We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

6,900
Open access books available

185,000
International authors and editors

200M
Downloads

Our authors are among the

154
Countries delivered to

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com



Chapter 4

Robots Liability: A Use Case and a Potential Solution

Alejandro Zornoza, José C. Moreno,
José L. Guzmán, Francisco Rodríguez and
Julián Sánchez-Hermosilla

Additional information is available at the end of the chapter

http://dx.doi.org/10.5772/intechopen.69888

Abstract

In this chapter, a system of distribution of responsibility for damages caused by robots is introduced, and its practical application on the results obtained in the real experiences at the University of Almería (Spain) is analyzed. The attribution of liability for damages produced by autonomous agents usually focuses the theoretical discussion on legal and ethical fields on robotics. The European Parliament adopted the report with recommendations to the Commission on Civil Law Rules on Robotics (2015/2103(INL)) in February 2017. This work includes the master guidelines that the European Commission should take into account to legislate this technology. In its attempt to attribute responsibility for damages caused by robots, the Committee considers that once responsible parties have been identified, their liability level should range, looking the robot's learning capability and the knowledge learned from its owner. This work proposes the use of responsibility setting matrix as a mechanism to distribute liabilities between the robot, the manufacturer, and the owner, depending on the knowledge programmed by the manufacturer and the one acquired by the robot (through its learning ability and the adjustments made by the owner), that would distribute the responsibility for damages among the three agents involved.

Keywords: Fitorobot, Robotic Liability Matrix, robots, liability, responsibility

1. Introduction

One of the main current issues about robotics is its liability regimes, which is especially referred to robots with a high grade of autonomy.

Assuming that the term "responsibility" is a cultural product (it is the result of the interaction between sociability and moral agency; hence, each state has his own liability

INTECH
open science | open minds

© 2017 The Author(s). Licensee InTech. This chapter is distributed under the terms of the Creative Commons Attribution License (http://creativecommons.org/licenses/by/3.0), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited. CC BY

rules), we are going to introduce the technology as objective element for rethinking the responsibility.

The first works of the European Union (EU) on this subject call for new rules which focus on how a machine can be held responsible for its acts or omissions, suggesting the possibility to create a new legal status only for robots.

This approach would have important social, political, legal, and economical implications that we are not going to expose in this work. On the one hand, we consider robots as tools. Without undertaking any ethical and philosophical evaluation, robots are created to do and to make human tasks. If the labor is physical (e.g., to carry weight) or is psychological (e.g., to perform complex mathematical operations), it is not relevant; they are products and will be considered as such. There is a tendency to humanize robots because they have "human behavior." This is a consequence of their artificial intelligence and the type of work they do, but the most important part of that term is "artificial"; this means that they are something manufactured by human being and could be controlled, limited, and, so, equated to any other product.

On the other hand, there are different kinds of robots, so there should be different types of responsibility. Some robots have physical structure, as care robots or social robots, and some robots only exist on a digital form, as trader bots. When it concerns damage, one usually thinks directly of a robot that hurts a person, but observing the different kinds of robots, the damage can be classified as physical (over persons or goods), economic, or spiritual (moral damage). In order to guide the discussion, it should be noticed that we have observed the state of the art to write these lines, so we have employed a real use case.

In this chapter, Fitorobot is introduced as case study. Further on, the existing product liability regimes will be explained and applied to robots. This issue will allow to show the EU's agreed targets and to expose the adequacy or insufficiency of the current law for it. Finally, a genuine proposal to rework the responsibility and the way to apply it will be described.

2. Fitorobot: a use case

Tracked mobile robots (TMR) are increasingly being used on rough off-road terrains for applications such as forestry, mining, agriculture, and army, and in general in many kinds of applications on unpaved terrains. These applications usually require robots to travel across unprepared terrains performing some activity or transporting materials.

In this chapter, we use a research project at the University of Almería (Spain), which is devoted to the development of an TMR to perform different tasks in greenhouses (especially those related with spraying activities), as a use case.

This TMR is called *Fitorobot* (see **Figure 1**); it is a robot with a mass of 756 kg (with the spray tank full), and the dimension is 1.5 m long × 0.7 m wide. It is driven by a powerful 20 HP gasoline engine. It is equipped with several sensors, but in the real tests carried out in this work, only four have been used (right track encoder, left track encoder, radar, and magnetic compass). The real track radius of the test bed is 0.15 m, but the calibrated track radius is 0.10 m. The distance

http://dx.doi.org/10.5772/intechopen.69888



Figure 1. Fitorobot [1].

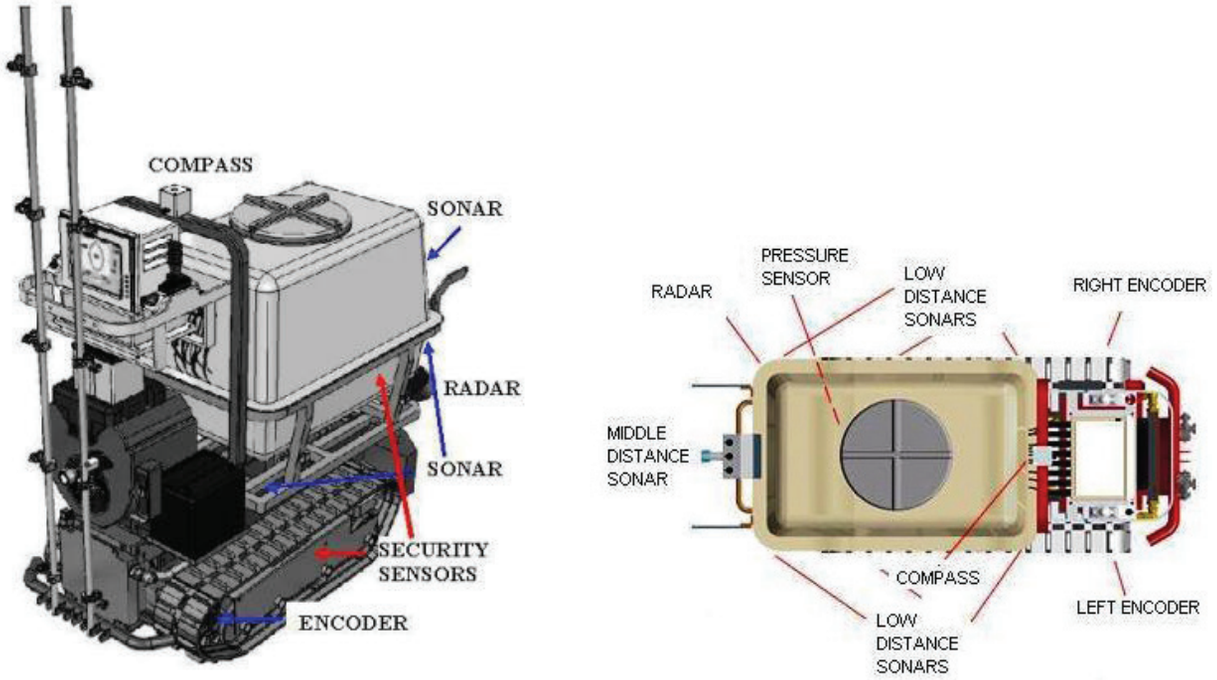
between the track centers is 0.5 m. More details about the features of this TMR can be found in Ref. [1].

Autonomous navigation of the mobile robot relies heavily on external sensor information. Therefore, the performance of the robot navigation will depend heavily on the installed sensors on the platform. Therefore, different types of sensors are installed on the platform, where some of them are redundant for the purpose of testing different configurations and for security reasons.

Table 1 summarizes the sensors installed on the Fitorobot. One middle-range sonar and four short-range sonars have been located in front and in each side of the platform, respectively. These sensors enable the robot to sense the environment and the greenhouse corridors. Odometry establishes the position and velocity of the robot, using two incremental optical encoders attached to the axle of rotation of the track motors. One radar and one magnetic compass measure the linear velocity and orientation of the vehicle, respectively. It is protected from unexpected obstacles in the environment by a security sensor composed of four tactile bars around the vehicle. Finally, a pressure sensor has been installed in the spraying hydraulic system to regulate the spraying controllers [1]. The sensor positions on the platform are shown in **Figure 2**.

Sensor	Mark, model	Range
Pressure	WIKA, ECO-1	0–250 bar
Sonar (middle dist.)	Siemens, Bero III	40–300 cm
Sonar (short dist.)	Siemens, Bero M18	15–100 cm
Magnetic compass	KVH, C100	0–360°
Radar	LH Agro, Compact II	0.08–17.3 m/s
Encoders	Sick, DRS61	0–360°
Security sensor	SafeWork, SKL25-40	On-off
Camera	Logitech 2 Mpixel QuickCam Sphere AF webcam	1600 × 1200

Table 1. Features of the installed sensors.



(a) Lateral view of the sensorial system

(b) Top view of the sensorial system

Figure 2. Sensors on the experimental platform [1].

A more detailed information about the actuator and control system of the Fitorobot can be found in Ref. [1].

3. Degrees of freedom for influencing the robot behavior

The behavior of a robot is given by the set of algorithms that constitute its artificial intelligence. These algorithms must allow the robot to develop the set of tasks for which it has been designed, respecting the fundamental laws of robotics [2]:

1. A robot may not injure a human being or, through inaction, allow a human being to come to harm.
2. A robot must obey the orders given by human beings except where such orders would conflict with the first law.
3. A robot must protect its own existence as long as such protection does not conflict with the first or second laws.

In Ref. [3], a revised set of five laws has been proposed:

1. Robots are multi-use tools. Robots should not be designed solely or primarily to kill or harm humans, except in the interests of national security.
2. Humans, not robots, are responsible agents. Robots should be designed and operated as far as practicable to comply with existing laws, fundamental rights, and freedoms, including privacy.
3. Robots are products. They should be designed using processes which assure their safety and security.
4. Robots are manufactured artifacts. They should not be designed in a deceptive way to exploit vulnerable users; instead, their machine nature should be transparent.
5. The person with legal responsibility for a robot should be attributed.

In order to fulfill the fifth law, some considerations about the possibilities to modify the robot's predefined behavior must be taken into account. So, in this chapter we consider five levels of behavior modification, from a closed architecture to a completely open architecture:

a. Closed architecture

In this category we include all robots with a non-customizable predefined behavior. Examples of this type of robots are the household robots, as Roomba by iRobot, and entertainment and leisure robots, as Robotic Balls by Sphero. A closed version for Fitorobot can be considered. So, the sole responsibility of the farmer is to use the robot properly.

b. Autotuning from user behavior

This category includes robots that learn from the user. The driverless car is one of the clearest example of mobile robots. With the last advances in this technology, it is possible that in the near future these cars, in order to satisfy the particular user expectations, incorporate an algorithm that learns from the user skills and his driving styles.

c. Setup parameter calibration

Algorithms controlling the robot behavior in general have parameters depending on the context or particular application of the robot. So, for example, if an artificial vision algorithm is used to navigate, there will be some parameters for the algorithm that will depend on the characteristic context for the environment navigated by the robot. For example, if we consider the Fitorobot presented in Section 2, we need to fix the umbral value for flooring color

so that the robot can determine the way to follow centering itself between plants. **Figure 3** shows two different types of flooring, a soil cropping greenhouse and an hydroponic crop greenhouse with mulching. For both cases different umbralization values must be chosen in a manual way, so if the incorrect value is chosen by the farmer, the greenhouse corridors will not be correctly identified, and the Fitorobot will collide with the plants causing a disaster with important economic consequences. **Figure 4** shows an example of segmented corridor.



(a) Soil cropping greenhouse (b) Hydroponic crop greenhouse with mulching

Figure 3. Different types of a greenhouses flooring.



Figure 4. An example of a segmented corridor marking the trajectory to follow by the Fitorobot.

Two types of calibration may be considered:

i. Automatic calibration: The algorithm automatically sets the parameters. In this case, the robot identifies the most adequate values for the particular environment. However, a problem may occur if the automatic calibration run under conditions other than those used in the calibration stage.

ii. Manual calibration: Parameters are set by the user. In this case, the user chooses the value for parameters, so the responsibility for the good performance of robot lies with the user.

d. High-level programming

There are robots that can be programmed but with certain limitations. So, the maker partially opens the architecture robot, so that the user can customize its behavior in a single way. An example of robot included in this category is Aisoy, a social robot by Aisoy Robotics, in which speaker, microphone, camera, contact sensors, and single movements of the head, eyelids, eyebrows, and body can be programmed in order to interact with a person.

e. Low-level programming

This is the most customizable category. All hardware and software elements can be modified in order to obtain a determined behavior. This category would include the research robots. However, other end-user robots may be also considered, although it is important to note that these robots will commonly be programmed by an intermediate agent between the maker, the user, and the engineer. From the user point of view, the robot would have a closed architecture. Fitorobot is also an example of this type of robot.

4. Robots liability

Action and responsibility are two close concepts; we are responsible for all of our actions because our behavior is guided by reasons. Something similar occurs with robots.

The Committee on Legal Affairs of the European Union considers that the more autonomous robots are, the less they can be considered simple tools in the hands of manufacturers, owners, or users. As a consequence, the ordinary rules on liability are insufficient.

To fully manage this issue, two work lines are proposed. The first one is to establish a direct link between the robot behavior and the knowledge learned from the owner. In this sense, the owner plays a teacher role, so the longer a robot's *education* has lasted, the greater the responsibility of its owner should be.

The second one is the possibility to create a specific legal status for the most sophisticated robots (named *electronic persons*) and to apply electronic personality to cases where robots make smart autonomous decisions or otherwise interact with the third parties. That last argument has an abstract, legal, and ethical meaning that is too different from lines of this work, so we are going to focus on the first point.

We will begin by saying that although the Committee highlights that robots need special rules based on their nature, robots are not so different from other industrial and smart mechanical tools. Robots are machines, and all machines can produce damages because they are badly handled or because they are defecting products.

One of the most serious problems is the European legal analysis of defect concept, in terms of finding similarities or contrast between different court points of view. The differences

of perspective and of process will have fundamental importance in the practical decision-making [4].

A summary of legal aspects about liability is presented in the following paragraphs. Considering robots as products, we are going to introduce different liability regimes depending on the subject, focusing on cases of shared liability for making a debate more interesting than the strict responsibility of the manufacturer or the user.

4.1. The liability of the producer

The manufacturer (or producer, as named Directive 85/374/EEC of 25 July 1985 on the approximation of the laws, regulations, and administrative provisions of the member states concerning liability for defective products) can mean:

1. The producer of a raw material and the manufacturer of a finished product or of a component part
2. The importer of the product
3. Any person putting their name, trademark, or other distinguishing features on the product
4. Any person supplying a product whose producer or importer cannot be identified

This multi-identity situation is a consequence of *pro consumatore* doctrine: an unknown producer cannot be a reason for unprotecting the consumer for damage caused by his products.

From a general point of view, we can say that a robot is composed of two things: software and hardware. In effect, it should do the distinction between producer and programmer, being the first person responsible for the whole electronic (sensors and actuators) and mechanical parts and the second person responsible for the intern process of the robot (learning capability, image processing, decision-making process, etc.).

Currently, and observing the lines of responsibility drawing by Directive 85/374/EEC, the difference between both is not relevant for the injured person, but in order to propose a new model and considering that the user can be the programmer sometimes, the labeling of the Directive could not be enough.

The producer can be responsible on two ways: based on detrimental qualities of his products (defective product) and based on the product behavior (negligence).

A product is defective when it does not provide the safety which a person is entitled to expect, taking all circumstances into account, including:

1. The presentation of the product.
2. The use to which the product could reasonably be expected to serve; in the legal practice, this also involves designing the product to avoid inappropriate use of situations. In consequence, it is not possible to avoid the defect product category just complying with the existing European and national regulations or requirements (e.g., ISO rules). In the field of

robotics, this consideration can be substantial, even if we talk about limiting the autonomy of robots as a way to prevent damages.

3. The time when the product was put into circulation.

The presentation of the product is not advertising and packaging only; it is the user manual and the information about components, characteristics, mode of use, and contraindications too. The presentation must be clear and detailed.

The injured consumer must prove that the damage is actual and caused by a defect in the product, but he/she does not have to prove the negligence or fault of the producer or importer; a causal link between the damage and the defect is enough.

Following the law, there are three categories of deficiencies: first, the product does not correspond to those of its same series (manufacturing defect). The second one consists of products that have a fault in its conception, so the deficiencies affect the whole series (design fault). And, third, the products have a perfect design and manufacturing but could be potentially dangerous if the user does not have adequate information to use it (defect of information).

In this respect, the Council of European Convention in 1977 rejected the adoption of an exclusive liability regime for dangerous products. This criterion has been followed later in the regulations on this matter. The dangerous product does not mean unsafe product: the safety is in relation to the handling and consequences arising from its use. A product can be dangerous and safe at the same time.

Observing this classification, we do not consider that a new category of responsibility for damages produced by a robot is necessary. Robots will be as unsafe as producers want to make them.

4.2. The liability of the programmer

We must distinguish two kinds of programmers: the programmer in the proper sense, whose job is to prepare the robot for be used by the customer, and the user-programmer, who is a user that programs the robot with the limitations established by the manufacturer.

4.2.1. The programmer

The programmer will be responsible for damages caused by robots when the damage is related to software failures, faults or errors.

A *failure* is an event that occurs when the correct service deviates, because it does not comply with the functional specification or because this specification did not adequately describe the system function. The *error* is defined as part of a system's total state that may lead to a failure. And, the cause of the error is called *fault* [5].

If a user acquired a closed architecture robot, any damage produced by the machine is the liability of programmer and so a product liability case.

The producer assumes to own all acts and omissions by his/her employees which occur during the course of their employments. If the programmer is employed by the manufacturer, the assumption of liability for one or the other is significant for his domestic political only. And, if the programmer is external, the producer assumes his/her acts too, because someone contributes to produce the final results but it has a procedural advantage: the producer can direct an action of compensation for damages to the external programmer.

4.2.2. The user-programmer

When the robot allows a degree of personalization, there is an effect of displacement of liability and voluntary assumptions of risk by the user.

This displacement cannot be understood as an absolute release of producer responsibility; the starting point in this regard should be the assumption that the user is not an expert on robotics and he/she only uses the technology that others create. Indeed, the assumption of risk by the user is limited to the risk that he/she may know. The duty of information about the conditions and mode of use has an essential role to play in this case.

The customizable robots are not "empty"; they have a minimum knowledge on which to work, given by the factory software. The distinction must be made between wrong customization-calibration and wrong basis programming. In the case of damage caused by a customized robot, the user would be responsible if the injury has originated in wrong final programming only. However, if the damage had occurred in any way, regardless of whether the robot was well or badly personalized because it is a factory of software problem, the user would not be responsible.

Taking a risk only becomes negligence if the conduct in risk-taking is unreasonable. Negligence is the interference with the duty to take care, causing damage or injury to another person.

4.3. The liability of the owner

By Directive 2011/38/EU of the European Parliament and of the Council of October 25 2011 on consumer rights, *consumer* means any natural person who is acting for purposes which are outside his trade, business, craft, or profession, and *trader* means any natural person or any legal person who is acting for purposes relating to his trade, business, craft, or profession.

Literally, the farmer who trades with his/her crops and acquires Fitorobot should not be covered by Consumer Law because the robot is intended for professional use. Nevertheless, there is an exception: tort rules are applied to all injured people, and as such the direct purchaser, the holder, the bystander, or the professional are identified.

Accordingly, the trader farmer who got Fitorobot is covered by some points of Consumer Law if he/she is injured by the robot, and there is no blame or negligence in his/her conduct, acting the preceding regimes, depending on the case.

To speak about the liability of the owner, a distinction is required: we must handle rightly the concepts of owner and user. The owner is the person that has purchased the product. An owner has duties which include machine's maintenance, preservation, and upgrading. The user is the person who can use the product. Both can be the same person or not.

In the case of Fitorobot acquired by a trader farmer, the farmer is the owner. If he/she has an employee who has permission to use Fitorobot, the employee is the user. In this case, the regime previously explained about dependence between producer and programmer is applied, and if a third party suffers a damage caused by the user's fault, he/she can act against the owner, and the owner can repeat the action against the user later. For this reason, we are going to do reference to *owner* always, because in our case he/she is the ultimate responsible. The owner will be responsible for negligence.

Negligence modulates the liability regimes of other subjects. The legal regime of product liability is imperative. That means that it cannot be modulated by the will of the parties, except if the doctrine *pro consumatore* is applied: any alteration of the manufacturer's liability should be for aggravating it. So, damages produced by robots as a result of errors in manufacturing, assembly, or design usually are producer's liability, but mechanical failures can be caused by not keeping the robot in good conditions, for example, and that is a responsibility of the owner.

In case of user/owner-programmer, the regime is explained previously for damages caused by wrong customization.

About it, the Committee notes that skills resulting from education given to a robot should not be confused with skills depending strictly on its self-learning abilities when seeking to identify the person to whom the robot's harmful behavior is actually due. In this sense, the Committee tries to express its concern about the "bad ideas" that the originator has and that expresses through a robot. The originator is not the programmer or the owner necessarily, but is the person who gives the order to the robot.

That approach of the Committee implies a necessary ethical and moral debate; should the learning capability be limited? Maybe yes, but the "forbidden actions" for a robot should be determined observing each kind of robot. For example, in an ideal scenario where the robot understands the order "do not hurt," a robotic surgeon cannot differentiate between hurting and cutting the skins with a scalpel.

Safety and legality are some of the guiding principles that all producers have to observe, and it is obvious that actions as to steal or to cross the road when traffic is fluid are orders that the robot should never fulfill. So, from a more pragmatic point of view, taking into account the state of the art, it must be determined if the executed order by the robot and the purpose for which is intended are related. A mobile robot with the ability to process orders that implies to make a displacement must be limited to avoid dangerous situations of abuse by the user.

The lack of restraints at the learning ability should be considered as programmer liability. It is without doubt a sign of design deficiency.

5. A potential solution: the Robotic Liability Matrix (RLM)

The model proposed by the Committee observed the learning ability of the robot to distribute the liability between the owner and the manufacturer. In effect, it looks like it is a proposal for robots with the ability to learn only, but the real intent of the Committee is to apply this rule to all autonomous robots. The reason for the confusion is that in this discourse "autonomy" and "ability to learn" concepts are intermingled, forgetting that these two concepts should be analyzed independently. On the one hand, robots can have different levels of autonomy, from a low level for tele-operated robots to a high level for robots with the ability to learn from the environment and react appropriately. On the other hand, also different levels for the ability to learn exist for different robots. For example, a classical industrial manipulator robot has no ability to learn, and an advanced social robot has to learn in order to be able to interact appropriately with humans. So, a robot can be autonomous in a particular task (e.g., a typical industrial task) and has no learning ability (e.g., the classical industrial robots previously mentioned), or has full learning ability and not be autonomous (e.g., a social robot with no ability to move).

Building on this preliminary proposal, our aim is to offer a liability distribution system that allows being applied to any type of autonomous robot, with or without a learning capacity.

5.1. Key factors

In order to ensure reliable results, the next elements have been identified as adequate and robust to endow the Matrix:

Environments. The environment, meaning as the particular situation in which a robot is present, influences on the way in which responsibility is distributed. The more complex it is, the more diligence the producer expects, and the more care the owner must take. In this first stage, we are using the next environment classification system [6]:

- Deterministic or nondeterministic: An environment is deterministic if the next state is perfectly predictable given the knowledge of the previous state and the agent's action.
- Static or dynamic: Static environments do not change, while the agent deliberates.
- Full or partial: A fully observable environment is one in which the agent has access to all information in the environment relevant to its task.
- Single or multiple agent: If there is at least one other agent in the environment, it is a multi-agent environment. Other agents might be apathetic, cooperative, or competitive.
- Known or unknown: An environment is considered to be "known" if the agent understands the laws that govern the environment's behavior.
- Episodic or sequential: Sequential environments require memory of past actions to determine the next action. Episodic environments are a series of one-shot actions, and only the current (or recent) percept is relevant.

- Discrete or continuous: A discrete environment has fixed locations or time intervals. A continuous environment could be measured quantitatively to any level of precision.
- Simulated or non-simulated: In a simulated environment, a separate program is used to simulate an environment, feed percepts to agents, evaluate performance, etc.

Black box equipment. Recording system aboard of the robot is required.

Sensors and actuators. Both are relevant to weigh the risks; the difference between strict liability and negligence of producer depends of the kind of sensors and actuators chosen. Sensors recover data, and actuators act in a consequence; so, the design of the robot must be completely adequate for the tasks. It is not about the quality only but is about safety too.

Mechanical structure. Is the frame of the robot, the skeleton. Safety is not a requirement to individual parts of robots only but is a reference to the whole package too. A dangerous design (cutting edges, heavy materials, or moving components) may affect the risk and harm evaluation.

Learning capability. Referred to real learning capability, and no mere appearance of learning process when the robot is executing a program or accessing to a cloud database where it could download instructions or information. Real learning is the ability to acquire data and elaborate information in order to complete its tasks (Mathias, 2004).

Levels of automation. There are multiple definitions for levels of automation. In order to provide clarity and consistency, we adopt the SAE International definitions for levels of automation, where Level 0 is *no automation* and Level 5 is *full automation* (SAE International).

Depending on the level, the Matrix modulates the responsibility of the producer, the programmer, the owner, or the user using other parameters as the learning capability, the initial knowledge acquired, and the robot architecture.

Human intervention. That factor is really voluble and must be observed carefully. As we said, different kinds of liability can be attributed to different subjects, observing the type of damage and the circumstances of the accident.

In some cases, the final damage can be produced by a concatenation of negligent facts, so it will be necessary to determine how much responsibility each has, and also take into consideration the autonomy of the machine.

5.2. Fundaments

Liability distribution cannot be done observing learning capability and knowledge learned from the owner only, because some software or hardware failures can be produced by the owner.

The Matrix works with identified and isolated accident situations, quantifying the level of implication of each subject in the accident. The accident is not an isolated fact, but is integrated by several and little facts whose consequence is the final accident. Each of one of these facts,

which we named *stages*, must bee analyzed and quantified separately. This process of individualization allows us to distinguish between software and hardware failures and human errors.

The stages are checked with an objective questionnaire designed over the user fault principle. If the user fails, the answer to the question is YES. If it is not, the answer is NOT. Each answer has a value of 1, so the final sum of all values determines the degree of involvement of the subject in the accident. Some answers need to be evaluated with reversed values or, as we say, "falsely answered"; it means some YES answers will sum a value of 1 into the NO answer lines and vice versa. That would happen when to keep a *pro consumatore* position could vitiate the results.

Once the analysis of stages is finished, we elaborate the Matrix over the values obtained in each stage, which will show how the liability must be distributed adding up the different values for each subject.

A simplified example of this is presented below.

The RLM has been adapted here to show simple and visually how it works, so that some elements as the types of environments, which have large and complex influence, are not going to be used. The stages have been reduced to one, so the facts happen in a linear manner. There are two tests, one for sensors and other for system. Each test has seven questions and must be remarked that they are oriented to get the liability of the user, so the last questions are about the fault or negligence and extenuating and aggravating circumstances. The questions that must be falsely answered are marked with the letter (F). The producer and the professional programmer on one side, and the owner and the user on the other, will be considered as the same person. The large version of the Matrix would be possible to establish a clear differentiation between each subject, observing the degree and timing of involvement of them.

We are going to use a closed version of Fitorobot. This means that the user cannot customize it. Fitorobot is equipped with different sensors, as a camera for detecting the center of greenhouse corridors using the color as reference, a set of sonars for the same task but when the crop is low, and a touch sensor to stop the activity when Fitorobot hits something. Fitorobot has a bug-testing function that the user must check previously to start the activity. If this test is not realized, Fitorobot does not show any failure signal. The robot has installed a bug-warning system but only works if the bug is produced during the activity. The user has a remote control with an emergency-stop button. The software of Fitorobot is updated to the last version; however, it is supposes that some users have noticed some problems with this update.

5.2.1. Sensor test

1. Has there been a sensor failure? (F)
2. If there is a bug-testing function, has it been executed? (F)
3. If there is a bug-warning system, has it been worked?
4. Is there any relation between the presence of the obstacle and the action/omission of the user?

5. Has it noticed fault or negligence?
6. Has it noticed some extenuating circumstance? (F)
7. Has it noticed some aggravating circumstance?

5.2.2. System test

1. Has there been a system failure? (F)
2. If there is a bug-testing function, has it been executed? (F)
3. Is the software updated? (F)
4. Does any error reports about the last update exist? (F)
5. Without the existence of a bug-warning system, would there be fault or negligence in the action-omission of the user?
6. Has it noticed some extenuating circumstance?
7. Has it noticed some aggravating circumstance?

5.2.3. Case 1

The user activates Fitorobot, but he/she does not check the system using the bug-testing function. Fitorobot starts its activity, moving around the corridors. The user has not verified the robot's working environment, so there is a box in the ground and both have the same color. The camera of Fitorobot cannot distinguish between the box and the ground, so it hits the box. The impact blocks the Fitorobot's chain drive causing the robot to cross the corridor, smashing some crops. The bug-warning system works correctly, and the user pushes the remote emergency-stop button after some seconds. **Table 2** shows responses for camera test. The responsibility values for user and producer, computed from the liability index L_j ($j = 1$ for user and $j = 2$ for producer) are given in (Eq. (1)):

$$L_j = \sum_{i=1}^{7} p_i \, V_{ji}; \; j = \{1, 2\} \tag{1}$$

where ρ_i is the weight for each question and V_{ji} is given by (Eq. (2))

$$V_{ji} = \begin{cases} 1 & \text{if } Q_i \text{ is marked with X or } Q_i \text{ is marked with F in } (j \bmod 2) + 1 \text{ row} \\ -1 & \text{if } Q_i \text{ is marked with N in row \#1 and 0 in other cases} \end{cases} \tag{2}$$

Are $L_1 = 5$ and $L_2 = 1$ in this case.

In this case $p_i = 1$ is chosen for i in {1, 2, 3, 4, 5, 7}, and $p_6 = 0.5$ is used, so that Question 6 does not cancel the full value for a direct question.

	Q1	Q2	Q3	Q4	Q5	Q6	Q7
YES			X	X	X		
NO	F	F				N	X

Table 2. Robotic Liability Matrix (RLM) for camera sensor.

The conclusion is that the damages produced by Fitorobot have been caused by the user's fault largely. The importance of the distribution in this case is that we are not observing the harmful event only, but the whole chain of events from the source to the end. If the user had used the bug-testing function previously, he would have discovered that the sensor was not working correctly. It may seem irrelevant because this does not alter the fact that the box is in the ground and Fitorobot could not have distinguished one thing from the other, but the negligent acts of the user are linked and must be weighed for their computation.

Moreover, it may seem irrelevant (even unfair) that the Question 1 declares the existence of a fail. It attaches one point of guilt to the producer when the accident would have happened in any case. It must be considered that a product is created to work correctly. The fact that the error has no any implication in this occasion does not reduce the responsibility of the producer, who has to guarantee that his/her products do not produce damages.

In the large version of the RLM, it would be possible to determine what extent the answer of Q1 affects to the liability distribution and not attribute a global value only.

5.2.4. Case 2

Taking the case above, we suppose that the touch sensor and the bug-warning system do not work, so the user is not alerted about the crush. Fitorobot keeps moving around the greenhouse smashing the crops. The user is alerted by the noise, and he uses the remote control to stop the robot. The same previous values as in Case 1 for weighing indexes p_i are used in this case.

From RLM for touch sensor (**Table 3**), the liability distribution with respect to this sensor is given by $L_1 = 3$ and $L_2 = 3$.

From RLM for bug-warning system (**Table 4**), the liability distribution with respect to this system is given by $L_1 = 3$ and $L_2 = 3$.

Taking results from RLMs in **Tables 2–4** into account, the user has 11 points of responsibility, and the producer has 7 points. But the most interesting data is the result of bug-warning system test;

	Q1	Q2	Q3	Q4	Q5	Q6	Q7
YES	F			X	X		
NO		F	X			N	X

Table 3. Robotic Liability Matrix (RLM) for touch sensor.

	Q1	Q2	Q3	Q4	Q5	Q6	Q7
YES	F		F	X	X		
NO		F				N	X

Table 4. Robotic Liability Matrix (RLM) for bug-warning system.

if the sequence is isolated, it is observed that the producer has more responsibility than the user in the accident. The main reason is that the producer/programmer is responsible about the good work of the internal components. Regarding this issue, the user has to be sure to keep the system updated and to stay informed about the errors published by the producer exclusively.

However, as mentioned above, the Matrix allows observing the damage chain as a whole, so in Case 2, the negligence of the user exceeds in 1 point of the liability of the producer. Into a system of distributive blame, the user is responsible at 61.11% and the producer at 38.89%.

It must be highlighted that the results do not involve that the producer has no any liability because the user is *guiltier*. The Robotic Liability Matrix is designed for distributing the liability, not for attributing it.

6. Conclusions

We have introduced Fitorobot, a successful practical case by the University of Almería, joined to a sample of Robotic Liability Matrix, a new fairer method to distribute the liability to all subjects involved in the use of robots.

In the first instance, the classic theories about product liability are applicable to robots that require continuous human supervision, but in the case of more autonomous robots the standard rules must be improved. The statement approach by the European Parliament, consisting of observing the learning ability to attribute the liability, may not be enough and equitable; all things considered, new products require new solutions.

The result contents in this paper show how the currently product guarantee scheme is obsolete when the product is autonomous in any degree, so the liability for damages does not correspond only to one person.

The Matrix has a competitive advantage over other systems: it will allow defining what would be the content of the guidelines for manufacturers and programmers. A guideline can work as self-regulation system when the Laws are not drafted or passed, and at the same time, it can be used by stakeholders to result in knowledge of the ethical, moral, and legal limits of their work. Guidelines really help reduce time to promote innovation and technology, because they use consensus-based standards, which are recognized by the industry.

The design of the Matrix is under development and improvement process, trying to adequate it to different kinds of robot and situations, but in isolate and laboratory cases, the Matrix is actually working.

Author details

Alejandro Zornoza[1]*, José C. Moreno[2], José L. Guzmán[2], Francisco Rodríguez[2] and Julián Sánchez-Hermosilla[2]

*Address all correspondence to: alejandro.zornoza.somolinos@gmail.com

1 Universidad Carlos III de Madrid, Madrid, Spain

2 Universidad de Almería, Almería, Spain

References

[1] Sánchez-Hermosilla J, Rodríguez F, González R, Guzmán JL, Berenguel M. A mechatronic description of an autonomous mobile robot for agricultural tasks in greenhouse. In: Barrer A, editor. Mobile Robots Navigation. InTech; 2010. pp. 583-560. DOI: 10.5772/900

[2] Asimov I. I, Robot. 6th ed. Greenwich, Conn.: Fawcett Publications; 1950. 192 p

[3] Winfield A. alanwinfield.blogspot.com [Internet]. October 30, 2013.Available from: http://alanwinfield.blogspot.com.es/2013/10/ethical-robots-some-technical-and.html [Accessed: March 25, 2017]

[4] Whittaker S. Liability for products. English Law, French Law, and European Harmonisation. 1st ed. Oxford University Press, New York, 2005. 752 p

[5] Avizienis A, Lapire JC, Randell B, Landwehr C. Basic concepts and taxonomy of dependable and secure computing. IEEE Transactions on Dependable and Secure Computing. 2004;**1**(1):11-33. DOI: 10.1109/TDSC.2004.2

[6] Russell SJ, Norvig P. Artificial Intelligence: A Modern Approach. 2nd ed. New Jersey: Prentice Hall; 2010. 1132 p.