Alma Mater Studiorum
Università di Bologna
Campus di Cesena

DIPARTIMENTO DI INFORMATICA – SCIENZA E INGEGNERIA
Corso di Laurea Magistrale in Ingegneria e Scienze Informatiche

Instinct and reflex inspired safety mechanisms for adaptive robots

Elaborato in:

Intelligent Robotic Systems

Relatore:
Prof. Andrea Roli

Presentata da:
Eleonora Bertoni

Correlatore:
Dr. Paolo Baldini

Correlatore:
Dr. Michele Braccini

Sessione I
Anno Accademico 2025-2026

"May your heart be your guiding key"

Introduction

The goal of adaptive robotics is to build fully autonomous robots capable of overcoming any unexpected change on their own with no need for human intervention. This can be achieved by modifying the operative strategy used to complete a task by exploiting, for example, neural networks and learning, evolution as an adaptation process or online adaptation. In particular, for online processes, robots try variations of their own controllers in real time exploring novel strategies as they operate to find a suitable one. During this exploration, a robot is vulnerable and exposed to damage. In a controlled setting, a human supervisor can intervene to avoid serious damages but in an autonomous deployment this is not a viable or desired solution. A single damage can permanently compromise the robot, making future operation impossible and replacement costly. For this reason, many efforts have been made to develop robots that can adapt to damage like animals can adapt to injuries. Nevertheless, even when these strategies are available, preventing damages remains preferable to recovering from them. In this context, how can an autonomous robot protect itself from damage especially while it is still learning or adapting?

In nature animals but also humans can count on a set of innate instincts and reflexes. Many of these are meant to protect the individual and trigger survival-oriented responses to the environment. This characteristic is evident in some reflex actions like blinking or moving away when an object suddenly approaches. It can also be observed in more complex behaviors triggered, for example, by fear that leads to avoiding certain potentially dangerous areas and situations. Given their importance in nature and based on the fact that it is common in robotics to take inspiration from nature and biology and that this often leads to interesting results, robots too could benefit from having a set of innate and ready-to-use safety mechanisms inspired by instincts and reflexes.

The contribution of this thesis lies in the design of a safety module based on innate instincts and reflexes to prevent different types of damage. The focus is on potential dangers that could be faced by a simple wheeled robot that freely navigates its environment. We provide an implementation of these mechanisms and propose a control architecture to equip the safety module to a robot subject to an Online Adaptation process.

The thesis is organized as follows:

- **Chapter 1 - Innate instincts and reflexes**: It reviews the literature on instincts and reflexes taking into consideration different perspectives on the topic. It examines their characteristics and purpose, focusing in particular on those that are useful for the survival and protection of the individual. Finally, it provides a set of examples of instincts and reflexes observed in nature.

- **Chapter 2 - Autonomous and adaptive robots**: It examines the relevant background for autonomous and adaptive robots covering Evolutionary Robotics, Online Adaptation and existing approaches that focus on damage recovery.

- **Chapter 3 - Robots equipped with innate instincts and reflexes**: It illustrates the contribution of the thesis. It describes the designed safety instincts and reflexes, the experimental setup, the proposed control architecture and the implementation details of each mechanism. It also describes an example of Online Adaptation process equipped with the instincts and reflexes and discusses the results and limitations.

- **Conclusion**: In this final chapter we summarize the work done and draw the conclusions.

Contents

List of Figures

List of Tables

Chapter 1

Innate instincts and reflexes

Innate instincts and reflexes are fundamental for the survival of both humans and animals. Literature is plenty of examples on how instincts like fear can help animals in successfully escaping from threats or on how reflex reactions to pain allow humans to prevent further injuries. Due to the importance of these two mechanisms, for many years, instincts and reflexes have been the topic of discussion among scholars and researchers coming from very different fields. For example W.M. Wheeler [1] states that there are substantial differences between his point of view as an entomologist and zoologist and that of a philosopher and psychologist: one studies the subject and explains the origin of these mechanisms observing animals while the other is more focused on the impact they have in humans.

1.1 Reflexes and instincts

We define a reflex as an immediate and involuntary motor reaction triggered by sensory stimuli [2]. Reflexes are automatic responses but they can happen without the subject being aware of their effects. Usually, the strength of the reflex is proportionate to the intensity of the stimulus. Sensory and motor neurons work together in what is known as the reflex arc, a neural pathway that allows animals to respond rapidly to stimuli, ensuring survival through protective and adaptive reactions. Simple reflexes involve a direct connection between sensory and motor neurons while more complex reflexes require multiple neurons and synapses to coordinate responses. Reflexes are useful for many different purposes: some reflexes quickly and efficiently regulate vital biological processes, while others have a protective or self-defense function needed to guarantee the survival of the organism. This defensive function is observed in reflexes such as the startle reflex, in infants, which provides an automatic response to an unexpected stimulus [2]. Reflexes are genetically determined, they are present since birth and no effort is needed to acquire them, as a consequence reflex responses are essentially invariant in healthy individuals and their examination can often provide valuable insights about the health of the individual and the presence of neurologic disorders [2].

On the other hand, instincts can be seen as the innate tendency to execute a particular complex behavior or a specific series of actions when facing a certain situation or to respond to a specific stimulus [3]. We refer to instincts also to all biological forces that guide the individual to the fulfillment of its needs. As reflexes, instincts are performed without relying upon prior experience and learning. This implies that instincts are an expression of innate biological factors that exist in most members of the species; however, details of their performance change based on the experience of the single individual.

Table 1.1 summarizes the key differences and similarities between instincts and reflexes.

1.2 Different perspectives on the topic

This topic has attracted and keeps attracting the attention of psychologists, philosophers and biologists, especially ethologists and physiologists. For this reason, instincts and reflexes have been investigated from different perspectives which, over the years, have given rise to many definitions.

For Wheeler [1], in 1921, an instinctive behavior originates from an im-

Instincts	Reflexes
Tendency to perform an entire behavior in certain conditions	Quick defensive motor reaction in response to a stimulus
They are innate and present since birth	They are innate and present since birth
Triggered by a stimulus or a situation	Triggered by a stimulus
They can slightly change based on the experience of the individual	They do not change based on the experience

Table 1.1: Similarities and differences between Instincts and Reflexes

pulse, namely an innate biological drive or a predisposition, and is characterized by some kind of necessity because the resulting reaction is almost obligatory. In addition, it is both highly adaptive and purposive even though the organism manifesting it is unaware of any purpose, other than an immediate one. He studied animal behavior, and in particular insects, to get insights on the relation between stimuli and responses. He states that insects best show the diversity and complexity of instincts because many species have endured the challenges and changes of geological time.

From an early and pure mechanistic point of view [1], we have Descartes's idea of reflex: animals are seen as machines, and every activity of the organism is a necessary reaction to some external or internal stimulus. It is a simple chain of reflexes to react to changes in the environment. In this definition we lose all the intentionality of the action. This connection between the stimulus and a definite response resembles the link cause-effect and is the fundamental purpose of the nervous structures in the animal body [4].

In 1951, in physiology, it is explained how fundamental it is that animals reactions to changes in the environment are directed towards the preservation of their existence [4] [5]. The lack of this exact correspondence with the environment would be lethal, just think how dangerous it could be if instead of being attracted to food, the animal were repelled by it, or if instead of running from fire the animal threw itself into the fire. This means that organisms survive only if they are in a balanced state with the environment and reflexes, machine-like and inevitable reactions of the organism, are the elemental units to do so. In short, the reflex can be seen as a survival-oriented response to the environment. It is worth mentioning that in this definition a reflex is innate, shared by all members of a species, permanent in the sense that it cannot be unlearned and the pattern of the reflexive reaction is quite constant and predictable. In this field we can sometimes find instincts and reflexes grouped together and it is discussed that there are no real differences between them. For example, the essential feeding instinct of a newly hatched chick that reacts by pecking towards any stimulus that catches its eyes works just like the defensive reflex of tilting our head or closing our eyelids when something flicks past our eyes. They also reject distinctions based on the complexity or on the origin, internal or external, of the stimulus.

According to neuroscience and psychology [6], instincts need to be studied focusing not only on the causal relationship with the environment but also on their inner origin, that can be seen as an automatic response to a specific stimulus, almost like a reflex, as well as an impulse, a pressure towards auto-preservation. This leads us to consider a behavior always in relation to specific motives that can be found in emotions or instincts such as hunger or fear as well as in simple reflexes. For this reason, the study of instincts and reflexes often overlaps with that of emotions. Human emotions evolved to ensure safety in various domains, and scientists have long studied the actions that impact basic survival and what triggers them as to reveal the nature of emotions. These actions are believed to be caused by the existence of a survival circuit and range from defensive reflexes, which are said to be executed in an obligatory, rapid manner to flexible, goal directed actions that result from thinking about possible future outcomes. It is believed that the same circuits are present in animals as well: rats are assumed to be fearful when they protect themselves from a predator and flies are assumed to be angry when they attack each other.

1.3 Examples

In this section, we will list some examples of instincts and reflexes to concretely show their usefulness and importance in keeping the organism safe and increasing its chances of survival and adaptation. These mechanisms have been observed in nature or through experiments.

Wheeler [1] studied insects and, as one of his examples, explains that red wood ants have evolved an instinctive defensive behavior: whenever an intruder comes close to their nest, they rush outside and spray it with acid. He also observed instinctive behaviors important for the survival of the species rather than the one of the single individual [2]. For example, some species show established patterns in courtship rituals, nest building and territorial behaviors. In addition, there are instincts important for newly hatched animals: for example sea turtles move towards the ocean and goslings, as investigated by Lorenz [7], follow their mother knowing it will keep them safe, even though they can be tricked because they need to learn how to identify their mother.

An interesting instinct studied by physiologists is the investigatory or "what-is-it?" reflex [4] [5]. It is important for both men and animals because it immediately leads them towards the slightest changes in their environment and to fully investigate them. It is clear that an animal without this attention mechanism could not survive long and humans would not be curious about the world around them. This curiosity and attention that animals and humans show towards their surroundings is also particularly important for threat assessment and detection [8]. In addition it is interesting that anxiety in humans can influence this predisposition: elevated anxiety levels usually mean a stronger attentional bias toward threatening or potentially threatening stimuli.

Another example is the self-defense instinct [4] [5] that is triggered as soon as the prey spots or hears the predator, giving it a chance to save itself by fleeing or hiding. If this reaction fired only when the predator had already touched the prey it would be useless, instead, it works because the prey successfully learns how to associate the sight and sounds produced by the predator with danger.

There is also a passive-defensive mechanism [4] [5] that can be seen as caution or fear. This can be triggered by specific weak or repetitive stimuli or by a strong and unusual stimulus caused, for example, by the sudden appearance of a predator. In this case the organism reduces the vigor of its behavior, sometimes even producing a freezing reaction. This could enhance the chances of survival because the prey could go unnoticed.

Similar behaviors to the previous two, are also studied in [9], where the authors explain, in terms of neural circuits, the origin of these mechanisms that allow animals to successfully avoid danger and escape from predators, and thus to survive. In order to do this, they explain how animals are able to timely detect threats and how they start the correct fight-or-flight responses. These defensive behaviors are innate capabilities of the nervous system. Mice evolved a couple of instinctive mechanisms: freeze and response. The imminence of the threat determines which one is needed to be activated. It is observed that freezing is chosen when the threats are remote, for example when the mouse detects a distant eagle in the sky. The reason for this choice is simple: stopping all movements makes the mouse more difficult to be identified as a prey and keeps it safe until the predator goes away, it also allows to save energy compared to escaping. However, when the threat is imminent, for example if the mouse has already been identified and the eagle is approaching, escape especially towards a close shelter is better. This is examined in depth in [10] where it is shown that mice can learn the location of a shelter and then, when in danger, they instinctively flee towards it. When facing an imminent threat animals need to perform an escape that involve choosing between moving away from the threat or moving toward a safe place. The first one is quicker but can have many disadvantages, for example it can lead the animal into dead-ends or makes it more easy to spot. The second one has better long-term value but it is more complex and needs time to compute the safest escape route considering all the available shelters. The collected data indeed show that the mouse escape behavior is an instinctive and goal-directed defensive behavior, rather than a simple reflex-like reaction. This is shown by the fact that when the mouse notices something dangerous while it is inside a shelter he does not try to escape. They also reflected on the fact that, although this defensive mechanism is instinctive, it is also conditioned by the mouse's knowledge of available shelters: if there is no memorized safe location the mouse could choose freezing instead. Similar decisions are also observed in the vinegar fly. When a non-imminent danger is detected the fly privileges a more controlled motor planning strategy, however, this can be overridden by a fast escape when the threat is dangerously near [9]. Finally the fight response can be seen as a last resort when all the other strategies have failed [8].

A further different perspective on fight-or-flight responses is given in [8], where an unified and general model, the Survival Optimization System, is proposed to explain how animals and humans protect themselves when they face danger and what important role fear plays in this process. Here, survival intelligence, defined as the capacity of minimizing local threats while adapting to novel threats, relies on both fixed and flexible or learned traits. This, again, highlights the significance of innate traits, such as instincts and reflexes, in dealing with highly risky situations especially when they can not be predicted or detected in time. It is stated that this quick instinctive reaction system resides in the oldest regions of the human central nervous system and this is coherent with the evolutionary benefits it brings.

Physiology takes into consideration the life instinct [4] [5] which corresponds to the realization of a goal: the preservation of life. It needs other instincts such as the alimentary reflex to feel hunger and continuously search for food, and the investigatory reflex to react to novel stimuli.

In [4] [5] it is observed that an animal in confinement attempts to free itself. This behavior is referred to as the freedom reflex and it is considered to be of fundamental importance for the organism survival because, without it, it would not be possible to overcome any obstacle.

In a study that suggests that animals should be used in the field of disaster management it is observed that animals instinctively react beforehand to the occurrence of a disaster [11]. The key idea of this unconventional approach is to comprehensively examine the abnormal behavior and reactions of animals to potential hazards and then apply these observations to human disaster response to detect natural but also man-made emergencies. It is still not known how animals can sense a disaster but they express anxiety or fear by hiding, flying away, making alarming sounds or exhibiting frantic behavior. It has been observed that animals tend to flee to the top of a mountain before and during an earthquake and dolphins moved to deeper water to avoid hurricanes.

Among reflexes, we can find various protective responses such as withdrawing from pain or even sneezing but also heart rate and breathing. Most of the time, we are unaware of these happening due to their involuntary nature but some of them can be more apparent than others [2].

The knee-jerk or stretch reflex is well-known for its widespread clinical use [2]. It is triggered by a tap on the tendon of the knee and when the knee flexor is stretched, some of the tension in the knee extensor dissipates. It does so in order to oppose sudden changes in the length of skeletal muscle. We can examine the flexor reflex and accompanying crossed extensor reflex activated by a painful stimulus [2]. This behavior makes sure that the foot is quickly removed from the harmful stimulus while also take care of maintaining balance. Again, we see a reflex whose purpose is to protect the organism. It is so quick that the pain is felt only after the withdrawal response is complete and this speed is particularly important especially to avoid more critical injuries that would be caused by a longer exposure to the harmful source.

Other useful reflexes involve the eye and the ear to protect them from overstimulation that could severely damage them by accident due to their sensitivity and fragility [2]. For this reason, the pupillary reflex control, changing the diameter of the pupil, the amount of light that can enter the retina. The eyeball is also protected from drying by a blink reflex and from injuries by a quick eyelid closure reflex that is fired when an object gets really close or even when the lashes or cornea are touched. There is also a reflex that protects the ear from potentially damaging sounds by contracting some of the ear muscles [2].

We can consider as protective reflexes really quick escape responses. For example, squids when startled perform an evasive action by contracting their mantle muscles that generates a jet of water [2]. Similarly, fish respond to vibrations carried through the water with a reflex contraction that results in a sudden movement in the opposite direction of the potential danger [6].

It has been observed that, in humans, some reflexes come into play since infancy [12]. They are called survival and primitive reflexes and their role is to facilitate the survival of the neonate. It is also shown that delays in the development of these reflexes can expose the newborn to serious threats. For example, the sucking reflex is fundamental for both breathing and swallowing and, as a consequence for eating. There are also reflexes that trigger motor reactions to prevent injuries for example when there is a sudden change in the position of the body.

To conclude this overview on instincts and reflexes, we take into consideration Valentino von Braitenberg's works on synthetic psychology [13]. Braitenberg's vehicles were originally thought experiments to understand which principles lead to intelligence [14]. They showed that even vehicles equipped with very simple sensor-motor connections can display complex behaviors. For our scope, the simpler vehicles are the most relevant ones because they allow to focus on the connection between stimuli and reaction. In particular, by coupling the light sensors to the motors, the vehicle can exhibit behaviors resembling attraction, avoidance, aggression, or fear that can be explained in terms of instincts and reflexes. This work represents an interesting bridge between biological mechanisms and robotics because it is rooted in neuroscience and biology but the use of wheeled vehicles places it closer to robots than animals.

Chapter 2

Autonomous and adaptive robots

One of the main goals in robotics is to build robots that are truly autonomous, that is, robots that are capable of surviving in complex environments without human intervention. In this context, adaptivity is one of the most important characteristics a robot should have. Animals, humans and other biological systems are naturally able to adapt to changes in themselves or in the environment but, for robots, this is more complicated. Biology identifies four different kinds of adaptation [14]: evolutionary adaptation, concerning how species change in the long term to face new and unsustainable environmental conditions, physiological adaptation, involving physiological adjustments in the individual, sensory adaptation, characterized by adjustments in the sense organs of the individual and, lastly, adaptation by learning.

In this chapter, we examine the relevant background for autonomous and adaptive robots and focus in particular on evolutionary adaptation and adaptation by learning. For this reason, we cover Evolutionary Robotics, Online Adaptation and approaches to achieve damage recovery.

2.1 Evolutionary robotics

The field of evolutionary robotics has emerged in the 1990s and has been widely recognized since 2000 after Nolfi and Floreano published the first book entirely on the subject [15]. Evolutionary robotics can be defined as the field that exploits evolutionary algorithms to design and optimize robots. This evolution can target the body/morphology of the robot [16] as well as its brain/controller [17] [18].

2.1.1 Evolutionary computation

Algorithm designers have used evolution as a source of inspiration for problem solving since the invention of computers [19]. The approach treats possible solutions as individuals in a population and needs a fitness function to evaluate the quality of each individual in relation to the problem we are facing. Finally, it needs to specify a set of selection and variation operators. Just like natural evolution, the selection operators pick phenotypes (candidate solution) for the next generation and decide which one will be the parents based on their fitness. At a lower level, the algorithm works with genotypes, objects that represent a possible solution and on which the variation operators, such as mutation and recombination, can act. Genotypes can be data structures such as array of bits, array of real numbers and trees. The history of evolutionary computations tells us that suitable combinations of these simple data structure is enough to find good solutions.

For example, in genetic programming a computer program solution is generated through an automatic domain-independent method. Computer programs are often represented as trees and, starting with randomly created solutions, the algorithm applies the Darwinian principles of natural selection, crossover, mutation, gene duplication and deletion [20]. Evolutionary algorithms have also been applied in the design domain to find new effective antenna designs, searching and testing even many unusual structures to find solutions that are very different from the ones proposed by human-designers [19]. This approach has also been used for black-box optimization and also to design machine learning algorithms.

In summary, the key features of Evolutionary Computation are its ability to work specifying only the solution encoding and the fitness function with no need for explicit assumptions about the problems, its flexibility that eases its application in combination with domain-specific methods, its robustness towards suboptimal solutions and its capability to produce unexpected and counter intuitive solutions [19].

2.1.2 From evolutionary computation to evolutionary robotics

As discussed in section 2.1.1, the idea of using evolutionary algorithms in robotics is highly justified by more than 40 years of successful applications in many scientific and technical fields where they have proven to be able to solve hard and challenging problems, even in situations where traditional approaches often do not work, for example when noise is present. The evolution of robots has also shown to achieve the same results [15]. In addition it can also find simpler or even unexpected, original solutions that designers or developers could not have thought of otherwise. This is particularly important because manual design of a robot that is both autonomous and adaptive is very difficult [21] [15]. It is possible because Evolutionary Robotics tackles the problem of designing robots from the embodied intelligence point of view, meaning that intelligence is not determined only by the brain of the animal or the control policy of the robot but also by all the complex interactions between the machine and the environment. As a consequence, the morphology of the robot, its sensors, its controller, the environment and all the interactions are studied as a whole instead of designing each part separately and combining them together only at the end of the process [22] [19] [21]. These components should not be divided and this becomes even more obvious when a robot with the “right” body can achieve a task with a simpler controller than the one evolved by a robot with the “wrong” body. Morphological computation states that physical characteristics of the robot body can perform part of the computations that would otherwise be handled by its controller [21].

Evolutionary algorithms and, as a consequence, Evolutionary Robotics take inspiration from natural evolution. The idea is that since natural evolution has led to the development of suitable lifeforms for different environments, artificial evolution can produce suitable robots for different applications [15]. Evolutionary Robotics is very different from other engineering fields that typically depend on well-established mathematical and physical principles because it is based on biological mechanisms, that are often obscure, meaning that the underlying dynamics are not deeply understood. [19]. As shown in [21], not only evolutionary approaches but also other techniques that use biological systems as their source of inspiration and focus on embodied dynamics have proved to be successful for the development of autonomous robots. Natural evolution took billions of years and complex mechanisms but we can not afford to do the same and there is no need to do so because, even though our knowledge on the intricate evolutionary mechanisms is quite limited, following and implementing the principles is more important and has proven to be enough [15]. As a consequence, the general trend is to copy tricks from natural evolution and, at the same time, simplify and speedup the process exploiting the artificial as much as possible. This is justified because natural evolution is extremely inefficient and, especially in pure real-world application, we can not afford wasting robots and time.

Understanding evolution principles is very challenging especially because it is not possible to directly observe the evolution of animals for long periods, therefore, the studies are limited to simple bacterial organisms and can cover just a few decades. An interesting application of evolutionary robotics is that it can help biologists in finding generalizations about natural evolving systems. Modeling and simulating the evolution can be useful to verify hypothesis and to obtain new insights about how and why some traits evolved [22] [21]. In conclusion, Evolutionary Robotics can be seen as a subfield of both engineering science and biology in the way it takes inspiration but also contributes to these fields [22].

2.1.3 Evolution as an adaptation process

Floreano [23] shows that the evolution of robots by means of Darwinian selection leads to the generation of complex and adaptive behaviors, which is fundamental for the creation of increasingly autonomous robots. In Evolutionary Robotics each genome describes the parameters of the controller of the robot or of its morphology. For example, in Floreano experiments [23], where robots are controlled by neural networks, each artificial genome encodes the weight configuration that determines, along with the information provided by the environment, the behavior. The process follows a genetic algorithm that starts with random weights, then, to emulate natural selection, keeps only the genomes with the highest fitness values for the next generation. Genomes are also paired for recombination and modified with a certain probability by random mutations. This can keep going for many generations until an acceptable behavior is found. In the end, if this process has taken place in simulation, the best genome can be employed in the real robot. Robots were able to develop a collision-free behavior because evolution exploits a characteristic of the robot body, favoring those individuals that moved forward, where it has more sensors. Robots also limited their own speed because sensory and motor refresh the rate at the same time and this means that a fast navigation had higher probability to lead to collisions with obstacles not detected in time. In conclusion, this shows that Evolutionary Robotics is an adaptation process that is able to achieve the desired behavior exploiting both the morphological and mechanical features of the robot, taking into account how the robot interacts with the environment [24] [23]. Thanks to this approach it is possible to obtain complex behaviors from simple solutions exploiting emergent behaviors [24].

2.1.4 The reality gap problem

Even though Evolutionary Robotics can be implemented directly on physical robots, the evolution process is often carried out in simulations because it could take a really long time otherwise [24] [19]. Moreover, there is a high risk of damaging the robot especially when testing random and poorly fitting behavior in the first generations. Human intervention is needed to change the environment back to the original configuration but it is still difficult to do all the experiments with the exact same conditions. For these reasons, the strategy is to evolve the controllers in simulation and to transfer them in the physical robot only at the end of the process. This leads to the reality gap problem: programs that perform well in simulations could not transfer well in the real world. This problem is accentuated in Evolutionary Robotics because it is an adaptive process that evolves the controller and/or the morphology of the robot based on the interactions with the environment and this could lead to the exploitation of every possible simplification or inaccuracy of the simulation. This issue can be mitigated by the use of accurate simulators, noise or online learning but it needs careful considerations [25].

2.1.5 Evolution and learning

Evolution and learning are both forms of biological adaptation, yet they differ in several ways [24]. On one side, evolution works on a population of many individuals to face slow environmental changes that could affect different generations. On the other side, learning modifies the behavior of a single individual during its lifetime and adopts many strategies to help it to adapt to new situations that can not be predicted at the generational level. The idea is to combine evolutionary with learning techniques to better adapt to quick changes that evolution could not overcome in time. This can be achieved building an evolutionary neural network using a fixed learning rule or it can be expanded evolving the way different rules are chosen. With only evolutionary techniques we obtain robots that are tailored to the environment used for the training while the goal of this strategy is to obtain a robot that knows how to adapt to partially unknown environments. For Eiben, learning is a must-have feature for robots that evolve in real-time [15]. In the robot, body and brain need to evolve together but the two could not work well together, and learning could mitigate this. In particular, in the infancy phase of the triangle of life the newborn robot should learn how to optimize, through trial-and-error learning, its performance and its inherited brain to better suit its body. Learning is also contemplated in the maturity stage, the ability to evolve and adapt after could improve the robot obtained after the offline optimization. In this case the robot learns/evolves while it operates [20].

Evolutionary robotics vs. reinforcement learning

Evolutionary Robotics and Reinforcement Learning are inspired by different principles but they both want to find the policy that leads the agent, acting in the environment, to the maximum reward [24]. The main difference is that in Evolutionary Robotics only the global value is considered while Reinforcement Learning works also with state-action pairs. This means that in Evolutionary Robotics all the information obtained during the life of the individual is lost and this makes it potentially weaker than Reinforcement Learning. At the same time, Evolutionary Robotics works better when facing partial observability and continuous domains such as robot control where it is difficult to define actions and states. In some cases, a convergence between the two methods for behavior design is possible [24]. However, Reinforcement Learning is not a good choice to automatically design the morphology of the body of the robot.

Online learning

Evolutionary Robotics often relies on a two-step offline approach where optimizations and evaluations only happen in the design phase which is then, followed by the operational phase [19]. This strategy is pretty common in engineering but it relies on the assumption that the environment will remain stable after deployment, or that the robot is able to deal with the changes because it has developed a robust behavior or it is equipped with the ability to learn. However, this is often not true because sometimes it is not possible to know in advance in which environment the robot will be deployed or it could change drastically over time. To solve this issue, the focus has shifted towards online versions of Evolutionary Robotics. These techniques keep the typical benefits of the offline counterparts while adding some unique features: each robot is situated in the environment and the optimization should not rely on external intervention. The robot moves around, this means that each evaluation starts with different initial conditions and, in addition, safety, in an unknown environment, is not guaranteed.

2.2 Online adaptation

It is clear from the issues highlighted in the previous section that, in order to develop autonomous robots, we must first build adaptive systems capable of continuously modifying their behavior at runtime in response to real-time feedback [26]. One of the possible ways to achieve this is through Online Adaptation where controllers are equipped with different mechanisms that enable the robot to change its behavior and adapt to new and challenging situations. For example, Online Adaptation can mean simply switching behaviors from a set of predefined ones or optimize some parameters of a neural network online. A performance function is needed to avoid ineffective behaviors and guide the process towards the design of an effective controller. The final goal is to maintain or even improve the performance during the entire lifetime of the robot and this never-ending nature of adaptation is what mostly distinguishes it from online learning [26].

The authors in [26] and [27] rely on an online and purely adaptive mechanism with no foreknowledge nor predefined behaviors, meaning that the robot needs to learn everything from scratch. The approach derives from evolutionary robotics but presents some important differences. First of all, the focus is on the phenotype instead of the genotype and the target of the optimization is a single individual instead of an entire population, as a consequence, it is not possible to rely on crossovers but only on mutations. In addition, the only information about the goal is provided by the fitness function that is evaluated during the entire epoch. The idea is to perturb the best found solution in the hope that minor changes lead to the exploration of similar and better behaviors [26].

Differently from evolutionary robotics, in [27] the focus moves from the genotype to the phenotype and in particular on phenotypic plasticity. In biology, a genotype with this property can generate many different phenotypes based on its development and on the environment in which it is situated. This permits to choose the phenotype that more suits the situation, favoring its affirmation on the evolutionary scale. As a consequence, this property is fundamental for adaptivity and provides an organism with the ability to build an internal model of the external world filtering only the information needed for its survival. This work uses Boolean Networks as controllers and an online adaptive mechanism with no mutations involved to obtain phenotypic plasticity in robots. The Boolean Network is generated at the beginning and only the couplings between the nodes and the sensors of the robot change. The robot automatically and continuously selects the best configuration based on the interaction with the environment and on a fitness function that acts as a selective pressure that guides the adaptation process. This work has been expanded in [28] to test also adaptive mechanisms that consider changing the internal structure of the boolean network with mutations by randomly negating some truth values or randomly changing the connections between nodes. The couplings between nodes and actuators were also affected. The robots performed well in different tasks such as navigation with collision avoidance and foraging. In addition, they gained insights on phenotypic plasticity showing that criticality can lead to the emergence of this property and to higher performances. These findings fuel the criticality hypothesis that states that critical systems are capable of balancing robustness to external perturbations and adaptivity.

In [29] and [26] the same online adaptation mechanism is shown to be effective with a different controller, Nanowire Networks. In this case, an offline approach would be infeasible due to the immutable nature of these networks. The choice to use Nanowire Networks is justified by the long term goal of building miniaturized robots that are capable of autonomously operate in unknown environments. In the paper, the local adaptation strategy performs a guided adaptation where the best solution found so far is perturbed to explore similar and hopefully better configurations. One of the limitations of this approach is that the starting configuration could be very far, in the landscape of possible solutions, and different from a good one and this could trap the adaptation process into local minima or require many edits. The proposed workaround could rely, at the beginning, on a random exploration that affects all the couplings until a good enough solution is found. Once such a solution is found the local adaptation process begins. The strategy proposed has been proved capable of adapting and maintaining good life-long performances in two different tasks: the t-maze task and foraging.

2.3 Robots that deal with damages

For a robot to be truly autonomous, it must deal with damages because they represent a concrete risk, especially when a robot is supposed to act in a complex or natural environment [30]. As a consequence, it is fundamental to increase robots resilience to faults because a single damage could compromise the entire system, especially in isolated environment where no human intervention is possible [31].

The ideal scenario is to create robots that can adapt like animals: robots should be able to find a compensatory behavior for many kinds of damage in the same way that animals can quickly adapt when injured [30].

In evolutionary robotics the self-repairing snake is an example of how it is possible to recover from damage [32]. The snake robot is built using shape memory alloy as muscles and an evolutionary algorithm that is able to find the proper sequence of muscle activation to compensate and adapt to the damage. The muscles of the snake are made of nitinol, a smart material that can change shape when the current passes through it. In particular, these nitinol wires can shorten at the expense of getting thicker to conserve the overall volume. Strings of bits encode Finite State Machines that describe the sequence of muscle activations at each time slot that is needed to make the robot move. Repeating patterns are very common and the genetic algorithm works on them with selection and mutation operators to evolve new and hopefully better solutions that enable the snake to travel longer distances. In the experiments the most used muscles are damaged and the evolution process is shown to be able to successfully adapt and move the robot relying on the remaining muscles.

A clear effort towards damage recovery is provided by Bongard's research [33] [21]. Here, a four-legged robot is able to overcome unexpected damages relying on continuous self-modeling to keep moving forward. In particular, the robot indirectly builds a model of its own morphology relying on an algorithm divided in three phases, each one executing a different evolutionary algorithm. First, in the simulation, the robot performs an action and records the data coming from the sensors and this information is used to generate a set of candidate self-models that predicts the sensory-actuation causal relationship. Then the models are used to select a new action, specifically the one expected to maximize the information that can be extracted about the way the robot interacts with the environment. This is determined by selecting the action for which the disagreement between the predicted sensory values is the highest. The action is executed on the real robot and the models updated with the new information. These two phases are alternated for several rounds and, in the final phase, only the most accurate model is used to evolve the desired behavior for the robot. Thanks to this model the robot is able to identify a damage by detecting unexpected sensor-motor patterns. When this happens the three phases are repeated to produce new models to cope with the changes and produce compensatory behaviors. In the experiments, a four-legged robot is damaged by shortening a leg and this technique show that the robot is able to successfully detect the damage and find new strategies to keep moving forward despite the damage.

The goal of the algorithm proposed in [30] is to obtain more robust and autonomous robots that are able to quickly adapt to damage thanks to a trial-and-error learning algorithm. The algorithm does not rely on a self-diagnosis phase followed by the selection of a contingency plans among some pre-defined ones, instead, there is an offline stage where the robot builds a behavior-performance map. The map contains all the thousands possible behaviors it can perform and a prediction of their performance. This map represents the robot's intuition on what can perform well and it is used as previous knowledge that guides the trial-and-error algorithm. In particular, it helps it choosing which behavior leads to better performances and in particular which is the best to compensate for the damage the robot has received, or for the drop in the performance. The map is created via the MAP-elites for which designers need to describe the space of possible behaviors considering only few dimensions along which behaviors vary. They need also to define a way to compute the performance. Then, in the simulation, this space is automatically searched to find a high-performing behavior for each point in the space or for each behavior type and to create the map. When the damaged robot is deployed the adaptation process starts a Bayesian optimization algorithm that intelligently selects a behavior from the map. This choice tries to balance exploitation of configurations that are predicted to perform well and exploration of those configurations where the confidence in the performance prediction is low. Once the behavior is tested on the physical robot the map is updated and the process is repeated until a behavior for which the performance of the damaged robot is good enough is found. This technique has been employed on two complex robots: a hexapod robot and a robotic arm. The hexapod robot is tested to see if it is able to cope with different potential damages such as a shortened leg, unpowered leg, missing leg, modified leg and even two missing legs. The robot is able to successfully choose those behaviors that do not rely on the damaged legs very often. The robotic arm is tested to see if it is able to release a ball into a bin with broken joints that block parts of the arm at different angles. Even in these scenarios the robotic arm is able to complete the task with a good accuracy. All these experiments show that this approach permits the robots to adapt to new environments and also to successfully cope with damage like animals with no need to specify a predefined strategy for all the kind of potential damage.

Another approach that does not rely on self-modeling or on any kind of a-priori knowledge is the work of Husbands's research group [34]. They focus on the exploitation of complex and embodied dynamics to generate adaptive and resilient behaviors. They believe that a robot equipped with suitable adaptive mechanisms can achieve the desired behavior by learning how to efficiently exploit its own embodied neuromechanical dynamics, especially chaotic ones. In particular chaos can lead to the exploration of many possible motor patterns. They developed artificial neural networks based on coupled oscillators to use them as controllers for robots. However, chaos is not originated by the controller alone, the overall system contributes to the generation of chaotic dynamics which comprehends also the body of the robot and the way it interacts with the environment. For example, sensory feedback has a major role in this as well. The core idea is to rely on a mechanism called "homeostatic sensory adaptation" that makes it possible to control the level of chaos of the system. In particular, this permits the robot to lower the level of chaotification when the performance is high and to switch to a higher level when the performance drops, for example because the robot has been damaged. In addition damage changes the body of the robots and the sensory inputs and this overall leads to more chaos that favors the exploration of new solutions useful for finding a compensatory behavior. The experiments involved a robotic 3-arm swimmer and also a four-legged robot that needs to learn how to coordinate their limbs to move straight ahead. Both the robots were able to successfully keep moving forward after one of the limbs was removed showing that it is possible to recover from the damage in real time.

Due to the importance of resilience to damage and damage recovery in the development of increasingly autonomous robots, these properties are also investigated in robots with minimal computational capabilities that are subject to Online Adaptation. For example, the authors in [31] investigated on how the performance of a robot controlled by a Boolean Network changes in relation to the number of faulty sensors. The experiments involved a simple foot-bot robot in a phototaxis task considering different types of damage: sensors when damaged stop producing any output or start sending wrong random values. The results show that Online Adaptation leads to a graceful degradation of the performance because it decreases slightly with the number of faulty sensors, showing that the robot is able to partially adapt after the drop in performance even though some information is permanently lost and it requires more time to improve when many sensors are missing. There are some interesting exceptions: if the damaged sensors are not many and especially when they are contiguous the robot can even benefit from this loss and increase its performance, probably because it permits to focus only on the relevant information. These results show that Online Adaptation can be the key for adaptation and achieving fault tolerance. This belief motivates further investigations on Boolean Networks and automatic fault recovery driven by an Online Adaptation process [18]. The previous work is expanded with some improvements on the adaptation mechanism that tries to overcome a fault starting from an existing controller instead of creating a new solution from scratch. In addition, a fitness function helps identifying faults and when the robots needs to find more effective behaviors. Moreover, the robot has been tested not only on a phototaxis task but also in a collision avoidance task and also with different kinds of damage. The authors not only tested different kinds of faulty sensors but also damaged wheels reducing the maximum reachable speed. The results demonstrated that Online Adaptation can successfully recover in most cases the performance of the robot to a level comparable to the one achieved before the damage.

Chapter 3

Robots equipped with innate instincts and reflexes

Many studies and ideas in the robotic field take their inspiration from nature and biology. For example, researchers have put a lot of effort in the design of robots whose morphology resembles animals such as hexapod robots based on insects [30] [24], robots that look like snakes [35] or birds [34]. These ideas impact not only the body of the robot but also its controller: this is particularly evident with Evolutionary Robotics although other bio-inspired approaches exist [34].

As illustrated in chapter 1, animals of different species and humans possess useful innate instincts and reflexes. For many of these mechanisms, the main function is to protect the organism from potential danger, enhancing its chances of survival. Since innate instincts and reflexes play a fundamental role in nature, we believe that robots could also benefit from possessing similar capabilities. Therefore, the purpose of this thesis is to explore the advantages that the integration of innate safety measures, inspired by instincts and reflexes, can bring to robotic systems. We believe this to be particularly useful for adaptive robots to protect them from runtime damages especially during the exploration of different strategies, that could lead the robot into dangerous scenarios.

3.1 Proposed instincts and reflexes

In order to equip a robot with instincts and reflexes, we can not simply copy the ones that exist in nature but we can use them as a source of inspiration to design new ones tailored specifically for a robot. Therefore, the first step of this work is to devise some mechanisms that could help in the prevention of some common damages that a simple wheeled robot with sensors could sustain during the execution of a task. Some damages could not only stop the robot from completing its current task but they could also permanently break it and compromise future deployments of the same robot. In this case, the only viable solution would be to replace the robot, however, this comes with substantial costs and is almost impracticable. This is the reason why it is so important to avoid danger as much as possible.

In the following sections, we describe the proposed instincts and reflexes.

3.1.1 Avoid falling

The purpose of the Avoid Falling mechanism is to prevent the robot from falling off edges as well as from sliding into gaps or holes in the terrain. We can think of this mechanism as an innate instinct that makes the robot wary of edges while keeping a safe distance from them and avoiding any situation where it is not possible to detect a surface or the ground below. It is clear that this instinct is extremely important to effectively save the robot because a wheeled robot that falls, especially from significant heights, could be severely damaged. In addition, even if no damage occurs, the robot could land in a position from which it can not move without external help, for example upside-down.

For all these reasons we think that any moving robot would benefit from the possession of this instinct. The only requirement for its implementation is to have a means to detect edges or holes that in the most straightforward way can be provided by common infrared ground sensors.

3.1.2 Avoid fast crashes

The idea is that a robot equipped with the Avoid Fast Crashes mechanism prepares for imminent impacts by significantly slowing down when approaching nearby walls or obstacles. This can be compared to those fast reflexes that trigger defensive reactions. This reflex is fundamental to protect the robot from the damage that could otherwise result from collisions with obstacles at full speed.

This mechanism can be implemented in any robot that can correctly detect walls and obstacles in front of it. For example, exploiting standard proximity sensors can be enough.

We have also considered the possibility of completely avoiding an obstacle but we choose this alternative because it is more general and suits more tasks. For example, if the robot needs to perform a foraging task where it is necessary to pick up an object, the object could be detected as an obstacle and the stricter mechanism, while possibly be more efficient in protecting the robot, it would stop the robot from reaching its goal. Instead, our mechanism permits the robot to approach objects more cautiously, enabling it to continue performing the task.

3.1.3 Avoid overheating motors

As the name suggests, the goal of the Avoid Overheating Motors safety mechanism is to prevent any damage that the robot could sustain as a consequence of an excessive overheating of motor components. The occurrence of this scenario can be prevented by monitoring the robot and by choosing not to always move at full-speed. This can be seen as an instinct that protects the robot by telling it that it should slow down after going at maximum power for too long. This mechanism not only keeps the robot safe from potential damages, but it can also permit to save energy. In a robot with temperature sensors located near the wheels it could be possible to detect and react directly to overheating.

3.1.4 Avoid extreme temperatures

The Avoid Extreme Temperatures mechanism should lead the robot away from areas in the environment that are exposed to extremely high or low temperatures outside the range of intended operation, between -40 and +70 Celsius degrees. A robot that stays in these zones for too long would see its components starting to have issues and deteriorate. Not only that, the same mechanism can be used to keep the robot away from fire or any other source of heat and this is crucial especially because batteries could explode. This mechanism can be associated with a reflex reaction triggered by pain because the robot equipped with this innate reflex would start retreating or escaping from the current location as soon as the temperatures are not bearable anymore.

In order to implement this reflex we need to build a system that is capable of monitoring the temperature at the robot's position. This can be easily achieved even in simple robots thanks to temperature sensors.

3.1.5 Avoid excessive tilts

We have also thought of an Avoid Excessive Tilts mechanism. The idea is to detect when the robot is dangerously tilted because when this happens the robot risks to tip over. This could damage the robot or stop it completely when it is not possible or desired to rely on external help to put the robot back in the right position. This works like an instinct that lets the robot know that certain surfaces are not safe and should not be navigated. For example, a robot equipped with this instinct would keep its distance from slopes or irregular terrains and lead it towards plain or safe zones.

It can be quite complex to equip a robot with this instinct because it requires a component that is able to correctly detect when the robot tilts. In simple robots this can be achieved by exploiting the accelerometer [36] they are often provided with but it can be not very precise in identifying slopes because, in a wheeled robot, some of its values can significantly vary even when the robot is simply turning around or bumps into something. For this reason, it is often combined with a gyroscope.

3.1.6 Avoid humidity and water

This Avoid Humidity and Water mechanism is very similar in principle to the Avoid Extreme Temperatures one. It is obvious how much damage a robot, equipped with a lot of electrical components, could receive after operating in a very humid environment or, even worse, after running into a pond of water. This could permanently impair the robot leaving us with no other choice than to replace it with a new one. For this reason, we can think of this mechanism as a reflex triggered by pain that keeps the robot safe from any situation that could get it wet.

The most basic implementation, available even for simple robots, could rely on humidity sensors to determine dangerous zones. However, water identification is not that straightforward.

3.2 Experimental environment

In this section we describe the environment used for this thesis. In particular, we have worked with a Thymio [37] robot operating inside the Webots Simulator [38].

3.2.1 Thymio robot

Thymio [39] is a simple and small mobile robot created by Mobsya for educational purposes. It is designed to help anyone discover technology and get closer to robots and for this reason it comes with preprogrammed behaviors and interfaces that allow to program the robot in an intuitive way by means of image blocks.

As illustrated in figure 3.1 the Thymio robot comes with many different sensors and actuators.

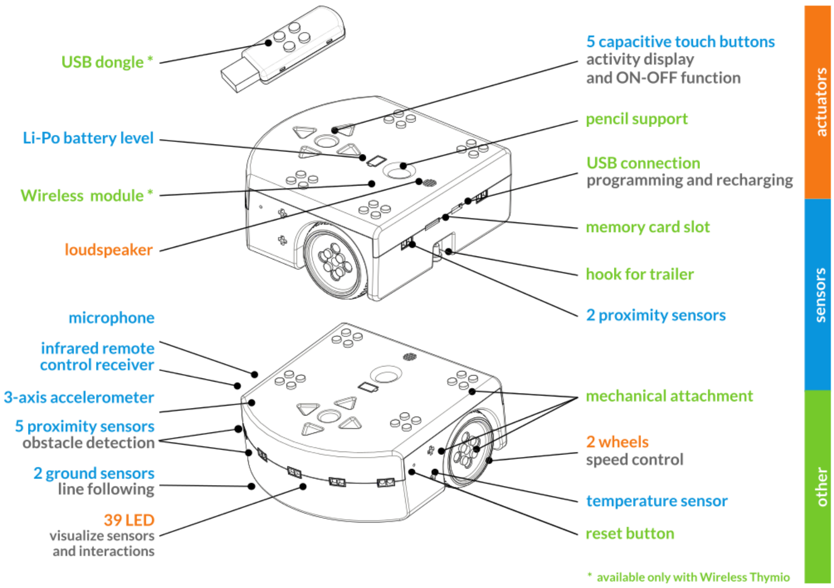

Figure 3.1: Thymio Specifications. Source: `http://aseba.wikidot.com/en:thymiospecifications`

In the following, we provide a filtered list that contains only those components that are relevant for our experiment:

- sensors:
  - 9 infrared sensors (range approximately 10 cm):
    - 5 proximity sensors placed in front of the robot
    - 2 proximity sensors placed in the back of the robot
    - 2 ground proximity sensors
  - 1 three-axis accelerometer
  - 1 thermometer
- actuators:
  - 2 direct-current motors linked to the wheels

3.2.2 Webots simulator

The Webots simulator is an open source and multi-platform desktop application that provides a complete development environment to model, program and simulate robots. It has been designed in 1998 by Cyberbotics Ltd. that still maintains it and updates it with new features. It has been widely adopted by professionals in industry, education and research [38].

We choose to work with the Webots simulator for its reliability and because it is one of the best available options to work with an accurate model of the Thymio II robot where almost all the sensors and actuators have been modeled and calibrated [37]. In addition, it allows to test robots in custom 3D environments with complex physics.

The main downside of working with the Webots simulator, besides the problems linked to the reality gap, is that the migration of the code towards a real Thymio robot is not straightforward. For this reason, we first considered the Playground simulator that is distributed with the Thymio Suite [40], the official set of tools to create a program for Thymio. However, this simulator is too simple and not very precise: for example, it is possible to work only in a 2D environment and both the accelerometer and thermometer are not simulated. As a consequence, it is impossible to create arenas with holes or slopes and to rely on these sensors preventing us from implementing most of the instincts and reflexes proposed. For these reasons, the Webots simulator better suits our needs.

To be precise, we have worked with an extension of the Thymio model provided by the Webots simulator. We need this extension to provide the simulated Thymio with sensing capabilities that are present on the real robot but not in the Webots version and also to enable it to better detect and react to danger.

The first workaround concerns the thermometer or temperature sensor because it is not supported in Webots. We simulate this adding a GPS sensor to Thymio and assigning temperature values to each point in the arena. In this way it is possible to simulate how a thermometer works by mapping each set of coordinates returned by the GPS to a temperature value.

The second extension is related to the ground proximity sensors. These sensors are centered and placed only on the front of the robot and this means that when Thymio moves backwards is left completely defenseless. To avoid this situation we extend Thymio by adding two ground proximity sensors located towards its back. We imagined to add these sensors within a bumper component that extends the sensing capabilities of the original Thymio, enabling it to properly detect dangerous edges on time when moving backwards. For this reason, in the simulation they are placed outside the chassis of the robot body. In addition, the same additional component is used to provide the robot with ground proximity sensors located towards the sides, specifically in front of the wheels. This also augments the robot capabilities and helps the robot to react before one of the wheels is already partially over the edge.

3.2.3 Proxy

Since we are working on the Webots simulator, it is not possible to directly use the code produced for the simulated Thymio on a real Thymio. We mitigate this problem by making the implementation of some instincts and reflexes reusable. For this purpose we have written the code in Python, supported in both cases, and created a Proxy Interface containing methods that the two Thymio robots should implement. These methods encapsulate the way the robot gets access to its sensors and actuators.

3.3 Control system architecture

For the architecture of our control system we have decided to rely on motor schemas, a concurrent and distributed model of computation [41, 42]. Thanks to motor schemas it is possible to build complex behaviors combining simpler ones in a cooperative way with no predefined hierarchy allowing each behavior to contribute to the robot's overall response. The general idea, shown in figure 3.2, is to adopt a modular approach that requires, for each behavior, to design a motor schema and the corresponding perceptual schemas containing all the relevant information about the environment. Each motor schema outputs an action vector that provides magnitude and orientation to describe how the robot should move. In the end, all these vectors are summed up together to produce a final vector that is actually used to move the robot.

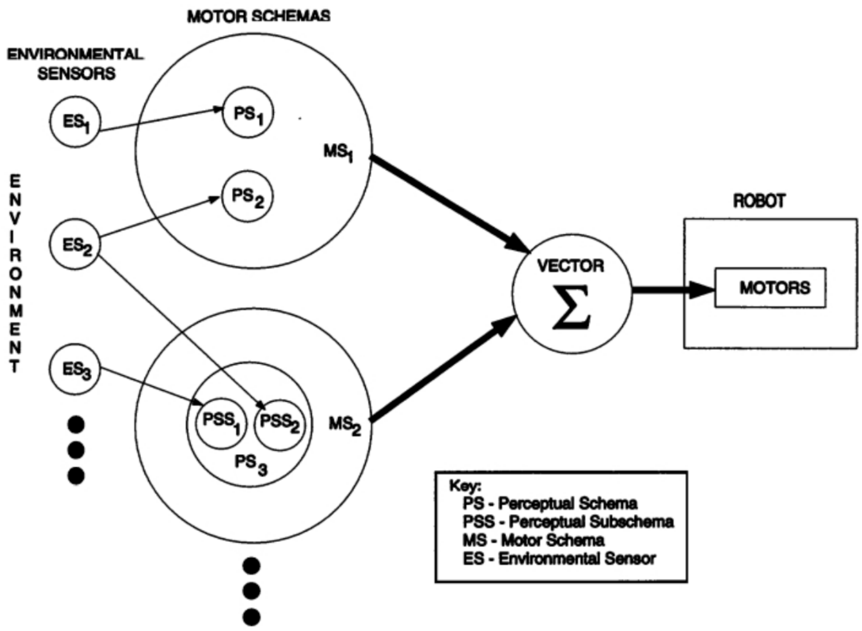

Figure 3.2: Relationship between motor schemas and perceptual schemas. Source: Ronald C. Arkin. Behavior-Based Robotics. The MIT Press, 1998

In order to follow this idea we need to create a motor schema for each instinct and reflex. Each defensive mechanism protects the robots from a single potential damage, for this reason, the corresponding motor schema considers only the sensors and the information that allows it to perceive and discourage a single dangerous behavior telling how the robot should move to avoid it. However, combining all of them, it is possible to keep the robot safe from multiple kind of damage. This allows to create a module of instincts and reflexes that can be combined, always by means of motor schemas, with already existing robot control policies meant to complete a certain task or reaching a goal. In order to do this, the original robot control policy needs to be expressed by means of motor schemas and produce an action vector. In this way, the addition of this module augments the original behavior with a set of safety mechanisms.

It is worth mentioning that different implementations are possible, for example we could use an instinct or reflex to directly control the wheels of the robot according to a set of rules. However, our approach allows to write a more elegant and modular solution that makes it easier to add, remove or modify a mechanism. In addition, it would also be more complicated to inject instincts and reflexes on an already existent task-achieving behavior while, thanks to this accurate organization, it does not require much effort. For example, it is possible to combine the safety module with neural network as long as it is possible to interpret its output as a polar vector.

At each step, all the action vectors are summed up together and the resulting vector needs to be used to give commands to the wheels. If the robot is controlled with a roto-transational model it can be used directly as it is considering the magnitude as the desired speed v and the orientation as the rotation angle w. However, in our case, we need a conversion from the roto-transational model to the differential steering model that transforms the vector with polar coordinates into velocities to assign to each wheel. In our specific case, we use the following formulas to transform v and w into the speeds to assign to the left wheel v_l and to the right wheel v_r:

$$v_l = v \cdot \cos\left(\omega + \frac{\pi}{4}\right)$$

$$v_r = v \cdot \sin\left(\omega + \frac{\pi}{4}\right)$$

These trigonometric transformations work with only positive values for v, while w can assume any value between $-\pi$ and π and, as a consequence, will determine whether the speeds will be positive or negative. Each combination is possible and so the robot can go forth or back by moving straight or rotating and also rotate in place.

3.4 Implementation of instincts and reflexes

In this section we describe the implementation details of each defensive mechanism. We implemented all the mechanisms proposed in section 3.1, with the exception of the Avoid Humidity and Water mechanism. None of the mechanisms involve learning, the idea is that, since instincts and reflexes are innate in nature, they should be ready to use and present in robots since the beginning.

3.4.1 Avoid falling

The Avoid Falling mechanism is implemented relying only on the ground proximity sensors of the extended Thymio described in section 3.2.2. A threshold has been set empirically based on experimental observations and determines the activation of the mechanism: when the values perceived by the proximity ground sensors surpass the threshold, this means that the robot is approaching a dangerous edge and that the safety mechanism should intervene. When this happens, the motor schema produces a vector representing a repulsive force going on the opposite direction and with magnitude equal to the maximum possible speed. The vector will point backward (-*pi* angle) if the danger is detected by the front proximity ground sensors and forward (0 angle) if the activation is determined by the ones on the back. In all the other cases a zero vector is returned.

Figure 3.3 shows that the control policy (in red) is not affected by the mechanism when the robot is not in danger, as a consequence the resulting direction vector (in blue) is equal to it. Instead, figure 3.4 shows the influence of the Avoid Falling mechanism (in green) and how the resulting direction vector successfully push the robot away from the border.

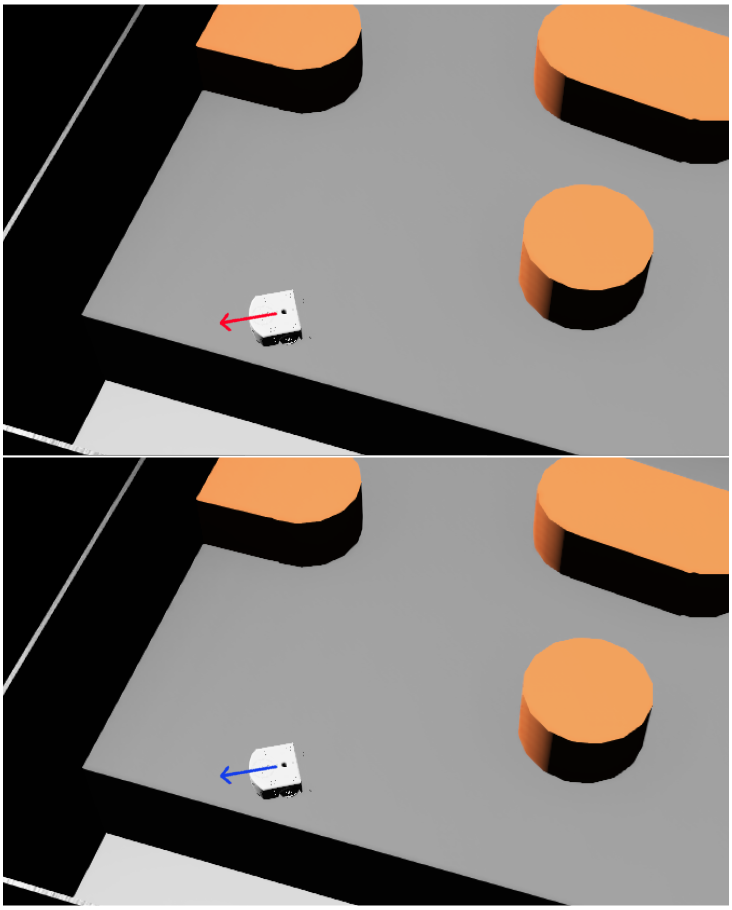

Figure 3.3: Behavior of the robot when it is approaching the edge but danger has still to be detected. The red vector representing the robot control policy and the blue vector representing the resulting vector are the same.

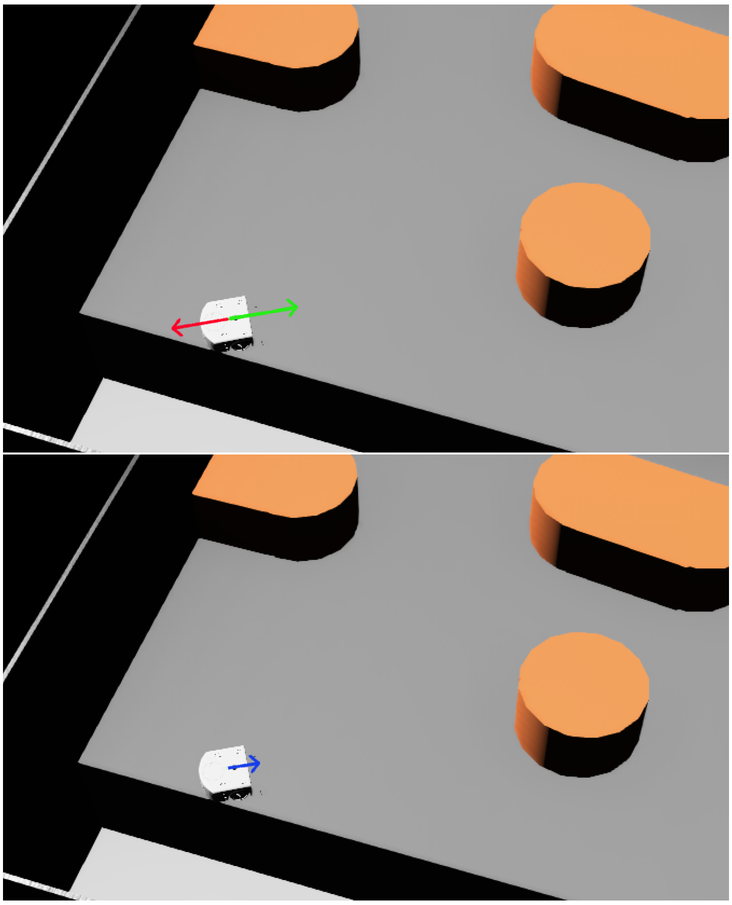

Figure 3.4: Behavior of the robot when it is approaching the edge and it is in danger. On the top, in addition to the red vector of the robot control policy, there is also the green vector that describes the action of the Avoid Falling mechanism. On the bottom, it shows how the resulting direction vector in blue pushes the robot away from the danger.

3.4.2 Avoid fast crashes

The Avoid Fast Crashes mechanism is implemented exploiting all the proximity sensors of Thymio. The threshold, empirically set, determines whether or not the safety mechanism should be activated: if one of the values perceived by the proximity sensors surpass the threshold, this means that the robot is going to hit an obstacle and it should prepare to the impact by slowing down. For this reason, the motor schema returns a vector representing the repulsive force exerted by the obstacle on the robot. The vector will point backward (-*pi* angle) if the danger is detected by the front proximity sensors and forward (0 angle) if determined by the sensors on the back. The magnitude equals to half the maximum speed because we do not need to discourage the behavior completely but just decrease the speed of the robot. In all the other cases a zero vector is returned.

Figure 3.5 illustrates how the behavior of the robot is not affected when it is going towards an obstacle that is quite distant and does not represent a risk. However, as shown in 3.6, when the robot is going to hit the obstacle, the Avoid Fast Crashes mechanism takes action. As a result, we see that the robot still hits the obstacle but at a slower speed, preventing it from taking more severe damage.

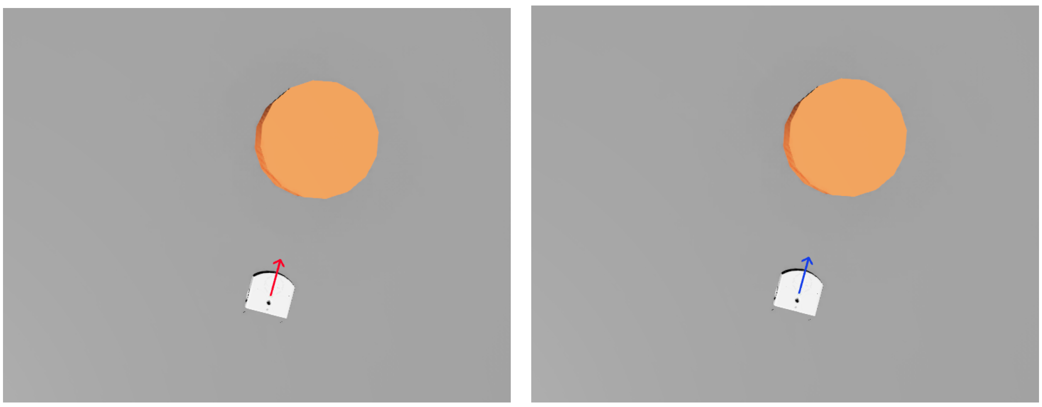

Figure 3.5: Behavior of the robot when it is approaching an obstacle that has yet to be detected. The red vector representing the robot control policy and the blue vector representing the resulting vector are the same.

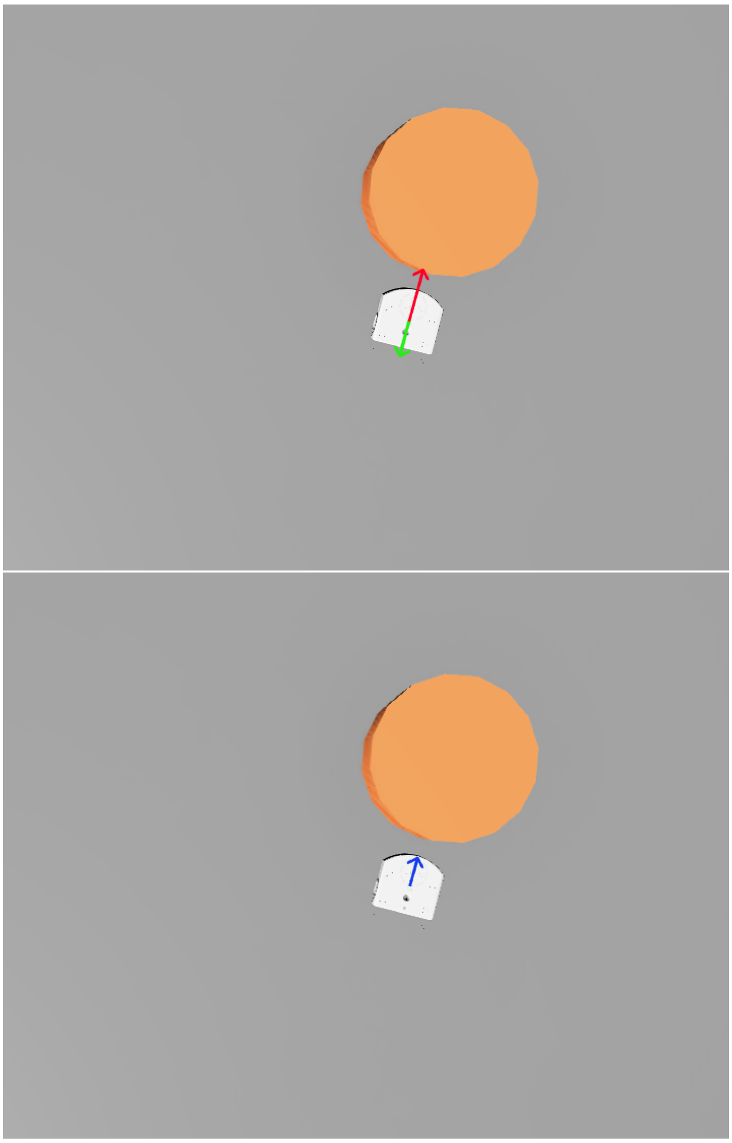

Figure 3.6: Behavior of the robot when it is dangerously near the obstacle. On the top, In addition to the red vector, representing the robot control policy, there is also the green vector that describes the action of the Avoid Fast Crashes mechanism. On the bottom, it shows how the resulting direction vector in blue lets the robot go forward but at a slower pace.

3.4.3 Avoid overheating motors

In order to implement the Avoid Overheating Motors mechanism we check the wheel velocities and count the number of steps to determine how long the robot has moved at maximum speed without never slowing down. The amount of time that is considered dangerous for the robot has been arbitrarily chosen and can be changed based on the robot needs. When the robot surpass the threshold the mechanism protects the robot by forcing it to slow down by a certain amount of time. The time that the robot needs to cool down the motors is also set arbitrarily and can be modified. In terms of motor schema this works by producing, for each step of the cool down period, a vector that goes in the opposite direction to the robot's motion. Once enough time has passed, it restarts returning zero vectors.

In figures 3.7 the robot control policy is not influenced by the mechanism until the robot has passed too much time at maximum speed. As shown in figures 3.8 to prevent the motors from overheating the mechanism makes the robot decrease its speed until it has cooled down.

Figure 3.7: Behavior of the robot when it is going forward at maximum speed for a duration that does not pose a danger for the motors. The red vector representing the robot control policy and the blue vector representing the resulting vector are the same.

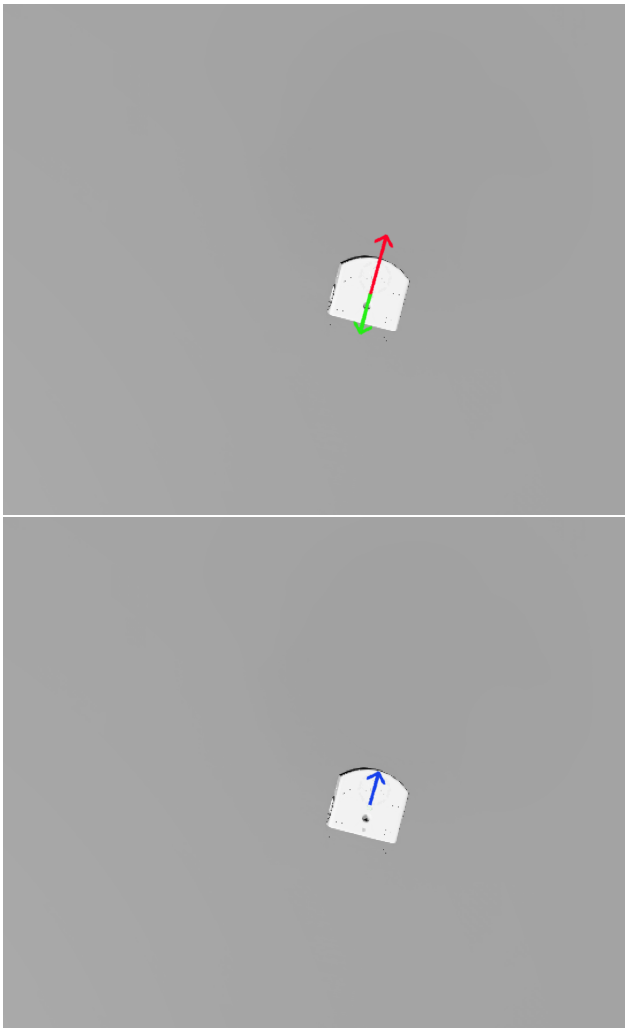

Figure 3.8: Behavior of the robot after it has moved at maximum speed for too long. This can damage the motors and thus the robot has to slow down for a while. On the top, in addition to the red vector representing the robot control policy, there is also the green vector that describes the action of the Avoid Overheating Motors mechanism. On the bottom, it shows how the resulting direction vector in blue lets the robot go forward but at a slower pace.

3.4.4 Avoid extreme temperatures

The Avoid Extreme Temperatures mechanism is implemented exploiting only the simulated temperature sensor or thermometer of the extended Thymio described in section 3.2.2. The idea is to protect the robot from going into areas where the temperatures are too high or too low, so it is possible to set two different thresholds. When the temperature at the current location of the robot surpass the threshold this means that the robot is entering a very dangerous area. As a consequence, the safety mechanism is activated and produces a vector in the opposite direction and with maximum speed as a magnitude to push the robot away from the zone. In all the other cases the motor schema returns a zero vector.

In figures 3.9 the safety mechanism does not have any effect on the robot control policy as long as it navigates areas with safe temperatures. However, as shown in figures 3.10 the Avoid Extreme Temperatures mechanism keeps the robot away from zones with high temperatures that can cause damage to the robot.

Figure 3.9: Behavior of the robot before reaching an area with dangerous temperatures. The red vector representing the robot control policy and the blue vector representing the resulting vector are the same.

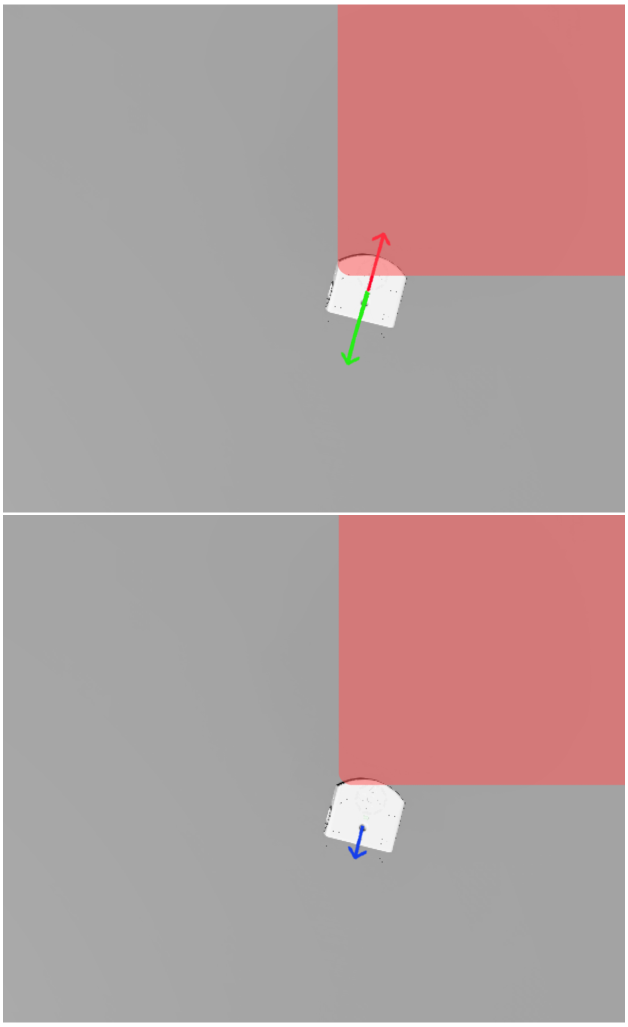

Figure 3.10: Behavior of the robot after it has reached an area with dangerous temperatures. On the top, in addition to the red vector representing the robot control policy, there is also the green vector that describes the action of the Avoid Extreme Temperatures mechanism. On the bottom, it shows how the resulting direction vector in blue pushes the robot away from the area.

3.4.5 Avoid excessive tilts

The Avoid Excessive Tilts mechanism is implemented relying exclusively on the values measured by the Thymio accelerometer. As explained in section

3.1.5, detecting tilts by using only an accelerometer can be tricky and other methods and sensors could be more suited for the task. Nonetheless, this is the only way for Thymio to gain information about tilts and it is enough to demonstrate the usefulness of this safety mechanism. In order to detect a tilt we need all three values returned by the 3-axis accelerometer and use them to compute the pitch and roll angles [43]. In particular, the pitch angle tells if the robot is tilting up or down while the roll angle tells if it is tilting towards the left or the right. We set up a threshold based on the experiments and if one of the two angles surpass the threshold this is an indicator that the robot is tilting on a dangerous terrain and should avoid going forward. We consider both angles and also a counter to discriminate strong but quick accelerations from potential threatening tilts.

Figure 3.11 shows that the Avoid Excessive Tilts mechanism does not modify the robot control policy when the robot is not tilting and far from the dangerous surface, here in the simulation a ramp, that could cause the robot to tip over. Instead, in figures 3.12 we can see how the Avoid Excessive Tilts mechanism perceive that the robot is tilting when trying to go up the ramp and intervene to keep the robot away from it and stopping it from moving forward.

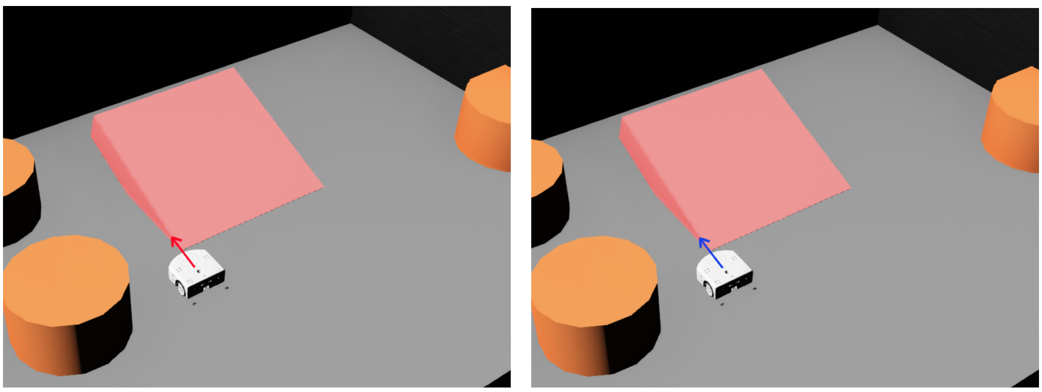

Figure 3.11: Behavior of the robot before reaching the ramp. The red vector representing the robot control policy and the blue vector representing the resulting vector are the same.

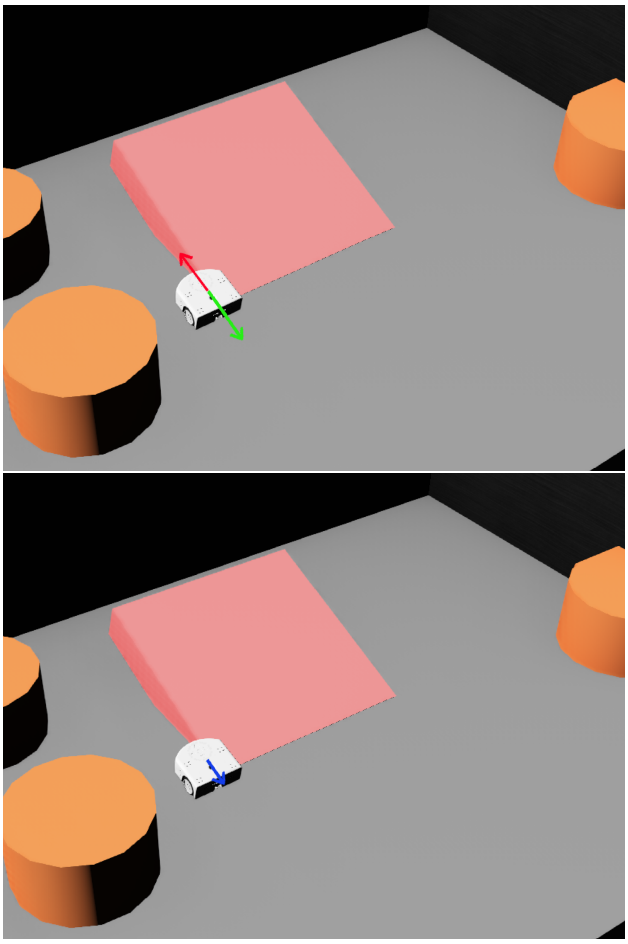

Figure 3.12: Behavior of the robot after it has reached the ramp and is trying to go up the ramp. On the top, in addition to the red vector representing the robot control policy, there is also the green vector that describes the action of the Avoid Tilts mechanism. On the bottom, it shows how the resulting direction vector in blue pushes the robot away stopping it from going up the ramp.

3.5 Online adaptation with instincts and reflexes

To better assess and test the usefulness of these mechanisms we have equipped them to a robot controlled by an Online Adaptation process that follows the same methodology presented in [26], [31] and [18]. In this section, we describe how the Online Adaptation works and how it is possible to combine it with the set of mechanisms that we designed based on instincts and reflexes.

3.5.1 Online adaptation process

The Online Adaptation Process relies on a simple feedforward Neural Network and its goal is to perform a collision avoidance task.

Neural network

The network has only three layers. The first one is directly connected to the sensory information and, therefore, to the proximity sensors that allow the robot to detect obstacles and walls. In this layer there is a node for each one of the proximity sensor and, at each step, the values of the sensors are normalized and fed as inputs to the network. The second layer is a hidden layer, here, the number of neurons is a parameter and can be tuned. Finally, the third and last layers have only two output neurons that, typically, are directly connected to the robot wheels.

Fitness function

The robot learns how to carry on the navigation with object avoidance task based only on the information provided by the fitness function and computes the performance at each step. The choice of the fitness function is considered the key components to evolve and automatically design controllers for robots that can adapt by expressing the task they need to achieve without telling how the robot should perform it [44]. The fitness function applied in this work derives from the one used in Floreano's experiments [17].

$$\Phi = V\left(1 - \sqrt{\Delta v}\right)(1 - i)$$

Where V is the average rotation speed of the two wheels normalized between 0 and 1. We consider positive velocities as they are and pick 0 for negative values to especially penalize when the robot chooses to move backwards and both speeds are negative. The next parameter, Δv, is the algebraic difference between the signed speed values of the wheels in absolute value. Finally, i is the maximum value among those measured by the proximity sensors normalized in the range [0, 1].

Online adaptation

The Online Adaptation process is carried out through successive epochs with a parameterized number of steps. At each step, all the values coming from the proximity sensors are computed and used as inputs to feed the Neural Network. The Network then computes the output, a polar direction vector that is used to move the robot. In addition, the fitness function is computed to measure the performance of the robot based on the values of the proximity sensors and the actual wheel velocities. The performance is updated at each step and is considered at the end of each epoch to determine whether the current evaluation is equal to or better than the previous one. In particular, to balance the trade-off between the exploration of new solutions and the exploitation of the best solution found so far, we alternate each epoch with an exploratory epoch whose goal is to search the space of possible configurations. Every even epoch starts an exploration to search for a better configuration, this is done by modifying the weights of the best network with random mutations. At the end of the exploration, if a better performance is achieved the new network is kept and a new evaluation begins. Specifically, this comparison is done at the beginning of the following (odd) epoch during which the best configuration is re-valuated.

Online adaptation equipped with safety mechanisms

Following the proposed architecture, integrating the designed safety mechanisms into the robot controlled by the Online Adaptation process is quite simple and straightforward and requires only minimal modifications. In order to do so, we just need to add, in each step, the computation of all the motor schemas related to the instincts and reflexes. Not only that, we also have to consider the output of the network as a separate motor schema. In this case the two values of the output are not used to directly control the wheels and move the robot. We have to interpret them differently, in particular, we need to obtain a the polar vector of direction and magnitude. This is done by considering as the direction angle the result of the multiplication between *pi* and the first output value. The second output value is also used as a multiplier for the maximum speed to determined the magnitude. Once we have done this, we just need to sum all the vectors, convert the resulting vector into wheel velocities to move the robot. Finally, the computation of the fitness function needs to be kept after the speed assignment so that the performance can be influenced by how the safety mechanisms affect the velocities.

3.5.2 Discussion

The integration of the innate safety mechanisms within a robot subject to an Online Adaptation process has proven to be useful in successfully avoiding damages. At the same time, it has also brought to light potential limitations and issues that require further consideration and future improvements.

The Avoid Falling mechanism successfully permits the robot to survive longer in an arena with a hole pushing it away from the cliff as soon as it is detected by the ground sensors. At first the robot fell when proceeding backward towards the edge because it had no way to detect it. For this reason and to better test the mechanism we considered the Thymio extension described in 3.2.2 that adds ground sensors behind the robot. Thanks to this the robot is protected both whether it is moving forward or backward. Despite this addition there are still some cases where the mechanism fails in saving the robot. This can happen because the robot detects an edge only when it is extremely close to it and this can limit the mechanism that can not remain active long enough or react in time to effectively push the robot away. We believe this can be mitigated by using a stronger magnitude to make sure that the safety measure prevails on other behaviors, or by making the effect of the mechanism last longer but these alternatives require further evaluations. Finally, a robot that is able to perceive a cliff before coming close to the border could certainly implement a more robust mechanism.

The Avoid Fast Crashes mechanism successfully makes the robot slow down when approaching an obstacle reducing the damage that would otherwise be caused by hitting the obstacle at a higher speed. The action of the Avoid Fast Crashes safety measure works correctly whether the robot is going towards an obstacle while moving forward or backward exploiting the proximity sensors located in the front and on the back of the robot. Even though the Online Adaptation process focuses on obstacle avoidance this mechanism is still useful to protect the robot when it is more vulnerable and more likely to hit walls and obstacles. This can happen especially when the robot is at the beginning of the adaptation process or during exploration phases where it can select poorly performance behaviors.

The Avoid Overheating Motors mechanism is able to force the robot to decrease its velocities after proceeding at full speed for too long. We believe that this kind of slowdown can be really helpful to prevent the overheating of the motors. However, the implementation provided here is quite simple and limited to when the robot is moving forward.

The Avoid Extreme Temperatures mechanism shows that it can successfully keep the robot away from the area with high temperatures in the arena. In particular, it prevents the robot from entering the dangerous zone as soon as the perceived temperature surpasses the threshold. The main limitation of our implementation is that the mechanism assumes that the robot is entering the area while facing it and moving forward. This means that the robot is not able to correctly avoid the areas with high temperatures when it moves backward. This is due to the fact that, with motor schema, we need to determine the direction of the vector field generated by the source of heat but we have no way to properly determine the orientation of the robot. In addition, the robot could stay near the area with harmful temperatures.

The Avoid Excessive Tilts mechanism correctly prevents the robot from moving forward when there is a risk of tilting and, consequently, of tipping over and taking damage. In the arena used for the experiments this translates as the mechanism stopping the robot from going up a ramp. As for the Avoid Extreme Temperatures mechanism, this one too has problems when the robot meets the ramp while moving backwards. The limitation is caused by the lack of information about the orientation of the robot with respect to the inclination that can not be inferred by the accelerometer values alone. In addition, as already anticipated before, the accelerometer is not very accurate for this task.

To summarize, the innate safety mechanisms have proved to be very useful for saving the robot from different sources of damage. The instincts and reflexes we have implemented are meant to protect the robot both from dangers it had no knowledge about and also to mitigate certain behaviors while it is vulnerable and adapting. The implementation and architecture proposed have shown to be more suited to integrate those mechanisms that not only can detect the danger but can also react differently based on which side the robot is facing the risk. For these reasons the Avoid Falling and Avoid Fast Crashes mechanisms that rely on different sets of proximity sensors, one in the front and one in the backward of the robot, are more effective than the others. In future works this could lead to try out different solutions that limit the robot to moving forward. Despite these limitations and considering this is the first application of a module of instincts and reflexes with Online Adaptation we believe these results to be enough to show which advantages the integration of these innate safety mechanisms can bring.

Conclusion

In this thesis we explored the possibility of equipping robots with innate safety mechanisms inspired by instincts and reflexes. In nature, these mechanisms play a fundamental role in survival by enabling organisms to avoid potentially dangerous situations or to react rapidly to stimuli. Starting from this observation, we investigated whether similar concepts could be applied to robotic systems, with particular attention to adaptive robots operating in dynamic and partially unknown environments.

In the first part of the work we focused on the analysis of instincts and reflexes to examine their characteristics and to highlight the importance of these innate mechanisms in nature. Then, we reviewed the literature for adaptive robotics to understand which properties are fundamental to build autonomous agents focusing in particular on solutions from Evolutionary Robotics and Online Adaptation. We observed that the ability to adapt to changes and to deal with damages is extremely important and many approaches take inspiration from animals and nature to build adaptive and resilient systems. In this context, especially during adaptation, robots are more vulnerable and could benefit from safety measures for damage prevention.

Taking these considerations into account, we proposed a set of instinct and reflex inspired safety mechanisms specifically designed for a wheeled robot. The mechanisms are meant to deal with different sources of potential damage such as falling from edges, high-speed collisions, motor overheating, extreme temperatures and excessive tilts. To facilitate their integration with existing robot controllers, we proposed a modular architecture based on motor schemas, enabling each mechanism to contribute independently to the final behavior of the robot. This allowed to equip the safety mechanisms to the robot controlled by an Online Adaptation process in an almost straightforward way.

The proposed approach was implemented and tested on a Thymio robot operating in the Webots simulator to evaluate its usefulness in situations where the robot continuously explores new behaviors. The experiments showed that the proposed mechanisms can effectively reduce exposure to several dangerous situations and can contribute to improve the robot's longevity. This is extremely useful during those phases in which the robot is particularly vulnerable, such as the early stages of adaptation or exploratory epochs. Among the implemented mechanisms, Avoid Falling and Avoid Fast Crashes proved to be the most effective. This is due to the availability of dedicated sensors positioned on both sides of the robot, which provide sufficient information to determine the location of the danger and generate an appropriate reaction. The other mechanisms also demonstrated their potential usefulness, although their effectiveness was limited by the available sensing capabilities and by the difficulty of accurately estimating the robot's orientation with respect to the source of danger.

In conclusion, this work represents a first step toward the integration of instinct and reflex inspired safety mechanisms within adaptive robotic systems. Although preliminary, the results suggest that this integration can increase the resilience and autonomy of robots that operate in unpredictable environments.

Bibliography

[1] W.M. Wheeler. On instincts. *The Journal of Abnormal Psychology*, 15: 295–318, 1920-1921.

[2] Genevieve Slomski. Animal reflexes. 2022. URL `https://www.ebsco.com/research-starters/anatomy-and-physiology/animal-reflexes`. Accessed: 17 April 2026.

[3] Wikipedia contributors. Instinct — Wikipedia, the free encyclopedia, 2026. URL `https://en.wikipedia.org/w/index.php?title=Instinct&oldid=1352338241`. [Online; accessed 22-April-2026].

[4] I.P. Pavlov. Conditioned reflexes: an investigation of the physiological activity of the cerebral cortex. In *Oxford Univ. Press.*, 1927.

[5] George Windholz. Pavlov's conceptualization of unconditional reflexes, or instincts, within the framework of the theory of higher nervous activity. *The Pavlovian Journal of Biological Science*, 22:123–131, 1987.

[6] Barbara L. Finlay Lisa Feldman Barrett. Concepts, goals and the control of survival-related behaviors. *Current Opinion in Behavioral Sciences*, 24:172–79, 2018.

[7] Konrad Z. Lorenz. The companion in the bird's world. *The Auk*, 54 (3):245–273, 07 1937. ISSN 1938-4254. doi: 10.2307/4078077. URL `https://doi.org/10.2307/4078077`.

[8] D. Mobbs, C.C. Hagan, T. Dalgleish, B. Silston, C. Prévost. The ecology of human fear: survival optimization and the nervous system. *Front. Neurosci*, 9, March 2015. doi: 10.3389/fnins.2015.00055.

[9] Q. Wu, Y. Zhang. Neural circuit mechanisms involved in animals' detection of and response to visual threats. neurosci bull. *Neurosci Bull*, 39:994–1008, June 2023. doi: 10.1007/s12264-023-01021-0.

[10] R. Vale, DA. Evans, T. Branco. Rapid spatial learning controls instinctive defensive behavior in mice. *Curr Biol*, 27:1342–1349, May 2017. doi: 10.1016/j.cub.2017.03.031.

[11] Kyoo-Man Ha. Animals as valuable instinctive and 'learned' beings in the field of disaster management: a comparative perspective. *Natural Hazards*, 75, December 2015. doi: 10.1007/s11069-014-1359-1.

[12] P Harjpal, RK Kovela, MI Qureshi. Promoting survival and primitive reflexes to prevent brain imbalance in premature infants: A scoping review of new insights by physiotherapists on developmental disorders. *Cureus*, August 2023. doi: 10.7759/cureus.43757.

[13] Valentino Braitenberg. *Vehicles: Experiments in Synthetic Psychology.* MIT Press, Cambridge, MA, 1986. ISBN 9780262521123.

[14] Alex Andrew. Understanding intelligence, by rolf pfeifer and christian scheier, mit press, cambridge, mass., 1999, isbn 0-262-16181-8. *Robotica*, 18:687–689, 11 2000. doi: 10.1017/S0263574700222820.

[15] A.E. Eiben. Real-world robot evolution: Why would it (not) work? *Frontiers in Robotics and AI*, Volume 8 - 2021, 2021. ISSN 2296-9144. doi: 10.3389/frobt.2021.696452. URL `https://www.frontiersin.org/journals/robotics-and-ai/articles/10.3389/frobt.2021.696452`.

[16] Hod Lipson and Jordan Pollack. Automatic design and manufacture of robotic lifeforms. *Nature*, 406:974–8, 09 2000. doi: 10.1038/35023115.

[17] D. Floreano, F. Mondada, D. Cliff, P. Husbands, J. A. Meyer, and S. Wilson. Automatic creation of an autonomous agent: Genetic evolution of a neural network driven robot. january 1994.

[18] P. Baldini, M. Braccini, A. Roli. Fault recovery through online adaptation of boolean network robots. *Sensors*, 25(18), 2025. ISSN 1424-8220. doi: 10.3390/s25185849. URL https://www.mdpi.com/1424-8220/25/18/5849.

[19] Agoston E Eiben and Jim Smith. From evolutionary computation to the evolution of things. *Nature*, 521(7553):476–482, 2015. doi: 10.1038/nature14544.

[20] John R. Koza, Martin A. Keane, and Matthew J. Streeter. What's AI Done for Me Lately? Genetic Programming's Human-Competitive Results . *IEEE Intelligent Systems*, 18(03):25–31, may 2003. ISSN 1941-1294. doi: 10.1109/MIS.2003.1200724. URL https://doi.ieeecomputersociety.org/10.1109/MIS.2003.1200724.

[21] Josh C. Bongard. Evolutionary robotics. *Commun. ACM*, 56(8):74–83, August 2013. ISSN 0001-0782. doi: 10.1145/2493883. URL https://doi.org/10.1145/2493883.

[22] Stephane Doncieux, Nicolas Bredeche, Jean-Baptiste Mouret, and Agoston E. (Gusz) Eiben. Evolutionary robotics: What, why, and where to. *Frontiers in Robotics and AI*, Volume 2 - 2015, 2015. ISSN 2296-9144. doi: 10.3389/frobt.2015.00004. URL https://www.frontiersin.org/journals/robotics-and-ai/articles/10.3389/frobt.2015.00004.

[23] D. Floreano and L. Keller. Evolution of adaptive behaviour in robots by means of darwinian selection. january 2010. URL https://doi.org/10.1371/journal.pbio.1000292. Accessed: 27 March 2026.

[24] Dario Floreano, Phil Husbands, and Stefano Nolfi. *Evolutionary*

Robotics, pages 1423–1451. 01 2008. ISBN 978-3-540-23957-4. doi: 10.1007/978-3-540-30301-5_62.

[25] Nick Jakobi, Phil Husbands, and Inman Harvey. Noise and the reality gap: The use of simulation in evolutionary robotics. volume 929, pages 704–720, 01 1995.

[26] P. Baldini, A. Roli, and M. Braccini. On the performance of online adaptation of robots controlled by nanowire networks. *IEEE Access*, 11: 144408–144420, 2023. doi: 10.1109/ACCESS.2023.3345224.

[27] Michele Braccini, Andrea Roli, and Stuart A. Kauffman. Online adaptation in robots as biological development provides phenotypic plasticity, 2020. URL https://arxiv.org/abs/2006.02367.

[28] Michele Braccini, Andrea Roli, Edoardo Barbieri, and Stuart A. Kauffman. On the criticality of adaptive boolean network robots. *Entropy*, 24(10), 2022. ISSN 1099-4300. doi: 10.3390/e24101368. URL https://www.mdpi.com/1099-4300/24/10/1368.

[29] Paolo Baldini. Online adaptation of robots controlled by nanowire networks. Master's thesis, Università di Bologna, 2022. URL https://amslaurea.unibo.it/id/eprint/25396/.

[30] Antoine Cully, Jeff Clune, and Jean-Baptiste Mouret. Robots that can adapt like natural animals. *CoRR*, abs/1407.3501, 2014. URL http://arxiv.org/abs/1407.3501.

[31] M. Braccini, P.Baldini, A. Roli. An investigation of graceful degradation in boolean network robots subject to online adaptation. In *Artificial Life and Evolutionary Computation*, pages 202–213, Cham, 2024. Springer Nature Switzerland. ISBN 978-3-031-57430-6. doi: 10.1007/978-3-031-57430-6_16.

[32] S. Mahdavi, P. Bentley. An evolutionary approach to damage recovery of robot motion with muscles. pages 248–255, 09 2003. ISBN 978-3-540-20057-4. doi: 10.1007/978-3-540-39432-7_27.

[33] J. Bongard, V. Zykov, H Lipson . Resilient machines through continuous self-modeling. *Science*, 314(5802):1118–1121, 2006. doi: 10.1126/science.1133687. URL `https://www.science.org/doi/abs/10.1126/science.1133687`.

[34] P. Husbands, Y. Shim, M. Garvie, A. Dewar, N. Domcsek, P. Graham, J. Knight, T. Nowotny, and A. Philippides. Recent advances in evolutionary and bio-inspired adaptive robotics: Exploiting embodied dynamics. *Applied Intelligence*, 51(9):6467–6496, 2021. doi: 10.1007/s10489-021-02275-9.

[35] Siavash Mahdavi and Peter Bentley. Innately adaptive robotics through embodied evolution. *Auton. Robots*, 20:149–163, 03 2006. doi: 10.1007/s10514-006-5941-6.

[36] Society of Robots. Sensors - accelerometer. `https://www.societyofrobots.com/sensors_accelerometer.shtml`, 2014. Accessed: 2026-06-04.

[37] Cyberbotics Ltd. Webots User Guide. `https://www.cyberbotics.com/doc/guide/thymio2?version=R2021a`, 2026. Accessed: 2026-05-26.

[38] Cyberbotics Ltd. Simulating your robots with Webots. `https://cyberbotics.com/`, 2026. Accessed: 2026-05-26.

[39] Mobsya. Thymio by Mobsya. `https://www.thymio.org/`, 2026. Accessed: 2026-05-26.

[40] Mobsya. Programming with Thymio Suite. `https://www.thymio.org/products/programming-with-thymio-suite/`, 2026. Accessed: 2026-05-26.

[41] Ronald C. Arkin. *Behavior-Based Robotics*. The MIT Press, 1998.

[42] Andrea Roli. Behaviour-based robotics – part 2. Intelligent Robotic System Course Slides.

[43] Michael Wrona. Roll and pitch angles from accelerometer sensors. `https://mwrona.com/posts/accel-roll-pitch/`, November 2020. Accessed: 2026-06-04.

[44] Andrew L. Nelson, Gregory J. Barlow, and Lefteris Doitsidis. Fitness functions in evolutionary robotics: A survey and analysis. *Robotics and Autonomous Systems*, 57(4):345–370, 2009. ISSN 0921-8890. doi: https://doi.org/10.1016/j.robot.2008.09.009. URL `https://www.sciencedirect.com/science/article/pii/S0921889008001450`.

Ringraziamenti

La stesura di questa tesi non sarebbe stata possibile senza la guida, il supporto e anche l'incoraggiamento dei miei relatori. Ringrazio quindi il Prof. Roli, Paolo Baldini e Michele Braccini.

Durante il mio percorso universitario ho avuto la fortuna di fare tante amicizie che si sono rivelate importanti tanto dentro all'università quanto fuori. Voglio ringraziare di cuore la mia amica Elisa che conosco dal primo giorno di università e con cui ho affrontato la preparazione di molti esami e progetti universitari. Non avrei saputo come fare senza di te. Ringrazio anche i miei cari amici Malloc, Rubba, Gaia e Pippo per avermi sempre ascoltata e per i consigli che mi hanno dato.

Ovviamente, anche il sostegno delle persone a me care al di fuori del contesto universitario è stato fondamentale per arrivare alla fine di questo percorso. Un grazie speciale alle mie amiche Bianca e Vittoria, su cui ho sempre potuto contare negli anni, sia per confidarmi sia per trovare un po' di svago nella vita di tutti i giorni. Ringrazio Fodda per avermi spronato con la sua caratteristica schiettezza. Un grazie anche ai miei amici Cenno, Farolf e Bra per avermi aiutata ad affrontare tutto con più leggerezza.

Infine, ringrazio i miei genitori e mio fratello per essermi stati sempre vicini e non avermi mai fatto mancare nulla.