Re-Active Vector Equilibrium: A Novel Method of Autonomous Vehicle Navigation Using Artificial Potential Fields

by

Cameron Frazier, B.Eng

A Thesis submitted to

the Faculty of Graduate and Post-Doctorate Studies

in partial fulfilment of

the requirements for the degree of

Master of Applied Science

in

Mechanical Engineering

University of Ottawa

Ottawa, Ontario, Canada

April 2015

Abstract

The use of potential field based navigation schemes in robotics has been limited by inherent local minima issues. Local minima traps, small passages, unstable motion, and targets positioned near objects all pose major concerns when using potential fields for local vehicle control. This work proposes a new algorithm, "Re-Active Vector Equilibrium" (RAVE) that mitigates many of these issues. The vehicle representation model is expanded to use multiple points subject to potential calculation and the addition of two forces, a velocity dependent "risk force" ($\vec{F}_{rsk}$) and a velocity and direction dependent "tangential force" ($\vec{F}_{tan}$). The vehicle representation model is also expanded from a single reactive point to a series of points that define the vehicle body, providing better and simpler vehicle control. This has the effect of simplifying the required calculations at the cost of increasing the calculation count. The risk force, $\vec{F}_{rsk}$, allows for dynamic adaptation to the immediate environment by acting in opposition to the net obstacle force, and is inversely proportional to the vehicle speed. The tangential force, $\vec{F}_{tan}$, encourages better wall-following behaviour and provides a biasing mechanism to resolve obstacle aligned with target local minima issues.

Acknowledgments

I would like the acknowledge and thank the following people with the utmost gratitude possible:

Professor Natalie Baddour for her constant support, advice, and most importantly, in giving me the chance to return to academia. Without her guidance, support and honest comments I would not have been greatly challenged to complete this research.

Professor Alex Ellery for his support in pursuing my Masters and guiding my focus onto robotics. I would also like to thank him for including me in the Kapvik project and the SEEG research group

Brian Lynch, PhD for his collaboration and partnership on many parts of the RAVE research. His enthusiasm and curiosity into the subject were key to my pursuing the line of research presented here.

My parents for their support throughout my Masters, and especially in these last few months; it has been invaluable to me completing my thesis.

My friends **Jean-François Thibault**, **Matt Cross**, **Chris Nicol** and the whole of the SEEG team, as well as the other residents of the Jo Wong Lab for their

understanding, constructive comments, and support over the past couple of years.

This is by no means a complete list – or even a plurity of that list – of those I should be thanking and so I'd like to say "Thank you" to all of those un-named but important people. Your help was valued.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

Since the advent of extra-terrestrial exploration with the successful landing of the Soviet *Luna 9* mission, launching exploration from the realm of science fiction into reality, the exploration of space has been led by robotic missions. From landing on the planets known since antiquity, to missions that extend to the reaches of our solar system and beyond, all space-faring nations have used robotic missions and probes to expand our reach into the cosmos. The dangerous task of exploring new, unknown, and hazardous worlds is given to these devices, albeit with close control and observation from commanders back on Earth. Particularly for the often ill-fated Martian exploration missions – the extended family of spacecraft and rovers[1] include examples from the Soviet, American, Russian, Japanese, European, British, Chinese, and soon also Indian space agencies – the ability for an exploration rover to safely achieve mission goals is paramount.

[1]The term "rover" is used when referring to robotic land vehicles specifically, and "vehicles" when any form of robotic vehicle (land, atmospheric, nautical) is applicable.

As missions reached beyond the orbit of the Moon, many performance-impeding issues were confirmed; the most prominent issue relevant to this work is the limited communication transmission windows and the delay in the transmission signal. The limited communication windows force missions to operate on fixed information schedules, potentially decreasing the operational duty cycle. For example a mission to Mars, with the communication window of only a couple hours per day at most, and a wide-ranging transmission delay in communications between, direct human control of vehicles becomes untenable. To increase the capability, reliability, and ruggedness of deployed rovers, the development of techniques for autonomous navigation (including visual odometry, tracking motion during and following a traverse, visual hazard detection and localization) were adapted for mission use and deployed. Advanced autonomy of the rover increases its capability through the ability to perform more complex or riskier actions, operations beyond line-of-sight as an example. Reliability is also increased by having the means to decrease the risk of mission-ending incidents, such as the miring of Mars Exploration Rover (MER) 'Spirit' in soft regolith. Rover autonomy also increases its ruggedness by permitting operation within a larger work envelope. The subject of this thesis is techniques to increase rover autonomy, specifically through obstacle avoidance and local control via reactive means.

Increasing the autonomy of exploration vehicles also permits the application of the developing field of "autonomous science". This relatively new field uses the robotic vehicle to detect possible scientific targets, to traverse to the most promising location, perform on-board analysis, and then to report back to Earth the developed results.

This field holds great potential as it would allow vehicles to operate without intensive supervision from commanders on Earth and permit longer distance missions and/or missions with many vehicles working cooperatively.

Existing terrestrial navigation techniques often cannot be directly employed on exploration rover missions owing to the limited capabilities of the flight vehicles. For example, limitations on available power for sensors, limited computational cycles for algorithm processing, insufficient temporary memory for data storage, and the lack of a global position referencing system all pose potential real-world limits for developed algorithms. One set of algorithms that have been proposed to address problems relating to rover motion and obstacle avoidance is based on *Artificial Potential Fields* (APF). In principle, these algorithms operate using a collection of repulsers (detected or known obstacles and the vehicle) and attractors (a traverse goal) modelled as point charges, negative for repulsers and positive for attractors. In this way, a map of the force gradients can be created. The vehicle path is then defined by the steepest descent gradient between the rover's current position and the end target – in effect the vehicle "flows downhill" to the target. The APF-based algorithms are desirable as they are relatively simple calculations and can provide smooth and efficient vehicle motion. Some authors describe them as elegant, a notion this author favours. There are several problems that appear in all APF work and are rooted in the resolution of local minima, the pools and eddies in the "downhill flow" that can trap a vehicle and prevent it from reaching the desired target, the global minimum. Additionally, in real-world applications, APF solutions are only deterministic in a broad sense as

many different paths may result from the same set of start and end poses, limiting usage to the current command paradigms used on modern missions.

The focus of APF systems is split into the areas of either global planning, where a known global map is available, or local/reaction planning, where only the immediate rover space is considered. For the global planner the work focuses on formulations and algorithms to remove local minima while providing efficient paths. Global planning work is of limited use in exploration vehicles as the global map is unknown, or at best, grossly incomplete. On the other hand, the use of APF as a local planner and reaction controller has been studied in many variations, and remains an interesting point of study. Regardless of usage, the variety of implementations demonstrates the adaptability of APF methods, even though their implementation is limited by well-recognized problems. Should these problems be overcome or circumvented, APF navigation could become a useful tool for planetary exploration.

1.1 Motivation

Current styles of robotic vehicle tele-operation schemes can be grouped into one of three arrangements of controller/vehicle interaction. First are the situations where the operator is local to the operation theatre and has access to situational information that goes beyond that returned by the vehicle. In addition, the communication link, be it wireless or wired, is of sufficient throughput and does not present a limiting bottleneck to operations. Examples of these vehicles are those used for Explosive

Ordinance Disposal (EOD)[3, 4] and those used in the clean-up of the Fukushima Daiichi nuclear power plant meltdown [5, 6]. These vehicles are dependant upon a local operator with access to additional situation information via line of sight or closed circuit television (CCTV) to ensure safe and efficient operation.

Second are the operation scenarios in which the local vehicle environment is unknown outside that recorded by the vehicle-mounted sensors and returned via high throughput communications links, similar to the first scenario. Absent the situational awareness provided by the off-vehicle sensors, the operator must exercise enhanced caution with navigating areas, and in general is subject to reduced vehicle performance in either speed or ability. Examples of these systems include the previously mentioned EOD robots, and also include vehicles such as Remotely Operated and Autonomous Underwater Vehicles (ROV and AUV respectively). A case could be made for the recently deployed Chinese rover, *Yùtù*[2] and similar lunar missions, given the reduced communications delay relative to other extra-terrestrial exploration vehicles. There is some overlap with respect to Unmanned Ground Vehicles (UGV), as the enhanced situational awareness for the operator is dependant upon the mission. ROV and AUV operations more fully represent the limits of this scenario, as there is a merging of high data throughput and severe isolation between the operator and vehicle. In an interview with Mr. Conrad, an active ROV operator in the offshore oil industry ([7]), it was made clear that the minimization of communication delay and the amount of data and its fusion into a clear presentation is key to efficient operation. In the case

[2] *Yùtù*,玉兔: Jade Rabbit

of offshore oil production, the communicated data includes multiple high-definition (HD) video streams, ROV telemetry, and large amounts of environment profiling data generated from sonar systems.

The final scenario is where the majority of planetary exploration missions lie, namely remote and isolated positions precluding external situational awareness, and presenting great challenges in terms of limited communication windows, signal delay and data throughput. With active examples in the Mars Exploration Rover (MER), Mars Science Laboratory (MSL), and Voyager 1 and 2 missions, these limits are a major driver for increased autonomy of exploration vehicles. For vehicles on Mars, the one-way communications delay varies between 4 and 21 minutes ([8, 9]) and for comparison, the Voyager missions have delays of just over 17 and 14 hours respectively. While there is limited autonomy aboard the MER and MSL missions, autonomy expanded beyond the earlier Mars Pathfinder/Sojourner mission to include global path planning, hazard avoidance and visual pose estimation [9]. For the MER missions, this limited autonomy has allowed for many instances where the vehicle was allowed to perform traverses safely and autonomously [10], though the more common situation relies on directed drives commanded from Earth. This scenario also applies to lunar polar missions, where vehicle position as it moves about a crater would create limited communication windows.

The common problem in all three of the above scenarios is that better detection and reaction to the current vehicle environment is required. Scenarios one and two attempt to solve the problem with a dependence upon numerous high data rate sensors

(including the operator as a "sensor input") and high throughput communications links. By providing the operator with a vast amount of clear data, the vehicle can be commanded safely and quickly. With the second and third scenarios, there is a restriction of the sensors to the vehicle, and with that an increased opportunity for autonomy and/or data pre-processing, at the cost of a potential reduction in situational awareness. Accordingly, with the reduction in situational awareness, the speed and fidelity at which the vehicle can operate and perform actions is reduced. Finally, the third scenario presents the limitations often found in planetary exploration vehicles, that is reduced data communication rates, limited communication windows with increased communication delays, along with unknown and potentially hazardous environments.

In all three scenarios, the operation and performance of the vehicle could be improved by increased vehicle autonomy. Increasing vehicle autonomy and minimizing dependence on low throughput data links will present exploration missions with the ability to perform autonomous science, at increased operational speeds, and with potentially increased robustness. Additionally, these improvements will present beneficial technologies to current terrestrial UGV, ROV, and AUV designs.

It is with this motivation that the research for this thesis is undertaken. The goal of this thesis is to provide an improvement to the ability of a tele-operated vehicle to operate in hazardous and remote environments, with reduced mission risk and an increased ability to operate free of terrestrial bonds. To this end, this thesis will investigate the use of a modified APF method for reactive navigation of assisted

tele-operations, either as direct control or as a waypoint following system.

1.2 Scope

The scope of this thesis is divided into two main, reinforcing topics: *i*) obstacle point sensing and force calculation, *ii*) calculation of vehicle control output.

Specifically, this thesis presents a novel algorithm for obstacle point sensing and force calculation. The presented algorithm is the core novelty of this thesis and will remain the focus throughout.

1.3 Goals

The goal of this thesis is the development of a re-active navigation algorithm, and its implementation and evaluation in both simulation and hardware. The goals of this research are to:

i) Develop a 2D APF-based reactive controller for a Lidar equipped rover platform

ii) Evaluate the developed control algorithm via simulations

iii) Demonstrate the developed control algorithm in hardware

1.4 Contributions

1.4.1 Multipoint Vehicle Model

Building upon the work of Khatib and others ([11–14]) who demonstrate using multiple *Points Subject to a Potential* in manipulator control, presented here is the use of a multiple point vehicle model in the Re-Active Vector Equilibrium algorithm. The use of a multi-point vehicle model simplifies the calculation of repulsive forces and permits the direct calculation of repulsive torques at act upon the vehicle. This contribution is discussed in Section 3.4.

1.4.2 Risk Force Amendments

Expanding on the work of Lynch ([15]), the *Risk Force* is developed to use a vehicle specific velocity decay coefficient (c_{decay}, see Equation 3.7) within the calculation of ϵ (see Equation 3.8). This change reframes the calculation of ϵ in terms of the vehicle properties, as opposed to a tunable parameter. This contribution is discussed in Section 3.2.6.

1.4.3 Simulation Environment

As part of this work, a simulation environment is presented and publicly made available through [16]. This simulator was developed using the Mathworks Matlab software

and features a modular design for maps, vehicle models, and obstacle detection models. As referenced, the simulator includes all maps used in this thesis, a vehicle model characterizing the Clearpath Robotics A200 Husky, and a line-of-sight lidar model. This contribution is discussed in Chapter 4.

1.5 Outline of the thesis

The outline of this thesis is as follows. Chapter 1 provides an introduction to the research and presents the motivation and scope of the work presented.

Chapter 2 presents the background theory of APF approaches to robotic ground vehicle navigation. Reviews of select papers are discussed, and a critical overview of prior research is presented. In addition, the Re-Active Vector Equilibrium (RAVE) algorithm developed for this thesis will be presented with a discussion of the related benefits and limitations.

Chapter 3 describes the details of the proposed RAVE algorithm. Each force is described and the overall algorithm is presented.

Chapter 4 discusses the implementation of the algorithm in simulation, in particular the simulation design, and the major modelling components. Also presented are the methods used in the genetic algorithm for parameter tuning. Also presented is the hardware used for the analogue demonstration. A summary of the hardware used, software frameworks employed, and developed support software is also included.

Chapter 5 discusses the results of the simulation testing and hardware proof-of-concept. This chapter is also divided into two major sections, for the simulation and analogue hardware evaluations, respectively.

Chapter 6 summarizes the work completed and results acquired from this research. Future work to extend this research is also presented and discussed in this chapter.

Chapter 2

Background and Literature Review

2.1 Background

2.1.1 Artificial Potential Fields

Artificial Potential Fields, (APF), are the presentation of a vector field, $\vec{F} = \nabla\mathbb{U}$, for the map of a potential function, $\mathbb{U}$ [17–20]. In this thesis, $\vec{A}$ represents a vector quantity, $\hat{A}$ represents the unit vector corresponding to vector $\vec{A}$, $\mathbb{A}$ represents a scalar map, A represents a scalar value, the magnitude of vector $\vec{A}$. Algorithms that employ APF for navigation operate using a collection of repulsers (such as detected or known obstacles) and attractors (such as the traverse goal). The nomenclature of artificial potential fields comes from the possibility that any or all of the repulsers or attractors may be an artificial or virtual entities [11, 17, 21–24]. These repulsers and attractors are modelled as negative and positive point charges respectively, with the sample location – the point for which the field is being calculated – being represented also

as a positive charge [21]. Using this aggregate collection, a map of the local energy potentials can be created, and hence a gradient (force) vector map. The vehicle path is then defined by the steepest descent gradient between the rover's current position and the end target. In effect, the vehicle "flows downhill" to the target [25].

The focus of APF systems is commonly split into the areas of global planning, where a known global map is available, and local/reaction planning, where only the immediate rover space is considered [26]. For the global planner, the work focuses on formulations and algorithms to remove local minima while providing efficient paths [15, 17, 18, 27, 28]. Global planning is of limited use in exploration vehicles as the global map is unknown or at best, grossly incomplete. However, the use of APF as a local planner and reaction controller has been studied in many variations, a short discussion is provided in Sections 2.3 and 2.4, and remains an interesting point of study[29–31]. Regardless of usage, the variety of implementations demonstrates the adaptability of APF methods, even though their implementation is limited by well-recognized problems which are discussed in Section 2.5. Should these problems be addressed, APF navigation could become a useful tool for planetary exploration.

2.1.2 Non-APF Approaches

Though not covered in this thesis, there are several other approaches to the path planning/local control problem. A brief discussion and summary of some approaches are presented below. Further discussion is not presented.

2.1.2.1 Graph Searching

When the search space (map) is known or assumed, potentially exhaustive graph searches can be employed to find valid and optimal paths between two points [32]. In the context of path planning, a graph is a set of nodes and links that represent the path solution space. In an unstructured arrangement, such as when using a rapidly exploring random tree, nodes can represent the sample set [33, 34]. In a structured environment, these nodes can represent cells within an occupancy map [35, 36]. In an occupancy map the mapped area is divided up onto sections, commonly a grid of squares, which are marked as either unoccupied or occupied. In the simplest arrangement an occupancy map defines the environment as a bitmap, see Figure 2.4 for an example of this representation. In either the structured or unstructured cases, the links between the nodes define the set of possible paths and can represent any desired metric that varies between the nodes, e.g. distance, speed, energy use, etc, and any of these can be used as the optimization metric. Graph-based searching algorithms seek to optimize this defined metric.

Naïve (uninformed) searches do not employ a heuristic mechanism to provide a progression bias. A good example of a naïve search is Dijkstra's algorithms. Dijkstra's algorithm is a logical method of progressively stepping through each vertex of a graph, calculating and storing the minimum cost to root (start node) [37]. The calculation of each successive node from the root is executed sequentially until the shortest path to the root is found.

Heuristic (informed) searches use a bias metric to reduce the search space when possible. Well-known examples are the A* and D* family of algorithms [35, 36, 38–41]. The A* search algorithm operates by comparing a heuristic, defined by an *evaluation function*, and preferentially expanding to the node which best minimizes that heuristic. For dynamic maps, the A* algorithm requires a complete recalculation of the resultant path. The D* algorithm extends the A* concept to include *backpointers* to connect nodes with respect to the next node in the path. This relative connectivity provides D* algorithm with the capacity to recalculate local regions of the map and allows for the inclusion of new or dynamic information.

2.1.2.2 Other Planning Methods

Many other rover motion planning methods exist for both local and global contexts. Included in these are simulated annealing, which probabilistically models the graph as an annealing solid [31, 42–44]. As a cooling solid re-crystallizes, the crystals seek the lowest energy state, and this low energy state translates into a near-optimal path. The iterative Markov chain optimization drives the inability of simulated annealing to provide an optimal path and the associated computational cost limits usage in online vehicle navigation. However, it is effective at providing near optimal solutions to large NP-hard problems, such as the well known "Travelling Salesman Problem" which seeks to find the shortest route that passes through all points in a graph [45]. Also existing are behaviour-based algorithms which execute defined behaviours given the appropriate situational triggers, including both low level behaviours ("move

forward" or "turn right") and high level behaviours (or schema) that are developed by compounding lower level behaviours ("avoid obstacle" or "move to target") [9, 46, 47]. The NASA rover *Sojourner* made use of specific behaviour actions to overcome specific defined situations.

Many of these other planning methods can be combined with an APF approach for path planning solutions [31, 44, 46]. In Arkin's paper, the presented behaviour schema uses velocity potentials for path planning, though as a modular part of the schema concept. Park et al. presented a synthesis of APF obstacle avoidance with simulated annealing to provide effective resolution of the local minima trap problem (Section 2.5.1)

2.2 APF Implementations

A common starting point to APF implementations is the work by Khatib et al. [11, 48, 49], where an artificial potential field method for manipulator motion planning was presented. Khatib defined each manipulator joint using a *Point Subject to a Potential*, PSP. The obstacles (objects, walls, etc) were defined with basic geometric elements (circles, boxes, triangles). Rigid links between each PSP were also considered as obstacles. With the rigid links between the joints known, Khatib then demonstrated obstacle avoidance of a 4-Degree of Freedom (DoF) manipulator using low-level control (Figure 2.1).

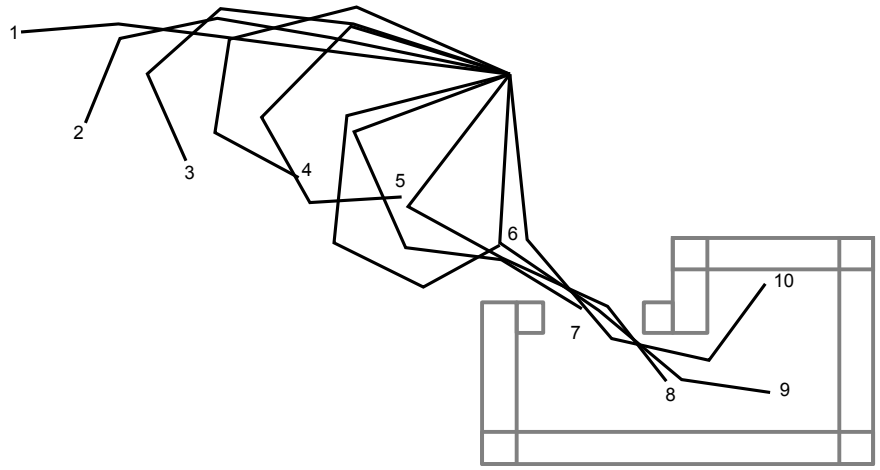


Figure 2.1: Displacement of a 4-DoF manipulator inside an enclosure. Adapted from [48]

Hogan and others further advanced APFs with the addition of various implementations of a viscous damping component [50–53]. Commonly, this damping component was rendered into a additional force that was aggregated with the attractive and repulsive forces. Hogan developed an implementation that utilised the damping term to account for motor impedances. Building upon Krogh, Tilove employed a viscous damping term that was added to the attractive velocity calculation and acted orthogonal to the goal direction (Equation 2.1):

$$\vec{v}_A = v_A^t \hat{e}_t + v_A^n \hat{e}_n \qquad [53] \qquad (2.1)$$

$$= \left[\pm \frac{(2v_t^2 \pm 4\alpha d_t)^{1/2} \mp v_t}{\alpha} \right] \hat{e}_t + \left[\pm \frac{(2v_n^2)^{1/2} \mp v_n}{\alpha} \right] \hat{e}_n \qquad [50] \qquad (2.2)$$

where $\vec{v}_A$ is the attractive control velocity vector, v_A^t is the attractive control velocity in the target direction, v_A^n is the attractive control velocity orthogonal to the target direction, d_t is the distance to the target, v_t is the current velocity in the target

direction, $\hat{e}_t$ is the unit vector in the target direction, v_n is the current velocity orthogonal to the target direction, $\hat{e}_n$ is the unit vector orthogonal to the target direction, and α is the maximum magnitude of acceleration. Subsequently, Tilove demonstrated the performance benefit of this damping component, showing a clear reduction in path oscillation.

Extending on the commonly-considered triad of forces (attractor, repulser and damping), additional forces can be further added to APF implementations [15, 29, 30, 54]. For example, Borenstein presents a force acting counter to the repulsive force that is dependant on the current repulsive bearing (Equation 2.3). The adaptation by Borenstein acts to limit damping during wall-following behaviours; while slowing the vehicle in head-on approaches, the repulsive component is in full effect. However, while when running perpendicular to the obstacle, the rover is subject to a reduced repulsive force. An example of this implementation is given by

$$\vec{F}'_R = w\vec{F}_R + (1-w)\,\vec{F}_R\,(-cos\theta) \tag{2.3}$$

where $\vec{F}_R$ is the repulsive force, $\vec{F}'_R$ is the modified repulsive force, w is a scaling constant, and θ is the repulsive force bearing.

Lynch ([15]) incorporates both a tangential and a risk force in addition to the three basic forces. The tangential force applies a small force acting perpendicular to the current vehicle velocity direction, which is biased with respect to the local obstacles. The risk force acts to mitigate the repulsive force at slower speeds. This allows vehicles to slow down as they approach obstacles and to overcome many local

minima issues. A key component in Lynch's formulation is ϵ, which is an exponential parameter (see Equation 2.4) that provides a velocity dependent scaling function.

$$\epsilon = e^{(-cv)} \qquad [15] \tag{2.4}$$

where c is a constant, and v is the vehicle forward velocity. Lynch also uses a proportional-integral-derivative (PID) controller to control the vehicle once a defined range to target threshold was achieved. Additionally Lynch uses a path history force to provide directional bias away from previously traveled areas.

This thesis builds upon components of these earlier works, namely Khatib's multiple PSP manipulators, the use of viscous damping, and modifies Lynch's risk and tangential forces, in order to propose the algorithm presented in Chapter 3.

2.3 Potential Field Generation

The generation of the scalar APF map (U) is accomplished through the calculation of the net field potential at each point within the desired area. At each point within the map ($U[x, y]$), the net potential field is calculated as a combination of the repulsive and attractive potentials. The gradient map for this net potential is the net force map (Equation 2.5) given by

$$\begin{aligned} \vec{F} &= -\nabla \mathbb{U} \\ &= -\left(\frac{\partial \mathbb{U}}{\partial x}\hat{e}_x + \frac{\partial \mathbb{U}}{\partial y}\hat{e}_y + \frac{\partial \mathbb{U}}{\partial z}\hat{e}_z \right) \end{aligned} \tag{2.5}$$

While attractive potentials are commonly defined as acting from a unique points,

repulsive potentials are generated from the surrounding environment. There four criteria that are desired for potential functions that are used to model an environment [21, 55]:

1. At distance from an obstacle, the isopotential lines should approach circular (2D) or spherical (3D) symmetry. This minimizes risks of local minima by approximating the obstacle as a featureless point.

2. Near an obstacle, the isopotential lines should follow the obstacle shape. Mapping to the obstacle features ensures that while in proximity the developed potential respects smaller elements of the obstacle.

3. The potential function should have a limited range of effect. [48]

4. The potential function and the resulting gradient must be continuous. [48, 51]

As shown in [48, 56, 57], $\vec{F}$ can be constructed as the summation of the separate potential gradients (Equations 2.6 and 2.7), in other words

$$\mathbb{U} = \mathbb{U}_R + \mathbb{U}_A \tag{2.6}$$

where U_R and U_A are repulsive and attractive potential so that the force can then be written as

$$\begin{aligned} \vec{F} &= -\left(\nabla \mathbb{U}_A + \nabla \mathbb{U}_R\right) \\ &= -\left(\vec{F}_A + \vec{F}_R\right) \end{aligned} \tag{2.7}$$

A necessary requirement for the use of this superposition of potentials is both the non-rotation of the field (Equation 2.8) and static nature of the point charges (Equation 2.9) [22, 51, 58]. These can be respresented mathematically as

$$\begin{aligned} curl\left(\vec{F}\right) &= \nabla \times \vec{F} \\ &= 0 \end{aligned} \tag{2.8}$$

$$\begin{aligned} div\left(\vec{F}\right) &= \nabla \cdot \vec{F} \\ &= 0 \end{aligned} \tag{2.9}$$

An example of this method is shown in Figures 2.2 and 2.3, where a field map ($\vec{F}$) consisting of two repulsers and one attractor point (R_i and A, respectively) is generated via the general functions defined in Equations 2.10, 2.11 and 2.12.

$$\vec{F}_A(P) = \nabla\left(\frac{1}{2}\mu\frac{1}{z_A^2}\right)\hat{e}_A = -\frac{\mu}{z_A}\hat{e}_A \tag{2.10}$$

$$\vec{F}_{R_i}(P) = \nabla\left(\frac{1}{2}kz_{R_i}^2\right)\hat{e}_{R_i} = kz_{R_i}\hat{e}_{R_i} \tag{2.11}$$

$$\vec{F}(P) = \vec{F}_A(P) + \sum_1^N \vec{F}_{R_i}(P) \tag{2.12}$$

These functions are examples adapted from [48], where P is a defined map point $[x, y]$ A is the attractor point, R_i is the i^{th} repulser point, z_{R_i} is the range from P to R_i, $\hat{e}_{R_i}$ is the unit vector from P to R_i, z_A is the range from P to A, $\hat{e}_A$ is the unit vector from P to A, $\vec{F}(P)$ is the net force vector at P, k and μ are respectively attractive and repulsive scaling constants. R_i represents the i^{th} repulser of the set of N repulsers.

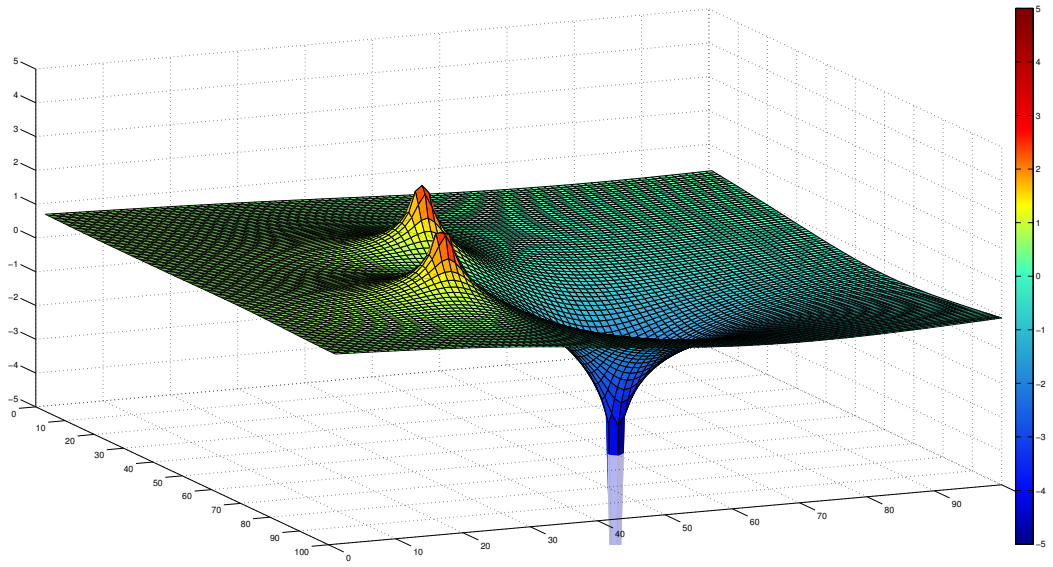


Figure 2.2: APF Surface Map based upon discrete repulser and attractor points

Here, a bitmap occupancy map is used to define the regions of repulsion as a series of discrete points, with the attractor positioned at $x = 10$, $y = 45$. With the use of many repulser points, geometrically irregular regions of exclusion can be defined. In this fashion, complex maps can be defined through the superposition of the individual maps [15]. An example of this progression is shown in Figures 2.4 through 2.7. Figure 2.4 shows the defined obstacle map as an occupancy grid. Cells which are part of obstacles are shaded in, white cells indicate free space. Figure 2.5 shows the repulsive potential map that results from the occupancy map shown in Figure 2.4. The repulsive function defined in Equation 2.13 is used to generate this map. The respulsive function is given by

$$u_R = 0.025 \cdot r^2 \tag{2.13}$$

where u_R is the repulsive potential, r is the range between calculation points, and

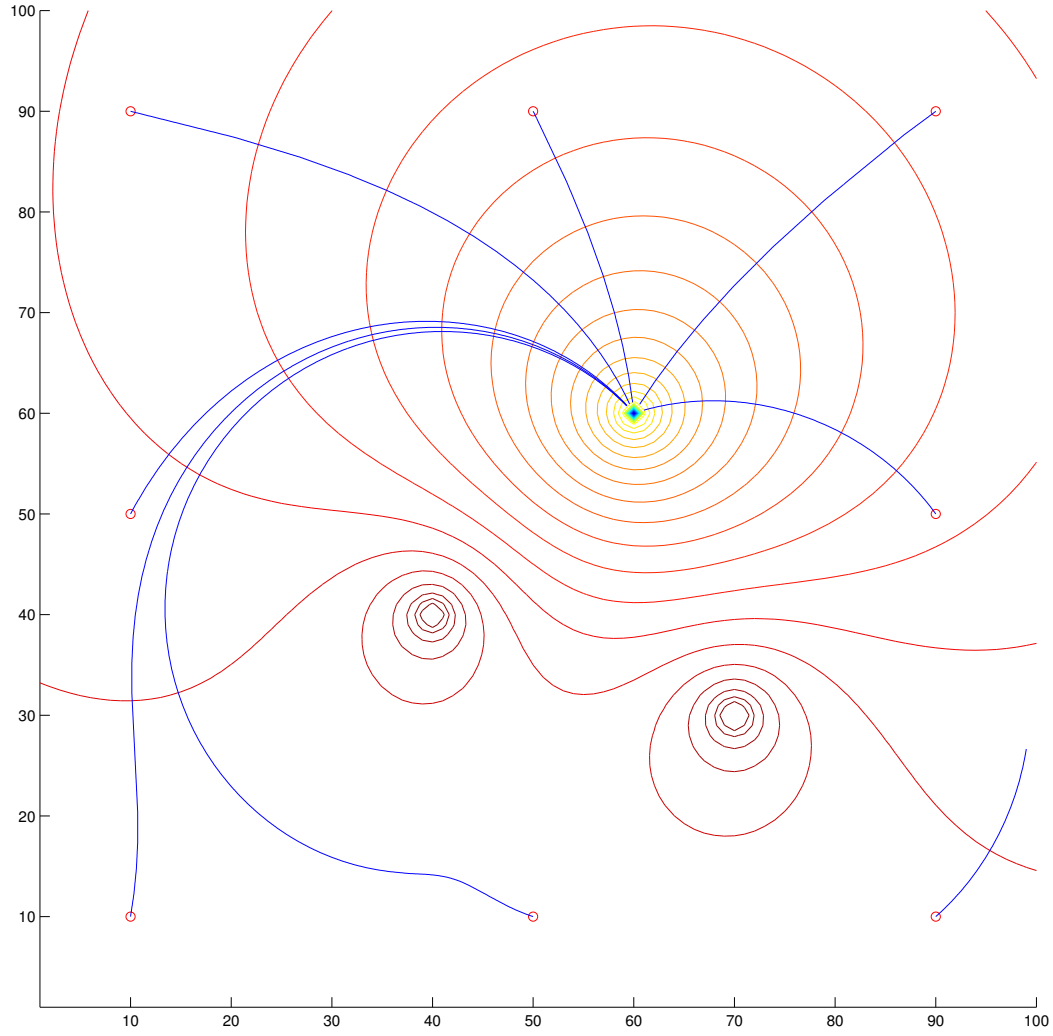


Figure 2.3: APF Contour Map with Gradient Decent Paths

0.025 is a scaling factor. Figure 2.6 shows the attractive potential map that results from a target located at $[10, 45]$ in a map with the same dimensions as that of Figure 2.4. The attraction function, defined in Equation 2.14, is used to generate this map. This attraction function is given by

$$u_A = -1.5 * (1/r)^{0.5} + C/r \tag{2.14}$$

where u_A is the attractive potential, r is the range between calculation point and

sample point, C is the constant value to ensure gradient at range, and 1.5 is a scaling factor.

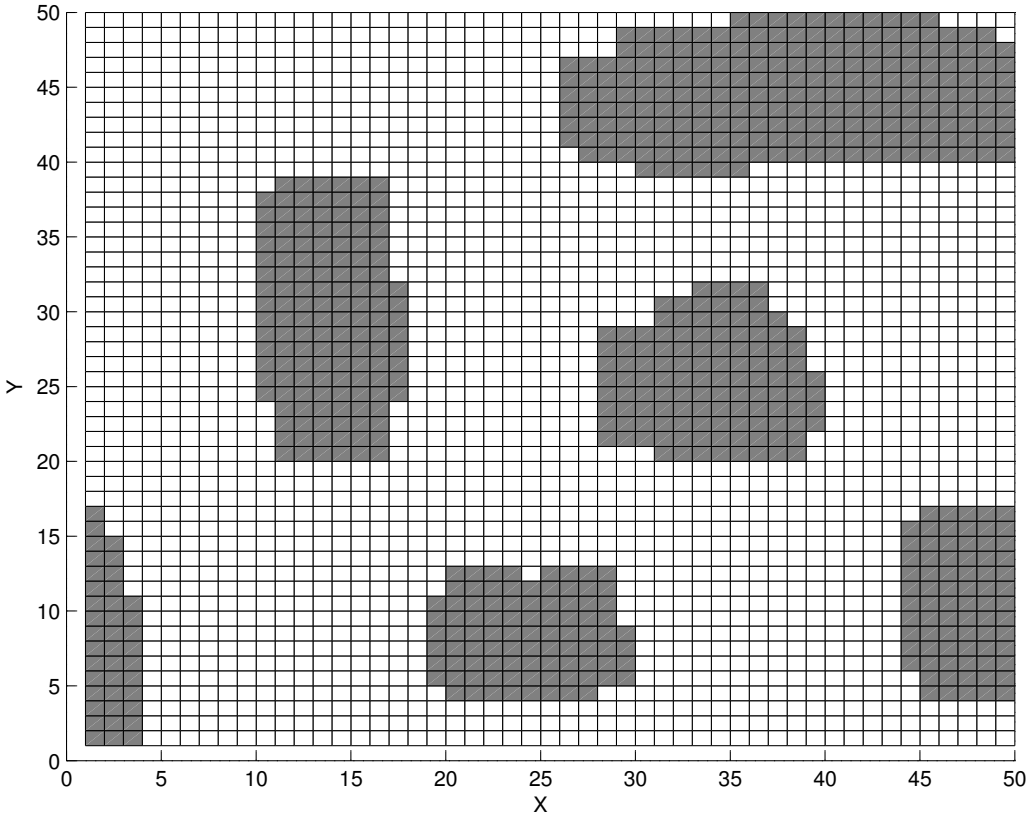


Figure 2.4: APF source obstacle bitmap. Occupied regions are indicated in grey, free regions are indicated in white.

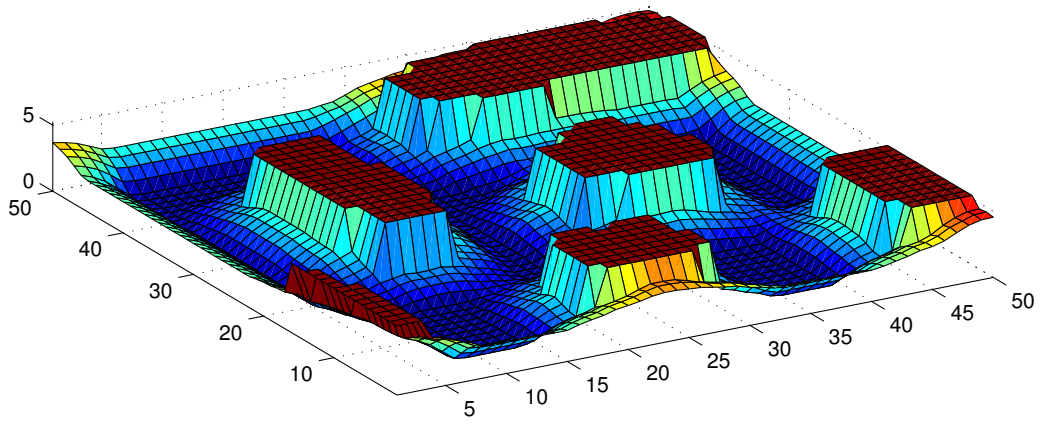


Figure 2.5: APF obstacle potential map. Power law based repulsion map generated from the the obstacle map generated in Figure 2.4

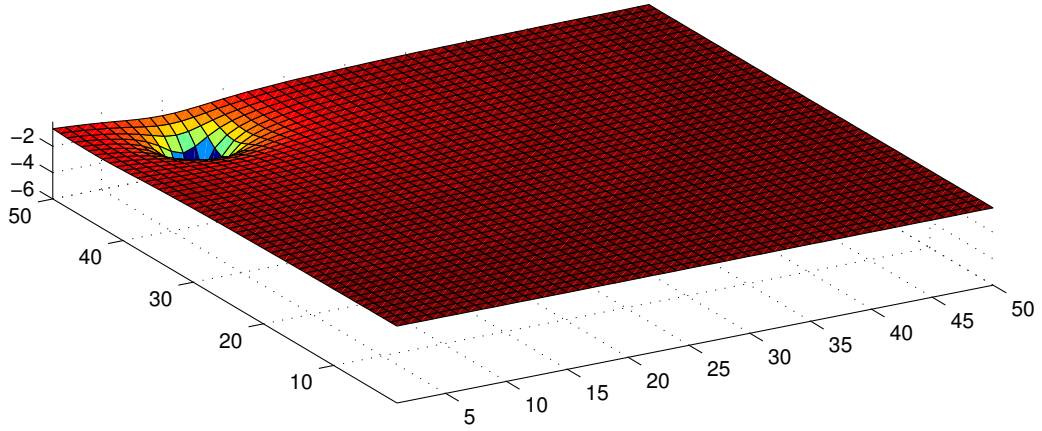


Figure 2.6: APF attractor potential map. Power law based attraction map generated with attractor point located at $x = 10$, $y = 45$

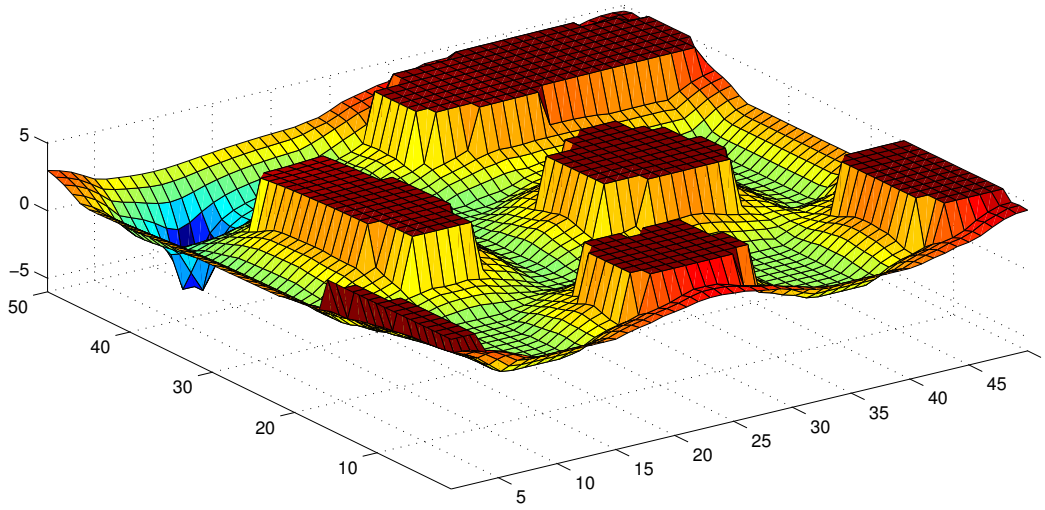


Figure 2.7: APF net potential map. The combination of the maps presented in Figures 2.5 and 2.5

Alternative definitions for potential functions and the creation thereof have been proposed by other works. A short list of those works is presented here. Khosla and Volpe proposed using a superquadratic shape function in their work in 3D, Equation 2.15, and an elliptical shape function for their work in 2D, [55, 59]. Similar work was also part of Chella's paper to model the detected environment [60]. These shape functions define the desired arrangement of the isopotential lines for a given object shape at some given range from the object.

$$\left[\left(\frac{x}{f_1(x,y,z)}\right)^{2n} + \left(\frac{y}{f_2(x,y,z)}\right)^{2n}\right]^{\frac{2m}{2n}} + \left(\frac{z}{f_3(x,y,z)}\right)^{2n} = 1 \tag{2.15}$$

where f_1, f_2 and f_3 are scaling functions, and m and n are exponential parameters. This method requires that the obstacle is a known shape primitive, limiting usage to known maps.

Haddad et al. proposed using a *rotational potential* which combines two repulsive potential functions [61]. Additionally they used an image based obstacle detection method that resulted in an probability weighted occupancy map that is not rectangular but trapezoidal. The resultant potential for each map cell was weighted by the probability, Equation 2.16. The attractive function used by Haddad et al. is show in Equation 2.17.

$$U_r = \sum_{i,j} \begin{cases} \begin{cases} \alpha^2 p_2 + p_1 & ,if\ d < d_0 \\ 0 & ,else\ wise \end{cases} & ,if\ P_{Obst}[i,j] \geq S^0 \\ \begin{cases} \alpha^2 P_{Obst}[i,j]\ p_t & ,if\ d < d_0 \\ 0 & ,else\ wise \end{cases} & ,if\ P_{Obst}[i,j] < S^0 \end{cases} \tag{2.16}$$

$$U_g = K_g \sqrt{d_g^2 + R^2} \tag{2.17}$$

$$U = U_g + U_r \tag{2.18}$$

$$p_1 = \frac{1}{2} k_1 \left(\frac{1}{d} - \frac{1}{d_0} \right)^2 \tag{2.19}$$

$$p_2 = \frac{1}{2} k_2 \left(d - d_0 \right)^2 \tag{2.20}$$

$$p_t = \frac{1}{2} k_t \left(d - d_0 \right)^2 \tag{2.21}$$

where U_g is the attractive potential function, U_r is the repulsive potential function, p_1, p_2, and p_t are supplementary potential functions, k_1, k_2, and k_t are gain constants, i and j are the map cell indices, α is the sine of the relative angle between the robot direction and the cell segment, d is the distance from the robot to $cell[i,j]$, d_g is the distance from the robot to the goal, d_0 is the distance limit beyond which a cell has no effect on the robot, $P_{Obst}[i,j]$ is the probability of $cell[i,j]$ containing an obstacle (the traversability danger), S^0 is the traversability danger limit that defines when a cell is considered traversable, K_g is the positive gain constant, R is a weighting function.

Kim and Khosla proposed the use of harmonic functions to develop a velocity potential function for use in obstacle avoidance [58]. Here the authors model the point vehicle as an element within a fluid. They then develop the velocity potential

to "flow" this point around obstacles, which have been decomposed into discrete flat panels for this flow analysis. One clear benefit of harmonic functions are the lack of local minima for scenarios where vehicles are modelled as point entities, though as the authors note, when a multiple point vehicles are used there is still possibilities of local minima forming. For these situations, they show the example of redundant manipulators and a two point vehicle, the authors introduced an additional control strategy. Also, similar to the work in this thesis, the harmonic potential velocity functions they propose (Equations 2.22 and 2.23) are only applicable to the space local to the vehicle. The proposed equations combine the velocities developed from uniform flow, flow towards the goal, and flow due to the repulsive obstacles.

$$-\phi_x = U\cos\alpha - \frac{\lambda_g}{2\pi}\frac{\partial}{\partial x}\log R_{ig} - \sum_{j=1}^{m}\frac{\lambda_j}{2\pi}\int_j R_{ig}dl_j \tag{2.22}$$

$$-\phi_y = U\cos\alpha - \frac{\lambda_g}{2\pi}\frac{\partial}{\partial y}\log R_{ig} - \sum_{j=1}^{m}\frac{\lambda_j}{2\pi}\int_j R_{ig}dl_j \tag{2.23}$$

where ϕ_x and ϕ_y are the velocity components induced from the obstacle panels, U is a coefficient defining the uniform flow (free-stream) velocity, α is the angle between the direction of flow and the x-axis, R_{ig} is the distance from the current sampling point to the goal, λ_j is the *strength* of the j^{th} panel, and dl_j is the j^{th} small panel element.

2.4 APF Reactive Navigation

In contrast to the static nature of constructing vehicle paths based upon APF algorithms, APF systems are also used for real-time obstacle avoidance [2, 29, 30, 47, 48, 53]. In this case, the vehicle (or manipulator in the case of [48]) reacts to obstacles detected within a defined range. As compared to the global path-finding problem using APFs, this limits the degrees of freedom of the problem and provides a lower computational cost option [62]. This reduced computational complexity is better suited for real-time vehicle operations [58].

With the real world realities imposed by various sensing technologies, various APF reactive navigation algorithms have been proposed. Borenstein and Koren have proposed both the Virtual Force Field (VFF) method and the follow-up Vector Field Histogram (VFH) method [1, 2]. In the VFF, the authors presented a mechanism that constructed an occupancy probability (histogram) map and weighting resulting repulsive potentials of each map cell by this probability. Given a noisy sensed environment, this provides an effective approach to reactive navigation. The VFF layout is shown if Figure 2.8.

With VFH, Borenstein and Koren modify the VFF concept to use a polar histogram map instead of the grid model used in VFF (Figure 2.9). Their motivation was to improve the performance and reliability of the VFF solution. In this method the region surrounding the robot is discretized into a polar histogram map, which then

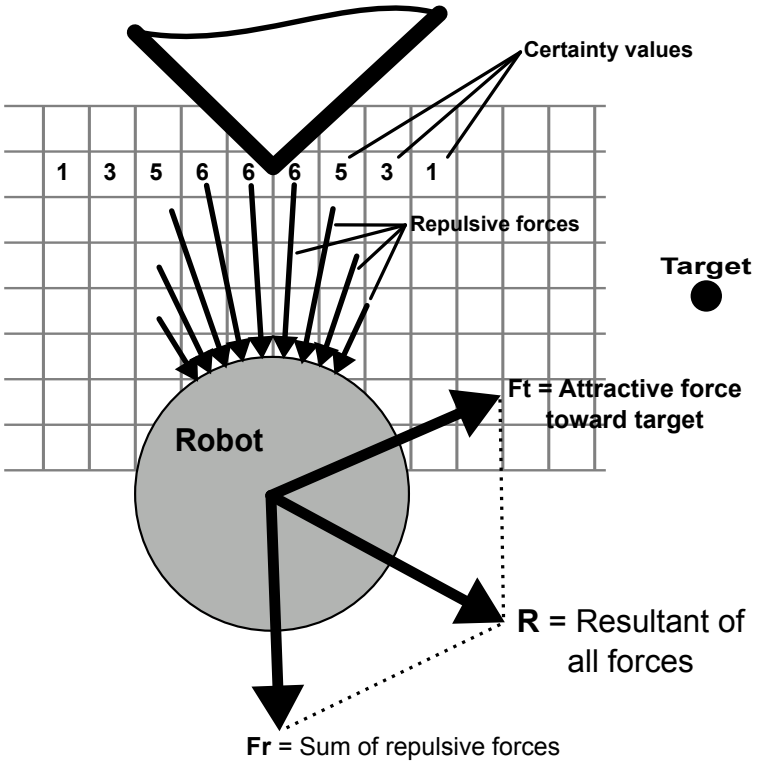


Figure 2.8: Virtual Force Field (VFF) concept, adapted from [1]

provides a weighted per-sector repulsive force that is normalized against detected obstacle density. Along with the repulsive force, the polar histogram also provides a reliable means of detecting clear travel directions, something that was not directly provided by VFF, and one of the goals of VFH.

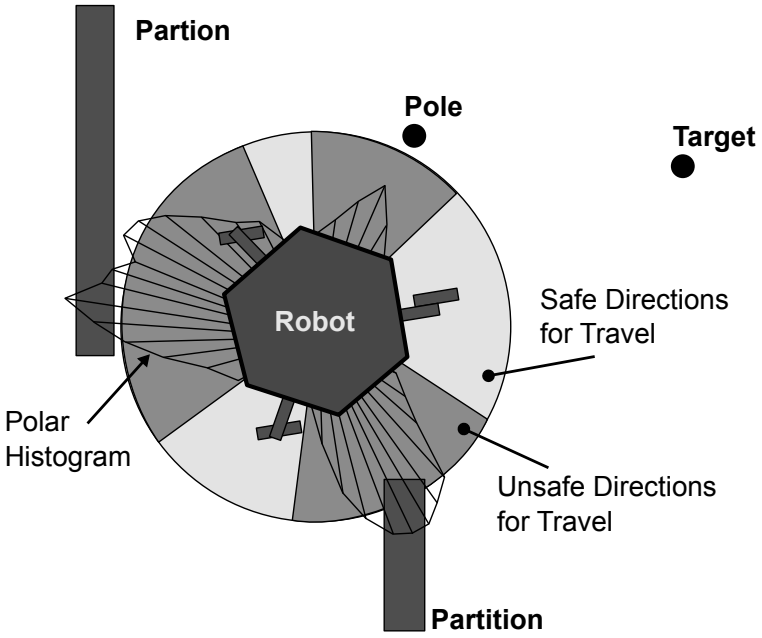


Figure 2.9: Vector Field Histogram (VFH) concept, adapted from [2]

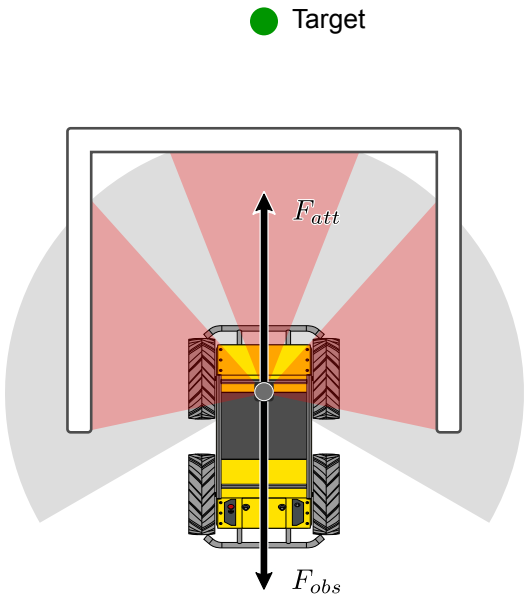
2.5 Known Concerns With APF

Several recurring issues with APF implementations have been well documented in the literature [19, 24–26], and are discussed below. The discussion by Koren in [24] provides a particularly in-depth review of the local minima trap (Section 2.5.1), narrow passages – referred to as the "Thread-the-Needle" problem here – (Section 2.5.2), undamped disturbances and undamped motion (Section 2.5.3). Later work by Ge and Cui in [29] (and also discussed in [63]) presents a fifth local minima state, referred to as *Goals Non-Reachable with Obstacles Nearby* (GNRON) and is presented in Section 2.5.4.

2.5.1 Local Minima Traps

The most prevalent problem inherent with an APF-based system is the susceptibility to local minima. A local minimum is a point within the APF where all exiting field gradients are positive. These local minima trap the vehicle to this point and prevent achievement of the goal. In a pure system of point repulsers, a point attractor local minimum may occur in a *trap situation*, such as in the scenario presented in Figure 2.10 and discussed in [24, 53].

Tilove discusses the generation of local minima points through the interaction of the repulsive fields generated by the map obstacles. Also discussed is the method of reducing the occurrences of local minimas through the use of an obstacle threshold distance beyond which a known obstacle is excluded from the algorithm. This



$$F_{net} = F_{att} - F_{obs} = 0$$

Figure 2.10: APF Concerns – Local Minima Trap

thresholding approach is use in RAVE, where any obstacle point beyond this threshold is ignored. Koren and Borenstein discuss using either a global path planner or a heuristic approach to extricate the vehicle from a local minima situation. Similarly, the use of a global path planner to avoid local minima conditions while using the RAVE method as the local planner is preferred, though this is beyond the scope of this thesis.

2.5.2 Narrow Openings – "Thread-the-Needle" problem

Passing through narrow passages, such as building door frames or boulders in a rock field, can present a local minima even though there is sufficient room for the vehicle

to pass, a problem known to some as "Thread-the-Needle" problem [9, 24]. As shown in Figure 2.11, the developed obstacle forces ($\vec{F}_{obs,i}$) are mirrored about the vehicle drive direction, resulting in a net obstacle force ($\vec{F}_{obs}$) that is in direct opposition to the target attractive force ($\vec{F}_{att}$), potentially resulting in zero net force ($\vec{F}_{net}$). The development of this situation is discussed by Koren and Borenstein in [24], with the observation that the occurrence of this situation is dependant upon the relative attractive and obstacle force magnitudes. With this understanding, occurrences of the "Thread-the-Needle" problem through the use of the *Risk Force* was sought in this thesis. This is described in detail in 3.2.6 and with the multi-point geometry presented in 3.2.

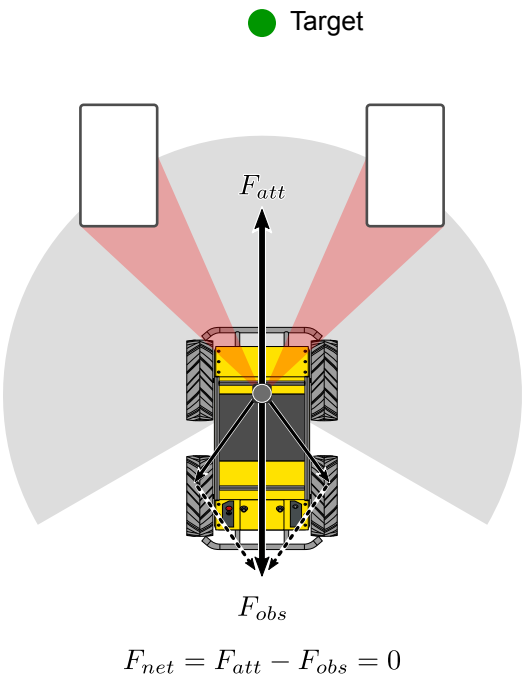


Figure 2.11: APF Concerns – "Thread-the-Needle" problem.

2.5.3 Undamped Responses

The proportional nature of the developed reactive forces can present situations of command oscillations. The two generalized cases presented in [24, 25] are the *undamped motion* and *undamped disturbances*.

When passing through narrow corridors, APF systems that have insufficient damping or update the system too slowly may become trapped in a state of oscillation due to *undamped motion* [24]. In this situation, the resultant forces push the vehicle away from an obstacle and subsequently overshoot the equilibrium path for the corridor. Reacting to now being too close to the opposing obstacle, the vehicle is pushed back across the equilibrium path. This situation repeats until either the local environment changes and the vehicle exits the corridor, or the oscillations increase until the vehicle impacts with the corridor wall. Figure 2.12 shows an example of this problem.

In [24], Koren and Borenstein present their experiments with the CARMEL platform which show this oscillatory behaviour. In their experiments, their vehicle clearly transitions between stable and unstable behaviour as it transitions between a "wide" hallway to a "narrow" hallway. In their analysis, they showed the link between the vehicle geometry (width), hallway width and vehicle speed.

When interacting with a single disturbance, APF systems that have insufficient damping or update the system too slowly may become trapped in a state of oscillation due to this *undamped disturbance*. In this situation, the resultant forces push the vehicle away from an obstacle and subsequently overshoot the equilibrium path to

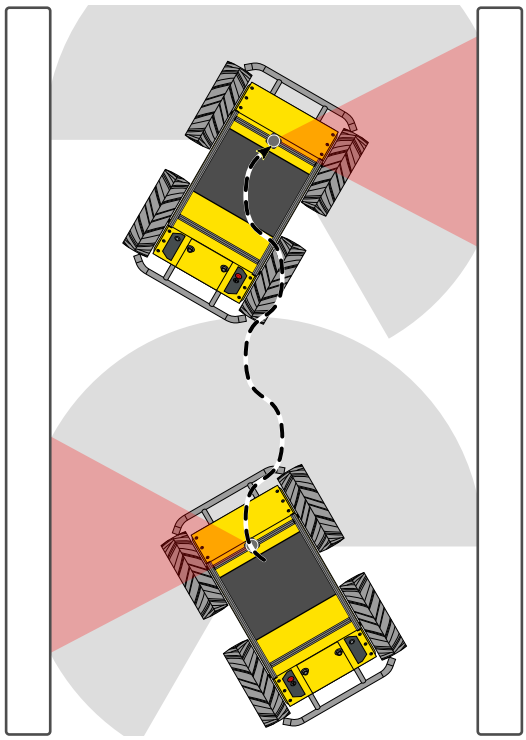


Figure 2.12: APF Concerns – Undamped Motion. Potentially occurring in narrow corridors, the vehicle oscillates about the corridor center, 'bouncing' between the walls.

the target. Reacting to the target overshoot a the vehicle is pulled back across the equilibrium path, repeating the cause of the issue. Figure 2.13 shows an example of this issue.

In [24], Koren and Borenstein present their experiments with the CARMEL platform, and clearly demonstrate this oscillatory behaviour. They also show that the stability condition for this behaviour is dependant upon vehicle size, speed and relative angle between the disturbance and the vehicle motion vector.

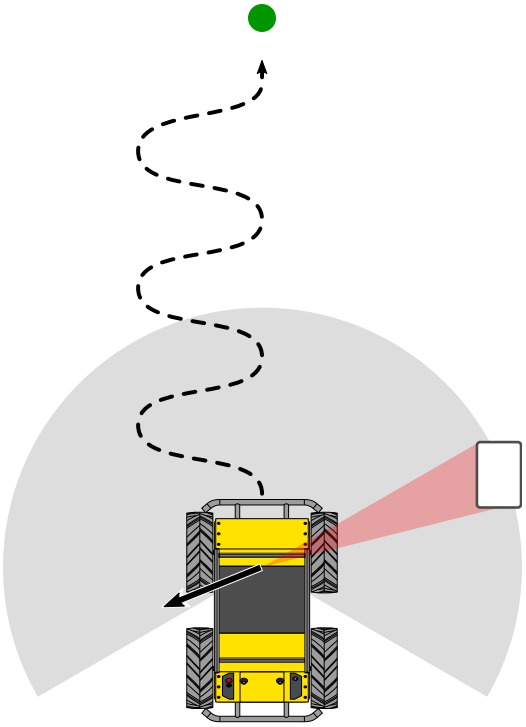
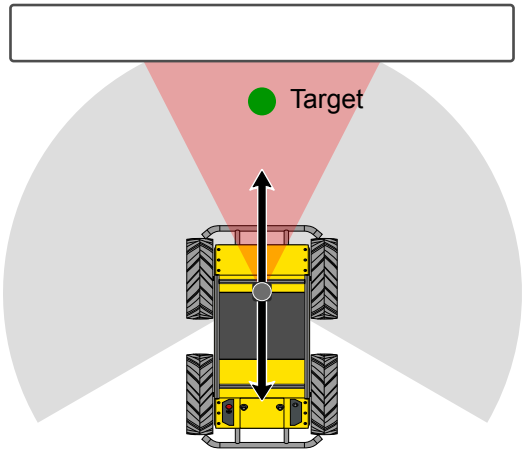


Figure 2.13: APF Concerns – Undamped Disturbance. Potentially occurring when passing lone obstacles, the vehicle oscillates about the target vector

2.5.4 Target Proximity to Obstacle

As highlighted in [29], it is common to present targets for APF systems that are placed at known safe positions within the environment. That is, targets are designated in positions which are a sufficient distance from obstacles to avoid, generating what Ge and Cui call *Goals NonReachable with Obstacles Nearby* (GNRON) conditions. This is shown in Fig 2.14.

In this situation, the attractive force of the target is negated by the obstacle force, while the target is between the obstacle and the rover. Ge and Cui proposed to resolve this issue through a scaling of obstacle repulsive forces that is inversely proportional



$$F_{net} = F_{att} - F_{obs} = 0$$

Figure 2.14: APF Concerns – Goals NonReachable with Obstacles Nearby (GNRON). Target proximity to obstacles shifts target minimum away from the obstacle.

to the distance from the vehicle to the target as well as the obstacle, a concept that was also used by Yin and Yin in [54]. This is similar to the *Risk Force* developed in Section 3.2.6, with the major difference that the *Risk Force* is applied generally and helps to resolve other issues (such as the "Thread-the-Needle" issue, as discussed in Section 2.5.2, whereas the proposal by Ge and Cui limits the effectiveness only to the GNRON situation.

Chapter 3

Development of the new RAVE algorithm

3.1 Introduction

In the work completed for this thesis, a system was developed to expand upon the past work presented in Section 2.1.1. Specifically, this work combines the use of the multiple point model proposed by Khatib ([11, 48]), a viscous damping component ([50–53]), the dynamic adaptation of the repulsive force ([15, 29, 30, 54]), and the inclusion of the risk and tangent forces from Lynch ([15]).

Titled *Re-Active Vector Equilibrium* (RAVE), the proposed method is a novel implementation of the following four main components:

- The vehicle as a collection of point charges
- The use of a linear and radial viscous damping force
- The use of a speed proportional tangential force
- The addition of a speed-proportional risk force

The elements of each of these points are explained in the following sections.

In a short overall summary, the proposed implementation of RAVE uses range data from a single laser scanner to obtain a scene of the current environment. This range data is compared at each perimeter point, and then transformed into sets of virtual forces and torques that act on the rover, which is considered a free Newtonian body. From these forces, the calculated reaction accelerations are used to develop rover drive wheel commands. The proposed Re-Active Vector Equilibrium (RAVE) method was developed to overcome several of the limitations discussed in Section 2.5 that are associated with traditional artificial potential field techniques. Specifically, the narrow openings (Section 2.5.2), undamped disturbances and undamped motion (Section 2.5.3), and GNRON (Section 2.5.4) limitations are addressed by RAVE. The general case of local minima traps (Section 2.5.1) is not addressed by RAVE.

3.2 Description of RAVE Model

The RAVE geometric model is shown in Figure 3.1, where the detected obstacle points (red circles) are used to calculate the net force ($\vec{F}_{net}$) at each perimeter point (small ◔). A collection of 10 perimeter points is used in this example, distributed as shown. The number of perimeter points and the associated vehicle safety buffer distance (grey circles) are chosen to provide complete coverage of the vehicle, although this does not imply equal coverage of the vehicle. The number and location of perimeter points is not fixed and any number and configuration may be, this is vehicle dependant.

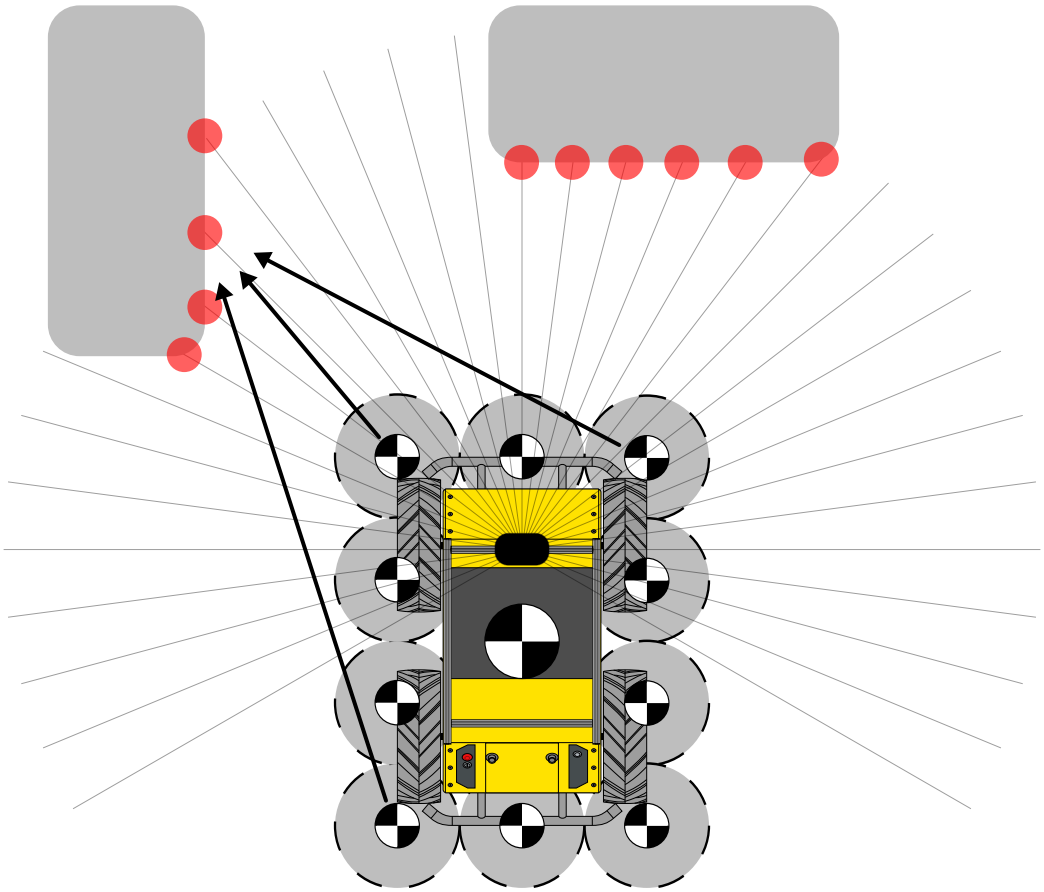


Figure 3.1: The geometric model of the RAVE method.

3.2.1 Perimeter Points

RAVE models the vehicle as a collection of point charges arranged about the vehicle perimeter (see Figure 3.2). This includes the vehicle geometry in the calculation reacted motion, without demanding additional vehicle information or requiring calculation of the per-angle buffer distance for each scan cycle. The per-angle buffer is a vehicle specific calculation that is needed to envelope a vehicle with an even buffer on all faces. Alternatively to the per-angle buffer, a central circular buffer may also be used, as shown in Figure 3.2. As each perimeter point is at a known vector from the

vehicle Center of Gravity (CofG, large ◓ in figures), reactive torques about the vehicle CofG can be directly calculated from the obstacle force calculated at each perimeter point. Additionally, the use of the perimeter points (small ◓ in figures) allows for vehicles of non-standard planar profiles to be readily defined without limiting vehicle mobility due to an oversized enveloping perimeter (see Figure 3.2). For multi-body or articulated vehicles, this multi-point arrangement can also be used to provide a simple means to adapt the net vehicle reaction without reformulation.

Throughout this thesis, 10 perimeter points are used as that number worked well with the modelled vehicle (Clearpath Husky A200 [64]), but this is not a requirement of RAVE. As a starting point, each vehicle side should have at least $m = (\text{Side Length}/(2 \times \text{Buffer Distance})) + 1$ perimeter points. This number of perimeter points will ensure that there is total vehicle coverage, though some areas may not have complete coverage and more points may be added at the discretion of the operator. Ideally the perimeter points are evenly spaced along each vehicle side, though again the operator should use their discretion if the vehicle is not of regular shape.

This thesis does not present work on dynamic perimeter points, that is, perimeter points which have relative positions that update at each calculation point, but the following discussion is provided.

For vehicles of a fixed nature, such as Husky used here, the perimeter points are static in nature. However this is not a requirement of RAVE. As proposed by Khatib in this work on manipulators, an operator can choose perimeter points that represent

moving points. So long as the relative position from the perimeter point to the vehicle CofG is known at the time of calculation, any set of perimeter points may be used. As an example a vehicle such as a tank may be represented as a set of static perimeter points for the chassis, and another set of points for the gun barrel which have relative positions which are updated at each calculation point.

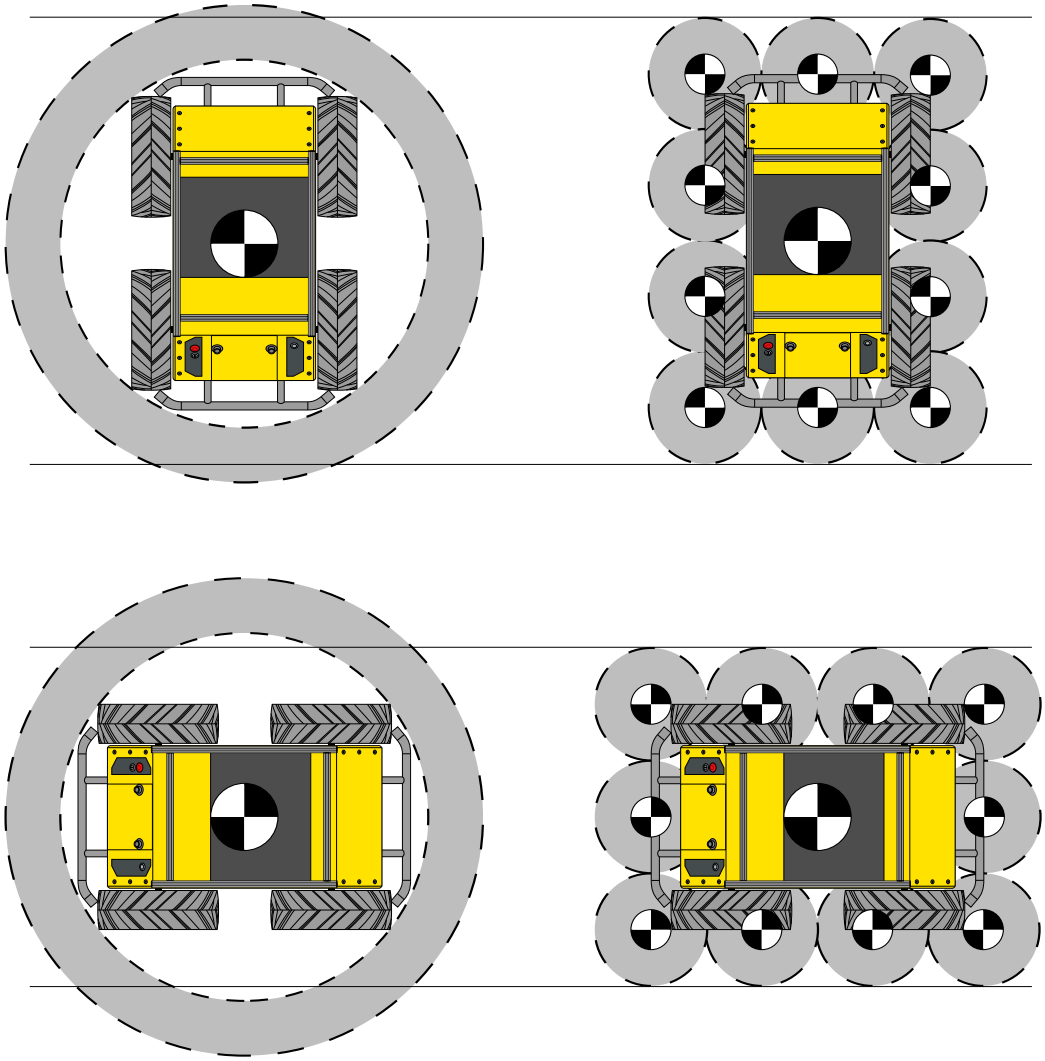


Figure 3.2: A geometric model of the RAVE method as compared to a single point charge with equal safety buffer radii (grey regions).

3.2.2 Description of RAVE Forces

A detailed description of the forces developed is presented in Sections 3.2.3 through 3.2.8.

3.2.3 Obstacle Force

The obstacle force ($\vec{F}_{obs}$, equation 3.1, Figure 3.3) presents the net repulsion force acting on the vehicle CofG and can be written as

$$\vec{F}_{obs} = \sum_{i=1}^{M} \sum_{j=1}^{N} -K_{obs}\ s\ \hat{e}_r\ {r_{i,j}}^{1-q} \tag{3.1}$$

where K_{obs} is a proportional gain value, s is the arc length of a sector for the defined scan resolution (z), q is the power law exponent used, M is the number of perimeter points, N is the number of obstacles, and $r_{i,j}$ represents the range of the obstacle point, j, to the perimeter point, i, $\hat{e}_r$ is the unit vector along $r_{i,j}$. The obstacle force

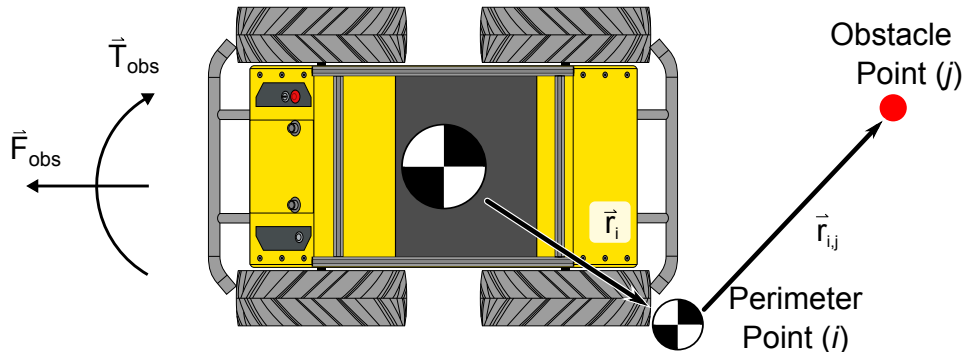


Figure 3.3: Re-Active Vector Equilibrium – Obstacle Force

is calculated in two steps, first by computing the force at each of the M perimeter point (small ◔) of the vehicle for each of the detected N obstacles ($\vec{F}_{obs,j}$), then as a

summation of these forces which acts through the vehicle centre. This formulation is based upon power law rules, discretized for the scan angular resolution. The inclusion of the arc length at range is to account for the varying linear point density. For a laser scanner with an angular resolution of 0.25°, obstacles that have an angular width of 3° at $2\,m$ will register with at most 10 points, while at $1\,m$ the same obstacle will have an angular width of 22.5° and register with at most 90 points. This causes a non-linear increase in the number of obstacle points applied to each obstacle magnifying the effective repulsive force. To remove this magnification effect each obstacle point calculation is scaled by the length of the laser resolution arc at the measured distance. The resultant obstacle torque ($\vec{T}_{obs}$, Equation 3.2) generated by the obstacles is calculated through computing the summation of the torques due to each M perimeter point force ($\vec{T}_{obs,i}$) acting about the vehicle CofG and is given by

$$\vec{T}_{obs} = \sum_{i=1}^{M} \vec{r}_i \times \vec{F}_{obs,i} \tag{3.2}$$

where $\vec{r}_i$ is the vector from the vehicle CofG to the i-th perimeter point, and r_i represents its magnitude.

3.2.4 Attractor Force

The attractor force ($\vec{F}_{att}$, Equation 3.3, Figure 3.4) represents the pull generated by the target point and acts through the vehicle's CofG.

The attractor force is linearly proportional to the distance from the target to the vehicle centre of gravity. Additionally, to overcome real world mechanical limitations

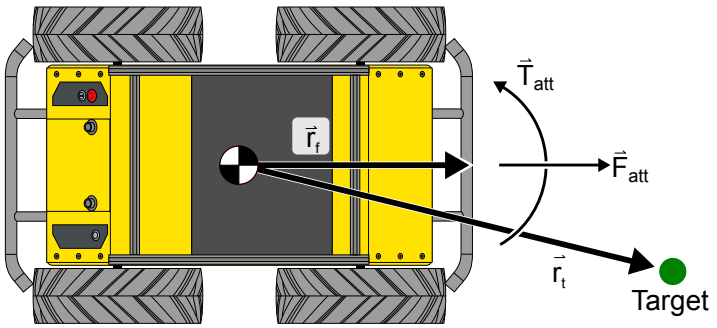


Figure 3.4: Re-Active Vector Equilibrium – Attractor Force

of stiction, friction and other low speed losses, there is a constant minimum force that is maintained to ensure motion towards the target, $\vec{F}_{att,min}$. A maximum force, $\vec{F}_{att,max}$, is also defined to prevent the attractive force from overpowering the system. $\vec{F}_{att,min}$ and $\vec{F}_{att,max}$ are set based upon the minimum and maximum desired vehicle acceleration and the vehicle mass, $F = ma$.

$$\vec{F}_{att} = K_{att} \cdot \vec{r}_t \tag{3.3}$$

where K_{att} is the attractor force proportionality constant and $\vec{r}_t$ is the vector from the target to the vehicle CofG.

The attractor torque ($\vec{T}_{att}$, Equation 3.4) is generated by the attractor force acting about the CofG. This is achieved by artificially applying $\vec{F}_{att}$ at the midpoint of the vehicle leading edge, similar to how a wagon is drawn by its handle. $\vec{T}_{att}$ is given by

$$\vec{T}_{att} = \vec{r}_f \times \vec{F}_{att} \tag{3.4}$$

where $\vec{r}_f$ is the vector between the CofG and the leading edge of the vehicle.

3.2.5 Drag Force

The drag force ($\vec{F}_{drg}$, equation 3.5, Figure 3.5) is in a direction opposite to the current direction of motion and is also linearly proportional (with proportionality factor $K_{drg,lin}$) to the vehicle forward velocity ($\vec{v}_x$). This force provides damping to the vehicle model, acting to smooth operations and limit abrupt command changes and can be written as

$$\vec{F}_{drg} = -K_{drg,lin}\ \vec{v}_x \tag{3.5}$$

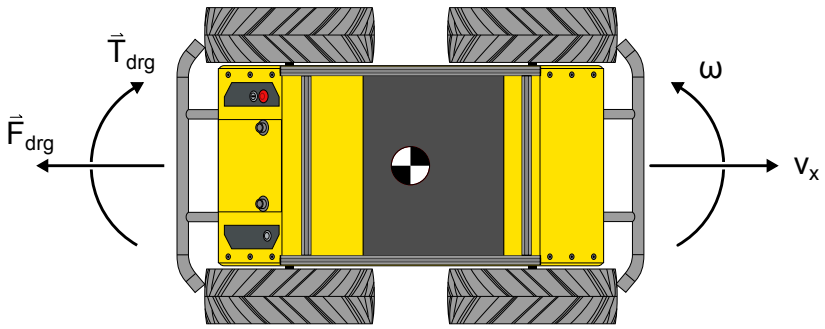


Figure 3.5: Re-Active Vector Equilibrium – Drag Force

The resulting drag torque ($\vec{T}_{drg}$, Equation 3.6) is similarly linearly proportional ($K_{drg,ang}$) to the current vehicle angular velocity ($\vec{\omega}_z$) and is given by

$$\vec{T}_{drg} = -K_{drg,ang}\ \vec{\omega}_z \tag{3.6}$$

3.2.6 Risk Force

The risk force ($\vec{F}_{rsk}$, equation 3.9, Figure 3.6) is in the opposite direction and is linearly proportional to the current obstacle force, $\vec{F}_{obs}$.

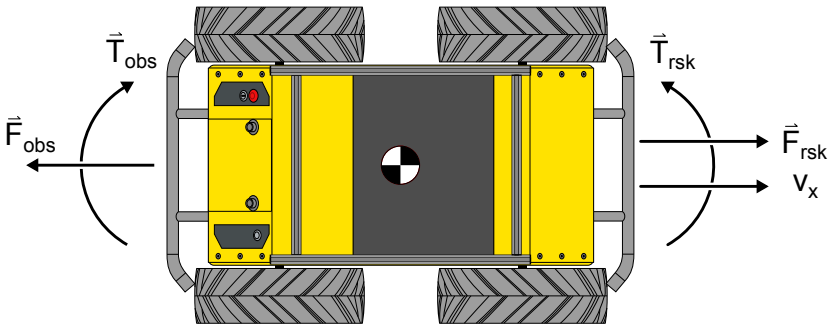


Figure 3.6: Re-Active Vector Equilibrium – Risk Force

The risk force provides for increased opposition of obstacles at high velocities while allowing for a "creeping up" effect at low velocities. The risk force provides a substantial effect into the RAVE algorithm, particularly with local minima avoidance. This is most apparent in "Thread-the-Needle" (see 2.5.2) and GNRON situations (see 2.5.4). In these situations, the risk force allows the vehicle to offset the balanced obstacle forces and slowly progress towards the target. The risk force is proportional to the exponential velocity factor, ϵ, which is used to provide a smooth transition from $\epsilon = 0 \rightarrow 1$ across the vehicle's velocity range ($0 \leq v_x \leq v_{max}$). The risk torque ($\vec{T}_{rsk}$, equation 3.10) is similarly in the opposite direction and linearly proportional to the obstacle torque via ϵ. The above comments can be written mathematically as

$$c_{decay} = 10 \cdot e^{-0.1 \cdot v_{max}} \tag{3.7}$$

$$\epsilon = e^{(-c_{decay} \cdot v_x)} \tag{3.8}$$

$$\vec{F}_{rsk} = -\epsilon \; K_{rsk} \; \vec{F}_{obs} \tag{3.9}$$

$$\vec{T}_{rsk} = -\epsilon \; K_{rsk} \; \vec{T}_{obs} \tag{3.10}$$

where c_{decay} is the velocity decay coefficient, K_{rsk} is the proportionality factor and v_{max} is the maximum forward speed defined for the vehicle. This ϵ formulation extends that presented by Lynch in [15] in the use of vehicle parameters to define c_{decay}, rather than a tunable parameter.

3.2.7 Tangential Force

The tangential force ($\vec{F}_{tan}$, equation 3.14, Figure 3.7) is generated perpendicular to the obstacle force, $\vec{F}_{obs}$, and is proportional to the exponential velocity factor, ϵ.

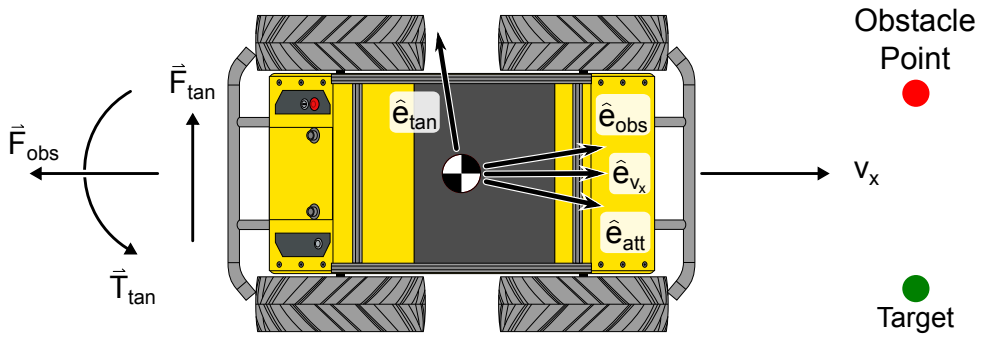


Figure 3.7: Re-Active Vector Equilibrium – Tangential Force

The tangential force is important to the avoidance of local minima conditions where $\vec{F}_{obs}$ is in opposition to the attraction force and an obstacle lies along the desired travel path, similar to the situation shown in Figure 3.7. In this situation, the tangential force will provide a bias force to break out of the local minima. As $\vec{F}_{tan}$ is proportional to ϵ, the force only appears when the vehicle has slowed to a safe speed upon detection of an obstacle, ensuring safe operation with no sudden turns. Force direction is determined by the heading direction parameter, η, and the bias direction

parameter, λ, which are given by

$$\eta = -\hat{e}_{v_x} \cdot \hat{e}_{obs} \tag{3.11}$$

$$\lambda = SGN(v_x)\,(\hat{e}_{v_x} \cdot \hat{e}_{tan})\,(\hat{e}_{v_x} \cdot \hat{e}_{att}) \tag{3.12}$$

where $SGN(v_x)$ indicates the sign of v_x, $\hat{e}_{obs}$ indicates the unit vector of $\vec{F}_{obs}$, $\hat{e}_{tan}$ indicates the unit vector of $\vec{F}_{tan}$, $\hat{e}_{att}$ indicates the unit vector of $\vec{F}_{att}$, $\hat{e}_{v_x}$ indicates the unit vector of $\vec{v_x}$.

The final direction bias sign, γ, is a random choice of either 1 or -1, if the forward velocity, v_x, is less than the tangential velocity limit, v_{tan}, and η is less than the heading direction limit, η_{lim}. Otherwise, γ is set to the sign of λ. Thus, γ is given by

$$\gamma = \begin{cases} RAND[1,-1] & : \begin{cases} v_x < v_{tan} \\ \eta <= \eta_{lim} \end{cases} \\ SGN(\lambda) \end{cases} \tag{3.13}$$

where $RAND[1,-1]$ indicates the random selection of either 1 or -1. The tangential force is given by

$$\vec{F}_{tan} = \epsilon\ \gamma\ K_{tan}\ R\,(\pi/2)\ \ \vec{F}_{obs} \tag{3.14}$$

where K_{tan} is the tangential force gain and $R\,(\pi/2)$ is a 90° rotation matrix.

$$R\,(\pi/2) = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \tag{3.15}$$

The resulting tangential torque ($\vec{T}_{tan}$, Equation 3.16) is generated in a fashion similar to $\vec{T}_{att}$. This is achieved by artificially applying $\vec{F}_{tan}$ at the midpoint of the

vehicle leading edge, similar to how a wagon is drawn by its handle. $\vec{T}_{tan}$ is given by

$$\vec{T}_{tan} = \vec{r}_f \times \vec{F}_{tan} \tag{3.16}$$

where $\vec{r}_f$ is the vector between the CofG and the leading edge of the vehicle.

3.2.8 Net Force

The net force ($\vec{F}_{net}$) and net torque ($\vec{T}_{net}$)acting on the vehicle are the sum of all of the previously described forces and torques. $\vec{F}_{net}$ and $\vec{T}_{net}$ are applied through the vehicle CofG and are given by

$$\vec{F}_{net} = \vec{F}_{obs} + \vec{F}_{att} + \vec{F}_{drg} + \vec{F}_{rsk} + \vec{F}_{tan} \tag{3.17}$$

$$\vec{T}_{net} = \vec{T}_{obs} + \vec{T}_{att} + \vec{T}_{drg} + \vec{T}_{rsk} + \vec{T}_{tan} \tag{3.18}$$

3.3 Control Loop

With $\vec{F}_{net}$ and $\vec{T}_{net}$ encompassing all of the external active ($\vec{F}_{att}, \vec{F}_{obs}, \vec{F}_{rsk}, \vec{F}_{tan}$), resistive forces ($\vec{F}_{drg}$), along with the associated torques, the vehicle can be considered as a free body on a frictionless plane. Thus the vehicle reactive acceleration can be found from equation 3.19, where m is the vehicle mass and I is the moment of inertia for a uniform plate with the same mass and external dimensions as the vehicle footprint.

$$\dot{\mathbf{x}} = \left\{ \begin{array}{c} \dot{v} \\ \\ \dot{\omega}_z \end{array} \right\} = \left[\begin{array}{cc} \vec{F}_{net,x}/m & 0 \\ \\ 0 & \vec{T}_{net,z}/I \end{array} \right] \tag{3.19}$$

The final equation is calculated from

$$\mathbf{x} = \begin{Bmatrix} v \\ \\ \omega_z \end{Bmatrix} + \begin{Bmatrix} \dot{v} \\ \\ \dot{\omega}_z \end{Bmatrix} \Delta t \tag{3.20}$$

where Δt is the timestep for the particular scan iteration.

It should be noted that physical limits of the vehicle (maximum wheel speeds, motor torque limits, etc) are not captured here. Likewise, acceleration limits are also not captured here. These are vehicle specific limits and need to be addressed prior to issuing wheel motor commands.

3.4 RAVE Algorithm

Algorithm 1 details out the high level procedure for the RAVE method. As can be seen, the general algorithm has a complexity which is driven by the calculation of $\vec{F}_{obs}$, with a complexity of $\mathcal{O}(n^2)$. However, the value of n^2 has an upper limit which is defined as $n^2_{max} = n_C n_S$, where n_S is the maximum number of laser scan points and n_C is the number of defined vehicle perimeter points. The remaining force and control law calculations run in constant time. Thus, for any implementation the algorithm has a fixed maximum execution time.

N = ImportedLaserScanObstacles()
M = PerimeterPoints()
// Calculate obstacle forces
$\mathbf{x}_k = \left\{ \begin{array}{cc} \vec{v}_X & \omega_z \end{array} \right\}^T$
for *i in N* **do**
  for *j in M* **do**
    $\vec{F}_{obs}(C_i) \mathrel{+}= -K_{obs}\ s\ \vec{r}_{i,j}{}^{1-q}$
  end
  $\vec{F}_{obs} \mathrel{+}= \frac{1}{M}\ \left(\vec{F}_{obs,C_i}\right)$
  $\vec{T}_{obs} \mathrel{+}= \frac{1}{M}\ \left(\vec{r}_{C_i} \times \vec{F}_{obs,C_i}\right)$
end
// Calculate attractor force
$\vec{F}_{att} = \vec{F}_{att} = \vec{F}_{att,min} \leq K_{att} \cdot d \leq \vec{F}_{att,max}$
$\vec{T}_{att} = \vec{r} \times \vec{F}_{att}$

// Calculate drag force
$\vec{F}_{drg} = K_{drg,lin}\ v_x\ \hat{\vec{e}}_X$
$\vec{T}_{drg} = K_{drg,ang}\ \omega_z\ \hat{\vec{e}}_Z$

// Calculate risk force
$\vec{F}_{rsk} = -\epsilon\ K_{rsk}\ \vec{F}_{obs}$
$\vec{T}_{rsk} = -\epsilon\ K_{rsk}\ \vec{T}_{obs}$

// Calculate tangential force
$\vec{F}_{tan} = \epsilon\ \gamma\ K_{tan}\ R\,(\pi/2)\ \ \vec{F}_{obs}$
$\vec{T}_{tan} = \vec{r}_f \times \vec{F}_{tan}$

// Calculate net vehicle force
$\vec{F}_{net} = \vec{F}_{obs} + \vec{F}_{att} + \vec{F}_{drg} + \vec{F}_{rsk} + \vec{F}_{tan}$
$\vec{T}_{net} = \vec{T}_{obs} + \vec{T}_{att} + \vec{T}_{drg} + \vec{T}_{rsk} + \vec{T}_{tan}$

// Control Loop
$\dot{\mathbf{x}} = \left\{ \begin{array}{c} \dot{v} \\ \dot{\omega}_z \end{array} \right\} = \left[\begin{array}{cc} \vec{F}_{net,x}/m & 0 \\ 0 & \vec{T}_{net,z}/I \end{array} \right]$

$\mathbf{x} = \left\{ \begin{array}{c} v \\ \omega \end{array} \right\} + \left\{ \begin{array}{c} \dot{v} \\ \dot{\omega}_z \end{array} \right\} \Delta t$

Algorithm 1: RAVE Algorithm

Chapter 4

Evaluation of the new RAVE algorithm

The evaluation of the RAVE algorithm was performed in several, iterative steps. First, a computer simulation was developed and used to evaluate the algorithm with both manual and genetic parameter tuning. Second, optimization of the various algorithm constant parameters was performed through manual and automatic means using the simulator. Finally, the selected algorithm parameters were incorporated into the test hardware and limited performance testing was performed.

4.1 Simulation

The simulator is a modular tool that was developed to permit flexible and adaptable simulation of the RAVE algorithm in an environment that closely matches the test hardware [16]. Incorporated into the simulator are the following main features:

- Line-of-sight discrete lidar model
- Non-holonomic vehicle model

- Dynamic or static maps

The individual aspects are discussed below.

4.1.1 Line-of-Sight Lidar Model

Lidar is a remote sensing technology that measures distance by illuminating a target with a laser and analyzing the reflected light.The Line-of-sight (LoS) Lidar model for the simulations was used to ensure that the surrounding environment is detected in a means that is equivalent to the Lidar installed on the test hardware, the Hokuyo URG-04LX-UG01[65]. This model works in conjunction with the developed parametric map representation to present a world of vector shapes and laser rays. Calculating the closest juncture point between the intersecting shape along each ray defines a set of obstacle points. Points are limited to being less then a defined scanning range limit, r_{scan}. In this fashion, an accurate local scan map of the surrounding world is created. Figure 4.1 shows an example of this laser model. Detected obstacle points are shown as small red dots in the above image. A grey arc indicates the sensor scanning range. The larger red circles are the defined perimeter points, their size representing the minimum obstacle range. The blue, green, red, yellow, longer black arrows (red arrowhead is off image) represent the the attractive, tangential, obstacle, drag, and risk forces respectively. The short black arrow presents the net force. Arrow length is relative to the respective force magnitude. The drag force (yellow arrow) is located at the body centre as the magnitude of the drag force was negligible at the time the

figure was generated.

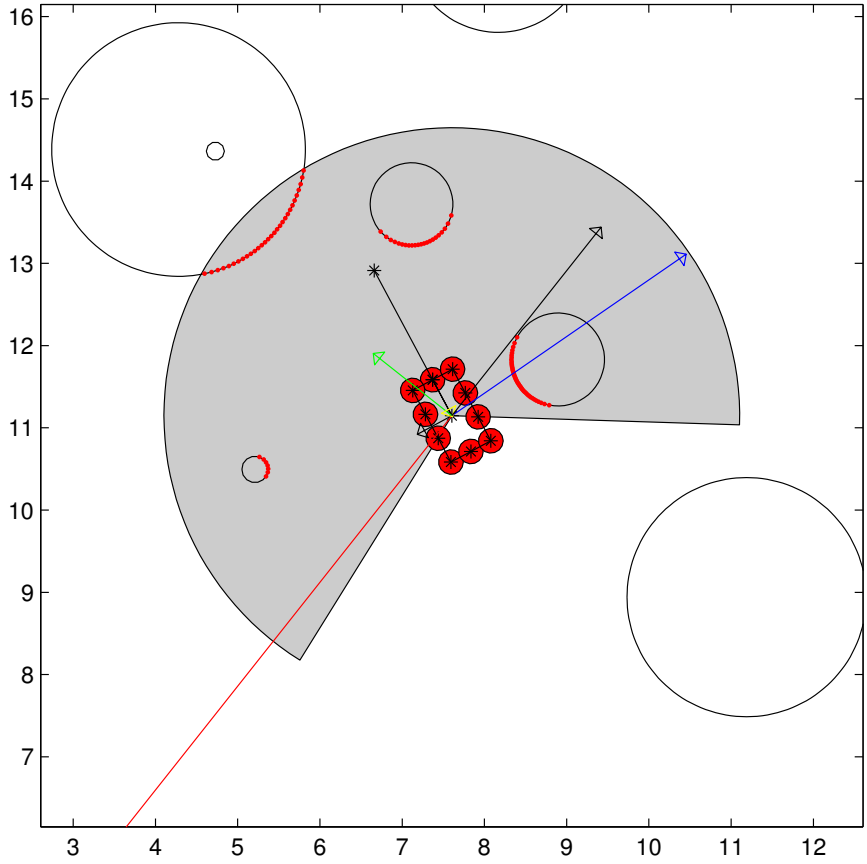


Figure 4.1: Example of Line-Of-Sight Lidar Model. 1° resolution shown.

4.1.2 Non-holonomic vehicle model

The vehicle model used for this simulation is based on the Clearpath Robotics A200 Husky. This is a skid-steer vehicle that is controlled via velocity control commands. These commands are composed of forward and angular velocity components, defined as

$$V = \left\{ \begin{array}{cccccc} \dot{x} & \dot{y} & \dot{z} & \dot{\psi} & \dot{\theta} & \dot{\phi} \end{array} \right\}^T \tag{4.1}$$

where $\dot{x}, \dot{y}$ and $\dot{z}$ represent the vehicle linear velocities and $\dot{\psi}, \dot{\theta}$ and $\dot{\phi}$ representing the angular velocities. $\dot{y}$, $\dot{z}$, $\dot{\psi}$, and $\dot{\theta}$ are defined to be 0 for 2D non-holonomic operation.

Within the simulator, vehicle position is defined in an absolute sense within the simulation map. For each timestep within the simulation the vehicle map position ($[X, Y, \Phi]$ for x-position, y-position and orientation from horizontal respectively) is updated with the distance traversed within that timestep. In this arrangement, the transition of vehicle position within the map from one time point (k) to the following time point ($k+1$) can be defined as

$$\begin{Bmatrix} X \\ Y \\ \Phi \end{Bmatrix}_{k+1} = \begin{Bmatrix} X \\ Y \\ \Phi \end{Bmatrix}_{k} + \begin{bmatrix} \dot{x}\ cos(\Phi_k)\Delta t \\ \dot{x}\ sin(\Phi_k)\Delta t \\ \dot{\phi}\Delta t \end{bmatrix}_{k} \tag{4.2}$$

with Δt being the simulation timestep, $\dot{x}$ represents the rover forward velocity, and $\dot{\phi}$ represents the rover angular velocity.

4.1.3 Dynamic or Static Maps

The maps developed for the simulation environment are defined parametrically and can be either static or randomized at time of generation. For randomly generated maps, the random number generator seed can be controlled should the user wish to revisit a particular generated map. Similar to previous simulations that also use basic geometric shapes ([29, 48, 50]), basic geometric elements are used to define the maps, circles and lines. This is different from previously used simulations where maps were defined as either a collection of discrete points or an occupancy grid [18, 20, 24]. Map

size, scaling, start and end points are defined in a map definition file. Also defined is the obstacle count and maximum obstacle size. The parametric nature of the maps allows for rapid and controlled variations between tests.

4.1.4 Simulation

Numerical simulations developed in the MathWorks Matlab software were used to investigate the improved capacities of RAVE with respect to the known issues associated with potential fields presented in Section 2.5. Only the specific cases (Sections 2.5.2 - 2.5.4) are presented here. The general local minima case (Section 2.5.1) was not tested explicitly as it is not a situation that RAVE was intended to resolve. The simulation testing matrix is shown in Table 4.1. Each series was tested 25 times against a controlled simulation map for each issue. The total test series sequence comprised a total of 900 simulations. Series were evaluated on a simple success rate comparison.

Each controlled simulation map varied the starting location for each test defined in Table 4.2. A unique pseudo-random number generator steam was created for each test set to ensure non-deterministic behaviour between sets. Within Matlab this was defined as using a Mersenne Twister algorithm with a random integer seed value of $\mathcal{U}(0, 2^{32} - 1)$ [66]. The six maps included two to represent the 'door-frame' problem (Section 2.5.2), and one each for the GNRON, narrow corridor, wide corridor, and single obstacle problems (Sections 2.5.3 & 2.5.4). Scenarios for the general local

Test Series	**# of PSP**	F_{obs}	F_{att}	F_{rsk}	F_{dmp}	F_{tan}
1	1	●	●	○	○	○
2	1	●	●	●	●	●
3	1	●	●	●	●	○
4	10	●	●	○	○	○
5	10	●	●	●	●	●
6	10	●	●	●	●	○

● Force was used ○ Force was not used

Table 4.1: Simulation testing matrix.

minima trap case (Section 2.5.1) are not tested as the RAVE algorithm was designed without a heuristic measure, and would thus be of limited improvement in these situations.

Test Series 1-3 use only a single, centrally located PSP (*Point Subject to a Potential*, Section 2.2), while Test Series 4-6 use 10 PSP points, distributed symmetrically and equally spaced on each side. Test Series 1 and Test Series 4 exclude the risk, damping and tangential forces. Test Series 2 and Test Series 5 include the risk, damping and tangential forces. All Test Series use the same force calculations. During initial testing, it was noticed that the tangential force could be detrimental in some cases, and hence the addition of Test Series 3 and Test Series 6. These replicate Test Series 2 and Test Series 5, with the exclusion of the tangential force.

Simulation Map	**Map Size**	**Start**	**Target**	**Variation**
Door Frame	20 x 20	[1 1]	[19 19]	$[\,\mathcal{U}(0,5)\;\mathcal{U}(0,5)\,]$
Obstacle Gap	20 x 20	[1 10]	[19 10]	$[\,\mathcal{U}(-1,0)\;\mathcal{U}(-2.25,2.25)\,]$
Hallway (Narrow)	20 x 20	[10 1]	[10 19]	$[\,\mathcal{U}(-5,5)\;\mathcal{U}(-1,0)\,]$
Hallway (Wide)	20 x 20	[10 1]	[10 19]	$[\,\mathcal{U}(-5,5)\;\mathcal{U}(-1,0)\,]$
GNRON	20 x 20	[8 1]	[10 10]	$[\,\mathcal{U}(-1,0)\;\mathcal{U}(-5,5)\,]$
Single Obstacle	20 x 20	[10 1]	[10 19]	$[\,\mathcal{U}(-2,2)\;\mathcal{U}(-1,0)\,]$

Table 4.2: Simulation Map Parameter Matrix

4.2 Simulation Values

The values for the constants used in the described simulations are listed in Table 4.3. These values were initially generated from the Genetic Algorithm (GA) described in Appendix B. Manual tuning was then done to result in the desired performance. These final tuned values are recorded in Table 4.3.

Constant	**Value**	**Units**	**Description**
Simulation Settings			
Δt	0.1	sec	Iteration timestep
d_{scan}	2.5	m	Scan range
d_{min}	0.2	m	Goal reached range
Algorithm Settings			
η_{lim}	2	deg	Heading tolerance
v_{tan}	0.0194	m/s	Rover speed tolerance
d_{buf}	0.15	m	PSP buffer distance
q	2.1	-	Obstacle Force Power
$F_{att,max}$	90	N	Max. Attractor force
$F_{att,min}$	10	N	Min. Attractor force
K_{obs}	60	-	Obstacle Force gain
K_{att}	2.5	-	Target Force gain
$K_{drg,lin}$	2	-	Drag Force Linear gain
$K_{drg,ang}$	10	-	Drag Force Angular gain
K_{tan}	0.05	-	Tangential Force gain
K_{rsk}	0.96	-	Risk Force gain
Genetic Algorithm Settings			
K_{angle}	0.02	-	Angular Change cost factor
K_{length}	0.5	-	Travel Distance cost factor
K_{time}	0.5	-	Travel Time cost factor
K_{range}	1	-	Range cost factor
K_{impact}	50	-	Impact cost factor
$K_{success}$	0.5	-	Success cost factor
Mating threshold	0.2	-	20% chance of variation

Table 4.3: Simulation Parameters

4.3 Vehicle Testing

The final test of the algorithm was proof-of-concept testing in an actual vehicle. The developed RAVE method was tested on a vehicle for both tele-supervision and waypoint following operations ([67, 68]). The gains and other systems settings were the same for both tests. Test hardware included a Clearpath Husky A200 vehicle, [64], a Hokuyo Laser Scanner, [65], and a joystick. Computation power was provided by a Intel i7 based laptop. During the waypoint following operation a Microstrain 3DM-GX3-25 inertial measurement unit (IMU) was added ([69]). The Robot Operating System (ROS, [70]) was employed to provide the high level abstraction framework, and all computational elements were implemented as ROS nodes. This testing is further discussed in Section 5.2.

Chapter 5

Results and Discussion

5.1 Simulation Results

Several types of simulations were performed to provide specific performance results to the local minima conditions presented in Section 2.5. Each simulation was performed with the same boundary conditions and vehicle configurations. Example results of Obstacle Gap Map simulation set are presented in Figures 5.1, 5.2, and 5.3. Full plots of the results are shown in Appendix A. In these selected restult sets, the progressive improvement from complete failure of the basic APF forces with a single PSP (Figure 5.1), to the 84% success of the RAVE forces without the tangential force with a single PSP (Figure 5.2), to finaly complete success of the RAVE forces without the tangential force with ten PSPs (Figure 5.3). Presentation of selected single simulations in Figures 5.4, 5.6, 5.8, 5.9, 5.10, and 5.11 show the vehicle path as a two colour line with red representing $0\,m/s$ and green as the vehicle maximum velocity of $0.5\,m/s$. Obstacles are represented as circles and lines. The vehicle starting

	Test Series (25 simulations)					
Map	**1**	**2**	**3**	**4**	**5**	**6**
Thread-the-Needle	0%	100%	100%	0%	100%	100%
Obstacle Gap	0%	0%	84%	0%	0%	100%
Hallway (Narrow)	56%	100%	100%	68%	100%	100%
Hallway (Wide)	100%	100%	100%	100%	100%	100%
GNRON	0%	100%	100%	0%	100%	100%
Single Obstacle	40%	100%	92%	48%	100%	100%
Test Series Average	33%	83%	96%	36%	83%	100%

Table 5.1: Simulation Results Matrix.

point is indicated as the green circle and the target point is shown as the red circle. The target is considered reached when the vehicle CofG is within 20 cm. The grey arc shows the laser scanning arc ($\pm 120°$, $0°$ is the forward velocity direction). Figures 5.5 and 5.7 show the velocity component for Figures 5.4 and 5.6. Table 5.1 lists the aggregate results of the simulation testing.

Performance improvements for the various RAVE components can be readily seen in Table 5.1. Comparing Test Series 1 versus Test Series 2 and Test Series 4 versus Test Series 5, the improvements of RAVE as a whole can clearly be seen with the rise of 50 and 47 percentage points respectively. This improvement holds across all maps.

The effectiveness of the use of multiple PSP points can be seen from the comparison of Test Series 1 - 3 against Test Series 4 - 6, respectively. Here, only small

improvement of 3 and 4 percentage points are seen on Test Series that do not include the tangential force. This improvement holds across all maps.

With the removal of the tangential force, comparing Test Series 2 with Test Series 3 and Test Series 5 with Test Series 6, improvements of 13 and 17 percentage points are seen, respectively. These improvements from the removal of the tangential force are due to increased success on the Obstacle Gap map. The Single Obstacle map shows a reduction in success between Test Series 2 with Test Series 3, both of which are single PSP cases. In all other maps there was no change in performance. This seems to indicate that the benefit of the tangential force is reduced or negated with the move to using multiple PSP points to define the vehicle.

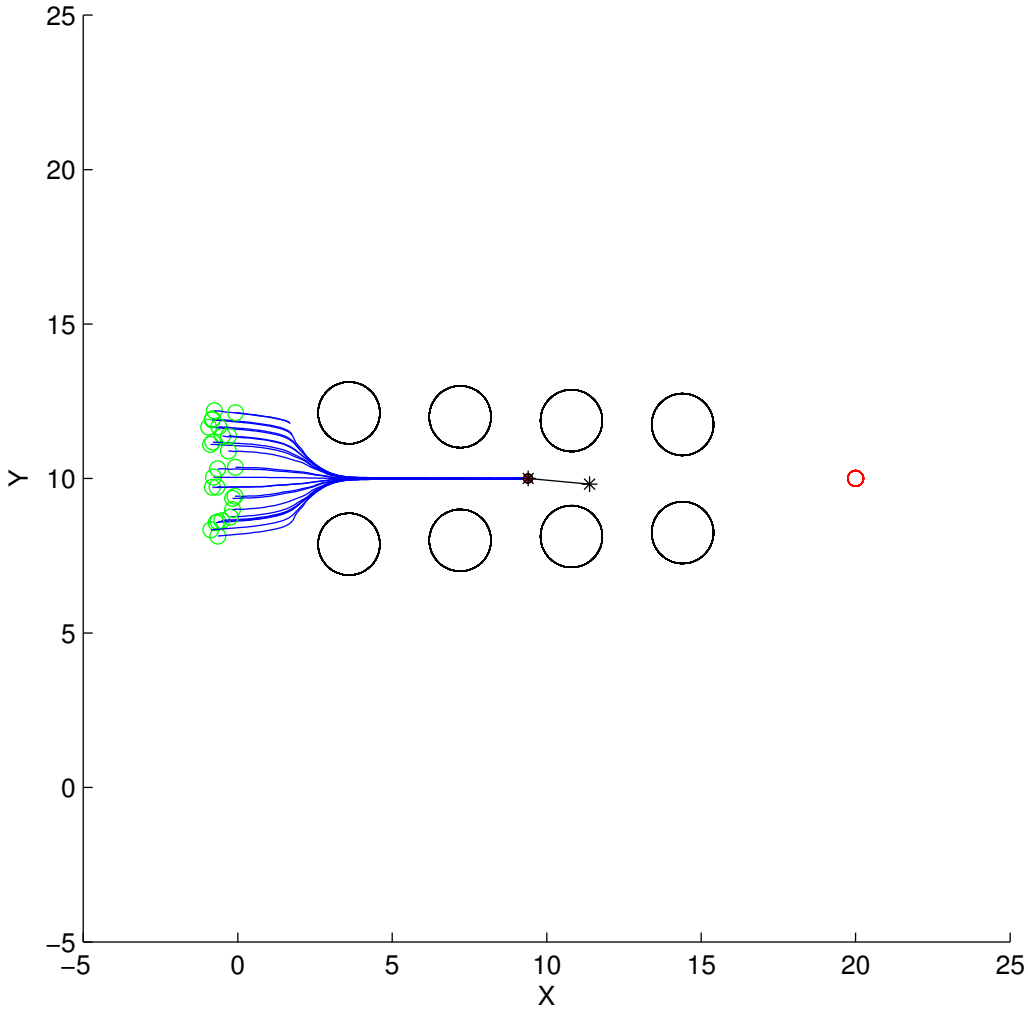


Figure 5.1: Test Series 1 - Obstacle Gap Map. 1 PSP used. Only attractive and repulsive forces used. Success: 0 of 25.

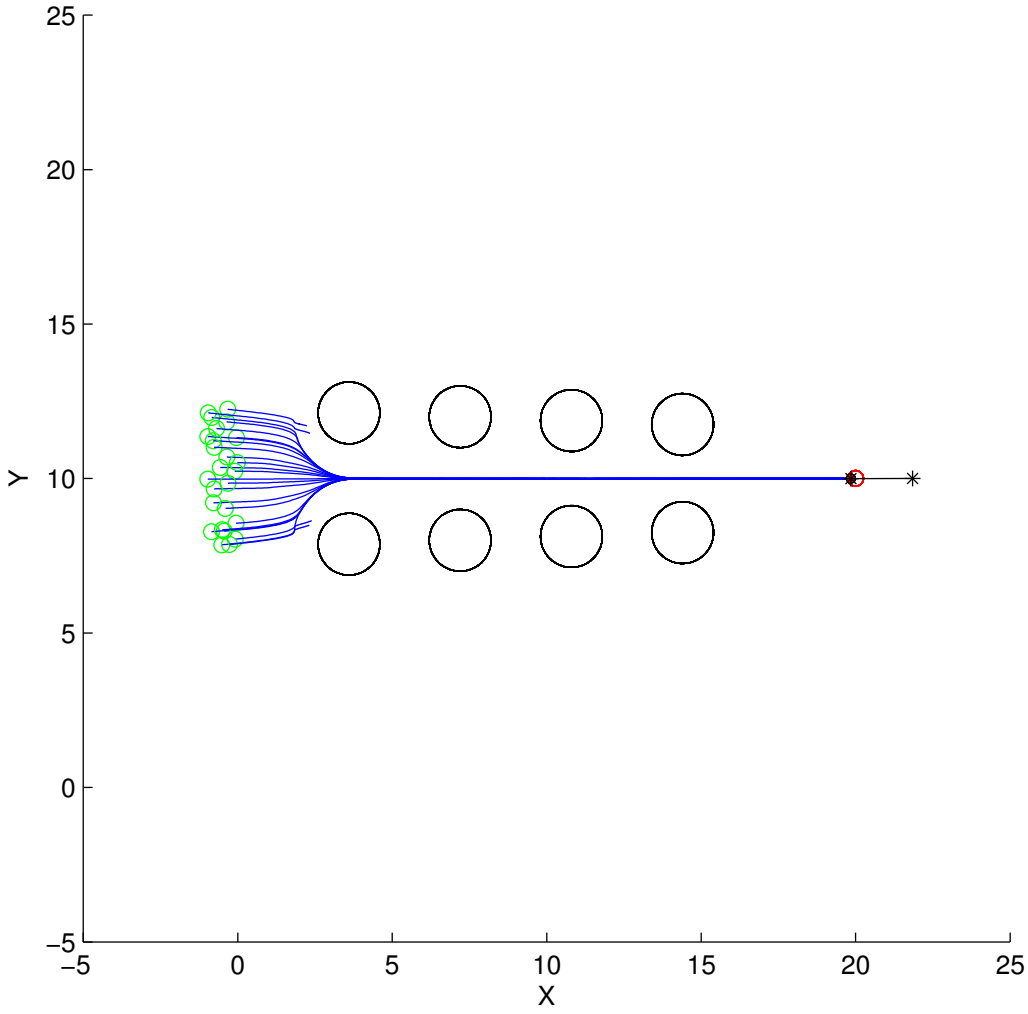


Figure 5.2: Test Series 3 - Obstacle Gap Map. 1 PSP used. All forces used except for tangential forces. Success: 21 of 25.

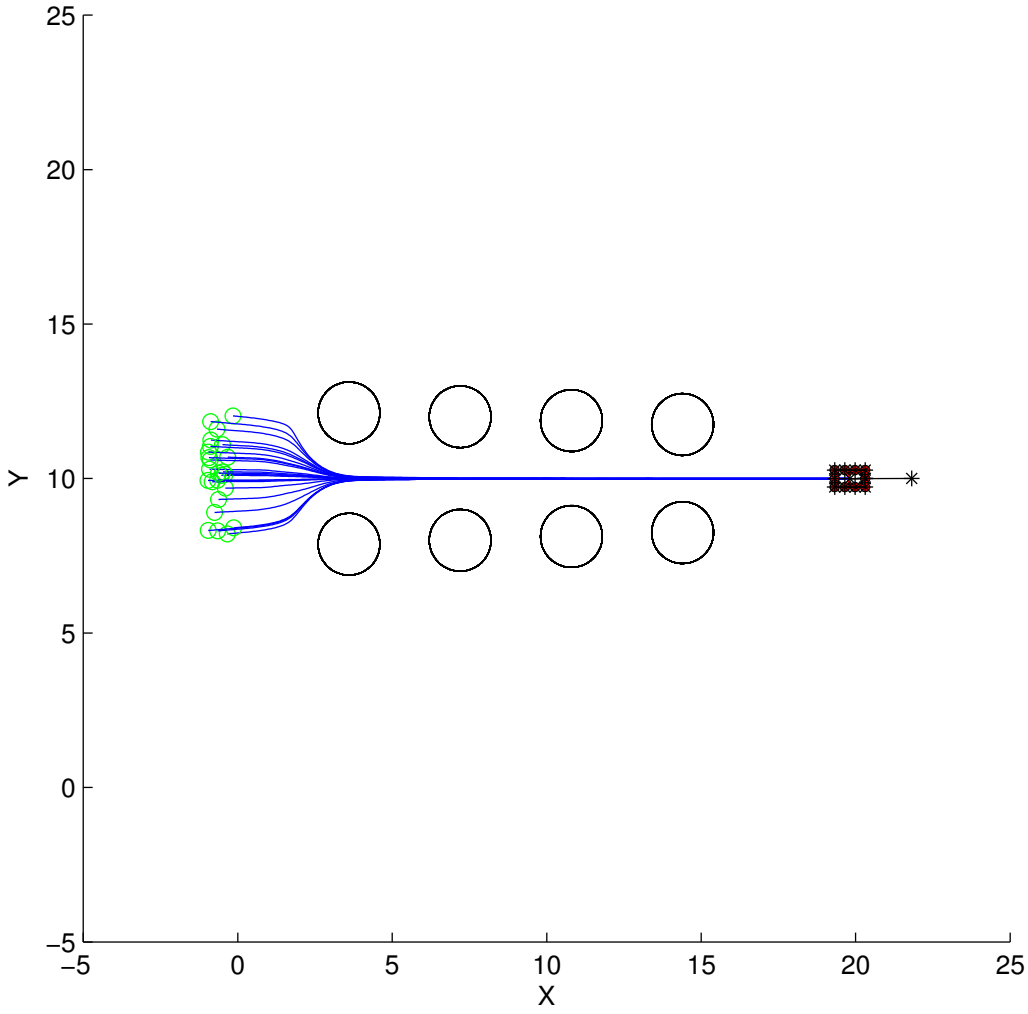


Figure 5.3: Test Series 6 - Obstacle Gap Map. 10 PSP used. All forces used except for tangential forces. Success: 25 of 25.

5.1.1 Specific Local Minima Simulation

Specific local minima cases are presented in Figures 5.4, 5.6, 5.8, 5.9, and 5.10. Each of these cases present stable performance without amplification of oscillations or obstacle impact.

5.1.1.1 "Thread-the-Needle" Problem

Figure 5.4 demonstrates the ability of the system to overcome the "Thread-the-Needle" problem as described in Section 2.5.2. This simulation was configured with a series of circular obstacles with ever decreasing spacing (2.25 m, 2.0 m, 1.75 m, and 1.5 m gaps were used) and with starting and ending points such that the vehicle should pass between all gaps. The purpose of this test was to demonstrate the effectiveness of the risk force (F_{rsk}) in overcoming the "Thread-the-Needle" issue. During the simulation, as the local minima is approached, the effect of the F_{rsk} is to reduce the net obstacle force. This reduction in net obstacle force reduces the vehicle speed, which increases the value of ϵ (see Equation 3.8). The increase in ϵ increases the risk force, allowing passage. This cycle repeats until equilibrium is met or ϵ is saturated at 1. The overall effect is to slow the vehicle as it approaches the obstacles, and then accelerate it once the vehicle centre passes the narrowest point of the gap. The direct path taken allows for the effects of angular velocity to be ignored. Figure 5.5 shows the change in forward velocity associated with the map in Figure 5.4. This figure shows that there is a rapid, smooth transition from the prescribed simulation

velocity to an equilibrium velocity that allows the vehicle to pass through the gap. Four points of velocity adjustment are seen, corresponding to each obstacle gap. The first two have only minimal effect and show as only small ripples in forward velocity, while the third and fourth obstacles show a much greater effect on the forward velocity. Similarly Figure 5.6 and Figure 5.7 demonstrate a similar effect when sharp edged doors are simulated. As the vehicle passes through a series of ever encroaching door frames, the F_{rsk} causes the vehicle velocity to decrease until passage through the doorway is possible.

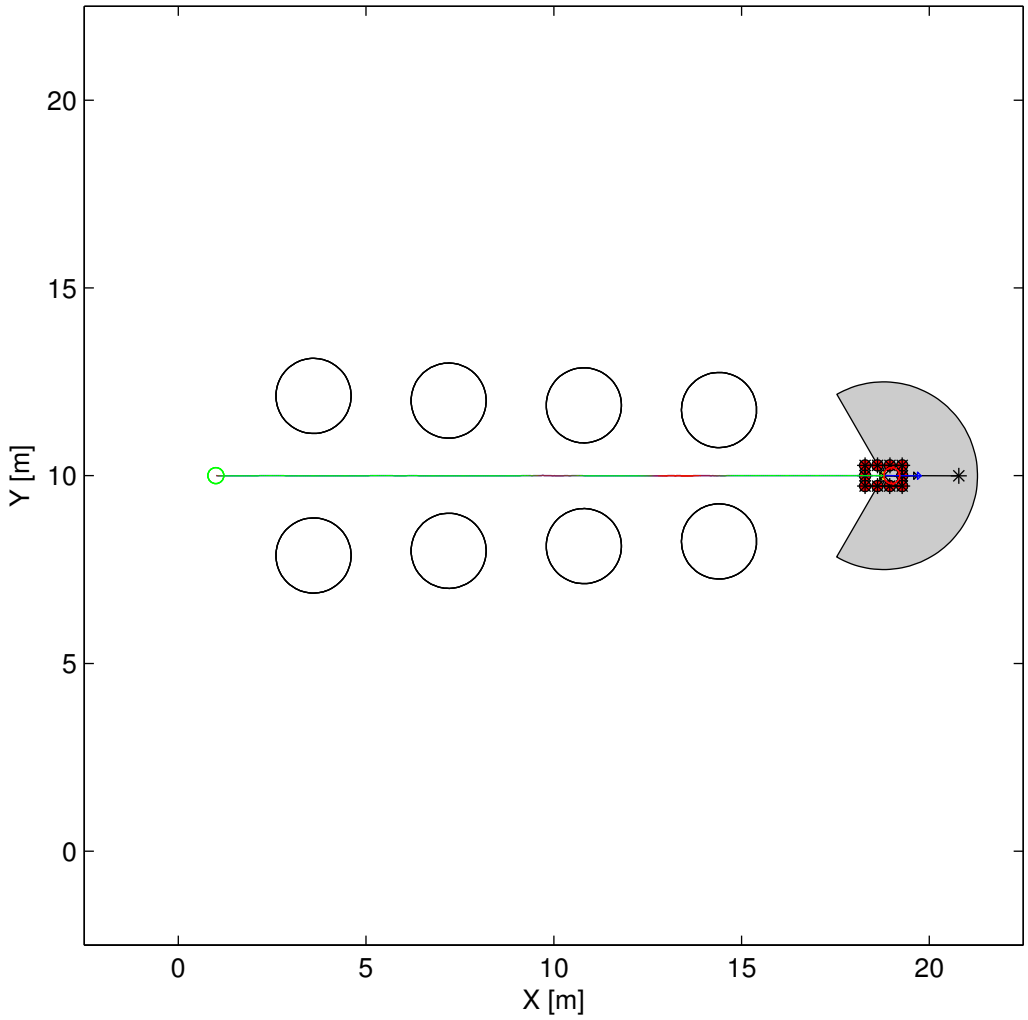


Figure 5.4: Passage through narrow openings. Progressively narrowing openings are overcome via F_{rsk}. Path colour corresponds to forward velocity, v_x, with green representing 0.5 m/s. Grey arc indicates sensor scanning range.

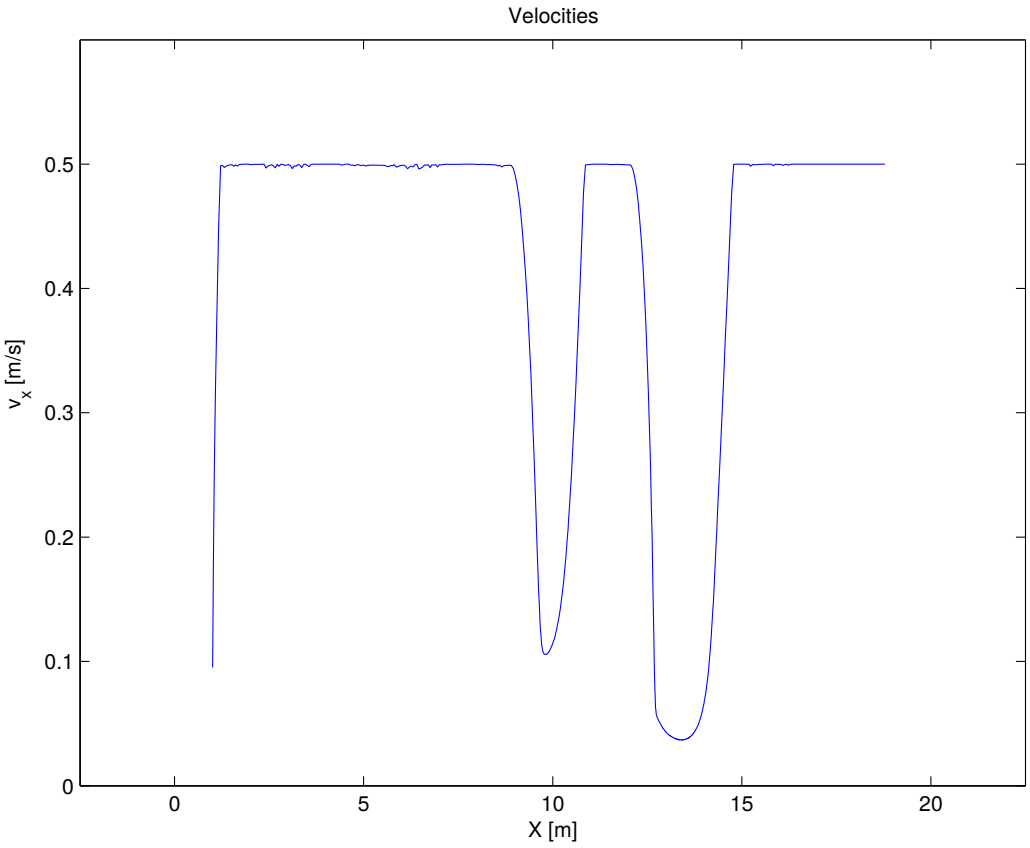


Figure 5.5: Velocity of vehicle through the map shown in Figure 5.4. Vehicle slowdown is the effect generated by the interaction of obstacle force, F_{obs}, and the risk force, F_{rsk}.

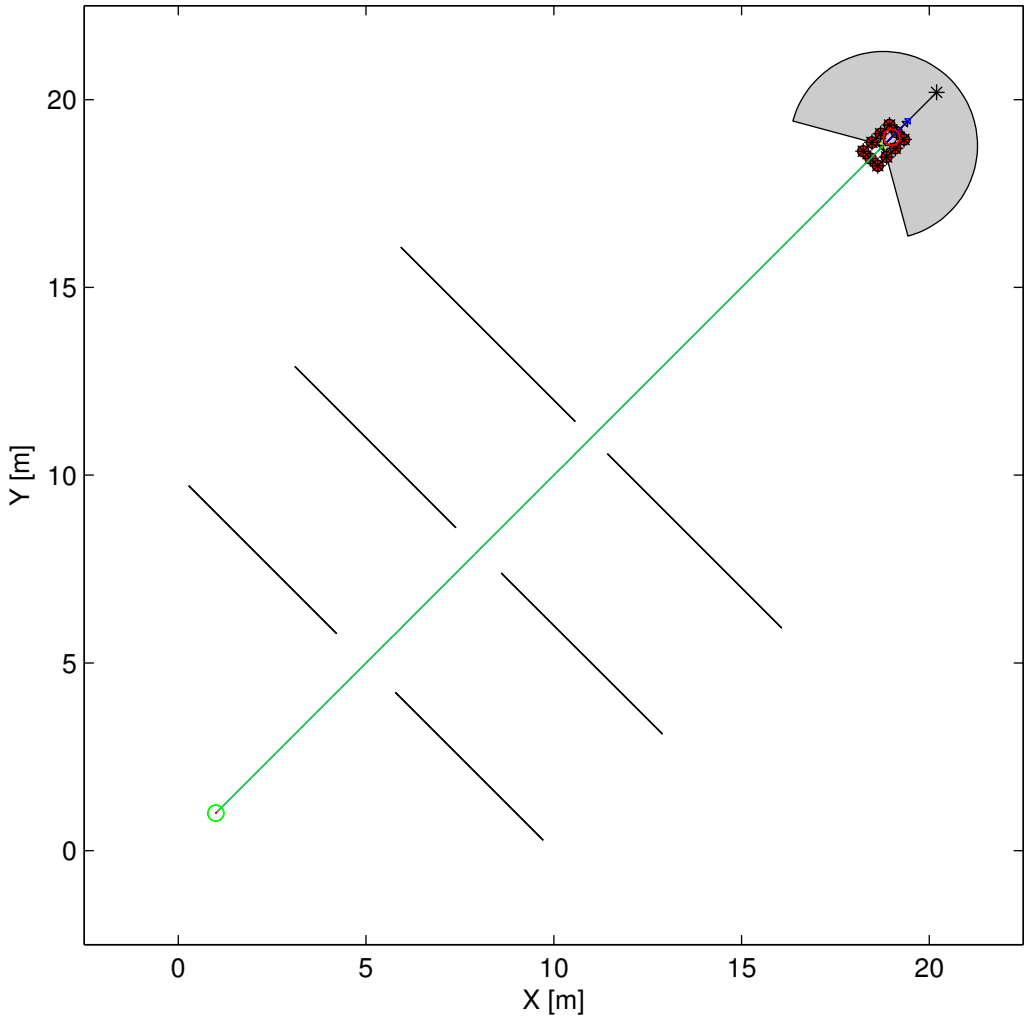


Figure 5.6: Passage through narrow openings. Progressively narrowing sharp openings are overcome via F_{rsk}. Openings are $2.21\,m$, $1.71\,m$, and $1.21\,m$ largest to smallest. Path colour corresponds to forward velocity, v_x, with green representing $0.5\ m/s$. Grey arc indicates sensor scanning range.

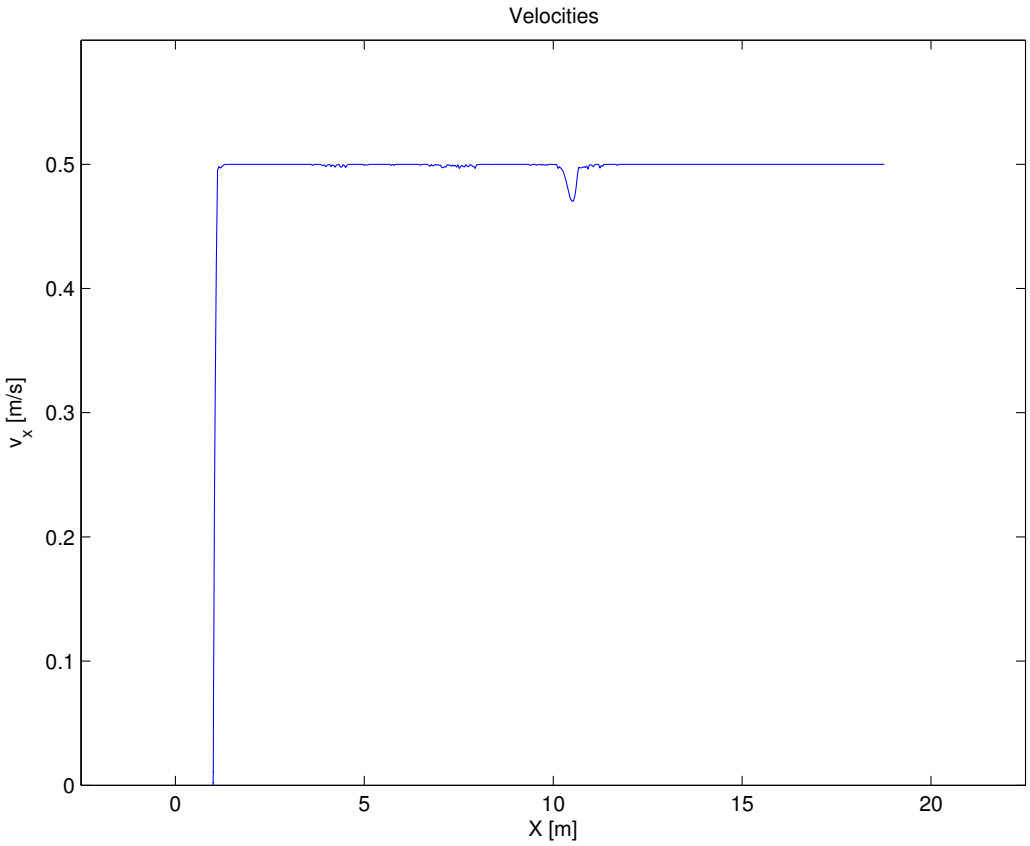


Figure 5.7: Velocity of vehicle through the map shown in Figure 5.6. Vehicle slowdown is the effect generated by the interaction of obstacle force, F_{obs}, and the risk force, F_{rsk}.

5.1.1.2 Undamped Responses

Figure 5.8 presents the performance within narrow passages, as a demonstration of overcoming the undamped motion problem described in Section 2.5.3. Here, the vehicle begins in a position which forces an avoidance response to enter the narrow passage. Once in the passage, the vehicle performs well, with no oscillations or other unexpected behaviour. Additionally, the vehicle path quickly centers within the passage and has no unstable response upon exit. This behaviour benefits from F_{rsk}, F_{tan}, and F_{dmp} to smoothly enter the passage, while the multiple perimeter points ensure that no extremity of the vehicle is at risk of obstacle impact.

Figure 5.9 presents the performance when meeting lone obstacles, as a demonstration of overcoming the undamped disturbance problem described in Section 2.5.3. Here, the vehicle begins in a position that forces an avoidance response around the presented obstacle. Without oscillation the vehicle slows, avoids the obstacle, and resumes a direct path to the target. This behaviour benefits from F_{rsk}, F_{tan}, and F_{dmp} to smoothly enter the passage, while the multiple perimeter points ensure that no extremity of the vehicle is at risk of obstacle impact.

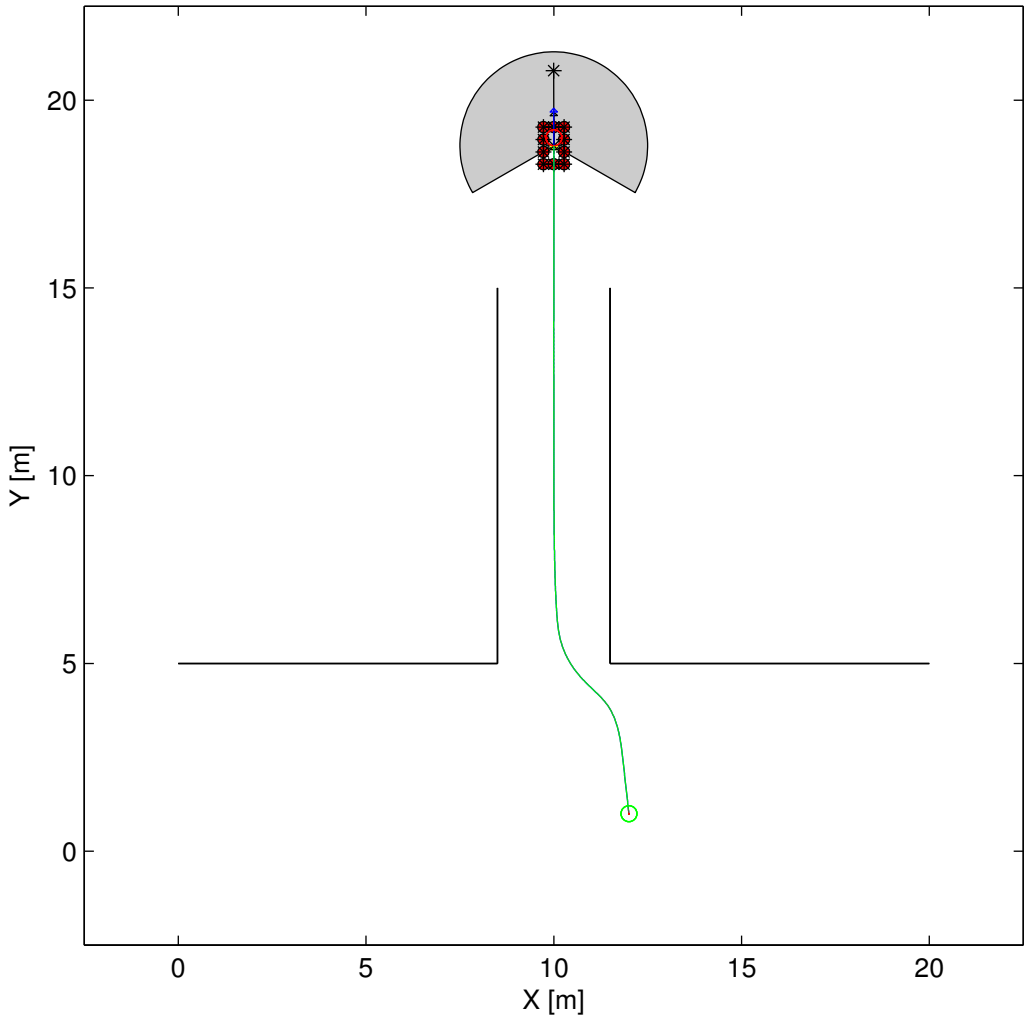


Figure 5.8: Stability through narrow passages. Effects of damping and multipoint perimeter stabilizes vehicle motion. Path colour corresponds to forward velocity, v_x, with green representing 0.5 m/s. Grey arc indicates sensor scanning range.

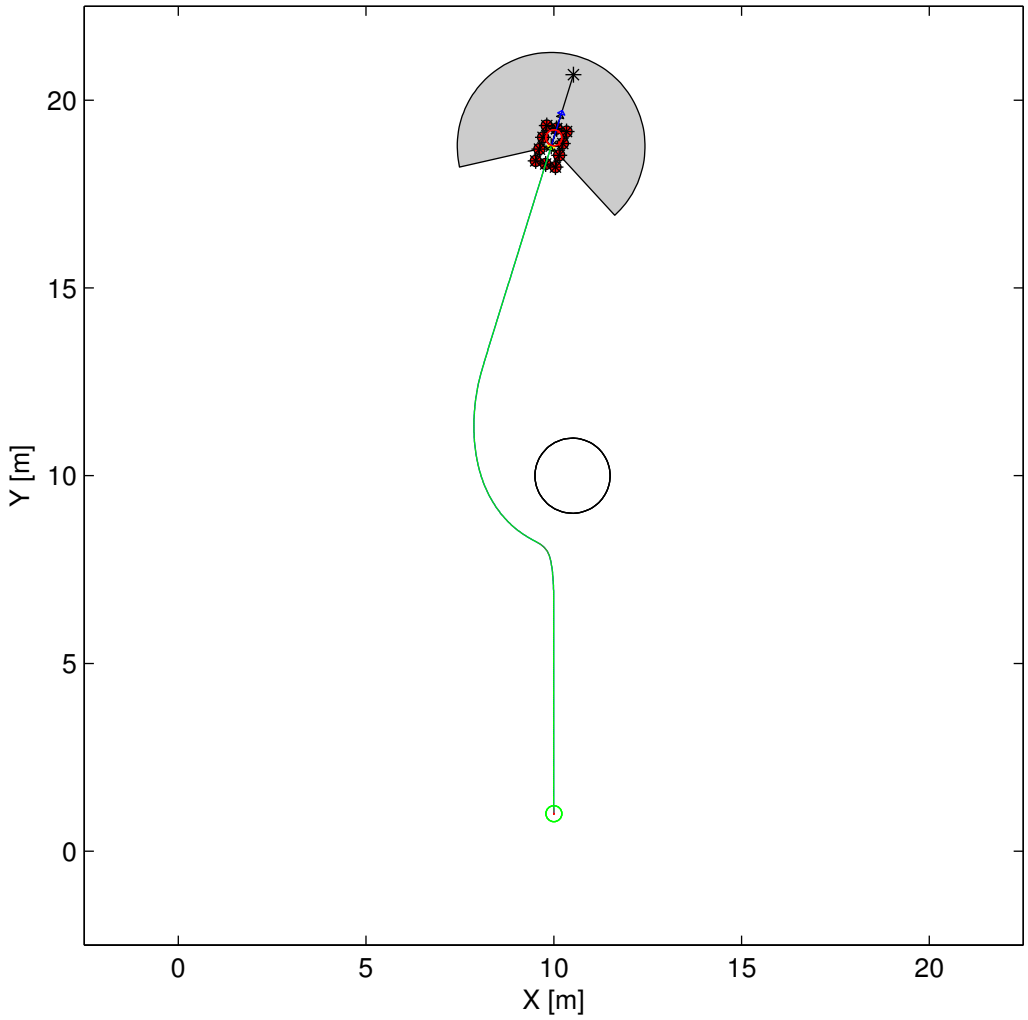


Figure 5.9: Stability around lone obstacles. Effects of damping and multipoint perimeter stabilizes vehicle motion. Path colour corresponds to forward velocity, v_x, with green representing 0.5 m/s. Grey arc indicates sensor scanning range.

5.1.1.3 GNRON

Figure 5.10 presents the performance under GNRON conditions, described in Section 2.5.4. In this case, the vehicle begins in a position with a trivial path to the target, but with an increasing F_{obs} as the target is approached. The vehicle slows and with the feedback mechanism that increases F_{rsk} as the vehicle slows, the target is reached without unstable oscillation. This behaviour relies heavily upon the effects of F_{rsk} to overcome the local minima.

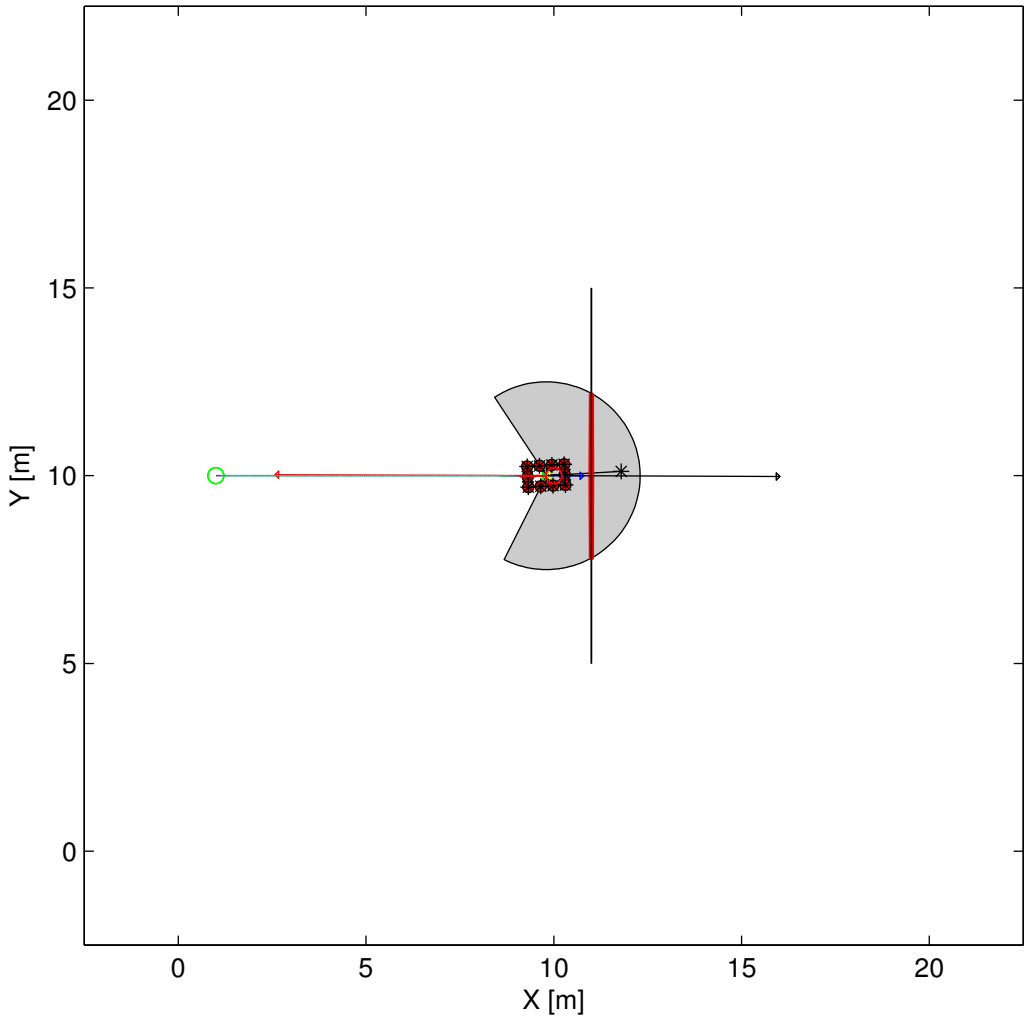


Figure 5.10: Stability in a GNRON condition. Effects of damping and multipoint perimeter stabilizes vehicle motion. Path colour corresponds to forward velocity, v_x, with green representing 0.5 m/s. Grey arc indicates sensor scanning range.

5.1.2 Simulated Rock Field

Example results for a random rock field are shown in Figure 5.11. Four separate simulations are shown, all starting from unique starting positions, and with a common target. For this simulation, 100 obstacles are randomly placed within a $45\,m \times 45\,m$ region ($x = \mathcal{U}(2.5, 47.5)\,m, y = \mathcal{U}(2.5, 47.5)\,m$) to create the rock field. Each rock has a randomly defined radius, $r_{obs} = |\mathcal{N}(0, \sigma^2)|$, where σ is a constant for all rocks generated. The start and target points are surrounded by a $1.5\,m$ obstacle stay-out clearance regions to ensure that invalid beginning and end target locations are not generated. For the generated distribution shown in Figure 5.11, a σ of $0.81\,m$ was used for the radius. As the same generated map was used for each test, the randomized starting points could not guarantee that a regular spacing would be had for each starting point from the maps obstacles.

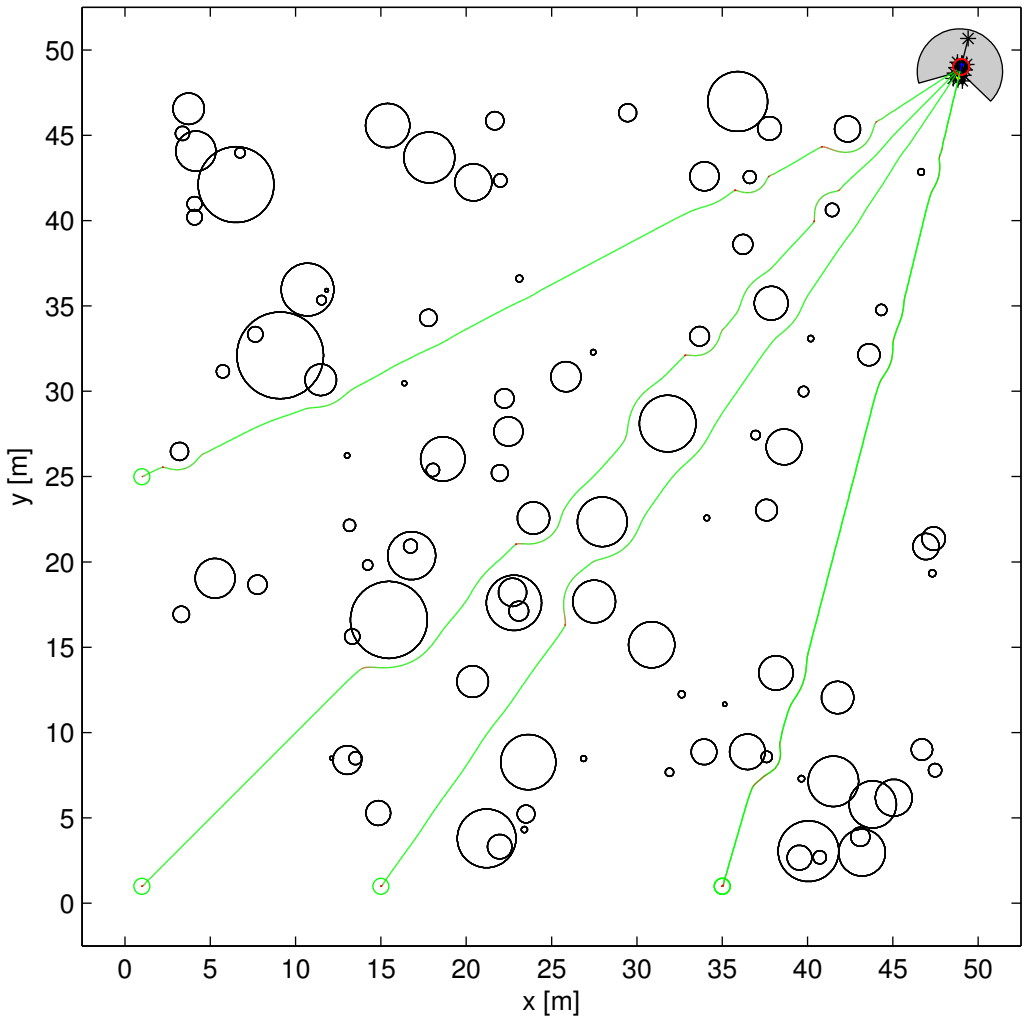


Figure 5.11: Example of successful paths found through a randomized rock field. Path colour corresponds to forward velocity, v_x, with green representing 0.5 m/s. Grey arc indicates sensor scanning range.

5.2 Deployment

Physical testing took place on the ground floor of the Canal Building of Carleton University. All test video was captured in one take, with no editing except for the final presentation layout. The joystick was used as the user input control of the target point for the tele-supervision case, and as a dead-man safety switch in both tele-supervision and waypoint navigation cases. No direct control of the vehicle was present; all control was through the user directed target point or in the user defined waypoints.

5.2.1 Tele-supervision

The video presented in [67] demonstrates tele-supervision utilizing the RAVE method. This is accomplished by adapting the position of the target attractor to be operator controlled. In this fashion, the operator input is combined with the RAVE outputs. This arrangement allows for the operator to provide the operational intent, while the vehicle can ensure that obstacles which may be unseen by the operator can be avoided. This deployment included the addition of an operator joystick, configured to position the target at the joystick input angle and at a range from 0 to 4 meters from the vehicle; this range correlates to $F_{att,max}$.

This test involved operator-controlled navigation through a narrow passage; for which a doorway between a corridor and a room was used. The test commenced in the corridor and controlling only the target point, the operator directed the vehicle

through the door. The vehicle was not initially aligned to the opening, and some caution on the part of the operator can been seen in [67]. The goal is readily achieved and some interesting behaviours developed. The two most interesting observations were the slowing of the vehicle as it passed through the narrow opening, a direct effect of F_{rsk}. The second interesting observation was of the organic reverse and passage alignment behaviour, a direct effect of the multiple corner points. As the vehicle turned to align itself to the passage opening, the front right corner point passed closer to the door frame, increasing the repulsive force. This increase in turn caused the vehicle to reverse and realign to the passage. On the return path, the vehicle was given a constant command to go forward, and without any other input, passed through the door, avoided a wall and began to track a centering path in the corridor.

Figure 5.12 shows a screen capture taken from [67], showing in-range detected obstacle points as small white squares, and all other detected points as small dots. The cyan circles are the defined perimeter points, their size representing the minimum obstacle range. The green dot is the operator controlled target point. The cyan and yellow arrows represent the the net and attractive forces, respectively. The red arrows represent the obstacle forces generated at each perimeter point. Arrow length is relative to the respective force magnitude. For clarity, the drag and tangential forces were not shown at the time of test.

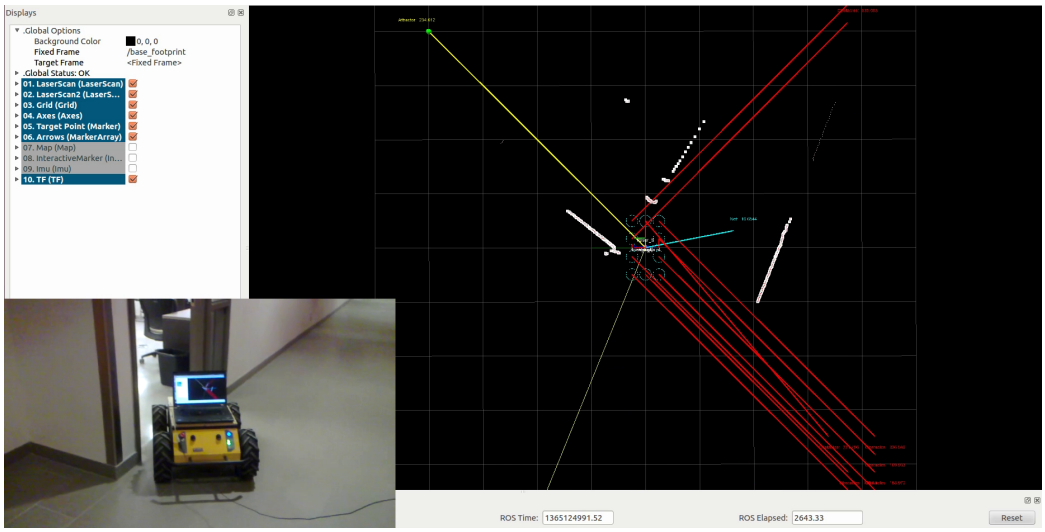


Figure 5.12: Deployment of RAVE for tele-supervision of vehicles. A Clearpath Husky A200 vehicle, [64], outfitted with a Hokuyo Laser Scanner, [65], negotiates a doorway under tele-supervision.

5.2.2 Autonomous Operation

Autonomous navigation utilizing the RAVE method is accomplished by providing a set of waypoints through which the vehicle must travel. This deployment included the addition of an inertial measurement unit (IMU) and an operator joystick to operate as a dead man's safety switch. To provide an estimation of vehicle position, the vehicle odometry and the IMU data were fused using a simple Extended Kalman Filter (EKF). At each scan, this position estimate is used to provide a range and bearing to the target waypoint relative to the vehicle.

Demonstration of this operation is shown in [68]. This demonstration comprises an approximate 65 m loop about the ground floor of the Carleton University Canal

Building, see Figure 5.13. Beginning at the lower left green dot, the vehicle proceeded clockwise through the defined waypoints. Of the interesting behaviours noted, the most interesting was the excellent demonstration of a GNRON local minima condition which occurred at the upper right corner of Figure 5.13. In this case, due to the placement of the waypoint, the vehicle could not achieve the target and stopped within a few centimetres of the target. Manually pushing the vehicle slightly toward the target failed to trigger a 'Waypoint Reached' condition, and then caused the vehicle to back away from the target, before settling back into the local minima. Upon review, it appears that a combination of position estimation error and incorrect buffer space considerations on the part of the operator led to this problem. Upon manually clearing the waypoint, the vehicle promptly continued onto the lower right waypoint with no other operator interaction.

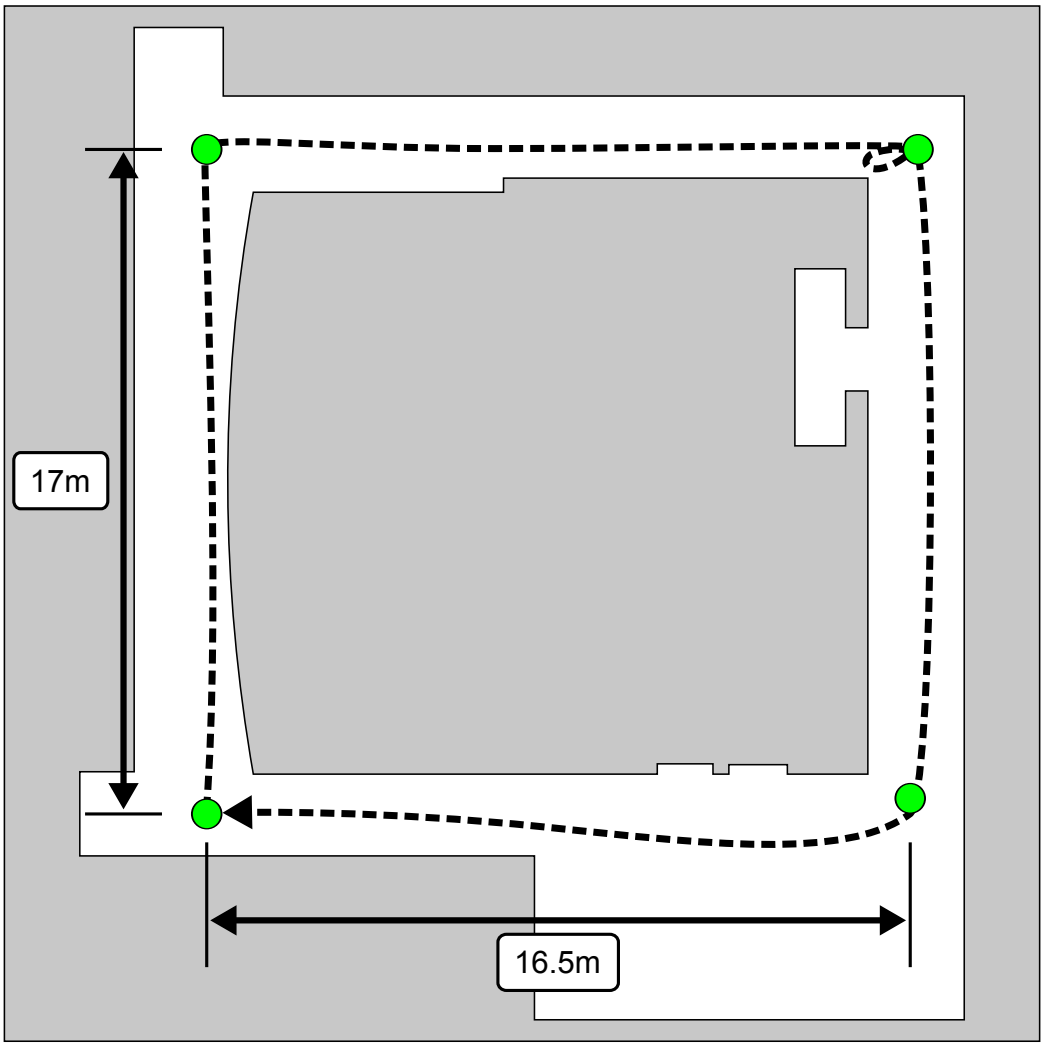


Figure 5.13: General Canal Building outline, ground floor. Autonomous navigation testing approximate path shown by the dotted line. Green circles are the defined waypoint. The path loops at the top right relate to a poorly placed waypoint that created an unachievable GNRON condition that was cleared manually.

Chapter 6

Summary and Conclusion

6.1 Summary

In this thesis, the RAVE method for reactive vehicle control was developed. RAVE has the motivation of improving reactive local control of autonomous vehicles using potential field based methods. This proposed method used a multipoint (several PSP, *Point Subject to a Potential*, Section 2.2) model for vehicle representation and incorporated repulsive, attractive, risk, and damping forces. These elements were taken from or inspired by existing works by the authors Khatib and Lynch ([15, 48]), with additional inspiration from many of the other contributors to the field.

The prime intent of RAVE is to overcome several of the known local minima problems that are inherent to potential fields. These local minima problems were described in Section 2.5. The specifically addressed problems are the "Thread-the-Needle", undamped motion, and *Goals NonReachable with Obstacles Nearby* (GNRON) situations. The "Thread-the-Needle" problem occurs when the vehicle attempts to pass

through narrow openings, such as between door frames or other obstacles. Undamped motion problems include both constant stimulation, such as a hallway or wall, and impulse situations where a lone obstacle is detected. The final situation addressed is the GNRON situation where the target goal may be physically accessible but the repulsive pressure from the local obstacle prevents the vehicle from reaching it.

A description of the RAVE method was presented in Chapter 3. As mentioned above, the RAVE algorithm used both the collection of repulsive, attractive, risk, and damping forces with a multipoint model for vehicle representation. A sensor – a 2D laser is used here – first generates a point map of the vehicle surroundings. For each point of the vehicle representation, the net repulsive forces from all detected points is calculated. Combining these forces and the respective torques to act about the vehicle Center-of-Gravity (CofG) creates the vehicle net obstacle force (Section 3.2.3). The attractive force is generated from the range from the vehicle CofG, with the torque acting from the vehicle front (Section 3.2.4). Viscous damping forces are generated proportional to the current vehicle linear and angular speeds (Section 3.2.5). The risk force is generated in direct opposition to the current obstacle force but inversely proportional to the vehicle speed (Section 3.2.6). The balance of forces as they act about the vehicle CofG is then resolved to a single force along the vehicle forward axis and a single torque about the vertical (Section 3.2.8). Applying this single force and torque as a virtual force to the vehicle body results in desired vehicle linear and angular accelerations. These accelerations then lead to desired vehicle linear and angular velocities (Section 3.3). Initially a fifth force, the tangential(Section 3.2.7),

was employed within the algorithm, but following testing it was found to have only a small if not detrimental effect on the RAVE simulation results.

The methods of simulation testing were presented in Chapter 4. A series of simulation tests were executed to capture the effects of different components of the RAVE algorithm. Six test series were defined (see Table 4.1) that compared the simulation success of using only the attractive and repulsive forces, all RAVE forces, and all RAVE force except for the tangential force for both a single vehicle PSP and for a multipoint vehicle model. These six test cases were each tested over six different maps that presented two cases of "Thread-the-Needle" situations, three cases of undamped motion situations, and one GNRON situation. For each map, 25 simulations were performed, with randomized starting points, for a total count of 900 simulations.

The results of testing were presented in Chapter 5 with full results presented in Appendix A. The simulation results demonstrated the capability of the RAVE method under the situations defined in Table 4.1, and are presented in Table 5.1. Comparing classical forces (Test Series 1 and 4) versus RAVEforces (Test Series 2 and 5, respectively), the improvements due to RAVE forces were shown with the rise in the success rate of 50 and 47 percentage points, respectively. The use of multiple PSP points was shown to benefit in the comparison of the single PSP scenarios (Test Series 1 - 3) versus multiple PSP scenarios (Test Series 4 - 6) with a small improvement of 3 and 4 percentage points versus Test Series that do not include the tangential force. The removal of the tangential force (Test Series 2 and 5 versus Test Series 3 and 6) show improvements of 13 and 17 percentage points, respectively. These improvements from

the removal of the tangential force are due to increased success on the Obstacle Gap map. This indicates that the benefit of the tangential force is reduced or negated with the move to using multiple PSP points to define the vehicle.

In addition to the simulation testing, the RAVE algorithm was implemented on a Clearpath A200G Husky rover as a proof of concept (Section 5.2). This vehicle was used in both tele-supervision and waypoint following situations. Though not conclusive, the vehicle showed good performance and responded well to both situations.

6.2 Conclusion

The series of simulation results and vehicle implementation presented here supports the claim that the RAVE algorithm has improved performance against several of the known local minima issues that plague potential field reactive navigation. As a local navigation controller, RAVE will still be dependant upon a global planner, and does not resolve or attempt to resolve all issues that exist with potential fields.

6.3 Future Work

Continued research on this topic will include parameter optimization of the various constants, and where possible, the replacement of constants with formulated values. Additionally, the modification or removal of the tangential force must be considered as testing showed that in its current state, the tangential force can be a detriment to

the performance of the RAVE system. Finally extension and validation of the RAVE algorithm to three dimensions, and packaging within an reactive navigation software package are possible endeavours.

Corrections

All page and section numbers refer to the initial submitted copy. Relevant page and section numbers in the corrected version may not be the same due to changes.

General

1. Colour - figure colours not addressed as these comments only pertains to the printed B/W copy

2. Colour - Link colour (references, figures, etc) changed to darkgrey from grey for better visibility on all versions

Abstract

1. Page ii - [EL] - Removed final line

Chapter 1

Introduction

1. Page 1 - [EL] - Slight rephrasing of first two sentences.

2. Page 9 - [EL] - Defined RAVE at initial usage

3. New Section - [MA] - Added in a new section that states the contributions of the thesis.

Chapter 2

1. Page 11 - [EL] - Indicated where the discussions on APF variations and issues are to be found in the thesis.

2. Page 17-18 - [MA] - Added more description of Brian's Risk forces, and PID controller.

3. Page 19 - [EL] - Added missing equation element definitions, and fixed equation ordering in text.

4. Page 20 - [EL] - Added missing equation element definitions

5. Page 27 - [EL] - Slight rephrase of second sentence of paragraph 2

6. Page 31 - [EL] - LaTeX typo: PFA $\rightarrow$ RAVE

7. Page 36 - [EL] - Typo: ib $\rightarrow$ in

Chapter 3

1. Page 38 - [EL] - Added statement of what APF limitations RAVE addresses and does not address to the end of Section 3.1.

2. Page 45 - [EL] - Changed "is in the opposite direction and linearly proportional" $\rightarrow$ "is in the opposite direction and is linearly proportional"

3. Section 3.2.1 - [EL] - Clarified several terms used in the first paragraph.

4. Section 3.2.6 - [MA] - Clarification of differences between Brian's work.

5. Section 3.3 - [EL] - Typo: $\vec{F}_{dmp} \rightarrow \vec{F}_{drg}$

6. Section 3.4 - [EL] - Clarification of $\mathcal{O}$ notation. Review of C++ code confirms $\mathcal{O}(n^2)$.

Chapter 4

1. Section 4.2 - [EL,MA] - Clarification to GAO, and section is moved to Appendix B, as it is not part of the core thesis.

Chapter 5

1. Section 5.1 - [EL] - Changed example Figures 5.1 - 5.3 to be have better context to each other. Also included text discussions of what is shown in the figures.

2. Section 5.1.1.1 - [EL] - Clarified Figures 5.4-5.7, improved descriptions, added description for Figure 5.6, and made minor corrections to annotation of Figure 5.5.

3. Section 5.1.1.2 - [EL] - 3rd sentence changed “uncontrolled behaviour” to “unexpected behaviour”

4. Section 5.1.2 - [EL] - Clarified final sentence.

List of References

[1] J. Borenstein and Y. Koren. Real-time obstacle avoidance for fast mobile robots. *IEEE Transactions on Systems, Man and Cybernetics*, 19(5):1179–1187, September 1989. ISSN 0018-9472. doi: 10.1109/21.44033.

[2] J. Borenstein and Y. Koren. The vector field histogram-fast obstacle avoidance for mobile robots. *IEEE Transactions on Robotics and Automation*, 7(3):278–288, June 1991. ISSN 1042-296X. doi: 10.1109/70.88137.

[3] Northrop Grumman. F6B: The Industry's Most Versatile Platform, March 2012. URL `http://www.northropgrumman.com/Capabilities/Remotec/Products/Pages/F6B.aspx`.

[4] QinetiQ North America. Dragon Runner 20: Small Unmanned Ground Vehicle, November 2013. URL `https://www.qinetiq-na.com/wp-content/uploads/data-sheet_dr-20.pdf`.

[5] Kazunori Ohno, Shinji Kawatsuma, Takashi Okada, Eijiro Takeuchi, Kazuyuki Higashi, and Satoshi Tadokoro. Robotic control vehicle for measuring radiation in Fukushima Daiichi Nuclear Power Plant. pages 38–43. IEEE,

November 2011. ISBN 978-1-61284-769-6, 978-1-61284-770-2, 978-1-61284-768-9. doi: 10.1109/SSRR.2011.6106792. URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6106792.

[6] Wei Boyu, Gao Junyao, Zhu Jianguo, and Kejie Li. Design of a Large Explosive Ordnance Disposal Robot. In *Second International Conference on Intelligent Computation Technology and Automation, 2009. ICICTA '09*, volume 3, pages 403–406, 2009. doi: 10.1109/ICICTA.2009.563.

[7] Michael Conrad. Private communication, December 2013.

[8] Charles D Edwards, A Barbieri, E Brower, P Estabrook, R Gibbs, R Horttor, J Ludwinski, R Mase, P Theisinger, and T Thorpe. *A Martian telecommunications network: UHF relay support of the Mars exploration rovers by the Mars global surveyor, Mars odyssey, and Mars express orbiters.* Pasadena, CA: Jet Propulsion Laboratory, National Aeronautics and Space Administration, 2004.

[9] M. Bajracharya, M.W. Maimone, and D. Helmick. Autonomy for Mars Rovers: Past, Present, and Future. *Computer*, 41(12):44 –50, December 2008. ISSN 0018-9162. doi: 10.1109/MC.2008.479.

[10] Ayanna M. Howard and Edward W. Tunstel. Intelligence For Space Robotics. *International Journal of Intelligent Computing in Medical Sciences & Image Processing*, 2(1):69–70, January 2008. ISSN 1931-308X, 2326-0068. doi:

10.1080/1931308X.2008.10644153. URL http://www.tandfonline.com/doi/abs/10.1080/1931308X.2008.10644153.

[11] O Khatib and J.-F. Le Maitre. Dynamic Control of Manipulators Operating in a Complex Environment. pages 267–282, Udine, Italy, 1978.

[12] Mircea Ivanescu. Artificial Potential Method for Control of a Tentacle Manipulator. In *8th IEEE International Conference on Intelligent Engineering Systems*, 2004.

[13] Shonal Singh, Bibhya Sharma, and Jito Vanualailai. Autonomous Control of a Mobile Manipulator. *World Academy of Science, Engineering and Technology*, (60):983–992, 2011.

[14] Babak Ranjbar, Javad Mahmoodi, Hasan Karbasi, Gholam Dashti, and Ali Omidvar. Robot Manipulator Path Planning Based on Intelligent Multi-resolution Potential Field. *International Journal of u-and e-Service, Science and Technology*, 8(1):11–26, 2015.

[15] Brian Lynch, Alex Ellery, and Fred Nitzsche. Two-Dimensional Robotic Vehicle Path Planning Based on Artificial Potential Fields. In *Proceedings of CSME Forum 2008*, Ottawa, March 2008.

[16] Cameron Frazier, Natalie Baddour, and Alex Ellery. RAVE-Matlab: Thesis Release. December 2014. doi: 10.5281/zenodo.13331. URL http://dx.doi.org/10.5281/zenodo.13331.

[17] E. Rimon and D.E. Koditschek. Exact robot navigation using artificial potential functions. *IEEE Transactions on Robotics and Automation*, 8(5):501–518, October 1992. ISSN 1042296X. doi: 10.1109/70.163777. URL `http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=163777`.

[18] F. Arambula Cosio and M . A. Padilla Castaneda. Autonomous robot navigation using adaptive potential fields. *Mathematical and Computer Modelling*, 40(9-10):1141–1156, 2004. ISSN 08957177. doi: 10.1016/j.mcm.2004.05.001. URL `http://resolver.scholarsportal.info/resolve/08957177/v40i9-10/1141_arnuapf.xml`.

[19] Bruno Siciliano and Oussama Khatib. *Springer Handbook of Robotics*. Springer, Berlin, 2008. ISBN 9783540239574 354023957X 9783540303015 3540303014 9783540382195 3540382194.

[20] S. S. Ge and Y. J. Cui. Dynamic Motion Planning for Mobile Robots Using Potential Field Method. *Autonomous Robots*, 13:207–222, 2002.

[21] Jing Ren, K.A. McIsaac, R.V. Patel, and T.M. Peters. A Potential Field Model Using Generalized Sigmoid Functions. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 37(2):477–484, 2007. ISSN 1083-4419. doi: 10.1109/TSMCB.2006.883866.

[22] K. P. Valavanis. Mobile Robot Navigation in 2-D Dynamic Environments Using an Electrostatic Potential Field. *Systems, Man and Cybernetics, Part A, IEEE Transactions on*, 30(2):187–196, March 2000. ISSN 10834427. URL `http://resolver.scholarsportal.info/resolve/10834427/v30i0002/187_mrni2deuaepf.xml`.

[23] P. Veelaert and W. Bogaerts. Ultrasonic potential field sensor for obstacle avoidance. *IEEE Transactions on Robotics and Automation*, 15(4):774–779, August 1999. ISSN 1042-296X. doi: 10.1109/70.782033.

[24] Y. Koren and J. Borenstein. Potential field methods and their inherent limitations for mobile robot navigation. In *Proc. IEEE Int. Conf. Robot. Autom.*, pages 1398–1404, Sacramento, California, USA, April 1991. IEEE Comput. Soc. Press. ISBN 0-8186-2163-X. doi: 10.1109/ROBOT.1991.131810. URL `http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=131810`.

[25] Jing Ren, K.A. McIsaac, and R.V. Patel. Modified Newton's method applied to potential field-based navigation for mobile robots. *IEEE Transactions on Robotics*, 22(2):384–391, 2006. ISSN 1552-3098. doi: 10.1109/TRO.2006.870668.

[26] R. Abiyev. Navigation of mobile robots in the presence of obstacles. *Advances in Engineering Software*, 41(10-11):1179–1186, October 2010. ISSN 09659978. URL `http://resolver.scholarsportal.info/resolve/09659978/v41i10-11/1179_nomritpoo.xml`.

[27] C.W. Warren. Global path planning using artificial potential fields. In *, 1989 IEEE International Conference on Robotics and Automation, 1989. Proceedings*, pages 316–321 vol.1, May 1989. doi: 10.1109/ROBOT.1989.100007.

[28] J. Barraquand, B. Langlois, and J.-C. Latombe. Numerical potential field techniques for robot path planning. pages 1012–1017 vol.2. IEEE, 1991. ISBN 0-7803-0078-5. doi: 10.1109/ICAR.1991.240539. URL `http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=240539`.

[29] S.S. Ge and Y.J. Cui. New Potential Functions for Mobile Robot Path Planning. *IEEE Transactions on Robotics and Automation*, 16(5):615–620, October 2000. ISSN 1042296X. doi: 10.1109/70.880813. URL `http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=880813`.

[30] J. Borenstein and Y. Koren. Real-time obstacle avoidance for fast mobile robots. *IEEE Transactions on Systems, Man, and Cybernetics*, 19(5):1179–1187, September 1989. ISSN 0018-9472. doi: 10.1109/21.44033. URL `http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=44033`.

[31] Min Gyu Park, Jae Hyun Jeon, and Min Cheol Lee. Obstacle avoidance for mobile robots using artificial potential field approach with simulated annealing. In *IEEE International Symposium on Industrial Electronics, 2001. Proceedings. ISIE 2001*, volume 3, pages 1530–1535 vol.3, 2001. doi: 10.1109/ISIE.2001.931933.

[32] Thomas Weise. *Global Optimization Algorithms Theory and Application.pdf*. 2nd edition, June 2009. URL `http://www.it-weise.de/`.

[33] Steven M. Lavalle. Rapidly-Exploring Random Trees: A New Tool for Path Planning. Technical report, 1998.

[34] Steven Michael LaValle. *Planning algorithms*. Cambridge University Press, Cambridge ; New York, 2006. ISBN 0521862051.

[35] Alex Nash, Kenny Daniel, Sven Koenig, and Ariel Felner. Theta*: Any-Angle Path Planning on Grids. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 1177–1183, Vancouver, Canada, July 2007. AAAI. ISBN ISBN 978-1-57735-323-2. URL `http://www.aaai.org/Papers/AAAI/2007/AAAI07-187.pdf`.

[36] Anthony Stentz. Optimal and Efficient Path Planning for Unknown and Dynamic Environments. Technical report, August 1993.

[37] E. W. Dijkstra. A note on two problems in connexion with graphs. *Numerische Mathematik*, 1(1):269–271, December 1959. ISSN 0029-599X, 0945-3245. doi: 10.1007/BF01386390. URL `http://link.springer.com/10.1007/BF01386390`.

[38] P.E. Hart, N.J. Nilsson, and B. Raphael. A Formal Basis for the Heuristic Determination of Minimum Cost Paths. *IEEE Transactions on Systems Science and Cybernetics*, 4(2):100–107, July 1968. ISSN 0536-1567. doi: 10.1109/TSSC.1968.300136.

[39] Peter E. Hart, Nils J. Nilsson, and Bertram Raphael. Correction to "A Formal Basis for the Heuristic Determination of Minimum Cost Paths". *SIGART Bull.*, (37):28–29, December 1972. ISSN 0163-5719. doi: 10.1145/1056777.1056779. URL `http://doi.acm.org/10.1145/1056777.1056779`.

[40] A. Stentz. Optimal and efficient path planning for partially-known environments. In *, 1994 IEEE International Conference on Robotics and Automation, 1994. Proceedings*, pages 3310–3317 vol.4, May 1994. doi: 10.1109/ROBOT.1994.351061.

[41] Anthony Stentz. The focussed D* algorithm for real-time replanning. *IJCAI*, 95:1652–1659, 1995.

[42] R.A. Rutenbar. Simulated annealing algorithms: an overview. *IEEE Circuits and Devices Magazine*, 5(1):19–26, January 1989. ISSN 8755-3996. doi: 10.1109/101.17235.

[43] M. Fleischer. Simulated annealing: past, present, and future. In *Simulation Conference Proceedings, 1995. Winter*, pages 155–161, December 1995. doi: 10.1109/WSC.1995.478718.

[44] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi. Optimization by Simulated Annealing. *Science*, 220(4598):671–680, May 1983. ISSN 0036-8075, 1095-9203. doi: 10.1126/science.220.4598.671. URL `http://www.sciencemag.org/content/220/4598/671`. PMID: 17813860.

[45] Jillian Beardwood, J. H. Halton, and J. M. Hammersley. The shortest path through many points. *Mathematical Proceedings of the Cambridge Philosophical Society*, 55(04):299, October 1959. ISSN 0305-0041, 1469-8064. doi: 10.1017/S0305004100034095. URL http://www.journals.cambridge.org/abstract_S0305004100034095.

[46] Ronald C. Arkin. Motor Schema - Based Mobile Robot Navigation. *The International Journal of Robotics Research*, 8(4):92–112, August 1989. ISSN 0278-3649, 1741-3176. doi: 10.1177/027836498900800406. URL http://ijr.sagepub.com.proxy.bib.uottawa.ca/content/8/4/92.

[47] Wesley H. Huang, Brett R. Fajen, Jonathan R. Fink, and William H. Warren. Visual navigation and obstacle avoidance using a steering potential function. *Robotics and Autonomous Systems*, 54(4):288–299, April 2006. ISSN 0921-8890. doi: 10.1016/j.robot.2005.11.004. URL http://www.sciencedirect.com/science/article/pii/S0921889005001995.

[48] O. Khatib. Real-Time Obstacle Avoidance for Manipulators and Mobile Robots. volume 2, pages 500–505. Institute of Electrical and Electronics Engineers, March 1985. doi: 10.1109/ROBOT.1985.1087247. URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1087247.

[49] Oussama Khatib, John J Craig, and Tomaś Lozano-Peŕez. *The Robotics review 1*. MIT Press, Cambridge, Mass., 1989. ISBN 0262111357 9780262111355.

[50] Bruce Krogh. A Generalized Potential Field Approach to Obstacle Avoidance Control. In *SME Conference Proceedings*, Bethleham, PA, USA, August 1984.

[51] Neville Hogan. Impedance Control: An Approach to Manipulation. pages 304 – 313, San Diego, CA, USA, June 1984.

[52] N. Hogan. Stable execution of contact tasks using impedance control. volume 4, pages 1047–1054. Institute of Electrical and Electronics Engineers, 1987. doi: 10.1109/ROBOT.1987.1087854. URL `http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1087854`.

[53] R.B. Tilove. Local obstacle avoidance for mobile robots based on the method of artificial potentials. pages 566–571. IEEE Comput. Soc. Press, 1990. ISBN 0-8186-9061-5. doi: 10.1109/ROBOT.1990.126041. URL `http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=126041`.

[54] Lu Yin and Yixin Yin. An improved potential field method for mobile robot path planning in dynamic environments. pages 4847–4852. IEEE, 2008. ISBN 978-1-4244-2113-8. doi: 10.1109/WCICA.2008.4593709. URL `http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4593709`.

[55] P. Khosla and R. Volpe. Superquadric artificial potentials for obstacle avoidance and approach. In *, 1988 IEEE International Conference on Robotics and Automation, 1988. Proceedings*, pages 1778–1784 vol.3, April 1988. doi: 10.1109/ROBOT.1988.12323.

[56] I. Kweon, Y. Kuno, M. Watanabe, and K. Onoguchi. Behavior-based mobile robot using active sensor fusion. In *, 1992 IEEE International Conference on Robotics and Automation, 1992. Proceedings*, pages 1675–1682 vol.2, May 1992. doi: 10.1109/ROBOT.1992.220137.

[57] In So Kweon, Y. Kuno, M. Watanabe, and K. Onoguchi. Behavior-based Intelligent Robot In Dynamic Indoor Environments. In *, Proceedings of the 1992 lEEE/RSJ International Conference on Intelligent Robots and Systems, 1992*, volume 2, pages 1339–1346, July 1992. doi: 10.1109/IROS.1992.594559.

[58] J. Kim and P.K. Khosla. Real-time obstacle avoidance using harmonic potential functions. *IEEE Transactions on Robotics and Automation*, 8(3):338–349, June 1992. ISSN 1042-296X. doi: 10.1109/70.143352.

[59] R. Volpe and P. Khosla. Artificial potentials with elliptical isopotential contours for obstacle avoidance. In *26th IEEE Conference on Decision and Control, 1987*, volume 26, pages 180–185, December 1987. doi: 10.1109/CDC.1987.272738.

[60] A. Chella, S. Gaglio, and R. Pirrone. Conceptual representations of actions for autonomous robots. *Robotics and Autonomous Systems*, 34(4):251–263, March 2001. ISSN 0921-8890. doi: 10.1016/S0921-8890(00)00121-4. URL `http://www.sciencedirect.com/science/article/pii/S0921889000001214`.

[61] H. Haddad, M. Khatib, S. Lacroix, and R. Chatila. Reactive navigation in outdoor environments using potential fields. In *1998 IEEE International Conference on Robotics and Automation, 1998. Proceedings*, volume 2, pages 1232–1237 vol.2, May 1998. doi: 10.1109/ROBOT.1998.677268.

[62] Jerome Barraquand, Bruno Langlois, and Jean-Claude Latombe. Robot Motion Planning with Many Degrees of Freedom and Dynamic Constraints. In *The Fifth International Symposium on Robotics Research*, Artificial intelligence, pages 435–444, Cambridge, Mass, 1990. MIT Press. ISBN 0-262-13253-2. URL `http://dl.acm.org/citation.cfm?id=112687.112734`.

[63] S. Shimoda, Y. Kuroda, and K. Iagnemma. Potential Field Navigation of High Speed Unmanned Ground Vehicles on Uneven Terrain. pages 2828–2833. IEEE, 2007. ISBN 0-7803-8914-X. doi: 10.1109/ROBOT.2005.1570542. URL `http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1570542`.

[64] Clearpath Robotics. Husky A200 UGV User Manual, 2011.

[65] Mori. Scanning Laser Range Finder URG-04LX-UG01 (Simple-URG): Specifications, August 2009. URL `http://www.hokuyo-aut.jp/02sensor/07scanner/download/products/urg-04lx-ug01/data/URG-04LX_UG01_spec_en.pdf`.

[66] Makoto Matsumoto and Takuji Nishimura. Mersenne Twister: A 623-dimensionally Equidistributed Uniform Pseudo-random Number Generator. *ACM Trans. Model. Comput. Simul.*, 8(1):3–30, January 1998. ISSN 1049-3301.

doi: 10.1145/272991.272995. URL http://doi.acm.org/10.1145/272991.272995.

[67] Cameron Frazier. SEEG - Potential Field Obstacle Avoidance - 2013-04-04, April 2013. URL http://www.youtube.com/watch?v=z8jz4OrNV80.

[68] Cameron Frazier. SEEG - Potential Fields - Waypoints - 2013-06-10, November 2013. URL http://www.youtube.com/watch?v=tpgmaVgcz9Y.

[69] Microstrain Sensing Systems. Lord Product Datasheet: 3DM-GX3-25, 2013.

[70] Open Source Robotics Foundation. ROS.org — Powering the world's robots, July 2014. URL http://www.ros.org/.

Appendix A

Full Simulation Results

A.1 Test Series 1

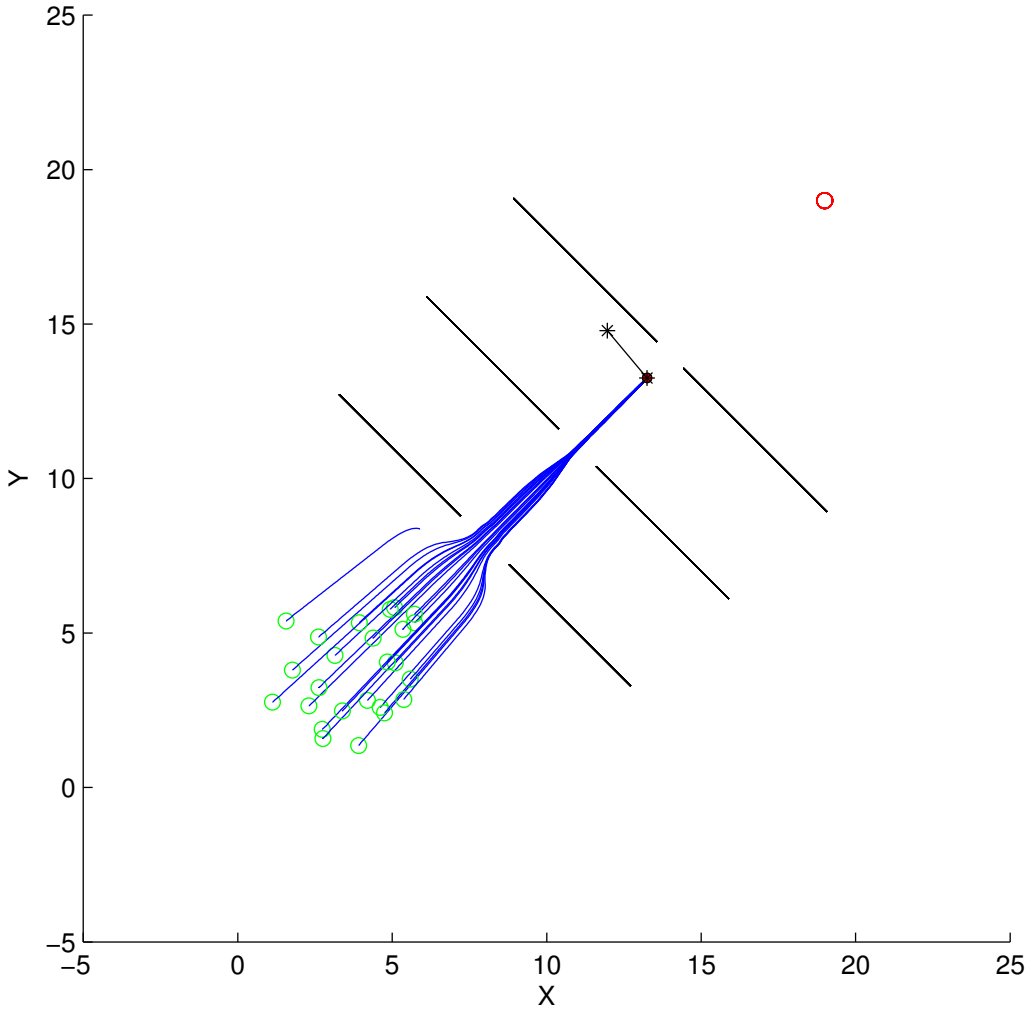


Figure A.1: Test Series 1 - Door Frame Map. 1 PSP used. Only attractive and repulsive forces used. Success: 0 of 25

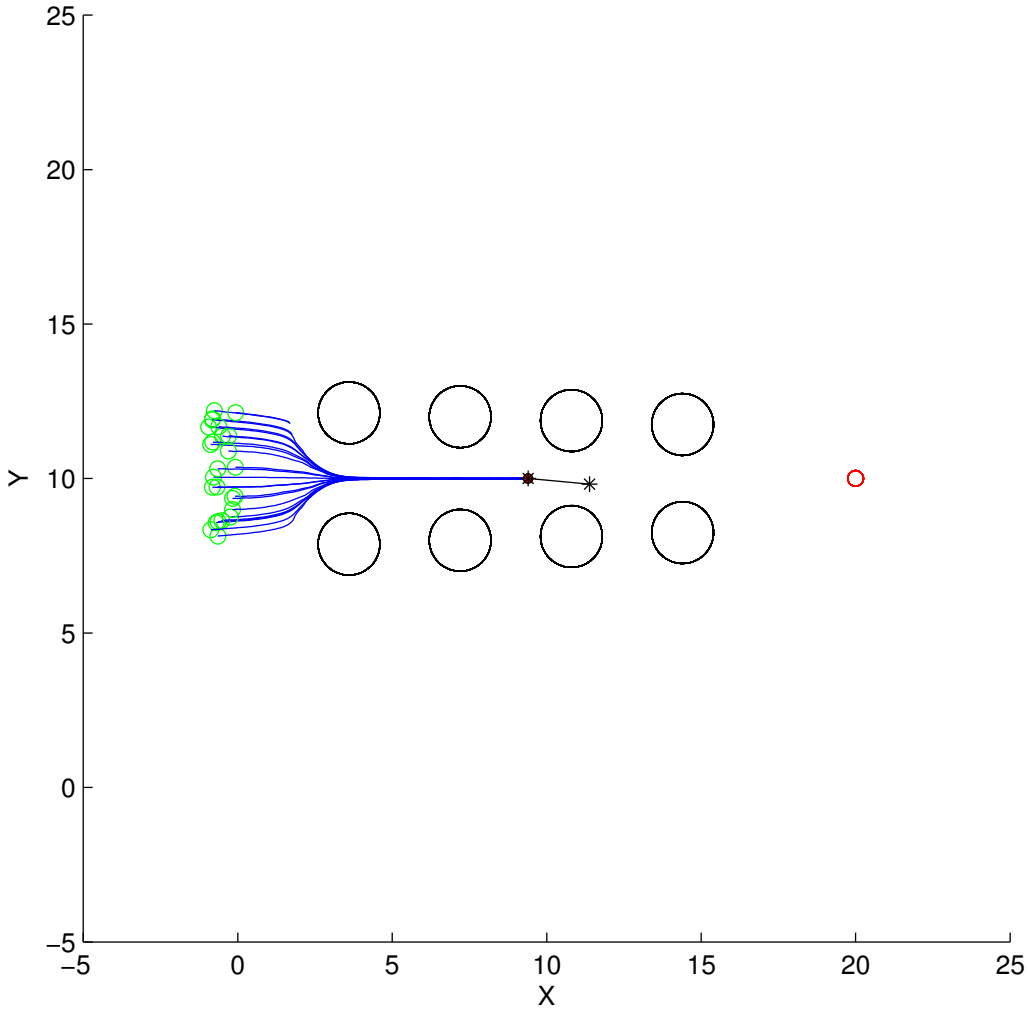


Figure A.2: Test Series 1 - Obstacle Gap Map. 1 PSP used. Only attractive and repulsive forces used. Success: 0 of 25

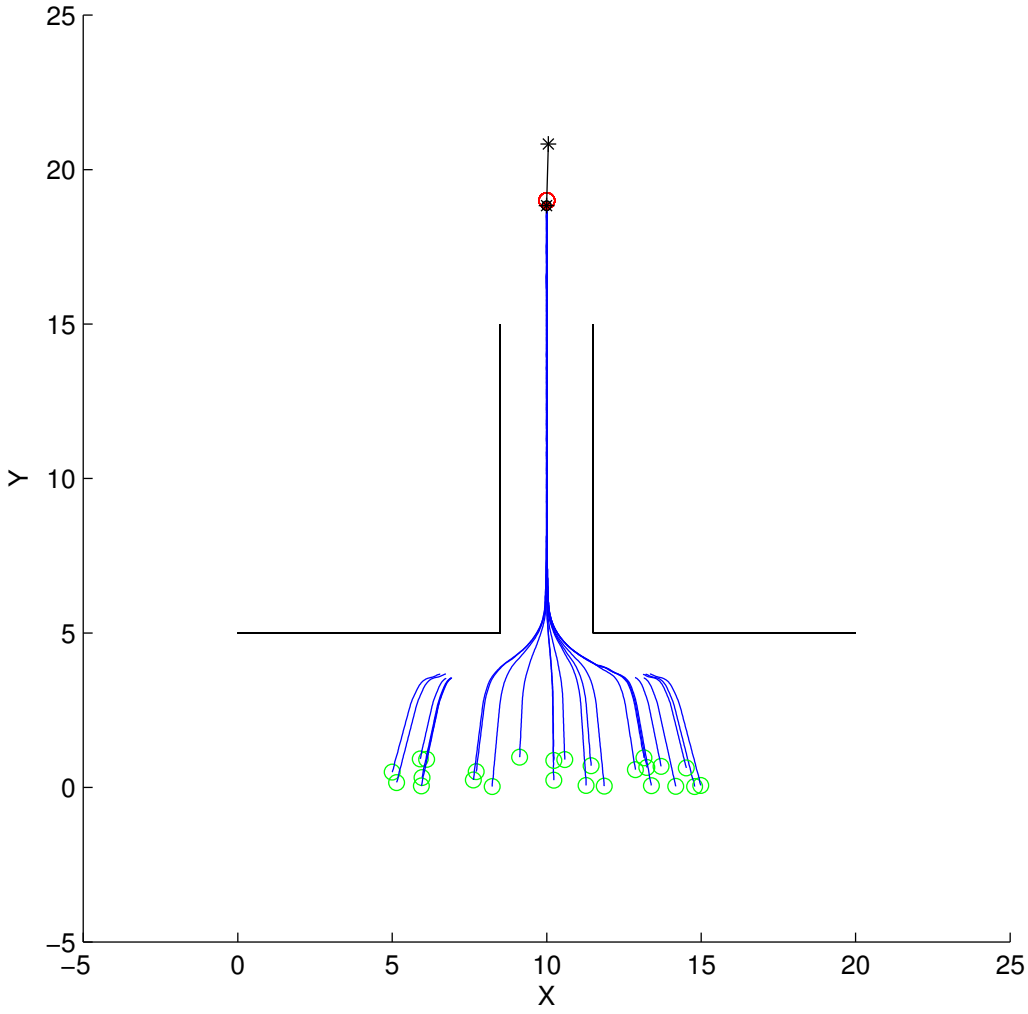


Figure A.3: Test Series 1 - Hallway (Narrow) Map. 1 PSP used. Only attractive and repulsive forces used. Success: 14 of 25

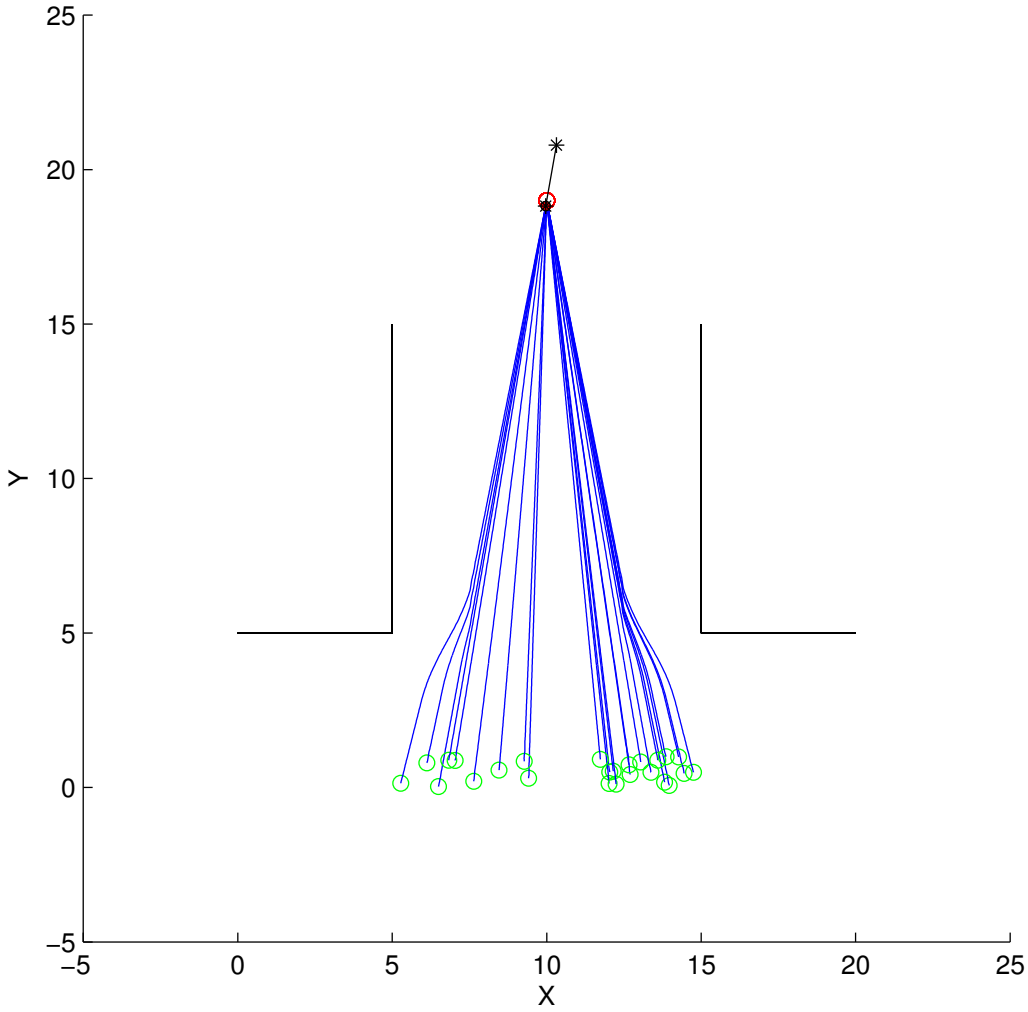


Figure A.4: Test Series 1 - Hallway (Wide) Map. 1 PSP used. Only attractive and repulsive forces used. Success: 25 of 25

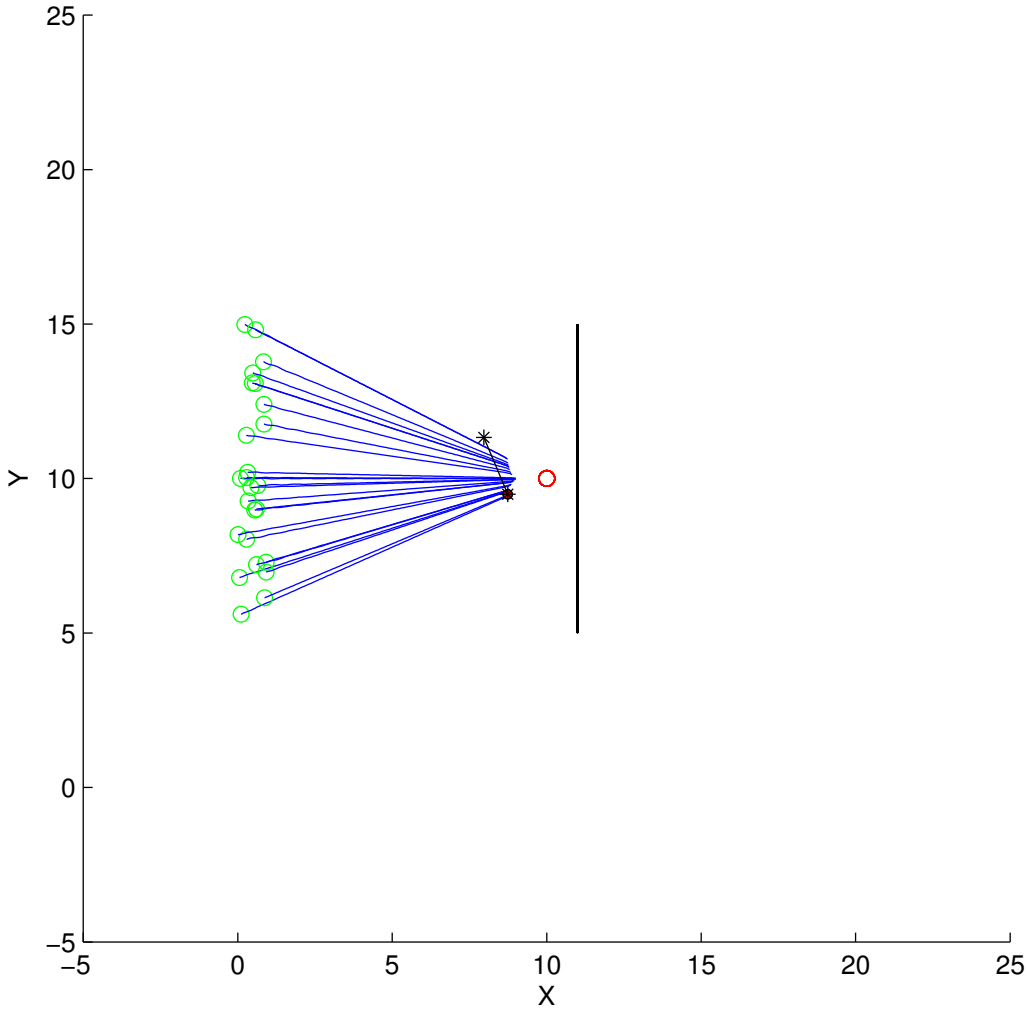


Figure A.5: Test Series 1 - GNRON Map. 1 PSP used. Only attractive and repulsive forces used. Success: 0 of 25

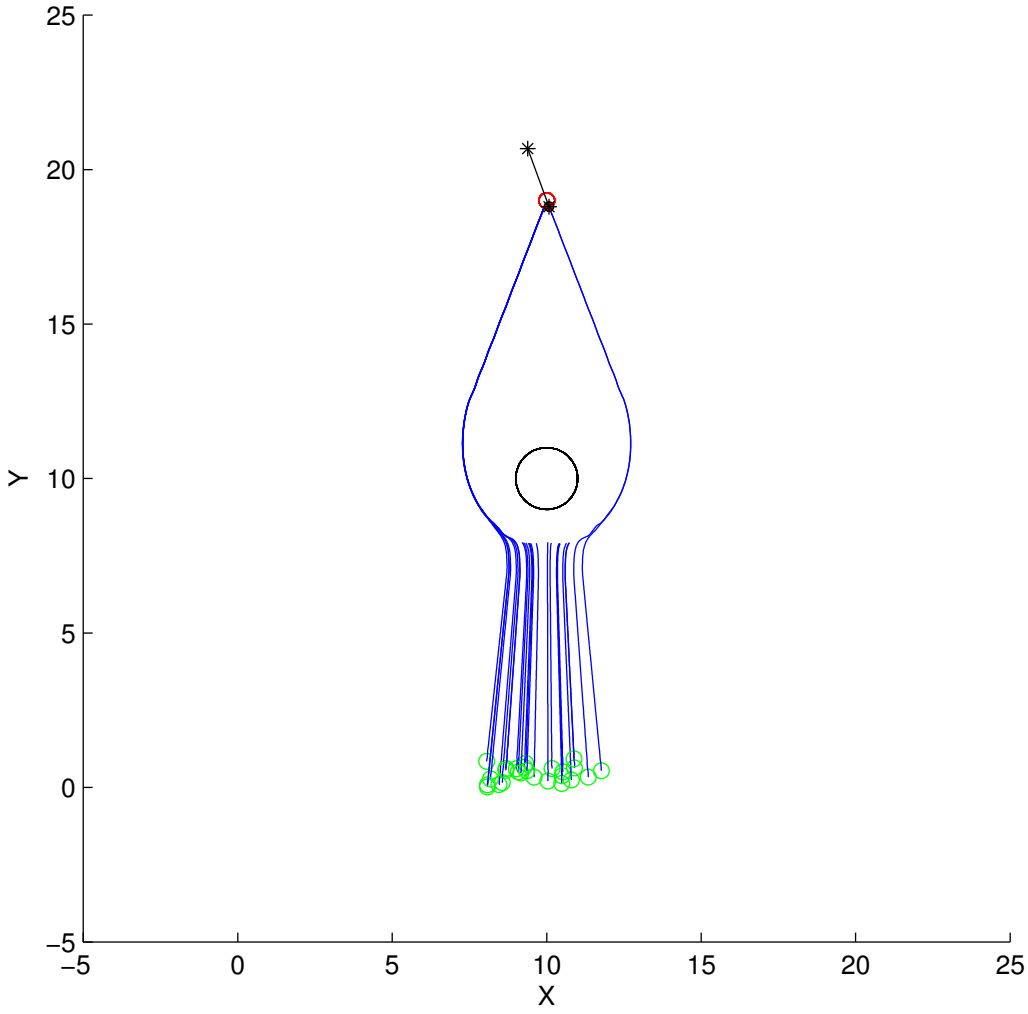


Figure A.6: Test Series 1 - Single Obstacle Map. 1 PSP used. Only attractive and repulsive forces used. Success: 10 of 25

A.2 Test Series 2

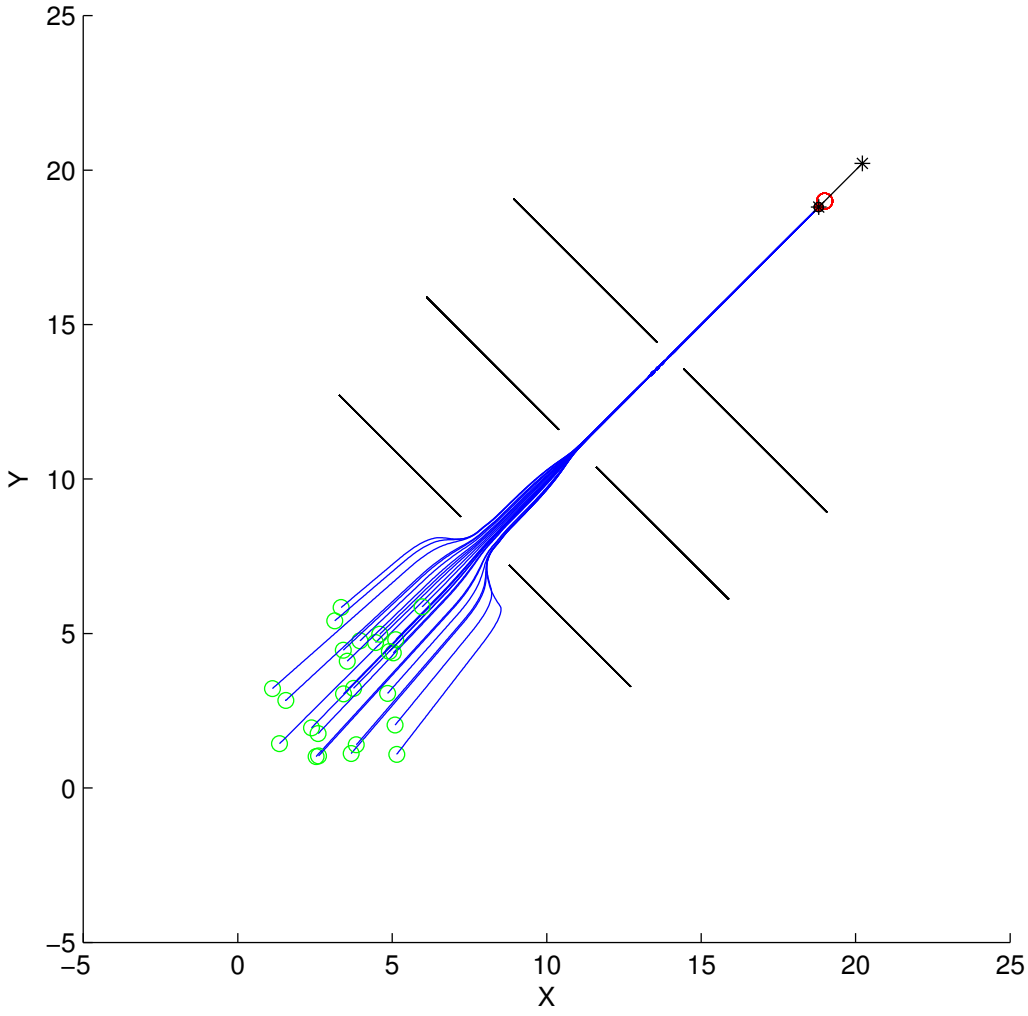


Figure A.7: Test Series 2 - Door Frame Map. 1 PSP used. All forces used. Success: 25 of 25

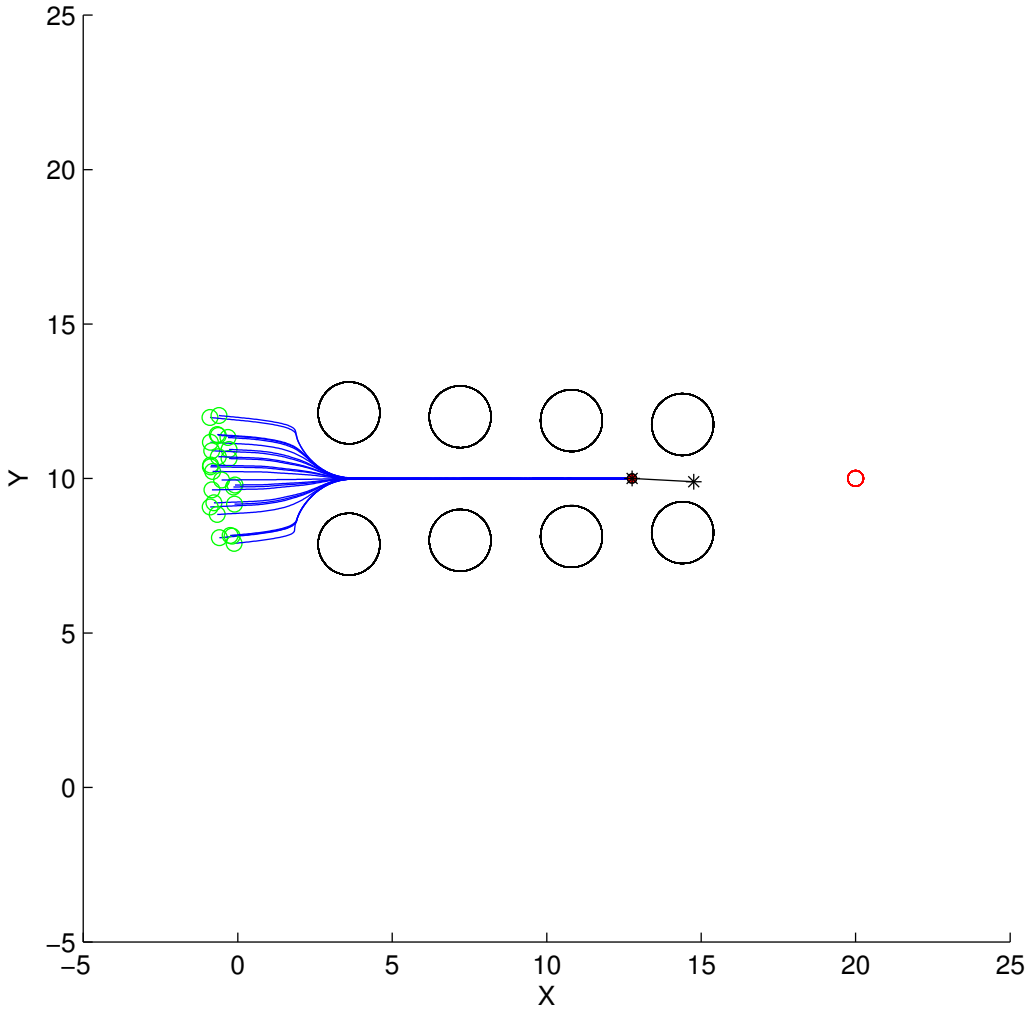


Figure A.8: Test Series 2 - Obstacle Gap Map. 1 PSP used. All forces used. Success: 0 of 25

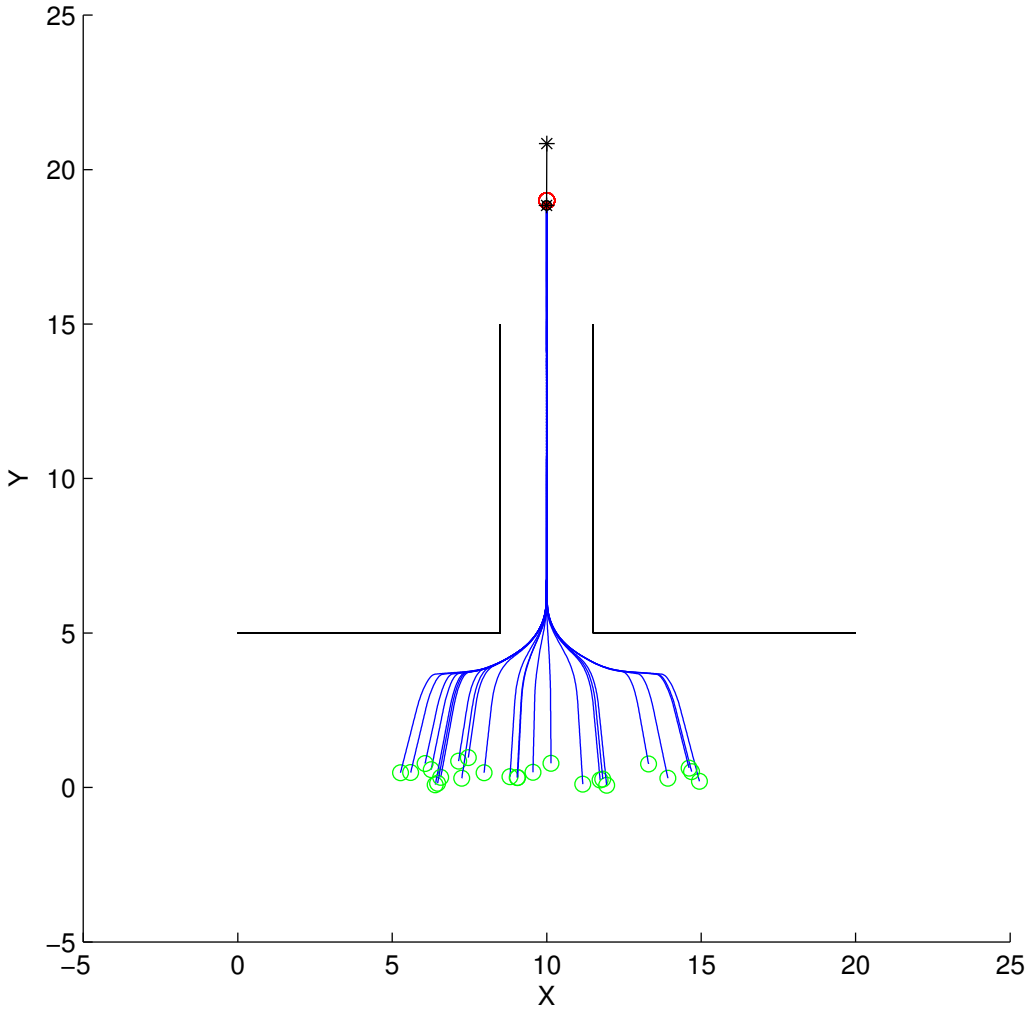


Figure A.9: Test Series 2 - Hallway (Narrow) Map. 1 PSP used. All forces used. Success: 25 of 25

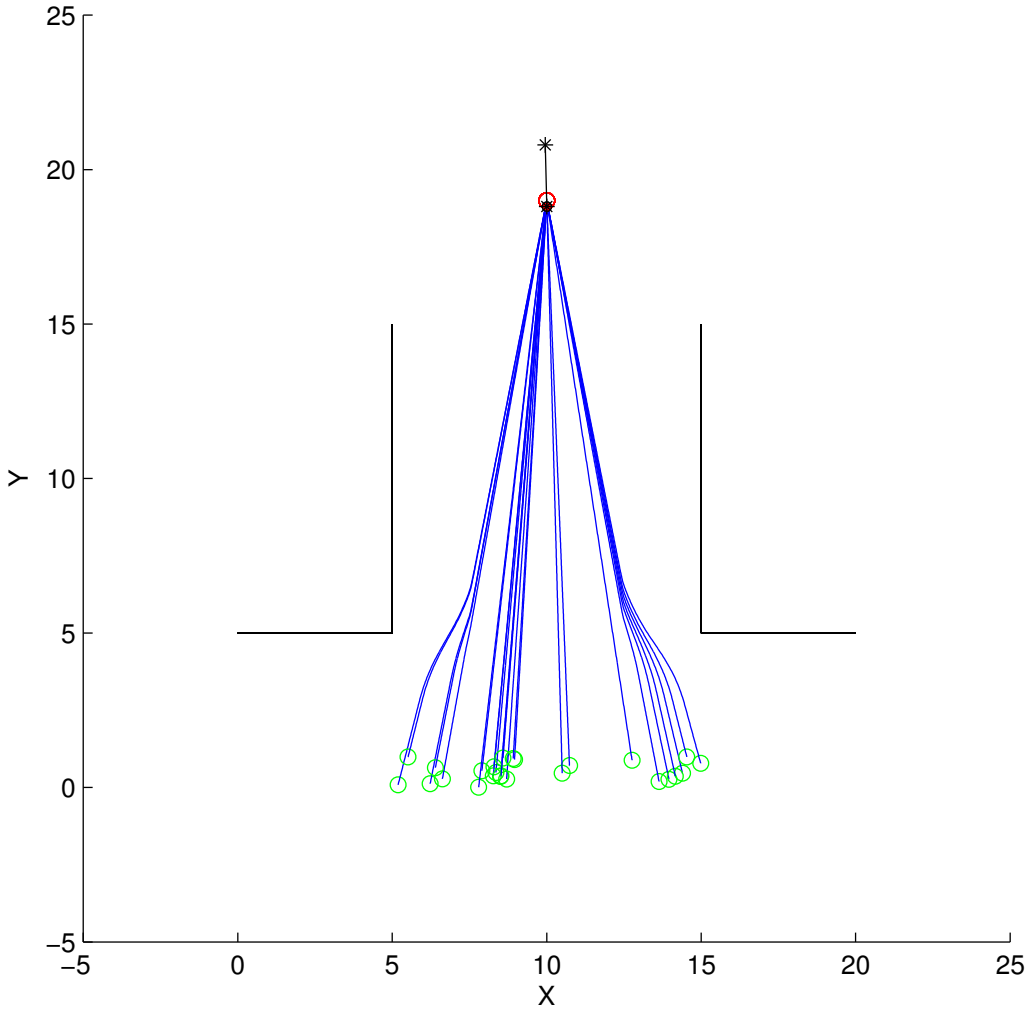


Figure A.10: Test Series 2 - Hallway (Wide) Map. 1 PSP used. All forces used. Success: 25 of 25

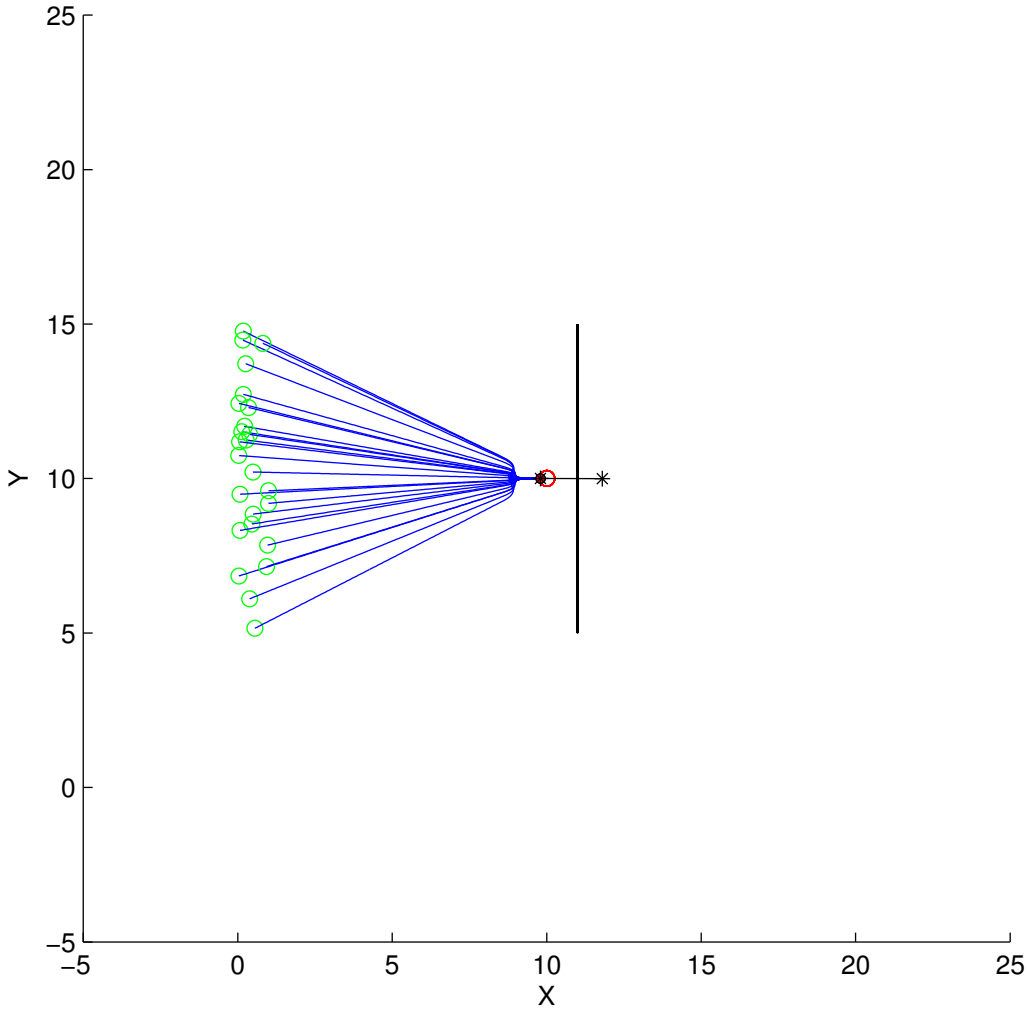


Figure A.11: Test Series 2 - GNRON Map. 1 PSP used. All forces used. Success: 25 of 25

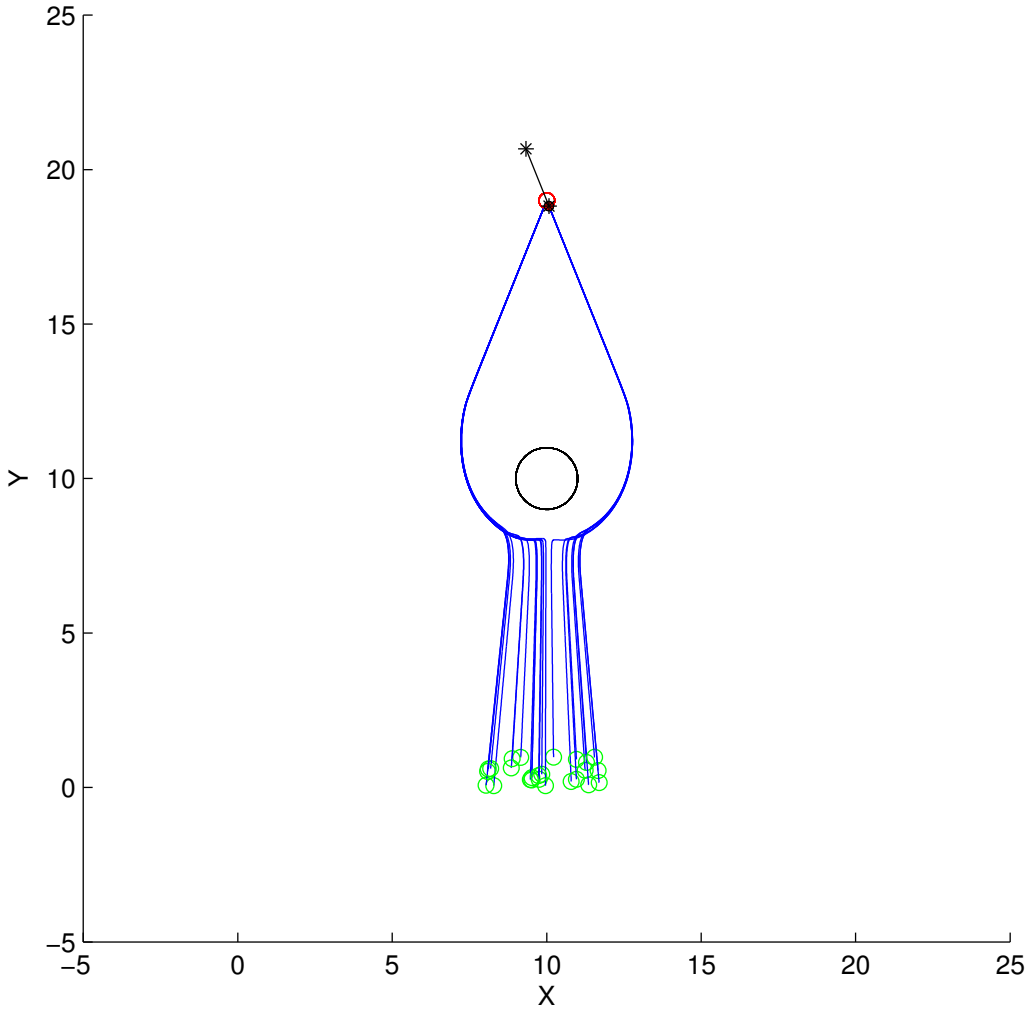


Figure A.12: Test Series 2 - Single Obstacle Map. 1 PSP used. All forces used. Success: 25 of 25

A.3 Test Series 3

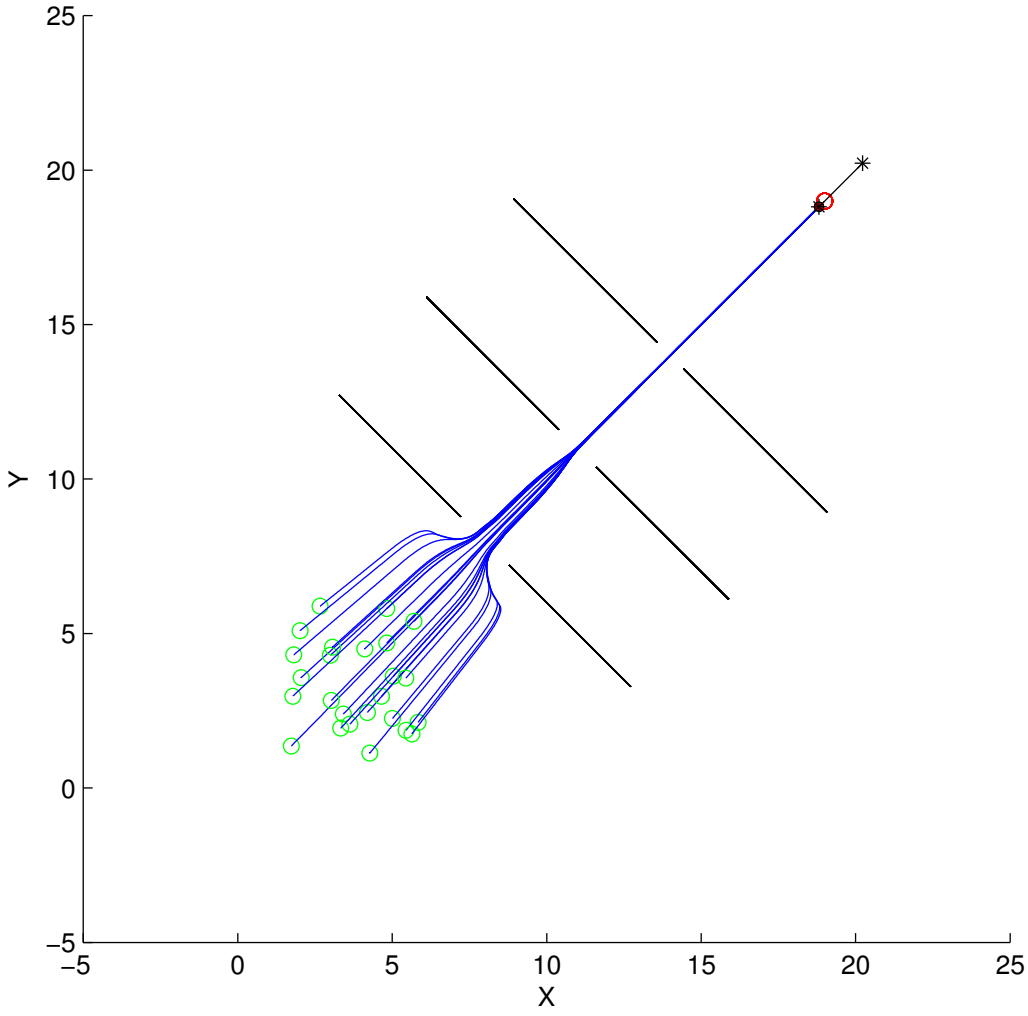


Figure A.13: Test Series 3 - Door Frame Map. 1 PSP used. All forces used save for tangential forces. Success: 25 of 25

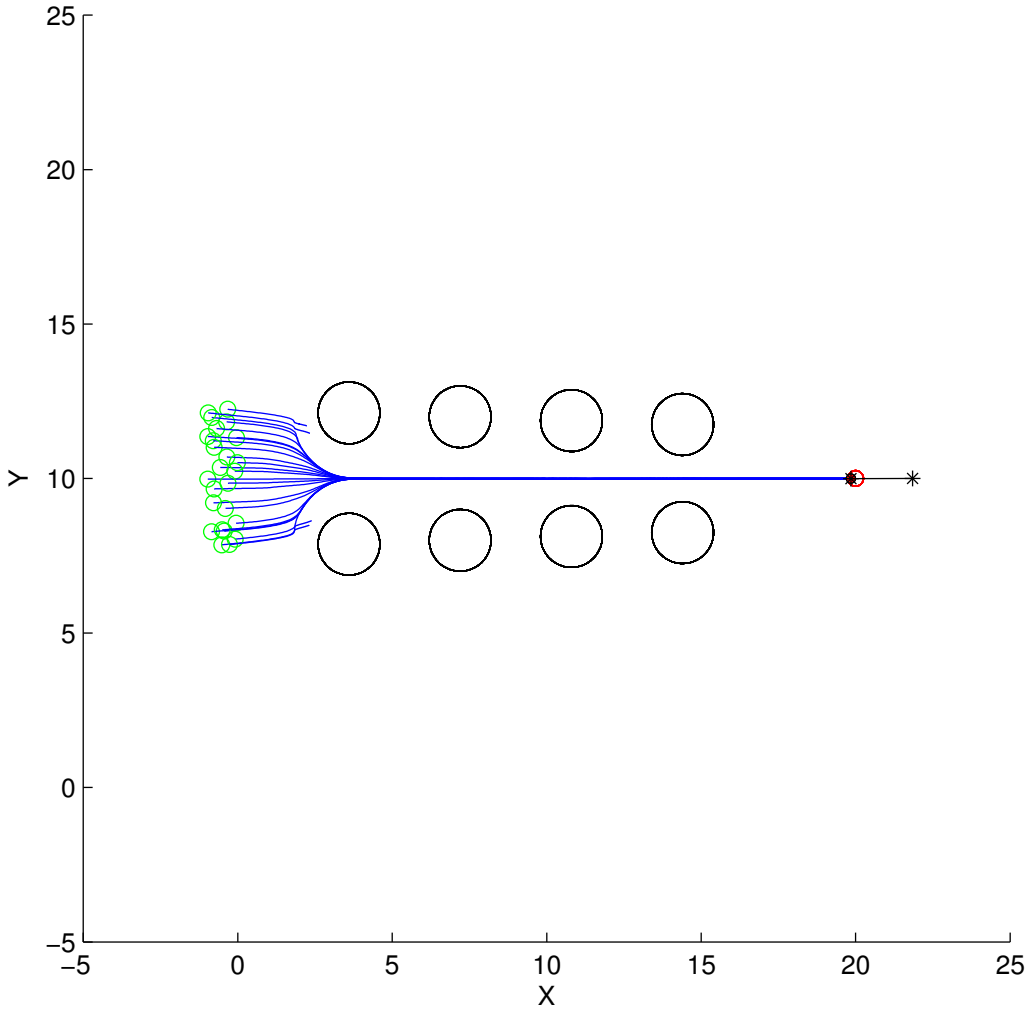


Figure A.14: Test Series 3 - Obstacle Gap Map. 1 PSP used. All forces used save for tangential forces. Success: 21 of 25

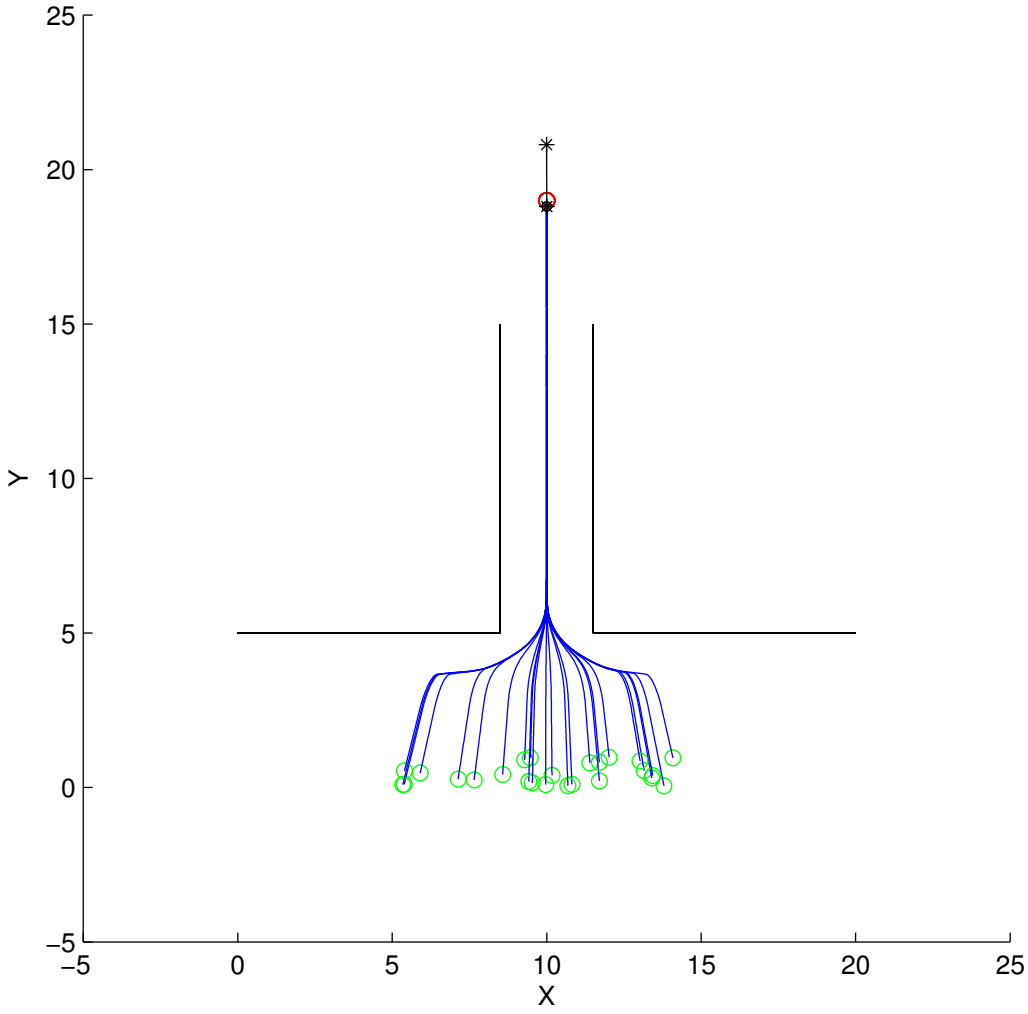


Figure A.15: Test Series 3 - Hallway (Narrow) Map. 1 PSP used. All forces used save for tangential forces. Success: 25 of 25

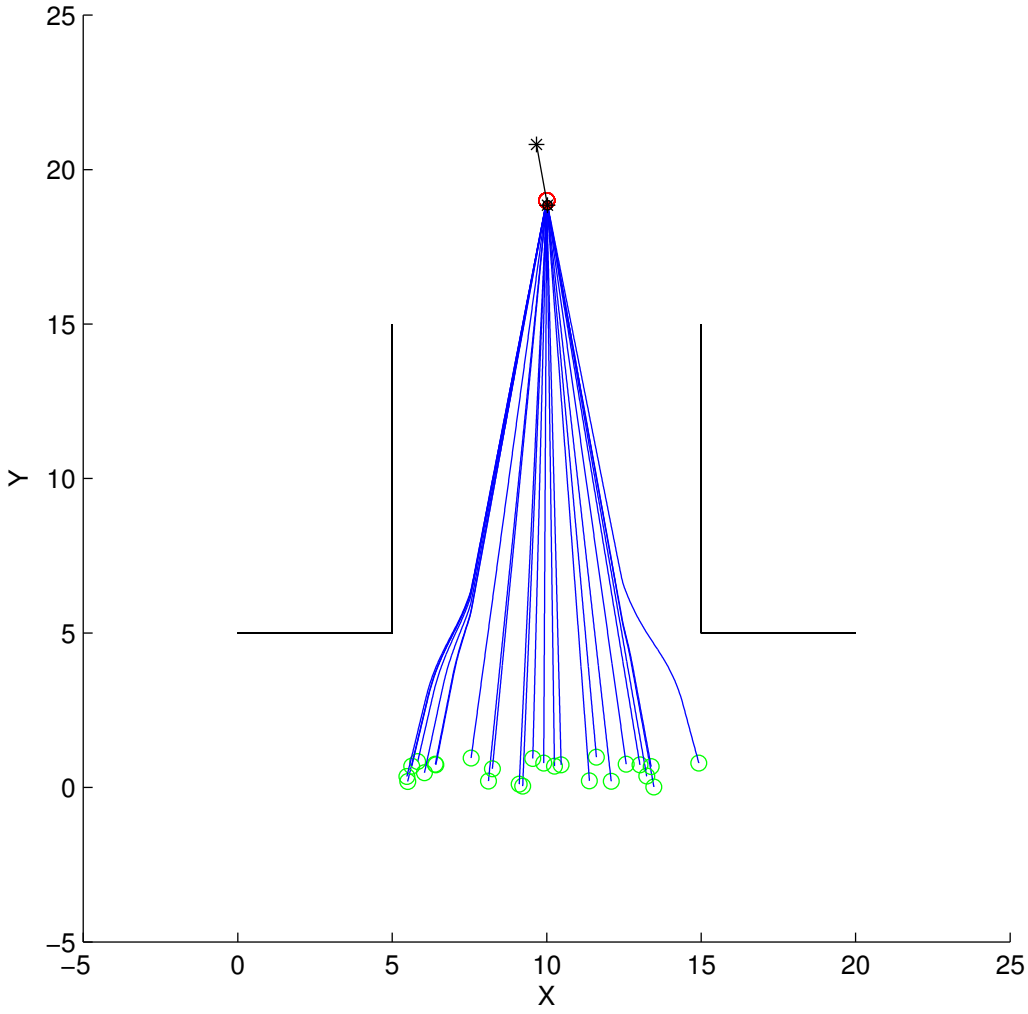


Figure A.16: Test Series 3 - Hallway (Wide) Map. 1 PSP used. All forces used save for tangential forces. Success: 25 of 25

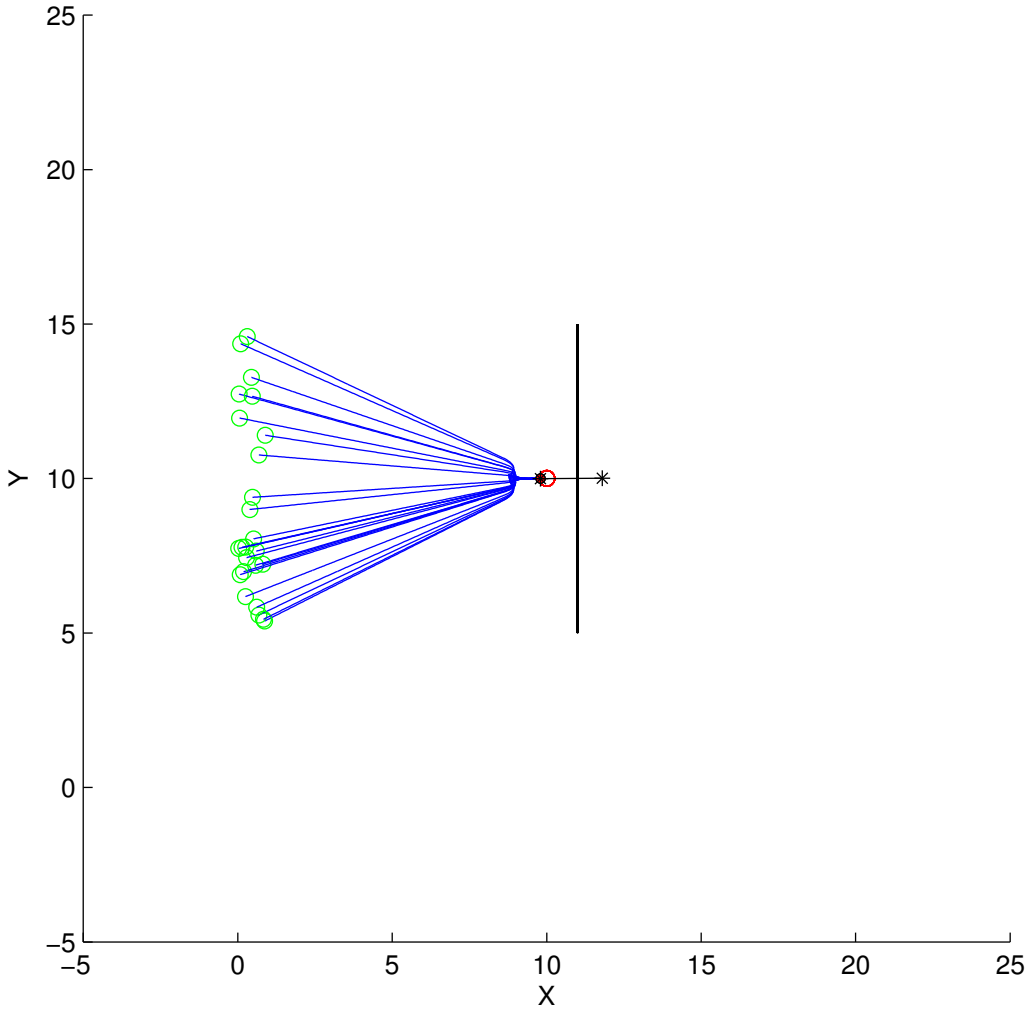


Figure A.17: Test Series 3 - GNRON Map. 1 PSP used. All forces used save for tangential forces. Success: 25 of 25

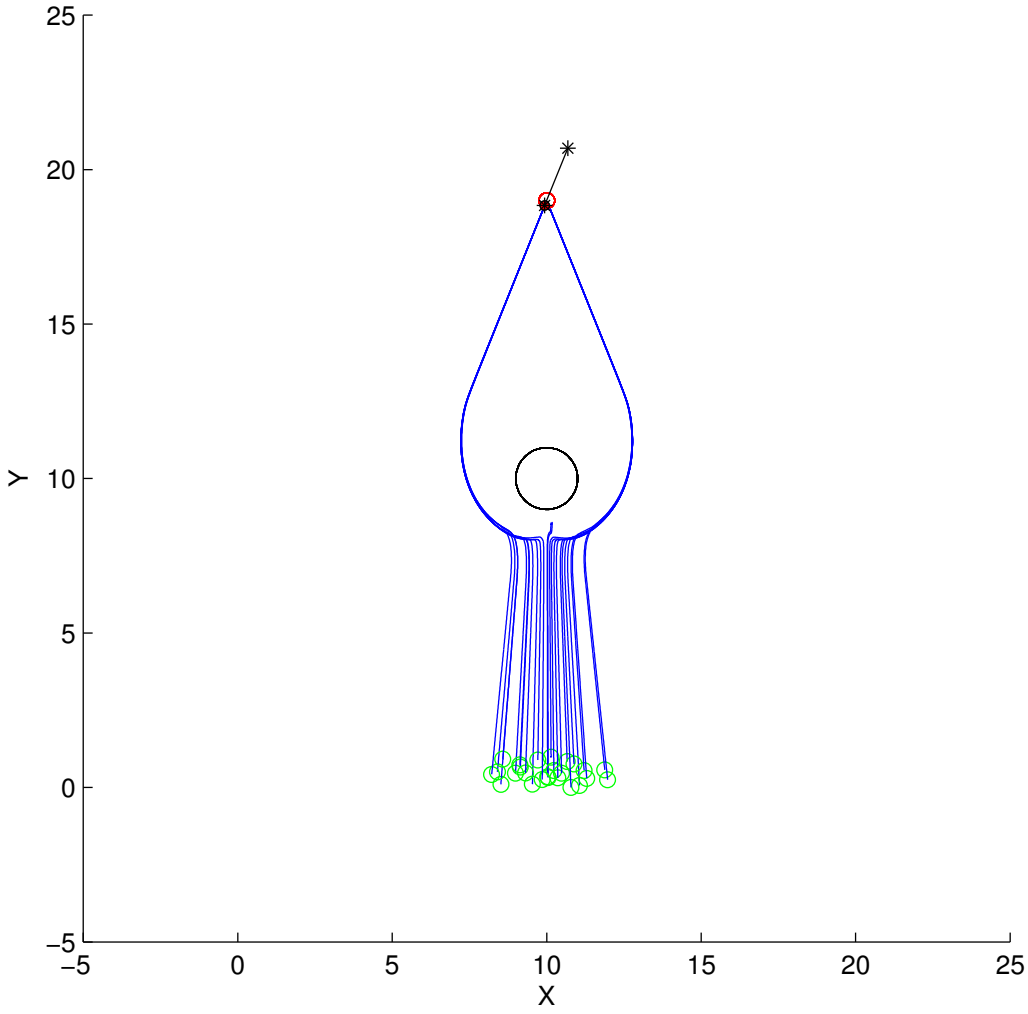


Figure A.18: Test Series 3 - Single Obstacle Map. 1 PSP used. All forces used save for tangential forces. Success: 23 of 25

A.4 Test Series 4

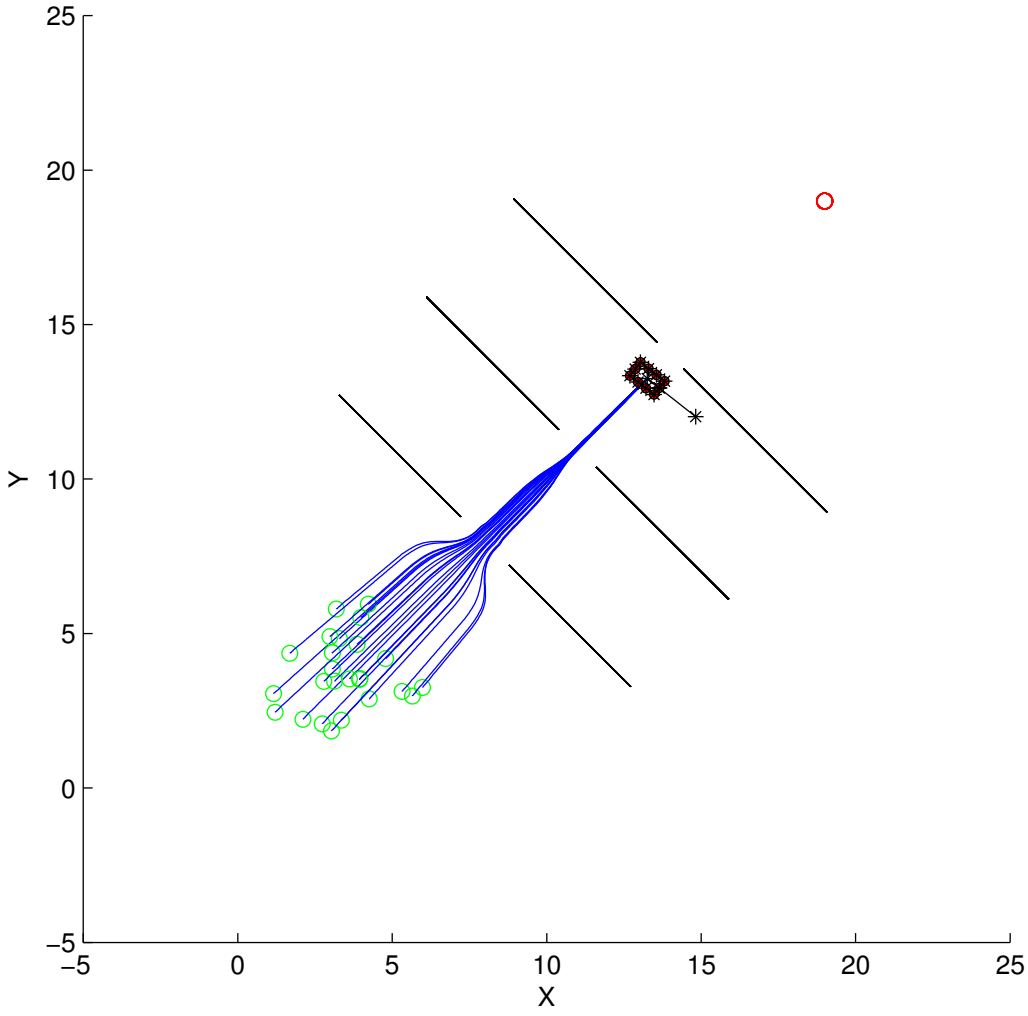


Figure A.19: Test Series 4 - Door Frame Map. 10 PSP used. Only attractive and repulsive forces used. Success: 0 of 25

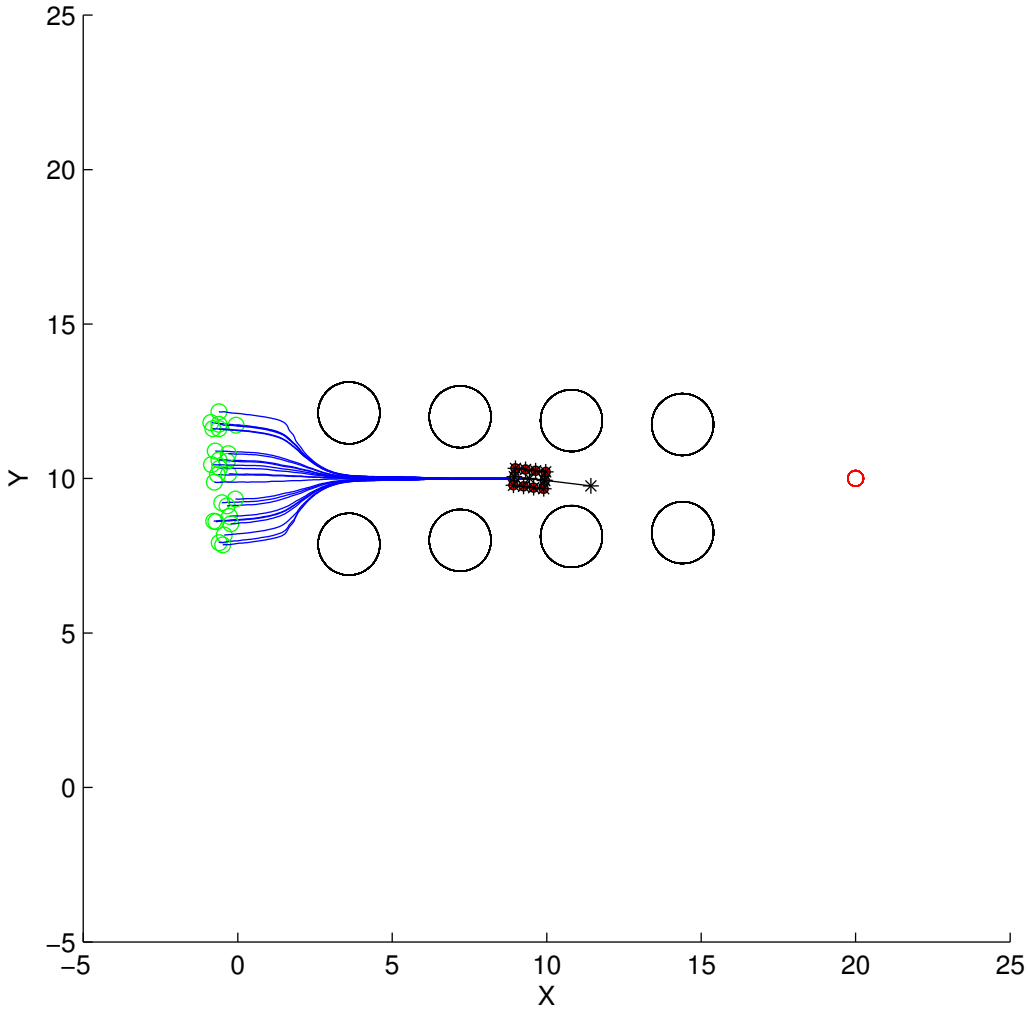


Figure A.20: Test Series 4 - Obstacle Gap Map. 10 PSP used. Only attractive and repulsive forces used. Success: 0 of 25

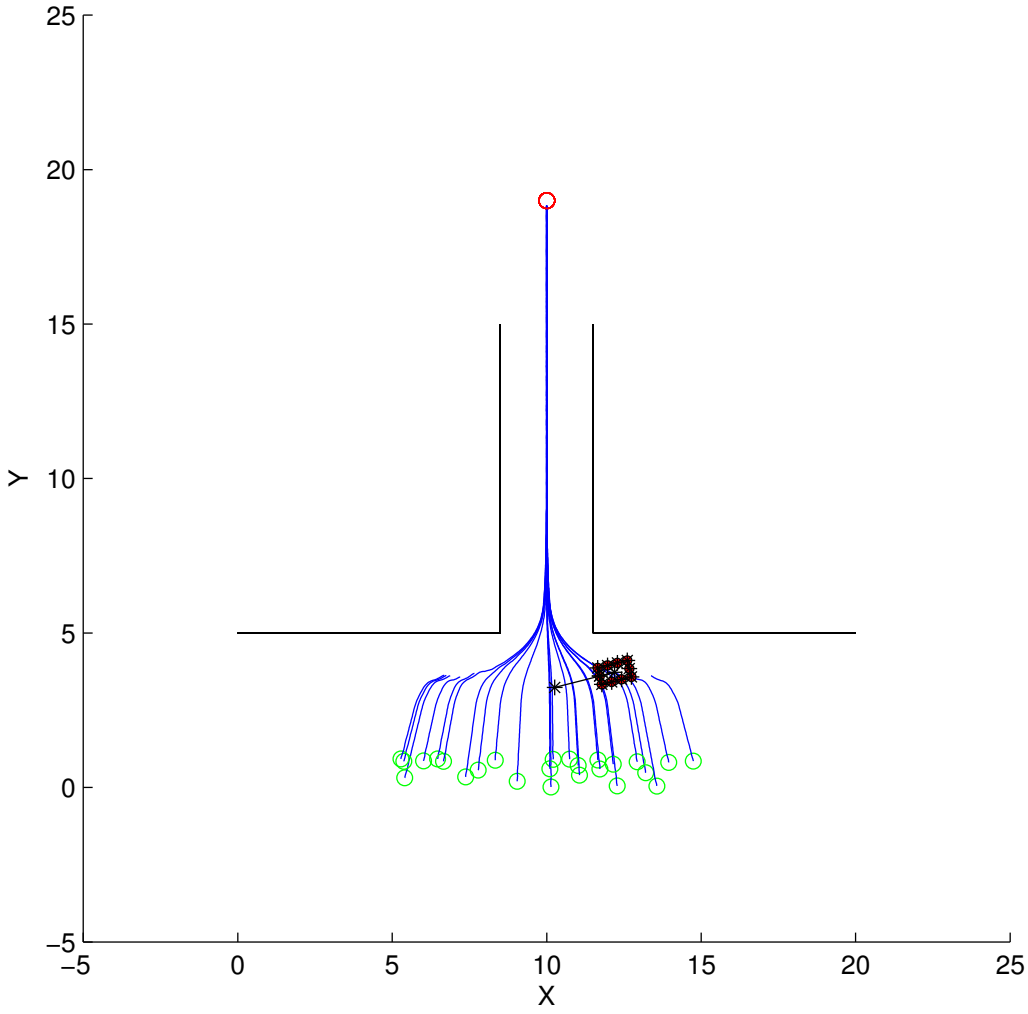


Figure A.21: Test Series 4 - Hallway (Narrow) Map. 10 PSP used. Only attractive and repulsive forces used. Success: 17 of 25

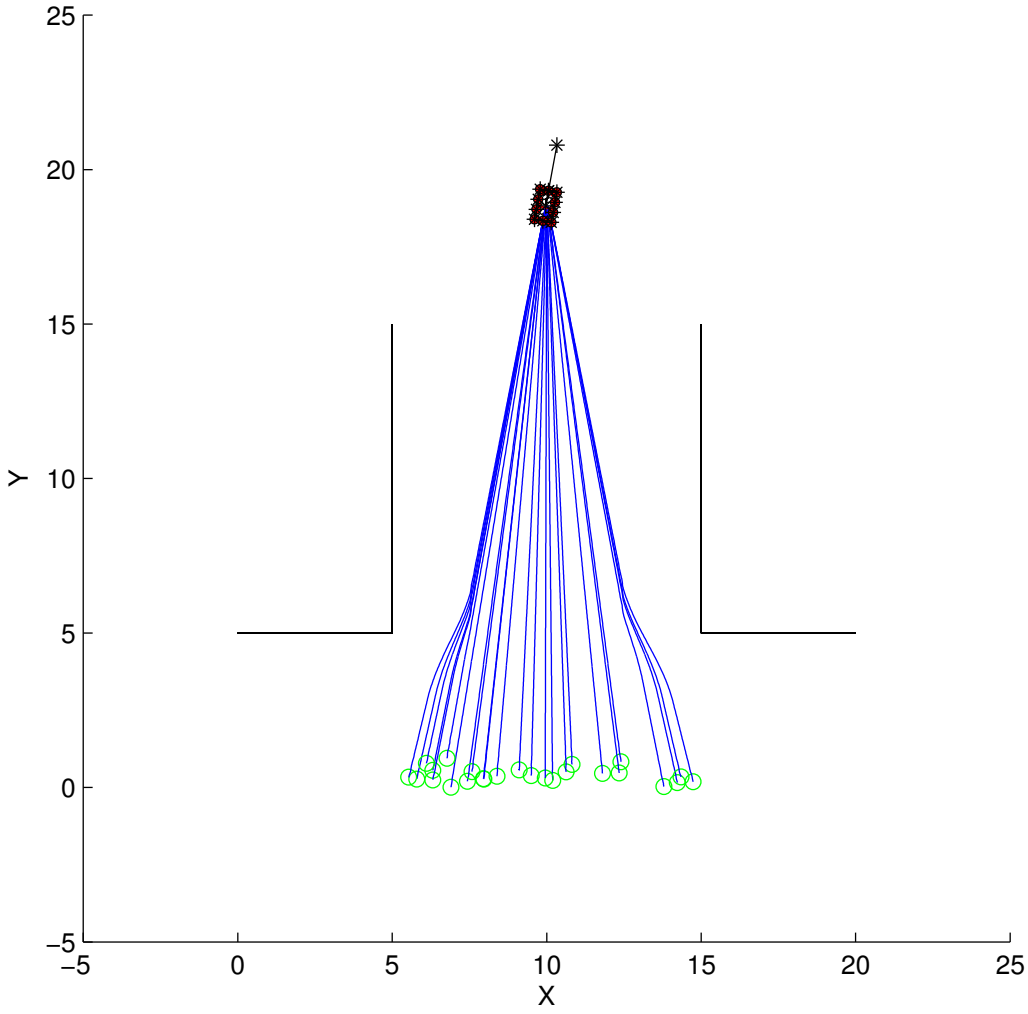


Figure A.22: Test Series 4 - Hallway (Wide) Map. 10 PSP used. Only attractive and repulsive forces used. Success: 25 of 25

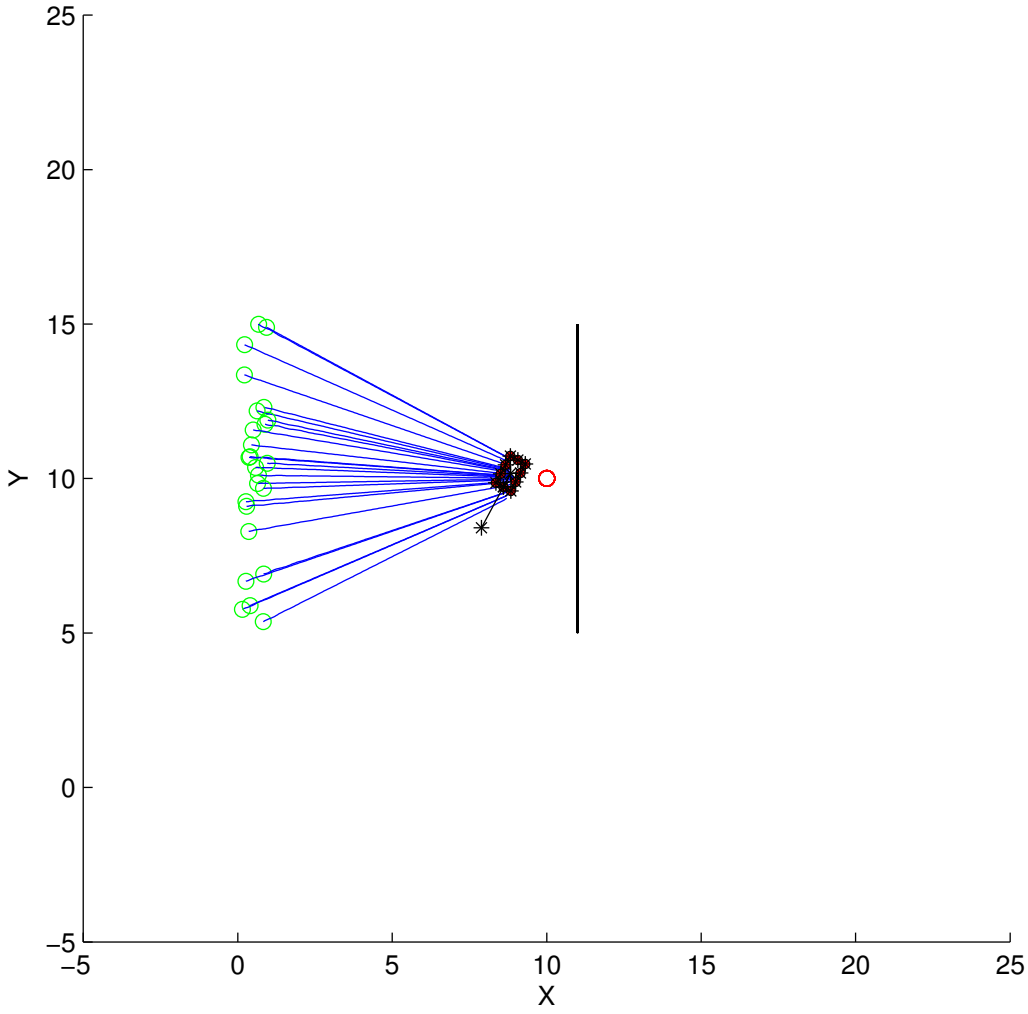


Figure A.23: Test Series 4 - GNRON Map. 10 PSP used. Only attractive and repulsive forces used. Success: 0 of 25

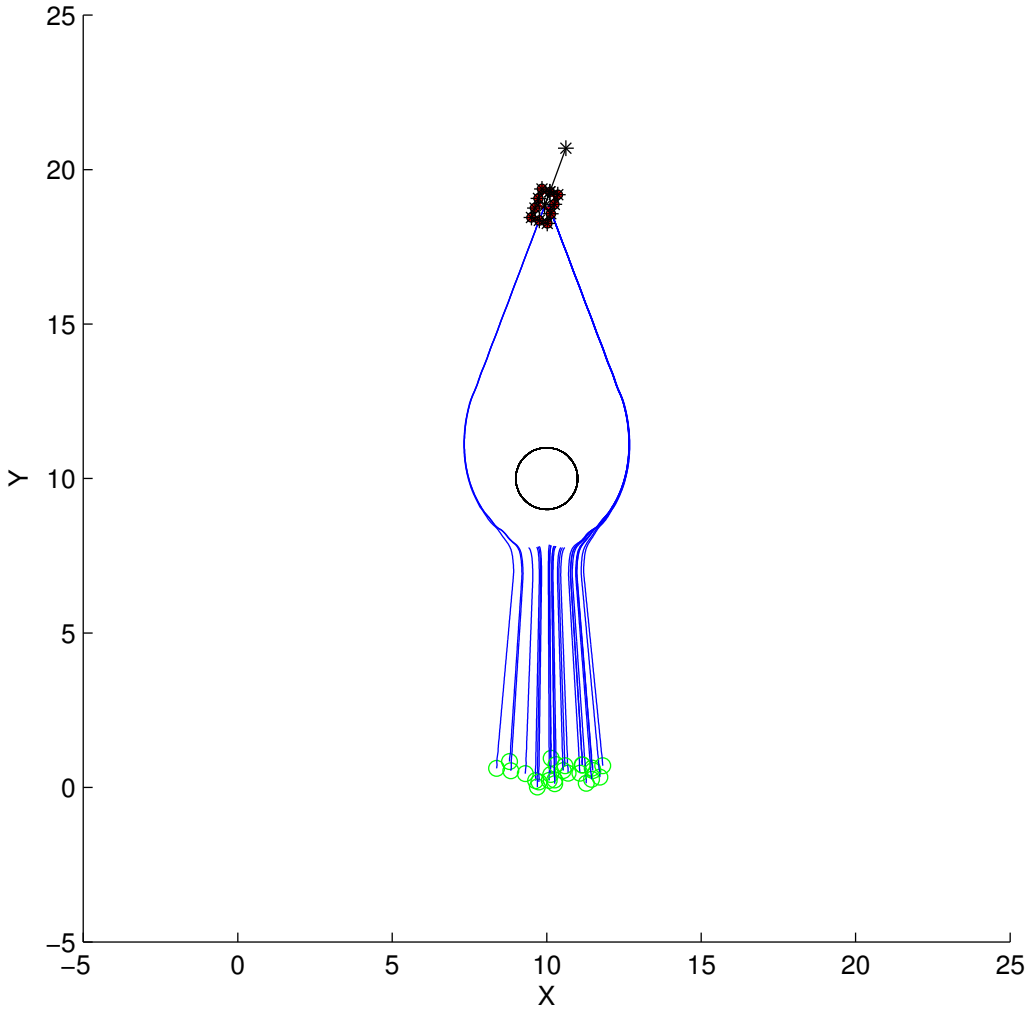


Figure A.24: Test Series 4 - Single Obstacle Map. 10 PSP used. Only attractive and repulsive forces used. Success: 12 of 25

A.5 Test Series 5

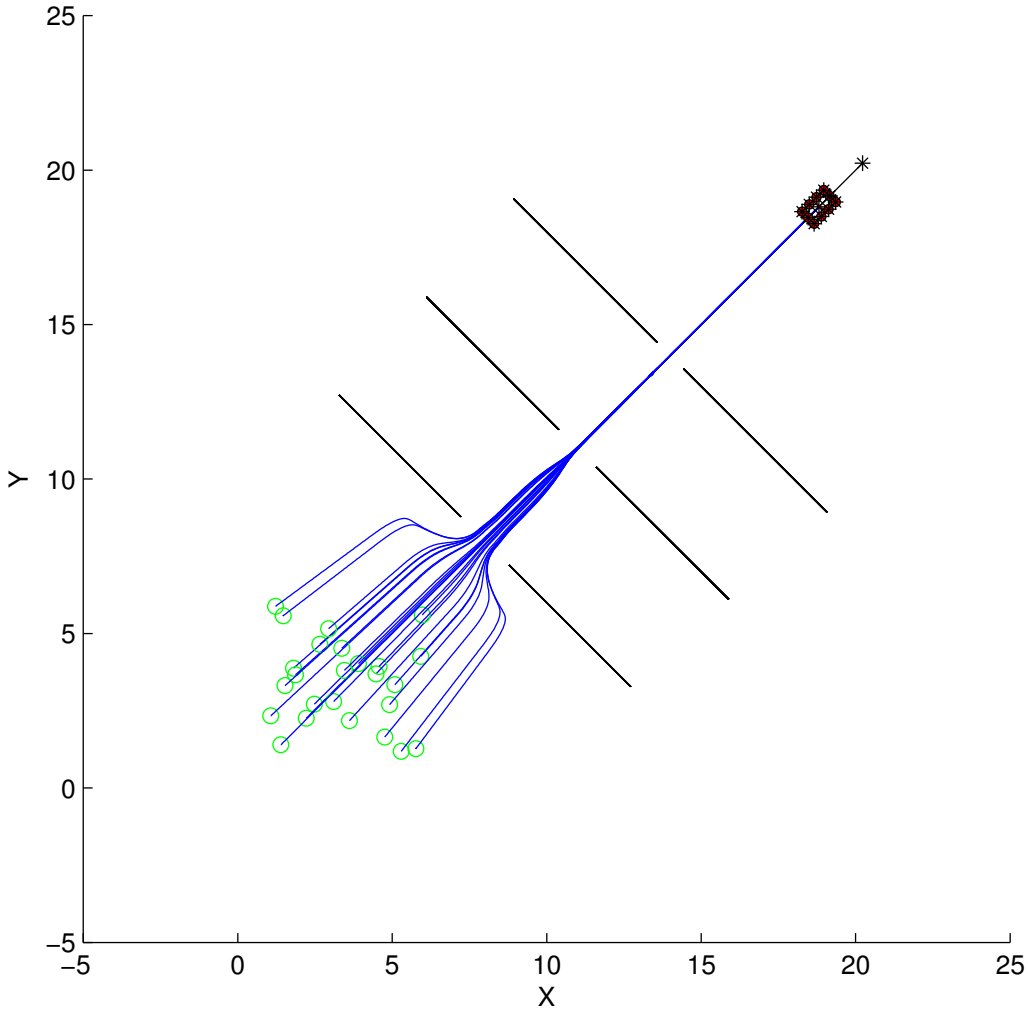


Figure A.25: Test Series 5 - Door Frame Map. 10 PSP used. All forces used. Success: 25 of 25

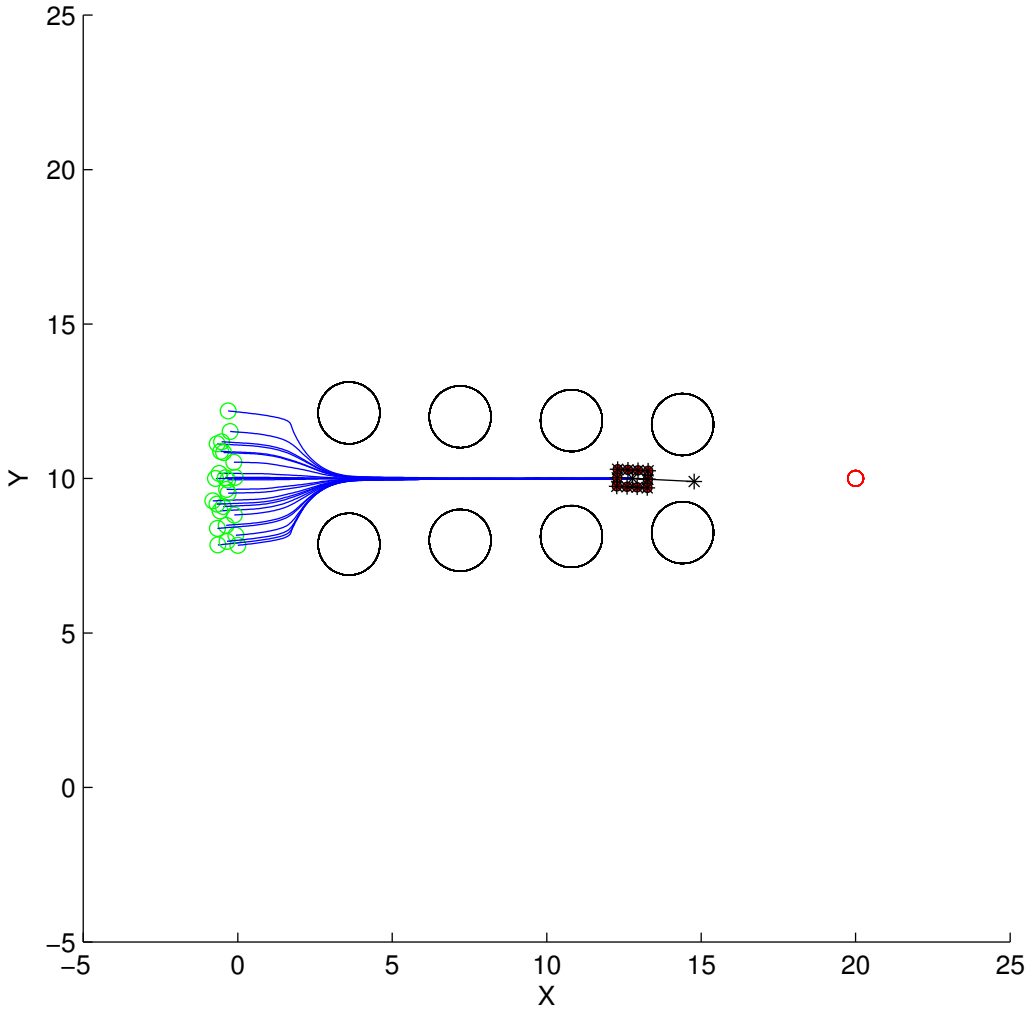


Figure A.26: Test Series 5 - Obstacle Gap Map. 10 PSP used. All forces used. Success: 0 of 25

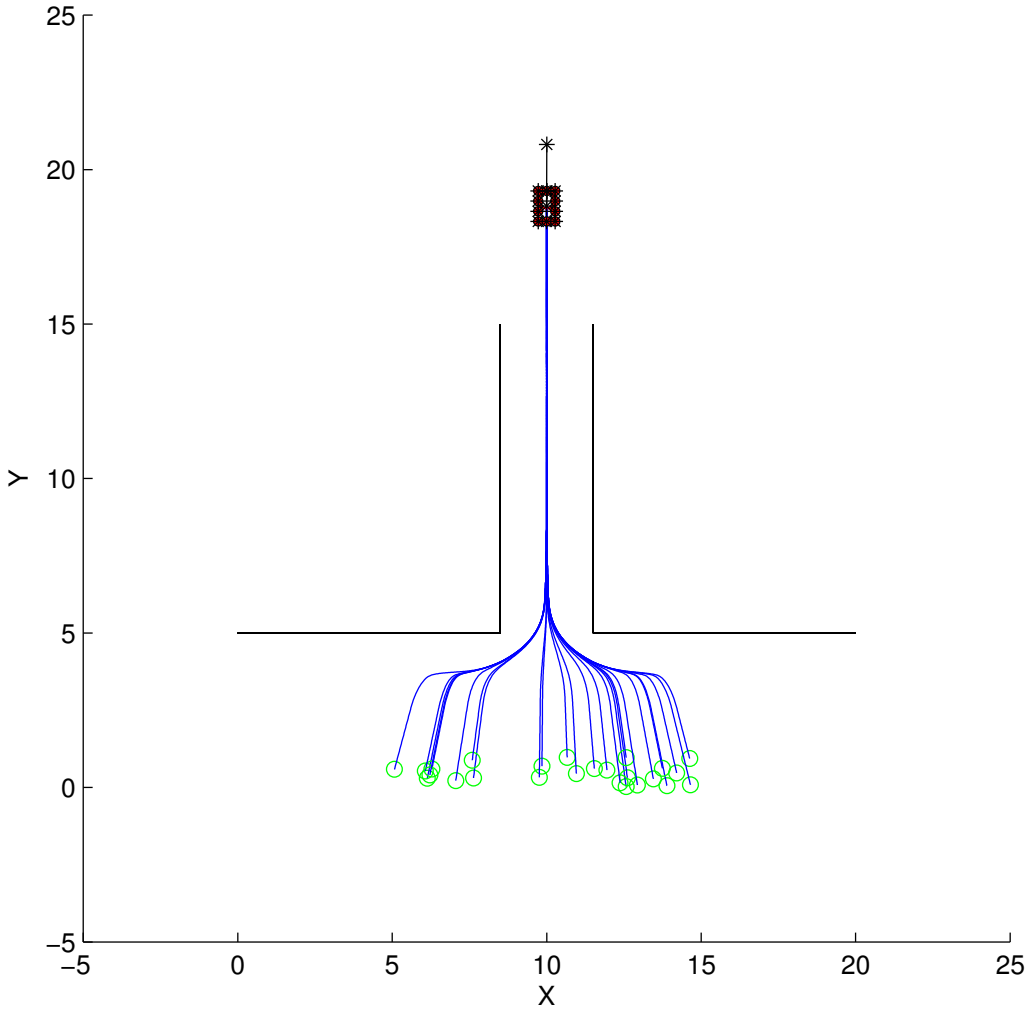


Figure A.27: Test Series 5 - Hallway (Narrow) Map. 10 PSP used. All forces used.

Success: 25 of 25

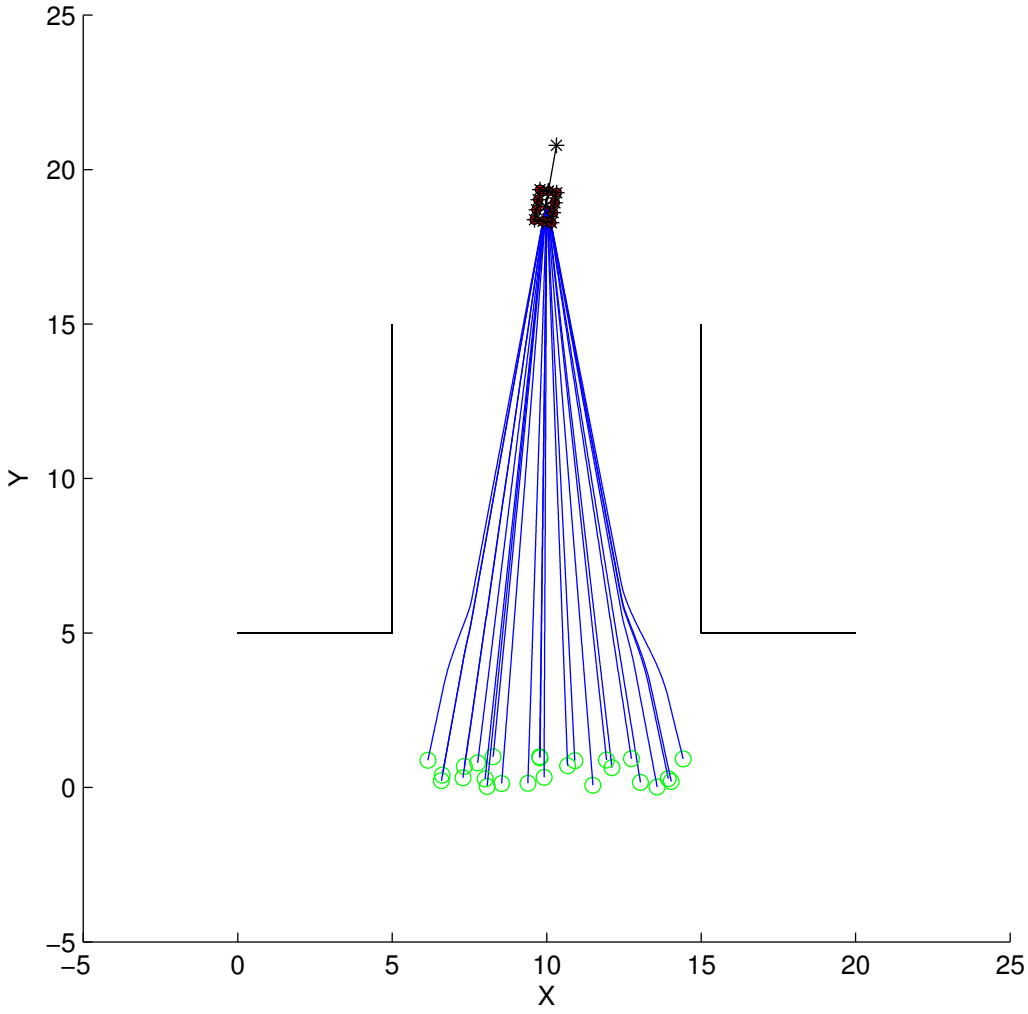


Figure A.28: Test Series 5 - Hallway (Wide) Map. 10 PSP used. All forces used. Success: 25 of 25

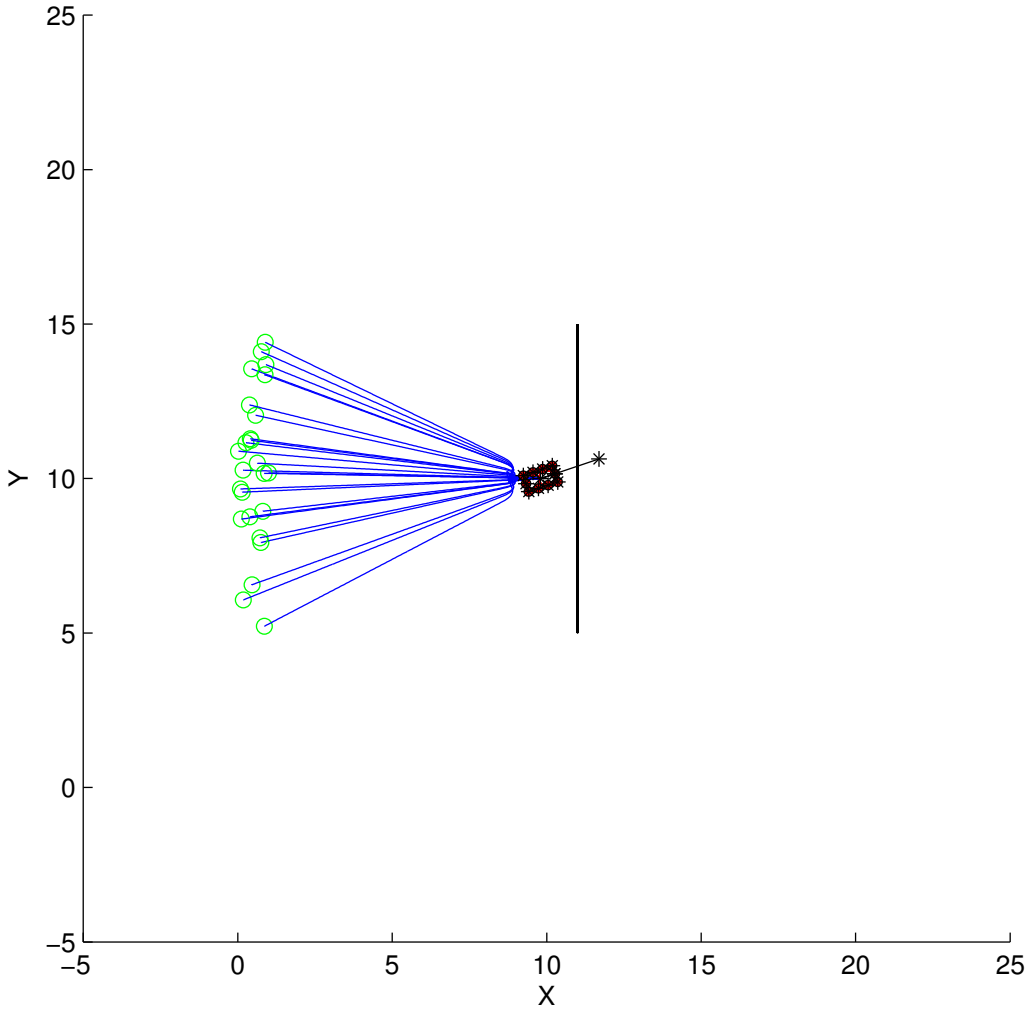


Figure A.29: Test Series 5 - GNRON Map. 10 PSP used. All forces used. Success: 25 of 25

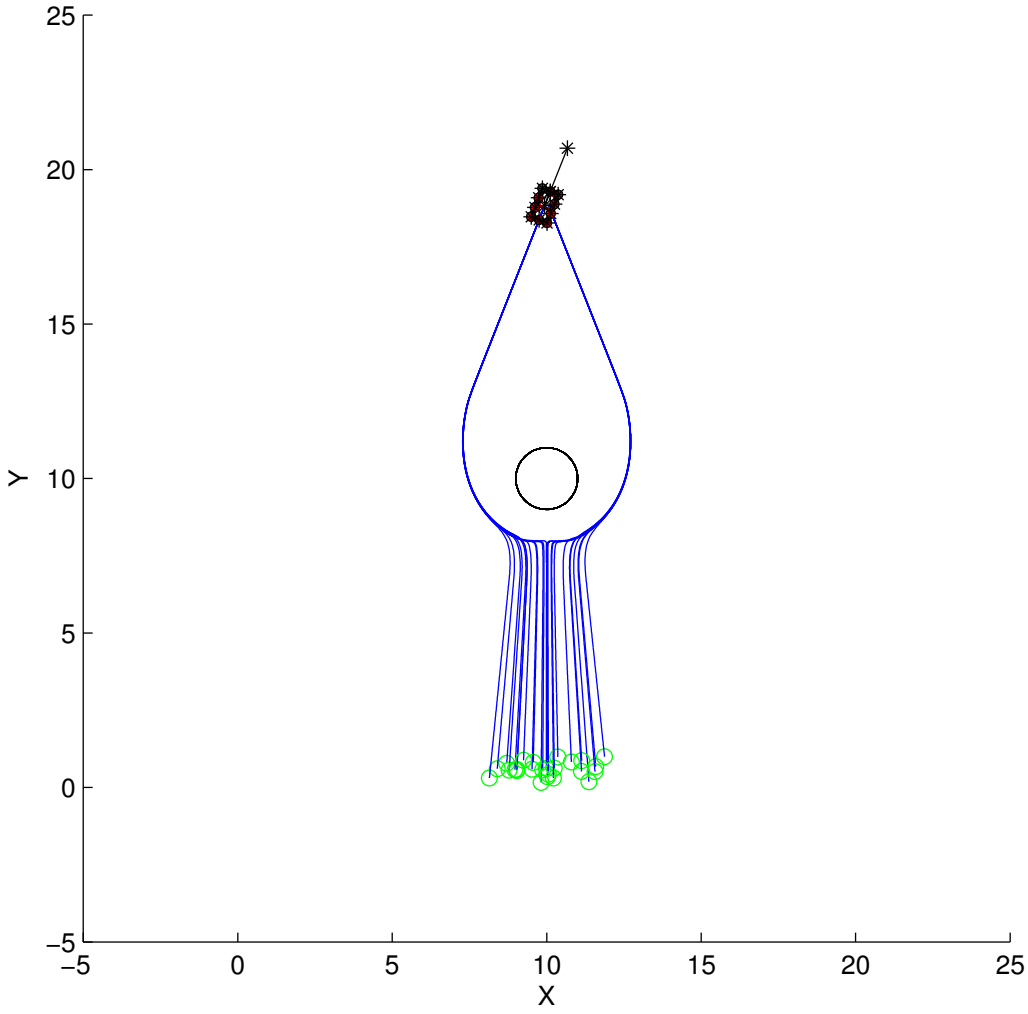


Figure A.30: Test Series 5 - Single Obstacle Map. 10 PSP used. All forces used. Success: 25 of 25

A.6 Test Series 6

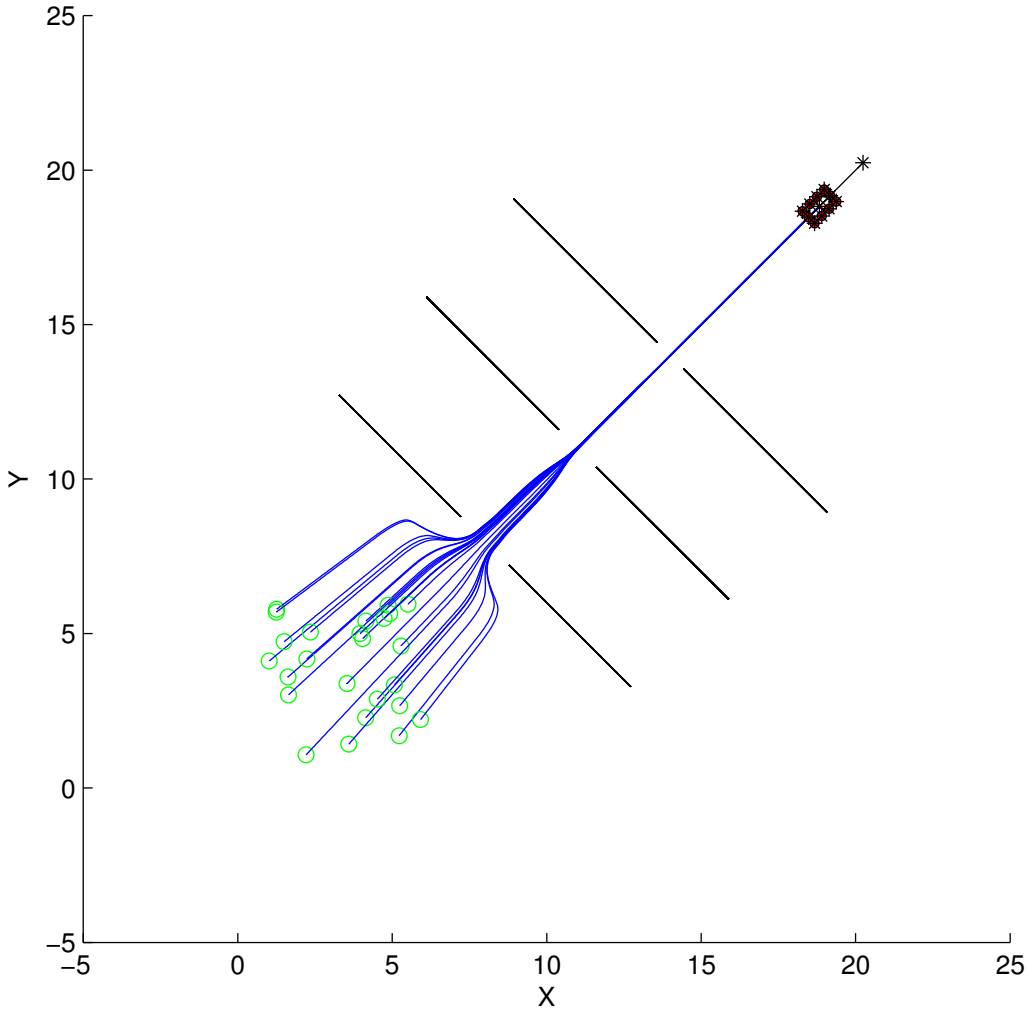


Figure A.31: Test Series 6 - Door Frame Map. 10 PSP used. All forces used save for tangential forces. Success: 25 of 25

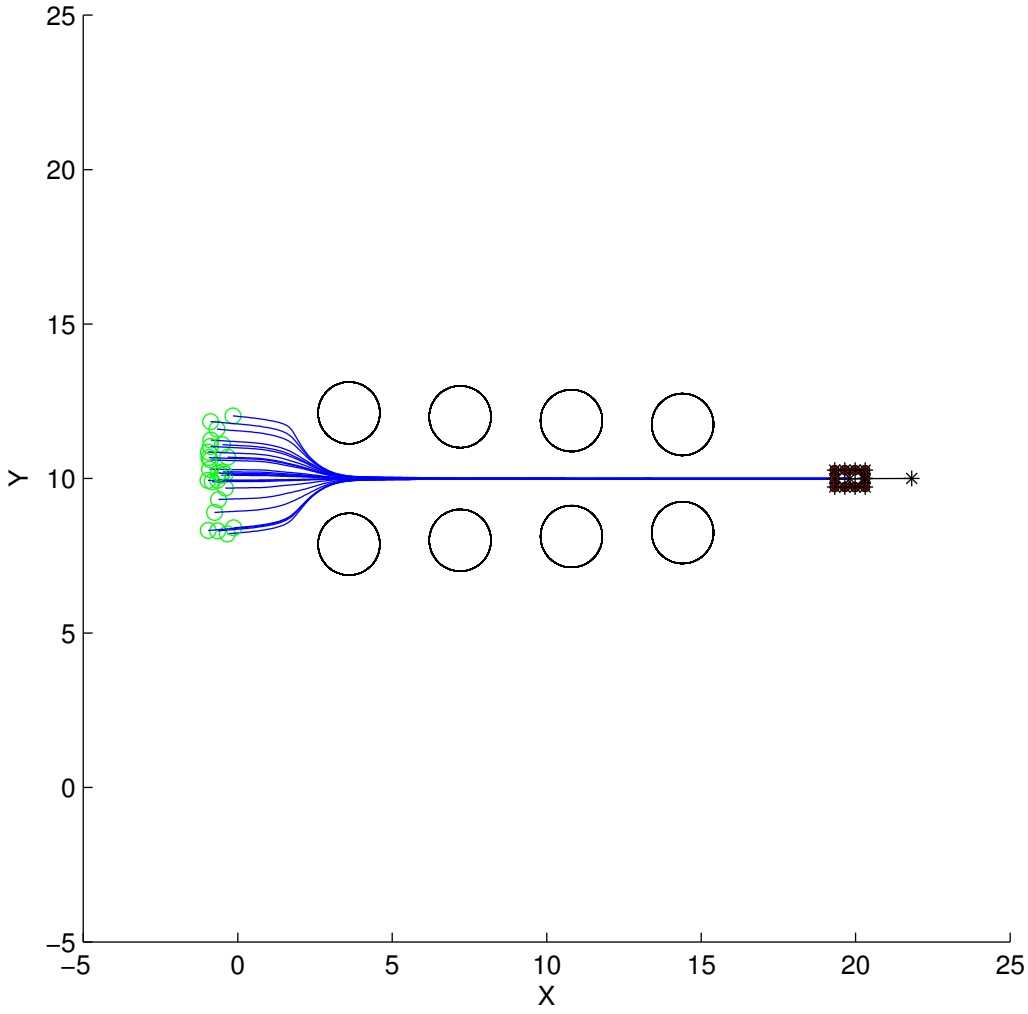


Figure A.32: Test Series 6 - Obstacle Gap Map. 10 PSP used. All forces used save for tangential forces. Success: 25 of 25

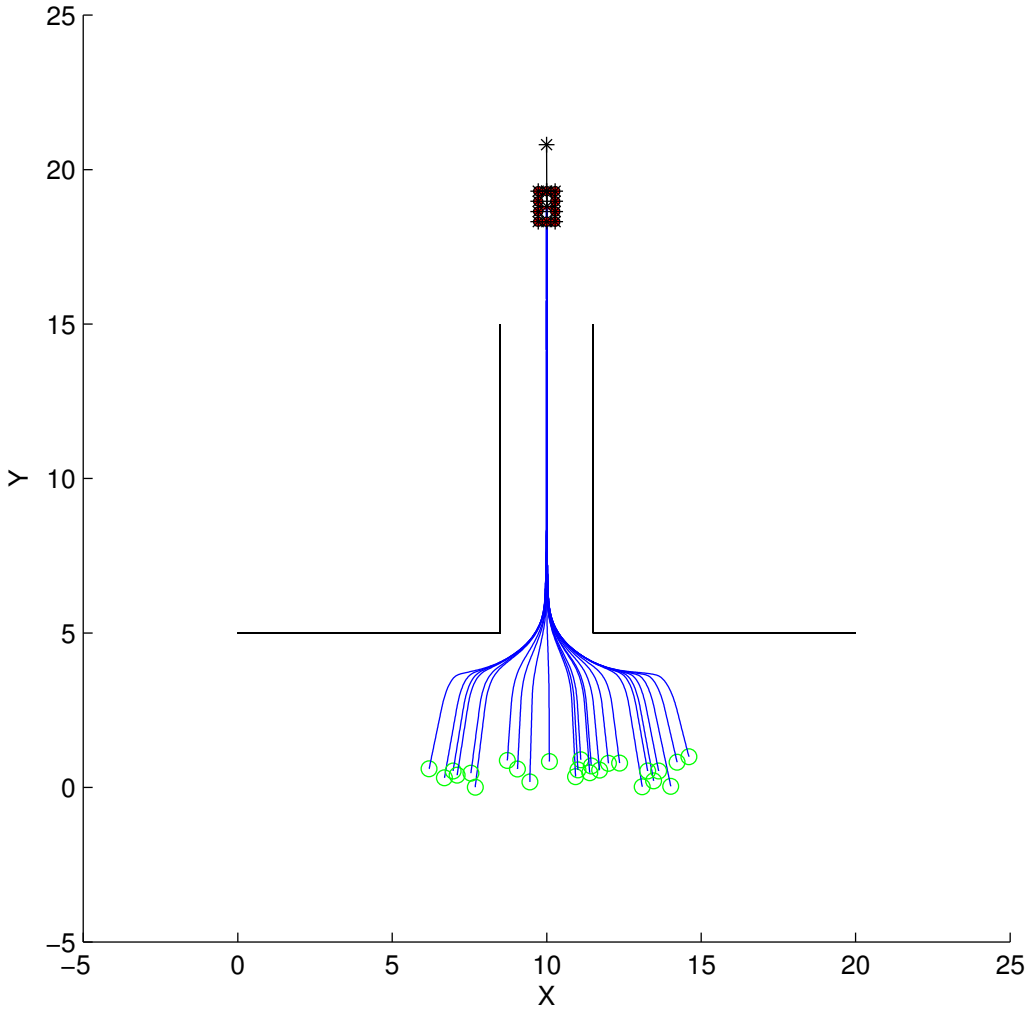


Figure A.33: Test Series 6 - Hallway (Narrow) Map. 10 PSP used. All forces used save for tangential forces. Success: 25 of 25

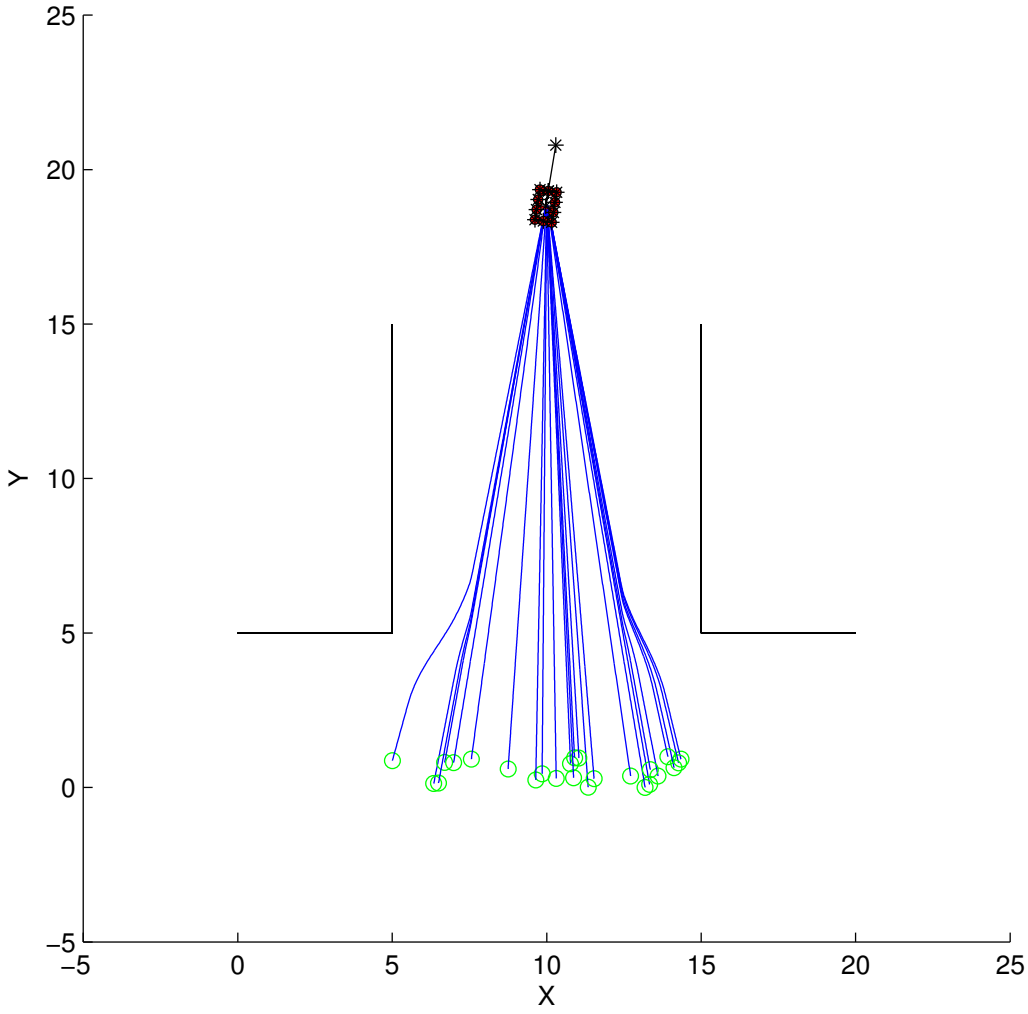


Figure A.34: Test Series 6 - Hallway (Wide) Map. 10 PSP used. All forces used save for tangential forces. Success: 25 of 25

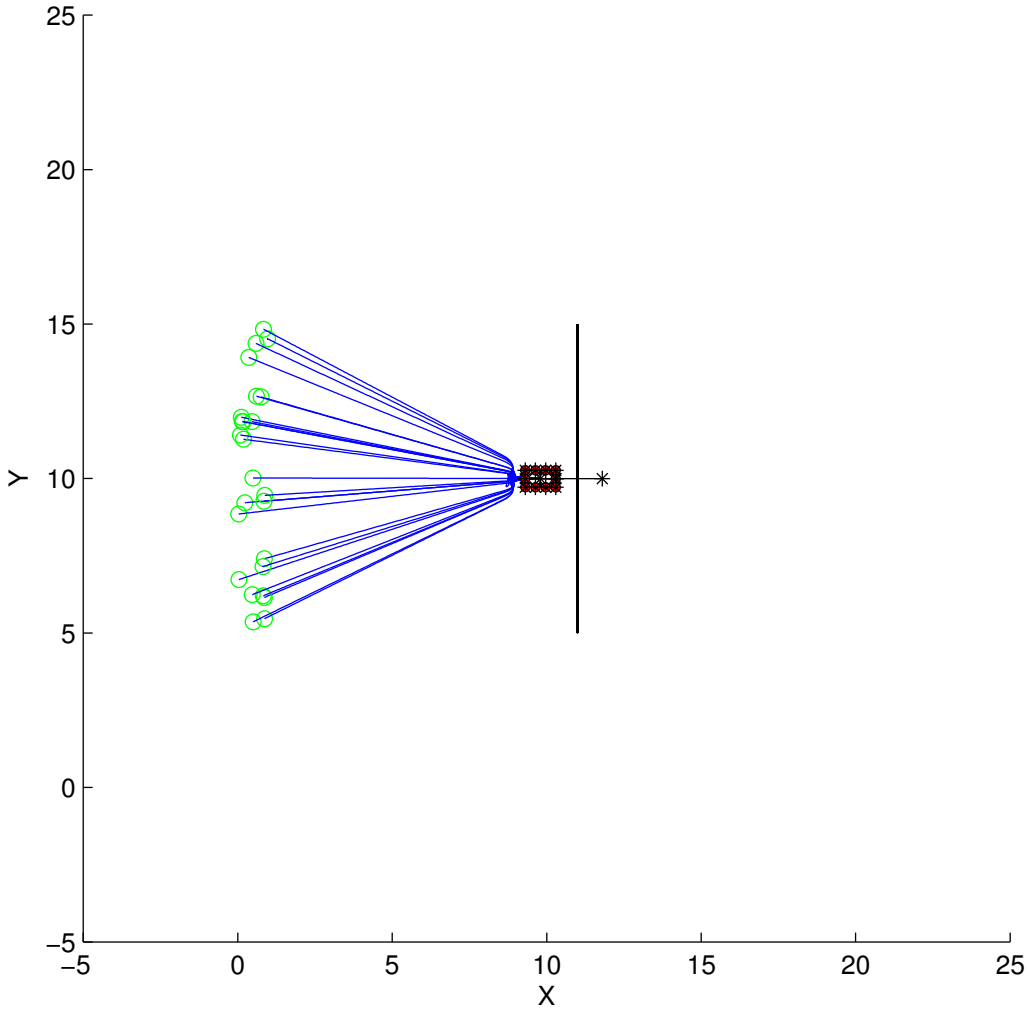


Figure A.35: Test Series 6 - GNRON Map. 10 PSP used. All forces used save for tangential forces. Success: 25 of 25

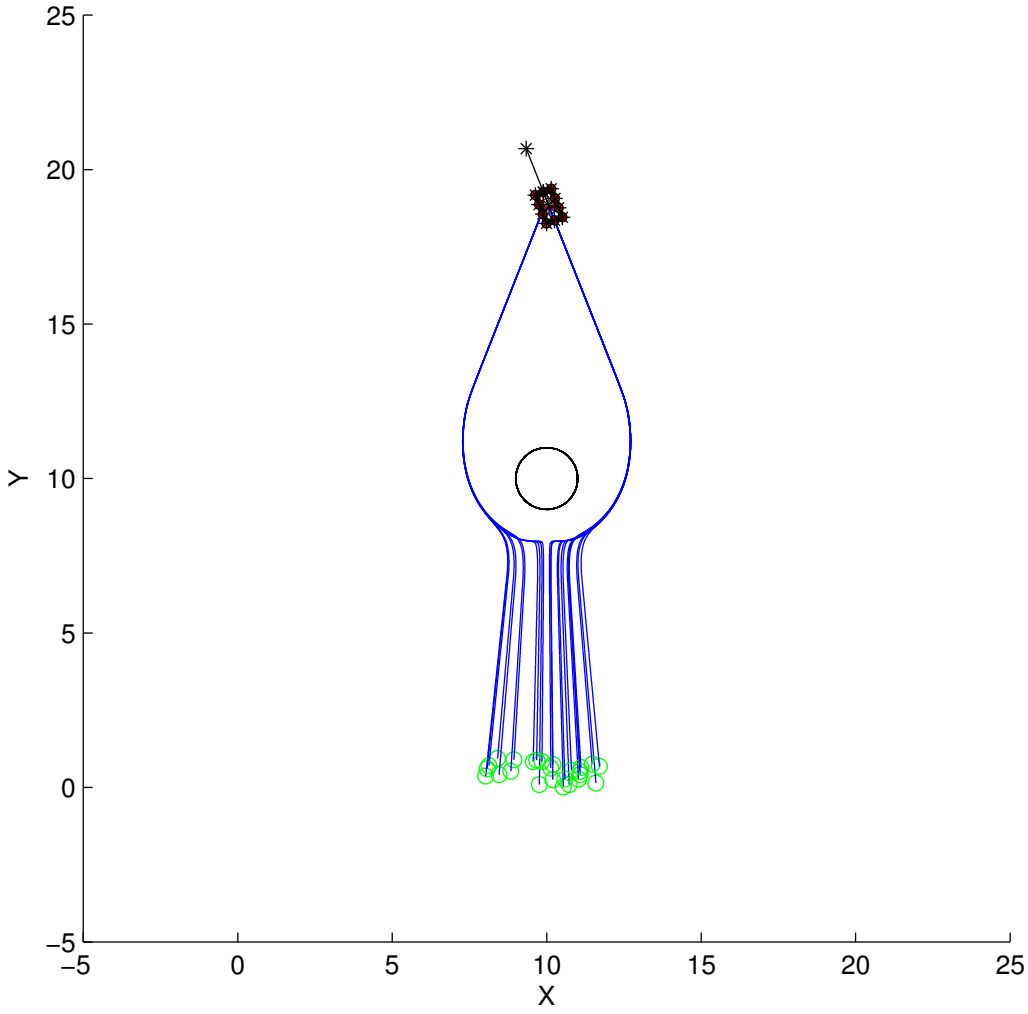


Figure A.36: Test Series 6 - Single Obstacle Map. 10 PSP used. All forces used save for tangential forces. Success: 25 of 25

Appendix B

Genetic Algorithm Optimization

With the sizable collection of parametric variables involved in this algorithm, manual initial parameter selection and tuning proved to be untenable. The non-linear and random nature of the developed simulation necessitated a genetic algorithm (GA) based approach to the selection of these parameters. Performing a more rigorous multi-variate simulation series would have taken a substantial amount of time and computing power. The number of simulations approaches n^{mn}; for $n = 12$ variables and only $m = 10$ variations of each this becomes $\sim 10^{192}$. A Monte-Carlo simulation may have produced an improved result than that presented in the genetic algorithm as it provides greater solution space coverage, but at the cost of many unused simulations, potentially wasting computer cycles. The use of a genetic algorithm was chosen because it uses a random approach with the guidance of the success of previous simulations. The purpose of the genetic algorithm was not to provide the precise

parameters to use, but to provide a starting point for manual tuning later. This optimization was run as a standalone exercise, and is not part of the RAVE algorithm. It is presented here as part of the discussion on how initial values for the RAVE parametric values were developed.

Genetic algorithm optimization (GAO) uses an evolutionary approach to optimize non-linear problems. Defining an individual as a single simulation, the variable parameters are then defined as that individual's chromosome, the unique information defining that individual. By defining a fitness metric, a cumulative score of simulation performance in this case, different individuals can be compared and ranked using the fitness function defined in Equation B.5. Taking a selection of the fittest individuals, offspring are created by merging the chromosomes. Additional limited variation is introduced to reduce the limiting effects of inbreeding. These offspring then form the population of the succeeding generation.

For this thesis the following genetic algorithm process was used, following the process discussed above:

1. An initial population of chromosomes, $\vec{P}$, is created. Each chromosome, $\vec{C_i}$, represents the set of parameters for a single simulation run.

2. Generate a generation specific map (simulated rock field type)

3. Perform the simulation using each chromosome against the generated map

4. Evaluate each chromosome using the fitness function. Only successful simulations are evaluated.

5. Select the top 10% of the evaluated population as the "fittest", introducing new, random chromosomes to fill if needed, $\vec{P}_{fit}$.

6. Generate a new population, $\vec{P}_c$, set by randomly "mating" pairs of $\vec{P}_{fit}$.

   (a) Select a "mother", $\vec{C}_m$, and "father", $\vec{C}_f$, chromosome.

   (b) Create a "child" chromosome, $\vec{C}_c = 0.5 * \left(\vec{C}_m + \vec{C}_f\right)$

   (c) Randomly add noise to each "child" chromosome value with a 20% chance of multiplying the "child" value by $\mathcal{N}(1, 0.0625)$

7. Set $\vec{P}$ to $\vec{P}_c$

8. Repeat steps from 2 for the number of desired generations.

This process is captured in Algorithm 2 below:

$$m_{Length} = K_L \frac{L_{Total}}{L_{Minimum}} \tag{B.1}$$

$$m_{Range} = K_R \frac{L_{Remain}}{L_{Minimum}} \tag{B.2}$$

$$m_{Curvature} = K_{Cv} \sum_{k=0}^{n-1} (\phi_{k+1} - \phi_k) \tag{B.3}$$

$$m_{Speed} = K_T \frac{L_{Total}}{t_{Total}} \frac{1}{\dot{x}_{Maximum}} \tag{B.4}$$

$$m_{Total} = m_{Length} + m_{Range} + m_{Curvature} + m_{Speed} \tag{B.5}$$

// Initialize
$\vec{P} = \left[\vec{C}_1, \vec{C}_2, \cdots, \vec{C}_i\right]^T$
for $n_{generations}$ **do**
    // Generate the map
    $M = GenerateMap()$
    foreach $\vec{C}$ ***in*** $\vec{P}$ **do**
        // Simulate
        $R = Simulate(\vec{C}, M)$
        // Evaluate Fitness
        $\vec{F}_i = ObjectiveFcn(R)$
    end
    // Select Fittest
    $\vec{P}_{fit} = \vec{P}$ **sorted by** $\vec{F}$; **select top** 10%
    while ***length***$(\vec{P}_{fit}) < 0.1 *$ ***length***$(\vec{P})$ **do**
        append new $\vec{C}$ **to** $\vec{P}_{fit}$
    end
    // Generate Children
    $\vec{P}_c =$ **mate**$(\vec{P}_{fit})$
end

Algorithm 2: Genetic Algorithm Optimization