W^n}{d^n} \tag{8}$$

where F_{rx} is a constant proportional to F_{cr}. Note that Eq. (8) holds also for systems that do not use grid-type world models. The resultant force in the X-direction is $F_r - F_{ct} \cos\alpha$, where F_{ct} is the target force constant. The reference command δ to the steering control loop is calculated by the equation

$$\tan\delta = \frac{F_r - F_{ct}\cos\alpha}{F_{ct}\sin\alpha} \tag{9}$$

Differentiating δ in Eq. (9) with respect to time yields

$$\dot{\delta} = -N\dot{d} + M\dot{\alpha} \tag{10}$$

where

$$N = \frac{nW^n f \cos^2\delta}{d^{n+1}\sin\alpha} \tag{11}$$

$$M = \left(1 - \frac{\tan\delta}{\tan\alpha}\right)\cos^2\delta \tag{12}$$

and

$$f = \frac{F_{rx}}{F_{ct}} \tag{13}$$

Note that for small δ, the parameter N is independent of δ and $M=1$.

While the robot may move in an arbitrary direction in the XY-plane at a velocity V, its velocity component in the X-direction is always given by

$$\dot{d} = V\sin\theta \tag{14}$$

By substituting Eq. (14) into (10) and the resulting equation into Eq. (7), we obtain

$$\tau\dddot{\theta} + \ddot{\theta} + k\dot{\theta} + kN(d,\alpha,\delta)\, V\sin\theta = kM(\alpha,\delta)\,\dot{\alpha} \tag{15}$$

Equation (15) is important since it describes the behavior of the robot in response to relative changes in the environment. Eq. (15) contains parameters of the robot (k, V) and the PFM (n, f). The environment, as related to the robot, is also represented in Eq. (15) by (d, α), and the response of the steering angle θ to changes in these parameters is shown.

If we assume that the robot moves almost parallel to the obstacle, and therefore is commanded by small angles δ that result in small angles θ, Eq. (15) can be rewritten

$$\tau\dddot{\theta} + \ddot{\theta} + k\dot{\theta} + kN(\alpha)V\theta = k\dot{\alpha} \tag{16}$$

We may assume that α is changing slowly, such that $k\dot{\alpha}$ is zero and N is constant. This results in a linear, time-invariant differential equation. For this condition, the characteristic equation of the Laplace transform of Eq. (16) is

$$\tau s^3 + s^2 + ks + kNV = 0 \quad (17)$$

Note that Eq. (17) holds true for continuous control systems. For sampled data systems, a delay e^{-sT} is introduced in the control loop and Eq. (17) becomes

$$\tau s^3 + s^2 + ks + kNVe^{-sT} = 0 \quad (18)$$

This delay T may be quite significant; sampling times of up to 3 seconds are reported in the literature [2]. However, if the sampling time T is very short, the delay can be approximated by $e^{-sT} = 1 - sT$.

Substituting this expression in Eq. (18) yields the following characteristic equation

$$\tau s^3 + s^2 + [k(1 - NVT)]s + kNV = 0 \quad (19)$$

In order to determine the stability limit of the moving robot, the *Routh stability criterion* may be applied to Eq. (19), which yields the following condition

$$1 - NV(\tau + T) > 0 \quad (20)$$

with N given in Eq. (11).

The condition stated in Eq. (20) clearly shows a major drawback of the PFM: *The configuration space might include regions in which Eq. (20) is not satisfied and the robot will begin to oscillate.* The condition is harder to be satisfied for larger V, τ, and T; for example, if a heavy vehicle moves at high speed, it is more likely that oscillations occur.

Obviously, unstable conditions exist also for the general environment-robot model in Eq. (15). However, for the general nonlinear time-varying model of Eq. (15) it is difficult to find a simple stability rule like the one given in Eq. (20).

The parameter that can be tuned to guarantee stability is f which is defined as the ratio between the *repulsive force constant* F_{cr} (defined in Eq. 1) and the *target force constant* F_{ct} (defined in Eq. 3). The value of f for a particular system should be determined experimentally, as explained in the following section.

4. The Experimental System

We have verified our theoretical considerations on our mobile robot, CARMEL (*C*omputer-*A*ided *R*obotics for *M*aintenance, *E*mergency, and *L*ife support). CARMEL is based on a commercially available mobile platform

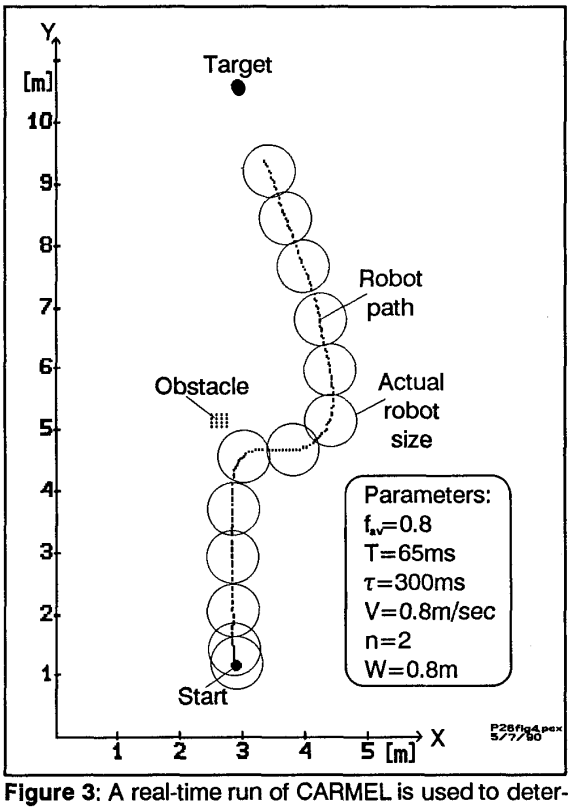


Figure 3: A real-time run of CARMEL is used to determine the minimal ratio between repulsive and attractive force constants.

with a unique three-wheel drive (synchro-drive) that permits omnidirectional steering [6]. This platform has a maximum travel speed of V_{max} = 0.8 m/sec and a maximum steering rate of Ω = 120 deg/sec; it weighs about 125 kg. A Z-80 on-board computer serves as the low-level controller of the vehicle.

We equipped this vehicle with a ring of 24 ultrasonic sensors; the ring has a diameter of 0.8m. Two computers were added to the platform: a 20Mhz, 80386-based AT-compatible that runs the VFF obstacle avoidance algorithm, and a PC-compatible single-board computer to control the sensors.

In the experiments described in this paper, the ultrasonic sensors were *disabled*, and obstacles were manually inserted into the *histogram grid*. This measure was necessary to insure that experiments were conducted under repeatable conditions, without noisy and inaccurate sensory data. However, in all experiments we used CARMEL, a *real* mobile robot, to ensure realistic conditions concerning system dynamics and sampling time.

A typical real-time run with CARMEL is depicted in Fig. 3. In this experiment CARMEL traveled from "*Start*" to "*Target*." A small obstacle (approx. 40×40cm) was inserted into the *histogram grid*. This experiment was performed to define the relative strength of the repulsive force (determined by F_{cr}, in Eq. 1) to the attractive force (determined by F_{ct}, in Eq. 3) such that the robot would just barely avoid the obstacle when approaching it head-on, at maximum speed. To ensure repeatability, the obstacle was manually programmed for subsequent experiments. Fig. 3 shows the resulting path of the robot. The actual position of the robot at one-second intervals is shown by circles which are plotted to scale according to the 0.8m diameter of CARMEL's sensor ring.

It should be noted that due to *quantization* and the finite size of the *active window* (see discussion in Section 2.3), F_{rx} (defined in Eq. 8) is not constant but rather fluctuates as a function of the robot's distance from an obstacle. Consequently, f (defined in Eq. 13) varies for our system. For this reason, we use an average value f_{av} to compare our experimental parameters with the analytical ones. With the experiment of Fig. 3, we determined $f_{av}=0.8$; this value was then used in all the experiments described in this paper.

5. Problems with Potential Field Methods

In the course of our experimental work with the VFF algorithm, we identified the following 4 significant problems that are inherent to PFMs and independent of the particular implementation:

1. Trap situations due to local minima (cyclic behavior).
2. No passage between closely spaced obstacles.
3. Oscillations in the presence of obstacles.
4. Oscillations in narrow passages.

The following sections present a detailed discussion.

5.1 Trap Situations Due to Local Minima

Perhaps the best-known and most often-cited problem with PFMs is the problem of *local minima* or *trap-situations* [1], [10]. A *trap-situation* may occur when the robot runs into a dead end (e.g., inside a U-shaped obstacle). Traps can be created by a variety of different obstacle configurations, and different types of traps can be distinguished. However, *trap-situations* can be resolved by heuristic or global recovery.

Trap-situations that are remedied with heuristic recovery rules are likely to result in a non-optimal path. For this reason we abandoned the heuristic recovery approach in favor of an integrated *global path planner* (GPP). With this method, the *local path planner* (LPP) monitors the robot's path; when a *trap-situation* is detected the GPP is invoked to plan a new path based on the available information.

5.2 No Passage Between Closely Spaced Obstacles

Fig. 4 shows a mobile robot at an attempt to pass among two closely spaced obstacles (e.g., passing through a door frame). With PFMs, the repulsive forces from obstacle 1 and 2 are combined into the two *lumped repulsive forces* F'_{r1} and F'_{r2}, respectively. The sum of all repulsive forces ($F_r = F'_{r1} + F'_{r2}$) points straight away from the opening between the two obstacles. Depending on the relative magnitude of the target-directed force F_t the robot will either approach the opening further, or it will turn away (as depicted in Fig. 4).

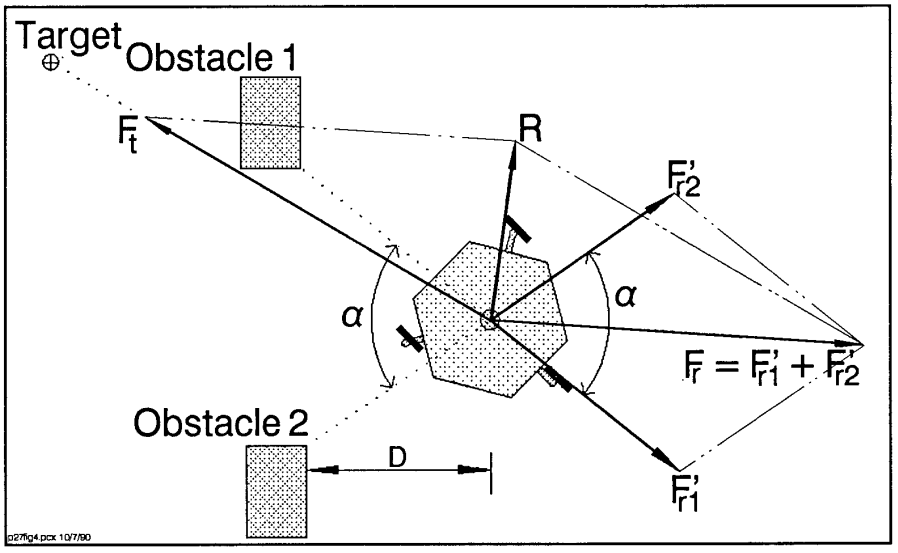


Figure 4:
Under PFM control, the robot does not pass among densely spaced obstacles.

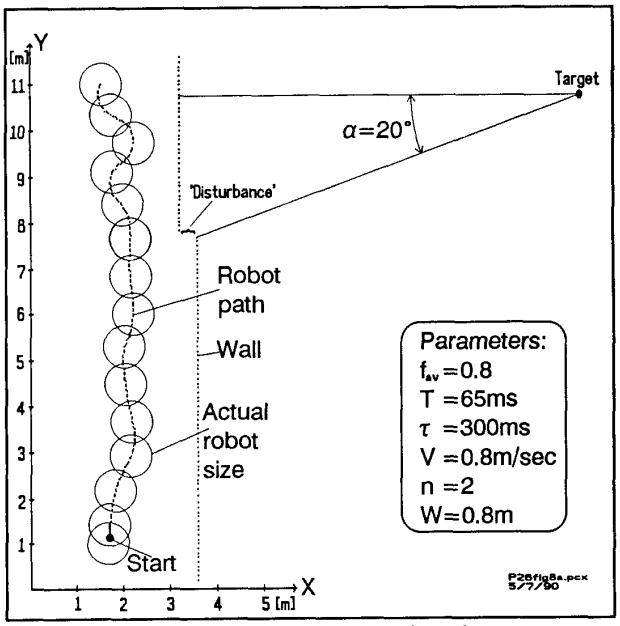


Figure 5: Instable motion results when the robot encounters a disturbance at $\alpha<\alpha_{cr}$, while traveling in the vicinity of an obstacle.

5.3 Oscillations in the Presence of Obstacles

One of the most significant limitations of potential field methods is their tendency to cause unstable motion in the presence of obstacles.

The mathematical conditions for the onset of unstable motion may be derived from the *robot-environment* model developed in Section 3. Substituting Eq. (11) for small δ (i.e., $\cos\delta \approx 1$) into Eq. (20) yields the stability condition

$$d^{n+1} \geq \frac{nV(\tau+T)W^n f}{\sin\alpha} \qquad (21)$$

When the robot travels alongside the obstacle under steady state conditions, the sum of the lateral forces is zero, so that

$$F_r = F_{cl}\cos\alpha \qquad (22)$$

Substituting Eq. (8) into Eq. (22) yields

$$\frac{F_r W^n}{d^n} = F_{cl}\cos\alpha \qquad (23)$$

Using f from Eq. (13) and substituting d from Eq. (23) into Eq. (21) we obtain the stability condition

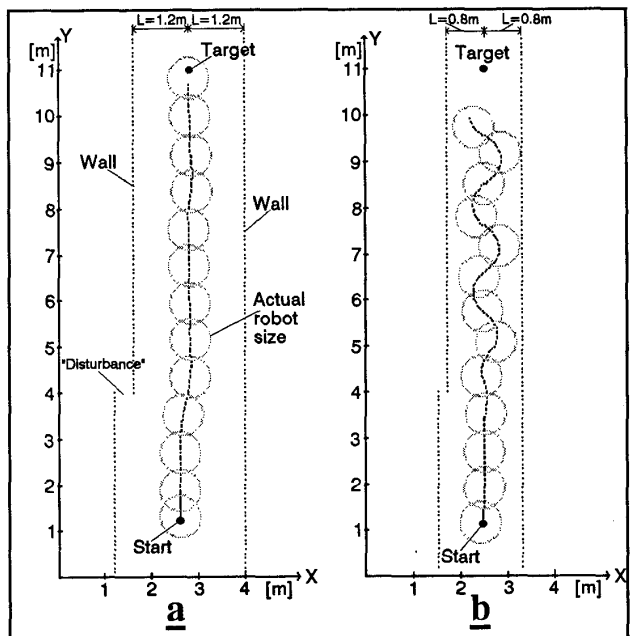
$$f \geq \cos\alpha \left[\frac{nV(\tau+T)}{W\tan\alpha} \right]^n \qquad (24)$$

Figure 5 shows what happens when condition (24) is not met. In this real-time experiment CARMEL traveled from "*Start*" to "*Target*." CARMEL was forced to travel alongside a wall which obstructed the robot's path. Note that the "wall" was manually programmed into the *histogram grid* and is shown as a straight line of dots which indicate filled cells in the *histogram grid*. Initially, the robot's path was fairly straight. However, a discontinuity in the wall (at $\alpha=20°$) caused the robot to enter into oscillatory and unstable motion.

5.4 Oscillations in Narrow Passages

A similar yet more severe problem with PFMs occurs when the robot travels in narrow corridors, in which the robot experiences repulsive forces simultaneously from opposite sides.

To analyze this problem we assume that the robot moves in the middle of a long passage as depicted in Fig. 6. At point Y=4m the left wall protrudes into the passage and causes a perturbation in the robot's path. Figure 6 shows the path resulting from



Stable motion in wide corridor V=0.8m/s

Unstable motion in narrow corridor. V=0.8m/sec.

Figure 6: Stable and unstable motion in corridors.

CARMEL traveling in a wide (case a) and a narrow (case b) corridor. The parameters in this actual run were $f_{av}=0.8$, $n=2$, $T=0.065$sec, $\tau=0.3$sec, and $V=0.8$m/sec. In the wide corridor, a sudden change in the width of the corridor causes the robot to smoothly adjust its path. In the narrow corridor, however, the sudden change excites the robot into unstable oscillations and eventually a collision. A thorough mathematical analysis shows that the stability criterion for this case is

$$L^{n+1} > 2nW^nV(T+\tau)f \qquad (25)$$

Eq. (25) is the stability limit which relates the robot parameters (W, V, and τ) to the PFM parameters (f and n,) and the environment parameter L.

6. Conclusions

Based on a rigorous mathematical analysis, we have presented a *systematic* overview and a critical discussion on the inherent problems of potential field methods (PFMs). Specifically, we have identified 4 distinct drawbacks. Two of these drawbacks are related to the possibility of oscillations which become apparent only when the PFM is implemented in a high-speed real-time system. Most researchers concentrate their efforts on simulation programs of potential fields; they don't seem to be aware of the substantial, possibly unresolvable problems that are bound to surface once actual implementation in an experimental system is attempted. Other researchers work with actual mobile robots, but at slow speeds which conceal the disadvantages of the PFMs.

For these reasons, we have abandoned potential field methods altogether, and developed a new method for fast obstacle avoidance. This method, called the *vector field histogram* (VFH) method, produces smooth, non-oscillatory motion, while sampling time and hardware are identical to those used in the VFF method. The VFH method was introduced in [4].

This work was sponsored by the Department of Energy Grant DE-FG02-86NE37969

7. References

[1] Andrews, J. R. and Hogan, N., "Impedance Control as a Framework for Implementing Obstacle Avoidance in a Manipulator." *Control of Manufacturing Processes and Robotic Systems*, Eds. Hardt, D. E. and Book, W., ASME, Boston, 1983, pp. 243-251.

[2] Arkin, R. C., "Motor Schema-Based Mobile Robot Navigation." *The International Journal of Robotics Research*, August 1989, pp. 92-112.

[3] Borenstein, J. and Koren, Y., "Real-time Obstacle Avoidance for Fast Mobile Robots." *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 19, No. 5, Sept/Oct, 1989, pp. 1179-1187.

[4] Borenstein, J. and Koren, Y., "Real-time Obstacle Avoidance for Fast Mobile Robots in Cluttered Environments." *The 1990 IEEE International Conference on Robotics and Automation*, Cincinnati, Ohio, May 13-18, 1990, pp. 572-577.

[5] Brooks, R. A., "A Robust Layered Control System for a Mobile Robot." *IEEE Journal of Robotics and Automation*, Vol. RA-2, No. 1, 1986, pp. 14-23.

[6] Cybermation, "K2A Mobile Platform." *Commercial Offer*, 5457 JAE Valley Road, Roanoke, VA 24014.

[7] Khatib, O., "Real-Time Obstacle Avoidance for Manipulators and Mobile Robots." *1985 IEEE International Conference on Robotics and Automation*, St. Louis, Missouri, March 25-28, 1990, pp. 500-505.

[8] Krogh, B. H. and Thorpe, C. E., "Integrated Path Planning and Dynamic Steering Control for Autonomous Vehicles." *Proceedings of the 1986 IEEE International Conference on Robotics and Automation*, San Francisco, California, April 7-10, 1986, pp. 1664-1669.

[9] Moravec, H. P. and Elfes, A., "High Resolution Maps from Wide Angle Sonar." *IEEE Conference on Robotics and Automation*, Washington D.C., 1985, pp. 116-121.

[10] Tilove, R. B., "Local Obstacle Avoidance for Mobile Robots Based on the Method of Artificial Potentials." *General Motors Research Laboratories, Research Publication GMR-6650*, September 1989.