Recipes, Beyond Computational Procedures

Gianmarco Tuccini §
gianmarco.tuccini@stud.unifi.it
Laura Corti ‡
l.corti@unicampus.it
Luca Baronti †
l.baronti@bham.ac.uk
Roberta Lanfredini §
roberta.lanfredini@unifi.it

ABSTRACT

Nowadays autonomous robots are pervasive in the manufacturing industry and are increasingly common in the domestic setting, particularly in the kitchen. Even though the kitchen robots have proven themselves useful, they have also shown inherent limitations. In this contribution we contend that these limitations arise from essential differences between computational procedures (i.e. programs), as originally described by Turing in his seminal 1936 paper, and recipes. Computational procedures formalise the actions of a person which computes with pencil and paper and thus concern themselves only with symbols. Recipes describe the actions needed to prepare dishes and thus are essentially coupled to the environment in which they are followed, the running time, the functional role of the hand; and the constitutive role of the body in the act of cooking. Unlike a program, a recipe is not a sufficient description to produce the expected result regardless of environment, body and time. Whilst in an industrial setting it is possible to thoroughly account for the differences between computing and cooking and thus formalise recipes as programs, in the domestic setting it proves impossible. Everyday recipes go too much beyond computational procedures because the "architecture" of a human cook, which determines in which way he perceives and acquires experience while following recipes, is shaped by his belonging to the world and not by the needs of a particular task.

1. Introduction

With the advent of Unimate (Islam & Rahman, 2013) designed and developed by George Devol and Joe Engelberger in the late 50's, the manufacturing

‡ Università Campus-Biomedico di Roma, Italy.
† University of Birmingham, UK.
§ Università Degli Studi di Firenze, Italy.

industry was set to a path of a gradual automation of repetitive or dangerous tasks. The automotive industry was one of the primary sectors where automatic mechanical units were employed. The needs of the sector demanded units that were reliable enough to carry over a specific, repetitive, task with high accuracy. At the same time, the units had to be general enough to execute a wide variety of different tasks. More advanced *robots*, like the Programmable Universal Machine for Assembly (PUMA) (Wise, 2005), were designed to perform progressively more complex activities. With more autonomous robots involved in the productive process, the human responsibilities slowly shifted from an operative to a supervisory role (e.g. constant monitoring and intervening).

Nowadays robots are so pervasive in the manufacturing industry that in many cases a constant human supervision is replaced in favour of a manager-by-exception model (Thurman & Mitchell, 1995) where human operators are not even physically present in the control room. Whilst the industrial sector is slowly transitioning into the *lights-out* (Jaikumar, 1986) model, where a factory is entirely run by robots able to autonomously carry over the whole process productive from the raw materials to the finite product, the automation is also becoming more present in the domestic contexts.

People have shown (Ray et al., 2008) a very positive attitude towards the adoption of robotic solutions able to tackle repetitive or menial domestic tasks. However, in contrast with the industrial environment, designing robots for domestic contexts pose additional difficulties. In some cases these challenges are represented by more variegated and uncertain environments to navigate whilst in others the uncertainty is inherent in the task itself. Kitchen robots designed to operate autonomously in a kitchen environment are perhaps the most interesting examples of robots that work under uncertain environments on a high qualitative task.

One of the most advanced representatives of the kitchen robots is The Moley Robotic Kitchen (*Moley*, 2020), composed by two-handed robot arms able to manipulate most of the commonly used kitchen utensils. Using video-recorded human chefs as reference, Moley is able to prepare a variety of different recipes simply replicating with its manipulators the actions performed by the human "trainer". Thus, Moley is able to prepare the trainer's recipe as long as the environment (e.g. the position and orientation of the required utensils) is as close as possible to the one used by the human trainer. The problems inherent in dealing with unforeseeable circumstances make apparently trivial tasks, like cutting an ingredient, difficult (Mu et al., 2019) to be executed automatically.

In order to better understand these limitations it is useful to remember that, in the most general case, robots are general-purposes computers programmed to interact with physical world objects, in function of the signal transmitted by their sensors. Ultimately, any robot is a general-purpose computer required to act in a physical world environment. On the one hand, this design allows the programming of the robot with a finite set of precise instructions, with clear advantages in terms of control and precision, especially useful in the context of performing repetitive tasks accurately. On the other hand, robot potential is bound by the computational limits of the computer. Therefore the problems the modern robots design is facing in dealing with *physical procedures* can be, at least partially, backtracked to the nature of the *effective procedures* computers were originally designed to solve.

The remainder of this paper we'll be structured as follows. First, we'll take a look at the theoretical foundation of modern computers, the formal system known as Turing Machine (Turing, 1936), in order to shed light on the properties of computations performed by modern computers guided by programs. We'll note that the original Turing machine was meant to model the activity of a person who computes with pencil and paper, a computor. We'll then trace back the properties of computations to those of the activity performed by a computor. Finally, we'll define the formal representation of computations, programs, as sequences of instructions which a computor may follow. We'll insist that the properties of programs derive from those of computation, which in turn rest on those of the activity performed by a computor.

Second, we'll take a look at the production of food in an industrial setting. We'll show that, whilst robots guided by recipes formalised as programs do produce food everyday, they may not be accurately modelled through the Turing machine formal system. In order to reliably produce food the robots need sensors, which must be explicitly modelled as oracles in their programs. Those oracles are a significant extension to the original Turing machine formal system. We'll argue that need for this extension stems from the difference between the activity of computing, which – even when performed by a human – operates on symbols to produce symbols and the activity of cooking, which operates on ingredients to produce dishes: the former activity is, in principle, independent from its context; the latter is not (e.g. because it depends on the passage of time). We'll also argue that this extension is enough to bridge the difference between computing and cooking only because the industrial setting is a controlled one. In an uncontrolled environment too many sensors would be needed and it wouldn't be feasible to model them as oracles. Without all the required oracles, the production of food wouldn't be assimilable to computing and thus recipes wouldn't be formalisable as programs.

Third, we'll look at cooking outside the industrial setting. We'll note that human cooks are able to follow recipes even though they are not presented in a formal fashion. We'll contrast human cooks with robots such as Moley, which are relatively autonomous but requires formalised recipes, and with food-processors, tools devoid of any autonomy which may be used to execute recipes but which do not follow recipes by themselves. On the basis of our analysis, we'll argue that the limitations of robots – and a fortiori food processors – with respect to humans stem from the relationship between robots and the environment, the functional role of the hand in the act of cooking and the relationship between human beings and robots. The "architecture" of a human cook, which determines in which way he perceives and acquires experience while following recipes, is shaped by his belonging to this world and not by the needs of a particular task. This architecture makes him more suited than a robot to cope with an uncontrolled environment such as that of the kitchen in order to reliably produce food. The human hand is also shaped by the relationship with this world and is endowed with a wide range of skills, such as feeling. Moreover, humans have a consciousness of the limits of their hands – and more generally of their body – which is both innate and acquired through experiences. The properties of the human hand and the consciousness of the limits are decisive to effectively cook outside an industrial setting, yet robots lack them and thus are not modelled in their programs. Finally, the relationship between human and kitchen robots confirms ever more the importance of the human body for the activity of cooking.

Finally, we'll conclude that – generally speaking – recipes go beyond computational procedures because the activity of cooking goes beyond the activity of computing. Outside of an industrial setting human-like abilities are needed to effectively follow recipes.

2. The Turing Machine

Modern general-purpose computers are physical implementations of the formal model known as Turing Machine (TM) (Turing, 1936). This formal model, equivalent to the λ-calculus devised by Church (Church, 1936), was proposed as a solution to the *Entscheidungsproblem* posed (Ackermann & Hilbert, 1928) by Hilbert. This *decision problem* questioned the existence of an effective procedure that was able to determine if any formula of the functional calculus is provable. The key aspect of this problem was how to interpret the terms "effective procedure". Turing's interpretation of an effective procedure was an algorithm computable by what he (Turing, 1948, p. 7) called *logical computing machine* (LCM), later known as Turing Machine. A Turing Machine was devised as an ideal machine composed by a *state machine* and an *head* able to read and write symbols present on a *tape* freely movable forward and backward. A finite set S defines all the possible symbols present on each location, or *cell*, of the tape. The tape is assumed to be either infinite or long "as required". At each step, the machine is instructed to carry out a specific action, defined from a limited set (e.g. read or write a symbol) based on the currently read symbol and the current internal state.

Any physical implementation of a TM poses limits to its original design, dampening its universality capabilities as a result. Even a fundamentally symbolic task, like computing a function of two numbers, can yield inaccurate results due to the physical nature of the computer and its limits (e.g. finite precision). In modern general purposes computers numerical errors are the result of the mismatch between the formal and physical world. That said, when the wanted result of a procedure is a symbolic answer, a physical computer based on the idea of a TM can still be a tool apt for the purpose. In this context, problems inherent to the physical implementation of the computer can be predicted or limited and their impact on the final result can generally be marginalised.

Not every limitation encountered by physical implementations of a TM, though, is of this kind. Some limitations are inherent to Turing's definition and may be ultimately traced back to the "reference implementation" of a Turing machine, which predates (or rather has motivated) the formal system itself. Infact, whilst the Turing machine is above all a formal system devised to solve the *Entscheidungsproblem* problem, it is presented by Turing himself as the model for a *person* who computes with pen and paper: a computor. The Turing machine owes its immediate and widespread success to the appeal of this very interpretation, which manages to root the formal properties and limitations of computation in the practicalities of the everyday activity of computing, which everyone knows well.

To better appreciate the *inherent* properties and limitations of TM, it is useful to revisit the analysis of human computation carried out by Turing in it's seminal paper (Turing, 1936).

3. A definition of computation

A computation is performed by a *computor*, who observes, writes and erases symbols on sheets of squared paper[1].

All the elements of a computation are finite. The computor uses a finite alphabet of symbols: even though it may be large, it always has an upper bound. His attention is finite, so at every moment he may observe only a limited number of symbols written in a limited number of squares. The possible "states of mind" which he can have are finite too. Lasty, at every moment he may have only one state of mind.

A computation progresses through local, progressive changes. The computor may write or erase symbols on the sheets of paper but only in the squares he is observing. Moreover, he may shift his focus on different squares but they have to be near to some of the squares he is already observing.

In every moment a computation is determined. The computor's behaviour is entirely determined by the symbols written in the squares he is observing and by his current state of mind. In turn, the status of the computation is entirely determined by the computor's behavior: nothing changes if he doesn't act.

Finally, a computation is not sensitive to time. If the computor wish, he can pause the computation anytime. He just has to write down the status of the computation and all the instructions needed to continue it later.

4. A definition of program

Together, these characteristics make it always possible to provide a formal description of a computation in the form of a finite sequence of instructions. Each instruction is conditional to a combination of states of mind and observed symbols of the computor, lists the sequence of observations, erasures and writes which the computor has to perform and indicates which states of mind the computor will have after the performance of the prescribed actions.

These sequences of instructions are programs.

The properties inherent to computation determines some notable properties of programs.

First of all, the intended result of a program is predetermined by the initial state of mind of the computor, the symbols he initially observes and the program itself. The initial state of mind and symbols, in fact, univocally determine which is the first instruction the computor has to follow, which in turns determines the

[1] From now on, we'll follow Sieg (1994) in describing Turing's analysis of human computation.

changes to the status of the computation and the next instruction he has to follow, and so on.

Secondly, the result of a program is independent from the context in which it is executed. Provided the computer is able to observe, erase and write symbols, he may produce the intended result everywhere and everytime, because the status of the computation changes only in consequence to his actions.

Thirdly, the result of a program is also independent from the time employed by the computor to execute it. If the computor stops following the prescribed instructions, the status of the computation also stops changing until he resumes the execution of the program.

Finally, the computor observes, writes and erase symbols. It doesn't matter how the symbols are represented: the computer may choose to "write" a symbol by putting some stones on a square of his sheet of paper or he may even systematically swap a symbol with another. Provided he is able to distinguish the different symbols, he is still able to produce the intended result.

5. Cooking in an industrial setting

It is worth asking whether these properties and limitations hold for an activity which, prima facie, is very different from computing: cooking. Is the production of food similar enough to computation that a Turing machine is a good model for it? By the same token, are the formal descriptions of the cooking procedures (i.e. the recipes) similar enough to programs that a Turing machine may execute them?

For a start, it must be noted that it is difficult to provide an univocal definition of recipes. According to the Oxford English Dictionary, a recipe is simply "a statement of the ingredients and procedure required for making something, (now) esp. a dish in cookery". Yet, recipes are used both in an industrial setting, to produce thousands of cakes in a day, and to prepare a simple meal. It is not at all obvious that the recipes used in these two different contexts are of the same kind.

We may look at the following simplified description of the industrial recipe for a Jaffa cake, intended as an aid for the product developers (Manley, 2001, p. 124).

Ingredient	Quantity
flour, weak cornflour	100.00

caster sugar	86.59
glucose syrup 80%	6.95
oil	2.57
fresh egg	69.52
amm. bic.	0.64
soda	0.50
ACP	0.50
glycerine	3.09
colour	0.10
added water	2

Critical ingredients

The quality of the egg is important and it is usual to use either freshly shelled whole eggs or carefully thawed frozen fresh eggs. The egg entrains the air and the batter is then pumped to a depositor. ... The syrups and glycerine are used as humectants to prevent the baked product from drying too much and to maintain a softer eating texture.

Mixing

This is usually done in two stages. Firstly, all the ingredients are blended together as a batch operation. This is followed by vigorous beating when air is incorporated to give a lower density. This latter stage is usually achieved as a continuous operation by passing the blended batter through a very high shear mixer inside a water cooled barrel under pressure (for example, an Oakes mixer). Air is injected into the mixer at a given rate and pressure to give a batter density of about 0.88 g/cc at around 19 °C. A back pressure valve at the exit of the mixer barrel gives better control of the pressure during mixing.

Even this simplified description shows that industrial recipes are by their nature precise, in order to allow the production of many identical dishes with no waste of time and resources. Recipes of this kind may be formalised and *are* regularly formalised as computer programs which industrial robots then execute, e.g. to produce Jaffa cakes.

It's worth noting, though, that to reliably execute these recipes industrial robots rely on their sensors, for which the original Turing machine formal system doesn't account.

Sensors are needed because of intrinsic differences between programs and recipes. For instance:

1. Unlike a program, a recipe is not a sufficient description to produce the expected result regardless of the type of ingredients. If the robot would systematically invert glycerine and pasta, keeping the ability to distinguish between the two types of ingredients unchanged, he would get different results.
2. Unlike a program, a recipe is non a sufficient description to produce the expected result regardless of the environment. The recipe instructs the robot to inject air in the mixer to give the batter a certain density at 19 ° C: the robot can follow the instruction correctly and still not manage to produce the intended result, e.g. because it not consider atmospheric pressure.
3. Unlike a program, a recipe is sufficient to produce the expected result only if it is executed in a certain time interval. The recipe instructs the robot to blend the ingredients and then to incorporate air into them. If the robot would suspend the execution of the recipe from 30 minute after the blending, it would be impossible for it to correctly incorporate the air because the ingredients would already be partially separated.

Industrial robots are thus better described through an extension to the Turing machine model, the oracle machine.

A *Turing oracle machine (o-machine)* is a Turing machine with an extra "read-only" tape, called the "oracle tape", upon which is written the characteristic function of some set A (called the *oracle*), and whose symbols cannot be printed over. The old tape is called the *work tape* and operates just as before. The reading head moves along both tapes simultaneously (Soare, 2009, p. 377).

Through sensors the robot is able to keep track of the physical parameters involved in food production, which are essential in order to produce the desired products. Through oracles, which model sensors, the recipes may be accurately formalised as sequences of instructions. These programs, though, are different from those of the Turing machine in that they are essentially coupled to the environment in which they are executed. We contend that this coupling is motivated from the differences between computing and cooking: whilst computing, either with pencil and paper or through a modern computer, is concerned only with symbols, cooking is concerned with the processing of concrete ingredients. Moreover, the formalisation of recipes through programs still requires that a sufficient number of sensors is available to control all physical parameters relevant for the computation: otherwise the intended result wouldn't be reliably produced. This is a reasonable requirement in an industrial setting but maybe not in a domestic setting.

6. Cooking in a domestic setting

We will now consider everyday recipes.

An everyday recipe, taken from BBC website and meant for students, has the following format.

- 2 tbsp olive oil
- 5 rashers smoked streaky bacon, roughly chopped
- 500g/1lb 2oz beef mince
- 1 onion, finely chopped
- ...

1. Heat 1 tablespoon of olive oil in a large pan over a medium heat. Add the bacon and cook for 3–4 minutes, until beginning to crisp. Remove from the pan with a slotted spoon and set aside on a plate.
2. Add the mince to the pan and cook over a high heat until well browned. Remove from the pan with a slotted spoon and set aside.
3. Heat the remaining 1 tablespoon of oil in the pan. Add the onion and cook for 3–4 minutes, until beginning to soften. Add the celery and carrots cook for 5–8 minutes, then season with salt and pepper.
4. ...

This recipe is meant to be easily followed by about any person in about any kitchen. Any person may decode the rough instructions and follow them effectively using its skill, both innate and learnt through experience (e.g. to perceive when something is "browned"), with no need for precise instruments (compare: "Air is injected into the mixer at a given rate and pressure to give a batter density of about 0.88 g/cc at around 19 °C.") or a strictly controlled environment. Whilst it is surely possible to build an industrial robot in order to execute this particular recipe, precisely formalised, in a controlled environment, is it possible to build a robot capable of executing also all the other recipes in the BBC website regardless of the environment? From a different point of view, is it possible to thoroughly describe any instance of food preparation through a formalised sequence of instructions just as it happens with instances of computations, so that a single universal robot needs only to know the sequence of instructions in order to execute them?

7. Robotics: from industries to home

Robotics is often described as the technology of the future, and it is frequently compared to computers, as in the article 'A robot in Every Home' by Bill Gates published in January 2008 in the magazine Scientific America or in the article 'the robots are coming' written by Elizabeth Corcoran for the magazine Forbes in 2006. Bill Gates, the co-founder of Microsoft, the world's largest software company, predicts that the next revolution will be robotics. A quotation from this article is fundamental in order to comprehend the urgency to think about robotics in a trans-disciplinary approach.

The challenges facing the robotics industry are similar to those we tackled in computing three decades ago. Robotics companies have no standard operating software that could allow popular application programs to run in a variety of devices. The standardisation of robotic processors and other hardware is limited. Whenever somebody wants to build a new robot, they usually have to start from square one. (...) And as I look at the trends that are now starting to converge, I can envision a future in which robotic devices will become a nearly ubiquitous part of our day-to-day lives. (Gates, 2007)

Gates' forecasting was met by seeing the data in the Executive Summary World Robotics 2019 by International Federation of Robotics. The industrial sector shows that in 2018, global robot installations increased by 6% (422,271 units) (Robotics, 2019) . The large part of installation remains in the automotive industry, followed by electrical/electronics, metal and machinery, plastics and chemical products, and food and beverages. On the other hand, in the sector of service robotics, the total number of machines sold in 2018 rose by 61% (more than 271,000 units), up from roughly 168,000 in 2017. Autonomous guided vehicles (AGVs) represent the largest fraction 41% of all units sold; the second largest category (39%) is inspection and maintenance robots. Service robots for defence represent 5% of the total number. (Robotics, 2019) Therefore, we can envision a future in which robots becomes a significant part of our life in a wide range of tasks, from assisting people in the domestic context to cooperate in the public sphere (Arras & Cerqui, 2005) (Baillie & Benyon, 2008). Particularly, the introduction of robotics in the domestic setting imposes a reflection on home as "a complex environment, designed for general use but shaped by individual needs and desires." (Baillie & Benyon, 2008)

8. Robots in the kitchen domain

According to D. Schneiderman (Schneiderman, 2010), the kitchen is historically the focal point of the house, where family members spend most of their time. The design of the kitchen must be based on efficiency and flexibility. If we are looking to the present condition in the kitchen domain, kitchen robots, or in French robot de cuisine, have long been a welcome and necessary presence in everyday life. The most common examples help the stakeholders primarily to chop and mix ingredients; the more sophisticated robots can cook an essential meal, as, for example, boiling pasta or roast meat. Are they so incredible and sophisticated machines to describe them as robots? This is a very fundamental issue concerning the definition of these tools. According to Paul Dumouchel and Luisa Damiano (Dumouchel, 2017), the use of the term 'cooking robot' does not fit with the original definition of robot because of its lack of autonomy. They prefer to use the term 'food processor' because of the automatic way in which kitchen robots work.

This paradigm is drastically changed when the Britain company, Moley, announced in 2015 that it would make the first robotic kitchen.(*Moley*, 2020) On April 2015, CNN titled "Robo chef: Would you trust a cook with no taste buds?" and the daily telegraph "Robotic hands cook any dish with the skill of master chef...then clean up". On the website, the company has recently announced that the robotics kitchen will be on the marketplace in 2020 and describe it as "hands with multiple joints, numerous actuated degrees of freedom, tactile sensors and sophisticated control systems. This is what allows MK1 to download a recipe and reproduce it exactly as the MasterChef would have cooked it, wherever you are in the world." Therefore, it is a close unit, equipped with two robotics arms with hands, an oven, an electric stove, and a dishwasher. The essential elements are the arms that can handle not only the kitchen equipment, like knife, plates, or spoon but also the ingredients. The human ability that the machine shows is based on a database of chef's actions, captured by cameras and sensors and translated digitally using gesture recognition. The user chooses a dish on the smartphone or tablet, and the robotics kitchen prepares it autonomously. The robot has a set of ingredients laid on the workspace and uses them to prepare a dish following a list of instructions. Unlike the automatic food processors that we have currently in the kitchen, Moley could be defined properly as a robot. Following Angelo's definition, "robot is a smart machine that does routine, repetitive, hazardous mechanical tasks, or performs other operations either under direct human command and control or on its own, using a computer with embedded software (which contains 12 previously loaded commands and instructions) or with an advanced level of machine (artificial) intelligence (which bases decisions and actions on data gathered by the robot about its current environment)." (Angelo, 2007)

9. A philosophical reflection on kitchen robots

A philosophical reflection on the robotics limitation must take into account three central topics: the relationship between robots and the environment, the functional role of the hand in the act of cooking and the relationship between human beings and robots.

The first issue concerns the relation between robotics and the environment. As we mentioned earlier, one of the key distinctions between industrial robotics and service robotics is precisely the characteristic of the first form of working in a structured environment, that is a closed setting in which the human presence is restricted, and a robot could act autonomously in this setting where the object to manipulate is in a well-known position. Instead, the service robots are designed to work and co-operate in the social setting. The case of Moley opens the way to a third mid-term position in which we have a service robot, whose function is to cook, which, however, has a close structure, that recalls an industrial robot. The protective glass that distances the human subject from the moving robotics limbs determines a substantial not only physical distance of the action. This reflection implies a second item, the time, that is, at the same time, a more general issue concerning the possibility to consider a recipe as a mechanical program of execution but also the specific example of robotics. From a general point of view, "Computers do have timing information in their internal clocks, but there is no timing in the Turing machine formalisms (and equivalents) for computers, and the architecture of the timing in a computer is not an evolutionary possibility." (Bickhard, 2009) The problem of time span becomes hard to handle for a robot struggling with the most complex recipes. If, for example, we think about the leavening of pizza, we realise how much it is conditioned by external conditions, such as the quality of flour and yeast, or humidity, or heat. In this case, the human being puts into practice the previous experience in assessing the degree of leavening of the dough and tries touching it. In the same example, a robotic cook, as Moley can be, will have much greater difficulty in evaluating leavening since the experience but also the ability to perceive play an essential role. The same example has a different development in the case of a food processor because, in this case, the tool is not autonomous in the management of time, but the timespan is set by the person who is cooking.

The example of pizza has already highlighted the other points to be dealt with, in particular the problem of personal experience in cooking. For example, imagination is essential when we are foreshadowing a new recipe or the creative copy of a recipe that has ingredients we do not like. Even more, the theme of experience comes into play when we consider the movement of the hand. In the philosophical debate, there are many positions focused on the important role of hand. Anaxagoras, for example, affirmed that the human being is the most intelligent animal because of the hand, and Aristotle argued that 'intelligence is the reason why we have a hand' (Russo, 2017). We should also recall Heidegger's ready-to-hand description, articulated in the third chapter of the first section of the first part of Being and Time, as an 'ontological categorical concept of beings as they are in themselves' (Chillón, 2017). The most relevant function is the active manipulation of objects, that comprehend power and precision grasping; for this reason, the hand has an 'operative opening' (Russo, 2017). Introducing manipulation in robotic systems is one of the biggest challenges today, not only from service robotics but also for medical devices such as upper limb prostheses. (Carrozza et al., 2006) (Raspopovic et al., 2014) (Sorgini et al., 2018) Even if we do not pay much attention to the variety of gestures we use, the analysis of movement shows that there is a wide variety of possible grasping. The power grasping is divided into three main sections, based on the shape of objects, cylindrical, spherical and hook; whilst in the precision grasping, in which it is central the function of thumb opposition, we can find three types of gesture: pinch, tripod and lumbrical. From a technical point of view, the robot is a physical complex system that comprehends sensors, able to sense or perceive features from the environment, a central controller and effectors and actuators, with which a robot can move and act. There are two main types of activities: the locomotion, that belongs to the capability to move around, and the manipulation, or capacity to 'handle objects.' (Matarić et al., 2007) In the case of Moley, the end-effector, which is the final part of the manipulator used to achieve the goal, is the hand, which manipulates the ingredients and uses the kitchen equipment. Even the importance of hand and the variety of possible human movements, when we consider a robotic body, we have to take into account also the limit of its body, for example, the fact that the hand is connected to an arm, that does not have complete freedom of movement. (Matarić et al., 2007)

If we have a consciousness of our limits in the use of the body, at the same time, innate and made by experience, a robot cannot be conscious of its limits; for this reason, it is fundamental studying the kinematics, in which are expressed the rules 'about the structures of the manipulator, describing what is attached to what' (Matarić et al., 2007). The attention is focused on joints, and in an anthropomorphic robot arm, as in the case of Moley, there are three main joints: the shoulder one, the elbow and the wrist joint. In the example mentioned above of pizza, the problem of manipulation is extremely central because pizza needs to be kneaded by the hands. kneading pizza has a long series of implications: first of all, kneading pizza requires certain freedom in the movement of the wrist and a dose of the strength that must be commensurate with the degree of softness of the dough. This would require a high number of degrees of freedom and skill, such as the ability to feel what is touched and to measure strength against this.

The last philosophical issue concerns the relationship between human beings and robots from a post-phenomenological point of view. This philosophical perspective focuses on the types of relationship between humans and technology, that should not be seen as passive ″object″ as opposed to a human subject. It investigates how technology plays a role in human-world relations. (Ihde, 1990) It is possible to find at least four types of technological mediation: 1. the first one is the ″embodiment relationship″ in which the technological device is linked closely with the subject and becomes part of the way we perceive the world, like an eyeglass; 2. the ″hermeneutic relations″, the second paradigm, focuses on the possibility to interpret the world directly through a technological device's display; 3. the third one is the ″alterity relations″ in which technology interacts like a human with us, becoming ″quasi-other″; 4. the last one is the background relation in which technology is around us and functions in the background, like the fridge. (Ihde, 1990)

In the domain of cooking, we can find the coexistence of many types of relationship with technology. The most immediate is undoubtedly the fourth, background relationship, in which some devices work regularly and we do not realise it until when we need it, as in the case of the fridge. On the other hand, the use of food processors shows a different scenario in which technology is used to support human dexterity by optimising time and precision. Therefore, we could define an intermediate way between embodiment and alterity relationship. In the case of Moley, the robot which cooks, the relationship could be definable as alterity because the robot functions in place of human beings.

10. Conclusion

In conclusion, the philosophical analysis points out clearly that the limits of robotics chef are not only linked with technical issues concerning robotics but also with the definition of recipe and cooking adopted. The alterity relation that we consider the best to describe the capability of cooking expressed by Moley could help to identify the difference between a functional perspective on recipe and a creative dimension of cooking, fit for human beings. According to Borghini (Borghini, 2015), one crucial issue is the difference between dish and recipe, which are often used in the same way even if they are distinct concepts. ''A dish is a stuff, a recipe is the idea. More precisely, a dish is a specific concoction of (typically perishable) edible stuff, such as those specific actions that led to this slice of pizza sitting on my kitchen counter. On the other hand, a recipe – in first approximation – comprises the array of repeatable aspects of a dish whose replication would deliver a dish of the same sort.'' (Borghini, 2015) Already this first clarification poses problems in attributing to a machine the possibility of having ideas. To elucidate this point, it is necessary to rethink the concept of recipes. Harari (Harari, 2016) and Bollini (Bollini et al., 2013) use a realistic interpretation of recipe as a list of ingredients and procedures for preparation; we can apply this definition to describe very well the robotics situation in which a robot picks up the ingredients and cook them.(Borghini, 2015) Therefore, it implies the concept of technical reproducibility that becomes technological reproducibility in the age of robotics. On the other hand, a constructivist definition introduces the 'human fiat' as the guiding principle in the process of a recipe. Therefore, "The identity of a recipe, that is, does not rest on specific ingredients or procedures; what matters, rather, is that whoever produces the recipe recognises it as such." (Borghini, 2015) This definition does not apply to robots but underlines equally essential aspects in the kitchen, such as the personal expertise that comes into play and that we have pointed out as a missing element in robotics practice.

To understand what expertise means and why there is a difference between humans and robots, we must refer to the embodiment as a dimension in which we acquire skills and abilities. In humans, we experience the world through the body, which plays an essential role in shaping our life in many ways and from different perspectives. (Corti & Bertolaso, 2020) We can not think of physical activity, like cooking, without a body able to sense, move and manipulate. Kerstin Dautenhahn said "Life and intelligence only develop inside a body"(Dautenhahn, 1998); therefore, in robotics, the body has not the same fundamental dimension and is considered as the simplest way to interact with the world. A robot can perform a task without this experience being additive of his or her skills because it is designed to act efficiently. The expertise is not only about dexterity or precision but concerns a more 'phenomenological' approach to the experience based on the acquisition of ability by doing; we can think about the difficulty of explaining all the individual steps of a specific recipe. Even the introduction of sensors, seen by Turing as oracles, in practice, cannot solve all the problems described because the model is still not a good fit for this activity. Although the sensors are highly specialised and have the function of receiving information from the world, they are not able to perceive the world in the same way as a human being due to technical and formal limits. Therefore, as we have mentioned in the abstract, the limitation of a computational perspective on the recipe is not only practical but also formal, linked to a descriptive inadequacy of the Turing machine with respect to the activity of cooking in an everyday setting.

REFERENCES

Ackermann, W., & Hilbert, D. (1928). *Grundzüge der theoretischen logik*. Springer-Verlag.

Angelo, J. A. (2007). *Robotics: A reference guide to the new technology*. Libraries Unlimited.

Arras, K. O., & Cerqui, D. (2005). Do we want to share our lives and bodies with robots? A 2000 people survey: A 2000-people survey. *Technical Report, 605*.

Baillie, L., & Benyon, D. (2008). Place and technology in the home. *Computer Supported Cooperative Work (CSCW), 17*(2-3), 227–256.

Bickhard, M. H. (2009). The biological foundations of cognitive science. *New Ideas in Psychology, 27*(1), 75–84.

Bollini, M., Tellex, S., Thompson, T., Roy, N., & Rus, D. (2013). Interpreting and executing recipes with a cooking robot. *Experimental Robotics*, 481–495.

Borghini, A. (2015). What is a recipe? *Journal of Agricultural and Environmental Ethics, 28*(4), 719–738.

Carrozza, M. C., Cappiello, G., Micera, S., Edin, B. B., Beccai, L., & Cipriani, C. (2006). Design of a cybernetic hand for perception and action. *Biological Cybernetics*, *95*(6), 629.

Chillón, J. M. (2017). Ready-to-hand in heidegger. Philosophy as an everyday understanding of the world and the question concerning technology. In *The hand* (pp. 115–126). Springer.

Church, A. (1936). An unsolvable problem of elementary number theory. *American Journal of Mathematics*, *58*(2), 345–363.

Corti, L., & Bertolaso, M. (2020). Embodiment from philosophy to life science and back. *Ludus Vitalis*, *27*(52), 137–142.

Dautenhahn, K. (1998). Embodiment and interaction in socially intelligent life-like agents. *International Workshop on Computation for Metaphors, Analogy, and Agents*, 102–141.

Dumouchel, P., & Damiano, L. (2017). *Living with robots*. Harvard University Press.

Gates, B. (2007). A robot in every home. *Scientific American*, *296*(1), 58–65.

Harari, Y. N. (2016). *Homo deus: A brief history of tomorrow*. Random House.

Ihde, D. (1990). *Technology and the lifeworld: From garden to earth*.

Islam, M., & Rahman, M. (2013). Design and fabrication of line follower robot. *Asian Journal of Applied Science and Engineering*, *2*(2), 27–32.

Jaikumar, R. (1986). Postindustrial manufacturing. *Harvard Business Review*, *64*(6), 69–76.

Manley, D. (2001). *Biscuit, cracker and cookie recipes for the food industry*. Elsevier.

Matarić, M. J., Maja, J., Arkin, R. C., & others. (2007). *The robotics primer*. Mit Press.

Moley. (2020). https://www.moley.com/

Mu, X., Xue, Y., & Jia, Y.-B. (2019). Robotic cutting: Mechanics and control of knife motion. *2019 International Conference on Robotics and Automation (Icra)*, 3066–3072.

Raspopovic, S., Capogrosso, M., Petrini, F. M., Bonizzato, M., Rigosa, J., Di Pino, G., Carpaneto, J., Controzzi, M., Boretius, T., Fernandez, E., & others. (2014). Restoring natural sensory feedback in real-time bidirectional hand prostheses. *Science Translational Medicine*, *6*(222), 222ra19–222ra19.

Ray, C., Mondada, F., & Siegwart, R. (2008). What do people expect from robots? *2008 Ieee/Rsj International Conference on Intelligent Robots and Systems*, 3816–3821.

Robotics, I. F. of. (2019). *Executive summary world robotics 2019 industrial robotics*.

Russo, M. T. (2017). The human hand as a microcosm. A philosophical overview on the hand and its role in the processes of perception, action, and cognition. In *The hand* (pp. 99–113). Springer.

Schneiderman, D. (2010). The prefabricated kitchen: Substance and surface. *Home Cultures*, *7*(3), 243–262.

Sieg, W. (1994). Mechanical procedures and mathematical experience. In A. George (Ed.), *Mathematics and mind* (pp. 71–117). Oxford University Press.

Soare, R. I. (2009). Turing oracle machines, online computing, and three displacements in computability theory. *Annals of Pure and Applied Logic*, *160*(3), 368–399.

Sorgini, F., Massari, L., D'Abbraccio, J., Palermo, E., Menciassi, A., Petrovic, P. B., Mazzoni, A., Carrozza, M. C., Newell, F. N., & Oddo, C. M. (2018). Neuromorphic vibrotactile stimulation of fingertips for encoding object stiffness in telepresence sensory substitution and augmentation applications. *Sensors*, *18*(1), 261.

Thurman, D. A., & Mitchell, C. M. (1995). Multi-system management: The next step after supervisory control? *1995 Ieee International Conference on Systems, Man and Cybernetics. Intelligent Systems for the 21st Century*, *5*, 4207–4212.

Turing, A. M. (1936). On computable numbers, with an application to the Entscheidungsproblem. *Proceedings of the London Mathematical Society*, *2*(42), 230–265.

Turing, A. M. (1948). Intelligent machinery. *Edinburgh University Press*, 395.

Wise, E. (2005). *Robotics demystified*. McGraw-Hill,