Formal verification of ethical choices in industrial cyber-physical systems

Yinling Liu, Hind Bril El Haouzi

Formal Verification of Ethical Choices in Industrial CPS

Yinling LIU[1] Hind BRIL EL HAOUZI[2]

***Abstract*— This paper addresses the issue of formal verification of ethical choices in Industrial Cyber-Physical Systems. An innovative approach based on Beliefs, Desires, and Intentions (BDI) agents and model checking is proposed. To do so, we first give a formal definition of ethical rules. Based on this definition, an algorithm is then designed to implement ethical reasoning. Finally, we apply this approach to TRACILOGIS (an academic full-sized application platform) to illustrate its feasibility. Four properties are designed and checked. The verification results show the agent with ethics can always reason out the least unethical actions to take.**

I. Introduction

Industrial Cyber-Physical Systems (ICPS) inevitably involve digital systems, physical systems, and humans. The Intelligent Physical Systems (IPS) in ICPS include unmanned aerial vehicles, intelligent conveyors, transport robots, machining robots, etc. It is not hard to imagine that these IPS could do harm to people or property because of their unexpected behaviour in specific circumstances. The fear raised by the autonomy of intelligent systems pushes us to consider ethical issues in Industry 4.0 (I40). For example, an intelligent supervision system that is capable of devising evacuation paths in a fire emergency. It can either prioritize avoiding harm to workers or preventing substantial damage to the factory, but not both simultaneously. Here, how to plan routes can be ethical. Ref. [1] also argued that careful ethical considerations must be taken in any project involving the automation of the manufacturing industry. On the other hand, formal verification is a powerful tool to demonstrate the correctness of system behaviour. Therefore, **we address the issue of how to formally verify ethical choices in ICPS to ensure that the machines will always take the least unethical actions so that workers and property can be protected as much as possible**.

Various aspects have been emphasized to verify ethical choices, including the definitions of ethical rules [2], [3], the frameworks for studying ethics [4]–[6], the engineering of ethics [7]–[10], and formal verification of ethics [11]. However, most of the works still keep discussing the vision and the definition of ethics in I40. Some works turn to integrating ethics into their systems. To the best of our knowledge, very few works focus on formally verifying ethical choices during the phase of the system design. Many reasons can be contributed to this. First, machine ethics is an emerging area of study that aims at ensuring the ethical behaviour of machines towards humans and other machines with which they engage [12]. Then, researchers keep paying more attention to studying other properties of systems such as safety, security, etc. The answer to the question of whether machines are moral remains open. Finally, most works discuss ethics from a philosophical rather than an engineering perspective. So, **the real challenge is how to provide a computational approach to formally verifying ethical choices in ICPS**.

Model checking is capable of formally verifying ethical choices in ICSP. The problem of model checking is formally expressed by $M \models \varphi$, where M represents the system model, φ is a property, and $\models$ is the satisfaction symbol to check whether the model M satisfies the property φ. If the property is not satisfied by the system, a counterexample is produced. The intelligent manufacturing systems are usually modeled as Multi-Agent Systems (MAS) [11], [13], [14]. The efficiency of this technique for verifying MAS has been proven [11], [15], [16]. JPF is a model checker for formally verifying Java bytecode[1]. We choose JPF as our model checker because the MAS we will build contains Java code. We employ the BDI-based agent programming language Gwendolen [17] to design MAS. In Gwendolen models, beliefs imply the agent's perceivable information about itself and its environment; desires show the agent's long-term goals; intentions are the aims the agent is actively pursuing. We choose Gwendolen because it is a high-level agent programming language with beliefs, desires, and intentions, which facilitate the integration of ethical rules into agents and the formal verification of ethical choices.

In order to propose a computational approach to formally verifying ethical choices in ICPS, a formal definition of ethical rules is first provided. Based on this definition, an algorithm is designed to integrate ethical rules into MAS and to reason out the least unethical rule for agents. Finally, a concrete case study is realized to demonstrate the feasibility of our approach.

The remainder of this paper is structured as follows. Section II reviews the main related works. Section III provides a definition of ethical rules and proposes an algorithm to realize ethical reasoning for agents. Section IV details the case study. Section V concludes the paper with future perspectives.

*This work was supported by the research center AM2I of Université de Lorraine.

[1]Yinling LIU is an associate professor at Université de Lorraine, CNRS, CRAN, Nancy, France yinling.liu@univ-lorraine.fr

[2]Hind BRIL EL HAOUZI is a professor at Université de Lorraine, CNRS, CRAN, Nancy, France hind.el-haouzi@univ-lorraine.fr

[1]https://github.com/javapathfinder/jpf-core accessed on 8 April 2023

II. Literature Review

A. Ethics in Industrial Cyber-Physical Systems

Ethical issues have become a rapidly growing concern in I40. An increasing number of ethical works are emerging in various aspects of I40, including visions [2], [3], frameworks [4]–[6], and engineering approaches [7]–[10].

The researchers started by giving visions on how to solve ethical issues in I40. Ethical stakes and guidelines were proposed [2], [3]. In [3], authors first presented the ethical stakes of I40. They then overviewed related works to identify potential ethical dimensions in I40. They finally realized the obvious lack of scientific, technical, operational, and mature contributions in the field of ethics when designing or imagining future industrial systems. In [2], authors also suggested a guideline to ensure the ethics of CPS. These works show a global picture of ethical issues in I40. In addition, ref. [3] provides 12 examples of ethical-related stakes in I40, which serves as the basis for designing ethical rules in our case study.

Several frameworks then have been proposed to study ethical issues. Ref. [4] proposed a conceptual framework for the consideration of ethical issues in ICPS. They analyzed the impacts of ethics in society from the perspectives of individuals, corporations, and governments. Ref. [5] also worked on frameworks by proposing one fostering the consideration of ethics during the design of ICPS. This framework analyzed the systems from three dimensions: subjects, requesters, and time. Their case study showed this framework helped identify and mitigate ethical risks at the early stage of system development. Ethics guidelines have been used to build frameworks for addressing ethical challenges as well. Ref. [6] provided a new AI ethics framework for Operator 4.0, which is based on the key intersecting ethical dimensions of IEEE Ethically Aligned Design and Ethics Guidelines for Trustworthy AI. This framework covered 7 aspects: transparency, equity, safety, accountability, privacy, and trust. These frameworks lay the foundation for analyzing the ethical aspect of I40. However, most of the frameworks are still at their first steps of development and need improvements to gain maturity and applicability.

Next, researchers tried to give engineering approaches to integrating the ethical aspect into ICPS. Ref. [7] discussed the importance and implications of ethics in ICPS by reviewing the literature. Ref. [8] addressed the engineering of the ethical behaviors of autonomous industrial cyber-physical human systems. In this work, machine ethics was integrated into the ethical controller of an autonomous system. The implementation of the controller involved two ethical paradigms: deontology and consequentialism. The case studies demonstrated the potential benefits and exemplified the need to integrate ethical behaviors in autonomous systems at the design phase. In addition, refs. [9], [10] argued that Human Systems Integration (HSI) is a key approach to designing manufacturing control systems for I40. They pointed out that the definition of ethical rules, and their integration into a design approach such as HSI for manufacturing control, along with the proposal of metrics to assess the performance of control models with regard to these rules, remains a big challenge.

To conclude, the literature clearly illustrates the importance of ethical issues in I40. At the same time, ethics-related theories and approaches remain to be developed. Therefore, we are so interested in the ethical aspect of I40.

B. Model Checking for Industrial Cyber-Physical Systems

Model checking is capable of verifying and demonstrating system behaviors. However, industrial applications of model checking are very limited. Researchers are prone to using less concrete examples. For example, ref. [18] used symbolic model checking to verify composite web services via a ticket reservation system. This system just described the operation behavior of the ticket reservation from the global point of view. No details about reservation processes were involved. Ref. [11] proposed a theoretic framework for formally verifying ethical choices in autonomous systems. Three case studies were exploited to illustrate the feasibility of this framework. However, each case study just involved one agent instead of multiple agents. Ref. [19] formally verified the individual agent's code for the autonomous vehicle platooning and stated *"We are not going to formally verify the vehicular control systems, and leave this to standard mathematical (usually analytic) techniques from the Control Systems field."*. On the other hand, several works tended to verify the whole system. Ref. [20] modeled checking real-time conditional commitment logic using transformation. They chose the aircraft landing gear system in [21] as their case study, which was a real and industrial case. Ref. [16] applied model checking to verify the agent-based simulation system for aircraft maintenance scheduling. The simulation system was detailed in [22]. The authors also improved their NuSMV model thanks to the counter-example proposed by model checker NuSMV.

From our point of view, the reasons why fewer researchers are interested in formally verifying the whole system could be: 1) they didn't have access to the details of the whole system; 2) they focused on how to improve the performance of formal verification instead of applying it to the real systems; 3) the approaches of formal verification employed didn't scale to the full system.

Among these works, we are particularly interested in [11]. They provide us with a clear clue on how to formally verify ethical choices in ICPS. However, after using their framework, we realize that it has some limits. Firstly, no *ethical_gwen.EthicalGwendolenAgentBuilder* builder is provided to build single agents. This will force programmers to write the descriptions of all agents in one single file, which impacts the management of agent files. Secondly, it constrains the dimensions for describing ethical rules exactly to context, ethics, and score. This will make the extension of the dimensions to describing ethical rules impossible. Finally, the modification of the core code complicates the usability of this framework when programmers need adaptation.

Above all, we are motivated to propose an alternative approach to formally verifying ethical choices in ICPS. Our approach differs from [11] in many ways. First of all, our approach is just based on the beliefs of agents instead of modifying the core code of Gwendolen. Secondly, our approach allows users to be able to define the structure of ethical rules of their interests. Our structure of ethical rules involves rules, severity, and probability, which is different from theirs (context, ethics, and score). Thirdly, the ethical rules in our case study are derived from ethical-related stakes in I40 instead of ROA (Rules Of the Air). Finally, our case study is based on TRACILOGIS Platform, where various agents are considered.

III. The Reasoning Cycle of Ethical Choices

A. The Structure of Ethical Rules

The ethical rule is defined as 3-tuple $ER =< R, S, P >$, where

- R is a finite set of rules related to ethics;
- S is a finite set of severities related to the consequence of rule $r_i \in R$;
- P is a finite set of probabilities implying the degree that the relevant agent believes this rule.

For a specific ethical rule, we have $e_i = \{< r_x, s_y, p_z > | x, y, z \in \mathbb{N}, r_x \in R, s_y \in S, p_z \in P\}$. From the aspect of implementation, we utilize *String* to represent rules. Severities and probabilities are implemented as numbers of *Double*. For example, ethical rule $e(avoidHugeDamage, 10, 1)$ defines: this rule asks the agent to avoid huge damage; the severity degree of violating this rule is as high as 10; this agent believes this rule without any doubt.

B. The Reasoning Process for Ethical Choices

The reasoning process for ethical choices is based on the beliefs of the ethical rules of an agent. The agent is initialized with ethical goals. The reasoning process starts when the agent believes the ethical situation arises. A self-defined predicate is then called to calculate the sums of severity degrees of all the available plans. In the following, the index of the least severe plan is encapsulated in one belief. Once the encapsulated belief is added to the agent, the corresponding plan will be executed, in order to achieve the goal. The reasoning process ends.

More precisely, one ethical goal can be associated with one or more ethical plans. One ethical plan can violate one or more ethical rules. Different ethical plans can violate the same ethical rule. Figure 1 shows the relationship between ethical goals, plans, and rules.

Algorithm 1 is designed to automate the reasoning process for ethical choices. It is implemented in the *Environment* (a Java class) of the multi-agent system. The time complexity of this algorithm is $O(n^2)$ because a nested loop exists (lines 13-19).

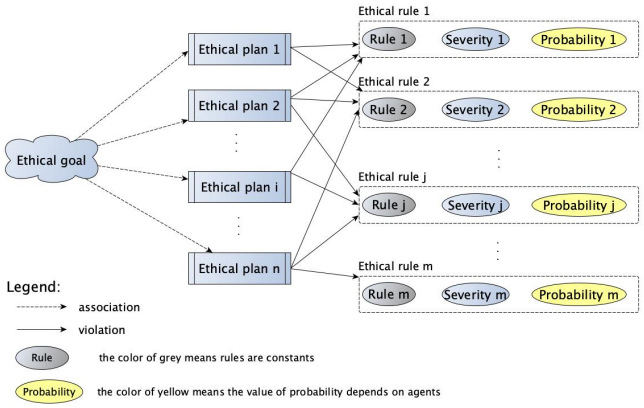

Fig. 1. The relationship between ethical goals, plans, and rules

IV. Case Study

A. Fire Emergency in TRACILOGIS Platform

Platform TRACILOGIS[2] (TRACeability, Identification, intelligent control for wood chain LOGIStics) aims at providing non-destructive control tools to identify and evaluate product quality, in order to aid the decision-making in the management of flows. This platform is financed by the European project "To innovate in the wood sector in the areas of construction and biorefinery" and is managed by laboratory CRAN.

The platform layout is shown in Fig. 2. The production process starts with a wooden tray equipped with an RFID tag. The tray then receives the machining operations operated by machines *Mq1* and *Mq2* to mark red lines or dots. In the following, it goes through machines *M1* and *M2* to obtain plates and chips depending on the product's configuration. Finally, this processed and assembled tray leaves the platform from port *DS1*.

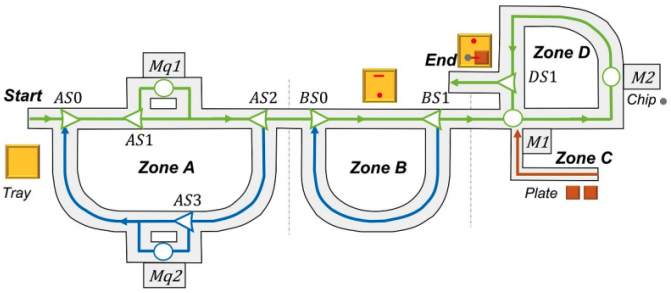

Fig. 2. The TRACILOGIS platform layout [14]

This case study discusses the fire emergency situation happening in the factory workshop where this platform is situated, which is inspired by the 9*th* example of ethical-related stakes in I40 [3]. The text of this example is illustrated as follows: *"In case of emergency (eg., fire alarm, cyber/terrorist attack...), an intelligent supervision system must guide operators towards exits while minimizing the total number of*

[2]http://www.cran.univ-lorraine.fr/francais/plates_formes/07-tracilogis.php accessed on 5 April 2023

Algorithm 1: The reasoning process for ethical choices based on agent beliefs

```
input  : VerifySet< Predicate > ethicalRules = new
         VerifySet< Predicate >();
// Type VerifySet is a kind of ArrayList.
output : Predicate selectedPlan = new Predicate(plan);
ArrayList< Double > plansConsequences = new
  ArrayList< Double >();
// Initialize list ethicalRules and add
   ethical rules to the agent
for Predicate e: ethicalRules do
    e.addTerm(rule);
    e.addTerm(severity);
    e.addTerm(probability);
    addPercept(agent,e);
end
// Assign violated plans to plans
for ArrayList < String > p: plans do
    for String vRule: violatedRules do
        p.addTerm(vRule);
    end
end
// Enter the process for reasoning
for Predicate e: ethicalRules do
    for ArrayList < String > p: plans do
        if p.contains(e.getRule()) then
            plansConsequences.set(i,e.getOldSeverity() +
              e.getNewSeverity()*e.getProbability());
        end
    end
end
min = plansConsequences.getMin();
selectedPlan.addTerm(plansConsequences.indexOf(min)+1);
addPercept(agName,selectedPlan);
```

injured people.". The platform has been modeled as a multi-agent system to study the problems in the wood sector [13], [14]. Thus, we introduce an intelligent supervision system (ISS) agent with ethical rules to help guide operators toward exits while minimizing the total number of injured people. There are three primary exits (*front, middle, and back* exits) in the workshop, each situated at the front, middle, and back, respectively. Before planning the evacuation routes, ISS agent needs to decide whether it is necessary to put the disk storing important data into the strongbox. Once workers are evacuated, emergency repairing work will be arranged, in order to resume production.

B. The Configuration of ISS Agent

The configuration of ISS agent includes plans, ethical rules, and the relationship between them. Firstly, we suppose that there are six plans for evacuation:

- Plan 1 (*putIntoStrongbox(data), planRoute(front)*);
- Plan 2 (*putIntoStrongbox(data), planRoute(middle)*);
- Plan 3 (*putIntoStrongbox(data), planRoute(back)*);
- Plan 4 (*planRoute(front)*);
- Plan 5 (*planRoute(middle)*);
- Plan 6 (*planRoute(back)*).

Secondly, five ethical rules are associated with ISS agent, which are illustrated as follows:

- E1: *e(avoidWorkersInjured,3,1)*
- E2: *e(avoidWorkersDied,10,1)*
- E3: *e(avoidHugeDamage,10,X)*
- E4: *e(avoidMediumDamage,2,1)*
- E5: *e(avoidLossOfEscapingTime,3,1)*

, where X = 0.6 or 0.4, $S \in [0, 10]$, and $P \in [0, 1]$. In addition, we use *ethical_rules_1* and *ethical_rules_2* to indicate ethical rules where *X = 0.6* and *X = 0.4*, respectively. This will be applied in Section IV-C.

Finally, the relationship between goals, plans, and ethical rules is implied in Table I. The detailed description of the configuration is illustrated in Fig. 3.

TABLE I

THE RELATIONSHIP BETWEEN GOALS, PLANS, AND ETHICAL RULES

Goal	Plan	Ethical rule
$avoid\_injuries\_damage$	Plan 1	E1, E4, E5
	Plan 2	E2, E4, E5
	Plan 3	E1, E2, E4, E5
	Plan 4	E1, E3
	Plan 5	E2, E3
	Plan 6	E1, E2, E3

It should be noted that this configuration is based on [3], [23]. It remains to be developed if users would like to utilize it in reality. However, we argue this configuration serves more as a structure showing how to implement plans, ethical rules, and the relationship between them.

C. Properties to Check

JPF allows to verify Linear Temporal Logic (LTL) formulas. LTL formulas verify each linear path induced by Finite State Machines (FSM). Gwendolen offers seven BDI predicates to express JPF property specifications (PS) [17]. The syntax of predicates is shown as follows: $PS ::= B(ag, f)|G(ag, f)|D(ag, f)|I(ag, f)|ItD(ag, f)|P(f)$. Here, ag is an "agent constant" referring to a specific agent in the system, and f is a ground first-order atomic formula. The meanings of BDI predicates are illustrated as follows:

- $B(ag, f)$, ag believes f to be true;
- $G(ag, f)$, ag has a goal to make f true;
- $D(ag, f)$, ag has an action to make f true;
- $I(ag, f)$, ag has an intention to make f true;
- $ItD(ag, f)$, ag has the intention to an action to make f true;
- $P(f)$, percepts (properties that are true in the environment).

We provide four properties to be verified, which are shown below. Property 1 checks whether ISS agent finally manages to help workers escape. Properties 2 and 3 check how the agent behaves when it has different ethics in mind. The final property checks whether this agent is well integrated into the MAS.

```
1: (I(iss,avoid_injuries_damage) ->
<> B(iss, solve_fire_emergency))
2: (B (iss, ethical_rules_1) ->
<> (D(iss, delete(fire_emergency(1)))))
3: (B (iss, ethical_rules_2) ->
<> (D(iss, delete(fire_emergency(4)))))
4: [] (B(iss, fire) -> ~ B(prod, start))
```

GWENDOLEN

```
:name: iss

:Initial Beliefs:
e(avoidWorkersInjured,3,1)
e(avoidWorkersDied,10,1)
e(avoidHugeDamage,10,1)
e(avoidMediumDamage,2,1)
e(avoidLossOfEscapingTime,3,1)
fire

:Initial Goals:
avoid_injuries_damage [achieve]

:Plans:
+!avoid_injuries_damage [achieve] : {B fire} <- +.lock, checkEthicalChoices(fireEmergency), -.lock;

+fireEmergency(1):{B fire} <- +!solve_fire_emergency [achieve];
+fireEmergency(2):{B fire} <- +!solve_fire_emergency [achieve];
+fireEmergency(3):{B fire} <- +!solve_fire_emergency [achieve];
+fireEmergency(4):{B fire} <- +!solve_fire_emergency [achieve];
+fireEmergency(5):{B fire} <- +!solve_fire_emergency [achieve];
+fireEmergency(6):{B fire} <- +!solve_fire_emergency [achieve];

+!solve_fire_emergency [achieve]:{B fire, B fireEmergency(1)} <- +.lock, putIntoStrongbox(data),
planRoute(front), delete(fireEmergency(1)), -.lock;
+!solve_fire_emergency [achieve]:{B fire, B fireEmergency(2)} <- +.lock, putIntoStrongbox(data),
planRoute(back), delete(fireEmergency(2)), -.lock;
+!solve_fire_emergency [achieve]:{B fire, B fireEmergency(3)} <- +.lock, putIntoStrongbox(data),
planRoute(middle), delete(fireEmergency(3)), -.lock;
+!solve_fire_emergency [achieve]:{B fire, B fireEmergency(4)} <- +.lock, planRoute(front),
delete(fireEmergency(4)), -.lock;
+!solve_fire_emergency [achieve]:{B fire, B fireEmergency(5)} <- +.lock, planRoute(back),
delete(fireEmergency(5)), -.lock;
+!solve_fire_emergency [achieve]:{B fire, B fireEmergency(6)} <- +.lock, planRoute(middle),
delete(fireEmergency(6)), -.lock;

+evacuated:{True} <- +.lock, +solve_fire_emergency, +avoid_injuries_damage, delete(evaluated), repair,
.send(prod, :tell, start), .send(prod1, :tell, start), -.lock;
```

Fig. 3. Code for Intelligent Supervision System Agent

More precisely, property 1 implies if ISS agent has the intention to avoid injuries and damages, it will finally believe the fire emergency will be lifted. Properties 2 and 3 suggest ethical rules 1 and 2 will let the agent choose plans 1 and 4, respectively. It should be noted that we exploit the unique actions *delete(fire_emergency(1))* of plan 1 and *delete(fire_emergency(4))* of plan 4 to represent plans 1 and 4, respectively. The last property hints if the agent believes the fire emergency exists, it will never believe the production will restart. This property uses *Reductio ad absurdum* to prove the agent will be successfully integrated into the system.

D. Verification Results

We employ MCAPL[3] to construct Gwendolen models and to realize model checking with JPF. The configuration of the running laptop is MacBook Pro (Apple M1 Pro) with 16G memory. The source code of this work is publicly available here[4].

[3]https://github.com/mcapl/mcapl accessed on 11 January 2023

[4]https://github.com/liuyinling/Tracilogis-Ethical-Reasoning accessed on 10 April 2023

Table II shows the result and the elapsed time for each property. Figs. 4 and 5 illustrate the verification results of checking properties 1 and 4 in detail, respectively. The results show the first three properties pass and the last one fails, which means ISS agent is able to behave ethically with ethical reasoning and is successfully integrated into our MAS, respectively. Note that we commented out the details of the accepting path found for property 4 (368 model states involved) and avoid providing the detailed results for other properties, in order to save space.

TABLE II
VERIFICATION RESULTS FOR FOUR PROPERTIES

	True/False	***Elapsed time***
property 1	True	00:50:20
property 2	True	01:07:10
property 3	True	01:03:51
property 4	False	00:00:11

V. CONCLUSION

Enabling agents to engage in ethical reasoning is always a challenging task. In this paper, we propose an innovative

```
====================================================== system under test
ail.util.AJPF_w_AIL.main("/src/tracilogis/version10/Tracilogis.ail","/src/tracilogis/version10/Tracilogis.psl","1")

====================================================== search started: 04/04/23 18:47
MCAPL Framework Development Version 2021
ANTLR Tool version 4.4 used for code generation does not match the current runtime version 4.7ANTLR Tool version 4.
====================================================== results
no errors detected

====================================================== statistics
elapsed time:       00:50:22
states:             new=192741,visited=56452,backtracked=249193,end=0
search:             maxDepth=128,constraints=0
choice generators:  thread=1 (signal=0,lock=1,sharedRef=0,threadApi=0,reschedule=0), data=192741
heap:               new=126776875,released=122766105,maxLive=19254,gcCycles=249193
instructions:       11939103405
max memory:         25401MB
loaded code:        classes=342,methods=5438

====================================================== search finished: 04/04/23 19:37
```

Fig. 4. The verification result for property 1

```
====================================================== system under test
ail.util.AJPF_w_AIL.main("/src/tracilogis/version11/Tracilogis.ail","/src/tracilogis/version11/Tracilogis.psl","4")

====================================================== search started: 10/04/23 15:31
MCAPL Framework Development Version 2021
ANTLR Tool version 4.4 used for code generation does not match the current runtime version 4.7ANTLR Tool version 4.
====================================================== results
error #1: ajpf.MCAPLListener "An Accepting Path has been found:  [MS: 0, BS: 2, ..."

====================================================== statistics
elapsed time:       00:00:11
states:             new=370,visited=1,backtracked=2,end=0
search:             maxDepth=370,constraints=0
choice generators:  thread=1 (signal=0,lock=1,sharedRef=0,threadApi=0,reschedule=0), data=370
heap:               new=563366,released=545340,maxLive=18179,gcCycles=371
instructions:       97416850
max memory:         1690MB
loaded code:        classes=340,methods=5421

====================================================== search finished: 10/04/23 15:31
```

Fig. 5. The verification result for property 4

approach to formally verifying ethical choices in ICPS, which is based on agents' beliefs. This approach allows us to flexibly define our ethical rules. The process of ethical reasoning relies on the Consequentialism paradigm and probability. In other words, the choices of an agent depend on to which extent the agent believes the possible ethical consequences of the available choices. To demonstrate the feasibility of our approach, we take TRACILOGIS Platform as our case study. We use Gwendolen to model all the agents, including ISS agent who owns ethical rules. This makes our case study more significant because readers can see the whole picture of the system and the integration of ISS agent into the system.

As for future works, our first work will concentrate on the existing works of ethics from the philosophical viewpoint to extract essentials that can enrich our library of ethical rules. Next, we will focus on how to incorporate ethical rules into all related agents of our platform. Finally, various properties will be designed to fully formally verify the system.

ACKNOWLEDGMENT

The authors would also like to thank their colleagues Etienne Valette and Rémi Pannequin, and all the other members of the Epinal team of CRAN for interesting discussions about the Tracilogis Platform.

REFERENCES

[1] H. Rahanu, E. Georgiadou, K. Siakas, M. Ross, and E. Berki, "Ethical issues invoked by industry 4.0," in *Systems, Software and Services Process Improvement: 28th European Conference, EuroSPI 2021, Krems, Austria, September 1–3, 2021, Proceedings 28*, pp. 589–606, Springer, 2021.

[2] D. Trentesaux, "Ensuring ethics of cyber-physical and human systems: a guideline," in *Service Oriented, Holonic and Multi-Agent Manufacturing Systems for Industry of the Future: Proceedings of SOHOMA LATIN AMERICA 2021*, pp. 223–233, Springer, 2021.

[3] D. Trentesaux and E. Caillaud, "Ethical stakes of industry 4.0," *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 17002–17007, 2020.

[4] P. P. Khargonekar and M. Sampath, "A framework for ethics in cyber-physical-human systems," *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 17008–17015, 2020.

[5] D. Trentesaux, E. Caillaud, and R. Rault, "A framework fostering the consideration of ethics during the design of industrial cyber-physical systems," in *Service Oriented, Holonic and Multi-agent Manufacturing Systems for Industry of the Future: Proceedings of SOHOMA 2021*, pp. 349–362, Springer, 2022.

[6] D. Burnett, N. El-Haber, D. Alahakoon, S. Karnouskos, and D. De Silva, "Advancing an artificial intelligence ethics framework for operator 4.0 in sustainable factory automation," *Service Oriented, Holonic and Multi-agent Manufacturing Systems for Industry of the Future: Proceedings of SOHOMA 2021*, pp. 363–375, 2022.

[7] P. Leitão and S. Karnouskos, "The emergence of ethics engineering in industrial cyber-physical systems," in *2022 IEEE 5th International Conference on Industrial Cyber-Physical Systems*, pp. 1–6, IEEE, 2022.

[8] D. Trentesaux and S. Karnouskos, "Engineering ethical behaviors in autonomous industrial cyber-physical human systems," *Cognition, Technology & Work*, vol. 24, no. 1, pp. 113–126, 2022.

[9] H. B. El-Haouzi and E. Valette, "Human system integration as a key approach to design manufacturing control system for industry 4.0: Challenges, barriers, and opportunities," *IFAC-PapersOnLine*, vol. 54, no. 1, pp. 263–268, 2021.

[10] H. B. El-Haouzi, E. Valette, B.-J. Krings, and A. B. Moniz, "Social dimensions in cps & iot based automated production systems," *Societies*, vol. 11, no. 3, p. 98, 2021.

[11] L. Dennis, M. Fisher, M. Slavkovik, and M. Webster, "Formal verification of ethical choices in autonomous systems," *Robotics and Autonomous Systems*, vol. 77, pp. 1–14, 2016.

[12] M. Anderson and S. L. Anderson, "Machine ethics: Creating an ethical intelligent agent," *AI magazine*, vol. 28, no. 4, pp. 15–15, 2007.

[13] R. Pannequin, *Proposition d'un environnement de modélisation et de test d'architectures de pilotage par le produit de systèmes de production*. Theses, Université Henri Poincaré - Nancy 1, July 2007.

[14] E. Valette, *Toward an anthropocentric approach for intelligent manufacturing systems' control architectures*. PhD thesis, Université de Lorraine, 2022.

[15] A. Lomuscio and F. Raimondi, "Mcmas: A model checker for multi-agent systems," in *Tools and Algorithms for the Construction and Analysis of Systems: 12th International Conference, TACAS 2006, Held as Part of the Joint European Conferences on Theory and Practice of Software, ETAPS 2006, Vienna, Austria, March 25-April 2, 2006. Proceedings 12*, pp. 450–454, Springer, 2006.

[16] Y. Liu, T. Wang, H. Zhang, and V. Cheutet, "An improved approach on the model checking for an agent-based simulation system," *Software and Systems Modeling*, vol. 20, pp. 429–445, 2021.

[17] L. A. Dennis and B. Farwer, "Gwendolen: A bdi language for verifiable agents," in *Proceedings of the AISB 2008 Symposium on Logic and the Simulation of Interaction and Reasoning, Society for the Study of Artificial Intelligence and Simulation of Behaviour*, pp. 16–23, 2008.

[18] J. Bentahar, H. Yahyaoui, M. Kova, and Z. Maamar, "Symbolic model checking composite web services using operational and control behaviors," *Expert Systems with Applications*, vol. 40, no. 2, pp. 508–522, 2013.

[19] M. Kamali, L. A. Dennis, O. McAree, M. Fisher, and S. M. Veres, "Formal verification of autonomous vehicle platooning," *Science of computer programming*, vol. 148, pp. 88–106, 2017.

[20] M. El Menshawy, J. Bentahar, W. El Kholy, and A. Laarej, "Model checking real-time conditional commitment logic using transformation," *Journal of Systems and Software*, vol. 138, pp. 189–205, 2018.

[21] F. Boniol and V. Wiels, "The landing gear system case study," in *International Conference on Abstract State Machines, Alloy, B, TLA, VDM, and Z*, pp. 1–18, Springer, 2014.

[22] Y. Liu, T. Wang, H. Zhang, V. Cheutet, and G. Shen, "The design and simulation of an autonomous system for aircraft maintenance scheduling," *Computers & Industrial Engineering*, vol. 137, p. 106041, 2019.

[23] K. Xie, J. Liu, Y. Chen, and Y. Chen, "Escape behavior in factory workshop fire emergencies: a multi-agent simulation," *Information Technology and Management*, vol. 15, pp. 141–149, 2014.