



HAL
open science

Governing Automated Vehicle Behavior

Nelson de Moura

► **To cite this version:**

Nelson de Moura. Governing Automated Vehicle Behavior. Computer Science [cs]. Sorbonne Universities, UPMC University of Paris 6, 2021. English. NNT: . tel-03680555

HAL Id: tel-03680555

<https://hal.science/tel-03680555>

Submitted on 28 May 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



SORBONNE UNIVERSITÉ

École doctorale 130: Informatique, Télécommunications et Électronique
Institut des Systèmes Intelligents et de Robotique

Thèse de Doctorat

Pour obtenir le grade de

DOCTEUR en informatique

Soutenance présenté par

Nelson DE MOURA MARTINS GOMES

GOVERNING AUTOMATED VEHICLE BEHAVIOR

Soutenu le 29 juin 2021 devant le jury composé de :

Dr. Fawzi NASHASHIBI	Rapporteur	INRIA
Dr. Catherine TESSIER	Rapporteur	Onera
Pr. Stéphane CHAUVIER	Examineur	Sorbonne Université
Pr. Véronique CHERFAOUI	Examineur	Université de Technologie de Compiègne
Dr. Ebru BURCU DOGAN	Examineur	Institut VEDECOM
Pr. Jean-Gabriel GANASCIA	Examineur	Sorbonne Université
Pr. Alan WINFIELD	Examineur	UWE Bristol
Pr. Raja CHATILA	Directeur de thèse	Sorbonne Université

Abstract

From a societal point of view, automated vehicles (AV) can bring many benefits, such as an increase of road safety and a decrease of congestion (Fagnant and Kockelman, 2015; Schwarting et al., 2018). However, due to the highly dynamic environments these vehicles can encounter, technical concerns about their feasibility and ethical concerns about their actions' deliberation, explainability and consequences (Lin, 2016; Gerdes and Thornton, 2016) seems to damper the enthusiasm and optimism that once ruled the automotive sector and the public opinion (Boudette, 2019).

This thesis addresses the core questions surrounding autonomous vehicles today: how they should account the road users sharing the same environment in their decision making and also how they should deliberate to act in dilemma situations. Firstly, a decision making process for the autonomous vehicle under general situations is proposed, and then, as a particular situation, the deliberation under certain collision, with other vehicles, pedestrians or static objects, is implemented.

The discussion focus on an hypothetical AV implementation in urban environments, where velocity is limited and many road users might interact with each other. Therefore, the behavior from other road users is a significant source of uncertainty for the AV (Madigan et al., 2019). In a first attempt to propose a decision making algorithm for the AV that contemplate both generic and dilemma situations, the behavior of road users was considered static during the execution of an AV policy.

A finite horizon MDP, with states representing only the AV's configuration, was implemented, leaving the road users' configuration to be accounted in the reward function as an evaluation of how much risk and performance each action would bring. With this state definition the transition probability was defined as static scalar constant, given that it would represent only the probability of the AV reaching an expected next state given a current one and an action. A dilemma scenario was defined as when all available actions for the AV would result in an accident, which is predicted by the outcome calculated in the reward function estimation. The results of this implementation can be seen in (de Moura et al., 2020).

In the aforementioned article and also in (Evans et al., 2020), a measure of the risk in an accident, called harm, was proposed. It accounts for the difference of velocity produced by the collision and for the vulnerability between the involved road users. This last parameter is proposed to be calculated via a statistical accidentology study from a region or a country, classifying the severity of a collision according to the difference of velocity between the concern road users. The harm represents a scaled measure of risk for each road user due to a collision and is used to deliberate on which action the AV should execute in a dilemma scenario situation.

On the subject of ethical deliberation, (de Moura et al., 2020) and (Evans et al., 2020) take different approaches. The former adapts the general idea of certain ethical theories (Rawlsian contractualism, utilitarianism and egalitarianism) into a mathematical formulation to choose the action based on the expected harm it would produce, which is then used as a deliberation method by the MDP for a state that presents a dilemma situation, instead of the usual value iteration criteria for general situations. The method proposed by (Evans et al., 2020) adds to the harm the concept of ethical valence, which represents the degree of social acceptability that is attached to the claims of the road users in the environment. Then both measures are used by defined moral profiles to deliberate on an action.

Coming back to the question about other road users' behaviors uncertainty, a different approach was considered for such parameter to the AV's decision making. Using an intersection as use case, a multi-agent simulation was implemented, with each decision making from other road users (pedestrians and vehicles) being executed by a set of deterministic rules, to represent the behavior of a standard driver/pedestrian. To estimate the intentions of other users, the AV compares its predictions by assuming an intention vector and updating an instance of a Kalman filter for each user. The distance between the estimate and the observation defines the probability that a user has the assumed intention. To take into account the interactions between the different users, we adopt a game theoretic approach with incomplete information (Harsanyi, 1967; Harsanyi, 1968b) to then calculate the Nash's equilibrium, giving the final AV's strategy.

Keywords: automated vehicles, decision-making, ethical decision-making, artificial moral agents, game theory, behavior prediction

Résumé

D'un point de vue sociétal, le véhicule à conduite automatisée (VA) peut nous offrir plusieurs avantages par rapport au trafic constitué seulement de véhicules avec conducteur, comme une augmentation de la sécurité routière et une diminution de la congestion routière (Fagnant and Kockelman, 2015; Schwarting et al., 2018). Cependant, à cause de l'environnement dynamique et complexe où ces systèmes doivent opérer, des questionnements concernant leur sûreté de fonctionnement et leur comportement vis-à-vis des autres utilisateurs se posent, en particulier sur la capacité du VA de délibérer sous dilemme éthique, d'expliquer ses choix et de mesurer les conséquences de ses actions (Lin, 2016; Gerdes and Thornton, 2016). Ces questions ont finalement ralenti l'enthousiasme naïf et inouï qui avait conquis l'opinion publique et une partie des constructeurs automobiles (Boudette, 2019).

Deux sujets sont abordés dans cette thèse concernant les défis du déploiement du VA dans le monde réel: comment prendre en compte les autres utilisateurs de la route dans le processus de prise de décision du VA dans un environnement urbain, et comment le VA doit délibérer sous une situation de dilemme éthique. D'abord un algorithme de prise de décision conçu pour traiter des situations génériques est proposé; ensuite, à partir de cet algorithme et dans le cadre d'une situation spécifique, quand une collision avec des autres véhicules ou des piétons devient inévitable, la délibération éthique est traitée.

La discussion concernant la prise de décision du VA est limitée à l'environnement urbain, où la vitesse est limitée et plusieurs utilisateurs, protégés par des coques de métal (véhicules) ou pas (piétons, cyclistes, etc.) interagissent les uns avec les autres. Le comportement des autres usagers est une source majeure d'incertitude pour le VA. Dans une première proposition d'un algorithme de prise de décision pour le VA, le comportement des autres utilisateurs a été considéré comme statique.

Un processus de décision Markovien (*Markov Decision Process*, MDP) fini avec des états qui représentent seulement la configuration du VA a été implémenté, avec la prise en compte de la position des autres utilisateurs dans l'évaluation de chaque action. Cette évaluation permet aussi de détecter les situations de dilemme éthique, qui sont traités de deux façons différentes: avec une optimisation (minimisation) du dommage (*harm* en anglais) prévu pour une collision

dans le cadre d'un processus délibération défini à partir d'une théorie éthique (de Moura et al., 2020) ou en considérant des profils moraux qui prennent en compte la valence éthique de chaque utilisateur et le dommage prévu (Evans et al., 2020).

La probabilité de transition entre états a été considérée comme statique pour le processus de prise de décision proposé. Cependant elle est en réalité dépendante du comportement de chaque utilisateur individuellement et de ses interactions avec les autres. Dans une généralisation de la démarche précédente, nous avons introduit une modélisation pour prédire ces comportements.

Une simulation multi-agents est nécessaire pour que l'interaction avec les autres usagers soit bien représentée. Pour estimer les intentions des autres usagers, le VA compare ses prédictions en utilisant un vecteur d'intentions et en mettant à jour un filtre de Kalman pour chaque usager. L'écart entre l'estimation et l'observation définit la probabilité qu'un utilisateur ait l'intention prédite. Pour tenir compte des interactions entre les usagers, nous adoptons une formulation de la théorie des jeux avec informations incomplètes qui permet de fournir le comportement de l'ensemble des agents (Harsanyi, 1967; Harsanyi, 1968b). Le point d'équilibre de Nash permet d'obtenir la stratégie finale du VA.

Mots-clés: véhicule automatisé, prise de décision, prise de décision éthique, agents moraux artificiel, théorie des jeux, prédiction de comportement

Remerciements

Primeiro eu gostaria de agradecer meus pais, Nelson e Daisy, pela confiança que eles sempre me deram e pelo apoio incondicional, em todos os momentos, desde os tempos da natação com treinos às 4:00, passando pela poli e pelas duas mudanças para o exterior. Tenho que agradecer também minha família aqui na Europa, tia Marilda e tio Henk, Tanja, Carol, Dennis, Janwillen, Mats e Louis pelas páscoas, natais, aniversários e por me ajudar em todos os momentos aqui. Sem todos vocês eu nunca teria chegado aonde cheguei.

Je veux remercier aussi Amanda, pour m'avoir eu comme co-loc pendant quatre mois et pour être toujours là quand je besoin. Et Bruno, Jero, Ale et Lucas pour tous les weekends, les fêtes et les invitations que j'ai refusés car je devrais travailler. Ebru, merci pour m'avoir choisi pour travailler dans votre projet et pour me donner la possibilité d'être dans MOB02. Mercedes, Elsa, Franck, JB, Jessy, Jean-Louis et Céline, je suis très content d'avoir appris un peu sur le domaine de chacun de vous, qui sont très éloignés de tout ce que j'ai étudié auparavant et de nos discussions autour des cafés dans les matins et des bières dans les après-midis. Merci beaucoup pour m'avoir reçu de manière si chaleureuse au sein de l'équipe. Le même est aussi vrai pour le VEH08, DT et Guillaume, Mohamed, Nicolas, Laurène, Florent, Flavie, Floriane et Marie, merci pour les cafés, les discussions et pour me soutenir pendant les trois années à VEDECOM. Je remercie aussi le CSE, pour l'expérience quasi-syndicale et le D1 pour la camaraderie et les afters.

Katherine, merci pour avoir facilité mon introduction à l'éthique, sujet avec lequel je n'ai jamais eu de contact avant le début de la thèse; sans ton aide toute le développement de cette thèse aura été beaucoup plus difficile. Merci pour les innombrables révisions de notre article (vraiment innombrables) et pour ta patience et compréhension quand je faisais des remarques sur les aspects fonctionnels dans tes raisonnements (c'est-à-dire tout le temps).

Et en dernier, et certainement le plus important, je remercie Raja. Au long des trois ans de thèse j'ai écouté plusieurs histoires de thésards qu'ont eu des difficultés avec ces directeurs de thèse. Je n'ai jamais compris ces histoires car pendant tout ma thèse tu m'as donné le support, le temps et la confiance que j'avais besoin. Merci beaucoup pour avoir eu la patience avec moi même quand je ne l'avais pas et pour le soutien indéfectible pendant ces derniers mois de thèse.

Chapter 0. Remerciements

This research has been conducted as part of the AVEthics Project funded by the French National Agency for Research, grant agreement number ANR-16-C22-008 and Horizon 2020 program AI4EU under grant agreement 825619.

Contents

Abstract	iii
Résumé	v
Remerciements	vii
List of Figures	xv
List of Tables	xviii
1 Introduction	1
1.1 Automated driving vehicles in society	1
1.1.1 A note on terminology	1
1.1.2 History	2
1.1.3 Current research status	9
1.2 Motivation of the thesis	12
1.3 Contributions of the thesis	14
1.4 Structure of the document	15

2	State of the Art	17
2.1	Decision Making	18
2.1.1	Related works	20
2.2	Ethical decision-making	25
2.2.1	Normative Ethics	25
2.2.2	Artificial moral agents	27
2.3	Behavior prediction	32
2.3.1	Vehicle prediction	33
2.3.2	Pedestrian prediction	38
2.4	Game theory	40
2.4.1	Related works	42
2.5	Conclusion	45
3	MDP for decision making	47
3.1	Theoretical background	47
3.1.1	AV’s Architecture	48
3.1.2	Markov Decision Process	50
3.2	AV decision-making model	55
3.2.1	State and action sets	55
3.2.2	Transition function	57
3.2.3	Reward function	59
3.3	Results and discussion	66

3.3.1	Value iteration	66
3.3.2	Simulation	68
3.3.3	MDP policy results	70
3.4	Conclusion	74
4	Dilemma Deliberation	77
4.1	Dilemma situations	78
4.2	The definition of harm	79
4.2.1	Δv for collision	82
4.2.2	Vulnerability constant	83
4.3	Ethical deliberation process	86
4.3.1	Ethical Valence Theory	86
4.3.2	Ethical Optimization	90
4.3.3	Value iteration for dilemma scenarios	94
4.4	Results and discussion	98
4.4.1	Using EVT as ethical deliberation	98
4.4.2	Using the Ethical Optimization profiles to deliberate	102
4.5	Conclusion	104
5	Road User Intent Prediction	107
5.1	Broadening the scope	108
5.2	Intent estimation	109
5.2.1	Simulated Environment	109

5.2.2	Agent's decision model	112
5.2.3	Estimating other agents' intent	119
5.3	Incomplete game model	125
5.3.1	Nash equilibrium	125
5.3.2	Harsanyi's Bayesian game	126
5.3.3	Decision making procedure	129
5.4	Simulation results	133
5.4.1	Multi-agent simulation	134
5.5	Final remarks	138
6	Conclusion	141
6.1	Final remarks	141
6.2	Future research perspectives	144
A	Simulation set-up	147
A.1	Webots simulation structure	147
A.2	Controlling a car-like vehicle	149
A.3	Controlling the pedestrian	153
A.4	Necessary modifications	154
	Bibliography	157

List of Figures

- 1.1 Automated vehicles envisioned in 1950 (source: (Kröger, 2016)) 3
- 1.2 Interior of the vehicle used to test the machine vision component proposed in (Dickmanns and Zapp, 1987) (source: (Daimler, 2016)) 5
- 1.3 Argo vehicle (source: (Parma, 1999)) 6
- 1.4 Sandstorm vehicle in its final accident (source: Urmson, Anhalt, Clark, et al., 2004) 7
- 1.5 Stanley vehicle in 2005 DARPA Grand Challenge (source: (Thrun et al., 2006)) 8
- 1.6 Boss vehicle in 2007 DARPA Urban Challenge (source: (Urmson, Anhalt, Bagnell, et al., 2008)) 9
- 1.7 Definitions of levels of automation (source: (SAE, 2018)) 10

- 2.1 Possible structures for the AV’s decision-making 19
- 2.2 Hierarchy of possible significations for behavior 33

- 3.1 Three tiered (3T) architecture 49
- 3.2 AV architecture adopted 49
- 3.3 Simplified example of an MDP 52
- 3.4 AV’s kinematic model 56
- 3.5 Example of state space discovery 57

3.6	State transition uncertainty for a_0 and a_3	58
3.7	Performance reward parameters	60
3.8	AV with all the proximity limits	61
3.9	Calculation of velocity projections	63
3.10	Workflow for s_{prox} calculation	64
3.11	Calculating the separation line	65
3.12	Initial simulation setup (other road users' configuration is represented by (x, y, θ, v) , not in scale)	69
3.13	Simulation environment	69
3.14	AV's behavior for $x_{\text{veh}} = 100, 80, 40\text{m}$	71
3.15	AV's behavior for $x_{\text{veh}} = 80, 40\text{m}$ with 5 transitions	72
3.16	AV's velocity, direction and trajectory for $x_{\text{veh}} = 80\text{m}$	72
3.17	AV's behavior for $x_{\text{veh}} = 100, 40\text{m}$ and $\theta_{\text{ped}} = \frac{3\pi}{4}\text{rad}$	74
3.18	Trajectory for $x_{\text{veh}}=40\text{m}$, $\theta_{\text{ped}}=\frac{3\pi}{4}\text{rad}$ for 5 transitions	74
4.1	Projection of future collision into previous states	97
4.2	Possible dilemma situation	99
4.3	Initial simulation setup (other road users' configuration is represented by (x, y, θ, v) , not in scale)	99
4.4	Collision simulation for situation 1	100
4.5	Second simulation setup (other road users' configuration is represented by (x, y, θ, v) , not in scale)	101
4.6	Collision simulation for situation 2	102
4.7	Trajectories for contractarian, utilitarian and egalitarian methods	103

5.1	New simulation environment	111
5.2	Road environment graph structure	115
5.3	Decision tree for vehicles	117
5.4	Decision tree for pedestrians	118
5.5	Illustration of the intersection interaction	135
5.6	Strategies used during situation 1	135
5.7	Strategies used during situation 2	136
5.8	Strategies used during situation 3	137
5.9	Modified vehicle decision for the no collision sub-tree	138
A.1	Main window of the simulator	148
A.2	Longitudinal control for the vehicle	150
A.3	Vehicle's acceleration sampling distribution	151
A.4	The pedestrian object	154
A.5	Pedestrian supervisor field	155

List of Tables

- 3.1 Parameters used in the state space discovery 58
- 3.2 MDP parameters used 66
- 3.3 Road users’ physical properties 70
- 4.1 Δv threshold used for fatality collisions 80
- 4.2 Possible valence hierarchy 87
- 4.3 Possible moral profiles for an AV 87
- 4.4 Optimization procedure based on the moral profile chosen 88
- 4.5 Valence hierarchy 100
- 4.6 AV’s harm for each possible collision for situation 1 101
- 4.7 Road users’s harm for each possible collision for situation 1 101
- 4.8 AV’s harm for each possible collision for situation 2 102
- 4.9 Road users’s harm for each possible collision for situation 2 102
- 4.10 Expected harms and $\sigma_{h_{exp}}$ for contractarian policy 103
- 4.11 Expected harms and Σh_{exp} for utilitarian policy 104
- 4.12 Expected harms and total cost for egalitarian policy 104

5.1	Example of reward table for a specific intent vector c	130
5.2	Rewards for player x	132
5.3	Intent probabilities	132
5.4	Conditional probabilities for $c_x = i_1$	132
5.5	Conditional payoffs for $c_x = i_1$	133

Chapter 1

Introduction

Contents

1.1 Automated driving vehicles in society	1
1.1.1 A note on terminology	1
1.1.2 History	2
1.1.3 Current research status	9
1.2 Motivation of the thesis	12
1.3 Contributions of the thesis	14
1.4 Structure of the document	15

1.1 Automated driving vehicles in society

1.1.1 A note on terminology

The expression "automated driving vehicles" or "automated vehicles", will be preferably used in this manuscript instead of autonomous vehicles. The technical meaning of an "autonomous machine" is a machine that operates on its own, once programmed and started, without human intervention. This is also the meaning of "automated machine"(Dictionary, 1989). Only environment and task complexity lead researchers in robotics to use "autonomous robots" in place of "automated" because the necessary capacities in terms of perception, decision making, learning, etc., are more diverse, and the tasks are more complex. However, "autonomous" and "autonomy" in the general language, in philosophy and in law carry the meaning of deciding one's own goals and making one's own decisions - including moral decisions - independently from other agents' influence, (Christman, 2008): "*...to govern itself, to be directed by considerations, desires,*

conditions and characteristics that are not simply imposed externally upon one but are part of what can somehow be considered one's authentic self". The decision-making algorithms proposed and discussed in this thesis account for usual vehicle behavior control but also for decisions having ethical stakes. This invalidates the use of autonomy as descriptive of such vehicles, since machines do not have morality on their own. Automated driving vehicles accomplish computations as programmed in their algorithms, designed by humans - including through machine learning methods - and are constrained by what these algorithms compute. It's therefore important to avoid any misleading terminology and to use the proper terms.

1.1.2 History

To explain from where the decision-making algorithms that govern automated vehicles (AV) originated, a short history about the development of artificial intelligence (AI) in general, with focus on mobile robotics, will be given, together with the past and current developments for automated vehicles. A picture about the current deployment of automated driving technologies on roads will also be presented, with the most popular use cases and main actors in the domain. This overview will situate the motivations and contributions of this thesis and help to put the research discussed in the next chapters into context.

The history of automated vehicles (AVs) is one that starts long before this century, and in fact even before the well known DARPA challenges (2005 Grand challenge and 2007 Urban challenge), that brought AVs into the public spotlight as a real and imminent innovation. The first interaction between the wide public and cars that can drive themselves was in the New York's world fair in 1939. At that time, GM imagined a trench-like system build into highways to keep vehicles separated. Then the driver could enable the automatic driving system and relax. All such vehicles based their autonomy in mechanical systems (Kröger, 2016) since any kind of computing machine technology was still in its beginnings.

Only in 1960's, with the development of computational systems, semiconductors and the establishment of artificial intelligence (AI), researchers start working towards reproducing some sort of intelligence embedded in a vehicle. By introspection, what makes an agent¹ become an intelligent agent (which hypothetically receives some sort of authorization from society to function with less or even without human supervision) is the possession of a model for the world, which can be used to draw conclusion about how a task can be accomplished using computational reasoning, can be enriched with additional information and allow the agent to execute tasks necessary to complete a goal (McCarthy and Hayes, 1969). However the exigence towards the performance level of said intelligent agent changed through the years. The first definition of artificial intelligence for an inanimate agent, the *Turing Test*, verify if some sort of machine is capable of producing an output indistinguishable from one originated from a human being

¹Which are considered by the literal provenance of the word, from the latin *agere*, or to do, disregarding its philosophical meaning.

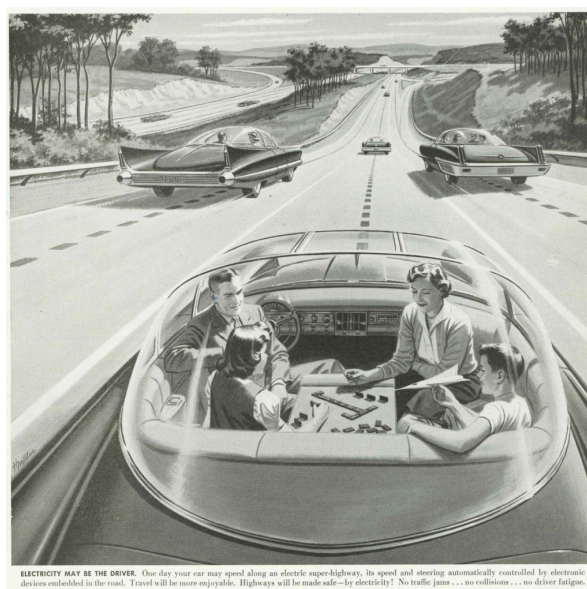


Figure 1.1 – Automated vehicles envisioned in 1950 (source: (Kröger, 2016))

(Turing, 1950). At this point the idea of an intelligent agent was connected to an equivalence between human and machine, but a more accurate definition is perhaps the one given by (Bellman, 1978): "the automation of activities associated with human thinking".

With each advancement in the field, the equivalence was replaced by optimality, defining an intelligent agent as a rational² machine capable of executing the best outcome (or expected best outcome under uncertainty) at each time given a mission (Russell and Norvig, 2009). The same "rationality" is used in (Winston, 1992) through the definition of AI as "the computations that make it possible to perceive, understand and act". This formulation of intelligence, born from the introspection of the human mind (McCarthy and Hayes, 1969), is the first robotic paradigm used to approach the implementation into a machine (Murphy, 2000).

Such concept of rationality, which is connected with the robot's capacity to reason and to construct a world model is based on a knowledge representation consisting of all the sensorial data and *a priori* information about the environment (Murphy, 2000). Planning, state spaces and transitions were used to construct this world model in the STRIPS planning system for the robot Shakey (Munson, 1971; Fikes and Nilsson, 1971), even if at the time the theoretical tool used to implement the planning was predicate logic, which does not offer the capacity to deal with the uncertainties that may exist in an open and dynamic world. But when one considers the entire concept of intelligence, only planning is not enough to reproduce, for example, the behaviors that

²Rational is used in (Russell and Norvig, 2009) as a property of someone or something that acts "doing the right thing". It is not clear if this model of rationality accounts a moral component into the evaluation of what is right. Therefore for this specific case, "the right thing" means quantitatively the best outcome without any moral consideration.

can be observed in animals given its complexity and the need for real-time execution (Brooks, 1986).

Intelligence as only defined by reason, and therefore with an artificial implementation based on a top-down approach like the hierarchical paradigm, practically limits a robot's capabilities to some aspects of what one would expect from an intelligent agent (Brooks, 1991). For example, the necessity for a real-time capacity to respond to stimulus can be in opposition to the execution of a planning routine at every iteration (and it certainly was in the 1960s, when computational capacity was much lower). It is from the search of a more "animal-like" responses that the reactive paradigm came about. The seminal paper of (Brooks, 1986) defines the main characteristics of this bottom-up approach (using a subsumption architecture): many distributed components, each one emulating a kind of behavior, interact with each other and with other components in different layers of increasing complex behavior. No planning is involved in this approach, thus the agents do not have a model of the environment or any type of memory, instead they react to what is perceived from the environment.

However, given the desire to implement an intelligent agent that has at least an equivalent capacity to a human (if not more) to execute some specific task, the reactive paradigm also falls short on finding a viable implementation (even though it produced many successful demonstrators of animal behaviors, as (Brooks, 1989) or even other models of reactive behavior, as (Firby, 1989)). Having no capacity to plan is not realistically possible, since some kind of high-level planning is necessary to execute long-term missions, map the environment and check the robot's performance (Murphy, 2000). So because of this outcome the planning step was added on top of a deliberative process proposed by the reactive approach, in a hybrid configuration. The reactive approach was a functional way to model the low-level real-time behavior for an agent but some knowledge representation of the world is always necessary to plan, as it is shown in (Payton, 1986), (Arkin, 1990), (Giralt et al., 1990) and (Noreils and Chatila, 1995).

From the 1990s forward, two other modules started to appear and compose robotics systems with perception, deliberation and action: communicate and learn (and given the diffuse nature of the behavioral approach, these two functions were implemented by specific components). The former is an essential source of information about other agents in the same environment, allowing a more predictable interaction to optimize the main goal of both. Learning, and more specifically reinforcement learning presented itself as an alternative to the fixed behaviors proposed by the reactive paradigm while using planning to find the best actions to be executed (Sutton, 1990). Perception will also adopt learning for machine vision (Brauckmann et al., 1994).

Throughout the development of the techniques, concepts and demonstrators of AI in robotics, AV research (in this context AV refers to the domain of mobile robotics concerned into producing vehicles for transportation) also took part in such progress from the 1980s. Given the highly dynamical environment they should be able to operate, they touch the main problem of hierarchical architectures, the need for a closed world assumptions, but a reactive approach is

also not completely adapted *a priori* to the problem, since planning and representation is still essential. Thus, since the beginning an hybrid approach was taken to build an AV architecture. Some of the main demonstrators of self-driving AVs will be discussed below³.

One of the first implementation of camera-based perception for AVs was done by (Tsugawa et al., 1979). Two cameras were used in a stereo mount to detect and avoid objects, with a maximum cruise velocity of 30km/h on private tracks. In 1987 (Dickmanns and Zapp, 1987) successfully tested an AV capable of lateral and longitudinal guidance by computer vision in a public highway for 20km with a maximal velocity of 96km/h. Such implementation extracted the road boundary markings from images produced by a CCD TV-camera to determine an estimation of the road geometry (curvature). Then this measure was updated at each iteration with the vehicle state using a Kalman filter and used to calculate the longitudinal and lateral control.

The progress of self-driving continued from 1987 to 1995 due to the Prometheus project, an European project that assembled car manufactures and research organizations to improve road traffic safety from a vehicle perspective and from an infrastructure perspective. At the same time the machine vision applied to AVs also was focus of research in the US and in Japan (Dickmanns, 2002). In (Dickmanns, Behringer, et al., 1994) a vision system composed of two platforms of two cameras with different focal points was used to interpret the situation ahead and behind the vehicle, with traffic sign detection, moving humans identification, obstacles and road lanes finding. In the same project, (Brauckmann et al., 1994) proposed a vision system of sixteen cameras, capable of detecting road vehicles all around, with a neural network capable of detecting the rear of vehicles, a lateral blind spot surveillance and a short range visual object detection to automate stop and go situations.



Figure 1.2 – Interior of the vehicle used to test the machine vision component proposed in (Dickmanns and Zapp, 1987) (source: (Daimler, 2016))

³A more exhaustive list of projects can be found in (Claussmann, 2019)

All of the developments cited were finally concentrated in the VITA II demonstrator (Ulmer, 1994), which had the following capacities: lane, distance and speed keeping, lane change, overtaking and collision avoidance. For the collision avoidance, two subsystems co-exist in the vehicle: one that calculate controls based in a potential field approach and another that uses a state transition model to represent all possible situations that may arise in traffic, invoking vehicle's maneuvers when necessary. Another European project, ARGO, also demonstrated a AV prototype capable to drive long distances without a driver only with passive sensors (two cameras and a speedometer) (Broggi, Bertozzi, et al., 1999). The road was reconstructed using monocular images while detection and tracking of other vehicles was done with pattern matching. For the vehicle control, a variable gain proportional controller with a non-holonomic bicycle model and a quintic polynomial approximation produced the final trajectory. This demonstrator drove 2000km in Italian highway network (not continuously). Problems related to illumination sensibility in image acquisition and some control instability in high speeds were observed along during tests.



Figure 1.3 – Argo vehicle (source: (Parma, 1999))

Then it came the DARPA (Defense Advanced Research Projects Agency) challenges. The race consisted in completing a 143 miles (approx. 230 km) route in the Mojave desert. No team were able to complete the very first challenge in 2004, with the Red team from Carnegie Mellon University traveling the farthest. The course route were given only 3 hours before the start of the race, so the trajectory could not be calculated beforehand and since it is a off-road challenge (thus no traffic infrastructure were available) the number of sensors needed to assure high speed automated driving increased in number and complexity in comparison with the previous attempts in highway environments. An effort from Red team was made to build a detailed map from the entire region where the race could take place, with possible routes, geographic characteristics, elevation and satellite images before the race. All this information was used, with the course waypoints, to calculate a pre-established path, which should be tracked by the vehicle during execution based on sensor readings (GPS for position and LIDAR and stereo vision for terrain perception). Many incidents were reported, some related to the inability to modified the pre-planned trajectory to avoid obstacles, but the one that provoked the final accident was due to an

excessive sharpness in the pre-planned trajectory, tendency to cut corners from the pure pursuit control and measurement errors in the GPS readings (Urmson, Anhalt, Clark, et al., 2004).



Figure 1.4 – Sandstorm vehicle in its final accident (source: Urmson, Anhalt, Clark, et al., 2004)

Since no team was able to finish the event, it was repeated in 2005. Five teams successfully finished the course this time, with the Stanford team as first. The winner vehicle also possessed a rich array of sensors as previously, from lasers, radars and cameras for environmental detection to IMU, GPS and wheel encoders for position measurements. The Red team vehicle already implemented an architecture close to the hierarchical paradigm, but the Stanford's vehicle went beyond and based its software in the three-layer architecture (will be detailed in the next chapter). All position estimation measures were calculated using an UKF with a modified estimation for GPS outages situations, where accurate vehicle modeling is necessary to maintain pose errors contained. No environment *a priori* map was used, with the vehicle executing the obstacle detection online and with laser measurements. Such detection is based on a probabilistic grid approach with parameters tuned by a discriminative machine learning algorithm. For road finding, since the laser range was insufficient given the speed necessary to complete the challenge, all readings of safe terrain from laser origin were projected into the perceived pixels of the camera image. Then, using a mixture of Gaussians the image pixels were classified and the drivable surface was determined. The base trajectory was pre-calculated and modified online to avoid obstacles while staying inside the course limits (Thrun et al., 2006).

As one can see, the complexity of the perception systems increased substantially since the days of (Dickmanns and Zapp, 1987; Brooks, 1986) in terms of price and capacity. The other main advancement that allowed such gain of performance is the availability of more capable computers. Most of the machine vision processing done in (Dickmanns and Zapp, 1987) was done in specific hardware; in (Thrun et al., 2006) the embedded processing unit had 6 processors. Advances in machine learning were also capitalized in machine vision for the road and object detection (although neural networks were already present in (Dickmanns, Behringer, et al., 1994)).

In 2007 the Urban Challenge took place, which consisted in a course of 97km through



Figure 1.5 – Stanley vehicle in 2005 DARPA Grand Challenge (source: (Thrun et al., 2006))

an urban environment. This time it was the Carnegie Mellon University team that won the first place. Their vehicle, called Boss, used a similar architecture from Stanley but with a focus on active sensing. Given the nature of the challenge two different navigation modules existed: one to generate a trajectory for road environments and another for non-structured zones (parking zones). In roads multiple trajectories were generated from the middle of the current lane, allowing the vehicle to choose the best lane to avoid static or dynamic obstacles. Outside roads, where there was no nominal orientation, the vehicle model was used to generate offline a set of possible maneuvers available for each state (in this case the state is composed by $(x, y, \theta, v)^4$). Then this space was searched online to form a trajectory, from the goal pose to the current position. Moving obstacles were tracked using or a fixed-body hypothesis or a point-based estimation and then classified as moving or not, while static objects first were identified in a instantaneous map to then be filtered out or added into the temporally map (Urmson, Anhalt, Bagnell, et al., 2008).

But the real differential with the previous challenge was the mission planning. The environment now had intersections, that needed to be managed in addition to the structured and unstructured environment. Thus three behaviors were available: lane driving, intersection handling and goal selection (for unstructured road). Each one of these behaviors had functional components that could be used if necessary, in a reactive structure (Urmson, Anhalt, Bagnell, et al., 2008).

From this point on there have been some initiatives from institutional projects, but the private initiative took hold from the domain and started to develop real applications with the technology displayed until this point (Anderson, Kalra, et al., 2016). Research beyond the 2010s started to focus on the study of V2X communications combined with automated driving, as shown by The Grand Cooperative Driving Challenge (Englund et al., 2016), an event in the molds of the DARPA Challenge about wireless communication usage in traffic. Two other examples are Broggi, Cerri, et al., 2015, a test in open street without driver in Parma, and Ziegler et al., 2014, an automated driving test in the same route where the first cross-country automobile journey took place, between Mannheim and Pforzheim in 1888.

⁴position in 2D, direction and velocity



Figure 1.6 – Boss vehicle in 2007 DARPA Urban Challenge (source: (Urmson, Anhalt, Bagnell, et al., 2008))

1.1.3 Current research status

Many applications for limited domain of operation are already present in the real world today. The most popular system deployed are small shuttles operating in closed environments and with limited velocity. For example, university campuses are very interesting environments to test such technology and also to inquire the users about the service's quality and its social acceptability (Berrada et al., 2020; Nordhoff et al., 2021). Even if from a technical standpoint they do not deliver what is expected – fully automated driving; Nordhoff et al., 2021 and Mouratidis and Cobeña Serrano, 2021 had maximal velocity of 18km/h with automatic longitudinal control but any obstacle avoidance needed to be executed by the operator and Berrada et al., 2020 had a maximal velocity of 30km/h – their deployment in specific domains is useful to start studying the societal effects that the automated driving can have in our society, which can vary from an acceleration of urban sprawl (Soteropoulos et al., 2019) to an improvement of public transport coverage in urban zones with few or infrequent buses (Mouratidis and Cobeña Serrano, 2021).

Usual car manufacturers already adopted level 1 or level 2 autonomy solutions. According to the definition given by (SAE, 2018), level 1 automation is characterized by longitudinal or lateral automatic control and level 2 both directions are controlled by the vehicle, but always under de strict driver supervision. The latter is the frontier between having the driver in the loop⁵, since from level 3 it is the machine that takes over – although in case of accidents or any type of failure the driver must be ready to take over. For example, a very common system nowadays

⁵To be more precise, the difference between both levels is that level 2 assumes that no object or event detection, recognition, classification, response preparation and response execution will be done by the vehicle, while level 3 internalize such tasks while maintaining the option to cede control for the driver (SAE, 2018)

is the adaptive cruise control (ACC), which controls the velocity of a vehicle when engaged, always checking ahead for obstacles. Another popular feature is the automatic emergency braking (AEB), a system capable of detecting obstacles in front of the vehicle and break under danger of collision. Its efficacy to reduce rear-end vehicle to vehicle (V2V) collision by was shown in Fildes et al., 2015 for low speed (30km/h to 50km/h), but these systems still have difficulty to prevent collisions with pedestrians in straight roads and in left and right turns (AAA, 2019).

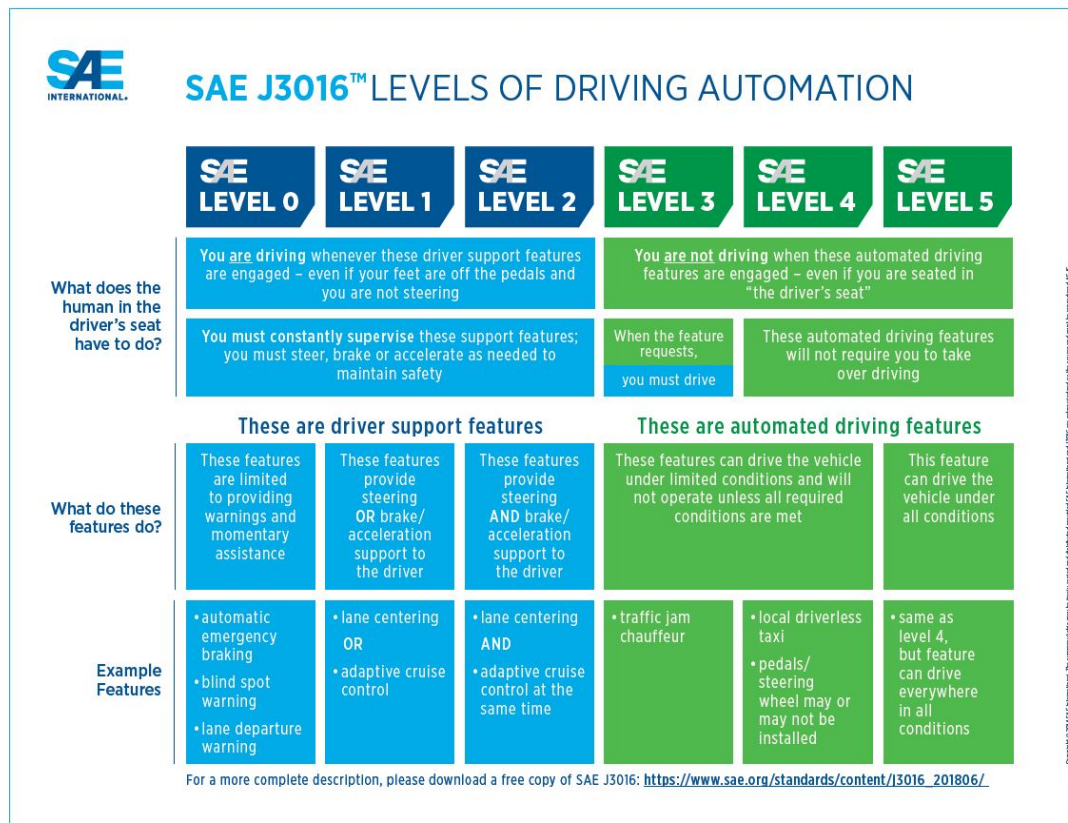


Figure 1.7 – Definitions of levels of automation (source: (SAE, 2018))

Both aforementioned systems are classified as level 1. A level 2 system could be composed for example by an ACC and a lane keeping system functioning at the same time, which is a combination done by some brands (Toyota, Tesla and Mercedes, for example). No examples of complete level 3 automation for open road usage⁶ exist in commercial cars today due to the lack of regulatory legislation up to 2020, since level 3, often called conditional autonomy, is the first stage of automated driving, thus, *a priori*, manufacturers could be liable for accidents during execution of the given system, even in misuse cases.

⁶The *open road usage* was added to exclude parking assistants from the comparison, given that they are classified as level 4 but they have a very specific *raison d'être*.

From January 2021 forward, 60 countries (including the EU, Japan and Canada) adopted a UN resolution that sets up the regulation of automated lane keeping systems (ALKS)⁷, limited to 60km/h and roads with separation between opposite directions and without pedestrians or cyclists. The regulation considers the system as primary driver and sets safety requirements for emergency maneuvers, transition demands and minimal risk maneuvers. With level 3 capabilities also comes two mandatory hardware components: a driver availability recognition system, which detects the driver's presence and readiness to take back control, and a data storage system, sort of a black box for the automated system (UNECE, 2020). Thus, level 3 capable vehicles should be expected in the near future.

Differently from the established car manufacturers, that avoid taking considerable risks, some companies are working in level 4 urban AVs. Two of them already have automated demonstrators working in the streets: Waymo and Cruise. The first, a spin-off from Google, is the company with the most experience and mileage concerning simulation and real kilometers driven in automated mode. It even created its own LIDAR sensor, which is used to give the vehicle a 360° vision of the environment. Cruise, a start-up that has as investors GM, Honda and Microsoft, has a automated taxi service in central San Francisco without backup driver since October 2020, after years of tests in the suburbs. Zoox, bought by Amazon in 2020, does not have vehicles retrofitted with sensors to be autonomous, it has taken the approach to design a vehicle (in this case a shuttle) from the ground up to be autonomous, with 4-wheel steering and bidirectional driving.

The case of Tesla for automated driving is more complex. Tesla's Autopilot has been present in the streets since 2014 and it has generated controversy given its array of accidents and problems, since its first accident in 2016 (Yadron and Tynan, 2016), when a Tesla drove under a perpendicular lorry after failing to distinguish the latter from the bright sky (Tesla, 2016), until another one in 2018 when a Model X hit a previously damaged median barrier at approximately 114km/h (Chokshi, 2020). In this last accident the National Transportation Safety Board, the federal agency that investigated the incident pointed that the accident was caused by the driver who has not paying attention to the road and that in 19 minutes of automated driving the driver kept its hands on the wheel for a total of 34 seconds (Chokshi, 2020).

In 2017, as the result of an investigation of the 2016 accident (BBC, 2017), the same agency had already issued two specific recommendations, for Tesla and all other manufacturers: to limit the use of automated systems to the conditions accounted during design and to make sure driver keep the focus on the road and their hands on the wheel (Chokshi, 2020). More recently it was announced that US federal regulators investigate 23 accidents involving Tesla's vehicles, potentially using AutoPilot (it is not clear yet if the system was enable and functioning at the moment of accident in all accounts). In these accidents there are instances when the Tesla crashed into a stopped police vehicle without decreasing its velocity, another rear-ended a police vehicle and one, which happened in February 2021, very similar to the 2016 lorry accident (Boudette, 2021).

⁷Despite the name, longitudinal control is also executed by the system

Tesla's new automated lane change system is the main focus of attention now. Since 2019 the Autopilot function can deliberate if a lane change is possible and necessary, and depending on the settings used, it can execute or ask the driver permission to execute. Up until 2019 the driver itself had to initiate the lane change. According to Barry, 2019, the functionality can cut off cars in a way that drivers do not usually do. Other problems are mentioned, like the difficulty to interact with high-speed vehicles coming from the rear during lane change and problems to merge into traffic. Thus, one of the characteristics that an AV should have (or at least some vehicle with advanced ADAS), the predictability of its actions by other drivers, sometimes is not present. By hindsight, maybe the system executes such maneuvers because it has a greater analytical capacity than a common driver, but there is no way to know since the inner works of the system are secret.

This privacy problem is also present in all other automated systems from other manufacturers, throwing all the responsibility to policymakers, that will need to establish the homologation process for automated driving systems. This secrecy also prevents the inspection or the investigation of these systems by the scientific community (or the public in general) at large. Despite an expected re-calibration on the hopes and dreams about deploying fully functional AVs in the roads at the beginning of this decade (Boudette, 2019), the technological development in the private domain has already solidified and it is advancing towards a real prototype.

1.2 Motivation of the thesis

Simply put, the biggest selling point for the deployment of automated vehicles is a possible decrease in the number of accidents, given that approximately 93.1% of them in the US are due to human error (NHTSA, 2008). Such decrease is taken as a condition for the licensing of automated systems according to guidelines from regulators (Luetge, 2017; Bonnefon et al., 2020). Automated driving is estimated to represent a significant decrease in the number of deaths in road accidents per year, which would be a gain for society as a whole, even if the gains of such revolution would be felt in its majority in high-income countries, where the average rate of death per 100.000 is 8.3 in comparison with 27.5 for low-income countries⁸ (WHO, 2018). There are other reasons to study and push for AVs in the streets, for example to reduce traffic and congestion (Narayanan et al., 2020), improve transportation efficiency (and therefore reduce pollution) (Wadud et al., 2016), increase accessibility to the elderly and the disabled (Fagnant and Kockelman, 2015), but spare lives is the most urgent issue.

It is undeniable that an automated system can, *a priori*, have a better analytical capacity than a human. This is even more evident if one considers afflictions that only affect humans, for example alcoholic intoxication, lack of attention, disregard towards signalization or the traffic code, drowsiness and many others. So, on the account of accidents involving these situations an AV can make a difference. However, the interaction between AV and other drivers can itself

⁸It is also important to mention that road traffic injuries are the eight leading cause of death for all age groups and the leading cause for the range of 5-29 years according to WHO, 2018.

introduce new sources of accidents. This interaction between machine and human is even more critical considering that it is expected from the AV the capacity to solve situations that would result in an accident, thus possibly producing a non-obvious behavior for the other road users. The AV's ability to account for these interactions and predict the reaction of all other road users is important and, by nature, these reactions are uncertain.

The AV itself also has some shortcomings. For example, every perception source has an inherent noise, maximal range and other specificities related to each type of sensor (for example GPS do not work well in dense urban areas and LIDAR measurements are disturbed by rain), there is a limited time for reasoning and hardware failures are always possible. But even if one assume that all the AV system and the infrastructure function at an ideal performance level, the unpredictability about the other road users remain (considering situations were there is an human component in the environment). Thus, even in an ideal world some uncertainty always exists and, because of that, the possibility of an accident happening cannot be disregarded (Goodall, 2014). According to Teoh and Kidd, 2017, in 2016, Google's AV registered 3 reportable incidents in tests at Mountain View, California. The comparison with real drivers shows that the AV is safer than a normal driver, but without statistical certainty (95% confidence interval). Looking the numbers of disengagement initiated by the backup driver, out of 13, 10 would have caused a contact. A more detailed study is done in Blanco et al., 2016, which also does not have enough data to reach a conclusion but observes the same tendency, this time adjusted by US national statistics⁹.

It is clear from this point that a probabilistic approach needs to be adopted at the planning component of the AV, to account the behavior of other road users and all the related uncertainty. But if an ethical dilemma situation presents itself in the planning, how the AV should deliberate? Firstly, an ethical dilemma is defined as a situation where harm will be inflicted by the AV toward other road users for every possible action executed (Evans et al., 2020). If the planning for some non-dilemma situation predicts a possible collision (i.e., there is a risk of harm for some of the users), the simple imperative that the AV must not cause accidents can be used to avoid completely this situation, and in this normal situation other parameters would be used to drive the decision, as for example the AV's performance given the chosen action, proximity to other road users, etc. But this imperative is invalid in a dilemma situation, and the other ones are not pertinent, since deliberation about the harm distribution must be the priority, which is dependent on how one thinks it is pertinent, fair, moral to distribute the risk of harm between each one in the environment. Such deliberation is necessarily an ethical one (Evans et al., 2020).

Considering the predicament mentioned, this is the main motivation of the thesis: **to investigate how an automated vehicle should deliberate under normal situation and in dilemma situations.** From a defined decision process for normal situations an additional component must be integrated, to enable the AV to reproduce some sort of moral deliberation, given a pre-defined ethical theory, to determine how the distribution of risk needs to be done if the planned situation

⁹Another interesting problem with a direct comparison of accidents is that all accidents are reported by AVs, which is not true for human drivers, creating a bias against the AV. Even so, the final conclusion in Blanco et al., 2016 says that only for the least severe crashes the AV is safer at a statistically significant level.

manifest itself in the real world.

The reasoning to argue that ethics is necessary in the decision-making component of an AV presented above might be classified as a top-down justification, since it starts discussing the general capabilities of an AV to them point where the need for ethical reasoning arises. The inverse direction could also be used to justify the same need: assuming that during the life-cycle of an AV and that because of the highly dynamic environment characterized by urban areas, at least once it might face a dilemma situation, maybe only in planning, but if it is predicted it might materialize itself. In this case, how it should deliberate? Should it choose at random, or should it beforehand have some capacity to choose one action over the other (Lin, 2016)? This consideration also assumes that it is improbable that an AV could be treated as a perfect implicit ethical machine, i.e. a system capable to avoid any type of ethical issues at all times (Moor, 2006).

1.3 Contributions of the thesis

The following contributions were implemented and tested through a multi-agent simulation of an environment containing vehicles, pedestrians and AV.

- **Definition of a harm measure for collisions:** Defined by the difference os velocity between the two road users involved in the collision and scaled using a constant that represents the inherent fragility of a road user.
- **Ethical deliberation according to different moral theories:** Rawlsian contractualism, utilitarianism and egalitarianism were used to inspire three different optimization criteria for the risk of harm in an accident.
- **Reward function combined with ethical consideration:** An ethical component assumes the role as reward function when necessary, projecting future dilemma situations into the present to actively avoid it.
- **Model of interaction between road users using an probabilistic game theory approach:** Accommodation of other road users probable behavior into the AV's decision-making structure using a game with incomplete information to model expected reward given each possible action for the AV.

Results from the ideas and propositions during this thesis are presented in the following publications:

Peer-Reviewed Journals**Title:** Ethical Decision Making In Autonomous Vehicles: The AVEthics Project**Authors:** Katherine Evans, Nelson de Moura, Stéphane Chauvier, Raja Chatila & Ebru Dogan**Journal:** Science and Engineering Ethics, Springer**Volume:** 26, **Pages:** 3285-3312, **Year:** 2020, **Reference:** Evans et al., 2020**Peer-Reviewed Conferences****Title:** Ethical decision making for autonomous vehicles**Authors:** Nelson de Moura, Raja Chatila, Katherine Evans, Stéphane Chauvier & Ebru Dogan**Conference:** IEEE Intelligent Vehicles Symposium**Place:** Las Vegas, US (virtual), **Date:** October 2020, **Reference:** de Moura et al., 2020

1.4 Structure of the document

Chapter 2 presents the state of the art in the most relevant domains addressed in this thesis, decision-making, ethical decision-making, behavior prediction and game theory. Comments about the structure and types of methods inside each domain is followed by the mention of a set of representative publications about the methods discussed.

Chapter 3 proposes the decision-making algorithm to process normal situations. Before diving into the implementation details, a discussion about the architecture used and the theoretical definition of a Markov decision process are done. Next, the details about the implementation of a MDP are given, which is followed by the policy determination procedure and the results obtained in simulation.

Chapter 4 proposes two possible deliberation procedures to determine an action to be executed in a dilemma situation. The chapter starts with the definition of what is a dilemma situation, to then propose a severity measure of an accident. The two deliberation methods are treated next and to finish the chapter, the simulation results are discussed.

Chapter 5 's main goal is to relax the assumption about the invariability of the other road users behaviors during the execution of the AV's policy. Starting from the determination of the errors that might drive the AV into a dilemma situation, the a new use-case and the definition of the deterministic decision process for other agents in the simulation is addressed. The AV's estimation of the other road users intent and the game theoretic model are proposed, in a first moment only considering pure strategies and then modifying the proposed framework to account for mixed strategies.

Chapter 6 closes this document with the final remarks and future research perspectives concerning this thesis thematic.

Appendix A includes a presentation of the simulation environment and set-up, based on Webots, to which we have made some adaptations.

Chapter 2

State of the Art

Contents

2.1 Decision Making	18
2.1.1 Related works	20
2.2 Ethical decision-making	25
2.2.1 Normative Ethics	25
2.2.2 Artificial moral agents	27
2.3 Behavior prediction	32
2.3.1 Vehicle prediction	33
2.3.2 Pedestrian prediction	38
2.4 Game theory	40
2.4.1 Related works	42
2.5 Conclusion	45

To propose a decision-making system capable to deal with the situations mentioned in Chapter 1 it is necessary to consider two main subjects. The first is the algorithms that reason about the world around the AV and allow it to make decisions in place of the driver during the mission. They usually consider criteria about the consequences of actions towards achieving a goal as the only (or the most important) measure to find the best action. Three domains from this first subject are explored in this thesis: how to predict the behavior of other agents, how to model interaction between agents and how to account for uncertainty.

The second subject concerns the fusion between ethical considerations and normal decision making. The resulting methods from this domain are grouped on what is called ethical decision-making methods; their imperative is to determine the best action according to some ethical theory of what is right, good or fair. A discussion about the scope of ethics needed to implement these methods and the one used in the algorithms that are proposed in subsequent

chapters will be followed by a report of the past and current state of the art in the field of artificial moral agents.

Two other domains are of importance, this time for the questions addressed in chapter 5. To relax the assumption that no interaction between road users happens during the AV's mission it is necessary to, first estimate the intention of each road user and then account for possible interactions. Therefore, a discussion about behavior prediction is due, as is one concerning game theory, the most appropriate tool to model interaction between agents that present some form of rationality.

2.1 Decision Making

As the title clearly states a decision making system uses its input, that may be sensor information, *a priori* data or any sort of exploitable data, with its decision structure to deliberate on what the automated¹ system (AS) should do, given a mission or an objective. Considering the hierarchical paradigm, decision-making is the embodiment of the plan step (for mobile vehicles motion planning² is also a part of the plan step, if it is not handle by the decision-making itself), and for the reactive paradigm the decision is indirectly implemented by the sum of all components in all layers.

There is a wide range of methods for decision making in many tasks and situations. Autonomous systems cover a wide range of applications, from the automated vehicles (aerial, terrestrial or aquatic) to robotic manipulators, but the discussion here will be centered in the decision making methods for automated vehicles. In the AV domain, the organization established by the hierarchical paradigm is often used to define the AV's architecture (more details on this point will be given in chapter 3), with the sense-plan-act layers. These three layers can also be divided in sub-components, which is typically the case of the planning layer: its task can be separated in strategic, tactic and operation procedures. Such classification is used in (González et al., 2016) and (Sharma et al., 2021) and in most of the AV published research, although with different names. Decision-making methods concerns exactly the planning layer, with the tactical component (path planner in the former and behavioral planner in the latter) being of special interest since it is where the global path is adapted to the environment constraints and the behavior of other road users.

The realization of this architecture can be done in different ways. One notable difference

¹The same reasoning as for the discussion between automated and autonomous vehicles is valid here; the term more commonly known is autonomous systems, but automated systems will be used.

²There might be some confusion related to the frontier between motion planning and decision-making. It will be considered that decision-making is more general than motion planning and thus can refer to decisions about other variables than the trajectory to be executed, which the output of each motion planning method necessarily refers to; for example, considering the AV context, the decision-making can reason about the necessity to adopt some general profile of defensive driving in certain situations, which does not have a direct relation with the trajectory to be followed.

concerns the flow of information inside an AV. According to (Schwartz et al., 2018), these organizations can be resumed to three representations, shown in figure 2.1 (they are fairly similar to the usual architecture for an AV, as it will be explored in section 3.1). The first and the second approaches have the perception and control components well defined, having as difference the separation or not from the motion planning and the higher decision-making process. Such difference depends upon the characteristics of the model, for example, if some internal motivation or long-term strategy is calculated as a separate process, then this result necessarily becomes the input for the motion planning, that may deal not only with the geometric constraints of the trajectory but also with the interaction with other road users. On the other hand, methods as Markov Decision Process (MDP) can fuse both procedures in one only component, even if some internal variable needs to be calculated as to guide the trajectory determination.

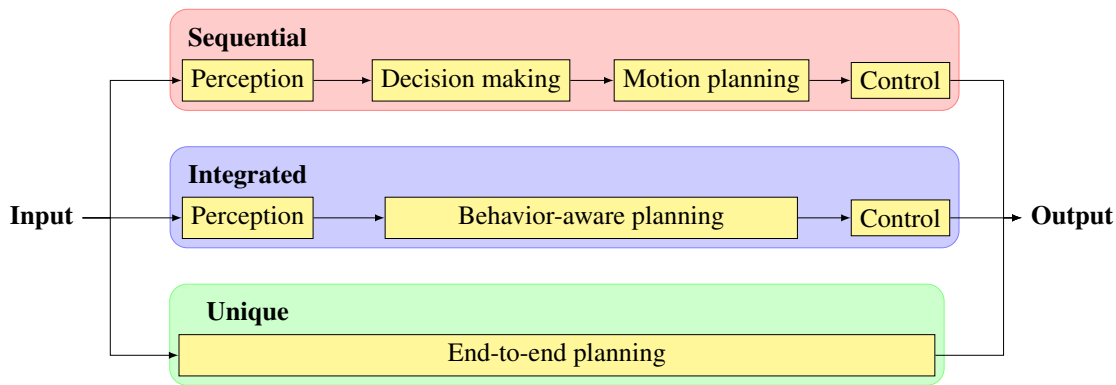


Figure 2.1 – Possible structures for the AV's decision-making

The third option has recently been in the spotlight due to the advancements delivered by deep-learning methods, which enables the learning procedure to be executed directly from the input data mass without any type of separation, rule abstraction or clear hierarchy. However, the first application of end-to-end driving was proposed years before, by (Pomerleau, 1988), which used a neural network to teach a vehicle how to keep driving in the same lane. Given the critical characteristic of automated driving, this implementation philosophy is somewhat contentious, since these systems function as black-boxes and that they are approximate methods without a measure of limits to which the output values can assume.

Considering the frameworks of possible methods and proposed organization for the flow of information, some examples of decision-making application into for AVs will be discussed next. A preference will be given for MDP-inspired decision-making algorithms, since it directly relates to all the methods proposed here, and, together with end-to-end learning based on neural networks, constitutes the two most used approaches for AVs decision-making implementation.

2.1.1 Related works

Given the importance of the uncertainties involved in automated driving, finite state machines are not commonly used to embody the decision-making of an AV. Markov decision processes are much more adapted to this task, which is observed by the number of publications that propose such methods. Aiming to solve a partially observable Markov decision process (POMDP) in real time, (Ulbrich and Maurer, 2013) organized the entire decision system using two levels: one that detects if a lane change is possible and/or beneficial, using two different signal processing networks, and a POMDP that has as input these networks and chooses the action to take the AV to a state from a defined set of eight possible states. These are defined by three boolean variables, that represents if the lane change is possible, if it is beneficial and if it is in progress, while the action set is composed by continue in lane, change lane and abort lane change. The main idea is to encapsulate into the POMDP all the sensor noise handling while maintaining a low complexity so as to allow real-time execution.

Still in the lane change use-case, (Wei et al., 2011) uses a point-based Markov decision process (QMDP) to implement a decision-making capable to account for three types of uncertainties: sensor noise, perception limitations and other vehicles' behaviors. Such interpretation of the usual MDP algorithm uses Monte Carlo sampling from the sensor readings to define the states which will have the reward calculated, instead of using a POMDP. The sensor uncertainty and the perception range limitation also modifies the transition probability calculation during the policy evaluation.

In (Brechtel et al., 2011) an MDP with a Dynamic Bayesian Network (DBN) as a transition model is proposed. The state space is discretized in rectangles of equal size so as to allow the definition of a finite set of states, while the use of an DBN gives the possibility to express multiple abstraction levels. The model resolution is obtained by applying a Monte Carlo approximation, that is solved partially offline and then refined online, allowing a real-time execution.

Two types of uncertainty are the focus of (Brechtel et al., 2014) proposed POMDP model: the environment evolution uncertainty and the limitation of sensor. In order to address these parameters, the transition probability function is defined by a DBN, similar to (Brechtel et al., 2011) but without the state space discretization, where the current state is related to the environment context, the planned trajectory and the action, transitioning to a new state with a certain probability. Finally, the model is solved using the Monte Carlo Value Iteration, an algorithm proposed in (Brechtel et al., 2013) which determines a discrete representation of the state space and calculates the α -vectors; such representation is formed offline and then used in real time for a T-intersection use-case.

Addressing the same two uncertainty sources, (Hubmann et al., 2018) defines a POMDP with a limited number of parameters, searching to determine the longitudinal acceleration necessary for the AV to cross an intersection. The lateral motion is considered to be extracted from

the road geometry. Each vehicle in the environment is represented in the state by its position, longitudinal velocity and its assumed path, which is not observable as the other two variables, but it is determined from the current state of the vehicle in question and it has its prediction uncertainty calculated by a naive Bayes instance. This POMDP is solved using an adaptive belief tree (ABT) proposed in (Kurniawati and Yadav, 2016), a method that uses an unweighted particle filter-like approach to create a belief tree while selecting with actions should be favored in the particle sampling using an upper confidence bound determination (UCB).

Bayesian change-point detection is used in (Galceran et al., 2017) to classify observed states from other vehicles based on predefined action-policies, or intentions, which are formed by low-level commands. These distributions are then sampled to determine the possible actions that need to be evaluated, for the entire environment (AV and other vehicles). Then the expected reward for each possible state are maximized to find the best policy to be executed.

Without a high penetration of automated vehicles in the real world vehicle-to-vehicle communication will remain based on non-verbal signals between drivers, as is today. The work in (Lenz et al., 2016) offers a method to participate in such exchange using Monte Carlo Tree Search (MCTS). It is defined that two vehicles are interacting if their respective reward functions are influenced by each other, which means that changes in the value function (the expected reward considering some fixed horizon) happen due to the behavior from another vehicle. In the defined context it is also assumed that all vehicles act in a rational manner, i.e. they try to maximize their reward function. The MCTS proposed is similar the ABT algorithm of (Kurniawati and Yadav, 2016) in the sense that it also uses a tree structure and the UCB algorithm to select which action should be used to expand the tree during its evaluation process. For each vehicle the reward function (in this case a cost function) is defined by the cost for the considered vehicle, itself defined by typical parameters like lane change cost or acceleration cost, calculated using a weighted sum, and a cooperative cost function defined as the sum of the vehicle cost and all other vehicles' cost multiplied by the cooperation factor. The first one is used in the tree expansion and the second updates the cost in the back-propagation. Multiple scenarios are tested, with different cooperation levels.

The starting ideal used in (Broadhurst et al., 2005) is to reason about the road users future paths and to assess the related danger from the combination of all of them. Pedestrians and vehicles were given geometric bodies (squares and rectangles respectively), with the latter following a front-wheel steering single track model. Two different goals are considered, straight line following and road following, so as to determine the future control inputs, which has its risk measured by a conditional probability, dependent of these control inputs. The integral resulting from the conditional probability is calculated using a Monte Carlo random sampling, which samples behaviors from different trajectories defined by different intentions (as stop, turn and stop, change lane, etc.). The end result allows the algorithm to detect intersections in the path of each road users, and according to the probability of this intersections to raise a warning of unsafe behavior in the environment.

Focusing in the intersection problem, (Barbier et al., 2018) proposes an POMDP to model the uncertainty related to sensor readings and other vehicle behavior but also key performance indicators to measure the calculated policy quality. The state evaluates is defined by the distance to the intersection and the speed of the ego-vehicle and the other vehicle together with the calculated intention from the other vehicle and the expectation for both vehicles in regards to each other. These three last variables, intention and expectations are calculated as proposed in (Lefèvre, Laugier, et al., 2012), which motivates the Bayesian network representation of the transition probability. A method known as POMCP, proposed in (Silver and Veness, 2010), solves the POMDP model using a combination of Monte Carlo update of the belief state and a MCTS representation. The key performance indicators are divided in four categories: safety, navigation, trust and comfort; they were used to evaluate two different policies calculated using different weights for the rewards' component terms. The set of indicators were able to represent the performance of the policy related to each of the categories and the trade-off available between them to improve the performance.

Decision-making can also be implemented using fuzzy logic; (Claussmann et al., 2018) employs a multi-criteria decision-making based on fuzzy logic to select the most appropriate trajectory to be executed by the AV in an highway scenario. A wide range of criteria is considered, from vehicle and passenger safety to passenger comfort and energy savings, to select which trajectory has the best behavior (if the AV should change lane or stay in the same one) and the most desirable acceleration/deceleration profile. The usefulness of such approach becomes clear due to the capacity to mix a fairly big number of criteria (eight in this publication) and reach a decision concerning multiple candidates of trajectories. Another application of fuzzy methods is the estimation of risk during driving. Using a two-level architecture (Derbel and Landry Jr., 2018) uses a fuzzy approach to measure the global risk of a situation.

Any type of negotiation between humans, that inevitably will fall under the AV's purview, can be focus of a learning method. Using the lane swap that might happen in an highway environment as an use-case, (Schmerling et al., 2018) proposed a learning algorithm capable of determine probability distributions over possible behaviors from human to human training data (obtained experimentally), to then determine a policy using conditional variational auto-encoders. Using as input the history of the interacting driver and the optimal response from the AV, this network is capable of estimating the future response of the driver, from a Gaussian mixture models used to evaluate the probability to observe some specific answer from the driver.

Lane changes are recurrently target of decision-making implementations, given its complexity in a highway scenario. Having as an objective the evaluation of a lane change desirability, (Scheel et al., 2018) adopted a bidirectional recurrent neural network (RNN), which can process information in both directions of time. According to the results, such arrangement allows the algorithm to improve its performance in detection when a lane change is possible in comparison with the same network exploring only the future and also in comparison with the results of an SVM-based risk assessment.

Markov decision processes, and particularly the partially observable variation, can take a prohibitive amount of time and computational effort to be solve, without even pointing out the necessity of variable discretization. One of the applications of deep learning has been the policy representation of such problems using a neural network, facilitating the application of reinforcement learning methods. (Qiao et al., 2018) compared the results of policies for crossing intersections from three different methods: one based on time-to-collision, another which implements calculates a POMDP via a long short-term memory (LSTM) neural network and a MDP model with hierarchical decisions. This structure divides the possible policies in different categories restricted only to certain states, and in the experiments conducted produces the safest policy, in comparison with the other two implementations.

Every learning method so far dealt with the decision-making component of the automated driving, always having the control as a separate piece of the pipeline, consisting on the information flows named sequential or integrated, in figure 2.1. The first example of end-to-end driving here comes from (Jaritz et al., 2018), with a deep reinforcement learning algorithm for race driving. Using a realistic rally game to extract the training data and a state decoder composed by convolutional neural networks (CNN) plus LSTM with a asynchronous actor critic (A3C) framework, which allows the network escape from local minima at training, the end result is a policy capable of driving the AV, although still with some crashes. The calculated policy also shown some generalization with tested in unseen tracks.

One of the main problems with black-box type learning methods is the lack of assurances about the existence of unsafe outputs, given the definition of safe for a specific context. To avoid this inconvenient (Mirchevska et al., 2018) coupled a reinforcement learning formulation with a safety verification of the next state given the action to be executed, allowing the end policy to guide the AV through a highway scenario, executing a target velocity and navigating through multiple vehicles, which sometimes demand lane change operations. The state used in the model corresponds to only thirteen variables, the AV's velocity itself and the relative velocities and distance towards other six vehicles considered in the policy calculation. Such limited set of features also favors a faster convergence rate than larger implementations.

Targeting pedestrian collision-free automated driving, (Pusse and Klusch, 2019) used an hybrid POMDP resolution method, combining the deep reinforcement learning and the approximated POMDP planning. The approximated belief tree built during the POMDP planning operation has its construction guided by the neural network, for each leaf expansion the new belief distribution is given by the network's output. As training data the collision history recorded in project GIDAS (German in-depth road accident study) are used, resulting in a policy capable of avoiding such collisions while remaining competitive in terms of performance and smoothness of driving.

Reinforcement learning does not restrain itself only to the use of deep learning methods to calculate the policies. In an application of inverse reinforcement learning to obtain driving styles from human drivers, (Kuderer et al., 2015) tried to determine parameters from real data

to improve the comfort during the automated driving. Features like longitudinal and normal acceleration, jerks, following distance and curvature were used to learn the closest policy that results in a behavior compatible to the one demonstrated by the driver.

2.2 Ethical decision-making

In more common applications an instance of decision making tries to emulate reasoning based on objective quantifiable concepts that are connected by logical relationships. Every bit of care is taken for ASs in general to adhere to strict limitations on performance, comfort, safety and so forth. However, as the number of applications where robots are employed without supervision increases, they get closer and closer to situations in which not only economical and scientific factors are important, but also certain moral features³ the consequences of an automated system's actions should be another parameter in its decision-making (Arkin, Ulam, and Wagner, 2012; Allen et al., 2000). The automated vehicle is an emblematic case, since controlling a one thousand to two thousand kilogram vehicle at 40km/h with unprotected humans around clearly demands an evaluation of the risk that actions might pose to multiple parties. Many other such morally salient cases exist and will continue to crop up in the future, such as the helper robot which will interact with the elderly, or more topically, the level of independence of "autonomous" weapons. As the point of depart of the discussion about artificial moral agents, the three main domains of normative ethics will be discussed to then detail some of the implementations of artificial moral agents proposed throughout the years.

2.2.1 Normative Ethics

Consequentialism

Consequentialist ethics can be defined by the idea that an action is morally acceptable, desirable or necessary in virtue of the results it produces (Gips, 1995; Alexander and Moore, 2020), rather than the nature of the action (or agent) themselves. Of course, before any type of evaluation can occur, what is of value needs to be defined. It is generally agreed upon that "the Good" is the value that must be targeted, i.e. an action is more morally justifiable if it increases "the Good". The definition of goodness varies according to the flavor of consequentialism considered. Taking the example of utilitarianism, the 'Good' which ought to be considered is underpinned by a monistic account of value, which implies that utilitarians hold that only one factor in a decision has moral salience, namely happiness or pleasure (Brink, 2018). Other consequentialist theories, on the other hand, may be underpinned by a pluralistic account of value, implying that multiple values ought to be seen as morally relevant in decision making: John Rawls, for instance, believed that the desirability of an action should also be connected with questions of justice, such as who receives this "Good", and how this effects inequalities in society.

Two main currents of thought from this domain of normative ethics will be discussed later on: utilitarianism, where the Good should simply be maximized; and egalitarianism, which defends that an equal distribution of the Good is morally preferable to its simple maximisation.

³The terms ethics and morals will be used interchangeably.

Two main critiques may be levelled against consequentialist theories. The first is that many consequentialist theories, by for instance focusing exclusively on the maximization of happiness, do not capture enough about what is seen to matter morally. Other features that many people might find important in moral decision making, such as the motivations of the agent, the circumstances of the action, or any pertinent rights or entitlements are often disregarded. There is no list of a priori forbidden actions, everything boils down to the consequences produced by the action, however apparently abhorrent it may be. In the AV context, this would represent the act of targeting an older person to save two younger ones for example, considering only the general maximisation of utility. From this conclusion came the idea of relaxing the maximization requirement, allowing the consideration that only a certain amount of utility may be sufficient. To this end, the second critique addresses the practical feasibility of these more monistic types of consequentialist theories, which compel an agent to consistently and categorically maximize utility, leaving little room for individual goals or preferences, or any weakness of the will on the part of the agent.

Deontology

This domain of ethics defends that an action should be considered good or evil according to its nature, denying the justification of an action based on its results and imposing a rigid and immutable constraint that cannot be changed, a norm, which is the case of the categorical imperative from Kant. For an action to be good, it needs to obey a categorical imperative, i.e. the rightness of an action is assessed by its adherence to a universal maxim or law which all rational agents autonomously accept (Allen et al., 2000). One clear consequence of this approach aligns with the second critique of consequentialist theories, namely, that the uncompromising recommendations that deontology may yield are too much for many human beings to live up to. Some other lines of thought consider more flexible interpretation of the pre-defined norms, allowing exceptions to be made according to the situation.

One of these interpretations, proposed by W.D. Ross, consider the norms to act morally as *prima facie* duties. Such an interpretation is useful in the case of a conflict between the duties, were it is argued that a stronger duty can supersede another. It also offers a list of basic duties that a person should observe and a priority list, that creates an hierarchical order between them in case of conflict. Asimov's three laws of robotics is also an example of normative framework based on the essential premise of deontological ethics.

Another class of interpretations about how to use norms to define if an action is morally acceptable is the contractarian deontological theories. An act would be immoral if it violates a framework of norms defined by a social contract and accepted by society. The origins of Rawls' theory of justice resides in deontology, since the two imperatives which ground his theory– that inequalities are not acceptable and if they exist, they should benefit the worse off–are basically norms. Later on, a version of these two laws will be adapted to the AV context.

One of the main critics of this type of moral reasoning is that sets of norms resemble more heuristic attempts to capture a deliberation process that would be better represented with an utilitarian approach (Gips, 1995). Another is the lack of consideration of information uncertainty and uncertainty in actions' consequences into the definition of the normative framework, although some recent works address this question (Alexander and Moore, 2020).

Virtue Ethics

Virtue ethics predates both previous ethical schools and departs from a focus on both actions and consequences. Its origins can be placed in ancient Greece, when ethical behaviour was understood in terms of an agent's virtues and dispositions, rather than an action's consequences or its rightness according to some normative framework. Thus, if an agent has a virtuous disposition—acting with wisdom, courage, temperance and justice (the four cardinal values proposed by Plato)—then right action would be a consequence of his reasoning and deliberation process.

One can already understand that such theory is based on highly symbolic values held by persons, the concept of a virtue, which can be defined as human excellence embodied in a list of character traits (Hursthouse and Pettigrove, 2016). Naturally, the substantive definition of these virtuous traits is the point of contention for this approach. Some, as Aristotle, argue that virtues are divided into intellectual virtues and moral virtues, with the first type has the possibility to be taught, while the second set must be learned practically through experience. This learning in practice is one of the main tenets of virtue ethics, allowing the agent to discern what to do to accomplish its intention through an action in any situation, and thus developing indirectly the virtue connected to this action.

2.2.2 Artificial moral agents

Artificial moral agents (AMA) is a qualifier that represents every application that tries to implement some sort of ethical reasoning. This goes from simple applications, like the ones proposed in (McLaren, 2003) and (Anderson, Anderson, and Armen, 2005) that are basically programs have ethical recommendations as an output, to (Thornton et al., 2017) and (de Moura et al., 2020) who propose ethical components close to real robotic applications. It should also be pointed out that in most cases, the aim of these approaches is to approximate human moral reasoning, able to consider a universal set of problems, using some form of generic formalism to translate the world into comprehensible data.

Another choice is to focus on specific applications, which is the case of the methods proposed in the next chapters, where the entire ethical deliberation process is designed to solve problems having a specific format and constant properties. Therefore, the question is not whether the agent is morally good or bad, since the approach itself does not aim to simulate moral agency.

Instead, what is important is that the artificial moral agent is capable of making ethical choices which align or coincide with what a human moral agent might decide in a similar context, using the same tools and available information. This argumentation is what motivates the use of the word automated instead of autonomous. Both cases, AMA for generic and specific cases, would be classified as explicit ethical agents using the classification proposed by (Moor, 2006), given that the ethical deliberation are coded into their systems, for more generic that they may be.

The role that uncertainty in information acquisition and processing has within any ethical deliberation (Anderson and Anderson, 2007) must also not be forgotten, because it can directly impact the end decision or invalidate the entire reasoning method. This is one of the main difficulties of the implementation of AMAs, since uncertainty is always present in measurement and most of the time it is not easy to estimate. Most of the works in the domain assume that the information received from other parts of the system is exact, avoiding this concern.

The three domains of normative ethics are viable options to be implemented into an AMA. Deontological and consequentialist approaches are ever present (Gips, 1995; Allen et al., 2000) with the Kantian categorical imperative and the utilitarian maximization principle respectively most of the time. Virtue based methods are less popular, and considering that an AMA does not have agency, an ethical reasoning method based on this approach could be proposed as having some sort of variables that represent virtues that might evolve through time, according to the success, failure and observed consequences of its actions. Another way to implement a virtue-based approach is to use the role morality concept (as is done in (Thornton et al., 2017)). According to (Gips, 1995), virtue-based systems are usually translated as deontological approaches for an application. The other two are the focus of much of the published works on the area, as we will explain shortly.

AMAs can be employed in a wide range of applications, from counseling a human's ethical decision to deciding whether a lethal action from a drone is ethically permissible. In the automated vehicle domain the ethical component is most of the time connected to the safety claim of the road users, meaning that there is always a risk of death involved. Therefore the domain is posed to be the first one that presents to laypeople an application that can present ethical behavior in some situations, given that it was already established that braking and relinquishing control from the AV to the driver is not always the best option (Lin, 2016).

Related works

Since the advent of autonomous systems and our expectations for their deployment in the real world, certain ethical concerns related to these systems have become well known. In (McLaren, 2006), two implementations of the casuistry approach, where a real problem is compared to an instance of a similar case so as to guide the decision-making, are reported one. The first compares ethical dilemmas questioning if the truth or a lie should be told. Adopting the point of view from the person that ponders what to do in a dilemma, the algorithm points out the

motivations as to why the truth should be told that are valid for the current situation and the base situation (the one loaded *a priori* in the machine to serve as a comparison point), that are more strongly connected to one case or that it is only valid in one instance.

The second program, proposed in (McLaren, 2003), operates as an ethical board review would, comparing past instances of deliberated cases and ethical codes. Then, given a new case to be analyzed, the system would give a list of possible pertinent codes, a list of pertinent cases and additional suggestions concerning the relationships that may exist between the cited ethical codes. Its deliberation procedure involves a two-step graph-mapping algorithm, where in the first each of the cases in the database are given a score according to the similarity to the one being evaluated, which is used with a A^* algorithm to establish the most similar cases.

Taking a similar approach, using recorded cases of ethical decisions as the baseline to the resolution of new ones, (Anderson, Anderson, and Armen, 2005) implements the principles of Biomedical Ethics (that are *prima facie* duties) into a program, which extract principles from some cases and use these to solve new cases. The implementation is divided in three components: a knowledge-based interface, that manages the selection of principles to be applied, a advisor module, that make the decision and a learning module, that abstract principles from particular cases. This last one uses inductive logic programming to generate the decision rules that are representative from the problems given as training and that might passed undetected given that an reflection of multiple cases is necessary to detect them.

Ethical constraints for autonomous weapons were addressed by (Arkin, Ulam, and Duncan, 2009) through the definition of a framework that contains a ethical governor, proposed in (Arkin, Ulam, and Duncan, 2009) and an ethical adaptor, proposed in (Arkin and Ulam, 2009). The first module restrains the lethal actions that a robot can take given constraints defined by the Laws of War and the rules of engagement, which are grouped in obligations and forbidden constraints for the current mission. Such constraints are applied to the robotic response calculated by the system, and if it is not a permissible action another one is selected and then executed. Two components define the action selection: one deals with evidential reasoning, extracting from perception, *a priori* information, target data and current behavior the logical assertions that will confront the current constraints; and the constraint applicator, that process such logical assertions, composed by the constraint interpreter, that translated the general language used to restrictions that should be applied in the current situations and the damage estimator, that estimates the consequences of an action in an utilitarian fashion, accounting for structural damage and loss of life. However it is assumed that accurate targeting with precise uncertainty correction is employed, which may be too optimistic to consider.

The ethical adaptor acts in the level of the constraints used to evaluate the chosen behavior, and more generally tries to modify the robot's behavior so as to insert an emotion-like deliberation during the behavior selection. It functions both as a reviewer of the robot's actions after the fact and it measures the effect of a violation of the LOW or ROE using emotion-based functions; both mechanisms have as objective the evolution of the robot's procedures so as to

prevent another violation of its constraints. As such, it works only by increasing the intensity of restrictions, never on the other direction. In (Arkin and Ulam, 2009), it is the second function that is discussed. It employs a measure of guilt towards violations of the restrictions, which can accumulate throughout the mission and result in then necessity to disable temporarily or permanently the lethal actions available to the robot until an after-action review is completed. In this context the guilt "feeling" works as a representation of the overall ability of the machine to behave ethically.

(Berreby et al., 2015) proposes a formal language to model a generic ethical dilemma, allowing the artificial agent to reason about an ethical choice during its occurrence using the information at hand. Taking the trolley dilemma as an example, a model of causality is established as an Event Calculus model, which enables the program to reason about responsibility of actions while taking into consideration the actual state of the world and also account prevented situations. This framework then uses the deliberation given by the formalization of the Doctrine of the Double effect (DDE), represented here by the nature-of-the-action, that measures if an action is appropriate from its inherent nature, the means-end and right-intention condition, that avoids negative intentions and instances that use negative actions to achieve positive ends, and the proportionality condition, that demands that positive and negative effects be proportional.

On the same line, trying to give the machine the capacity to deliberate similarly as a person might do in ethical dilemma situations, (Bonnemains et al., 2018) propose formal tools to describe a situation and some models of ethical principles that can be used with this descriptions to achieve an ethically motivated decision. Three ethical frameworks are proposed, based on utilitarian and deontological ethics and another inspired by the Doctrine of Double effect. Given a particular situation, it is described as logical facts which are taken into account to deliberate on a decision by a respective profile; the utilitarian approach chooses the decision that present the greater good having also the lesser evil; the deontological approach judges the decision by its nature; the DDE is implemented with the same formalization than the proposed in (Berreby et al., 2015). The same assumption made in (Arkin, Ulam, and Wagner, 2012) and in (Berreby et al., 2015) is present here: all decisions and facts are considered to be certain, hence no uncertainty is taken into account inside the deliberation process.

Using a new approach to decide under ethical dilemma, (Leben, 2017) proposes a criteria based on Rawls' Maximin principle as an alternative to an utilitarian approach or the usage of the Doctrine of the Double effect, outlining an algorithm to implement such a principle while using a "veil of ignorance" to conceal information that could create a bias. The procedure applied in a dilemma, using an automated vehicle in a inevitable collision as an example, is to put every involved road user in an original bargaining position of fairness to find the final decision. The biggest difference from the utilitarian approach is that the proposed decision procedure operates trying to increase the worst possible option that might happen, while the utilitarian approach always based its decision in maximizing its expected criteria in uncertain situations, without looking to what might happen in the worst possible option.

But one can only go so far with hypothetical situations. Parameters such as dynamic interactions between agents and uncertain action consequences must be taken into account to produce fair, realistic, and acceptable ethical deliberation. Focusing on a situation when an AV must decide between a collision with pedestrians or a physical barrier, (Pickering et al., 2019) used an estimation of the AV's deformation against the barrier and statistical data to define a risk velocity for the pedestrian collision. A bi-linear model of the deformation is tuned to be similar to a finite element analysis (FEA) instance of a full-frontal collision, and from that model the maximum deformation that can be produced so as the injuries involved are minor is obtained. The pedestrian injury severity is then represented by the vehicle's velocity obtained by a statistical analysis of collisions in another study. Then the deliberation becomes clear if one or the other possible decision, collide with the barrier or with the pedestrian results in minor injuries, but the case when it is equally bad to do either is not treated.

Taking a design approach to infuse ethical constraints in the control component of an AV, (Thornton et al., 2017) uses concepts from deontology, consequentialism and virtue ethics to define a Model Predictive Control (MPC) that respect ethical constraints by design. Such MPC is proposed in a straight road, with shoulder and an obstacle ahead of the vehicle. Every choice that may be taken in this scenario have an ethical component: if the vehicle uses the other lane while there is a double solid yellow line it will break the traffic code; if it uses the shoulder lane then it needs to be unoccupied and can only be used for brief periods of time; and if the stop behavior is chosen then the mobility property of the vehicle is affected. Firstly, the path tracking cost is dependent to the necessity to deviate from the reference trajectory while the steering angle have two constraints from different philosophic origins: a limit on the steering angle limits (in a deontological sense) and a cost function connected to its rate of change (in a consequentialist sense), a process to avoid the obstacle with absolute certainty and the adherence to the traffic code, which is treated as a soft constraint that can be set with a high weight (thus approach a deontological behavior) or with a low weight (approaching a consequentialist behavior). Two different experiences are made, one with different weight values from each of the reward parameters and another simulating a form of role morality connected to an emergency vehicle goal.

2.3 Behavior prediction

An essential role of the AV during execution is the capacity to predict other road users' behaviors, given the environment that they are in. Multiple information sources can be used to predict what road users intent to do next, from the road structure itself to wireless communications and traffic rules, without forgetting the road users' dynamics itself. The most immediate use of these methods are in advanced driving assistance systems (ADAS) to detect possible dangers to the driver and passengers, but without the need to deliberate on an action based on this information.

Different types of road users are present and interact between themselves in urban environments. A useful classification of this environment is the definition of vulnerable road users (VRU), which assemble pedestrians, cyclists and two-wheeled drivers into one single category that distinguishes itself for the lack of protective "shell" (WHO, 2018) surrounding the user. Every one of these groups is the focus of multiple studies to predict the behaviors of each specific road users type, but the two most studied groups are pedestrians and car-like vehicles, given that 66% of deaths in EU roads in 2018 were pedestrians or vehicle passengers (Commission, 2018). Pedestrians and car-like vehicle passengers are the only two types of road users considered in this thesis.

According to (Rudenko et al., 2020), motion prediction procedures contain three main elements:

- Stimuli: represents the agent's motivation to execute the trajectory observed. It can be divided in two parameters, the agent motion intent and other influences, exerted by the environment itself (and in this case the road infrastructure) or by other road users.
- Modeling approach: encodes the agent's reasoning; given a stimuli composed of an intent and the environment influence, it produced as output what the agent will do in the future to accomplish its intent.
- Prediction: the format of the output from the previous element; it can be the a trajectory itself to a probability distribution given some possible maneuvers for example.

The reasoning model is the main component of the behavior prediction. Three different approaches may be taken to reason about an agent behavior: focus on a physical description of the agent to predict on a short-term basis, try to learn these behaviors from data and reproduce it generically or reason about each agent motivations to then find the correct behavior. In these three options the term behavior represents a different quantity being estimated. Figure 2.2 shows which is the meaning of behavior in a short Δt , a mid-range Δt and a long-term Δt prediction⁴.

⁴Roughly speaking, one can consider a short Δt as less than 1 second, and a long-term Δt as more than 4 seconds

This structure can be identified in the survey done in (Rudenko et al., 2020), where physics-based methods predict short-term movements, pattern-based methods try to learn features and cues from past agents' trajectories and planning-based methods that reason about the agent's motivation.

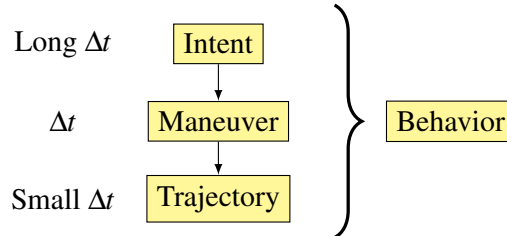


Figure 2.2 – Hierarchy of possible significations for behavior

A different classification is proposed by (Lefèvre, Vasquez, et al., 2014), where three different types of behavior prediction are also proposed, with the first pertaining to the physical prediction, short-term models that takes into account the mechanical properties of the agent. The difference between both classifications comes from the other two categories, maneuver based prediction and interaction based prediction, which represents the capacity to observe a specific maneuver being executed, already accounting the possible intention determination that lead the agent to choose a maneuver and, lastly, changes in behavior during interactions between road users. What becomes clear from this classification is that the three levels as defined are not disjoint, they can complement each other, going from prediction close to the seconds scale to accounting interactions between agents, as to plan and execute a mission.

In comparison between the classification proposed by (Rudenko et al., 2020) the learning method does not need to restrain itself only to a maneuver abstraction level, it can determine the intent of a road user using the data available, that usually is also composed by interactions with other road users. Clearly, the proposed organization it is not completely rigid in its definition. But, since it is more pragmatic, it considers the implementation method for behavior prediction, thus interactions between agents are considered at every level. To understand the body of work in behavior prediction (which is also present in some decision-making methods too, since the former frequency is a part of the latter) it is important to consider the type of method being used to implement the prediction model and what is really being observed as a behavior.

2.3.1 Vehicle prediction

Methods that use the physical prediction approach usually search the minimization of measurement errors while predicting the position of the vehicle in the near future. Aligned with the ability to predict the behavior in short-term periods of time, one the main use cases of such approach is to estimate risk of collision given a specific situation. Considering a cooperative group of vehicles in a road with the possible presence of cyclists and pedestrians, (Batz et al.,

2009) uses unscented Kalman filters (UKFs) to predict trajectories during a specified Δt and thus monitor the distance between pairs of agents, taking into account the physical dimensions of each road user. The distance is calculated from a set of vectors defined by the Minkowski sum between the uncertainty ellipsis, calculated from the covariance estimated by the UKF, the possible rotations of the rectangle that represents the body of the road user, which also considers a range of values extracted from the covariance matrix, and the mean position. Inside a group of vehicles, one of them is designated as leader, which will execute the probabilistic prediction, check if all vehicles are at a minimal distance from each other, and if there are not then signal that the situation is dangerous.

The trigger time of AEB systems is a critical parameter that might prevent a collision or even reduce its severity. It is in the short term that the real need of activation needs to be checked, which is proposed in (Kaempchen et al., 2009) to be determined considering the trajectories of all road users involved. Only if all possible trajectories evaluated result in a collision that the vehicle must trigger the braking (reason why only the maximal acceleration is considered to determine the trajectory). One sensible point is the determination of which trajectories should be checked, which is done by sampling the Kamm's circle⁵ to obtain the different values of longitudinal and cross accelerations that might be applied in a short period of time. Given the infinite number of possible trajectories, a search through the circle is done using the approximate collision intersection area between the involved parties. When no zero area is detected after a gradient descent search, the AEB must be triggered.

In (Brännström et al., 2010) a linear bicycle model is used to model vehicles in near accident scenarios and assess the risk of collisions. These models are then tested to verify if there are possible collisions in the future, and if there is, to determine and execute different policies according to the driver's preferences and the other vehicle's actions. Observations that, for example, drivers tend to use constant steering angle velocity followed by a constant steering angle (Godthelp, 1986) during normal driving or in evasive situations, specially in the moose test⁶, that the steering velocity is correlated with the posterior steering angle (Breuer, 1998) were used to model possible maneuvers to be tested. They are a combination of swerving and acceleration or braking, always determined to avoid contact with other road users, given that each one of them are considered as rectangles. The system is triggered if no possible safe maneuver that respects the driver's preferences (orientation position and its rate of change and acceleration value and its rate of change) and proceeds to a two-step verification: considering the other road user as deterministic, it finds the maneuvers that the ego-vehicle should execute, and also does the opposite for the other vehicle. As such, it can improve the timing of AEB activation as assess how the driver can avoid collisions.

Following the same line of (Brännström et al., 2010), the uncertainties connected to road geometry detection and measurement are accounted in the sampling of possible trajectories

⁵Encompass all possible direction of the forces between the interaction of the tires and the road.

⁶The moose test is a high-speed avoidance maneuver test that tries to emulate a vehicle avoiding an obstacle ahead maintaining its velocity constant during a lane change and going back afterwards

by (Petrich et al., 2013). Each lane center-line is approximated by a cubic Hermite curve, and the mean value of the vehicle's state (estimated by an EKF) is projected into this center-line, defining a way-point. Since there is also uncertainty in the lane's determination, they are also considered to be distributed normally, thus making the assignment of a dynamic object to a lane by an χ^2 -test possible. Each lane, to be considered as a viable hypothesis, need to score higher than a determined threshold in the test. Then all possible trajectories are predicted by the EKF, considering constant acceleration and that the vehicle always follows the orientation of the center-lane. These hypothesis over time are evaluated by a multivariate cumulative sum algorithm, used to detect small and moderate mean shifts in probabilistic distributions, which is adapted to consider the vehicle's possible lane change.

As it can be seen, for the physical methods approach the most used implementations to model the vehicle are constant models, for example constant velocity (CV), constant acceleration (CA) or constant turn model (CT) with a bicycle model to represent the dynamics. But more complexity can be necessary in some contexts, even more in dangerous situations that tend to approach the vehicle to high lateral accelerations (Polack et al., 2017). Kalman filters are the go to method to account for uncertainties, but there are implementations that use Monte Carlo methods or dynamic Bayesian network (DBN) to relax the Gaussian incertitude assumption (Rudenko et al., 2020). Other methods that can be applied are potential fields derived algorithms and interaction multiple model (IMM) implementations.

Using maneuver recognition, with pre-determined maneuvers, and a constant yaw rate and acceleration (CYRA), (Houenou et al., 2013) achieves an improved prediction accuracy in comparison if only the physical model was used to predict the trajectory. Three types of maneuvers, keep lane, change lane and turn at intersection, are used to generate different trajectories using the current state and the road's parameters. A Malahanobis distance between the current position and the center-lane is used to define the correct lane and, by consequence, the maneuver being executed. The selection of the most appropriate one is done by a cost function that accounts for the comfort during the execution of the trajectory (which means no high normal accelerations) and a time parameter to punish longer maneuvers. The physical prediction is done using the CYRA model, as if the vehicle was a determinist agent. This physical prediction and one of the sampled trajectories are fused using a pre-defined weight function.

Also considering maneuvers, but this time taking an offline learning approach to learn following, overtaking and flanking maneuvers, (Firl and Tran, 2011) uses a Hiden Markov Model (HMM) to represent each one of them. The learning method chosen is an expectation-maximization-type (EM) algorithm, with the online recognition being done by calculating the value given by $\text{argmax}_i P(\lambda_i|O) \propto P(O|\lambda_i) \cdot P(\lambda_i)$, with λ_i being the HMM model and O the observation vector. The likelihood $P(O|\lambda_i)$ is modeled as a mixture of Gaussians (MoG), with is obtained, together with the prior, in training. Still, roads with different spatial disposition and interaction between more than one vehicle are not considered.

Further improvements came in (Tran and Firl, 2013) where a bi-dimensional Gaussian

process was used to model traffic situations at intersections and an UKF to estimate the trajectory. The parameters of the Gaussian process are obtained by maximizing the log marginal likelihood given some training data, which classify the behavior of other vehicles as turn left, turn right or go straight. With the most probable maneuver identified, the correct Gaussian process is used in the update step of the UKF to correct the prediction. In another implementation, (Tran and Firl, 2014), the same structure is used, but this time the algorithm combines 3-dimensional Gaussian processes with a particle filter, instead of the previous UKF. Another dimension was added to the processes to allow training the model with data that originated from different intersections and similarly, the process that correctly represents the situation is used to calculate the weight of each particle in the filter.

The road structure configuration exert an important influence into the behavior of all road users. Together with distances to surrounding vehicles and road characteristics, (Gindele et al., 2010) employs a dynamic Bayesian network (DBN) to model the relationship between position, context, behavior and trajectory. Using a bicycle model without slip to model vehicles and six possible maneuvers (free ride, following, acceleration, sheer out, overtake and sheer in), it predicts the most probable trajectory, which is approximated by a cubic Bézier curve. The resulting filter is solved using a particle filter approach to calculate the update equation. The model is further improved in (Gindele et al., 2013) adding a learning step to determine the correct policy model for other road users given a specific context.

Up until now probabilistic methods composed the entirety of algorithms discussed, but in pattern approach and also later, in planning-based methods, two other type of methods will start to appear. The use of logical reasoning is one of them. in (Hummel et al., 2008), it description logic is used to describe all the possible different forms and configurations of intersections that may exist in a city. A case-based approach is taken by (Vacek et al., 2007) to interpret situations and deliberate on a maneuver to be executed. Each case is composed by a description of a situation that might occur and the correct behavior that needs to be executed. During the execution new knowledge can be added, possibly modifying a case solution.

Deep learning is the other technique that is increasingly popular in behavior prediction. Long short-term memory (LSTM) is used as a recurrent neural network (RNN) in (Cho et al., 2019) in the implementation of the encoder, interaction and prediction components. The first models patterns of trajectory of other vehicles, the second measures both multi-modality (how many situation can occur) and robustness slackness related to a rule (exclude situations that cannot occur) while the third effectively predicts the trajectory and the robustness of a rule. Maneuvers are considered as a logical relation, formalized by a signal temporal logic (STL) approach, which has a the robustness slackness measure defined to create an order of priority in a set of rules. The last component of the algorithm is a controller, implemented by an MPC.

Another work that uses deep learning is (Messaoud et al., 2021), which considers the environment surrounding the ego-vehicle as a grid, to encode drivable areas and all vehicle present into the input vector to the encoder layer. Then a middle layer, called attention layer, decides

which vehicles must be prioritized when the trajectory is being defined. The output of this layer is fed to the decoder layer, that executes the trajectory prediction, formed by two networks: the LSTM decoder and two fully connected layers separated by a leaky rectified linear unit (ReLU). Each output of this last network represents the probability of the predicted trajectories.

Pattern approaches can use pre-defined maneuvers or offline learning methods but with a defined set of maneuvers. Popular methods are GPs, HMMs, DBNs, neural networks (Rudenko et al., 2020). But there are also methods that use learning to model and detect the intention of other road users. Intersections are one of the most dangerous road environments, given the number of accidents that are observed in them. The work in (Lefèvre, Laugier, et al., 2012) proposed to use the difference between what is expected from road users from what they intent to do as a measure of risk. A DBN model is used to estimate the vehicles' motion while the behavior intention and expectation is estimated jointly. Both the context at the moment of estimation and the interaction between road users, through the jointly estimation, are considered to measure the risk. An RNN with LSTMs is used in (Zyner et al., 2018) to predict the intent of other road users in non-signalized intersections. The proposed method is applied in tangential roundabouts⁷, with lidar data as input.

In an implementation of a system capable to predict vehicle behavior in generic environments, (Bonnin et al., 2012) proposes the use of an hierarchy of cases, where each node represents a specific behavior that might be detected in different locations. Inside each node there is multiple models, itself defined by a set of features and classifiers. A node is activated based on GPS readings and road infrastructure information collected from an *a priori* map, and it competes with its children nodes to find the one that better fits the situation.

Uniting physical, pattern and planning based approaches into one single framework, (Lefkopoulos et al., 2021) implemented a predictor that uses an interacting motion model Kalman filter (IMM-KF) with 6 different longitudinal and lateral behaviors for each vehicle, which passes through an selection based on the detection of collision during prediction, given the size of each vehicle. Then an hierarchical approach allows the algorithm to define an order of priority, taken into account the Highway Code, decoupling the multiple prediction of N vehicles into N single prediction operations.

Focusing on a long horizon prediction of local vehicles in highways, (Xin et al., 2018) used two LSTM networks, one that concentrates into determining the driver's intention, and a second dedicated to predict the future trajectory. All the data used in classification and prediction is obtained by the ego vehicle.

⁷Tangential roundabouts are characterized by higher speeds given that the entry curve is tangential to the center island; this configuration differs from the radial approach, adopted mostly in Europe, that ensures the entry into the roundabout through a deliberate turn maneuver to reduce speed.

2.3.2 Pedestrian prediction

The pedestrian's behavior estimation methods can also be divided into the physical, pattern and planning approaches, with probabilistic and data-driven philosophies being the most popular implementations. Since, the trajectory prediction in the algorithms proposed in chapters 3, 4 and 5 will be made considering a constant velocity model, only a short state of the art will be commented here, nonetheless containing the most used approaches in the domain. A more detailed survey can be found in (Ridel et al., 2018) and for deep learning based approaches, (Ahmed et al., 2019).

Considering the context of an automated vehicle, it is important to determine the behavior of an pedestrian not only in correspondence with interactions with vehicles but also taking into account the interaction between pedestrians. Using bird-eye images to as support to predict the pedestrian behavior, (Coscia et al., 2018) represents the agent by its position and velocity. This last one is the focus of a prediction dependent on four parameters: semantics of the position, observation from training data, constant velocity inertia and a destination probability. The semantics of a position is supported by measures of desirability of the pixel, according to the number of pedestrian trajectories that passed close to it and resistivity, which represent the proximity of obstacles.

In contract to social forces prediction methods, which are usually hand crafted, (Alahi et al., 2016) uses a LSTM network to model not only the constant and predictable movement that originates from attraction and repulsion approaches, but also disruptive behaviors, as for example the response of a pedestrian when it perceives a crowded environment. To account the interaction between pedestrians the hidden states from an LSTM are shared with neighboring LSTMs.

Taking a probabilistic approach to predict long-term behavior of pedestrians and their intent was done in (Karasev et al., 2016) by modeling the latter as a function of the goal, considered hidden (but with a goal space known) and the state. This unknown goal is represented by the posterior probability of goal and state given past observations, and it is updated using a Rao-Blackwellized particle filter. Then the intent function, which is considered as a sample from an MDP policy, is estimated not by the optimal policy, but by an approximation, since to use the optimal solution would be to consider the pedestrian completely rational limiting too much the choices of possible trajectory. In the policy it is considered too traffic signals states, enabling the model to predict a stop from the pedestrian due to a red light.

As it can be seen, the methods for predicting behavior for vehicles and pedestrians are almost the same, with the exception that for pedestrians there is another source of information available: the body language, and more importantly for intention estimation, the head orientation. The 3D body language of the pedestrian is modeled in (Quintero et al., 2014) using a Gaussian process dynamic model (GPDM), one for each hypothetical behavior, for example starting to walk, stopping and walking combined with left and right. A feature vector for each pedestrian is formed by the 3D coordinates of specific body joints and their velocities, which has the dimension

reduced by the GPDM and used to train the correct model. A naive Bayes classifier is used to classify an observed pedestrian according to the correct model and a maximum *a posteriori* (MAP) decision rule determines the correct hypothesis.

2.4 Game theory

Game Theory assembles analytical tools that may be used to model a situation when an agent's best action depends on its expectations from the actions executed by other agents, and vice-versa (Ross, 2001). Its first foundations, in the economics context, were established in (Von Neumann and Morgenstern, 1966), with the following definition: *They⁸ have their origin in the attempts to find an exact description of the endeavor of the individual to obtain a maximum of utility, or, in the case of the entrepreneur, a maximum of profit.* In this definition three main entities can be identified:

- Individual: it represents the agent, the decision-maker that interacts with others.
- Utility: quantify the agent's preference towards some object or event, expressing these preferences into all possible choices of action.
- "Maximum of": what the individual wants from the interaction; corresponds to some deliberation method to determine the action with the most utility.

The concept of utility came about as a quantity that represented subjectively some form of fulfillment for the agent, thus its main objective is to maximize it. The relationship between choices and utility that emanates from them is represented by the utility function. It allows the mathematical formalization for the maximization operation that is executed by every agent. Two forms of utility function exists, an ordinal⁹ form, which express a preference order without a particular meaning for the magnitude of these utilities, and a cardinal one, where magnitude matters (Ross, 2001), such that the utility measures directly the desired property. One example of a cardinal utility functions is when such function represents a currency.

The definition of a game is the realization of some situation when an agent can only maximize some utility if it anticipates the behavior of other agents. Such abstractions allows different situations to be represented and analyzed by the same set of tools. Examples of different backgrounds that may use a game formulation are the economical domain, to study social interactions and even in robotic implementations. As for the anticipation towards other agents decision, it is based on the rationality model assumed to explain the behavior of the other agents. In normal life, rational behavior means that an agent look for the best mean to achieve a goal (Harsanyi, 1976).

Such definition of rationality is close but not exactly the same as the classical one, that consider not only a set of means but also multiple ends. It is assumed as constraints that the decision-maker have perfect access to all relevant knowledge concerning its environment,

⁸Problems arising from economic behavior

⁹When the type of utility function is not mentioned, it should be considered as an ordinal function.

cognitive capacity to deliberate on an action and clear preferences about its action consequences (Osborne and Rubinstein, 1994; Simon, 1955). Equation 2.1 formulates the usual optimization executed by the classic rationality assumption under certainty, where \mathbb{A} is the set of available actions, $g(\cdot)$ is the action-consequence function and $U(\cdot)$ is the utility function.

$$a^* = \underset{a \in \mathbb{A}}{\operatorname{argmax}} U(g(a)) \quad (2.1)$$

However, the classical model of rationality have clear downfalls and approximations. This subject is pointed out in (Simon, 1955), where a number of adjustments are discussed. For example, in real situations the perfect information hypothesis is almost never realistic, each agent only has access to some part of the total information necessary to define the entire situation. Other two problems pointed out are the limited cognitive capacity of each agent, which may impede the optimization of the utility if it needs a complex quantification to be determined and an inability of the classical rationality to model situations when utility preferences are not clearly comparable (Simon, 1955). Another practical problem is the limited period of time available to deliberate on an action. Even though these limitations are well known, the classical rational model is still widely adopted but sometimes some *a priori* constraints are relaxed, notably the perfect access to information.

The properties below are useful to characterize different types of games and to establish the conditions in which a game unfolds (Osborne and Rubinstein, 1994).

- Interaction between agents:
 - Non-cooperative: games with no communication or enforceable agreements (Nash, 1951), where the individuals decide on their own volition (Shoham and Leyton-Brown, 2008).
 - Cooperative: games with communications and/or enforceable agreements, where reasoning as a collective is possible.
- Decision time:
 - Strategic game: each player makes only one decision, at the beginning of the game; all players decide at the same time.
 - Extensive game: each player has the liberty to make a decision and revisit it whenever it wants.
- Information:
 - Perfect information: All agents have access to all information about the other players.
 - Imperfect information: Some information might not be accessible for the agents.

Depending on the type of game an analytical solution might exist. Non-cooperative games are the most studied type of game, since it involves competition between individuals and because cooperative games can be remodeled into non-cooperative bargaining games (Harsanyi and Selten, 1992). In this context, a game that considers the classical model of rationality is guaranteed to have an equilibrium point (Nash, 1951).

One of the most important interaction scenarios to be considered is when the decision makers do not have knowledge about preferences and motivations of the other with which it interacts. This incomplete game can be transformed into a complete game according to (Harsanyi, 1967) if a joint probability function about these unknown information exists, then the incomplete game is equivalence to a game with complete information and any equilibrium point determined in this equivalent game represents also an equilibrium point for the original game (Harsanyi, 1968b). Chapter 5 will use this transformation to estimate the behavior of other road users, given that the AV does not know their motivation.

2.4.1 Related works

As one would expect, all the instances where a game theoretic approach was taken involves interaction with other road users. It can be used in physical, pattern or planning approaches. In a racing scenario, where two vehicles try to overtake one another, (Wang et al., 2019) adds a prediction step into the game theoretic controller that is able to estimate how much the other vehicle will yield to the ego-vehicle. Collision is avoided because both vehicles share the same collision constraint into their controller. One player use the same game-theoretic controller while the other uses an MPC, which allows the demonstration of typical racing behaviors, as blocking and overtaking. This game controller was inspired by the implementation in (Spica et al., 2020) for two drones racing, but with the addition of the bicycle model to describe the dynamics of the vehicles involved.

The previous algorithm depends on reliable information about the other player, hence it is susceptible to measurement errors. To fix such vulnerability (Notomista et al., 2020) proposes to use the algorithm proposed in (Wang et al., 2019) as a high-level controller (notably named Senna), while implementing a low-level planning algorithm (this one named Prost) to deal with possible unreliable information. This second algorithm uses a control barrier function as a constraint to mitigate uncertainties in the opponent's model or in its state estimation.

In (Turnwald et al., 2016) the Nash equilibrium is used to reason about the interactions between agents. The use-case considered is a robot moving inside a crowded environment. Two game models are considered, one static, where all players take decision simultaneously, and one dynamic, where decision comes sequentially. The cost functions¹⁰ is defined as a sum of two

¹⁰Cost function, payoff function and utility function are three names that represent the same entity, the utility function, as explained in the beginning of the section 2.3.

terms, one involving the state of the robot and another accounting for the interaction with other agents (notably if there is a collision predicted or how close a human pass another). Contrary to the last two examples, where the utility function to be optimized was the difference in position of two racing drivers in reference to how ahead one is from the other, the utility function here is based more on ordinal preferences.

One simple situation where interaction between agents is vital to achieve an acceptable behavior is the lane change maneuver. Usually the AV has to "negotiate" with a surrounding vehicle(s) to insert itself into another lane. Vehicle to vehicle communications can help a lot in this negotiation, but even in a far future there will be manual vehicles in highways. Therefore, in (Meng et al., 2016) a method to evaluate if a change of lane can be executed consists basically in a Nash equilibrium calculation, where the actions for the AV are to change or not the lane, and to the other vehicle are to decelerate or accelerate. The payoffs for the AV are the difference between the maximal cruise velocity in two situations: if it continues in the current lane (which is the initial one) and if some safety gap is respected during the lane change operation; otherwise the payoff is defined as a cost. For the other vehicle the payoffs are defined as the inverse of the acceleration (or the absolute value of deceleration), also if the security criteria is respected.

The lane change use-case is also explored in (Ding et al., 2018) but now using a learning approach. The maneuver is considered to be composed by two moments, first when one vehicle adjust its state to allow an other vehicle to merge, and second when the merging maneuver happens. Two possible controllers for the operation are examined; the first one, called asynchronous, controls the vehicles with a controller trained using a deep Q-learning algorithm that acts as a single agent, hence it does not take into account the behavior of other vehicles. The second controller, called synchronous, is implemented using a Markov game¹¹ to account the interaction between both users. The reward function in this case has a cost term connected with the occurrence of a collision, and every Markov game with a stationary policy has at least one Nash equilibrium. It is observed that the first scheme fails constantly, while the second is successful after a reasonable amount of training episodes.

Interaction between the AV and a human driver is usually treated using an open-loop information structure, as was in (Spica et al., 2020) for example. The approach taken in (Fisac et al., 2019) is to consider a fully coupled interaction model, with two levels: a strategic one where interactions are treated integrally and dynamics in a simplified way; and a tactical level, which uses the opposite arrangement, since it operates in short time periods. The result of the strategic level (and the value functions calculated) is used as the best outcome for the tactical level, that takes such information in considerations in its planning, solving a high-fidelity model of each vehicle with a receding horizon approach. A Stackelberg game is used to model the game in the high-level layer, with the AV as the leader.

It is well known that human drivers make decisions based on experience and emotions,

¹¹Also called stochastic game, is a game defined as a Markov system, with the payoff function equal to the value function.

while most of the proposed methods for implementations in AVs are based on procedural analysis of the environment and prediction of the other assuming that their behavior can be approximated by another procedural representation. The work done in (Ji and Levinson, 2020) tries to estimate the behavior difference between the real decision taken during a discretionary lane change situation and one that is taken using a game theoretic approach with payoff function defined by safety and time-saving terms with adjusted parameters using real data collected by the NHTSA. The result of simulations with different initial gaps and initial velocities are that greater initial distances allow the vehicles to pursue more selfish non-cooperative behaviors, while high-velocity allows a fast maneuver but worsens the total payoff since also extends the period in which a dangerous choice can be made by one of the drivers.

2.5 Conclusion

As it was seen in the section about decision-making methods for the automated vehicle, two main approaches dominate the domain: Markov-based application or learning methods. The method proposed here is based on the first approach, with a key simplification to avoid considering the changes in other road users' behavior into the decision process. Without such simplification it would be necessary to consider a complex model to calculate the transition probability, as it is done in (Brechtel et al., 2014) for example, and adopt a sampling-based solving algorithm, a common resolution method in many MDP and POMDP applications (Ulbrich and Maurer, 2013), (Hubmann et al., 2018), (Wei et al., 2011). The proposed decision-making is designed to work in any type of environment, urban or in highways, and it was tested in an urban scenario.

The literature about ethical decision-making is rich on thought experiments but less numerous on real applications. Most works deal with logic rules of some sort, but there are examples using other approaches such as trajectory control (Thornton et al., 2017) or collision severity analysis Pickering et al., 2019. The methods proposed in this thesis to deliberate under ethically sensitive situations are based on consequentialist ethics and are specific to situations where the ethical consequences need to be accounted for, in opposition to Thornton et al., 2017 that includes by design the ethical component of the decision in situations where it is not necessary. Another difference from the literature is that, during the ethical deliberation uncertainties play a main role, differently to the applications proposed by (Bonnemains et al., 2018) and (Berreby et al., 2015) that relies only in deterministic logical relations for example.

Finally, the last part of the thesis deals with the prediction of other road users' behavior and how they should modify its decision accordingly. An incomplete game is then modeled using the behavior probabilities for the other road users, applying the Harsanyi's transformation to enable a resolution via Nash equilibrium. Two main differences from the standard application of game theory in the current literature can be pointed out: it is used to make the AV's decision more robust, given the max-min nature of the Nash equilibrium itself and the influence of the uncertainties into the final deliberation; this last one is rarely seen in game-theoretic applications ((Meng et al., 2016), for example). The approach, as was used here, is similar to (Turnwald et al., 2016) in that he models the interaction between multiple agents in a higher symbolic level, but the proposition will go beyond since it also accounts for uncertainties; although such use is second to the most popular application: to model and exploit agent interaction in low symbolic level, as for example in the control component of agents (Wang et al., 2019), (Notomista et al., 2020).

Each symbolic option of the game is represented by behavior profiles, modeled using CV (constant velocity) or CA (constant acceleration) policies for pedestrians or vehicles. This method represents a typical physical-based approach application to behavior prediction, given that the prediction happens in the dynamic level and that the interval range considered for the implementation is a small one (Rudenko et al., 2020). A similarity analysis is then executed based on the expected behavior given the assumed profile, that generates a prediction at each

iteration using a Kalman filter, and the observation of the environment evolution, typical of pattern-based approaches such as (Houenou et al., 2013). The behavior probabilities calculated with this approach are used by the game model to deliberation on the best strategy.

Chapter 3

MDP for decision making

Contents

3.1 Theoretical background	47
3.1.1 AV's Architecture	48
3.1.2 Markov Decision Process	50
3.2 AV decision-making model	55
3.2.1 State and action sets	55
3.2.2 Transition function	57
3.2.3 Reward function	59
3.3 Results and discussion	66
3.3.1 Value iteration	66
3.3.2 Simulation	68
3.3.3 MDP policy results	70
3.4 Conclusion	74

3.1 Theoretical background

This chapter starts with a discussion on the architecture most commonly used for AVs, considering the reactive and hierarchical paradigms approaches presented in chapter 1, which is followed by the theoretic definition of a Markov Decision Processes (MDP), from its components to the conditions of uniqueness and existence for the policy calculation. Section 3.2 details the specification of the MDP formulation used in de Moura et al., 2020 and the results obtained in simulation, together with the value iteration procedure used to calculate the resulting policy are discussed in section 3.3.

3.1.1 AV's Architecture

Before talking about the MDP implementation for an AV decision-making, it is necessary to discuss the architecture used in automated vehicles, which appear after the subsumption architecture, cited in chapter 1, reached its "capacity ceiling". After implementing an airplane controller using a subsumption architecture, Hartley and Pipitone, 1991 noted that one flaw in such organization is the lack of modularity, since each behavior layer interferes with the one below only in terms of suppressing it or allowing it to change the output of the controller. This means that complexity scales fast according to the number of layers and changes in one layer needs to be met with changes in all superior layers.

Since the beginnings of the reactive approach other architectures were proposed, similar to the subsumption although having key differences with respect to it. One common point in these architectures was the presence of three layers, one containing no internal state, another containing information about the past and one last with predictions about the future, according to Gat, 1998. Some examples of such organization are the AuRa robot (Arkin, 1990), the task control architecture, notably used in (Simmons et al., 1997) and in many robots designed for NASA (Murphy, 2000) and the 3-tiered (3T) architecture (Gat, 1991). Every single one of these, since there are modified the ideas put forth by subsumption to be more modular and to allow planning, are classified as belonging to the hybrid paradigm (Murphy, 2000).

The three layers mentioned earlier and present in many of the hybrid approach architectures can be divided in (using the nomenclature presented in Gat, 1998):

- **Controller:** It is the layer that communicates with the actuators representing the low-level algorithms. Usually it does not have any internal states, it deals with the execution of some behavior passed from higher layers.
- **Sequencer:** Its role is to choose which behavior the controller must execute, while receiving information about the general planning from the upper layer.
- **Deliberator:** Executes the time consuming strategic planning tasks.

As it can be seen in figure 3.1 the relationship between layers are modified to input and output, differently from the suppression relation employed in the reactive paradigm. This structure is curious because it is outdated by the three levels of control during a driver task, presented in Michon, 1979: strategic level, tactical level and operational level (apparently there are no direct connection between the these works). The highest level deals with the global planning of the mission, the route determination, traffic level evaluation during the mission and possibly the risks involved in the chosen route. The tactical level deals with local constraints, adapting the route to interactions with other road users, while the operational level deals with the execution of the trajectory that possibly was modified by the previous layer. The same organization later appeared in Donges, 1999, with a different nomenclature (navigation, guidance, stabilization)

and is adopted by multiple AV decision-making publications (Ulbrich and Maurer, 2013).

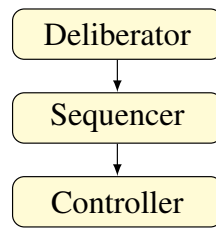


Figure 3.1 – Three tiered (3T) architecture

Nowadays the three layers architecture is the dominant one, being the one used in the Stanley vehicle at the DARPA challenge (Thrun et al., 2006) for example, and also in the majority of publications that deal with the decision-making of autonomous vehicles. Hence, the vehicle architecture considered here uses such organization in its planning section, as it is shown in figure 3.2.

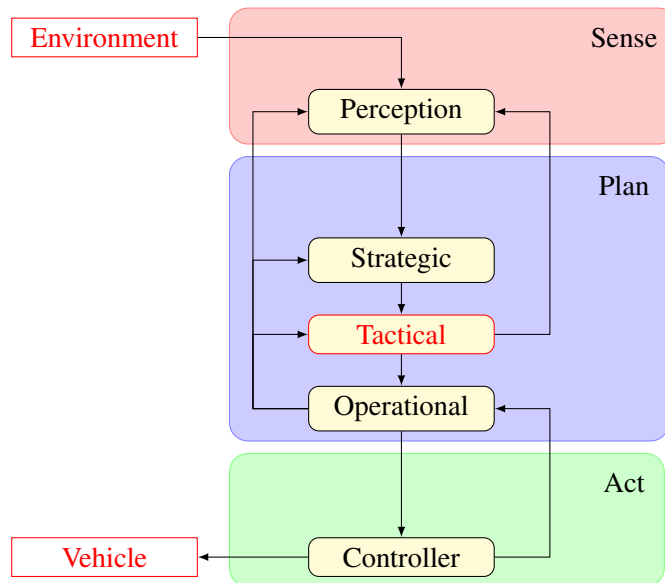


Figure 3.2 – AV architecture adopted

The perception node involves all communication, localization and mapping procedures together with sensor data acquisition. It can receive information from the operational layer and the tactical layer (data from the strategic layer may come using the tactical link since the latter is expected to function at a higher frequency than the former; the same is valid with the controller data and the operational layer link to perception). The plan step is formed by the three layers discussed earlier, with the strategic layer calculating the global route, the tactical adjusting this route to local constraints and the operational one converting the resulting route into a trajectory

for the controller. The proposed architecture works according to the hybrid approach, since the sense plan act structure is respected, but the tactical and operational layers are not limited by the strategy layer in their capacity to react to external stimuli.

The discussion in the next sections and next chapters will focus mostly in the tactical layer, considering the operational layer and the controller component only to explain how the implementation was done to allow the simulation of the entire system. An *a priori* route will be considered as given, and the perception component will be short-circuited by the simulation supervisor that can detect the configuration of every agent at any time.

3.1.2 Markov Decision Process

About the MDP choice

As it can be seen in the chapter 2, specifically in the section that presents the state of the art for decision-making, two main types of algorithms are used to give a robot, or in our case a vehicle, the capacity to function without supervision and in a flexible way, so as to deal with uncertainties and variations of environment parameters that are common in dynamic scenarios. One is the Markov-based algorithms and another is the deep learning (DL) based methods. Both have advantages and inconveniences, but one of them cannot, at least not currently, does not have some essential properties to be used as a decision-making model for AVs.

To model the deliberation process, in normal and ethical situations, a Markov decision process will be proposed in the rest of this chapter. This was done because this method offers a well defined theoretic framework to deal with uncertainty into action selection and also because it has three main characteristics that deep learning does not have (Barredo Arrieta et al., 2020):

- **Comprehensability:** The ability to represent its learned knowledge in some representation that is understandable by humans.
- **Explainability:** The ability to explain in human terms why a decision was taken.
- **Transparency:** If the model is understandable (there are different levels of transparency)^a.

^aThe important point is that deep learning methods are not transparent, the discussion about different levels of transparency is left to Barredo Arrieta et al., 2020

Given that every single decision made by an AV needs to be verified and validated, the three characteristics are essential, because not only one must be able to explain why an AV did what it did. It is also necessary to be certain concerning the limits of operation and which reactions

might occur in extreme situations, mechanically and ethically to be able to define when dilemma situations might occur or not; for example, for MDP it is possible to verify the convergence of its policy calculation, which theoretically is not possible with DL methods.

And the point of this document is to propose methods of ethical deliberation inspired in ethical theories, it is more coherent to do it in an understandable fashion. Of course, this does not mean that any type of ethical oriented decision-making using deep learning is possible, nor that it should not be researched and implemented, just that considering the points made previously this was not the choice made here.

MDP definition

A Markov decision process (MDP) is a stochastic process that has the Markov property, i.e., that is memoryless and that assigns values, positive or negative, to state transitions, according to the desirability of such transition (Garcia and Rachelson, 2013).

Definition 1 *Markov Property*

Let $\{X(t), t \geq 0\}$ be a stochastic process defined on a measure space (Ω, \mathcal{F}, P) , with Ω being the sample space, \mathcal{F} being the σ -algebra of Ω and P the probability measure on \mathcal{F} . $\{X(t), t \geq 0\}$ is a Markov process if:

$$P(X(t_{n+1}) \in \mathcal{F} \mid X(t_n), \dots, X(t_0)) = P(X(t_{n+1}) \in \mathcal{F} \mid X(t_n)) \quad (3.1)$$

The 5-tuple (S, A, T, R, γ) defines an MDP. What each component represents is detailed below:

Components of an MDP

- State space ($s_i \in S$): represents all possible process configurations.
- Action set ($a_i \in A$): represents the set of all possible actions available; triggers the transition from one state to another.
- Transition probability (T): represents the probability whether, given a state, executing an action takes the process to some state; formulated as $p(s_{t+1} \mid s_t, a_t)$.
- Reward function (R): quantifies how good or bad the transition to another state was according to a defined criteria.
- Discount constant (γ): represents the factor used to adjust the utility at a time $t+n$, $n \in [0, \infty]$ to the present (time t); defined as $\gamma \in [0, 1]$.

What makes an MDP a rather popular method to model stochastic processes is the ability to expand on the capacities from a usual Markov chain and add uncertainty into the transition between states. Figure 3.3 illustrates an MDP with three possible states and two actions. From the state s_0 , if the action a_0 is executed, there is a probability of $p_{t_n}(s_1 | s_0, a_0)$ that the process at t_{n+1} will be at the state s_1 and $p_{t_n}(s_2 | s_0, a_0)$ that it will instead be in s_2 , with $p_{t_n}(s_2 | s_0, a_0) + p_{t_n}(s_1 | s_0, a_0) = 1$.

The reward depends on the current state, the chosen action and the final state. If, from s_0 , the action a_0 is chosen and the process ends up in state s_2 the reward would be $r_{t_n}(s_0, a_0, s_2)$. The expected reward from state s_0 and action a_0 can be calculated as $r_{t_n}(s_0, a_0) = \sum_{i \in S'(s_0)} p_{t_n}(i | s_0, a_0) \cdot r_{t_n}(s_0, a_0, i)$, with $S'(s_0)$ being the set of possible next states from s_0 .

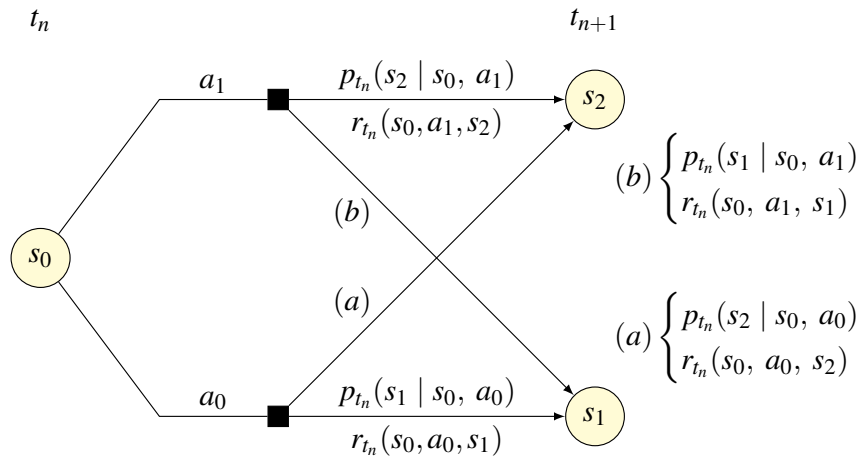


Figure 3.3 – Simplified example of an MDP

The output of an MDP is a policy, a data structure that encloses multiple decision rules, one for each time epoch. Each one of these decision rules give, for each state, the optimal action to be executed according to the transition probabilities and the reward function chosen (Puterman, 1994c). A policy usually is represented by the greek letter π . The decision rules can be Markov-based (the input is the current state) or history-based (the input is a series of past states and actions) and they can also be deterministic (one input gives one action) or stochastic (one input gives a distribution of possible actions). Equation 3.2 shows a policy based in Markov-deterministic decision rules.

$$\pi = (d_t(s_t), d_{t+1}(s_{t+1}), \dots, d_{N-1}(s_{t+1})) \quad (3.2)$$

There are multiple criteria to select which action should be executed at each state to transfer the process to another state, supposedly closer to the objective. The criteria that will be used for the rest of this document is the discounted reward criteria, which uses a discount

factor to scale down the influence of future rewards into the current decision. This constant also allows the convergence of the value function in infinite horizon MDPs. Solving an MDP means obtaining a policy that has the best reward throughout the entire execution, based on the current transition and all the future transitions that the process might choose. Equation 3.3 defines the value function, that evaluates the total amount of reward that a policy produces, and equations 3.4 show how to calculate the optimal policy can be determined.

$$V^\pi(s_t) = E \left[\sum_{t=0}^{\infty} \gamma^t r_t(s_t, a_t) \mid s_0 \right] \quad (3.3)$$

$$\begin{cases} V^*(s_t) &= \max_{a \in A} E \left[\sum_{t=0}^{\infty} \gamma^t r_t(s_t, a_t) \mid s_0 \right] \\ \pi^* &\in \operatorname{argmax}_{\pi \in \Pi^{HS}} V^\pi(s_t) \end{cases} \quad (3.4)$$

The calculated optimal policy in 3.4 was considered to be the most general possible, having the history-based stochastic type (HS). However, Markov-based deterministic policies are easier to calculate. The proposition 1 shows that for each history-based stochastic policy there exist a Markov-based stochastic (MS) one with equal value function¹, hence allowing the consideration of a Markov-based stochastic policy as a solution.

Proposition 1 *Equivalence of history and Markov based policies:*

Let $\pi \in \Pi^{HS}$ be a stochastic history-based policy; for each initial state there exists a Markov-based stochastic policy $\pi' \in \Pi^{MS}$ such that:

$$V_{\gamma}^{\pi'}(s) = V_{\gamma}^{\pi}(s) \quad (3.5)$$

To further verify that the optimal policy can be obtained using a deterministic policy, first the optimal equation needs to be determined. An infinite horizon will be considered, since the simplification for finite horizon can be easily done. The following simplification assumptions will be adopted:

MDP Optimality Assumptions

- Rewards and transition probabilities are stationary: they do not vary according to time.
- Rewards are bounded: $|r(s, a)| \leq M < \infty$.
- The state space is discrete.

¹The proof of this proposition can be found in Garcia and Rachelson, 2013 or Puterman, 1994b.

Since rewards and transition probabilities are stationary through time, then the policy itself can be considered stationary, i.e. being composed by only one decision rule, $\pi = (d(s_t), d(s_{t+1}), \dots, d(s_{t+1}))$. All these assumptions are valid regarding the implementation discussed in this and the next chapters. Then, the optimal value equation for infinite horizon MDPs is defined by equation 3.6, called *Bellman equation*.

$$V(s) = \max_{a \in A} \left[r(s, a) + \gamma \sum_{s_j \in S} p(s_j | s, a) V(s_j) \right] \quad (3.6)$$

Transforming equation 3.6 into the vector form the operator L on V can be defined, as shown in equation 3.7, for $\forall \pi \in \Pi^{MS}$.

$$LV = \max_{\pi \in \Pi^{MS}} (r_\pi + \gamma P_\pi V) \quad (3.7)$$

From it the proposition 2 can be established², stating the equivalence between deterministic and stochastic policies under the previous assumptions. The space \mathcal{V} refers to the space of all functions $V : S \rightarrow \mathbb{R}$.

Proposition 2 *Equivalence between stochastic and deterministic policies:*

For all $V \in \mathcal{V}$ and $0 \leq \gamma \leq 1$

$$\max_{\pi \in \Pi^{MD}} (r_\pi + \gamma P_\pi V) = \max_{\pi \in \Pi^{MS}} (r_\pi + \gamma P_\pi V) \quad (3.8)$$

Finally, from the stationary equivalent policy, it is only necessary to prove the existence of an optimal policy. For that, the next theorem,^{1,3} establishes the existence and uniqueness of an optimal solution for the *Bellman* equation, while theorem 2⁴ shows that there is a optimal policy π^* connected to such V^* .

Theorem 1 *Existence and uniqueness of Bellman equation solution*

If $V = LV$, for $V \in \mathcal{V}$, then $V = V^*$ and is the only optimal solution for the Bellman equation.

²Proof found in Puterman, 1994a; the proof of this proposition is done during the proof of theorem 1.

³The proof of existence can be found in Puterman, 1994a, and of uniqueness in Garcia and Rachelson, 2013.

⁴Proof found in Garcia and Rachelson, 2013.

Theorem 2 *Existence of optimal policy*For $\gamma < 1$:

- $\pi^* \in \Pi^{MS}$ is an optimal policy $\Leftrightarrow V^{\pi^*}$ is a solution of $V = LV$ and $V^{\pi^*} = V^*$
- Any stationary policy $\pi^* \in \operatorname{argmax}_{\pi \in \Pi^{HA}} r_{\pi} + \gamma P_{\pi} V^*$ is an optimal policy.

Then, under the conditions assumed earlier, an MDP has a unique optimal value solution and an optimal policy connected to it. How to determine such policy will be discussed in subsection 3.3.1.

3.2 AV decision-making model

In this section the physical model of the AV will be detailed, together with the definition of the state set and action set (subsection 3.2.1), the transition probability function (subsection 3.2.2) and the reward function (subsection 3.2.3).

3.2.1 State and action sets

State Set

Unlike many MDP (or POMDP) implementations that consider the state as the descriptor of the environment including other road users' configuration together with the AV's configuration as the state, the definition used here only includes the AV configuration, as seen in 3.9. All other road users are considered through their interaction with the AV in the reward function. The state is composed by the AV's middle rear-axis point coordinates (x, y) , direction θ , scalar velocity v and steering angle ϕ .

$$s_t = [x_t, y_t, \theta_t, v_t, \phi_t] \quad (3.9)$$

For the AV's model, a non-holonomic single-track vehicle model without slippage was chosen, as it can be seen in figure 3.4. The correspondent model, using front-wheel driving, is given by equation 3.10.

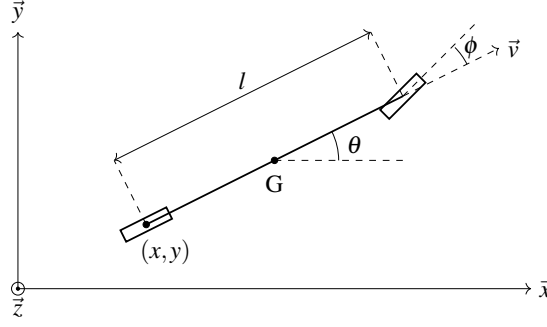


Figure 3.4 – AV's kinematic model

$$\begin{cases} \dot{x}_{t+1} &= v_t \cos \theta_t \cos \phi_t \\ \dot{y}_{t+1} &= v_t \sin \theta_t \cos \phi_t \\ \dot{\theta}_{t+1} &= \frac{v_t}{l} \sin \phi_t \\ \dot{v}_{t+1} &= a_{t,1} \\ \dot{\phi}_{t+1} &= a_{t,2} \end{cases} \quad (3.10)$$

Action set

Both acceleration and steering angle rate of change are defined as discrete actions for the MDP (3.11). The action set is then defined as the set of pairings considering the possible values for \dot{v} and $\dot{\phi}$ (values are presented in table 3.1).

$$a_t = (\dot{v}_t, \dot{\phi}_t) \quad (3.11)$$

State space discovery

Using this finite number of actions, the state space can be determined expanding the initial state during N transitions using the vehicle model given by 3.10 and each action. Each transition is defined to last Δt_{trans} . Such expansion is clearly exponential, requiring a limited number of transitions to form the state space for the problem. All actions were used at each state to generate possible next states. Considering N_a as the number of possible actions (i.e. size of A), equation 3.12 gives the total number of states. The state space discovery produces a tree of states,

with each level corresponding to a specific time epoch and the leaves being the terminal states.

$$N_s = \sum_{i=0}^N N_a^i \quad (3.12)$$

During the propagation of states, each one is checked if it remains in the bounds of the environment being simulated. If it is outside, then for this specific state the action chosen is not used and the state is removed from the tree. Another criteria to eliminate possible states is the mechanical limit for the steering angle of the AV. If the current state already achieve the maximum angle, then all actions that increase this angle are discarded. The same is valid for null or negative velocities coupled with actions with negative acceleration, since velocities smaller than zero are not considered in this implementation.

Figure 3.5 depicts an example of how the state space discovery is done for an hypothetical MDP with 2 transition and 3 possible actions. Starting from s_0 , the states s_4 , s_5 , s_6 and the ones that remain to be discovered from s_1 and s_2 are the terminal states of this process.

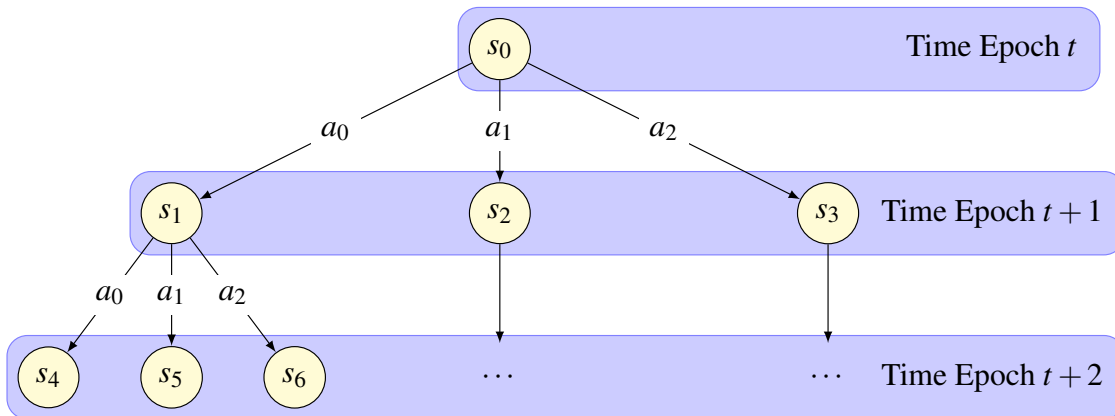


Figure 3.5 – Example of state space discovery

All constants that concern the state space discovery used in the implementation discussed here are displayed in table 3.1.

3.2.2 Transition function

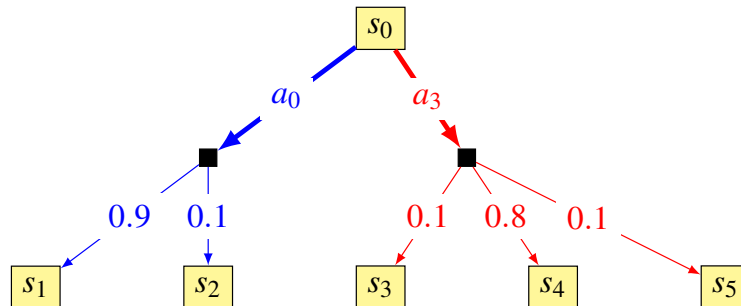
The main source of uncertainty in planning for AVs comes from other road users' behavior estimation. Since this knowledge is considered to be known during the planning (hence it is also why the state is defined entirely by the AV's configuration) the only source of uncertainty

Table 3.1 – Parameters used in the state space discovery

Parameter	Possible values
$\dot{v}(m/s^2)$	$-1, 0, 1$
$\dot{\phi}(^\circ/s)$	$-40, -20, 0, 20, 40$
$\Delta t_{trans}(s)$	0.5
N (number of transitions)	4
N_a (number of actions)	15

becomes the imprecision due to the vehicle model considered. The transition probability was defined using the steering angle rate of change, i.e. there are no uncertainty concerning changes in velocity.

In each set of actions with equal acceleration, the chosen action will have a probability of 0.8 of being successful, meaning that for state s_0 , in figure 3.6, the action a_3 takes AV to s_4 with probability of 0.8 and to s_3 or s_5 with probability of 0.1 each. For example, if $a_3 = (-20, -1)$, there is an 0.8 probability that the next state will be the correct one, given the model equations and the current state, 0.1 probability that the process ends up in the next state from $(-40, -1)$ and 0.1 that it will arrive in the state from $(0, -1)$. In cases when only one neighbor exists, notably when the steering angle is at its mechanical limit, then the transition probabilities become 0.9/0.1 (action a_0).

Figure 3.6 – State transition uncertainty for a_0 and a_3

The choice of parameters for the transition probability was made arbitrarily. A possible tire slippage, that is not modeled by the single track model presented in figure 3.4, is represented by the incertitude in the AV's steering angle rate of change with values (0.1 and 0.8) inspired by MDP grid-like applications that are frequently used as an example of Markov processes. Such definition is enough to display the entire AV's behavior but it is without saying that for a real application some form to calculate this probability, including the uncertainty of the other road users' behavior, is necessary (and it will be presented in chapter 5).

3.2.3 Reward function

The reward function indicates to the system which actions are preferable in regards to achieve a certain goal. It is based on two parameters: performance (s_{perf}) and action consequences (s_{conseq}), as given by equation 3.13.

$$R(s_t, a_t, s_{t+1}) = s_{\text{perf}} + s_{\text{conseq}} \quad (3.13)$$

Performance term

Performance for the vehicle is measured as lateral distance to the strategic path, direction offset and time to arrival at goal point. Equation 3.14 calculates the reward using three variables: q_{lat} represents the lateral distance to the trajectory, $\Delta\theta$ is the offset angle between the direction of the AV and the trajectory and q_{eta} represents the approximated time of arrival (ETA).

$$s_{\text{perf}} = w_{\text{lat}} \cdot q_{\text{lat}} + w_{\text{dir}} \cdot \Delta\theta + w_{\text{eta}} \cdot q_{\text{eta}} \quad (3.14)$$

Both variables q_{lat} and q_{eta} are actually offsets from the real lateral distance (d_{lat}) and the approximated time to arrival (t_{eta}), as equations 3.15 and 3.16 show. For the lateral distance, since the ideal is that $d_{\text{lat}} = 0$, the constant r_{lat} sets the reward of this ideal situation. The same idea is applied with r_{eta} , where it represents the maximum allowed ETA before q_{eta} becomes a cost (both w_{lat} and w_{eta} are positive).

$$q_{\text{lat}} = r_{\text{lat}} - d_{\text{lat}} \quad (3.15)$$

$$q_{\text{eta}} = t_{\text{eta}} - r_{\text{eta}} \quad (3.16)$$

This approximated ETA is calculated using the projection of the AV's velocity onto the direction of the distance between the front axis middle point ($\mathbf{p}_{\text{fr}}^{\text{AV}}$) and the trajectory goal (equation 3.17). More details about how this projection is done will be presented by equation 3.21. It is an hypothetical measure, since the vehicle does not necessarily has the same direction than this distance vector, but is enough to capture some idea of how much time it will take to arrive at the goal given the current speed and direction.

$$t_{\text{eta}} = \frac{d_{\text{obj}}}{v_{\text{av}}^{\text{proj}(d_{\text{obj}})}} \quad (3.17)$$

If the AV has a velocity equal to zero, then equation 3.14 is replaced by a fixed cost c_{eta} . The weights w_{lat} , w_{dir} , w_{eta} together with the constants r_{lat} and r_{eta} are calibrated heuristically, enabling the AV to arrive at its goal following a trajectory given by the strategic layer of the AV. Since the strategic layer is not of concern here, the trajectory comes from the simulation's input and is the same throughout the execution.

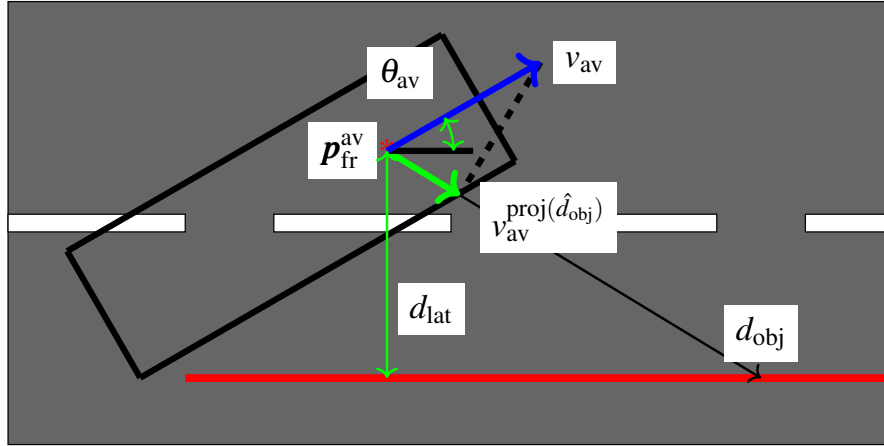


Figure 3.7 – Performance reward parameters

Action consequences term

The action consequences score is measured by two parameters, adherence to the traffic code (s_{traf}) and proximity to other road users (s_{prox}), as equation 3.18 indicates. The values s_{traf} and s_{prox} can assume negative values or 0, with the latter approximately 10 times lower than the former. This difference is necessary since s_{prox} has two functions: evaluate how close the AV is from other road users and detect collisions; if no collisions are detected the term s_{traf} must always prevail in case of traffic code violation. In case of a collision, as said before, the entire reward function (equation 3.13) is replaced by a negative constant costs.

$$s_{conseq} = s_{traf} + \sum_{i=1}^{i < N_{ag}} s_{prox}^{av,ru_i} \quad (3.18)$$

However, traffic code violations are not considered to be constitutive of a dilemma situation event. If, for example, there is one action that violated the traffic code and all others end up in collisions, then the AV will execute the action that violates traffic code to avoid the accident. The objective of this implementation is to focus in unavoidable collisions, not in the ethical and legal question of a possible trade-off between code violation and possible accidents. A cascade of

if-elses emulates some traffic rules into the AV's decision process by adding the resulting costs. If no infringement is detected, $s_{\text{traf}} = 0$.

If the AV's velocity surpasses the legal limit, then the cost c_{vel} is added to $s_{\text{traf}} = 0$, but if the acceleration is negative, then r_{acc} is also added, to favor actions that decrease the velocity in this context. Five points are considered to analyze a possible AV intrusion in the opposite lane or in the sidewalk: its four corners and the rear axis middle point. If two of these points are in the opposite lane, then $s_{\text{traf}} = s_{\text{traf}} + 2 \cdot c_{\text{oplane}}$ (if it is inside the sidewalk and the opposite lane the cost accumulate).

Traffic code evaluation

- Velocity above limit: add the cost c_{vel} to reward.
- AV in opposite lane: add the cost c_{oplane} to reward for each point in the opposite lane.
- AV in the sidewalk: add the cost c_{sidewalk} to reward for each point in the sidewalk.

To ascertain if the AV is too close to another road user, a minimal distance frontier is defined in figure 3.8. These security zones are the spaces between the vehicle's body and the red lines. All positions, orientations and velocities used to evaluate these parameters correspond to the time epoch t , the same as the current state (unless it is said otherwise). These frontiers represent an implicit risk measure for the reward function, since their role is to verify if breaking while maintaining the same direction is still a valid collision avoidance strategy for this state and action and if it is not, the action's reward is negatively impacted.

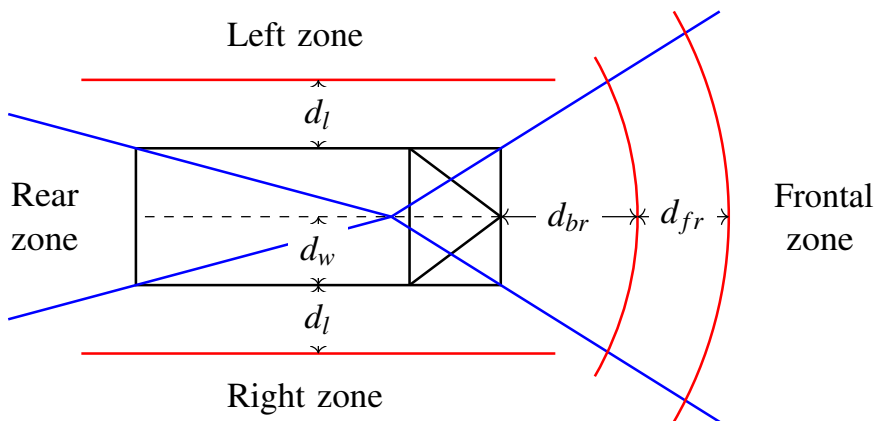


Figure 3.8 – AV with all the proximity limits

The region around the vehicle are divided in front, left, right and back. The front is defined as the region inside the frontal cone, delimited by the lines formed by the two frontal

rectangle's corners and the frontal axis middle point (blue lines in figure 3.8). The left zone is defined by the two left corners and the frontal axis middle point; the right zone is defined analogously, while the rear zone is defined by the rear cone. The frontier checked depends on with region the closest road user corner is. If it is inside the frontal zone, then the frontier defined by $d_{fr} + d_{br}$ will be used to verify if there is a potential dangerous situation, while if it is in one of the lateral zones, d_l is used. No frontier in the rear is defined.

Lateral frontiers are placed at a distance equal to d_l from the AV's body (d_w). Equation 3.19 gives the calculation of the d_l , with all parameters corresponding to the current time epoch t . This variable takes into account the lateral security distance d_{sec_l} , that is a static input of the simulation, and the lateral displacement of the other road user during t and $t + 1$, evaluated by $\Delta t_{trans} \cdot \|\vec{v}_{ru}^{proj(\hat{\theta}_{av}^\perp)}\|$. To calculate this displacement, the velocity of the other road user is projected perpendicularly to the AV's direction (equations 3.20 and 3.21), this last defined by 3.22.

$$d_l = \Delta t_{trans} \cdot \|\vec{v}_{ru}^{proj(\hat{\theta}_{av}^\perp)}\| + d_{sec_l} \quad (3.19)$$

$$\vec{v}_{ru}^{proj(\hat{\theta}_{av}^\perp)} = \vec{v}_{ru} - v_{ru}^{proj(\hat{\theta}_{av})} \quad (3.20)$$

$$v_{ru}^{proj(\hat{\theta}_{av})} = \hat{\theta}_{av} \cdot \vec{v}_{ru} \quad (3.21)$$

$$\hat{\theta}_{av} = [\cos(\theta_{av}), \sin(\theta_{av})] \quad (3.22)$$

The frontal limit is given by d_{fr} plus d_{br} , the breaking distance given v_{av} (the AV's velocity at t) and calculated by equation 3.24. The distance d_{fr} , as d_l did, also takes into account the displacement of other road users during Δt_{trans} , but now using the term $\Delta t_{trans} \cdot v_{ru}^{proj(\hat{\theta}_{av})}$ to do so, with the velocity calculated previously by equation 3.21. The constant d_{sec_f} is the frontal security distance, similar to the previous d_{sec_l} , while the deceleration constant \dot{v}_{break} is also a static input of the simulation.

$$d_{fr} = -\Delta t_{trans} \cdot v_{ru}^{proj(\hat{\theta}_{av})} + d_{sec_f} \quad (3.23)$$

$$d_{br} = \frac{1}{2} \frac{(v_{av})^2}{\dot{v}_{break}} \quad (3.24)$$

It is assumed as an hypothesis that if an emergency breaking happens the AV will continue at the same direction ($\phi_t = 0$), thus no term related to the change in direction is necessary in equations 3.19 and 3.23. The behavior of the other road user in the calculation of these equation is that it will continue in the same direction, with the same velocity.

If a road user is predicted to remain outside the security frontiers, then $s_{prox}^{av,ru_i} = 0$; otherwise the correspondent score becomes a cost, calculated by equation 3.25. To check this

condition the position of the road user at $t + 1$ is used. In equation 3.25 there is a static cost, c_{st} , to punish the next state due to a frontier violation, a static weight w_v , defined heuristically and the difference of velocity variations.

$$s_{\text{prox}}^{\text{av,ru}_i} = c_{st} + w_v \cdot \left[\left(v_{\text{ru}_{i,t+1}}^{\text{proj}(\hat{d}_{t+1})} - v_{\text{av}_{t+1}}^{\text{proj}(\hat{d}_{t+1})} \right) - \left(v_{\text{ru}_{i,t}}^{\text{proj}(\hat{d}_t)} - v_{\text{av}_t}^{\text{proj}(\hat{d}_t)} \right) \right] \quad (3.25)$$

Each of the difference of velocities refers to a time epoch. Defining the distance vector between the AV and the considered road user (RU_i) as the vector from the AV's front axis middle point ($\mathbf{p}_{\text{fr}}^{\text{av}}$) to the closest corner of the road user's body, the difference of velocity refers to the projection of each velocity onto this distance vector.

This measure shows if the AV is converging on (Δv negative) or diverging from (Δv positive) the road user with the proposed action, while the weight w_v (always negative) controls how important it is to avoid a situation where the AV tends to cross other road user's immediate direction while being too close to it. The equation 3.21 is used to calculate the projected velocities, but with the distance vector (\hat{d}_t) instead of the AV's direction vector. Figure 3.9 given an example of the evaluation procedure of this velocity variation for $t + 1$.

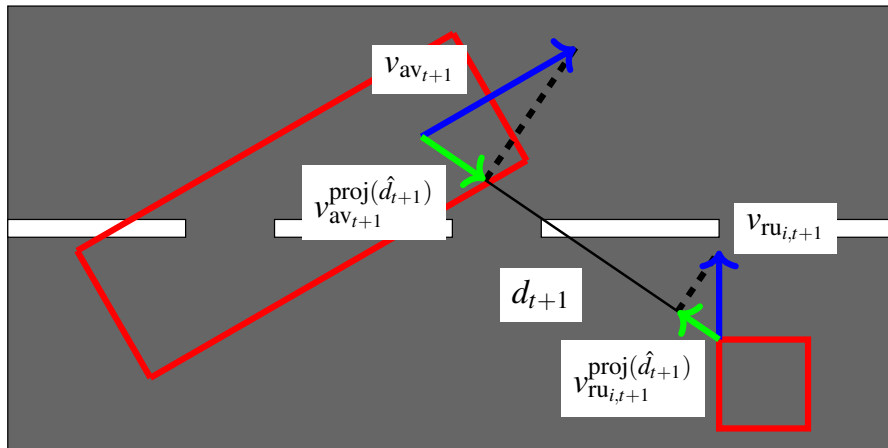
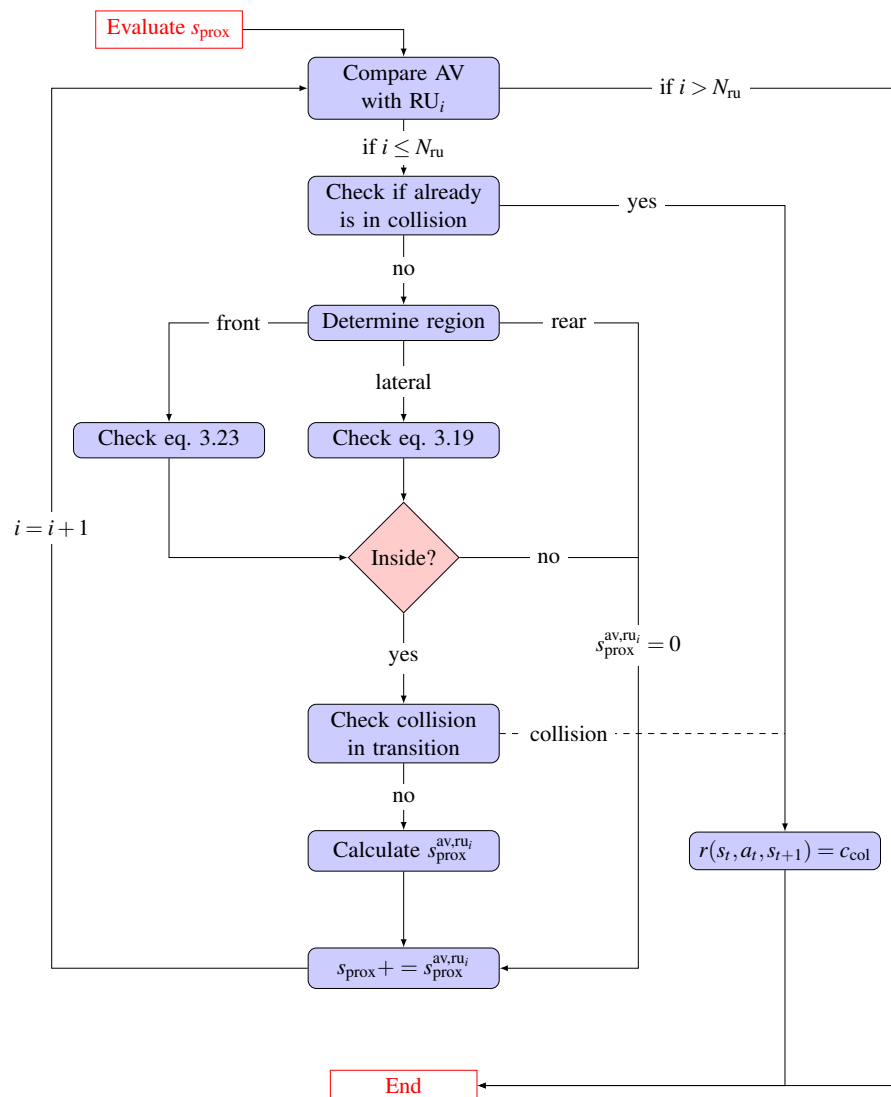


Figure 3.9 – Calculation of velocity projections

Collision detection

To detect collisions during a transition, each road user's position (including the AV) is predicted ten times during a transition period Δt_{trans} . Each road user (including the AV) is represented by a rectangle (pedestrians are particularly represented by squared), reflecting their real dimensions. This evaluation is done before the verification if the road user is too close to the AV, as it can be seen in the workflow presented by figure 3.10.

Figure 3.10 – Workflow for s_{prox} calculation

In two moments the algorithm checks for collisions involving the AV: at the beginning of the procedure to evaluate $s_{\text{prox}} += s_{\text{prox}}^{\text{av,ru}_i}$ in equation 3.18, to determine if the AV already collided and if the road user i is detected to be inside the safety frontier. In this last case, the collision is checked during the transition. If it is detected, the algorithm stops and sets 3.18 to c_{col} . Two different situations can be labeled as a collision, an intersection with the rectangle that represents the other road user, or if the AV is out of the bounds of the environment being simulated. The limits of the state space are considered to be rectangular walls.

According to Gottschalk et al., 1996, one possible way to verify if two convex 2d-

polygons intersect each other is to search for an axis where the projection of both polygons do not intersect each other. If this axis exist, then the line perpendicular to such axis separates both polygons. It is only necessary to check, for each polygon, a set of lines parallel to each side, since the separation line, if it exists, is parallel to one of the edges of the polygon; hence, since a rectangle have two parallel sides, only two lines need to be checked, per rectangle. This theorem is originated from the hyperplane separation theorem, presented in theorem 3 (Boyd and Vandenberghe, 2004):

Theorem 3 *hyperplane separation theorem*

Suppose C and D are nonempty disjoint convex sets ($C \cap D = \emptyset$); then there exists $a \neq 0$ and b such that $a^T x \leq b$ for all $x \in C$ and $a^T x \geq b$ for all $x \in D$. The hyperplane $\{x \mid a^T x = b\}$ is called a separating hyperplane for the sets C and D .

Figure 3.11 shows an example of the procedure for the rectangle $V_0V_1V_2V_3$. One of the sides taken as a separating axis, V_0V_1 , generating the axis r , has an intersection between the projection of the square $P_0P_1P_2P_3$ and the projection of the rectangle, which is the edge V_0V_1 itself. But the second side chosen as separation axis, V_0V_3 (line l) does not have an intersection between projections, therefore it is a separation axis (and a line perpendicular to it separates both rectangles).

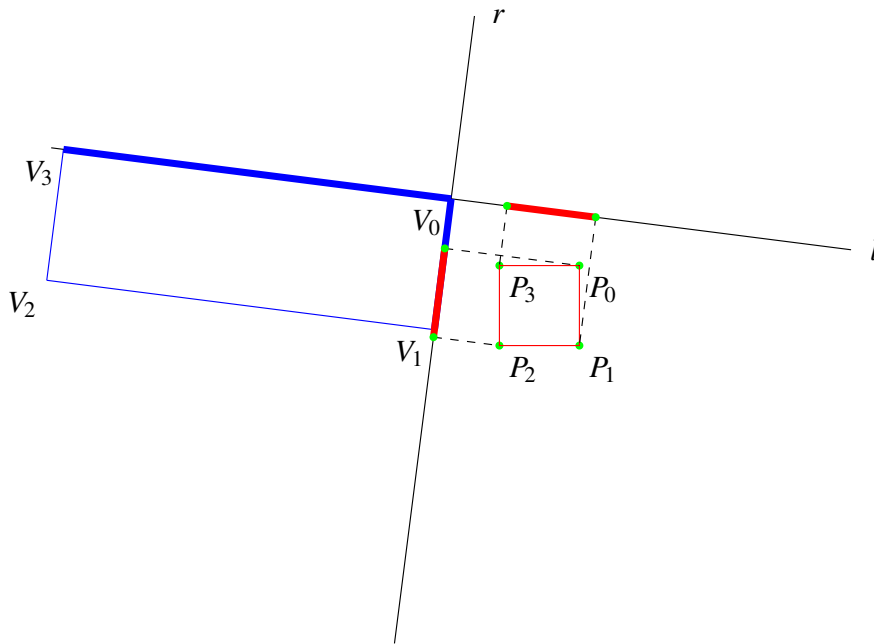


Figure 3.11 – Calculating the separation line

Table 3.2 shows the values of all constants used in the reward calculation. The information

about the parameters for the AV (mass, acceleration, etc.) and for the other road users in the simulation will be given in the next section.

Table 3.2 – MDP parameters used

Parameter	Value	Parameter	Value	Parameter	Value
w_{lat}	60	w_{θ}	-50	w_{eta}	-35
r_{lat}	10	r_{eta}	30	c_{eta}	-1250
c_{vel}	-1000	r_{acc}	750	c_{oplane}	-50
$c_{sidewalk}$	-1500	w_v	30	d_{sec_f}	5m
d_{sec_l}	0.5m	Δt_{trans}	0.5s	c_{st}	-1000
c_{col}	-15000	γ	0.9		

In the proposed implementation the γ value was arbitrarily set to 0.9, given that only four transitions are chosen and that the current AV needs to stay aware of future collisions and transgressions. A lower γ might dampened the costs given to collisions. As for the other constants, they impact the reward and consequently the behavior of the AV in the following manner:

- The parameters related to the proximity term of the reward (c_{st} , w_v and c_{col}) should have a highly punitive nature to the vehicle with no possibility of generating a positive reward. The decision if it should be calculated is controlled by the frontier around the AV (d_{sec_l} , d_{sec_f}).
- The parameters related to traffic code violations (c_{vel} , r_{acc} , c_{oplane} and $c_{sidewalk}$) have a similar nature but they are always verified.
- The parameters related to the performance have the possibility to produce positive rewards, which are controlled in value by the constants r_{lat} and r_{eta} . The cost of not driving is expressed by c_{eta} . How important each of the parameters related to lateral distance, direction offset and estimated time of arrival is defined by the three weights w_{lat} , w_{θ} , w_{eta} , respectively.

3.3 Results and discussion

3.3.1 Value iteration

The output of a MDP algorithm is a policy π^* which, for each state, yields the optimal action to be executed, as discussed in section 3.1. It was already determined that this optimal

policy is Markov-based, deterministic and that it exists. The only remaining piece of the puzzle is a tool to calculate it given an MDP formulation with the assumptions declared before.

$$V^*(s) = \max_{a \in A} \left[r(s, a) + \gamma \sum_{s_j \in S} p(s_j | s, a) V(s_j) \right] \quad (3.26)$$

Equation 3.26 needs to be solved to obtain the policy. There are many ways to solve a Bellman equation, but the one adopted here will be the value iteration, where in each iteration the value for all states is calculated, until an equilibrium is reached. At this point the value function with the selected actions will be the solution, and the optimal policy can be determined by the actions that produced such values (Garcia and Rachelson, 2013). This algorithm is shown in 1.

At each iteration the mean squared error of the difference between each value calculation for every state is calculated. Such iteration can also be represented using the L operator, as is in equation 3.27. The algorithm is stopped if the calculated error becomes smaller than $\varepsilon = 0.5$ (defined heuristically) and the current value iteration is then used by equation 3.28 to obtain the policy.

$$V^{t+1} = LV^t \quad (3.27)$$

Algorithm 1 convergence is verified by theorem 4 in Puterman, 1994a.

Theorem 4 *Optimal solution via value iteration*

Let $V \in \mathcal{V}$, $\varepsilon > 0$ and let V^t satisfy 3.27 for $n \geq 1$; then:

1. V^t converges in norm to V^*
2. The stationary policy defined in equation 3.28 is ε -optimal

Algorithm 1: Value Iteration**Data:** Spaces S and A , functions R and P , constant γ , environment data e **Result:** Policy $\pi^*(s)$, Optimal Value $V^t(s)$

```

1  $t = 0$ 
2 do
3   for every  $s_i \in S$  do
4      $V^{t+1}(s_i) = \max_{a \in A} [R(s_i, a, s', e) + \gamma \cdot \sum_{s'_j} P(s'_j | s_i, a) V^t(s'_j)]$ 
5      $t = t + 1$ 
6   end
7 while  $MSE(V^{t+1} - V^t) > \epsilon$ 
8 for every  $s_i \in S$  do
9   
$$\pi^*(s_i) = \operatorname{argmax}_{a \in A} V^t(s_i) \tag{3.28}$$

10 end

```

3.3.2 Simulation

Configuration

Using the WeBots simulator (Michel, 2004), the aforementioned algorithm was tested in the scenario shown by figure 3.12. A real depiction of the simulation environment is shown in figure 3.13. While the AV is traveling in its lane, two pedestrians, P1 and P2 suddenly cross into the street (both start moving at $t_{\text{simul}} = 4s$). Every behavior from other road users are considered to be known; for example, the other vehicle's behavior is assumed to be the will to cruise the entirety of the straight road with $10m/s$. For the two pedestrians, before they start moving the AV consider that they will remain stopped. When they start moving, then the AV assumes that they will cross the road.

All MDP parameters and the physical information of road users' are shown in table 3.3.

Vehicle control

The obtained policy is used as a one shot fashion. Since the state space has a tree structure, from the current state to all other possible ones, with a fixed number of layers (which is equal to four in this implementation), the policy has the same structure, with an action for each state. Hence, starting from the root, one can determine the sequence of action that should be executed.

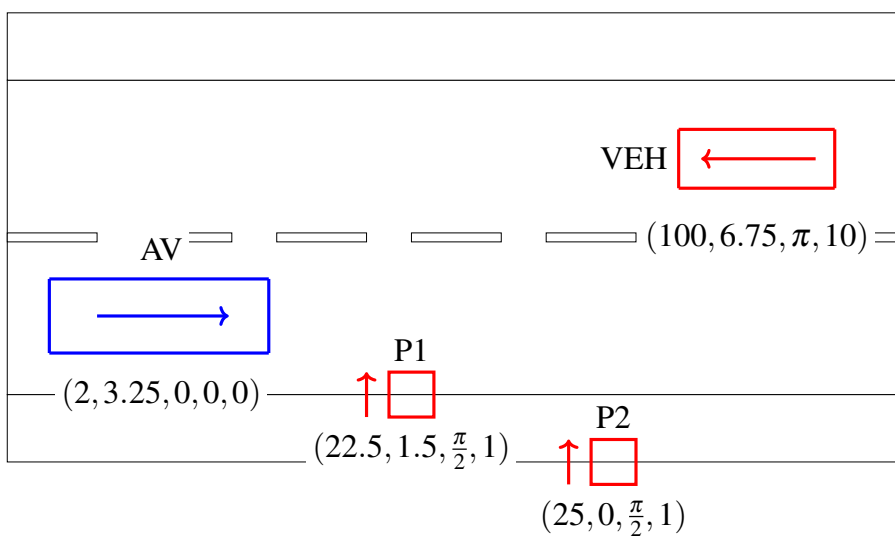


Figure 3.12 – Initial simulation setup (other road users’ configuration is represented by (x, y, θ, v) , not in scale)

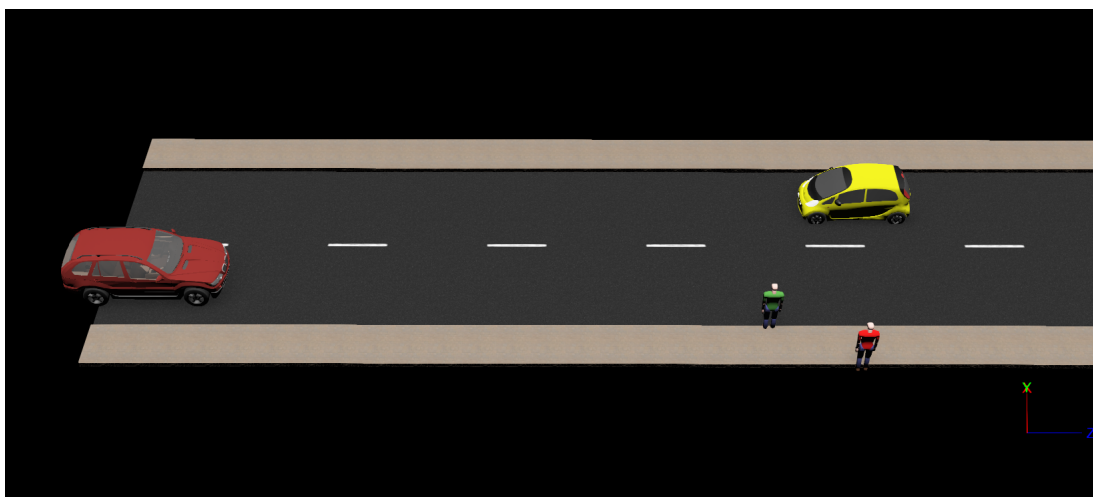


Figure 3.13 – Simulation environment

It is this sequence of actions that is used to command the AV.

Every action sequence originated from a policy is used during one second, considering that the four transitions can plan the trajectory for two seconds. After this period the MDP is re-initiated, with a different state space, given that the position of the AV changed, and a new policy originates another trajectory. The reason why this two transition buffer is necessary will be made clear in the next chapter, but it can be said that the AV, at the end of the policy execution,

Table 3.3 – Road users’ physical properties

	AV	Pedestrian	Vehicle
Mass (kg)	2105 + 80	80	1065 + 80
Width, Height (m)	4.853, 1.65	0.625, 0.625	3.475, 1.310
Wheelbase (m)	2.933	–	2.55
Acceleration (m/s^2)	1	–	1
Deceleration (m/s^2)	–7	–	–7
Steering angle limits ($^\circ$)	25	–	25

should be able to know that no collision is predicted before the policy recalculation. Such justification accounts for the need to keep one transition as buffer. As it will be explained in subsection 4, two actually need to be kept.

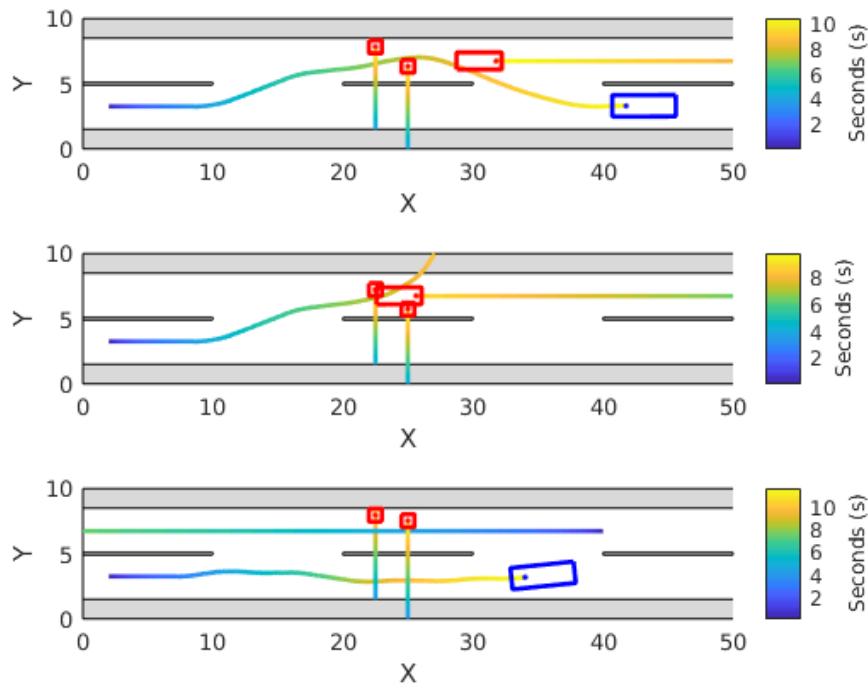
Inside the simulation the velocity of car-like objects are controlled by a cruising control, meaning that they decelerate and accelerate using a constant acceleration value. The steering angle has a clear mechanical limit, as has its rate of change. For the pedestrians, a constant velocity is used in its displacement, with the body’s articulation movements being controlled by the simulator’s controller.

3.3.3 MDP policy results

The AV should not produce accidents or critical situations on its own. Varying the initial position of the vehicle (x_{veh}) and the velocity direction of one of the pedestrians (θ_{ped}) relative to figure 3.12, different AV behaviors can be observed, for this precise road configuration. Each policy predicts two seconds ahead, with one second used for control ($(\dot{v}, \dot{\phi})$ are directly fed into the AV), before the policy being recalculated.

The AV needs at least two transitions to verify if the next action will not cause an accident, and that a state has actions which will not necessarily lead to a dilemma situation. And if it does, this fact needs to be signaled to the entire state space to propagate the high cost connected to a collision to the respective states. It is because of this necessity to look ahead of time that the execution of a policy is limited to $N - 2$ (N is given by table 3.1), before the recalculation of the policy. The weights of equations 3.14 and 3.25 were set heuristically to obtain a result representative of the MDP defined in section 3.2.3.

The first of the AV’s behavior plots shown in figure 3.14, for $x_{veh} = 100m$ is acceptable (but not ideal given that it should have stopped and waited the crossing), avoiding the pedestrian and returning to the other lane afterwards. All road user states are considered observable and known, therefore at the beginning of a policy calculation the AV can assert correctly all its states

Figure 3.14 – AV's behavior for $x_{veh} = 100, 80, 40\text{m}$

and predict changes throughout the decision horizon. But since the AV only has the capacity to predict two seconds ahead of time, it starts the maneuver without knowing how it will end.

This can be seen more clearly in the second plot, for $x_{veh} = 80$ when after the beginning of the maneuver, it reaches a critical situation and simply collides with the wall beyond the sidewalk (every limit of the environment is defined as a static wall). But if five transitions is used to plan the policy the collision is avoided, as it can be seen in figure 3.15. This is because, according to the Y position graphic at figure 3.16 the vehicle under four transitions do not see clearly the other vehicle approaching and thus places himself ahead of the vehicle in the implementation with five transitions. The moment where collision becomes unavoidable is signaled by point 15, where the orientation for the first case takes a turn to the left, while in the second it turns to the right and accelerates to return to its lane in time.

Adding one mode transition to the policy produces a viable trajectory, but increases the total calculation time from $t = 60.1\text{s}$ to $t = 802.4\text{s}$, due to the exponential nature of the state space discovery process.

Starting the other vehicle at $x_{veh} = 40\text{m}$ (third plot in figure 3.14) creates a situation in which the AV cannot execute the avoidance trajectory. It is perhaps intuitive for a human driver to stop and wait for the pedestrians, but here the AV only reduces its velocity at a certain point and continues to move until the path is clear. Even with five transitions (second plot in figure 3.15)

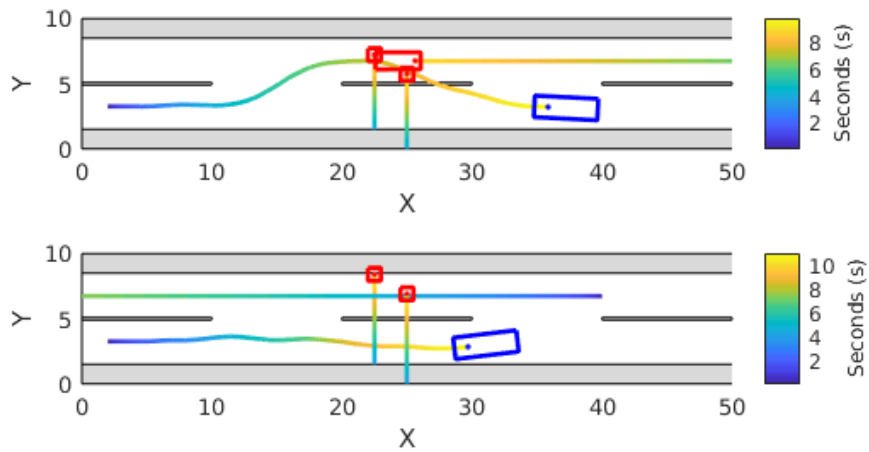


Figure 3.15 – AV’s behavior for $x_{veh} = 80, 40m$ with 5 transitions

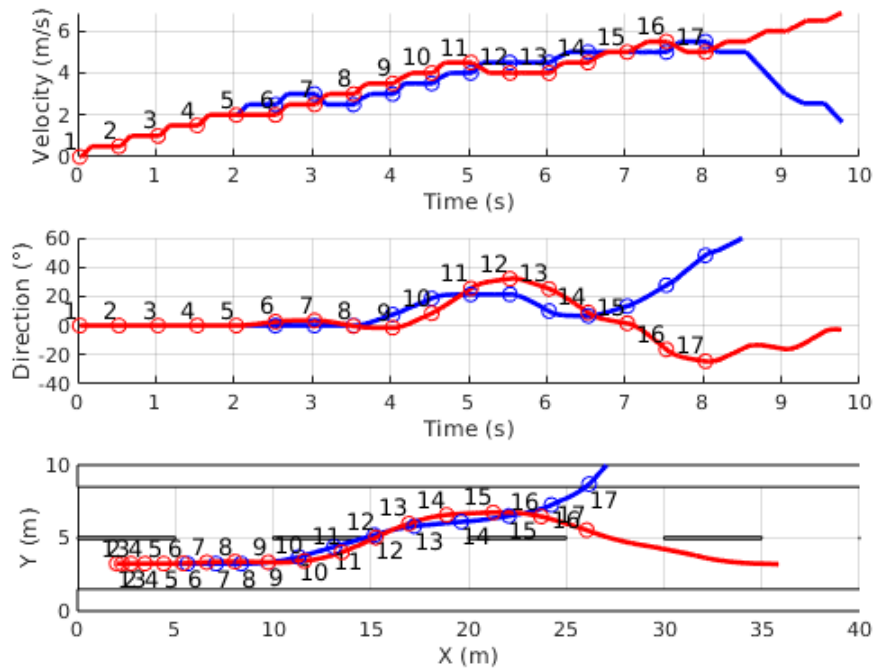


Figure 3.16 – AV’s velocity, direction and trajectory for $x_{veh} = 80m$

the behavior is similar, meaning that it needs more transitions to receive the delayed reward from the future and understand that a stop and go behavior is the right one in this context (including in the cases where it can use the opposite lane without ending up in an accident).

Some conclusions can be drawn from these three different situations:

- I The decision-making horizon needs an appropriate length to anticipate short-term behaviors from other road users. However, the exponential nature of the state space discovery process (subsection 3.2.1) must be avoided.
 - i One way to do this is to sample the most probable next states given a current state and an action, focusing only on these discovered next states (as some POMDP solvers do). In theory, considering a fixed transition function, if a bigger number of transition were to be used, so as to only eliminate the states inside the decision horizon deemed not useful to the AV, the complexity of the resolution might be decreased. However, the heaviest step to solve this MDP is the evaluation of the reward function, not the iteration in itself, and since it is the reward the indicator of pertinence, this solution would not accomplish what is expected.
 - ii Another possible solution involves the inclusion of all other road users into the state and to evaluate a transition probability function capable to identify the most pertinent states, given the interaction between all users. This could break the exponential character of the problem if the actions of the AV were subjected to the same treatment as the states of the other users, i.e. not all actions need to be considered, only the ones that somewhat optimize one transition.
- II In reality the decision horizon is dependent of the AV's velocity. If it increases too much the capacity to observe sufficiently ahead of time needs to increase, in the case that transition time (Δt_{trans}) is fixed. One can see the beginning of this problem in the situations commented above, but the real problematic becomes evident if one considers high speed limits. Planning should occur in small steps of trajectory, even more since the trajectory planned is based on piece-wise linear segments. Without a small enough transition time it is clear that the maneuver capability of the AV decreases rapidly.
- III Risk evaluation is also necessary to stop the AV from starting a maneuver that cannot be completed in one policy, as it was done in figure 3.14. Defining the state space without a fixed number of transitions and using the end of a maneuver as a sign of policy termination would allow the AV to start and end in low-risk states.

Changing the direction of the first pedestrian from $\frac{\pi}{2}$ rad to $\frac{3\pi}{4}$ rad creates a different situation, but the results obtained are similar. Figure 3.17 shows the trajectory at $x_{veh} = 100$ m and 40 m and figure 3.18 at $x_{veh} = 40$ m for five transitions.

In comparison with figure 3.14, figure 3.17 produces similar behavior, where an additional transition allows the AV to avoid an accident. For $x_{veh} = 40$ and 5 transitions, it makes the AV choose to accelerate to avoid an accident, which in turn causes it to invade the sidewalk. With

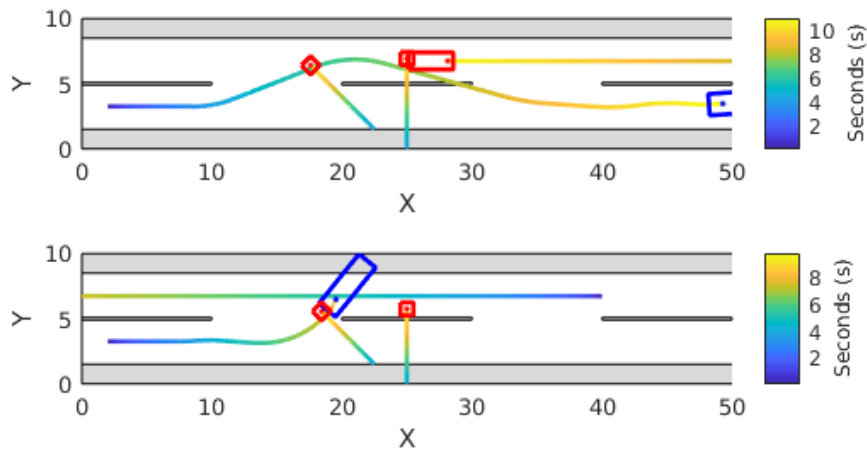


Figure 3.17 – AV's behavior for $x_{veh} = 100, 40\text{m}$ and $\theta_{ped} = \frac{3\pi}{4}\text{rad}$

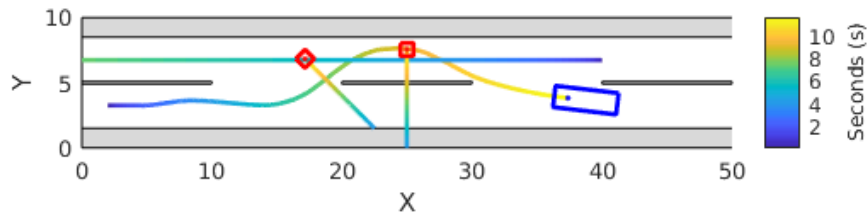


Figure 3.18 – Trajectory for $x_{veh}=40\text{m}$, $\theta_{ped}=\frac{3\pi}{4}\text{rad}$ for 5 transitions

only four transitions the collision cost was not detected, only the sidewalk invasion cost is, which fix the future outcome to a collision with the environment's border.

Another critical point in the implementation is the weights used for the reward function. All the considered weights and constants were defined heuristically for the specific use case considered. Even so, the best values obtained still do not reproduce an ideal trajectory, which can be seen in figure 3.14 between [5, 15] at the X axis. The small but perceptible change in AV's direction, an oscillation in its direction, is not illegal but rather pointless.

3.4 Conclusion

This chapter analyzed an proposition of an MDP algorithm for the decision-making of an AV under normal situations. Before its definition, a discussion about which architecture is used in automated vehicles in general was addressed and the question of why a Markov-based algorithm was chosen instead of a deep-learning approach was also answered afterwards. After the MDP model was formally defined, from the necessary components to the existence and uniqueness

of its resulting optimal policy to subsequently define all the particularities of the model for this instance.

The proposed algorithm was tested in a simulation environment, considering multiple configurations from other road users. It was observed that, for the heuristic weights chosen for the reward function, the AV behaved correctly in most situations, but some problems were flagrant. The exponential nature of the state set discovery did not allowed the use of an appropriate horizon prediction to correctly account the behavior of the other road users with the AV's velocity worsening this problem if the transition time is kept the same. No guarantee is done towards the finalization of a maneuvers before the end of the prediction horizon, which demands at least a risk assessment to avoid terminating in an unsafe state.

Chapter 4

Dilemma Deliberation

Contents

4.1 Dilemma situations	78
4.2 The definition of harm	79
4.2.1 Δv for collision	82
4.2.2 Vulnerability constant	83
4.3 Ethical deliberation process	86
4.3.1 Ethical Valence Theory	86
4.3.2 Ethical Optimization	90
4.3.3 Value iteration for dilemma scenarios	94
4.4 Results and discussion	98
4.4.1 Using EVT as ethical deliberation	98
4.4.2 Using the Ethical Optimization profiles to deliberate	102
4.5 Conclusion	104

Up until now dilemma situations were not treated by the algorithm proposed in chapter 3, even if when the collision was inevitable, notably the case in graphic 3.14. Two methods to treat such situation will be exposed in this chapter: the ethical valence theory (EVT), proposed by Evans et al., 2020 and the ethical optimization, proposed in de Moura et al., 2020. Both methods use to guide its decision the estimated harm that may happen in an accident, while the EVT also uses the ethical valence with a deliberation procedure to choose the most appropriate action. This chapter ends with the explanation about how the value iteration, used to obtain the MDP policy, is modified to account for dilemma scenarios and a discussion about the results of both publications.

4.1 Dilemma situations

At each step of its trajectory, the AV should be able to tell whether a situation constitutes a dilemma worthy of moral consideration. This situational classification is necessary to determine if the AV must act according to ethical constraints or simply concrete objectives. Three rules can be defined to represent the AV's responsibilities towards the other road users in its environment and its own passengers and they must be followed at all times. If one or more of these rules are violated across all possible actions, this indicates that the non-dilemma portion of the AV's decision-making cannot cope with the consequences of all possible actions, and thus ethical deliberation is required for the AV to act in an acceptable way. Here harm is defined as the negative consequences suffered by a human after some type of collision with another road user.

AV duties towards other road users

- The lives of the passenger(s) must not be put in harm's way.
- The lives of the road users in the environment must not be put in harm's way.
- Traffic regulations must be followed.

Interactions between road users and the AV are covered by the two first rules. If the geometric representation of two road user (including the AV), during a transition, intercept each other, a collision is considered to have occurred. As it was done in the last chapter, a safety frontier around the AV can be defined to discourage the execution of actions which would remove the possibility of breaking without swerving to avoid an accident. But even so the AV could be exposed to dilemma situation, since the main uncertainty source comes from the behavior of the other road users. The general risk of a dilemma can also be minimized with a conservative behavior from the AV, however it can only go so far if it adopts a precautionary approach since it also must produce a performance capable to justify its existence.

There is a certain antagonism between the two first rules and the third one. One can ask himself if the former should take precedence over the other, so it is certainly desirable that the AV should also be sensitive to the interplay between ethical and legal behavior. In this sense, when there is a conflict in dilemma situations between harm to humans on one hand, and adherence to the traffic code on the other, the standard behavior adopted will be that the avoidance of the former should take precedence. As such, the MDP algorithm must be defined so as to express this priority in ways independent from the influence of the temporal discount rate. However, after the transgression, the AV must return to a 'safe' state, guaranteeing that another collision does not arise as a direct consequence of its original action choice. If this is the case, then for the AV's decision-making algorithm all actions, at the first moment, end in collision, considering the proposed policy calculations that will be presented in the next sections.

Make the AV conform to traffic laws by design is not a subject that is widely addressed in

the literature. Generally, legal conformity presents challenges related to the interpretation of laws which can be vague, admit exceptions, or be internally incoherent; the resolution of all of which may demand some degree of common sense reasoning in order to be solved (Prakken, 2017). Additionally, with adherence to traffic laws comes the need to embed relatively abstract norms, used in laws to map concrete behavior, into an AV, and more broadly, into an autonomous system (Leenes and Lucivero, 2014). Some authors have already attempted to implement some portions of various traffic codes — related to circulation and behavior — into an autonomous vehicle, such as Rizaldi et al., 2017 (German legislation). Categorically, these attempts have been made using logic-based approaches to emulate constraints, representing only the procedural demands which usually compose a traffic code.

Given these considerations, when the duty obliging traffic code adherence is adequately defined, the entirety of the given traffic code does not need to be exhaustively implemented. To simplify the implementation of this evaluation algorithm, a set of logical rules will represent the procedural rules present in the specific scenario simulated. This set of rules should almost always allow the AV to cruise in a lawful manner. Exceptions to the code and the resolution of conflict between rules will not be covered here, the latter being treated as an ethical decision (even if ideally the procedures to solve conflicts between rules present in traffic codes should be used where possible).

For example, in a straight line domain, without pedestrian strips or semaphores, and with a solid double line the following logic rules can be used:

Simplification of traffic rules

- Do not cross over into the opposite lane.
- Do not drive onto the sidewalk.
- Do not surpass the speed limit.

This simplified rules are the same as the ones used in the last chapter to calculate the cost connection to traffic code infringement, in subsection 3.2.3. In a real implementation, where the AV should target full automation (level 4 or more) in specific and generic environments, the actual set of rules will be extended well beyond these three rules and include also the conciliation process between these rules.

4.2 The definition of harm

The purpose of considering 'harm' in ethical deliberation is to measure the risk for AV passengers and other road users involved in a hypothetical collision. For decades, the main

measurement variable to estimate this property has been the difference of velocity between the two implicated road users at the moment of collision (Δv) (Evans, 1994; Jurewicz et al., 2016; Martin and Wu, 2018; Hussain et al., 2019).

Most of the research conducted within the domain of vehicle collisions uses historical accident data to analyze the influence of Δv in collisions. To quantify injury, two metrics are popular: risk of fatality and the Abbreviated Injury Scale (AIS) (MacKenzie et al., 1985). The latter is preferred, since it is important to consider not only fatal collisions but those that can inflict severe damage (referred to as MAIS3+, which indicates that at least one injury in some region of the body is above AIS3, a scale going from 0 to 6). In the European Union, this metric is used as standard to measure road accidents (Weijermars et al., 2018).

All Δv used as thresholds for severe injuries are indicated in table 4.1, along with their source (typically, an injury is considered as ‘severe’ if it indicates a MAIS3+ injury probability of 10%¹). For the pedestrian case, the value was obtained from Kröyer, 2015, which considers severe injury as having an ISS (Injury Severity Score, defined as the squared sum of AIS for the three most severely injured body regions) larger than 9, which is stricter than MAIS3+. Lateral crashes are covered by near side (driver’s side) and far side (passenger’s side). For single vehicle collisions, the same Δv defined for collisions between vehicles is used.

Table 4.1 – Δv threshold used for fatality collisions

Collision type	Contact	Δv value	Taken from
Pedestrian collision	-	6.94 m/s	(Kröyer, 2015)
Vehicle collision	Frontal	7.78 m/s	(Jurewicz et al., 2016)
	Rear	10.56 m/s	(Jurewicz et al., 2016)
	Near side	5.56 m/s	(Jurewicz et al., 2016)
	Far side	6.39 m/s	(Jurewicz et al., 2016)

The data presented in Jurewicz et al., 2016 was collected by the National Highway Traffic Safety Administration (NHTSA), published in Bahouth et al., 2014, considering injuries in the front seat, with a seat-belt, without rollover, with a passenger age ranging from 16 to 55, involving passenger vehicles and heavy vehicles. In Kröger, 2016 the data comes from the Swedish STRADA database, possessing a complete registry of pedestrian collision between 2004 and 2008.

This retrospective analysis of reported accidents has some drawbacks. According to Rosén et al., 2011, the data may be biased, since it is only collected across a small set of countries. Also, in the pedestrian case, age is an important feature (Kröyer, 2015), therefore the age distribution in the studied population plays a role which is unaccounted for in the resulting curve. Under-reporting of non-dilemma cases (Martin and Wu, 2018), estimation of collision velocities (Rosén et al., 2011), negligence of a vehicle’s mass and geometry (Martin and Wu,

¹This is the threshold used by some automotive companies to define an unacceptable risk of collision.

2018; Mizuno and Kajzer, 1999) and the use of different methodologies to evaluate AIS scores (Weijermars et al., 2018) also reduce the precision of such an approach.

Given that the previous method presents problems when applied to specific situations (despite it generalizing relatively well across a population), accounting for contextual information is necessary. The collision interaction between vehicles can be approximated by a damper-spring-mass system, where the initial velocity of each vehicle is projected onto the axis \mathbf{n} (normal to contact plane between both vehicles) and \mathbf{t} (tangential to contact plane). One example of such approach is Pickering et al., 2019, that uses a sprint bi-linear model to estimate the severity of a frontal collision might have if to avoid a pedestrian the AV needs to collide with a static object. The approach that will be considered here should be able to handle any specific situation, being capable to represent multiple collision configurations, from all frontal collisions types to variations of lateral impacts.

The collision can be divided in two phases: the first is the compression, where the deformation of each vehicle happens with the transfer of kinetic energy into to the deformation process, followed by the restitution, when the vehicles plastic deformation settles, and part of the energy transferred to deformation returns as kinetic energy (Smit et al., 2019). Given the multiple possible collision configurations and the objective of having an approximated measure of the severity, an impact the point-mass collision approach will be adopted (Brach and Brach, 2011), with some simplifications, to model the interaction between road users and the AV in case of collision. The following simplification assumptions are adopted:

Collision mechanics simplificatory assumptions

1. No sliding during the collision (when the vehicle skid tangentially the other instead of deform in the normal direction)
2. Dissipation forces are negligible.
3. Crash interface between vehicles is a point.
4. Duration of the crash is infinitesimal.

With the collision mechanics defined, one can start to reason what "harm" should represent. This should refer to a single road user, during a collision. Thus, a plausible evaluation would be $h \propto \vec{J}, c_{vul}$, h representing the road user's harm, J for the impulse applied during collision and c_{vul} for a measure of physical vulnerability, which will be explained in subsection 4.2.2. Considering the point-mass approach, the expression then becomes $h \propto \Delta\vec{p}, c_{vul}$, resulting in the equation 4.1, for a road user k :

$${}^k h(s_t, a_t, s'_t) = {}^k c_{vul} \cdot \left(\|\vec{v}_f - {}^k \vec{v}_i\| \right) \quad (4.1)$$

The collision velocity is calculated using the conservation of linear momentum, that will be explained in subsection 4.2.1. The variable v_f , represents the collision velocity for both road users, k and l . The velocities ${}^l\vec{v}_i$ and ${}^k\vec{v}_i$ are measured at the imminence of impact for k and l .

4.2.1 Δv for collision

To calculate the velocity after collision, v_f , used in equation 4.1, both road users (k and l) are considered as punctual masses and all dissipation forces are neglected. From the original equations of the momentum, given by 4.2 and 4.3 for each road user, which can be brought together in one single equation, 4.4, which represents the conservation of linear momentum. The masses m_k and m_l correspond to the total mass of each road user (if it is a vehicle, then its mass plus the passengers' mass)

$$m_k {}^k\vec{v}_f - m_k {}^k\vec{v}_i = {}^k\mathbf{J} \quad (4.2)$$

$$m_l {}^l\vec{v}_f - m_l {}^l\vec{v}_i = {}^l\mathbf{J} = -{}^k\mathbf{J} \quad (4.3)$$

$$\left(m_k {}^k\vec{v}_f - m_k {}^k\vec{v}_i\right) + \left(m_l {}^l\vec{v}_f - m_l {}^l\vec{v}_i\right) = 0 \quad (4.4)$$

Usually to solve the system of equations presented by equations 4.2 and 4.3 it is necessary to "find" two more equations (six variables and only four equations, two for \hat{x} component and two for the \hat{y} component). Then comes the particular knowledge about the dynamics of collision restitution from the coefficient of restitution e and the impulse ratio μ , that is the relation between the normal and tangential impulse (no relation to the friction coefficient, that frequently is represented by the same letter) (Brach and Brach, 2011).

Given that it is essential for the harm to estimate the danger that a collision might pose and that such collision can be represented roughly by a spring-mass system (dissipation forces are negligible, so the presence of the damper is not necessary), it would be interesting to evaluate the final velocity present at equation 4.1 in the moment of maximal acceleration, which is the most important information to measure injury for passengers/pedestrians. This moment is the end of the compression phase, when the vehicles compose a single structure, right before the restitution takes place and separates them. Thus, this final velocity can be considered equal for both vehicles, giving the following equation:

$$\left(m_k {}^k\vec{v}_i + m_l {}^l\vec{v}_i\right) = (m_k + m_l) \cdot \vec{v}_f \quad (4.5)$$

With equation 4.5 it is finally possible to calculate the final velocity for each road user \vec{v}_f used in the harm equation 4.1, without any particular specificity about the vehicle body properties. For collisions with pedestrians, it is considered that the pedestrian's \vec{v}_f is equal to the AV's \vec{v}_i , which implicitly means that the linear momentum of the pedestrian is negligible in comparison with a vehicle, given the difference in mass, approximately by a factor of 10 and the difference of velocity, that can also reach a factor of 10. Thus, the AV's velocity variation due to the collision can be neglected, and its harm is equal to zero.

This simplification was adopted considering that the most common variables used to predict injury for pedestrians, according to the current research about collisions between pedestrians and vehicles, are the type of vehicle involved (Mizuno and Kajzer, 1999), specially due to the height of bonnet leading edge, parameter that can explain why SUVs are more dangerous to pedestrians than other vehicle types and the mass of the entire vehicle which usually is bigger than the average due to its size (Malczyk et al., 2012; Simms and Wood, 2006), along with the impact velocity. Therefore, this danger that is not captured by the velocity is the focus of another parameter, which accounts for the inherent vulnerability of a road user, together with other parameters. This subject comes into discussion with the definition of the vulnerability constant in equation 4.1.

For collisions with static objects, the same reasoning which was used with vehicle to vehicle collisions is applied, but in this case v_f is equal to zero. In the last chapter, the borders of the simulation environment were defined as static walls (subsection 3.2.3); when the AV leaves the bounds of simulation it considers that the final velocity in a collision (at the end of the respective transition) is equal to zero to evaluate its harm in a direct collision with this wall.

4.2.2 Vulnerability constant

A plethora of studies exist in accidentology addressing collisions between vehicles and pedestrians or only between vehicles. The role of vulnerability constant c_{vul} is to represent the inherent physical vulnerability of each road user during a collision using the information captured by these statistical studies in a way to complete the information given by the velocity variation described in subsection 4.2.1. Lets consider the following scenario: a frontal collision between two vehicles with a Δv equal to $30km/h$ and another between a vehicle and a pedestrian with the same Δv . Lets first disregard the c_{vul} and define the harm as only being the variation of velocity of a road user due to a collision.

According to equation 4.1 the harm for each vehicle (named l and k , with ${}^l\vec{v}_i = (20, 0)$ and ${}^k\vec{v}_i = (-10, 0)$ and equal mass) would be the norm of the difference between the final velocity, which is the mass weighted sum of their velocities (equation 4.8), and their initial velocity, at this moment not using c_{vul} . This value, expressed by equations 4.6 and 4.7 needs to be compared with the same situation happened with the vehicle l but this time with a pedestrian p .

$${}^k h = (\|\vec{v}_f - {}^k \vec{v}_i\|) = \|(5,0) - (-10,0)\| = 15 \quad (4.6)$$

$${}^l h = (\|\vec{v}_f - {}^l \vec{v}_i\|) = \|(5,0) - (20,0)\| = 15 \quad (4.7)$$

$$\vec{v}_f = \frac{m_k {}^k \vec{v}_i + m_l {}^l \vec{v}_i}{(m_k + m_l)} = (5,0) \quad (4.8)$$

What can be observed from the comparison between the equations 4.6 and 4.7 with 4.9 and 4.10 is that for the pedestrian collision the harm would be 25% higher in comparison with the vehicle case.

$${}^p h = (\|\vec{v}_f - {}^p \vec{v}_i\|) = (20,0) \quad (4.9)$$

$${}^l h = 0 \quad (4.10)$$

$$\vec{v}_f = {}^l \vec{v}_i \quad (4.11)$$

One can argue that this conclusion can be interpreted as an evidence that in this case the collision with the pedestrian can be more severe. But two important details are not included in this procedure:

- Multiple essential information are left out of the calculation. For example, the type of vehicle is an important information, giving a hint about the mass and geometry of the vehicle; the configuration of collision also is important, since frontal collisions and lateral collision may present different risks for the passengers.
- The graduation between the harm's value and the probability of an MAIS3+ injury is not straightforward either. It cannot be said, for example, that there is a 25% higher change that the collision with the pedestrian result in such injury than in comparison with the vehicle collision.

The objective of the vulnerability constant c_{vul} is to add these information about the road users and also scale the harm measure to be an accurate representation of the danger coming from the collision, considering all different information. Its determination should come from the Δv curves available in the literature, that connects a certain value of Δv with a probability of an injury MAIS3+. However, given the number of parameters the determination of such constant is rather complex. The following parameters should be considered to define the constant:

Parameters necessary to define the constant of vulnerability

- Type(s) of vehicle(s) involved (minis, SUVs, sedans, etc.).
- Configuration of collision (frontal, oblique, lateral, etc.).
- Homogeneous absolute velocity interval.

Both first and second parameters are self-explanatory. The third refers to the fact that Δv is only a difference of velocities, therefore to have a $\Delta v = 30\text{km/h}$ is different when one of the vehicles have $v_1 = 100\text{km/h}$ and if it have $v_1 = 30\text{km/h}$. This parameter guarantees that accident data relative to fast corridors in cities does not pollute the set, pushing up the probability of injury for a certain Δv .

For each combination of the three parameters presented above, one curve Δv versus probability of MAIS3+ injury would be necessary. If it existed, then, assuming some specific curve form for the vulnerability constant, it could be fitted to give the harm the same proportional increase according to the risk of injury. The necessary offer of graphs do not exist today, all the open databases concerning traffic accidents usually does not comport the velocity before collision nor the type of vehicle involved (the french database does not have neither, while the NHTSA has only the first, both also without the injury measurement based on the AIS scale).

The choice made to generalize all information about the correlation between the physiological impact of an accident into humans, together with the parameters connected with the compatibility between vehicles (in V2V collisions) and collision configuration, using the statistical data of accidents, tries to simplify the procedure of estimating the harm considering all these variables. All different modes of collision could be added into the collision mechanics, but would necessitate a 3D dynamic model of the vehicle to calculate all the forces involved, including a precise model of the interaction between road and tires. And to account the inherent vulnerability of the user would be more difficult, maybe with some relation between acceptable levels of short pulses of acceleration for humans. As said before, an accidentology approach also has drawbacks, but all the data, in the format proposed, does not need to be validated since it comes from real accidents.

As the determination of the initial velocity of an vehicle right before a collision and the aforementioned curves themselves are subject of entire projects and because the necessary databases are not open to public, the vulnerability constant was simplified in the implementation presented at the end of this chapter. As such, the c_{vul} was considered equal to one, given that the velocities used in simulation were no higher than 10m/s , originating Δv s that do not signal a critical difference of risk in the collisions studied later on.

4.3 Ethical deliberation process

Given an hypothetical dilemma situation, the AV must be able to act in an ethical manner, using explainable methods and basing its decision in criteria that is clear and well known. This section offers two different methods to achieve such decision procedure: the ethical valence theory and the ethical optimization. This section ends with an explanation concerning the necessary adaptation into the value iteration to allow the AV to switch from the reward-value optimization to one of these ethical deliberation processes.

4.3.1 Ethical Valence Theory

The ethical valence theory was proposed in Evans et al., 2020 as a method to mitigate the claims that each road user has about the AV's behavior and also about their right of safety in a public space. It considers two variables to take a decision, the ethical valence, which tries to capture an hierarchy of protection based on *a priori* criteria, and the harm, already defined in the precedent section. Then an ethical deliberation procedure, based on these variables, define which road user should be safeguarded against a possible collision.

Valence definition

The purpose of a valence is to represent the degree of social acceptability that is attached to the claims of the road users in the vehicle's environment. In this sense, the claims of certain road users can be more or less 'acceptable' to satisfy via the vehicle's action selection. The valences, in so far as they are rooted in the phenomenal signature of individuals, then track various physical characteristics which are seen to carry social importance: height, age, gender, helmet-wearing-cyclist, or stroller-pushing-adult, all of which are detectable by the object classification algorithms of the AV. Importantly, the determination of the strength of these valences is accomplished through a type of ranking or hierarchisation, which associates a road user's claim with a certain class or category of valence, as shown by table 4.2. In this way, depending on the amount or detail of the valence features under consideration, there can be more or less valence categories.

In this example for instance, two features are used: age and type of road user. The classification was created considering recent studies which suggest that western societies prefer to spare the young and vulnerable (understood in terms of exposure to injury) in AV collisions (Awad et al., 2018). In the case of multiple people, vehicles or agglomeration of pedestrians, the entity that has the larger number of users with a high classification has the preference. Between an AV with passengers C and F and another with C and D, the latter is considered to have a higher valence.

Table 4.2 – Possible valence hierarchy

Feature 1	Feature 2	Classification
Young (0 – 18 years)	Pedestrian	A
Old (65+ years)	Pedestrian	B
Young	Vehicle passenger	C
Old	Vehicle passenger	D
Adult (18 – 65 years)	Pedestrian	E
	Vehicle passenger	F

In cases where the chosen valence features are minimal or simple (such as in the example above) the likelihood that multiple road users will have the same valence, but differing claims, increases. In this sense, there may be certain situations wherein the harm measurement becomes the decisive factor in action selection. In these cases, the vehicle satisfies the strongest claim in its environment, protecting the person whose welfare is most severely impacted, due either to a dangerous context (high velocity difference) or to an inherent vulnerability (detected by the vulnerability constant). This simple maximization of welfare, however, is complicated by the operational moral profile, which specifies the claim mitigation process between those passengers inside the car, and those road users outside of it. To this end, two possible moral profiles can be seen in table 4.3. Risk is considered severe if Δv surpasses the limits defined in table 4.1.

Table 4.3 – Possible moral profiles for an AV

Moral profiles	Definition
Risk averse altruism	Protects the road user with the highest valence as long as AV passenger's risk is not severe.
Threshold egoism	Protects AV passengers as long as risk for other road users with higher valence than the AV is not severe.

None of these profiles perfectly resemble any traditional moral theory, or if anything, resemble various positions along the spectrum of egoistic rationality (Parfit, 1984). This is intentional, as these profiles are designed to capture various degrees of compromise between the claims and valences of the AV's passengers and those of the other agents within the AV's environment. These profiles often reinforce the idea that a certain degree of morally admirable partiality is possible, or perhaps even necessary in AV behavior, in order to best align with user expectations, or to garner user trust (Gerdes and Thornton, 2016; Keeling et al., 2019). The profiles listed in table 4.3 are likewise non-exhaustive and represent somewhat factually opaque renditions of the profile types the Ethical Valence Theory can accommodate. In these versions, the role of the harm calculation is important, as it is the principal factor which informs the various consequences of the AV's actions, due to trade-offs between the passenger(s) claims and those of

the other agents in the vehicle's environment.

Moral deliberation

Once informed by the valences and harms, the AV can deliberate on an action, a step which is crucially guided by the operational moral profile. Each moral profile indicates a unique form of deliberation, as shown in table 4.4. It is perhaps worth restating that the moral profiles—and for that matter, ethical deliberation itself—is only used if the AV detect a possible dilemma situation. Otherwise, concrete, goal-driven planning is operative, using standard decision-making criteria.

Table 4.4 – Optimization procedure based on the moral profile chosen

Moral profiles	Deliberation
Risk averse altruism	Minimize the expected harm of the road user with the highest valence until the AV's collision becomes severe
Threshold egoism	Minimize the expected harm of the AV until the risk to a road user with a higher valence becomes severe

Each profile requires a different implementation. Using the risk-averse altruism case as an example, to deliberate, the AV's state (represented with the same variables as in subsection 3.2.1), environment state (e , which contains the position and velocity of all agents in the environment), highest road user valence (η) and maximum Δv , are the input. The action that should be executed (a_η), is the output. As a first step, all harm measurements for possible actions and the proceeding states (represented by the state space S' , composed by the states reached after one single transition) need to be calculated. Here the decisional horizon is equal to one transition, since the accident will follow immediately afterwards. This is done first by solving equations 4.5 and 4.1. Only one road user is implicated with the AV in a collision. The representation of other road users follow the same procedure used in chapter 3: they have a static behavior and their current velocity is used to predict their movements in the future, with no uncertainty involved.

Algorithm 2: Calculation of all possible harms

```

1 for all  $a_i \in A$  do
2   for all  $s'_i \in S'$  do
3      $\vec{v}_f \leftarrow$  calculate final velocities (equation 4.5)
4      ${}^k h(s_i, s'_i, a_i) \leftarrow$  calculate harm for all road users (including AV, equation 4.1)
5   end
6 end
```

If all possible outcomes have a velocity difference which is larger than Δv (the road user's initial velocity minus the AV's velocity), then the collision is severe and the safety of the AV's passenger is prioritized. In the considered profile, the chosen action minimizes the expected harm for the AV. It should be pointed out that Δv changes according to collision type (as can be seen in table 4.1). The transition probability is used to calculate the expected harm ($h_{exp}(s_i, a_j)$, equation 4.12), which represents a mean harm value for a road user k , given that for one state s_i and action a_j different states s'_i can be reached, and therefore different collisions can happen. The position of all road users and the observation of the AV's state is considered to be perfect (no uncertainty in these measures).

$${}^k h_{exp}(s_i, a_j) = \sum_{s'_i \in \mathcal{S}'} p(s'_i | s_i, a_j) {}^k h(s_i, s'_i, a_j) \quad (4.12)$$

If the set of admissible actions according to Δv , A_η , is not empty, the chosen action minimizes the road user's expected harm with the highest valence for the actions $\in A_\eta$. Otherwise the action that minimize the harm for the road user with the highest valence is chosen; if multiple minimal actions still exist, then the one that minimizes the AV's expected harm is chosen. This process is shown by algorithm 3.

Algorithm 3: Action selection

```

1  $A_\eta \leftarrow$  all actions in  $A$  that ( $\|{}^{RU}\vec{v}_i - {}^{AV}\vec{v}_i\| \leq \Delta v$ )
2 if  $A_\eta = \emptyset$  then
3   |  $a_\eta = \operatorname{argmin}_{a \in A} {}^{AV} h_{exp}(s_i, a_j)$ 
4 else
5   |  $a_c \leftarrow \operatorname{argmin}_{a \in A_\eta} {}^{RU} h_{exp}(s_i, a_j)$ 
6   | if Multiple  $a_c$  exists then
7     |  $a_\eta = \operatorname{argmin}_{a_c} {}^{AV} h_{exp}(s_i, a_c)$ 
8   | else
9     |  $a_\eta = a_c$ 
10  | end
11 end
```

Passing from the AV's harm minimization to the road user's harm minimization may appear to be an extreme position in comparison with other alternatives, such as the possible minimization of both quantities. An infinite number of compromises can be imagined between the AV and road users, however in our examples here both moral profiles oppose each other to maximize the safety of only one road user. For the threshold egoism profile, only the action deliberation process shown by algorithm 3 would change.

4.3.2 Ethical Optimization

The original idea behind the optimization procedures discussed in this section is the adaptation of an ethical reasoning into a mathematical procedure given a specific context. Currently and in the close future the existence of some robot capable of capturing all the nuances of human reasoning and behavior, while being able to reason at all levels, from symbolic to practical scenarios, is highly improbable. So the legwork from theory to the application of an ethically-driven form of decision-making must be partly done by the design of the decision-making model itself.

Three ethical principles were chosen to be applied to the collision dilemma situation. The first discussed will be the Rawlsian contractarianism, based on the theory of justice, proposed by Rawls, 1971. Then, the more familiar theory of utilitarianism will be considered, subsequently followed by egalitarianism, an adaptation of the utilitarian idea but also evaluating the scale of disparity between users. The discussion is concentrated on consequentialism methods, with Rawlsian contractarianism and egalitarianism serving as starting points for reflection about the drawbacks of utilitarianism.

Contractarian optimization

According to Rawls, 1971, justice is an inviolable right of humans, in which the only compromise acceptable in the case of an unequal distribution of rights guaranteed upon people by justice would be the closest possible to equality. The concept of fairness intervenes when two or more persons are interacting, and fairness should determine which procedures and practices can be used in their competition or cooperation, or the conditions under which one individual should recognize another individual's claim without the feeling of being forced or taken advantaged of. These two properties are highly desirable in the context of an AV decision making that might need to make decision in ethical dilemma situations.

From Rawls, 1958, the two principles of justice proposed are: "*..., each person participating in a practice, or affected by it, has an equal right to the most extensive liberty compatible with a like liberty for all; ..., inequalities are arbitrary unless it is reasonable to expect that they will work out for everyone's advantage ...*". More specifically, the expression "*must work out for everyone's advantage*" excludes the justification of disadvantages for the worst of given the advantages offered for the most well of.

Translating the generic context used here to the narrow case of justice as is applied to dilemma decision scenarios, one can say that every road user must be awarded the most extensive right to safety as is compatible with the safety of others, and that differences in safety across road users should benefit those road users who are most at risk. In the imminence of a collision, having the same amount of safety means that ideally the harm should be equally distributed for all involved road users (involved here refers to all road users in reach of the AV that might be

struck by it), if no other action is available that protects the most vulnerable road users.

Practically speaking, starting from the most equal state, when the standard deviation is minimal for an action a that is available to the AV in a dilemma situation (equation 4.13; $h_{exp}(a)$ represents the set ${}^i h_{exp}(a)$, $0 \leq i \leq n$, n being the number of road users in the scene) an hierarchy can be created, from the road user most protected to the least one. Then the largest expected harm, considering this classification, should be minimized, as long as the harm for all other road users involved does not increase (equation 4.14).

$$a_{int} = \underset{a \in A}{\operatorname{argmin}} \sigma_{h_{exp}}(a) \quad (4.13)$$

$$\underset{a}{\operatorname{Min}} \{ {}^i h_{exp}(a) \}, \text{ subject to } {}^j h_{exp}(a) \leq {}^j h_{exp}(a_{int}) \quad (4.14)$$

This optimization model is the one used in de Moura et al., 2020, which results are presented in the later subsection 4.4.2. But this formulation can still be improved to represent more faithfully the ideas of justice proposed by Rawls. The author says that justice is a primitive concept, that arises when morality is applied to the negotiation between the involved parties to define practices that would be acceptable for all parties. It is the sentiment that comes from the enforcement of these practices that configures fairness. If one party holds that some claim is not legitimate, then to them the decision is unfair. Each one can have a definition of what is legitimate and what is not.

If one considers the traffic code as the practice that deals with the opposing claims from each road user, which in turn would assume that what is just is defined by this code, one has an authentic picture of the relations between users in modern days. Walking back from this commitment would certainly create an unjust action (without even discussing the legal implications of such a decision), hence from this observation comes the first constraint: **during the deliberation for an action that will inevitably cause harm, the traffic code must continue to be followed.**

So fairness, in a dilemma scenario, would be characterized as the evaluation that a concerned road user would perform concerning the claim which underpinned an AV's action. Let's say that, in such a situation the right for safety is the main parameter to be considered when an action is selected, since intuitively the claims connected to it are, at least in their majority and assuming an acceptable level of pertinence, legitimate. But it would not be legitimate if to respect some claim from one road user the AV should risk the safety of other users not involved in the traffic situation, which means that **road users not involved in the interactions that generated the dilemma situation cannot be put in harms-way.**

For pedestrians this constraint can be described roughly as "the AV must not invade the

sidewalk as a tactic to decrease the harm in a collision" considering all possible options available, since in doing so it inevitably will put other lives in danger that are not necessarily concerned by the interaction on the road. The general idea expressed by this limitation was also proposed by the guideline 9 in Luetge, 2017. The interpretation of what is legitimate used here is minimalist and tries to capture a common sense argument since this legitimacy determination has an important personal component involved.

One last modification should be done in the previously proposed deliberation procedure. Given the emphasis that inequalities should only be accepted if the worst off is favored, a search for the action that minimizes the greatest expected harm while maintaining the same level of harm for everybody else was proposed. But such a minimization can be pushed even further: **the biggest expected harm should be minimized as long as the expected harm for the other road users do not surpass a danger threshold.** However, such threshold can only be defined comparing the harm value before the accident and its consequences; such a comparison is not available today.

Lets assume that this threshold can somehow be defined, for each type of road user. Then, three scenarios might appear, for all considered actions, and are displayed below. The set of actions considered below is formed by the original available actions minus the ones that violate the traffic code and that brings danger to road users not concerned with the road environment activity.

New contractarian deliberation

1. All road users have an expected harm below the threshold: the AV can execute the action that distributes the harm in the best way possible.
2. Some road users' expected harm surpass the limit and some do not: the action to be selected must
 - (a) Reduce the expected harm of the biggest number of road users possible to below the threshold.
 - (b) The action should not send road users above the limit, i.e. the one that are below the threshold given the most equal action (determined the same way as before, the action that has the smallest standard deviation) should remain below the limit (but its expected harm can be increased).
 - (c) Accumulate the biggest reduction of expected harm for the road users above the danger limit.
3. All road users' expected harm surpass the limit: same behavior from the first case.

The proposed deliberation adheres better to the notion of justice expressed by Rawls than the first proposition, given the road environment context. But some barriers still exist for its

implementation, the most notable being the definition of this danger threshold, possibly related to the MAIS3+ injury probability level. Somehow, this limit helps to define which claim has preference in the mitigation process, together with the value of the expected harm itself. One could suggest an implementation without this *a priori* threshold, calculating the median of all expected harms and setting it to be the limit; in such case every situation would be treated by the second item of the new proposition.

Utilitarian optimization

This approach is rooted in the Greatest Happiness Principle, which states that: "*actions are right in proportion as they tend to promote happiness, wrong as they tend to produce the reverse of happiness. By happiness is intended pleasure, and the absence of pain; by unhappiness, pain, and the privation of pleasure*" Mill, 1859. Happiness, in this definition, refers to the happiness of all people, not only for the individual that is deliberating about an action. Differently from the previous theory discussed, which aimed at defining a just framework to arbitrate conflicts and generate agreeable compromises, utilitarianism has its origins in trying to explain the motivation of individuals to take actions and as a base to legal and social reform at the time, using this theory to define what is morally good or right (Driver, 2014); which, according to the theory, is fundamentally connected to the pursuit of pleasure (Brink, 2018).

This idea of the greatest good for everyone embodies the definition of utility, every decision should be taken according to the maximization of utility, for the common good. Using the second formulation of the stated principle, one can minimize the total amount of expected harm to find the "least wrongful action", as equation 4.15 shows, the definition of the deliberation rule for this criteria is rather straightforward, the minimization of the expected harm.

$$a_{eth} = \underset{a}{\operatorname{argmin}} \sum_{i=0}^n i h_{exp}(a) \quad (4.15)$$

Egalitarian optimization

One of the biggest drawbacks of the utilitarian maximization approach is that everything rests on general maximization, without any concern towards how the utility is shared among the participants. This poses a real problem for the use case discussed here, because the total expected harm can be minimized for some action while condemning some participants to severe injuries instead of allowing a higher total expected harm for a more safe action for everyone. It is also clear that there is a need to consider the distribution of harm so as to avoid extreme situations like the one used previously as an example.

An approach based on egalitarianism can be used to solve the harm distribution problem. Equality discussion often refers to income distribution, which is an important subject for society, but lets consider that the claim for safety due to each of the road users is more urgent in its strength, demanding a stricter representation of equality in dangerous situations. Thus, in a comparison between the two use cases, the latter should have less "tolerance" with possible inequalities.

One re-interpretation of the utility principle adds how well the utility is distributed as another parameter of the goodness of an action, besides the total amount of good produced by it. Such a view of distribution and total quantity as equal measures of goodness is called communal egalitarianism by Broome, 2017. One clear conclusion of this approach is that the general good can be increased without the need to increase the total amount of good. Such a constraint is translated into the harm optimization process by the minimization of the cost function composed by the total expected harm and the variance of such expected harm for all involved road users, as equation 4.16 shows. The index i refers to each individual possibly involved in the accident.

$$a_{eth} = \underset{a}{\operatorname{argmin}} \sum_{i=0}^n \operatorname{var}_{h_{exp}}(i, a) \cdot {}^i h_{exp}(a) \quad (4.16)$$

The variance is used as weight for each expected harm so as to modulate the proportionality of the amount of harm expected for a specific road user and its comparison with the quantity of harm estimated for the other. Another important detail of the proposed criteria is the use of variance and not standard deviation, with amplifies the importance of inequalities into the distribution of harm.

4.3.3 Value iteration for dilemma scenarios

The value iteration procedure explained in subsection 3.3.1 is a typical implementation of a method to solve an MDP. But, after all the discussions about how the AV should account for collision harm and the ethical consequences an action may present, it is clear that such implementation does not support a way to use both methods to calculate the resulting policy. Therefore, some changes in the value maximization need to be made, so as to, when the operation done by equation 3.27 in algorithm 1 is executed, the action corresponding to the dilemma state be the one chosen by one of the deliberation methods discussed previously, not by the reward calculation proposed in chapter 3.

Some states may have actions that cause collisions and actions that do not. In this case those that cause accidents are discarded during the iteration procedure, since it should be the main prerogative of the AV the avoidance of any collision every time that it is possible. Only the safe actions have their value function calculated and consequently, their action considered as the

optimal solution for a given state. If a dilemma situation arises, then or the ethical value theory is used to deliberate on an action, in which case the valence of all the road users involved is needed as an input of the algorithm; or the ethical optimization might be used, to choose an action that optimized the expected harm according to the three optimization measures defined in subsection 4.3.2. Algorithm 4 shows the resulting modified value iteration to be used to solve the MDP.

During the value iteration, the switch between a normal evaluation and a dilemma evaluation happens if the set of actions for a state that do not result in accident, $A_{\text{norm}}^{s_i}$, is empty (line 5). This set is calculated during the estimation of the reward function; it starts having the same content as A but have actions removed if they end up in accident. If all actions ends up in an accident, which is the same as saying that A_{norm} is empty, then the ethical deliberation method chosen must deliberate on an action (line 6) and the value function correspondent to the current state s_i and the chosen action a_{eth} will receive the collision cost. If there is some action that do not cause a collision, then the value $V^{t+1}(s_i, a_j)$ is calculated normally, only using the actions inside the set of admissible actions A_{norm} (line 11).

This abrupt change in the value iteration estimation might raise questions about the assumptions made to guarantee the existence and convergence of the optimal calculation via value iteration. Such assumptions are still valid (stationary and bounded rewards, subsection 3.1.2), since the value adopted is a constant and establishes unequivocally the action to be executed for a state. Intuitively speaking, the only problem that such approach could induce is a periodical switch between both calculation methods, but the convergence theorem also remains valid (theorem 4).

Algorithm 4: Value iteration including dilemma situations**Data:** Spaces S and A , functions R and P , constant γ , environment data e **Result:** Policy $\pi^*(s)$, Optimal Value $V^t(s)$

```

1  $t = 0$ 
2 do
3   for every  $s_i \in S$  do
4      $A_{norm}^{s_i}$  determination
5     if  $A_{norm}^{s_i} = \emptyset$  then
6        $a_{eth} = \text{Ethical\_deliberation}(s_i, A, e, P)$ 
7        $V^{t+1}(s_i, a_{eth}) = c_{col}$ 
8        $A_{select}^{s_i} = \{a_{eth}\}$ 
9     else
10      for every  $a_j \in A_{norm}^{s_i}$  do
11         $V^{t+1}(s_i, a_j) = [R(s_i, a_j, s', e) + \gamma \cdot \sum_{s'_j} P(s'_j | s_i, a) V^t(s'_j)]$ 
12      end
13       $A_{select}^{s_i} = A_{norm}^{s_i}$ 
14    end
15     $V^{t+1}(s_i) = \max_{a \in A_{select}^{s_i}} V^{t+1}(s_i, a_i)$ 
16     $t = t + 1$ 
17  end
18 while  $MSE(V^{t+1} - V^t) > \epsilon$ 
19 for every  $s_i \in S$  do
20   
$$\pi^*(s_i) = \underset{a \in A}{\operatorname{argmax}} V^t(s_i) \tag{4.17}$$

21 end

```

One should also consider the future consequences of these actions that do not provoke an accident immediately. It is possible that the AV choose an action that currently does not cause an accident, but that in the future ends up in a dilemma situation. In such occasions this action should be discarded. More serious than that, when the only actions available for the AV are certain to provoke dilemma situations in the future the AV needs some capacity to compare both sets of actions from different time epochs. This is done by projecting the parameters (at $t + n$) used to deliberate ethically on which action must be chosen towards the present (at t). Figure 4.1 illustrates how the mechanism works.

Red states show that a collision occurred during the transition, and in such a case the AV breaks and stops. State $s_{2,2}$ is critical, since all its actions will cause an accident in all available transitions. When the deliberation is done for $s_{1,0}$, which is not critical *per se*, it realizes that $s_{2,2}$ is. Thus $s_{1,0}$ is considered to be critical and the collision information resulting from the chosen action determined by the ethical deliberation in $s_{2,2}$ is transferred to $s_{1,0}$ to be used in its ethical

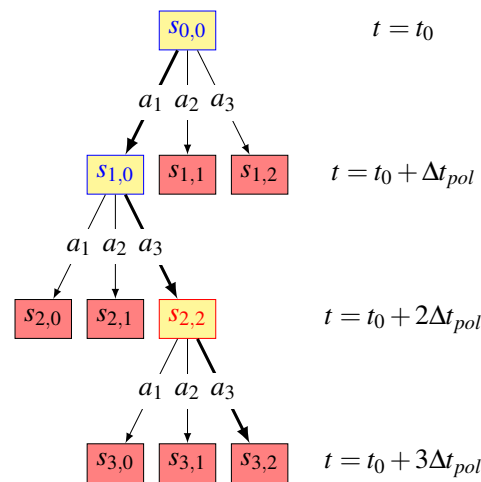


Figure 4.1 – Projection of future collision into previous states

deliberation. The deliberation at $s_{1,0}$ is done with the collision information (expected harm) for a_1 and a_2 at $t + \Delta t_{\text{trans}}$ and the information determined by $s_{2,2}$ at $t + 2\Delta t_{\text{trans}}$ for a_3 . The same procedure is later done for $s_{0,0}$, with its information for a_1 being the expected harm for the action chosen in $s_{1,0}$.

Comments on the effect of the number of people in the ethical deliberation

How the ethical deliberation methods account for the number of persons that participate in the collision while being in the same road user (AV's passengers or many pedestrians close together) is an important detail. In the EVT this is controlled by the moral profile, that uses the expected harm and the valence to choose an action. In the two examples of moral profiles given by table 4.4, the risk averse altruism considers only the road user with the highest valence and each valence inside the AV, with the maximal valence representing the road user and the others having a tiebreak function. The expected harm is calculated for the entire vehicle, no consideration is made concerning differences in passenger placements. This could be done calculating the harm for each of the passengers, but in this case using their placement as another parameter in the calculation of the vulnerability constant.

However, the same is not valid for the ethical optimization approach. In the simulation results that will be presented next, the vehicles (AV and the other vehicle) were assumed to be carrying only one passenger, but let's assume that there are more passengers inside. The correct would be to give each passenger its own harm and that the total harm for the AV in a collision would be the sum of all passengers' harms, which is compatible with the utilitarian philosophy. This could create a disequilibrium in the deliberation because five medium intensity harms could

beat one high harm for a pedestrian for example. Another option would be to keep using only one harm per road user, even if there is multiple persons inside; such hypothesis would allow a better account of the harms when the harm baseline is low, for example when the AV's velocity is low but in high velocities it could favor one individual instead of multiple. Thus, for the contractarian and egalitarian the most coherent choice is to consider one harm per human being in the environment but create the constraint that when the action that concerns multiple road users are evaluated all harms must be evaluated, instead of the one by one basis from the contractarian; the egalitarian would function the same way as the utilitarian, the sum of all expected harms, with the variance as weights.

4.4 Results and discussion

4.4.1 Using EVT as ethical deliberation

In this publication an example of application of the ethical valence theory was given, starting from a dilemma scenario, to compare the results that each of the deliberation methods (risk-averse altruism and threshold egoism) would produce. The situation is illustrated by figure 4.2, where the AV is represented by the red vehicle, the yellow being a vehicle with driver and the person in the middle of the road a pedestrian. Only three outcomes are possible, depending on the action that the AV takes: collision with the wall if it turns right, collision with the pedestrian if it continues with the same direction and frontal collision with the other vehicle if it turns left. The direction is considered the only variable since the acceleration for this case is set to be negative.

The transition probability used to calculate the expected harm is the same proposed in the last chapter, subsection 3.2.2. Only the changes in direction carry a probability of transition, with the structure 0.1-0.8-0.1 for the action that continues in the same direction and 0.9-0.1 for the two extremity actions. For the vehicle's position prediction the same model used previously was adopted here, a single-track kinematic non-holonomic model without slippage (figure 3.4), with $\Delta t_{\text{trans}} = 0.5\text{s}$.

To calculate the vulnerability constant, c_{vul} , the data available in Kröyer, 2015 and Jurewicz et al., 2016 are used in equation 4.18, with $\text{Prob}_{\text{MAIS3+}}(\Delta v)$ being the probability of MAIS3+ injury given a Δv , difference of initial velocities before the collision. Since the interval of velocities considered here is small, this curve was chosen to demonstrate how the idea of c_{vul} should work.

$$c_{vul} = \frac{1}{1 - \text{Prob}_{\text{MAIS3+}}(\Delta v)} \quad (4.18)$$

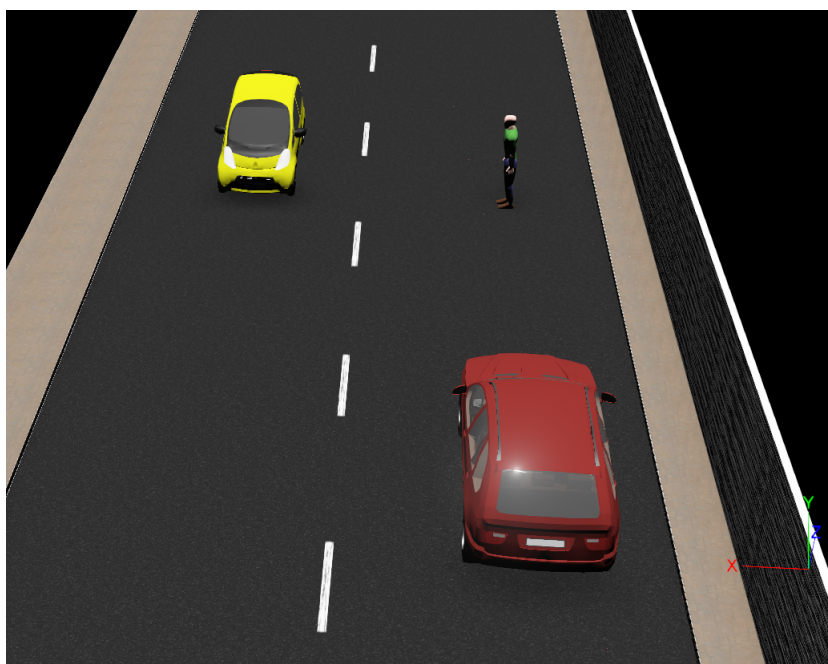


Figure 4.2 – Possible dilemma situation

Figure 4.3 shows the collision simulation with all the road users’ initial states in a situation hereby defined as *situation 1*.

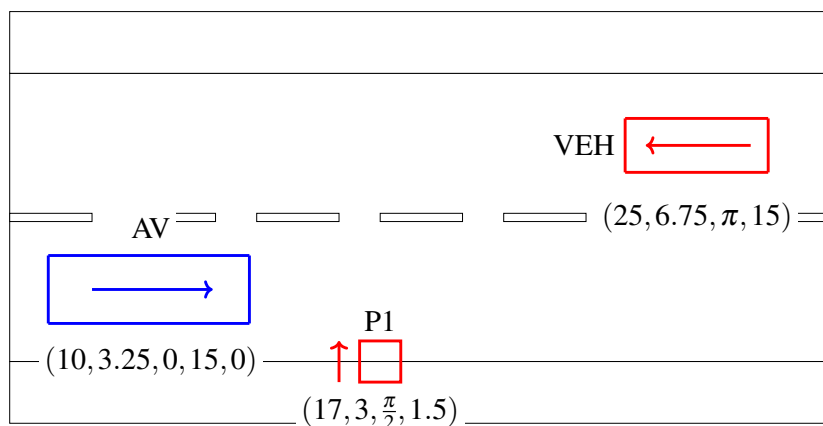


Figure 4.3 – Initial simulation setup (other road users’ configuration is represented by (x, y, θ, v) , not in scale)

Table 4.5 shows the preference order, given the valences for each road user in figure 4.4.

In situation 1, Δv are equal to 23.1 m/s for the AV-vehicle (frontal collision), 14.1 m/s for

Table 4.5 – Valence hierarchy

Road User	Valences	Classification
AV	C, F, F	3°
Vehicle	C, D	2°
Pedestrian	A	1°

the AV-pedestrian (pedestrian collision) and 14.2 m/s for AV-wall (frontal collision). Comparing these values with the limits established in table 4.1, one can conclude that all actions pose a serious risk for the AV and all other road users. Following the risk-averse altruism profile would entail choosing to run over the pedestrian, since the AV must be prioritized (Δv is above the limit, therefore the AV's harm is minimized, selecting the red cell in table 4.6; such a procedure was seen in algorithm 3). Table 4.6 shows the harm and expected harm calculated for the AV considering each possible collision.

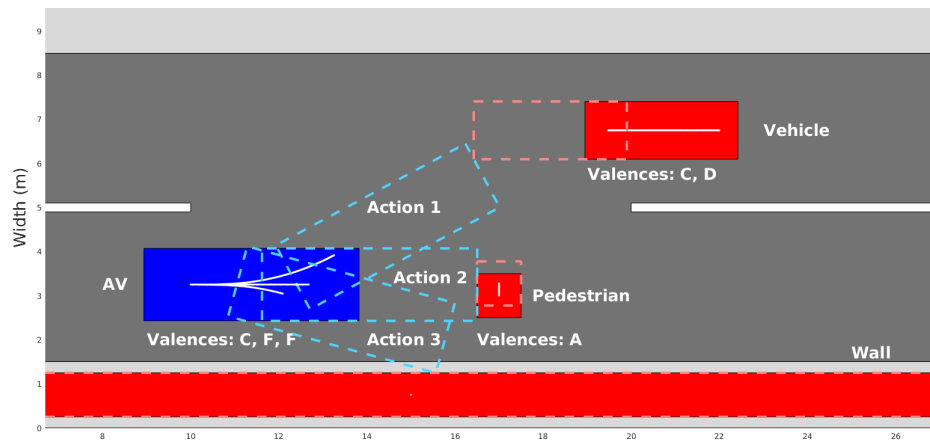


Figure 4.4 – Collision simulation for situation 1

If the AV was configured to have threshold egoism as its operational moral profile, the choice would be to collide with the wall, since the pedestrian's and vehicle's valences are higher, according to the table 4.5 (both Δv are above the limit, thus the road users with valences higher than the AV have their expected harm minimized, resulting in the blue cell at table 4.7). Table 4.7 presents in its first column the nominal road user's harm, while in the second and third columns the vehicle's expected harm and the pedestrian's expected harm, obtained using the transition probability. Since the wall is a static object, its harm and expected harm is zero (only human safety is considered; historical, cultural or affective value to a static object like a tree or a monument are disregarded).

Figure 4.5 shows *situation 2*, where the initial AV's state is $(10, 3.25, 0, 7.5, 0)$ and the position and velocity of the other vehicle also are changed. The new consequences of each action are displayed by figure 4.6, changing the collision with the other vehicle from frontal to lateral.

Table 4.6 – AV's harm for each possible collision for situation 1

	AV's harm	AV's exp. harm
Veh. col.	8.77	7.02
Ped. col.	0	2.46
Wall col.	15.80	12.64

Table 4.7 – Road users's harm for each possible collision for situation 1

	Road user's h	Vehicle's h_{exp}	Pedestrian's h_{exp}
Veh. col.	16.80	15.12	1.57
Ped. col.	15.71	1.68	12.57
Wall col.	0	0	1.57

There is a change in collision severity, given that the velocity differences Δv would be 14.87 m/s, 5.63 m/s and 6.07 m/s for vehicle to vehicle, to pedestrian and to wall collision, meaning that the two last ones do not surpass the severe threshold.

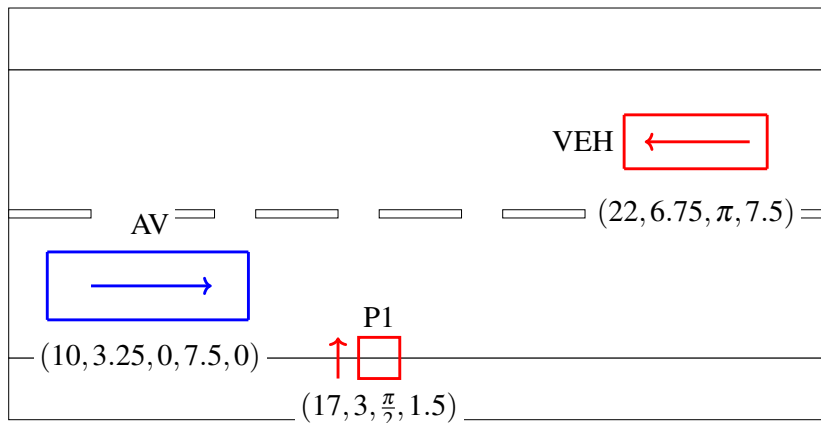


Figure 4.5 – Second simulation setup (other road users' configuration is represented by (x, y, θ, v) , not in scale)

Using risk-averse altruism as the operative moral profile results in the wall collision action being executed (the road user that has the highest valence has its expected harm minimized), resulting in the action represented by the red cell in table 4.8 (in here, since two action have the same expected harm the second highest valence had its exp. harm minimized, which is the other vehicle) and for the threshold egoism the chosen action would be collision with the pedestrian (In this case the AV's expected harm would be minimized), resulting in the blue cell at table 4.9. Tables 4.8 and 4.9 are analogous to tables 4.6 and 4.7, respectively.

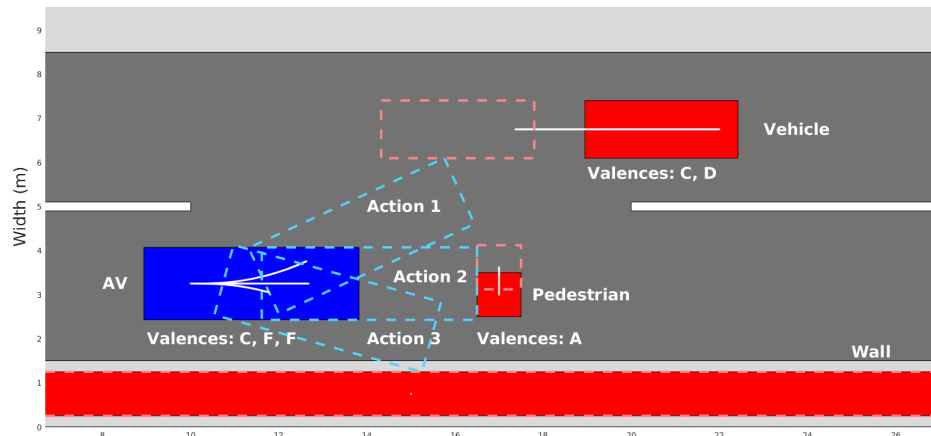


Figure 4.6 – Collision simulation for situation 2

Table 4.8 – AV's harm for each possible collision for situation 2

	AV's harm	AV's exp. harm
Veh. col.	5.10	4.08
Ped. col.	0	1.12
Wall col.	6.07	4.86

Table 4.9 – Road users's harm for each possible collision for situation 2

	Road user's h	Vehicle's h_{exp}	Pedestrian's h_{exp}
Veh. col.	10.85	9.76	0.56
Ped. col.	5.63	1.08	4.51
Wall col.	0	0	0.56

4.4.2 Using the Ethical Optimization profiles to deliberate

Here the simulation presented in subsection 3.3.3 are slightly modified to produce a dilemma situation, forcing the AV to use the ethical optimization methods proposed previously. The ethical dilemma scenario is created moving the other vehicle abruptly in just one time step (also, the x coordinate of P1 and P2 are changed to 20 and 22.5, respectively). At this point the AV must deliberate about an action. The velocity difference between each road user (Δv) is not significant enough to be identified in the fatality probability versus Δv graphic given by Hussain et al., 2019 and Richards, 2010, thus the constant c_{vul} used for the collisions will be equal to one.

As it can be seen in figure 4.7, the result for all cases is a collision with the pedestrian. But each policy chooses a different action, which imposes different collision consequences. Take the contractarian policy, which has its results displayed in table 4.10: instead of decreasing the velocity right before the collision (actions 1 through 5), it chooses to maintain it ($v_{AV} = 4m/s$) to escape a collision with the other vehicle (action a_5). The AV ends up colliding with the first pedestrian (in the next calculated policy) with ($v_{AV} = 4m/s$).

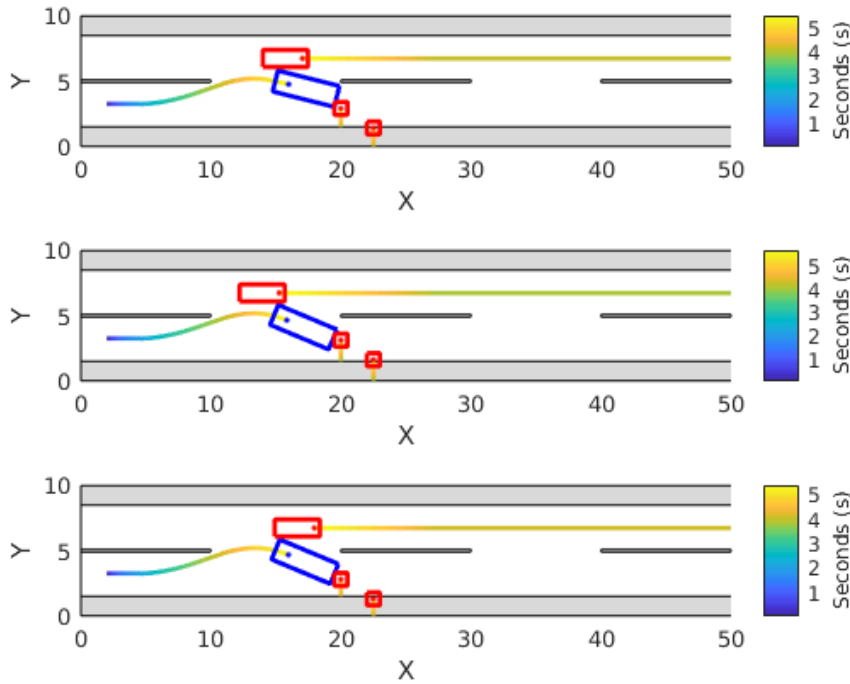


Figure 4.7 – Trajectories for contractarian, utilitarian and egalitarian methods

Table 4.10 – Expected harms and $\sigma_{h_{exp}}$ for contractarian policy

Action ($m/s^2, \circ/s$)	AV	P1	VEH	P2	$\sigma_{h_{exp}}$
a_0 (-1, -40)	3.64	0	6.94	0	2.89
a_1 (-1, -20)	3.64	0	6.94	0	2.89
a_2 (-1, 0)	3.61	0	6.90	0	2.87
a_3 (-1, 20)	3.54	0	6.77	0	2.82
a_4 (-1, 40)	3.49	0	6.76	0	2.81
a_5 (0, -40)	0.383	4.38	0.730	4.00	1.83

For the utilitarian policy, (second plot of figure 4.7), action a_0 is chosen because it minimizes the total amount of harm, as can be seen in table 4.11. It decreases the velocity to $v_{AV} = 3.5m/s$ and then increases to $v_{AV} = 4m/s$ so as to escape the other vehicle. The final collision velocity, in the next calculated policy is $v_{AV} = 3.1m/s$.

Table 4.11 – Expected harms and Σh_{exp} for utilitarian policy

Action ($m/s^2, \circ/s$)	AV	P1	VEH	P2	Σh_{exp}
a_0 (-1, -40)	0.364	3.90	0.694	0	4.96
a_1 (-1, -20)	3.28	0.433	6.25	0	9.96
a_2 (-1, 0)	3.63	0	6.93	0	10.57
a_3 (-1, 20)	3.58	0	6.83	0	10.40
a_4 (-1, 40)	3.55	0	6.81	0	10.36

The egalitarian policy, trying to achieve a compromise between total expected harm minimization and dispersion minimization, actually increases the velocity right after the ethical deliberation, and ends up colliding with the pedestrian with $v_{AV} = 4.5m/s$. Action a_{10} predicts that both pedestrians will be struck by the AV. This policy, in this example, produces more negative consequences than the other two, but maybe with another formulation, one that calibrates more the importance of distributing correctly the risks of a collision without allowing the harm be concentrated in one user, while the other share more of the harm, but only slightly. This method can represent a compromise between a contractarian and a utilitarian policy in such formulation.

Table 4.12 – Expected harms and total cost for egalitarian policy

Action ($m/s^2, \circ/s$)	AV	P1	VEH	P2	Total
a_0 (-1, -40)	3.77	3.34	7.20	0	95.06
a_1 (-1, -20)	3.76	3.05	7.18	0	98.07
a_5 (0, -40)	0.385	4.69	0.735	0	50.00
a_6 (0, -20)	3.46	2.73	6.61	0	77.50
a_{10} (1, -40)	0.431	4.91	0.824	4.60	46.05

4.5 Conclusion

Two ethical deliberation methods were proposed in this chapter. A formal definition for an ethical dilemma situation in the automated vehicle context was given in the first section, to afterwards proceed to the definition of the harm in a collision, including its main parameters, the difference of velocity due to the collision and the constant of vulnerability. The harm is then used by the EVT or by one of the ethical optimization methods to deliberate on an action. From this point a small modification for the value iteration algorithm that solves the MDP model from last chapter is proposed, allowing the AV to switch between deliberation procedures when an ethical situation is detected.

The two deliberation methods were then tested in two different scenarios. The EVT was used in an hypothetical situation where the AV was to decide if it should collide with the vehicle in the other lane, the pedestrian right in front of with the wall in the right. With a higher AV velocity used in situation 1 the profiles mandated the protection of the AV for the risk-averse altruism, selecting the collision with the pedestrian; and the road user with the highest valence, the pedestrian, for the threshold egoist profile and therefore selecting a collision with the wall. In situation 2, when the AV's velocity was decreased to $7.5m/s$ the situation inverted concerning the collision with the pedestrian and the wall, both of which was not considered severe anymore.

All of the profiles for the ethical optimization were tested in the same scenario from chapter 3, using the MDP model with the modified value iteration. From the three each one chose the collision with the pedestrians, but with different end velocities, with the utilitarian profile having the lowest one. Given the end velocity from all three instances, around $4m/s$ this was the correct choice, not generating severe injuries for the pedestrian, due to the rather low velocity, and avoiding a frontal collision with the vehicle in the other lane, that had a speed of $10m/s$.

Chapter 5

Road User Intent Prediction

Contents

5.1 Broadening the scope	108
5.2 Intent estimation	109
5.2.1 Simulated Environment	109
5.2.2 Agent's decision model	112
5.2.3 Estimating other agents' intent	119
5.3 Incomplete game model	125
5.3.1 Nash equilibrium	125
5.3.2 Harsanyi's Bayesian game	126
5.3.3 Decision making procedure	129
5.4 Simulation results	133
5.4.1 Multi-agent simulation	134
5.5 Final remarks	138

In the previous chapters the proposed AV decision-making procedure considered the behavior of other road users as constant throughout the prediction, as if no interaction happened inside the environment. This kind of assumption can create a planning procedure that is not capable to capture all the variability present in each situation, ignoring an important, if not essential, source of uncertainty. This chapter focuses on correcting this issue using three algorithms: the first one to determine the behavior presented by other road users from pre-defined profiles, which are then used by the AV to estimate the most probable behaviors; and finally, the AV deliberates on which behavior profile it should adopt while at the same time trying to increase its resilience to errors in the prediction of the other road users' behaviors.

5.1 Broadening the scope

In an ideal world the AV would not encounter any dilemma situations. Given the abundant number of sensors one would assume to be embedded in the vehicle and scattered around cities and highways and fast, reliable and low-latency communication with other road users and the cloud combined with enough processing power to execute all necessary operations, *a priori* passengers can be safely driven anywhere, anytime. However, this is a naive assumption since the quantity of hardware in a vehicle will always be minimized by car manufacturers due to cost concerns, there will always be some imprecision in sensors measurements and also due to sheer number of possible behaviors that might be originated by the inevitable interaction between manual vehicles (or vehicles that do not have any type of communication capability, i.e., no V2X communication capability), automated ones and vulnerable road users, at least in urban areas. Considering the universe of difficulties that the AV can face, three categories of errors can be defined:

Possible errors in automated driving

- Material error: due to failure in sensors, processing unit or actuators of the vehicle.
- Design error: represent mostly errors in the software components of the vehicle.
- Interaction error: unexpected behavior and imprecise measurement from other road users can create situation that are not accounted for.

The first type of error refers to problems occasioned by the failure of hardware during the automated driving task. This risk can be eliminated, and probably will be, with the adoption of robustness standards and fail-safe strategies to take the vehicle to a safe configuration in such occasions. Some inspiration can certainly be drawn from the aeronautical industry, that for decades deals with this type of errors. The critical situation studied in (de Moura et al., 2020) and discussed at subsection 4.4.2 was inspired by this type of error since during the simulation an abrupt change in the position of the other vehicle is considered to be caused by a detection error due to a sensor failure.

Design errors are a real problem in any software component, and even more in complex systems. This category encompass the errors that originate from bugs and invalid assumptions made during the design and implementation of the entire automated system for an AV. Such source of problems can also be mitigated with the advancement of programming practices, verification and validation procedures for the developed software and a robust homologation process, which, as discussed in chapter 1, is not yet set in stone.

This chapter focuses on the third problem: the capacity to predict and correctly account the behavior of other road users and their interaction with each other. First, the AV must be capable to predict all possible behaviors of other road users, given the environment constraints that may

exist, which can be different for each road user, and the influence of interactions between road users in the selection of these behaviors; and second, it must be able to estimate the probability that a certain behavior will be adopted for a road user. It is based on these two elements that the AV's decision-making must work to produce a safe behavior or at least be able to plan beforehand its actions in possible dilemma situations that might appear.

Next section will present the new configuration for the simulation, assuming a multi-agent formulation to enable the study of the possible interactions between agents. The decision-making procedure executed independently by each road user will be detailed, followed by the AV's estimation procedure of each decision taken by other road users. Afterwards, an additional procedure to increase the robustness of the AV's decision is proposed.

5.2 Intent estimation

Given that the scope of the application was enlarged from the original AV's decision-making proposed in earlier chapters to also include the other road users' behavior prediction and the interaction between all agents¹ in the environment, a new simulation scenario is necessary, where each road user acts and reacts independently (subsection 5.2.1). To do so, every single one of them (it will be considered only vehicles and pedestrians) need its own behavior selection procedure (subsection 5.2.2). And from the AV's perspective, it should be able to measure the probability that each road user is acting according to the AV's expectation (subsection 5.2.3).

5.2.1 Simulated Environment

Figure 5.1 represents the new environment to be simulated. An intersection is one of the most challenging environments for an AV, because if it does not have any semaphores (which is the case here) the negotiation between road users might depend on non-verbal communication or movement cues that are not immediately distinguishable by a machine. Each leg of the T-intersection has 50 meters of length, with an intersection in the form of a square with a 10x10 meters of surface. A manual driver is represented by the blue vehicle, that might turn right² or continue straight; the AV is represented by the red vehicle, that might turn left or right; and finally the pedestrian can cross in any of the two pedestrian strips in the scene.

Each road user behavior is determined by its intent, which is a variable that defines what the road user wants to accomplish in the environment. The intent, c_i , is composed by two components (g_i, f_i) , where g_i represents the goal that the road user has (for example, the vehicle

¹From now on agents refers to all the other road users and the AV.

²When the turn maneuver is mentioned the frontal direction of the vehicle in question defines the left and right sides.

might want to continue straight in the intersection) and f_i represents the qualitative behavior that a road user might produce during an interaction with another road user. For the vehicles in figure 5.1 for example, this represents if the intersection priority, as defined by the traffic code, will be respected, while for the pedestrian it signals if, in a situations where it does not have the priority, it will nevertheless cross the road.

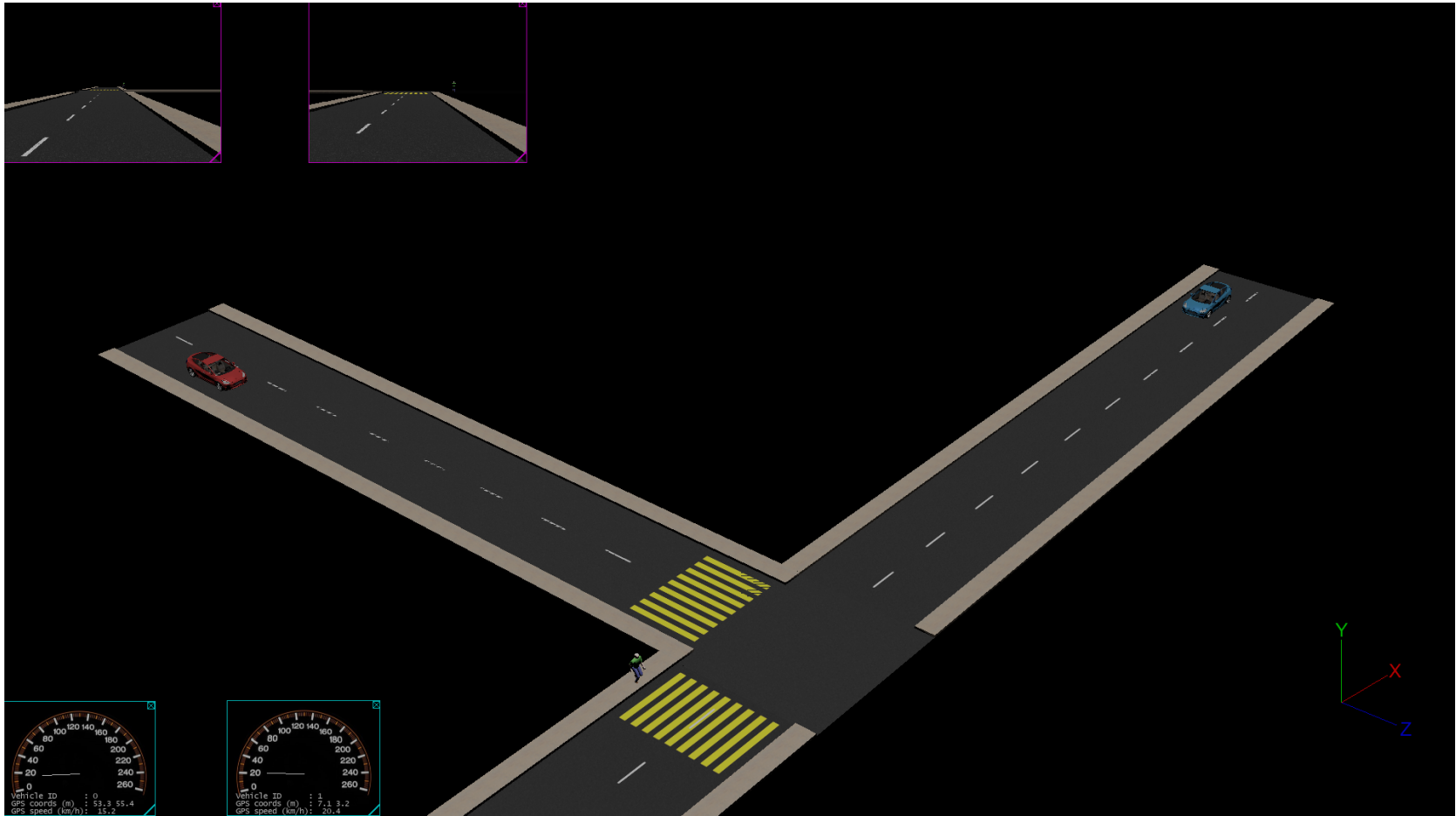


Figure 5.1 – New simulation environment

5.2.2 Agent's decision model

Two different types of agents coexist with the AV in the simulation: vehicles (representing manually driven cars) and pedestrians. To implement an independent decision-making inside each pedestrian and vehicle, a decision tree will be used. This component tries to emulate the behavior of a road user if it follows its intent, according to a trajectory (g_i is connected to a pre-defined trajectory) and to some key behavior connected to a salient situation (f_i indicates generically what will be done). All road users are represented by position, direction and velocity, $s = [x_t, y_t, \theta_t, v_t]$, at some time epoch t .

The intent of each agent in the environment is translated into its longitudinal and lateral control. The goal determines the path³ to be followed, for the pedestrian, which has a predefined trajectory, and for the vehicle, which always follows the middle-lane line. The reference path determines the lateral control of each road user. As for the longitudinal control, it is determined by the (or absence of) interaction with other road users. According to such interaction, one strategy might be adopted as a way to accomplish the road agent's intent. The box below shows all defined strategies for each of the two possible agent types (the strategies are explained in detail at the appendix A).

Available strategies for vehicle and pedestrian	
<ul style="list-style-type: none"> • Vehicle – Cruise – Caution – Follow – Stop – Emergency stop – Swerve 	<ul style="list-style-type: none"> • Pedestrian – Walk – Stop – Run

Starting with the vehicle's strategies, when cruise is the chosen strategy, the vehicle will maintain its speed or it will continue to accelerate until it reaches the speed limit (with constant acceleration). Caution represents the necessary decrease of speed due to the proximity of some other agent. Follow is reproduced by copying the velocity of another agent in front. Stop is used to reduce the velocity and eventually stop using a reasonable braking, while emergency stop uses the maximum deceleration possible and stops in the shortest distance possible. Finally, swerve reduces the velocity and also updates the trajectory proposed by g_i to avoid a collision.

Walk represented the use of a constant speed for the pedestrian, while stop is the immediate reduction of the velocity to zero. Run simply multiplies the speed of the pedestrian in question

³Path is defined in this context as a sequence of waypoints that the agent must pass.

by a factor of two.

Each strategy for a road user is determined from a pairwise comparison between the agent in question and all others in the environment, as illustrated by the algorithm 5. Firstly, the trajectory of all agent is determined during a certain prediction horizon, defined by the variable N_t (line 1), assuming a default strategy, which for vehicle is cruise and walk for pedestrians. As for the motion models for vehicles and pedestrians, the former uses same one used in previous chapters, represented by equation 3.10, while the latter uses a simple point-mass model. Considering these predicted states, the road user in question, indexed by i , is "compared" (how this comparison is done will be explained shortly) to every other road user, indexed by j , depending on the type of each one (lines 6, 8 and 14).

The output of each interaction is the predicted strategy for the agent i in respect to agent j . If there are n road users in the scene, then at the end of all comparisons for agent i there will be $n - 1$ possible strategies (originated from lines 6, 8 or 14). From this set the final strategy ρ_i is calculated simply by checking which $\rho_{i,j}$ has the higher priority, which grows from top to the bottom, using the previous box as reference. Thus, for the vehicle cruise has the lowest priority and swerve has the highest; for the pedestrian walk has the lowest and run the highest.

All road users have access to some shared information, that are the environment structure (dimensions of roads and localization of each structure in the global map), the position of each road user and the decision procedure in itself, and exclusively for the other road users, the intent of each agent. The AV does not know the others' intent, only its own. Such choice is justified with the assumption that the driver and the pedestrian can subjectively estimate the intention of each other, and to model this subjective process is not the main objective in this chapter. What the methods presented in this section and in the next ones try to achieve is to reproduce the behavior of real road users in specific situations, to then enable the AV to identify such behaviors, not necessarily to emulate completely this subjective process of intent estimation.

Next, the comparisons between predicted trajectories for pedestrians and vehicles will be explained.

Algorithm 5: Decision procedure for agent i

Input: Environment state $s^t = \{s_i^t\}$, $i \in [0, N_t]$ time epoch t
Output: Strategy set according to i , $\{\rho_j\}$, $j \in [0, N_t]$ for t

```

1  $s = \{s^t, s^{t+1}, \dots, s^{t+N_t}\} = \text{agents\_prediction}(s^t)$ 
2 for every agent  $s_i$ ,  $i \in [0, N_t]$  do
3   if agent  $i$  is a vehicle then
4     for every other agent  $j$ ,  $j \in [0, N_t] - i$  do
5       if agent  $j$  is a vehicle then
6          $\rho_{i,j} = \text{veh\_decision}(\{s_i^t\}, \{s_j^t\})$ 
7       else
8          $\rho_{i,j} = \text{ped\_decision}(\{s_i^t\}, \{s_j^t\})$ 
9       end
10    end
11  else
12    for every other agent  $j$ ,  $j \in [0, N_t] - i$  do
13      if agent  $j$  is a vehicle then
14         $\rho_{i,j} = \text{veh\_decision}(\{s_i^t\}, \{s_j^t\})$ 
15      end
16    end
17  end
18   $\rho_i = \text{strategy\_calculation}(\{\rho_{i,j}\})$ 
19 end

```

Vehicle model

For a road user that is a vehicle, to compare its trajectory to someone else's, as done in lines 6 and 14, four logical variables are calculated: if a collision happens, if one vehicle is too close to the other, if there is an intersection ahead and who has priority over the other. Collisions and proximity are checked using the mechanisms proposed in chapter 3, section 3.2, which are the idea of frontiers for the proximity detection and the hyperplane separation theorem for the collision detection. To determine the location of each road user inside the simulated environment, a graph-like structure is used to construct the road environment, which is represented by figure 5.2. According to the position and the geometrical dimensions of the vehicle, it is placed in one of the nodes of the road network graph, which automatically determines also the possible next nodes for vehicles.

With the vehicles placed in one of the nodes, their predicted trajectory can be projected into the graph. Each straight road has a reference lane and an opposite one; it is adopted that the reference lane is the one that goes from the extremity not connected to the intersection, which makes the opposite lane definition straightforward. The intersection has a more complex set of possible destinations.

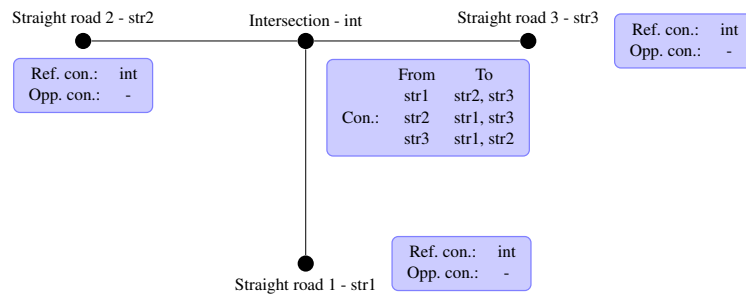


Figure 5.2 – Road environment graph structure

The priority is calculated according to where the vehicle is. For example, if two vehicles are in *str1* and *str2*, two roads in the graph-like structure in figure 5.2, the priority to access the intersection node is given to the one in the right side, considering that the right is measured as the smallest angle between them. If one of them is already inside the intersection it automatically has the priority. It was also defined that the pedestrian has priority over the vehicle if both are in the same node⁴.

The comparison between agents if *i* is a vehicle is displayed by algorithm 6. For each time epoch predicted (from t to $t + N_t$) the proximity is verified; if they are close to each other, *j* might be in front of *i*, in which case the latter should follow the former (line 4); or both vehicles might collide without being on the same lane. If so, the collision time and the necessary strategy to avoid the collision are registered inside the variable **colResult** (line 7). Then the priority is determined in two specific cases, if *j* is a vehicle and both are in the imminence of entering a intersection (line 15) or if *j* is a pedestrian (line 17). The final strategy is calculated by the function in line 19, which uses a decision tree, detailed in figure 5.3, to select the correct strategy.

All leafs in the tree have the final strategies according to the four variables calculated in algorithm 6, which is case for cruise, caution and stop when there is no collision; if there is a collision, then two specific labels are used: try to stop and dynamic stop. The first one refers to the adoption of the stop strategy if there is enough space and a reasonable deceleration is sufficient to do it and if not the stop degenerates to the emergency stop strategy. Similarly, the dynamic stop checks by prediction what the vehicle can do to avoid the accident, if stop is enough, which can also degenerate to a emergency stop or if a swerve strategy is necessary. All these tests are made by the *checkCollision* function in line 7.

⁴A different approach could have been taken, for example defining that the pedestrian has the priority only inside a pedestrian strip, while the vehicle has the priority if the pedestrian is crossing outside the strip.

Algorithm 6: Comparison if i is a vehicle

Input: Agent i and j trajectories ($\{s_i\}, \{s_j\}$) in $[t, t + N_t]$, starting time t_0
Output: Calculated strategy for i , ρ_i

```

1 for each time epoch in  $\{s_i\}$  and  $\{s_j\}$  do
2   if  $checkProximity(s_i^t, s_j^t) \ \&\& \ type(j) \neq pedestrian$  then
3     isClose = true
4     if  $shouldFollow(s_i^t, s_j^t, road\_graph)$  then
5       following = true
6     end
7     if  $checkCollision(s_i^t, s_j^t, colResult)$  then
8       if  $!following$  then
9          $\rho_i = colResult$ 
10        Break
11      end
12    end
13  end
14 end
15 if  $localization(s_i^t) \neq intersection \ \&\& \ type(j) \neq pedestrian$  then
16   hasPriority =  $checkV2V\_Priority(i, j)$ 
17 else if  $type(j) == pedestrian$  then
18   hasPriority =  $checkV2P\_Priority(s_i^t, s_j^t, road\_graph)$ 
19  $\rho_i = decideStrategy(isClose, following, colResult, hasPriority)$ 

```

The strategy selection is dependent from the four logical variables calculated in the vehicle comparison (algorithm 6): proximity, collision, placement and priority. This last one can be modified by the second element of the intent, f_i , to create an unexpected behavior. For example, if one vehicle should yield in an intersection its f_i can say that it should enter it anyway, independently from the other user. The variable f_i has the function to reproduce irresponsible behavior in the simulation, and the challenge from the AV's point of view is to anticipate such behavior using the cues from the other vehicle's movement. How this variable affects the decision tree of a road user will be explained with an application in subsection 5.4.1.

Longitudinal control is defined by the strategy chosen while the lateral control is guided by the middle-lane line. To calculate the final control for vehicles, given g_i and the current position the controller shown in equation 5.1 is used. It is the same controller proposed by (Hoffmann et al., 2007) and used in (Thrun et al., 2006), but here the dynamic terms were neglected. It tries to correct the direction offset and the lateral distance between the desired trajectory and the current one and adjusts the steering angle to make the current direction intersects the tangent line from the desired trajectory at k/v units of distance (Snider, 2009).

$$\phi_{t+1} = (\theta_{traj} - \theta_t) + \text{atan} \left(\frac{k \cdot d_{lat}}{v_{long}} \right) \quad (5.1)$$

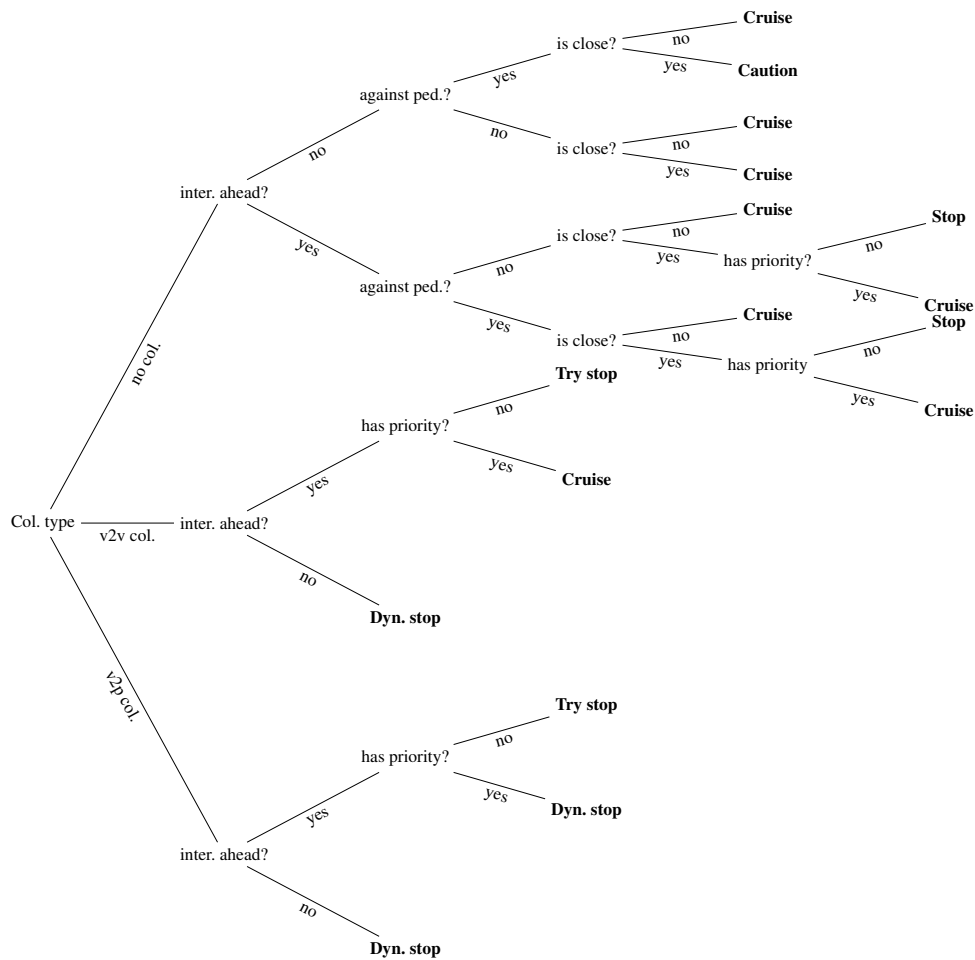


Figure 5.3 – Decision tree for vehicles

Pedestrian model

Pedestrians also have a method to compare the vehicle’s behavior with its own. It is considered unnecessary to compare the behavior of two pedestrians, thus it is assumed that no interaction exclusively between pedestrians cause a change in behavior, leaving only the comparison with other vehicles to be done. The pedestrian strategy is obtained using the algorithm 7 but this time using the current road location of the pedestrian instead of the logical variable related to the existence of an intersection ahead, as is done for vehicles. All trajectories considered for the pedestrian involves crossing the street according to the environment proposed in figure 5.1.

Given that the agent *i* is a pedestrian, its priority is calculated at line 12 of algorithm 7, in

the same manner that it is calculated when i is a vehicle and j a pedestrian. The parameter f_i also controls this priority decision, where it can overturn the *yield* if necessary.

Algorithm 7: Comparison if i is a pedestrian

Input: Agent i and j trajectories ($\{s_i\}$, $\{s_j\}$) in $[t, t + N_t]$, starting time t_0
Output: Calculated strategy for i , ρ_i

```

1 for each time epoch in  $\{s_i\}$  and  $\{s_j\}$  do
2   if  $checkProximity(s_i^t, s_j^t) \ \&\& \ type(j) \neq \text{pedestrian}$  then
3     isClose = true
4     if  $checkCollision(s_i^t, s_j^t, colResult)$  then
5        $\rho_i = colResult$ 
6       Break
7     end
8   end
9 end
10 placement = find_Place( $s_i^t$ )
11 if placement  $\neq$  sidewalk then
12   hasPriority = checkP2V_Priority( $i, j$ )
13 end
14  $\rho_i = decideStrategy(isClose, placement, colResult, hasPriority)$ 

```

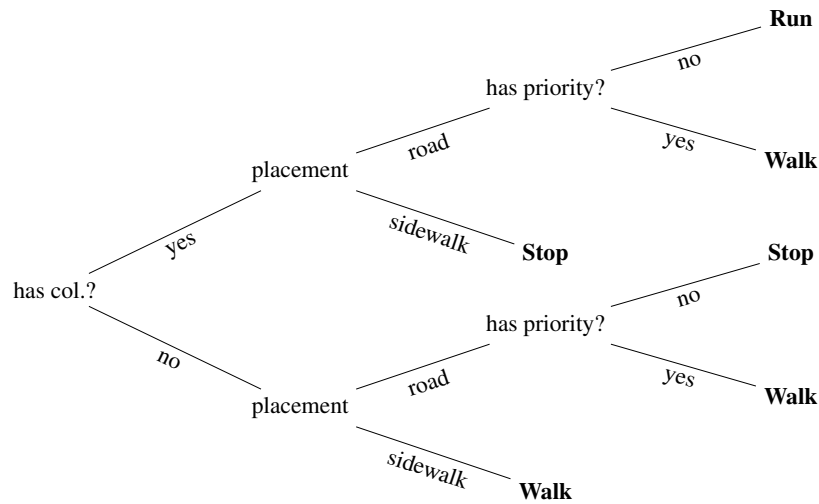


Figure 5.4 – Decision tree for pedestrians

The lateral direction of the pedestrian, if its trajectory requires change in direction, is done instantaneously, as is changes in its nominal speed of $1.5m/s$.

5.2.3 Estimating other agents' intent

Given the previous decision model, the AV can now make use of it to predict the behavior of all other road users, assuming an *a priori* intent for each one and then find the most probable intents. The following information is available for the AV at the beginning of each iteration: the position of all agents, its own goal g_i and the decision process used by other agents. This last one is considered here as a justified assumption for the AV to have due to the common knowledge about the way a human drives or behaves as a pedestrian, based on the traffic code impositions on the infrastructure and on the behavior of all agents and some introspection.

At the beginning of the simulation the AV faces the question of which intent hypothesis best matches each agent, making it possible to explain why an agent is behaving as observed. The entire scene is explained by the information vector c , defined by the union of every agent intent as displayed in the expression 5.2 for N agents. Considering that the AV's intent is always represented by the subscription 0 then $\bar{c} = (c_i)$, $i \in [1, N]$ is the vector that needs to be estimated by the AV.

$$c = (c_0, \dots, c_N) = ((g_0, f_0), \dots, (g_N, f_N)) \quad (5.2)$$

Using all possible values for c the AV tries to predict the next strategy, and consequently, the future trajectories of all other road users. Considering the example given in figure 5.1, where there are three road users and assuming that each one of them has 2 possible g_i values and 2 f_i values, it totals 64 possible c configurations. One might question why the intention of the AV is considered if this estimation about the entire environment is done from the AV's perspective and as such its intent is known; this is because it is important to understand what the other agents are thinking or expecting as a behavior from the AV, which does not necessary mean to exploit a situation where the others have a misguided impression about the AV's intent. To profit from a wrong estimation done by another road user is not so straightforward since it is desirable that the AV maintain a defensive stance when driving, although it is a subject that merits more discussion and exploration at a later time.

The proposed road-map of the algorithm consists in the following: from time t to $t + N_t$ the AV predicts the strategies of all road users according to an assumed intent vector c and propagates their positions into the future using the prediction step of a Kalman filter. Then, at $t + N_t$, an observation is made, which is compared with the predicted result to evaluate the probability of c being the right intent vector for the environment. After the calculation of $p(z_{t+1} | c_k, s_{t+1})$, where z_t represents the observation, all the predicted values are modified according to the update step of the Kalman filter. The algorithm 8 shows how this procedure implemented, being executed at each simulation step, Δt . Given this short period for the agents to develop their strategies and that physical cues to become detectable, the update of the EKF and the probability calculation procedure are only executed after N_t transitions.

The algorithm starts collecting the observation of each road user at line 2, to then proceed to the prediction of the position at this time epoch at line 4. The total number of hypothesis instances is represented by $N_{\text{ins}} = N_{\text{int}}^{N_{\text{ag}}}$, N_{int} being the number of intents per agent and N_{ag} the number of agents in the simulation (assuming that each agent has an equal number of possible intents). Both prediction and update steps are represented below as a single procedure, given that for this representation all agents' state was concatenated into a single vector, as $z_t = \bigcup_{i=0}^{N_{\text{ag}}} (z_{i,t})$ and $s_t = \bigcup_{i=0}^{N_{\text{ag}}} (s_{i,t})$. As such, all calculated values are also generalized to represent the entire collection of agents.

Algorithm 8: Estimation of $P_{t+1}(c)$

Input: Previous state S_t , Previous distribution P_t
Output: Estimated state S_{t+1} , Current distribution P_{t+1}

```

1 begin
2    $z_{t+1} \leftarrow \text{get\_observations}()$ 
3   for each hypothesis  $c_k$  considered, given  $0 \leq k \leq N_{\text{ins}}$ , do
4     prediction  $\begin{cases} \bar{\mu}_{t+1}^{[k]} \leftarrow \mathcal{T}(\mu_t^{[k]}, c_k) \\ \bar{\Sigma}_{t+1}^{[k]} \leftarrow G_{t+1} \Sigma_t G_{t+1}^T + R \end{cases}$ 
5   end
6   if  $t \% N_t == 0$  then
7     for each hypothesis  $c_k$  considered do
8       update  $\begin{cases} w_{t+1}^{[k]} = p(z_{t+1} | c_k, s_{t+1}) \cdot w_t^{[k]} \\ K_{t+1} = \bar{\Sigma}_{t+1}^{[k]} H_{t+1}^T \cdot (H_{t+1} \bar{\Sigma}_{t+1}^{[k]} H_{t+1}^T + Q)^{-1} \\ \mu_{t+1}^{[k]} \leftarrow \bar{\mu}_{t+1}^{[k]} + K_{t+1} (z_{t+1} - h(\bar{\mu}_{t+1}^{[k]})) \\ \Sigma_{t+1}^{[k]} \leftarrow (I - K_{t+1} H_{t+1}) \bar{\Sigma}_{t+1}^{[k]} \end{cases}$ 
9     end
10    calculate  $P_{t+1}(c) = \frac{w_t^{[k]}}{\sum_{j=0}^{N_{\text{ins}}} w_{t+1}^{[j]}}$  for all  $k \in [0, N_{\text{ins}}]$ 
11  else
12     $P_{t+1}(c) = P_t(c)$ 
13  end
14 end
```

If the current time epoch is a multiple of N_t the update step is executed (line 8), where the weight given for this hypothesis is calculated, together with the usual update step of a Kalman filter. The probability connected to the hypothesis c is the normalized value of each one of the calculated weights (line 10). One point remains to be discussed: the calculation of $p(z_{t+1} | c_k, s_{t+1})$, which quantify the similarity between what was observed and what was assumed *a priori* given the c_k chosen.

Kalman filter for agent prediction

This part will focus in the application of the Kalman filter for a single road user, as such all the formalism used here refers to a single road user, differently from the variables mentioned in algorithm 8. Instead, the respective parts of each vector μ_t and z_t are referenced as $s_{i,t}$ and $z_{i,t}$, i being the index for the road user. For each agent, one instance of an EKF is used. Following the prediction equations given previously for the vehicle, shown in 5.3, and for pedestrians, 5.4, the Kalman matrices G_t and H_t are defined for vehicles in expressions 5.5 and 5.6 and for pedestrians in 5.7 and 5.8.

$$s_{t+1}^{\text{veh}} = s_{t+1}^{\text{av}} = \begin{cases} x_{t+1} &= x_t + \Delta t \cdot v_t \cos \theta_t \cos \phi_t \\ y_{t+1} &= y_t + \Delta t \cdot v_t \sin \theta_t \cos \phi_t \\ \theta_{t+1} &= \theta_t + \Delta t \cdot \frac{v_t}{l} \sin \phi_t \\ v_{t+1} &= v_t + \Delta t \cdot u_1 \\ \phi_{t+1} &= \phi_t + \Delta t \cdot u_2 \end{cases} \quad (5.3)$$

$$s_{t+1}^{\text{ped}} = \begin{cases} x_{t+1} &= x_t + \Delta t \cdot v_t \cos \theta_t \\ y_{t+1} &= y_t + \Delta t \cdot v_t \sin \theta_t \\ v_{t+1} &= v_t + \Delta t \cdot u_1 \\ \theta_{t+1} &= \theta_t + \Delta t \cdot u_2 \end{cases} \quad (5.4)$$

The state of each other road user that is a vehicle also contains the steering angle, since, due to the strategy estimation, the control using the trajectory defined by g_i needs to be calculated. The pedestrian's state remains being expressed by the previous four variables (x, y, θ, v) .

$$G_t^{\text{veh}} = \begin{bmatrix} 1 & 0 & -\Delta t \cdot v_t \sin \theta_t \cos \phi_t & \Delta t \cdot \cos \theta_t \cos \phi_t & -\Delta t \cdot v_t \cos \theta_t \sin \phi_t \\ 0 & 1 & \Delta t \cdot v_t \cos \theta_t \cos \phi_t & \Delta t \cdot \sin \theta_t \cos \phi_t & -\Delta t \cdot v_t \sin \theta_t \sin \phi_t \\ 0 & 0 & 1 & \frac{\Delta t}{l} \sin \phi_t & \frac{\Delta t v_t}{l} \cos \phi_t \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (5.5)$$

$$H_t^{\text{veh}} = I_5 \quad (5.6)$$

$$G_t^{\text{ped}} = \begin{bmatrix} 1 & 0 & -\Delta t \cdot \cos \theta_t & \Delta t \cdot v_t \sin \theta_t \\ 0 & 1 & \Delta t \cdot \sin \theta_t & \Delta t \cdot v_t \cos \theta_t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5.7)$$

$$H_t^{\text{ped}} = I_4 \quad (5.8)$$

The convergence of an Kalman filter is assured for the systems that obey the pre-requisites given by theorem 5, proved in (Krener, 2003). The system considered is given in the form of 5.9, which is uniformly observable if the initial state can be reconstructed with the output and its $(n - 1)$ first derivatives, n being the total number of state variables (Gauthier et al., 1992).

$$\begin{cases} \dot{x} = f(x, u) \\ y = h(x, y) \end{cases} \quad (5.9)$$

Continuity of a function $f : X \rightarrow Y$ can be represented by the Lipschitz condition, given by the expression 5.10. The Euclidean norm is used as the metric for the spaces X and Y , with $L \in \mathbb{N}$ and $x_1, x_2 \in X$.

$$\|f(x_1) - f(x_2)\| \leq L \cdot \|x_1 - x_2\| \quad (5.10)$$

Finally, the theorem below says that for the conditions assumed the EKF converges exponentially to the correct result.

Theorem 5 *Convergence of EKFs*

Suppose that:

- *The system in question is uniformly observable and satisfies the Lipschitz condition.*
- *The second derivative of $f(x, u)$ is bounded.*
- *μ_t and Σ_t are the solutions of the extended Kalman filter, where Σ_0 is positive definite and $s_0 - \mu_0$ is sufficiently small*

Then $\|s_t - \mu_t\| \rightarrow 0$ exponentially as $t \rightarrow \infty$.

When a Kalman filter is used to estimate the evolution of some system through time the output is a Gaussian distribution. This precise point is used to measure the similarity between

what is being assumed as an hypothesis for the behavior of the agents and what is really being observed.

Mahalanobis distance and similarity estimation

Usually when one wants to measure the similarity between two vectors the Euclidean norm is used, if these two vectors are defined as points in an Euclidean space. For example, if one have $u, v \in \mathbb{R}^5$, then the distance between the points is given by expression 5.11, where u_i and v_i are the elements of each one of the respective vectors.

$$d(u, v) = \sqrt{\sum_{i=0}^4 (u_i - v_i)^2} \quad (5.11)$$

The same is not valid for the output of the Kalman filter. The distance can be calculated between the observation and the distribution mean, but an importance source of information will be left out of the measurement, the covariance. Therefore one needs to evaluate the similarity between a point, the observation obtained, and a distribution, the prediction of each agent state maintained using the EKF, which can be done using the Mahalanobis distance expressed by equation 5.12. This type of norm measures the distance between a sample from a distribution and its center.

$$M(x) = [(x - \mu)^T \cdot \Sigma^{-1} \cdot (x - \mu)]^{\frac{1}{2}} \quad (5.12)$$

So, to measure the coherence of an hypothesis c_k used to predict the state of each agent, the observation is considered as a sample of the state distribution.

$$M_t^{[k]} = \left[(z_t - \mu_t^{[k]})^T \cdot (\Sigma_t^{[k]})^{-1} \cdot (z_t - \mu_t^{[k]}) \right]^{\frac{1}{2}} \quad (5.13)$$

It is well known that the quadratic form of a Gaussian variable is distributed according to the chi-squared distribution (Bensimhoun, 2009). To be more precise, the cumulative sum of squares of n independently normally distributed Gaussian variables, with mean equal to 0 (or the null vector) and covariance equal to the identity matrix (I_n) is called central chi-squared distribution with n degrees of liberty. This distribution will be used to calculated the probability $p(z_{t+1} | c_k, s_{t+1})$. However, as said before, the central chi-squared distribution is only *a priori* valid for $\mathcal{N}(0, I_n)$ which is not the case for the distributions treated here.

If one considers the expression $(z_t - \mu_t^{[k]})$ a new variable, called U , it can be assumed that $U \sim \mathcal{N}(0, \Sigma_t)$ since z_t is being used as a point of this distribution, thus the difference must be zero. Then the theorem 5.1.3 on page 199 of (Mathai and Provost, 1992) can be stated:

Theorem 6 *Distribution of $M_t^{[k]}$*

If $U \sim \mathcal{N}(\mu_u, \Sigma)$ and Σ is positive semi-definite, then a set of necessary and sufficient conditions for $U^T A U \sim \chi_n^2(\delta^2)$ is:

- $\text{trace}(A\Sigma) = n$ and $\mu^T A \mu = \delta^2$
- $\Sigma A \Sigma A \Sigma = \Sigma A \Sigma$
- $\mu^T A \Sigma A \mu = \mu^T A \mu$
- $\mu^T (A\Sigma)^2 = \mu^T A \Sigma$

Given that $A = \Sigma^{-1}$ and $\mu_u = 0$ the non-centered chi-squared distribution from the theorem becomes centered due to $\delta^2 = \mu^T A \mu = 0$ and all the conditions are met. Equation 5.14 defines the opposite value of the cumulative probability of the ellipsoid defined between the observation and the predicted state. Such ellipsoid, if small, shows that the observation is close to the center of the distribution, meaning that the hypothesis k might be pertinent. The probability that the Mahalanobis distance d_M is smaller than the calculated value $M_t^{[k]}$ is the same as the cumulative probability of χ_n^2 , which value is commonly known.

$$v_t^{[k]} = 1 - P(d_M \leq M_t^{[k]}) \quad (5.14)$$

Finally, the probability of an observation given an assumed hypothesis and the current state can be defined by equation 5.15.

$$p(z_t | c_k, s_t) = \frac{v_t^{[k]}}{\sum_{i=0}^{N_{\text{ins}}} v_t^{[i]}} \quad (5.15)$$

Using the algorithm 8, that estimates the probability of each possible intent for the agents being true, the AV can now know which strategy it should use if all road users act according to the decision-making procedure proposed in algorithm 5. One detail was left out of the decision process: the variable f_k , which controls the behavior of a road user during the interaction. Such variable is used to emulate instances where an agent does not behave as it should: for example, if a vehicle does not yield for a pedestrian crossing the road. The AV might not be able to identify such behavior and thus react sub-optimally to it.

As such, the AV needs to choose a strategy considering a deviation from normal behavior coming from other road users. To maximize the utility of the worst possible scenario is an operation embodied by the Nash equilibrium in a game theory approach. But the same detail that created the necessity to establish the estimation process of c_k is also present if this approach is applied: the agents do not know which is the intention of the other ones around them. Everything done up until now compared these two by two to obtain an expected strategy given an assumed intent; it is necessary now to consider the collective set of calculated strategies and the estimated agents' intent and their probabilities of being correct to arrive in a coherent decision for the AV.

Such configuration requires an incomplete game, where part or all the players do not have access to some essential information. For such cases the method proposed in (Harsanyi, 1967) can be used to reach a formulation of the game that has a defined solution.

5.3 Incomplete game model

As discussed in chapter 2, in an incomplete game the players do not have access to some information about the other players. Without such information it may be impossible to calculate the utility of each possible action that might be taken (if the utility function form or some other parameter is not known) and therefore to find an equilibrium point of the problem, if it exists. To counteract the lack of information, this incomplete game can be converted into a complete imperfect one in which the Nash equilibrium is applicable.

5.3.1 Nash equilibrium

In a non-cooperative n -player game each player has at its disposal a set of different strategies⁵, which, according to the idea of a game, produces different utilities for the player, each one also depending on the choices of other road users. Every road user tries to anticipate the others' decision and maximize its utility and it is from this interaction between every player that the notion of equilibrium point emerges.

A pure strategy is defined as a single action that an agent may execute, while a mixed strategy is represented by a probability distribution over different actions (or similarly, over pure strategies). As one may expect the interaction between agents happens through the choice of a strategy; if it is assumed that all agents are rational, i.e. that they search to maximize their preferences (deterministic reward for pure strategies and expected reward for mixed ones), then

⁵Used in this sense, strategy here is something more general than the longitudinal control profiles proposed earlier, even though when this theoretical discussion is applied to the problem treated in this chapter, this general sense is compatible with the definition of strategy used in subsection 5.2.3.

each agent tries to anticipate the choice of the other agents and counter it with the best strategy from its point of view. This same procedure is repeated inside other agents.

When, for every agent, there is no other better strategy than the one chosen, i.e. the current strategy is self-countering, then an equilibrium point was reached (Nash, 1950). It has been proven in (Nash, 1950) and (Nash, 1951) that every finite game has an equilibrium point, which configures the best collective reward for all the n -players in a non-cooperation game. The following definition from (Osborne, 2002) resumes the idea of Nash equilibrium:

A Nash equilibrium is an action profile (or a strategy) a^ with the property that no player i can do better by choosing an action different from a_i^* , given that every other player j adheres to a_j^* .*

However, the existence of an equilibrium point is only assured in the case of mixed strategies (Osborne, 2002). This concept is essential to define what each agent does in response to an interaction and it is the main tool to solve a game model. It assumes that each agent is capable to predict the behavior of all other agents, which entails that the intent of each agent needs to be perfectly known by the AV, which is not realist. It is at this point that a Bayesian game formulation is necessary.

5.3.2 Harsanyi's Bayesian game

In a n -player game with complete information all agents know the entirety of the necessary information to predict the choices of the other agents, all their possible strategies and their preferences. However, if these details are not accessible one can argue that each agent maintain a subjective probability distribution over the possible alternatives. The conversion from incomplete to complete information proposed in (Harsanyi, 1967) uses this concept of probability over unknown information to try to find a complete game formulation that has similar properties to the original one.

As said before, each player tries to anticipate the behavior of all others. In a incomplete game, this means that player 1 has some expectation over the behavior of player 2, which in turn also has some expectations about player 1. Such sequence can go on forever, since now player 1 also has an expectation over the expectation of player 2 has of him and so on. The Bayesian game approach model these expectations as subject⁶ probability distribution over the behavior of all other players (back later on).

⁶This probability, differently from the objective definition, refers to the belief of one agent about some phenomena which need not be the same for each agent (Savage, 1954), therefore creating a graded belief towards this phenomena (Hajek, 2019). The objective definition, according to (Harsanyi, 1967), refers to the frequentist interpretation of probability.

One can postulate that all the subjective probability distributions considered by each agent are the marginal distributions of some original, base probability distribution that has as parameters the information not known by the players, as indicated in expression 5.16, where R refers to the probability distribution and c to the unknown information (which will be the intent vector for the environment). Then the original game and the complete game formed by the substitution of those subjective probability distributions by the marginal probability from this base distribution, according to the definition 2, are equivalent, which entails that both have the same resulting decision rule and the same Nash equilibrium point(s).

$$R(c) \begin{cases} R(c_1, c_2, \dots, c_n | c_0) \\ R(c_0, c_2, \dots, c_n | c_1) \\ \vdots \\ R(c_0, c_1, \dots, c_{n-1} | c_n) \end{cases} \quad (5.16)$$

Definition 2 *Bayes-equivalency*

Let G be an incomplete game and G^* be a complete one, both in standard form (defined by expression 5.18). Both games are *Bayes-equivalent* if for a player j :

- *The two games must have the same strategy (S_i) and information (C_i) spaces (also called intent).*
- *Both games must have the same expected value function for the payoffs.*
- *The subjective probability distribution R_i from player i must satisfy the following relationship:*

$$R_i(c_0, c_1, \dots, c_{i-1}, c_{i+1}, \dots | c_i) = R_i^*(c_0, c_1, \dots, c_{i-1}, c_{i+1}, \dots | c_i) \quad (5.17)$$

Form of a game

At this point it is necessary to give more details about the standard form of a game and its normal form, which will appear subsequently. The standard form of a particular incomplete game G , defined in (Harsanyi, 1967), has the composition shown in expression 5.18. In this context S_i represents the strategy space, C_i the information space (which space of possible values for the intent vector), V_i represents the expected value for the payoff⁷ and R_i the base probability

⁷In (Harsanyi, 1967), the function V is defined as the common form of the payoff functions of each player, which is only known by the respective player.

distribution.

$$G^* = \{S_0, \dots, S_n; C_0, \dots, C_n; V_0, \dots, V_n; R_i\} \quad (5.18)$$

The normal form of a game is presented in definition 3 (Shoham and Leyton-Brown, 2008). It is defined directly using actions, but it can also be defined in function of the strategies that are available. If these strategies are pure, i.e. they represent only one action, then nothing changes. But if one needs to consider mixed strategies then the total payoff needs to be calculated as an expected payoff.

Definition 3 *Normal-form game*

A finite n -person game in normal form is defined by the following parameters:

- N , the number of players.
- $A = A_1 \times A_2 \times \dots \times A_n$, the finite set of actions.
- $u = (u_1, \dots, u_n)$, where $u_i : A \rightarrow \mathbb{R}$ is the payoff function for the player i .

When the game is defined in the normal form, the strategies are also normalized. The difference from the usual definition is that the normalized ones are actually functions from the intent space of the road user in question to the strategy space.

Solving the equivalent game

The necessary and sufficient conditions for two games, the original G and the complete one G^* , to be equivalent is enunciated by theorem 7, found in (Harsanyi, 1968b). Basically, it says that if two games are equivalent, according to the definition 2, they need to have the same strategy and attribute spaces, the same payoff functions and must respect the relationship described in 5.16. Games where the hypothesis that the original distribution exist can be called consistent games, and those that do not, inconsistent. It will be assumed here that the game involving the interaction between road users is a consistent one, although even if a game is inconsistent, the equivalency can be extended to this case (Harsanyi, 1968a).

In the theorem below the normal form of the game is cited. This game formulation, differently from the extensive game that exposes all possible outcomes for each player, is composed of only three elements: the number of players, the set of available strategies for each player and the combined payoff function, that for each player gives its payoff according to its current strategy, there is not a dependency between players in this calculation.

Theorem 7 *Bayes-equivalency*

Let G be an incomplete game and G^* be a Bayesian equivalent game to G . To an n -tuple strategy s^* of G^* also be an equilibrium point in G is it necessary and sufficient that in the normal form $\mathfrak{N}(G^*)$ of the game G^* , s^* be an equilibrium point.

The estimated probability by algorithm 8 will serve as the base probability to calculate the equilibrium points of the equivalent game. This discrete original distribution, calculated by the Mahalanobis distance and the chi-square distribution, is itself based on the conclusion obtained from the comparison about what each road user should do in the considered situation.

According to the determination of each strategy proposed in the previous section each strategy is a pure one, consisting of a single action possible according to the situation at hand. The Nash equilibrium point is not guaranteed to exist in such situations, but it will be assumed that it does exist. Its determination, given the implementation specifications of this context, is similar to the example presented in (Harsanyi, 1968b).

5.3.3 Decision making procedure

The main objective of this procedure is to account for every possible situation in the AV's decision-making, so as to avoid instances where the AV is not prepared for a behavior from some road user. Of course, only the behaviors represented by the considered strategies are involved into the deliberation, any other behavior that cannot be approximated by them is not considered. This is a limitation of the procedure, but if one assumes that such strategies can be extracted from real data, one can find a sufficient set of strategies to represent all possible reactions during road users interactions. From the equivalent game formulation, the Nash equilibrium is calculated the payoff values variations according to the intent vector probabilities and the strategies considered.

Each agent will use a reward function that is similar in form to the MDP implementation displayed in subsection 3.2.3. For each intent vector hypothesis there will be a table of payoffs. These tables are in turn connected to the probability that such situation is actually happening. Considering the three agents in the scene represented by figure 5.1, the intent vector is formed by three components, c_1 , c_2 and c_3 and the normalized strategies are represented by the Greek letters ν , γ and ρ , respectively for each agent. They are functions that take as input the current value of the respective c_i and output the strategy that should be used in this given situation.

$$c = (c_1, c_2, c_3) \mapsto (\nu, \gamma, \rho) \quad (5.19)$$

Then, for each possible value of the c vector, the payoff is calculated, giving different values for each type of strategy, represented by the Greek letters presented earlier with the

subscription that indicated the strategy index, as it can be seen in table 5.1. In this table agent one's strategy is represented by v in the row direction while the combination of the possible values for γ and ρ are in the column direction. For clarity reasons each agent will be considered to have only two possible strategies, even if the vehicles have six possible strategies and pedestrians three. Multiple tables like the one below are calculated for different combinations of the intent vector c .

Table 5.1 – Example of reward table for a specific intent vector c

		$c = (c_1, c_2, c_3)$			
$v \setminus \gamma\rho$	$\gamma^1\rho^1$	$\gamma^1\rho^2$	$\gamma^2\rho^1$	$\gamma^2\rho^2$	
v^1	$r_c^{1,11}$	$r_c^{1,12}$	$r_c^{1,21}$	$r_c^{1,22}$	
v^2	$r_c^{2,11}$	$r_c^{2,12}$	$r_c^{2,21}$	$r_c^{2,22}$	

Table 5.1 exemplifies the reward calculated for each strategy given an intent vector. According to the definition of the Nash equilibrium point, no agent can achieve a better outcome, so if the system is solved from the viewpoint of agent one, it is not necessary to calculate all the rewards again and find the equilibrium point for agent two and three.

With the probability distribution $P_{t+1}(c)$ calculated in algorithm 8, the total estimated reward is determined by equation 5.20 for a certain combinations of strategies. The functions $f_1 : c \rightarrow \sigma$, f_2 and f_3 represent the normalized strategy being evaluated. For example, f_1 can assume four possible values according to the number of intents agent one can have (two in this example) and the number of possible strategies (also two); if, for agent one $f_1 = 21$ this means that for the first intent the agent will choose the strategy 2 and for the other intent, the strategy 1. Using this codification the estimated payoff matrix is formed having as row the possible values for f_1 and as column the combination of f_2 and f_3 . Constants N_{c_1} , N_{c_2} and N_{c_3} represent the total number of intents for each agent.

$$\mathbb{E}[r^{v^{f_1}, \gamma^{f_2}, \rho^{f_3}}] = \sum_{i=0}^{N_{c_1}} \sum_{j=0}^{N_{c_2}} \sum_{k=0}^{N_{c_3}} P_{t+1}(c_i, c_j, c_k) \cdot r_{c_i, c_j, c_k}^{v^{f_1(c_i)}, (\gamma^{f_2(c_j)}, \rho^{f_3(c_k)})} \quad (5.20)$$

But if one considers that the intent of agent one is known *a priori*, only the conditional expected reward is necessary, cutting the number of evaluations needed (equation 5.21). Then, the conditional estimated payoff matrix will have two rows (represented in the equation below by the index l), one for each of the possible strategies of the agent one, and the columns are all the possible strategies for agents two and three $((\gamma^{11}\rho^{11}), (\gamma^{11}\rho^{12}), (\gamma^{11}\rho^{21}), \dots, (\gamma^{22}\rho^{22}))$, a total

of sixteen columns).

$$\mathbb{E}[r^{v^l} \gamma^{f_2} \rho^{f_3} | c_1] = \sum_{j=0}^{N_{c_2}} \sum_{k=0}^{N_{c_3}} P_{t+1}(c_j, c_k | c_1) \cdot r_{c_j, c_k}^{v^l, (\gamma^{f_2(c_j)} \rho^{f_3(c_k)})} \quad (5.21)$$

The optimal strategy in a Nash sense is calculated by equation 5.22 using a maxmin operation, consisting of a minimum possible reward in the column, represented by the index h , for each possible normalized strategy for the other agents, and then a maximization operation to find the best strategy overall in the minimized row. Such strategy, as expected, will give the security that at the worst possible situation the payoff is maximized but not necessarily that the highest possible payoff is obtained in the best scenario.

$$v^* = \operatorname{argmax}_l \min_h \mathbb{E}[r^{v^l} \gamma^{f_2} \rho^{f_3}] \quad (5.22)$$

It is the strategy dictated by v^* that should be executed by the AV, together with the lateral reference given by its trajectory. This normalized strategy, that depends only on the current state of agent one, is a pure strategy since the relationship between current intent and strategy is deterministic. According to the theorem 7 this equilibrium point will also maximize the payoff from the original incomplete game.

Example of application

To better explain the calculation of the Nash equilibrium point and the use of equations 5.19 to 5.22 an example will be provided below. It will be composed of two players, x and y , that each can have two different intents i_1 and i_2 and execute the actions a^1 and a^2 . This example is similar to the one found in (Harsanyi, 1968b), but it will serve to explain how to apply the method used to calculate the equilibrium point with more than two players. Table 5.2 gives the reward for each possible situation for player x , while the rewards for player y are the same but multiplied by -1 . The circled numbers are the saddle points for each situation, the rows give the x actions and the columns the y actions.

The functions v and γ represent the strategy for agents x and y respectively given their intent. Table 5.3 gives the intent probability, which is the output of the algorithm 8.

So, with these elements it is already possible to calculate the normalized strategies for the road user x . There are two possible ways to proceed to the calculation of the Nash equilibrium point: or the equation 5.20 can be used to calculate the expected payoffs for all possible intents of x or, since its intent is known by itself, equation 5.21 can be applied. The second choice will be

(a) Reward for intentions $c_1 = (i_1, i_1)$			(b) Reward for intentions $c_2 = (i_1, i_2)$		
$x \backslash y$	$\gamma(i_1) = a^1$	$\gamma(i_1) = a^2$	$x \backslash y$	$\gamma(i_2) = a^1$	$\gamma(i_2) = a^2$
$v(i_1) = a^1$	$r_{c_1}^{1,1} = \textcircled{5}$	$r_{c_1}^{1,2} = 10$	$v(i_1) = a^1$	$r_{c_2}^{1,1} = 10$	$r_{c_2}^{1,2} = -25$
$v(i_1) = a^2$	$r_{c_1}^{2,1} = 3$	$r_{c_1}^{2,2} = -5$	$v(i_1) = a^2$	$r_{c_2}^{2,1} = -5$	$r_{c_2}^{2,2} = \textcircled{-8}$

(c) Reward for intentions $c_3 = (i_2, i_1)$			(d) Reward for intentions $c = (i_2, i_2)$		
$x \backslash y$	$\gamma(i_1) = a^1$	$\gamma(i_1) = a^2$	$x \backslash y$	$\gamma(i_2) = a^1$	$\gamma(i_2) = a^2$
$v(i_2) = a^1$	$r_{c_3}^{1,1} = 9$	$r_{c_3}^{1,2} = -31$	$v(i_2) = a^1$	$r_{c_4}^{1,1} = 1$	$r_{c_4}^{1,2} = \textcircled{-4}$
$v(i_2) = a^2$	$r_{c_3}^{2,1} = \textcircled{15}$	$r_{c_3}^{2,2} = 27$	$v(i_2) = a^2$	$r_{c_4}^{2,1} = -24$	$r_{c_4}^{2,2} = -5$

Table 5.2 – Rewards for player x

Table 5.3 – Intent probabilities

$x \backslash y$	$c_2 = i_1$	$c_2 = i_2$
$c_1 = i_1$	0.5	0.1
$c_1 = i_2$	0.1	0.3

made here and the x intent will be equal to i_1 , resulting in the conditional probabilities calculated in table 5.4 (The probabilities P_{t+1} are indicated by p).

Table 5.4 – Conditional probabilities for $c_x = i_1$

$x \backslash y$	$c_y = i_1$	$c_y = i_2$
$p(c_y c_x = i_1)$	0.833	0.167
$p(c_y c_x = i_2)$	0.25	0.75

Simplifying the equation 5.21 to two players gives:

$$\mathbb{E}[r^{v^1 \gamma^{f_2}} | c_x = i_1] = \sum_{j=0}^{N_{c_2}} p(c_y = i_j | i_1) \cdot r_c^{v^1, (\gamma^{f_2}(c_j))} \quad (5.23)$$

To determine the conditional payoff table, each row is formed by a possible strategy for x , $v(i_1) = a^1$ or $v(i_1) = a^2$, while the columns are determined by the possible strategies for y , indicated by γ . The values that it can assume are displayed below. It becomes clear then the definition of the function f_2 for the general case: if one considers the first item in the box below,

$f_2(i_1) = a^1$ and $f_2(i_2) = a^1$, which defines γ^{11} (the numbers indicate the strategy index for each intent of y).

- $\gamma(c_y = i_1) = a^1$ and $\gamma(c_y = i_2) = a^1$ or γ^{11}
- $\gamma(c_y = i_1) = a^1$ and $\gamma(c_y = i_2) = a^2$ or γ^{12}
- $\gamma(c_y = i_1) = a^2$ and $\gamma(c_y = i_2) = a^1$ or γ^{21}
- $\gamma(c_y = i_1) = a^2$ and $\gamma(c_y = i_2) = a^2$ or γ^{22}

The calculation of the conditional payoff for the cell in line one, column one is:

$$\begin{aligned} \mathbb{E}[r^{v^1\gamma^{11}} | c_x = i_1] &= p(c_y = i_1 | c_x = i_1) \cdot r_{c_1}^{1,1} + p(c_y = i_2 | c_x = i_1) \cdot r_{c_2}^{1,1} = \\ &= 0.833 \cdot 5 + 0.167 \cdot 10 = 5.833 \end{aligned}$$

And for the cell in line two, column three:

$$\begin{aligned} \mathbb{E}[r^{v^2\gamma^{21}} | c_x = i_1] &= p(c_y = i_1 | c_x = i_1) \cdot r_{c_1}^{2,2} + p(c_y = i_2 | c_x = i_1) \cdot r_{c_2}^{2,1} = \\ &= 0.833 \cdot (-5) + 0.167 \cdot (-5) = -5 \end{aligned}$$

Finally, the conditional payoff table is given by 5.5.

Table 5.5 – Conditional payoffs for $c_x = i_1$

$x \backslash y$	γ^{11}	γ^{12}	γ^{21}	γ^{22}
v^1	5.833	0	10	4.167
v^2	1.667	1.167	-5	-5.5

To find the best strategy to be adopted, in a Nash sense, it is only necessary to apply the minimal to each row, giving 0 to v^1 and -5.5 to v^2 and then to maximize this result, finding that the best strategy is v^1 , given that $c_x = i_1$.

5.4 Simulation results

Currently only the multi-agent simulation proposed in subsection 5.2.2 is ready to offer some results; the intent estimation procedure proposed in subsection 5.2.3 has been implemented

but not entirely to offer consistent outputs. Therefore, the results presented here will concern only the interaction between the multiple independent agents in the simulation. The simulated scene is given by the figure 5.1, with the same three road users. Given the limitations cited, the two vehicles are implemented as manual drivers with the decision tree component, and the simulation pertains only to the interaction between these road users.

5.4.1 Multi-agent simulation

According to the figure 5.1, there are two possible goals (g_i) for each agent. The red vehicle can turn right or left, the blue vehicle can continue straight or turn right and the pedestrian can cross the intersection to the right or continue straight to the other pedestrian strip. These options are combined with the second element, of the intent, f_i , which is the reaction during the interaction, creating four possible intents for each road user. For each vehicle and in this context f_i indicates if they will obey an yield imperative, since in an intersection without semaphores the vehicle on the right always has priority; and for the pedestrian it defined which is the reaction of the pedestrian in a dangerous situation: if it starts to run or if it freezes and stop in the middle of the road.

Three different scenes will be displayed as a result:

- **Situation 1:** Red vehicle turns right, blue vehicle turns right and pedestrian crosses to the right.
- **Situation 2:** Red vehicle turns right, blue vehicle continues straight and pedestrian crosses to the right.
- **Situation 3:** Red vehicle turns right, blue vehicle turns right and do not give priority and pedestrian crosses the pedestrian strip on top.

All the details about the vehicle object implementation, the implementation of each strategy and the simulation itself can be found in appendix A.

Situation 1

An image from the output of the simulation when all road users are approaching the intersection can be seen in figure 5.5. The same situation is shown (not exactly the same time epoch) in figure 5.6, where it can be seen on the right side the three agents moving and on the left the trajectory colored with the respective strategy in execution at each position.

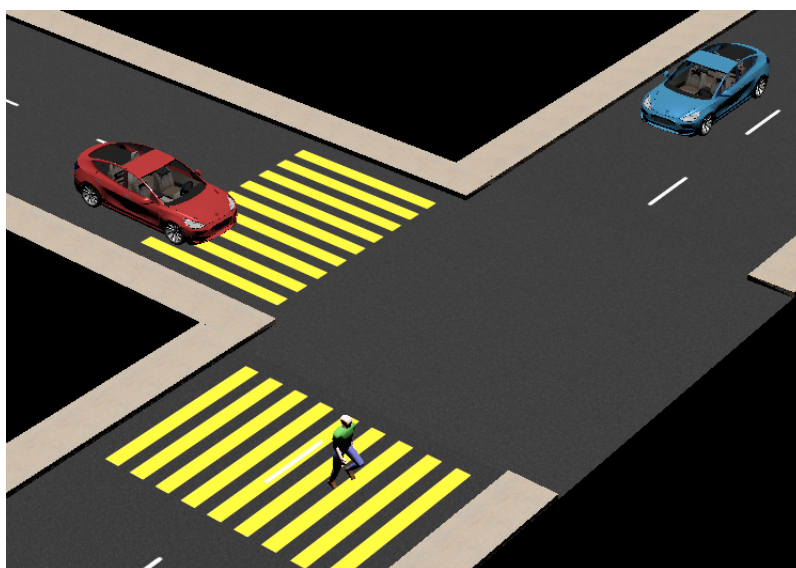


Figure 5.5 – Illustration of the intersection interaction

Considering that the pedestrian starts to move at $t = 7s$, one can clearly see that the red rectangle agent (which represents the red vehicle at figure 5.5) detects that the pedestrian is crossing the road and therefore has the preference. It decreases the velocity to stop in the intersection and avoid a collision, until the point that it can be sure that no collision with this pedestrian can happen, then it resumes the cruise strategy and turns downwards.

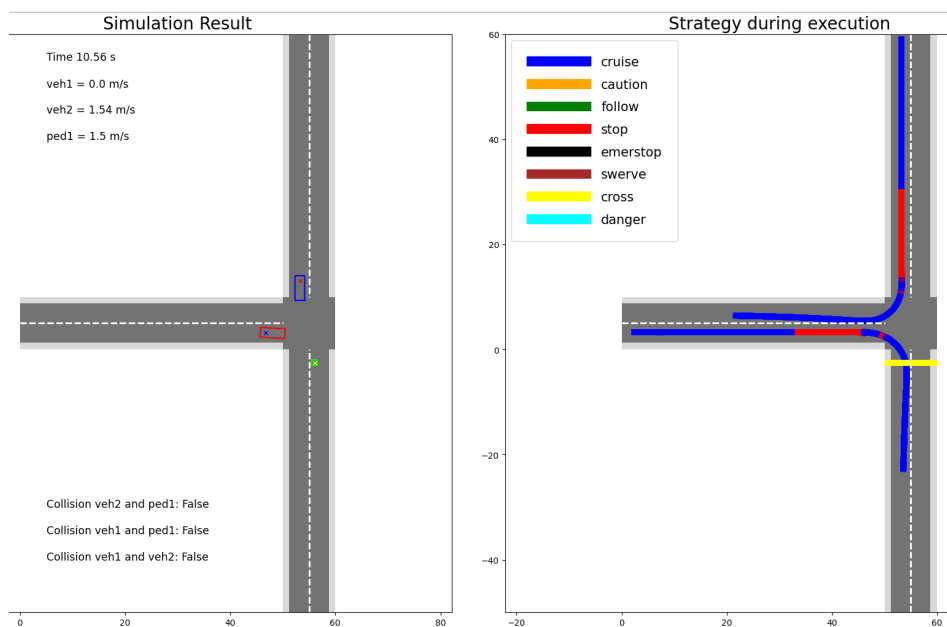


Figure 5.6 – Strategies used during situation 1

The other vehicle (blue rectangle, blue car) stops first because it detects a collision with the red agent during the turning trajectory, but resumes the cruise soon after. The pedestrian does not detect any danger since the red vehicle waits for it to get to the opposite lane to start moving again.

Situation 2

In this situation the blue vehicle now will continue straight ahead instead of turning right. As it can be seen in figure 5.7, it detects that there is a pedestrian that might cross the street and that the red vehicle has the priority over him, and because of these reasons it engages the stop strategy very soon into the simulation, stopping right at the intersection. The red vehicle, given that it needs to decelerate to turn and that in that specific instance the acceleration sampled was $2.5m/s^2$ (the discussion about the acceleration sampling can be found in appendix A), did not need to stop and wait for the pedestrian to arrive at the opposite lane.

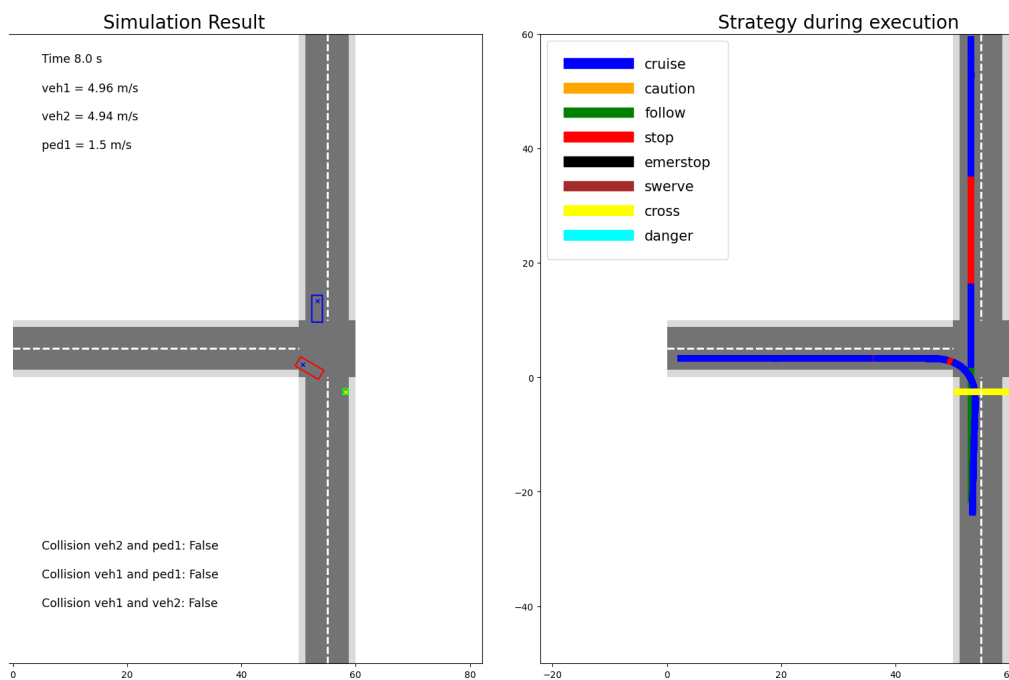


Figure 5.7 – Strategies used during situation 2

After the red vehicle turned, the blue vehicle starts the intersection crossing, it detects that the red vehicle is in front and then start to follow him, as it can be seen by the green trajectory in figure 5.7. Both vehicles continue cruising until the end of the simulation.

Situation 3

This situation emulates a scenario where one of the vehicles, in this case the blue one, do not abide the priority to the other vehicle and the pedestrian. However, since the idea is to represent a human driver, the decision making is still sensible to collisions. Looking at the figure 5.8 one can see that there is one interval in the blue's trajectory that is a emergency stop strategy, which is exactly due to a possible collision with the pedestrian and the red vehicle. When the pedestrians freeze in reaction to the proximity of the blue car the red vehicle also engages in a controlled stop at the intersection, thus avoiding the collision altogether.

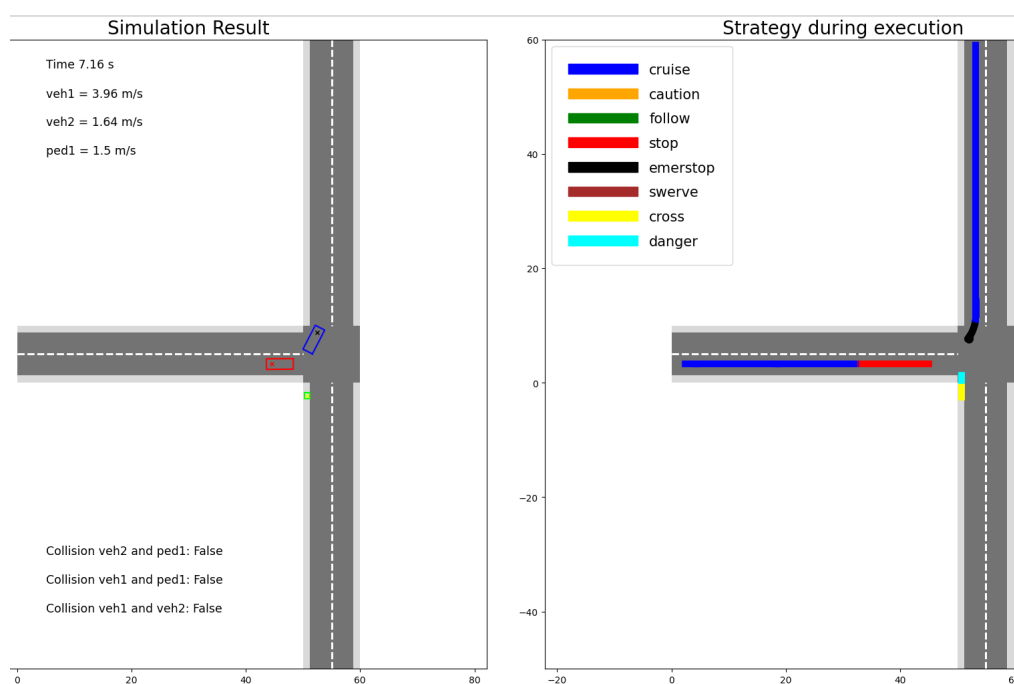


Figure 5.8 – Strategies used during situation 3

The parameter f_i from the blue vehicle overrides the common decision tree for vehicles shown in figure 5.3. In the no collision branch, the intermediary nodes related to the priority are eliminated, since the blue vehicle will not yield in any situation; the new branch taken these modifications into account can be seen in figure 5.9. The collision branches continue with the same structure, which is why the blue vehicle, when it perceives a collision, uses the emergency stop strategy.

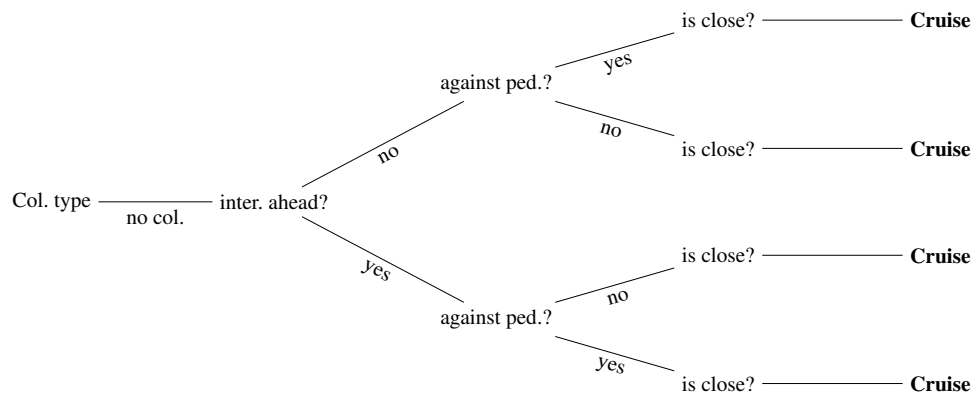


Figure 5.9 – Modified vehicle decision for the no collision sub-tree

5.5 Final remarks

In chapter 3, to allow the definition of the MDP state as only the AV's configuration it was assumed that, from the start of the policy execution, after its calculation, the intentions of other road users did not change, which entails that their direction and velocity remains the same. This continuity allows the AV to predict their trajectory only using their current configuration. The main goal of this chapter was to propose a method capable to predict this possible change in intent during the policy execution, so as to relax this assumption made. Three main components form the prediction algorithm: the deterministic decision-making that emulate other vehicles and pedestrians, the probabilistic distance estimation, that calculates the probability of some intent given an observation of the current situation and an incomplete game model to adapt the choice of strategy to the current interaction between agents. Finally, the results of the multi-agent simulation using the decision-making models are shown.

However, some considerations about the proposed intent estimation method needs to be made. During the entire chapter the deterministic decision-making model proposed was suggested to be used by the AV as a way to predict the other road users strategy and as the decision-making process of the other road users themselves. This would work with the estimation of intent by the AV, but the other capability, the game model to account the interactions between road users, is supposed to close the gap between the simulation and real data. Thus, to test the incomplete game formulation in a simulation it would be necessary to find another decision-making procedure to be implemented in the agents, preferably one that mimics real humans using real data.

Another concern related to the game model is that it is based on the assumption that the Nash equilibrium point could be reached using a pure strategy, which entails that for each agent, given its state and intent, the strategy taken is deterministic. These agents are human drivers and

pedestrians, that have personalities and different motivations, thus such assumption that every single one would act equally is a strong one and must be relaxed. A mixed strategy would be more suited for the problem, creating an additional degree of liberty according to the specificity of each component that predicts an agent behavior. To allow such consideration it would be necessary to update the proposed intent estimation given by algorithm 8 to a nested particle filter, capable to consider multiple hypothesis of intent and for each intent multiple hypothesis of strategy. Afterwards the proposed Nash equilibrium point determination would also need to be modified to deal with mixed strategies.

Chapter 6

Conclusion

Contents

6.1 Final remarks	141
6.2 Future research perspectives	144

6.1 Final remarks

In this thesis the question concerning how the automated vehicle should deliberate on decisions, given its perception information, mission data and the presence of other road users in its surroundings was treated. Both normal situations for a human driver and the hypothetical case of an ethical dilemma situation were addressed. The starting point of the discussion around automated vehicles, and more generally artificial intelligence and automated mobile robots, was developed in chapter 1, from the early days of artificial intelligence with the definition of the Turing Test, the development of the first automated mobile robots, the paradigms to orient the architecture of such robots and the development of driveless vehicles during this process. Some of the most important breakthroughs in the area were cited, for example the works of Dickmanns and Zapp, 1987 and the two prototypes that won the DARPA challenges, Thrun et al., 2006 and Urmson, Anhalt, Bagnell, et al., 2008.

Having done a review of the history of automated systems, the current state of the art in the domain of automated vehicles was finally described, from most useful advantages that one hopes the AV technology delivers, the current maturity level of the deployment of such robots in the real world and what is currently being commercialized by the automotive industry. The most promising prototypes of level four automated driving were also discussed, with a special attention to the Tesla's implementation of automated driving.

Chapter 2 discusses the state of the art of each domain addressed in the theses: decision-making, ethical decision-making, behavior prediction and game theory. Each subject starts with an explanation about the organization of the concepts that define the field of research to then proceed towards the analysis of a set of representative publications, recent and older, from each domain.

The decision-making algorithm for an AV is proposed and explained in chapter 3. Starting from the discussion about implementation paradigms from chapter 1, the choice of architecture assumed, so as to model the decision-making process with the correct responsibilities and functionalities, was defined to be the 3-tiered architecture Gat, 1998. From the theoretical side, the formulation of a Markov Decision Process has presented next, focusing in the existence and uniqueness of the soon to be calculated policy, given the assumptions made about the system. Finally, the formulation of the MDP is presented, with the hypothesis that the behavior of the other road users does not change throughout the execution of the policy, which enabled the definition of the state to contain only the AV's configuration while the other road users' configuration were accounted for in the reward function. The results obtained from the simulation are discussed next, not before the definition of the simulation environment and the value iteration method. The following conclusions could be drawn from such experimentation:

Conclusions from Chapter 3 experiments

- The necessary horizon to sufficiently account for the behaviors of other road users is in direct conflict with the exponential nature of the defined state space.
- The prediction horizon and the AV's velocity also have an antagonistic relationship, given that the transition time remains fixed.
- Or some risk evaluation is necessary to stop the AV from starting a maneuver that ends in an unsafe state for the current prediction horizon or the idea of prediction horizon should be modified to be dependent to the maneuver completion.

Ethical dilemma situations are treated using the methods proposed in chapter 4. First, one needs to define what are dilemma situations, which is done in the first section of chapter 4 to then step into the definition of the parameters used to calculate the harm of a collision, the difference of velocity due to the physical interaction between road users and the vulnerability constant. Using this concepts two types of ethical decision-making procedures are presented: the Ethical Valence theory (EVT), that uses the ethical valence assigned to each road user, in combination to the predicted harm, to deliberate; and the ethical optimization (EO), three different handcrafted minimization procedures that are inspired by the utilitarian, egalitarian and Rawlsian contractarian ethics. Both deliberation methods were tested, with the EVT in an hypothetical solution and assuming a specific definition of features for the ethical valences while the ethical optimization was tested as an MDP criteria for ethical decision, in the same configuration of the tests executed in the previous chapter. In both implementations the algorithm correctly calculated

that the collision with the opposite vehicle would be too dangerous, and opted for the collision with the pedestrian or for the collision with the wall in the frontier of the simulated environment.

Conclusions from Chapter 4 experiments

- EVT: According to the moral profile chosen, the deliberation system protected or the AV's passengers or the most vulnerable users.
- EO: The three actions produced the same result, a collision with the pedestrian, but with different intensities; the utilitarian approach gives the smallest collision velocity for the AV with the pedestrian.

The main subject of the chapter 5 revolves around the relaxation of the assumption made in chapter 3, that the behavior of other road users did not change during the MDP policy execution. To eliminate such assumption it is necessary to account the interaction between each road user, which entails the estimation of their motivations, their intent. To entry in the matter at hand, the types of errors that an AV is subject to are discussed, so as to isolate the error that might produce an ethical dilemma situation, when other users act in an unforeseen way from the AV's point of view. From this point an deterministic decision model for vehicles and pedestrians is proposed, making possible the implementation of a multi-agent simulation and allowing the AV to predict what will be the answer of the other road users given some interaction between them. Using this prediction the AV can test which intent describes the behavior of the other users the best, according to this deterministic decision model, and then calculate the probability that some road user has a determined intent given by the distance between the prediction made by the Kalman filter and the observation.

This is only enough to estimate the intent of each road user, not to determine which action it will execute next. For such task it is necessary to apply an game theory model of the interaction, more specifically the interaction at a precise time is modeled as an incomplete game, with the intent of the other agents, that are estimated by the previous algorithm, as the unknown information. Calculating the Nash equilibrium point of the equivalent Bayesian game then gives the adopted strategy for each of the agents in the environment. However, the algorithm described only accounts for pure strategies, which does not guarantee the existence of the Nash equilibrium. To allow mixed strategies as an end result of the procedure, it is necessary to modify the intent estimation process so as to account multiple strategies for a single intent being compared, and then also expand the Nash equilibrium determination procedure as well.

Using this procedure of intent estimation allows the correct evaluation of the transition probability for the MDP proposed in chapter 3, which means that a more precise account of the expected harm and the multiple possible outcomes in a dilemma situation can be determined and deliberated upon. Not only this but the simulation itself is closer to reality due to the independence of other road users behaviors from the AV's controller. All could result in an proposition of a decision-making with ethical deliberation capabilities that accounts for behavior uncertainty, an

implementation yet not studied.

6.2 Future research perspectives

Multiple lines of research can be branched out from the work done in this theses. Some of them are presented and commented below, in no specific order:

New decision-making system using an POMDP Given that, in the intent estimation for other road users, their positions are predicted using a Kalman filter there is a probability distribution around the state (which are now defined by the configuration of each road user, including the AV), then it would be more appropriate to consider a partially observable Markov decision process to model the decision-making, instead of the model proposed in chapter 3. Of course, this comes with more complications since it is harder to calculate the optimal policy for such cases, often an approximate solution suffices due to the computational cost involved to find the optimal solution. However, the same problems observed in the MDP implementation still need to be solved, notably the antagonistic relationship between the predicted horizon and the capacity to account to other road users behavior (which can be better answered by a state space sampling methods, as the one that Kurniawati and Yadav, 2016 proposes) and the addition of some measure to verify that a maneuver, when started, is at least simulated until its end. In this case the structure of ethical and normal decision making deliberation retains the same functionality as before, if a dilemma situation is detected then the ethical deliberation takes place and a highly negative reward is attached to the situation.

Definition of other road users' behavior using machine learning In chapter 5 the decision process of other vehicles and pedestrians were defined in a deterministic fashion by decision trees. One improvement that could be made is to learn a driver or a pedestrian behavior from real data using a machine learning approach, as Kuderer et al., 2015 did. The resulting behavioral model could produce simulated agents that reproduce more realistically the dynamics of driving, for example acceleration, deceleration, respect to the traffic code, etc. Such method would also allow the test of multiple types of driving profiles, from a more cautious to a more aggressive driver, with the same being valid for the pedestrian. Specifically for pedestrians, there is an entire field of research about the prediction of intentions of pedestrians using its pose, head orientation and position. Certainly the capabilities of the intent estimation would be improved with more realistic methods.

Fusion between the normal reward criteria and the ethical deliberation The ethical deliberation proposed in chapter 4 are only used if a dilemma situation is detected, which might not necessarily happen for every real dilemma situation. Detection of some phenomena almost always has to deal with false negatives instances of detection and given the gravity of a dilemma situation even the existence of a false negative must be avoided at all times.

Therefore, it would be more suitable to fuse both objective and ethical criteria into one hybrid deliberation process. Both components would be accounted at every action taken, with one predominant in normal situations and the other in ethical dilemmas, so as to minimize the loss of performance given that the other component could influence the system towards non-optimal decisions (in the context of a non-dilemma situation) and the same during ethical deliberation. Another motivation for this modification is that up until this point, outside the dilemma situation, the selection criteria for action is entirely based on an objective regard of the situation, but an argument can be made that some instances in normal driving should involve an ethical component for an AV (even if for a normal driver this is not the case), so as to encompass possible passive risks due to each AV's action. For example, if there is a cyclist in the same lane as the AV, should it overtake him? From a performance point of view, yes, but the increase of proximity between both agents would mean that the risk of something go wrong and this vulnerable road user end up entangled in a collision increases. Such fact ideally should also be accounted, which is possible using a hybrid decision process.

Exploration of a virtue ethics approach for ethical learning As established in chapter 2, virtue ethics says that morally good actions flows from a good moral character, which is composed by the correct virtues. Implement such an approach in a machine is not as straightforward as is for deontology or consequentialist based methods, since some sort of character needs to be established and than it must evolve through practice, so as to learn to act ethically. The reason why this is proposed here as a possible research perspective is that such system would be interesting to define the weight of each component in a hybrid decision criteria. Using the framework proposed in Arkin and Ulam, 2009 as a model (the emotions considered in the publication is exchanged by the virtues), one could simulate different outcomes for many dilemma situations and assess the consequences of each action to then backtrack to modify the virtue so as to prevent some actions to be execute while allowing others to. The main question is how to represent a virtue, and to be sure that it represents the same as we, humans, think it should. Or even some artificial virtues could be proposed specifically to produce outcomes that could be considered morally correct.

Verification and validation of stochastic decision-making systems One of the key problems with stochastic models of decision is that certainty cannot be guaranteed. So, to evaluate the performance of some system and to check its safety operational properties different indexes are needed. The necessity of such tools increase with the critical aspect of the system, which means that decision-making algorithms for automated vehicles should pass through verification and validation (V&V) using strict measures of performance and safety. One example of V&V for AVs is Barbier et al., 2018 using KPI to evaluate a POMDP performance.

Real-time implementation and test with a real AV To target a real testing session of the algorithm some important modifications are necessary. The first one is the addition of the perception procedure and the delays involved in the extraction of information before the start of the deliberation process. Also, the measurement uncertainties must be combined with the estimation procedures proposed here. The second question to be treated is the

possible real-time execution of the deliberation. Given the computational effort necessary, the limitations of an embedded system (even if it is a vehicle, it has a far less limitation than a drone for example) and the already observed computation times it is fairly possible that real-time performance is off the table. Nonetheless improvements in calculation effort should be made to increase the responsiveness of the AV. One possible solution is the implementation of parts of the algorithm using GPU, that can massively decrease the processing time of highly parallel tasks. Considering only the evaluation of the reward in chapter 3, it could be parallelized since each calculation is independent from each other. Another algorithm that could take advantage of such operation is the intention estimation nested particle filter structure.

Appendix A

Simulation set-up

To test the algorithms proposed in this thesis, the simulator called WeBots was used. It is a 3D simulation environment where dynamic simulations can be done using any type of object, the already available modeled ones or personalized structures created using the constructs available. Most of its applications are in the field of robotics, as for example industrial robots like the UR3, UR5 and UR10 arm robots from Universal Robots, Spot, the dog-like robot from Boston Dynamics and Nao, the humanoid robot from Aldebaran Robotics. It also has pre-modeled vehicles, like the Tesla model 3 and many others. In this appendix it will be presented the main structure of simulation, details about the control of vehicles and pedestrians and the modifications made in the simulator to unlock new capabilities that were necessary for the simulations proposed in the thesis.

A.1 Webots simulation structure

The simulation structure is similar to other simulators, offering as the main representative entity nodes, that are any type of object in the 3D scene. In the AV context, the road environment in the experiments described at chapters three, four and five was constructed with pieces of straight roads and an intersection, as the figure A.1 shows. For the physical interaction between agents, the Open Dynamics Engine (ODE), an open-source library, is used to calculate rigid bodies dynamics.

The behavior of objects can be controlled externally using a controller, which can be a program written in Python, C++, Matlab or Java. The controllers used, one for the vehicles¹

¹The AV uses the same control functionalities described here combined with the algorithms proposed throughout the thesis.

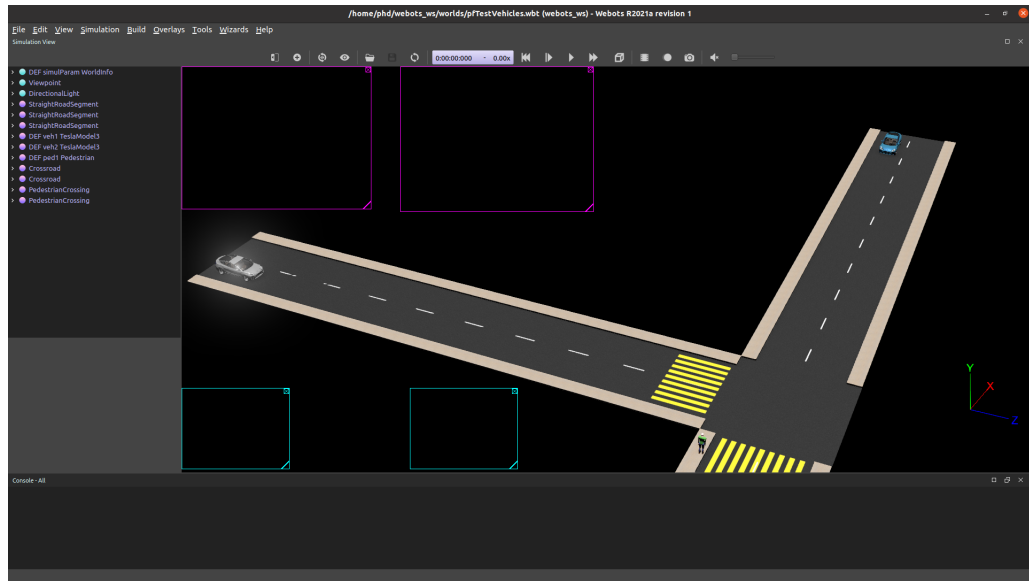


Figure A.1 – Main window of the simulator

and another for the pedestrian, were implemented in C++. It is also possible to use perception sensors, cameras, LiDARs, radars and wireless communications, but given that in a simulation it is possible to find the positions and velocities of all moving objects, this was the favored option as a method to determine the inputs of the decision-making. Also, the implementation of perception routines would add a complexity level that is outside the scope of this thesis (although it is an important component that should be considered in a later time).

The controllers for both vehicles and the pedestrians have one component in common: the interface between the control functions and the WeBots objects. This concerns uniquely the observation acquisition, since the control reference dispatching is specific according to the controller (and it will be detailed later on). For the observation acquisition, each object being controlled can access the main node tree of the simulation, where every object in the scene have a respective node, and pull the position and velocities for each moving object at each given time.

Every simulation done in this document uses the time step of $10ms$, i.e. the frequency in which the simulator calculates all the physical reactions and updates the positions and velocities from the objects accordingly. Each time step happens synchronously; after the dispatch of a control reference each object executes a function called `step`, that signals to the simulator that it can execute another iteration. If there are other objects that did not yet execute this function, the controller routine is blocked in this function until all other controllers call the same function from their side. Then, after every controller signaled the simulator it calculates another step and unblocks all the controllers.

In the next sections the particularities about each of the controllers used will be explained,

and in the last section the modification of one of the simulator libraries, that consist in the creation of one special function to control the car-like vehicle and a modification of the pedestrian model, will be treated.

A.2 Controlling a car-like vehicle

One of the implemented controllers concerns the car-like vehicles. Apart from the observation acquisition already described, it builds upon the capabilities of the WeBots libraries called Driver and Car, the former providing all the functionalities that a human driver usually has in a vehicle, including functions to steer and to accelerate/brake, and the latter completing the former library with functions that are not usually available to the common driver, for example to change the blinking period of the indicator or getting physical parameters of the car.

Two types of longitudinal control are available: speed mode and dynamic mode. The first one emulates a cruising control system, increasing and decreasing the velocity of the vehicle according to a velocity reference, using a constant acceleration value. In this mode the throttle and the brake are not directly used to control the vehicle. This was the function used to change the vehicle's velocity during the simulations for chapters 3, 4 and 5 (this last chapter has some particularities that will be commented in the last section). As for the dynamic mode, it allows to control the vehicle with the throttle, the gears and the braking directly, according to the type of engine of the chosen vehicle. Electric vehicles like the Tesla have only two gears, in a 1+1 configuration (forward and backward), while the combustion vehicles have six or seven gears, in a 5(6)+1 configuration. Given the needs and limitations of the simulations developed throughout the thesis the speed control was enough to model the vehicles' behavior.

Velocity, acceleration and steering selection

As said in chapter 3, the steering and the velocity of the vehicle are defined by the action selected by the policy being executed. The pair steering angle / velocity is directly sent to the controller using the respective functions from the Car library with the speed mode activated. This produces a cruise control-like behavior, with a constant fixed acceleration. Steering in this case is managed by the MDP, given that the reward of each couple state / action depends on the reference trajectory.

For the simulations in chapter 5, the Stanley controller, given by equation 5.1, takes care of defining the steering angle to correct the vehicle's trajectory towards the reference, both for the AV and for the other vehicle. For the velocity, it is given according to the strategy chosen, and it is applied to the vehicle using the cruising control from the speed mode, but this time with a variable acceleration between iterations. In figure A.2, each point of a hypothetical trajectory is

displayed; for each point the controller calculates a velocity limit, that depends on the curvature and on the velocity limit imposed by the traffic code considered, together with an acceleration to be used between the current and the next point, at least until next iteration.

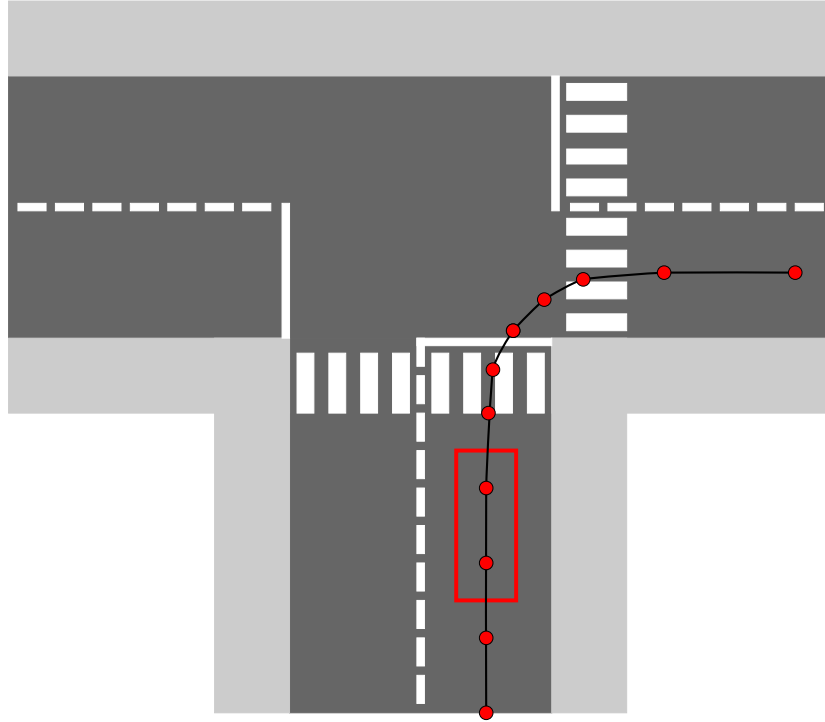


Figure A.2 – Longitudinal control for the vehicle

For all trajectories, the curve trajectory was obtained with a 3rd order Bézier curve, which was stitched into the two straight line trajectories at the middle of the road segments. Using this curvature, the maximal velocity was calculated, given the friction limits of the vehicle, by equation A.1, where μ is the tire's friction coefficient, g is gravity and κ the curvature at the trajectory point being considered. This expression comes from the equality between the centripetal force and the lateral friction, simplified so as to be equal to the maximum longitudinal friction of only one tire where the correct would be to have a dynamic model capable to determine the lateral force in all four tires given the dynamics of the car's body and the slippage influence in the friction force. But since the vehicle model used was kinematic, this approximation was applied, which did not result in slippage during the turning operation.

$$v_{l,\max} = \sqrt{\frac{g\mu}{\kappa}} \quad (\text{A.1})$$

The maximal velocity to execute a turn operation was determined to be 5.41m/s , close to

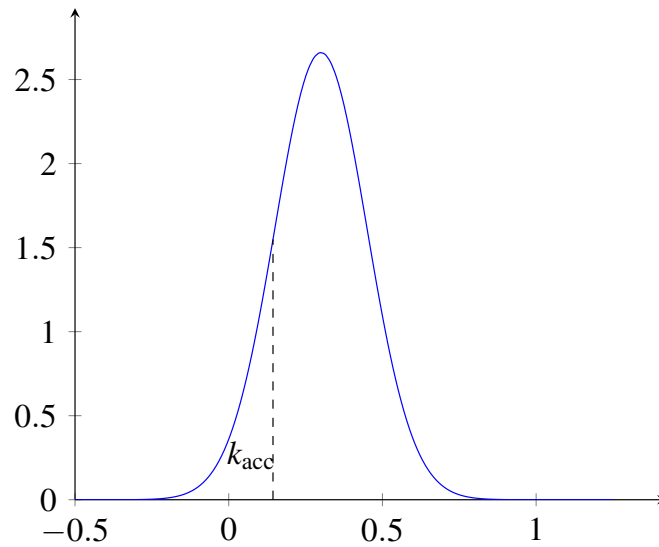


Figure A.3 – Vehicle's acceleration sampling distribution

20km/h, which is a reasonable speed to execute a 90° curve. With this velocity determined, the adjustment for all the trajectory points is done modifying the velocity in the points located before the curve start, to arrive in it with the correct velocity. In the straight sectors the maximal velocity is equal to the velocity limit established by the implementation of the traffic rule, 40km/h. As for the longitudinal acceleration, a normal distribution given by the figure A.3 is used to define the factor which multiplies the maximal acceleration from the vehicle used, the Tesla Model 3. Its maximal acceleration is 6.945 m/s^2 , which is then multiplied by the value sampled from a normal distribution defined as $\mathcal{N}(0.3, 0.15)$.

But since only positive velocities are used the part of the Gaussian that refers to negative values needs to be disregarded. Adding to this the fact that a minimum acceleration is necessary results in the definition of the constant k_{acc} , setting the minimal acceleration to be 1 m/s^2 , which has a probability of approximately 15% of being drawn (corresponds to the left area of the dashed line in figure A.3). This sampling is useful to test multiple accelerations profiles, and once defined this acceleration is used throughout the simulation, it is not changed depending on the situations like the deceleration. Next, each of the available strategies for the vehicle at the chapter 5 will be detailed.

Cruise, caution and follow strategies

The cruise strategy simply adopts the velocity limit given by the traffic code (or the maximal velocity in curves, $v_{l,max}$) as the reference for the cruising control. If the vehicle has a higher velocity before the intersection, it calculates how soon it needs to decelerate to enter the

turning trajectory with the correct velocity. The same value to accelerate is used to decelerate initially, and if it is not enough to arrive at the curve's entry point with the correct velocity, then the necessary deceleration is calculated using the Torricelli's equation (A.2), that determines the resulting velocity $v_{f,x}$ according to the initial one $v_{i,x}$, the acceleration (or deceleration) considered a_x and the distance between initial and final points, Δs_x , all in one direction.

$$v_{f,x}^2 = v_{i,x}^2 + 2a_x \cdot \Delta s_x \quad (\text{A.2})$$

The caution strategy cuts in half the velocity of the entire trajectory while it is being used, including in curves. And to execute the follow strategy, a vehicle needs to equate its velocity to the one of the vehicle in front, hence the necessary deceleration is again calculated using equation A.2 to achieve the correct velocity given the distance between both vehicles. This behavior, if applied once and not updated during long stretches of time can cause a collision (which then would result in a stop strategy), but it is refreshed at every iteration so if the vehicle in front reduces its velocity the deceleration is again recalculated, allowing the maintenance of a safe buffer distance.

Stop, emergency stop and swerve strategies

Two situations can trigger a stop strategy: when a collision is detected and when the vehicle does not have the priority to access the intersection. In the first case, using the position in which the collision was predicted, the deceleration necessary to stop 5 m from this position is calculated, again using equation A.2. If the deceleration necessary is higher than the one available (and since the speed mode is being used the maximal deceleration possible is equal to the max. acceleration, 6.945 m/s^2) then the emergency stop is activated. It simply consists of applying the maximal deceleration until the vehicle stops or when the collision is no longer predicted.

In the second case, given that the environment is always perfectly observable, i.e. the road graph structure illustrated in 5.2 is always available, so it is always possible to stop before an intersection. If a vehicle perceives that another is driving to the intersection and that it has the priority, then the stop procedure starts, with a calculated deceleration until the entry point of the intersection.

The swerve strategy usually refers to the capacity of the vehicle to accelerate and change direction to avoid a collision that usually cannot be avoided by braking². Given the environment used, the strategy of changing direction is never used, and therefore it was not implemented. On the other hand, if the stop and the emergency stop are not enough, then an increase of velocity is tested; if it works, it is adopted as a swerve, if not then the emergency stop strategy is used.

²If one assumes that the increasing the velocity in a dangerous situation is not allowed, so as to mitigate a possible collision severity, the swerve can only happen when braking is not enough to avoid an accident.

A.3 Controlling the pedestrian

Differently from the vehicle case, there is not a special library with control functions for pedestrians, but there is an example of a controller available as an application example preinstalled that makes the pedestrian "walk". The controller implemented then uses some parameters from this application, notably the joint angles. The pedestrian object (figure A.4) in the simulator is defined by thirteen joints:

- Head angle
- Left shoulder angle, left elbow angle, left hand angle.
- Right shoulder angle, right elbow angle, right hand angle.
- Left hip angle, left knee angle, left foot angle.
- Right hip angle, right knee angle, rights foot angle.

Each one of the joints has eight angle values that are used to give a sense that the pedestrian is moving, when it is actually being displaced at the same time that its joint angles are being modified. One can even see in figure A.4 that the foot of the pedestrian penetrates the geometric representation of the sidewalk, since the physical interaction between them is not enabled. These joint angles function in a circular fashion, after eight iterations it start again but in a value that is the natural continuation of the last, creating the illusion of a fluid movement.

As for the pedestrian control, it is rather straightforward; its velocity is changed as it was a point of mass and the same is valid for its direction. The only feature implemented was the timer to start, where the controller only starts the movement after certain amount of time.

The trajectory of the pedestrian is defined by five points in simulation (but it could be used many points as needed), that are executed linearly. Its orientation is defined to be the same as the segment from its position towards the next trajectory point. As it was said in chapter 5, danger for a pedestrian is defined as a situation when the vehicle is closer than five meters, triggering a change of strategy from walk to stop or to run. Walk is the behavior mentioned until now, the execution of the trajectory in a linear piece-wise fashion, with a constant velocity of 1.5 m/s , while the stop strategy is characterized simply by the instantaneous change in velocity to zero. And the run strategy multiplies the pedestrian velocity by two, to 3 m/s .

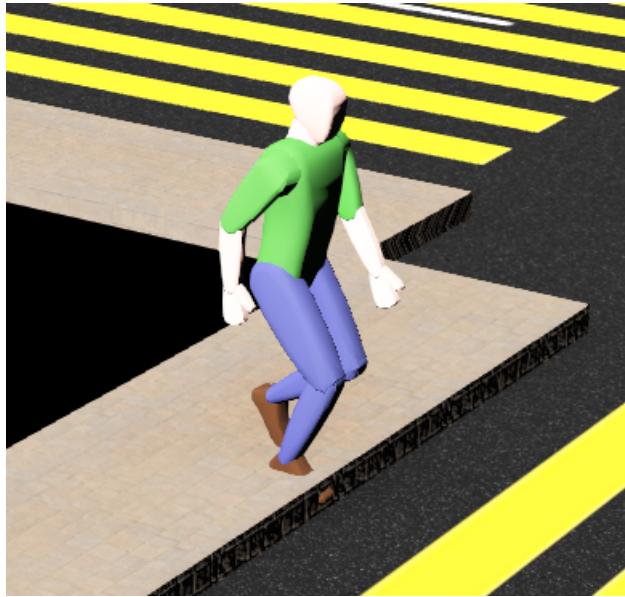


Figure A.4 – The pedestrian object

A.4 Necessary modifications

During the explanation of the control modes for the vehicle it was said that for the speed mode, which is the one used to control the vehicle in all simulations, the acceleration used by the cruising control is fixed to be the maximal acceleration possible given the car model used. However, the strategies for the vehicle need to change its acceleration to brake under some maximal distance and it is also necessary to change the acceleration used in the cruising control so as to apply the sampled one, for the simulations done in chapter 5. To create this functionality a modification was made to the Driver library.

All libraries for WeBots are implemented in C, to then be referenced by an C++ interface. In the Driver C file, a call to a new function, that changes the acceleration of the angular motor that controls the tires depending on the desired linear acceleration (or deceleration) for the vehicle, makes the acceleration change possible. With the library compiled, the effects of the changes in acceleration tested were correctly predicted by equations 3.10.

Another modification done was related to the capacity of the objects to access the node tree to retrieve positions and velocities from the other objects in the simulated scene. To be able to access such node tree, a robot should have the status of supervisor. For the vehicle this option is already available at the simulator interface (which means that it is listed in the parameters file of this particular car model), but not for the pedestrian. Figure A.5 shows the pedestrian node with the already modified field.

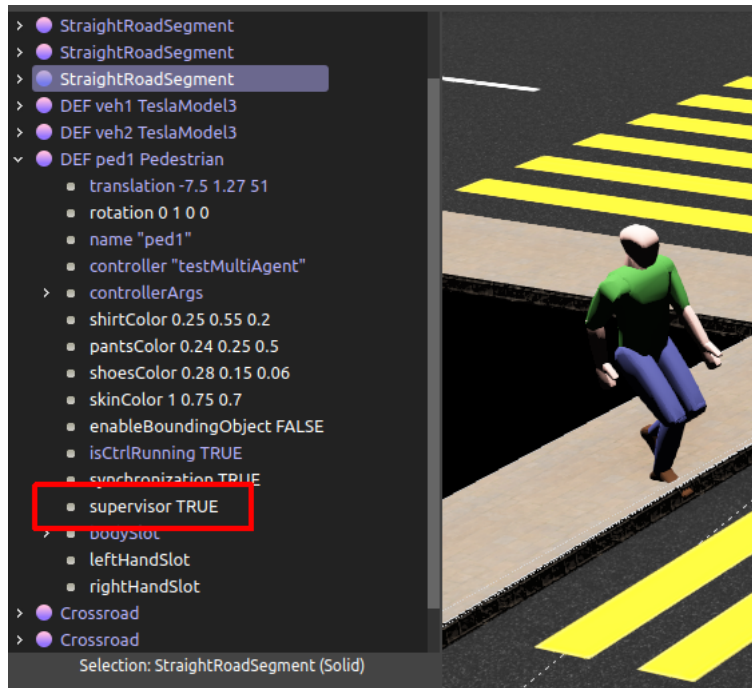


Figure A.5 – Pedestrian supervisor field

To enable this option one entry was added in the PROTO file of the pedestrian model, which is the file that lists all existent parameters of an object, including motors' parameters, colors of the object's parts, name of the object, initial joint values, etc. In this way the pedestrian can obtain the observation from the environment the same way that the vehicle does, directly from the simulator. Apart from these two modifications, the simulator is used as it is available in the WeBots site.

Bibliography

- AAA (2019). *Automatic emergency braking with pedestrian detection*. Report. American Automotive Association. URL: <https://newsroom.aaa.com/wp-content/uploads/2020/11/Pedestrian-Detection-Full-Research-Report-Oct-2019.pdf> (visited on 03/01/2021).
- Ahmed, Sarfraz, M. Nazmul Huda, Sujan Rajbhandari, Chitta Saha, Mark Elshaw, and Stratis Kanarachos (2019). “Pedestrian and Cyclist Detection and Intent Estimation for Autonomous Vehicles: A Survey”. In: vol. 9. 11. MDPI. DOI: 10.3390/app9112335.
- Alahi, A., K. Goel, V. Ramanathan, A. Robicquet, L. Fei-Fei, and S. Savarese (June 2016). “Social LSTM: Human Trajectory Prediction in Crowded Spaces”. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 961–971. DOI: 10.1109/CVPR.2016.110.
- Alexander, Larry and Michael Moore (2020). *Stanford Encyclopedia of Philosophy: Deontological Ethics*. Stanford University. URL: <https://plato.stanford.edu/entries/ethics-deontological/> (visited on 05/08/2021).
- Allen, Colin, Gary Varner, and Jason Zinser (2000). “Prolegomena to any future artificial moral agent”. In: *Journal of Experimental & Theoretical Artificial Intelligence* 12.3, pp. 251–261. DOI: 10.1080/09528130050111428. URL: <https://doi.org/10.1080/09528130050111428>.
- Anderson, James M., Nidhi Kalra, Karlyn D. Stanley, Paul Sorensen, Constantine Samaras, and Tobi A. Oluwatola (2016). *Autonomous Vehicle Technology: A Guide for Policymakers*. Santa Monica, CA: RAND Corporation. DOI: 10.7249/RR443-2.
- Anderson, Michael and Susan Leigh Anderson (Dec. 2007). “Machine Ethics: Creating an Ethical Intelligent Agent”. In: *AI Magazine* 28.4, p. 15. DOI: 10.1609/aimag.v28i4.2065. URL: <https://ojs.aaai.org/index.php/aimagazine/article/view/2065>.
- Anderson, Michael, Susan Leigh Anderson, and Chris Armen (2005). “MedEthEx: Toward a Medical Ethics Advisor.” In: *AAAI Fall Symposium: Caring Machines*, pp. 9–16.
- Arkin, Ronald C, Patrick Ulam, and Brittany Duncan (2009). *An ethical governor for constraining lethal action in an autonomous system*. Tech. rep. Georgia Institute of Technology.

- Arkin, Ronald C. (1990). “Integrating behavioral, perceptual, and world knowledge in reactive navigation”. In: *Robotics and Autonomous Systems* 6.1. Designing Autonomous Agents, pp. 105–122. ISSN: 0921-8890. DOI: [https://doi.org/10.1016/S0921-8890\(05\)80031-4](https://doi.org/10.1016/S0921-8890(05)80031-4). URL: <https://www.sciencedirect.com/science/article/pii/S0921889005800314>.
- Arkin, Ronald C. and Patrick Ulam (2009). “An ethical adaptor: Behavioral modification derived from moral emotions”. In: *2009 IEEE International Symposium on Computational Intelligence in Robotics and Automation - (CIRA)*, pp. 381–387. DOI: 10.1109/CIRA.2009.5423177.
- Arkin, Ronald Craig, Patrick Ulam, and Alan R. Wagner (2012). “Moral Decision Making in Autonomous Systems: Enforcement, Moral Emotions, Dignity, Trust, and Deception”. In: *Proceedings of the IEEE* 100.3, pp. 571–589. DOI: 10.1109/JPROC.2011.2173265.
- Awad, Edmond, Sohan Dsouza, Richard Kim, Jonathan Schulz, Joseph Henrich, Azim Shariff, Jean-François Bonnefon, and Iyad Rahwan (Aug. 2018). “The Moral Machine experiment”. In: *Nature* 563.7729, pp. 59–64. ISSN: 1476-4687. DOI: 10.1038/s41586-018-0637-6. URL: <https://doi.org/10.1038/s41586-018-0637-6>.
- Bahouth, George, Jill Graygo, Kennerly Digges, Carl Schulman, and Peter Baur (2014). “The Benefits and Tradeoffs for Varied High-Severity Injury Risk Thresholds for Advanced Automatic Crash Notification Systems”. In: *Traffic Injury Prevention* 15.sup1. PMID: 25307378, S134–S140. DOI: 10.1080/15389588.2014.936011. URL: <https://doi.org/10.1080/15389588.2014.936011>.
- Barbier, Mathieu, Christian Laugier, Olivier Simonin, and Javier Ibañez-Guzmán (Nov. 2018). “Probabilistic Decision-Making at Road Intersections: Formulation and Quantitative Evaluation”. In: *ICARCV 2018 - 15th International Conference on Control, Automation, Robotics and Vision*. Singapur, Singapore, pp. 1–8. URL: <https://hal.inria.fr/hal-01940392>.
- Barredo Arrieta, Alejandro, Natalia Díaz-Rodríguez, Javier Del Ser, Adrien Bennetot, Siham Tabik, Alberto Barbado, Salvador Garcia, Sergio Gil-Lopez, Daniel Molina, Richard Benjamins, Raja Chatila, and Francisco Herrera (2020). “Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI”. In: *Information Fusion* 58, pp. 82–115. ISSN: 1566-2535. DOI: <https://doi.org/10.1016/j.inffus.2019.12.012>. URL: <https://www.sciencedirect.com/science/article/pii/S1566253519308103>.
- Barry, Keith (May 22, 2019). *Tesla’s Updated Navigate on Autopilot Requires Significant Driver Intervention*. Consumer Reports. URL: <https://www.consumerreports.org/autonomous-driving/tesla-navigate-on-autopilot-automatic-lane-change-requires-significant-driver-intervention/> (visited on 03/04/2021).
- Batz, T., K. Watson, and J. Beyerer (June 2009). “Recognition of dangerous situations within a cooperative group of vehicles”. In: *2009 IEEE Intelligent Vehicles Symposium*, pp. 907–912. DOI: 10.1109/IVS.2009.5164400.

- BBC (Sept. 12, 2017). *Tesla Autopilot 'partly to blame' for crash*. British Broadcast Company. URL: <https://www.bbc.com/news/technology-41242884> (visited on 03/01/2021).
- Bellman, Richard Ernst (1978). *An Introduction to Artificial Intelligence: Can Computers Think?* Boyd & Fraser Publishing Company.
- Bensimhou, Michael (2009). “N-dimensional cumulative function, and other useful facts about gaussians and normal densities”. In: *Jerusalem, Israel, Tech. Rep*, pp. 1–8.
- Berrada, Jaâfar, Ilyes Mouhoubi, and Zoi Christoforou (2020). “Factors of successful implementation and diffusion of services based on autonomous vehicles: users’ acceptance and operators’ profitability”. In: *Research in Transportation Economics* 83. Thredbo 16 conference, p. 100902. ISSN: 0739-8859. DOI: <https://doi.org/10.1016/j.retrec.2020.100902>. URL: <https://www.sciencedirect.com/science/article/pii/S0739885920301001>.
- Berreby, Fiona, Gauvain Bourgne, and Jean-Gabriel Ganascia (2015). “Modelling moral reasoning and ethical responsibility with logic programming”. In: *Logic for programming, artificial intelligence, and reasoning*. Springer, pp. 532–548.
- Blanco, Myra, Jon Atwood, Sheldon M Russell, Tammy Trimble, Julie A McClafferty, and Miguel A Perez (2016). *Automated vehicle crash rate comparison using naturalistic data*. Tech. rep. Virginia Tech Transportation Institute. URL: <https://vtechworks.lib.vt.edu/handle/10919/64420> (visited on 03/08/2021).
- Bonnefon, Jean-François, David Černý, John Danaher, Nathalie Devillier, Veronica Johansson, Tatiana Kovacicova, Marieke Martens, Milos Mladenovic, Paula Palade, Nick Reed, et al. (2020). *Ethics of Connected and Automated Vehicles: Recommendations on road safety, privacy, fairness, explainability and responsibility*. Tech. rep. European Commission - Expert Group on Ethics of Connected and Automated Vehicles. DOI: 10.2777/035239. URL: https://ec.europa.eu/info/news/new-recommendations-for-a-safe-and-ethical-transition-towards-driverless-mobility-2020-sep-18_en (visited on 03/08/2021).
- Bonnemains, Vincent, Claire Saurel, and Catherine Tessier (2018). “Embedded ethics: some technical and ethical challenges”. In: *Ethics and Information Technology* 20.1, pp. 41–58.
- Bonnin, S., F. Kummert, and J. Schmüdderich (Sept. 2012). “A Generic Concept of a System for Predicting Driving Behaviors”. In: *2012 15th International IEEE Conference on Intelligent Transportation Systems*, pp. 1803–1808. DOI: 10.1109/ITSC.2012.6338695.
- Boudette, Neal E. (July 17, 2019). *Despite High Hopes, Self-Driving Cars Are 'Way in the Future'*. The New York Times. URL: <https://www.nytimes.com/2019/07/17/business/self-driving-autonomous-cars.html> (visited on 03/08/2021).
- (Mar. 23, 2021). *Tesla's Autopilot Technology Faces Fresh Scrutiny*. The New York Times. URL: <https://www.nytimes.com/2021/03/23/business/teslas-autopilot-safety-investigations.html?smid=url-share> (visited on 03/23/2021).

- Boyd, Stephen and Lieven Vandenberghe (2004). *Convex Optimization*. USA: Cambridge University Press. ISBN: 0521833787.
- Brach, Raymond M and R Matthew Brach (2011). *Vehicle accident analysis and reconstruction methods*. 2nd. SAE International.
- Brännström, M., E. Coelingh, and J. Sjöberg (2010). “Model-Based Threat Assessment for Avoiding Arbitrary Vehicle Collisions”. In: *IEEE Transactions on Intelligent Transportation Systems* 11.3, pp. 658–669. DOI: 10.1109/TITS.2010.2048314.
- Brauckmann, Michael E, Christian Goerick, Jürgen Gross, and Thomas Zielke (1994). “Towards all around automatic visual obstacle sensing for cars”. In: *Proceedings of the Intelligent Vehicles’ 94 Symposium*. IEEE, pp. 79–84.
- Brechtel, Sebastian, Tobias Gindele, and Rüdiger Dillmann (2011). “Probabilistic MDP-behavior planning for cars”. In: *2011 14th International IEEE Conference on Intelligent Transportation Systems (ITSC)*. IEEE, pp. 1537–1542.
- (2013). “Solving Continuous POMDPs: Value Iteration with Incremental Learning of an Efficient Space Representation”. In: *Proceedings of the 30th International Conference on International Conference on Machine Learning - Volume 28*. ICML’13. Atlanta, GA, USA: JMLR.org, III–370–III–378.
- (2014). “Probabilistic decision-making under uncertainty for autonomous driving using continuous POMDPs”. In: *17th International IEEE Conference on Intelligent Transportation Systems (ITSC)*. IEEE, pp. 392–399.
- Breuer, Joerg J (1998). “Analysis of driver-vehicle-interactions in an evasive manoeuvre-results of ‘moose test’ studies”. In: *Proc. 16th ESV Conf., Paper*. 98-S2. NHTSA.
- Brink, David (2018). *Mill’s Moral and Political Philosophy*. Stanford University. URL: <https://plato.stanford.edu/entries/mill-moral-political/> (visited on 03/31/2021).
- Broadhurst, A., S. Baker, and T. Kanade (2005). “Monte Carlo road safety reasoning”. In: *IEEE Proceedings. Intelligent Vehicles Symposium, 2005*. Pp. 319–324. DOI: 10.1109/IVS.2005.1505122.
- Broggi, A., P. Cerri, S. Debattisti, M. C. Laghi, P. Medici, D. Molinari, M. Panciroli, and A. Prioletti (2015). “PROUD—Public Road Urban Driverless-Car Test”. In: *IEEE Transactions on Intelligent Transportation Systems* 16.6, pp. 3508–3519. DOI: 10.1109/TITS.2015.2477556.
- Broggi, Alberto, Massimo Bertozzi, Alessandra Fascioli, C Guarino Lo Bianco, and Aurelio Piazzi (1999). “The ARGO autonomous vehicle’s vision and control systems”. In: *International Journal of Intelligent Control and Systems* 3.4, pp. 409–441.
- Brooks, Rodney A (1986). “A robust layered control system for a mobile robot”. In: *IEEE journal on robotics and automation* 2.1, pp. 14–23.

- (1991). “Intelligence without reason”. In: *Artificial intelligence: critical concepts* 3, pp. 107–63.
- (1989). “A Robot that Walks; Emergent Behaviors from a Carefully Evolved Network”. In: *Neural Computation* 1.2, pp. 253–262. DOI: 10.1162/neco.1989.1.2.253. eprint: <https://doi.org/10.1162/neco.1989.1.2.253>. URL: <https://doi.org/10.1162/neco.1989.1.2.253>.
- Broome, John (2017). “Equality”. In: *Weighing Goods*. John Wiley & Sons, Ltd. Chap. 9, pp. 174–201. ISBN: 9781119451266. DOI: <https://doi.org/10.1002/9781119451266.ch9>. URL: <https://onlinelibrary.wiley.com/doi/abs/10.1002/9781119451266.ch9>.
- Cho, K., T. Ha, G. Lee, and S. Oh (2019). “Deep Predictive Autonomous Driving Using Multi-Agent Joint Trajectory Prediction and Traffic Rules”. In: *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 2076–2081. DOI: 10.1109/IROS40897.2019.8967708.
- Chokshi, Niraj (Feb. 25, 2020). *Tesla Autopilot System Found Probably at Fault in 2018 Crash*. The New York Times. URL: <https://www.nytimes.com/2020/02/25/business/tesla-autopilot-ntsb.html> (visited on 03/01/2021).
- Christman, John (2008). *Autonomy in moral and political philosophy*. Tech. rep. URL: <https://plato.stanford.edu/entries/autonomy-moral/>.
- Claussmann, Laurene (2019). “Motion planning for autonomous highway driving: a unified architecture for decision-maker and trajectory generator”. PhD thesis. Université Paris-Saclay (ComUE).
- Claussmann, Lauréne, Marie O’Brien, Sébastien Glaser, Homayoun Najjaran, and Dominique Gruyer (2018). “Multi-Criteria Decision Making for Autonomous Vehicles using Fuzzy Dempster-Shafer Reasoning”. In: *2018 IEEE Intelligent Vehicles Symposium (IV)*, pp. 2195–2202. DOI: 10.1109/IVS.2018.8500451.
- Commission, European (2018). *How is your country doing? Road safety in 2018*. Tech. rep. European Union. DOI: 10.2832/39063. URL: <https://op.europa.eu/en/publication-detail/-/publication/043ee22b-643b-11eb-aeb5-01aa75ed71a1> (visited on 03/20/2021).
- Coscia, Pasquale, Francesco Castaldo, Francesco A.N. Palmieri, Alexandre Alahi, Silvio Savarese, and Lamberto Ballan (2018). “Long-term path prediction in urban scenarios using circular distributions”. In: *Image and Vision Computing* 69, pp. 81–91. ISSN: 0262-8856. DOI: <https://doi.org/10.1016/j.imavis.2017.11.006>. URL: <https://www.sciencedirect.com/science/article/pii/S0262885617301853>.
- Daimler (2016). *The PROMETHEUS project launched in 1986: Pioneering autonomous driving*. URL: <https://media.daimler.com/marsMediaSite/en/instance/ko.xhtml?oid=13744534&relId=1001&fromOid=13744534&resultInfoTypeId=172&viewType=>

grid&sortDefinition=PUBLISHED_AT-2&thumbScaleIndex=1&rowCountIndex=5&fromInfoTypeId=40626#prevId=7408570 (visited on 02/24/2021).

- de Moura, N., R. Chatila, K. Evans, S. Chauvier, and E. Dogan (2020). “Ethical decision making for autonomous vehicles”. In: *2020 IEEE Intelligent Vehicles Symposium (IV)*, pp. 2006–2013. DOI: 10.1109/IV47402.2020.9304618.
- Derbel, Oussama and René Landry Jr. (Feb. 2018). “Belief and fuzzy theories for driving behavior assessment in case of accident scenarios”. In: *International Journal of Automotive Technology* 19.1, pp. 167–177. ISSN: 1976-3832. DOI: 10.1007/s12239-018-0016-1. URL: <https://doi.org/10.1007/s12239-018-0016-1>.
- Dickmanns, Ernst D (2002). “The development of machine vision for road vehicles in the last decade”. In: *Intelligent Vehicle Symposium, 2002. IEEE*. Vol. 1. IEEE, pp. 268–281.
- Dickmanns, Ernst D and Alfred Zapp (1987). “Autonomous high speed road vehicle guidance by computer vision”. In: *IFAC Proceedings Volumes* 20.5, pp. 221–226.
- Dickmanns, Ernst Dieter, Reinhold Behringer, Dirk Dickmanns, Thomas Hildebrandt, Markus Maurer, Frank Thomanek, and Joachim Schiehlen (1994). “The seeing passenger car’ VaMoRs-P””. In: *Proceedings of the Intelligent Vehicles’ 94 Symposium*. IEEE, pp. 68–73.
- Dictionary, Oxford English (1989). *Oxford english dictionary*.
- Ding, G., S. Aghli, C. Heckman, and L. Chen (2018). “Game-Theoretic Cooperative Lane Changing Using Data-Driven Models”. In: *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 3640–3647. DOI: 10.1109/IROS.2018.8593725.
- Donges, Edmund (1999). “A Conceptual Framework for Active Safety in Road Traffic”. In: *Vehicle System Dynamics* 32.2-3, pp. 113–128. DOI: 10.1076/vesd.32.2.113.2089. URL: <https://www.tandfonline.com/doi/abs/10.1076/vesd.32.2.113.2089>.
- Driver, Julia (2014). *The history of Utilitarianism*. Stanford University. URL: <https://plato.stanford.edu/entries/utilitarianism-history/> (visited on 03/31/2021).
- Englund, C., L. Chen, J. Ploeg, E. Semsar-Kazerooni, A. Voronov, H. H. Bengtsson, and J. Didoff (2016). “The Grand Cooperative Driving Challenge 2016: boosting the introduction of cooperative automated vehicles”. In: *IEEE Wireless Communications* 23.4, pp. 146–152. DOI: 10.1109/MWC.2016.7553038.
- Evans, Katherine, Nelson de Moura, Stéphane Chauvier, Raja Chatila, and Ebru Dogan (2020). “Ethical Decision Making in Autonomous Vehicles: The AV Ethics Project”. In: *Science and Engineering Ethics* 26 (6), pp. 3285–3312. ISSN: 1471-5546. DOI: 10.1007/s11948-020-00272-8. URL: <https://doi.org/10.1007/s11948-020-00272-8>.
- Evans, Leonard (1994). “Driver injury and fatality risk in two-car crashes versus mass ratio inferred using Newtonian mechanics”. In: *Accident Analysis & Prevention* 26.5, pp. 609–616.
- Fagnant, Daniel J. and Kara Kockelman (2015). “Preparing a nation for autonomous vehicles: opportunities, barriers and policy recommendations”. In: *Transportation Research Part A*:

Policy and Practice 77, pp. 167–181. ISSN: 0965-8564. DOI: <https://doi.org/10.1016/j.tra.2015.04.003>. URL: <https://www.sciencedirect.com/science/article/pii/S0965856415000804>.

- Fikes, Richard E and Nils J Nilsson (1971). “STRIPS: A new approach to the application of theorem proving to problem solving”. In: *Artificial intelligence* 2.3-4, pp. 189–208.
- Fildes, B., M. Keall, N. Bos, A. Lie, Y. Page, C. Pastor, L. Pennisi, M. Rizzi, P. Thomas, and C. Tingvall (2015). “Effectiveness of low speed autonomous emergency braking in real-world rear-end crashes”. In: *Accident Analysis & Prevention* 81, pp. 24–29. ISSN: 0001-4575. DOI: <https://doi.org/10.1016/j.aap.2015.03.029>. URL: <https://www.sciencedirect.com/science/article/pii/S0001457515001116>.
- Firby, Robert James (1989). “Adaptive execution in complex dynamic worlds”. PhD thesis. Yale University.
- Firl, Jonas and Quan Tran (2011). “Probabilistic Maneuver Prediction in Traffic Scenarios”. In: *ECMR*, pp. 89–94.
- Fisac, J. F., E. Bronstein, E. Stefansson, D. Sadigh, S. S. Sastry, and A. D. Dragan (2019). “Hierarchical Game-Theoretic Planning for Autonomous Vehicles”. In: *2019 International Conference on Robotics and Automation (ICRA)*, pp. 9590–9596. DOI: 10.1109/ICRA.2019.8794007.
- Galceran, Enric, Alexander G Cunningham, Ryan M Eustice, and Edwin Olson (2017). “Multi-policy decision-making for autonomous driving via changepoint-based behavior prediction: Theory and experiment”. In: *Autonomous Robots* 41.6, pp. 1367–1382.
- Garcia, Frédéric and Emmanuel Rachelson (2013). “Markov Decision Processes”. In: *Markov Decision Processes in Artificial Intelligence*. John Wiley & Sons, Ltd. Chap. 1, pp. 1–38. ISBN: 9781118557426. DOI: <https://doi.org/10.1002/9781118557426.ch1>. URL: <https://onlinelibrary.wiley.com/doi/abs/10.1002/9781118557426.ch1>.
- Gat, Erann (1991). “Reliable Goal-Directed Reactive Control of Autonomous Mobile Robots”. UMI Order No. GAX91-23728. PhD thesis. USA.
- (1998). “Three-Layer Architectures”. In: *Artificial Intelligence and Mobile Robots: Case Studies of Successful Robot Systems*. Cambridge, MA, USA: MIT Press, pp. 195–210. ISBN: 0262611376.
- Gauthier, J.P., H. Hammouri, and S. Othman (1992). “A simple observer for nonlinear systems applications to bioreactors”. In: *IEEE Transactions on Automatic Control* 37.6, pp. 875–880. DOI: 10.1109/9.256352.
- Gerdes, J. Christian and Sarah M. Thornton (2016). “Implementable Ethics for Autonomous Vehicles”. In: *Autonomous Driving: Technical, Legal and Social Aspects*. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 87–102. ISBN: 978-3-662-48847-8. DOI: 10.1007/978-3-662-48847-8_5. URL: https://doi.org/10.1007/978-3-662-48847-8_5.

- Gindele, T., S. Brechtel, and R. Dillmann (July 2010). “A probabilistic model for estimating driver behaviors and vehicle trajectories in traffic environments”. In: *13th International IEEE Conference on Intelligent Transportation Systems*, pp. 1625–1631. DOI: 10.1109/ITSC.2010.5625262.
- (Oct. 2013). “Learning context sensitive behavior models from observations for predicting traffic situations”. In: *16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013)*, pp. 1764–1771. DOI: 10.1109/ITSC.2013.6728484.
- Gips, James (1995). “Towards the Ethical Robot”. In: *Android Epistemology*. Cambridge, MA, USA: MIT Press, pp. 243–252. ISBN: 0262061848.
- Giralt, Georges, Raja Chatila, and Marc Vaisset (1990). “An Integrated Navigation and Motion Control System for Autonomous Multisensory Mobile Robots”. In: *Autonomous Robot Vehicles*. New York, NY: Springer New York, pp. 420–443. ISBN: 978-1-4613-8997-2. DOI: 10.1007/978-1-4613-8997-2_31. URL: https://doi.org/10.1007/978-1-4613-8997-2_31.
- Godthelp, Hans (1986). “Vehicle Control During Curve Driving”. In: *Human Factors* 28.2. PMID: 3733107, pp. 211–221. DOI: 10.1177/001872088602800209. URL: <https://doi.org/10.1177/001872088602800209>.
- González, David, Joshué Pérez, Vicente Milanés, and Fawzi Nashashibi (2016). “A Review of Motion Planning Techniques for Automated Vehicles”. In: *IEEE Transactions on Intelligent Transportation Systems* 17.4, pp. 1135–1145. DOI: 10.1109/TITS.2015.2498841.
- Goodall, Noah J (2014). “Ethical decision making during automated vehicle crashes”. In: *Transportation Research Record* 2424.1, pp. 58–65.
- Gottschalk, S., M. C. Lin, and D. Manocha (1996). “OBBTree: A Hierarchical Structure for Rapid Interference Detection”. In: *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques*. SIGGRAPH '96. New York, NY, USA: Association for Computing Machinery, pp. 171–180. ISBN: 0897917464. DOI: 10.1145/237170.237244. URL: <https://doi.org/10.1145/237170.237244>.
- Hajek, Alan (2019). *Interpretations of Probability*. Tech. rep. URL: <https://plato.stanford.edu/archives/win2012/entries/probability-interpret/>.
- Harsanyi, John C and Reinhard Selten (1992). *A general theory of equilibrium selection in games*. 2nd. The MIT Press. ISBN: 978-0-262-08173-3.
- Harsanyi, John C. (1967). “Games with Incomplete Information Played by “Bayesian” Players, I-III. Part I. The Basic Model”. In: *Management Science* 14.3, pp. 159–182. ISSN: 00251909, 15265501. URL: <http://www.jstor.org/stable/2628393>.
- (1968a). “Games with Incomplete Information Played by ‘Bayesian’ Players, Part III. The Basic Probability Distribution of the Game”. In: *Management Science* 14.7, pp. 486–502. DOI: 10.1287/mnsc.14.7.486. URL: <https://doi.org/10.1287/mnsc.14.7.486>.

- (1968b). “Games with Incomplete Information Played by "Bayesian" Players, I-III. Part II. Bayesian Equilibrium Points”. In: *Management Science* 14.5, pp. 320–334. ISSN: 00251909, 15265501. URL: <http://www.jstor.org/stable/2628673>.
 - (1976). “Advances in Understanding Rational Behavior”. In: *Essays on Ethics, Social Behavior, and Scientific Explanation*. Dordrecht: Springer Netherlands, pp. 89–117. ISBN: 978-94-010-9327-9. DOI: 10.1007/978-94-010-9327-9_6. URL: https://doi.org/10.1007/978-94-010-9327-9_6.
- Hartley, R. and F. Pipitone (1991). “Experiments with the subsumption architecture”. In: *Proceedings. 1991 IEEE International Conference on Robotics and Automation*, 1652–1658 vol.2. DOI: 10.1109/ROBOT.1991.131856.
- Hoffmann, Gabriel M., Claire J. Tomlin, Michael Montemerlo, and Sebastian Thrun (2007). “Autonomous Automobile Trajectory Tracking for Off-Road Driving: Controller Design, Experimental Validation and Racing”. In: *2007 American Control Conference*, pp. 2296–2301. DOI: 10.1109/ACC.2007.4282788.
- Houenou, A., P. Bonnifait, V. Cherfaoui, and W. Yao (2013). “Vehicle trajectory prediction based on motion model and maneuver recognition”. In: *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 4363–4369. DOI: 10.1109/IRoS.2013.6696982.
- Hubmann, Constantin, Jens Schulz, Marvin Becker, Daniel Althoff, and Christoph Stiller (2018). “Automated driving in uncertain environments: Planning with interaction and uncertain maneuver prediction”. In: *IEEE Transactions on Intelligent Vehicles* 3.1, pp. 5–17.
- Hummel, Britta, Werner Thiemann, and Irina Lulcheva (2008). “Scene Understanding of Urban Road Intersections with Description Logic”. In: *Logic and Probability for Scene Interpretation*. Ed. by Anthony G. Cohn, David C. Hogg, Ralf Möller, and Bernd Neumann. Dagstuhl Seminar Proceedings 08091. Dagstuhl, Germany: Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, Germany. URL: <http://drops.dagstuhl.de/opus/volltexte/2008/1616>.
- Hursthouse, Rosalind and Glen Pettigrove (2016). *Stanford Encyclopedia of Philosophy: Virtue Ethics*. Stanford University. URL: <https://plato.stanford.edu/entries/ethics-deontological/> (visited on 05/08/2021).
- Hussain, Qinaat, Hanqin Feng, Raphael Grzebieta, Tom Brijs, and Jake Olivier (2019). “The relationship between impact speed and the probability of pedestrian fatality during a vehicle-pedestrian crash: a systematic review and meta-analysis”. In: *Accident Analysis & Prevention* 129, pp. 241–249.
- Jaritz, Maximilian, Raoul de Charette, Marin Toromanoff, Etienne Perot, and Fawzi Nashashibi (2018). “End-to-End Race Driving with Deep Reinforcement Learning”. In: *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2070–2075. DOI: 10.1109/ICRA.2018.8460934.

- Ji, A. and D. Levinson (2020). “Estimating the Social Gap With a Game Theory Model of Lane Changing”. In: *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–10. DOI: 10.1109/TITS.2020.2991242.
- Jurewicz, Chris, Amir Sobhani, Jeremy Woolley, Jeff Dutschke, and Bruce Corben (2016). “Exploration of vehicle impact speed–injury severity relationships for application in safer road design”. In: *Transportation research procedia* 14, pp. 4247–4256.
- Kaempchen, N., B. Schiele, and K. Dietmayer (2009). “Situation Assessment of an Autonomous Emergency Brake for Arbitrary Vehicle-to-Vehicle Collision Scenarios”. In: *IEEE Transactions on Intelligent Transportation Systems* 10.4, pp. 678–687. DOI: 10.1109/TITS.2009.2026452.
- Karasev, V., A. Ayvaci, B. Heisele, and S. Soatto (May 2016). “Intent-aware long-term prediction of pedestrian motion”. In: *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2543–2549. DOI: 10.1109/ICRA.2016.7487409.
- Keeling, Geoff, Katherine Evans, Sarah M. Thornton, Giulio Mecacci, and Filippo Santoni de Sio (2019). “Four Perspectives on What Matters for the Ethics of Automated Vehicles”. In: *Road Vehicle Automation 6*. Ed. by Gereon Meyer and Sven Beiker. Cham: Springer International Publishing, pp. 49–60. ISBN: 978-3-030-22933-7.
- Krener, Arthur J. (2003). “The Convergence of the Extended Kalman Filter”. In: *Directions in Mathematical Systems Theory and Optimization*. Ed. by Anders Rantzer and Christopher I. Byrnes. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 173–182. ISBN: 978-3-540-36106-0.
- Kröger, Fabian (2016). “Automated Driving in Its Social, Historical and Cultural Contexts”. In: *Autonomous Driving: Technical, Legal and Social Aspects*. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 41–68. ISBN: 978-3-662-48847-8. DOI: 10.1007/978-3-662-48847-8_3. URL: https://doi.org/10.1007/978-3-662-48847-8_3.
- Kröyer, Höskuldur R.G. (2015). “Is 30km/h a ‘safe’ speed? Injury severity of pedestrians struck by a vehicle and the relation to travel speed and age”. In: *IATSS Research* 39.1, pp. 42–50. ISSN: 0386-1112. DOI: <https://doi.org/10.1016/j.iatssr.2014.08.001>. URL: <https://www.sciencedirect.com/science/article/pii/S0386111214000235>.
- Kuderer, Markus, Shilpa Gulati, and Wolfram Burgard (2015). “Learning driving styles for autonomous vehicles from demonstration”. In: *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2641–2646. DOI: 10.1109/ICRA.2015.7139555.
- Kurniawati, Hanna and Vinay Yadav (2016). “An Online POMDP Solver for Uncertainty Planning in Dynamic Environment”. In: *Robotics Research: The 16th International Symposium ISRR*. Ed. by Masayuki Inaba and Peter Corke. Cham: Springer International Publishing, pp. 611–629. ISBN: 978-3-319-28872-7. DOI: 10.1007/978-3-319-28872-7_35. URL: https://doi.org/10.1007/978-3-319-28872-7_35.

- Leben, Derek (2017). “A Rawlsian algorithm for autonomous vehicles”. In: *Ethics and Information Technology* 19.2, pp. 107–115.
- Leenes, Ronald and Federica Lucivero (2014). “Laws on Robots, Laws by Robots, Laws in Robots: Regulating Robot Behaviour by Design”. In: *Law, Innovation and Technology* 6.2, pp. 193–220. DOI: 10.5235/17579961.6.2.193. URL: <https://doi.org/10.5235/17579961.6.2.193>.
- Lefèvre, S., C. Laugier, and J. Ibañez-Guzmán (June 2012). “Risk assessment at road intersections: Comparing intention and expectation”. In: *2012 IEEE Intelligent Vehicles Symposium*, pp. 165–171. DOI: 10.1109/IVS.2012.6232198.
- Lefèvre, Stéphanie, Dizan Vasquez, and Christian Laugier (2014). “A survey on motion prediction and risk assessment for intelligent vehicles”. In: *ROBOMECH Journal* 1.1. DOI: 10.1186/s40648-014-0001-z.
- Lefkopoulos, V., M. Menner, A. Domahidi, and M. N. Zeilinger (Jan. 2021). “Interaction-Aware Motion Prediction for Autonomous Driving: A Multiple Model Kalman Filtering Scheme”. In: *IEEE Robotics and Automation Letters* 6.1, pp. 80–87. ISSN: 2377-3766. DOI: 10.1109/LRA.2020.3032079.
- Lenz, David, Tobias Kessler, and Alois Knoll (2016). “Tactical cooperative planning for autonomous highway driving using Monte-Carlo Tree Search”. In: *2016 IEEE Intelligent Vehicles Symposium (IV)*, pp. 447–453. DOI: 10.1109/IVS.2016.7535424.
- Lin, Patrick (2016). “Why Ethics Matters for Autonomous Cars”. In: *Autonomous Driving: Technical, Legal and Social Aspects*. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 69–85. ISBN: 978-3-662-48847-8. DOI: 10.1007/978-3-662-48847-8_4. URL: https://doi.org/10.1007/978-3-662-48847-8_4.
- Luetge, Christoph (2017). “The German ethics code for automated and connected driving”. In: *Philosophy & Technology* 30.4, pp. 547–558.
- MacKenzie, Ellen J, Sam Shapiro, and James N Eastham (1985). “The Abbreviated Injury Scale and Injury Severity Score: levels of inter-and intrarater reliability”. In: *Medical care*, pp. 823–835.
- Madigan, Ruth, Sina Nordhoff, Charles Fox, Roja Ezzati Amini, Tyron Louw, Marc Wilbrink, Anna Schieben, and Natasha Merat (2019). “Understanding interactions between Automated Road Transport Systems and other road users: A video analysis”. In: *Transportation Research Part F: Traffic Psychology and Behaviour* 66, pp. 196–213. ISSN: 1369-8478. DOI: <https://doi.org/10.1016/j.trf.2019.09.006>. URL: <https://www.sciencedirect.com/science/article/pii/S1369847819300531>.
- Malczyk, Axel, Gerd Müller, and Tina Gehlert (2012). “The increasing role of SUVs in crash involvement in Germany”. In: *Ireland: IRCOBI*.
- Martin, Jean-Louis and Dan Wu (2018). “Pedestrian fatality and impact speed squared: Cloglog modeling from French national data”. In: *Traffic injury prevention* 19.1, pp. 94–101.

- Mathai, A. M. and S. Provost (1992). *Quadratic forms in random variables : theory and applications*. 1st. Dekker. ISBN: 0-8247-8691-2.
- McCarthy, John and Patrick J Hayes (1969). “Some philosophical problems from the standpoint of artificial intelligence”. In: *Machine Intelligence* 4, pp. 463–504.
- McLaren, Bruce M (2006). “Computational models of ethical reasoning: Challenges, initial steps, and future directions”. In: *IEEE intelligent systems* 21.4, pp. 29–37.
- (2003). “Extensionally defining principles and cases in ethics: An AI model”. In: *Artificial Intelligence* 150.1. AI and Law, pp. 145–181. ISSN: 0004-3702. DOI: [https://doi.org/10.1016/S0004-3702\(03\)00135-8](https://doi.org/10.1016/S0004-3702(03)00135-8). URL: <https://www.sciencedirect.com/science/article/pii/S0004370203001358>.
- Meng, F., J. Su, C. Liu, and W. Chen (2016). “Dynamic decision making in lane change: Game theory with receding horizon”. In: *2016 UKACC 11th International Conference on Control (CONTROL)*, pp. 1–6. DOI: 10.1109/CONTROL.2016.7737643.
- Messaoud, K., I. Yahiaoui, A. Verroust-Blondet, and F. Nashashibi (Mar. 2021). “Attention Based Vehicle Trajectory Prediction”. In: *IEEE Transactions on Intelligent Vehicles* 6.1, pp. 175–185. ISSN: 2379-8904. DOI: 10.1109/TIV.2020.2991952.
- Michel, Olivier (2004). “Cyberbotics Ltd. Webots™: Professional Mobile Robot Simulation”. In: *International Journal of Advanced Robotic Systems* 1.1, p. 5. DOI: 10.5772/5618. URL: <https://doi.org/10.5772/5618>.
- Michon, John A. (1979). *Dealing with danger*. Tech. rep. Verkeerskundig Studiecentrum, Rijksuniversiteit Groningen Haren, The Netherlands. URL: http://www.jamichon.nl/jam_writings/1979_dealing_with_danger.pdf (visited on 04/05/2021).
- Mill, John Stuart (1859). “Utilitarianism (1863)”. In: *Utilitarianism, Liberty, Representative Government*, pp. 7–9.
- Mirchevska, Branka, Christian Pék, Moritz Werling, Matthias Althoff, and Joschka Boedecker (2018). “High-level Decision Making for Safe and Reasonable Autonomous Lane Changing using Reinforcement Learning”. In: *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, pp. 2156–2162. DOI: 10.1109/ITSC.2018.8569448.
- Mizuno, Koji and Janusz Kajzer (1999). “Compatibility problems in frontal, side, single car collisions and car-to-pedestrian accidents in Japan”. In: *Accident Analysis & Prevention* 31.4, pp. 381–391.
- Moor, J. H. (2006). “The Nature, Importance, and Difficulty of Machine Ethics”. In: *IEEE Intelligent Systems* 21.4, pp. 18–21. DOI: 10.1109/MIS.2006.80.
- Mouratidis, Kostas and Victoria Cobeña Serrano (2021). “Autonomous buses: Intentions to use, passenger experiences, and suggestions for improvement”. In: *Transportation Research Part F: Traffic Psychology and Behaviour* 76, pp. 321–335. ISSN: 1369-8478. DOI: <https://doi.org/10.1016/j.trf.2021.05.008>.

//doi.org/10.1016/j.trf.2020.12.007. URL: <https://www.sciencedirect.com/science/article/pii/S1369847820305921>.

- Munson, John H (1971). *Robot planning, execution, and monitoring in an uncertain environment*. Tech. rep. Stanford Research Institute Menlo Park CA.
- Murphy, Robin R. (2000). *Introduction to AI Robotics*. 1st. Cambridge, MA, USA: MIT Press. ISBN: 0262133830.
- Narayanan, Santhanakrishnan, Emmanouil Chaniotakis, and Constantinos Antoniou (2020). “Shared autonomous vehicle services: A comprehensive review”. In: *Transportation Research Part C: Emerging Technologies* 111, pp. 255–293. ISSN: 0968-090X. DOI: <https://doi.org/10.1016/j.trc.2019.12.008>. URL: <https://www.sciencedirect.com/science/article/pii/S0968090X19303493>.
- Nash, John (1951). “Non-Cooperative Games”. In: *Annals of Mathematics* 54.2, pp. 286–295. ISSN: 0003486X. URL: <http://www.jstor.org/stable/1969529>.
- Nash, John F. (1950). “Equilibrium points in n-person games”. In: *Proceedings of the National Academy of Sciences* 36.1, pp. 48–49. ISSN: 0027-8424. DOI: 10.1073/pnas.36.1.48. eprint: <https://www.pnas.org/content/36/1/48.full.pdf>. URL: <https://www.pnas.org/content/36/1/48>.
- NHTSA (2008). *National Motor Vehicle Crash Causation Survey*. Tech. rep. National Highway Traffic Safety Administration. URL: <https://crashstats.nhtsa.dot.gov/Api/Public/ViewPublication/811059> (visited on 03/05/2021).
- Nordhoff, Sina, Victor Malmsten, Bart van Arem, Peng Liu, and Riender Happee (2021). “A structural equation modeling approach for the acceptance of driverless automated shuttles based on constructs from the Unified Theory of Acceptance and Use of Technology and the Diffusion of Innovation Theory”. In: *Transportation Research Part F: Traffic Psychology and Behaviour* 78, pp. 58–73. ISSN: 1369-8478. DOI: <https://doi.org/10.1016/j.trf.2021.01.001>. URL: <https://www.sciencedirect.com/science/article/pii/S1369847821000012>.
- Noreils, Fabrice R and Raja G Chatila (1995). “Plan execution monitoring and control architecture for mobile robots”. In: *IEEE transactions on robotics and automation* 11.2, pp. 255–266.
- Notomista, G., M. Wang, M. Schwager, and M. Egerstedt (2020). “Enhancing Game-Theoretic Autonomous Car Racing Using Control Barrier Functions”. In: *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 5393–5399. DOI: 10.1109/ICRA40945.2020.9196757.
- Osborne, Martin J and Ariel Rubinstein (1994). *A course in game theory*. MIT press.
- Osborne, Martin J. (2002). *An Introduction to Game Theory*. 1st. Oxford University Press, Inc. ISBN: 0195322487.

- Parfit, Derek (1984). *Reasons and persons*. OUP Oxford. ISBN: 978-0-1982-4908-5. DOI: 10.2307/2107444.
- Parma, University of (1999). *The ARGO project*. URL: <http://www.argo.ce.unipr.it/ARGO/english/index.html> (visited on 02/24/2021).
- Payton, David (1986). “An architecture for reflexive autonomous vehicle control”. In: *Proceedings. 1986 IEEE International Conference on Robotics and Automation*. Vol. 3. IEEE, pp. 1838–1845.
- Petrich, D., T. Dang, D. Kasper, G. Breuel, and C. Stiller (2013). “Map-based long term motion prediction for vehicles in traffic environments”. In: *16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013)*, pp. 2166–2172. DOI: 10.1109/ITSC.2013.6728549.
- Pickering, James E., Mateusz Podsiadly, and Keith J. Burnham (2019). “A Model-to-Decision Approach for the Autonomous Vehicle (AV) Ethical Dilemma: AV Collision with a Barrier/Pedestrian(s)”. In: *IFAC-PapersOnLine* 52.8. 10th IFAC Symposium on Intelligent Autonomous Vehicles IAV 2019, pp. 257–264. ISSN: 2405-8963. DOI: <https://doi.org/10.1016/j.ifacol.2019.08.080>. URL: <https://www.sciencedirect.com/science/article/pii/S2405896319304136>.
- Polack, Philip, Florent Althé, Brigitte d’Andréa-Novel, and Arnaud de La Fortelle (2017). “The kinematic bicycle model: A consistent model for planning feasible trajectories for autonomous vehicles?” In: *2017 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, pp. 812–818.
- Pomerleau, Dean A. (1988). “ALVINN: An Autonomous Land Vehicle in a Neural Network”. In: *Proceedings of the 1st International Conference on Neural Information Processing Systems. NIPS’88*. Cambridge, MA, USA: MIT Press, pp. 305–313.
- Prakken, Henry (2017). “On the problem of making autonomous vehicles conform to traffic law”. In: *Artificial Intelligence and Law* 25.3, pp. 341–363. DOI: 10.1007/s10506-017-9210-0.
- Pusse, Florian and Matthias Klusch (2019). “Hybrid Online POMDP Planning and Deep Reinforcement Learning for Safer Self-Driving Cars”. In: *2019 IEEE Intelligent Vehicles Symposium (IV)*, pp. 1013–1020. DOI: 10.1109/IVS.2019.8814125.
- Puterman, Martin L. (1994a). “Discounted Markov Decision Problems”. In: *Markov Decision Processes*. John Wiley & Sons, Ltd. Chap. 6, pp. 142–276. ISBN: 9780470316887. DOI: <https://doi.org/10.1002/9780470316887.ch6>. URL: <https://onlinelibrary.wiley.com/doi/abs/10.1002/9780470316887.ch6>.
- (1994b). “Infinite-Horizon Models: Foundations”. In: *Markov Decision Processes*. John Wiley & Sons, Ltd. Chap. 5, pp. 119–141. ISBN: 9780470316887. DOI: <https://doi.org/10.1002/9780470316887.ch5>. URL: <https://onlinelibrary.wiley.com/doi/abs/10.1002/9780470316887.ch5>.
- (1994c). “Model Formulation”. In: *Markov Decision Processes*. John Wiley & Sons, Ltd. Chap. 2, pp. 17–32. ISBN: 9780470316887. DOI: <https://doi.org/10.1002/9780470316887>.

ch2. URL: <https://onlinelibrary.wiley.com/doi/abs/10.1002/9780470316887.ch2>.

- Qiao, Zhiqian, Katharina Muelling, John Dolan, Praveen Palanisamy, and Priyantha Mudalige (2018). “Pomdp and hierarchical options mdp with continuous actions for autonomous driving at intersections”. In: *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, pp. 2377–2382.
- Quintero, R., I. Parra, D. F. Llorca, and M. A. Sotelo (Oct. 2014). “Pedestrian path prediction based on body language and action classification”. In: *17th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pp. 679–684. DOI: 10.1109/ITSC.2014.6957768.
- Rawls, John (1958). “Justice as Fairness”. In: *The Philosophical Review* 67.2, pp. 164–194. ISSN: 00318108, 15581470. URL: <http://www.jstor.org/stable/2182612>.
- (1971). *A Theory of Justice*. Rawls. The Belknap.
- Richards, DC (2010). “Relationship between speed and risk of fatal injury: pedestrians and car occupants”. In:
- Ridel, D., E. Rehder, M. Lauer, C. Stiller, and D. Wolf (Nov. 2018). “A Literature Review on the Prediction of Pedestrian Behavior in Urban Scenarios”. In: *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, pp. 3105–3112. DOI: 10.1109/ITSC.2018.8569415.
- Rizaldi, Albert, Jonas Keinholtz, Monika Huber, Jochen Feldle, Fabian Immler, Matthias Althoff, Eric Hilgendorf, and Tobias Nipkow (2017). “Formalising and Monitoring Traffic Rules for Autonomous Vehicles in Isabelle/HOL”. In: *Integrated Formal Methods*. Ed. by Nadia Polikarpova and Steve Schneider. Cham: Springer International Publishing, pp. 50–66. ISBN: 978-3-319-66845-1.
- Rosén, Erik, Helena Stigson, and Ulrich Sander (2011). “Literature review of pedestrian fatality risk as a function of car impact speed”. In: *Accident Analysis & Prevention* 43.1, pp. 25–33. ISSN: 0001-4575. DOI: <https://doi.org/10.1016/j.aap.2010.04.003>. URL: <https://www.sciencedirect.com/science/article/pii/S0001457510001077>.
- Ross, Don (2001). *Stanford Encyclopedia of Philosophy: Game Theory*. Stanford University. URL: <https://plato.stanford.edu/entries/game-theory/> (visited on 03/31/2021).
- Rudenko, Andrey, Luigi Palmieri, Michael Herman, Kris M Kitani, Dariu M Gavrila, and Kai O Arras (2020). “Human motion trajectory prediction: a survey”. In: *The International Journal of Robotics Research* 39.8, pp. 895–935. DOI: 10.1177/0278364920917446. URL: <https://doi.org/10.1177/0278364920917446>.
- Russell, Stuart and Peter Norvig (2009). *Artificial Intelligence: A Modern Approach*. 3rd. USA: Prentice Hall Press. ISBN: 0136042597.

- SAE, On-Road Automated Driving (ORAD) committee (June 2018). *Taxonomy and Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles. SAE J3016*. Standard. DOI: https://doi.org/10.4271/J3016_201806.
- Savage, Leonard J. (1954). *The Foundations of Statistics*. 1st. John Wiley & Sons. ISBN: 0-486-62349-1.
- Scheel, Oliver, Loren Schwarz, Nassir Navab, and Federico Tombari (2018). “Situation Assessment for Planning Lane Changes: Combining Recurrent Models and Prediction”. In: *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2082–2088. DOI: 10.1109/ICRA.2018.8462838.
- Schmerling, Edward, Karen Leung, Wolf Vollprecht, and Marco Pavone (2018). “Multimodal Probabilistic Model-Based Planning for Human-Robot Interaction”. In: *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3399–3406. DOI: 10.1109/ICRA.2018.8460766.
- Schwarting, Wilko, Javier Alonso-Mora, and Daniela Rus (2018). “Planning and Decision-Making for Autonomous Vehicles”. In: *Annual Review of Control, Robotics, and Autonomous Systems* 1.1, pp. 187–210. DOI: 10.1146/annurev-control-060117-105157.
- Sharma, Omveer, N.C. Sahoo, and N.B. Puhan (2021). “Recent advances in motion and behavior planning techniques for software architecture of autonomous vehicles: A state-of-the-art survey”. In: *Engineering Applications of Artificial Intelligence* 101, p. 104211. ISSN: 0952-1976. DOI: <https://doi.org/10.1016/j.engappai.2021.104211>. URL: <https://www.sciencedirect.com/science/article/pii/S0952197621000580>.
- Shoham, Yoav and Kevin Leyton-Brown (2008). *Multiagent systems: Algorithmic, game-theoretic, and logical foundations*. Cambridge University Press.
- Silver, David and Joel Veness (2010). “Monte-Carlo Planning in Large POMDPs”. In: *Advances in Neural Information Processing Systems*. Ed. by J. Lafferty, C. Williams, J. Shawe-Taylor, R. Zemel, and A. Culotta. Vol. 23. Curran Associates, Inc. URL: <https://proceedings.neurips.cc/paper/2010/file/edfbe1afcf9246bb0d40eb4d8027d90f-Paper.pdf>.
- Simmons, Reid G., Richard Goodwin, Karen Zita Haigh, Sven Koenig, Joseph O’Sullivan, and Manuela M. Veloso (Dec. 1997). “Xavier: Experience with a Layered Robot Architecture”. In: *SIGART Bull.* 8.1–4, pp. 22–33. ISSN: 0163-5719. DOI: 10.1145/272874.272878.
- Simms, C K and D P Wood (2006). “Pedestrian Risk from Cars and Sport Utility Vehicles - A Comparative Analytical Study”. In: *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering* 220.8, pp. 1085–1100. DOI: 10.1243/09544070JAUT0319. URL: <https://doi.org/10.1243/09544070JAUT0319>.
- Simon, Herbert A. (1955). “A Behavioral Model of Rational Choice”. In: *The Quarterly Journal of Economics* 69.1, pp. 99–118. ISSN: 00335533, 15314650. URL: <http://www.jstor.org/stable/1884852>.

- Smit, Stefan, Ernst Tomasch, Harald Kolk, Michael Alois Plank, Jürgen Gugler, and Hannes Glaser (Jan. 2019). “Evaluation of a momentum based impact model in frontal car collisions for the prospective assessment of ADAS”. In: *European Transport Research Review* 11.1, p. 2. ISSN: 1866-8887. DOI: 10.1186/s12544-018-0343-3. URL: <https://doi.org/10.1186/s12544-018-0343-3>.
- Snider, Jarrod M (2009). *Automatic steering methods for autonomous automobile path tracking*. Report. Carnegie Mellon University. URL: https://www.ri.cmu.edu/pub_files/2009/2/Automatic_Steering_Methods_for_Autonomous_Automobile_Path_Tracking.pdf (visited on 04/23/2021).
- Soteropoulos, Aggelos, Martin Berger, and Francesco Ciari (2019). “Impacts of automated vehicles on travel behaviour and land use: an international review of modelling studies”. In: *Transport Reviews* 39.1, pp. 29–49. DOI: 10.1080/01441647.2018.1523253. eprint: <https://doi.org/10.1080/01441647.2018.1523253>. URL: <https://doi.org/10.1080/01441647.2018.1523253>.
- Spica, R., E. Cristofalo, Z. Wang, E. Montijano, and M. Schwager (2020). “A Real-Time Game Theoretic Planner for Autonomous Two-Player Drone Racing”. In: *IEEE Transactions on Robotics* 36.5, pp. 1389–1403. DOI: 10.1109/TR0.2020.2994881.
- Sutton, Richard S. (1990). “Integrated Architectures for Learning, Planning, and Reacting Based on Approximating Dynamic Programming”. In: *Machine Learning Proceedings 1990*. Ed. by Bruce Porter and Raymond Mooney. San Francisco (CA): Morgan Kaufmann, pp. 216–224. ISBN: 978-1-55860-141-3. DOI: <https://doi.org/10.1016/B978-1-55860-141-3.50030-4>. URL: <https://www.sciencedirect.com/science/article/pii/B9781558601413500304>.
- Teoh, Eric R. and David G. Kidd (2017). “Rage against the machine? Google’s self-driving cars versus human drivers”. In: *Journal of Safety Research* 63, pp. 57–60. ISSN: 0022-4375. DOI: <https://doi.org/10.1016/j.jsr.2017.08.008>. URL: <https://www.sciencedirect.com/science/article/pii/S002243751730381X>.
- Tesla (June 30, 2016). *A Tragic Loss*. Tesla. URL: <https://www.tesla.com/blog/tragic-loss> (visited on 03/08/2021).
- Thornton, Sarah M., Selina Pan, Stephen M. Erlien, and J. Christian Gerdes (2017). “Incorporating Ethical Considerations Into Automated Vehicle Control”. In: *IEEE Transactions on Intelligent Transportation Systems* 18.6, pp. 1429–1439. DOI: 10.1109/TITS.2016.2609339.
- Thrun, Sebastian, Mike Montemerlo, Hendrik Dahlkamp, David Stavens, Andrei Aron, James Diebel, Philip Fong, John Gale, Morgan Halpenny, Gabriel Hoffmann, et al. (2006). “Stanley: The robot that won the DARPA Grand Challenge”. In: *Journal of field Robotics* 23.9, pp. 661–692.
- Tran, Q. and J. Firl (June 2013). “Modelling of traffic situations at urban intersections with probabilistic non-parametric regression”. In: *2013 IEEE Intelligent Vehicles Symposium (IV)*, pp. 334–339. DOI: 10.1109/IVS.2013.6629491.

- Tran, Q. and J. Firl (June 2014). “Online maneuver recognition and multimodal trajectory prediction for intersection assistance using non-parametric regression”. In: *2014 IEEE Intelligent Vehicles Symposium Proceedings*, pp. 918–923. DOI: 10.1109/IVS.2014.6856480.
- Tsugawa, Sadayuki, Teruo Yatabe, Takeshi Hirose, and Shuntetsu Matsumoto (1979). “An automobile with artificial intelligence”. In: *Proceedings of the 6th international joint conference on Artificial intelligence-Volume 2*, pp. 893–895.
- Turing, Alan Mathison (1950). “Computing machinery and intelligence”. In: *Mind* 59.236, p. 433.
- Turnwald, Annemarie, Daniel Althoff, Dirk Wollherr, and Martin Buss (2016). “Understanding human avoidance behavior: interaction-aware decision making based on game theory”. In: *International journal of social robotics* 8.2, pp. 331–351. DOI: 10.1007/s12369-016-0342-2.
- Ulbrich, S. and M. Maurer (2013). “Probabilistic online POMDP decision making for lane changes in fully automated driving”. In: *16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013)*, pp. 2063–2067. DOI: 10.1109/ITSC.2013.6728533.
- Ulmer, Berthold (1994). “Vita ii-active collision avoidance in real traffic”. In: *Proceedings of the Intelligent Vehicles’ 94 Symposium*. IEEE, pp. 1–6.
- UNECE (2020). *UN Regulation on Automated Lane Keeping Systems is milestone for safe introduction of automated vehicles in traffic*. Regulation. United Nations Economic Commission for Europe. URL: <https://unece.org/transport/press/un-regulation-automated-lane-keeping-systems-milestone-safe-introduction-automated> (visited on 03/01/2021).
- Urmson, Chris, Joshua Anhalt, Drew Bagnell, Christopher Baker, Robert Bittner, M. N. Clark, John Dolan, Dave Duggins, Tugrul Galatali, Chris Geyer, Michele Gittleman, Sam Harbaugh, Martial Hebert, Thomas M. Howard, Sascha Kolski, Alonzo Kelly, Maxim Likhachev, Matt McNaughton, Nick Miller, Kevin Peterson, Brian Pilnick, Raj Rajkumar, Paul Ryski, Bryan Salesky, Young-Woo Seo, Sanjiv Singh, Jarrod Snider, Anthony Stentz, William Whittaker, Ziv Wolkowicki, Jason Ziglar, Hong Bae, Thomas Brown, Daniel Demitrish, Bakhtiar Litkouhi, Jim Nickolaou, Varsha Sadekar, Wende Zhang, Joshua Struble, Michael Taylor, Michael Darms, and Dave Ferguson (2008). “Autonomous driving in urban environments: Boss and the Urban Challenge”. In: *Journal of Field Robotics* 25.8, pp. 425–466. DOI: <https://doi.org/10.1002/rob.20255>. URL: <https://www.onlinelibrary.wiley.com/doi/abs/10.1002/rob.20255>.
- Urmson, Christopher, Joshua Anhalt, Michael Clark, Tugrul Galatali, Juan Pablo Gonzalez, Jay Gowdy, Alexander Gutierrez, Sam Harbaugh, Matthew Johnson-Roberson, Hiroki Kato, et al. (2004). “High speed navigation of unrehearsed terrain: Red team technology for grand challenge 2004”. In: *Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, Tech. Rep. CMU-RI-TR-04-37*.

- Vacek, S., T. Gindele, J. M. Zollner, and R. Dillmann (June 2007). “Situation classification for cognitive automobiles using case-based reasoning”. In: *2007 IEEE Intelligent Vehicles Symposium*, pp. 704–709. DOI: 10.1109/IVS.2007.4290199.
- Von Neumann, John and Oskar Morgenstern (1966). *Theory of Games and Economic Behavior*. 3rd. Princeton University Press. ISBN: 978-0691130613.
- Wadud, Zia, Don MacKenzie, and Paul Leiby (2016). “Help or hindrance? The travel, energy and carbon impacts of highly automated vehicles”. In: *Transportation Research Part A: Policy and Practice* 86, pp. 1–18. ISSN: 0965-8564. DOI: <https://doi.org/10.1016/j.tra.2015.12.001>. URL: <https://www.sciencedirect.com/science/article/pii/S0965856415002694>.
- Wang, Mingyu, Zijian Wang, John Talbot, J Christian Gerdes, and Mac Schwager (2019). “Game Theoretic Planning for Self-Driving Cars in Competitive Scenarios.” In: *Robotics: Science and Systems*.
- Wei, Junqing, John M Dolan, Jarrod M Snider, and Bakhtiar Litkouhi (2011). “A point-based mdp for robust single-lane autonomous driving behavior under uncertainties”. In: *2011 IEEE International Conference on Robotics and Automation*. IEEE, pp. 2586–2592.
- Weijermars, Wendy, Niels Bos, Annelies Schoeters, Jean-Christophe Meunier, Nina Nuyttens, Emmanuelle Dupont, Klaus Machata, Robert Bauer, Katherine Perez, Jean-Louis Martin, Heiko Johansson, Ashleigh Filtness, Laurie Brown, and Pete Thomas (2018). “Serious Road Traffic Injuries in Europe, Lessons from the EU Research Project SafetyCube”. In: *Transportation Research Record* 2672.32, pp. 1–9. DOI: 10.1177/0361198118758055. URL: <https://doi.org/10.1177/0361198118758055>.
- WHO (2018). *Global status report on road safety 2018*. Tech. rep. World Health Organization. URL: <https://www.who.int/publications/i/item/9789241565684> (visited on 03/05/2021).
- Winston, P.H. (1992). *Artificial Intelligence*. A-W Series in Computerscience. Addison-Wesley Publishing Company. ISBN: 9780201533774.
- Xin, L., P. Wang, C. Chan, J. Chen, S. E. Li, and B. Cheng (Nov. 2018). “Intention-aware Long Horizon Trajectory Prediction of Surrounding Vehicles using Dual LSTM Networks”. In: *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, pp. 1441–1446. DOI: 10.1109/ITSC.2018.8569595.
- Yadron, Danny and Dan Tynan (July 1, 2016). *Tesla driver dies in first fatal crash while using autopilot mode*. The Guardian. URL: <https://www.theguardian.com/technology/2016/jun/30/tesla-autopilot-death-self-driving-car-elon-musk> (visited on 03/01/2021).
- Ziegler, J., P. Bender, M. Schreiber, H. Lategahn, T. Strauss, C. Stiller, T. Dang, U. Franke, N. Appenrodt, C. G. Keller, E. Kaus, R. G. Herrtwich, C. Rabe, D. Pfeiffer, F. Lindner, F. Stein, F. Erbs, M. Enzweiler, C. Knöppel, J. Hipp, M. Haueis, M. Trepte, C. Brenk, A. Tamke, M.

Ghanaat, M. Braun, A. Joos, H. Fritz, H. Mock, M. Hein, and E. Zeeb (2014). “Making Bertha Drive—An Autonomous Journey on a Historic Route”. In: *IEEE Intelligent Transportation Systems Magazine* 6.2, pp. 8–20. DOI: 10.1109/MITS.2014.2306552.

Zyner, A., S. Worrall, and E. Nebot (July 2018). “A Recurrent Neural Network Solution for Predicting Driver Intention at Unsignalized Intersections”. In: *IEEE Robotics and Automation Letters* 3.3, pp. 1759–1764. ISSN: 2377-3766. DOI: 10.1109/LRA.2018.2805314.