Investigation of Local Minima in Autonomous Potential Field Agents/Vehicles in Pure Dynamic Environment

A Thesis Submitted to the University
of Kent
For the Degree of Doctor of Philosophy
In Electronic Engineering

By
Thomas Statheros

2013

To the Memory of my Parents and Grandmother

Abstract

Autonomous vehicle navigation can be divided into two major areas of research: Collision avoidance and Track-Keeping. This study focuses on Collision avoidance which is one of the major issues that unmanned autonomous vehicles have to face. Collision avoidance may be further grouped into classical and soft computing based categories. Classical techniques are based on mathematical models and algorithms, while soft-computing techniques are based on Artificial Intelligence. In this study, we focus on the Classical techniques and more specifically in the Potential Field Methods.

The potential field algorithms rapidly gained popularity due to their simplicity and elegance. In other words, Potential Field Methods are generic, computationally efficient and generate naturally smooth trajectories. On the other hand, PFM algorithms experience local minima. Nevertheless, local minima for PFM are extensively studied in different environments; they have never studied in a Pure Dynamic Environment (PDE). PDE is a new dynamic environment in which all its elements are guaranteed to be dynamic at their initial state.

In this way we have managed to identify and define the causes of Potential Field Agent local minima and trajectory inefficiencies in a number of collision scenarios within PDE. To efficiently and accurately identify and define these causes of local minima and trajectory inefficiencies, we have introduced the novel concept of the Monovular Autonomous Agent Correlation. Based on this concept we have identified and mathematically defined the Trajectory Equilibrium State (TES) for the first time. This state is responsible for local minima and trajectory inefficiencies of Monovular Autonomous Agents in PDE.

Because of TES identification and definition we have designed a rule based mathematical algorithm that efficiently navigates the Autonomous Agents out of local minima and trajectory inefficiencies in PDE in a number of generic collision scenarios. The algorithm's performance is tested in a number of simulated water based collision scenarios.

Contents

List of Figures

List of Tables

Publications by Thomas Statheros Related to this Thesis

Journal Publications

1. Autonomous Ship Collision Avoidance Navigation Concepts, Technologies and Techniques. T Statheros, G Howells, K McDonald-Maier, 2008, Journal of Navigation, Volume 61, Issue 1 (independently **cited by 38, within the 10 most cited journal of journal of navigation in 2009**)

2. Trajectory Equilibrium State Detection and Avoidance Algorithm for Multi-Autonomous Potential Field Mobile Robots. T Statheros, G Howells, K McDonald-Maier, Jul 2007, IET Electronics Letters, Volume 4, Issue 15

Conferences

3. A Novel Potential Field Algorithm and an Intelligent Multi-classifier for the Automated Control and Guidance System (ACOS)
T. Statheros, G. Howells, K.D.McDonald Maier, P. Lorrentz. Jul 2009, ICINCO 2009, Milan, Italy

4. Automated Control and Guidance System (ACOS): An Overview.
T. Statheros, M. Defoort, S. Khola, K.D. McDonald-Maier, W.G.J. Howells, A. Kokosy, J. Palos, W. Perruquetti, T. Floquet, D. Boulinguez. Jul 2006, RASC, Canterbury, UK

5. An Intelligent Fast-Learning Multi-Classifier System based on Weightless Artificial Neural Architectures. G. Howells, S. Khola, T. Statheros, K.D. McDonald-Maier. Jul 2006, Sixth International Conference on Recent Advances in Soft Computing (RASC), Canterbury, UK

Journal papers to be published

6. Analysis of Monovular and Biovular Potential Agents in Pure Dynamic Environment (PDE).

7. Potential Field Algorithms' differences between static and dynamic environment.

Acknowledgments

First of all, I would like to thank my Ph.D. supervisor Dr. Gareth Howells who has been very supportive to my research views and ideas as well as giving great guidance for the whole period of my Ph.D. I would also like to thank Prof. Dr. Klaus D McDonald-Maier from University of Essex for his advice and support, while I was working in the Automated Control and Guidance System (ACOS) project.

During my Ph.D. in Kent University, I have enjoyed valuable discussions related to my research work with several friends and colleagues. Therefore, I would like to thank:
Mixalis Pediaditakis, Afroditi Pina, Evangelos Papoutsis, Nikos Giannakis, Philippos Asymakopoylos, Pavlos Sklikas, Vassilis Taralis and Huiling Zhu.

Finally, I would like to thank my family, and especially my sister Stamatia for the encouragement and support.

Chapter 1

Introduction

1. Introduction

Autonomous vehicle navigation is based on Collision Avoidance (CA) and Track-Keeping. Track-keeping is related to the control methods that allow the vehicle to follow a certain predefined segment of trajectory. CA is the method that obstacles are avoided via the generation of a different hierarchical path between the vehicle current position and its target destination. In this study we focus on the Collision avoidance. We have two major parts of path planning: on-line path planning and off-line path planning. Off-line path planning refers to global environment path planning and is not real-time while on-line path planning refers to the real-time path planning. In addition, CA could be further grouped into classical and soft computing based categories. Classical techniques are based on mathematical models and algorithms, while soft-computing techniques are based on Artificial Intelligence. In this study, we focus on the Classical techniques and more specifically on the Potential Field Methods, which is a real-time path planning method. In chapter 2, a review of the CA methods related to autonomous unmanned surface vehicle takes place along with its comparison with human's cognitive abilities.

The potential field algorithms have extensively researched due to their simplicity and elegance. In other words, Potential Field Methods are generic, computational efficient and generate naturally smooth trajectories. On the other hand, PFM algorithms experience local minima. Local minima causes the autonomous potential field agent/vehicle to stop or to oscillate while it is trapped into a group of local coordinates. The local minima for PFM are extensively studied in different environments, but what cause them in in a Pure Dynamic Environment (PDE) haven't been identified. We have defined as a PDE, a new dynamic environment in which all its elements are guaranteed to be dynamic at their initial state. We have decided to investigate the causes of local minima in PDE, since it is desirable to accurately understand the performance of the Classical Potential Field Algorithms without the 'noise' static environment. In addition, the vehicle/agent dynamic model is based on a point-mass model for the same reason.

To investigate the performance of PFM in PDE, we have designed and implemented a Potential Field Method (PFM) based on Virtual Force Field (VFF) of a classical PFM [1-3]. Moreover, this modified PFM incorporates processing power oriented improvements by design changes of the Active Window (AW). An analytical presentation of the design and the implementation of this algorithm are presented in chapter 3; we have named this algorithm Active Window Single Point Potential Field (AWSPPF) Algorithm.

We have tested this algorithm in an extensive number of cross collision scenarios, and based on these results, the need of a new agent concept has arisen. This new agent concept reveals local minima behaviour in Pure Dynamic Environment. We have named the agent concept "Monovular" and the novel concept that identified local minima in PDE "Monovular Autonomous Agent Correlation" (MAAC). In this concept the agents in the same local environment are identical in both dynamic/kinematic models and algorithm implementation. The word Monovular is inspired by the biological term Monovular , which means from the same ovum (egg), and The MAAC is inspired by the Signal Processing Concept of Autocorrelation. Based on this concept, we have identified and mathematically defined the Trajectory Equilibrium State (TES) for the first time. This state is responsible for local minima and trajectory inefficiencies of Monovular Autonomous Agents in PDE. We explain the TES state, as well as the MAAC in chapter 4. The concept was published for first time by the author of this thesis in [4].

Based on the above, an efficient algorithm that identifies and avoids local minima in PDE of both Monovular and Biovular agents/vehicles in local environment is presented. This method is a combination of the AWSPPF and a rule based mathematical algorithm. The algorithm performance is tested in a number of generic water based collision scenarios, which are presented in chapter 5; some of these performance results are published by the author in [4-6].

1.1 Motivation of the Thesis

Humans perform navigation to a satisfactory level, but their critical decisions are highly subjective. This can lead to error and potentially, to collision. The human navigation operation could lead in collision is called real-time path planning. Therefore, to replace this human operation, a rigorous research is carried out for a number of years in real-time path planning section of autonomous navigation.
One of the most efficient real-time path planning algorithms, as we will analyse in chapter 2 is the Potential Field Methods (PFM).

The efficiency of this algorithms rely on its properties, which are:

- Low processing power
- The collision avoidance trajectory to be realistic in relation to own agent dynamic model with minimal agent modeling. In other words, PFM generates realistic smooth trajectory, so the Potential Agent can follow instantly the generated track segment in safe critical situations.

Therefore, we have selected to evaluate the PFM among other path planning and collision avoidance algorithms due to its elegance and simplicity, as well as to its natural smooth trajectories generation.

On the other hand, as it has repeatedly been underlined in many studies [7], the PFM are mainly suffering from two major drawbacks, local minima. When the autonomous potential agent is close to local minima in a confined environment experiences trajectory oscillations. The main causes for trajectory oscillations in a static environment have been identified in [7], also a number of solutions for local minima have been discussed in [1, 7, 8]. Nonetheless, none of these studies have examined and defined the existence and the reason of local minima in PDE. Therefore, we need to identify the reasons, which cause local minima in PDE; this way we can vastly improve the performance of the PFM as we will analytically prove in chapter 5.

The basis for the identification of local minima in PDE was an extensive testing of two uniquely defined Potential Autonomous vehicles/agents in the same dynamic environment. Based on these results, the inclination of the algorithm for local minima was observed in the extent to make us able to identify a new Potential Field Algorithms new State that causes the Potential Autonomous agents to have poor performance.

Therefore, the main motivation of this study is: to evaluate and improve the performance of classical Potential Field Methods by identifying the cause of local minima in PDE, to mathematically define and predict the cause of local minima in this environment, as well as to design a mathematical method that improves Potential Field Performance in PDE.

1.2 Research Aims and Challenges

The main aim of this research was to examine the use of the Potential Field Algorithms in Pure Dynamic Environment, as well as to improve their performance without the algorithms losing their simplicity and elegance. Based on this aim a number of challenges were immersed. It is necessary to underline that simplicity and elegance of a collision avoidance algorithm in a dynamic local environment of fast agents/vehicle is very important, since the decision time is very limited. Nevertheless, the existence of local minima in this type of environment, due to fast speeds, could be more safety critical than in static environment.

Therefore, the first challenge of this research was to select the appropriate Potential Field Algorithm variation to accommodate the simplicity and elegance of the classic methodologies, which could possibly improve its computational capabilities. As we have referred to the above Potential Field Algorithms that are extremely computationally efficient they also have problems with local minima. In this stage the elimination of local minima was not yet the target. Therefore, we designed a Potential Field Algorithm that accommodates the features of a classical PFM [1, 3, 9]. We have achieved this by designing the Active Window Single Point Potential Field (AWSPPF) Algorithm that we have mentioned in the above section. In this algorithm we have used the concept of Active Window, which is introduced in [2] but we have further improved the computational performance of the algorithm by using a circular (AW). The algorithm's mathematical analysis and the mathematical justification of the circular AW is presented in chapter 3.

The AWSPPF algorithm exhibits local minima in static environment but is more computational efficient than the classical methodologies. The second challenge of the research was to clearly identify the cause of local minima in PDE. This challenge took place by testing the AWSPPF algorithm in a number of cross collision scenarios, which revealed the inclination of the algorithm to have bad performance when it is on specific initial conditions with another dynamic agent/vehicle/obstacle.

At this stage the 3rd challenge of this research was to accurately define this similarity. We have achieved this by the introduction of the novel concept of Monovular and Biovular agents/vehicles, as well as the Monovular Autonomous Agent Correlation. Based on this new technique we managed for the first time to define the cause of local minima of Potential Field Agents in PDE, as well as to predict them long before the actual local minima occurrence.

Finally, the last challenge of this study was to utilise the local minima prediction to design an algorithm that will prevent the Potential Field Algorithm from local minima in PDE. This algorithm is described in chapter 5, and the author of this thesis has published the initial concept in [4] for Monovular agents/vehicles and in [5] for Biovular agents/vehicles.

1.3 Thesis Structure

This thesis is divided in 6 chapters, which are:

Chapter 1 presents the introduction

Chapter 2 provides both a spherical understanding about autonomous water based vehicles for collision avoidance (CA), and a theoretical background of the reviewed work. It also includes mobile robots algorithms, since the genesis of a number of water based autonomous navigation methods took place in mobile robots domain first. In particular, we review the Potential Field Algorithms (PFA).

In this chapter 3 we design and implement a Potential Field Method (PFM) that accommodates the majority of the features of a classical PFM. Moreover, this algorithm incorporates processing power oriented improvements by redesigning the Active Window (AW). We use this algorithm to evaluate the PFM capabilities in the newly defined Pure Dynamic Environment (PDE), without the overhead of the dynamic/kinematic models of the autonomous agent.

In chapter 4 we introduce the novel concept of Monovular Autonomous Agent Correlation (MAAC). The significance of this concept is to identify performance inefficiencies of the Potential Fields algorithms in Pure Dynamic Environment PDE. These performance inefficiencies show that the potential field algorithm experience local minima not only in static environment but also in dynamic one.

In chapter 5 we have designed a rule based mathematical algorithm that efficiently navigates the Autonomous Agents out of local minima and trajectory inefficiencies in PDE in a number of generic collision scenarios. The algorithm's performance is tested in a number of simulated water based collision scenarios in both Monovular and Biovular Agents/Vehicles.
Chapter 6 presents the conclusion and future work.

Chapter 2

2. The generic problem of collision in navigation of water based vehicles and mobile robots and the autonomous navigation techniques

This chapter provides both a spherical understanding about autonomous water based vehicles for collision avoidance (CA), and a theoretical background of the reviewed work. It also includes mobile robot algorithms, since the genesis of a number of water based autonomous navigation methods took place in mobile robots domain first. In particular, we review the potential field algorithms (PFA).

Autonomous ship navigation can be divided into two major areas of research: Collision Avoidance (CA) and Track-Keeping (TK). This study focuses on the former Collision Avoidance that may be further grouped into classical and soft computing based categories. Classical techniques are based on mathematical models and algorithms that deal with determinism, while soft-computing techniques are based on Artificial Intelligence (AI) (algorithms deal with uncertainty). The areas of AI for autonomous ship collision avoidance that are examined in this chapter are: evolutionary algorithms, fuzzy logic, expert systems, neural networks (NN), as well as, a combination of all of them (hybrid system). For the areas of the classical techniques we mostly focus on potential field methods for both mobile robots and autonomous water based vehicles.

Nevertheless, in this chapter we also compare the human's cognitive abilities for CA with different techniques for autonomous CA algorithms. Humans perform navigation to a satisfactory level, but their critical decisions are highly subjective. This can lead to error and potentially, to collision. To limit the human subjective factor, therefore, the International Marine Organisation (IMO) has defined the international rules for collision avoidance (COLREGs) [10-14]. The effectiveness of COLREGs in human ship navigation is also investigated within this study.

2.1 Contrasting human cognitive abilities and intelligent algorithms for ship collision avoidance

Collision avoidance is one of the major issues that mariners face. Therefore, it is not surprising that autonomous ship navigation success depends on the development of efficient real-time intelligent algorithms for collision avoidance. Most of these algorithms attempt to imitate human piloting cognitive abilities [15]. To understand better the cognitive abilities of the "captain" within the ship, and their similarities or differences to the intelligent navigation algorithms, we must take into consideration all the human operations that are performed for collision avoidance purposes. These operations can be understood better by observing the factors that influence ship collision avoidance. These factors illustrated in Figure 1 are now introduced and discussed.

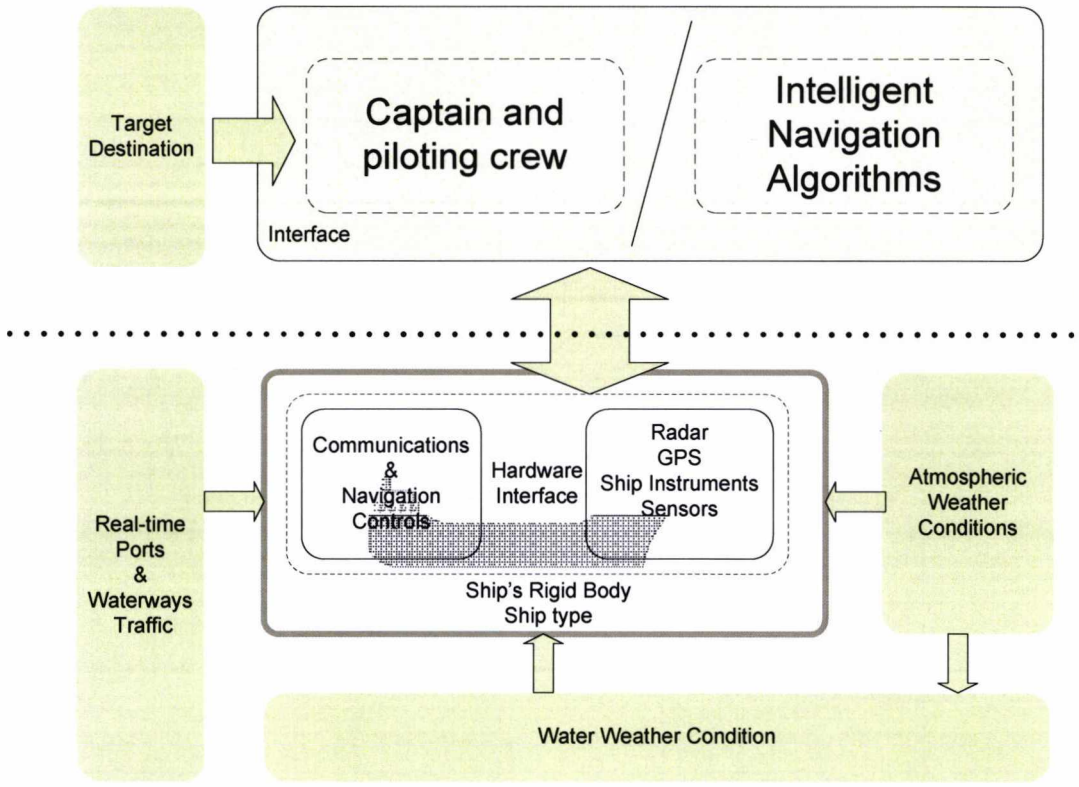


Figure 2-1: Ship navigation factors influence collision avoidance.

1. The *ship type* (e.g. sail boat, speed boat, commercial ship, passenger ship etc.) defines the properties of the transport (e.g. manoeuvrability, speed, Pay-load, weather conditions tolerance etc.) within the water medium. Each type of ship employs different kinds of evasive manoeuvres for collision avoidance, since speed and agility can differ significantly from one ship type to another. For each ship type, the captain and crew have special training. Then again, the CA autonomous navigation algorithms "understand" ship type as "ship dynamic" or " ship mathematical model" (e.g.[16-19]). These models provide a prediction of ship behaviour (based on inertial and fluid mechanics rules) but sometimes are very simplistic producing erroneous predictions or very complex to operate in real-time.
2. All forms of *sea traffic*, (e.g. a vessel, during its journey from the port to the target destination, can collide with another vessel or an obstacle). We can divide the types of traffic into two main categories: 1) traffic within confined environment (e.g. ports or Canals) [20] and 2) traffic in open sea waterways. In both traffic categories, we have to take into consideration the additional traffic complexity caused by the under surface environment (seabed level, wrecks and other underwater obstacles), as well as, the random dynamic and static sea obstacles. Ship collision avoidance in a confined environment is based on the guidance via the communication between the local traffic control station (port or canal traffic control) and the piloting crew of the ship. On the other hand, open sea collision free navigation is based on the communication of the captain and the encountered ship or ships, as well as, the COLREGs guidelines. It is important to note that most of the intelligent algorithms for ship navigation do not communicate among themselves or with the traffic control station. Therefore, these algorithms calculate the safe and optimal trajectory for collision avoidance [21-26] relying on first, the current state (speed, direction) of each encounter vessel or ship and second, the COLREGs[10, 11].
3. The *weather conditions* in the water and in the atmosphere influence every aspect of the ship navigation. The weather manipulates each type of ship differently (e.g. sailing boat or engine powered vessels). Collision avoidance in different weather conditions requires different evasive manoeuvres from the piloting crew. For example, in severe weather conditions, the ship manoeu-

vres have to combine safety (avoid capsizing or sinking) and collision avoidance concurrently. In most autonomous navigation algorithms, the sea weather conditions are rarely considered [27].

4. Last but not least, *on-board technologies* assist the ship-crew to navigate safely and efficiently. Nowadays, maritime aid technologies include Global Positioning System (GPS), Radar, Automatic Radar Plotting Aid (ARPA) [27, 28] and Atmospheric and Water Weather Monitoring Instruments. A considerable number of the algorithms for ship autonomous collision avoidance consider that the ships are equipped with GPS and ARPA [29, 30]. But none of these algorithms take into consideration all the instrumentation, navigational data and communication for collision avoidance that most of the modern ships have available.

Each of the above factors and any combination of them require human operation for ship collision free navigation. These operations are subjective to each individual captain and his or her training and cognitive abilities. The clarification or the human cognitive abilities for collision avoidance is the key point of the design, of any nature of the CA intelligent algorithms. Consequently, the investigation of human cognitive demands for collision avoidance [15] is useful. Part of this investigation shows that mariners' preference of collision avoidance manoeuvres varies quite significantly [31]. In real-life tasks, piloting crews frequently make course changes up to 30 degrees to minimise uncertainty but there is also a significant number of occasions when piloting crews are reluctant to make large course or speed changes [32], since there is a trade-off between Collision Avoidance (CA) and track-keeping (TK). It is worth noting that one of the reasons for the Titanic tragedy [33] was the unwillingness of the piloting crew to change the ship's speed.

Ship navigators have the ability to cope with CA by planning ahead a sequence of possible evasive manoeuvres, which are updated in real-time to sustain a safe path for the ship. The sequence of these manoeuvres assembles the evasive trajectory of the ship that can be considered as knowledge-based [34, 35] and has a degree of uncertainty. This degree of uncertainty can be minimised by COLREGs. However, COLREGs define the vessels actions for collision avoidance between two ships. Therefore, even if COLREGs are in place to dictate the decisions of evasive actions, the analysis and performance of these guidelines are highly subjective

(especially in many-ship encounters) that incorporate uncertainty and unpredictability. These can lead to marine accidents [36, 37].

Marine accidents can also be reduced by ship navigation aid technologies used by an experienced officer. Nowadays, the majority of commercial and transportation ships are equipped with Automated Radar Plotting Aid (ARPA). The ARPA replaces the Radar handmade plots for the graphical representation of navigational status of the tracking objects. Then again, it has been observed that mariners are more likely to overlook the COLREGs when performing "get away" manoeuvres with the support of ARPA [38]. Subsequently, the use of ARPA can have negative results when are operated by inexperienced officers, since reported data shows that 56% of major maritime collision includes violation of "the rules of the road" (COLREGs) [39, 40].

In addition to the above, radar navigation has its risk for collision, since the probability of the target ship reaching the future position varies [41]. To minimise the above problem, and further to improve the collision avoidance of a ship, an electronic-mapping intelligent support for ship navigator system is proposed by [42]. A further study of how mariners have to cross alleys safely is detailed in [43].

Another technique to reduce the violation of collision avoidance rules by the piloting crew is by knowledge acquisition [44]. Knowledge acquisition refers mainly to the information of the near future (1sec-1min) trajectory of possible encounter to the own-ship surface vehicles. Furthermore, a quantitative risk assessment of the possible collision avoidance manoeuvre can be automatically produced by the obtained marine data [45]. A similar study of how to provide suitable navigation information for the mariners and a risk analysis of evasive manoeuvres is proposed [46, 47].

In conclusion, the lawful collision free guidance of a ship requires a highly trained and experienced officer in charge [48] who will minimise the human error in ship navigation by utilising appropriately ARPA, knowledge acquisition and safety domain processing intelligent support systems. On the other hand, intelligent algorithms for CA may possibly suppress the navigational error to zero, since they approach the collision avoidance problem in a more objective way than humans. Nevertheless, the real-time demands of the collision avoidance navigation, the vastly poor (compared to humans) pattern recognitions performance and the one-dimensional or non existed communication operations among these algorithms

(between them or with humans) place them far below of the average piloting crew collision avoidance abilities. From all the above challenges the autonomous navigation algorithms have to face, in this study, we focus on improving their performance in real-time, dynamic and local environment. More specifically, we have chosen to investigate and improve the performance of Artificial Potential Field Algorithms due to their simplicity and elegance.

2.2 Water based methods for autonomous collision free navigation

The collision avoidance problem is manly solved for two different classes of methodologies, the global and the local. The global approach generates the optimal trajectory based on a selected optimization criteria e.g. path length, safety etc. In this case, the geometry of the surrounding environment (obstacles) between the starting point and the target destination (or waypoint) is fully defined (known environment). The algorithms restricted to this class of solution have a major disadvantage; they are computationally intensive. For this reason, they are only used for off-line path planning and can't be used for real-time obstacle avoidance in an unpredictable static and/or dynamic environment. On the other hand, the algorithms based on the local approach utilise partial knowledge of the surrounding environment and the trajectories don't guarantee global optimality and which sometimes they encounter local minima [7]. The advantage of these algorithms is their ability for on-line path planning, since they are not computationally intensive.

The algorithmic encoding of the above methodologies can be grouped into the following three categories:

- **Mathematical models and algorithms**. The mathematical models refer to the precise mathematical description of ships' dynamics and its neighbouring environment. The mathematical algorithms use a sequence of strict definitions to solve the collision problem. In other words, these mathematical algorithms can be described as measuring algorithms to solve the collision avoidance problem in autonomous ship navigation.
- **Soft computing - Evolutionary algorithms, Neural Networks and Fuzzy Logic**. The part of Artificial intelligence that consists mainly of Neural Net-

works (NN) [49-51], Fuzzy Logic [52, 53], evolutionary algorithms [54-56] and expert-systems [57]. Neural Networks are famous for their unique learning capabilities. Fuzzy logic can simplify complex computations due to its high mathematical abstraction. Evolutionary algorithms approach the CA problem by exploiting their optimisation capabilities. These properties can aid ship CA.

- **Hybrid autonomous navigation systems.** Hybrid autonomous navigation systems [58, 59] propose a possible optimal combination of all, or a subsection of the above methods for collision free ship navigation.

2.2.1 Water based collision avoidance mathematical models and algorithms

Currently, collision avoidance problems at sea have also been addressed by mathematical models and algorithms. These models and algorithms simulate a variety of factors that influence water vehicle based collision avoidance, such as, ship's dynamics, ship's vector of motion (map location, speed and direction), ship's manoeuvres and trajectories, etc.

Most of the ship's dynamics mathematical models [17] consider the ship to have six degrees of freedom as shown in figure below.

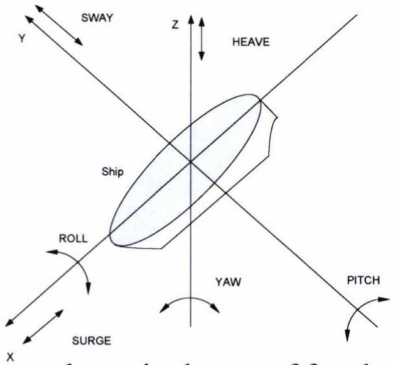


Figure: boat six degree of freedom

$$\begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \\ \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} 6x6 \\ Mass \\ Matrix \end{bmatrix}^{-1} \times \begin{bmatrix} Surge \\ Sway \\ Roll \\ Pitch \\ Heave \\ Yaw \end{bmatrix}$$

Matrix of the ship six degrees of freedom

The above Equation expresses the six degrees of freedom of a ship based on Newton's second law of motion. The mathematical algorithms that generate a sequence of real-time manoeuvres can be based on static, kinetic, dynamic and model of matrix nature of the problem [16], and various methods for solving them [22].

On the other hand, some other models that are focused on the safe path obtain the safe path trajectory with different methods but with almost the same assumptions. In particular, these assumptions are: first, the potential collision occurs in the

open sea (no land or water depth). Second, the target ships do not change their velocity (speed and course). Finally, the encountered ships do not communicate amongst themselves.

[23] proposes an Utilisation Potential Collision Threat Area (PCTA). This algorithm is founded on a single change of course AND/OR the speed of the own-ship. The principle of the PCTA is that the vector of the own-ship is outside the dangerous area that is defined. The above model forms a general guideline for a ship to follow the safe path but not the optimal trajectory. An optimal trajectory method proposed by [22] dictates that a series of delicate evasive manoeuvres of the own-ship has to be performed. The problem of evasive manoeuvres is modelled as a non-linear programming task. The non-linearity of the tasks depends on the kinetics of the own-ship. The overall approach of the model is that the safe course deviation is based on the nonlinear admittance restrictions. A more specific own-ship course optimisation model for a predefined marine environment can be achieved by point-mass models for ship motion [60]. An alternative model for ship get away manoeuvres in confined waters is also the rigid-body dynamical model for ship motion [61, 62] based on stochastic optimum control [63]. On the other hand, a specialised model for collision free overtaking is proposed [64].

More general methods for collision avoidance include the modelling of the own-ship and its immediate environment [65]. For example, [18] proposes a tanker realistic model for collision avoidance manoeuvres between strange-ships and other offshore installations and obstacles.

The collision avoidance problem can be seen from a different angle, like the Line of Sight Counteraction Navigation (LOSCAN) algorithm [10] in which the problem of two ship encounters is solved by reversing the idea of a traditional missile proportional navigation, recognising that the target is to avoid the strange-ship. The main concept of the algorithm is to generate acceleration commands in order to increase the misalignment between the ships relative velocity and the line-of-sight.

Finally, in the classical mathematical methodologies for collision avoidance we can include the potential field algorithms that we will extensively review in section 2.4.

2.2.2 Soft computing techniques for water based collision avoidance

2.2.2.1 Hybrid systems

Ships form a non-linear and ill-defined system. This means that ships and their immediate environments constitute an extremely complex system to be described in terms of precise mathematical models. Even if we have precise mathematical models for an autonomous system, the time for the decision making of the system will potentially not be reasonable for a real-time application under, for example, severe weather and navigation conditions. For the above reasons, a fuzzy approach [52] to navigational systems has been considered. However, sometimes fuzzy logic itself is insufficient, so neuro-fuzzy or fuzzy hybrid expert-systems come to complete the picture of the autonomous ship navigation challenge. A general explanation of fuzzy logic expert-systems and rule-based control is presented in [57].

Neural Networks [50, 66], have succeeded in many applications [51], including 2 dimensional robot navigation [67], with their distinct ability to learn. In addition, NN with the aid of Fuzzy Logic can form neuro-fuzzy systems. This combination of neural networks and fuzzy systems is proposed by [68]. This intelligent guidance system is based on the introduction of neuro-fuzzy networks multi-step ahead predictor for ship obstacle avoidance. The approach is generic, includes the line of sight concept and its use can be extended to aircraft, and missile guidance problems where the dynamics change significantly and unpredictably. After all, a data-fusion algorithm generates the desired waypoints of the own-ship route.

Going further to more complex hybrid systems for ship autonomous navigation, it is necessary to introduce the Potential Field Algorithms (PFA) that originate from the mobile robots research [2]. The PFAs and the hybrid system that incorporate those algorithms are reviewed in detail in section 2.3 of this chapter.

Finally, a hybrid system for collision avoidance and track-keeping is proposed by [59]. These systems combine Fuzzy logic, expert systems and state space H_∞ [59]. The collision avoidance is carried out by the fuzzy expert system. The system utilises a knowledge-base of facts and rules with the aid of an inference engine. The inference engine is also responsible for the simulation of the expert system decisions for ship collision avoidance. Finally, a robust state space H_∞ controller guides the autopilot safely on the route that is predetermined by the fuzzy-

expert system. The main purpose of the state space H_∞ control is to keep the closed-loop system stable by the use of an optimal control law. This control law can be defined by a transfer function involving exogenous inputs (weather, seaways traffic) with endogenous ship control actions maintaining the control actions minimal. At the end of the algorithm's routines, the H_∞ autopilot system materialises the avoidance action dictated by the fuzzy collision avoidance expert system under the worst exogenous systems inputs.

2.2.2.2 Evolutionary algorithms

An alternative technique for collision avoidance involves the utilisation of evolutionary techniques in the framework of *evolutionary computation* [69], which are a collection of stochastic optimisation algorithms loosely based on biological evolutionary theory of Charles Darwin [70]. Evolutionary computation is in general, an optimisation tool. More specifically, it is a search strategy for an infeasible large search space. In addition to the above, evolutionary algorithms have proved their potential for solving complex real world [27, 54] problems. The underlying generic principle of these techniques is based upon the "survival of the fittest". For instance, in collision avoidance, they maintain a population of assign paths, and through a process of variation and selection, find a near-optimum solution. Finally, they also constitute an interesting category of heuristic search [71], which is also to aid the autonomous ship navigation. We define heuristic search as the technique that does not follow a strict algorithmic solution to a problem. More specifically, heuristics are a group of effective rules guiding a system to perform search in a problem space.

The category of GAs has been widely employed successfully in mobile robots [55] where ship collision free navigation is analogous to the similar problem of safe navigation of a mobile robot. Therefore, the evolutionary method named evolutionary planner navigator (EP/N) system originally designed for mobile robot evasive steering and path generation in predefined environment presented in [56] is potentially useful for ship navigation and it has been further modified for ship encounter free navigation in [24]. This study adds the concept of time to the system. The element of time allows evaluating the system's behaviour and performance under real-time constraints (e.g. moving obstacles). Finally, it introduces own-ship variable speed, so a safe-path of a ship can be tracked in dynamic or static environ-

ment. All of the above variables can be controlled and optimised by the aid of evolutionary theory by using genes to form chromosomes. Each gene contains information such as the ship coordinates. In the above study, each chromosome has a variable-length sequence of genes. These genes specify: the coordinates (x_i, y_i) of the turning points, the speed of the ship and interconnections between the genes within the same chromosome. A more simplistic model of collision avoidance based on a single gene approach is proposed by [72]. This gene contains only the geographical position of the ship (latitude, longitude). On the other hand, a more realistic genetic algorithm for ship collision avoidance is proposed by [27]. This genetic algorithm coding introduces a gene with additional information to that in the conventional coding. These genes include, additionally to the position and speed of own-ship described in the above, the weather conditions, which are defined as "noise" (wind, wave and sea current). The mathematical representation of every chromosome is shown in equation 2.

$$x^n = (s_1{}^n, s_2{}^n, ..., s_m{}^n), \quad 1 < m \leq k \quad \text{(Eq. 2)}$$

Where:

- x is the chromosome,
- s is the gene,
- K is the length of the chromosome (number of genes in one chromosome).

In this system, the chromosome length is also a variable quantity and it is strongly dependant on the navigation conditions, (weather or traffic density). In other words, when no ship or obstacle is in the own-ship vicinity, the chromosome for the ship's safe trajectory contains only two genes, which is a straight line between the planned route waypoints. On the other hand, when additional obstacles that pose a potential threat of collision are added to the ship's neighbourhood, the enhanced system is modelled by the introduction of two genes in the variable length chromosome. A representation of the chromosomes of the study [27] is shown in Figure 3.

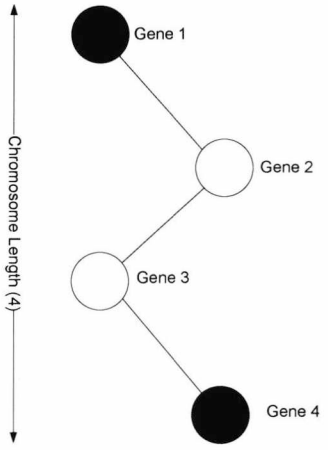


Figure 2-2: Chromosome structure of study [51]

The black genes of **Figure 2-2** represent the current and the target ship location. They are parts of the planned route. The white genes are randomly selected within the boundaries between the two black ones.

2.3 Potential Field Methods

The Potential Field concept is one of the classic methodologies for local autonomous navigation and manipulators control[8, 73-75]. The simple principle of the Potential Field Method (PFM) for the autonomous navigation of mobile vehicles is the combination of imaginary forces. These forces act upon the robot so that their resultant force performs simultaneously both steering and collision avoidance. The PFM has been initially suggested by [73] in 1983 and in 1985 [76]. In this approach, the collision avoidance problem is distributed between different levels of control, allowing the autonomous vehicle to deal with real-time dynamic local environment.

Initially, the collision avoidance problem was centred on the development of collision free path planning algorithms [77-79]. These algorithms involve high-level control and provide the low level control with a path free of collision. This way the low-level control is limited to execute elementary operations of Track-Keeping. The disadvantage of this method is that the autonomous vehicle interaction with the unknown and/or dynamic local environment is constrained by the time-cycle of the higher control, which is a magnitude slower than the low-level. Consequently, the autonomous vehicle real-time reaction is inadequate for collision avoidance in a complex evolving environment. In contrast, the Potential Field Methodology greatly

extends the low-level control to perform more complex operation by combining the environment sensing feedback with the low level of control.

This methodology was not initially intended to replace the high-level operations or to solve planning problems. The focus was on the real-time "spontaneous" autonomous vehicle reaction (reactive control algorithm)[80] in an unknown and/or dynamic environment e.g. between two waypoints defined by a high-level planning algorithm.

The essence of the simple potential method for collision avoidance is a differential equation that combines the autonomous robot and its environment in a unified system. The general concept of the potential field algorithms is illustrated in Figure 4. Potential field related forces are illustrated. $\vec{F}_T$ is the vector of the force between the autonomous vehicle and the desired waypoint or target destination. $\vec{F}_R$ is the vector of the force between the obstacle and the autonomous vehicle. The resultant force $\vec{F}_P$ from the above two forces is the direction of the ship for obstacle avoidance.

The potential field algorithms rapidly gained popularity in autonomous navigations for the purpose of collision avoidance. This popularity lies on the fact of potential field principle of simplicity and elegance. It is important to note that in mathematics a solution to a problem is elegant when it is computationally economical and general (not ad-hoc solution). A study on the historical perspective of mathematical elegance is presented in [81]. In this sense, a simple PFM for collision avoidance can be developed in a short period of time, it is computational efficient, generates naturally smooth trajectories and provides acceptable results. On the other hand, PFM algorithms experience two major drawbacks, local minima and trajectory oscillations that we will discuss in section 2.3.3 of this chapter.

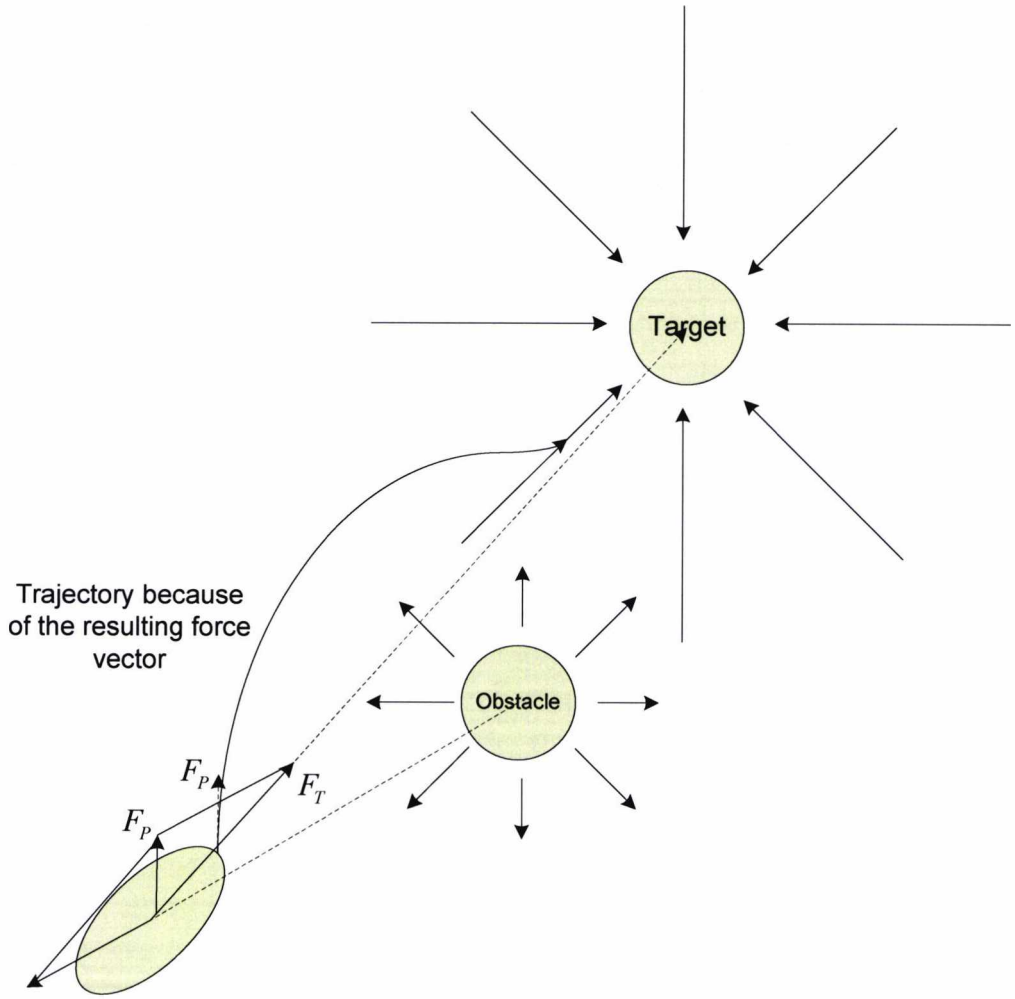


Figure 2-3: potential field algorithms general concept illustration for collision avoidance

2.3.1 Local and global potential fields for static environment

Initially, local methodologies employed the potential field methods (PFM) to be used as on-line path planning algorithms due to computational efficiency. In general technical terms, this type of algorithms for on-line path planning- fill the local Cartesian workspace with repulsive potential function that represents the obstacle to be avoided, and usually constant magnitude vectors pointing in the direction of the navigation goal[82]. We have to underline that in most occasions the control vector is exclusively calculated for only the instantaneous location of the robot within this local workspace.

In later years the potential field algorithms also utilised the global methodologies for off-line path planning. Broadly speaking, in global PFM the autonomous vehicle entire configuration space (c-space) is filled with repulsive potential energy functions representing the obstacles and the navigational goal with minimum energy [83]. In this case we have to underline that all control vectors are calculated based on all possible positions and directions of the robot within the configuration space. Therefore, these algorithms are difficult to be used for real-time or dynamic obstacle avoidance due to heavy computational requirements.

An example of global methodology is the harmonic potential fields (HPF) [84]; the use of this principle eliminates local minima. Nevertheless, as we will see later in the next section, harmonic potential fields can also be used in a local sense avoiding dynamic obstacle [85]. In addition, an example of the HPF has been used for ship collision avoidance in a known marine environment [86].

A similar approach to harmonic potential field algorithms is presented in [87]. In this instance the potential field is generated based on electrostatic potential that is developed through a resistor network. The resistor network represents the robots environment. This navigational space is mapped by Level mapping and binary mapping, and it is described by a scalar potential and a vector current field. The produced map is called Occupancy map. The algorithm produces an approximately optimal trajectory but the real-time capability of the algorithms depends on the cell resolution.

Note that the local methodologies can be categorised as the ones that deal with unknown static environment or dynamic environment. An illustration of a PFM solving the collision avoidance problem in an unknown static environment is presented in [88]. The method describes the environment sensed by the sensors as a polar histogram grid. In this case the algorithm employs a two stage data deduction. In the first stage the deduction is made by the histogram grid while in the second one by the polar histogram. A previous work of the same author combines potential field algorithms and certainty grids [2] for real-time collision avoidance. This combination is suitable for inaccurate sensor data (e.g. ultrasonic) and enables continuous robot motion. Nonetheless, the method suffers from local minima [7].

A different approach of dimensionality reduction for a local navigational environment is employed in [89]. The potential field was designed based on steady-state heat transfer with variable thermal conductivity. The obstacles are expressed as very low conductivity (K) while the free-space is as very high K. In this instance, the simulation results are generated in a static known environment. There is no proof that the method is viable in a dynamic environment, since time dimension is not taken in to consideration. On the other hand, the real advantage of this method is the description of the obstacles in a simple geometrical domain, despite the actual geometrical complexity of the obstacles.

A behaviour potential filed based approach for collision avoidance in local surroundings is explored by the [90]. In this occasion, the basic idea is to build an imaginary field that represents the behaviour of the robot. The algorithm connects the local and global optima while the robot moves. The method has better results than a simple PFM for static obstacles.

Another Potential Field global methodology is presented in [91], which utilises an improved Artificial Potential Field regression based search. In this case of research, the regression search is used to optimise the planned path generated by the improved Artificial Potential Field method.

To summarise, global methodologies algorithms are mainly used in well known static navigational environment [92] while the local ones are mostly used in unknown environment. A combination of both a local and a global method is presented in [93].

2.3.2 Local potential fields for dynamic environment

Moving to the dynamic local environment, we can find a less number of potential field algorithms. An interesting method is a view-time Potential field for moving obstacle, which is developed in [94]. The view-time is defined as the time period between two samples. In each sample, the position and the velocity is monitored and recorded. From this information the algorithm calculates the next probable position and velocity of the moving obstacle based on the probability density function of the obstacles' trajectory. The mobile robot is assumed to be a point-mass model while the moving obstacle a circle. This method is proven in an environment that includes maximum two obstacles.

The next PFM for dynamic unknown environment relies on harmonic potential fields that we first met in global path planning methods. This real-time path planning algorithm uses harmonic potentials for avoiding single and multiple moving obstacles [85]. The harmonic potential introduced in the global path planning eliminates all possible local minimum in a known navigational space. In these methods the Laplace's equation is solved numerically over the whole state space. This makes the process slow and not viable for real-time obstacle avoidance. In the proposed work [85], the use of analytical solutions to Laplace equations makes the real-time collision avoidance possible. Nevertheless, the harmonic potential field ensures a trajectory free of local minima only in static environment.

In addition, a potential field algorithm dealing with collision avoidance of multi dynamic obstacles depends on relative positions, velocities and acceleration is proposed in [95]. In this case, the own robot follows the goal with similar moving trends and avoids the obstacles with contrary moving trends. This method produces a reasonable but not optimal own robot trajectory.

An improved Potential Field algorithm, which is based on a cost function, is presented in [96]. The algorithm performs collision avoidance and Simultaneous Localisation and Mapping (SLAM). The cost function of this algorithm contains two

distances one from the autonomous agent to the obstacle and one from the autonomous agent to the target destination.

Finally, a hybrid Potential Field Algorithm dealing with both static and dynamic environment is introduced in [97]. In the first study, a Fuzzy Potential Field algorithm uses two Fuzzy Mamdani and TSK models to develop the total attractive and repulsive acting forces.

2.3.3 The potential field methods limitations and proposed solution

In a course of experimental work is indentified that the PFM suffer from local minima and trajectory oscillations [1, 7, 8]. A systematic criticism of PFM and the proof that these problems are inherited to all algorithms that rely on this principle takes place in [7]. The most common problems are: no passage between close distant obstacles, U shape traps, oscillations in narrow passages and oscillations in presence of obstacles.

In this section, we categorise the algorithms that minimise or eliminate the local minima in a local static and dynamic environment. On the other hand, the author of this thesis briefly presents his own work [4] on the identification of local-minima in Pure Dynamic Environment (PDE). Additionally, the mathematical algorithm that predicts and prevents these local minima is also mentioned. Finally, we also review the algorithms that mitigate the problem of trajectory oscillations.

2.3.3.1 Methods which solve or minimise potential field local minima in static environment

Initially, the research on local minima was concentrated on their identification, by searching if the potential field autonomous robot is within a local minima state. For this state to be recognized, a mathematical condition was usually used. In [2] a local minima trap condition monitors the angle between the course of the robot and the actual angle from its goal. When the angle is more than a specific value, the algorithm switches the robot's guidance from PFM to the wall-following method,

until the trap condition expires. The main disadvantage of this method is that the robot loses the collision avoidance capability of the PFM while it is within a trap state.

In later years, a potential function that unifies the wall-following algorithm with the PFM [98] is introduced. The method increases the complexity of the potential function from scalar potential field (SPF) to a vector based (VPF). In this way, it minimises the local minima compared to SPF, by circulating the closed equipotential contours. The VPF of [98] is tested in static known environment but potentially, it can be applied in unknown and dynamic environment due to low computational requirements. Nevertheless, the algorithm can experience trajectory oscillation, in particular, while the autonomous vehicle is crossing large obstacles in clattered navigational environment.

A different approach to the local minima problem comes in life by a path planning PFM [99]. This algorithm has a deterministic method for escaping the local minima by relating the free space with a global skeleton defined by a number of local methods. The skeleton curves are the loci of the maxima of the potential field function, which it is proportional related to distance between the autonomous vehicle and the obstacles. The local minima are avoided by following the local maxima of potential field function, as well as, by taking slice projections through critical points. This method is comparable to Voronoi Diagram path planning and on most occasions has better results.

There is a number of solutions for local minima that focus on a particular area of the problem. A potential function specially designed to solve the specific problem of local minima by the name: goal non reachable with obstacle nearby (GNRON) is presented in [100]. For this reason, the algorithms' potential function takes in the consideration the relative distance between the robot and the goal. This way verifies that the goal is always the global minima, as well as, taking the shortest distance to the goal, if we compare it with the routing distance generated by a conventional potential field method. Then the selection of the potential function parameters is critical, so the function is free of local minima. The method is partly verified by simulated results in a known static environment.

Most of the above algorithms solve the PFM local minima by modifying the potential function. In an alternative way a class of a cost functions was created by [101] to guide a point mass robot within a number of spherical bounded obstacles in Euclidean n-space. A subclass of the above functions is defined, in a way that the actual mechanical system will inherit the behaviour of the gradient lines of the cost function. Later work of the same authors fully define the collision avoidance amongst obstacle in spherical domain [101, 102]. In this generalised sphere world each obstacle is represented by an arbitrary number of disjointed disks attached to a larger disk. One-parameter family of this type of sub-functions guarantees elimination of local minima only in a perfectly known stationary environment.

A potential field method for path planning that relies on Laplace's Equations and it doesn't exhibit local minima by nature is developed in [84]. The origin of the harmonic function for autonomous navigation took place in early 90ies [103, 104] and satisfies the min-max principle. In this approach the properties of harmonic functions (solution of the Laplace Equations) are utilised. This technique for local minimum solution is also possible by using finite element method [105], which discretizes the configuration space to compute the potential function. This type of algorithms generates smooth trajectories for the guided autonomous vehicle and eliminates spontaneous creation of local minima. Nevertheless, their solution requires much higher processing power than a standard PFM [1, 8] for this reason it could face real-time collision incapacity in a dynamic or unknown environment. The computational time relates to the grid sizes, since harmonic functions decay rapidly. This has an impact to the system's performance in large navigational regions. A possible solution to this problem could be an efficient segmentation of the configuration space [86].

Finally, a behavioural based solution for local minima is presented in [90] named the behaviour information potential field (BIPF). This is a path planning algorithm, which relies on local environment representation by a 2-D Cartesian Histogram Grid and behavioural rules build from the imaginary navigation field. The method connects the global optima with the local optima by the BIPF. The method is able to

solve a local minima trap in a case of closely placed or overlapping obstacles in static unknown environment.

2.3.3.2 Methods which predict, solve or minimize potential field local minima in dynamic environment

The identification and definition of the local minima is firstly discussed in [4], this is part of the author's work that is explained extensively in chapter 4 and 5. In this study for first time we identify, define and predict local minima of the PFMs in Pure Dynamic Environment (PDE) of local multi agents/vehicles. The navigation collision scenario focuses on the case that the algorithm guides each agent/vehicle independently although the algorithm principle is identical for each one of the agent/vehicles. This approach is a combination of a novel rule-based mathematical algorithm and the AWSPPF Algorithm, which based on the Virtual Force Field (VFF) [2] navigational method. The need for the above combinational algorithm is due to PFM inability to guide efficiently multi-autonomous agents/vehicles in the same environment due to a Trajectory Equilibrium State (TES). TES is a new state that dramatically degrades the performance of the PFM and is firstly identified and defined within this study. The algorithm is extremely efficient in both computational and trajectory economy terms. In addition, in [106] we address the generic nature of the above novel Potential Field Algorithm, which allows it to be efficient, not only when applied to multi-autonomous agents, but also when applied to collision avoidance between a single autonomous agent and an obstacle displaying random velocity.

2.3.3.3 Methods which eliminate or minimize the trajectory oscillation of the potential field algorithms

Finally it is important to mention the inherent trajectory oscillation problem of PFM, especially in narrow passes and nearby obstacles the problem has been identified in [2, 7]. For the smooth operation of PFM a low pass filter for steering is employed in [2]. The filter response depends on the grid resolution and the sampling period. When an overly energetic steering control takes place a digital low pass filter limits the steering-rate command.

In addition a modified Newton's method is proposed [107] to solve this problem by adjusting the gradient. The gradient operation is forced by finding suitable direction to quadratic approximation than linear as the gradient descent method suggests. In this study the trajectory oscillation greatly improved in the considered scenarios. Nonetheless, the environment was known and static since the method requires high computational time.

Finally, a PFM in combination with improved Kalman filter is presented in [108] to remove trajectory oscillations due to sensors' disturbances from environmental noise.

2.3.4 Potential field methods in marine navigation based on mathematical, hybrid and protocol based systems.

Potential field methods have been also used for marine autonomous navigation. The potential methodologies adopted in marine environment are mainly harmonic potential field, Virtual Force Field (VFF) [2] and a combination of them with AI to form hybrid systems.

A harmonic potential field method (HPFM) for autonomous ship navigation is presented in [86]. In this study the HPFM is responsible to autonomously navigate the own ship in constrained water areas and tracks. It is tested in two water situations: traffic separation scheme and narrow channel. With this method a single ship collision can automatically follow the sea navigational regulation, while avoiding a static obstacle.

A simpler on-line PFM named virtual force field (VFF) has been widely used for local navigation both as a stand alone algorithm, and as a foundation to a series of hybrid [109] or mathematical navigational algorithms [110]. In ship autonomous navigation, [53] introduces a fuzzy logic autonomous navigation algorithm based on Virtual Force Field (VFF), which satisfies COLREGs. For the purpose of autono-

mous ship navigation, a Modified Virtual Force Field (MVFF) is used. This method is suitable for both track-keeping and collision avoidance. Furthermore, the algorithm has the ability to handle an immediate static and dynamic environment. From Figure 2-4, we can see the VFF concept of two forces at any given point in time. It is apparent that the VFF concept cannot provide a track-keeping capability. The Modified algorithm provides true track-keeping as well as collision avoidance in static and dynamic ship environment within COLREGs guideline. This is achieved by the addition of a perpendicular force $\vec{F}_p$ to the desired course as shown in Figure 2-4.

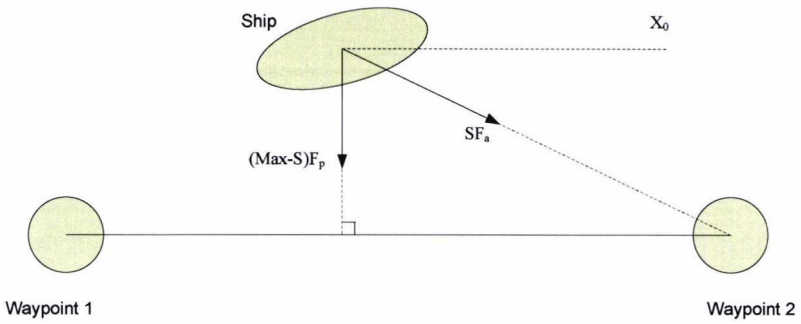


Figure 2-4: Modified VFF with the addition of $\vec{F}_p$ force vector

Another category of hybrid systems for collision avoidance is proposed by [11]. This technique is based on heuristic search reinforced by fuzzy relational products [111] and COLREGs. Generally, COLREGs define a guideline for collision avoidance between two ships. On the other hand, the applications utilising the heuristic search technique can cope with many-ship encounters, but at the same time they introduce an inclination to violate COLREGs. The inclination of COLREGs to be violated in a multi-ship environment can be minimised by the method of [11].

Finally, a new reactive navigation technique based on Artificial Potential Fields for autonomous sailboats is explored in [112]. This technique addresses both sailboat complex kinematics and the unpredictable nature of the trust force.

2.4 Conclusion

Collision avoidance is a complex multi-task problem. The degree of complexity depends on the state of exogenous (weather and waterways traffic density) and endogenous (ship type, onboard technologies, etc.) navigation influencing factors.

In comparison to "intelligent" machines, humans navigate ships satisfactory. The degree of human navigation ability depends on both the level of experience and the psychological status of each individual. On the other hand, human beings are highly subjective, and this can lead to accidents. The subjective nature of humans is sometimes reinforced by the ship intelligent support systems (GPS, Radars, ARPA, etc.) They provide the piloting crew with additional navigation information that can reduce the sense of collision danger and lead to COLREGs violation. The international rules for collision avoidance at sea (COLREGs) have been laid down to minimise the subjective nature of humans. Even if COLREGs are fully defined, the human interpretation of them is still subjective since ship navigation manoeuvres are performed in real-time, and sometimes, under demanding exogenous inputs.

The subjective nature of humans can only be removed from ship navigation when humans are no longer responsible for ship navigation. A variety of experimental water based and mobile robots algorithms navigation systems have been briefly explained in this chapter.

1. Most of the mathematical collision avoidance methodologies are effective when the exogenous inputs are not extreme. In case of extreme exogenous input the ship dynamics introduce non linearity, and computation complexity. For the mathematical collision avoidance algorithms to perform in real-time a level of environment and dynamic model abstraction must take place. One of the most important classical methodologies for collision avoidance is potential field methods. This method naturally supports environment abstraction, as well as, it greatly extends the low-level control to perform more complex operation by combining the environment sensing feedback with the low level of control. This has as effect these algorithms to unify kinematic planning with the dynamic exe-

cution problem. On the other hand, PFM algorithms experience two major drawbacks, local minima and trajectory oscillations. In this chapter we have reviewed a number of methods that either minimize or completely solve these problems.

2. The effectiveness of evolutionary algorithms for ship autonomous navigation depends on the optimisation method that is followed. The optimisation method depends on the fitness function [54]. The fitness function is fixed for the whole algorithm and sometimes is not sufficient for ship navigation under all different exogenous conditions. This can also lead to inefficient or random system performance.
3. A Combination of technologies such as Neural Networks, Fuzzy Logic, expert-system and mathematical algorithm can form a ship autonomous navigation system. These systems aim to use the advantages inherent within each component technology.
4. Hybrid systems look very promising. But, they require a high level of intelligence to harmonically merge the different AI technologies together. On the other hand, Machine intelligence is not absolutely proven up against human intelligences. Within the hybrid systems we find in a number of occasions the potential field algorithms due to their elegance and simplicity..

Chapter 3

3. Potential Field Algorithm Design for the Evaluation of the Potential Field Methods Performance in Pure Dynamic Environment (PDE)

In this chapter we design and implement a Potential Field Method (PFM) that purposefully accommodates the main features of a classical PFM [7]. Moreover, it incorporates processing power oriented improvements by design change of the Active Window (AW) that we analyse and justify in section 3.2. We use this algorithm to evaluate the PFM capabilities in the newly defined Pure Dynamic Environment (PDE), without the overhead of the dynamic/kinematic models of the autonomous agent. In this way the collision avoidance agent capabilities can be isolated and studied individually. The method of the collision avoidance capabilities examination is explained in the next chapter and is based on the novel concept of Monovular Agents Correlation.

In the above paragraph, we have referred to PDE that we define for first time as the environment that contains only dynamic agents and/or obstacles. A combination of dynamic and static agent/obstacles is not allowed at the beginning of the collision scenario. A static state of an agent or obstacles at this environment is only acceptable when derives from a previous dynamic state. In this manner, the possibility of a future static agent state caused by a previous dynamic state is also taken into consideration. As we will see in the next chapter this part of the definition permits us to identify local minima of an agent in PDE.

In this study we have focused on the performance of PFM in PDE, since for the purpose of safe collision avoidance in PDE two main requirements have to be satisfied:

- Low processing power requirements, since dynamic environment is more demanding than static.
- The collision avoidance trajectory to be realistic in relation to own agent dynamic model with minima agent modeling. In other words, the PFM

generates realistic smooth trajectories, so the Potential Agent can follow instantly the generated track segment in safe critical situations.

Therefore, we have selected to evaluate the PFM among other path planning and collision avoidance algorithms due to its elegance and simplicity, as well as its natural smooth trajectories generation.

In more detail, as we have referred in the previous chapter, the most important of PFM advantages are the following:

- Elegant algorithm, which means generic nature and processing efficient
- Simultaneous steering and collision avoidance
- Collision avoidance capability is not limited of the time cycle of higher levels of control
- It generates natural smooth trajectories that spontaneously accommodate the autonomous agent's dynamic capabilities with only minor modifications.

On the other hand, as it has repeatedly been underlined in many studies, the PFM are mainly suffering from two major drawbacks, local minima and trajectory oscillations. Nonetheless, none of these studies have examined and define the existence and the reason of local minima in PDE. We need to identify the reason causes local minima in PDE, since this way we can vastly improve the performance of the PFM as we will analytically prove in chapter 5.

For that reason, therefore, in this chapter we design a PFM that accommodates all major advantages and disadvantages of the classical PFM. In addition, it has an improved Active Window shape; it is generic and free of agent specific dynamics and sensory characteristics. In this way, the algorithm's collision capability can be monitored and evaluated accurately. We have named this algorithm Active Window Single Point Potential Field (AWSPPF) Algorithm. The algorithm is based on a point-mass model, and all the attractive and repulsive forces are calculated based on a single point, which represents the agents/obstacles centre of gravity.

The above algorithm is based on the classical Potential Field function that we analyse in section 3.2.2 and on the concept of Active Window. The decision for the design basis for this algorithm is based on the combination of the Potential Algorithm review of chapter 2 section 2.3 of the previous chapter in combination with the PFM comparison for collision avoidance in PDE of section 3.1 of this chapter. As we mention previously, the Active Window is improved and mathematical justification of the Active Window new shape is presented in section 3.2.1.

In section 3.2.2 we analyse the AWSPPF Algorithm mathematically, and in section 3.2.3 we analyse how we can manipulate the Minimum Distance (MD) of the agent of any obstacle within the working space. Finally, in 3.2.4 we present the software design and the implementation of the algorithm, as well as its performance results in both static and dynamic environment.

3.1 AWSPPF Potential Field algorithm Design for the Evaluation of the PFM performance in Pure Dynamic Environment (PDE)

In this section, we justify the PFM algorithms we have selected as a base for the design of the Active Window Single Point Potential Field (AWSPPF) algorithm. We use this algorithm in order to evaluate the PFMs performance in PDE. This selection is based on the review of the PFMs of section 2.3 chapter 2. In this section we have reviewed the following main categories of Potential Field Algorithms:

- Local
    - Static environment
    - Dynamic environment
- Global

For the purpose of this study we focus on local PFMs, since all global are mostly used for off-line path planning and are computational demanding. Therefore, these algorithms are not suitable to be used for real-time and/or dynamic obstacle avoidance.

In the local algorithms we can find two main categories, PFM for static environment and PFM for dynamic environment. Nevertheless, in this study we focus on the PDE, a reference to PFMs for static environment is important, since a number of dynamic environment algorithms are based initially on a static environment ones.

An illustration of PFM solving the collision avoidance problem in an unknown static environment is presented in [88]. A previews work of the same author combines potential field algorithms and certainty grids [2] for real-time collision avoidance. This combination is suitable for inaccurate sensor data (e.g. ultrasonic) and enables continues robot motion. Nonetheless, the method suffers from local minima [7].

A different approach of dimensionality reduction for a local navigational environment is employed in [89]. The potential field was designed based on steady-state heat transfer with variable thermal conductivity. There is no proof that the method is viable in a dynamic environment, since time dimension is not taken in to consideration. On the other hand, the real advantage of this method is the description of the obstacles in a simple geometrical domain, despite the actual geometrical complexity of the obstacles.

In this algorithm again the basic concept of PFM is used in combination of the obstacle description as a steady-state heat transfer with variable thermal conductivity.

All the above algorithms share a similar and calculation efficient Artificial Potential Field algorithm based on the principle of the classical PFM with the main difference of the obstacle description, world model segmentations and processing. Nevertheless, these algorithms still suffer from local minima.

A behaviour potential filed based approach for collision avoidance in local surroundings is explored by the [90]. In this occasion, the basic idea is to build an imaginary field that represents the behaviour of the robot. The algorithm connects the local and global optima while the robot moves. The method has better results than a simple PFM for static obstacles.

This algorithm is computational intensive, since the global and local optima have to be calculated prior to the path planning.

Moving to the dynamic local environment, we can find a less number of potential field algorithms. An interesting method is a view-time Potential field for moving obstacle, which is developed in [93]. The view-time is defined as the time period between two samples. In each sample, the position and the velocity is monitored and recorded. From this information the algorithm calculates the next probable position and velocity of the moving obstacle based on the probability density function of the

obstacles' trajectory. The mobile robot is assumed to be a point-mass model while the moving obstacle a circle. This method is proven in an environment that includes maximum two obstacles.

The algorithm is based on a PFM that both attractive and repulsive forces are variable based on the own agent distance from the target destination and the approaching obstacle. The repulsive force is applied to the predicted dynamic obstacle location based on its current velocity and direction. The prediction is based on the concept of random walks and probability density function (PDF).

This algorithm is tested with manually adjusted of Potential Field parameters as: field gains and view periods. The algorithm is not tested against its base potential field algorithm so its performance improvement is not stated. In addition, there is only one type of collision avoidance scenario, which is not categorised and fully defined.

The next PFM for dynamic unknown environment relies on harmonic potential fields that we first met in global path planning methods. This real-time path planning algorithm uses harmonic potentials for avoiding single and multiple moving obstacles [85]. The harmonic potential introduced in the global path planning eliminates all possible local minimum in a known navigational space. In these methods the Laplace's equation is solved numerically over the whole state space. This makes the algorithm processing intensive and not viable for real-time obstacle avoidance. In the proposed work [85], the use of analytical solutions to Laplace equations makes the real-time collision avoidance possible.

Nevertheless, the analytical harmonic PFM has to perform obstacle approximation for the harmonic functions' boundary conditions, it can only solve the dynamic collision avoidance problem based on a series of instantaneous static environments. Furthermore, harmonic PFM cannot predict local minima due to kinematic characteristic of the dynamic environment. The algorithm's processing requirements are much higher than the classical Potential Field Methodologies and strongly depend on the surrounding obstacle shapes.

From the analysis in section 2.3 chapter 2 of PFM in static and dynamic environment, we can conclude that we have the following categories related to processing power:

- The ones that are based on the classical Artificial Potential Field methodology, which are processing efficient and having problems with local minima
- The ones that are free of local minima and are computationally intensive

Since no previous investigation of the local minima behaviour took place In Pure Dynamic Environment (PDE) we have decided to methodically investigate the performance of the classical methodology of PFM.

It is important to note that in this study we don't focus on the calculation efficiency of the obstacle/agent/s descriptions or position accuracy, since we are interested in the collision avoidance behaviour of the actual collision avoidance algorithm within the PDE. The inaccuracies of the sensory data could dilute the behaviour of the Potential Field algorithm behaviour. Therefore, we use agent/s known positions and shapes to exclusively study the collision capabilities of the Potential Field Algorithm based only on agent/s' spatial and velocity correlation.

Therefore, the main requirement we have taken into consideration for safe collision avoidance in PDE among different collision avoidance PFM is the required processing power and the smoothness of the generated trajectory with any use of filtering or kinematic modelling. On this basis we have selected to use the classical potential field algorithm interpretation of [9]s based on a point-mass model with the use of a circular Active Window (AW).

The design, the mathematical analysis and the implementation of the above algorithm is described in the next sections of this chapter.

3.2 Pure Dynamic Environment (PDE) Potential Field Evaluation Method Design

For the purpose of the Potential Field Methods (PFMs) evaluation in PDE we have design a Potential algorithm that relies on the classical Potential Field Methodology interpretation of the Virtual Force Field (VFF) concept based on a point-mass agent model. In addition, the evaluation algorithm does not model obstacles based on the processing intensive certainty grid. Finally, the Active Window (AW) is not square but circular.

First of all we examine the VFF algorithm differences from the classic PFM, which are:

- The Potential Field Attractive Force is constant, and always greater or equal to the repulsive force (this minimises oscillations and simplifies the design of the original PFM definition).
- The Potential Field Agent (PFA) is located in the middle of a virtual square window. The effect of the repulsive force from any obstacle only takes place within this square window, which is named Active Window (AW). This eliminates the potential field calculations when the agent is in a safe distance from the obstacle/s.
- Obstacles are described based on the density of the Active Window grid [3], in contrast with the original PFM in which obstacles description was based on the composition of primitives [1].

As we have prior referred, based on VFF algorithm we have designed the PFM method for PFMs evaluation in PDE. This Potential Field Method has the following difference in relation to VFF:

- The active window is circular and not square as shown in Figure 3-1. As we will describe in the following section, it is not efficient to calculate if an obstacle is located within a square AW in comparison to a circular AW.

- We have removed the AW grid that describes the obstacles. The AW grid of the VFF was a certainty grid that describes the obstacles probabilistically based on range sensors modelling. In our case, we assume that the obstacles have known accurate positions at all times. The target of the research is the collision capabilities of the PFM and not the range sensor modelling. Nevertheless, the grid approach of VFF method has a second use, which is to describe the shape of the obstacle within the AW. This description is not efficient either, since requires multiple additions of repulsive Potential Field vectors to depend on the number of grid points that the obstacle occupies.
- Finally, the repulsive and attractive forces of the PFM are only allowed to take place in one point of the obstacle. In this way the shape of the obstacle cannot couple into the trajectory of the Potential Agent, and the collision avoidance capability of the algorithm is intact from the shape of the obstacle.

As we have mentioned in the beginning of this section, and based on the algorithms properties, we have named this Potential Field Algorithm: Active Window Single Point Potential Field (AWSPPF) Algorithm.

3.2.1 The Active Window Shape of the AWSPPF Algorithm

We can justify the use of Circular Active Window (AW) over the Rectangular one by examining the calculations are needed for the processor to decide if a dynamic obstacle is within: first the Square AW, and second the Circular AW.

For the square AW we assume that the angle θ between the own Potential Field Agent's (PFA) heading and the line that connects the dynamic obstacle/agent instantaneous position with the own PFA is known. Second, the D_{AWMD}, which is the minimum distance from the centre of Potential Field Agent to the boundary of the Square AW is also known.

Third, we have to calculate the Square AW border distance from the own PFA for this specific angle θ. Finally, we have to compare if the obstacle current distance

from the own PFA is equal or less than the border distance at this specific angle, as is shown in Figure 3-1.

We can calculate the above mathematically in the following steps:

1. To calculate the Square AW border distance from the own (PFA) for this specific angle θ we have:

$$D_{AWVAD} = \frac{D_{AWMD}}{\cos\theta} \qquad (3.1)$$

2. Then we have to calculate if:

$$D_{AWVAD} \geq D_{ob} \qquad (3.2)$$

Where

θ is the angle between the own PFA heading and the line connects the dynamic obstacle/agent momentary position with the own Potential Field Agent as shown in Figure 3-1.

D_{AWMD} Active Window Minimum Distance is the minimum distance from the centre of Potential Field Agent to the boundary of the Square AW as shown in Figure 3-1.

D_{AWVAD} Active Window Variable Angle Distance is the distance between the centre of the Potential Field Agent and the square active window boundary in angle θ as shown in Figure 3-1.

D_{ob} is the distance of the Potential Field Agent and the dynamic obstacle/agent as shown in Figure 3-1.

Note: the active window rotates according to the heading for the PFA. Therefore, the D_{AWMD} relates always to the heading angle.

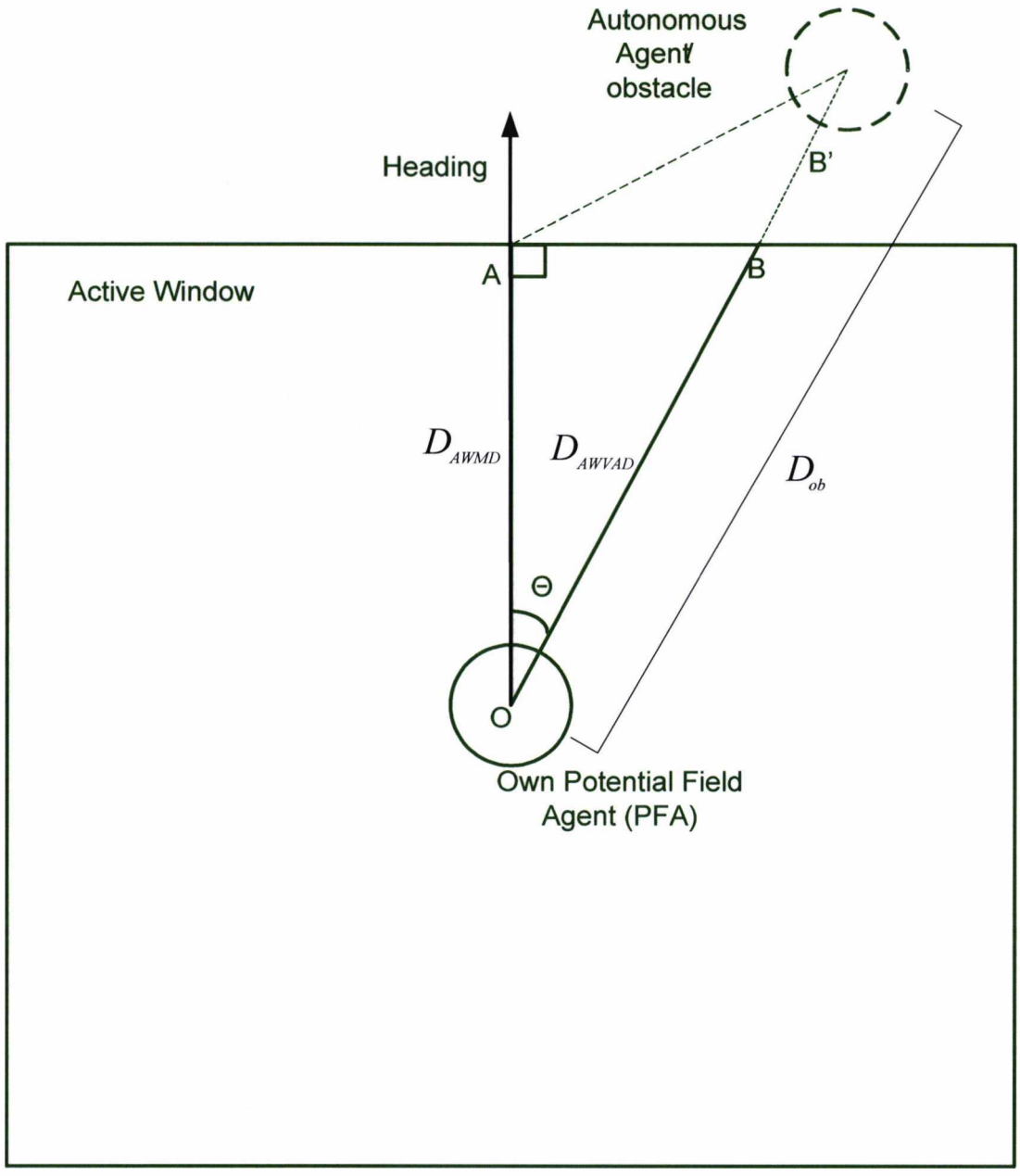


Figure 3-1: VFF Square Active Window

On the other hand, when the active window is circular we only have to calculate if $D_{AWCR} \geq D_{ob}$, since D_{AWCR} is known and always the same for any angle between the Own Potential Field Agent heading and the dynamic obstacle/agent.

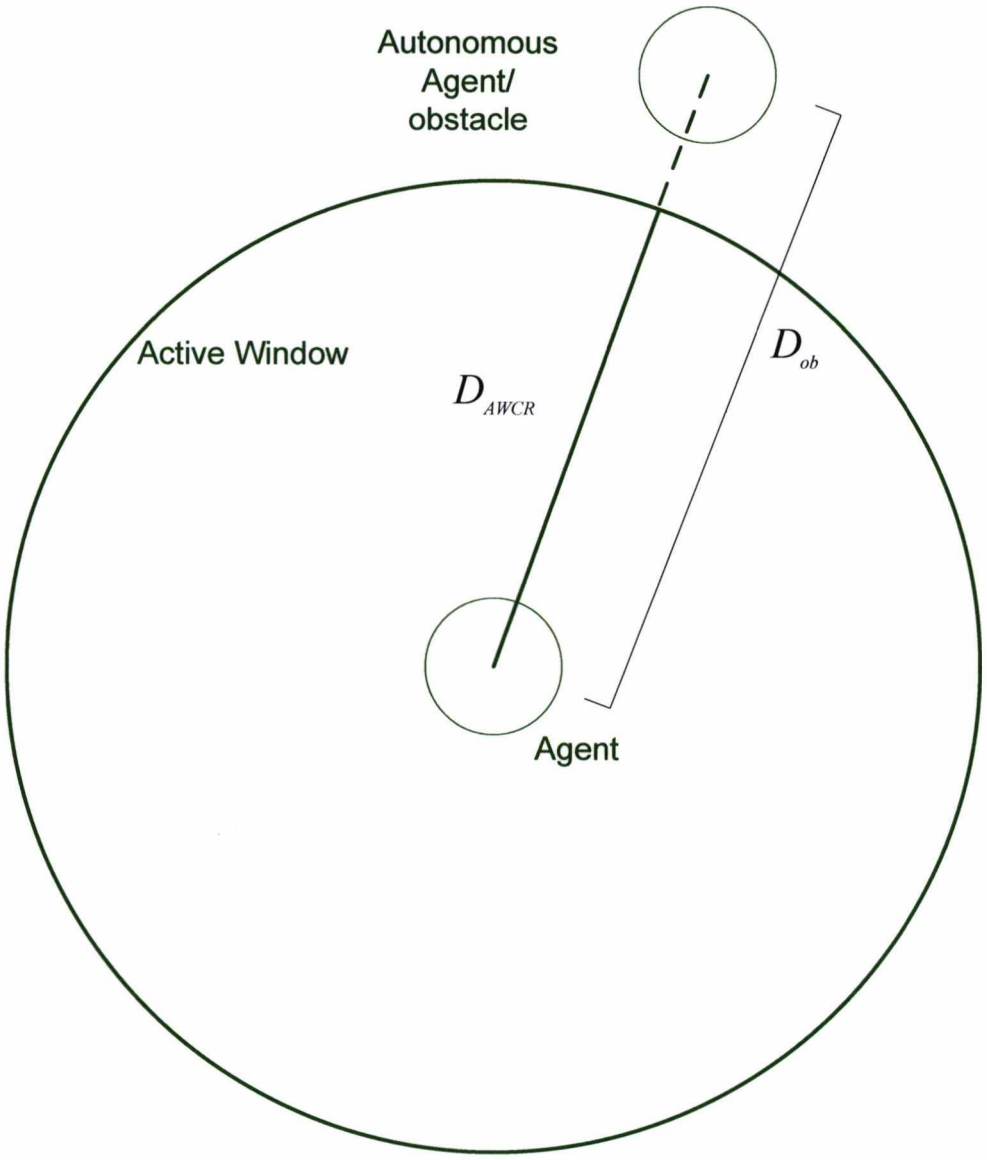


Figure 3-2: Potential Field Method (PFM) circular Active Window

Where

D_{AWCR} is the Active Window Circle Radius.

From the above mathematical comparison of the Potential field Active Windows we can conclude that the circular AW has the following advantages in PDE over the square one:

1. The processor needs less processing power or memory (e.g. use of lookup table) to determine if a dynamic obstacle/agent is within a circular AW. This

is because when the AW is circular the processor doesn't have to calculate equation **Error! Reference source not found.**(3.3) for every iteration.

2. The agent has equal distance from all points of the AW, which means that the collision avoidance safety distance modelling of the Potential Field Agent is more efficient. For example in the square AW the distance that the repulsive force vector takes places for first time depends on the heading angle between the dynamic obstacle and the own agent. This distance difference doesn't have a true meaning to the collision avoidance safety of the own agent.

3.2.2 Active Window Single Point Potential Field (AWSPPF) Algorithm Mathematical Analysis

The conceptual representation of the AWSPPF Algorithm is illustrated in Figure 3-3 below.

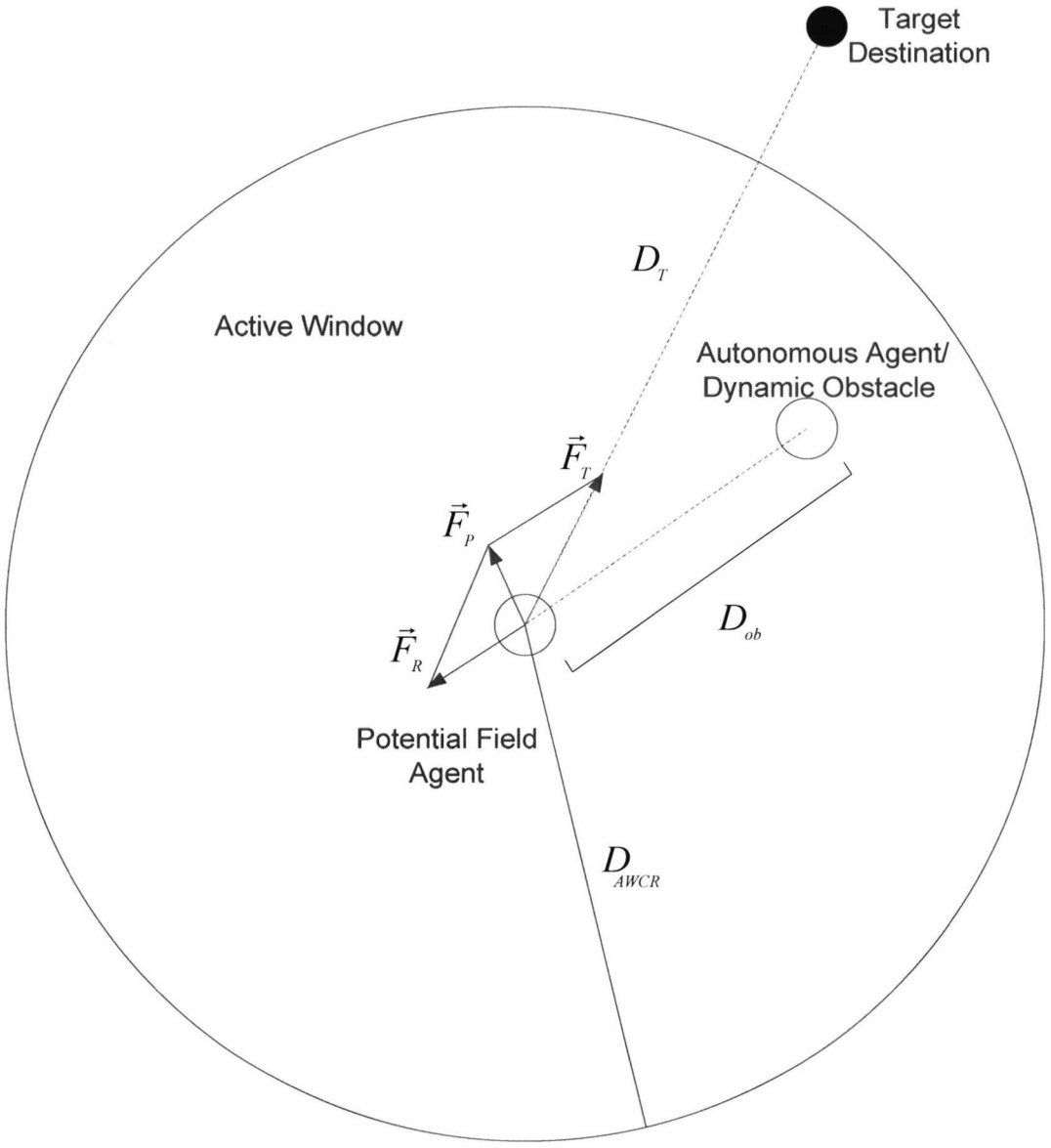


Figure 3-3: AWSPPF Algorithm concept

This Potential Field Algorithm is described by the following equations:

$$\vec{F}_P = \vec{F}_T + \vec{F}_R \qquad \text{if } D_{AWCR} \geq D_{ob} \qquad (3.4)$$

$$\vec{F}_P = \vec{F}_T \qquad \text{if } D_{AWCR} < D_{ob}$$

And

$$\left|\vec{F}_T\right| \geq \left|\vec{F}_R\right| \qquad (3.5)$$

Where

$\vec{F}_T$ is the attractive force form the autonomous agent target destination

$\vec{F}_R$ is the repulsive force due to obstacle location in the configuration space and

$\vec{F}_P$ is the direction of the autonomous agent.

More specifically the attractive force magnitude satisfies the following equation:

$$F_T = F_{CT}\left(\frac{x_t - x_o}{D_T}\hat{x} + \frac{y_t - y_o}{D_T}\hat{y}\right) \qquad \text{if } D_T \neq 0$$

$$\vec{F}_T = 0 \qquad \text{if } D_T = 0 \qquad (3.6)$$

Where

F_{CT} is the attractive force constant,

D_T is the autonomous agent distance from the target,

x_t, y_t are the target destination coordinates,

x_o, y_o are the current autonomous agent coordinates

On the other hand, the repulsive force is described by the vector:

$$\overrightarrow{F_R} = -\frac{F_{CR}W^n}{D_{ob}{}^n}\left(\frac{x_{ob} - x_o}{D_{ob}}\hat{x} + \frac{y_{ob} - y_o}{D_{ob}}\hat{y}\right) \qquad \text{if } D_{AWCR} \geq D_{ob}$$

$$\vec{F}_R = 0 \qquad \text{if } D_{AWCR} < D_{ob} \qquad (3.7)$$

Where

F_{CR} is the repulsive force constant,

W is the width of the agent,

D_{ob} is the autonomous agent distance from the obstacle,

x_{ob}, y_{ob} are the obstacle coordinates,

D_{AWCR} is the Active Window Radius

n is a positive integer.

In Equation 3-4, describes the intensity and the direction of the field that is produced by a point-mass obstacle.

We use the above equations based on the following conditions:

$$\text{When } D_T = 0 \text{ then } F_T = 0 \text{ and } F_R = 0 \qquad (3.8)$$

We have chosen to use a traditional and processing efficient speed control equation, which is derived from the equation

$$\vec{F}_P = \vec{F}_T + \vec{F}_R$$

$$\vec{V} = \frac{\left|\vec{F}_P\right|}{F_{P_{MAX}}} V_{MAX} \left(\frac{x_{ob} - x_o}{D_{ob}} \hat{x} + \frac{y_{ob} - y_o}{D_{ob}} \hat{y} \right) \qquad (3.9)$$

The magnitude of the above equation is equal to (3.10) equation and corresponds to the agent speed:

$$\left|\vec{V}\right| = \frac{\left|\overrightarrow{F_P}\right|}{F_{P_{MAX}}} V_{MAX} \qquad (3.11)$$

Where

$F_{P_{MAX}}$ is the maximum magnitude of F_P, which is equal to:

$$F_{P_{MAX}} = F_T \tag{3.12}$$

The above equation is true due to Virtual Force Field definition [9].

Therefore the equations (3.11) and (3.9) from (3.12) become:

$$\left|\vec{V}\right| = \frac{\left|\overrightarrow{F_P}\right|}{F_T} V_{MAX} \tag{3.13}$$

$$\vec{V} = \frac{\left|\vec{F}_P\right|}{F_T} V_{MAX} \left(\frac{x_{ob} - x_o}{D_{ob}} \hat{x} + \frac{y_{ob} - y_o}{D_{ob}} \hat{y} \right) \tag{3.14}$$

An example of the AWSPPF algorithms performance is illustrated in Figure 3-4. Autonomous agent avoids a circular static obstacle of 50m radius, the active window is 300m radius and the Minimum distance that we are going to explain in the next chapter is 100m. The number of times that the equation **Error! Reference source not found.** doesn't have to be calculated, if the algorithm is calculated one time every second is around 78. This number varies based on the active window radius and the number of calculation per time unit. A more comprehensive performance analysis of this algorithm in static and dynamic environment takes place in 3.2.4.

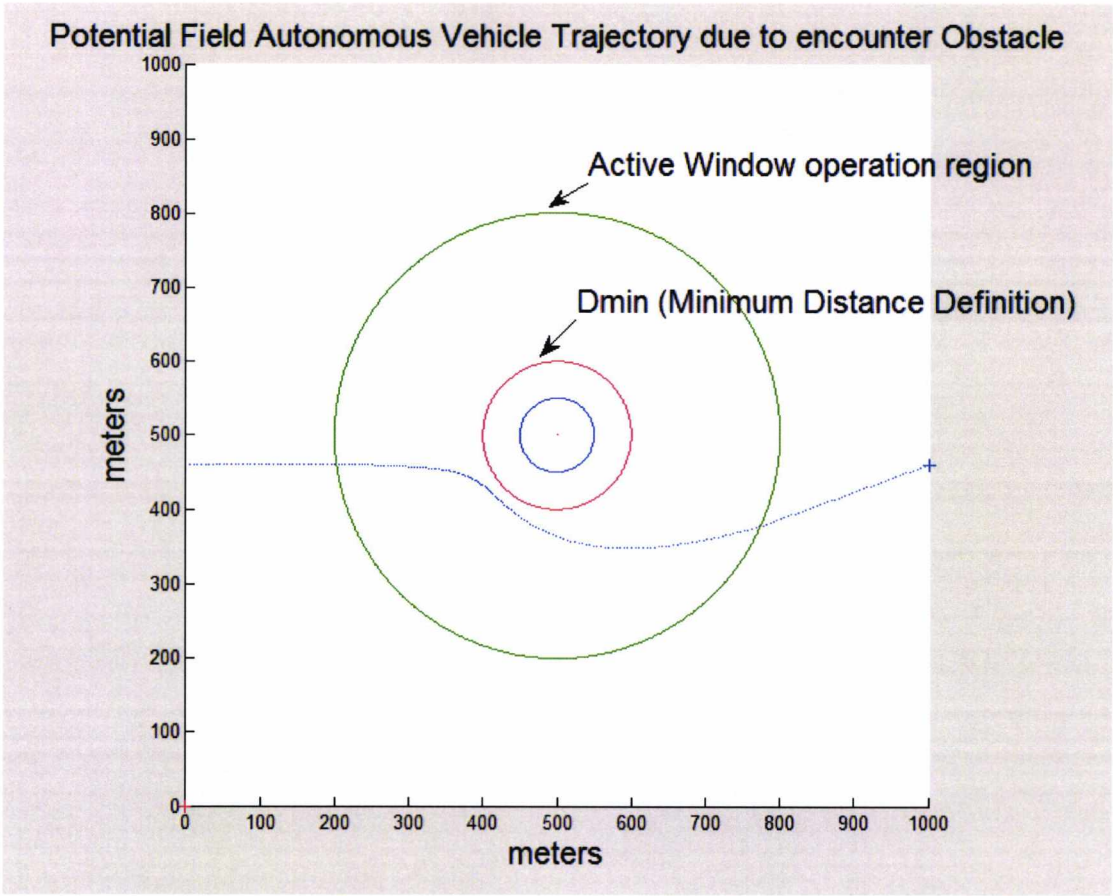


Figure 3-4: Potential Autonomous Vehicle simulation based on Circular Active Window

3.2.3 Active Window Single Point Potential Field AWSPPF) Algorithm Minimum Distance (MD) Definition

In all classical Potential Field Algorithms we have local minima when the Navigational Potential vector is equal to zero. According to the (3.14) when F_P is equal to zero the speed of the Potential Field Agent (PFA) is zero too, when the agent follows a point-mass model without inertia. Therefore, when F_P is equal to zero we also have the PFA Minimum Distance from the encounter obstacle. From the above we can determine the Minimum Distance of the PFA, since:

$$\vec{F}_P = \vec{F}_T + \vec{F}_R = 0 \Leftrightarrow$$

$$\Leftrightarrow \vec{F}_T = -\vec{F}_R \Rightarrow D_{\min} = W \sqrt[n]{\frac{F_{cr}}{F_{ct}}} \tag{3.15}$$

We can prove the above based on the equations (3.6) and (3.7) when $D_T \neq 0$ and $D_{AWCR} \geq D_{ob}$.

Therefore we have:

$$\vec{F}_R = \frac{F_{CR} W^n}{D_{ob}{}^n} \left(\frac{x_{ob} - x_o}{D_{ob}} \hat{x} + \frac{y_{ob} - y_o}{D_{ob}} \hat{y} \right)$$

$$\vec{F}_T = F_{CT} \left(\frac{x_t - x_o}{D_T} \hat{x} + \frac{y_t - y_o}{D_T} \hat{y} \right)$$

From the above and equation (3.15) we have:

$$F_{CT} \left(\frac{x_t - x_o}{D_T} \hat{x} + \frac{y_t - y_o}{D_T} \hat{y} \right) = -\frac{F_{CR} W^n}{D_{ob(\min)}{}^n} \left(\frac{x_{ob} - x_o}{D_{ob(\min)}} \hat{x} + \frac{y_{ob} - y_o}{D_{ob(\min)}} \hat{y} \right)$$

Therefore:

$$F_{CT} = \frac{F_{CR} W^n}{D_{ob(\min)}{}^n} \Leftrightarrow F_{CR} W^n = F_{CT} D^n{}_{ob(\min)} \Leftrightarrow D_{ob(\min)} = W \sqrt[n]{\frac{F_{CR}}{F_{CT}}}$$

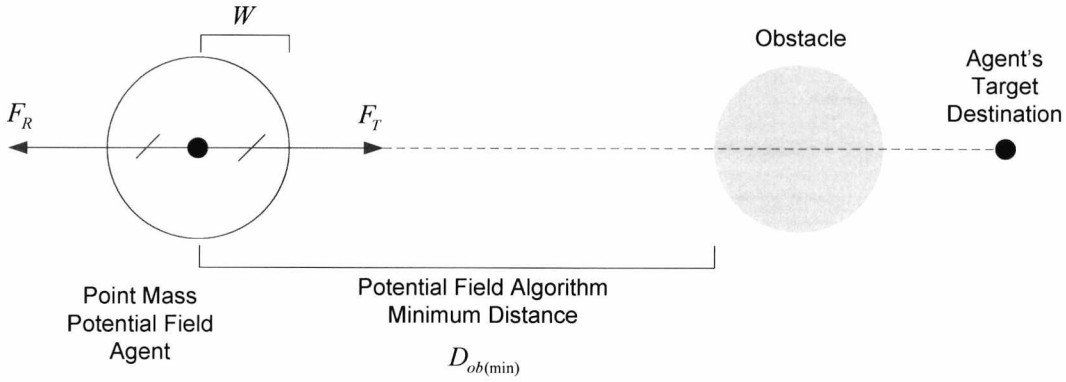


Figure 3-5: AWSPPF Algorithm Minimum Distance

In Figure 2-1is illustrated the AWSPPF vectors when the PFA has Minimum Distance from an obstacle. From equation (3.15) we can understand that the ratio $\sqrt[n]{\frac{F_{CR}}{F_{CT}}}$ has to be greater than 1 for the PFA to avoid collision in both cross and head-on collision scenarios. It is important to be noted that the algorithm can handle different size obstacles by adjusting the algorithm's Minimum Distance.

3.2.4 Active Window Single Point Potential Field (AWSPPF) Algorithm Software Design and Implementation

The implementation of the VFF is based on the following flow chart shown in Figure 3-6. It is assumed that the width of the agent and the obstacle are known, as well as their initial coordinates.

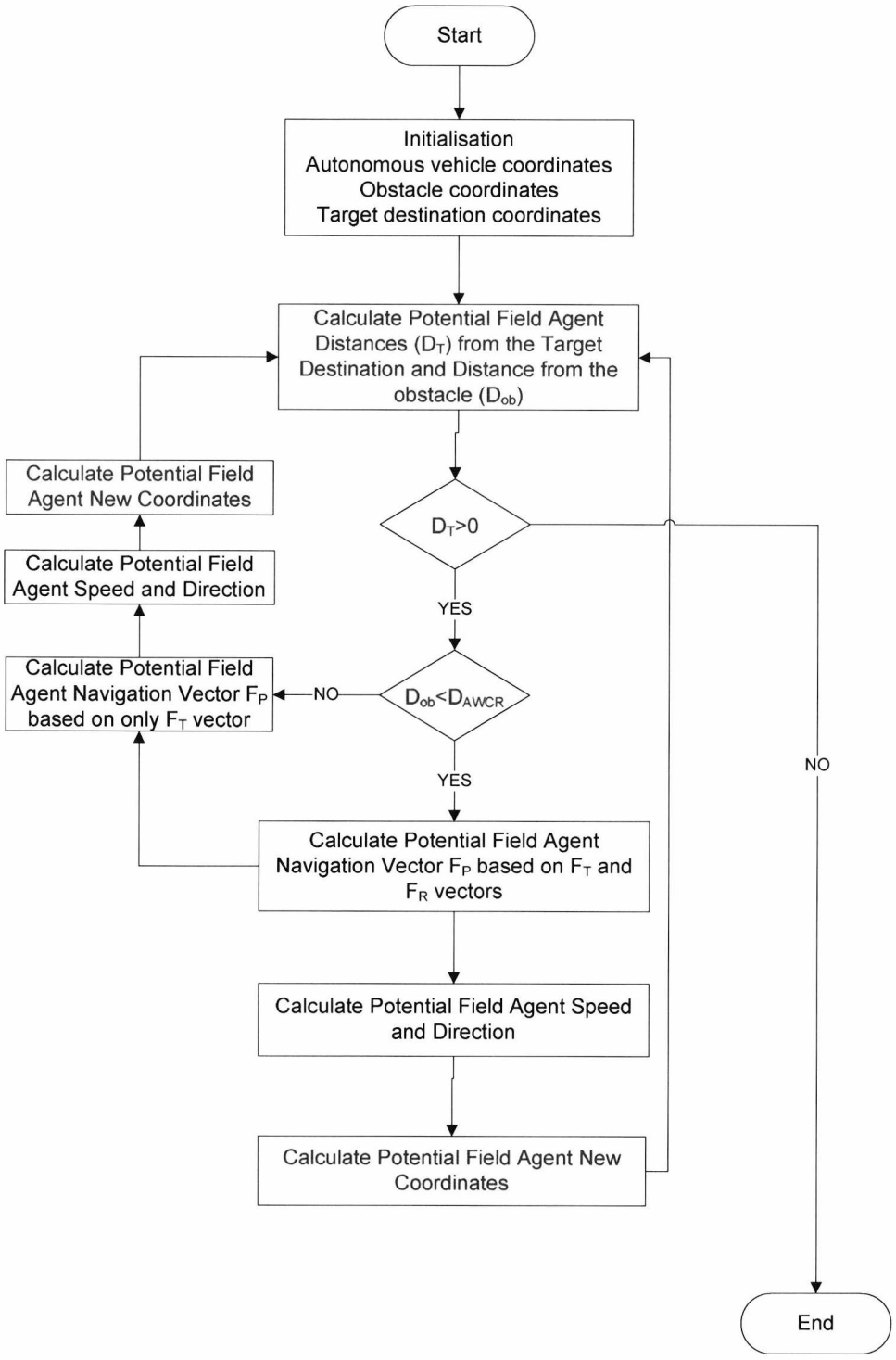


Figure 3-6: AWSPPF Algorithm implementation flowchart

The algorithm is implemented in Matlab and its representation could be based on the following pseudo code:

1. Initialise Potential Field Agent (PFA) size
2. Initialise PFA coordinates values
3. Initialise obstacle size
4. Initialise obstacle coordinates

5. Initialise Active Window Circle Radius D_{AWCR}
6. Calculate PFA distance from its target destination D_T
7. Calculate PFA distance from the obstacle D_{ob}
8. Initialise Sampling Rate
9. While(D_T>0)
    a. Initialise while (D_T>0) loop counter to zero and increment the counter in the next iteration
    b. If(D_{ob}<D_{AWCR})
        i. Calculates Virtual Attractive Force Vector
        ii. Calculates Virtual repulsive Force Vector
        iii. Calculates Resultant Artificial Potential Force Field vector.
        iv. Calculate the PFA speed based on the Magnitude of the Resultant Artificial Potential Force
        v. Calculate New Coordinates based on the Time sample specified
    c. Else
        i. Initialise else counter to zero and increment the counter in the next iteration
        ii. Calculates Virtual Attractive Force Vector
        iii. Calculates Resultant Artificial Potential Force Field vector (based only on the Target destination vector).
        iv. Calculate the PFA speed based on the Magnitude of the Resultant Artificial Potential Force
        v. New Calculate New Coordinates based on the Time sample specified
    d. Calculate D_{ob}
    e. Calculate D_T
10. Break while(D_T>0) loop
11. End //Target Destination has been reached

We can analyse the above pseudo code as follows:

1. Initialise Potential Field Agent (PFA) size. We define the PFA to have a circular shape with radius equal to $R_{agent} = \frac{W}{2}$. Where W is the width of the agent.
2. Initialise PFA coordinates values. We PFA coordinates are X_0, Y_0.
3. Initialise obstacle size. We consider the obstacle to be circular with radius $R_{obstacle}$.
4. Initialise obstacle coordinates. The obstacle coordinates are X_{ob} and Y_{ob}.
5. Initialise Active Window Circle Radius D_{AWCR}
6. Calculate PFA distance from its target destination D_T

$$D_T = \sqrt{(X_T - X_0)^2 + (Y_T - Y_0)^2}$$

7. Calculate PFA distance from the obstacle D_{ob}

$$D_{ob} = \sqrt{(X_{ob} - X_0)^2 + (Y_{ob} - Y_0)^2}$$

8. Initialise Sampling Rate

9. While(D_T>0)
    a. Initialise while (D_T>0) loop counter to zero and increment the counter in the next iteration. This counter is used to monitor the total time the agent needed to arrive to its target destination.
    b. If(D_{ob}<D_{AWCR})
        i. Calculate Virtual Attractive Force Vector from (3.6)

$$\overrightarrow{F_T} = F_{CT}\left(\frac{x_t - x_o}{D_T}\hat{x} + \frac{y_t - y_o}{D_T}\hat{y}\right)$$

        ii. Calculate Virtual repulsive Force Vector from (3.7)

$$\overrightarrow{F_R} = \frac{F_{CR}W^n}{D_{ob}^{\ n}}\left(\frac{x_{ob} - x_o}{D_{ob}}\hat{x} + \frac{y_{ob} - y_o}{D_{ob}}\hat{y}\right)$$

        iii. Calculate Resultant Artificial Potential Force Field vector.

$$\vec{F}_P = \vec{F}_T + \vec{F}_R$$

iv. Calculate the PFA speed based on the Magnitude of the Resultant Artificial Potential Force from (3.13)

$$V = \frac{\left|\vec{F}_P\right|}{F_{P_{MAX}}} V_{MAX}$$

v. Calculate New Coordinates based on the Time sample specified T_S

$$x = x_0 + \frac{\left|\vec{F}_P\right|}{F_{P_{MAX}}} V_{MAX} T_S \cos\theta$$

$$y = y_0 + \frac{\left|\vec{F}_P\right|}{F_{P_{MAX}}} V_{MAX} T_S \sin\theta$$

c. Else

i. Initialise else counter to zero and increment the counter in the next iteration

ii. Calculate Virtual Attractive Force Vector

$$\overrightarrow{F_T} = F_{CT}\left(\frac{x_t - x_o}{D_T}\hat{x} + \frac{y_t - y_o}{D_T}\hat{y}\right)$$

iii. Calculate Resultant Artificial Potential Force Field vector (based only on the Target destination vector).

$$\vec{F}_P = \vec{F}_T$$

iv. Calculate the PFA speed based on the Magnitude of the Resultant Artificial Potential Force

$$V = \frac{\left\|\vec{F}_T\right\|}{F_{P_{MAX}}} V_{MAX}$$

v. Calculate New Coordinates based on the sample time period T_S

$$x = x_0 + \frac{\left|\vec{F}_P\right|}{F_{P_{MAX}}} V_{MAX} T_S \cos\theta$$

$$y = y_0 + \frac{\left|\vec{F}_P\right|}{F_{P_{MAX}}} V_{MAX} T_S \sin\theta$$

d. Calculate D_{ob}

$$D_T = \sqrt{(x_T - x_0)^2 + (y_T - y_0)^2}$$

e. Calculate D_T

$$D_{ob} = \sqrt{(x_{ob} - x_0)^2 + (y_{ob} - y_0)^2}$$

10. Break while(D_T>0) loop
11. End //Target Destination has been reached

It is important to be noted that we use the counters to monitor the PFA time and trajectory length to reach its target destinations.

$$S_{total} = \sum_{i}^{k} \sqrt{\Delta x_i^2 + \Delta y_i^2}$$

Where k is equal to the number of iterations that the While(D_T>0) loop needs to terminates

$k \in \square$

Where S_{total} is the total trajectory length that the agent has covered to reach its target destination.

In the following paragraphs, we have tested the collision capabilities of the above generic potential field algorithm AWSPPF. This test takes place based on the characteristics of an Autonomous Unmanned Surface Vehicles USV (or Autonomous Unmanned Water based Vehicle). For this scenario we consider the agent to be circular and have radius of 9m, maximum speed of 5.14m/sec (or 10nm/h). For simplicity, we assume a Cartesian coordination system with x axis pointing to North and an anticlockwise angle increment. The AWSPPF Algorithm characteristics are set to be Dmin = 100m and the Active Window (AW) equal to 1000m.

The algorithm performance is tested first in static environment and then in dynamic environment. For the static environment we have considered the following 5 collision scenarios, in order to illustrate the performance of the algorithm near and on aliment among the initial autonomous vehicle position, the target destination and the obstacle. In these scenarios the obstacle has a radius of 50m.

The initial coordinates of the autonomous surface potential field vehicle are (0, 460) and its target destination coordinates are (1000, 460), the obstacle coordinates are (500, 500). We keep the obstacle coordinates the same for all collision scenarios. The second set of coordinates is: initial position (0, 480), target destination (1000, 480). The third set of coordinates is: initial position (0, 490), target destination (1000, 490). The forth set of coordinates is: initial position (0, 499), target destination (1000, 499). Finally, the fifth set of coordinates is: initial position (0, 500), target destination (1000, 500). In this case we have full aliment of initial position, obstacle and target destination. We have selected these coordinates to illustrate gradual performance degradation of the algorithm in relation to the aliment of the autonomous vehicle initial position, target destination and obstacle.

For the first set of coordinates:
Autonomous Surface Vehicle Initial Position: (0, 460)
Autonomous Surface Vehicle Target Destination: (1000, 460)
Obstacle coordinates (500, 500)

The performance of the algorithm related to the trajectory generation it is illustrated in Figure 3-7. The relation of the distance between the autonomous vehicle and the obstacle versus time in Figure 3-8, and the autonomous vehicle speed versus time in Figure 3-9.

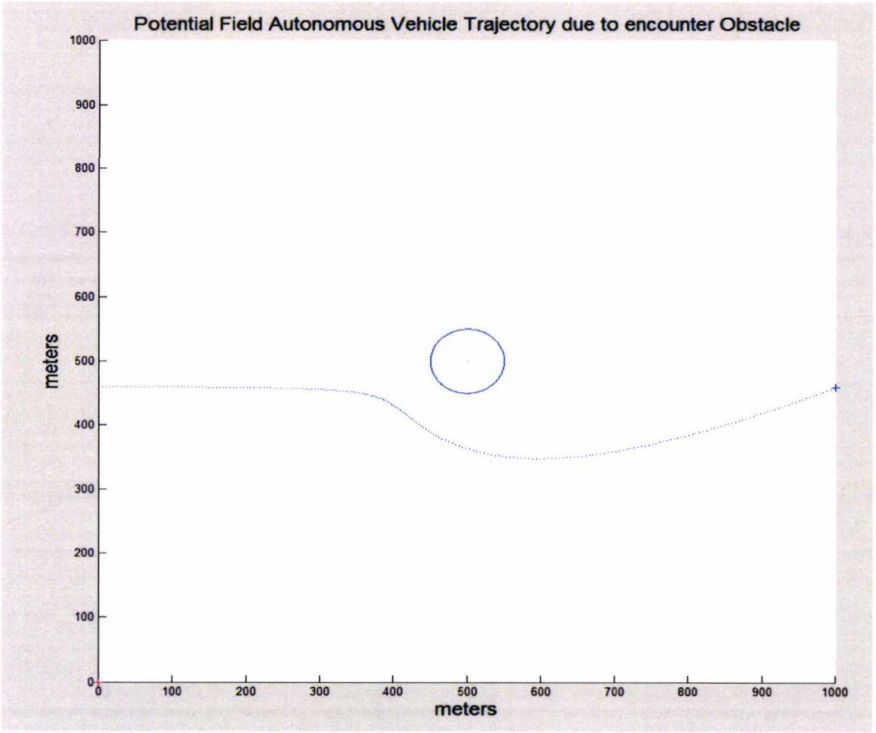


Figure 3-7: Collision avoidance scenario of autonomous agent and obstacle with obstacle out of aliment between initial position and target destination by 40m.

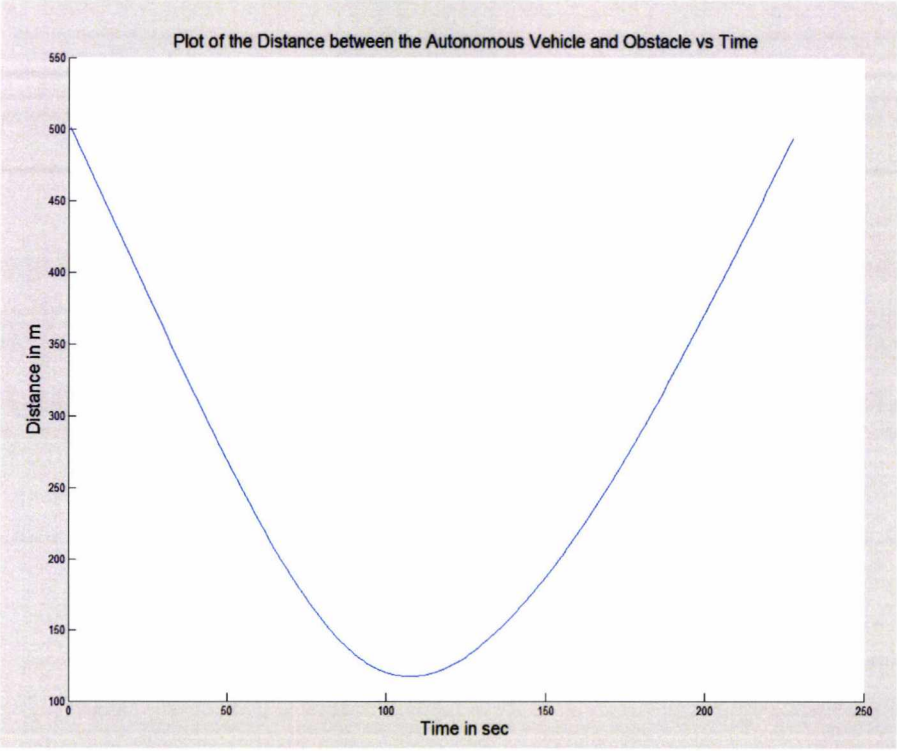


Figure 3-8: Autonomous agent |Distance from obstacle vs Time when the autonomous agent and the obstacle are out of aliment by 40m.

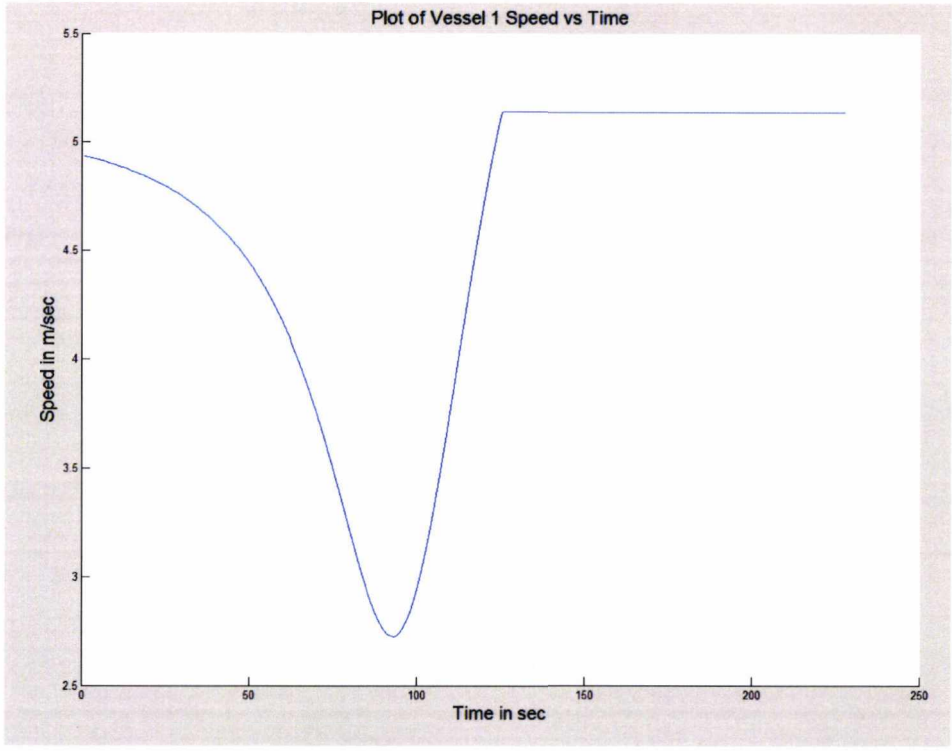


Figure 3-9: Autonomous agent Speed Variation vs Time when the autonomous agent and the obstacle are out of aliment by 40m.

The performance highlights of the AWSPPF Algorithm when the autonomous surface vehicle is out of aliment with the obstacle by 40m are the following:

$D\min = 117.5m$ at 107sec

$V\min = 2.723m/\sec$ (or 5.3 Nautical Miles (NM)) at 93sec

Max speed duration: from 126sec to 229sec

$t_D = 229\sec$

$l_T = 1,043.4m$

Where:

$D\min$ is the minimum distance between the autonomous surface vehicle trajectory and the obstacle.

$V\min$ is the minimum speed that the autonomous agent took in some point following its trajectory from initial position to target destination.

t_D is the time duration of the autonomous surface vehicle between its initial position and its target destination.

For the Second set of coordinates:

Autonomous Surface Vehicle Initial Position: (0, 480)

Autonomous Surface Vehicle Target Destination: (1000, 480)

Obstacle coordinates (500, 500)

The performance of the algorithm related to the trajectory generation it is illustrated in Figure 3-10. The relation of the distance between the autonomous vehicle and the obstacle versus time in Figure 3-11, and the autonomous vehicle speed versus time in Figure 3-12.

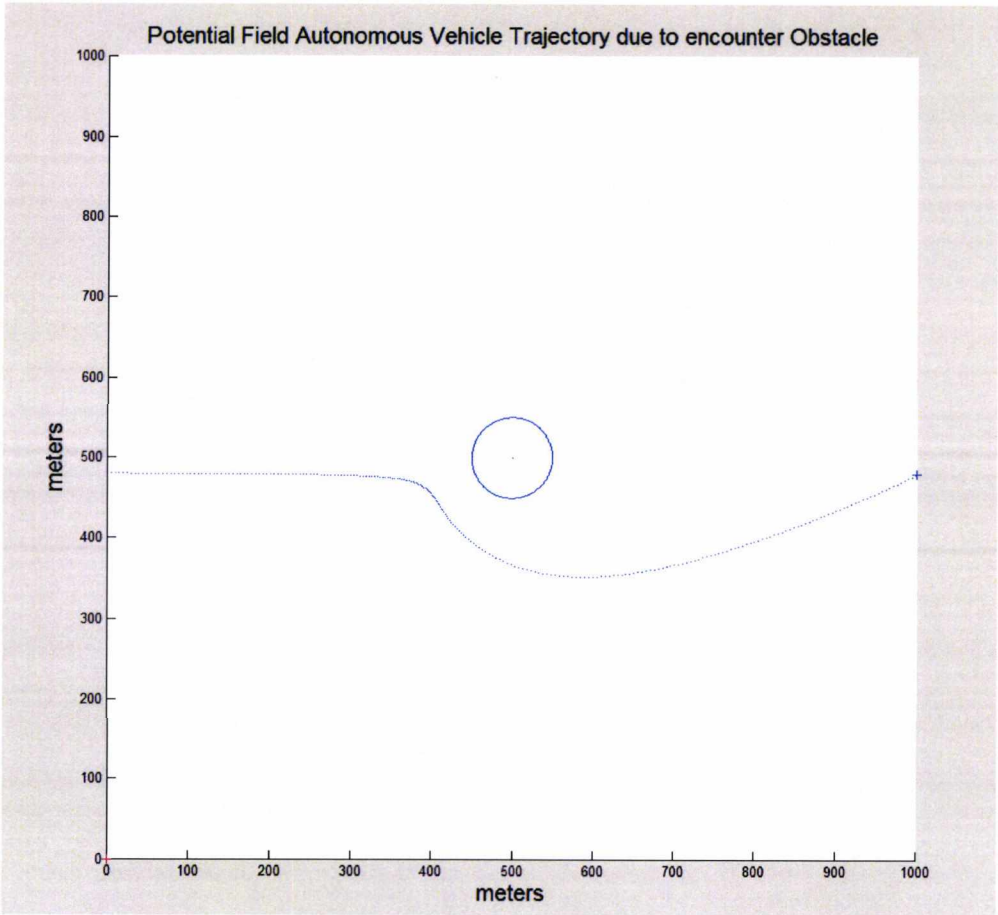


Figure 3-10: Collision avoidance scenario of autonomous agent and obstacle with obstacle out of aliment between initial position and target destination by 20m.

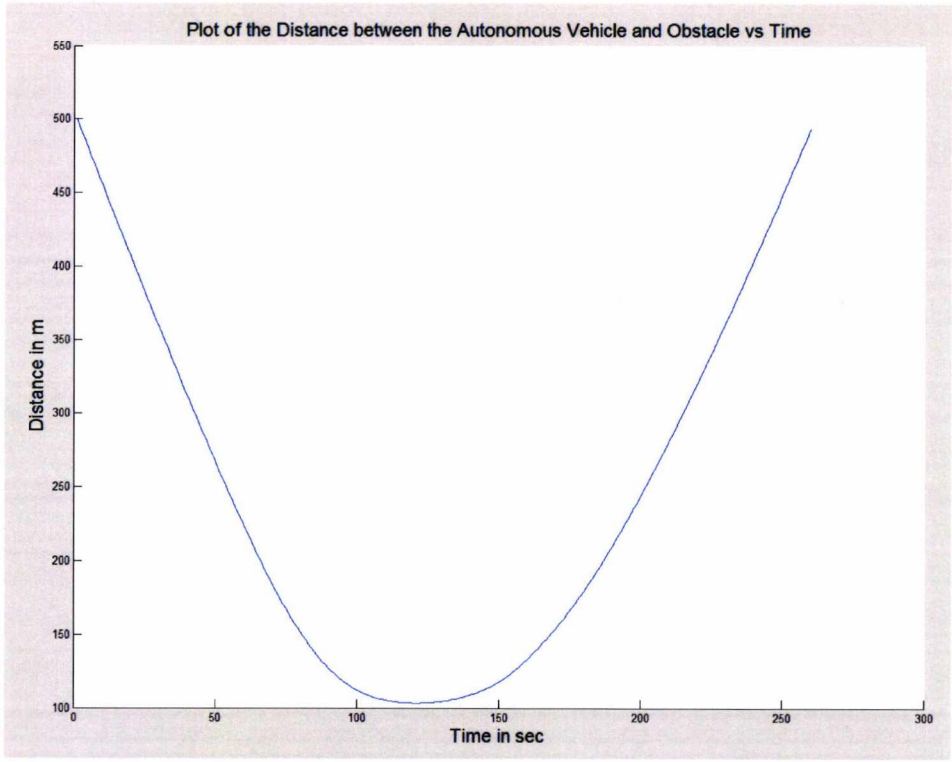


Figure 3-11: Autonomous agent Distance from obstacle vs Time when the autonomous agent and the obstacle are out of aliment by 20m.

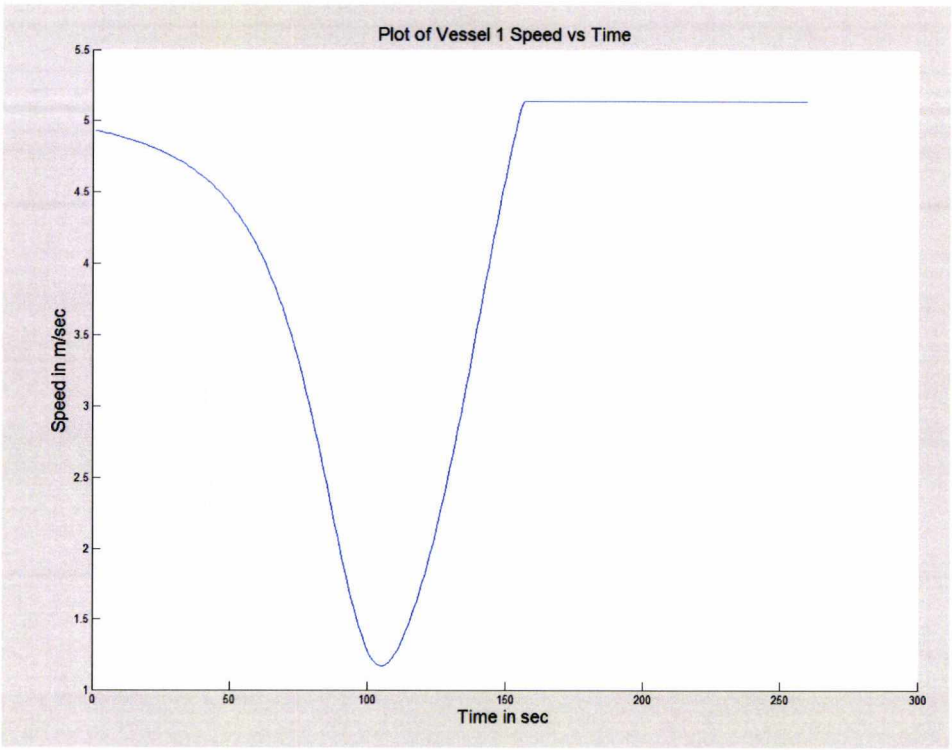


Figure 3-12: Autonomous agent Speed Variation vs Time when the autonomous agent and the obstacle are out of aliment by 20m.

The performance highlights of the AWSPPF Algorithm when the autonomous surface vehicle is out of aliment with the obstacle by 20m are the following:

$D\min = 107.7m$ at 107sec

$V\min = 1.849m/\sec$ (3.6 NM) at 99sec

Max speed duration: from 141sec to 245sec

$t_D = 245\sec$

$l_T = 1,064.1m$

For the third set of coordinates:

Autonomous Surface Vehicle Initial Position: (0, 495)

Autonomous Surface Vehicle Target Destination: (1000, 495)

Obstacle coordinates (500, 500)

The performance of the algorithm related to the trajectory generation it is illustrated in Figure 3-13. The relation of the distance between the autonomous vehicle and the obstacle versus time in Figure 3-14, and the autonomous vehicle speed versus time in Figure 3-15.

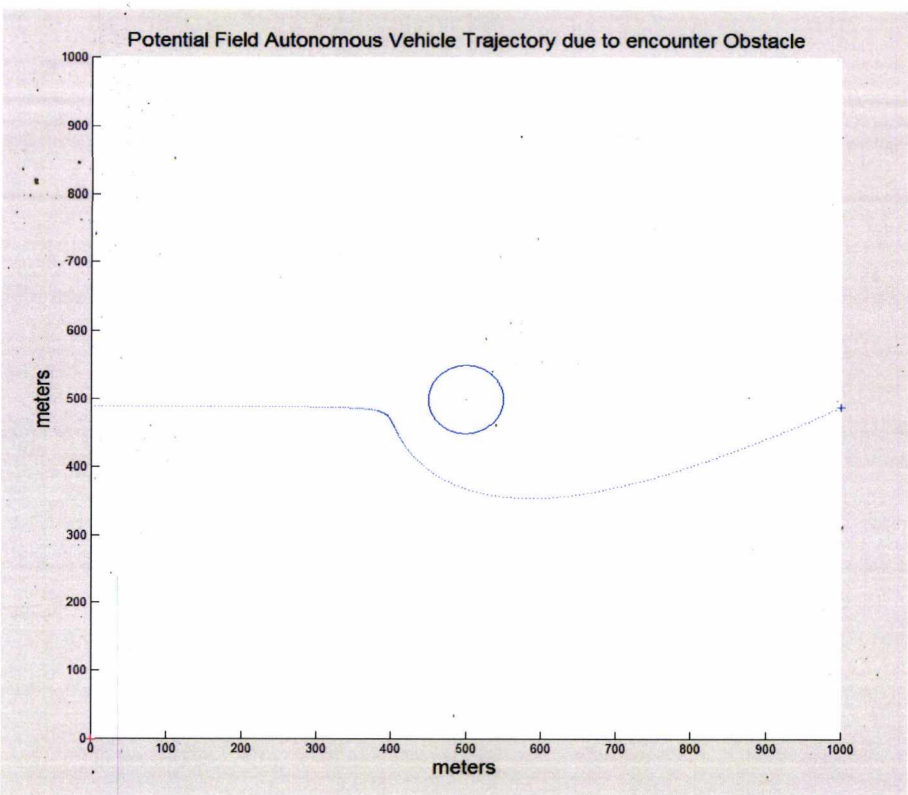


Figure 3-13: Collision avoidance scenario of autonomous agent and obstacle with obstacle out of aliment between initial position and target destination by 10m

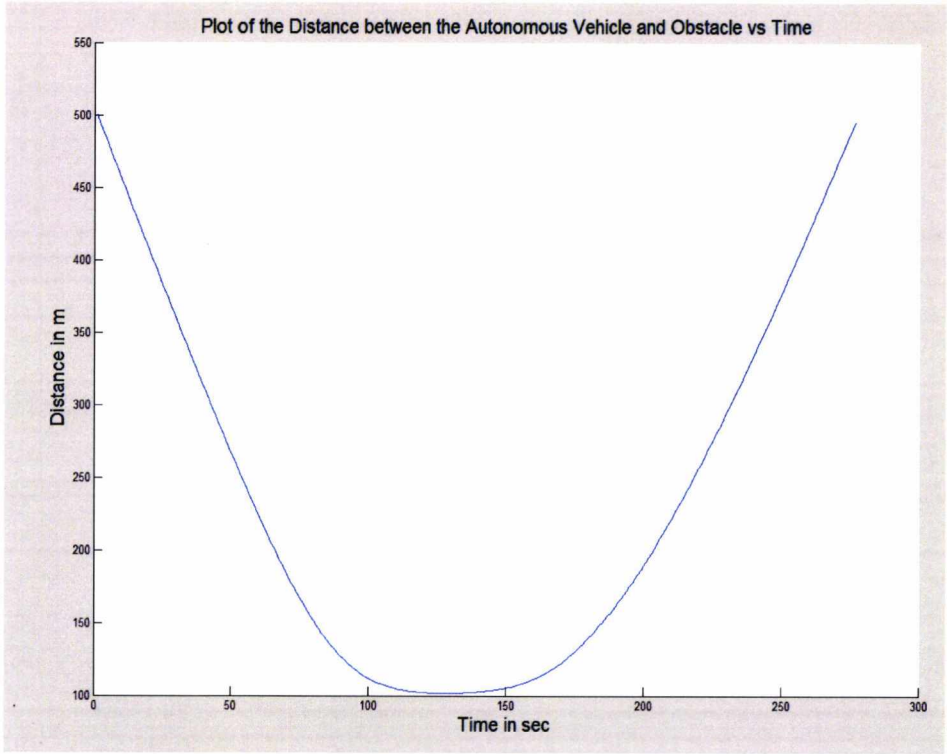


Figure 3-14: : Autonomous agent Distance from obstacle vs Time when the autonomous agent and the obstacle are out of aliment by 10m.

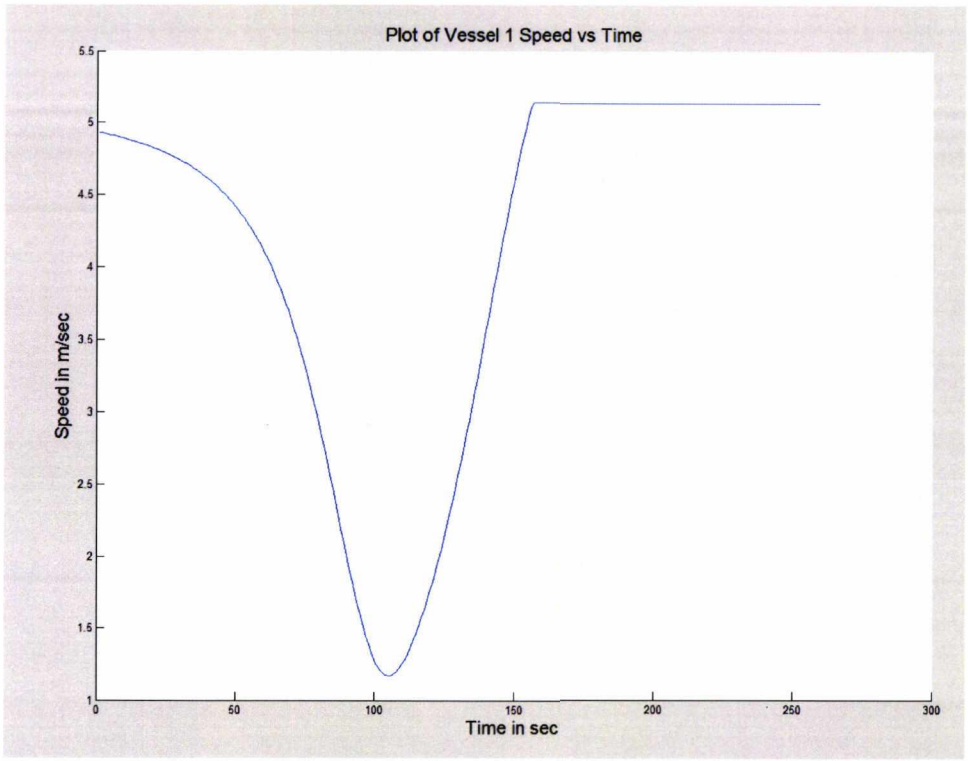


Figure 3-15: Autonomous agent Speed Variation vs Time when the autonomous agent and the obstacle are out of aliment by 10m.

The performance highlights of the AWSPPF Algorithm when the autonomous surface vehicle is out of aliment with the obstacle by 10m are the following:

$D\min = 101.6m$ at for a time period between 125sec to 131 sec

$V\min = 0.734m/\sec$ (1.43 NM) at 111sec

Max speed duration: from 174sec to 278sec

$t_D = 278\sec$

$l_T = 1,089m$

For the fourth set of coordinates:

Autonomous Surface Vehicle Initial Position: (0, 499)

Autonomous Surface Vehicle Target Destination: (1000, 499)

Obstacle coordinates (500, 500)

The performance of the algorithm related to the trajectory generation is illustrated in Figure 3-16. The relation of the distance between the autonomous vehicle and the obstacle versus time in Figure 3-17, and the autonomous vehicle speed versus time in Figure 3-18.

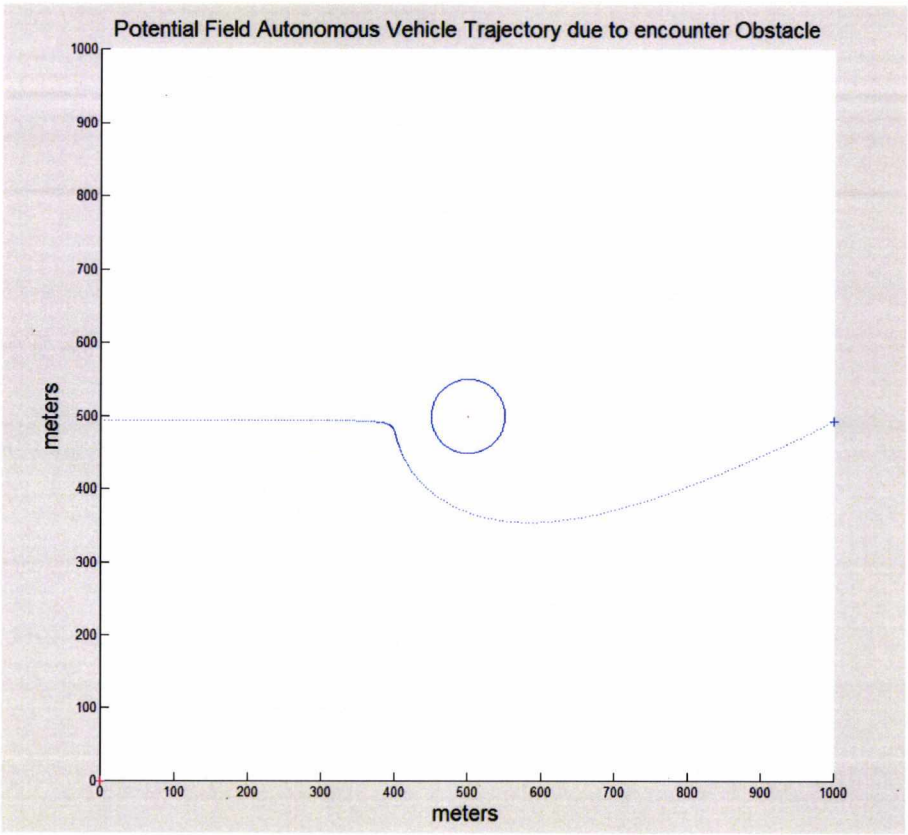


Figure 3-16: Collision avoidance scenario of autonomous agent and obstacle with obstacle out of aliment between initial position and target destination by 1m.

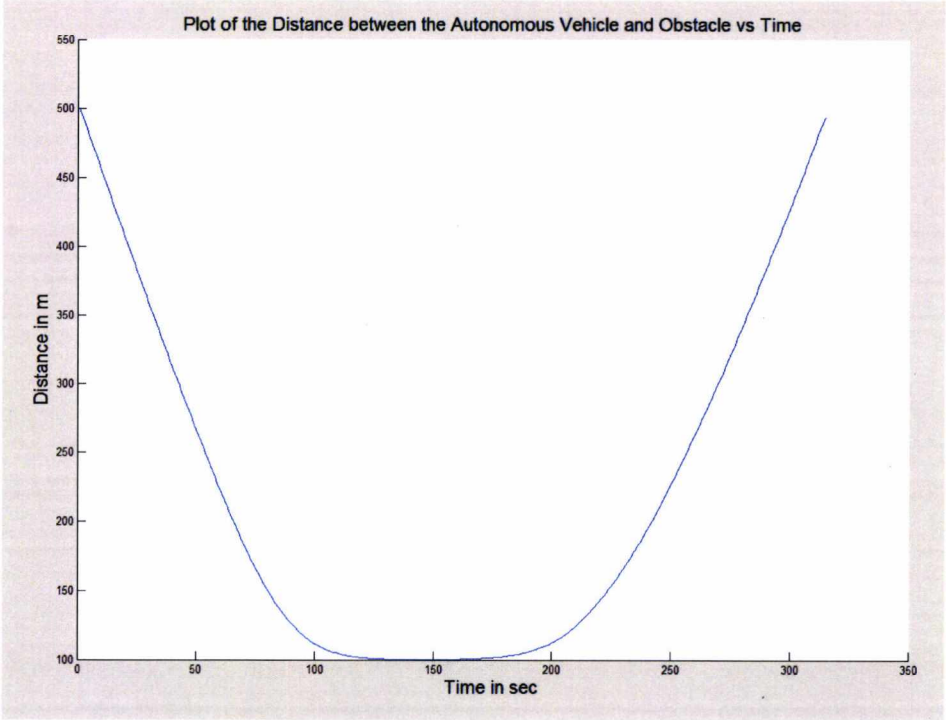


Figure 3-17: Autonomous agent Distance from obstacle vs Time when the autonomous agent and the obstacle are out of aliment by 1m.

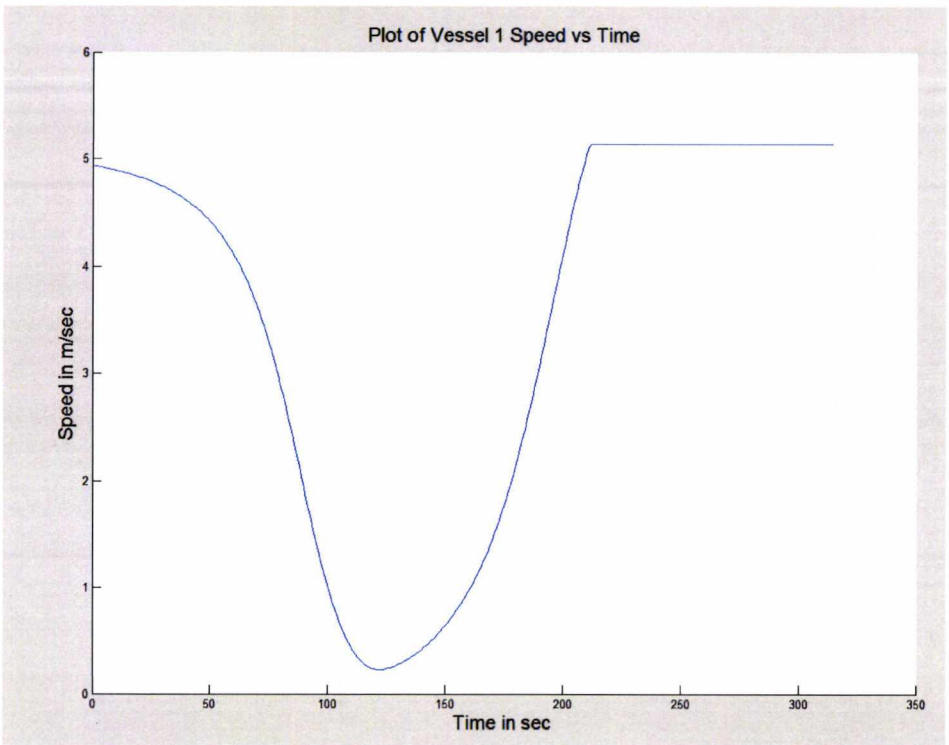


Figure 3-18: Autonomous agent Speed Variation vs Time when the autonomous agent and the obstacle are out of aliment by 1m.

The performance highlights of the AWSPPF Algorithm when the autonomous surface vehicle is out of aliment with the obstacle by 1m are the following:

$D\min = 100.3m$ at for a time period between 125sec to 131 sec

$V\min = 0.237m/\sec$ (0.46NM) at 111sec

Max speed duration: from 174sec to 278sec

$t_D = 316\sec$

$l_T = 1,098m$

For the fifth set of coordinates:

Autonomous Surface Vehicle Initial Position: (0, 500)

Autonomous Surface Vehicle Target Destination: (1000, 500)

Obstacle coordinates (500, 500)

The performance of the algorithm related to the trajectory generation it is illustrated in Figure 3-19.

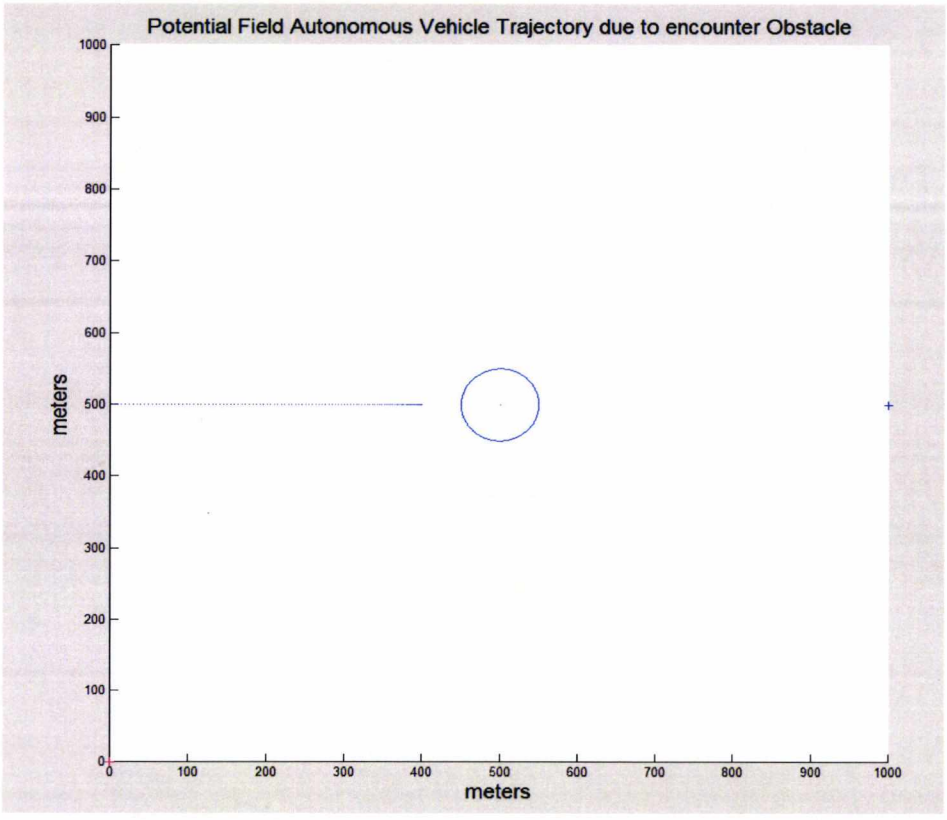


Figure 3-19: Collision avoidance scenario of autonomous agent and obstacle with obstacle out of aliment between initial position and target destination of 0m (Potential field algorithm local minima and navigational deadlock).

As we can see from the above figure the agent is within a local minima deadlock.

From the above we can notice that the trajectories of the first 4 collision scenarios are relatively similar but if we observe them closely, they have significant variation in the autonomous vehicle speed, trajectory length agent minimum distance from the obstacle. The performance of the algorithm becomes worse in relationship to the aliment between the autonomous vehicle initial position, the static obstacle and the autonomous agent target destination. The autonomous potential vehicle exhibits local minima when its target destination current position and target destination are in perfect aliment. Therefore, based on the simulation results, the AWSPPF algorithm is processing efficient, has smooth trajectories and exhibits local minima in static environment.

For the dynamic environment we have considered 4 collision scenarios based on the same initial coordinates for autonomous vehicle and dynamic obstacle, and we only alter the speed of the dynamic obstacle. The dynamic obstacle has constant speed and direction. Based on this scenario we demonstrate that the algorithm is susceptible to the combination of geometrical symmetries and speed correlation between the autonomous vehicle and the dynamic obstacle.

The initial coordinates for the autonomous vehicle and the dynamic obstacle are:
Autonomous surface vehicle initial coordinates: (0, 500)
Autonomous surface vehicle target destination coordinates: (1000, 500)
Autonomous surface vehicle initial and max speed: 5.14m/sec
Obstacle initial coordinates: (800, 100)

We examine for difference constant speeds for the dynamic obstacle which are:
Obstacle constant speed1: 4.5m/sec
Obstacle constant speed2: 5m/sec
Obstacle constant speed3: 5.14 m/sec
Obstacle constant speed3: 6.5m/sec

We have chosen the above initial coordinates to achieve geometrical symmetry between the autonomous agent, the dynamic obstacle and their current trajectories projection crossing point.

In Figure 3-20 is illustrated the trajectory of the autonomous vehicle when the dynamic obstacle has speed around 14% lower that the agent algorithmic max speed.

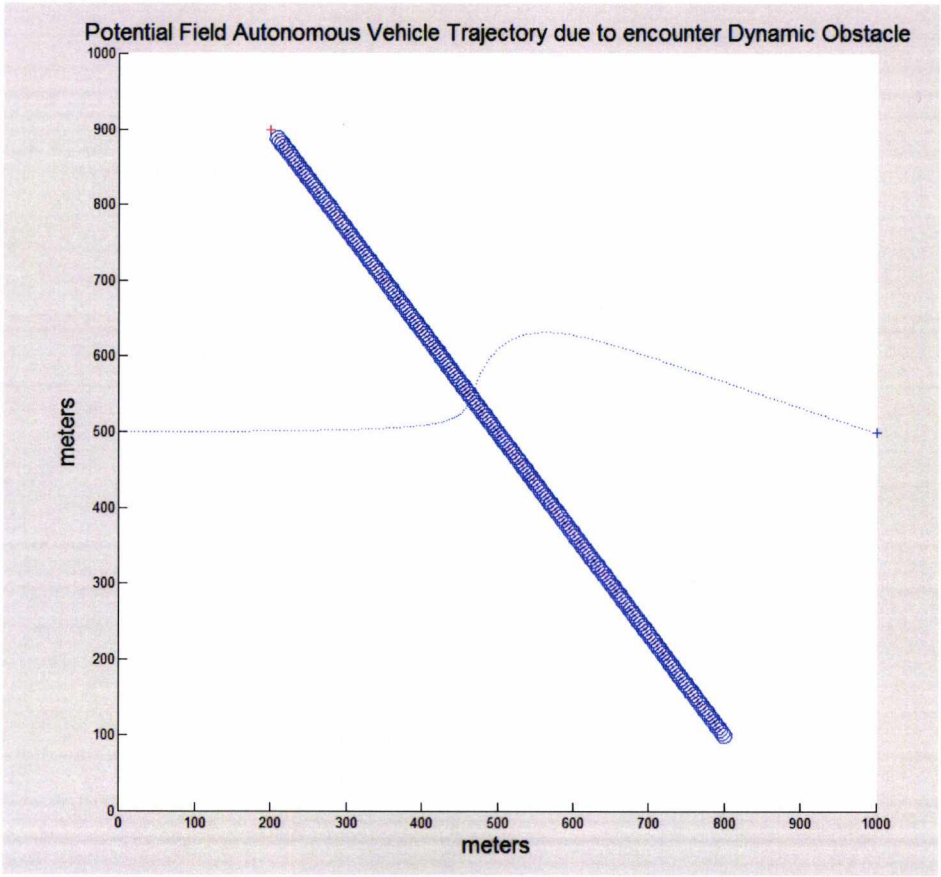


Figure 3-20: Autonomous agent trajectory when in geometrical symmetrical collision scenario with dynamic obstacle of constant speed and direction. Speed difference between autonomous vehicle and dynamic obstacle14.2%.

In Figure 3-21 is illustrated the trajectory of the autonomous vehicle when the dynamic obstacle has 2.8% lower speed than the maximum speed of the autonomous vehicle.

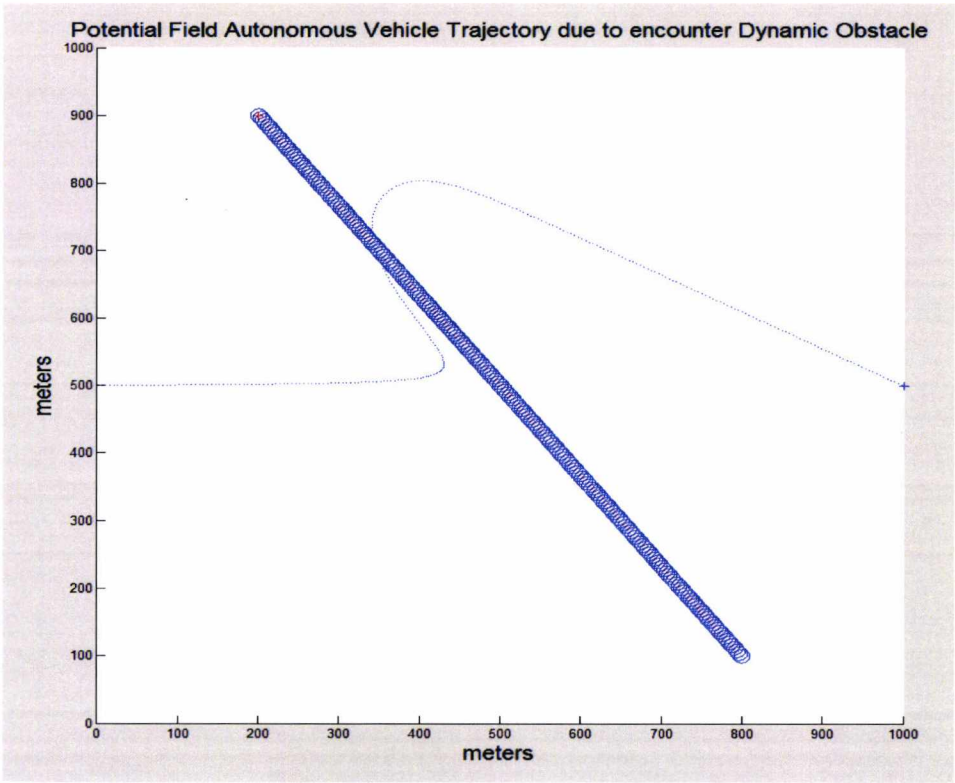


Figure 3-21: Autonomous agent trajectory when in geometrical symmetrical collision scenario with dynamic obstacle of constant speed and direction. Speed difference between autonomous vehicle and dynamic obstacle 2.8%.

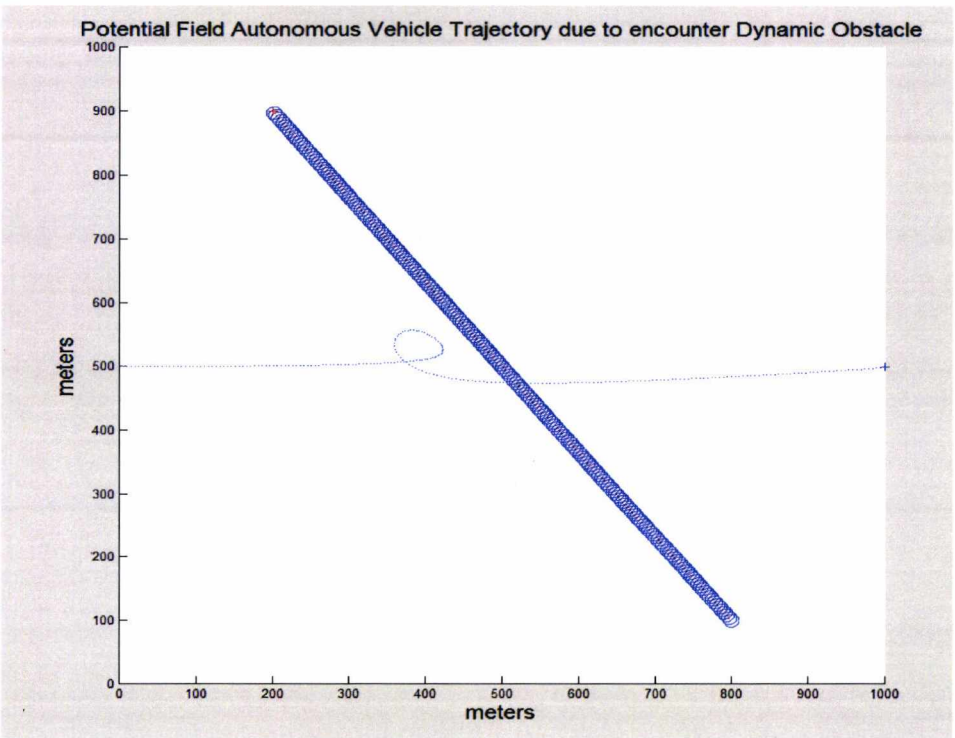


Figure 3-22: Autonomous agent trajectory when in geometrical symmetrical collision scenario with dynamic obstacle of constant speed and direction. Speed difference between autonomous vehicle and dynamic obstacle 0%.

As we can observe from Figure 3-21 and Figure 3-22, when the speed of the dynamic obstacle approaches the maximum algorithmic speed of the potential field algorithm, the performance of the algorithm degrades in terms of average speed, trajectory length and trajectory smoothness. This degradation takes place due to both spatial symmetry and the speed magnitude correlation of the agents. We analytically explain the above in Chapter 4.

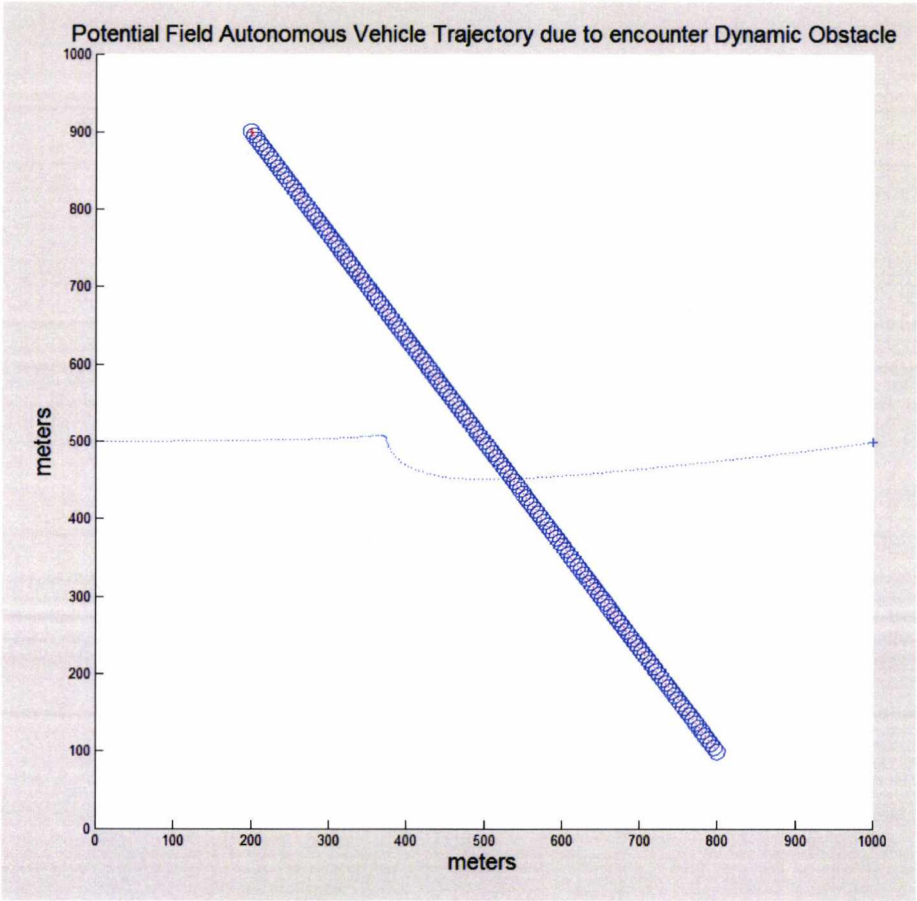


Figure 3-23: Autonomous agent trajectory when in geometrical symmetrical collision scenario with dynamic obstacle of constant speed and direction. Speed difference between autonomous vehicle and dynamic obstacle 12.45%

On this other hand, when the dynamic obstacle speed is greater than algorithmic by a reasonable amount the performance is satisfactory, as is illustrated in Figure 3-23. An analytic performance of the AWSPPF algorithm in relationship with a dynamic obstacle is presented in Chapter 5, along with a new efficient algorithm that avoids these inefficiencies.

From all the above results we come to the conclusion that first, the AWSPPF algorithm accommodate the advantages and disadvantages of the classical Potential Field Methods, as well as include an efficient Active Window (AW).

Second, the AWSPPF algorithm performance degrades in a dynamic environment. This degradation takes place when the initial coordinates of the autonomous vehicle, the dynamic obstacle and crossing point of their initial headings form a geometrical symmetry. With this observation as a guide, we have identified and defined the local minima in a dynamic environment, and more specifically, in Pure Dynamic Environment (PDE). The identification and definition of local minima of the AWSPPF algorithm in PDE takes place in the next chapter.

Chapter 4

4. The Concept of Monovular Autonomous Agent Correlation (MAAC) and how to Identify Navigational Deadlocks

In this chapter we introduce the novel concept of Monovular Autonomous Agent Correlation (MAAC). The significance of this concept is to identify performance inefficiencies of the Potential Fields algorithms in Pure Dynamic Environment (PDE). These performance inefficiencies show that the Potential Field Algorithms experience local minima not only in static environment but also in dynamic.

We have defined The PDE in the previous chapter, as the environment contains only dynamic agents and/or obstacles. The word Monovular is inspired by the biological term Monovular, which means from the same ovum (egg). Therefore, we define as Monovular Agents the agents that are identical not only in terms of kinematic or dynamic models but also in hardware and software terms. We have introduced the word Monovular to materialise the novel concept of MAAC, which is interdisciplinary inspired from the concept of autocorrelation in signal processing. Autocorrelation is a mathematical tool for finding repeating patterns e.g. identified periodic signals under the noise floor or missing fundamental frequencies etc. Different fields of engineering and AI (e.g. signal processing, pattern recognition, cryptanalysis etc.) define the autocorrelation in a different way.

In collision avoidance, we have aimed to use the concept of MAAC to identify repeating inefficient trajectories of the Autonomous Agents in pure dynamic environment (the pure dynamic environment for MAAC consists of only two Monovular agents initially). The principle of easier identification is that MAAC amplifies their worst navigational behaviour when they are in the same dynamic environment. Therefore, this behaviour is easier to be captured and defined. The use of the MAAC concept for finding repeating trajectory or behavioural patterns among dynamic agents is proved in the generic Potential field algorithm, which we have described in the previous section. At this point, we would like to note that the concept could possibly be applicable for other autonomous navigation algorithms but its effectiveness is not tested. Therefore, it is beyond the purpose of this Thesis to prove it.

In this study therefore, we have focused on the identification of Potential Field algorithm repeating inefficient trajectories in pure dynamic environment. Based on these trajectories identification, we have also been enabled to identify and define the causes that encourage such behaviour navigation patterns of the Monovular Agents.

The method we have followed to utilise the MAAC concept in order to identify and define repeated inefficient trajectories in Potential Field Algorithms is summarised in the next steps:

1. Define the Autonomous Monovular Agents, and the size of local Pure Dynamic Environment.
2. Place two Monovular Agents within the same local environment, and try to identify repeating inefficient trajectories.
3. Identify the generic causes of the inefficient repeating trajectories of the autonomous Monovular agents.
4. Define the causes of the inefficient repeating trajectories of the Potential Field Monovular Agents while navigating in the Pure Dynamic Environment.

Initially, we have to define a local pure dynamic environment that is correlated to the autonomous agents kinematic characteristics e.g. manoeuvrability and speed. In other words, in this step we refer to the local environment size, which will be appropriate for the specific agents' dynamics.

The second step is to place the Monovular Agents in the above local environment so we can examine their collision avoidance performance in different initial conditions. In this step we have to experiment with different combinations of both initial coordinates and velocity vectors of the agents to identify any possible repeated patterns of inefficient trajectories. With the term velocity vector we describe the directions and the speed of the agent. The angle of this vector corresponds to the direction, and the magnitude corresponds to the speed of the autonomous agent. This velocity vector is generated by the Potential Field Algorithm that navigates the autonomous agents in real-time.

In the third step, we have to examine the agents' behaviour while experience inefficient trajectories. In this way we will be able to identify the causes that result the autonomous agent to behave in this way in pure dynamic environment.

Finally, when we have identified these causes it is important to define them. So, we will be able to mathematically distinguish when the autonomous agents are in the beginning and when they are in the end of these inefficient trajectories.

As we will analytically illustrate in the next subsections of this chapter based on the above steps, we have identified that this type of potential field algorithms experience long inefficient trajectories that can lead into deadlocks. These inefficient trajectories are caused by local minima. This means that local minima are inherited by potential field algorithms not only in static environment but also in pure dynamic environment.

Furthermore, we have mathematical defined and grouped the causes of the potential Monovular autonomous agent behaviour under a new state we have named Trajectory Equilibrium State (TES). The author has published this concept for first time in [4]. This State is responsible for local minima of Monovular Autonomous Agents in the same dynamic environment.

4.1 Trajectory Equilibrium State (TES) Identification and Definition in Cross Collision Avoidance, based on Monovular Potential Agents in Pure Dynamic Environment (PDE)

In this section we apply the generic steps of MAAC concept for the identification and definition of repeated inefficient trajectories. Based on these steps we have identified that for this scenario the inefficient trajectories are caused by the geometrical symmetries among the agents' current coordinates and their target destinations. These geometrical symmetries result the agent's Potential Field Navigational vectors to have strong behavioral correlation, which trap them in TES. Therefore, the Potential Field agents are in this state when they mutually experience

local minima in Pure Dynamic Environment. In the most extreme cases of TES both agents are led to navigational deadlock.

Local minima have only been identified in static environment for Potential field algorithms. Therefore, in this study, we have identified local minima in the newly defined Pure Dynamic Environment (PDE). In addition, by analyzing the Potential Field local minima causes in PDE, we have realised that the causes are quite different than the causes which take place in static environment. In static environment the causes of local minima are based on the shape of the static obstacles when are in close proximity with the potential field agent. On the other hand, in PDE the local minima are caused by the relative position and the velocity vectors among the Monovular agents' current position and their target destination. This type of correlation is, as we referred to the above paragraph, a symmetry that causes Potential Field vectors of the agents to be strongly correlated. With the aid of the generic steps of MAAC concept we are able to mathematically define the Potential Field agents' generic symmetries that cause local minima in PDE.

To put in simulation practice the MAAC generic steps, we have focused on the characteristics of an average Unmanned Surface Vehicle (USV). For example, the configuration space decision for the simulation of the USV was based on the USV maximum speed, size and navigation algorithm. More specifically the steps are the following:

Step 1: Define the Autonomous Monovular Agents, and the size of local Pure Dynamic Environment.

We define the size of the configuration space based on the Monovular agents' kinematic capabilities. Initially, we follow a simplified model as is described below.

We consider two ideal Potential Field Autonomous Agents that have the size of an average USV. The Potential Field Algorithm is used is the AWSPPF algorithm, which described in the previous chapter. The agents are represented by a circle with equal radius of 9m and can reach a maximum speed of 5.14m/sec (10NM). Their local navigational environment covers a surface of 1000x1000m. We use ideal point

mass autonomous potential field agent models to identify the Potential Field algorithm trajectory that is generated in PDE without the noise produced of agents' kinematic capabilities or static obstacles.

In other words, we use PDE to isolate the effect, of any static obstacle, could cause upon the autonomous agent trajectory. Therefore, the navigation trajectory followed by the autonomous agent it will be only related to navigation capabilities of the algorithm in Pure Dynamic Environment (PDE). Furthermore, we have also isolated the agents' trajectories from the effect could cause its kinematic capabilities, in case it doesn't allow the agent to follow the potential field velocity vector accurately. For this reason at this stage, we have used point-mass autonomous agents.

Step 2: Place two Monovular Agents within the same local environment, and try to identify repeating inefficient trajectories.

From the previous step we have defined the Monovular Agent's kinematics models and their local PDE. We place the autonomous Agents in this predefined PDE, so each agent "reacts" to the position and the velocity vector of the other. Then by changing the agents' initial conditions we try to identify repeating inefficient trajectories. The initial conditions refer to when the repulsive force of the Potential Filed Agents take place for first time.

A group of these relative initial conditions cause the autonomous agent to experience long and inefficient trajectories like the ones in Figure 4-1. In this example, the distance between the two Monovular Vehicles while trying to reach their target destination is illustrated in Figure 4-2. Vehicle1 on the left hand side (initial position) is slightly faster that the vehicle2 on the right hand side (initial position). In Figure 4-3 is illustrated the speed diagram of Vehicle1 versus Time, and in Figure 4-4 the speed diagram of Vehicle2 versus Time. By observing these two diagrams we can distinguish that the lower speed, of the slightly faster vehicle, takes a much higher speed value than the lower speed value of the slightly slower speed vehicle.

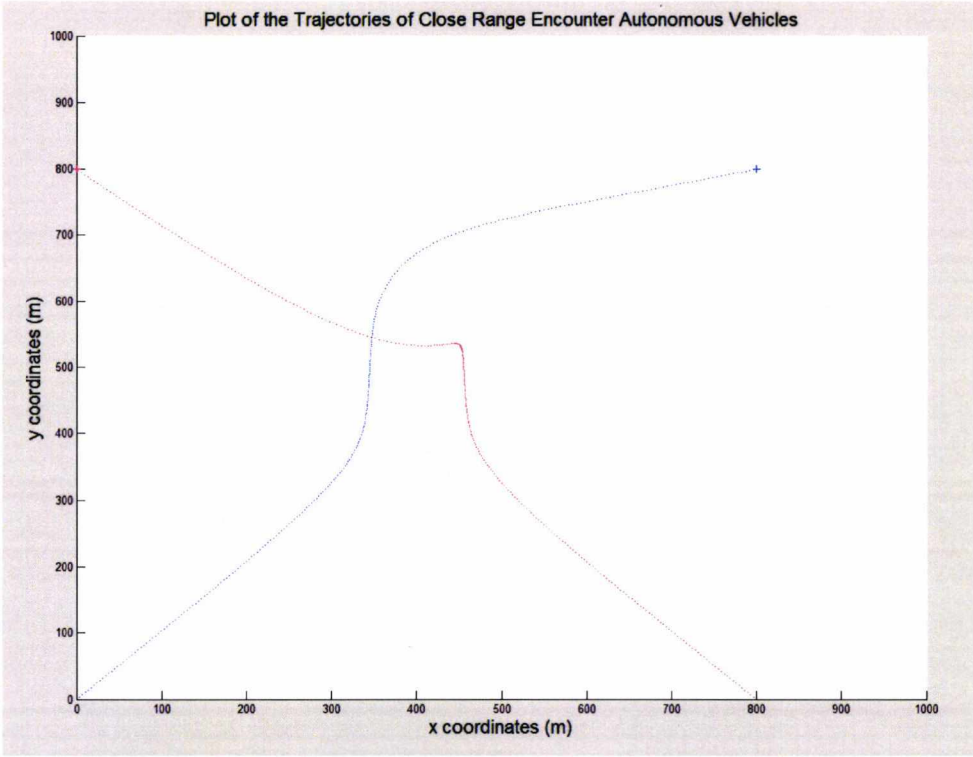


Figure 4-1: Identification of long and inefficient trajectories in Pure Dynamic Environment (PDE) of Potential Field Monovular Vehicles/Agents.

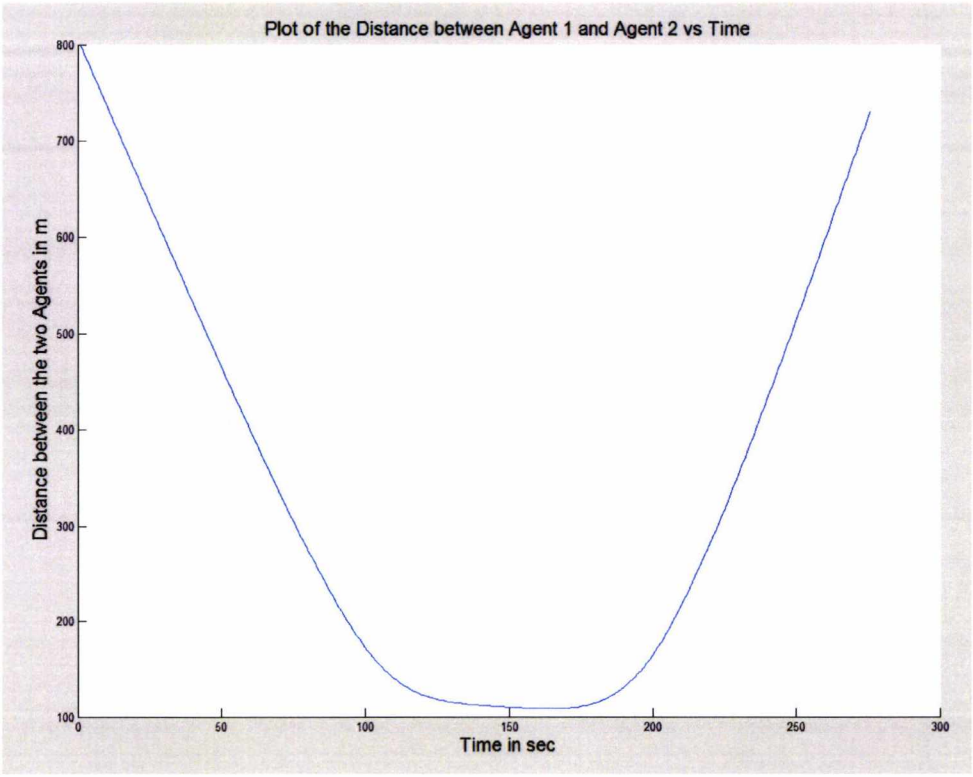


Figure 4-2: Autonomous Vehicle1 |Distance from Autonomous Vehicle2 vs Time when the Monovular Agents are guided by the AWSPPF algorithm.

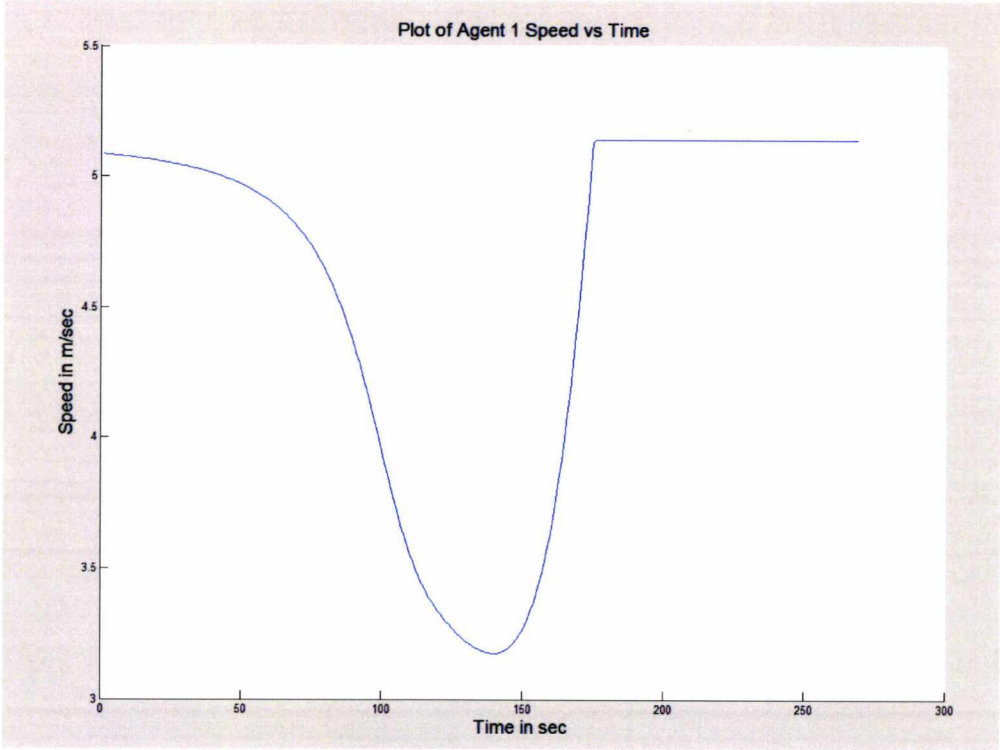


Figure 4-3: Autonomous Vehicle1 Speed Variation due to Autonomous Vehicle2 vs Time when the Autonomous Vehicle is guided by the AWSPPF algorithm.

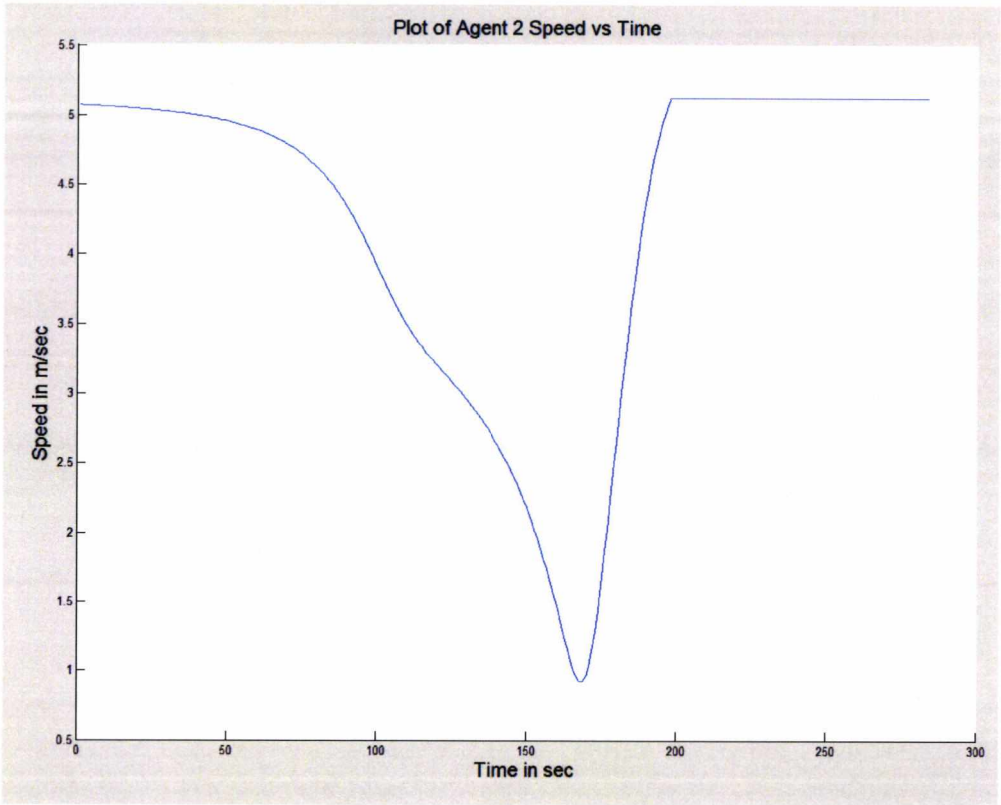


Figure 4-4: Autonomous Vehicle2 Speed Variation due to Autonomous Vehicle1 vs Time when the Autonomous Vehicle is guided by the AWSPPF algorithm.

In some cases these trajectories end in a navigational deadlock as is illustrated in Figure 4-5, and both the autonomous agents do not reach their target destination.

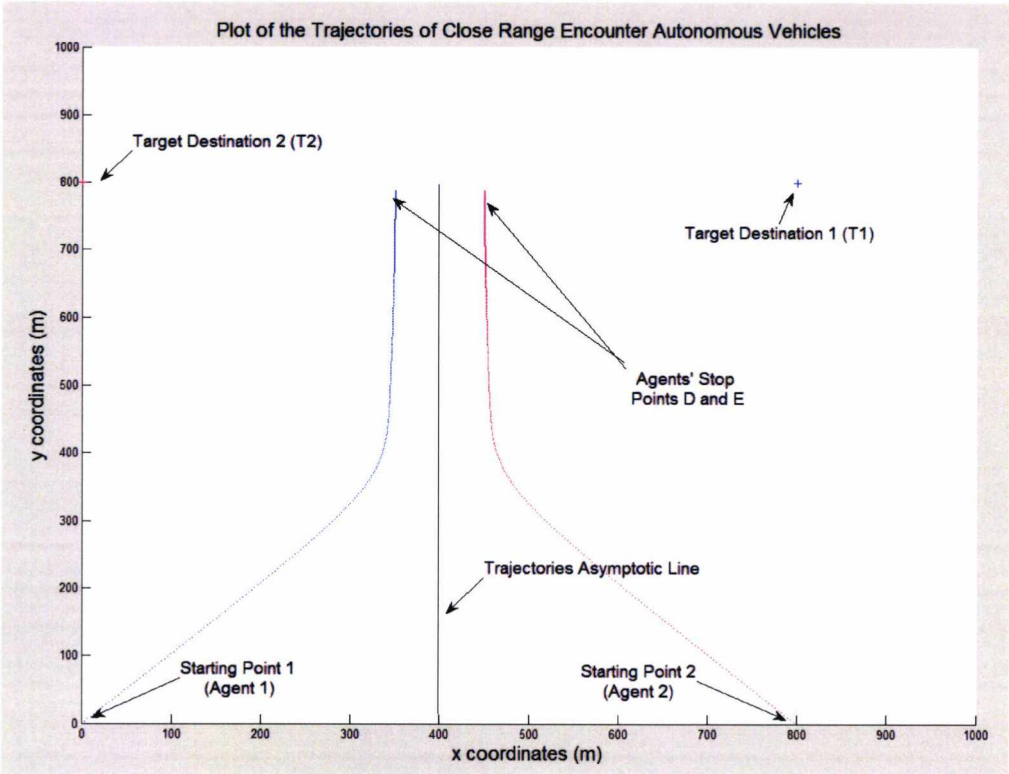


Figure 4-5: Identification of autonomous agents/vehicles deadlocks in PDE of Potential Monovular agents guided by the AWSPPF algorithm.

In Figure 4-5 we can see that the trajectories are symmetrical and are separated by an asymptotic line. In this case, the distance between the two Monovular Vehicles while are in a navigational deadlock is illustrated in Figure 4-6. Vehicle1 and Vehicle2 have the same diagrams of Speed versus Time that are illustrated in Figure 4-7 and Figure 4-8.

It has been possible for us to constantly reproduce the ill performance of the Potential Field Monovular agents of Figure 4-5 based on similar initial relative conditions.

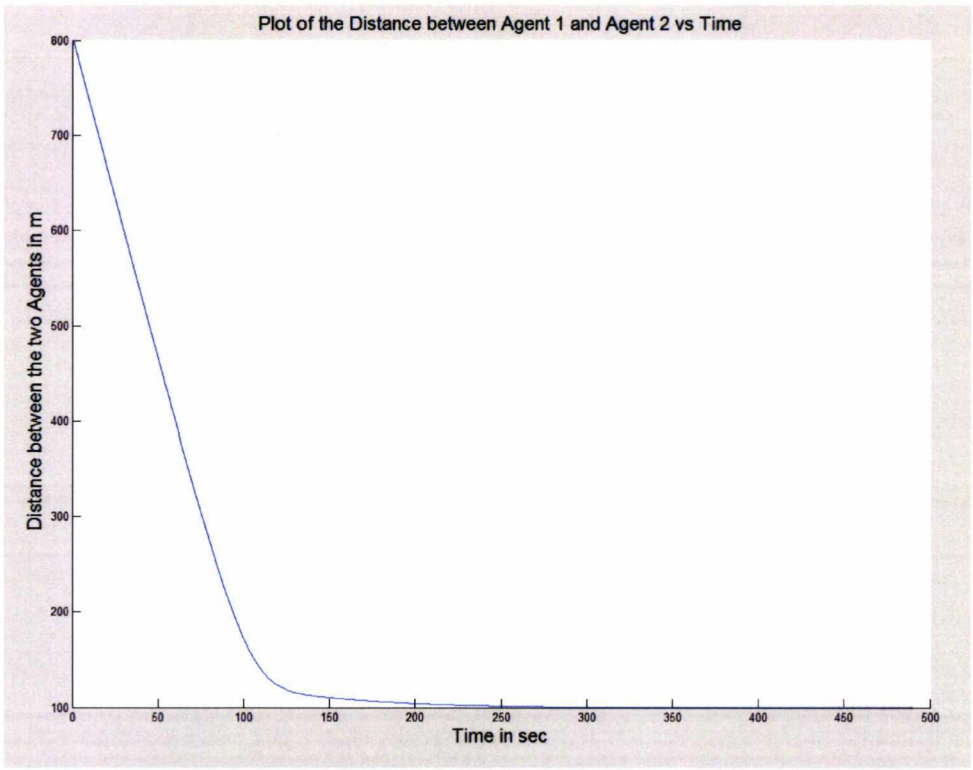


Figure 4-6: Autonomous Vehicle1 |Distance from Autonomous Vehicle2 vs Time when the Monovular Vehicles are in navigational deadlock.

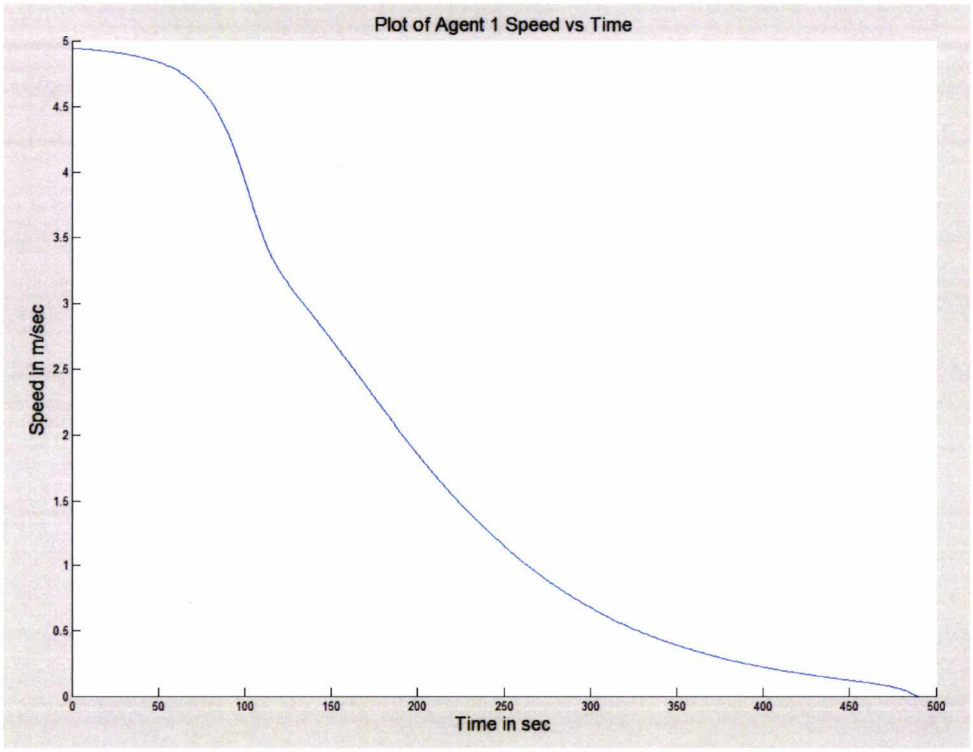


Figure 4-7: Autonomous Vehicle1 Speed Variation due to Autonomous Vehicle2 vs Time when the Autonomous Vehicles are in navigational deadlock.

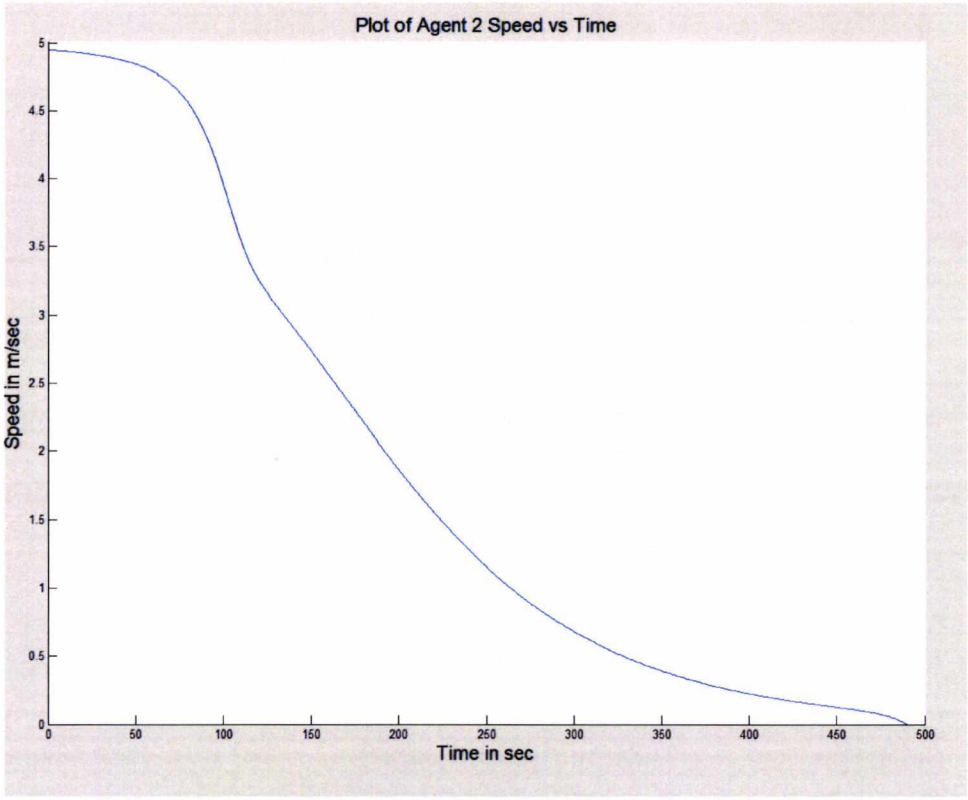


Figure 4-8: Autonomous Vehicle2 Speed Variation due to Autonomous Vehicle1 vs Time when the Autonomous Vehicles are in navigational deadlock.

Step 3: Identify the generic causes of the inefficient repeating trajectories of the autonomous Monovular agents.

If we observe the behaviour of both the navigational vectors of the two Monovular agents' trajectories that are illustrated in Figure 4-1 and Figure 4-5 in time domain, we can identify the following:

- The agents' directional vectors are strongly correlated to the rate of change of both direction and magnitude.
- The agents' speed are strongly correlated
- The coordinates of the autonomous agents are symmetrical or close symmetrical.

Step 4: Define the causes of the inefficient repeating trajectories of the Potential Field Monovular Agents while navigating in the PDE.

Based on the previous steps we have identified that the Monovular Agents that are located in a PDE and satisfy relative conditions correlation and certain coordinates' symmetries experience inefficient Trajectories and deadlocks. In this step we define the exact symmetries that cause these deadlock or inefficient trajectories. In addition, we have defined a new state the Potential Monovular Agents are in while experience inefficient trajectories. We have named this new state Trajectory Equilibrium State (TES). The most extreme case of this state is when the Potential Monovular Agents deadlock and never reach their target destinations we have named this state Absolute TES. On the other hand, when the autonomous agents reach their target destination but their trajectories are inefficient they are in a similar state that we have called Close TES. As we will analytically describe in the next paragraphs both Absolute and Close TES derive from the same symmetry we have called Potential Monovular Agent Symmetry (PMAS), but in close TES the symmetry is not 100% accurate.

In the following paragraphs we mathematically define the PMAS in relation to Monovular Agents' coordinates and Target Destination coordinates. The Potential algorithm we have used for this definition is the Active Window Single Point Potential Field (AWSPPF) algorithm we have designed in chapter 3.

Initially, by simulation experiments, we have established that the PMAS, which is illustrated in Figure 4-9 can cause an Absolute TES trajectory of Figure 4-5. PMAS is formed based on the coordinates of the agents' initial positions and their target destinations.

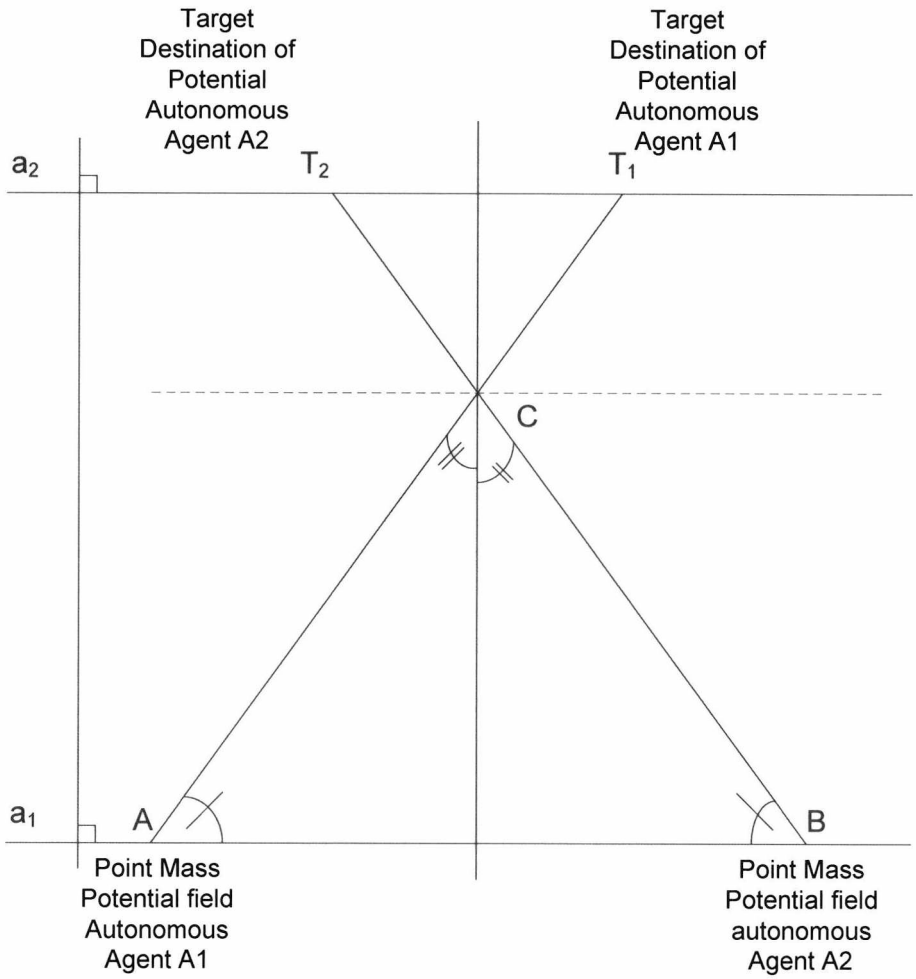


Figure 4-9: PMAS Generic symmetries between the initial positions A and B of the autonomous agents and their target destination, which results in absolute TES.

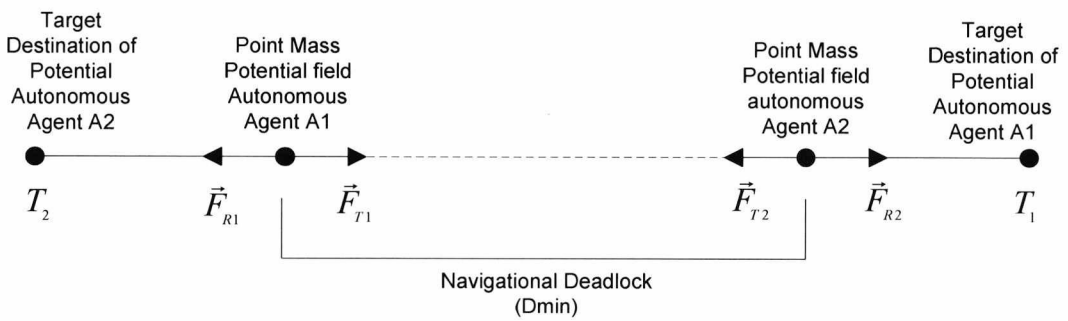


Figure 4-10: Absolute TES final state due to PMAS.

From Figure 4-9 we can derive the generic equations and conditions that govern PMAS. These equations and conditions are the following:

$$AT_1 = BT_2 \tag{4.1}$$

$$AC = BC \;\&\; AC < AT_1 \tag{4.2}$$

$$CT_1 = CT_2 \;\&\; CT_1 < AT_1 \tag{4.3}$$

T_1T_2 parallel to AB (4.4)

Based on the above equations we can conclude that the symmetries are satisfied based on concentric circles (with centre C) relation between the Potential Autonomous Agents' coordinates and their Target Destinations coordinates as shown in Figure 4-11.

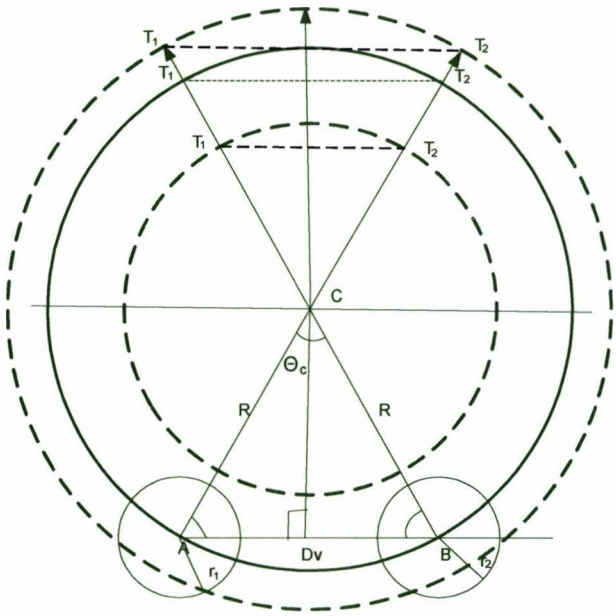


Figure 4-11: PMAS symmetry concentric relation between the Potential Monovular Agents and their target destinations coordinates.

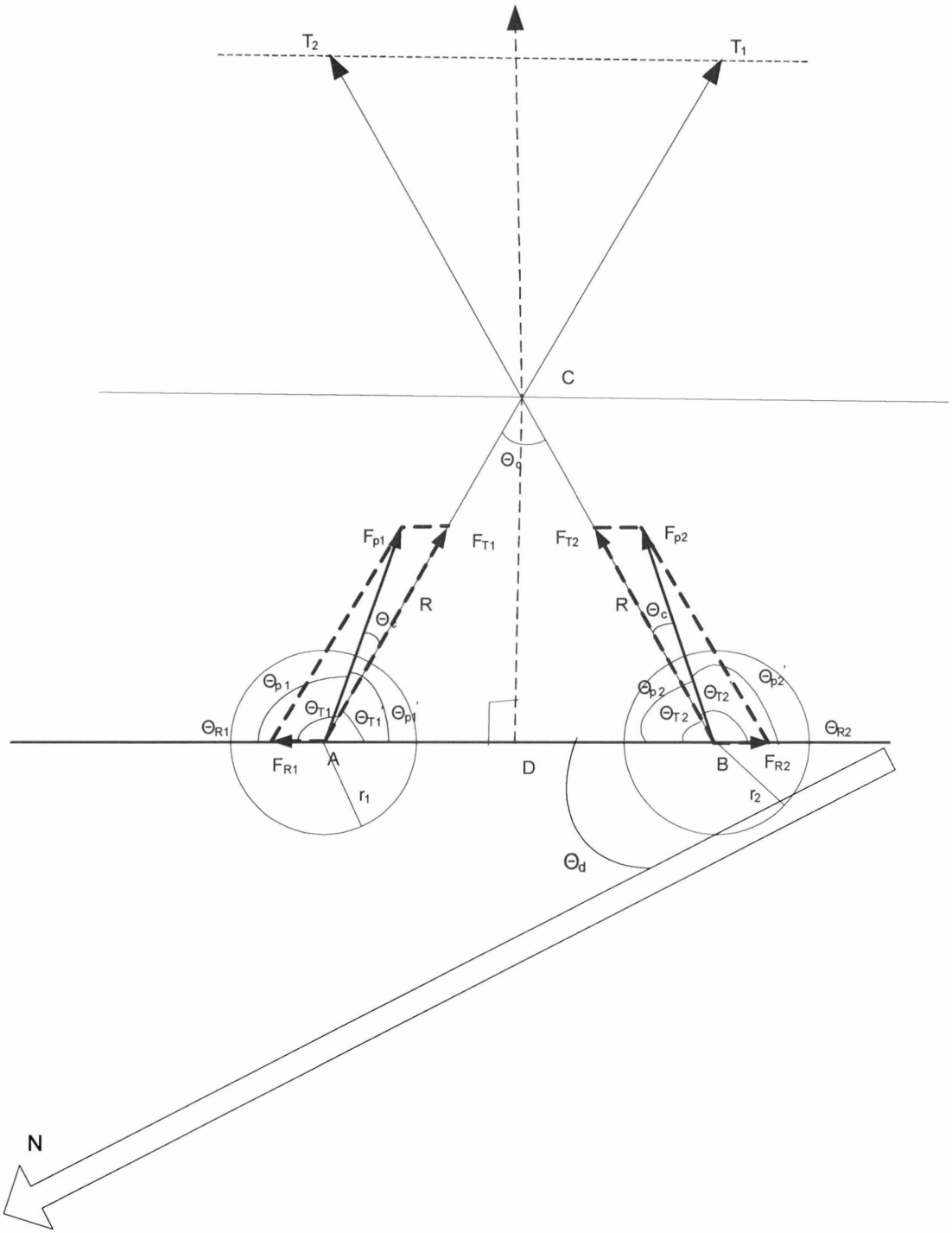


Figure 4-12: Potential Agents Vectors analysis when in PMAS.

In Figure 4-12 are illustrated the Monovular Potential Agents navigational vectors and their angles due to PMAS symmetry. From the (4.1), (4.2), (4.3) and (4.4) is obvious that angles θ_{T1}' and θ_{T2} are equal, and also θ_{T1} and θ_{T2} are supplementary.

Therefore, the following equations are satisfied:

$$\theta_{T1}' = \theta_{T2} \tag{4.5}$$

$$\theta_{T1} + \theta_{T2} = 180 \tag{4.6}$$

Where

θ_{T1}' and θ_{T2} are the angles of ABC triangle of PMAS as indicated in Figure 4-12.

θ_{T1} and θ_{T2} direction angles towards the Monovular Potential Agents Target Destinations T1 and T2

To understand the effect of PMAS on the Monovular Potential Agents we consider two Monovular agents, which are located in the same PDE and are guided by identical Potential Field repulsive and attractive forces (Monovular Agents). These forces are described by the following equations:

Repulsive force equations of the two Monovular agents (due to each other in PDE):

$$\overrightarrow{F_{R1}} = \frac{F_{CR} W^n}{D^n} \left(\frac{x_2 - x_1}{D} \hat{x} + \frac{y_2 - y_1}{D} \hat{y} \right) \tag{4.7}$$

And

$$\overrightarrow{F_{R2}} = \frac{F_{CR} W^n}{D^n} \left(\frac{x_1 - x_2}{D} \hat{x} + \frac{y_1 - y_2}{D} \hat{y} \right) \tag{4.8}$$

Attractive force equations of the two Monovular agents due to their target destinations

$$\overrightarrow{F_{T1}} = F_{CT} \left(\frac{x_2 - x_1}{D_T} \hat{x} + \frac{y_2 - y_1}{D_T} \hat{y} \right) \tag{4.9}$$

And

$$\overrightarrow{F_{T2}} = F_{CT} \left(\frac{x_1 - x_2}{D_T} \hat{x} + \frac{y_1 - y_2}{D_T} \hat{y} \right) \tag{4.10}$$

The Potential Field Navigation equation for agents 1 and 2.

$$\vec{F}_{P1} = \vec{F}_{T1} + \vec{F}_{R1} \tag{4.11}$$

And

$$\vec{F}_{P2} = \vec{F}_{T2} + \vec{F}_{R2} \tag{4.12}$$

Where

D is the distance between the two agents' centre of gravities

(x_1, y_1) and (x_2, y_2) are the coordinates of the centre of gravities of the agents

F_{CR} Potential Field algorithm repulsive force constant

F_{CT} Potential Field algorithm Attractive force constant

W is the agent width

We can see from the above equations that because the two agents are Monovular their repulsive force constants F_{CR}, their attractive force constants F_{CT} and their W are equal.

F_{P1}, F_{P2} the potential field resultant force of the Monovular Agent 1,2

F_{T1}, F_{T2} the target destination attractive force of the Monovular Agents 1,2

F_{R1}, F_{R2} the repulsive forces of the Monovular Agents 1,2

θ_D is the angle between the True North, which is our level of reference and the line defined by the points of A and B. The line segment AB is equal to the distance of the two Monovular agents' centre of gravities. We consider $\theta_D = 0$ for simplicity and clarity of the equation (4.20) to (4.23) that we will come across later in this chapter.

θ_{P1}, θ_{P2} are the angles of the direction of the two resultant potential force vectors of the Monovular Agents 1,2.

θ_{T1}, θ_{T2} the angles of the directions of the attractive Potential Vector towards the Target Destinations of the Monovular agents 1, 2.

θ_{R1}, θ_{R2} the angles of the directions of the Repulsive Potential Vector of the Monovular agents 1, 2.

D is the distance between the two Monovular agents' centre points (A, B).

D_{T1}, D_{T2} is the autonomous agent distances from their targets destinations based on the agents' current location.

$\mathrm{T}_1(x_{t1}, y_{t1})$ and $\mathrm{T}_2(x_{t2}, y_{t2})$ are the target destination coordinates of the Monovular agents 1, 2.

$\mathrm{A}(x_1, y_1)$ and $\mathrm{B}(x_2, y_2)$ are the Monovular agents' current coordinates (instantaneous coordinates).

From Figure 4-12 we also have:

$$\theta_{T1} = 180 - \theta'_{T1}, \theta_{T2} = 180 - \theta'_{T2}$$

$$\theta_{P1} = 180 - \theta'_{P1}, \theta_{P2} = 180 - \theta'_{P2}$$

And from the above definition of distances D, D_{T1} and D_{T2} we have:

$$D = AB,\ D_{T1} = AT_1 \text{ and } D_{T2} = BT_2$$

From equations (4.8) we have:

$$\overrightarrow{F_{R1}} = \frac{F_{CR} W^n}{D^n} \left(\frac{x_2 - x_1}{D} \hat{x} + \frac{y_2 - y_1}{D} \hat{y} \right) \qquad (4.13)$$

And

$$\overrightarrow{F_{R2}} = \frac{F_{CR} W^n}{D^n} \left(\frac{x_1 - x_2}{D} \hat{x} + \frac{y_1 - y_2}{D} \hat{y} \right) \qquad (4.14)$$

From the above two equations we have

$$F_{R1} = -F_{R2} \qquad (4.15)$$

Therefore

$$\theta_{R1} = -\theta_{R2} \qquad (4.16)$$

From the PMAS symmetry we have equation(4.6), which is equivalent to:

$$\theta_{T1} + \theta_{T2} = 180 \Leftrightarrow \theta_{T1} = 180 - \theta_{T2} \qquad (4.17)$$

From the cosine and sine definition we can write the equations (4.7), (4.8), (4.9) and (4.10) in the following form:

$$\vec{F}_{T1} = F_{T1}\cos\theta_{T1}\hat{x} + F_{T1}\sin\theta_{T1}\hat{y} \qquad (4.18)$$

And

$$\vec{F}_{T2} = F_{T2}\cos\theta_{T2}\hat{x} + F_{T2}\sin\theta_{T2}\hat{y} \qquad (4.19)$$

The above equations describe the Attractive force of the Monovular Potential Field Agents in terms of cosine and sine, and the equation below describes the Repulsive force of the Agents in terms of cosine and sine.

$$\overrightarrow{F_{R1}} = \frac{F_{CR}}{D_{ob}^{\ 2}}\left(\cos\theta_{R1}\hat{x} + \sin\theta_{R1}\hat{y}\right) \qquad (4.20)$$

And

$$\overrightarrow{F_{R2}} = \frac{F_{CR}}{D_{ob}^{\ 2}}\left(\cos\theta_{R2}\hat{x} + \sin\theta_{R2}\hat{y}\right) \qquad (4.21)$$

From equations (4.16), (4.17), (4.18), (4.19), (4.20) and (4.21) we can rewrite the equation (4.11) and (4.12) for both agents:

$$\overrightarrow{F_{P1}} = (F_{T1}\cos\theta_{T1} + F_{R1}\cos\theta_{R1})\hat{x} + (F_{T1}\sin\theta_{T1} + F_{R1}\sin\theta_{R1})\hat{y} \qquad (4.22)$$

And

$$\overrightarrow{F_{P2}} = (F_{T2}\cos\theta_{T2} + F_{R2}\cos\theta_{R2})\hat{x} + (F_{T2}\sin\theta_{T2} + F_{R2}\sin\theta_{R2})\hat{y} \qquad (4.23)$$

The two Potential Field autonomous agents are Monovular, and they have identical Potential Field attractive force magnitude, therefore:

$$\left|\overrightarrow{F_{T1}}\right| = \left|\overrightarrow{F_{T2}}\right| \qquad (4.24)$$

If we rewrite equation (4.22) from equations (4.15), (4.16), (4.17), and (4.24) we have:

$$F_{P1} = (F_{T2}\cos(180-\theta_{T2}) - F_{R2}\cos(-\theta_{R1})\hat{x} + (F_{T2}\sin(180-\theta_{T1}) - F_{R2}\sin(-\theta_{R2}))\hat{y}$$

Therefore

$$F_{P1} = (-F_{T2}\cos(\theta_{T2}) - F_{R2}\cos(\theta_{R1})\hat{x} + (F_{T2}\sin(\theta_{T1}) + F_{R2}\sin(\theta_{R2}))\hat{y}$$

Therefore the Potential Field Force navigational magnitude is equal to:

$$\|F_{P1}\| = \sqrt{[-(F_{T2}\cos\theta_{T2} + F_{R2}\cos\theta_{R2})]^2 + (F_{T2}\sin\theta_{T2} + F_{R2}\sin\theta_{R2})^2} =$$
$$\sqrt{(F_{T2}\cos\theta_{T2} + \cos\theta_{R2}F_{R2})^2 + (F_{T2}\sin\theta_{T2} + F_{R2}\sin\theta_{R2})^2} = \|F_{P2}\|$$

(4.25)

Therefore, we have proved that $\left|\overrightarrow{F_{P2}}\right| = \left|\overrightarrow{F_{P1}}\right|$ (4.26)

In addition, we can prove that

$$\theta_{P1} = 180 - \theta_{P2} \quad (4.27)$$

From the cosine definition we have:

$$\cos\theta_{P1} = \frac{(F_{T1}\cos\theta_{T1} + F_{R1}\cos\theta_{R1})}{\sqrt{(F_{T1}\cos\theta_{T1} + \cos\theta_{R1}F_{R1})^2 + (F_{T1}\sin\theta_{T1} + F_{R1}\sin\theta_{R1})^2}}$$

(4.28)

Based on equations (4.15), (4.16), (4.17), (4.24) and (4.26) the above equation becomes:

$$\cos\theta_{P1} = \frac{(-F_{T2}\cos\theta_{T2} - F_{R2}\cos\theta_{R2})}{\left|\overrightarrow{F_{P2}}\right|} =$$

$$-\frac{(F_{T2}\cos\theta_{T2} + F_{R2}\cos\theta_{R2})}{\left|\overrightarrow{F_{P2}}\right|} = -\cos\theta_{P2}$$

Therefore

$$\cos\theta_{P1} = -\cos\theta_{P2} \Leftrightarrow \theta_{P1} = 180 - \theta_{P2} \qquad (4.29)$$

Finally, based on the PMAS, we can also prove that velocity magnitudes of the two agents have the same magnitude. Base on the equation (3.13) of chapter 3, (4.24) and (4.26) we have:

$$\left|\overrightarrow{V_1}\right| = \frac{\left|\overrightarrow{F_{P1}}\right|}{\left|\overrightarrow{F_{T1}}\right|} V_{MAX} = \left|\overrightarrow{V_2}\right| \qquad (4.30)$$

From the equations (4.26), (4.29) and (4.30) we proved that the two Monovular potential field agents' resultant instantaneous forces have the same magnitude and supplementary angles when in PMAS. PMAS also causes the same instantaneous speed magnitude.

After we have analysed the results of PMAS when the $t = 0$ (t is the time from the moment the Monovular Potential Agents are with its other active window), it is time to examine if the PMAS symmetry is still valid after time equal dt. If, for this period we assume that the speed of both agents is constant, and their directions unchanged, then we have:

$$ds_1 = V_1 dt \qquad (4.31)$$

$$ds_2 = V_2 dt \qquad (4.32)$$

From (4.30) we have that

$$ds_1 = ds_2 = ds \qquad (4.33)$$

To calculate the agents' new coordinates after time dt we have:

(x coordinates for Agents 1 and 2)

$$x_{1+dt} = x_1 + ds\cos\theta_{P1}$$

$$x_{2+dt} = x_2 + ds\cos\theta_{P2}$$

And

(y coordinates for Agents 1 and 2)

$$y_{1+dt} = y_1 + ds\sin\theta_{P1}$$

$$y_{2+dt} = y_2 + ds\sin\theta_{P2}$$

From the equation $\theta_{P1} = 180 - \theta_{P2}$ that we have proved earlier, we have:

$$x_{1+dt} = x_1 - ds\cos\theta_{P2} \qquad (4.34)$$

$$x_{2+dt} = x_2 + ds\cos\theta_{P2} \qquad (4.35)$$

And

$$y_{1+dt} = y_1 + ds\sin\theta_{P2} \qquad (4.36)$$

$$y_{2+dt} = y_2 + ds\sin\theta_{P2} \qquad (4.37)$$

From the above equations we conclude that the PMAS is maintained after dt period of time, since the Agents move equally but in opposite directions on x axis, and equally and in the same direction on y axis.

We approximate dt numerically for the simulation purposes. Therefore, we calculate the speed of the Monovular Agents numerically. In this numeric calculation we assume that the speed and direction of the Agents remain constant in every iteration.

Therefore:

$$S_1 = \sum_{i=0}^{n} |V_{1i}| t_i = \sum_{i=0}^{n} |V_{1i}| t_i = S_2 \qquad (4.38)$$

Additionally, for every iteration of the summation we have $\theta_{P1} = 180 - \theta_{P2}$. While i of the above equation is getting higher, the value of θ_{P1} increases and the value of θ_{P2} decreases. When $\theta_{P1} = 180^\circ$ and $\theta_{P2} = 0^\circ$ the Monovular Potential

Agents are in navigation deadlock. The distance between the Agents when they are in navigation deadlock is given by:

$$\vec{F}_{P1} = \vec{F}_{T1} + \vec{F}_{R1} = 0 \Leftrightarrow \vec{F}_R = -\vec{F}_T \text{ when}$$

$$D = D_{\min} = W \sqrt[n]{\frac{F_{cr}}{F_{ct}}}$$

Because $D = D_{\min}$ at points A, B of Figure 4-13, we have navigational deadlock of the two agents.

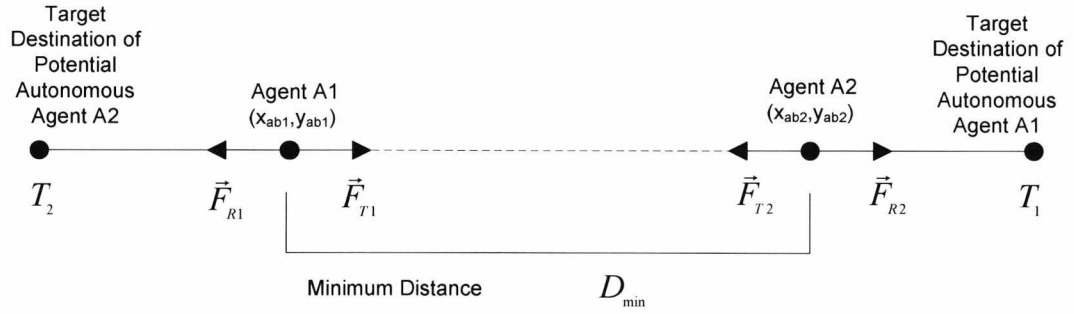


Figure 4-13: Absolute TES deadlock coordinates of the Monovular agents

The two vehicles will stop in a straight line with points (x_{1ab}, y_{1ab}) and (x_{2ab}, y_{2ab}). Where (x_{1ab}, y_{1ab}) and (x_{2ab}, y_{2ab}) are the deadlock coordinates of the two Monovular agents in Absolute TES.

This proves that Potential field algorithm of the AWSPPF nature experience local minima in pure navigation environment. Therefore, the vehicles will stay in this point indefinitely.

To summarise, we have proved that PMAS cause the Potential Navigation Vectors of the Monovular Potential Agents to have equal magnates and supplementary angles of their navigational Potential Vectors at all times. In addition, we can predict the exact deadlock when we know the coordinates of the Monovular Potential Agents Target Destinations. If we don't know the Agents' Target Destinations the possible dead-locks are illustrated in Figure 4-14.

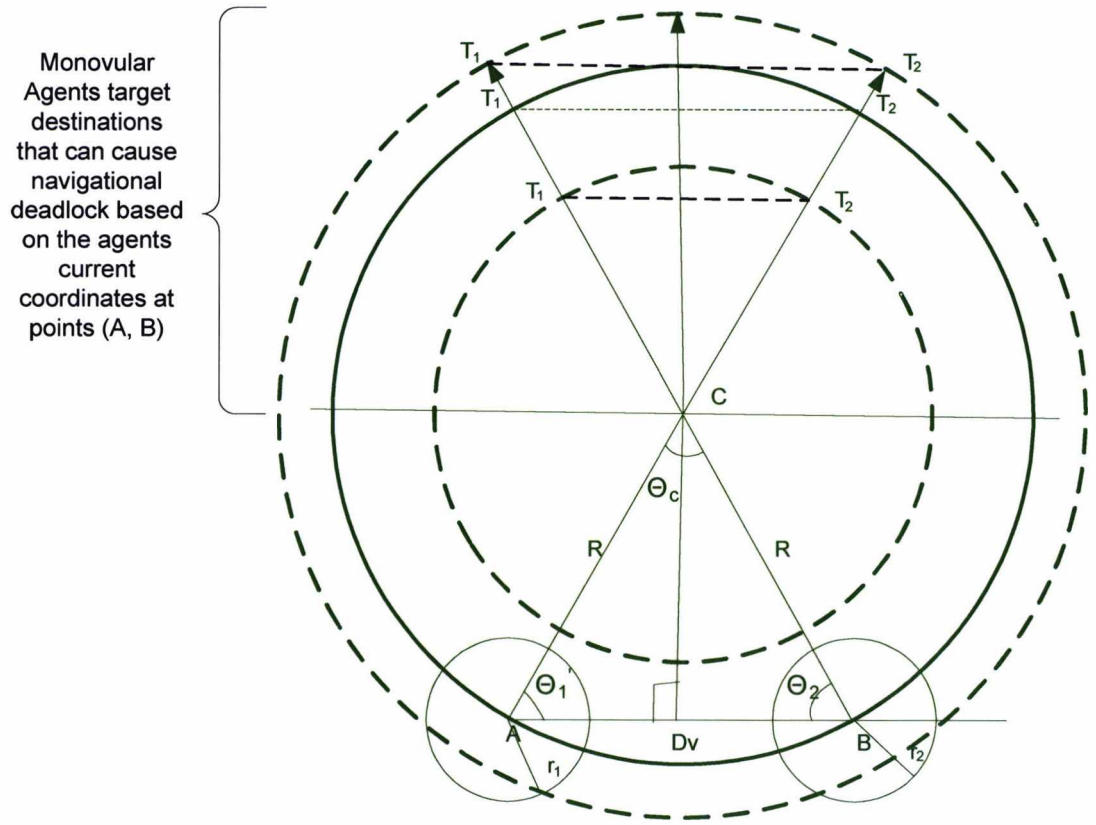


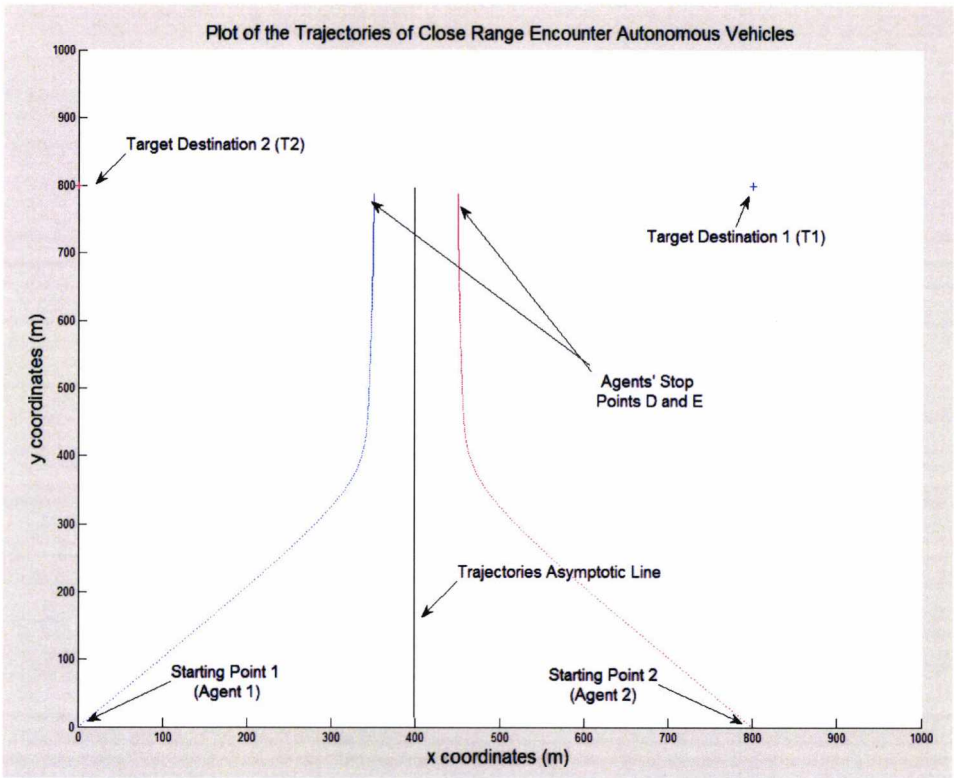
Figure 4-14: the set of Monovular agents target destination coordinates that can cause navigational deadlocks.

From the above figure we can also conclude that the PMAS symmetry applies to all the agents that have circular coordinates of equation (4.39) and satisfy the PMAS equations.

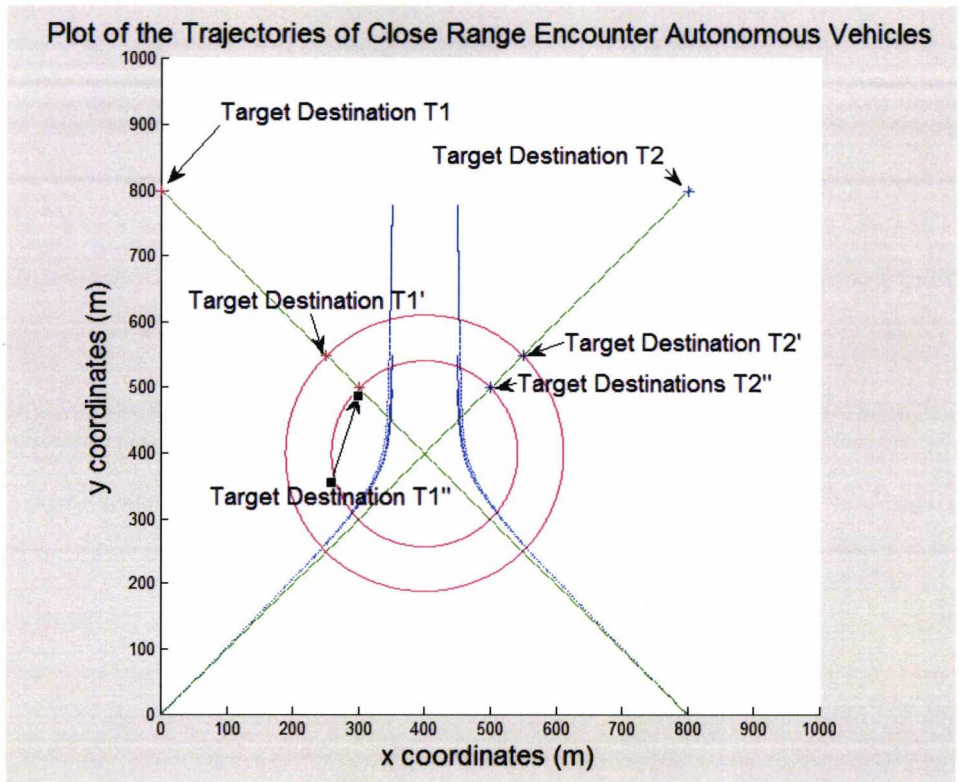
$$(x - x_c)^2 + (y - y_c)^2 = R^2 \qquad (4.39)$$

From the above we have mathematically proved that the PMAS is generic symmetry that when its related equations (4.1), (4.2), (4.3) and (4.4) are satisfied the Monovular Potential Agents are within Absolute TES. In Absolute TES the Agents have mirror inefficient trajectories as shown in 4-15.

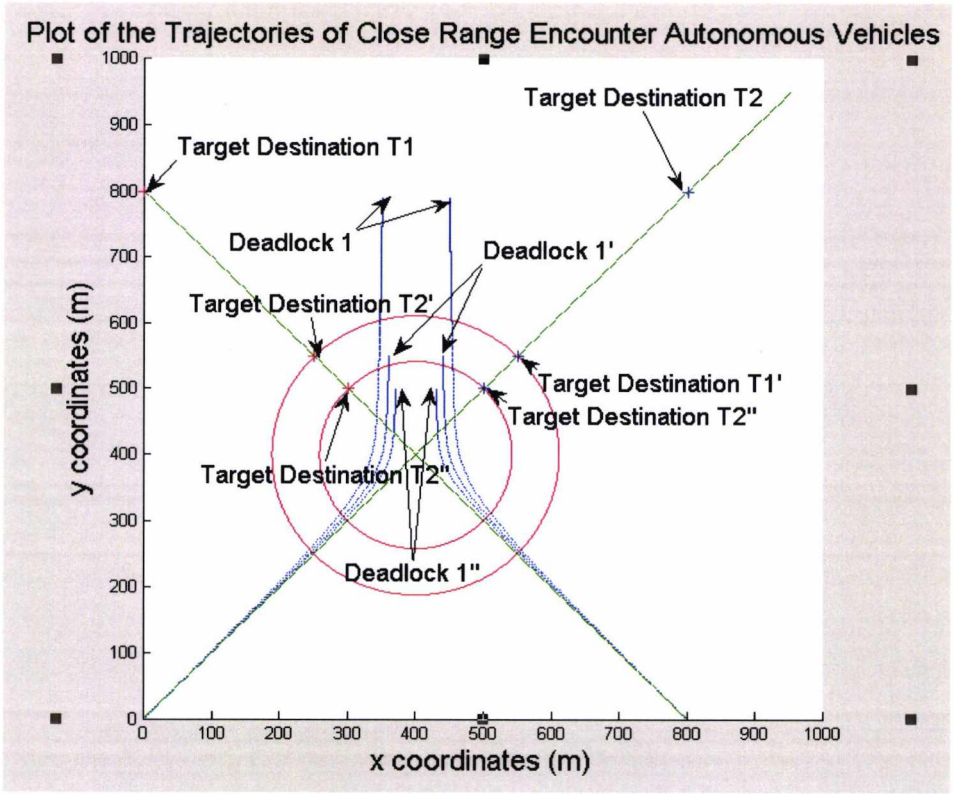
We can practically test that the PMAS is generic based on three simulation results that have derived from four variations of Agents' Target Destinations Coordinates that maintain the validity of the equations (4.1), (4.2), (4.3) and (4.4).



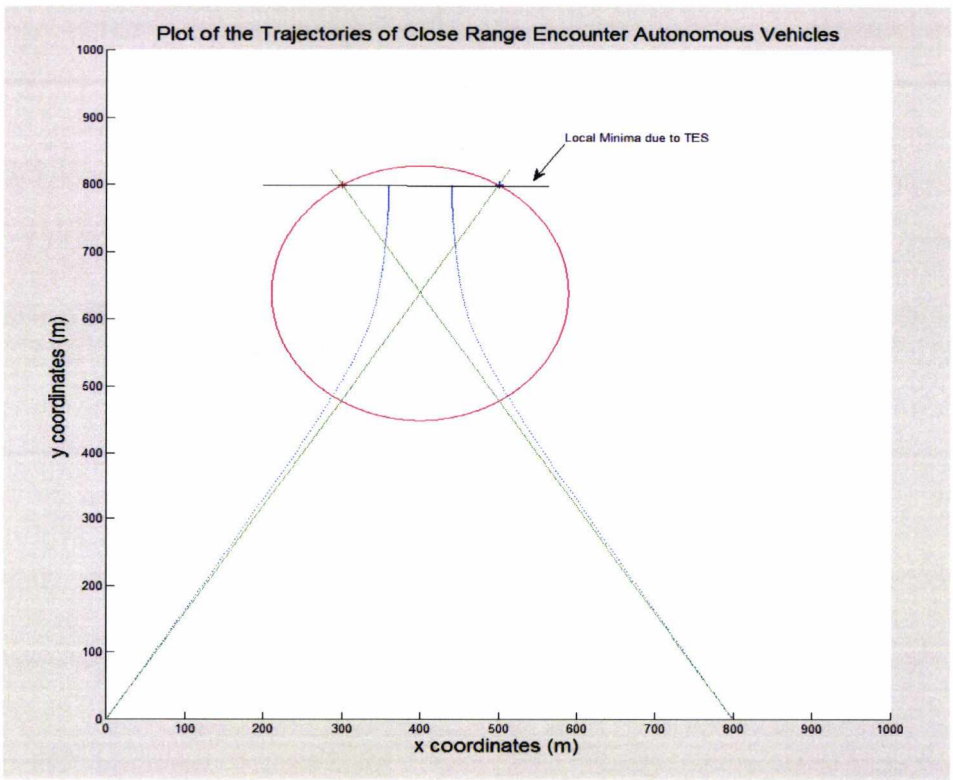
4-15: TES Monovular Potential Agents Mirrored Trajectories due to PMAS



4-16: TES three sets of inefficient deadlock trajectories due to generic PMAS with difference Monovular Potential Agents Target Destinations. These trajectories were generated based on the same algorithmic Dmin and different radius circles that cross the symmetries straight lines.



4-17: TES three sets of inefficient deadlock trajectories due to generic PMAS with difference Monovular Potential Agents Target Destinations. These trajectories were generated based on the different algorithmic Dmin for deadlock clarity, and different radius circles that cross the symmetries straight lines.



4-18: Inefficient deadlock trajectories due to generic PMAS altering the symmetries angles.

In 4-16 we can observe three Monovular Potential Agents deadlocks caused by the PMAS with two different Agents Target Destinations, causing the Agents to deadlock in two different coordinates' sets. As we can observe from this figure the two different Agents' Target sets of Target Destinations coordinates lying on the PMAS lines without altering the symmetry's angles. In 4-17 we have the same Agents' Target Destinations type that doesn't change the PMAS angles from the previous figure but with different algorithm Dmin definition for clarity. Finally, in 4-18 we have changed the PMAS symmetry's angles, and we still have a deadlock between the two Monovular Potential Agents. Therefore, based on the above simulation results, we have also tested experimentally the generic validity of PMAS.

From all the above observations and mathematical definitions, we can accurately define Absolute and Close TES. Absolute TES is the state of two Monovular Agents, in which during their motion for avoiding collision, they maintain PMAS symmetry until navigational deadlock. On the other hand, Close TES is the state of two Monovular Agents, in which during their motion for avoiding collision maintain close PMAS symmetry.

An example of inefficient trajectories due to Close TES is illustrated in Figure 4-1, the close PMAS symmetry between the two Monovular Potential Agents causes inefficient trajectories. In other words, we have Close TES when the PMAS is not "perfect", this means that the coordinates and/or the speed are slightly different than the ones cause Absolute TES.

4.2 The Biovular Agents Concept and how to identify inefficient trajectories

Based on the concept of the Monovular Autonomous Agent that we have explained in section 4.1 we can introduce the concept of the Biovular Agents. The word Biovular is inspired by the biological term Biovular, which means from different ova (eggs). Therefore, in this case, we define Biovular the agents that are different in one or more aspects of their dynamic, hardware and software characteristics. The word Biovular is introduced to materialise the Biovular Autonomous Agent Correlation (BAAC) concept, which is interdisciplinary inspired in the same fashion as the MAAC concept from the concept of cross-correlation. To link the BAAC with MAAC concept is important to underline that autocorrelation is cross-correlation of a function with itself. Furthermore, Cross-correlation is similar to convolution of two functions. One of the frequent applications of cross-correlation is the identification of long in duration signal based on a shorter known feature of this signal.

In collision avoidance, we have aimed to use the concept of Biovular Autonomous Agents Correlation (BAAC) to identify the performance inefficiencies of two non identical (Biovular) potential field agents within the same pure dynamic environment. To achieve this we need first to define a way to distinguish between the two autonomous agents and identify the level of the autonomous agents' differentiation. For this reason we have introduced the term of Autonomous Agents Degree of Consanguinity (AAC). Autonomous Agents Consanguinity specifies the level of relationship between of the two Agents' dynamics, hardware and software.

We have proven that Biovular Potential Field algorithms within the same Pure Dynamic Environment (PDE) that have trajectories with high-cross correlation experience inefficient trajectories.

In addition, the higher the correlation of other quantities (e.g. rate of turn, direction etc.) of the two autonomous Agents the higher the chance of long inefficient trajectories and navigational deadlocks.

Based on the results of the MAAC we have identified that similar or identical reflected trajectories can be caused in PDE based on relative initial conditions (e.g. speed, direction and target destination of the autonomous agents). Biovular Potential Agents are not exception but are more tolerant of the cause of Absolute Trajectory Equilibrium State (TES).

The method we have followed to put in practice the BAAC concept in order to identify repeated inefficient trajectories in Biovular Potential Field algorithms, which causes inefficient trajectories are borrowed by the MAAC. The performance of the Potential Field Method when a Potential Field Agent is within the same PDE with a dynamic obstacle, which have a degree AAC with this Agent, can be observed in the following simulation example. More specifically, the collision scenario to examine the AWSPPF algorithm performance is the following:

The autonomous potential Field algorithm characteristics for the Vehicle 1are:
Algorithmic maximum speed: Vmax = 5.14 m/sec
Algorithmic minimum distance: Dmin = 100m
Local environment: 1000x1000m

The second vehicle is a dynamic obstacle with constant speed of 5.14 m/sec and constant direction.
The initial coordinate of the Vehicle and the Dynamic Obstacle are the following:
Autonomous Vehicle: (0, 0)
Dynamic Obstacle: (0, 800)
Target Destination Autonomous Vehicle: (800, 800)
Target Destination Dynamic Obstacle: (0, 800)
The trajectory results of the Biovular Vehicle are illustrated in Figure 4-19. The Potential Autonomous Vehicle Distance from the Biovular Vehicle is indicated in Figure 4-20. Finally, the Potential Autonomous Vehicle Speed Variation due to Biovular Agent is shown in Figure 4-21.

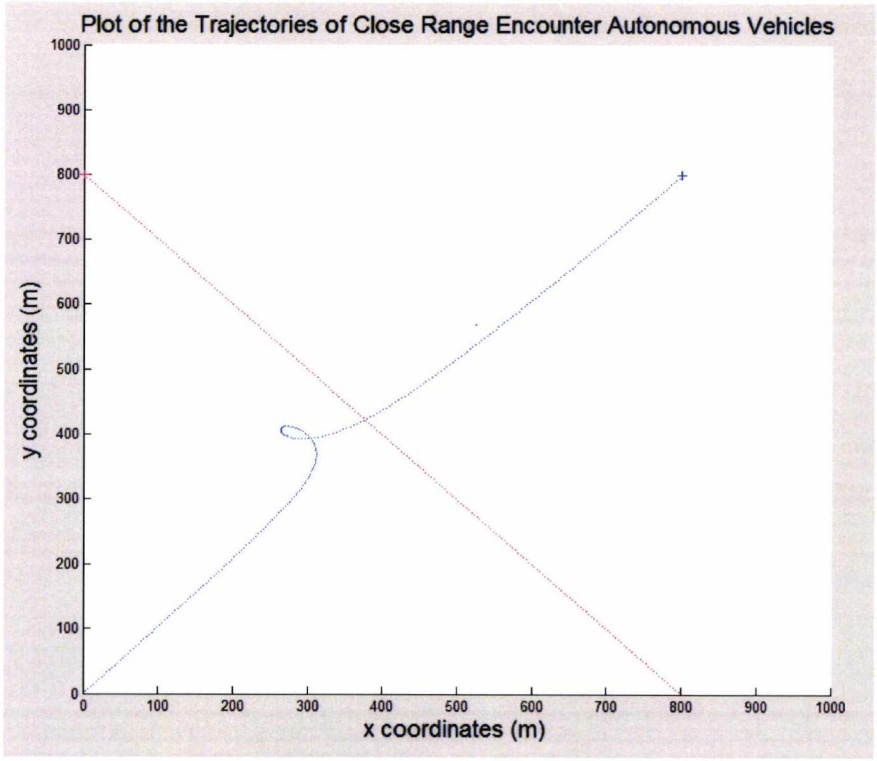


Figure 4-19: Trajectory of AWSPPF Autonomous Vehicle when in cross collision scenario with a Biovular Vehicle in symmetrical coordinates and of the same initial speed.

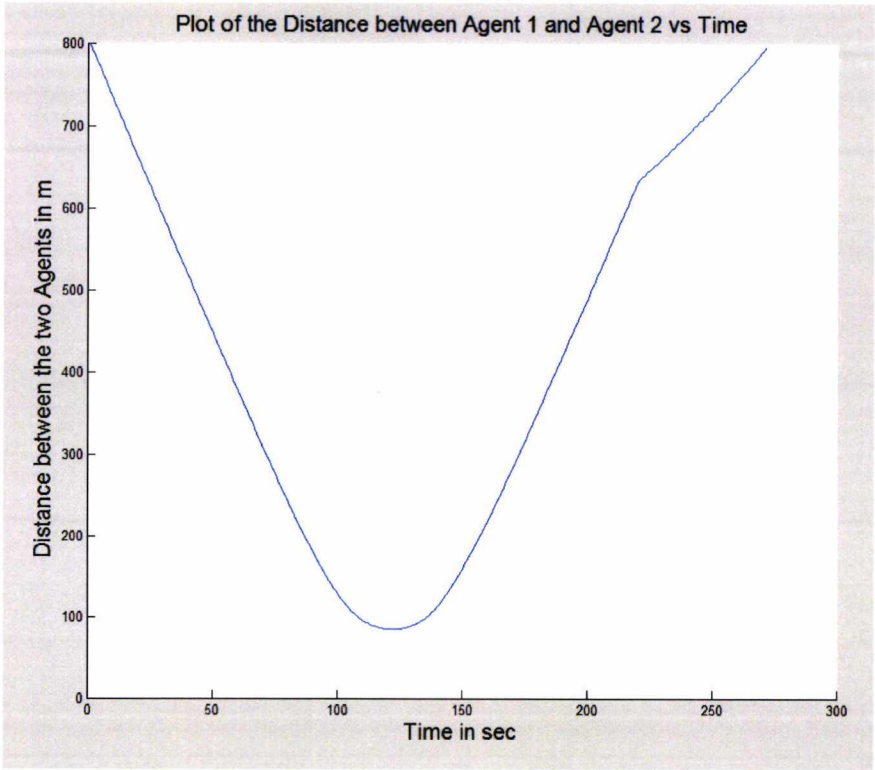


Figure 4-20: Autonomous Vehicle |Distance from Biovular Agent vs Time when the Autonomous Vehicle is guided by the AWSPPF.

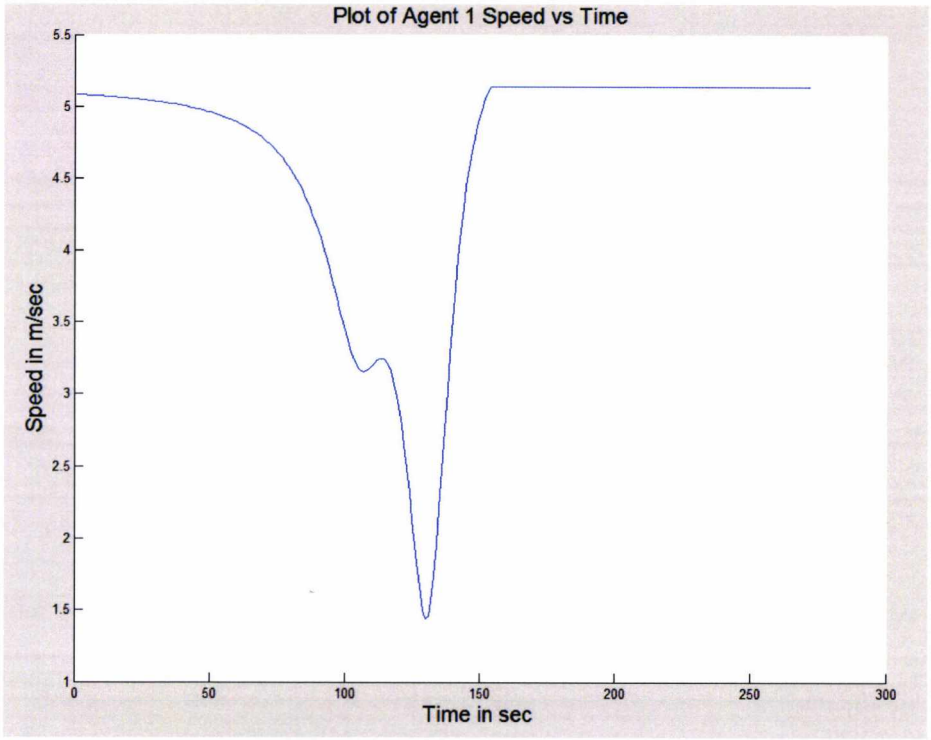


Figure 4-21: Autonomous Vehicle Speed Variation due to the Biovular Agent vs Time when the Autonomous Vehicle is guided by the AWSPPF algorithm.

From the Figure 4-19 we can observe that the Potential Autonomous Vehicle the trajectory is inefficient, since performs a not necessary loop to avoid the Biovular Vehicle. In Figure 4-20, we can observe that when we have Biovular Agents/Vehicles the minimum distance we have defined for the Artificial Potential Field Agents is not valid. Since the theoretical minimum distance of the Artificial Potential Agent is 100m and the actual from the simulation 80m.The reason of this reduction of the minimum distance relies on the fact that the Biovular Agent doesn't vary its speed. If the speed of the Biovular agent is higher enough of the Artificial Potential Field Agent then the two agents could collide. Finally, in Figure 4-21 we can observe an oscillation of the Potential Autonomous Vehicle speed.

From the above results, we have concluded that the PMAS that causes Absolute TES in Monovular Potential Agents, it also causes Close TES in Biovular Agents. We have to note that one of the biovular agents has to be Potential Field Agent for the above to be true. This is very important, since a Biovular Vehicles/Agents are more likely to take place in a real cross collision scenario.

Finally, it is important to say again that the Monovular Autonomous Agents Correlation (MAAC) concept enables us to create a method to identify TES in both Monovular and Biovular Agents/Vehicles.

Chapter 5

5. Trajectory Equilibrium State (TES) Avoidance with the Aid of Monovular Autonomous Agent Correlation MAAC

In the previous chapter we introduced the method that predicts the existence of local minima in Pure Dynamic Environment (PDE). This prediction was possible with the identification and definition of the causes that can lead in local minima. These causes are the initial coordinates geometrical symmetries and speed and direction correlation of the encounter autonomous vehicles/agents. We have identified and defined these causes based on the novel concept of MAAC. In addition, we have observed a new state that the Monovular encounter vehicles/agents are in when their initial conditions are as described above, and we have named this state Trajectory Equilibrium State (TES).

In this chapter based on all above new concepts, as well as using the AWSPPF Algorithm of chapter 3, we introduce a novel algorithm that predicts and avoids this TES and dramatically improved the AWSPPF performance when in TES.

The new algorithm guides each robot independently, although the algorithmic principle is identical for each robot. This approach is a combination of a novel rule-based mathematical algorithm and the AWSPPF navigational method. This method represents the classical processing efficient Potential Field Algorithms with efficient circular Active Window (AW). The need for the above combinational algorithm is due to AWSPPF inability to guide efficiently multi-autonomous vehicles/agents in the same environment due to a Trajectory Equilibrium State (TES). The test results of the novel algorithm yield much improved results than the AWSPPF without a major cost on processing power, and no cost at all on the trajectory smoothness of the autonomous Vehicles/Agents.

5.1 Trajectory Equilibrium State (TES) Detection and Avoidance in Cross Collision Scenario of Monovular Agents

As we have fully described in the previous section, Trajectory Equilibrium State (TES) occurs when two Monovular agents form the PMAS symmetry. The symmetry is illustrated in Figure 5-1.

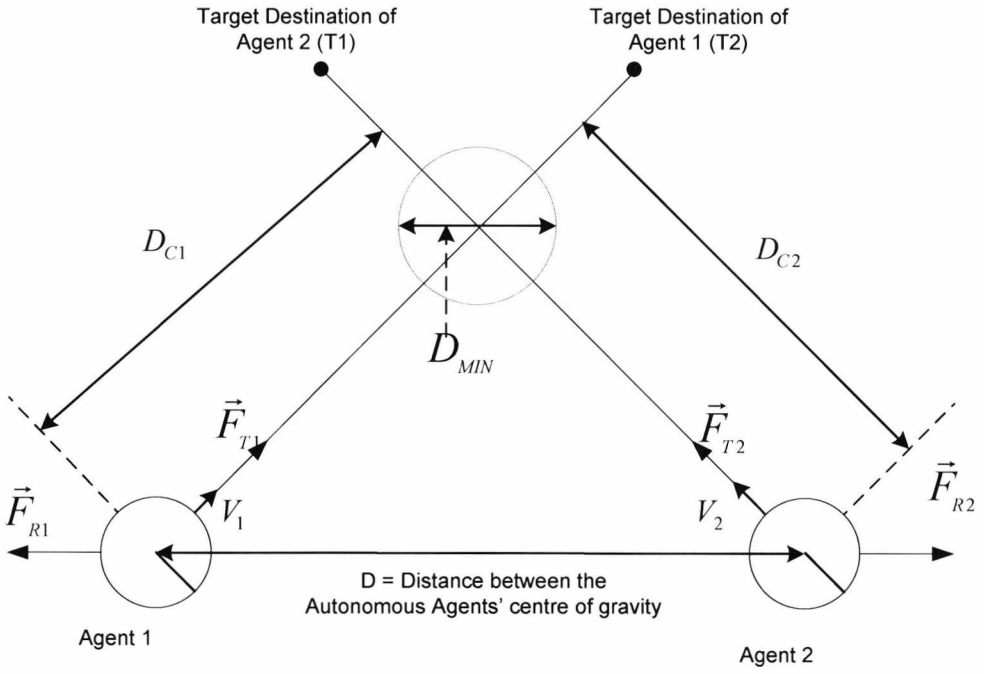


Figure 5-1: PMAS symmetry that causes Absolute TES.

We have managed to isolate the causes that force the Autonomous Potential Field Vehicle/Agent in TES based on the new concept of Monovular Autonomous Agent Correlation (MAAC). We have defined and analysed this concept in chapter 4. We have identified two types of TES, the *Absolute* and the *Close*.

In Absolute TES, based on the PMAS symmetry, we have the distances of each Monovular Potential Vehicle/Agent from the crossing point C to be D_{C1} and D_{C2}, and their instantaneous velocities to be V_1 and V_2 to follow the equations:

$$D_{C1} = D_{C2} \text{ and } \left|\overrightarrow{V_1}\right| = \left|\overrightarrow{V_2}\right| \tag{5.1}$$

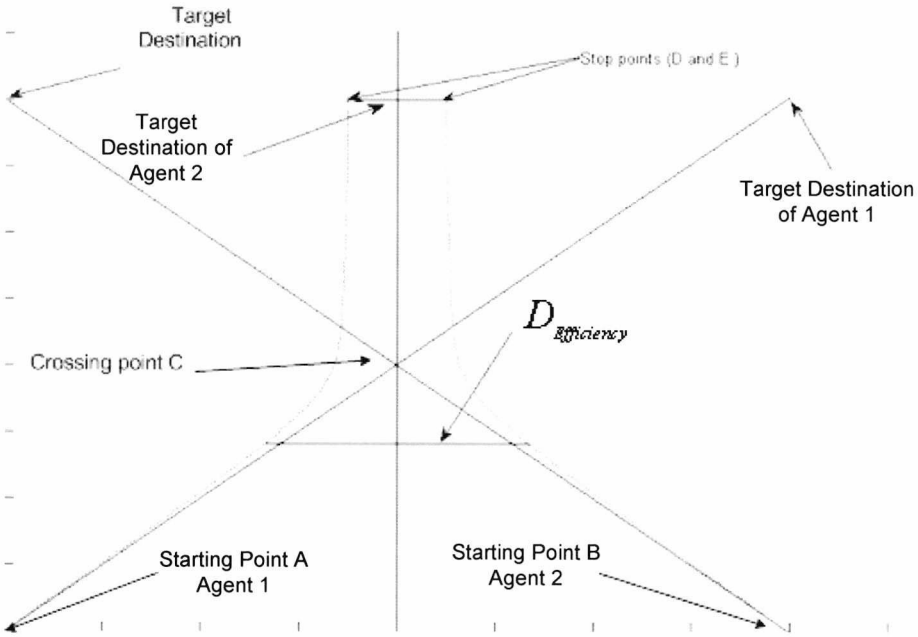


Figure 5-2: Absolute TES Monovular Agent Trajectories (deadlock)

The two Vehicles that are guided by the AWSPPF algorithm divert their trajectories to avoid collision as is indicated in Figure 5-2. This trajectory diversion leads to autonomous navigational deadlock and both Monovular Potential Vehicles/Agents stop in points D and E (Dmin) without reaching their target destinations T_1 and T_2. The Monovular Potential Agents will only stop without reaching their target destination in Absolute TES, where all PMAS equations are accurately satisfied. If the PMAS equations are not accurately satisfied but close to be satisfied, the Monovular Potential Agents are in a Close TES.

PMAS equations:

$$AT_1 = BT_2 \tag{5.2}$$

$$AC = BC \;\&\; AC < AT_1 \tag{5.3}$$

$$CT_1 = CT_2 \;\&\; CT_1 < AT_1 \qquad (5.4)$$

T_1T_2 parallel to AB (5.5)

In other words we have Close TES when:

$$D_{MIN} \leq D < D_{Efficiency} \qquad (5.6)$$

In equation (5.6), D_{MIN} is the minimum distance between the two Monovular Potential Agents so the non-linear effect of the equations $\overrightarrow{F_{R1}} = \frac{F_{CR}W^n}{D^n}\left(\frac{x_2 - x_1}{D}\hat{x} + \frac{y_2 - y_1}{D}\hat{y}\right)$ and F_{R2} is not apparent. $D_{Efficiency}$ is based on the percentage of direction diversion of each agent from its original direction (without the presence of the other agent) towards their target destinations. For this simulation we have found experimentally that 5% direction diversion of its agent original direction vector gives satisfactory results. An example of Close TES is illustrated in Figure 5-3.

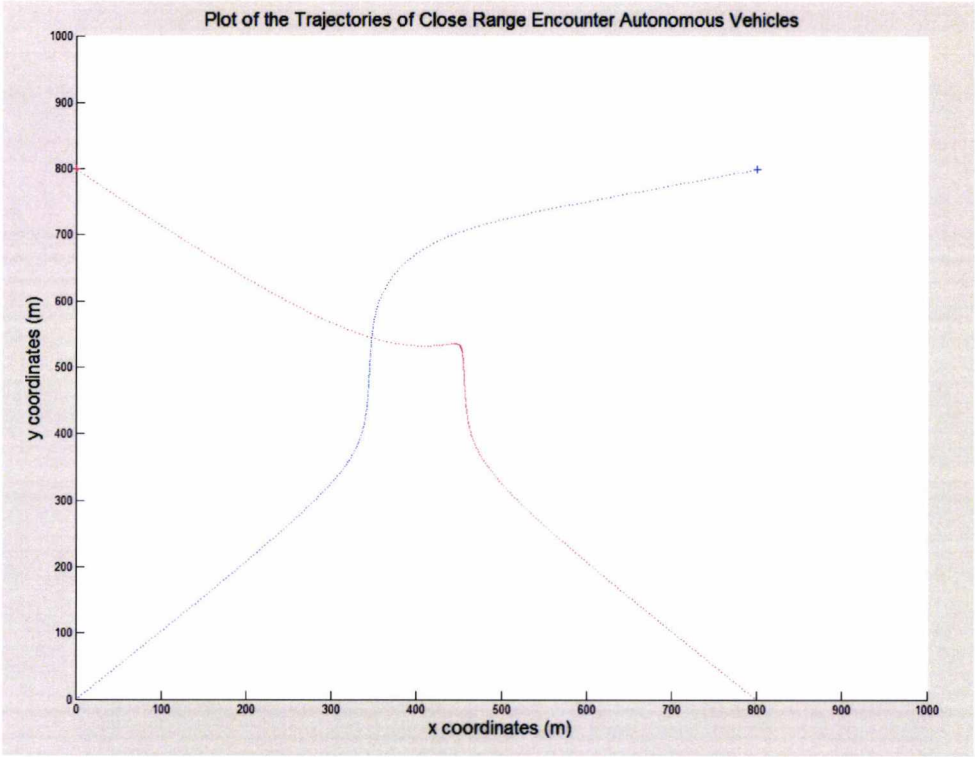


Figure 5-3: Close TES Monovular Potential Agents' inefficient trajectories

When the Monovular Potential Vehicles/Agents are in either Absolute or Close TES we have designed a novel rule based mathematical algorithm that avoids TES. Close and Absolute TES are defined in Chapter 4. We explain this algorithm below.

TES Detection and Avoidance: This algorithm maintains close to straight line efficient trajectories for the Agents in cases of possible collision by adjusting separately their speeds. The TES Detection and Avoidance algorithm requires the following input data to function: their coordinates (x_1, y_1) and (x_2, y_2), the Agents' instantaneous velocities V_1 and V_2 (instantaneous velocity is equal to $\dot{s}$ where s is distance) at these coordinates.

The TES Detection and Avoidance algorithm operation is based on the following steps which execute independently within each Autonomous Agent.

- The instantaneous velocity vectors (V_1 and V_2) include the instantaneous direction for each Agent. Based on these directions, the algorithm calculates if there is a crossing point between the two Agents' trajectories. If so, the al-

gorithm calculates the crossing point coordinates (x_C, y_C), and the distances D_{C1} and D_{C2} from the crossing point C Figure 5-2.

- The next step is to verify if equation 5-7 is satisfied.

$$D \leq D_{Efficiency} \qquad (5\text{-}7)$$

If the above equation is satisfied, the algorithm continues to the next step, otherwise it returns to the beginning.

- The algorithm now determines if the Agents are in Absolute or Close TES.
- If in Absolute TES, the random speed generator is activated individually for both robots until the equilibrium of speeds and distances from the crossing point C is broken.
- If in Close TES, the algorithm reduces the speed of the Agent with the bigger distance from the crossing point C, so the condition, D greater than $D_{Efficiency}$, is maintained.
- When an Agent passes the crossing point C while the above step is satisfied, the Agent that has altered its speed regains its original speed.

We have tested the effectiveness of the above algorithm for both Absolute and Close TES avoidance for Monovular Potential Agents that are guided by AWSPPF algorithm. The simulation scenarios that we have tested these algorithms are the same with the scenarios of the previous chapter, therefore:

The autonomous Agents have the characteristics of an average Unmanned Surface Vehicle (USV). The Vehicles are represented by a circle with radius of 9m and can reach a maximum speed of 5.14m/sec or (10NM/hour). Their local navigational environment covers a surface of 1000x1000m. We use ideal point mass model for the Vehicles to identify the Potential Field algorithm trajectory that is generated in Pure Dynamic Environment (PDE) without the noise produced by the Vehicles' kinematic capabilities or static obstacles.

The initial Monovular Autonomous Vehicles coordinates are:

Vehicle1: (0, 0)

Vehicle2: (0, 800)

Target Destination Vehicle1: (800, 800)

Target Destination vehicle2: (0, 800)

Vehicle1 algorithmic max speed: 5.14m/sec or 10NM

Vehicle2 algorithmic max speed: 5.14m/sec or 10NM

The performance results of the AWSPPF algorithm without the use of the TES Prediction and Avoidance rule based mathematical algorithm, are illustrated in Figure 5-4.

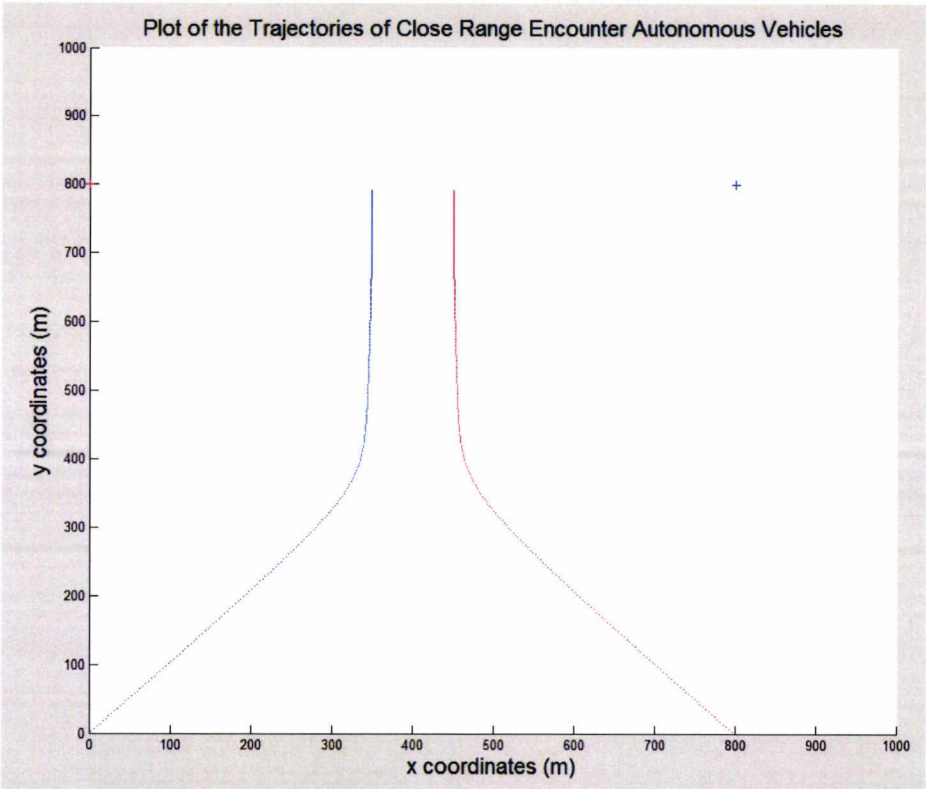


Figure 5-4: Absolute TES of the AWSPPF Algorithm without the TES Detection and Avoidance algorithm.

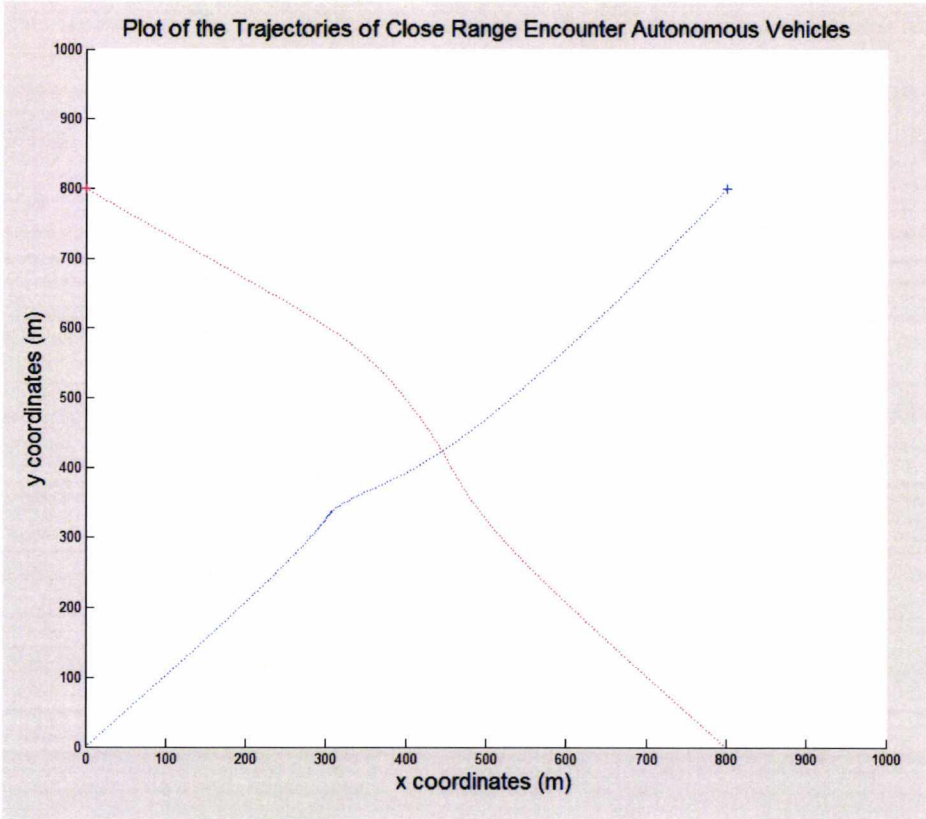


Figure 5-5: Absolute TES of the AWSPPF Algorithm with the TES Detection and Avoidance algorithm.

The algorithm is within local minima, and the Vehicles never reach their target destination. On the other hand, when the Autonomous Vehicles are aided by the new TES Detection and Avoidance algorithm, we have the trajectory results that are illustrated in Figure 5-5. In this case both Autonomous Vehicles reach their target destination. The performance of AWSPPF aided by the TES Detection and Avoidance mathematic algorithm, in terms of trajectory length and trip duration for the Absolute TES, are illustrated in Table 5-1.

Table 5-1: Absolute TES results between the AWSPPF algorithm vs the AWSPPF in combination with the TES Detection and Avoidance algorithms.

Algorithms	Trajectory Length of Vehicle1	Trajectory Length of Vehicle2	Trip Duration Vehicle1	Trip Duration Vehicle2
AWSPPF	Deadlock (local minima)	Deadlock (local minima)	Deadlock (local minima)	Deadlock (local minima)

TES Avoidance algorithm	1.1351x1000m	1.1467x1000m	266	231

We can achieve Close TES behaviour of the AWSPPF algorithm when it guides Monovular Vehicles that have the same coordinate symmetry with the above collision scenario, but their initial speed varies slightly.

First, we examine the case that their initial speed varies only by around 1% (The vehicles speeds are 5.1m/sec for vehicle1 and 5.14 for vehicle2). The trajectory of the two Autonomous Vehicles guided by the AWSPPF algorithm is illustrated in Figure 5-6.

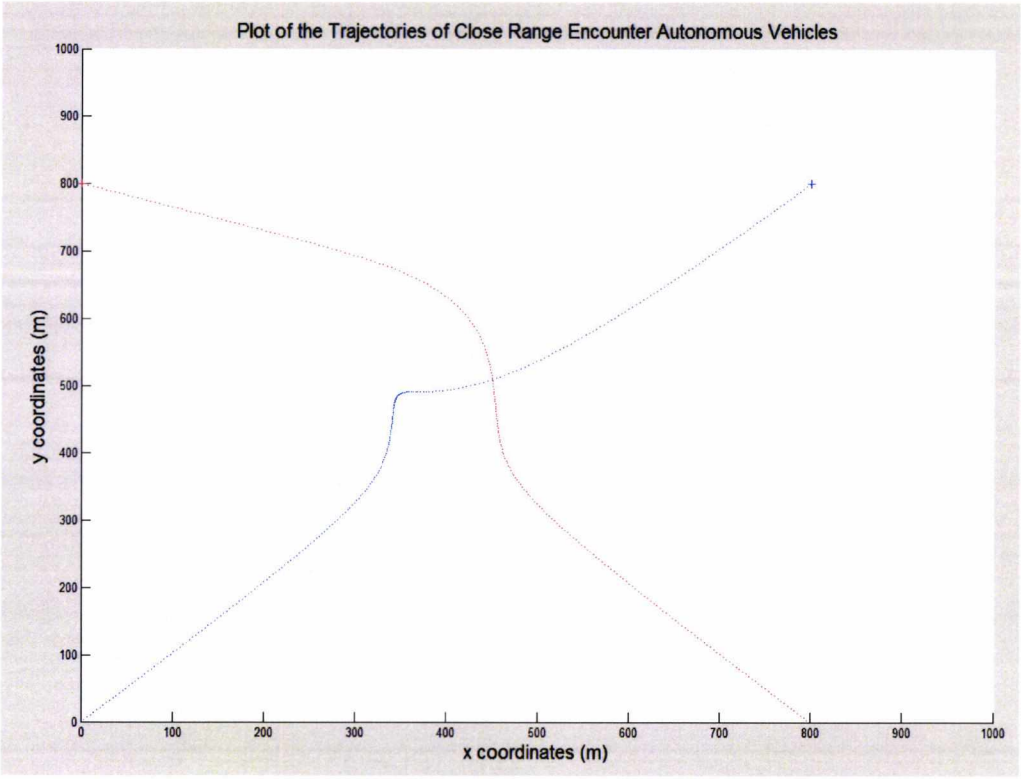


Figure 5-6: Close TES when both the Autonomous Vehicles are guided by the AWSPPF algorithm.

For the same cross collision scenario, when the two autonomous vehicles are guided by the TES avoidance AWSPPF algorithm, the autonomous Vehicles follow the trajectories are illustrated in Figure 5-7. The performance comparison of the two

algorithms in relation to the autonomous vehicles trajectory length and trip duration is illustrated in Table 5-2.

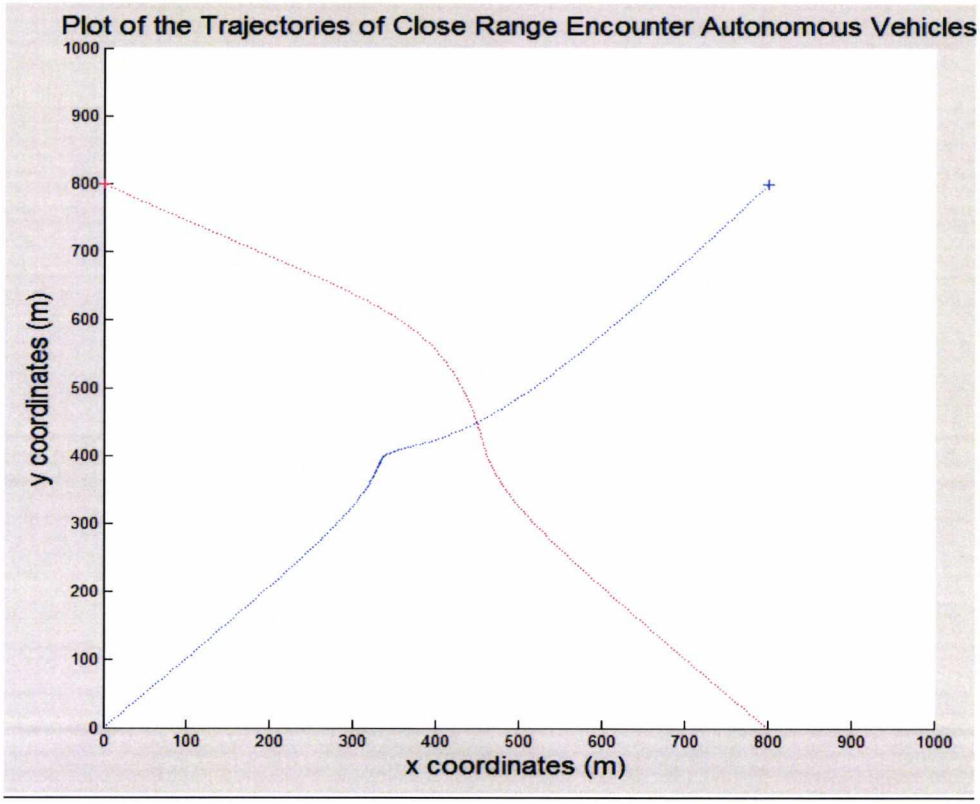


Figure 5-7: Close TES is avoided when both Autonomous Vehicles are guided by the AWSPPF aided by the TES Detection and Avoidance algorithm.

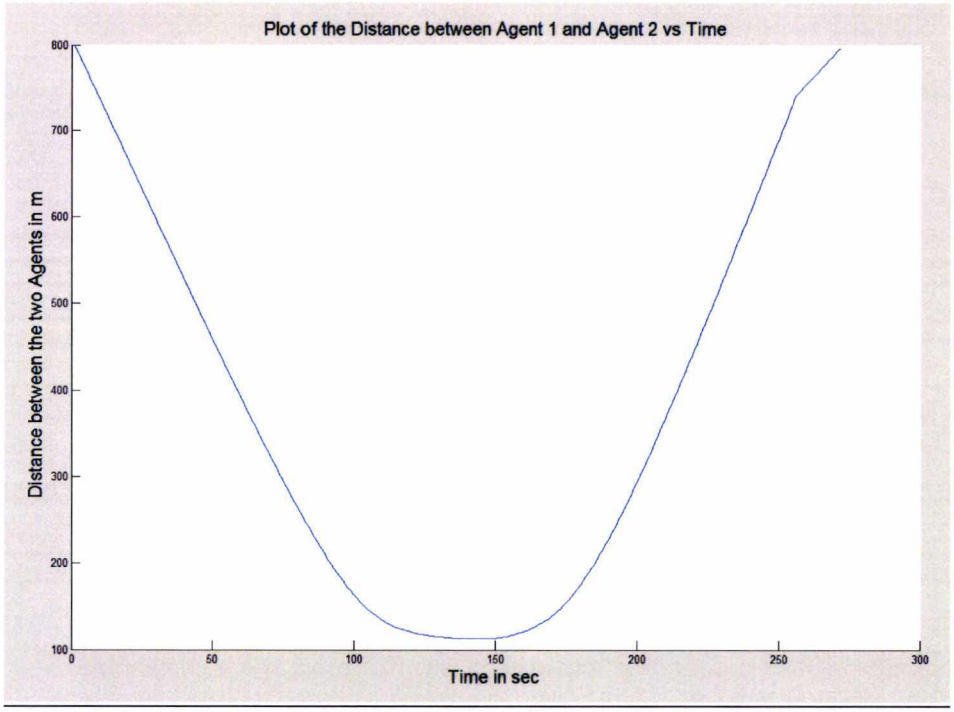


Figure 5-8: Autonomous Vehicle1 Distance from Autonomous Vehicle2 vs Time when the Monovular Vehicles are guided by the AWSPPF algorithm in Close TES.

Table 5-2: Trajectory Length and Trip Duration comparison between TES Detection and avoidance aided AWSPPF and not aided AWSPPF.

Algorithms	Trajectory Length of Vehicle 1	Trajectory Length of Vehicle 2	Trip Duration Vehicle 1	Trip Duration Vehicle 2
AWSPPF	1,1885	1,2236	279	263
With TES Detection and Avoidance	1,1419	1,1575	262	232

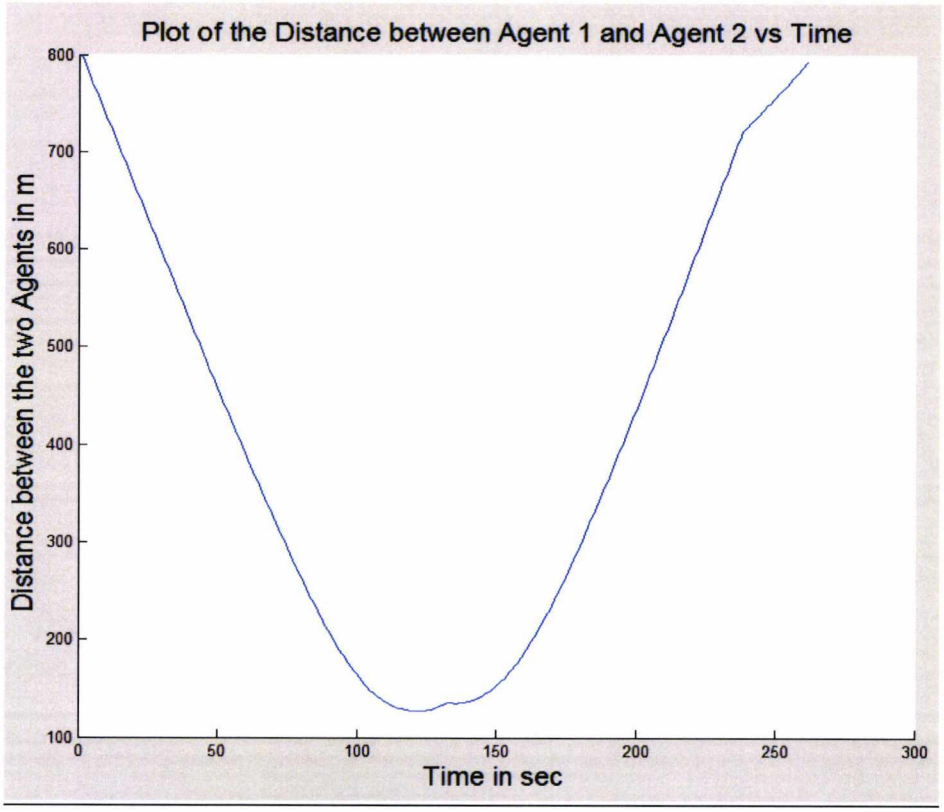


Figure 5-9: Autonomous Vehicle1 Distance from Autonomous Vehicle2 vs Time when the Monovular Vehicles are guided by the AWSPPF aided by the TES detection and avoidance algorithm.

As we can observe from the Table 5-2, the performance of the AWSPPF algorithm has considerably improved in terms of trajectory length, trip duration, and trajectories smoothness of both Vehicles when the TES Detection and Avoidance algorithm is used. In addition, as we can see from the comparison of Figure 5-8 and Figure 5-9 the algorithm safety is improved, since the actual minimum distance between the Vehicles is larger and the duration of similar distances with the actual minimum distance is shorter. These results are summarised in Table 5-3.

Table 5-3: Actual Minimum Distance and Minimum Distance Occurrence comparison between TES Detection and avoidance aided AWSPPF and not aided AWSPPF.

	Actual Minimum Distance (m)	Minimum Distance occurrence (sec)	Algorithm Minimum Distance (m)
AWSPPF	112.5	143	100
TES Avoidance algorithm	127.1	122	100

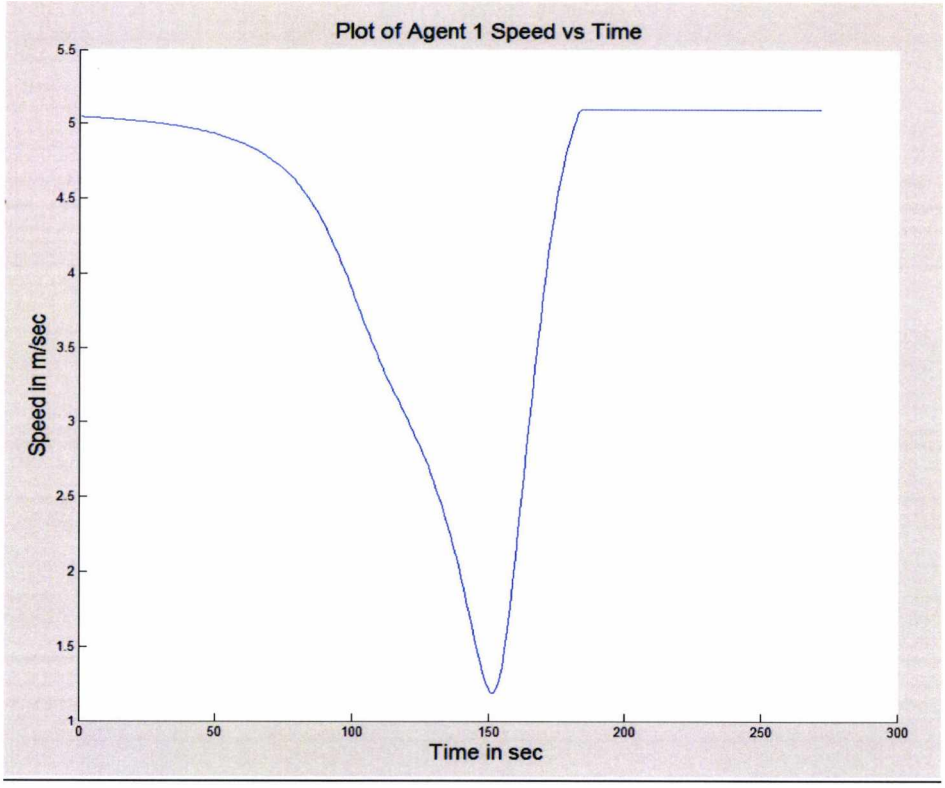


Figure 5-10: Autonomous Vehicle1 Speed Variation due to Autonomous Vehicle2 vs Time when the Autonomous Vehicle is guided by the AWSPPF algorithm.

The autonomous Vehicle1 speed variation graph when is guided by the AWSPFF algorithm is illustrated in Figure 5-10, and when is guided by the same algorithm but in combination with the TES Detection and Avoidance Algorithm is illustrated in Figure 5-11. From these two figures we can observe that the lower speed of Vehicle1 is lower when guided by the TES Detections and Avoidance AWSPPF algorithm than by the AWSPPF algorithm alone. On the other hand, the average speed of Vehicle1 is higher, as indicated in Table 5-2, since its trip has shorter duration. For the second Autonomous Vehicle the old and new algorithm speed comparison is illustrated in Figure 5-12 and Figure 5-13. For the second vehicle the average speed is really improved in the whole duration of the trip.

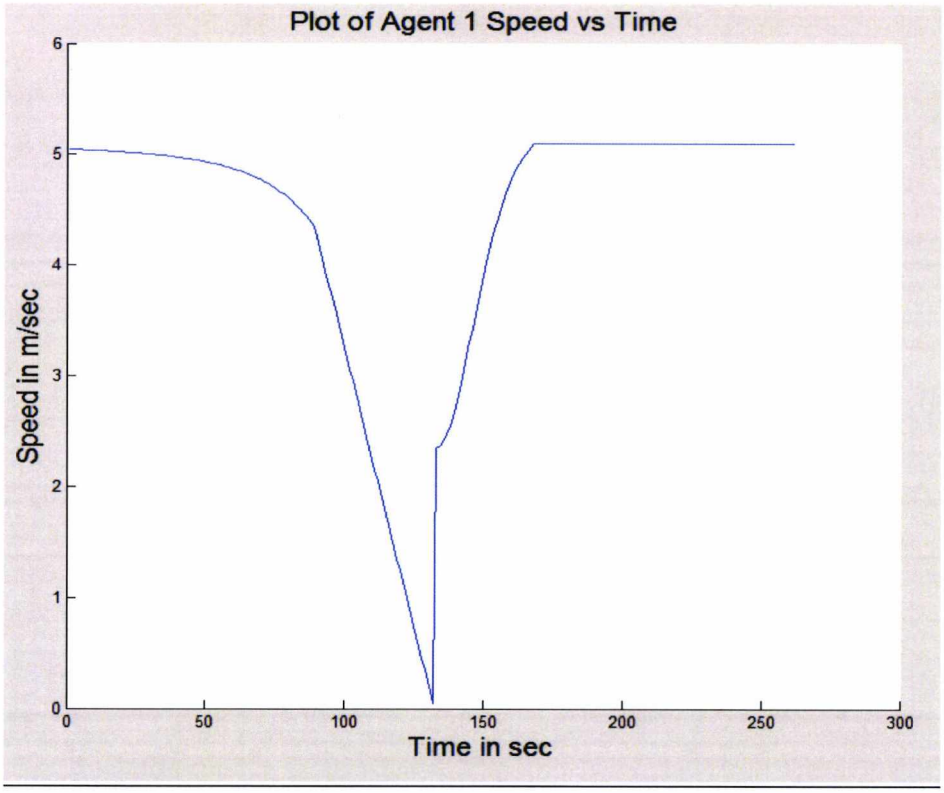


Figure 5-11: Autonomous Vehicle1 Speed Variation due to the Autonomous Vehicle2 vs Time when the Autonomous Vehicle is guided by the AWSPPF aided by the TES Detection and Avoidance algorithm.

The two Vehicles minimum speed comparison between TES Detection and Avoidance AWPSSF algorithm and AWPSSF algorithms are illustrated in Table 5-4.

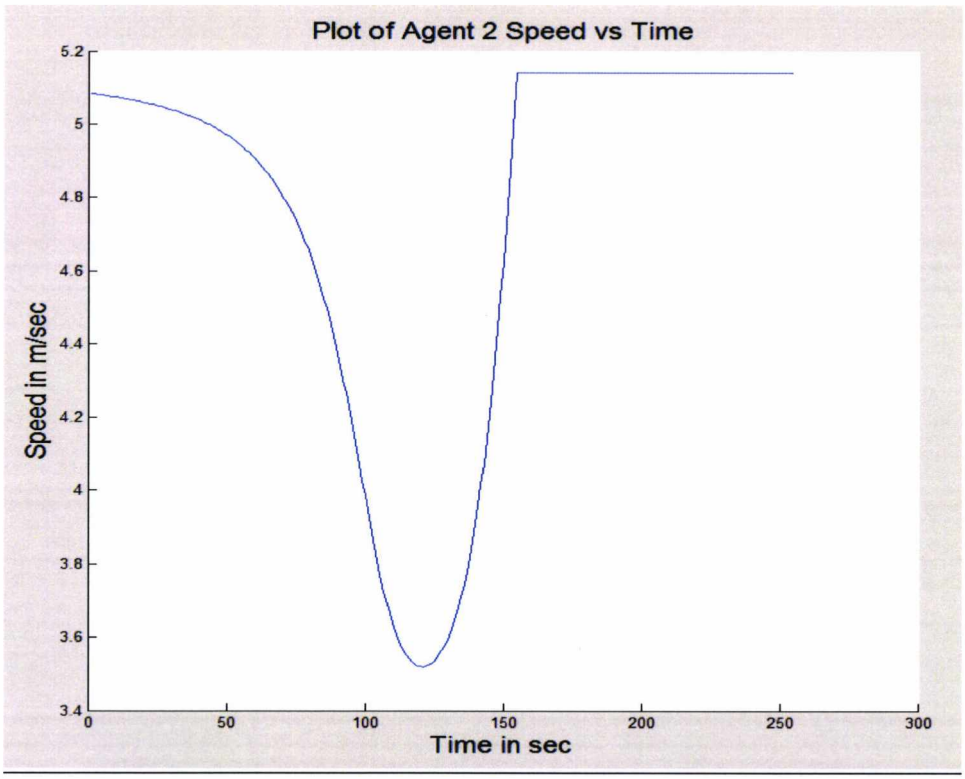


Figure 5-12: Autonomous Vehicle2 Speed Variation due to the Autonomous Vehicle1 vs Time when the Autonomous Vehicle is guided by the AWSPPF

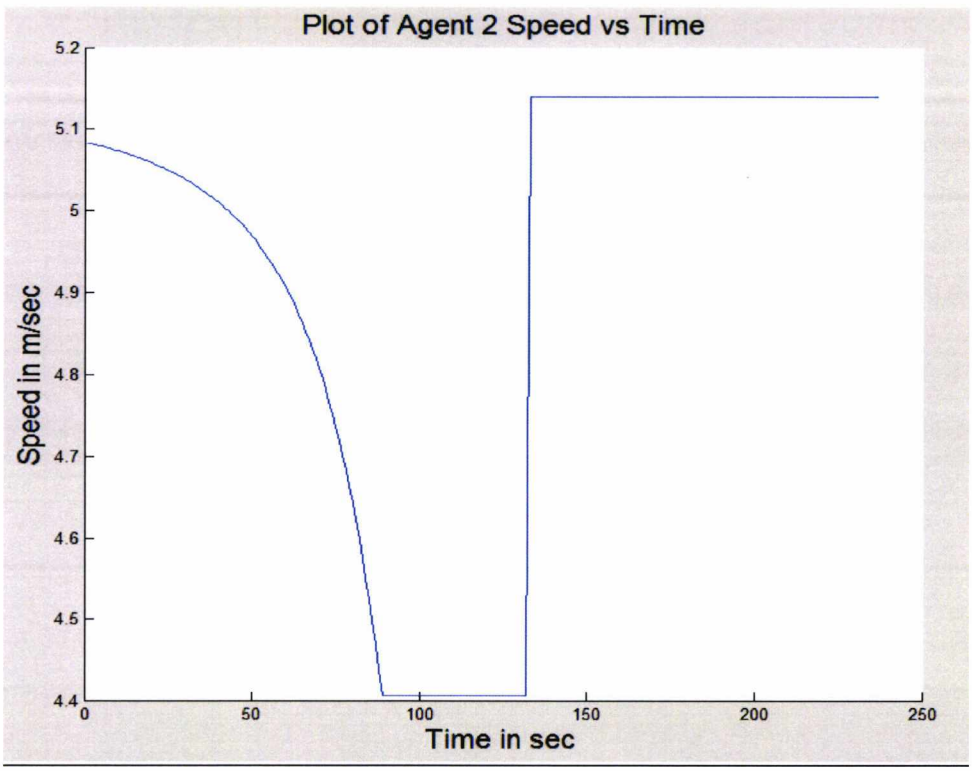


Figure 5-13: Autonomous Vehicle2 Speed Variation due to the Autonomous Vehicle1 vs Time when the Autonomous Vehicle is guided by the AWSPPF aided by the TES Detection and Avoidance algorithm.

Table 5-4: Vehicle1 and Vehicle2 Minimum Speed comparison between TES Detection and avoidance aided AWSPPF and not aided AWSPPF.

Algorithms	Minimum Speed of Vehicle1 (m/sec)	Minimum speed occur-rence Vehicle1 (sec)	Minimum speed of Vehicle2	Minimum speed occur-rence Vehicle2 (sec)
AWSPPF	1.19	151	3.52	121
TES Avoid-ance algorithm	0.05	132	4.407	89 to 131

5.2 Trajectory Equilibrium State (TES) Detection and Avoidance in Biovular Agents

The same mathematical TES Detection and Avoidance algorithm we used in the previous section for Monovular Vehicle/Agent, we can use in Biovular Vehicle/Agent. We have introduced the Biovular Agent in Chapter 4. In case of Biovular Vehicles/Agents, the performance of the AWSPPF algorithm is significantly improved in safety, trajectory smoothness and length. This demonstrates the generic nature of the TES Detection and Avoidance rule based mathematical algorithm.

The scenario to examine the AWSPPF TES Detection and Avoidance collision avoidance algorithm performance for Biovular Vehicles is the following:

The autonomous potential Field algorithm characteristics for the Vehicle 1are:
Algorithmic maximum speed: Vmax = 5.14 m/sec
Algorithmic minimum distance: Dmin = 100m
Local environment: 1000x1000m

The second vehicle is a dynamic obstacle with constant speed of 5.14 m/sec and constant direction.

The initial coordinate of the Vehicle and the Dynamic Obstacle are the following:
Autonomous Vehicle: (0, 0)
Dynamic Obstacle: (0, 800)
Target Destination Autonomous Vehicle: (800, 800)
Target Destination Dynamic Obstacle: (0, 800)

In Figure 5-14 is illustrated the trajectory that is generated when the Autonomous Potential Field Vehicle1 is guided by the AWSPPF algorithm, and in Figure 5-15 is illustrated the trajectory of the Potential field Vehicle1 when is guided by the same algorithm aided by the TES Detection and Avoidance Algorithm. As we can observe from these figures, the trajectory of the new algorithm is much more efficient in

terms of length and smoothness. The trajectory length, and the trip duration different between the two algorithms are indicated in Table 5-5.

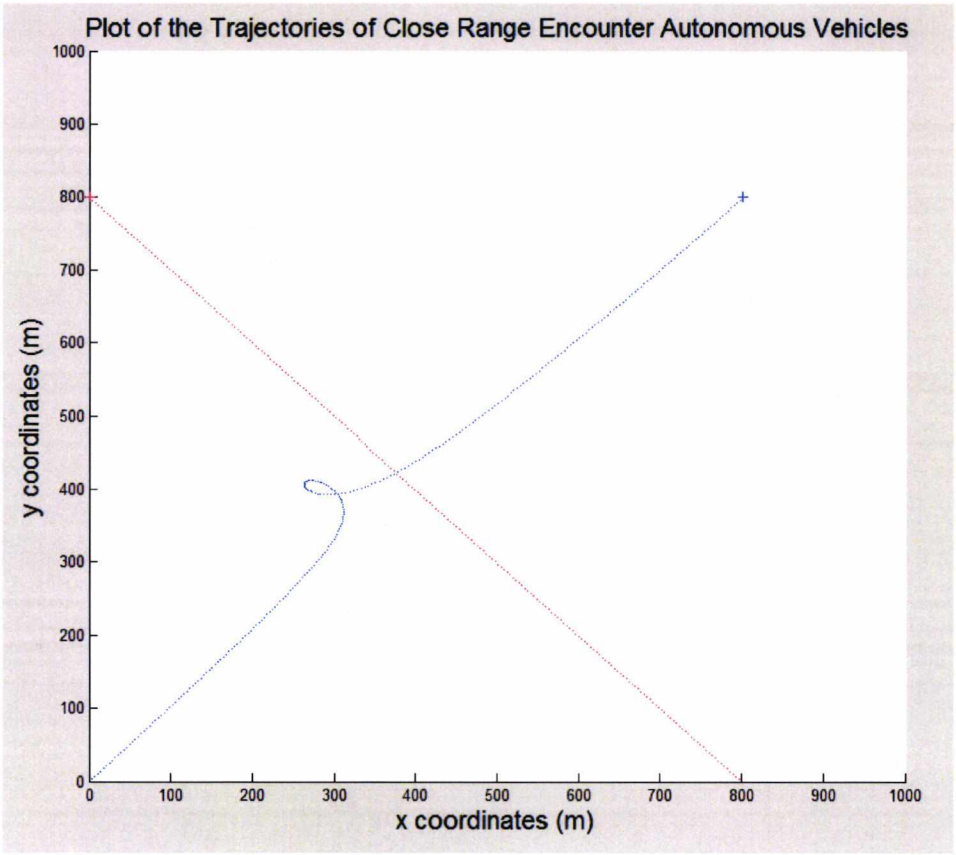


Figure 5-14: Trajectory of AWSPPF autonomous vehicle without TES avoidance algorithm

Table 5-5: TES Detection And Avoidance algorithm Trajectory Length and Trip Duration improvement over the AWSPPF algorithm.

Algorithms	Trajectory Length of Autonomous Vehicle (m)	Trip Duration Autonomous Vehicle (sec)
AWSPPF	1,256.9	273
TES Avoidance algorithm	1,166.1	262

In addition, the TES Detection and Avoidance algorithm critically improves the safety of the algorithm, since not only increases the actual minimum distance from the Dynamic Obstacle (or Biovular Agent/Vehicle), but also ensures that the theoretical Minimum Distance of the algorithm is not going to be exceeded . For example, the algorithm we have under test in this collision scenario has an algorithmic (theoretical) Minimum Distance from any obstacle of 100m. As we can

see from Table 5-6, the Actual Minimum Distance for the specific collision scenario when only AWSPPF is used is around 85 meters.

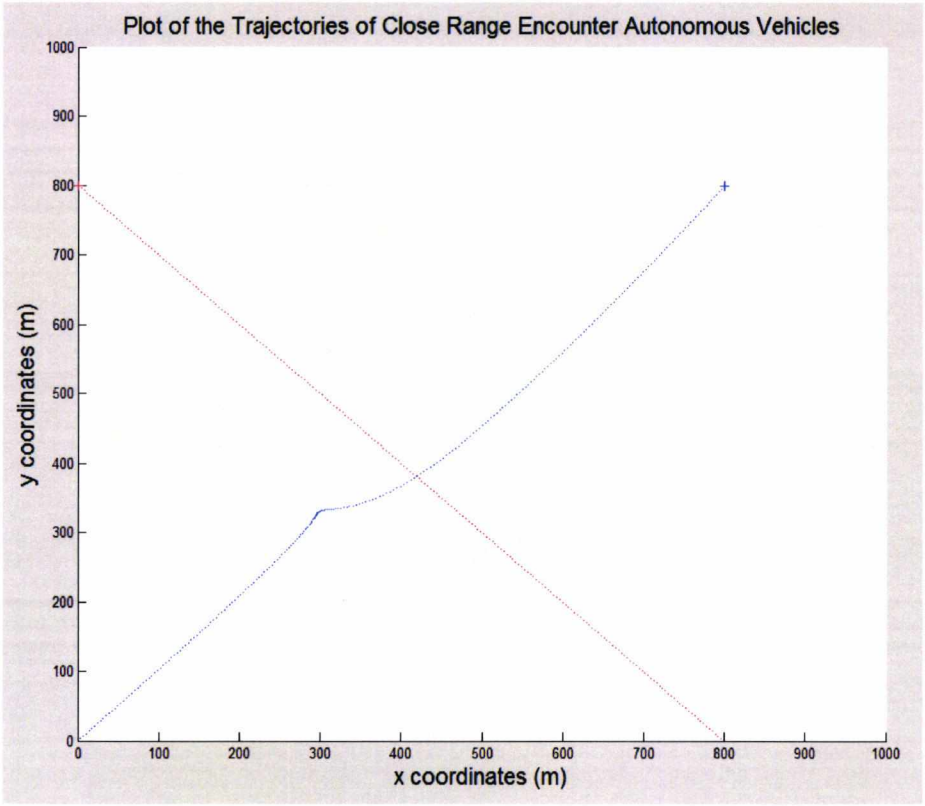


Figure 5-15: Trajectory of AWSPPF algorithm with TES Detection and Avoidance Algorithm

This occurs because the Dynamic Obstacle is much faster of the Autonomous Vehicle when Obstacle and Vehicle are in close proximity. On the other hand, when the TES Detection and Avoidance algorithm aids the AWSPPF algorithm, the actual Minimum Distance of the specific collision scenario is much larger than the algorithmic Minimum Distance at 119.7m.

Finally, in Figure 5-18 is illustrated the Autonomous Vehicle Speed Variation vs Time when the vehicle is guided by the AWSPPF, and in Figure 5-19 is illustrated the Autonomous Vehicle Speed Variation vs Time when the vehicle is guided by the AWSPPF aided by the TES Detection and Avoidance algorithm. The Minimum Speed value and its occurrence in time is illustrate in Table 5-7.

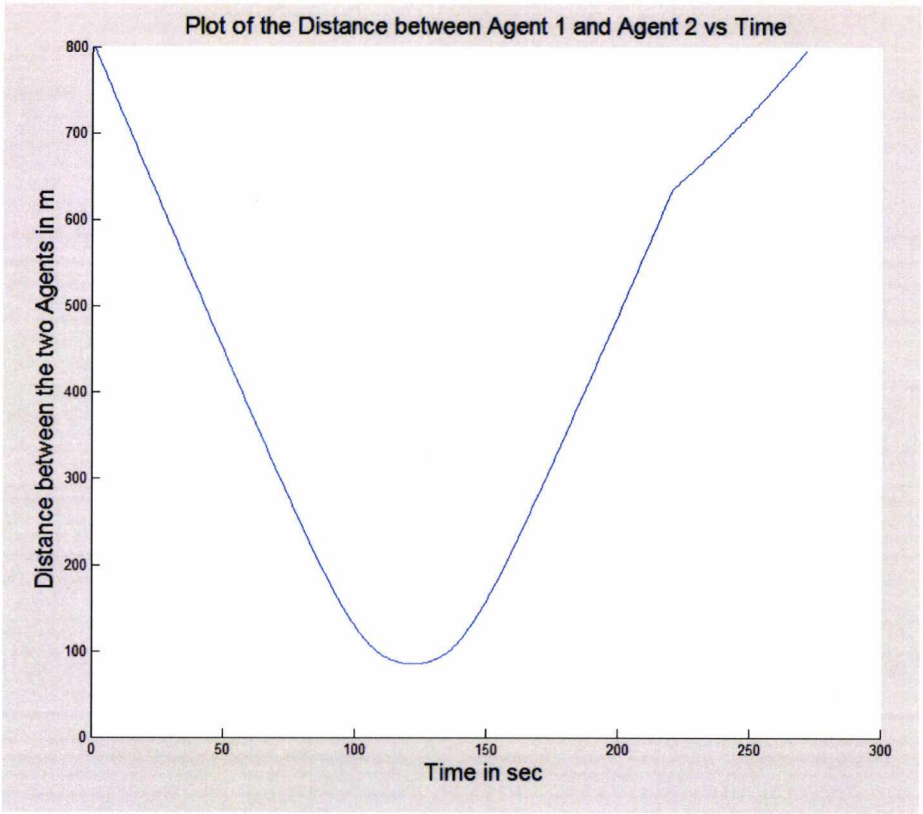


Figure 5-16: Autonomous Vehicle |Distance from Active Obstacle vs Time when the Autonomous vehicle is guided by the AWSPPF algorithm.

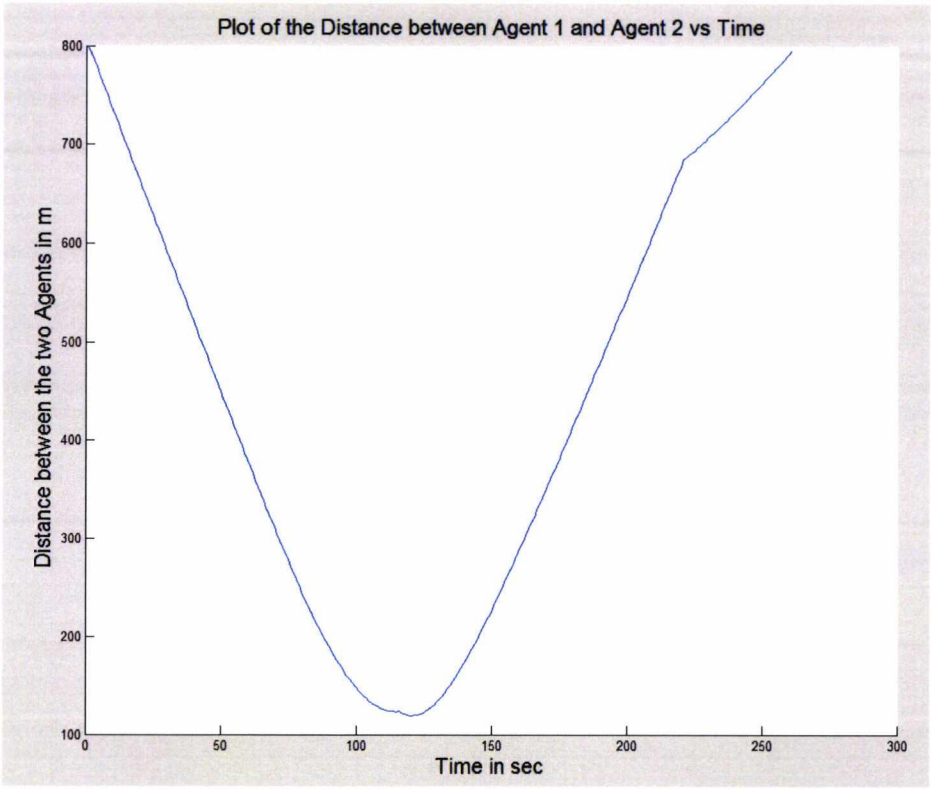


Figure 5-17: Autonomous Vehicle |Distance from Active Obstacle vs Time when the Autonomous Vehicle is guided by the AWSPPF aided by the TES Detection and Avoidance algorithm.

Table 5-6: Comparison of actual Minimum Distance and Theoretical (Algorithmic) Minimum Distance

	Actual Minimum Distance (m)	Algorithmic Minimum Distance (m)
AWSPPF	85.25	100
TES Avoidance algorithm	119.7	100

Nevertheless, the AWPFFS aided by the TES Detection and Avoidance algorithm indicates much smaller Minimum Speed than the AWSPPF algorithm, the average speed of the former is higher. In addition, the Minimum Speed of the TES Detections and Avoidance aided algorithm Minimum Speed occurs much faster than the AWSPPF algorithm. This way the TES Detections and Avoidance algorithm ensures that the Theoretical Minimum Distance of the algorithm will not be exceeded.

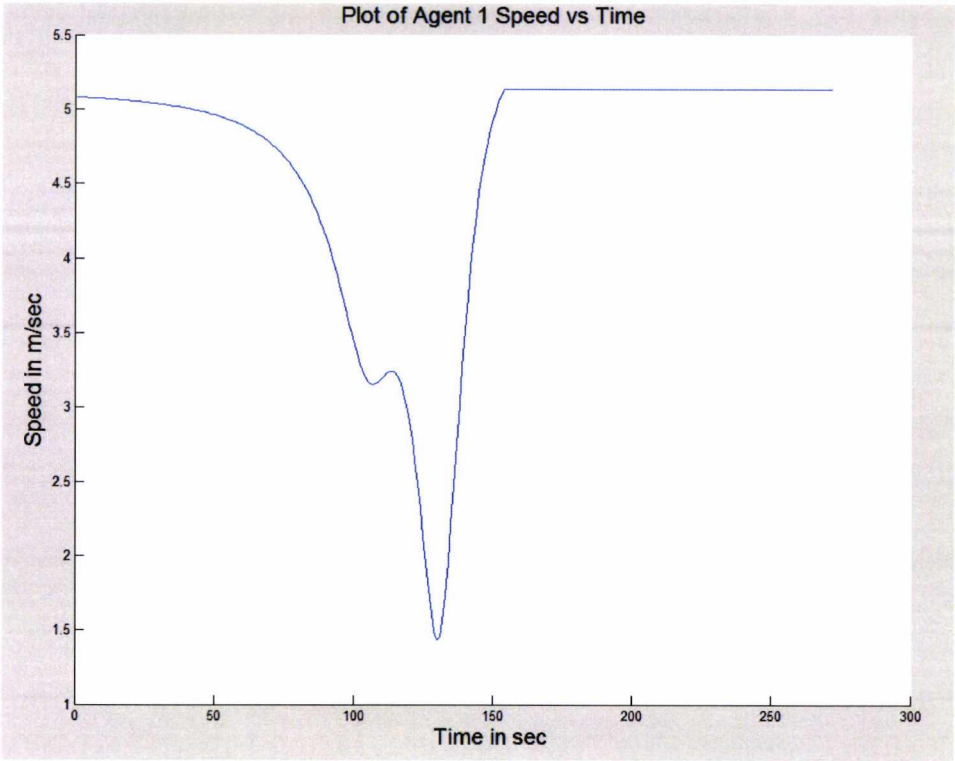


Figure 5-18: Autonomous Vehicle Speed Variation due to the Dynamic Obstacle vs Time when the Autonomous Vehicle is guided by the AWSPPF algorithm.

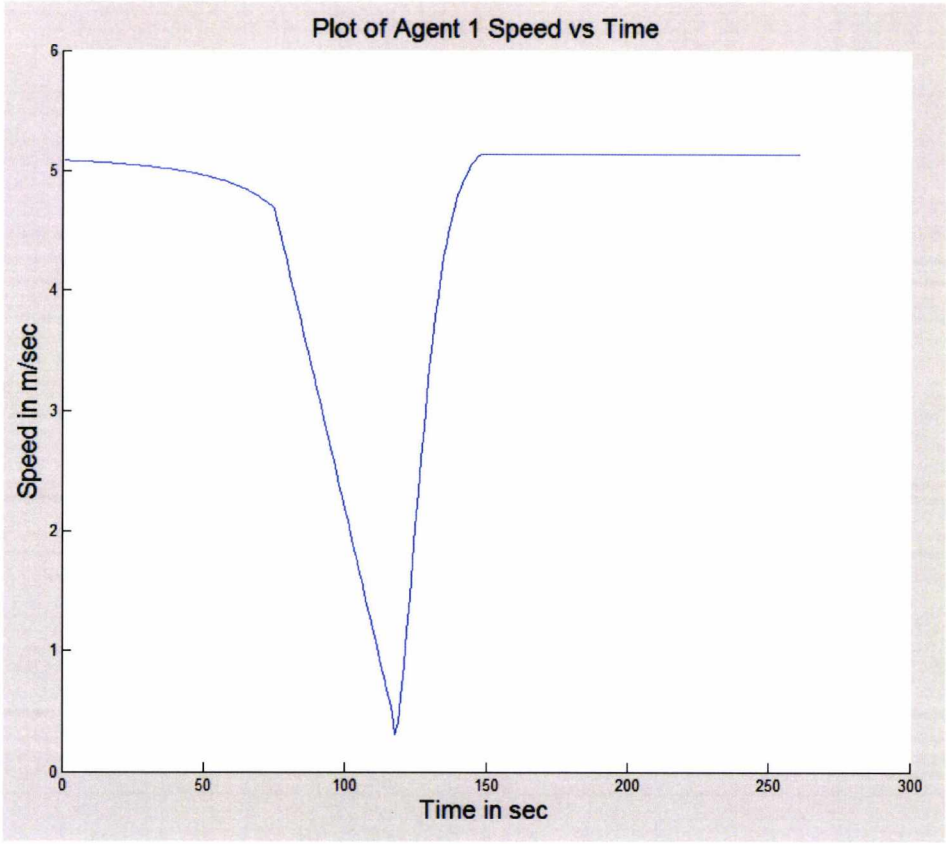


Figure 5-19: Autonomous Vehicle Speed Variation due to the Dynamic Obstacle vs Time when the Autonomous Vehicle is guided by the AWSPPF algorithm aided by the TES Detection and Avoidance Algorithm.

Table 5-7: Autonomous Vehicle Minimum Speed and its occurrence in time while travelling from its initial potion to its Target Destination.

Algorithms	Minimum Speed of the Autonomous Vehicle1 (m/sec)	Minimum speed occurrence (sec)
AWSPPF	1.437	130
TES Avoidance algorithm	0.3106	118

5.3 Conclusion

In this chapter, firstly, we have mathematically designed the novel TES Detection and avoidance algorithm. The mathematical design of this algorithm was possible due to novel concept of Monovular Autonomous Agent Correlation (MAAC), which we introduced and explained in the previous chapter. Secondly, we have demonstrated the performance benefits that the Potential Field algorithms indicate when they are aided by the TES Detection and Avoidance algorithm. It is important to be noted that the TES Detection and Avoidance algorithm is extremely processing efficient. Therefore, it maintains the elegant character of the Potential Field Methods with only a very small processing overhead.

The Potential Field Algorithm we have used is a single point Virtual Force Field (VFF) algorithm with efficient circular Active Window. We have named this algorithm, Active Window Single Point Potential Field (AWSPPF) algorithm.

We have presented the collision avoidance performance improvements of the AWSPPF algorithm when aided by the TES Detection and Avoidance algorithm by a number of collision scenarios. In these collision scenarios we have used both Monovular and Biovular Vehicles in Pure Dynamic Environment (PDE). The simulation results of the above collision scenarios are demonstrated in sections 5.1 and 5.2. These performance improvements that were demonstrated in the above sections are related to safety, trajectory length, trajectory smoothness, trip duration and average speed for both Monovular and Biovular Vehicles.

Chapter 6

Conclusion and Future Work

6. Conclusion and Future Work

In this study we have managed to improve the performance of the classical Potential Field algorithm in Dynamic environment. We have achieved this by following a number of steps.

First, we have designed a simplified Virtual Force Field (VFF) Algorithm. We have simplified the algorithm in the obstacle description and not in collision avoidance capabilities. We have simplified the obstacle and sensor descriptions in a simulation environment in order to examine only its collision avoidance capabilities in a more challenging dynamic local environment. We have also justified the efficiency of a circular Active Window (AW). Therefore, we have named this algorithm Active Window Single Point Potential Field (AWSPPF) algorithm, since it is based on point-mass model without detailed obstacle descriptions. The AWSPPF algorithm accommodates the advantages and disadvantages of the classical Potential Field Methods. In this way, based on this algorithm, we have initially explored the capabilities of processing efficient Potential Field Algorithms in Dynamic Environment.

The AWSPPF algorithm performance degrades in a dynamic environment. This degradation takes place when the initial coordinates of the autonomous vehicle, the dynamic obstacle and the crossing point of their initial/current headings form a geometrical symmetry. With this observation as a guide, we have identified the local minima in a dynamic environment, and more specifically, in Pure Dynamic Environment (PDE).

The identification of local minima of the AWSPPF algorithm in PDE took place with the novel concept of Monovular Autonomous Agent Correlation (MAAC), which is inspired from the Autocorrelation concept that is used in Signal Processing. Based on this concept we have developed a method that identifies inefficient agents' trajectories when in PDE. We have identified the inefficient trajectories, which in some occasions deadlock by following the 4 steps that utilise the MAAC concept:

1. Define the Autonomous Monovular Agents and the size of local Pure Dynamic Environment.

2. Place two Monovular Agents within the same local environment and try to identify repeating inefficient trajectories.
3. Identify the generic causes of the inefficient repeating trajectories of the autonomous Monovular Agents.
4. Define the causes of the inefficient repeating trajectories of the Potential Field Monovular Agents while navigating in the Pure Dynamic Environment.

After defining the AWSPPF Agents within their related local environment, we have identified repeated trajectory inefficiencies and deadlocks. We have identified these Potential Agents/Vehicles trajectory inefficiencies after running a number of simulated cross collision scenarios. With the aid of these scenarios, we have understood the generic causes that make the AWSPPF Monovular Agents to have inefficient trajectories, and which are:

- The agents' directional vectors are strongly correlated to the rate of change of both direction and magnitude.
- The agents' speed is strongly correlated
- The coordinates of the autonomous agents are symmetrical or close symmetrical.

The above generic causes of inefficient trajectories take place when the Potential Agent is in a state that we have identified and called Trajectory Equilibrium State (TES).

In the fourth step, we have defined and proved that when a Monovular Potential Field Agent is within TES, their coordinates, and their Target Destinations form a symmetry. We have named this symmetry Potential Monovular Agents Symmetry (PMAS). PMAS forces Potential Field navigation vectors of the Monovular Potential Agents to have supplementary directions or close supplementary in absolute terms. Furthermore, based on how close the Monovular Potential Agents are in PMAS we can predict the nature of TES (Absolute or Close). In the case of Absolute TES, we can also predict the coordinates of local minima (if Monovular Agents' Target

Destinations are known from both Agents). Finally, we have tested the PMAS in Biovular Agents; in this case we have also observed close TES but not Absolute TES. This means that PMAS is a generic symmetry that negatively affects Potential Field Algorithm of VFF type. The concept of TES is first published in [4] for Monovular Potential Agents, and in [5] for Biovular Potential Agents.

We have managed to improve the performance of the AWSPPF algorithm when it guides Potential Monovular and/or Biovular Agents/Vehicles that satisfy PMAS symmetry. That was materialised by aiding AWSPF algorithm with the novel TES Detection and Avoidance algorithm. The concept and the effectiveness of the above algorithm is published by the author of this thesis in [4] for Monovular Agents, and in [5] for Biovular Agents.

We have demonstrated the collision avoidance performance improvements of the AWSPPF algorithm when aided by the TES Detection and Avoidance algorithm by a number of cross collision scenarios. In these collision scenarios we have used both Monovular and Biovular Vehicles in Pure Dynamic Environment (PDE). The AWSPPF performance improvements in both Monovular and Biovular vehicles indicated the generic nature of the TES Detection and Avoidance algorithm. The simulation results of the cross collision scenarios are demonstrated in sections 5.1 and 5.2. The performance improvements that were demonstrated in the above sections are related to safety, trajectory length, trajectory smoothness, trip duration and average speed for both Monovular and Biovular Vehicles.

It is important to be noted that the TES Detection and Avoidance algorithm is computationally very efficient. Therefore, it maintains the elegant character of the VFF type Potential Field Methods with only a very small processing overhead.

6.1 Thesis Contribution

The main contribution of this study is the performance improvement of classical VFF type Potential Field Algorithms in Pure Dynamic Environment. We have

selected to improve the performance of these algorithms after a comprehensive review of a number of real-time collision avoidance algorithms; part of the review study is published by the author in [6]. The performance improvement of VFF type Potential Field Algorithms that took place is based on a number of novel concepts and algorithms.

Initially, we have justified the use of a Circular Active Window in Potential Field Algorithms. Secondly, we have designed the Active Window Single Point Potential Field (AWSPPF) algorithm. This algorithm uses a simplified point-mass model and the elegant collision avoidance capabilities of Virtual Force Field (VFF) type algorithms.

VFF type Potential Field Algorithms suffer from local minima, but are extensively used in real-time path planning for their elegance and simplicity. The existence of local minima has only been identified in static environment. In this study for first time, we have identified the existence of local minima in dynamic environment—more specifically, Pure Dynamic Environment (PDE) that we have newly defined.

To identify local minima existence in Potential Field Agents/Vehicles in PDE, we have introduced and defined the concept of Monovular Agent and Monovular Autonomous Agent Correlation (MAAC). Based on MAAC we have managed to identify local minima in PDE. In addition, we have mathematically defined the causes of local minima in PDE, which are related to a newly defined symmetry and we have named this symmetry PMAS. Furthermore, we have proved that PMAS is maintained throughout the collision avoidance stages between the two Monovular Potential Agents. As long the PMAS is maintained during the time domain of the Agent trajectory generation, the Agents are within TES. Absolute TES causes the Monovular Potential Agents to be in a future navigational deadlock, and Close TES causes the Agents to have inefficient trajectories. The author has published the initial concept of TES in [4].

Finally, based on the generic definitions of PMAS and TES, we have designed a new efficient rule based mathematical algorithm that predicts and avoids TES in PDE. Extensive simulation results of the performance improvements of the AWSPPF

algorithm aided by the TES Detection and Avoidance algorithm have been presented in chapter 5.

6.2 Future Work

One of the next steps of this study relates to the optimisation of the TES Detection and Avoidance algorithm.

The first optimisation of the TES Detections and Avoidance algorithm is to define the relationship between the Potential Field Autonomous Agents/Vehicles collision angle and the initialisation of theTES Detection and Avoidance Algorithm.

When the collision angle of the autonomous agent is equal to 180 degrees, The TES Detection and Avoidance algorithm is not able to avoid deadlocks. In other words, the TES Detection and Avoidance algorithm cannot avoid local minima in head on collision scenario. Therefore, it is important, in the future, to improve the TES Detection and Avoidance algorithm, so, it will be able to avoid local minima in head on collision scenario.

The TES Detection and Avoidance algorithm is tested in water based autonomous vehicle. Therefore, the algorithm has to be adapted to follow the "rules of the road" (COLREGs) guidelines.

Finally, we would like to note that the concept of MAAC could possibly be beneficial for other on-line collision avoidance navigation algorithms. Therefore, an investigation of the performance of these algorithms based on the MAAC method could identify possible inefficient trajectories in PDE and also define what has caused them.

1. Khatib, O., *Real-Time Obstacle Avoidance For Manipulators And Mobile Robots.* International Journal Of Robotics Research, 1986. **5**(1): p. 90-98.
2. Borenstein, J. and Y. Koren, *Real-time obstacle avoidance for fast mobile robots.* Systems, Man and Cybernetics, IEEE Transactions on, 1989. **19**(5): p. 1179.
3. Borenstein, J. and Y. Koren, *Real-time obstacle avoidance for fast mobile robots in cluttered environments.* Robotics and Automation, 1990. Proceedings., 1990 IEEE International Conference on, 1990: p. 572 - 577.
4. Statheros, T., G. Howells, and K. McDonald-Maier, *Trajectory equilibrium state detection and avoidance algorithm for multi-autonomous potential field mobile robots.* Electronics Letters, 2007. **43**(15): p. 799-801.
5. Statheros, T., G. Howells, and K. McDonald-Maier. *A Novel Potential Field Algorithm and an Intelligent Multi-classifier for the Automated Control and Guidance System (ACOS).* in *Sixth International Conference on Informatics in Control, Automation and Robotics (ICINCO 2009).* 2009 (July). Milan, Italy.
6. Statheros, T., G. Howells, and K. McDonald-Maier, *Autonomous ship collision avoidance navigation concepts, technologies and techniques.* Journal Of Navigation, 2008. **61**: p. 129-142.
7. Koren, Y. and J. Borenstein, *Potential field methods and their inherent limitations for mobile robot navigation.* Robotics and Automation, 1991. Proceedings., 1991 IEEE International Conference on, 1991: p. 1398.
8. Khatib, O., *Real-time obstacle avoidance for manipulators and mobile robots.* Robotics and Automation. Proceedings. 1985 IEEE International Conference on, 1985. **2**: p. 500.
9. Borenstein, J. and Y. Koren. *Real-time obstacle avoidance for fast mobile robots in cluttered environments.* in *Robotics and Automation, 1990. Proceedings., 1990 IEEE International Conference on.* 1990.
10. Wilson, P.A., C.J. Harris, and X. Hong, *A line of sight counteraction navigation algorithm for ship encounter collision avoidance.* Journal Of Navigation, 2003. **56**(1): p. 111-121.
11. Lee, Y.I. and Y.G. Kim, *A collision avoidance system for autonomous ship using fuzzy relational products and COLREGs*, in *Intelligent Data Engineering And Automated Learning Ideal 2004, Proceedings*. 2004, Springer-Verlag Berlin: Berlin. p. 247-252.
12. Salinas, C.F., *Collision regulations discussions.* Journal Of Navigation, 2002. **55**(3): p. 501-505.
13. Kemp, J., *Collision regulations - Discussion.* Journal Of Navigation, 2002. **55**(1): p. 145-146.
14. Belcher, P., *A sociological interpretation of the COLREGS.* Journal Of Navigation, 2002. **55**(2): p. 213-224.
15. Robert, G., et al., *Cognitive demands of collision avoidance in simulated ship control.* Human Factors, 2003. **45**(2): p. 252-265.
16. Lisowski, J., *A simulation study of various approximate models of ship dynamics in collision avoidance problem.* Found. Contr. Eng., 1985. **10**(2): p. 176-183.
17. Browning, A.W., *A Mathematical-Model To Simulate Small Boat Behavior.* Simulation, 1991. **56**(5): p. 329-336.

18. Yavin, Y., et al., *Collision avoidance by a ship with a moving obstacle: Computation of feasible command strategies.* Journal Of Optimization Theory And Applications, 1997. **93**(1): p. 53-66.
19. Abril, J., J. Salom, and O. Calvo, *Fuzzy control of a sailboat.* International Journal Of Approximate Reasoning, 1997. **16**(3-4): p. 359-375.
20. El-Kader, F.A., et al., *An integrated navigation system for Suez Canal (SCINS).* Journal Of Navigation, 2003. **56**(2): p. 241-255.
21. Lisowski, J. and R. Smierzchalski, *Computer simulation of safe path maneuver and trajectory avoiding collision at sea.* In Joint Proceedings, Gdynia Maritime Academy Housechule Bremerhaven, 1994.
22. Lisowski, J. and R. Smierzchalski, *Assigning of safe and optimal trajectory avoiding collision at sea.* in Proc. 3rd IFAC Workshop Contr. Appl. Marine Syst., 1995: p. 346-350.
23. Graczyk, T., et al., *Methods to assign the safe maneuver and trajectory avoiding collision at sea.* Proc. 1st Int. Conf. Marine Technol., 1995: p. 495-502.
24. Smierzchalski, R. and Z. Michalewicz, *Modeling of ship trajectory in collision situations by an evolutionary algorithm.* Ieee Transactions On Evolutionary Computation, 2000. **4**(3): p. 227-241.
25. Smierzchalski, R. and Z. Michalewicz, *Adaptive modeling of a ship trajectory in collision situations at sea.* 1998: p. 342.
26. Hong, X., C.J. Harris, and P.A. Wilson, *Autonomous ship collision free trajectory navigation and control algorithms.* 1999. **2**: p. 923.
27. Zeng, X.M., *Evolution of the safe path for ship navigation.* Applied Artificial Intelligence, 2003. **17**(2): p. 87-104.
28. Pedersen, E., *On the effect of plotting performance by the errors of pointing targets in the ARPA system.* Journal Of Navigation, 1999. **52**(1): p. 119-125.
29. Zeng, X.M., M. Ito, and E. Shimizu, *Collision avoidance of moving obstacles for ship with genetic algorithm.* 2000: p. 513.
30. Xiao-Ming, Z. and M. Ito, *Planning a collision avoidance model for ship using genetic algorithm.* 2001. **4**: p. 2355.
31. Curtis, R.G., *Deternination of mariners' reaction times.* Journal Of Navigation, 1978. **31**: p. 408-417.
32. Cahill, R.A., *Collision and their causes*. 1983: London: Fairplay.
33. Lord, W., *A Night to Remember*. 1955: New York: Holt, Rinehart & Winston.
34. Rasmussen, J., *Skills, Rules, And Knowledge - Signals, Signs, And Symbols, And Other Distinctions In Human-Performance Models.* Ieee Transactions On Systems Man And Cybernetics, 1983. **13**(3): p. 257-266.
35. Stefik, M., *Introduction to Knowleds Systems*. 1995, San Francisco: Morgan Kaufmann Publishers, Inc.
36. James, M.K., *The Timing Of Collision-Avoidance Maneuvers - Descriptive Mathematical-Models.* Journal Of Navigation, 1994. **47**(2): p. 259-272.
37. Wagenaar, W.A. and J. Groeneweg, *Accidents At Sea - Multiple Causes And Impossible Consequences.* International Journal Of Man-Machine Studies, 1987. **27**(5-6): p. 587-598.
38. Lee, J.D., & Sanquist, T.F., *Maritime Automation.* Automation and human performance: Theory and applications, 1996: p. 365-384.
39. Cockroft, A.N., *Collision at Sea.* Safety at Sea, 1984: p. 17-19.
40. Perrow, C., *Normal accidents*. 1984: Basic Books.

41. Imazu, H., A.M. Sugisaki, and Q. Min, *The Possible Collision Risk in Radar Navigation.* Journal of Navigation in Japan, 1979. **65**: p. 75-81.
42. Dmitriev, S.P., et al., *System of intelligent support of a ship navigator for collision avoidance.* Journal Of Computer And Systems Sciences International, 2003. **42**(2): p. 256-263.
43. Gung, W.S., *The Research of Safely Coursing and Collision Avoidance between Ships in Traffic Separation Schemes under the Situation of Crossing and Overtaking*. 1990, National Taiwan Marine University: Taiwan.
44. Hammer, A. and H. K., *Knowledge Acquisition for Collision Avoidance Maneuver by Ship handling Simulator.* International Conference on Marine Simulation and Ship Maneuverability, 1990: p. 245-252.
45. Jones, K.D., *Desision making when using collision avoidance system.* J. Navigation, 1978. **31**: p. 173-180.
46. Colley, B.A., R.G. Curtis, and C.T. Stockel, *A Marine Traffic Flow And Collision Avoidance Computer-Simulation.* Journal Of Navigation, 1984. **37**(2): p. 232-250.
47. Lamb, W.G.P., *The Calculation Of Marine Collision Risks.* Journal Of Navigation, 1985. **38**(3): p. 365-374.
48. Zhao, J., *Maritime Collision and Liability*, in *Dept of Ship Science*. 1996, University of Southampton.
49. Braspenning, P.J., F. Thuijsman, and A.J.M.M. Weijters, *Artificial Neural Networks: An introduction to ANN Theory and Practice*. 1995, Berlin: Springer.
50. Anderson, J.A., *An Introduction to Neural Networks*. 1995, Cambridge: MIT Press.
51. Patterson, D.W., *Artificial Neural Networks: Theory and Applications*. 1996, Singapore: Prentice Hall.
52. Zadeh, L.A., *Fuzzy Sets.* Information and Control, 1965. **8**: p. 338-353.
53. Lee, S.M., K.Y. Kwon, and J. Joh, *A fuzzy logic for autonomous navigation of marine vehicles satisfying COLREG guidelines.* International Journal Of Control Automation And Systems, 2004. **2**(2): p. 171-181.
54. Back, T., *Evolutionary Algorithms in Theory and Practice: Evolution Strategies, Evolutionary Programming, Genetic Algorithms*. 1996: Oxford University Press.
55. Lin, H.S., X. J, and Z. Michalewicz, *Evolutionary algorithm for path planning in mobile robot environment.* IEEE World Congress on Computational Intelligence., Proceedings of the First IEEE Conference, 1994(1): p. 211 - 216.
56. Jing, X., et al., *Adaptive evolutionary planner/navigator for mobile robots.* Evolutionary Computation, IEEE Transactions, 1997. **1**(1): p. 18 - 28.
57. Efstathiou, J., *Expert systems, fuzzy logic and rule-based control explained at last.* transactions Of The Institute Of Measurement And Control? (Trans Inst MC), 1988. **4**: p. 198-206.
58. Chohra, A., A. Farah, and M. Belloucif, *Neuro-fuzzy expert system E_S_CO_V for the obstacle avoidance behavior of intelligent autonomous vehicles.* Advanced Robotics, 1999. **12**(6): p. 629-649.
59. Hwang, C.N., J.M. Yang, and C.Y. Chiang, *The Design of Fuzzy Collision Avoidance Expert System Implemented by H-Infinity autopilot.* Journal of Marine Science and Technology, 2001. **9**: p. 25-37.

60. Skjong, R. and K.M. Mjelde, *Optimal evasive manoeuvre for ship in an environment of fixed installations and other ships.* Identification and Control, 1982. **3**: p. 211-222.
61. Yavin, Y., G. Zilman, and T. Miloh, *A Feasibility Study Of Ship Maneuverability In The Vicinity Of An Obstacle - A Stochastic-Control Approach.* Computers & Mathematics With Applications, 1994. **28**(8): p. 63-76.
62. Yavin, Y., T. Miloh, and G. Zilman, *Parametric Study Of Ship Maneuverability In Laterally Restricted Waters - Stochastic-Control Approach.* Journal Of Optimization Theory And Applications, 1995. **85**(1): p. 59-74.
63. Lewis, F.L., *Optimal Estimation: With an Introduction to Stochastic Control Theory*. 1986: John Wiley & Sons.
64. Curtis, R.G., *A Ship Collision Model For Overtaking.* Journal Of The Operational Research Society, 1986. **37**(4): p. 397-406.
65. Burns, R.S., G. Blackwell, and S. Calvert, *An automatic guidance, navigation and collision avoidance system for ships at sea.* IEE Colloquium on Control in Marine Industry, 1988: p. 3/1-3/3.
66. Khanna, T., *Foundations of neural networks*. 1990, New York: Addison-Wesley.
67. Xianyi, Y., *A Neural Network Approach to Real-Time Collision-Free Navigation of 3-D.O.F. Robots in 2D.* ICRA, 1999: p. 23-28.
68. Harris, C.J., X. Hong, and P.A. Wilson, *An intelligent guidance and control system for ship obstacle avoidance.* Proceedings Of The Institution Of Mechanical Engineers Part I- Journal Of Systems And Control Engineering, 1999. **213**(I4): p. 311-320.
69. Vonk, E., l.C. Jain, and R.P. Johnson, *Automatic Generation of Neural Network Architecture Using Evolutionary Computation.* Vol. 14. 1997: World Scientific.
70. Glick, T.F. and D. Kohn, *Darwin on Evolution*, ed. T.F. Glick and D. Kohn. 1996, Indianapolis, Indiana, USA: Hackett Puplishing Company, Inc.
71. Stewart, B.S., C.F. Liaw, and C.C. White, *A Bibliography Of Heuristic-Search Research Through 1992.* Ieee Transactions On Systems Man And Cybernetics, 1994. **24**(2): p. 268-293.
72. Ito, M., F. Zhang, and N. Yoshida, *Collision avoidance control of ship with genetic algorithm.* Proceedings of the IEEE International Conference on Control Applications, 1999. **2**: p. 1791 - 1796.
73. Andrews, J.R.a.H., pp., *Impedance Control as a Framework for Implementing Obstacle Avoidance in a Manipulator. Control of Manufacturing Processes and Robotic Systems*, 1983: p. 243-251.
74. Khosla, P. and R. Volpe. *Superquadric artificial potentials for obstacle avoidance and approach.* in *Robotics and Automation, 1988. Proceedings., 1988 IEEE International Conference on.* 1988.
75. Volpe, R. and P. Khosla, *Manipulator control with superquadric artificial potential functions: theory and experiments.* Systems, Man and Cybernetics, IEEE Transactions on, 1990. **20**(6): p. 1423.
76. Khatib, O. *Real-time obstacle avoidance for manipulators and mobile robots.* 1985.

77. Lozanoperez, T., *Automatic Planning Of Manipulator Transfer Movements.* Ieee Transactions On Systems Man And Cybernetics, 1981. **11**(10): p. 681-698.
78. Brooks, R., *Solving the find-path problem by good representation of free space.* IEEE transactions on systems, man, and cybernetics, 1983. **13**: p. 190-197.
79. Lozano-Perez, T., *Spatial Planning: A Configuration Space Approach.* Computers, IEEE Transactions on, 1983. **C-32**(2): p. 108.
80. Arkin, R.C., *Behavior based robotics*. 2nd print ed. 1999: Cambridge; London: The MIT Press.
81. Ken Clements and N. Ellerton, *Historical Perspectives on Mathematical Elegance - To What Extent is Mathematical Beauty in the Eye of the Beholder?* Illionois State University, 2006.
82. Ding, F.-g., et al. *AUV local path planning based on virtual potential field.* in *Mechatronics and Automation, 2005 IEEE International Conference*. 2005.
83. Warren, C.W. *Global path planning using artificial potential fields*. in *Robotics and Automation, 1989. Proceedings., 1989 IEEE International Conference on*. 1989.
84. Connolly, C.I., J.B. Burns, and R. Weiss. *Path planning using Laplace's equation*. in *Robotics and Automation, 1990. Proceedings., 1990 IEEE International Conference on*. 1990.
85. Feder, H.J.S. and J.J.E. Slotine. *Real-time path planning using harmonic potentials in dynamic environments*. in *Robotics and Automation, 1997. Proceedings., 1997 IEEE International Conference on*. 1997.
86. Chaojian, S., Z. Mingming, and P. Jing. *Harmonic Potential Field Method for Autonomous Ship Navigation*. in *Telecommunications, 2007. ITST '07. 7th International Conference on ITS*. 2007.
87. Valavanis, K.P., et al., *Mobile robot navigation in 2-D dynamic environments using an electrostatic potential field.* Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on, 2000. **30**(2): p. 187.
88. Borenstein, J. and Y. Koren, *The vector field histogram-fast obstacle avoidance for mobile robots.* Robotics and Automation, IEEE Transactions on, 1991. **7**(3): p. 278.
89. Yunfeng, W. and G.S. Chirikjian. *A new potential field method for robot path planning*. in *Robotics and Automation, 2000. Proceedings. ICRA '00. IEEE International Conference on*. 2000.
90. Dongyue, C., Z. Liming, and C. Xiong. *Mobile robot Path planning based on Behavior Information Potential Field in unknown Environments*. in *Robotics and Biomimetics, 2004. ROBIO 2004. IEEE International Conference on*. 2004.
91. Guanghui Li, et al., *An efficient improved artificial potential field based regression search method for robot path planning.* Mechatronics and Automation (ICMA), 2012 International Conference, 2012.
92. Charifa, S. and M. Bikdash. *Comparison of geometrical, kinematic, and dynamic performance of several potential field methods*. in *Southeastcon, 2009. SOUTHEASTCON '09. IEEE*. 2009.
93. Krogh, B. and C. Thorpe. *Integrated path planning and dynamic steering control for autonomous vehicles*. in *Robotics and Automation. Proceedings. 1986 IEEE International Conference on*. 1986.

94. Yun Seok, N., L. Bum Hee, and K. Nak Yong. *A view-time based potential field method for moving obstacle avoidance*. in *SICE '95. Proceedings of the 34th SICE Annual Conference. International Session Papers*. 1995.
95. Lu, Y. and Y. Yixin. *An improved potential field method for mobile robot path planning in dynamic environments*. in *Intelligent Control and Automation, 2008. WCICA 2008. 7th World Congress on*. 2008.
96. Xi Bin Wang, C.S., Guo Rong Zhao, Jing Li Huang, *Obstacles Avoidance for UAV SLAM Based on Improved Artificial otential Field.* Applied Machanics and Materials, 2013. **241 - 244**.
97. Mohammed Abdel Kareem Jaradat, Mohammed H. Garibeh, and E.A. Feilat, *Autonomous mobile robot dynamic motion planning using hybrid fuzzy potential field.* Soft Computing, 2012. **16**(1).
98. Masoud, A.A. and M.M. Bayoumi. *Robot navigation using the vector potential approach*. in *Robotics and Automation, 1993. Proceedings., 1993 IEEE International Conference on*. 1993.
99. Canny, J.F. and M.C. Lin. *An opportunistic global path planner*. in *Robotics and Automation, 1990. Proceedings., 1990 IEEE International Conference on*. 1990.
100. Ge, S.S. and Y.J. Cui, *New potential functions for mobile robot path planning.* Robotics and Automation, IEEE Transactions on, 2000. **16**(5): p. 615.
101. Rimon, E. and D.E. Koditschek. *Exact robot navigation using cost functions: the case of distinct spherical boundaries in <e1>E</e1>ⁿ*. in *Robotics and Automation, 1988. Proceedings., 1988 IEEE International Conference on*. 1988.
102. Rimon, E. and D.E. Koditschek, *Exact robot navigation using artificial potential functions.* Robotics and Automation, IEEE Transactions on, 1992. **8**(5): p. 501.
103. Connolly, C.I. *Applications of harmonic functions to robotics*. in *Intelligent Control, 1992., Proceedings of the 1992 IEEE International Symposium on*. 1992.
104. Connolly, C.I. *Harmonic functions and collision probabilities*. in *Robotics and Automation, 1994. Proceedings., 1994 IEEE International Conference on*. 1994.
105. Pimenta, L.C.A., et al., *Robot navigation based on electrostatic field computation.* Magnetics, IEEE Transactions on, 2006. **42**(4): p. 1459.
106. Statheros, T., G. Howells, and K. McDonald-Maier, *A Novel Potential Field Algorithm and an Intelligent Multi-Classifier for Autonoted Control and Guidance System (ACOS).* Sixth International Conference on Informatics in Control, Automation and Robotics (ICINCO 2009), 2009.
107. Ren, J., K.A. McIsaac, and R.V. Patel, *Modified Newton's method applied to potential field-based navigation for mobile robots.* Ieee Transactions On Robotics, 2006. **22**(2): p. 384-391.
108. Le-Jie, Z., H. Zeng-Guang, and T. Min. *Kalman filter and vision localization based potential field method for autonomous mobile robots*. in *Mechatronics and Automation, 2005 IEEE International Conference*. 2005.
109. Im, K.-Y. and S.-Y. Oh, *An extended virtual force field based behavioral fusion with neural networks and evolutionary programming for mobile robot navigation.* Proc. of the 2000 Congress on Evolutionary Computations, 2000. **2**: p. 1238-1244.

110. Kwang-Young, I. and O. Se-Young, *An extended virtual force field based behavioral fusion with neural networks and evolutionary programming for mobile robot navigation.* Evolutionary Computation, 2000. Proceedings of the 2000 Congress on, 2000. **2**: p. 1238-1244.
111. Bandler, W. and L.J. Kohout, *Semantics Of Implication Operators And Fuzzy Relational Products.* International Journal Of Man-Machine Studies, 1980. **12**(1): p. 89-116.
112. C. Petres, M.-A. Romero-Ramirez, and F. Plumet, *A potential field approach for reactive navigation of autonomous sailboats.* Robotics and Autonomous Systems, 2012. **60**(12).