INSTITUTO SUPERIOR TÉCNICO
Universidade Técnica de Lisboa

Motion Planning for a Rhombic-Like Vehicle Operating in ITER Scenarios

Daniel Alexandre Leal Espinha Fonte

Dissertação para obtenção do Grau de Mestre em
Engenharia Mecânica

Júri

Presidente:	Prof. Doutor João Rogério Caldas Pinto
Orientador:	Doutor Alberto Manuel Martinho Vale
Co-Orientador:	Prof. Doutor Miguel Ayala Botto
Vogal:	Prof. Doutor Jorge Manuel Mateus Martins

Lisboa, Junho de 2011

À Nô

Aos meus pais

Abstract

The dissertation addresses the motion planning problem stated for an autonomous rhombic vehicle that will operate in the buildings of the International Thermonuclear Experimental Reactor (ITER). The problem is addressed following two distinct headings. A complete refinement planning strategy is first presented that decouples the big motion planning problem into modules that are easier to solve. This strategy heavily relies on path deformation as path optimization techniques for which two novel methods are presented. For the complete definition of the rhombic vehicle motion, a trajectory design algorithm is also proposed, which takes into account the obstacle clearance and vehicle dynamic constraints. The second strategy opts for a trajectory planning philosophy with focus on a particular randomized planning technique, the Rapidly-exploring Random Tree (RRT). The main dissertation contribution comprises the combination of a robust RRT-planner with embedded vehicle kinematics with the efficiency of a stochastic optimization method, the Simulated Annealing (SA).

As a requirement for the latter approach the dissertation conducts a detailed analysis on the rhombic vehicle motion including the development of different vehicle behavior models.

Gathered simulated results show the advantage of the refinement strategy on handling feasible and reliable trajectories in cluttered scenarios such as those found in ITER. The trajectory planning strategy reveals some limitations with future perspectives directed to a refinement-like strategy.

The outcome and developments of the dissertation provided a valuable contribute on the area of optimization of trajectories for the rhombic vehicle, supporting the accomplishment and won, by Instituto Superior Técnico (IST), of two Fusion for Energy (F4E) grants in the area of Remote Handling (RH), in which the author of the dissertation participates as full active fellowship student.

Keywords: ITER, Remote Handling, Rhombic, Mathematical Modeling, Motion Planning, Mobile Robotics

Resumo

A tese propõe abordar o problema de planeamento de movimento para um veículo autónomo com configuração rômbica que irá operar nos edifícios do ITER (International Thermonuclear Experimental Reactor).

Para resolver esta problemática, foram seguidas duas estratégias distintas de planeamento. A primeira, define-se como uma abordagem de refinamento que divide o problema complexo de planeamento de movimento em subproblemas de solução mais acessível. Esta abordagem assenta fortemente na aplicação de técnicas de optimização baseadas na deformação de caminhos, área em que esta tese apresenta duas novas contribuições científicas. Para a definição completa do movimento do veículo ao longo dos caminhos, é proposto um método de avaliação de trajectórias que para além de considerar a proximidade do veículo aos obstáculos, inclui também restrições ao nível da dinâmica. A segunda estratégia propõe explorar a planeamento directo de trajectórias com especial enfoque no algoritmo aleatório, Rapidly-exploring Random Tree (RRT). A principal contribuição desta tese para esta filosofia de planeamento consiste no acoplamento de um planeador robusto baseado na RRT, que integra explicitamente a cinemática rômbica, com a eficácia de um algoritmo de optimização, o Simulated Annealing (SA).

Como condição necessária para o desenvolvimento desta última abordagem e colmatar a falta de documentação acerca da configuração rômbica, a tese apresenta diferentes modelos comportamentais para a simulação e análise cuidada do movimento do veículo.

Resultados obtidos em simulação, demonstram a vantagem da utilização de uma estratégia faseada de planeamento para a obtenção de soluções exequíveis para o veículo nos cenários confinados do International Thermonuclear Experimental Reactor (ITER). O planeamento directo de trajéctórias revela grandes limitações no âmbito deste projecto, vinculando-se o seu desenvolvimento para estratégias mais direccionadas para o refinamento.

Este estudo contribuíu de forma valiosa para a realização e obtenção, por parte do Instituto Superior Técnico (IST), de duas grants concedidas pela Fusion for Energy (F4E) na área de Remote Handling (RH) na qual o autor da tese participa como investigador bolseiro.

Palavras-Chave: ITER, Manipulação Remota, Configuração Rômbica, Modelação Matemática, Planeamento do Movimento, Robótica Móvel

Acknowledgments

My first word of appreciation goes to my supervisor and responsible for this adventure, Doctor Alberto Vale, for his insightful suggestions and encouragement during this work, for all the thesis proof reading, but most of all, for always listening attentively and caring. I also thank to Professor Miguel Ayala Botto, for accepting the co-supervision of this dissertation, for his concern and helpful comments.

My sincere gratitude to the Director of ISR/IST, Professor Victor Barroso, and President of IPFN/IST, Professor Carlos Varandas, for all the material and logistic support. This work has also been partially supported by the grant project F4E-2008-GRT-016 (MS-RH), which is funded by the European Joint Undertaking for ITER and the development of F4E. I gratefully acknowledge the support.

A special thanks to Professor Jean-Paul Laumond, who, with his geniality, first introduced me to the motion planning problem and to Professor Isabel Ribeiro, who always pushed me to go further thanks to her passion for science and research.

Thanks also to my colleague and co-worker, Filipe Valente, for his friendship, helpful ideas in this project and for always having time to come up with great CAD models.

A big thanks to Diogo Silvério and Nelson Homem; I will never forget the time we spent together.

I owe my parents for their interminable support, for always being there during the good times and bad times.

Finally, I want to thank my girlfriend, Nô, for always being there, besides the distance between Paris and Lisbon, for always reminding me what matters the most and for all her patience and love.

Thank you all to the others that were important for me during this period and contributed to my success.

Contents

List of Tables

List of Figures

List of Acronyms

2D Two Dimensional

3D Three Dimensional

4WS Four-Wheel Steering

AI Artificial Intelligence

AGV Automated Guided Vehicle

CAD Computer-Aided Design

CD Cell Decomposition

CDT Constrained Delaunay Triangulation

CoG Center of Gravity

CPRHS Cask and Plug Remote Handling System

CTS Cask Transfer System

DoF Degrees of Freedom

DP Docking Port

DT Delaunay Triangulation

EKM Extended Kinematic Model

HCB Hot Cell Building

ICR Instantaneous Center of Rotation

IO ITER Organization

IST Instituto Superior Técnico

ITER International Thermonuclear Experimental Reactor

OP Obstacle Point

PR Pure Rolling

RH	Remote Handling
RRT	Rapidly-exploring Random Tree
RWS	Rolling Without Slipping
SA	Simulated Annealing
TB	Tokamak Building
TES	Trajectory Evaluator and Simulator
VV	Vacuum Vessel
VVPC	Vacuum Vessel Port Cell
WMR	Wheeled Mobile Robot
WV	Wheeled Vehicle

List of Symbols

$\mathcal{A}$	set notation for robot
$\mathcal{C}$	configuration space
$\mathcal{C}_{free}$	free space
$\mathcal{C}_{obst}$	obstacle space or obstacle region
C_{li}	longitudinal ith wheel stiffness
C_{si}	lateral/cornering ith wheel stiffness
$\boldsymbol{e_n}$	vector position of F_n, $n = \{1, \cdots, N\}$
F	front wheel index
$\boldsymbol{F_e}$	elastic force
$\boldsymbol{F_r}$	repulsive force
$\boldsymbol{F_n}$	nth external force acting on the rigid body, $n = \{1, \cdots, N\}$
F_{li}	longitudinal force of the ith wheel
F_{si}	lateral/side force of the ith wheel
G	vehicle Center of Gravity (CoG)
$\mathcal{G}$	graph that captures the structure of $\mathcal{C}_{free}$
i	front or rear wheel index, $i = \{F, R\}$
I_z	inertia moment around the CoG of the vehicle/rigid body
J	jerk
Ker	kernel or null space of a matrix
K_d	damping gain
K_e	elastic gain
K_t	torsional gain
K_r	repulsive gain

m	mass of the vehicle or rigid body
M_s	steering wheel torque
M_d	driving wheel torque
M	vehicle wheel base $\equiv$ distance between the front and rear wheel
M_i	distance between the ith wheel and the vehicle CoG
N_i	normal reaction of the ground on the ith wheel
$\mathcal{O}$	obstacle region
$\boldsymbol{p_j}$	jth path point/pose
$\boldsymbol{q}$	configuration of the vehicle/robot/rigid body, $\boldsymbol{q} \in \mathcal{C}$
$\boldsymbol{q_I}, \boldsymbol{q_G}$	initial and goal configuration that define the motion query
$\boldsymbol{q_j}$	jth path pose
r	wheel radius
r'	vehicle turning radius
R	rear wheel index
s_j	speed at the jth path point
S_h	hth triangle cell sequence, $h = \{1, \cdots, H\}$
T_n	nth triangle cell, $n = \{1, \cdots, N\}$
$\boldsymbol{v}$	vehicle linear velocity
$\boldsymbol{v_i}$	linear velocity of the ith wheel
$\boldsymbol{v_i^{eff}}$	effective linear velocity of the ith wheel
v_l	wheel longitudinal linear velocity
v_s	wheel lateral linear velocity
W	weight force applied on the wheel
$\mathcal{W}$	set notation for world, space where the robot moves
$\boldsymbol{x}$	sate of the vehicle/robot, $x \in \mathcal{X}$
$\boldsymbol{x_I}, \boldsymbol{x_G}$	initial and goal state that define the motion query

α_i	slip angle of the ith wheel
λ	vehicle yaw rate $\equiv$ vehicle angular acceleration
β	vehicle side-slip angle
β_i	side-slip angle of the ith wheel
π_l, π_s	longitudinal and lateral wheel plane
Σ	wheel longitudinal slip
σ	wheel longitudinal slip ratio
μ	wheel coefficient of friction
τ	continuous path
τ^s	sampled path
θ_i	steering angle of the ith wheel
θ	vehicle/rigid body orientation
φ	wheel angular velocity

Chapter 1

Introduction

The theme of the dissertation is integrated in the International Thermonuclear Experimental Reactor (ITER) project, a world - wide research experiment that aims to explore nuclear fusion as a viable resource of energy for the coming years. Besides this major scientific objective, ITER aims to demonstrate that the future fusion power plants can be safely and effectively maintained through Remote Handling (RH) techniques, due to the restrictive presence of humans in activated areas.

Among the various RH systems that are expected to operate in ITER, the dissertation focus on a large and complex transporter unit that was chosen for the transfer of heavy and contaminated loads between the two main buildings of ITER, the Tokamak Building (TB) and the Hot Cell Building (HCB). The overall transporter is moved by a air-cushioned vehicle that has autonomous capabilities and a kinematic rhombic configuration.

This chapter describes in more depth this large-scale project, providing the required insight on fusion energy. It underlines the major challenges involved, with particular interest on mobile robotics aspects issued to the above referred RH system. The chapter finishes by presenting the main objectives and contributions of the dissertation.

1.1 Scientific and Industrial Context

1.1.1 Historical Overview of Tokamak-based Fusion

The research on fusion energy has begun approximately in the mid of 1950s, when fusion machines started to be used as an experimental tool to control fusion reactions, in countries such as the Soviet Union or the United States of America (USA). By that time, the practical application of the fusion power was thought as a challenging and distant realization. A few years later in the Soviet Union, experiments with tokamak – a doughnut shaped magnetic chamber – devices begun, predicting a possible successful application of fusion energy in a period of 20 years (see Figure 1.1-left).

Experiments continued to be performed and in 1991, the Joint European Torus (JET), depicted in Figure 1.1-center, achieved the world's first controlled release of fusion energy, using a mixture of

Deuterium and Tritium as the fueling gas [Wes00], [PS03]. Also, important experiments were conducted at the Tokamak Fusion Test Reactor (TFTR) in Princeton, USA (see see Figure 1.1-right). Together, JET and TFTR were important projects on the development of fusion power in a tokamak confinement basis, by showing that significant amounts of it could be produced. Other devices such as, the Tore Supra Tokamak [HMB+04] and the TRIAM-1M (Tokamak of the Research Institute for Applied Mechanics) in Japan [ISN+99], produce exceptional results by providing long-duration fusion experiments.

It is worth noting that the majority of magnetic fusion research to this day has followed the tokamak approach [URLFP], but other approaches exist, such as the concept of "mechanical fusion" through the use of lasers, which is not covered here.

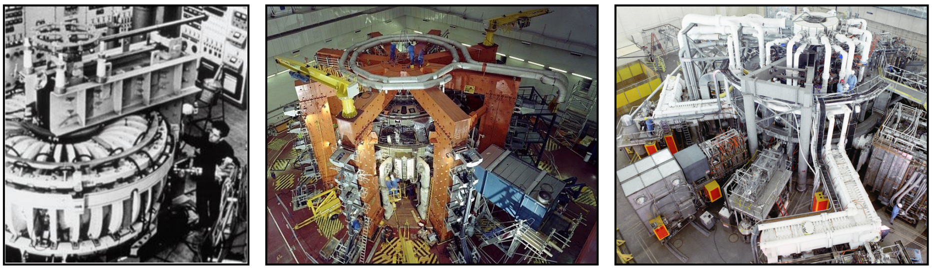


Figure 1.1: Left: first tokamak was developed, in the late of 1950s, in Russia. Center: view of the JET tokamak, the largest European fusion device. Right: The TFTR operated from 1982 to 1997.

1.1.2 ITER, the "Way" to Fusion Energy

Besides the scientific interest in all the previous referred experiments, there is a practical need for developing and exploring fusion as a source of energy for the humankind benefit. The shortage predictions on liquid fuels, especially with the inevitable oil extraction decline, require an urgent development and exploration of new sources of energy. As illustrated in Figure 1.2-left the current energy supply policy is mostly based on fossil fuels — oil, coal an natural gas — representing almost 80% of the total energy consumption. To worsen this scenario, the world population is expected to growth from 6 to 9 billion people until 2050 [URLWP] resulting on an expressive raise of the energy demand (see Figure 1.2-right). According to [Sch06], no single technology is likely to provide all of the world's future energy needs and replace the actual oil-based energy infrastructure. It is important to achieve a more sustainable mix of fossil fuels but, more importantly, develop an energy consumption-frame based on new technologies and alternative energies such as solar, geothermal and nuclear, fission and fusion, power.

In this context fusion energy may play a major rule in the future. Future fusion plants will need fuel to work and produce energy. This fuel will be based on a mixture of Deuterium and Tritium, two hydrogen isotopes, which are readily available on earth; The Deuterium is extracted from the sea water and Tritium will be produced through Lithium, which exists in abundance on the earth's crust. The basic principle the fusion reaction consists in "fusing" these two hydrogen isotopes. As products, this reaction will release Helium nucleus, neutrons and a big amount of energy. This fusion reaction, will be inherently safe, with no possibility of "meltdown" or "runaway reactions", two possible accidental situations that may arise in

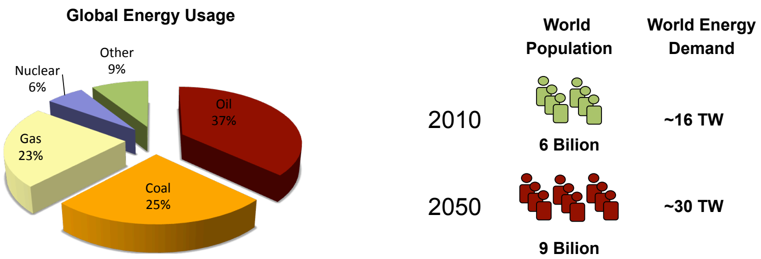


Figure 1.2: Global energy picture: current energy supply policy [URLED] (left); Growth predictions for world population and energy demand (right).

the traditional fission power plants. Also, when looking at long term radioactive decay of the material, it can be affirmed, and now citing [URLIO], that “the fusion reaction will produce no long-lived waste; within 100 years, the radioactivity of the materials will have diminished in such a significant way, that the materials can be recycled for use in future fusion plants”.

Therefore, fusion energy can be presented as a sustainable, environmentally acceptable and safe energy. In a long term, it will likely be a viable alternative for carbon-free production of large scale energy source. The only drawback of fusion power is that it still requires further development to become reality.

A major project pursuing the fusion dream is the ITER, which in Latin means “the way”, a metaphor that truly demonstrates the ambitions of this project.

The idea of ITER originally appeared on the Geneva superpower summit in 1985. Currently composed of seven different members (European Union (EU), Japan, USA, Russian Federation, India, China and the Republic of Korea), ITER represents the largest scientific international project in the earth [URLF4Eb, URLCor]. The plant construction has already begun at the selected site of Cadarache, in the south of France. The first plasma operations expected to be performed in 2018.

One of the ITER’s objectives, is to release 10 times as much energy as it will use to initiate the fusion reaction. For 50 MW of input power, ITER will generate 500 MW of output power, working on pulses of around 7 minutes. With non-commercial purposes, ITER will provide design basis for the technology to be used in the Demonstration Power Plant (DEMO) in 2030, a project that will bridge the gap between ITER and the first Fusion plant.

1.2 The Need for Remote Handling and Challenges Involved

The ITER will be composed by two main buildings, the TB lodging the tokamak reactor and the HCB, which will work mainly as a support area. Figure 1.3-left gives an impression of these buildings.

During ITER lifetime, the internal components of the Vacuum Vessel (VV) of the reactor, such as the Blanket and Divertor modules detached in Figure 1.3-right, will become activated due to the exposition to highly energetic neutrons released during the fusion reaction. Also, these in-vessel materials might get contaminated with small amounts of radioactive dust like Tritium radwaste. Hence, the components that provide the base functions for the ITER machinery will need to be periodically upgraded and inspected. To manage such operations and providing that human presence will be not authorized in activated areas, the ITER maintenance system will mostly rely on RH devices.

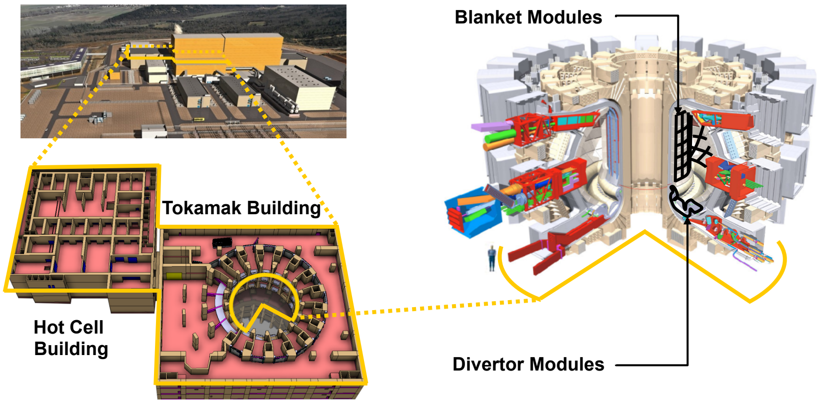


Figure 1.3: Left-bottom: CAD models of the TB and HCB. Right: an artistic impression of the tokamak device of ITER with the Blanket and Divertor modules highlighted in black.

The foreseen RH equipment will have a great impact on the design and assembly of the remaining ITER components, for instance, on building structural aspects and interfaces. During ITER operation, the RH equipment will be required to operate under specific and adverse conditions that are characterized by high gamma radiation, poor visibility spaces, some level of magnetic field and in some cases, ultra-high vacuum clean conditions. These conditions will significantly compromise the correct functioning and durability of fixed and mobile electronic devices (e.g. sensors). Moreover, the cluttered and geometrically complex environment and the particular nature of the RH problem, with heavy and large components to be handled, make the ITER's RH a unique challenge with an unprecedented complexity.

The development and test of reliable RH equipments is important so to guarantee a timely integration of this key technology on the overall project conclusion. For this reason the ITER Organization (IO) has distributed different RH procurement packages among its domestic agencies to hasten this developing process. Each procurement package refers to a particular RH system as described in detail in Appendix A.

The dissertation focus on aspects related to a particular RH system, a large transporter that will transfer components inside the buildings of ITER and is presented next with extended considerations.

1.2.1 Cask Transfer Using Rhombic-like Vehicles

The Cask and Plug Remote Handling System (CPRHS) represented in Figure 1.4-left, is a large and complex transport unit that has been adopted as the reference solution to transport heavy and contaminated components between the TB and the HCB. These components include the already mentioned Divertor cassettes and the Blanket modules from the interior of the reactor, but also some of the RH systems listed in Appendix A, such as the In-Vessel Viewing System (IVVS) and the In-Vessel Transporter (IVT).

Each CPRHS is composed by three sub-systems:

- **The cask envelope** – Container that enclosures the in-vessel components and the RH tools to be transported. This "transporter casing" also embodies additional equipments like an handling tractor to lift and position the load and a manipulator for bolting, cutting and welding inspections;

- **The pallet** – This component serves as interface between the cask and the Cask Transfer System (CTS), which is described next. The pallet is also equipped with an handling platform to support the cask load and help on docking procedures;
- **The Cask Transfer System** – The CTS, which can act as an autonomous vehicle, is composed by a set of air casters, which provide high load capacity to the overall transporter. When underneath the pallet the CTS transports the entire CPRHS but it can also move independently of the pallet and cask.

From past studies related to the CTS design [RLAF97], a decision was made to adopt this vehicle with a rhombic kinematic configuration with two pairs (one for spare purposes) of drivable and steerable wheels positioned on the front and rear of the vehicle and two swivel wheels on the sides (see Fig.1.4-right). This kinematic configuration endows the transporter with an high maneuvering ability and flexibility, which are key traits when considering the cluttered nature of the ITER environments, as illustrated in Fig.1.4-left.

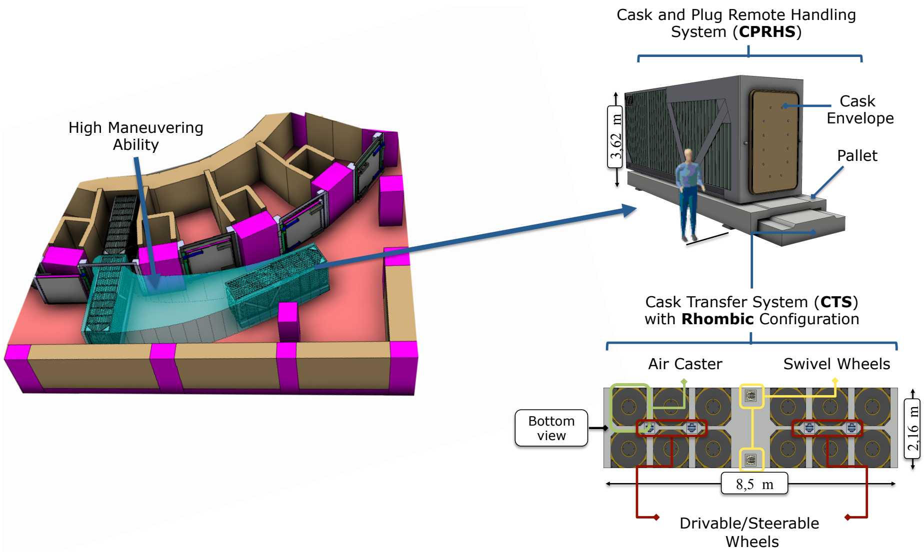


Figure 1.4: Left: Maneuvering ability of the CPRHS. Right: schematic view of the CPRHS and the CTS with its rhombic configuration and air cushion system.

The geometry of the CPRHS and payload vary according to the cask and components to be transported. As a reference, the largest CPRHS dimensions are 8.5m x 2.62m x 3.62m (length x width x height).

From this time forward, the term "rhombic vehicle" is used whenever the presented considerations can be used indifferently for the CPRHS or the CTS. This unified designation is clarified later in Section 2.1 with a precise system definition.

1.3 Mobile Robotics Challenges for the Rhombic Vehicle

During ITER operation, the rhombic vehicle is expected to perform different missions, which include the transfer of in-vessel components and RH tools between the TB and the HCB, the rescue and recover of other rhombic vehicles and also parking operations in the HCB to exchange casks.

The above listed missions may be performed autonomously requiring the vehicle to move along optimized prescribed routes. So far, the in-charge RH teams of IO have decided for a flexible guidance solution as opposed to rails or any other hard-guidance solutions. A question remains whether these routes will be implemented on the floor, or, instead, used as virtual trajectories on a non contact guidance.

Guidance and navigation issues are discussed in more detail in Appendix A. As later explained in Chapter 3 and without loss of generality, the dissertation considers the motion planning problem as finding a trajectory regardless of the system guidance used.

Moving an autonomous vehicle in a particular environment is known to be a complex and interdisciplinary problem that has been broadly addressed both by robotics and control research communities. For the specific case of the rhombic vehicle and regardless off the flexible guidance solution to be adopted in the future, its navigation and guidance system can be described within the same integrated framework, as depicted in Figure 1.5.

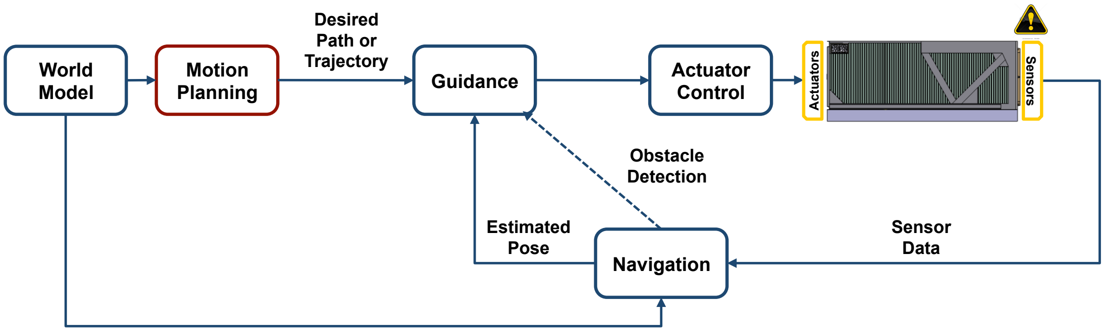


Figure 1.5: Schematic view of a generic navigation and guidance framework.

Within this framework three different problems can be stated corresponding to the following systems or modules:

1 – Motion planning – Refers both to the evaluation of a path or trajectory connecting a start and goal vehicle configuration and that must be followed by the vehicle. In addition to these configurations the motion planning module also receives as input the geometric model of the environment. Motion plans are evaluated resorting to proper motion planning techniques [Lat91, Lav03] and may be required to comply with different optimization requirements, such as obstacle clearance, length of the plan or even vehicle differential constraints (coming from its kinematics or dynamics);

2 – Guidance – The guidance system receives as input the motion plan evaluated in the previous planning stage. Working as an on-line application, this module is responsible for determining the appropriate feedback control law (supposing a feedback scheme) that enables the vehicle to correctly track the provided reference and still avoid the obstacles. Once the control law is determined by the guidance system, it must be "translated" to a lower level controller. For instance, to achieve a specific wheel velocity and steering angle, the actuators of the vehicle wheels must be actuated by driving and steering torques.

3 – Navigation – The performances of the guidance system highly depend on the navigation module, which resorting on the raw data from sensors, provides the means to: (1) estimate the vehicle pose (position plus the orientation), with respect to a given environment frame or reference (e.g. the path). This is usually performed by a localization system; (2) Avoid obstacles during motion. This is possible by using adequate collision avoidance features based on sensory fusion capabilities or anti-collision alarm systems based on hardware applications (e.g., infrared light curtains);

Mapping capabilities were not referred since ITER environments are fairly static and map information is know *a priori*. Another important issue that is convenient to refer is that the harsh environments of ITER, due to the high radiation levels, will reduced the spectrum of applicable electronic sensors and actuators thus requiring adequate procurement. For instance, considering that the rhombic vehicle is only partially shielded with respect to radiation, the placement of dedicate localization sensors may be required in the building rather on the vehicle (see the call for attention in Figure 1.5).

1.4 Novelties and Major Contributions

The dissertation focus on the motion planning problem inherent to the autonomous navigation problem of the rhombic vehicle. As underlined in the last section, the motion planning comprises only one of the required stages to achieve a full autonomous level. Even though, it will allow to tackle important issues involved in the challenging assignment of moving a large and heavy duty vehicle in the confined scenarios of ITER.

To handle this general problem, the dissertation has been roughly structured into two different parts corresponding to two major dissertation objectives:

- **Vehicle Modeling Analysis**
  The first part of the dissertation addresses the mathematical modeling of the rhombic vehicle, which design is still in a very early stage, accounting only with virtual mock-up designs. To achieve a more comprehensive understanding of the vehicle capabilities one needs to develop simulation tools capable of capturing the behavior of this RH unit. Moreover, the atypical rhombic configuration of this vehicle turns out to be barely described in the literature. Research has been performed mostly on Four-Wheel Steering (4WS) vehicles whose mechanical configuration may be simplified to the bicycle model, which is close to the rhombic configuration. Considering the lack of dedicated bibliography, the dissertation carries out a complete study regarding the kinematic and dynamic behavior of the rhombic vehicle.
- **Development of motion planners**
  This part aims to develop motion planners, capable of handling the motion planning problem of the rhombic vehicle, which is formulated in Section 3.3 . For that and after providing the sufficient insight on the motion planning subject, two different approaches are pursued each one following a specific planning methodology: (1) using a decoupled refinement strategy, composed of a path planning, path optimization and trajectory design stages. This approach heavily relies on path deformation techniques to obtain feasible and optimized trajectories and present two novel methods within the scope of path optimization; (2) Benefiting from the modeling analysis carried

out in the first part, the second approach follows a trajectory planning philosophy using a particular randomized planning technique, the Rapidly-exploring Random Tree (RRT). The planners herein developed are able to explicitly handle the vehicle kinematic and dynamic constraints. The main dissertation contribution of this second part comprises the combination of a RRT-planner with embedded vehicle kinematic simulator with the Simulated Annealing (SA) optimization method. As explained later in Chapter 3, more precisely in Subsection 3.1.2, this consists on planning under differential constraints, an higher instance of motion planning also referred to as trajectory planning.

Some of the developments presented in the dissertation have been presented in the following international events:

- 7th IFAC Symposium on Intelligent Autonomous Vehicles (IAV 2010), September 2010, Lecce, Italy, with the presentation of an authored scientific publication [FVVR10];
- 1st IPFN Workshop, in Lisbon, Portugal, with a poster presentation entitled of "A motion planning methodology for rhombic-like vehicles for ITER remote handling operation";
- 26th Symposium on Fusion Technology (SOFT 2010), in October 2010, Porto, Portugal with a co-authored scientific publication [VVFR10].

An additional authored work entitled of "Path Optimization for Rhombic-Like Vehicles: An Approach Based on Rigid Body Dynamics" [FVVR11], was recently accepted for publication in the Proceedings of the 15th IEEE International Conference on Advanced Robotics (ICAR 2011), to be held in Tallinn, Estonia, on June 20-23, 2011.

1.5 Dissertation Organization

The dissertation is structured as follows. Chapter 2 presents an extended analysis on the motion behavior of the rhombic vehicle, including the development of different simulation models. In Section 2.2 a pure kinematic model is developed based on the wheel kinematic constraints. In Section 2.3, the analysis is extended to a dynamic point of view in an attempt to construct a more reliable and realist simulation model. This chapter closes with Section 2.4, with an alternative modeling approach, which allows to achieve a trade-off between model accuracy and complexity.

Chapter 3 discusses the need and interest of the motion planning. The formulation of the general motion planning problem comprises Section 3.1. A survey on the available techniques for solving motion planning problems is presented in Section 3.2. Section 3.3 comprises the end of this chapter by concretely formulating the motion planning problem for the rhombic vehicle.

Introduced the theoretical basis of the motion planning problem, Chapter 4 and Chapter 5 then present the developed planning techniques based on two different planning strategies.

Chapter 4 covers a decoupled and refinement planning strategy comprising three planning stages: the path planning, path optimization and trajectory evaluation. This strategy encloses two different approaches. Section 4.1 presents a combinatorial-based approach, which combines the Constrained Delaunay Triangulation (CDT) (Subsection 4.1.1) with a new path optimization grounded on the elastic bands concept (Subsection 4.1.2). Section 4.2, following a randomized approach, first presents the RRT algorithm in subsection 4.2.1 and finishes with the presentation of a novel path optimization method in Subsection 4.2.2 that fully explores the rhombic vehicle capabilities. Finally, Section 4.3 handles the trajectory evaluation problem, by presenting a method that considers both safety constraints (from the obstacle proximity) and vehicle dynamic constraints on the evaluation of speed profiles for the optimized paths.

Chapter 5 proposes to tackle the motion planning problem of the rhombic vehicle by following a trajectory planning approach and relying on the RRT algorithm. The general problem of trajectory planning is formulated in Section 5.1. Different RRT-based planners, available in the literature, are presented and discussed in Section 5.2. This chapter closes by proposing a new RRT-based planner that is capable of yielding both differential constraints and optimization requirements in the search for a feasible planning solution.

Chapter 6 presents the outcome of the simulated results on ITER and other simulated environments.

The dissertation is concluded in Chapter 7 with a critical discussion on the presented modeling and planning approaches. The chapter closes by underlying relevant issues and referring future work perspectives.

Chapter 2

Mathematical Modeling

A necessary and important step in simulation and control oriented studies is the design of a model capturing the behavior of the systems being studied. For instance, planning techniques developed in Chapter 5 require the explicit use of system simulators to predict and plan feasible trajectories. This preliminary stage is usually referred to as mathematical modeling or system modeling and comprises the subject of this chapter.

2.1 System Definition

A system usually comprises a component or a group of interacting components. Therefore, an important step that usually precedes the modeling phase, is the system definition, i.e., the definition of the component or group of components whose properties required to be modeled. This step is rather important when leading with complex systems as it is the case of the CPRHS and comprises the definition of: (1) the systems boundaries defining the virtual or physical limits of the system and inherent components and (2) the system states, which described the system condition and provide useful information for its monitoring, simulation or even control.

In the model developing process, we are concerned about knowing how the inputs acting on the system force the system states (some chosen as outputs) to behave and find a mathematical description for these relations.

As referred in Chapter 1, the dissertation focuses on aspects related with the motion of the CPRHS, a complex assembly system composed by three different modules which will have putative contributions on the overall behavior of the rhombic vehicle. Furthermore, different configurations may appear such as different cask and CTS dimensions, the CTS operating alone or in the CPRHS mode.

Without loss of generality and to avoid misinterpretations and tedious specifications, the dissertation assumes a single system representation for the modeling process, which is represented in Figure 2.1. This planar representation works well since the modeling process will be carried according to Two Dimensional (2D) perspective of the rhombic vehicle motion. As shown in Figure 2.1, the adopted system definition relies on a set of reference parameters which define (without freezing) the system boundaries enabling an easy system interchangeability (CPRHS/CTS, different types of CTS or casks) and operating conditions (e.g., transported load, Center of Gravity (CoG) position).

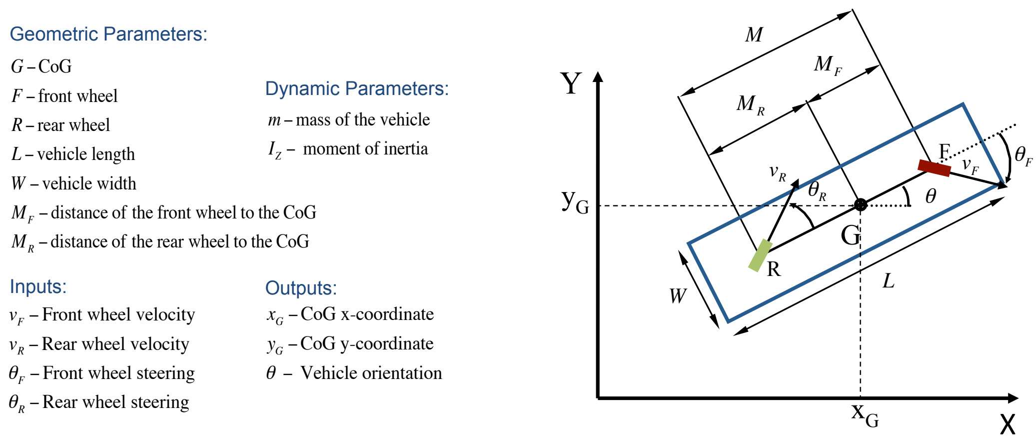


Figure 2.1: System representation.

Figure 2.1 also unveils the chosen inputs for the modeling process which are the linear velocities on the front and rear wheels (i.e., steering plus the wheel linear speed). The systems states vary depending on the modeling perspective (e.g. kinematic or dynamic aspects, chosen frame) and will be referred at the proper time. In what concerns the outputs, it can be stated that the objective of this chapter is to develop different mathematical models capable of describing the evolution of a vehicle reference point (e.g., the CoG) and the vehicle orientation, or in other words, the vehicle's pose, as function of the chosen inputs.

The following basic assumptions are also valid for this system definition:

- The vehicle is considered to be a solid rigid body. The only movable parts are the wheels, which are considered to be undeformable;
- The vehicle moves on a hard, flat surface; Its motion is characterized only on two dimensions (position and orientation);
- The vehicle is considered to be symmetric with respect to its longitudinal and lateral axes;

Another substantial matter concerning the modeling process is the model complexity. The mathematical model has to be detailed as possible to represent accurately and completely all the essential properties of the vehicle system. On the other hand, simplicity is required for efficient and fast simulations and to guarantee model tractability, for instance, for future control purposes. A strategy commonly found in the literature and that is followed in this dissertation, consists on beginning with a simple model that captures the essential of the system behavior. Then, after this first model analysis it may be desirable to construct a more accurate one, incorporating, for example, more realistic assumptions and considering different behavioral aspects.

In accordance to what was stated above, this chapter is organized as follows. The kinematic model of the rhombic vehicle is first derived in Section 2.2, based on the assumption that the Rolling Without Slipping (RWS) constraint affecting the vehicle wheels is satisfied. Taking into consideration the nature of the transportation problem and characteristics of the rhombic vehicle, the modeling process is extended to a dynamical point of view in Section 2.3, bringing slippage phenomena into discussion. Section 2.4 closes this chapter by proposing an alternative model, an extended version of the kinematic model proposed in Section 2.2, which too accounts for slippage phenomena and provides a good tradeoff between model accuracy and simplicity.

2.2 Kinematic Modeling

This section is devoted to the derivation of the kinematic model of the rhombic vehicle and analysis of its motion capabilities and singularities. The purpose of this kinematic study is to establish the mathematical equations describing the relationship between the temporal variations of the vehicle pose (remember, position and orientation) and the linear velocities on the wheels. It consists on a pure geometrical study that is carried out without considering vehicle dynamic properties such as mass, inertia or friction. Also, at this point, forces or torque inputs are not considered.

2.2.1 Nonholonomic Constraints on Wheels

The system defined in the previous section can be decomposed in two primary elements or subsystems: the vehicle body, which is the dissertation primary interest and whose motion is desirable to simulate and the wheels that through the contact forces with the ground, provide the means to generate the vehicle motion.

The overall vehicle motion depends heavily on the kinematics behind of these two subsystems, namely on the wheels. A main feature of the kinematic model of the Wheeled Mobile Robots (WMRs) and the Wheeled Vehicles (WVs), is the presence of nonholonomic constraints due to the RWS and Pure Rolling (PR) conditions between the wheels and the ground. Before unveil the importance of these constraints to the task of kinematic modeling, the meaning of nonholonomic constraint shall first be explained.

One of the first aspects introduced in books about WMRs is the term “constraints” (refer, for instance, to [Lau98] and [SK08]). These constrains are mathematical formulas describing the limitations of the system and are typically divided on the two following types:

- **Nonholonomic constraints** – Constraints that appear in the velocity space of the system and cannot be integrated to position or configurations constraints. These constraints limit the space of possible velocities and can also be referred to as velocity or kinematic constraints. These constraints limit the instantaneous admissible motions of the wheeled system, but they do not restrict the possible achievable configurations;
- **Holonomic constraints** – This type of constraints usually leads to a loss of the system’s Degrees of Freedom (DoF). They are also referred to as configuration constraints.

The most familiar example of a nonholonomic system – system with more than one nonholonomic constraint – is demonstrated by a parallel parking maneuver with a car-like vehicle. When parking, the car is not capable of sliding sideways due to velocity restrictions on the wheels. However, by a combination of driving and steering maneuvers, it can be correctly placed in the parking space. If the restrictions caused by the environment obstacles are ignored, the car can be placed in any configuration, despite its reduced maneuvering ability.

The rhombic vehicle is also an nonholonomic vehicle with velocity restrictions on the wheels. However, as discusses in Subsection 2.2.4, it presents an increased maneuvering ability when compared to the remaining nonholonomic vehicles.

The RWS and the PR conditions, which are assumed to be satisfied, restrict the mobility of the wheeled system with respect to velocities, resulting on nonholonomic constraints and, therefore, enclose information about the overall system's motion. Hence, it seems appropriate to base the formulation of the system's kinematic model on these constraints. This idea was first presented in [ACB91]. In this work the authors proposed a general framework for modeling WMRs with different wheel configurations. Together with [CBA96], these are among the first studies to address the topic of WMR modeling based on nonholonomic constraints. The developments presented in the following, for the unicycle and the rhombic vehicle, are based on the general framework proposed in [ACB91].

2.2.2 Kinematic Model of the Unicycle

The equations of motion for the rolling disk, also referred to as unicycle, are readily available in the literature. For the completeness of the dissertation and since it is an important element of the system definition established in Section 2.1, its kinematics is presented here, unfolding the general framework later used in Subsection 2.2.3 to determine the kinematic model of the rhombic vehicle.

The unicycle system, represented in Figure 2.2, can be described by the following configuration (state) vector :

$$\boldsymbol{q} = \begin{bmatrix} x \\ y \\ \theta \end{bmatrix}, \tag{2.1}$$

where (x, y) are the Cartesian coordinates and θ, the orientation of the wheel with respect to the x-axis.

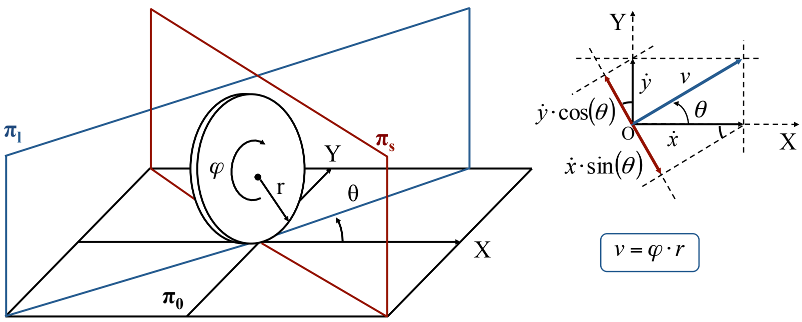


Figure 2.2: The unicycle system: Generalized coordinates and the RWS constraint.

The unicycle system is subjected to the following RWS constraint,

$$\dot{x} \cdot \sin\theta - \dot{y} \cdot \cos\theta = 0 \tag{2.2}$$

which entails that the linear velocity of the contact point of the wheel has zero component in the orthogonal direction (π_s) to the vertical wheel plane (π_l). Constraint 2.2 can be written under a matrix formulation as presented next.

$$\begin{bmatrix} \sin\theta & -\cos\theta & 0 \end{bmatrix} \cdot \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = 0 \Rightarrow A \cdot \dot{\boldsymbol{q}} = 0 \tag{2.3}$$

According to [ACB91], the homogeneous system just presented can be used to formulate a kinematic model of the unicycle system. All the admissible generalized velocities for the unicycle are constrained to belong to a reduced subset of solutions, which is given by the kernel (Ker) or null space of the constraint matrix A. In other words, the movement of the unicycle is hidden in the null space of the constraint matrix A. In a more formal way it can be stated:

For a specific homogeneous system $A\boldsymbol{x} = 0$ (with A being a mxn matrix), the subset of vectors (column vector with n entries), solution to this system, is given by the null space of the matrix A. The null space is thus be defined as

$$Ker(A) = \{\boldsymbol{x} \in \Re^n \mid A\boldsymbol{x} = 0\}, \tag{2.4}$$

for which the following prepositions are valid:

P1: $\mathbf{0} \in Ker(A)$

P2: $\boldsymbol{V}, \boldsymbol{U}, \in Ker(A) \Rightarrow \boldsymbol{V} + \boldsymbol{U} \in Ker(A)$

P3: $C \in \Re^n, \boldsymbol{V} \in Ker(A) \Rightarrow C \cdot \boldsymbol{V} \in Ker(A)$

In this case, the null space of the matrix A is given by

$$Ker(A(\boldsymbol{q})) = \begin{bmatrix} \cos\theta & 0 \\ \sin\theta & 0 \\ 0 & 1 \end{bmatrix}, \tag{2.5}$$

and all the generalized velocities (i.e., all the solutions of $A\boldsymbol{x} = 0$) are given by the following linear combination:

$$\dot{\boldsymbol{q}} = \begin{bmatrix} \cos\theta \\ \sin\theta \\ 1 \end{bmatrix} \cdot \iota_1 + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \cdot \iota_2 \tag{2.6}$$

where, ι_1 and ι_2, are any real scalars. In this case, these scalars can be defined as, $\iota_1 = v$ and $\iota_2 = \varphi$, which are the linear and angular velocities of the unicycle, respectively, yielding

$$\dot{\boldsymbol{q}} = \begin{bmatrix} \cos\theta \\ \sin\theta \\ 1 \end{bmatrix} \cdot v + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \cdot \varphi \tag{2.7}$$

The system of equations in (2.7) completely define the kinematic model of the unicycle.

2.2.3 Kinematic Model of the Rhombic Vehicle

The procedure just presented can be used to establish the kinematic model for the rhombic vehicle system. For the purposes of the dissertation and as discussed in Section 2.1, it would be desirable to develop a kinematic description of the rhombic vehicle motion with an explicit representation of the linear speeds (v_F, v_R) and steering angles (θ_F, θ_R) of the front (F) and rear (R) wheels, as represented in Figure 2.3. The state vector chosen to be the output for this model should represent the temporal variations of the

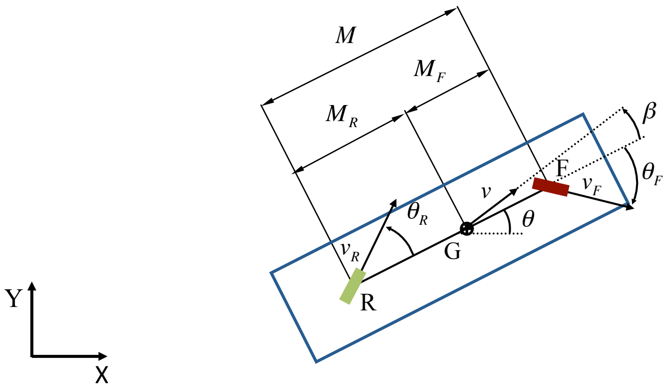


Figure 2.3: Generalized coordinates of the rhombic vehicle.

vehicle pose for a specific point reference. Without loss of generality, the position of the CoG of the vehicle is chosen as the reference point, as detailed schematically in Figure 2.3. The chosen state vector for the rhombic vehicle's pose is then,

$$\boldsymbol{q} = \begin{bmatrix} x_G \\ y_G \\ \theta \end{bmatrix} \tag{2.8}$$

Under the RWS assumption, the system is subjected to the following two nonholonomic constraints, one for each wheel:

$$\dot{x}_F \cdot \sin(\theta + \theta_F) - \dot{y}_F \cdot \cos(\theta + \theta_F) = 0 \tag{2.9a}$$

$$\dot{x}_R \cdot \sin(\theta + \theta_R) - \dot{y}_R \cdot \cos(\theta + \theta_R) = 0 \tag{2.9b}$$

with (x_F, y_F) and (x_R, y_R), being the Cartesian coordinates of the front and rear wheel, respectively. Using the rigid body geometric constraints the following equations can be established:

$$x_F = x_G + M_F \cdot \cos\theta \Rightarrow \frac{d}{dt} \Rightarrow \dot{x}_F = \dot{x}_G - M_F \cdot \sin\theta \cdot \dot{\theta} \tag{2.10a}$$

$$y_F = y_G + M_F \cdot \sin\theta \Rightarrow \frac{d}{dt} \Rightarrow \dot{y}_F = \dot{y}_G + M_F \cdot \cos\theta \cdot \dot{\theta} \tag{2.10b}$$

$$x_R = x_G - M_R \cdot \cos\theta \Rightarrow \frac{d}{dt} \Rightarrow \dot{x}_R = \dot{x}_G + M_R \cdot \sin\theta \cdot \dot{\theta} \tag{2.10c}$$

$$y_R = y_G - M_R \cdot \sin\theta \Rightarrow \frac{d}{dt} \Rightarrow \dot{y}_R = \dot{y}_G - M_R \cdot \cos\theta \cdot \dot{\theta} \tag{2.10d}$$

Using the geometric constraints (2.10), the nonholonomic equations (2.9) become

$$\dot{x}_G \cdot \sin(\theta + \theta_F) - \dot{y}_G \cdot \cos(\theta + \theta_F) - M_F \cdot \cos\theta_F \cdot \dot{\theta} = 0 \tag{2.11a}$$

$$\dot{x}_G \cdot \sin(\theta + \theta_R) - \dot{y}_G \cdot \cos(\theta + \theta_R) + M_R \cdot \cos\theta_R \cdot \dot{\theta} = 0, \tag{2.11b}$$

which in the matrix is the same as having

$$\begin{bmatrix} \sin(\theta+\theta_F) & -\cos(\theta+\theta_F) & -M_F \cdot \cos\theta_F \\ \sin(\theta+\theta_R) & -\cos(\theta+\theta_R) & +M_R \cdot \cos\theta_R \end{bmatrix} \cdot \dot{\boldsymbol{q}} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \Rightarrow C(\boldsymbol{q}) \cdot \dot{\boldsymbol{q}} = 0. \tag{2.12}$$

As for the case of the unicycle, solving the null space of the constraint matrix C allows to determine the set of admissible generalized velocities for the rhombic vehicle, i.e., all the attainable values for $\dot{q}$. The null space of C is given by (computed with MATHEMATICA software application [URLMat])

$$Ker(C(\boldsymbol{q})) = \begin{bmatrix} \frac{M_R \cdot \cos\theta_R \cdot \cos(\theta+\theta_F) + M_F \cdot \cos\theta_F \cdot \cos(\theta+\theta_R)}{\sin(\theta_F - \theta_R)} \\ \frac{M_R \cdot \cos\theta_R \cdot \sin(\theta+\theta_F) + M_F \cdot \cos\theta_F \cdot \sin(\theta+\theta_R)}{\sin(\theta_F - \theta_R)} \\ 1 \end{bmatrix}, \tag{2.13}$$

which can be seen as a basis for the null space of C. Using the properties ($P1, P2$ and $P3$) previously introduced in Subsection 2.2.2 and as it is demonstrated in Appendix C, the following vector

$$T = \begin{bmatrix} \frac{M_R \cdot \cos\theta_R \cdot \cos(\theta+\theta_F)}{M} \\ \frac{M_R \cdot \cos\theta_R \cdot \sin(\theta+\theta_F)}{M} \\ \frac{\sin\theta_F \cdot \cos\theta_R}{M} \end{bmatrix} \cdot \iota_1 + \begin{bmatrix} \frac{M_F \cdot \cos\theta_F \cdot \cos(\theta+\theta_R)}{M} \\ \frac{M_F \cdot \cos\theta_F \cdot \sin(\theta+\theta_R)}{M} \\ \frac{-\sin\theta_R \cdot \cos\theta_F}{M} \end{bmatrix} \cdot \iota_2, \tag{2.14}$$

where, $\iota_1 = \iota_2$, also belongs to the null space of the constraint matrix C.

The longitudinal components of v_F and v_R must be equal, since they are rigidly connected through the vehicle chassis.

$$v_F \cdot \cos\theta_F = v_R \cdot \cos\theta_R \Rightarrow \frac{v_F}{\cos\theta_R} = \frac{v_R}{\cos\theta_F} \tag{2.15}$$

Therefore, the real scalars, ι_1 and ι_2, can be chosen as to match available vehicle inputs as follows.

$$\iota_1 = \frac{v_F}{\cos\theta_R} \quad and \quad \iota_2 = \frac{v_R}{\cos\theta_F} \tag{2.16}$$

Introducing (2.16) into (2.14), T simplifies to

$$T = \begin{bmatrix} \frac{M_R \cdot \cos(\theta+\theta_F)}{M} \\ \frac{M_R \cdot \sin(\theta+\theta_F)}{M} \\ \frac{\sin\theta_F}{M} \end{bmatrix} \cdot v_F + \begin{bmatrix} \frac{M_F \cdot \cos(\theta+\theta_R)}{M} \\ \frac{M_F \cdot \sin(\theta+\theta_R)}{M} \\ \frac{-\sin\theta_R}{M} \end{bmatrix} \cdot v_R, \tag{2.17}$$

Recalling that $T \in Ker(C)$, it can be stated that $C(\boldsymbol{q})Ker(\boldsymbol{q}) = C(\boldsymbol{q})T = 0$ and the sate vector $\dot{q}$ can be described as

$$\begin{bmatrix} \dot{x}_G \\ \dot{y}_G \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \frac{M_R \cdot \cos(\theta+\theta_F)}{M} \\ \frac{M_R \cdot \sin(\theta+\theta_F)}{M} \\ \frac{\sin\theta_F}{M} \end{bmatrix} \cdot v_F + \begin{bmatrix} \frac{M_F \cdot \cos(\theta+\theta_R)}{M} \\ \frac{M_F \cdot \sin(\theta+\theta_R)}{M} \\ \frac{-\sin\theta_R}{M} \end{bmatrix} \cdot v_R. \tag{2.18}$$

The scalars ι_1 and ι_2 in Equation (2.14) may assume different meanings or have no relation with the actual control inputs available. For this reason, the system of equations in (2.18) is called the kinematic model of the rhombic vehicle.

Another interesting structure for the kinematic model was found in [WQ01], for an equivalent vehicle configuration (4WS vehicle). The proposed model is the following

$$\begin{bmatrix} \dot{x}_G \\ \dot{y}_G \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \cos(\theta+\beta) \\ \sin(\theta+\beta) \\ \dfrac{\cos\beta \cdot [\tan\theta_F - \tan\theta_R]}{M} \end{bmatrix} \cdot\ v, \tag{2.19}$$

where,

$$\beta = \arctan\left(\frac{M_R \cdot \tan\theta_F + M_F \cdot \tan\theta_R}{M}\right), \tag{2.20}$$

is the side-slip angle of the vehicle and

$$v = \frac{v_F \cdot \cos\theta_F + v_R \cdot \cos\theta_R}{2 \cdot \cos\beta}. \tag{2.21}$$

The main feature distinguishing this model from the one presented in (2.18), is the fact that it allows the simulation of the vehicle motion directly through the desired longitudinal vehicle speed (v) instead of imposing individual linear speed for each wheel.

2.2.4 General Considerations

Since the kinematic model of the rhombic vehicle has already been presented, it is now appropriate to extend some considerations about the rhombic vehicle motion.

1. Motion capabilities and maneuvering ability

Depending on the steering and driving configurations, the rhombic vehicle is able to perform different types of motion (see Figure 2.4): (a) Crab-like motion or crab steering, i.e., wheels steered in equal angles, in x-direction and (b) in y-direction, (c) turning on spot, with both wheels in the perpendicular (90°) position and, (d) any combination of these types of motion, i.e., rotation around the Instantaneous Center of Rotation (ICR) denoted with an "I" from now on.

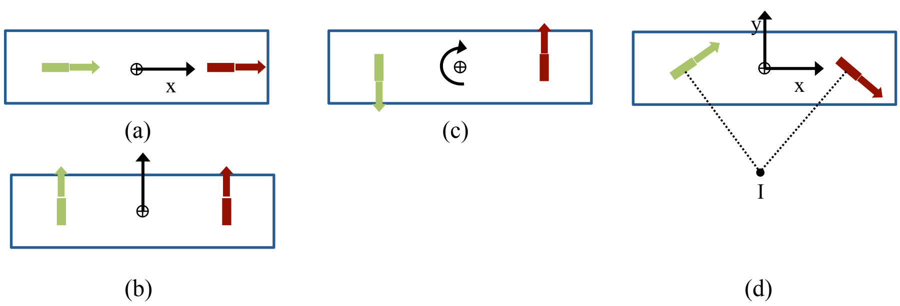


Figure 2.4: Motion capabilities of the rhombic vehicle.

Moreover, the mentioned rhombic configuration allows the vehicle to change not only its curvature radius but also the location of ICR. In fact, the variety of possible drive configurations endows this mobile platform with an increased flexibility and maneuvering ability. These are important traits, when considering the complexity and the cluttered nature of some environments, such as those found in ITER.

The maneuvering advantage of the rhombic configuration becomes even clearer when it is compared with car-like platform as presented next.

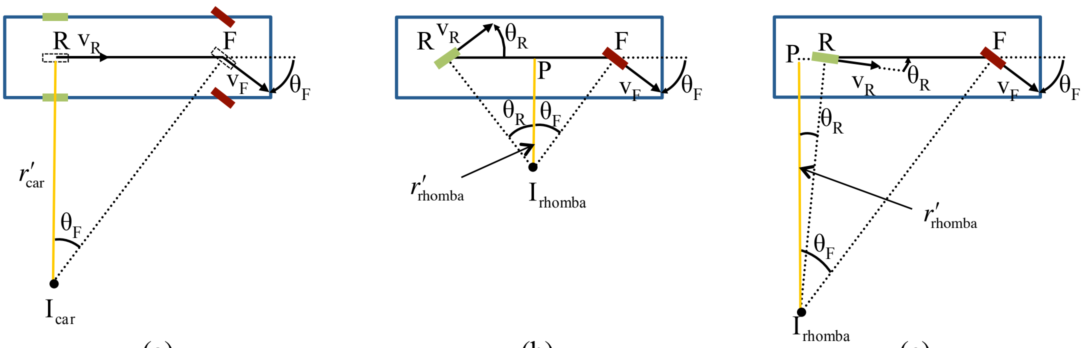


Figure 2.5: Evaluation of the turning radius: (a) On a car-like configuration. (b)-(c) On the rhombic-like configuration with the wheels turning in the opposite (b) and in the same direction (c).

Observing Figure 2.5 and considering the triangles (I_{car}, R, F) and (I_{rhomba}, R, F), the turning radius of the car-like vehicle can be defines as

$$r'_{car} = \frac{\overline{RF}}{\tan\theta_F}, \tag{2.22}$$

whereas the turning radius of the rhombic vehicle can be expressed as function of θ_F as follows

$$r'_{rhomba} = \frac{\overline{PF}}{\tan\theta_F.} \tag{2.23}$$

The relation (2.23) is valid, when the wheels turn in the same or in opposite directions. However, it should be mentioned that the length $\overline{PF}$ varies accordingly to these two situations. Taking this into account, the following comparison can be established:

- $r'_{rhomba} < r'_{car}$: when the rear wheels of the rhombic vehicle turn in the opposite direction to the front ones (see Figure 2.5-b),
- $r'_{rhomba} > r'_{car}$: when the rear wheels of the rhombic vehicle turn in the same direction than the front ones (see Figure 2.5-c).

Hence, the rhombic vehicle presents a maneuvering advantage over the car-like vehicle when the two wheels are turned in opposite directions. Remember, for instance, the illustrative case given in the left part of Figure 1.4. The relative reduction of the turning radius can be expressed as

$$\begin{aligned} |r'_{rhomba} - r'_{car}|/r'_{rhomba} &= \left|\left[\overline{PF} - \overline{RF}\right]/\tan\theta_F\right|/r'_{rhomba} \\ &= \left|\left[\overline{PF} - L\right]/\tan\theta_F\right|/r'_{rhomba} \\ &= \left|\left[\overline{RP}\cdot\tan\theta_R\right]/\tan\theta_F\right|/r'_{rhomba} \\ &= |r'_{rhomba}/\tan\theta_F|/r'_{rhomba} \\ &= tan\theta_R/\tan\theta_F \end{aligned} \tag{2.24}$$

2. Singularities and sensible drive configurations

The multitude of steering and driving configurations of the rhombic vehicle requires a strict coordination in order to achieve a sensible driving and prevent vehicle physical damage. For instance, the two configurations presented in Figure 2.6, are not sensible and will ultimately result in sideways dragging of one of the wheels or even maybe structural damage of the vehicle (components). It is up to the guidance system (recall Figure 1.5) to ensure that the mechanical constraints are satisfied and these singularities are replaced by sensible steering and velocity inputs.

Figure 2.6: Certain drive configurations must be avoided in order to prevent vehicle damage.

Other singularities may appear when the drive configuration brings the ICR of the vehicle to the center of a steering wheel. When this condition is verified, one of the wheels may skid the other one, violating the RWS condition, as it can be observed in Figure 2.7-top. Notice, however, that the two configurations presented in Figure 2.7-bottom, shall not be considered as a singularity, since the driving wheel forces the other wheel to rotate in place and wheel slippage does not occur.

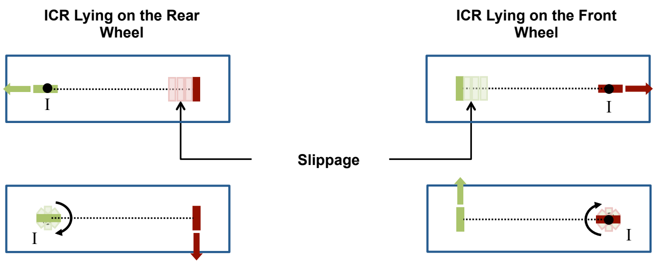


Figure 2.7: Singular drive configurations: insensible drive configurations forces one of the wheels to drag (top); The singular drive configurations allow the vehicle to rotate in place without causing slippage on wheels (bottom).

2.3 Dynamic Modeling

The kinematic models (2.18) and (2.19) presented in the previous section, capture the essential of the rhombic vehicle kinematics and are well suited for control and simulation purposes. However, the non-slipping assumptions in which the derivation of the kinematic model was based, will hardly be satisfied in reality, specially when tire deformation occurs. Consequently, every time the vehicle experiences slippage phenomena, for instance, in braking and acceleration phases (Figure 2.8-a) or in turning maneuvers (Figure 2.8-b), the kinematic model will not be able to correctly predict the system behavior providing a low simulation accuracy.

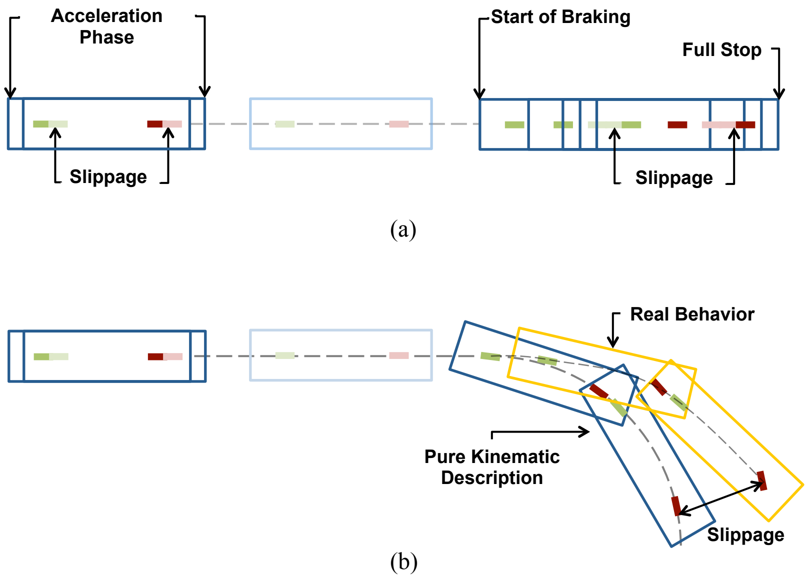


Figure 2.8: Slippage phenomena: (a) on accelerating and braking phases and, (b) on turning maneuvers, where significant wheel slippage may occur.

Most part of simulation and motion control related studies use a simple kinematic model to describe the movable system motion. The argument is usually based on the low speeds, low accelerations and the lightly loaded conditions under which these vehicles operate. In fact, the rhombic vehicle will operate under considerably low velocities, not exceeding the $0,2\ m \cdot s^{-1}$ but it will have to perform heavy duty work, which may significantly increase tire deformation and consequently slippage phenomena. In these conditions, control strategies developed based on the RWS kinematic models such as the one presented in (2.18), will hardly meet the requirements for the trajectory tracking task, encountering serious difficulties to maintain the vehicle over the reference trajectory. Also, the violation of the RWS constraints may have serious consequences on the accuracy of the localization system[1].

The remote operations related to the rhombic vehicle operation are issued with the typical demanding requirements of nuclear scenarios, which require high performances and accuracy. All these these facts encourage the author of the dissertation to assess a dynamic modeling approach, where wheel an vehicle slippage are taken into consideration as to obtain a more accurate simulation model of the rhombic vehicle.

Once the RWS condition is no longer respected, it is common to consider the complete or particular dynamics of the vehicle. For instance, in [OMN96], a focus was given to the vertical motion of the vehicle with the purpose of conceiving a suspension system. Another area of interest for automotive manufacturers, is the longitudinal control of the vehicle, with several systems already implemented in nowadays cars such as, the cruise control, anti-lock brake and traction control systems. For such type of applications the longitudinal description of the vehicle motion is commonly used. Despite these studies, great part of the literature on vehicle dynamics is devoted to the study and analysis of the vehicle motion

[1] In this case, sensory fusion often appears as a solution to attain the high accuracy mobility requirements, combining exteroceptive based localization systems with inertial sensors such as gyroscopes and accelerometers.

as a whole.

In general, vehicle dynamic approaches, resort on the classic mechanics of the rigid body, namely using the Newton-Euler or Lagrange equations, in order to develop more realistic models. Using these equations it is possible to derive the relations between the applied forces on the vehicle and the produced accelerations and trajectory motions.

2.3.1 Pneumatic Modeling

The knowledge of the forces and moments generated at the tire-ground contact area is essential for the completeness of any vehicle dynamic study. In fact, the vehicle motion is a consequence of the emergence of lateral and longitudinal forces on the vehicle tires. This type of phenomena comprises the scope of pneumatic modeling. For a comprehensive understanding of this subject, the reader is referred to [Pac06], a complete book about tire mechanics and modeling.

The interaction between a tire and the ground is characterized accordingly to the deformations of these bodies in the contact area, usually referred to as contact patch area. According to the type of deformation, four different cases can be established, with different targeted applications, as shown in Figure 2.9:

1. Flexible tire on a deformable surface – allowing the study of the motion of off-road vehicles, for instance on loose soil terrains;
2. Rigid tire on a deformable surface – adequate for studying the high pressure tire performances on a deformable terrain;
3. Rigid tire on a hard surface – mostly applied on railways vehicles;
4. Flexible tire on a hard surface – mostly applied on studies addressing the motion of ground vehicles on hard surfaces.

The latter case is assumed in the dissertation.

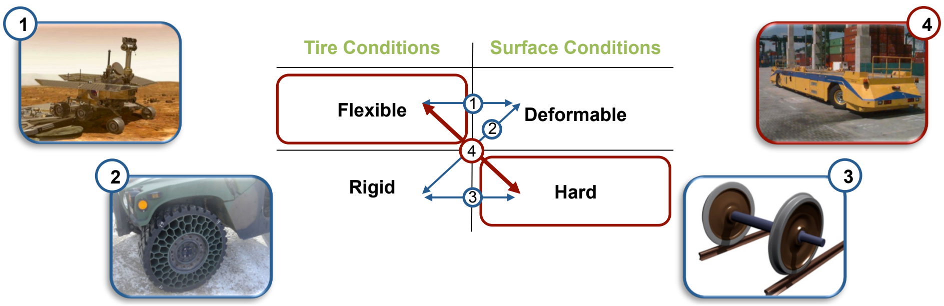


Figure 2.9: Tire and surface conditions originate different types of tire-ground studies.

For the case of a drivable and steerable wheels such as those found in the rhombic vehicle, tire-ground interactions originate the appearance of different moments and forces, which are illustrated in Figure 2.10 and listed in Table 2.1:

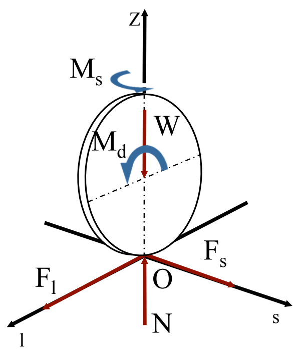


Figure 2.10: Applied forces and moments on a drivable and steerable wheel of the rhombic vehicle.

Table 2.1: Description of the forces applied on the vehicle wheel.

Parameter	Description
M_d	Driving torque
M_s	Steering/aligning torque
W	Weight force applied on wheel
F_l	Longitudinal tire force
F_s	Lateral/side tire force
N	Normal reaction of the ground

Several approaches have been proposed to extract these efforts resulting in different dynamic contact models. However, two main references shall be referred here: the well known Pacejka's model [BNP87], which is based on an empiric approach and the LuGre's model [CL97], this one, based on an analytic approach. For a survey on the different pneumatic models it is recommended [SCM01]. Despite the diversity of models, the majority of the approaches use the same variables to deduce these efforts and a special attention is given to the definition of the longitudinal F_l and lateral or side F_s forces acting on the wheel as function of slippage phenomena that occurs on the the wheel.

In what concerns wheel slip motion, there are two important phenomena to retain:

- **Lateral slip** – As illustrated in Figure 2.11-a, when the wheel negotiates turn, the lateral force F_s generated by the interaction between the tire and the ground, causes the wheel to transverse along a direction away from the wheel's plane. The angle between the wheel's direction and the wheel's movement plane is known as slip angle (α) and measures the wheel lateral slip.
- **Longitudinal slip** – Besides the lateral slip, the tire deformation (due to W) shown in Figure 2.11-b, causes the wheel to slippage. Under the PR assumption, the wheel's linear velocity is given by

$$v_l = r \cdot \varphi.$$

However, this is not the case of a deformed wheel, i.e, $v_l \neq r \cdot \varphi$. Hence, the longitudinal slip, here denoted by Σ, is equal to

$$\Sigma = r \cdot \varphi - v_l, \tag{2.25}$$

which can also be characterized by the longitudinal slip ratio defined as

$$\sigma = \frac{r \cdot \varphi - v_l}{\max\left(|r \cdot \varphi|, |v_l|\right)} \quad with \quad \sigma \in [-1, 1]\,. \tag{2.26}$$

In the previous equation, $\sigma = 0$ indicates no wheel slippage whereas $|\sigma| = 1$ means complete slippage, i.e., the wheel is not moving linearly despite of its angular rotation ($\sigma = 1$) or the wheel is blocked ($\sigma = -1$). On an acceleration phase, the wheel angular velocity is increased and $r \cdot \varphi >> v_l$ yielding a positive longitudinal slip ratio($\sigma > 0$). When braking, $r \cdot \varphi << v_l$ the slip ratio assume negative values($\sigma < 0$).

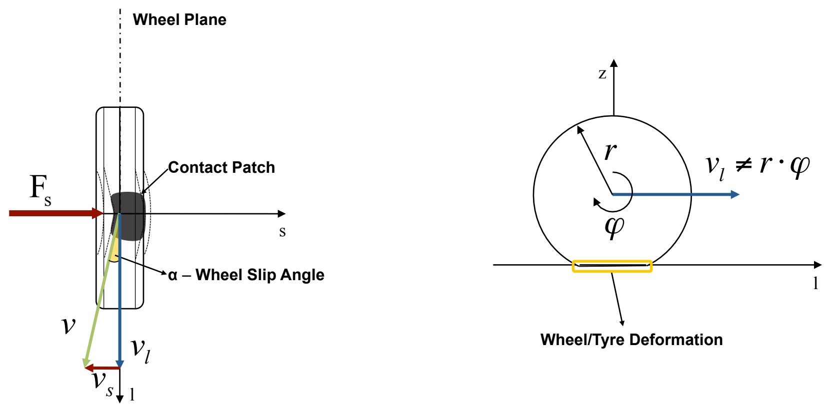


Figure 2.11: Tire deformation due to: the lateral force (left) and the wheel supported weight (right).

The Pacejka model, defined in [BNP87], also referred to as "The Magic Formula" (so named since the formulation has no underlying physical rationale) presents a description of the evolution of the lateral force F_s and the longitudinal force F_l, with the wheel slip angle α and the longitudinal slip ratio σ, respectively. However, the Pacejka's model requires the evaluation of numerous parameters related with pneumatic properties (dimensions, material, pressure), ground characteristics (humidity, nature of pavement) as well on the vehicle configuration such as, the weight distribution which influences the vertical wheel forces (W or N), which are unsettled in this premature design phase of the rhombic vehicle.

The dissertation follows a simpler approach, which is commonly adopted to simplify the dynamic model. Figure 2.12 illustrates the evolution of the longitudinal and lateral force with the lateral and longitudinal slip, respectively. A careful analysis of this data shows the existence of a pseudo-linear relation between these pair of variables. Therefore, for small values of lateral slip and longitudinal slip, the following simplified linear relation is valid [BNP87]:

$$\begin{cases} F_l = C_l \cdot \sigma \\ F_s = C_s \cdot \alpha \end{cases} \quad with \quad C_s, C_l > 0, \tag{2.27}$$

where the constants C_l and C_s are the longitudinal and cornering stiffnesses, respectively. These two constants are defined as the slope at vanishing slip values and are determining parameters for the basic linear handling and stability behavior of ground vehicles. The values of C_l and C_s depend on the tire and the ground conditions, vehicle load and other properties. These two parameters should be determined beforehand or estimated on-line with appropriate estimation algorithms. Due to the limitations of this study, constant contact stiffnesses is assumed for the front and rear wheel. This assumption can be considered reasonable, since during the rhombic vehicle journeys dramatic changes on the ground and tire conditions are not expected.

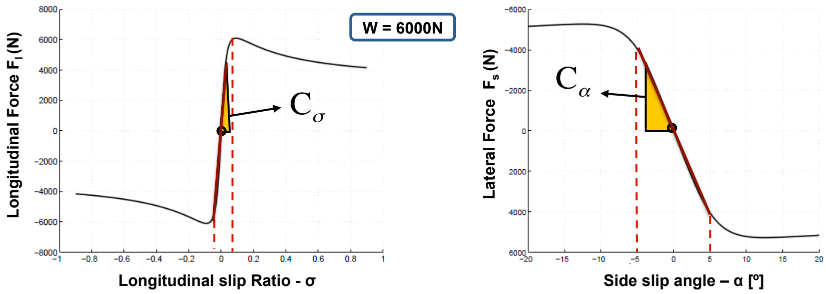


Figure 2.12: Evolution of the longitudinal force F_l and lateral force F_s with the longitudinal slip ratio σ and the slip angle α, respectively.

2.3.2 Dynamic Model of the Rhombic Vehicle

Once determine the interaction forces between the wheels and the ground the assessment of the global vehicle motion behavior can now proceed under a dynamic perspective. In addition to the previously (Section 2.1) presented assumptions, the derivation of the dynamic model of the rhombic vehicle is based on the following new or redefined hypotheses:

- The wheels are no longer considered as an undeformable solid; slippage phenomena is included on the dynamic analysis;
- The adhesion conditions on the rear and front wheels are considered to be equal;
- The CoG of the vehicle is located along its longitudinal axis and it is taken as reference point for the vehicle state vector;
- The vehicle suspension system is not considered. For this reason, the effects of rolling, pitching and load transfer are not included;
- Both longitudinal and lateral motions are considered to provide an accurate location of the vehicle;
- Counter-clockwise angles and torques are considered to be positive;
- The dynamic model is derived following the Newton-Euler formulation.

The vehicle dynamics with the forces acting on it is represented in 2.13. From now on, the index $i = \{F, R\}$ refers to the front and rear wheel, respectively. In the diagram of Figure 2.13, β_i denotes the side-slip angles of the wheels.

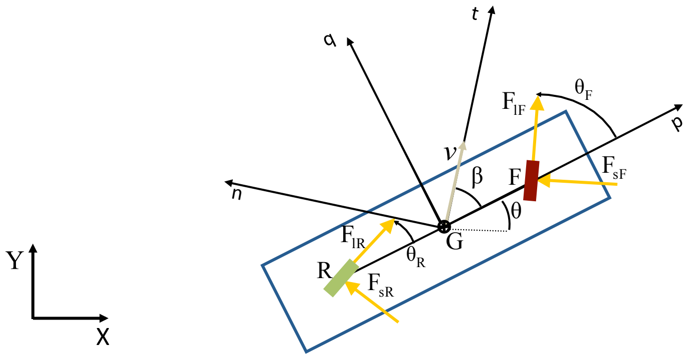


Figure 2.13: Dynamics diagram of the rhombic vehicle.

The derivation of the dynamic model is given in three different sets of coordinate frames (three different models), the global coordinate frame (X, Y), the local vehicle frame (q, p) and the vehicle velocity frame (n, t).

2.3.2.1 Dynamic Model in the Global Frame

In the global coordinate frame, the motion of the CoG of the vehicle along the X-axis is given by

$$F_x = m \cdot \ddot{x}_G. \tag{2.28}$$

Setting up the equilibrium of forces along the X-axis yields,

$$\begin{aligned} m \cdot \ddot{x}_G \quad &= \quad F_{lF} \cdot \cos(\theta + \theta_F) - F_{sF} \cdot \sin(\theta + \theta_F) \\ &\quad\quad + F_{lR} \cdot \cos(\theta + \theta_R) - F_{sR} \cdot \sin(\theta + \theta_R). \end{aligned} \tag{2.29}$$

Similarly, along the Y-axis the vehicle motion is described as

$$F_y = m \cdot \ddot{y}_G, \tag{2.30}$$

with the vertical forces summing up to

$$\begin{aligned} m \cdot \ddot{y}_G \quad &= \quad F_{lF} \cdot \sin(\theta + \theta_F) + F_{sF} \cdot \cos(\theta + \theta_F) \\ &\quad\quad + F_{lR} \cdot \sin(\theta + \theta_R) + F_{sR} \cdot \cos(\theta + \theta_R). \end{aligned} \tag{2.31}$$

Setting up the equilibrium of momentum about the vertical axis ate the CoG of the vehicle chassis, we obtain the following

$$\begin{aligned} I_z\ddot{\theta} \quad &= \quad F_{lF} \cdot \sin\theta_F \cdot L_F + F_{sF} \cdot \cos\theta_F \cdot L_F \\ &\quad\quad - F_{lR} \cdot \sin\theta_R \cdot L_R - F_{sR} \cdot \cos\theta_R \cdot L_R. \end{aligned} \tag{2.32}$$

The equations (2.29), (2.31) and (2.32), can be assembled on a matrix form as follows.

$$\begin{bmatrix} \ddot{x}_G \\ \ddot{y}_G \\ \ddot{\theta} \end{bmatrix} = \begin{bmatrix} \frac{\cos(\theta+\theta_F)}{m} & \frac{\cos(\theta+\theta_R)}{m} \\ \frac{\sin(\theta+\theta_F)}{m} & \frac{\sin(\theta+\theta_R)}{m} \\ \frac{\sin\theta_F \cdot L_F}{I_z} & -\frac{\sin\theta_R \cdot L_R}{I_z} \end{bmatrix} \cdot \begin{bmatrix} F_{lF} \\ F_{lR} \end{bmatrix} + \begin{bmatrix} -\frac{\sin(\theta+\theta_F)}{m} & -\frac{\sin(\theta+\theta_R)}{m} \\ \frac{\cos(\theta+\theta_F)}{m} & \frac{\cos(\theta+\theta_R)}{m} \\ \frac{\cos\theta_F \cdot L_F}{I_z} & -\frac{\cos\theta_R \cdot L_R}{I_z} \end{bmatrix} \cdot \begin{bmatrix} F_{sF} \\ F_{sR} \end{bmatrix} \tag{2.33}$$

2.3.2.2 Dynamic Model in the Local Frame

For the vehicle coordinate frame (p,q), attached at the CoG of the vehicle, the following expressions can be established:

$$m \cdot \ddot{p} = F_x \cdot \cos\theta + F_y \cdot \sin\theta; \tag{2.34}$$

$$m \cdot \ddot{q} = -F_x \cdot \sin\theta + F_y \cdot \cos\theta. \tag{2.35}$$

Using (2.29) and (2.31), equations (2.34) and (2.35) can be rewritten as

$$m \cdot \ddot{q} \;\; = \;\; F_{lF} \cdot \cos\theta_F - F_{sF} \cdot \sin\theta_F + F_{lR} \cdot \cos\theta_R - F_{sR} \cdot \sin\theta_R \tag{2.36}$$

and

$$m \cdot \ddot{p} \;\; = \;\; F_{lF} \cdot \sin\theta_F + F_{sF} \cdot \cos\theta_F + F_{lR} \cdot \sin\theta_R + F_{sR} \cdot \cos\theta_R \tag{2.37}$$

Equations (2.36) and (2.37) together with (2.32) comprise the vehicle dynamics on the local vehicle coordinate frame. The same model can be represented in a more convenient way as follows.

$$\begin{bmatrix} \ddot{q} \\ \ddot{p} \\ \ddot{\theta} \end{bmatrix} = \begin{bmatrix} \frac{\cos\theta_F}{m} & \frac{\cos\theta_R}{m} \\ \frac{\sin\theta_F}{m} & \frac{\sin\theta_R}{m} \\ \frac{\sin\theta_F \cdot L_F}{I_z} & -\frac{\sin\theta_R \cdot L_R}{I_z} \end{bmatrix} \cdot \begin{bmatrix} F_{lF} \\ F_{lR} \end{bmatrix} + \begin{bmatrix} -\frac{\sin\theta_F}{m} & -\frac{\sin\theta_R}{m} \\ \frac{\cos\theta_F}{m} & \frac{\cos\theta_R}{m} \\ \frac{\cos\theta_F \cdot L_F}{I_z} & -\frac{\cos\theta_R \cdot L_R}{I_z} \end{bmatrix} \cdot \begin{bmatrix} F_{sF} \\ F_{sR} \end{bmatrix} \tag{2.38}$$

2.3.2.3 Dynamic Model in the Velocity Frame

Sometimes it is useful to derive the dynamic model of the vehicle in the velocity coordinate frame (n,t) which in terms of coordinates can be defined as (v, β, λ). To develop this last dynamic model, the same reasoning developed to achieve (2.33) and (2.38) is followed here, now considering the normal and tangential directions of the velocity vehicle frame (n,t). Note that the vehicle motion along the axes defining this frame, can be expressed as[2]

$$F_n = m \cdot \ddot{n} = m \cdot \frac{v^2}{r'} = m \cdot v \cdot (\dot{\beta} + \lambda), \tag{2.39}$$

and

$$F_t = m \cdot \ddot{t} = m \cdot v. \tag{2.40}$$

Considering this, the equilibrium of forces along the n-axis and the t-axis yields,

$$\begin{aligned}\dot{v} &= \tfrac{1}{m} \cdot [F_{lF} \cdot \cos(\theta_F - \beta) - F_{sF} \cdot \sin(\theta_F - \beta) \\ &\quad + F_{lR} \cdot \cos(\beta - \theta_R) + F_{sR} \cdot \sin(\beta - \theta_R)]\end{aligned} \tag{2.41}$$

and

$$\begin{aligned}\dot{\beta} &= \tfrac{1}{m \cdot v} \cdot [F_{lF} \cdot \sin(\theta_F - \beta) + F_{sF} \cdot \cos(\theta_F - \beta) \\ &\quad - F_{lR} \cdot \sin(\beta - \theta_R) + F_{sR} \cdot \cos(\beta - \theta_R)] - \lambda.\end{aligned} \tag{2.42}$$

Noting the correspondency between $\ddot{\theta}$ and $\dot{\lambda}$, Equation (2.35) leads to

$$\begin{aligned}\dot{\lambda} &= \tfrac{1}{I_z} \cdot [F_{lF} \cdot \sin\theta_F \cdot L_F + F_{sF} \cdot \cos\theta_F \cdot L_F \\ &\quad - F_{lR} \cdot \sin\theta_R \cdot L_R - F_{sR} \cdot \cos\theta_R \cdot L_R].\end{aligned} \tag{2.43}$$

Equations (2.41)-(2.43) form the vehicle dynamics in the coordinate frame (n,t). Rearranging terms, the following matrix representation can be obtained:

$$\begin{aligned}\begin{bmatrix} \dot{v} \\ \dot{\beta} \\ \dot{\lambda} \end{bmatrix} &= \begin{bmatrix} \frac{\cos(\theta_F - \beta)}{m} & \frac{\cos(\beta - \theta_R)}{m} \\ \frac{\sin(\theta_F - \beta)}{m \cdot v} & -\frac{\sin(\beta - \theta_R)}{m \cdot v} \\ \frac{\sin\theta_F \cdot L_F}{I_z} & -\frac{\sin\theta_R \cdot L_R}{I_z} \end{bmatrix} \cdot \begin{bmatrix} F_{lF} \\ F_{lR} \end{bmatrix} \\ &+ \begin{bmatrix} -\frac{\sin(\theta_F - \beta)}{m} & \frac{\sin(\beta - \theta_R)}{m} \\ \frac{\cos(\theta_F - \beta)}{m \cdot v} & \frac{\cos(\beta - \theta_R)}{m \cdot v} \\ \frac{\cos\theta_F \cdot L_F}{I_z} & -\frac{\cos\theta_R \cdot L_R}{I_z} \end{bmatrix} \cdot \begin{bmatrix} F_{sF} \\ F_{sR} \end{bmatrix} - \lambda \cdot \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}\end{aligned} \tag{2.44}$$

2.3.2.4 Implementation Details

One of the issues that remains to explain is how to perform the evaluation of the longitudinal slip ratio and slip angle for the specific simulation requirements of the rhombic vehicle. As discussed in Subsection 2.3.1, slippage phenomena causes the wheels to roll with linear velocities that are usually in directions which slightly deviate from the longitudinal axes of the wheels, giving rise to the side-slip angles, α_F and α_R. These effective velocities, here in denoted as, $\boldsymbol{v_F^{eff}}$ and $\boldsymbol{v_R^{eff}}$, for the front and rear wheel, respectively, and the wheel slip angles, are depicted in the auxiliary kinematic diagram illustrated in Figure 2.14.

The relation between the wheel slip and steering angles and side-slip of the front an rear chassis (β_F, β_R) is promptly extracted using the pure geometric relations as follows.

[2]This term corresponds to the centripetal force. By differentiating the relation of the arc length $s = r' \cdot (\theta + \beta)$ with respect to time, yields $v = r' \cdot (\dot{\theta} + \dot{\beta})$. Remember that the angle $(\theta + \beta)$ corresponds to the angle between the vehicle velocity, v, at the CoG and the global fixed coordinate system (X, Y).

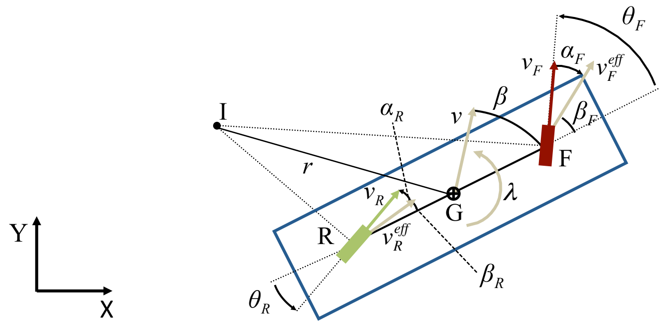


Figure 2.14: Rhombic vehicle kinematics diagram on a turning maneuver and considering slippage phenomena.

$$\alpha_i = \theta_i - \beta_i, \tag{2.45}$$

where,

$$\begin{aligned} \beta_F &= \arctan\left[\tfrac{v+M_F\cdot\dot{\theta}}{u}\right] \\ \beta_R &= \arctan\left[\tfrac{v-M_R\cdot\dot{\theta}}{u}\right] \end{aligned} \tag{2.46}$$

The longitudinal slip ratio for the front and rear wheel is evaluated with the following equation

$$\sigma_i = \frac{v_i - v_i^{eff}\cdot cos\alpha_i}{\max(|v_i|, |v_i^{eff}\cdot cos\alpha_i|)}. \tag{2.47}$$

Should be noted that Equation (2.47) slightly differs from the one presented in (2.26). According to the system definition presented in Section 2.1, the aim is to simulate the vehicle motion through the input of the linear speeds on the wheels (v_F, v_R), rather than using the wheel angular speed. Hence, in (2.47) the term v_i refers to the inputed wheel linear speed, whereas v_i^{eff} is the effective rolling wheel speed. To evaluate the longitudinal wheel slip ration, only the longitudinal component of the wheel effective velocity ($v_i^{eff}\cdot cos\alpha_i$) is taken into account. The modulus of the effective wheel velocity can be evaluated through the following expressions,

$$v_F^{eff} = \sqrt{u^2 + (v + M_F\cdot\dot{\theta})^2} \tag{2.48a}$$

$$v_R^{eff} = \sqrt{u^2 + (v - M_R\cdot\dot{\theta})^2} \tag{2.48b}$$

for the front and rear wheel, respectively.

In (2.46) and (2.48), u and v refer to the longitudinal and lateral vehicle velocity, respectively, and are accessed through the dynamic model on the vehicle frame. If other frame is chosen, the following correspondency shall be considered,

$$u = \dot{x}_G\cdot\cos\theta + \dot{y}_G\cdot\sin\theta = v\cdot\cos\beta \tag{2.49a}$$

$$v = -\dot{x}_G\cdot\sin\theta + \dot{y}_G\cdot\cos\theta = v\cdot\sin\beta. \tag{2.49b}$$

For the evaluation of the longitudinal and lateral forces recall the simplified linear assumption to approximate the nonlinear tire characteristics (2.27). Considering equal cornering stiffnesses (C_s) and longitudinal stiffness (C_l) for the front and the rear wheels the evaluation of these forces is given by

$$F_{si} = \begin{cases} C_{si} \cdot \alpha & if \quad C_{si} \cdot \alpha \leq \mu \cdot N_i \\ \mu \cdot N_i & otherwise \end{cases} \tag{2.50}$$

and

$$F_{li} = \begin{cases} C_{li} \cdot \sigma & if \quad C_{li} \cdot \sigma \leq \mu \cdot N_i \\ \mu \cdot N_i & otherwise \end{cases} . \tag{2.51}$$

F_s and F_l are clipped to the maximum allowable friction force given by the static coefficient of friction μ multiplied by the wheel normal force N_i.

2.4 Extended Kinematic Model

The dynamic model presented in the previous section provides a more truthful description of the rhombic vehicle behavior and shall be adopted in preference to the simpler kinematic one, whenever high accuracy simulations are needed. A shortcoming of this model is that it does not provide sufficient insight for control design due to its increased complexity. The motion control problem of the rhombic vehicle does not integrate the objectives of the dissertation. However, one of the stated purposes for this modeling stage was to follow an alternative modeling approach that could provide a trade-off between model accuracy and simplicity for possible incoming control oriented studies.

To handle this objective, the philosophy of the Extended Kinematic Models (EKMs) is explored in this section on an attempt to explicitly model vehicle slippage phenomena on a kinematic framework. It was shown in Section 2.3 that tire interaction forces are evaluated as function of two main parameters, the longitudinal slip ratio (σ) and the slip angle (α). In this section and by relying on simple mechanic considerations, this slippage phenomena is introduced as perturbations to the system on a kinematic basis.

In Section 2.2, the nonholonomic RWS assumption was used to derive the kinematic model, entailing that the linear velocity of the contact point of the wheel is perfectly aligned with the wheel plane. However, in situations where the slippage phenomena is not negligible, the direction of the real wheel velocity is no longer aligned with the wheel's plane (recall Figure 2.14). In order to take slippage phenomena the new EKM has to consider the real directions of the rear and front wheels velocities. Thus, the formulation presented in Section 2.2 will be extended here, for the creation of the EKM model, in which the nonholonomy is no longer respected in its strictly sense.

Recalling the equations in (2.9) and following the vehicle kinematic diagram in Figure 2.14, in the presence of wheel slippage, the system is subjected to the following two "nonholonomic constraints", one for each wheel

$$\dot{x}_F \cdot \sin(\theta + \theta_F + \alpha_F) - \dot{y}_F \cdot \cos(\theta + \theta_F + \alpha_R) = 0 \tag{2.52a}$$

and

$$\dot{x}_R \cdot \sin(\theta + \theta_R + \alpha_F) - \dot{y}_R \cdot \cos(\theta + \theta_R + \alpha_R) = 0. \tag{2.52b}$$

Applying the geometric constraints defined in (2.10), the equations in (2.52) can be organized in the following matrix form,

$$\begin{bmatrix} \sin(\theta+\theta_F+\alpha_F) & -\cos(\theta+\theta_F+\alpha_F) & -M_F\cdot\cos(\theta_F+\alpha_F) \\ \sin(\theta+\theta_R+\alpha_R) & -\cos(\theta+\theta_R+\alpha_R) & M_R\cdot\cos(\theta_R+\alpha_R) \end{bmatrix}\cdot\dot{\boldsymbol{q}} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \Rightarrow H(\boldsymbol{q})\cdot\dot{\boldsymbol{q}} = 0. \quad (2.53)$$

The null space of the constraint matrix H, $Ker(H(\boldsymbol{q}))$, contains all the admissible values for $\dot{\boldsymbol{q}} = [\dot{x}_G \quad \dot{y}_G \quad \dot{\theta}]^T$. Performing the same steps as in Appendix C for the constraint matrix C, the $Ker(H(\boldsymbol{q}))$ can be represented in the following matrix form

$$\begin{bmatrix} \dot{x}_G \\ \dot{y}_G \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \frac{M_R\cdot\cos(\theta+\theta_F+\alpha_F)}{M} \\ \frac{M_R\cdot\sin(\theta+\theta_F+\alpha_F)}{M} \\ \frac{\sin(\theta_F+\alpha_F)}{M} \end{bmatrix}\cdot\ v_F^{eff} + \begin{bmatrix} \frac{M_F\cdot\cos(\theta+\theta_R+\alpha_R)}{M} \\ \frac{M_F\cdot\sin(\theta+\theta_R+\alpha_R)}{M} \\ \frac{-\sin(\theta_R+\alpha_R)}{M} \end{bmatrix}\cdot\ v_R^{eff}, \quad (2.54)$$

Redefining the longitudinal slip in (2.25) to

$$\Sigma_i = v_i - v_i^{eff}\cos\alpha_i, \quad (2.55)$$

the EKM model in (2.54) can be rewritten as

$$\begin{bmatrix} \dot{x}_G \\ \dot{y}_G \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \frac{M_R\cdot\cos(\theta+\theta_F+\delta_5)}{M} \\ \frac{M_R\cdot\sin(\theta+\theta_F+\delta_5)}{M} \\ \frac{\sin(\theta_F+\delta_5)}{M} \end{bmatrix}\cdot\ (v_F\delta_1-\delta_2) + \begin{bmatrix} \frac{M_F\cdot\cos(\theta+\theta_R+\delta_6)}{M} \\ \frac{\sin(M_F\cdot\theta+\theta_R+\delta_6)}{M} \\ \frac{-\sin(\theta_R+\delta_6)}{M} \end{bmatrix}\cdot(v_R\delta_3-\delta_4), \quad (2.56)$$

where $\{\delta_1,\delta_2,\delta_3,\delta_4,\delta_5,\delta_6\}$ are model disturbance perturbations defined as

$$\delta_1 = \tfrac{1}{\cos\delta_5}, \quad \delta_2 = \tfrac{\Sigma_F}{\cos\delta_6}, \quad \delta_3 = \tfrac{1}{\cos\delta_6}, \quad \delta_4 = \tfrac{\Sigma_R}{\cos\delta_6}, \quad \delta_5 = \alpha_F, \quad \delta_6 = \alpha_R. \quad (2.57)$$

This EKM model provides the means to simulate the vehicle motion considering slippage phenomena, whenever slippage disturbances, i.e., α_i and Σ_i are accessible providing a good trade-off between model accuracy and simplicity.

Chapter 3

Motion Planning – An Overview

During the last decades, motion planning has emerged as an important and productive research topic in different research communities. The meaning of the term “planning” may vary depending on the field where it is applied. For instance, in Artificial Intelligence (AI), planning techniques are often used to solve discrete and continuous problems (e.g. puzzles and brain teasers), or within the control communities, where motion planning techniques arouse interest in the design of algorithms to find open loop strategies for nonlinear systems.

Also in robotics, the motion planning is a fundamental issue, in particular within the WMRs field, in which the dissertation is inserted. Many WMRs, such as the rhombic vehicle, are designed to move autonomously around its environment. To perform this task, the WMR needs to know how to move in its environment in order to achieve one or multiple specified goal locations without colliding with the obstacles. As shown in Section 3.1 this is exactly the problem that motion planning techniques try to solve.

The history of motion planning is quite recent. The first known published work addressing a motion planning technique appears in the AI field, in 1969 with [Nil69]. The Visibility Graphs were introduced later [Loz78], on a later pursuit of this work. The notion of configuration space, which is explored in Subsection 3.1.1, was introduced in the beginning of the 80s [Loz83], providing an unified geometrical formulation for the development of motion planning algorithms. Still in the 80’s, the motion planning is first formulated in the robotics field, appearing related to a benchmark problem known as *“The Piano Mover’s Problem”* [SS83, SS84], which was concerned with the following task: how to move a piano from on room to another without hitting the obstacles. Due to its challenging nature, this problem attracted the attention of many scientists over the next years. Hence, up until 1990 a large number of planning techniques have been proposed, some of them providing exact solutions [Can88, Lat91]. Latombe’s book [Lat91], constitutes an excellent survey on the progress and developed approaches during the mentioned period.

For the most part of the problems, the suitability of exact algorithms is limited due to its computational complexity. To overcome this inconvenience, several approaches have been proposed, relying mostly on approximate cell decompositions and on heuristic approaches (refer to Chapters 6 and 7 in [Lat91]).

Randomized approaches appear later in the 90's with more practical planners [BL91, KSLO96, Lav98]. Most of the recent work has been focused on the study of the main properties of these randomized techniques [KKL98, SO98] with several applications developed to solve difficult problems [HKL$^+$98, ADS02]. A complete survey on randomized algorithms can be found in [MR95]. Section 3.3 addresses some of these techniques while, in Chapter 5 a focus is given to a particular randomized planner, the RRT [Lav98].

Similar works to the Latombe's book, surveying the techniques and topics on motion planning, can be found in [Lau98, Lav03]. For a thorough overview on more recent research on mobile robots please refer to [Jing08].

Motion planning problems exist in abundance and as pointed out in [Lav03], "these are exciting times to study planning algorithms and contribute to their development and use". For some motivational examples refer to Appendix D.

3.1 Problem Formulation

Despite the broad class of motion planning problems, they are closely connected at the beginning and can be uniformly formalized. In the following, some basic ingredients arising in nearly every type of motion planning problem:

- **Agent** – It refers to different things depending on the application; The agent can be a robot, a decision maker, the controller, the digital actor, the molecule or even a person if applying a planned investment strategy.
- **Configuration** – Planning involves searching in a configuration space that captures all the possibilities that can arise. The configuration usually provides specific information about the agent (e.g., robot pose (position + orientation) or the location of a tile in a puzzle).
- **Initial and goal states** – A motion planning problem involves moving from a specified initial configuration, to a desired goal configuration, the "frontier conditions" that form the motion query to planner.
- **Geometric model of the World** –To solve the motion planning problem, the world $\mathcal{W}$ in which the plan is applied, must be classified and modeled. Typically motion planning problems are solved in 2D or Three Dimensional (3D) worlds, i.e, $\mathcal{W} = \mathbb{R}^2$ or $\mathcal{W} = \mathbb{R}^3$, respectively.
- **Plan** – Comprises the solution of the motion planning problem. It usually specifies a sequence of actions, a strategy to be carried out that brings the agent from the initial configuration to the goal configuration (e.g, open loop control law, a path to be followed, a strategy of investment).
- **Optimization Criteria** – The plan shall comply with some requirements; At a first instance, the output plan should cause the arrival of the agent to the goal configuration. This main concern is related to the geometric feasibility of the solution found. Then at a second stage, it is possible to entail higher level requirements such as optimized performances during the execution of the plan.

The motion planning problem can be mathematically formalized for WMRs as follows. Consider a point-like WMR, $\mathcal{A}$, in an environment composed by a set of obstacles $\mathcal{O} = \{\mathcal{O}_1 ... \mathcal{O}_n\}$. Given the initial and desired goal configuration for $\mathcal{A}$ in the $\mathcal{W} = \mathbb{R}^2$, find a collision free path τ, specifying a sequence of intermediate configurations connecting the initial and goal configurations, or report failure if there is no solution.

This basic formulation corresponds to *"The Piano Mover's Problem"* [SS83, SS84] mentioned earlier and is represented in Figure 3.1.

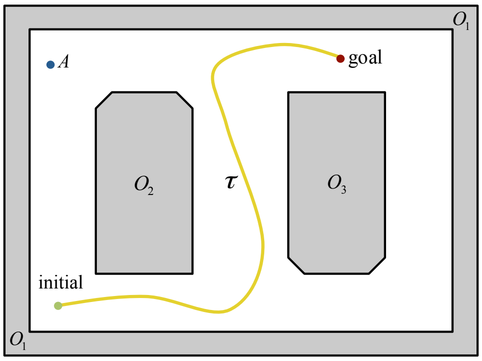


Figure 3.1: The basic motion planning problem for a point-like WMR (in blue). The path (in yellow) connects the initial (in green) to the goal (in red) configuration.

3.1.1 The Configuration Space

Let $\mathcal{C}$, denote the the configuration space [Loz83], which is the state space, in control terminology, arising in the motion planning.

Using $\mathcal{C}$-space notions, assume $\boldsymbol{q} \in \mathcal{C}$, as the configuration of $\mathcal{A}$ in $\mathcal{W}$. The path, τ, can thus be defined as a continuous map $\tau : [0,1] \rightarrow \mathcal{C}$, where $\tau(0) = \boldsymbol{q_I}$ and $\tau(1) = \boldsymbol{q_G}$, are the initial and goal configurations for $\mathcal{A}$, respectively. To be consider a viable solution, τ must be collision-free.

The collision-free requirement can be easily understood by introducing the notion of configuration-obstacle-space, $\mathcal{C}_{obst}$. $\mathcal{C}_{obst}$ basically represents a transcription of the obstacles $\mathcal{O} \subset \mathcal{W}$ for $\mathcal{A}$, such that, $\mathcal{C}_{obst} \subseteq \mathcal{C}$. A configuration $\boldsymbol{q}$ is said to be "*in colliding*‘" if it intersects at least one obstacle in $\mathcal{W}$. $\mathcal{C}_{obst}$ can thus be defined as all the colliding configurations in $\mathcal{C}$,

$$\mathcal{C}_{obst} = \{\boldsymbol{q} \in \mathcal{C} \; ; \; \mathcal{A}(\boldsymbol{q}) \cap \mathcal{O} \neq \emptyset\} \tag{3.1}$$

and the collision-free configuration space or "*free space*", simply defined as,

$$\mathcal{C}_{free} = \mathcal{C} \setminus C_{obst} \tag{3.2}$$

Considering the same $\mathcal{W}$ presented in Figure 3.1, Figure 3.2 conceptually illustrates the basic motion planning problem and the resulting topologies for the different *C-spaces*. As it can be seen, the topology of the $\mathcal{C}_{obst}$ changes if the geometry of $\mathcal{A}$ is considered. It also gives an idea about the complexity of an explicit representation of $\mathcal{C}_{obst}$ or $\mathcal{C}_{free}$ for WMRs with more DoFs (e.g, translation plus rotation).

It is important to know that some algorithms depend on the explicit representation of $\mathcal{C}_{obst}$, namely the exact or combinatorial methods presented in Subsection 3.2.1. Other planning techniques, as the

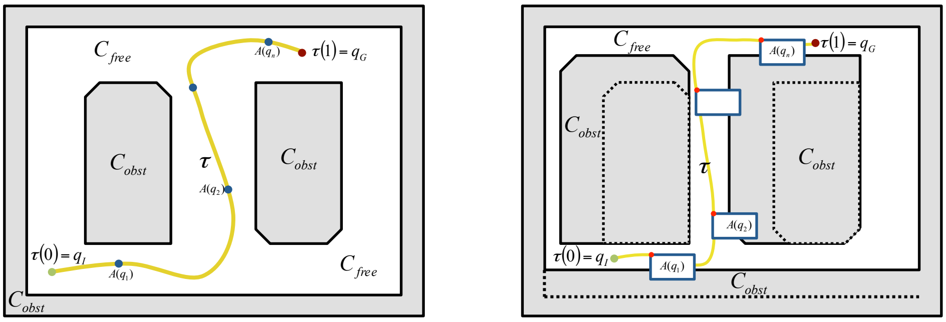


Figure 3.2: $\mathcal{C}$-space topologies: for a point-like WMR (left) and for a rigid WMR that can only translate (right).

sampling-based planning algorithms introduced in 3.2.2, evade the complexity of constructing $\mathcal{C}_{obst}$, by implicit representing the environment and the robot model resorting on collision detection modules. Nevertheless, $\mathcal{C}$-space notions are important. They provide a certain level of abstraction, allowing to use the same formulation for different problems and sometimes solve them with the same planning algorithms.

3.1.2 Planning Under Differential Constraints

In the above formulation, time dimension is only implicitly modeled by determining that the successive configurations have to be performed one after another. Moreover, the WMR is allowed to move freely in any direction aside from the $\mathcal{C}_{obts}$. This formulation is referred to as *holonomic motion planning*.

Besides the global constraints that arise from obstacles and restrict the set of allowable configurations, other constraints exist that that bound the allowable velocities or accelerations for the WMR at each point. These restrictions can be considered as local constraints and are usually referred to as differential constraints. Therefore, the motion planning problem has to be brought to an higher instance, where both, global and local constraints are considered. The motion planning problem correspond thus to a search in a feasible configuration space $\mathcal{C}_{feas}$ (i.e., the space of feasible velocities and/or accelerations) rather than in $\mathcal{C}_{free}$.

Within this higher instance of motion planning, two different problems can be presented:

- **Nonholonomic motion planning**

  The notion of *nonholonomic motion planning*, was introduce in 1986 by J-P. Laumond [Lau86], to describe the motion planning problem of WMRs involving non integrable constraints on the state velocities as the ones introduced in Subsection 2.2.1. The planning methods applied to nonholonomic systems are usually referred to as *steering methods* [Dub57, RS90, ST91, SL90].

- **Kinodynamic[1] motion planning**

  Beyond the kinematic restrictions sometimes it is necessary to consider dynamic constraints which restrict the robot motion in the accelerations space. This leaded to the emergence of kinodynamic

[1] In the motion planning literature, nonholonomic planning is mostly concerned with the kinematic (nonholonomic) constraints and kinodynamic planning with nonholonomic dynamic constraints. However, in nonlinear control terminology, kinodynamic planning is encompassed by the definition of nonholonomic planning.

motion planning, which consists on solving the motion planning while satisfying both kinematic and dynamic constraints associated with the WMR.

Planners addressing these two problems output plans in form of trajectories (instead of simple paths) that directly comply with the natural and feasible motions of the WMR. For this reason this planning instance is often referred to as trajectory planning, which is covered in Chapter 5. Notice the difference to the classic trajectory design, which has commonly been divided into two separated problems: (1) determine a collision free path, which ignores differential constraints; (2) apply a trajectory evaluator that parameterizes the path in terms of time or velocities and fulfills the WMR differential constraints. Chapter 4 closely follows this approach. For further details please refer to Section 3.3, where it is formulated the complete motion problem for the rhombic vehicle as well the proposed methodologies to solve it.

3.2 Motion Planning Algorithms

This section starts by reviewing some of the most used classical motion planning algorithms. The second part is dedicated to randomized motion planning algorithms, with focus on two approaches that have drawn more attention in the planning-algorithm community. For each algorithm, the basic concepts are explained together with illustrative examples.

3.2.1 Classical Motion Planning

These classical motion planning methods presented here are based on an explicit representation of the environment. Most part of low dimensional motion planning problems (e.g. the 2D problem presented in Section 3.1), can be solved using these approaches. Yet, the cost of an explicit representation of $\mathcal{C}_{obst}$ it is not always worthy. The use of the methods described below is not exclusive and can be combined in a single architecture such as in [MN04]. Three classical methods are presented here:

1- Cell Decomposition

The main idea behind the Cell Decomposition (CD) methods, is to divide or decompose $\mathcal{C}_{free}$, into a finite number of regions, called cells. CD algorithms are classified according to the nature of the CD yielded: (1) Exact Cell Decomposition; these methods generate an exact decomposition of $\mathcal{C}_{free}$ and can alternatively be referred to as complete algorithms, since their either find a solution or correctly report failure. Figure 3.3-left provides depicts a triangle CD. (2) Approximate approaches; the CD is based on a discretization of $\mathcal{C}_{free}$. These algorithms are referred to as resolution complete, i.e., the completeness of the solution is only guaranteed for a given resolution of the $\mathcal{C}_{free}$. If a solution exists, the algorithm will find it in finite time. However, if a solution does not exist, the algorithm may run forever. The resulting cells have a simple and predefined shape such as the quadratic shape used in Figure 3.3-right to exemplify an approximate CD.

As shown later in Chapter 4, where it is described in more detail triangle CD algorithm, a key element of the CD methods is the connectivity graph $\mathcal{G}$ of the cells, which allows to efficiently capture the structure

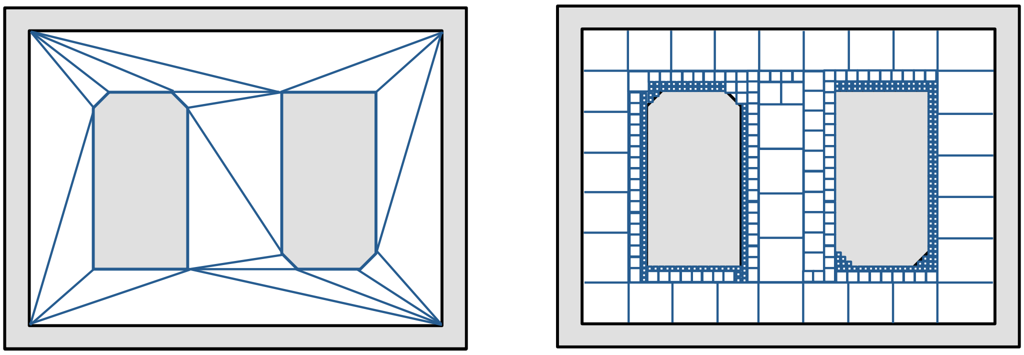


Figure 3.3: Left: exact CD using triangle cells. Right: approximate CD.

of $\mathcal{C}_{free}$ and solve the motion query as a graph search problem.

2 – Roadmap methods

Unlike the CD methods that try to reach the $\mathcal{C}_{free}$ shape by decomposing it into a group of cells, the roadmap based methods, as the name suggests, try to reach the connectivity $\mathcal{C}_{free}$ by creating a roadmap, i.e., topological graphs. Once the roadmap is defined, the query motion is solved as a graph search problem. The existing approaches differ in the way this roadmap is constructed, which is usually linked to a specific criterion that shall be optimized. For instance, the shortest-path roadmap can be evaluated by means of the Reduced Visibility Graph [Lat91], which is an improved version of the Visibility Graphs introduced in 1978 by Lozano [Loz78]. In Figure 3.4-left, the Reduced Visibility Graph returns the yellow shortest roadmap by connecting the obstacle tangent edges and the edges that are formed by vertices that are mutually visible from each other (and also tangent to obstacles). The blue roadmap corresponds to the one returned by the seminal Visibility Graph method. The motion query problem is then solved by connecting any pair of objective configurations to the closest configuration in the roadmap.

The solution obtained with the Reduced Visibility Graph is optimal in the sense that it guarantees the minimum distance between the start and goal configuration. However, it also presents the minimum clearance possible over the obstacles, which is undesirable for the most part of robot applications. Instead of generating the shortest path it might be convenient to compute a path that maximizes clearance from obstacles. This through the use of a different type of roadmap method which is usually called of Generalized Voronoi Diagram or Retraction Method [Can85, DY82]. This method returns a roadmap that is kept as far as possible from $\mathcal{C}_{obst}$, as illustrated in Figure 3.4-right. Note that the set of collision-free configurations composing this roadmap are obtained by considering a point-like robot approach. What if the solution path was followed by a real rigid body? Would it still continue to be a maximum clearance path? This issue is revisited later in Chapter 4.

3 – Potential fields

The potential fields approach [Kha85] is one of the most popular approaches for path planning and also as collision avoidance method. The concept behind the potential fields approach is a little different from the approaches that have been describe so far. In this technique, $\mathcal{C}$ is modeled as a potential function and the robot is taken as a point moving through the potential descending gradient. Most formulations use an attractive term causing the movable system to approach the goal and a repulsive term to penalize

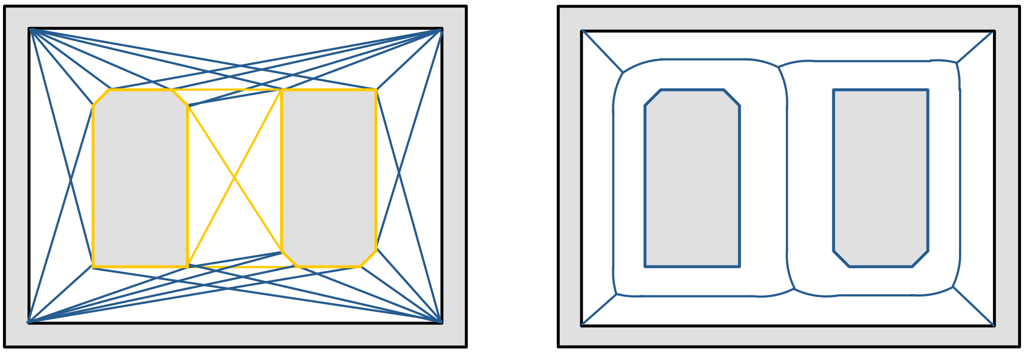


Figure 3.4: Left: in blue the roadmap obtained with the Visibility Graph and in yellow the roadmap returned by the Reduced Visibility Graph. Right: the generalized Voronoi diagram yields a maximum clearance roadmap.

configurations closer to obstacles, as depicted in Figure 3.5. The principal inconvenient of this method is to get trapped on a local minimum. This can be avoided by using harmonic functions that do not allow local minima or simply performing a random movement to leave the local minimum.

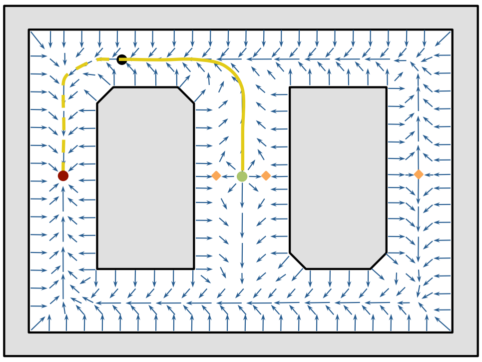


Figure 3.5: The potential field, repeals the point-like robot (in black) and conducts it from the initial configuration (in green) to the attracting goal configuration (in red). Possible local minimums (yellow diamonds).

3.2.2 Randomized Motion Planning

The randomized approaches start to appear in the 90's with [BL91] and gone through interesting developments during the next years [KSLO96, MR95, Lav98, HKL$^+$98, ADS02, Lav03]. In fact, randomized methods have proven to be an option that deserves to be considered when tackling challenging motion planning problems. These methods are capable of handling high-dimensional problems in acceptable algorithmic-execution times and tackle important problem constraints aside from the geometric ones arising from obstacles. Even so, these approaches are based on a weaker notion of completeness, which, unlike the combinatorial planners, do not guarantee that the motion problem will be solved. They are usually considered to be probabilistically complete, i.e., theoretically, the probability of finding a solution converges to 1 when the algorithm is given a infinite time.

An important characteristic of these methods is that they are based on a sampling scheme that avoids the explicit representation of $\mathcal{C}$, reason why they are alternatively referred to as sampled-based planners. As no explicit representation of $\mathcal{C}$ exists, randomized planners make use of a binary collision checker to test whether a specified configuration $\boldsymbol{q}$ is feasible (collision-free) or unfeasible (i.e., colliding with obstacles).

The two methods that have attracted more attention and have been more successful during the last years are the Probabilistic Roadmap Methods (PRMs) [KSLO96] and the RRTs [Lav98], are briefly described in the following.

1 – Probabilistic Roadmap Method

Despite being a roadmap-based method the PRM is completely different from the combinatorial roadmaps referred in Subsection 3.2.1. The PRM method is composed of two stages:

- **Learning phase:** the roadmap is constructed by randomly sampling $\mathcal{C}$ as shown in Figure 3.6-left. The roadmap is then constructed by connecting collision-free neighbor vertices, i.e., vertices laying in $\mathcal{C}_{free}$ that can be connected by a continuous path in $\mathcal{C}_{free}$ (this is performed by a local planner);
- **Query phase:** the initial and goal configuration are first connected to the roadmap. The motion query problem is then solved by searching an admissible solution as illustrated in Figure 3.6-right. When the aim is to choose the shortest-path, this tasks can be carried out using AI graph search methods, such as the Dijkstra [Dij59] or its improved heuristic version A* [HNR68].

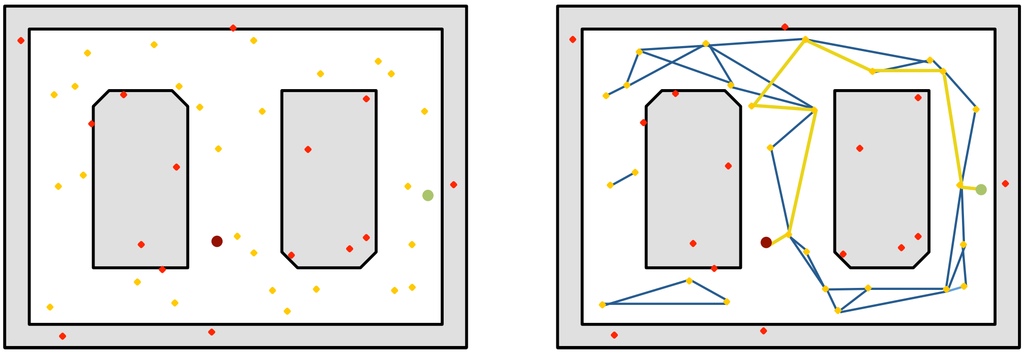


Figure 3.6: The PRM: collision free samples in yellow-$\diamond$, colliding samples in red-$\diamond$ (left) and obtained roadmap in blue (right) during the learning phase. In the query phase, the initial and goal configurations are connected via the yellow shortest-path (right).

For static environments the obtained roadmap can be reused to process other motion queries. This is the reason why PRM is sometimes classified as multiple motion query method.

The process described above comprise the basic functioning of PRM but there are several extensions of this method. For instance in [BK00] a single query planner is proposed entitled "Lazy PRM". In this planner a roadmap is built in the entire $\mathcal{C}$ and not only in the free space $\mathcal{C}_{free}$. The idea behind this "lazy" planner is to reduced the number of collision checks needed in the learning phase, by postponing this collision avoidance feature to the query phase when the initial roadmap is already evaluated.

One problem may arise when the solution path must pass through narrow corridors or passages. As it can be imagined, the probability of a sample fall in this region is very low and the connectivity of the roadmap in $\mathcal{C}_{free}$ is hard to obtain. To cope with this problem, several extensions have been proposed such as [HKL+98] or more recently in [MRA03].

An important issue when using the PRMs is concerned with sampling aspects. Many samples distributed in $\mathcal{C}_{free}$ do not improve the connectivity of the roadmap and are therefore almost useless. To overcome this issue [SLN99] proposed a PRM based on the Visibility Graph introduced in Section 3.2.1. The main idea is to retain only the collision-free samples that are "visible" by any two other configurations. Other research efforts trying try to handle up this issue can be found in [HJRS03, AKC04, KH04].

2 – Rapidly-exploring Random Tree

The RRT first presented in [Lav98]. In the basic RRT algorithm, a tree is incrementally grown from the initial configuration, exploring the $\mathcal{C}_{free}$ on an attempt to reach the goal configuration (see Figure 3.7-left). In each step, a random sample is taken in $\mathcal{C}$. If this sample is collision-free, then the algorithm tries to connect it to the nearest node of the growing tree.

One of the best known variants of the RRT algorithm is the Bi-directional RRT method [LK01] which grows two trees, rooted at the initial and goal configurations, on an attempt to reach the same collision-free sample. This variant is illustrated in Figure 3.7-right.

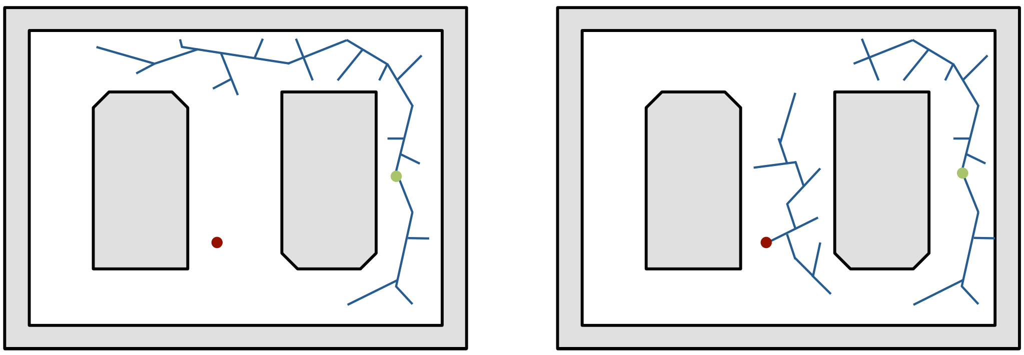


Figure 3.7: Left: the RRT algorithm grows a tree (in blue) from the initial configuration (green-o) that explores $\mathcal{C}_{free}$ on an attempt to reach the goal configuration (red-o). Right: dual-tree variant of the RRT algorithm.

Being an incremental search method, the RRTs algorithm is only capable of solving single motion queries, i.e., the motion planning query is defined with a single pair of start and goal configurations. Moreover, the information obtained during the search process is unused for posterior motion queries. Despite these facts and since they do not use preprocessing phase, these methods are generally faster than multi-query planners.

Further details on this motion planning method are given later in Chapter 4 and Chapter 5, where different planners are developed to solve the motion planning of the rhombic vehicle, which is formulated and described in the following section.

3.3 Motion Planning Problem for the Rhombic Vehicle

So far, this chapter has addressed the motion planning problem in a comprehensive way, given a mathematical formulation for the basic motion planning problem, referring to some motivational applications (Appendix D) and finally providing a survey on the motion planning techniques available in the literature. Chapter 1 has also vaguely introduced the need to address the motion planning problem, contextualizing it in the autonomous navigation task of the rhombic vehicle. Notwithstanding, the concrete motion planning for the rhombic vehicle as still not been formulated. Thereby, this section impacts the specific motion planning problem of the rhombic vehicle within transfer and other operations in ITER. The motion problem is completely described and the solutions proposed in the dissertation to tackle it, are unveiled.

During ITER lifetime, the rhombic vehicle will be frequently asked to move autonomously from its current position to a desired or goal location. These "motion requests", from now on referred to as motion queries, will be frequent and might appear in different contexts and scenarios, such as in nominal cask transfer operations between the Vacuum Vessel Port Cells (VVPCs) in the TB and the Docking Ports (DPs) in the HCB, in parking maneuvers and also in recovery or rescue missions.

A common feature within the operations described above, is that they involve an autonomous driving of the vehicle between two configurations. These operations could equally be performed "manually" by human remote operation. However, the autonomous level is desirable as to provide an increased reliability to the operations, since some of the vehicle configurations are hard to achieve manually due to space confinement. This also allows to decrease faults and avoid repetitive human interventions.

Presently, a major IO and Fusion for Energy (F4E) RH teams concern consists in unravel the impact of these missions on the design and assembly of the different ITER components, as to guarantee a timely and effective integration of the rhombic vehicle, as a RH key technology, on the overall project. In particular, it is important to gather information about the space required for the safe convey of radioactive material, detect possible conflicts with the current building design and analyze compromising and hard-solving situations such as rescue or recovery operations.

Planning and analyzing these autonomous missions, either on the present design stage or later during ITER operation, are hard and time consuming tasks, if performed manually. Computerized techniques are thus vital to reduce the cost and time and increase the quality of such operations. Moreover, providing that CAD models of the ITER environments are available as well the information about the start and goal objectives for the different rhombic vehicle missions, the employment of motion planning techniques previously presented in Section 3.2, appear as a powerful tool to plan and analyze the required motions.

The problem and the challenges involved are better understood if grounded in the formulation presented in Section 3.1 as follows.

1. **Agent** – The agent performer of the plan, i.e., the solution for the motion query, is the rhombic vehicle. The motion planning module considers this vehicle as a movable system that behaves under the considerations presented in Section 2.1. The planning solutions depend directly on the rhombic

vehicle properties (e.g. geometric and kinematics).

2. **Configuration** – The configuration adopted corresponds to the pose of the rhombic vehicle defined with respect to a global fixed frame as the one defined in Figure 3.8. This configuration comprises the cartesian coordinates of the vehicle center (C) and the vehicle orientation, i.e., $\boldsymbol{q} = [x_C \quad y_C \quad \theta]^T$. Despite being an important element on the motion planning problem formulation, not every planner explicitly deals with the configuration definition. For instance, in the dissertation, this only occurs on the planning strategy followed in Chapter 5. Moreover, the motion query may be established based on a smaller configuration definition comprising only the position of the vehicle.
3. **Start and goal configurations** – define the start and final poses forming the motion query to the vehicle.
4. **Geometric model of the World** – model of the scenario, i.e., the occupied volumes of the scenario where the rhombic vehicle has to move and that constitute relevant information for the definition of a collision free plans. The motion planning of the rhombic vehicle is solved in a 2D world for which the original CATIA models, which are in 3D, must be converted to 2D projections on the floor level. Environments such as the ones found in the TB or in the HCB are well structured and therefore, they can be modeled as a set of planar walls composed of line segments connected with two points, as depicted in Figure 3.8 in item 2.
5. **Plan** - solution to the motion query, this plan must correctly drive the rhombic vehicle from the starting to the goal pose and without collisions. In the dissertation, and without loss of generality, it is assumed that this plan consist on a trajectory defined for the vehicle center, or in a complementary way, for each wheel of the vehicle. As shown in Figure 3.8, different representations can be adopted, such as the spanned area or the occupied volume by the vehicle.
6. **Optimization Criteria** – In addition to the free collision criteria, the trajectory plan shall comply with other requirements. Considering the large dimensions of the vehicle and the cluttered nature of the environments, each planned path shall satisfy the following criteria: (1) maximize the clearance to the obstacles to reduce the risk of collision; (2) minimize the distance traveled to prevent the need to recharge the on-board batteries during the journeys; (3) guarantee smooth transitions to avoid jerky motions and reduce the number of maneuvers.

The planning methods surveyed in Section 3.2 provide the means to efficiently solve the basic geometric problem described in Section 3.1. However for more concrete and complex scenarios as the one described above the same techniques are insufficient to completely solve the motion planning problem. For example, a set of free-collision motions can be determined for the rhombic vehicle that cause it to arrive to the desired destination. However, feasibility and safety criteria upon execution is not guaranteed. To achieve feasible and optimized solutions for rigid bodies such as the rhombic vehicle and account for optimization criteria the seminal forms of motion planning methods must be improved and coupled with additional techniques.

Note that the desired final output is an optimized trajectory, which entail that vehicle kinematic and dynamic constraints are satisfied. One approach is to regard a trajectory as a decoupled solution composed

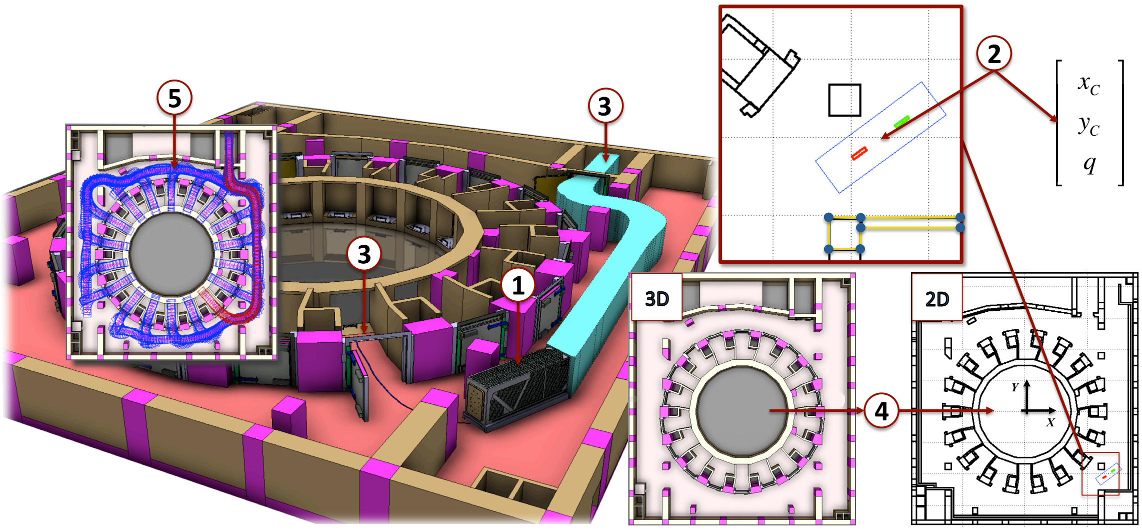


Figure 3.8: The motion planning problem for the rhombic vehicle: 1- rhombic vehicle, 2- pose, 3- start and goal conditions, 4- geometric model (3D and 2D), 5-optimized trajectories for each VVPC in the TB.

of a path defining a set of vehicle configurations or points that should be followed by the vehicle and associated velocity profile. This option raises the possibility of using path planning methodologies to first achieve a geometric description of the vehicle motion and then couple it with a time parameterized law that embeds vehicle dynamic assumptions. This strategy is followed in chapter 4 to solve the motion planning problem of the rhombic vehicle, with small improvements that include path optimization techniques. It will be referred to as a decoupled or refinement planning strategy, since the solution plan is progressively improved. The term path is used whenever the velocity component is not relevant.

On the other hand, Subsection 3.1.2 discussed the existence of an higher planning instance, which allows to directly evaluate trajectory solutions that inherit vehicle differential constraints. This alternative approach is addressed in Chapter 5 and is referred to as trajectory planning.

Chapter 4

A Refinement Strategy for Motion Planning

This chapter proposes to tackle the motion planning problem stated for the rhombic vehicle in Section 3.3, by following a decoupled and refinement strategy, which is depicted in Figure 4.1 and is composed by the following three stages:

1. **Geometric path evaluation** – Provided that the geometric model of the environment and the initial and goal configurations of the vehicle are given, an initial collision-free path connecting the two objective configurations is obtained;
2. **Path optimization** – This stage transforms the initialization path into an optimized solution that satisfies different optimization criteria;
3. **Trajectory evaluation** – The last stage considers how the vehicle should move along the path in order to meet time and vehicle differential constraints.

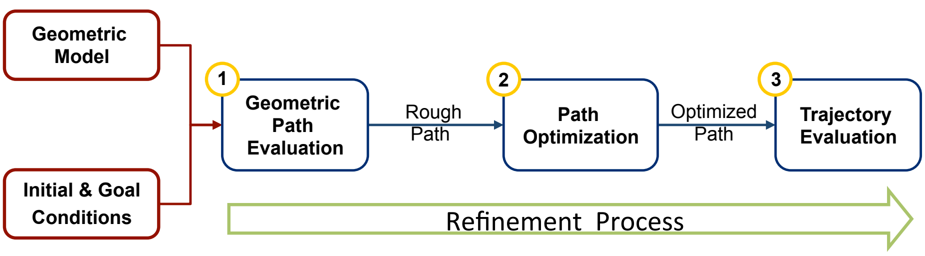


Figure 4.1: Motion Planning methodology: a three stage refinement strategy.

In the following of this chapter two approaches are presented that handle the first and second stage of this general framework. Each approach assumes different assumptions for the path generation and optimization. In particular, they employ techniques with distinctive algorithmic natures. For this reason they are presented in separate sections as follows:

- Section 4.1 presents a combinatorial based approach, which assumes that the outputted trajectories of the planning stage shall represent a unique reference to be followed by both wheels of the rhombic vehicle, as illustrated in Figure 4.2-left. The geometric path evaluation is carried using a point-like robot approach by means of a combinatorial method named of CDT (Subsection 4.1.2). A dedicated path optimization algorithm is presented in Subsection 4.1.2 based on the Elastic Bands approach [QK93a, QK93b] which guarantees collision free paths for the vehicle, in addition to other optimizing criteria. This refinement strategy, has been developed under a procurement grant F4E-2008-GRT-016 (MS-RH), with sparse contributions by the dissertation author. A full description of the method is important for the understanding and contextualization of the subsequent approach. Hence, a brief description is given in this chapter with the main dissertation contributions, for this approach, highlighted in *italics* and moved to appendix to save space.

- The second approach assumes an increased vehicle flexibility, where the wheels are allowed to follow independent trajectories (see Figure 4.2-right). The geometric path evaluation is performed using the RRT algorithm described in Subsection 3.2.2, which explicitly considers the vehicle geometry thus evading the previous point-like robot approach and directly returning collision-free paths for the vehicle. Drawing inspiration in the former approach and on rigid body dynamics, a different path optimization algorithm is developed (Subsection 4.2.2), which is capable of fully exploring the high maneuvering ability of the rhombic vehicle. Of particular note the fact that the outcome and developments of this approach has provided a valuable contribute on the area of optimization of trajectories for the rhombic vehicle, supporting the won of a procurement grant from F4E F4E-GRT-276-01 (MS-RH), in the area of RH.

The terms adopted for Section 4.1 and 4.2, which are "combinatorial approach" and "randomized approach", respectively, refer to the algorithmic nature of the geometric path evaluation module.

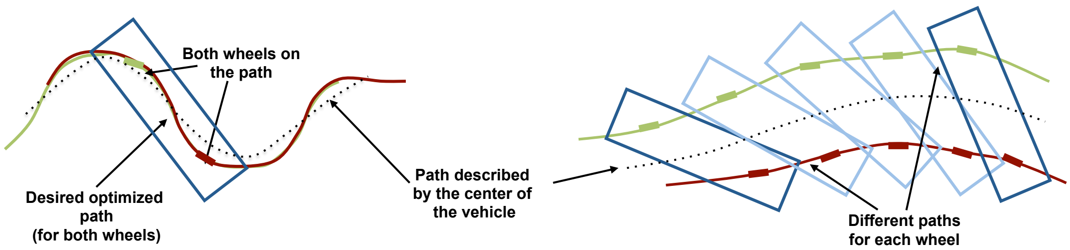


Figure 4.2: Left: the rhombic vehicle following a unique path reference for both wheels. Right: executed motion by the rhombic vehicle while following independent references for each wheel.

Lastly, Section 4.3 presents an algorithm for trajectory evaluation to completely define the rhombic vehicle motion along the obtained optimized paths in stage 2. The proposed algorithm suits any of the approaches presented in Section 4.1 and 4.2.

4.1 Combinatorial Approach

Combinatorial or exact algorithms were vaguely introduced in Section 3.2.1. These methods explicitly represent $\mathcal{C}_{obst}$ by applying combinatorial discrete techniques and do not resort to approximations. This is an important property since it allows to efficiently determine if the motion query has a geometric feasible (or collision-free) solution. This is in contrast to the sampling-based planning algorithm presented in Section 4.2.

The good geometric properties of the adopted 2D representation of the ITER environment models, such as low dimensionality and convexity, allow the use of a combinatorial planning approach, an option that is often left apart due to (explicit) modeling issues. Therefore, for the geometric path evaluation, a triangle CD method is adopted, but other combinatorial methods could be used, such as the Voronoi Diagram [Can85, DY82] or other roadmap-based methods.

4.1.1 Constraint Delaunay Triangulation

From the existent CD methods, a triangle cell arrangement was adopted, using the Delaunay Triangulation (DT). Each triangle cell resulting from the DT is composed both by real environment edges (footprint of walls) and virtual edges. However, as represented in Figure 4.3-left, the naive DT algorithm may return virtual edges that cross real walls since it considers only points. To overcome this issue and include the obstacles constraints, it is proposed the used of an improved implementation of the DT algorithm, the CDT [Chew87], illustrated in Figure 4.3-right.

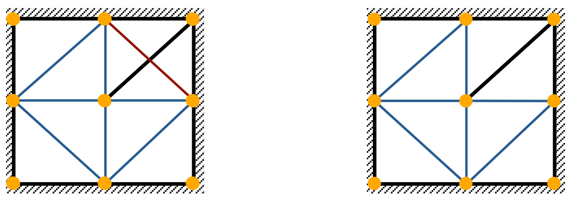


Figure 4.3: Triangle CD on a 2D map with: the DT (left) and the CDT (right).

The overall procedure to determine the initial geometric path is represented in Figure 4.4, for the reference scenario used in Chapter 3, and can be described as follows:

1. For a specified scenario, the CDT is applied to the corresponding 2D map yielding the triangle CD, as illustrated in Figure 4.4-a. This corresponds to a connectivity graph $\mathcal{G}$ that captures the structure of $\mathcal{C}_{free}$. Let T denote the set of N triangle cells composing $\mathcal{G}$,

$$T = \{T_n \quad | \quad n = 1, \dots, N\}; \tag{4.1}$$

2. To handle a specific motion query, i.e., to connect the initial configuration, $\boldsymbol{q_I}$, to the goal configuration, $\boldsymbol{q_G}$, the next step determines which cells, herein denoted by T_I and T_G, contain these two configurations. Recall that a point-like robot approach is assumed and therefore the orientation of the vehicle is ignored; These initial configurations are defined as simple Cartesian points;

3. Using the cell adjacency property, all the possible triangle cell sequences (depicted in Figure 4.4-e) connecting T_I to T_G and composed by consecutive cells that do not share a real edge are evaluated. The desired cells in the sequences are connected by virtual edges, since feasible solutions cannot cross walls represented by real edges. Let

$$S = \{S_h \quad | \quad h = 1, \ldots, H\}, \tag{4.2}$$

be the set of all cell sequences. Due to its complete nature, if such a sequence does not exist, the algorithm states that there is no solution for the proposed query;

4. Considering the last approach, the final step converts the shortest cell sequence into an ordered sequence of points connected by line segments. First, $\boldsymbol{q_I}$ is connected to the middle edge point of the two first cells in the sequence. Then, the middle edge points of two consecutive cells of the sequence are taken as path sample points and are linked by a straight line. Finally, the middle edge point of the last two cells in the sequence is connected to $\boldsymbol{q_G}$.

To increase the efficiency of the CDT algorithm, the dissertation proposes to use the A^* algorithm [HNR68] in alternative to the exaustive search performed in step 3. Simulations experiments shown how this algorithm can dramatically fasten the search for the shortest triangle cell sequence in complex scenarios such as the TB, that are composed off numerous triangle cells.

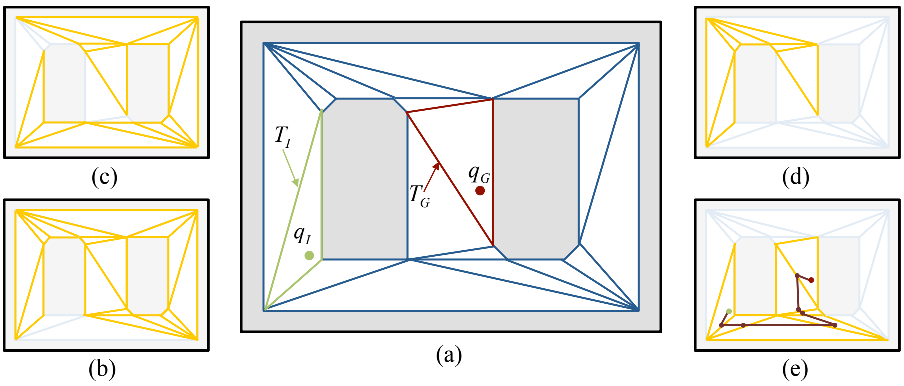


Figure 4.4: Evaluation of the initial geometric path: (a) Triangle CD with initial and goal conditions. (b)−(e) Possible triangle cell sequences. (e) Shortest triangle cell sequence and corresponding geometric path.

4.1.2 Elastic Bands

The CDT combined with the A^* is able to efficiently (regarding the very low computational cost) determine a geometric path connecting $\boldsymbol{q_I}$ to $\boldsymbol{q_G}$. However, the feasibility upon execution is not guaranteed as the obstacle avoidance for a rigid body such as the rhombic vehicle remains an issue. Note that the combinatorial algorithm performs a search on the $\mathcal{C}_{free}$ for a point-like robot, which is topologically different from the $\mathcal{C}_{free}$ belonging to the rhombic vehicle (recall Figure 3.2). Another undesirable effect resulting from this geometric solution is the curvature discontinuity, which causes the vehicle to stop every time it mets a path discontinuity. The geometric solution is thus unfeasible to be followed by the vehicle.

The path smoothness constraint can be overcame by applying a B-spline interpolation method [dBo78], yielding a smoother path. Moreover this technique allows to increase the path resolution, which is also desirable. However two main issues remain to be solved in order to meet some of the criteria requirements defined in Section 3.3:

- Guarantee a collision-free path; this requires the improvement of the path clearance, by enlarging the minimum distance from the vehicle to obstacles (as a collision avoidance feature);
- Reduce path looseness in order to get shorter and smoother paths without slacks.

To tackle the referred issues, it is proposed the use of an optimization method based on the elastic bands concept. The elastic bands were proposed in [QK93a, QK93b], but they are closely related to another approach that first appear in the computer vision field, which was called "snakes" [KWT88].

Following the elastic bands approach, the path is modeled as an elastic band which can be compared to a series of connected springs subjected to two types of forces as illustrated in Figure 4.5:

- Internal or elastic forces ($\boldsymbol{F_e}$) – These forces simulate the Hooke's elasticity concept [KZ95, BJdW02]. This allows to simulate the path as a stretched band;
- External or repulsive forces ($\boldsymbol{F_r}$) – The obstacle clearance is achieved using repulsive forces to keep the path away from obstacles.

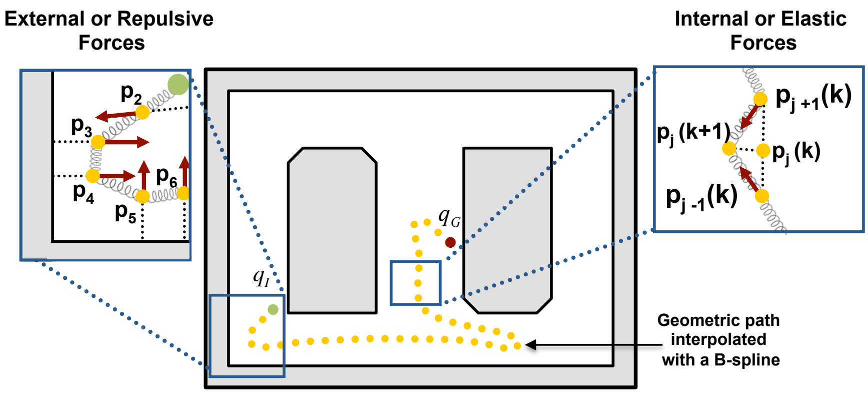


Figure 4.5: Elastic band formulation: internal or elastic and external or repulsive forces.

When submitted to these artificial forces, the elastic band is deformed over time becoming a shorter and smoother path, increasing clearance from obstacles.

The elastic force acting on each path point can be evaluated by

$$\boldsymbol{F_e}(\boldsymbol{j}) = K_e \cdot [(\boldsymbol{p_{j-1}} - \boldsymbol{p_j}) - (\boldsymbol{p_j} - \boldsymbol{p_{j+1}})] \tag{4.3}$$

with K_e being the elastic gain and $\boldsymbol{p_j}$, the j^{th} path point. The path elastic behavior is controlled through K_e; high values of K_e stretch the path while smaller values of K_e provide more flexibility to the path deformation.

The repulsive contributions from the obstacles can be achieved in many different ways. This in fact the main issue when adopting the elastic bands concept and comprises the main contribution of the dissertation for this approach. The simpler approach would evaluate the repulsive force magnitude as function of the distance between each path point and the respective nearest Obstacle Point (OP). This would effectively increase the path clearance around obstacles but disregarding completely the geometric constraints of the vehicle as a rigid body. Therefore, the path optimization presented here and published in [FVVR10], proposes the use of an innovative repulsive scheme implementation that directly considers the vehicle geometric constraints and complies with the constraint of a unique reference for both wheels. The details for the evaluation of the repulsive forces are given in Appendix E.

With the elastic and repulsive forces evaluated, the path is then iteratively updated according to the following equation:

$$\boldsymbol{p_{j,new}} = \boldsymbol{p_{j,old}} + \xi \cdot \boldsymbol{F_{total}}(\boldsymbol{p_{j,old}}) \tag{4.4}$$

scaled by some factor ξ and where the total force contribution is given by

$$\boldsymbol{F_{total}}(\boldsymbol{p_{j,old}}) = \boldsymbol{F_e}(\boldsymbol{p_{j,old}}) + \boldsymbol{F_r}(\boldsymbol{p_{j,old}}) \tag{4.5}$$

The overall path optimization process is illustrated in Figure 4.6 for a motion query requesting a motion between the upper and lower opposite corners of the map. Using a combinatorial approach, the shortest feasible triangle cell sequence is evaluated and transformed into a geometric path. This initial solution is smoothed via the B-spline interpolation to satisfy the minimum curvature constraint. As it can be observed in Figure 4.6-left, the vehicle occupancy displays small clearance[1] making this solution path is unfeasible. To improve obstacle clearance and path smoothness, the solutions must be driven through an optimization module where the path is modeled as an elastic band. The path deformation process is well illustrated in Figure 4.6-center with the plot of the evolution of the spanned area by the vehicle. Figure 4.6-right displays the output of this optimization module, a shorter, smoother and safer path when compared to the initial geometric solution.

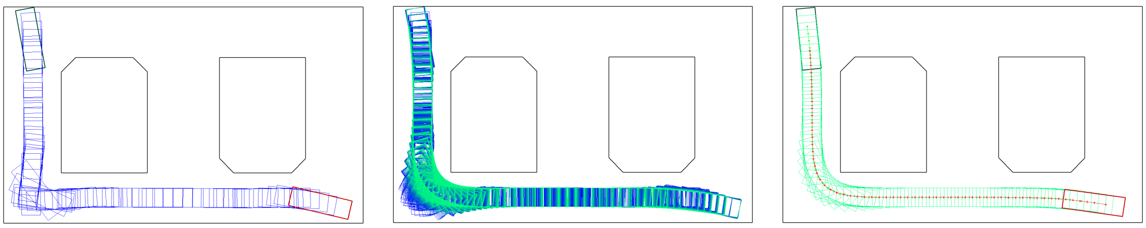


Figure 4.6: Path optimization with elastic bands: geometric path (left), path deformation (center) and vehicle occupancy area over the optimized path (right).

The described planning and optimization approach does not considers the inclusion of maneuvers as part of the solution path and maneuvers can greatly simplify the motion planning problem. Considering the large dimensions of the rhombic vehicle, maneuvers frequently arise as the only solution to safely solve

[1]In this approach, the geometric path may also return collisions with obstacles.

the space confinement or even to overcome vehicle orientation constraints. To handle up these issues, the path planning and optimization approach herein described was extended to incorporate the design and planning of the required maneuvers. Full details are again remitted to Appendix E and [VVFR10]. The important to retain is that this extended approach still relies on the elastic bands and the optimized path, with maneuvers included, ensure obstacle clearance and length optimization of the global solution path.

4.2 Randomized Approach

The path planning and optimization approach presented so far was applied to evaluate most part of the nominal operations in the TB and the HCB showing a good proficiency [VRF$^+$09, VVFR10, FVVR10]. Despite of the good performances obtained, this approach presents some limitations, namely:

- The geometric path evaluation module does not allow to include the vehicle orientation in the motion query; Consequently, it can not be defined the exact configuration with which the vehicle shall start and finish its motion.
- Assuming the same reference for both vehicle wheels restrict the native rhombic vehicle maneuvering capabilities, significantly reducing the advantage of using such a kinematic configuration.
- The vehicle geometry is not explicitly included in the path optimization module. During the optimization process, the vehicle occupancy area or poses along the path along are determined as function of the path points positioning. These raises some issues such as how to deform the path and how to automatically identify maneuvers as underlined in Appendix E.

The importance of these aspects is better realized by drawing a specific example for a rhombic vehicle mission. Figure 4.7-left illustrates part of the scenario in the TB, where a CPRHS, acting as a rescue vehicle, has to assist another CPRHS that is docked in a specific VVPC. For the particular case where both wheels are constrained to follow the same path, the planning approach of the previous section, with its inherent limitations, is not able to found an optimized path that positions the vehicle on the desired goal configuration (Figure 4.7-center). However, as shown in Figure 4.7-right, this would be possible by letting the vehicle wheels to follow independent references.

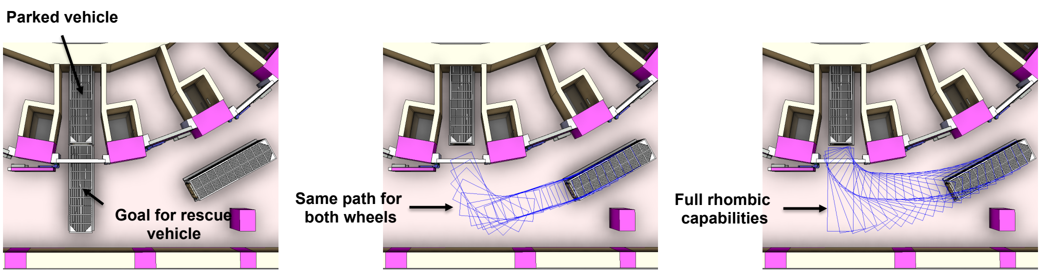


Figure 4.7: Rescue operation: To assist the parked CPRHS, the rescue-like CPRHS must be perfectly aligned with it (left). Planned solution for the same path for both wheels (center). Planned solution with independent references for each wheel.

The same observation is valid for other situations, where the exploration of the rhombic vehicle maneuverability appears as the only option to puzzle out the space confinement or to simplify the motion problem, for instance, in parking operations where parallel maneuvers can dramatically increase energy saving and ease parking logistics.

All these aspects encouraged the development of a new path planning and optimization approach capable of profiting from the high maneuvering ability of the rhombic vehicle and tackle the limitations referred above.

The randomized approach proposed in this section, aims to handle in a different manner the first and second stage of the general refinement strategy and is described next.

4.2.1 Rapidly-exploring Random Tree

To evaluate the geometric path this new approach uses a randomized method, the RRT algorithm, which was only conceptually described in Subsection 3.2.2. The RRT algorithm was first presented in [Lav98] as randomized data structure suitable for a broad class of motion planning problems. The adoption of this planning technique stems from the fact that it naturally extends to the nonholonomic planning problem (which is addressed in Chapter 5) in contrast with other popular randomized approaches such as the PRM [KSLO96] or the Randomized Potential Field [BL91]. Moreover, RRTs explicitly handle the vehicle geometry during the search and are thus able to directly generate collision-free paths for rigid bodies (in contrast with point like robot approaches).

The seminal form [Lav98] of the RRT algorithm grows a single tree from the initial configuration ($\boldsymbol{q_I}$), until one of its branches reaches the goal configuration ($\boldsymbol{q_G}$). Algorithm 1 presents the pseudocode for this most basic RRT variant for an holonomic problem and is described as follows.

In each iteration, a configuration is randomly sample yielding $\boldsymbol{q_{rand}}$. The *growTree* in step 11, selects the closest node (or configuration) in the tree to the sampled configuration $\boldsymbol{q_{rand}}$, using a specified distance metric (ρ). The dissertation the most simple and commonly used metric that treats $\mathcal{C}$ as Cartesian space and defines a Euclidean-based metric as follows.

$$\rho(\boldsymbol{q_1}, \boldsymbol{q_2}) = w_t \cdot \|\boldsymbol{X_1} - \boldsymbol{X_2}\| + w_r \cdot \mathcal{F}(\theta_1, \theta_2), \tag{4.6}$$

which is a weighted metric with the translation component $\|X_1 - X_0\|$ using a standard Euclidean norm, and the positive scalar function $\mathcal{F}(R_1, R_2)$ returning an approximate measure of the distance between two rotations within the interval $[-\pi, \pi]$. The rotation distance is scaled relative to the translation distance via the weights w_t and w_r.

In step 12, an attempt is made to add a new edge to the tree, from $\boldsymbol{q_{near}}$ towards $\boldsymbol{q_{rand}}$. These edges are created using either a single step operation (Extend-variant), i.e., a motion with a increment step ϵ, or by performing a multistep operation (Connect-variant [KL00b]) where ϵ is iterated until an obstacle or $\boldsymbol{q_{near}}$ is reached. The later approach is favored in holonomic problems whereas the Extend-variant is preferred for problems that involve differential constraints. Note that the RRT performs a sampling search in $\mathcal{C}$ avoiding an explicitly representation of $\mathcal{C}_{free}$. Therefore, the last mentioned step inherits

Algorithm 1: single-tree RRT

function: $\tau \leftarrow$ `singleRRT(`$\boldsymbol{q_I}, \boldsymbol{q_G}, \mathcal{C}$`)`
  $\tau \leftarrow$ `tree(`$\boldsymbol{q_I}$`)`
  while $time \leq time_{max}$ **do**
    $\boldsymbol{q_{rand}} \leftarrow$ `randomConfig(`$\mathcal{C}$`)`
    $\boldsymbol{q_{new}} \leftarrow$ `growTree(`$\tau, \boldsymbol{q_{rand}}$`)`
    if $\boldsymbol{q_{new}} \wedge \rho(\boldsymbol{q_{new}}, \boldsymbol{q_G})$ [a] $< \zeta$ **then**
      return `extractSol(`$\boldsymbol{q_{new}}$`)` [b]
    $\tau \leftarrow$ `tree(`$\boldsymbol{q_I}$`)`
  return *failure*
function: `growTree(`$\tau, \boldsymbol{q_{target}}$`)`
  $\boldsymbol{q_{near}} \leftarrow$ `nearestNeighbor(`$\tau, \boldsymbol{q_{target}}$`)`
  $\tau \leftarrow \tau +$ `newEdge(`$\boldsymbol{q_{near}}, \boldsymbol{q_{target}}$`)` [c]
  return $\boldsymbol{q_{new}}$

[a] distance metric
[b] constructs solution by traveling up the tree
[c] creates new edge from $\boldsymbol{q_{near}}$ to $\boldsymbol{q_{rand}}$

Algorithm 2: dual-tree RRT

function: $\tau \leftarrow$ `dualRRT(`$\boldsymbol{q_I}, \boldsymbol{q_G}, \mathcal{C}$`)` [a]
  $\tau_a, \tau_b \leftarrow$ `tree(`$\boldsymbol{q_I}$`)`,`tree(`$\boldsymbol{q_G}$`)`
  while $time \leq time_{max}$ **do**
    $\boldsymbol{q_{rand}} \leftarrow$ `randomConfig(`$\mathcal{C}$`)`
    $\boldsymbol{q_a} \leftarrow$ `growTree(`$\tau_a, \boldsymbol{q_{rand}}$`)`
    if $\boldsymbol{q_a}$ **then**
      $\boldsymbol{q_b} \leftarrow$ `growTree(`$\tau_b, \boldsymbol{q_a}$`)`
      if $\boldsymbol{q_b}$ **then**
        if $\rho(\boldsymbol{q_a}, \boldsymbol{q_b}) < \zeta$ **then**
          return `extractSol(`$\boldsymbol{q_a}, \boldsymbol{q_b}$`)`
    $\tau_a, \tau_b \leftarrow \tau_a, \tau_b$
  return *failure*

[a] inherits `growTree` and from Algorithm 1

a collision avoidance module which allows to determine wether a given configuration is collision-free or not. The search process runs until $\boldsymbol{q_G}$ is reached (or an ϵ-boundary of $\boldsymbol{q_G}$) or the time elapsed exceeds a specified constant.

An hasty improvement can be obtained by replacing $\boldsymbol{q_{rand}}$ for $\boldsymbol{q_G}$ a certain portion of iterations (5 % or 10%) as to bias the RRT growth. As discussed in [LK00], this causes the tree to converge faster to $\boldsymbol{q_G}$ when compared to the basic RRT method. Another interesting improvement discussed in [KL00b, LK00], consists on growing two trees, rooted at $\boldsymbol{q_I}$ and $\boldsymbol{q_G}$. One tree is grown towards a random configuration, as before, while the other grows towards any configuration of its counterpart, with the tree switching roles between iterations. This variant is outlined in Algorithm 2.

To gain understanding of the RRTs, Figure 4.8 represents the construction process of a single-tree RRT framed at 40 (a), 500 (b) and 2000 (c) samples.

4.2.2 Path Optimization Based on the Rigid Body Dynamics

The geometric path evaluation stage previously presented is able to generate collision-free paths that account for the vehicle geometry. However, as for the case of the combinatorial approach (Section 4.1), it is obtained a rough and low quality solution, which presents small clearance over the obstacles and induces jerky motions that do not favor tracking purposes. The employment of a path optimization technique is thus required as to achieve a feasible solution that can be followed by the rhombic vehicle. Moreover, the limitations of the former refinement strategy dictated the development of a new path optimization technique able of exploring the high maneuverability of the rhombic vehicle.

Taking into consideration all theses aspects, this section proposes a novel path optimization technique,

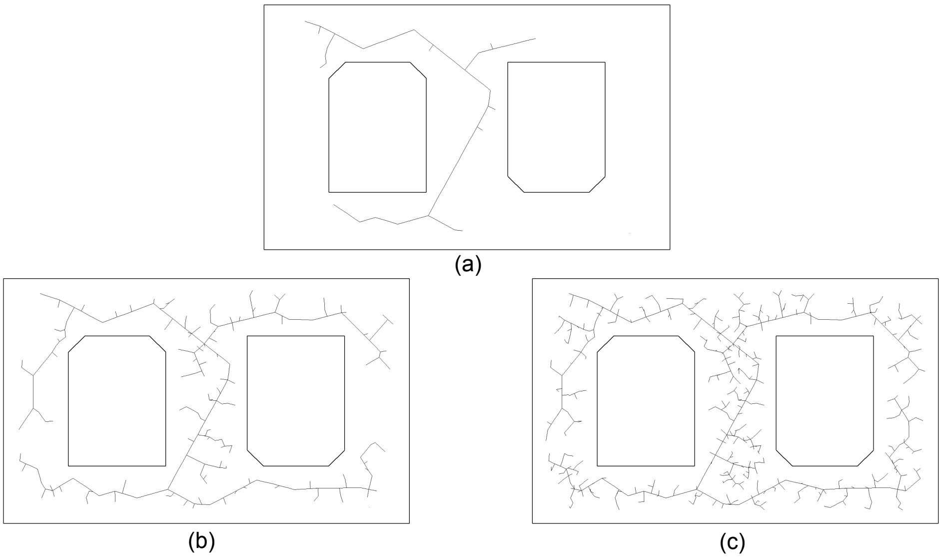


Figure 4.8: The RRT quickly expands in a few directions to explore the free space of the environment.

which conserves the idea of path optimization as path deformation problem from the elastic bands. However, by drawing inspiration on the dynamics of rigid bodies and taking advantage of the explicit representation of the rough query path, which defines a set of collision-free vehicle configurations from $\boldsymbol{q_I}$ to $\boldsymbol{q_G}$, this new method evades the formulation of paths as particle systems.

This new method is schematized in Figure 4.9 and can be summarily described as follows. In the path optimization process, each of the consecutive vehicle poses that form the rough query path is treated as a rigid body that is connected with its adjacent poses like a convoy through internal interactions and subjected to external-repulsive forces produced by obstacles in its vicinity. Hence, the path optimization becomes a path deformation problem, which relies on the principles of rigid body dynamics to iteratively simulate the evolution of each pose on the optimization process. In particular, it is proposed to subject each vehicle pose in the query path to two types of efforts:

- **Internal efforts** – Consecutive poses are kept connected through virtual elastic and torsional springs, which simulate the Hooke's elasticity concept and originate **elastic forces** and **torsional torques**. These efforts guarantee smoothness on deformation and help to shorten the path;
- **External efforts** – **Repulsive forces** repel the rigid poses from obstacles, acting as a collision avoidance feature. Moreover, force eccentricity originates **repulsive** rotating **torques**, which readapt poses orientation maximizing clearance over the obstacles.

In the following some notions on rigid body dynamics are reviewed easing the ensuing implementation of this method.

4.2.2.1 Dynamics of Rigid Bodies

For this study's purposes, the use of rigid body dynamics is restricted to the case of *general plane motion*, i.e., the particles composing the rigid body move in parallel planes and their motion is neither

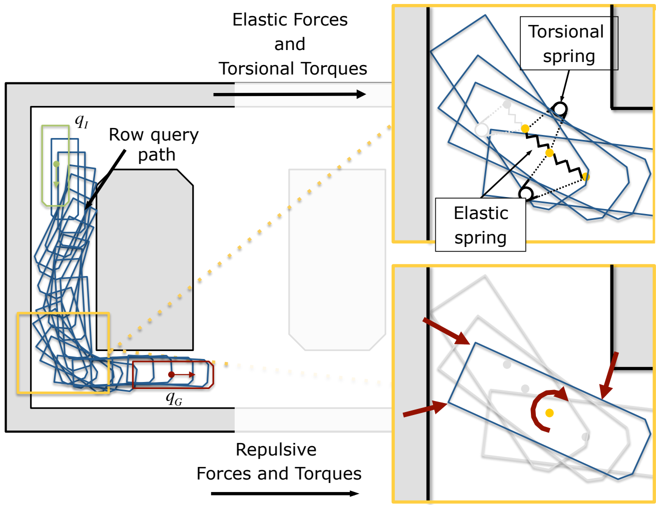


Figure 4.9: Optimization based on the rigid body dynamics: the elastic forces and torsional torques help to smooth and shorten the path while the repulsive forces and torques maximize clearance from obstacles.

characterized by pure rotational nor pure translational movements.

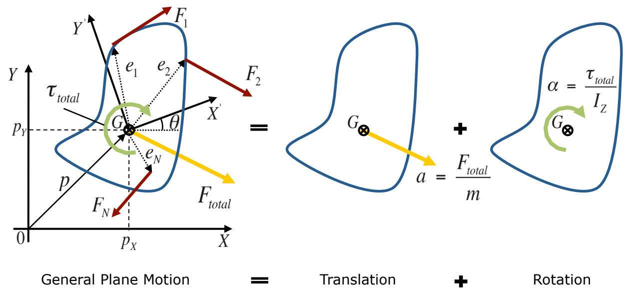


Figure 4.10: Kinetics of rigid bodies: the general plane motion of a rigid body can be decomposed on a translation and a rotation about G.

Consider that a rigid body with a CoG denoted by G, as represented in Figure 4.10, is acted by N external forces, $\boldsymbol{F_n}$, with $\{n = 1, \cdots, N\}$. Following the *Newton's Second Law* and taking the rigid body as a system of particles, the dynamics of G, with respect to the inertial frame OXY, is given by

$$\boldsymbol{F_{total}} = \sum_{n=1}^{N} \boldsymbol{F_n} = m\boldsymbol{a}, \tag{4.7}$$

where m is the mass of the body and $\boldsymbol{a}$ is the linear acceleration of G. The dynamics of the rigid body motion relative to the its body frame, $CX'Y'$, is given by

$$\boldsymbol{\tau_{total}} = \sum_{n=1}^{N} \boldsymbol{F_n} \times \boldsymbol{e_n} = I_z \alpha, \tag{4.8}$$

which entails that the resultant torque about G, $\boldsymbol{\tau_{total}}$, is a vector with the direction of the angular acceleration, α, and magnitude $I_z\alpha$. In (4.8), I_z is the moment of inertia around the perpendicular axis passing through G, whereas $\boldsymbol{e_n}$ corresponds to the position vector of $\boldsymbol{F_n}$ relative to the reference frame $GX'Y'$. For the case of uniformly accelerated motion, which is adopted in this formulation, $\boldsymbol{a}$ and α assume constant values over time.

As it is represented in Figure 4.10 and, from the kinetics viewpoint, the general plane motion of the rigid body can be decomposed as the combination of a translation with linear acceleration, $\boldsymbol{a}$, and a rotation about G with angular acceleration, α, given by (4.7) and (4.8), respectively. The linear, $\boldsymbol{v}$, and angular, ω, velocities of the rigid body's CoG can be obtained through integration of $\boldsymbol{a}$ and α over time, t, as follows,

$$\boldsymbol{v} = \boldsymbol{v^0} + \boldsymbol{a}t \tag{4.9}$$

$$\omega = \omega^0 + \alpha t \tag{4.10}$$

where $\boldsymbol{v^0}$ and ω^0 are the initial linear and angular velocities.

The position, $\boldsymbol{p}$, and orientation, θ, of the rigid body can be accessed through the integration of (4.8) and (4.9), yielding

$$\boldsymbol{p} = \boldsymbol{p^0} + \boldsymbol{v^0}t + \frac{1}{2}\boldsymbol{a}t^2 \tag{4.11}$$

$$\theta = \theta^0 + \omega^0 t + \frac{1}{2}\alpha t^2, \tag{4.12}$$

where $\boldsymbol{p^0}$ and θ^0 are the initial position and orientation of the rigid body's CoG. Equations (4.7) - (4.12) completely describe the general plane motion of a rigid body, relating displacement, velocity and acceleration to the external forces, which are the cause of motion.

4.2.2.2 Implementation

Let $j = \{1, \dots, J\}$ be the index of the consecutive vehicle poses composing the query path, each defined by a configuration vector

$$\boldsymbol{q_j} = \begin{bmatrix} \boldsymbol{p_j} \\ \theta_j \end{bmatrix}, \tag{4.13}$$

where $\boldsymbol{p_j}$ and θ_j denote the position and the orientation of the pose $\boldsymbol{q_j}$ relative to a fixed reference frame, respectively. It is stated that $\boldsymbol{q_1} = \boldsymbol{q_I}$ and $\boldsymbol{q_J} = \boldsymbol{q_G}$.

The elastic force, $\boldsymbol{F_e}$, and the torsional torque, τ_t, evaluated for the vehicle pose at $\boldsymbol{q_j}$ are:

$$\boldsymbol{F_e}(\boldsymbol{q_j}) = K_e \cdot [(\boldsymbol{p_{j+1}} - \boldsymbol{p_j}) + (\boldsymbol{p_{j-1}} - \boldsymbol{p_j})] \tag{4.14}$$

$$\tau_t(\boldsymbol{q_j}) = K_t \cdot [(\theta_{j+1} - \theta_j) + (\theta_{j-1} - \theta_j)], \tag{4.15}$$

where K_e, the elasticity gain and, K_t, the torsional gain, control the elastic and torsional avoidance behavior on the path deformation, respectively.

The evaluation of the external efforts due to obstacle proximity relies on a heuristic-based collision detector module, which is capable of determining the set of i-nearest OPs to each sampled pose $\boldsymbol{q_j}$. The overall procedure to handle the evaluation of the repulsive forces and torques is illustrated in Figure 4.11 and can be described as follows:

1. Let $i = \{1, \ldots, I\}$ denote the index of the i-th OP relative to a specific pose $\boldsymbol{q_j}$. Let $\boldsymbol{u_{j,i}}$ be the vector

$$\boldsymbol{u_{j,i}} = \boldsymbol{V_{j,i}} - \boldsymbol{O_{j,i}}, \tag{4.16}$$

taken from each OP, $\boldsymbol{O_{j,i}}$, and the corresponding vehicle nearest point, $\boldsymbol{V_{j,i}}$.

2. To improve clearance during path deformation, distance-dependent repulsive forces are defined, where each pair of points $(\boldsymbol{O_{j,i}},\ \boldsymbol{V_{j,i}})$ determines a repulsive contribution. For a specific vehicle pose, $\boldsymbol{q_j}$, the repulsive contributions are defined as

$$\boldsymbol{r_{j,i}} = \frac{\boldsymbol{u_{j,i}}}{\|\boldsymbol{u_{j,i}}\|} \cdot f(\|\boldsymbol{u_{j,i}}\|) \tag{4.17}$$

where,

$$f(\|\boldsymbol{u_{j,i}}\|) = \max(0, F_{max} - \frac{F_{max}}{d_{max}} \cdot \|\boldsymbol{u_{j,i}}\|). \tag{4.18}$$

In (4.19), a maximum allowable magnitude, F_{max}, is assigned to avoid outsized values in the close vicinity of the obstacles. d_{max} denotes the distance up to which the repulsive force is applied.

3. For each pose $\boldsymbol{q_j}$, the total repulsive force is defined as

$$\boldsymbol{F_r}(\boldsymbol{q_j}) = \sum_{i=1}^{I} \boldsymbol{r_{j,i}}. \tag{4.19}$$

Using (4.7), the net repulsive torque around the j-th pose CoG is defined as

$$\tau_r(\boldsymbol{q_j}) = \sum_{i=1}^{I} \boldsymbol{r_{j,i}} \times \boldsymbol{e_{j,i}}. \tag{4.20}$$

The repulsive and elastic forces are combined on a total force contribution as,

$$\boldsymbol{F_{total}}(\boldsymbol{q_j}) = \boldsymbol{F_r}(\boldsymbol{q_j}) + \boldsymbol{F_r}(\boldsymbol{q_j}). \tag{4.21}$$

Similar approach is valid for the torsional and repulsive torques acting on each pose $\boldsymbol{q_j}$. This leads to the definition of a net torque expressed as,

$$\tau_{total}(\boldsymbol{q_j}) = \tau_t(\boldsymbol{q_j}) + \tau_r(\boldsymbol{q_j}). \tag{4.22}$$

The total force and the total torque are represented in Figure 4.11.

Once determined the efforts acting on each pose, the ensued motion is evaluated through the principles of rigid body. Equations (4.7) and (4.8), are rewritten as

$$\boldsymbol{a_j} = \frac{\boldsymbol{F_{total}(q_j)}}{m} - K_d \boldsymbol{v_j} \tag{4.23}$$

$$\alpha_j = \frac{\tau_{total}(\boldsymbol{q_j})}{I_z} - K_d \omega_j, \tag{4.24}$$

which provide the linear and angular accelerations for a specific pose $\boldsymbol{q_j}$. The last term in the right hand side of (4.23) and (4.24) represent damping effects introduced to reduce the oscillatory motion during path deformation. They are controlled through K_D, herein set equally for both the translational and the rotational motion components. Notice that both m and I_z in (4.23)-(4.24), do not refer to real

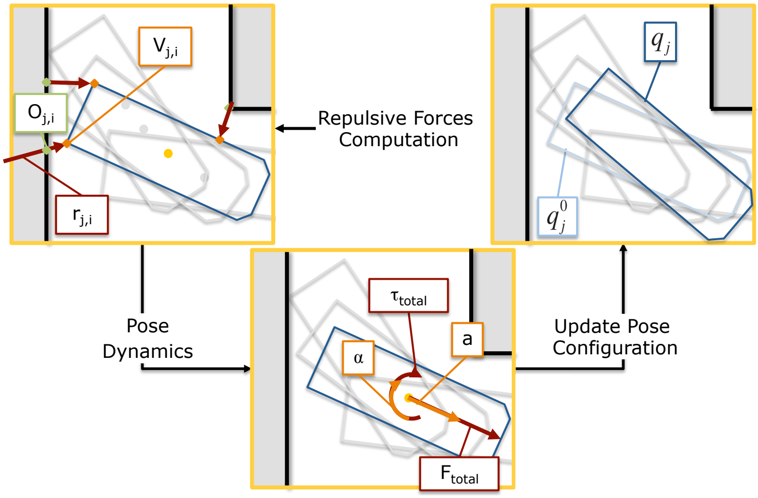


Figure 4.11: Path optimization flow based on the rigid body dynamics.

vehicle parameters but rather to simple scalars determining the resistance of each pose to change its configuration.

From a starting configuration in the query path, $\boldsymbol{q_j}$, this pose is updated iteratively according to the set of equations (4.8)-(4.11), where the referred initial conditions are the previous iterated pose in this process. The stopping criteria is defined by setting a maximum number of iterations.

4.2.2.3 A Numerical Approximation

To solve the problem stated above the use of numerical integration methods is required. The simpler Euler integration method tends to be numerically unstable and less accurate for small rates [AP98], meaning that the path may oscillate widely and not reaching a stable configuration. This paper proposes the use of the Leapfrog method, which is a modified version of the Verlet method [HLW03] and is nicely discussed in [FL63]. The Leapfrog method is commonly used to integrate Newton's equations of motion offering a greater stability when compared to the simpler Euler method.

Assume that the time discretization interval is Δt and represent as $\boldsymbol{v_j^{k\Delta t}} = \boldsymbol{v_j^k}$ the value of the linear velocity of the pose iterated from $\boldsymbol{q_j}$ at time instant $k\Delta t$. According to the flow diagram represented in Figure 4.12, the Leapfrog algorithm can be described for the previous stated approach as follows: (a) the linear, $\boldsymbol{a_j^k}$, and the angular, α_j^k, accelerations are evaluated at a given time step k using (4.23) and (4.24) with v_j and ω_j given by (4.30) and (4.31) and $\boldsymbol{F_{total}}$ and τ_{total} evaluated at $\boldsymbol{q_j^k}$;

(b) the corresponding velocities are calculated for the next "half step" (i.e., $k + \dfrac{1}{2}$) as

$$\boldsymbol{v_j^{k+1/2}} = \boldsymbol{v_j^{k-1/2}} + \Delta t \boldsymbol{a_j^k}, \tag{4.25}$$

$$\omega_j^{k+1/2} = \omega_j^{k-1/2} + \Delta t \alpha_j^k; \tag{4.26}$$

(c) accordingly, the iterated configuration of the pose $\boldsymbol{q_j}$ at time instant $k+1$ is then updated from the iterated configuration of the pose at time instant k as

$$\boldsymbol{q}_j^{k+1} = \begin{bmatrix} \boldsymbol{p}_j^{k+1} \\ \\ \theta_j^{k+1} \end{bmatrix} = \boldsymbol{q}_j^k + \Delta t \begin{bmatrix} \boldsymbol{v}_j^{k+1/2} \\ \\ \omega_j^{k+1/2} \end{bmatrix} \tag{4.27}$$

for $k \geq 0$. Note that $\boldsymbol{q}_j^0 = \boldsymbol{q}_j$, the pose at the query path.

It remains an issue how to evaluate the velocity at the next half-step when only the starting conditions are given. To get the calculation started and, as suggested in [FL63], the following simple approximation is used

$$\boldsymbol{v}_j^{1/2} = \boldsymbol{v}_j^0 + \frac{\Delta t}{2}\boldsymbol{a}_j^0 \tag{4.28}$$

$$\omega_j^{1/2} = \omega_j^0 + \frac{\Delta t}{2}\alpha_j^0. \tag{4.29}$$

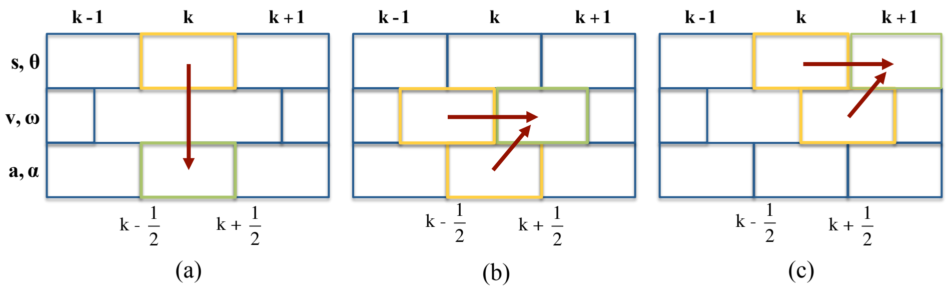


Figure 4.12: Flow diagram of the Leapfrog algorithm: evaluation of $\boldsymbol{a}_j^k$ and α_j^k (left); evaluation of $\boldsymbol{v}_j^{k+1/2}$ and $\omega_j^{k+1/2}$ (center); update of $\boldsymbol{q}_j^{k+1}$ (right).

The velocities at any given instant k are interpolated with

$$\boldsymbol{v}_j^k = \frac{1}{2}[\boldsymbol{v}_j^{k+1/2} + \boldsymbol{v}_j^{k-1/2}] \tag{4.30}$$

$$\omega_j^k = \frac{1}{2}[\omega_j^{k+1/2} + \omega_j^{k-1/2}]. \tag{4.31}$$

Figure 4.13 unveils a little the performance of the overall randomized approach for the same motion query used in Subsection 4.1.2. Further results and experiments on this approach are given in Chapter 6.

4.3 Trajectory Evaluation

The output of the path optimization module of the planning methodology is a collision free path suitable for execution. However, to achieve a realistic plan, it is necessary to determine how the rhombic vehicle should move along the path satisfying dynamic constraints. For that purpose the optimized paths should be parameterized in terms of velocities, i.e., converted into trajectories.

The velocity is an important issue within the rhombic vehicle motion; The designed trajectories must guarantee that the vehicle performs its motion in the shortest time period satisfying energy minimization

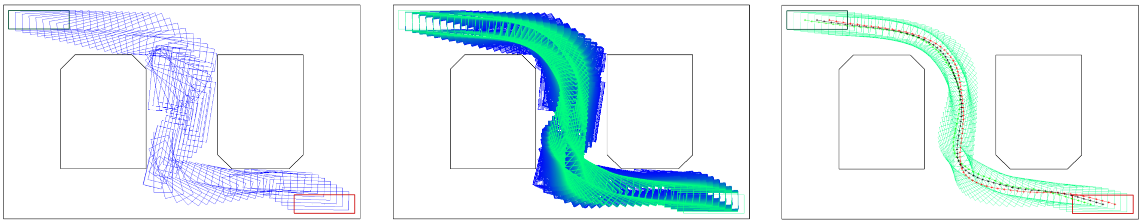


Figure 4.13: Path optimization based on rigid body dynamics: geometric path returned by the RRT (left), path deformation based on the rigid body motion simulation (center) and final optimized path (right).

requirements for air-cushion system and motor batteries. On the other hand, safety requirements are mandatory and the risk of collision shall be reduced. Given the cluttered environment where the TCS moves, an initial approach might define the vehicle velocity profile as a function of the distance to the obstacles. The velocity assumes low values when the vehicle is closer to the obstacles obstacles. Otherwise, the velocity could be greater, under the safety levels. However, this approach would be unable to handle vehicle dynamic constraints ignoring, for instance, the constraints on the admissible accelerations.

The purpose of this section is to present a trajectory evaluation method that is able to handle vehicle dynamic constraints and also incorporate safety requirements. This method integrates the last stage of the refinement planning strategy followed in this chapter and suits any of the two optimization approaches previously presented (Section 4.1 and 4.2).

In the following, and before unveiling the algorithmic details, some general notions about the mathematics behind motion are presented.

4.3.1 Motion profiles

The problem of trajectory evaluation is commonly find in a wide variety of applications specially on electro-mechanical systems. The so-called point-to-point motion, is by far the most used technique for this purposes [URLMP]. This means that from a stop position, an objet is accelerated to a constant velocity, and then decelerated till zero acceleration and velocity are reached at the arrival position. The two profiles more commonly used are the S-curve profile and the trapezoidal profile, represented in Figure 4.14-left and Figure 4.14-right, respectively.

As depicted in Figure 4.14-left, the S-curve profile consists on 7 different phases, which are described below:

- In phase I, motion start from the rest with a linearly increasing acceleration until the maximum allowable acceleration is reached;
- Motion continues at constant (maximum) acceleration during phase II;
- In phase III, comprises a deceleration until the maximum velocity allowed is achieved;
- In phase IV, the motion is performed at zero acceleration or constant velocity until deceleration phase begins;

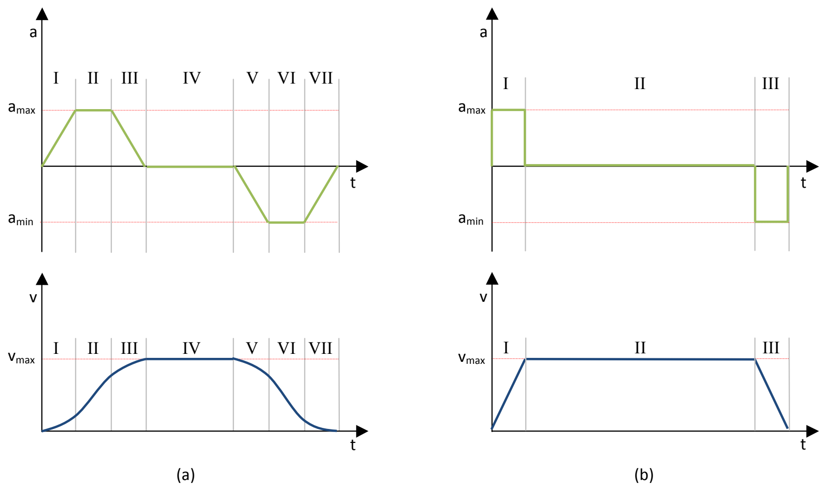


Figure 4.14: Speed and acceleration profiles: the S-curve (left) and the trapezoidal profile (right).

- The profile is repeated in a symmetric way as phases I,II and III.

The trapezoidal profile is simpler. As illustrated in Figure 4.14-right, the acceleration profile corresponds to a subset of the S-curve and it is only composed by phases II (constant acceleration), phase IV (constant velocity) and phase VI (constant deceleration), implying instantaneous transitions between phases. In its turn, the S-curve spreads the change of the acceleration over time. The motion parameter that defines the change in the acceleration is usually referred to as "Jerk". Hence, this value is infinite in the trapezoidal profile while in the S-curve profile it assumes a constant value.

Note that the Jerk is often ignored in the most part of motion related studies since it does not assume a usual physical meaning as for the case of the acceleration or velocity. However, this parameter can be extremely useful for some kind of applications. For instance, tuning the Jerk to small values may effectively prevent vehicle load oscillations, reduce vibrations and increase machine accuracy.

For completeness, the basic math required for these two profiles is stated here for the 2D case.

The motion for the trapezoidal profile is described with the equations (4.10) and (4.8). Retyping yields,

$$\boldsymbol{p} = \boldsymbol{p^0} + \boldsymbol{v^0}t + \boldsymbol{a}\frac{t}{2}^2 \tag{4.32}$$

and

$$\boldsymbol{v} = \boldsymbol{v^0} + \boldsymbol{a}t \tag{4.33}$$

For the S-curve profile, the equations are somehow similar but they contain and higher order derivative as to include the Jerk (J):

$$\boldsymbol{p} = \boldsymbol{p^0} + \boldsymbol{v^0}t + \boldsymbol{a}\frac{t}{2}^2 + \frac{1}{6}\boldsymbol{J}t^3 \tag{4.34}$$

$$\boldsymbol{v} = \boldsymbol{v^0} + \boldsymbol{a}t + \frac{1}{2}\boldsymbol{J}t^2 \tag{4.35}$$

Remember that the zero (upper) indexed variables refer to initial conditions.

Through the use of these equations, it is possible to determine the periods of acceleration and/or deceleration, tune motion performances or simply determine the position of the targeted system in any specific time instant (t).

4.3.2 Trajectory Evaluation for the Rhombic Vehicle

Before proceeding with the presentation of the method for the trajectory evaluation, consider the following assumptions:

- The trajectory evaluation task will be treated as a 2D problem. Even though, a 1D approach could also have been followed;
- The resulting set of trajectory points, will be given in equal number and keeping the same Cartesian coordinates of the inputed path points;
- It is considered the trapezoidal profile referred in Section 4.3.1;
- The output from the path optimization module is assumed to comply with the data structure presented in table 4.1:

Table 4.1: Expected output data for the optimization algorithm.

Symbol	Description
$\boldsymbol{p_F}$	Cartesian coordinates of the Front wheel path points
$\boldsymbol{p_R}$	Cartesian coordinates of the Rear wheel path points
$\boldsymbol{p_C}$	Cartesian coordinates of the Vehicle center path points
ϕ_F	angle between consecutive p_F points
ϕ_R	angle between consecutive p_R points
ϕ_C	angle between consecutive p_C points

For redundancy purposes, the trajectory evaluation is performed in all the three given references. From now on, and to simplify the incoming equations, consider only the general variables, $\boldsymbol{p}$ and ϕ.

The input data available for the trajectory evaluation module, are the reference path points:

$$\boldsymbol{p_j}, \tag{4.36}$$

with $j = \{1, ..., J\}$ and J being the number of reference path points.

The Equations 4.32 - 4.33 can be presented in a discrete form as follows. The velocity in each reference point, is updated according to the acceleration, which, to simplify the computation is kept fixed during the time interval (Δt_j), which is variable and not given as input.

$$\boldsymbol{v_{j+1}} = \boldsymbol{v_j} + \Delta t_j \boldsymbol{a_j}, \tag{4.37}$$

This is commonly referred to as the Euler approximation [URLEI], which was previously (in section 4.2.2.3) disregarded due to numerical issues.

The position of the reference points is updated by the average velocity during the time interval (Δt_j). Using the Trapezoidal rule [URLTR], which provides better accuracy than the simpler Euler approximation, and under the constant acceleration assumption, the average velocity during a time period is the average of its beginning velocity and ending velocity as follows,

$$\boldsymbol{p_{j+1}} = \boldsymbol{p_j} + \Delta t_j (\frac{\boldsymbol{v_j} + \boldsymbol{v_{j+1}}}{2}). \tag{4.38}$$

By substituting Equation 4.37 into Equation 4.38, one defines the updated position in terms of the previous step position, velocity, and acceleration as,

$$\boldsymbol{p_{j+1}} = \boldsymbol{p_j} + \Delta t_j \boldsymbol{v_j} + \frac{\Delta t_j^2}{2} \boldsymbol{a_j}. \tag{4.39}$$

At this point, we have all the math necessary to elaborate a routine for generating velocity profiles for the reference points given in table 4.1. The routine proposed is the following:

1. To include obstacle safety requirements, the speed at each reference point, here denoted as (s_j), is firstly defined as the linear function of the minimum distance (d_j) [2]. A maximum allowable speed (s_{max}) is set to this profile, so to integrate kinematic constraints;

$$s_j = \begin{cases} s_{min} & if \quad d_j < d_{saf} \\ \alpha(d) & if \quad d_{saf} < d_j < d_{th} \\ s_{max} & if \quad d_j > d_{th} \end{cases} \tag{4.40}$$

2. Each reference point is then updated with the corresponding velocity vector, which can be easily obtained through the following equation,

$$\boldsymbol{v_j} = s_j \langle \cos\phi_j, \sin\phi_j \rangle; \tag{4.41}$$

3. Consider zero initial condition for the velocity an acceleration: $\boldsymbol{v_0} = \boldsymbol{0}$ and $\boldsymbol{a_0} = \boldsymbol{0}$;
4. For $j = \{2, ..., J-1\}$, the following steps are iteratively performed
   (a) Evaluate Δt_j through Equation 4.38;
   (b) With Δt_j from (a), evaluate $\boldsymbol{a_j}$ using Equation 4.37;
   (c) If $a_j \in [a_{min}, a_{max}]$ (dynamical feasible transition) proceed to the next iteration from the step (a), otherwise, follow step (d);
   (d) Using the correct feasible acceleration (a_{min} or a_{max}) re-evaluate Δt_j using Equation 4.39;
   (e) Update the new velocity, $\boldsymbol{v_{j+1}}$, using Δt_j from (d) and Equation 4.37;
5. The evaluated profile is feasible if $\boldsymbol{v_J} = \boldsymbol{0}$. Otherwise the steps (a) to (e) shall be repeated using a reverse-time approach, i.e., $j = \{J, ..., 1\}$ and imposing $\boldsymbol{v_N}$ to the zero vector;

Figure 4.15 illustrates some of the main steps of this routine. The forward routine guarantees dynamic feasible transitions between points but is insufficient to guarantee $\boldsymbol{v_J} = \boldsymbol{0}$ (Figure 4.15-b). The backward routine overcomes this situation and assures $\boldsymbol{v_J} = \boldsymbol{0}$, and maintain all the transitions dynamic feasible

[2] d_j is measured between the vehicle pose relative to j^{th} reference point and the closest obstacle

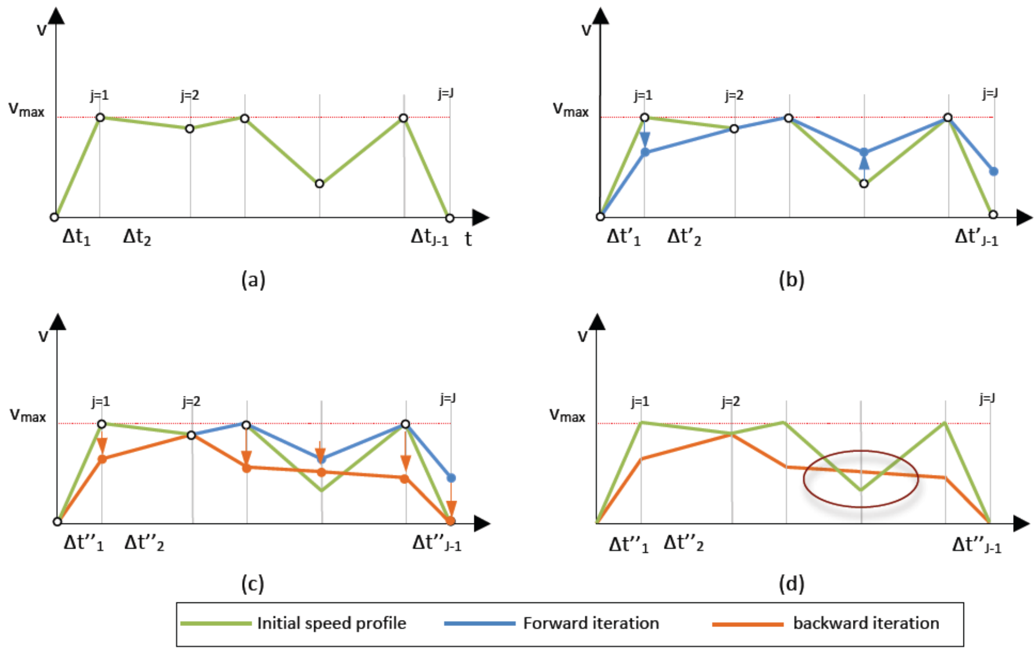


Figure 4.15: Trajectory evaluation: (a) initial speed profile based on safety-obstacle clearance requirements; (b) The forward routine guarantees dynamic feasible transitions between points but is insufficient to guarantee $\boldsymbol{v_J} = \mathbf{0}$; (c) The backward routine guarantees that all the transitions are dynamic feasible; (d) The evaluated speed profile may violate the safety-based speed profile.

(Figure 4.15-c). However, as it is depicted in Figure 4.15-d the generated profile may violate the safety requirements integrated in the speed profile s. This is an undesirable situation that can lead the vehicle to prohibitive velocities with respect to the obstacle proximity.

To sidestep this issue, it is proposed the following greedy solution; Suppose the existence of D violating points, with $D \subset J$. For each $p_d \in D$ the routine above described is repeated in the forward sense from the p_d to p_J ($j = \{d, ..., J\}$) and in the backward sense from p_d to p_1 ($j = \{d, ..., 1\}$) and assuming that $\boldsymbol{v_d} = \boldsymbol{s_d}\langle\cos\phi_j, \sin\phi_j\rangle$ and $\boldsymbol{v}$ given being the last obtained velocity profile. The process is repeated till $D = \emptyset$.

According to our experiments, whose data is later presented in Chapter 6, feasible trajectories can be achieved within a reduced number of iterations and generating a negligible computational overhead to the overall refinement planning strategy.

Chapter 5

Motion Planning under Differential Constraints

Chapter 4 handled the motion planning problem of the rhombic vehicle by following the classical and decoupled motion planning approach that first solves the basic path planning problem and then designs a velocity profile that satisfies the vehicle differential constraints. Optimization requirements where integrated in the solution plan through the employment of path optimization techniques.

In the present section, the motion planning problem is brought to an higher instance, where both the global (from obstacles) and the local (from vehicle kinematics and dynamics) constraints are considered in a unified approach that directly yields trajectory plans. To solve this trajectory planning problem that is formulated in Section 5.1 as a planning in the state space problem, different RRT-based planners are presented and discussed in Section 5.2. The most popular variations of the RRT method, the single and dual-tree RRT, are first extended to fit this new planning problem in Subsection 5.2.1. Despite their robustness, RRTs generate low quality solutions due to its random nature and usually perform poorly in constrained environments. For this reason, Subsection 5.2.2 discusses the implementation of the RRT-Blossom [KP06], a novel variation of RRT that is designed to perform well in highly constrained environments, and the Transition-Based RRT (diminished to RRT-T) [JCS08] in Subsection 5.2.3, which allows to conduct a search on continuous cost spaces. The main dissertation contribution of this chapter comprises Section 5.3 and consists on the combination of the RRT-Blossom and RRT-T into a single RRT planner that is capable of yielding both differential constraints and optimization requirements in its search process. The proficiency of this planner within the ITER motion planning framework is only later analyzed in Chapter 6.

5.1 Problem Formulation: Planning in the State Space

As presented in [LK00], the trajectory planning problem can be formulated in terms of the following six components:

1. **State Space:** A topological space, $\mathcal{X}$ that somehow serves the same purposes as $\mathcal{C}$ (introduced in Section 3.1). $\mathcal{X}$ defines the attainable states by the vehicle and it is defined as $\boldsymbol{x} = (\boldsymbol{q}, \dot{\boldsymbol{q}})$. Note that the state can include higher order derivatives if necessary, but such systems are not considered in the dissertation.
2. **Boundary values:** $\boldsymbol{x_I} \in \mathcal{X}$ and $\boldsymbol{x_G} \in \mathcal{X}$, which define the motion query.
3. **Inputs:** A set, $\mathcal{U}$ specifying the complete set of controls or actions that can affect the state. In the discrete form notation it is used $\mathcal{U}_d$.
4. **Incremental simulator:** Given the current state, $\boldsymbol{x}(t)$, and inputs applied over a time interval, $\{\boldsymbol{u}(t')|t \leq t' \leq t + \Delta t\}$, compute $\boldsymbol{x}(t + \Delta t)$.
5. **Collision detector:** A binary-valued function, $E : \mathcal{X} \rightarrow \{true, false\}$, that determines whether global obstacle constraints are satisfied from state $\boldsymbol{x}(t)$ to $\boldsymbol{x}(t + \Delta t)$.
6. **Metric:** A real-valued function, $\rho : \mathcal{X} \times \mathcal{X} \rightarrow [0, \infty]$, which specifies the distance between pairs of points in $\mathcal{X}$.

The trajectory planning problem is thus generally viewed as a search in the state space $\mathcal{X}$ for a continuous path from an initial sate $\boldsymbol{x_I}$ to a goal state $\boldsymbol{x_G}$ or goal region $\mathcal{X}_G \subset \mathcal{X}$. The set of global obstacle constraints are imposed in $\mathcal{X}$ and any solution trajectory must keep the set of states within this set. The collision detector reports whether a given state, x, satisfies the global constraints. This set, formerly defined as $\mathcal{C}_{free}$ for $\mathcal{C}$ is now defined as $\mathcal{X}_{free}$, i.e., the set of all states that satisfies the global constraints. The local differential constraints, aimed to be handled explicitly in this planning instance, are expressed in a differential state model or transition equation $\dot{\boldsymbol{x}} = f(\boldsymbol{x}, \boldsymbol{u})$ and also effected through the definition of a set of feasible inputs integrated on the system simulator. The trajectory planning problem can thus be defined as a search in a feasible state space $\mathcal{X}_{feas}$ (i.e., the space of feasible velocities and/or accelerations) rather than only in $\mathcal{X}_{free}$. The solution path is not directly expressed as as path through $\mathcal{C}_{free}$, as in Chapter 4, but is instead derived from an the action trajectory via integration of the state transition equation.

As overviewed in Subsection 3.1.2, the introduction of differential constraints in the motion planning problem emerges two different planning instances: (1) the nonholonomic planning, referring to problems that involve non-integrable constraints on the state velocities, in addition to the global constraints, and where $\boldsymbol{x} = \boldsymbol{q}$ (revisit, for instance, the state space model in (2.18)) for $\boldsymbol{q} \in \mathcal{C}$, and (2) the kinodynamic planning, which includes both velocity and acceleration constraints and the state, $\boldsymbol{x} \in \mathcal{X}$, is defined as $\boldsymbol{x} = (\boldsymbol{q}, \dot{\boldsymbol{q}})$ as in (2.33).

The experiments pursued in the dissertation fall in the former planning instance, which could, somehow, prevent the reformulation of the motion planning problem in Section 3.3. However, this new formulation eases the comprehension of the planners presented and developed in this chapter and serves as basis for future developments addressing the kinodynamic problem.

5.2 RRT-based planners

The trajectory planning problem presented above follows closely the incremental search paradigm, raising the interest on the employment of randomized and sampling methods such as the RRT to solve the trajectory design problem. Note that in general, it is more challenging to design a roadmap-based method (e.g., [KSLO96]) due to the increasing difficulty of connecting numerous pairs of states in the presence of differential constraints. Furthermore, very little can be achieved with combinatorial techniques in the context of differential constraints. In its turn, the RRT method, presented in Subsection 3.2.2 and explored in Subsection 4.2.1 for a holonomic path planning instance, was originally developed for handling differential constraints. This encouraged the examination of the performances of the RRT method to the specific motion planning problem of the rhombic vehicle.

5.2.1 Single and Dual-tree RRT

In fact, the Algorithm 1 and 2, are easily extended to the trajectory planning formulation previously presented as shown in the pseudocode in Algorithm 3 and 4, where:

- *pickControl*: selects the action control by trying all the possible inputs defined in $\mathcal{U}_d$ and choosing the one that yields a new state as close as possible (in term of ρ) to the target state, $\boldsymbol{x_{target}}$;
- *newEdge*: creates a new edge from the state $\boldsymbol{x}$ using the control input u for a single (Extend-variant) or maximal (Connect-variant) number of time steps;
- *sim*(x, u): computes the state $x(t+\Delta t)$of the vehicle after the application of the control input u to the state $x(t)$ (a constant time step is assumed in the dissertation);
- *failure*$(x(t), u, x(t+\Delta t))$: tests whether the transition from $x(t)$ to $x(t+\Delta t)$, using the control input u, incurs a collision (collision detection module) or violates the differential vehicle constraints.

Figure 5.1 shows how the reachability graph (i.e., the set of attainable configurations (or states) by the vehicle) of the rhombic vehicle[1] differ with the number of discretized control actions considered in $\mathcal{U}_d$.

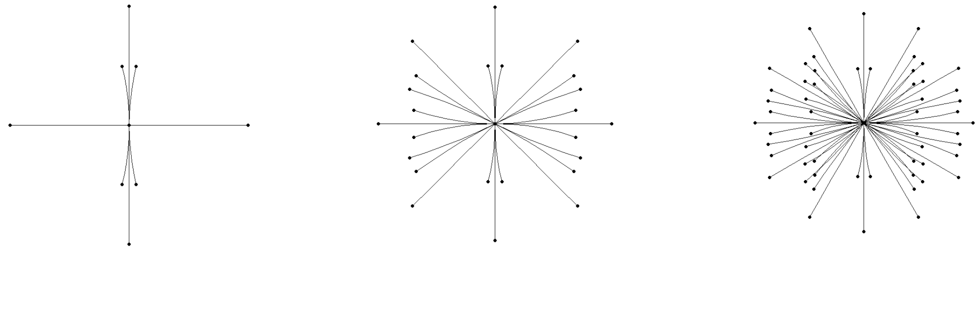


Figure 5.1: Reachability graph for the rhombic vehicle considering $|\mathcal{U}_d|$ equal to (from left to right): 18, 50 and 98.

[1]The reachability graph differs depending on the system being simulated.

Algorithm 3: single-tree RRT (under differential constraints)

function: $\tau \leftarrow$ `singleRRT`$(\boldsymbol{x_I}, \boldsymbol{x_G}, \mathcal{X})$
  $\tau \leftarrow$ `tree`$(\boldsymbol{x_I})$
  while $time \leq time_{max}$ **do**
    $\boldsymbol{x_{rand}} \leftarrow$ `randomConfig`$(\mathcal{C})$
    $\boldsymbol{x_{new}} \leftarrow$ `growTree`$(\tau, \boldsymbol{x_{rand}})$
    if $\boldsymbol{x_{new}} \wedge \rho(\boldsymbol{x_{new}}, \boldsymbol{x_G}) < \zeta$ **then**
      return `extractSol`$(\boldsymbol{x_{new}})$
    $\tau \leftarrow$ `tree`$(\boldsymbol{x_I})$
  return *failure*
function: `growTree`$(\tau, \boldsymbol{x_{target}})$
  $\boldsymbol{x_{near}} \leftarrow$ `nearestNeighbor`$(\tau, \boldsymbol{x_{target}})$
  $\boldsymbol{u_{best}} \leftarrow$ `pickControl`$(\boldsymbol{x_{near}}, \boldsymbol{x_{target}})$
  if $\boldsymbol{u_{best}}$ **then**
    $\tau \leftarrow$ `newEdge`$(\boldsymbol{x_{near}}, \boldsymbol{u_{best}})$[a]
  return $\boldsymbol{x_{new}}$
function: `pickControl`$(\boldsymbol{x}, \boldsymbol{x_{target}})$
  $d_{min}, \boldsymbol{u_{best}} \leftarrow \rho(\boldsymbol{x}, \boldsymbol{x_{target}})$
  for $\boldsymbol{u} \in \mathcal{U}_d$ **do**
    $\boldsymbol{x_{new}} \leftarrow$ `sim`$(x, \boldsymbol{u})$
    if `failure`$(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{x_{new}})$ **then**
      continue
    $d \leftarrow \rho(\boldsymbol{x_{new}}, \boldsymbol{x_{target}})$
    if $d_{min} < d$ **then**
      $d_{min}, u_{best} \leftarrow d, \boldsymbol{u}$
  return $\boldsymbol{u_{best}}$

[a] using the Extend variant

Algorithm 4: dual-tree RRT (under differential constraints)

function: $\tau \leftarrow$ `dualRRT`$(\boldsymbol{x_I}, \boldsymbol{x_G}, \mathcal{X})$[a]
  $\tau_a, \tau_b \leftarrow$ `tree`$(\boldsymbol{x_I})$,`tree`$(\boldsymbol{x_G})$
  while $time \leq time_{max}$ **do**
    $\boldsymbol{x_{rand}} \leftarrow$ `randomConfig`$(\mathcal{X})$
    $\boldsymbol{x_a} \leftarrow$ `growTree`$(\tau_a, \boldsymbol{x_{rand}})$
    if $\boldsymbol{x_a}$ **then**
      $\boldsymbol{x_b} \leftarrow$ `growTree`$(\tau_b, \boldsymbol{x_a})$
      if $\boldsymbol{x_b}$ **then**
        if $\rho(\boldsymbol{x_a}, \boldsymbol{x_b}) < \zeta$ **then**
          return `extractSol`$(\boldsymbol{x_a}, \boldsymbol{x_b})$
    $\tau_a, \tau_b \leftarrow \tau_a, \tau_b$
  return *failure*

[a] inherits `growTree` and `pickControl` from Algorithm 3

For illustrative purposes and to gain understanding of the RRTs within a nonholonomic philosophy, Figure 5.2 represents the construction process of a single-tree RRT framed at 20 (left), 100 (center) and 500 (right) samples.

5.2.2 RRT-Blossom

As shown later on the simulation results provided in Chapter 6, the RRT method, under differential constraints, can perform poorly (in terms of computational efficiency), specially on highly constrained environments. This is an unexpected result since highly constrained environments increase the geometric constraints and reduces the search space. The most difficult problems should be the ones that are neither

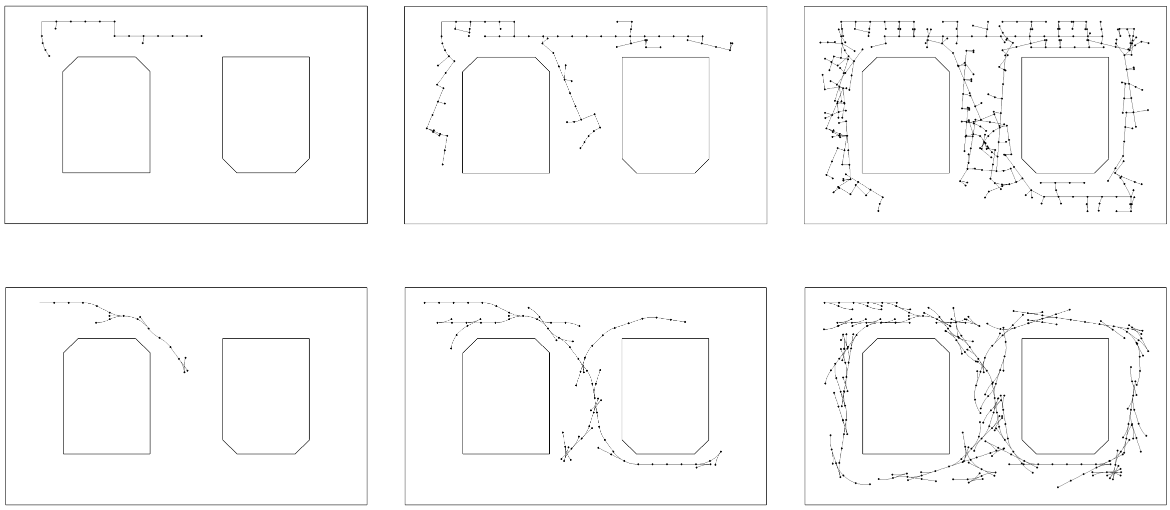


Figure 5.2: Single-tree expansion for $|\mathcal{U}| = 10$ (top) and $|\mathcal{U}| = 18$ (bottom).

highly-constrained, or highly under-constrained, but somewhere in the middle of these two extremes. In [KP06], a new RRT variant, called of RRT-Blossom, is presented for which the authors claimed an increased robust performance in highly constrained environments, while retaining the RRT's performance in under-constrained problems. This section explores the implementation of this variant in an attempt to increase the robustness of a RRT-based planner in a trajectory planning instance.

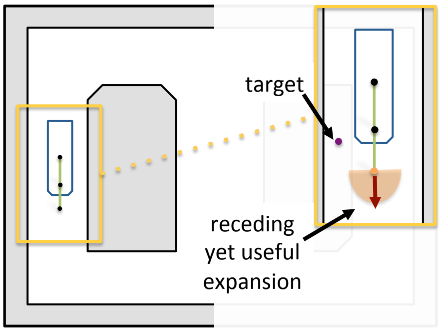
The RRT variations discussed so far share a very useful property that accounts for their "rapidly-exploring" characteristic, which is implicit in the basic method of construction. That is, newly added edges never regress into already explored areas. Although receding edges may seem undesirable because many of them will regress to already explored areas, some of these receding edges may be beneficial. For instance Figure 5.3-left, shows an example of a receding edge that could be useful to continue expanding the tree. Expanding such edges is beneficial because they provide tree growth in iterations that are otherwise wasted, which sustains the rate of node creation.

The RRT-Blossom introduces the idea of allowing tree expansions that recede from the target, but do not regress to areas that are already explored by the tree. Furthermore, the algorithm instantiates all the eligible edges[2] when expanding a node and not just the single best one, as depicted in Figure 5.3-right and Figure 5.4. According to [KP06], this "blossom-like" instantiation has little negative cost, since in constrained regions these expansions would eventually be instated anyhow, thus generating only negligible overhead. The regression constraint introduced in [KP06] is sustained by using an approximation, which assumes that a newly added edge is considered regressing if there is a node other than its parent that is closer to it. The Figure 5.3-right, excerpted from [KP06], explains the idea.

The pseudocode for the RRT-Blossom is presented in Algorithm 5. Note that the *nodeBlossom* function iterates over all controls and expands all the neighbor nodes that do not violate the collision detection (step 8 in Algorithm 5)) and regression constraints (step 10 in Algorithm 5).

As reported in [KP06], the regression constraint implemented in Algorithm 5 can be problematic for

[2]Hence the 'blossom" moniker.



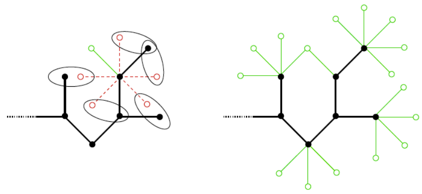


Figure 5.3: Left: example or a receding yet useful expansion. The orange half-disk shows the receding from target direction for the orange node. The red arrow indicates an useful expansion. Right: the left subfigure shows all possible extensions for a particular node; all the dashed expansions are regressing since other node other than parent are closer to them (indicated with loops). In the right subfigure, only the green edges are suitable for expansion.

nonholonomic systems as the rhombic vehicle studied in the dissertation. This is illustrated in Figure 5.4. The left path of the vehicle is expanded first, but all its following paths result in collision. Although the middle path would be feasible, it is disallowed because it would cause regression into space already explored by the left path.

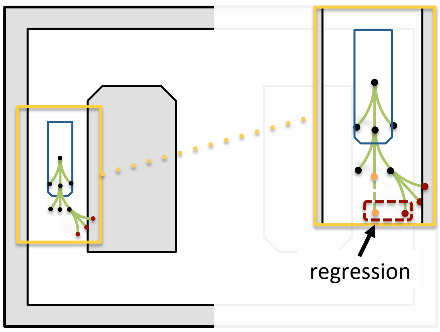


Figure 5.4: Regression constraint; the green-dashed expansion is blocked by an extant nonviable edge.

The solution presented in [KP06] to this problem is based on viability concept. A node is considered viable if the system can evolve from it indefinitely, or can reach the goal before failure. Therefore, given the viability status of each node beforehand, the problem can be solved by preventing nonviable nodes from blocking viable expansions in the tree. Unfortunately, the viability of edges is not known beforehand, so it must be deduced while expanding nodes of the tree. This can be achieved by implementing the method defined in the finite-state machine of Figure 5.5 (extracted from [KP06]). A remanning issue to be considered in this algorithm is the case where a dormant deadlock occurs in the tree. The author in [KP06] presents a sketched solution for it.

The implementation of the Algorithm 5 is straightforward, since it is a simple extension of the dual-tree RRT. However, the implementation of the viability conditions of Figure 5.5 was vaguely described in

Algorithm 5: RRT-Blossom

function: growTree($\tau, \boldsymbol{x_{target}}, \mathcal{X}$) $\quad \boldsymbol{x_{near}} \leftarrow$ nearestNeighbor($\tau, \boldsymbol{x_{target}}$)
  $\boldsymbol{x_{new}} \leftarrow$ nodeBlossom($\boldsymbol{x_{near}}, \boldsymbol{x_{target}}, \tau$)
  return $\boldsymbol{x_{new}}$

function: nodeBlossom($\boldsymbol{x}, \boldsymbol{x_{target}}, \tau$)
  for $\boldsymbol{u} \in \mathcal{U}_d$ **do**
    $\boldsymbol{x_{new}} \leftarrow$ sim($\boldsymbol{x}, \boldsymbol{u}$)
    if failure($\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{x_{new}}$) **then**
      continue
    if regression($\boldsymbol{x}, \boldsymbol{x_{new}}, \tau$) **then**
      continue
    $\tau \leftarrow$ newEdge($\boldsymbol{x}, \boldsymbol{u}$)
  return the new node closest to $\boldsymbol{x_{target}}$

function: regression($\boldsymbol{x_{parent}}, \boldsymbol{x_{new}}, \tau$)
  for $n \in \tau$ **do**
    if $\rho(n, \boldsymbol{x_{new}}) < \rho(\boldsymbol{x_{parent}}, \boldsymbol{x_{new}})$ **then**
      return True
  return False

[KP06]. The author of the dissertation proceed with its own understanding, which fortunately produced better results, increasing the RRT - planner performances. Figure 5.6 gives an example of the RRT - Blossom performance under the general map adopted in the dissertation.

5.2.3 Transition-based RRT

An issue from the employment of the Algorithms 3-5, is the poor quality of the resulting trajectories, as a consequence to the large number of points and fluctuations on the inputs. Conscious of this problem and on the emerging interest on the employment of randomized techniques, some authors start to consider the problem of trajectory refinement as a post-processing step to the trajectory planning phase [CSL00, LBL02]. Following a refinement strategy does not suit the purposes of this chapter, which aims to develop an unified trajectory planning approach that could lead to optimized trajectories. Hence, this approach is set aside for now.

[JCS08] presented an interesting RRT variant, the T-RRT, which combines the strength of the RRT algorithm (on rapidly growing random trees towards unexplored spaces) with the efficiency of optimization methods that uses transitions to accept or reject a new potential state. This variant is used in a novel RRT-based planner proposed in the next section, and for that reason it is summarily described in the following.

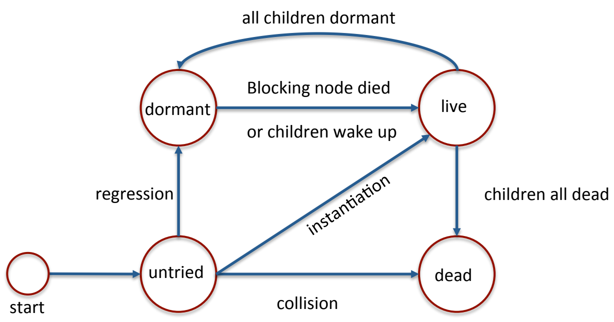


Figure 5.5: Finite-state machine for the evolution of the viability status of an edge.

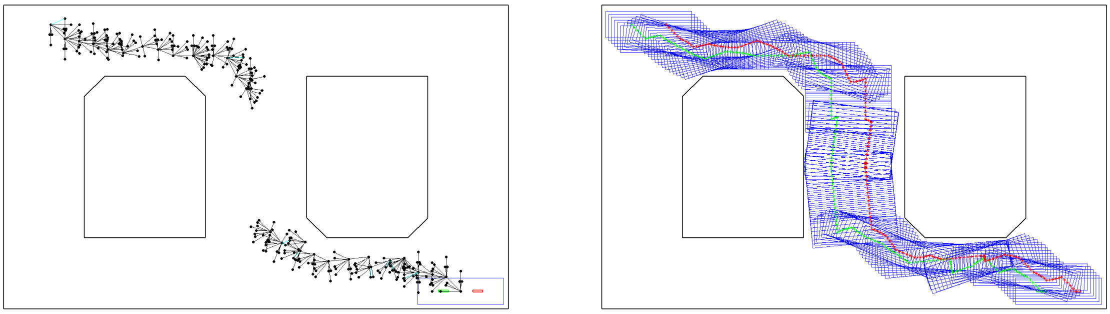


Figure 5.6: Left: RRT planner blossoming $\mathcal{C}_{free}$. Right: final generated path with front wheel path in red and rear wheel path in green.

Algorithm 6 presents the pseudocode for the T-RRT variant. The increment step ϵ on the *newEdge* function, in step 6, is chosen small enough to approximate well the cost variation between $\boldsymbol{x_{near}}$ and $\boldsymbol{x_{new}}$. While the *randomConfig* function in step 4 of Algorithm 4 ensures the bias towards unexplored free regions of the space, the goal of the *transitionTest* (steps 8-24 in Algorithm 6) is to filter irrelevant expansions regarding the search of low cost trajectories. Additionally *minExpCtrl* forces the planner to maintain a minimal exploration rate avoiding the blocking of the search process. In the following it is given a detail explanation of these two functions.

A. Transition Test

In the *transitionTest* function, starts by filtering the configurations whose cost is higher then a given threshold c_{max}. Then, a probability of acceptance of a new configuration is defined by comparing its cost c_j relatively to the cost of its parent c_i. This transition probability[4] is defined as

$$p_{ij} = \begin{cases} exp(\frac{\Delta c^*_{ij}}{K * T}) & if \quad \Delta c^*_{ij} > 0 \\ 1 & otherwise \end{cases} \tag{5.1}$$

where:

- $\Delta c^*_{ij} = \frac{c_j - c_i}{d_{ij}}$ defines the cost variation. Downhill transitions are automatically accepted (step 11 in Algorithm 6) whereas uphill transitions have lowest chance on being accepted;
- K is a constant that can be used to normalize the cost;

[4] Using the Metropolis criterion

Algorithm 6: Transition-based RRT

function: `growTree`(τ,$\boldsymbol{x_{target}}$)[3]

  $\boldsymbol{x_{near}} \leftarrow$ `nearestNeighbor`($\tau, \boldsymbol{x_{target}}$)

  $u_{best} \leftarrow$ `pickControl`($\boldsymbol{x_{near}}, \boldsymbol{x_{target}}$)

  if u_{best} **then**

    if `transitionTest`($c_{near}, c_{new}, d_{near-new}$) **and** `minExpCtrl`($\tau, \boldsymbol{x_{near}}, \boldsymbol{x_{rand}}$) **then**

      $\tau \leftarrow$ `newEdge`($\boldsymbol{x_{near}}, u_{best}$)

return $\boldsymbol{x_{new}}$

function: `transitionTest`(c_i, c_j, c_{ij})

  if $c_j > c_{max}$ **then**

    return False

  if $c_j < c_i$ **then**

    return True

$p = exp(\frac{-(c_j - c_i)/d_{ij}}{K * T})$

  if $rand(1) < p$ **then**

    $T = T/\kappa$

    nFail $= 0$

    return True

else

    if $nFail > nFail_{max}$ **then**

      $T = T * \kappa$

      $nFail = 0$

    else

      $nFail = nFail + 1$

    return False

- T is the "temperature" that controls the difficulty level of transition tests; Low temperatures limit the expansion to low positive variations of cost while high temperatures permit expansions to on higher cost variations;

Also note that the *transitionTest* function performs an adaptive tuning of the temperature parameter using nFail, the number of consecutive the criterion defined in (5.1) discards an expansion (see step 15 and 20 in Algorithm 6).

B. Minimum Expansion Control

The adaptive tuning of the temperature above described can present a side effect, which consist on a a too slow expansion of the tree and in a excessive refinement. The *minExpCtrl* function forces the planner to keep exploring new regions by controlling the ration between exploration and refinement steps; A new state generate $\boldsymbol{x_{new}}$ is considered to participate in a refinement step when $\rho(\boldsymbol{x_{near}}, \boldsymbol{x_{rand}})$ is less that ϵ and on the contrary, supposed to participate in a exploration step when this distance exceeds ϵ. Thus, *minExpCtrl* acts by rejecting explorations that make the ratio between exploration and refinement mode lower that a given minimal value. Figure 5.7, nicely illustrates the impact of this function on the tree search process.

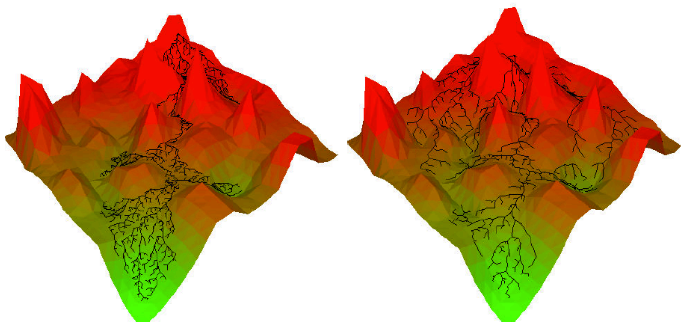


Figure 5.7: Impact of the *minExpCtrl* function on the RRT algorithm; Without control the tree tends to refine the search, which slows down the planner performances (left). With the expansion control activated the planner is forced to explore new regions of the space (right).

It remains to define the cost function c that measures the cost associated with the tree expansion from $\boldsymbol{x_{near}}$ to $\boldsymbol{x_{target}}$. Considering the optimization criteria presented in Section 3.3, it was formulated the following cost function

$$c = w_c \cdot \frac{1}{C} + w_s \cdot S + w_d \cdot D, \tag{5.2}$$

where, C measures the clearance, S measure the rotational smoothness and D measures the distance or length, of the tree transition and are defined as follows. Assume that the expansion, from $\boldsymbol{x_{near}}$ to $\boldsymbol{x_{target}}$, involves J sampled poses. Therefore, the total clearance of this expansion can be defines as

$$C = \sum_{j=1}^{J} C(\boldsymbol{x_j}), \tag{5.3}$$

where

$$C(\boldsymbol{x_j}) = min(\|\boldsymbol{u_{j,i}}\|), \tag{5.4}$$

with $u_{i,j}$ defined in (4.16).

The smoothness of the tree expansion is given by

$$S = \sum_{j=1}^{J-1} \|\mathcal{F}(\theta_{j+1} - \theta_j)\|, \tag{5.5}$$

where $\mathcal{F}$ represents a function that normalizes the twisting angle in $[-\pi, \pi]$. S only accounts for the rotational part of the motion between $\boldsymbol{x_{near}}$ and $\boldsymbol{x_{target}}$.

Finally, D measures the distance travelled between the two configurations and is defined as

$$D = \rho(\boldsymbol{x_{near}}, \boldsymbol{x_{target}}), \tag{5.6}$$

with ρ denoting the metric assumed in (4.6).

In (5.2), w_c, w_s and w_d scale the relative weight of each cost function component.

5.3 A Robust RRT Planner for Continuous Cost Spaces.

Experiments performed with the T -RRT in Algorithm 6, demonstrate that this variant has serious difficulties on solving the nonholonomic planning problem of the rhombic vehicle, requiring unreasonable running times to solve the motion queries. Intuitively the T- RRT variant could not perform well based on the simple original RRT search structure since it successively discards instantiations in order to find costly acceptable expansions. On the other hand, the RRT - Blossom revealed a good proficiency, with respect to computational aspects (see the reported data in Chapter 6). This encouraged the development of a combined planner that could merge the ideas of both RRT - Blossom and T-RRT, as to achieve a robust planner that could also met optimization requirements. In the dissertation this new combined variant is referred to as T-B -RRT. The changes proposed are best framed on Algorithm 7 and with the modified finite-state machine in Figure 5.8.

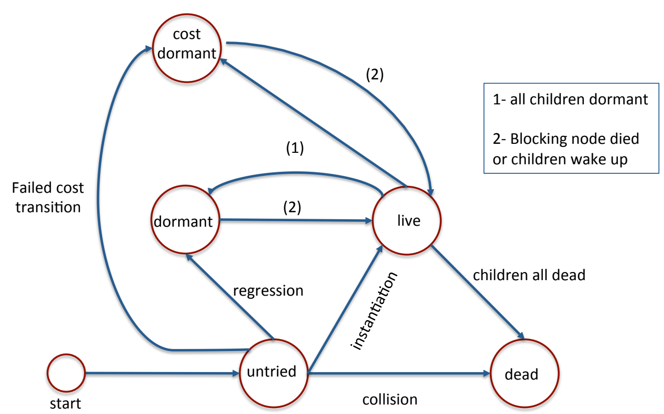


Figure 5.8: Finite-state machine for the T-B - RRT.

Figure 5.9-top shows the T-B - RRT planner exploring the $\mathcal{C}_{free}$ with the objective of finding a smooth path, while in the example on the bottom, the same planner performs on a more balanced fashion on

Algorithm 7: T-B - RRT

function: `growTree`$(\tau, \boldsymbol{x}_{\boldsymbol{target}}, \mathcal{X})^5$

  $\boldsymbol{x}_{\boldsymbol{near}} \leftarrow$ `nearestNeighbor`$(\tau, \boldsymbol{x}_{\boldsymbol{target}})$

  $\boldsymbol{x}_{\boldsymbol{new}} \leftarrow$ `nodeT-Blossom`$(\boldsymbol{x}_{\boldsymbol{near}}, \boldsymbol{x}_{\boldsymbol{target}}, \tau)$

  return $\boldsymbol{x}_{\boldsymbol{new}}$

function: `nodeT-Blossom`$(\boldsymbol{x}, \boldsymbol{x}_{\boldsymbol{target}}, \tau)$

  for $\boldsymbol{u} \in \mathcal{U}_d$ **do**

    $\boldsymbol{x}_{\boldsymbol{new}} \leftarrow$ `sim`$(\boldsymbol{x}, \boldsymbol{u})$

    if `failure`$(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{x}_{\boldsymbol{new}})$ **then**

      continue

    if `regression`$(\boldsymbol{x}, \boldsymbol{x}_{\boldsymbol{new}}, \tau)$ **then**

      continue

    if `transitionTest`$(c_{near}, c_{new}, d_{near-new})$ **then**

      continue

    $\tau \leftarrow$ `newEdge`$(\boldsymbol{x}, \boldsymbol{u})$

  return the new node closest to $\boldsymbol{x}_{\boldsymbol{target}}$

an attempt to improve both clearance and smoothness of the path. These preliminary results show the capacity of this innovative RRT-based planner on entailing optimization concerns during the search for a solution path. But it also anticipates some troubles on annulling the oscillations inherent to the random nature of the RRT algorithm.

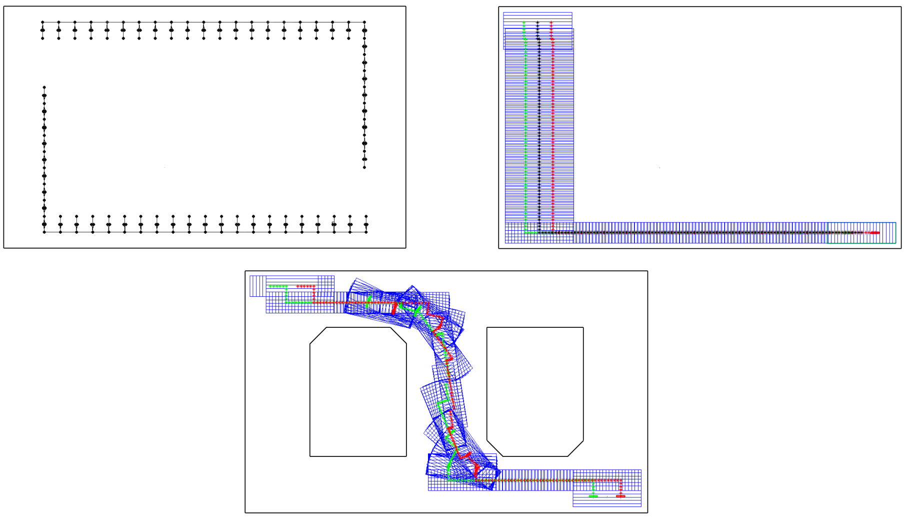


Figure 5.9: Top: T-B - RRT searching for a smooth path ($w_s = 1, w_c$ and $w_d = 0$) on a unobstructed environment; Bottom: solution obtained with $w_s = 1$, $w_c = 0.8$ and $w_d = 0.2$ for the general map.

Chapter 6

Experiments and Simulation Results

The dissertation focused on aspects related with the motion of a heavy and large RH vehicle that will operate in the confined buildings of ITER.

The unusual kinematic rhombic configuration adopted for this vehicle, which is barely described in the literature, sustained a pioneer analysis on aspects related with its motion, including the development of simulation models that follow different levels of abstraction. Unfortunately, the author of the dissertation has experienced some difficulties on correctly implement the dynamic model presented in Section 2.3. This eventuality truly compromises the presentation of relevant results concerning the mathematical modeling of the vehicle. In truth to be told, one of the major objectives was to compare the vehicle behavior of the vehicle under a kinematic and dynamic perspective on an attempt to understand and assess the influence of dynamic phenomena on the performances of the vehicle. For this reason, this chapter does not include results concerning the developments presented in Chapter 2 and they are postponed as future work.

However, the main focus of this dissertation was the development of planning strategies that could tackle the specific motion planning problem of the rhombic vehicle within the ITER framework.

Along the dissertation, some simulations results were presented, mainly due to illustrative purposes and to ease the understanding of the concepts presented. This chapter presents in a more comprehensive way the set of performed experiments conducting a careful analysis on the gathered results. The chapter is roughly organized in the same way as the presented concepts. It conducts a performance analysis on the involved planning techniques regarding pertinent issues and computational efficiency. Notwithstanding, the main focus will be given to the quality and feasibility of the obtained solutions concerning the optimization requirements specified in Section 3.3.

6.1 Simulation Setup

The set of conducted experiments and provided illustrations were developed in the MATLAB platform. In particular, it was used a Matlab-based software tool, the Trajectory Evaluator and Simulator (TES), which was developed under the grant F4E-2008-GRT-016 (MS-RH) [RVR+10] and includes the planners developed in Chapter 4 and 5 as well the kinematic model defined in Chapter 2.

Since the main concern of the dissertation is to investigate the ability of the developed planners to generate feasible and optimized trajectories under the scope of the RH operations in the ITER, the defined experimental setup gives preference to ITER simulated environments, more specifically to the levels L1 of the TB and HCB represented in Figure 6.1. Illustrative maps are also included and are defined on the button to illustrate each problem and to provide an extended and more generalized discussion.

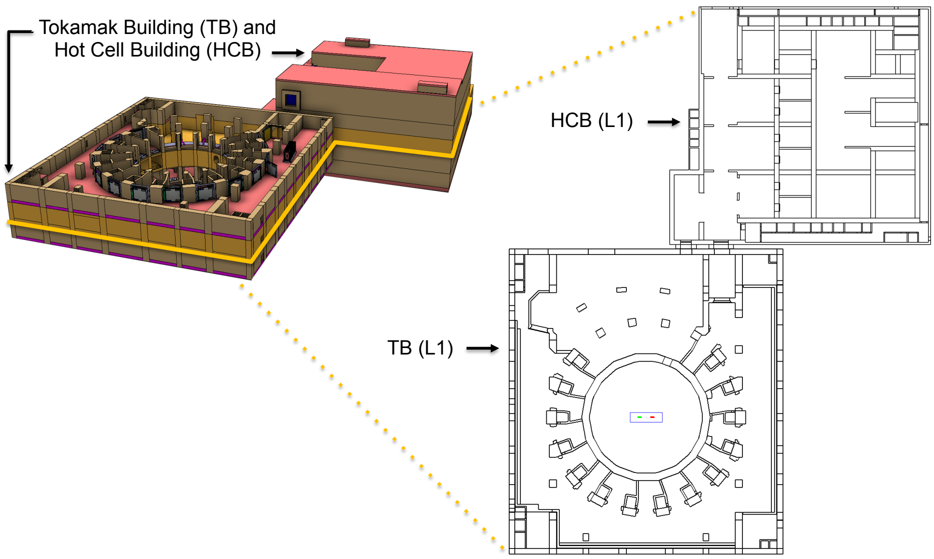


Figure 6.1: Impression of the original 3D CATIA models of the ITER buildings (left) and assumed 2D projections of the selected L1-levels of the TB and HCB.

Table 6.1 resumes the vehicle configuration assumed for the most part of the experiments (recall the system definition in Section 2.1), concerning geometric parameters. Additional vehicle configurations are promptly referred, when necessary.

Table 6.1: Vehicle geometric parameters.

Parameter	M	M_F	M_R	L	W
Value [m]	3.4	1.7	1.7	8.5	2.62

To evaluate the proficiency of the proposed planning techniques, three scenarios are presented entailing different motions queries, which are depicted in Figure 6.2 and described as follows:

- **Scenario I**: given the starting configuration at the lift in the TB, the vehicle shall dock inside a specific Vacuum Vessel Port Cell. The mission requires the vehicle to move in a highly confined space, in particular in the entrance of the VVPC;
- **Scenario II**: the rhombic vehicle leaving the lift from the TB, must dock in a specified DP inside the HCB, with a specific orientation. This mission involves passing through a narrow port that gives entrance to the DP room;
- **Scenario III**: the queried motion consists in a simple and general operation between two arbitrary

configurations on an illustrative map. Note hat the vehicle must start and finish its motion with opposite directions.

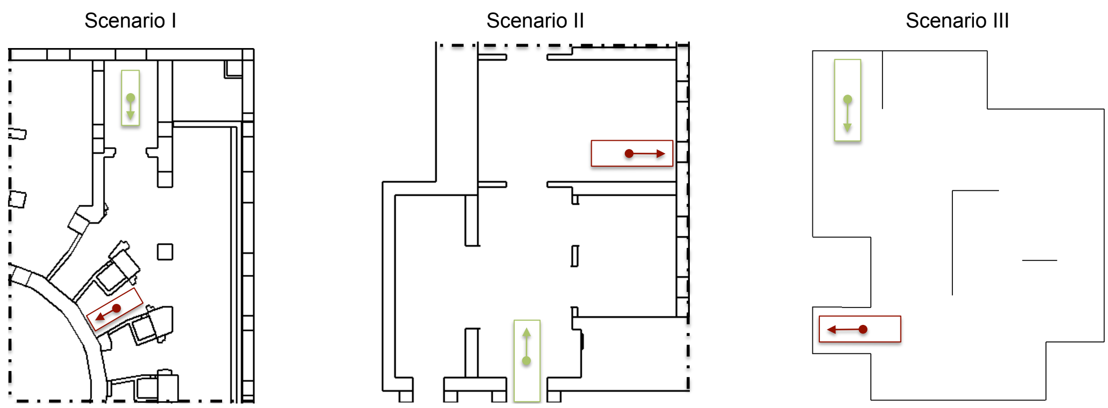


Figure 6.2: Motion query for: a nominal operation in the TB (left), a docking procedure in the HCB (center) and for a general mission between two arbitrary configurations (right).

6.2 Refinement Strategy

Two refinement approaches were presented in Chapter 4: (1) a combinatorial approach relying on the combination of the CDT and the A^* with an optimization technique based on the elastic bands concept; (2) a randomized approach that couples the RRT method with a path optimization technique that deforms and optimizes the path by explicitly considering the vehicle geometry.

The experiments are partitioned according to each planning stage (geometric path evaluation, path optimization and trajectory evaluation) sustaining a comparison between the two approaches.

6.2.1 Geometric Path Evaluation

1- CDT combined with the A^*

The evaluation of the geometric path in the first approach does not raise any computational issues due to complete nature and computational efficiency of both the CDT and A^* methods. Therefore no references are given to timing results. Figure 6.2 illustrates the triangular CD yielded by the CDT, the shortest triangle cell sequence returned by A^* (on the top) and the respective geometric paths (on the bottom). Note that at this stage, and following this combinatorial approach, the generation of collision-free paths is not guaranteed. The geometric paths generated by the A^* are in fact collision-free if considering a point-like robot approach. However when sampling the spanned area by the vehicle envelope, as in Figure 6.3-bottom, this is no longer verified. Moreover, the fact that the motion query is defined through starting and ending points for the vehicle wheels, rather then the complete vehicle poses, complicates the achievement of the correct vehicle positioning, which is required, for instance, in docking procedures.

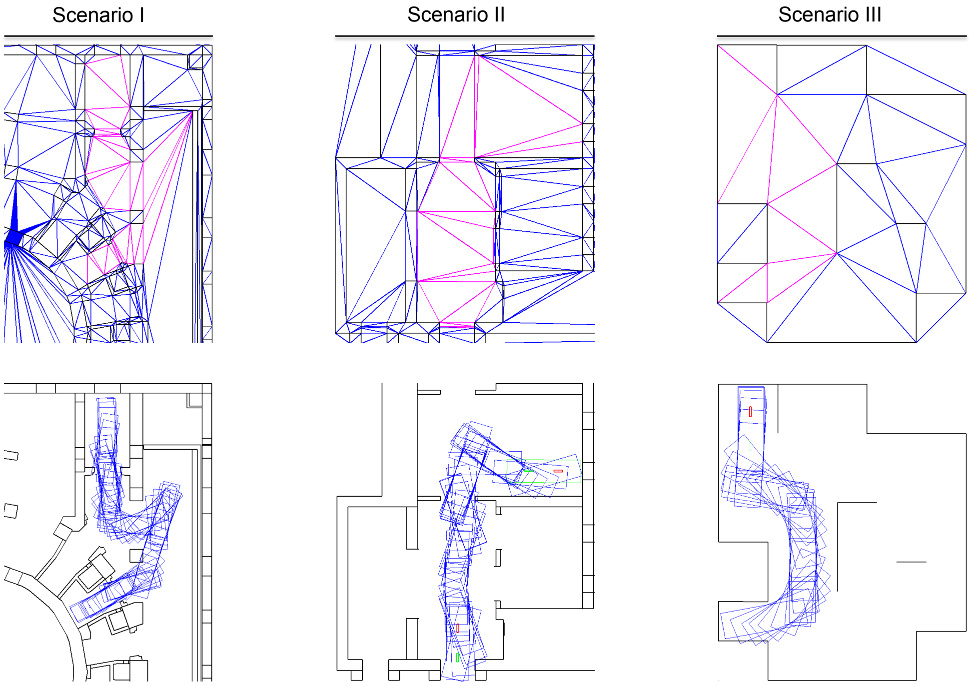


Figure 6.3: Top: Triangular CD (in blue) and shortest cell sequence (in magenta) evaluated by the A^* algorithm for Scenario I (left), II (center) and III (right). Bottom: Spanned area by the vehicle over the geometric path.

2- RRT-based approach

The second approach relies in a randomized sampled-based approach, the RRT method, which brings the computational efficiency into play. Section 4.2 presented the single-tree and dual-tree holonomic RRT as possible solutions for the geometric path evaluation, anticipating a better performance of the dual-tree version (Algorithm 3). The experiments performed in the dissertation corroborate this fact as shown by the run-time results in the boxplot of Figure 6.4. The dual-tree presents a clear advantage over the simpler single-tree in all the three analyzed scenarios. The sampled runs resulting in a timeout situation, i.e., when the planner does not solve the motion query within the specified time limit, are excluded from the averages. Therefore, the data presented for scenario I, which registered 42 timeouts over the 50 runs with the single-tree RRT method, represents an underestimate of the true computational cost. This confirms the difficulty of the single-tree RRT on solving motion queries on highly confined spaces such as the TB.

Figure 6.5 shows a geometric path retuned by the dual-tree RRT planner for the motion query of scenario I. A feature that stands out, is the collision-free status of the obtained geometric path, which contrasts with the results in Figure 6.3. Another important detail is the precise starting and arrival of the vehicle to the desired goal configurations. As discussed in Subsection 4.2.1, these are inherent properties of the RRT method, which explicitly considers the vehicle geometry during the search for a solution to the motion query. Apart from the low quality solution in unison with the former approach, the RRT method presents two main drawbacks:

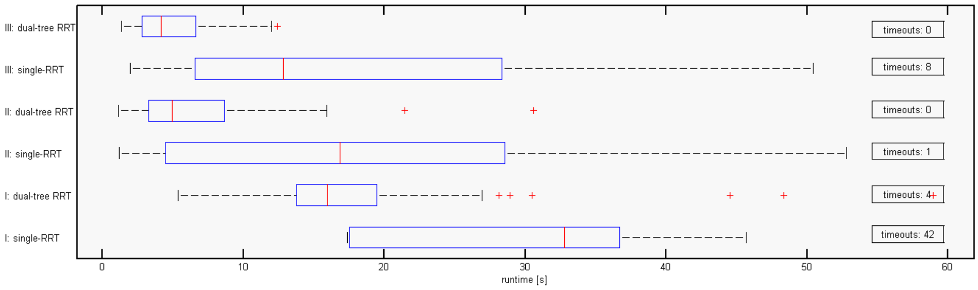


Figure 6.4: Single and dual-tree RRT algorithm runtimes in seconds, over 50 runs and considering a maximum runtime of 60 seconds.

- the fact that the solution path may contain unnecessary maneuvers for the arrival at the goal configurations. This issue is well illustrated in Figure 6.5 at the exit of the lift and entrance of the VVPC;
- the computational cost, which is no longer negligible as in the case of the CDT plus A^*; Although, the runtimes are kept acceptable as show in figure 6.4, when using the dual-tree RRT.

From now on, the dual-tree RRT is considered as reference for the geometric path evaluation, under the randomized approach.

Figure 6.5: Geometric path yielded by the dual-tree RRT planner for a motion query defined in the TB; Spanned area by the vehicle in blue and path described by the center of the vehicle in black.

6.2.2 Path Optimization

Both the combinatorial and randomized-based approaches shown to be insufficient to guarantee the generation of feasible paths that can be followed by the vehicle. The provided techniques are good at delivering an initial solution, but they must be coupled with more sophisticated methods as to return higher quality solutions. For this purpose, two path optimization methods were developed on a path deformation basis. The main idea is to improve the clearance over the obstacles, shorten and smooth the obtained geometric paths. Despite grounded on the same optimization requirements, recall from the beginning of Chapter 4, that each method was designed with different assumptions, with the particularity that the second (randomized-based) bridges some gaps of the former (combinatorial-based) one.

To assess the performances of each path optimization method a common data framework is defined based on the following three measurements:

- **Clearance**: the clearance for a given pose configuration (defined explicitly or sampled) over the path, is defined as the minimum distance between the vehicle at that pose and the obstacles, i.e,

$$C(\boldsymbol{q_j}) = min(\|\boldsymbol{u_{j,i}}\|). \tag{6.1}$$

  The total and average clearance, C and Γ_C, are defined for a path with J samples as,

$$C = \sum_{j=1}^{J} C(\boldsymbol{q_j}), \quad \Gamma_C = \frac{C}{J}. \tag{6.2}$$

  The bad clearance (BC) is evaluated over a path with

$$BC = \sum_{j=1}^{J} C_{bad}(\boldsymbol{q_j}), \tag{6.3}$$

  where $C_{bad}(\boldsymbol{q_j}) = C_{min} - C(\boldsymbol{q_j})$, if $C(\boldsymbol{q_j}) < C_{min}$, otherwise C_{bad} equals zero. In the dissertation, and following ITER specifications, C_{min} is set to 0.3 meters.

- **Length**: path length is defined as the distance travelled by the vehicle center throughout the path. This measure is decomposed in a translational component,

$$LT = \sum_{j=1}^{J-1} \|(\boldsymbol{p_{j+1}} - \boldsymbol{p_j})\|, \tag{6.4}$$

  and in a rotational component,

$$LR = \sum_{j=1}^{J-1} \|\mathcal{F}(\theta_{j+1} - \theta_j)\|, \tag{6.5}$$

  where $\mathcal{F}$ represents a function that normalizes the twisting angle in $[-\pi, \pi]$.

- **Smoothness**: smoothness can be quantified resorting to the path derivate. Using a simple forward finite difference to approximate the derivate in pose $\boldsymbol{q_j}$ both for the translational and the rotational components, as

$$ST(\boldsymbol{q_j}) = \|\boldsymbol{p_{j+1}} - \boldsymbol{p_j}\| \tag{6.6}$$

  and

$$SR(\boldsymbol{q_j}) = \|\mathcal{F}(\theta_{j+1} - \theta_j)\|, \tag{6.7}$$

  the average and standard deviation for the translational and rotational smoothness of the path, denoted as Γ_{ST}, Γ_{SR}, Λ_{ST} and Λ_{SR}, are evaluated.

1- Elastic Bands Approach

For the ensuing experiments consider the parameters in table 6.2.

Table 6.2: Reference parameters for the elastic bands approach.

Scenario	K_e	K_r	d_{max}	F_{max}	iterations
I	0.4	0.1	1.2	1	80
II	0.3	0.1	1	1	50
III	0.2	0.1	1	1	130

The optimization process, using the elastic bands approach, is well represented in Figure 6.6. The initial rough geometric path, which is not collision-free for the presented scenarios, is continuously deformed (center column) through the action of repulsive and elastic forces that act on the path points. For the specific case of scenario I, the path deformation was represented through the evolution of the path points. This prompts the fact that the deformation acts upon the path points and not on vehicle poses; As explained in Appendix E, the collision-free status of the path is achieved through implicit considerations on the vehicle geometry. Although, the two others impressions are presented showing directly the vehicle occupancy area making the optimization process more perceptible. This iterative process leads to a shorter, smoother and safer path, as illustrated in Figure 6.6 -right and supported by the data presented in table 6.3.

An attentive analyze of table 6.2 reveals a significative increase of the path clearance, less relevant on scenario III, which is explained due to the reduced space on the approaching of the goal configuration. The shortening of the path is more evident on the rotational component which, together with ΓSR ends up to be a good measure for the effectiveness of this approach on handling smoother paths.

According to the experiments performed, should any local minima occur in the elastic band iterative process, this does not trap the robot in any fixed configuration. Moreover, smoothness is not compromised and the output of this optimization is still a shorter, smoother and safer path (as illustrated in Figure 6.6-right).

The main shortcoming of this approach arrises from the restriction that obliges both vehicle wheels to follow the same, which naturally reduces the maneuvering ability of the vehicle and consequently the resulting evaluated paths. Also the optimization is unable to solve the issue of the correct positioning of the vehicle as shown by the initial (in green) and goal (in red) sampled configurations of the vehicle.

2- The Rigid Body Dynamics Approach

For the optimization method based on the rigid body dynamics, the same data framework is used. The relevant algorithm parameters are given in table 6.4 for the gathered results herein presented.

The obtained geometric paths with the dual-tree RRT algorithm, are depicted in Figure 6.7-left. As previously reported in Subsection 4.2.1, these initial solutions are collision-free paths that despite their efficiency, cause the vehicle to achieve the desired goal configuration. The center column of Figure 6.7

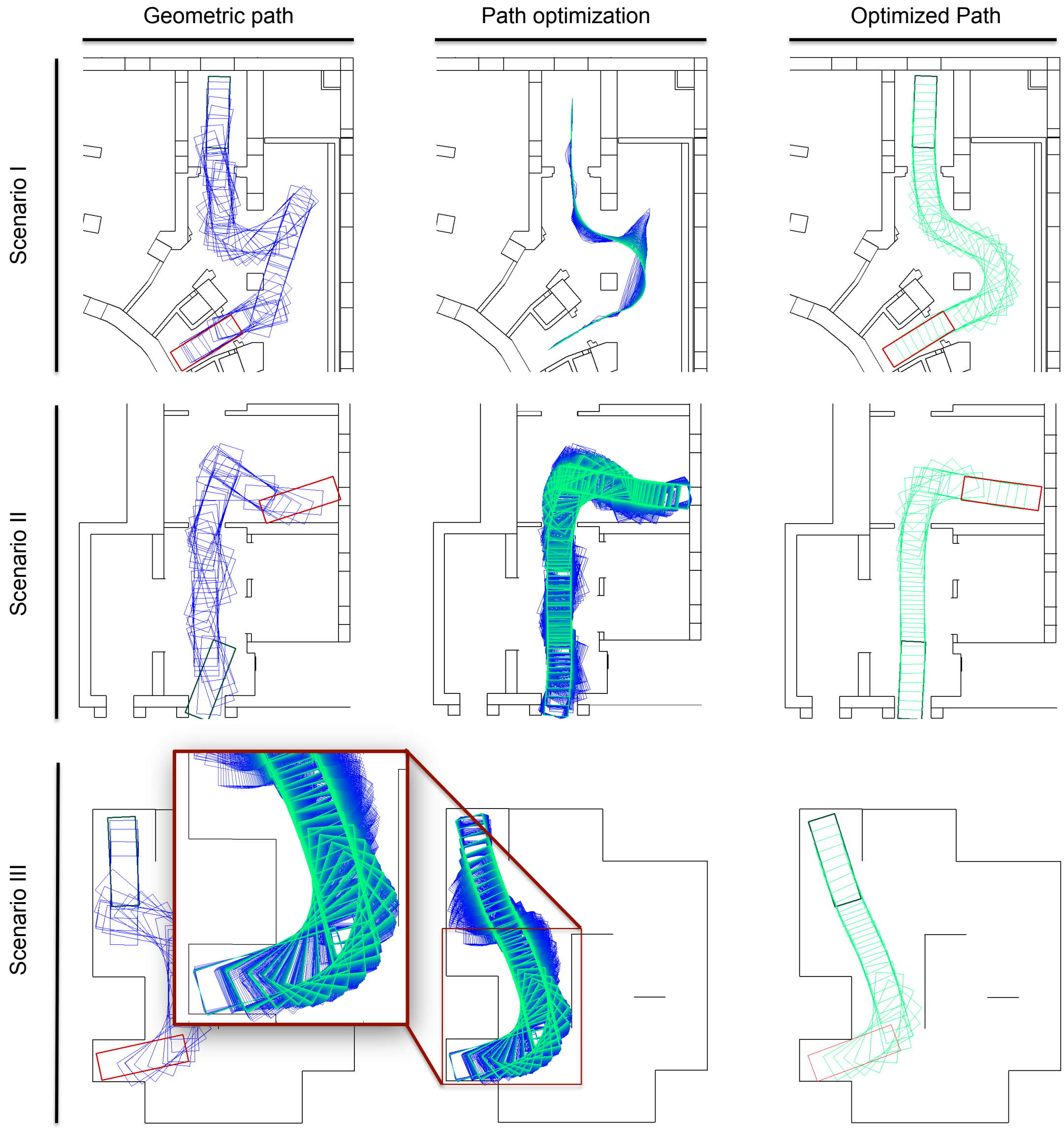


Figure 6.6: Path optimization with elastic bands: the rough and low quality paths (left) are optimized through a deformation process leading to the generation of smoother, shorter and safer paths.

provides an impression of the evolution of the consecutive vehicle poses along the path. Each pose treated as a rigid body is repealed from the obstacles iteratively increasing the overall path clearance. Moreover, the elastic and torsional interactions maintain the poses connected an ensure smooth transitions between vehicle poses.

The data presented in table 6.5 reveals the effectiveness of this optimization method, with all the reference measurements concurrent to the same traits: reduction of path looseness with the increase of the translation and rotational smoothness, path shortening, annulling the randomness of the RRTs and finally the augment of path clearance.

Figure 6.8 proposes an additional scenario referring to a parking operation of the rhombic vehicle, in the HCB of ITER. The environment is less constrained when compared to the previous examples but it

Table 6.3: Path optimization performance for the elastic bands approach.

Scenario	Path	Clearance			Length		Smoothness			
		C	BC	ΓC	LT	LR	ΓST	ΓSR	ΛST	ΛSR
I	Geometric	36.18	0.42	0.86	42.70	6.37	0.75	0.11	0.37	0.13
	Optimized	67.30	0.11	1.07	36.71	3.98	0.59	0.06	0.44	0.05
II	Geometric	29.82	0.87	1.07	32.30	6.15	0.83	0.16	0.38	0.22
	Optimized	67.72	0.18	1.69	29.26	1.94	0.73	0.05	0.46	0.08
III	Geometric	31.25	0.49	1.25	23.27	4.41	0.86	0.16	0.42	0.19
	Optimized	32.08	0.04	1.19	21.53	1.83	0.77	0.07	0.28	0.08

Table 6.4: Reference parameters for the rigid body dynamics approach

Scenario	K_e	K_t	K_d	d_{max}	F_{max}	iterations
I	3	40	1	2	1	90
II	1	10	1	1	1	215
III	1	20	1	1	1	220

Table 6.5: Path optimization performance for the rigid body dynamics approach.

Scenario	Path	Clearance			Length		Smoothness			
		C	BC	ΓC	LT	LR	ΓST	ΓSR	ΛST	ΛSR
I	Geometric	25.91	2.44	0.51	43.8	6.85	0.89	0.14	0.23	0.12
	Optimized	43.88	0.12	0.88	36.70	3.60	0.75	0.07	0.19	0.06
II	Geometric	27.21	3.75	0.53	43.47	2.75	0.87	0.06	0.28	0.08
	Optimized	82.65	0.06	1.62	32.40	1.69	0.65	0.03	0.17	0.03
III	Geometric	26.19	1.71	0.82	24.65	10.42	0.80	0.34	0.32	1.10
	Optimized	31.67	0.17	0.99	22.46	8.21	0.72	0.27	0.14	1.10

is filled with different parked vehicles that play the role of obstacles for for the mission purposed. The proposed example evidences the ability of this method (combined with the RRT algorithm) on exploring the high maneuverability of the rhombic vehicle, a key trait on parking maneuvers, but it also stresses two important issues: (1) The path reaches a state equilibrium that is loosely shaped by the initial geometric path in shown in Figure 6.8 and hence the optimized path may include unnecessary maneuvers for the achievement of the final goal; (2) The external forces can continuously push the poses, locally stretching the path. This phenomenon makes the optimized path less smooth on some parts and can weaken the condition of a free-collision path (see Fig. 6.8-c). So far, this issue has been managed by a careful tuning of the spring gain values, Ke and Kt in equations (4.14) and (4.15).

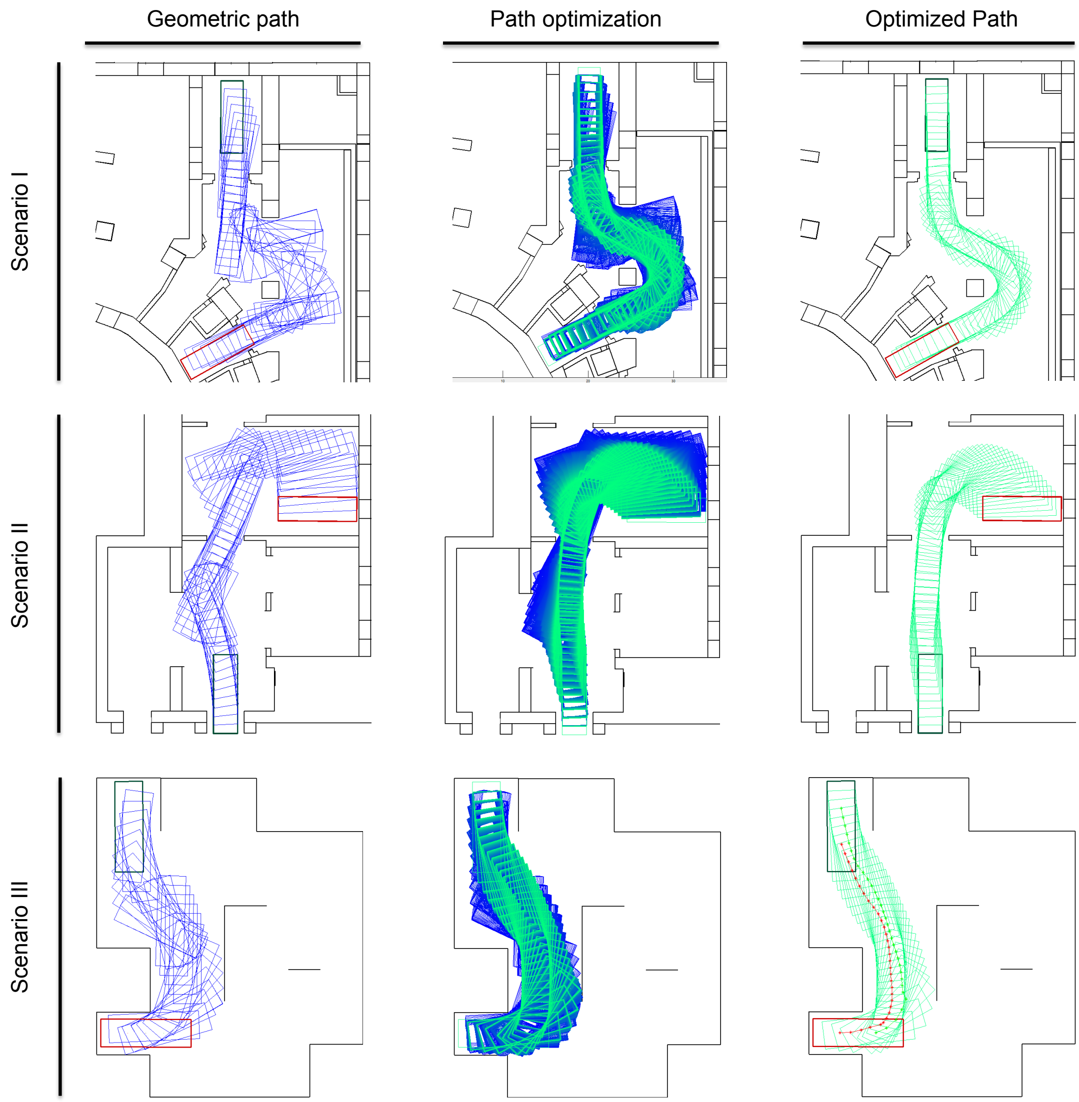


Figure 6.7: Geometric path yielded by the dual-tree RRT planner for a motion query defined in the TB; Spanned area by the vehicle in blue and path described by the center of the vehicle in black.

6.2.3 Trajectory Evaluation

The final step of the above referred refinement approaches transforms the optimized paths into trajectories, by associating a velocity profile to the paths. As referred in Section 4.3 this would be carried out in agreement with both obstacle clearance and vehicle dynamic constraints. For the following experiments consider the parameters in table 6.6.

Figure 6.9 refers to a vehicle journey that must be accomplished to a specific VVPC and for which an optimized path was generated using the the combinatorial-based approach, as illustrated in Figure 6.10. The upper graphic in Figure 6.9 shows the evolution of the path clearance, i.e., the minimum distance from the vehicle to the obstacles, as a function of the sampled poses. The provided data confirms that the path is feasible in terms of safety constraints, since the clearance trough the path is over the minimum safety distance, which was set to 0.3 meters. The second plot refers to the speed profile evaluated based on the

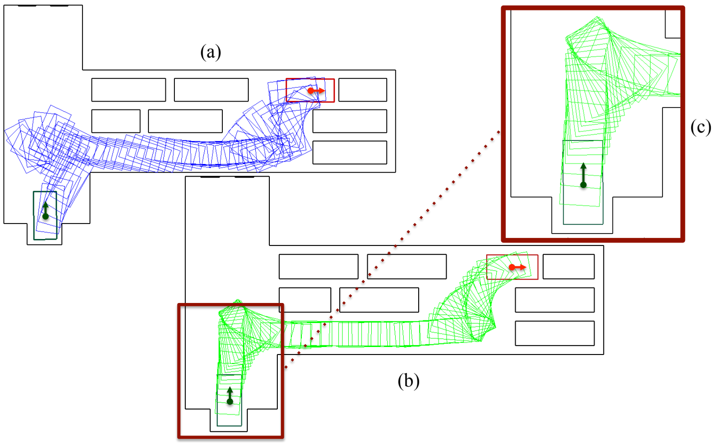


Figure 6.8: (a) Provided geometric path by the RRT algorithm. (b) Optimized path. (c) Local stretching of the path.

Table 6.6: Relevant parameters for the trajectory evaluation.

d_{saf} $[m]$	d_{th} $[m]$	s_{min} $[m.s^{-1}]$	s_{max} $[m.s^{-1}]$	a_{min} $[m.s^{-2}]$	a_{max} $[m.s^{-2}]$
0.3	1	0.05	0.5	0.01	-0.01

obstacle clearance, therefore, resembling the clearance path evolution. This speed profile does account for vehicle kinematic constraints integrating the maximum allowable speed on the wheels s_{max}. This (clearance-based) profile violates the admissible maximum accelerations for the vehicle as it illustrated in the third plot. To handle vehicle dynamic constraints, the specific routine described in Subsection 4.3.2 iterates the clearance-based speed profile in order to achieve dynamic feasible transitions. As depict in the last plot, this operation leads to small corrections in the speed profile, which for the specified example only happen at the beginning and end of the journey.

A final consideration to Figure 6.10, which nicely illustrates the speed evolution along the optimized path. Apart from the starting acceleration and final deceleration phases that inherit slow motions, the speed is also reduced at the exit of the lift and entrance of the VVPC due to the space confinement. This journey is performed by the vehicle in 227 seconds ($\sim 3,7$ minutes).

6.3 General Considerations

Quantitative parallels between the combinatorial and refinement approach are considered no worthy regarding the different guidance assumptions in which they are based. However it is convenient to present a summary and qualitative evaluation of the two approaches as provided next.

A - Combinatorial Approach:

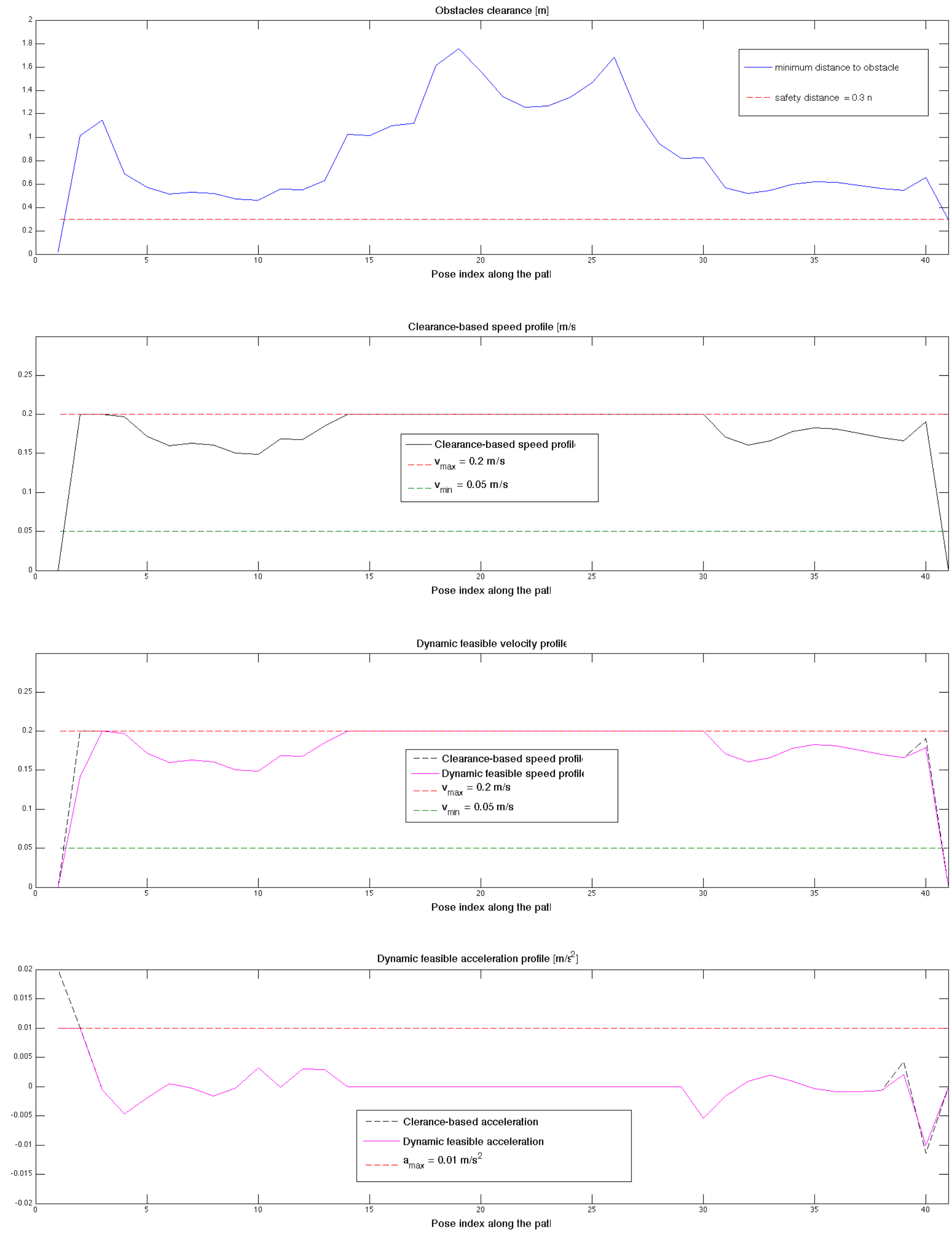


Figure 6.9: Trajectory evaluation: evolution of path clearance (upper plot); Speed profile based on the path clearance (second plot); Speed profile with and without vehicle dynamic constraints (third plot); Acceleration profile with and without dynamic considerations (bottom plot).

- **Pos**: The geometric path evaluation is fast and does not raise computational issues; The overall approach is compliant with a guidance constraint, which entails that both wheels should follow the same path. The combinatorial approach is thus suitable for line guidance approaches.

- **Cons**: The geometric path evaluation method is unable to handle motion queries that involve the specification of the vehicle orientation. As a consequence, the final optimized path does not guarantees that the vehicle starts or finishes its motion with the desired $\boldsymbol{q_I}$ and $\boldsymbol{q_G}$, respectively. The optimized path does not explicitly involves the vehicle geometry, which is only indirectly

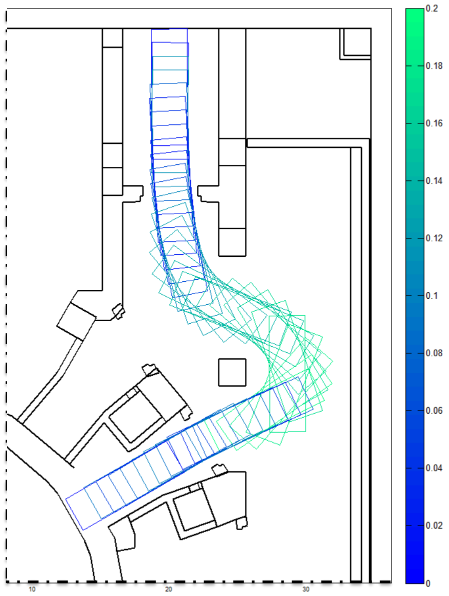


Figure 6.10: Speed vehicle map for a journey to a VVPC.

consider on the evaluation of the repulsive forces. The compliance with line guidance requirements limits the overall approach, which is incapable of profiting from the full rhombic vehicle capabilities.

B - Randomized Approach:

- **Pos**: The RRT algorithm handles the geometric path evaluation by explicitly solving the specified motion queries in terms of a set of vehicle configurations connecting $\boldsymbol{q_I}$ to $\boldsymbol{q_G}$. Benefiting from the en explicit definition of the vehicle motion, the optimization algorithm fully explores the rhombic vehicle maneuvering ability and considers directly the vehicle geometry during the path deformation. Also weak differential constraints, such as path smoothness, are guaranteed. The outputted optimized trajectories from this approach consist of two independent references for each vehicle wheel to follow or alternatively a reference for the vehicle center (which also includes the vehicle orientation). Thus, this approach is more suitable for a free-roaming approach, where the vehicle is able to track virtual trajectories.

- **Cons**: The inherent randomness of the RRT algorithm reflects on the low quality paths, which may inclusive contain unnecessary movements for the achievement of $\boldsymbol{q_G}$. Therefore, the optimization algorithm that post processes these solutions, may not be able to break down these undesired motions ending up to be loosely shaped by the initial geometric path. Additionally, the local minima situations may appear weakening the collision-free status of the path.

6.4 Trajectory Planning

So far, the vehicle differential constraints have not been explicitly integrated in the planning phase. In fact, the planning approaches whose experiments were presented in Section 6.2 have decoupled the big problem of motion planning into smaller problems that are easier to solve relying on weaker guarantees that the differential constraints are satisfied. This section is dedicated to the evaluation of the performances of the RRT-based planners developed in Chapter 5 which explicitly handle vehicle differential constraints.

Using the same experimental setup as in Section 6.2 (with respect to the chosen scenarios) and to gain some understanding on the complexity of trajectory planning, in the following it is proposed a comparison between the single and dual-tree RRT algorithms[1], with average computational performances presented in table 6.7. The data in the "NN" and "CC" refer to the number of nearest neighbor searches and checks for collisions with obstacles, respectively. The rows shown in *italics* indicate test cases for which the averages presented represent significant underestimates of the true costs, consequence of the higher number of runs that leads to a timeout situation.

Table 6.7: Average computational performances over 50 runs for the single and dual-tree RRT (nonholonomic version). $runtime_{max} = 120s$ and $|\mathcal{U}_d| = 25$.

Scenario	Algorithm	Runtime	Iterations	CC	NN	nodes	timeouts
I	single-tree RRT	–	–	–	–	–	50
	dual-tree RRT	63.94	43.63	7094.81	92.61	541.50	14
II	single-tree RRT	*60.74*	*117.01*	*20285.68*	*117.01*	*703.55*	*28*
	dual-tree RRT	29.23	156.36	20192.57	312.71	1878.28	1
III	single-tree RRT	*54.36*	*183.54*	*35197.15*	*1102.23*	*1102.23*	*37*
	dual-tree RRT	54.13	88.23	15220.41	176.31	1059.7	2

The first remark goes to the leading performances of the dual-tree RRT which, as in the holonomic case reported in Subsection 6.2.1, is more effective on solving the proposed motion queries for each scenario. The evidences are found in the smaller running times of this algorithm combined with an increased number of runs that succeed on finding a solution. Notice, however, the difference between these running times and the ones reported for the holonomic dual-tree RRT in the boxplot of Figure 6.4. This gives a good idea of the higher complexity of the trajectory planning problem.

To further increase the RRT planner performance, the RRT - Blossom variant was implemented (see Algorithm 5) . The average performances given in table 6.8 shown a clear gain in time and rate of success of the Blossom planner, registering zero timeouts. Of particular note is how the the RRT - Blossom reduces the number of collision and nearest neighbor checks. The resulting augment in the number of the tree nodes is due to the instantiation of all eligible edges during the search process (recall, for instance,

[1]The implementation includes some ideas discussed in [CL01], as to avoid keeping tracking of unsuccessful edge expansions.

Figures 5.3 and 5.4).

Table 6.8: Average computational performances over 50 runs of the dual-tree RRT - Blossom. $runtime_{max} = 120s$ and $|\mathcal{U}_d| = 25$.

Scenario	Runtime	Iterations	CC	NN	RC	nodes	timeouts
I	13.53	30.56	6522	58.74	576.56	7344.50	0
II	36.11	86.50	18886.10	167.73	1882.81	2096.82	0
III	20.12	56.20	13441.94	111.62	1475.34	13954.50	0

As to assess the performances of the RRT planners on handling quality paths, in particular considering the three established optimization measurements, i.e., clearance, length and smoothness of the solution plan, Chapter 5 presents another RRT variant, the T-RRT. The experiments performed with this variant led to poor results with the planner frequently failing to find a solution for the maximum runtime considered (120 seconds). This allows to conclude that if no improvements are made to the original algorithm, which was presented in [JCS08] for a holonomic planning problem, the suitability for nonholonomic problems, involving expensive numerical integration and collision detection, is seriously compromised.

To overcome this computational issue, Chapter 5 ends up proposing a newer RRT-based planner that combines ideas from both the RRT - Blossom and T - RRT. Table 6.9 refers to the computational performances of the T-B -RRT showing that the planner, which integrates the same optimization concerns of the T-RRT, is able to bring the running times to reasonable levels clearly outperforming the T-RRT and reaching performances comparable to the dual-tree RRT. This is major improvement. When compared to the performances of the original blossom variant, Table 6.9 shows a decrease of performance, which is coherent with the inclusion of the optimization considerations.

Table 6.9: Average computational performances over 50 runs of the T-B - RRT. $runtime_{max} = 120s$, $|\mathcal{U}_d| = 25$, $w_{cl} = 1$, $w_l = 1$ and $w_s = 1$.

Scenario	Runtime	Iterations	nodes	timeouts
I	37.48	27.04	6697	4
I	63.45	53.57	13143	10
I	55.23	67.76	16816	9

The optimization performances of this new planner are compared with the ones obtained with the RRT - Blossom in table 6.10. As previewed in Section 5.3 with the presentation of some preliminary results, the optimization concerns integrated in the T-B - RRT variant endows the planner with some capabilities to improve the solution quality. The data in table 6.9 apparently corroborates this fact by presenting a notable increase of the path clearance. However, since the clearance (C) is a cumulative value that depends on the number of poses along the path, this consideration is only valid if considering that in both experiments, the number of poses is approximately equal. Unfortunately, this cannot been confirmed since the number of poses was not stored during the experiments. If in on hand, there is no reason to believe on the existence of a relevant difference in this number (the planners are similar in their algorithm nature and used the same performance values (e.g., ϵ and $|\mathcal{U}_d|$)) the values presented for LT and LR suggest a poor performance of the planner in terms of length optimization which not complying with the expectations. The poor performance in the length optimization may in fact be related with an increased number of poses in the solutions of the T-B - RRT planner. The only true evidence in Table 6.9 is ΓC, which is an average value and increases in all the three experiments, supporting the idea of a real improvement on path clearance. The remaining data in Table 6.9 does not allow to present

Table 6.10: Average optimization performances of the RRT - Blossom and T-B - RRT. $runtime_{max} = 120s$, $|\mathcal{U}_d| = 25$, $w_{cl} = 1$, $w_l = 1$ and $w_s = 1$.

Scenario	Algorithm	Clearance			Length		Smoothness				Timeouts
		C	BC	ΓC	LT	LR	ΓST	ΓSR	ΛST	ΛSR	
I	RRT - Blossom	66.21	2.95	0.50	39.69	4.93	0.3	0.04	0.11	0.03	0
	T-B - RRT	132.5	1.39	0.86	43.93	5.43	0.28	0.04	0.13	0.03	4
II	RRT - Blossom	61.89	7.13	0.48	37.21	4.48	0.29	0.03	0.13	0.04	2
	T-B - RRT	167.40	2.19	1.27	40.17	3.89	0.31	0.03	0.12	0.03	10
III	RRT - Blossom	73.89	5.18	0.55	30.11	12.74	0.30	0.09	0.12	0.61	0
	T-B - RRT	186.46	2.21	1.06	46.57	16.89	2.75	0.10	0.13	0.59	9

good considerations on the performances of this planner on the adopted confined scenarios suggesting some limitations of this planning strategy to generate optimized trajectories. Figure 6.11 confirms this fact, by presenting the generated solution of the T-B - RRT to the motion query in scenario I. As it can be observed, the generated trajectory reveals some optimization concerns but it is far from achieve the optimization performances, in terms of path clearance and path smoothness, of the solutions provided in Section 6.2.

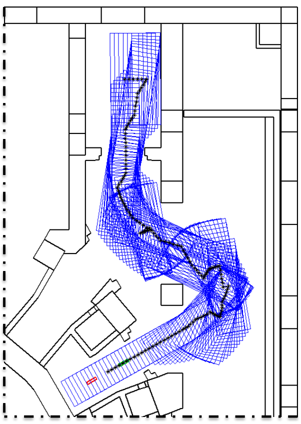


Figure 6.11: Trajectory generated by the dual-tree T-B - RRT planner for the motion query defined in the TB.

Chapter 7

Conclusions and Future Work

The dissertation addresses the motion planning problem stated for an autonomous rhombic vehicle that will operate in the buildings of the ITER. For that purpose, two distinct strategies were followed with the objective of finding feasible and optimized plans for the rhombic vehicle motion inside the confined spaces of those buildings.

First, a refinement strategy is proposed which divides the motion planning problem in three different and successive stages: the geometric path evaluation, the path optimization and the trajectory evaluation. In its turn, the second strategy follows a more direct and unified trajectory planning philosophy that enables to directly output trajectory solutions as solution for the motion planning problem.

In the former strategy two different approaches are followed that satisfy different guidance requirements for the vehicle. These requirements entail that the planned solutions must be either compliant with the constraint that both wheels must follow the same path or with an alternative solution, which accepts that independent references can be given for the vehicle wheels (and achieve an increased maneuvering flexibility). In what concerns the geometric path evaluation and optimization, the two approaches consider different planning and optimization techniques.

The combinatorial approach chooses the CDT as the method to provide a triangle CD and derive a search graph for the specified motion queries. To hasten the search for a solution this method is coupled with the A^* algorithm, which allows to quickly evaluate the geometric path even for complex scenarios such as the TB in ITER. The fact that these methods follow a point-like robot approach, with the solution constituting a single path to be followed by the wheels, restrict the definition of the motion queries to the Cartesian points of the vehicle initial and goal configurations. Moreover, the solution returned is not guaranteed to be collision-free for a rigid body such as the rhombic vehicle and weak differential requirements are not included (e.g. path smoothness). Notwithstanding, the approach is complete, which means that for any motion query, it either finds a solution or correctly reports that no solution exists.

To increase the path quality a post optimization method is applied to the geometric path solution, that is based on a path deformation technique called of elastic bands. The dissertation works this concept to achieve an innovative deformation scheme that is capable of integrating the geometric constraints (in a implicit way) of the vehicle during the path deformation and still comply with the line guidance constraint on the wheels. The inclusion of maneuvers is also considered within this planning and optimization approach. Note that none of these issues was addressed in related studies. Gathered simulated results with this approach show that it can be used to determine feasible optimized paths in the very confined scenarios of ITER. The outcome of this work is currently available in [FVVR10].

To tackle the second guidance requirement, the second approach relies on a randomized technique, the RRT, for the evaluation of the geometric path. This planning technique allows to explicitly determine the vehicle start and goal configuration in the motion query and it generates a collision-free path fully describing the vehicle motion in terms of consecutive poses. Additionally, maneuvers are automatically included as the solution to solve the orientation constraint imposed on both the initial and goal configuration. The main drawback of the algorithm is the fact that it relies on a weaker notion of completeness, which does not guarantee that a solution if found for a given finite time thus raising computational issues. However, simulated results shown that the dual-tree RRT is able to find a solution in a reasonable amount of time. An extra issue is the inclusion of unnecessary maneuvers on the geometric solution path. To handle this problem, the author of the dissertation proposes the idea, as future development, of developing a pruning technique that he thoughts that can be developed based on a graph-search - like problem, combining the RRT with the A^* algorithm.

As for the case of the combinatorial geometric stage, this randomized approach is also not able to provide quality solutions in what concerns the smoothness, length and clearance of the path. To overcome this inconvenient and fully explore the rhombic vehicle capabilities the dissertation proposes a novel path optimization technique, which based on both the elastic bands and rigid body concepts, is able to explicitly include the vehicle geometry on the path deformation and fully profit from the increased maneuverability of the rhombic vehicle. This is possible due to the fact that the proposed method evades the common approach that formulates paths as particle-systems and path deformation as a pseudo static simulation. Instead it dynamically simulates the motion of the consecutive poses of the path as rigid body systems bringing the vehicle geometry into play. Gathered results show the good proficiency of this method on handling feasible an reliable paths in cluttered scenarios such as those in ITER generating much more clearer, smoother and shorter paths when compared to the initial rough geometric ones.

The fully determine the vehicle motion, the dissertation proposes a trajectory evaluation method that is capable of combining both the proximity of the vehicle to the obstacles and kinematic and dynamic constraints of the vehicle. This method is compliant with any of the above referred path planning and optimization approaches.

These two approaches developed under the refinement planning strategy allow to bridge the gap between planning and execution phases providing, in a off-line instance, optimized trajectories that will certainly ease the task of real-time collision avoidance systems. The main shortcoming of these approaches is the fact that they are closely related to project requirements. For instance, the second approach, which is strongly linked to the rhombic capabilities and therefore its extension to other configurations is compromised. The former approach is suitable for other configurations that too use line guidance approaches, even though the satisfaction of the kinematic constraints must be assessed.

For future developments, the author recommends or foresees the necessity of including a dynamic tuning feature that is capable of optimizing the gains associated with the deformation process (K_e, K_r) for the combinatorial approach and (K_e, K_t) for the randomized one, which are required to be tuned on time depending on the map and vehicle geometry used. So far this issue has been managed manually for each studied scenario. In addition this tuning feature shall consider the stretching phenomenon reported in Figure 6.8, for instance adapting locally different gains wherever the band stretch is abnormal.

The outcome from this work was reported in two scientific articles [FVVR10, FVVR11] and provided useful background work on the area of optimization of trajectories for the rhombic vehicle, supporting the accomplishment and won, by Instituto Superior Técnico (IST), of two F4E grants in the area of RH, in which the author of the dissertation participates as full active fellowship student.

During the path and optimization process, the refinement planning strategy above referred does not take into consideration the differential constraints of the vehicle. This means that certain portions of the outputted paths may not be feasible for the vehicle to follow. The trajectory evaluation stage tries to handled this issue by imposing limits on the actuators. However, if simulating a trajectory tracking with the outputted trajectories as open loop laws, this would cause the appearance of some slightly deviations of the ideal path (or trajectory) whenever the limits of the control inputs are reached.

To handle this issue and further explore the nonholonomic instance of the rhombic vehicle motion planning problem, Chapter 6, proposes to follow a trajectory planning philosophy by means of the use of the RRT algorithm. Some RRT-based planners are presented always with the purpose of increasing the computational performance of the planner and include optimization concerns during the search for a solution. The chapter ends up by proposing a novel variant for the RRT algorithm, named of T-B - RRT that outperforms the T-RRT [JCS08] in computational issues and the RRT-Blossom [KP06] in what concerns the quality of the path, the two variants in which the new planner drew inspiration on. Unfortunately, the gathered results on the chosen confined scenarios did not provide satisfactory results in what concerns the quality of the final trajectory. In fact, the poor quality solutions resulting from the fluctuations on the inputs and the large number of nodes in the tree, is hard to annul even when considering optimization issues. However, it should be pointed out that this second planning strategy was not so explored as the first one and requires further studies, for instance, assess the interference of the cost function weights and other parameters associated with the SA optimization algorithm in the performance of the planner.

Considering all what was stated above, it can be concluded that the decoupled and refinement approach is more appealing because it provides much more reliable results in what concerns the quality of the solutions and therefore it must be adopted as the main reference for future incoming activities concerning the evaluation of optimized trajectories for the rhombic vehicle. Notwithstanding, the results obtained with the RRTs in the trajectory planning strategy shall not be ignored.

The dissertation's author believes that the most promising direction for this problem shall pass through the combination of the exploration strength of the RRT algorithm, for instance, the RRT - Blossom variant, with trajectory optimization methods (note the difference with path optimization). This way, it is possible to conceive a planning refinement strategy able to tackle both the vehicle differential constraints, handled during the RRT search and optimization requirements included during the trajectory optimization process.

Now considering the mathematical modeling part of the dissertation, a further step involves the accomplishment of effective simulation experiments integrating the dynamic model of the rhombic vehicle. As underlined along the thesis, these experiments will provide a valuable outcome to understand how dynamic phenomena affect the vehicle performances. For instance, it would be interesting to simulate the vehicle following an optimized trajectory generated with the planning techniques developed in the dissertation, using both the kinematic and dynamic model and appreciate the differences. These experiments will determine the greater or less urgency on developing a 3D dynamic model capturing the overall vehicle behavior, including suspension, load transfer and why not, air cushioning phenomena.

Bibliography

[ACB91] B. D'Andrea-Novel , G. Campion and G. Bastin, "Modelling and control of nonholonomic Wheeled Mobile Robots", in Proceedings Of the 1991 Institute of Electrical and Electronics Engineers (IEEE) International Conference on Robotics and Automation (ICRA), pp. 1130-1135, 1991.

[ADS02] N.M. Amato, K.A. Dill and G. Song, "Using motion planning to Study Protein Folding Pathways", Journal of Computational Biology, vol. 9(2), pp. 149-168, 2002.

[AKC04] D. Aarno, D. Kragic and H.I. Christensen, "Artificial potential biased Probabilistic Roadmap Method", In Proceedings of the IEEE International Conference on Robotics and Automation, vol.1 , pp.461 - 466, May 2004.

[AP98] M.U. Ascher and L.R. Petzold, "Computer methods for ordinary differential equations and differential-algebraic equations", Society for Industrial and Applied Mathematics, 1998.

[AS02] N. M. Amato and G. Song, "Using motion planning to Study Protein Folding Pathways", Journal of computational Biology, vol.9, no.2, pp.149-168, 2002.

[BJdW02] F. Beer, R. Johnston and DeWolf, "Mechanics of Materials", McGraw Hill, 2002. KZ95,BJdW02

[BK00] R. Bohlin and L.E. Kavraki, "Path planning using lazy PRM", In Proceedings of the International Conference on Robotics and Automation. vol. 1, pp. 521-528 2000.

[BL91] J. Barraquand and J.-C. Latombe, "Robot motion planning: A Distributed Representation Approach", International Journal of Robotics Research, vol. 10(6), pp. 628-649, 1991.

[BNP87] E. Bakker, L. Nyborg and H.B. Pacejka, "Tyre modeling for use in vehicle dynamics studies", International congress and expo of Society of Automotive Engineers (SAE), pp. 2190-2204, Warrendale, Pensylvanie (USA), 1987.

[Can85] J. F. Canny, "A Voronoi Method for the Piano-Movers Problem", In Proceedings of the International Conference on Robotics and Automation, pp. 530-535, St. Louis (USA), Mar 1985.

[Can88] J. F. Canny, "The complexity of robot motion planning", MIT Press, 1988.

[CBA96] G. Campion, G. Bastin and B. D'Andrea-Novel, "Structural Properties and Classification of Kinematic and Dynamic Models of Wheeled Mobile Robots", IEEE Transactions on Robotics and Automation, vol. 12, no. 1, pp. 47-62, Feb 1996.

[CBES09] J. Cortès, S. Barbe, M. Erard and T. Simèon, "Encoding molecular motions in voxel maps", In Proceedings of the 2009 IEEE International Conference on Robotics and Automation, pp. 2550-2555, Kobe (Japan), Jul 2009.

[CERA] "CERA says peak oil theory is faulty". Energy Bulletin, Cambridge Energy Research Associates (CERA). Available: `http://www.energybulletin.net/22381.html. Retrieved2008-07-27`.

[Chew87] L.P. Chew, "Constrained Delaunay Triangulation" in Proceedings of the third annual symposium on Computational geometry Waterloo, pp. 215-222, Ontario (Canada), 1987.

[CL97] C. Canudas de Wit and P. Lischinsky, "Adaptive friction compensation with partially known dynamic friction model", International Journal of Adaptive Control and Signal Processing, vol 11, no.1, pp. 65-80, ,1997.

[CL98] C.J. Campbell and J.H. Laherrere, "The end of cheap oil", Sci. Amer., vol.278, no3, pp.78, Mar. 1998.

[CL01] P. Cheng and S. M. Lavalle, "Reducing metric sensitivity in randomized trajectory design", In IEEE/RSJ International Conference on Intelligent Robots and Systems, 2001.

[CSL00] P. Cheng, Z. Shen and S. M. LaValle, "Using randomization to find and optimize feasible trajectories for nonlinear systems", In Proceedings Annual Allerton Conference on Communications, Control, Computing, pages 926–935, 2000.

[Dij59] E.W. Dijkstra, "A note on two problems in connexion with graph", Numerische Mathematik, vol. 1, pp.269 - 271, 1959.

[dBo78] C.R. De Boor, "A Practical Guide to Splines", Springer-Verlag, 1978.

[Dub57] L.E. Dubins, "On curves of minimal length with a constraint on average curvature, and with prescribed initial and terminal positions and tangents", American Journal of Mathematics, vol. 79, pp. 497-516, 1957.

[DY82] C. OíDunlaing and C.K. Yap, "A retraction method for planning the motion of a disc", Journal of Algorithms, vol.6, pp. 104 - 111, 1982.

[FL63] R.P. Feynman, R.B. Leighton and M. Sands, "The Feynman Lectures on Physics", Addison Wesley, vol I, Sec. 9.6; 1963.

[FLAE05] E. Ferrè, J.-P., Laumond, G., Arechavaleta and C. Estevès, "Progresses in assembly path planning", In Proceedings of the International Conference on Product Lifecycle Management, pp. 373 - 382, 2005.

[FVVR10] D. Fonte, F. Valente, A. Vale and I. Ribeiro. “A Motion Planning Methodology for Rhombic-like Vehicles for ITER Remote Handling Operations”, In Proceedings of the IAV 2010- 7th IFAC Symposium on Intelligent Autonomous Vehicles, Lecce, Italy, 2010.

[FVVR11] D. Fonte, F. Valente, A. Vale and I. Ribeiro. “Path Optimization for Rhombic-Like Vehicles: An Approach Based on Rigid Body Dynamics”, to be published in the Proceedings of the International Conference on Advanced Robotics (ICAR) 2011 - 15th IEEE International Conference on Advanced Robotics, Tallinn, Estonia, Jun 2011.

[GA03] G. Song and N.M. Amato, “A motion planning Approach to Folding: From Paper Craft to Protein Folding”, IEEE Transactions on Robotics and Automation, vol. 20(1), pp. 60-71, Feb 2004. Also, In Proc. IEEE Int. Conf. Robot. Autom. (ICRA), pp. 948-953, Seoul, Korea, May 2001. Also, Technical Report, TR00-017, Department of Computer Science and Engineering, Texas Agricultural and Mechanics (A&M) University, Jul 2000.

[GDI+09] C.G. Gutièrrez, C. Damiani, M. Irving, J.-P. Friconneau, A. Tesini, I. Ribeiro and A. Vale, “ITER Transfer Cask System: Status of design, issues and future developments”, In Proceedings of 9th International Symposium on Fusion Nuclear Technology, China, Oct 2009.

[GO04] R. Geraerts and M. H. Overmars, “Clearance Based Path Optimization for Motion Planning. In IEEE International Conference on Robotics and Automation”, vol. 3, pp.2386-2392, San Diego (USA), 2004.

[GS07] S.K. Gehrig and F. J. Stein, “Collision Avoidance for Vehicle - Following Systems”, On IEEE Transactions on Intelligent Transportation Systems”, vol. 8 (2), pp. 233-244, 2007.

[HJCA09] R. Highfield, V. Jamieson, N. Calder and R. Arnoux , “ITER: A brief history of fusion”, New Scientist Magazine, Oct 2009 [Online].Available: `http://www.newscientist.com/article/dn17952-\ac{ITER}-a-brief-history-of-fusion.html`

[HJRS03] D. Hsu, T. Jiang, J. Reif and Z. Sun, “The bridge test for sampling narrow passages with probabilistic roadmap planners”, In Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), vol.3, pp. 4420 - 4426, Nov 2003.

[HKL+98] D. Hsu, L.E. Kavraki, J.-C. Latombe, R. Motwani and S. Sorkin, “On Finding Narrow Passages with Probabilistic Roadmap Planners”, In P. Agarwal, L.E. Kavraki , and M. Mason, editors, “Robotics: The Algorithmic Perspective”, pp. 141-153, A.K. Peters, 1998.

[HLM99] D. Hsu, J.-C. Latombe and R. Motwani, “Path planning in expansive configurations spaces”, International Journal Computacional Geometry and Applications, vol. 4, pp. 495-512, 1999.

[HLW03] E. Hairer, C. Lubich and G. Wanner, “Geometric numerical integration illustrated by the Stormer Verlet method, 2003, Acta numerica, vol.12, pp. 399-450.

[HMB+04] D. van Houtte, G. Martin, A. Bècoulet, J. Bucalossi, G. Giruzzi, G.T. Hoang, Th. Loarer, B. Saoutic and the Tore Supra Team, "Recent fully non-inductive operation results in Tore Supra with 6 min, 1 GJ plasma discharges", in Nuclear Fusion, vol. 44, no.5, Apr 2004.

[HNR68] P. E. Hart, N.J. Nilsson and B. Raphael, "A Formal Basis for the Heuristic Determination of Minimum Cost Paths", IEEE Transactions on Systems Science and Cybernetics, vol.4, pp. 100 - 107, Jul 1968.

[ISN+99] S. Itoh, K.N. Sato, K. Nakamura, H. Zushi, M. Sakamoto, K. Hanada, E. Jotaki, K. Makino, S. Kawasaki, H. Nakashima and A. Iyomasa, "Recent progress in the superconducting Tokamak TRIAM-1M", Nucl. Fusion, vol.39, 1999.

[JCS08] L. Jaillet, J. Cortes and T. Simeon, "Transition-based RRT for Path Planning in Continuous Cost Space", 2008 IEEE/RSJ International Conference on Intelligent Robots and Systems, Nice, France, Sept., pp. 22-26, 2008.

[Jing08] X.-J. Jing, "Motion Planning", InTech Education and Publishing, 2008.

[KH04] H. Kurniawati and D. Hsu, "Workspace importance sampling for probabilistic roadmap planning", In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), vol.2, pp. 1618 - 1623, 2004.

[Kha85] O. Khatib,"Real-time obstacle avoidance for manipulators and mobile robots", In Proceedings of the IEEE International Conference on Robotics and Automation, 1985, pp. 500-505.

[KJCL97] M. Khatib, H. Jaouni, R. Chatila and J.-P. Laumond, "Dynamic path modification for car-like nonholonomic mobile robots", In Proceedings of the IEEE International Conference on Robotics and Automation, vol. 4, pp. 2920-2925, Albuquerque (USA), 1997.

[KKL98] L.E. Kavraki, M.N. Kolountzakis and J.-C. Latombe. "Analysis of Probabilistic Roadmaps for Path Planning", IEEE Transactions on Robotics and Automation, vol. 14(1), pp. 166-171, 1998.

[KL00a] J.J. Kuffner and J-C. Latombe, "Interactive manipulation planning for animated characters", In Proceedings of the Pacific Graphics, 2000.

[KNII03] J.J. Kuffner, K. Nishiwaki, M. Inaba and H. Inoue. "Motion planning for humanoid robots", In Proceedings International Symposium on Robotics Research, 2003.

[KP06] M. Kaisiak and M. van de Panne, "RRT-blossom: RRT with a local flood-fill behavior", In proceedings of International Conference on Robotics & Automation, ICRA 2006, pp. 1237 - 1242, Orlando, FL, 2006.

[KSLO96] L.E. Kavraki, P. Svestka, J.-C. Latombe and M.H. Overmars, "Probabilistic Roadmaps for Path Planning in High-Dimensional Configuration Spaces", IEEE Transactions on Robotics and Automation, vol. 12(4), pp. 566-580, 1996.

[Kuf01] J. J. Kuffner. Some Computed Examples [using RRT-Connect]. [Online], 2001. Available at http://www.kuffner.org/james/plan/examples.html

[KWT88] M. Kass, A. Witkin and D. Terzopoulos, “Snakes: active contour models”, International Journal on Computer Vision, vol.1 (4), pp. 321-331, 1988.

[KZ95] F. Kang and S. Zhong-Ci, “Mathematical Theory of Elastic Structures”, Science Press, Beijing (China), 1995.

[Lav98] S.M. LaValle, “Rapidly-exploring Random Trees: A New Tool for Path Planning”, Technical Report (TR) 98-11, Computer Science Dept., Iowa State University, 1998.

[Lav03] S.M. LaValle, “Planning algorithms”, 1999-2003. Available at http://msl.cs.uiuc.edu/planning

[Lat91] J.-C. Latombe, “Robot motion planning”, Kluwer Academic Publishers, 1991.

[Lau86] J.-P. Laumond, “Trajectories for mobile robots with kinematic and environment constraints”, In Proceedings International Conference on Intelligent Autonomous Systems, pp 346-354, 1986.

[Lau98] J.-P. Laumond, “Robot motion planning and control”, Springer, 1998.

[LBL02] F. Lamiraux, D. Bonnafous and O. Lefebvre, “Reactive trajectory deformation for nonholonomic systems: application to mobile robots”, In Proceedings of the International Conference on Robotics and Automation, ICRA ’02, pp. 3099 - 3104, Washington, USA, 2002.

[LBL04] F. Lamiraux, D. Bonnafous and O. Lefebvre,“Reactive path deformation for nonholonomic mobile robots”, In Proceedings of the IEEE International Conference on Robotics and Automation, vol. 20, pp. 967-977, 2004.

[LK00] S.M. LaValle and J.J. Kuffner, Jr., “Rapidly-exploring Random Trees: Progress and prospects”, in Workshop of Algorithmic foundations of robotics, 2000.

[KL00b] J. J. Kuffner Jr and S.M. Lavalle, “RRT-Connect: An efficient approach to single-query path planning”, in IEEE Int. Conf. on Robotics and Automation, 2000.

[LK01] S.M. Lavalle and J. J. Kuffner, “Rapidly-exploring random trees: Progress and prospects”, In B. R. Donald, K. M. Lynch, and D. Rus, editors, Algorithmic and Computational Robotics: New Directions, pages 293–308. A K Peters, Wellesley, MA, 2001.

[LK05] M. Lau, J.J. Kuffner, “Behavior Planning for Character Animation”, Symposium on Computer Animation, 2005.

[Loz78] T. Lozano-Perez and M. Wesley, “A. An Algorithm for Planning Collision-Free Paths Among Polyhedral Obstacles”, Communications of the Association for Computing

Machinery (ACM), vol. 22, no. 10, Oct 1979, pp.560-570. Also, IEEE Tutorial on Robotics, IEEE Computer Society, pp.65-175, 1983. Also, International Business Machines Corporation (IBM) Research Report RC7171, Jun 1978.

[Loz83] T. Lozano-Perez, "Spatial Planning: A Configuration Space Approach", IEEE Transactions on Computers, vol. 32, no. 2, pp.108-120, Feb 1983

[LSL98] J.-P. Laumond, S. Sekhavat and F. Lamiraux, "Guidelines in nonholonomic motion planning for mobile robots", In J-P., Laumond, editor, Robot motion planning and Control, pp 1-53. Springer-Verlag, Berlin, 1998.

[Mat04] M. MCTSukawa,"Recent results and modification program of JT-60 toward high integrated performance", IEEE Transactions on Plasma Science, vol.32, pp.135 - 143, May 2004.

[MN04] E. Masehian and M.R Naseri, "A voronoi diagram-visibility graph-potential field compound algorithm for robot path planning", Journal Robotic Systems, vol. 21, no. 6, pp. 275-300, 2004.

[MR95] R. Motwani and P. Raghavan, "Randomized Algorithms", Cambridge University Press, 1995.

[MRA03] M. Morales, S. Rodriguez and N.M. Amato, "Improving the connectivity of PRM roadmaps", In Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), vol.3, pp. 4427 - 4432, Taipai (Japan), Nov 2003.

[MTAB92] E. Mazer, G. Talbi, J.M. Ahuactzin and P. Bessière, "The Adriane's clew algorithm", In Proceedings of the International Conference of Society of Adaptive Behavior, Honolulu (USA), 1992.

[Nil69] N.J. Nilsson, "A mobile automaton: An application of Artificial Intelligence techniques", In 1st International Conference on Artificial Intelligence, pp. 509 - 520, 1969.

[NMK+09] M. Nakahirab, Y. MCTSumotoa, S. Kakudatea, N. Takedaa, K. Shibanumaa and A. Tesini, "Design progress of the ITER Blanket Remote Handling equipment", Symposium on Fusion Technology (SOFT)-25, vol. 84, pp.1394-1398, Jun 2009.

[OMN96] A. Oustaloup, X. Moreau and M. Nouillant, "The crone suspension", Control Engineering Practice, vol.4, no.8, pp. 1101-1108, 1996.

[Pac06] H. B. Pacejka, "Tire and Vehicle Dynamics", Second Edition, Society of Automotive Engineers Inc., 2006.

[PIJ+07] J. Palmer, M. Irving, J. Järvenpää, H. Mäkinen, H. Saarinen, M. Siuko, A. Timperi and S. Verho, "The design and development of Divertor Remote Handling equipment for ITER", SOFT-24, vol. 82, Issue 15, pp.1977-1982, Oct 2007.

[PLM06] R. Pepy, A. Lambert and H. Mounier, "Path Planning using a Dynamic Vehicle Model", Information and Communication Technologies (ICTTA 2006), vol. 1, pp.5376-5381, 2006.

[PS03] J. Pamela, E. R. Solano and JET and European Fusion Development Agreement (EFDA) contributors, "Overview of JET results", Nuclear Fusion, vol. 43, no. 12, pp. 1540 - 1554, 2003.

[PSE03] J. Popovic, S. Seitz and M. Erdmann, "Motion Sketching for Control of Rigid-Body Simulations", ACM Transactions on Graphics, vol. 22 (4), pp. 1034-1054, Oct 2003.

[PSL02] J. Pettrè, T. Simèon and J.-P. Laumond, "Planning human walk in virtual environments", In Proceedings of the IEEE/RSJ IROS,vol. 3, pp. 3048 - 3053, 2002.

[QK93a] S. Quinlan and O. Khatib, "Towards real-time execution of motion tasks", In R. Chatila and G. Hirzinger, editors, Experimental Robotics, Springer-Verlag, 1993.

[QK93b] S. Quinlan and O. Khatib, "Elastic bands: Connecting path planning and control", In Proceedings of the IEEE International Conference on Robotics and Automation, vol. 2, pp. 802-807, Atlanta (USA), 1993.

[RH91] M.H. Raibert and J.K. Hodgins, "Animation of dynamic legged locomotion", In Computer Graphics (Proceedings of SIGGRAPH 91) - Annual Conference Series. ACM SIGGRAPH, vol. 25 (4), pp. 349 - 358, Jul 1991.

[RLAF97] M.I. Ribeiro, P. Lima, P. Aparício and R. Ferreira, "Conceptual Study on Flexible Guidance and Navigation for ITER Remote Handling Transport Casks", In Proceeding of the 17th IEEE/NPSS Symposium on Fusion Engineering, San Diego (USA), pp. 969-972, 2007.

[RLA98] M.I. Ribeiro, P. Lima and P. Aparício, "Geometric Feasibility of the ITER Air Cushion Remote Handling Casks and Extensions for Free Roaming Navigation", Technical Report, RT-402-98, Instituto de Sistemas e Robótica, Instituto Superior Técnico, June 1998.

[RS90] J.A. Reeds and L.A. Shepp, "Optimal paths for a car that goes both forwards and backwards", Pacific J. Math.,vol. 145(2), pp.367-393, 1990.

[RVR+10] M.I. Ribeiro, A. Vale, P. Ruibanys, V. Queral, D. Fonte, F. Valente, C.Reig, E. Gazeau and P. Lima, "Activities related to the development of an ATS prototype and Cask Transfer System Virtual Mock-up"- Final Report – F4E-2008-GRT-016 (MS-RH), Oct. 2010.

[Sch06] K.R., Schultz, "Why Fusion? A discussion of energy alternatives", Control Systems Magazine, IEEE, vol.26, issue:2, pp.32-34, Apr 2006.

[SCM01] J. Stephant, A. Charara and D. Meizel, "Contact roue/sol: comparaison Journèes Automatique et Automobile de modèles d'efforts", In Journèes Automatique et Automobile, LAP-ENSEIRB-Universitè Bordeaux I (France), 2001.

[SK08] B. Siciliano and O. Khatib, "Handbook of Robotics", Springer, 2008.

[SL90] P. Souères and J.-P. Laumond, "Shortest paths syndissertation for a car-like robot", In IEEE Transactions on Automatic Control, pp. 672-688, 1996.

[SLN99] T. Simèon, J.-P. Laumond and C. Nissoux, "Visibility based probabilistic roadmaps for motion planning", In Proceedings of the International Conference on Intelligent Robots and Systems- IROS'99, vol.3 , pp.316 - 1321, Out 1999.

[SO98] P. Svestka and M.H. Overmars, "Probabilistic Path Planning". In J.-P. Laumond, editor, "Robot motion planning and Control", pp. 255-304, Springer-Verlag, 1998.

[SS83] J. T. Schwartz and M. Sharir, "On the Piano Movers' Problem II: General Techniques for Computing Topological Properties of Real Algebraic Manifolds", Advances in Applied Mathematics, vol. 4, pp 298-351, 1983.

[SS84] J. T. Schwartz and M. Sharir, "On the Piano Movers' Problem V: The Case of a Rod Moving in Three-Dimensional Space Amidst Polyhedral Obstacles", Communications on Pure Applied Mathematics, vol. 37, pp 815-848, 1984.

[ST91] H. Sussman and G. Tang, "Shortest paths for the Reeds-Shepp car: A worked out example of the use of geometric techniques in nonlinear optimal control", Technical Report SYNCON 91-10, Dept. of Mathematics, Rutgers University, 1991.

[TTA10] L. Tapia, S. Thomas and N.M. Amato, "A motion planning Approach to Studying Molecular Motions", Communications in Information and Systems, vol. 10, no. 1, pp. 53-68, 2010.

[TTTA10] L. Tapia, X. Tang S. Thomas and N.M. Amato, "Roadmap-Based Methods for Studying Protein Folding Kinetics", Technical Report, TR06-011, Parasol Laboratory, Department of Computer Science, Texas A&M University, Oct 2006.

[URL10] `http://www.iter.org/org/Pages/DAs.aspxlastviewed15/04/2010`.

[URLCor] European Commission Cordis website: `http://cordis.europa.eu/fetch?CALLER=NEWSLINK_EN_C&RCN=25711&ACTION=D`

[URLCS] School of Computer Science - Carnegie Mellon (CMU) website: `http://www.cs.cmu.edu/afs/cs/academic/class/16741-s07/www/lecture5.pdf`

[URLDEMO] Wikipedia website [DEMO]: `http://en.wikipedia.org/wiki/DEMO`

[URLED] World energy demand: `http://www.paulchefurka.ca/WEAP2/WEAP2.html`

[URLEI] Wikipedia website [Euler integration method]: `http://en.wikipedia.org/wiki/Euler_method`

[URLF4Ea] Fusion for Energy website [Understanding fusion]: `http://fusionforenergy.europa.eu/understandingfusion/merits.aspx`

[URLF4Eb] Fusion for Energy website [Media corner]: http://fusionforenergy.europa.eu/mediacorner/newsview.aspx?content=270

[URLFF] Fusion versus fission: http://www.diffen.com/difference/Nuclear_Fission_vs_Nuclear_Fusion

[URLFP] Wikipedia website [Fusion power]: http://en.wikipedia.org/wiki/Fusion_power

[URLIO] ITER website: http://www.iter.org/

[URLKK] Kineo CAM: http://www.kineocam.com/

[URLMat] Wolfram Mathematica website: http://www.wolfram.com/products/student/mathforstudents/index.html

[URLMP] Motion Profiles: http://www.pmdcorp.com/news/articles/html/Mathematics_of_Motion_Control_Profiles.cfm

[URLRKI] Wikipedia website [Runge Kutta integration method]: http://en.wikipedia.org/wiki/Runge-Kutta_method

[URLSE] Wikipedia website [Stiff equations]: http://en.wikipedia.org/wiki/Stiff_equation

[URLTR] Wikipedia website [Trapezoidal rule: http://en.wikipedia.org/wiki/Trapezoidal_rule

[URLVI] Wikipedia website [Verlet integration algorithm]: http://en.wikipedia.org/wiki/Verlet_integration

[URLVVI] Yale lecture notes [Verlet Velocity integration method]: http://xbeams.chem.yale.edu/~batista/vaa/node60.html

[URLWP] World population forecast: http://www.photius.com/rankings/world2050_rank.html

[VRF+09] A. Vale, M.I. Ribeiro, D. Fonte, F. Valente and E. Silva, "Activities Related To the Development of an Air Transfer System Prototype and Cask Transfer System Virtual Mock-Up", Interim Report.

[VVFR10] F. Valente, A. Vale, D. Fonte and I. Ribeiro, "Optimized Trajectories of the Transfer Cask System in ITER", In the 26th Symposium on Fusion Technology (SOFT 2010), Porto, Portugal, Oct. 2010.

[Wes00] J. Wesson, "The science of JET", JET Joint Undertaking, Abungdon, Oxon, 2000 [Online]. Available: http://www.jet.edfa.org/documents/wesson/wesson.html.af

[WQ01] D. Wang and F. Qi, "Trajectory Planning for a Four-Wheel Steering Vehicle", Proceedings of the 2001 IEEE ICRA, Korea, 2001.

[YBEL05] E. Yoshida, I. Belousov, C. Esteves and J.-P. Laumond, “Humanoid motion planning for Dynamic Tasks”, In IEEE-Robotics and Automation Society (RAS) International Conference on Humanoid Robots, Tsukuba (Japan), 2005.

[YES+07] E. Yoshida, C. Esteves, T. Sakaguchi, J.-P. Laumond and K. Yokoi, “Smooth Collision Avoidance: Practical Issues in Dynamic Humanoid Motion”, Conference on Intelligent Robots and Systems, IEEE/RSJ International, pp. 827 - 832 2007.

Appendix A

Description of the Remote Handling Systems of ITER

In addition to the Cask and Plug Remote Handling System (CPRHS) described in Chapter 1, five other Remote Handling (RH) systems are expected to operate in International Thermonuclear Experimental Reactor (ITER), which correspond to the following procurement packages:

- **Blanket RH system**
  According to [NMK+09], the ITER Blanket wall, depicted in Figure A.1, will be composed by approximately 440 modules, 4.5 t each, disposed toroidally in the Vacuum Vessel (VV). As already discussed, these components will have to be periodically replaced and inspected. The maintenance of these components will be carried out by a specific RH subsystem, the In-Vessel Transporter (IVT), partially illustrated in Figure A.1-top. The IVT consists of an articulated ring rail, umbilically supported by special IVT transporters placed at specific Vacuum Vessel Port Cells (VVPCs), a rail-mounted vehicle and a telescopic arm with proper end-effectors and tools. This RH equipment is currently under the responsibility of the Japanese Domestic Agency (DA).
- **Divertor RH system**
  The Divertor cassettes are essential components of the ITER reactor. They are the only components allowed to touch the hot confined plasma and are responsible for the removal of the helium ashes, which is a harmful product resultant from the fusion reaction. In scheduled ITER maintenance interventions, the Divertor system will also need to be replaced.
  The Divertor system, comprises 54 cassettes, each weighing about 9 t [PIJ+07]. Handling this type of components in ´the confined interior of the VV will certainly be an hard RH task.
  Continuously developed in the European Union (EU)-DA, the handling equipment foreseen for these operations is comprised by a Cassette Toroidal Mover (CTM) and a Cassette Multifunctional (CMM), both illustrated in Figure A.1-bottom. These two modules were designed to remove the cassettes through the VV ports located at the Divertor level. While the CMM will move in radial rails to place and remove the cassettes in the front of the VV, the CTM will execute toroidal movements do deliver and collect the cassettes along the VV. According to [PIJ+07], the CMM will also be responsible to handle the port closure plate and other diagnostic assemblies.
  After the developed Divertor Test Platform (DTP), which aimed to demonstrate the Divertor RH process, the EU-DA has already constructed a new DTP2 at Temper, Finland, to prove and

consolidate the design of this RH subsystem.

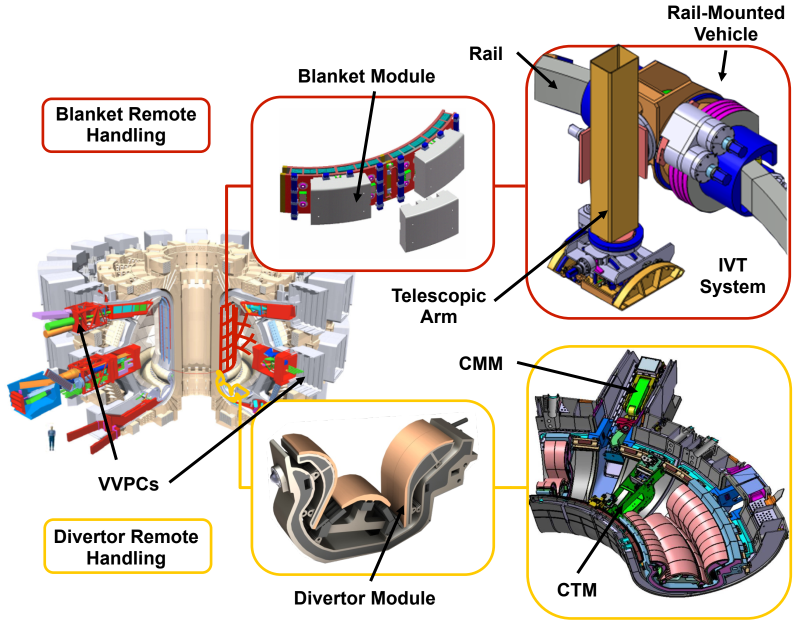


Figure A.1: Top: IVT and the manipulator system. Bottom: the Divertor cassette RH equipment deployed in the VV.

- **Hot Cell RH system**

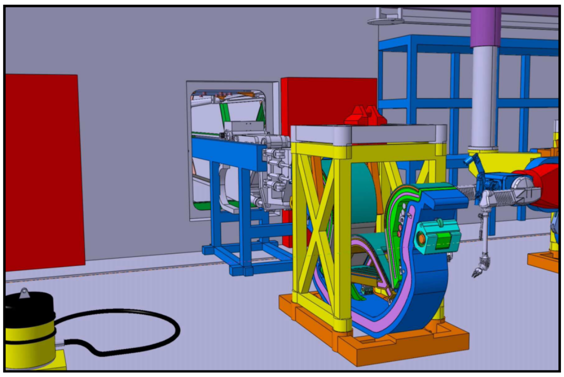
  The RH equipment present in the Hot Cell Building (HCB) (being procured directly by the ITER Organization (IO)), will be responsible for the refurbishment, testing, and disposal of components that have become activated by neutron exposure in the VV on the Tokamak Building (TB). Waste materials will be treated, packaged, and temporarily stored in the Hot Cell facility before being handed over to the French authorities. The Hot Cell facilities will also serve as a test stand to validate RH procedures and train RH operators.

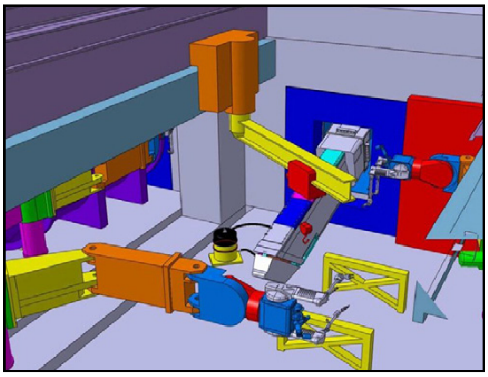
  The Hot Cell RH equipment (see Figure A.2) is composed by several devices:

  - Boom-style RH transporters;
  - Jib cranes transporters;
  - Lifting jigs;
  - Dexterous tele-manipulators;
  - Viewing systems;
  - Inspection systems;
  - Cleaning equipment.

- **In-Vessel Viewing System**

  The In-Vessel Viewing System (IVVS), will be used to conduct inspections on the VV, either after plasma pulses (1 day) or after an operational week. Even though it can be considered a diagnostic system, the IVVS will likely conduct to some RH interventions, reason why it is considered here.





(a) (b)

Figure A.2: Some of the RH operations and equipment needed in the HCB: (a) divertor cassette refurbishment; (b) Refurbishment plugs.

The IVVS is composed by six identical units of laser scanning probes that can act as a combined monitoring system. These units will be disposed along the VV with directions converging in couples, as depicted in Figure A.3-a, to reduce the toroidal distance between pairs (given the non-uniform spacing of the IVVS ports).

The inspections will be performed there upon plasma pulses, on a non-vented environment, where the vacuum, temperature and radiation conditions will be, by far, the most demanding when compared to the other RH operations.

- **Neutral Beam (NB) RH system**

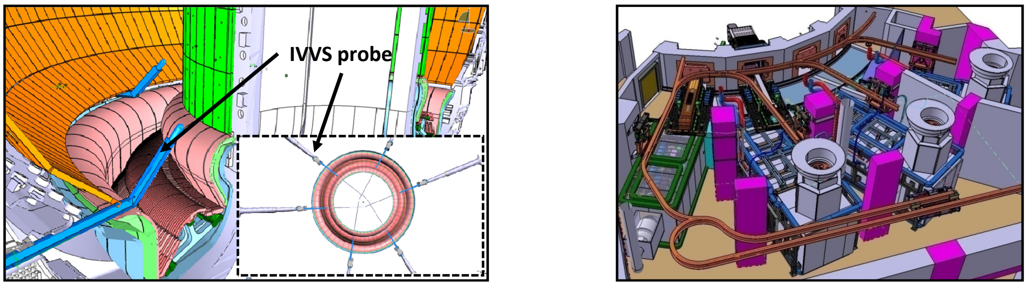
  In the NB cell, represented in Figure A.3-b, routine remote maintenance procedures are needed in the NB injectors for the removal and replacement of

  - the Cesium oven fueling system,
  - the beam source and
  - the beam line components.

  To perform such operations, different RH equipments will be available in the NB cell:

  - (50t) Monorail crane, equipped with special lifting interfaces. This system will assure the transportation of the various components to a specific transfer area in order to get out of the NB cell towards the HCB;
  - Transport cradle, specifically designed for the 26 t NB source-accelerator;
  - Force feedback manipulator arm and various tooling;
  - Special end-effectors for the installation and removal of the diagnostic tubes located in the upper level;
  - Auxiliary devices for temporary storage and transportation.

  The dimension of the components to be handled in the NB are similar to the Divertor cassettes but they are much more heavier (recall that the NB source-accelerator weights 26 t). The EU-DA is currently in charge of this RH system.



(a) (b)

Figure A.3: (a) The IVVS probes inside the VV; There will be available 6 equal units disposed equatorially with directions converging in pairs. (b) NB cell with a rail crane system.

Appendix B

Topics for the Guidance and Navigation of the Rhombic Vehicle

During International Thermonuclear Experimental Reactor (ITER) operation the rhombic vehicle is expected to operate in three different operational modes [1]:

- **Automatic** – The rhombic vehicle follows autonomously a physical or virtual path or trajectory. The operator can only perform monitoring operations together with basic actions (start and shutdown of the vehicle);
- **Semi-automatic** – The operator is free to control the vehicle velocity, but the vehicle is still in charge of autonomously follow a geometric reference by steering the wheels;
- **Manual mode** – the operator fully controls the vehicle.

The transportation system will be based on flexible or non-contact guidance approaches [GDI$^+$09], as opposed to rails or any other hard-guidance solutions. Within the flexible guidance approaches, three different navigation and guidance concepts are possible for the rhombic vehicle[RLAF97]:

- **Automated Guided Vehicle (AGV)** – the vehicle follows a physical path implemented on the floor. This is usually refer to as line guidance approach. Different guided paths can be used such as, optical, magnetic (inductive/active) tapes or buried wired systems. The AGV-like solution would present simpler control requirements an higher reliability but an undesirable low flexibility, since both vehicle wheels would have to follow the same physical paths;
- **Mobile robot** – acting as a free-roaming platform, the vehicle is able to operate in all the free space available and follow virtual references. Apart from the increased flexibility, this solution would be more robust, allowing for rescue interventions.
- **Mixed Vehicle** – the vehicle has the capabilities of normally following a physical path but, if necessary, it can leave this physical path and switch to a virtual reference, using appropriate virtual tracking methodologies. Both the AGV or mobile robot can be used as the primary navigation solution. This results is a more robust, reliable and flexible mobile platform.

Regardless off the solution that will be adopted for the rhombic vehicle, its navigation and guidance system can be described within the same integrated framework, as depicted in Figure B.1.

[1] The view expressed here consists in a redefinition of the classification presented in [RLA98], and reflects the sole author's understanding on this topic.

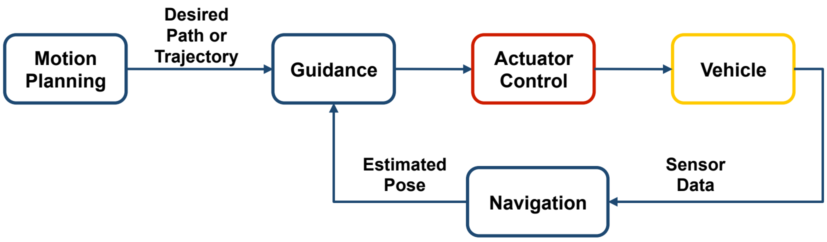


Figure B.1: Schematic view of a generic navigation and guidance framework.

In particular, three main problems can be stated:

- **Motion planning** – Refers to both the evaluation of a physical path or a virtual reference to be followed by the vehicle;
- **Navigation** – This stage comprises the estimation of the vehicle pose (position plus the orientation), with respect to a given environment frame. This module can also comprise other features such as mapping, collision avoidance and sensory fusion capabilities;
- **Guidance** – Consists on following a specific reference by means of the appropriate motion control laws, while avoiding collisions with obstacles.

The guidance methodologies used depend on the navigation mode adopted but, two distinct problems, which are often erroneously used as synonymous, can be stated:

- **Path Following** – A path provides a geometric description of the vehicle motion (e.g., Cartesian points or vehicle poses). It is considered to be a sequence of collision-free motions connecting the initial and final chosen vehicle configurations. In the path following problem the vehicle must follow this geometric path, starting from an initial configuration that may be on or off the path. When off the path and whenever a locally collision-free path exists, the vehicle may reach the path by looking for the nearest point, otherwise a local planner shall be used. The motion control problem consists in following the geometric reference (e.g., a buried wire or a virtual path). It is a time independent task, which is only concerned with the geometrical mismatch between the vehicle and the path to be followed. The vehicle speed is usually provided by a velocity controller.
- **Trajectory tracking** – A trajectory can be defined as a time or velocity parameterized path (i.e., trajectory = path + time/velocity) . Trajectories define the vehicle motion based on time, on dynamic assumptions (e.g., constraints on vehicle velocity and acceleration). The problem of trajectory tracking can thus be defined as the following of a virtual geometric path with a time function associated. The motion control problem consists on minimizing a function error that includes the mismatch between the actual vehicle configuration (provided by the localization system) and the virtual vehicle configuration that perfectly follows the desired trajectory.

Once the control law is determined by the guidance module, it must be "translated" to a lower level, i.e., to achieve a specific wheel velocity or steering angle, the actuators of the vehicle must receive torque inputs, e.g. driving and steering torques.

The performances of the guidance system highly depend on the navigation system which, resorting on the raw data from sensors, provides the means to:

- Estimate the pose of the vehicle – This evaluation is performed based on a localization system. In the case of a AGV-type vehicle, and disregarding the case of virtual paths, the localization system should evaluate this information relatively to the physical path being followed. In a mobile robot mode, the localization must be made with respect to a given world frame, providing a periodic estimation of the vehicle's pose.
- Avoid obstacles during motion. This is possible by using adequate collision avoidance features based on sensor data processing tools or anti-collision alarm systems based on hardware applications such as infrared light curtains.

Appendix C

Changing the basis of the Null Space

The null space of C given in 2.12 was computed with the MATHEMATICA software application [URLMat] yielding

$$Ker(C(\boldsymbol{q})) = V = \begin{bmatrix} \frac{M_R \cdot \cos\theta_R \cdot \cos(\theta+\theta_F) + M_F \cdot \cos\theta_F \cdot \cos(\theta+\theta_R)}{\sin(\theta_F - \theta_R)} \\ \frac{M_R \cdot \cos\theta_R \cdot \sin(\theta+\theta_F) + M_F \cdot \cos\theta_F \cdot \sin(\theta+\theta_R)}{\sin(\theta_F - \theta_R)} \\ 1 \end{bmatrix}, \tag{C.1}$$

which can be seen as a basis for the null space of C. Recalling the properties referred in Subsection 2.2.2, the vector V may be multiplied by any real scalar that the resulting new vector will still continue to belong to the null space of $C(\boldsymbol{q})$. It may then be defined a new vector $U \in Ker(C)$ as follows

$$U = \frac{\sin(\theta_F - \theta_R)}{M} \cdot V = \begin{bmatrix} \frac{M_R \cdot \cos\theta_R \cdot \cos(\theta+\theta_F) + M_F \cdot \cos\theta_F \cdot \cos(\theta+\theta_R)}{M} \\ \frac{M_F \cdot \cos\theta_R \cdot \sin(\theta+\theta_F) + M_F \cdot \cos\theta_F \cdot \sin(\theta+\theta_R)}{M} \\ \frac{\sin(\theta_F - \theta_R)}{M} \end{bmatrix}, \tag{C.2}$$

that rearranged results on[1],

$$U = \begin{bmatrix} \frac{M_R \cdot \cos\theta_R \cdot \cos(\theta+\theta_F)}{M} \\ \frac{M_R \cdot \cos\theta_R \cdot \sin(\theta+\theta_F)}{M} \\ \frac{\sin\theta_F \cdot \cos\theta_R}{M}] \end{bmatrix} + \begin{bmatrix} \frac{M_F \cdot \cos\theta_F \cdot \cos(\theta+\theta_R)}{M} \\ \frac{M_F \cdot \cos\theta_F \cdot \sin(\theta+\theta_R)}{M} \\ \frac{-\sin\theta_R \cdot \cos\theta_F}{M} \end{bmatrix} \tag{C.3}$$

Again, recalling preposition $P3$, notice that the following vector T also belongs to the null space of $C(\boldsymbol{q})$

$$T = \begin{bmatrix} \frac{M_R \cdot \cos\theta_R \cdot \cos(\theta+\theta_F)}{M} \\ \frac{M_R \cdot \cos\theta_R \cdot \sin(\theta+\theta_F)}{M} \\ \frac{\sin\theta_F \cdot \cos\theta_R}{M} \end{bmatrix} \cdot \iota_1 + \begin{bmatrix} \frac{M_F \cdot \cos\theta_F \cdot \cos(\theta+\theta_R)}{M} \\ \frac{M_F \cdot \cos\theta_F \cdot \sin(\theta+\theta_R)}{M} \\ \frac{-\sin\theta_R \cdot \cos\theta_F}{M} \end{bmatrix} \cdot \iota_2 \tag{C.4}$$

if $\iota_1 = \iota_2$.

[1] $sin(\theta_F - \theta_R) = \sin\theta_F \cdot \cos\theta_R - \sin\theta_R \cdot \cos(\theta_F)$

Appendix D

Motion Planning: Motivational Examples and Applications

Motion planning techniques are currently being applied to solve different problems on different fields of research. Some motivational examples are presented next.

Assembly Puzzles

Discrete and continuous puzzles have always aroused interest among humans. Can you imagine these problems being solved by planning algorithms? In fact, familiar problems such as the Rubik's cube illustrated in Figure D.1-left, or the sliding-tile puzzle, can be solved by using discrete planning techniques. However, brain teasers such as the one shown in Figure D.1-right, were created to frustrate both humans and planning algorithms since they involve an higher level of difficulty and require to be treated at a continuous instance. Benchmark problems like this are particularly important since they encourage the development of new and more efficient planning algorithms.

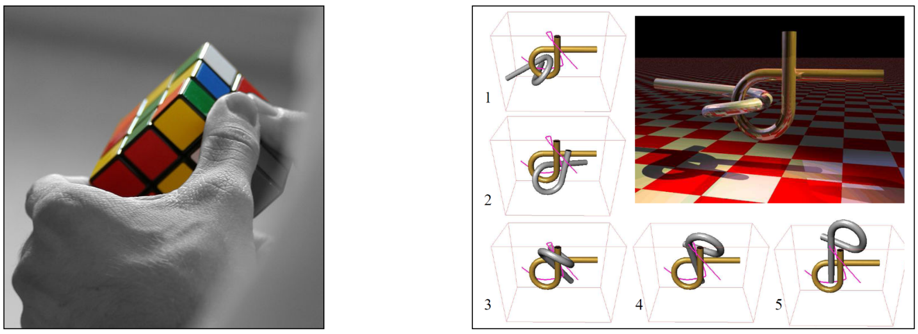


Figure D.1: Left: the Rubik's cube. Right: the Alpha 1.0 Puzzle was posted as a research benchmark by Nancy Amato at Texas A&M University.

Computer Animation and humanoid robots

Nowadays, it is common to find human-like characters exhibiting more and more sophisticated human behavior in video games. In this field, planning algorithms may soon play a major contribution by allowing to determine the animation of characters with a higher level of abstraction, i.e., the motion of each character is generated automatically resembling realistic motions. An illustration of this type of planning-based animation is presented in Figure D.2-left. Pre-computed animations are often feasible but they restrict the free hand of the user. Moreover, in this type of problems, the game developers deal with perfect world models and are not confronted with sensing and perception problems of mobile robotics, easing the motion planning task. Nevertheless, there is still a large division among planning-algorithm and video-game developing communities and a major effort must be done to adapt existing planning techniques to video-game animations as in [KL00a, PSL02].

The interest for planning techniques can also be found in the humanoid research area. Apart from the huge developments made in the automatic control field, which allow for instance, humanoid robots to maintain the equilibrium, recovering from losses of balance or to climb stairs, the development and use of planning capabilities towards higher levels of automation are still remaining issues to be solved. Still, some research efforts can already be found, which give a step forward and tackle these issues. See for example [KNII03, YBEL05] or the study [YES$^+$07], which has an illustration in Figure D.2-right.

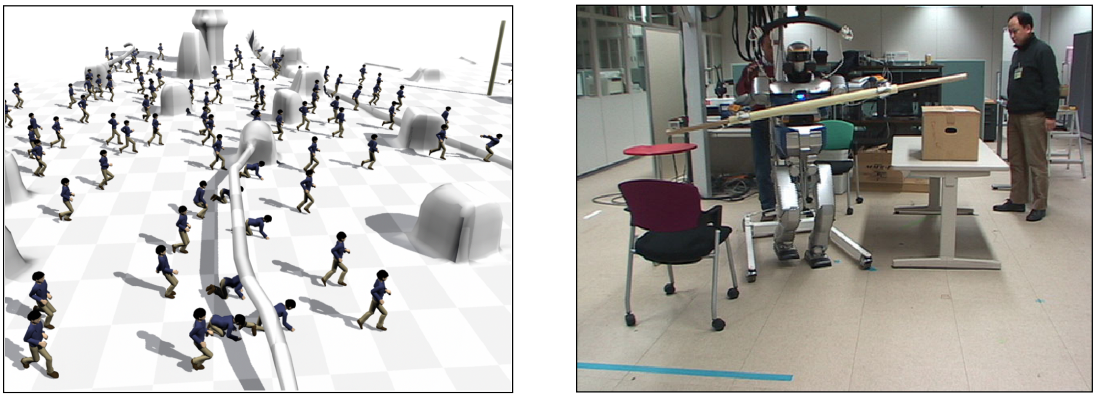


Figure D.2: Left: motion planning algorithms were used to compute the motion of 100 digital actors crossing a scenario with different types of obstacles [LK05]. Right: HRP-2 humanoid performing bar-carrying task and locomotion with stable collision free planned motions.

Assembly Planning

Assembly planning has several applications, for instance in automotive (dis)assembly and virtual prototyping. Here, planning algorithms allow answering to some questions such as: Is it possible to assemble the motor on the car cavity? In which steps and order should the seat be handled in order to reach the inside of the car without collisions? What is the best way to repair a broken piece of an assembled product? If each part of the product is considered as a separated rigid body, then the problem can be solved by determining the motions that each element shall perform to enable a specific part to reach a desired goal location. As shown in Figure D.3 , these applications assume a particular importance in the automotive industry, with big companies like Volvo, Renault, Ford and other corporations benefiting from specific motion planning software such as the one produced at Kineo CAM (Kineo Computer Aided Motion) [URLKK], a spin-off French company founded by important pioneers in the motion planning field, namely Jean-Paul Laumond, Tierry Simèon and Florence Lamiraux. As it can be imagined, these contributions allow to considerably reduce the time required to simulate the production environment, create new assembly processes and help manufacturers to better analyze the risk involved on cost-saving alternatives.

Computational Biology

This may be one of the farthest fields from the realm of traditional robotics where planning algorithms

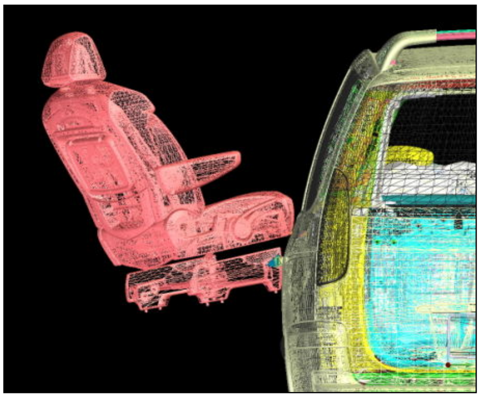


Figure D.3: An automotive assembly task, which involves inserting a seat in the car body cavity.

are applied. In computational biology, motion planning algorithms are used at a molecular level to solve two major problems: the protein folding illustrated in Figure D.4 and drug design. For the protein folding problem, the aim is to understand better how the molecules behave in the folding process by planning and analyzing their motions. Note that many diseases such as Alzheimer's and Parkinson's, are associated with protein misfolding and aggregation. Due to the difficulty on experimentally observe these processes, planning algorithms together in cooperation with other computational tools, can drastically hasten this knowledge process. [CBES09, AS02, YES^{+}07, TTA10, TTTA10] report some of the latest developments in this field.

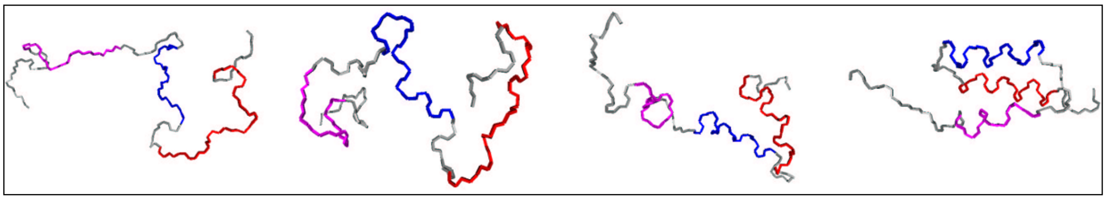


Figure D.4: Snapshots of folding paths found by a motion planner [GA03] for the protein A.

Autonomous Robots Nearly every task that an autonomous robot has to accomplish, involves a motion from one configuration to another. However, the problem of moving autonomously in a particular environment it is a complex and interdisciplinary subject. This problematic implies, among other things, that the robot is capable of avoiding obstacles, estimate precisely its location and know, where, how and when to move somewhere. In the autonomous robotics field, this is valid for an indoor Wheeled Mobile Robot (WMR), an industrial robot like a fixed manipulator arm or even for an autonomous underwater vehicle. The motion planning task does not solve the entire problem but it allows tackling an important issue involved on this challenging assignment: from its initial configuration, how should the robot move in order to reach a desired goal configuration while avoiding collisions with obstacles. In fact and as introduced in Chapter 1, for a given environment and robot model, motion planning techniques will provide the means to find a feasible sequence of motions that assure non-contact with obstacles and guarantee other operating conditions such as kinematic or even robot dynamic constraints.

Appendix E

Sparse Dissertation Contributions on the Elastic Bands Approach

A - Evaluating the Repulsive Forces during Path Deformation

The approaches for modelling the elastic contribution are very similar on most of the related published studies [QK93b, GS07]. However, the repulsive interactions with the environment can be achieved in many different ways. Herein an approach is presented that considers that path optimization must include both geometric and kinematic robot properties. For instance, a feasible and optimized path for a specific robot or vehicle configuration may become unfeasible or lead to a reduced path quality, if other configurations are used. With this purpose, the scheme herein proposed for the repulsive forces computation takes directly into account the vehicle configuration.

Using a collision detector algorithm, the nearest Obstacle Point (OP) to each vehicle pose might be considered. The use of a single OP as a reference to determine the repulsive forces may not be satisfactory to maintain clearance from obstacles, and therefore, a larger set of obstacle points, such as the k-nearest (k-OPs), must be considered. This will lead to a more balanced repulsive contribution ensuring effectiveness on most situations. Henceforth, on this formulation, it is considered the set of references formed by the nearest OP to each one of the four vehicle faces as illustrated on Figure E.1. The overall procedure to evaluate the repulsive force for each path point Pi is the following:

1. The initial poses are determined based on the constraint of both wheels placed over the path and are sampled considering:
   - Forward way: the rear wheel is fixed on each $\boldsymbol{p_j}$. Then, the path point $\boldsymbol{p_j}$ closest to the front wheel position along the path is determined. $\boldsymbol{p_j}'$is such that

     $$\|\boldsymbol{p_j} - \boldsymbol{p_j}'\| \leq M \tag{E.1}$$

     where M, remember, defines the distance between front and rear wheels. The points $\boldsymbol{p_j}$ and $\boldsymbol{p_j}'$ define the vehicle pose;
   - Backward way: the same procedure is repeated, now fixing the front wheel for each $\boldsymbol{p_j}$, as if the vehicle was executing the path moving backwards. Let $l = \{1, \ldots, L\}$ denote the index of the lth OP considered on each $\boldsymbol{p_j}$ related pose and $i = \{F, B\}$ referring to the forward or backward way. $\boldsymbol{t_{i,l}}$ is the vector defined by the lth OP ($O_{i,l}$) and each wheel point (W_F = rear

wheel and W_B = front wheel), as illustrated in Figure E.1,

$$\boldsymbol{t_{i,l}} = W_i - O_{i,l}. \tag{E.2}$$

2. This vector defines the repulsive force direction taking into account the position of the wheels. To maintain clearance from obstacles, the force magnitude must vary inversely with the distance of the poses to the obstacles. To carry out this geometric consideration, let $\boldsymbol{u_{i,l}}$ denote the vector taken from the $(O_{i,l})$ to the vehicle nearest point $V_{i,l}$,

$$\boldsymbol{u_{i,l}} = V_{i,l} - O_{i,l}. \tag{E.3}$$

3. Each pair of points $(O_{i,l}, V_{i,l})$ determines a repulsive contribution defined on $\boldsymbol{p_j}$ given by,

$$\boldsymbol{r_{i,l}(p_j)} = \frac{\boldsymbol{t_{i,l}}}{\|\boldsymbol{t_{i,l}}\|} \cdot f(\|\boldsymbol{u_{i,l}}\|), \tag{E.4}$$

where,

$$f(\|\boldsymbol{u_{i,l}}\|) = \max(0, F_{max} - \frac{F_{max}}{d_{max}} \cdot \|\boldsymbol{u_{i,l}}\|). \tag{E.5}$$

In (E.5), a maximum allowable magnitude, F_{max}, is assigned to avoid outsized values in the close vicinity of the obstacles. d_{max} denotes the distance up to which the repulsive force is applied.

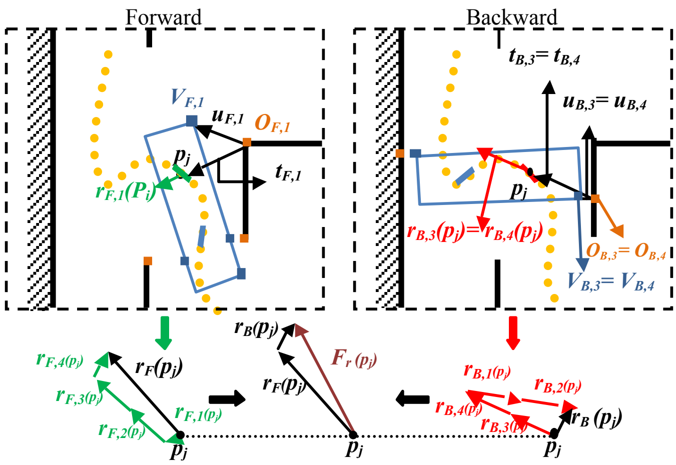


Figure E.1: Evaluation of repulsive forces: rear wheel in $\boldsymbol{p_j}$ (left) and front wheel in $\boldsymbol{p_j}$ (right). Dot blue and dot orange points represent the closest vehicle surface and obstacle points, respectively.

B - Including Maneuvers on Path Deformation

The combinatorial path planning and optimization approach described in Section 4.1 must be improved to support one or multiple maneuvers. As illustrated in Figure E.2, a path with a maneuver requires splitting the path in two other sub-paths with a constraint: the final point of the first sub-path must be the initial point of the next sub-path. The point where the maneuver occurs, i.e., where the path is splitted, defines the location where the vehicle stops and inverts its direction of movement.

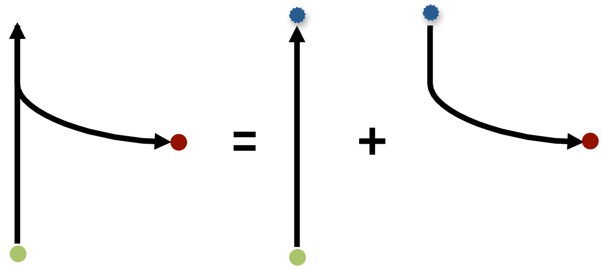


Figure E.2: Planning maneuvers: problem formulation; $\boldsymbol{q_I}$ in green, $\boldsymbol{q_G}$ in red and maneuver point in blue.

In case n maneuvers are required, the path has to be divided in $n+1$ sub-paths with n stopping points for the vehicle to invert its direction of movement. Thus, the solution is applying the path optimization to each sub-path. The decision of including maneuvers in the path is taken when a path without maneuver is not feasible or does not fulfill the minimum safety distance to the obstacles. The number of points of maneuver is decided manually. Figure E.3 illustrates an example for a maneuver planning problem.

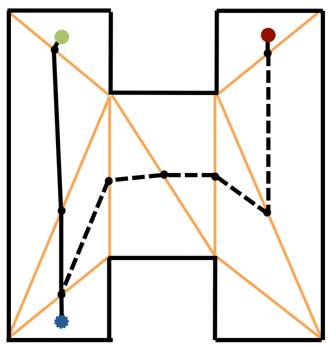


Figure E.3: Geometric path planning: triangle CD (orange), first segment roadmap (black) and second segment roadmap (dashed black)

However, there is an additional constraint when considering maneuvers. As underlined along Section 4.1, the path should be common to both wheels. However, both wheels cannot follow the entire path. If moving on forward direction, the front wheel only starts few meters ahead from the beginning, but finishes exactly in the final point. On the other hand, the rear wheel starts in the initial point but finishes before the final point. This shift corresponds to the wheelbase (M). Consequently, this constraint should also be taken into consideration when evaluating maneuvers. Between two consecutive sub-paths of a maneuver, there is a coincident segment of both sub-paths with a length greater or equal to the distance between the wheels (see Figure E.4).

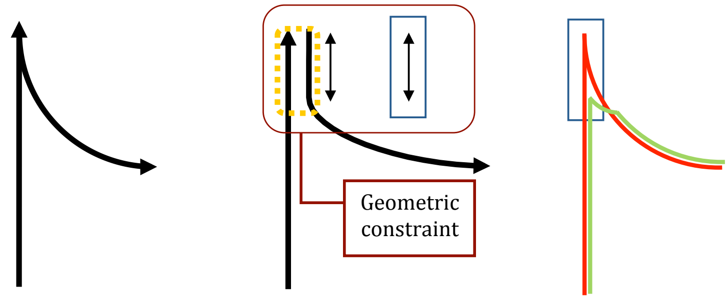


Figure E.4: Example of a maneuver, from left to right: a path, splitting the path in two sub-paths with the wheelbase geometric constraint, and the path described by the front and rear wheels.

Figure E.5 presents an example and the respective iterations to optimize the trajectory with maneuver based on the combinatorial approach presented Section 4.1 and using the above mentioned ideas.

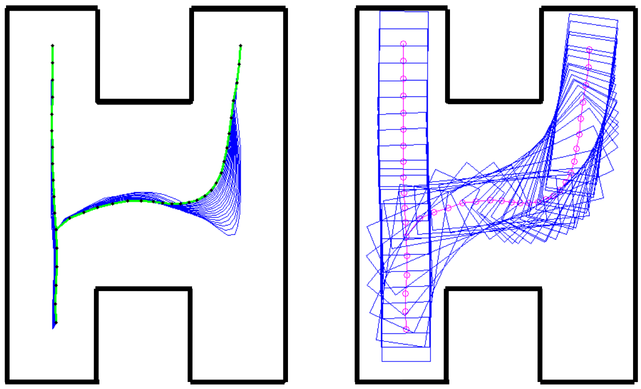


Figure E.5: Path optimization with maneuver included.

A remark should be made saying that each point of maneuver is common to the consecutive sub-paths, but it is not fixed. The algorithm requires an initial point of maneuver and then adjusts its position for optimizing the path and reducing the risk of collision. This point is manually defined, similarly to the definition of $\boldsymbol{q_I}$ and $\boldsymbol{q_G}$ for the typical motion query.