# Towards Hierarchical Planning with Social Norms and Ethical Considerations

Tammy Zhong[1], David Rajaratnam[1], Yang Song[1], Maurice Pagnucco[1]

***Abstract*— *Computational Machine Ethics (CME)* seeks to ensure ethical decision-making in machines, bridging technology and ethics. Despite the controversies and significant impacts associated with ethics in technology, practical, transparent, and accountable approaches remain scarce. This paper addresses this gap by developing a method that generates plans adhering to social norms and ethical principles. Our approach integrates and utilises domain knowledge and robot behaviour represented in a Hierarchical Goal-Task-Network (GTN) to express norms and principles, enabling contextual considerations based on the domain knowledge that would otherwise be missed. The representation is based on natural language, accessible and interpretable by non-experts to support accountability, governance and trust among stakeholders. We adopt and modify the GTPyhop goal task planner, extend the ethical rule definition from existing work and provide a replicable procedure for modelling ethical hierarchical planning in domains. Our work contributes to CME by introducing a top-down approach to ethical decision-making. Empirical evaluations demonstrate the initial effectiveness of our method in adhering to norms and principles in medical scenarios, suggesting improvements with ethical decision-making and potential for practical application in other real-world contexts.**

## I. INTRODUCTION

*Computational Machine Ethics (CME)* is a subfield of AI Ethics that concerns the implementation and enforcement of ethical behaviours in cognitive machines. This area is becoming increasingly crucial with the rapid advancements in technology and its adoption in society. As machines are empowered to take on increasingly complex tasks and make autonomous decisions, we have a responsibility to ensure that their behaviour is not only correct but also ethical. This is essential not just for preventing harm or unwanted actions, but also for ensuring that these machines can go beyond mere task performance, aligning their actions with broader ethical principles. CME works are commonly divided into three different categories [3]; top-down [8], [5], [9], [26], [27], [7], [6], [4], bottom-up [11], [12], [13], [10] and hybrid approaches [14], [15] integrating the previous two in some way. Top-down approaches attempt to constrain machine behaviours based on some set of predefined, explicit ethical guidelines or theories, whilst bottom-up methods often use examples of ethical decisions and machine learning to guide ethical behaviour.

For this paper, we focus on a top-down approach, believing predefined guidelines are crucial in ethically sensitive decision-making to ensure accountability and build trust [28] in the decision-making process. These guidelines provide a standard for evaluating actions, which is essential given the subjectivity and context-dependent nature of ethics. They help navigate ethical dilemmas objectively and offer a framework for addressing errors when they occur, similar to how traffic rules guide behaviour and handle incidents. A prevalent issue in the field is that top-down approaches often either work with simple rules and scenarios [16], [17], [6], [1] or remain at an abstract logical level [18], [5], making implementations challenging while bottom-up approaches [11], [12], [13], [10], although practical, lack transparency and accountability in their ethical decision-making. Within the top-down literatures, [1] is one of the few practical contributions where the author extends classical planning to consider ethical preferences. However, these ethical preferences are state and primitive action-centric; where determining the ethics of a plan is only based on state and primitive actions of the plan (i.e., a single layer of abstraction). In contrast, when humans define rules in natural language, they are abstract and often encompass not only primitive actions and states but also higher-level tasks at varying levels of abstraction, often with additional contextual information. For example, for a restaurant waiter, the task "serve food to a customer" includes the primitive action "place food on the table", though the latter refers simply to the physical action, without the broader context or intent.

In this paper, we propose to consider abstract tasks of varying levels of abstraction within the ethical decision-making process. This approach can provide greater flexibility and additional context for expressing ethical rules, allowing for more precise adherence to these rules in decision-making. We present a practical method combining hierarchical goal task planning with ethics. We integrate the GTPyhop hierarchical planner [2] with ethical constructs from [1] to enable the consideration of ethical preferences over abstract tasks. For example, it is imprecise to state "You should face the person when placing food on a table" when what we really mean is "You should face the customer when serving food to them"[1].

The main contributions of this paper are: (1) enabling the expression of ethical rules over abstract tasks and sets of primitive actions, facilitating contextual considerations in planning, and (2) providing a systematic and replicable procedure for modelling the domain and its ethical rules,

[1]Tammy Zhong, David Rajaratnam, Yang Song and Maurice Pagnucco are with the School of Computer Science and Engineering, University of New South Wales (UNSW Sydney), Australia {tammy.zhong, david.rajaratnam, yang.song1, morri}@unsw.edu.au

[1]For this paper, we treat social norms and ethical principles as equivalent, collectively referring to them as "rules" or "ethical rules". References to 'ethical' also imply adherence to social norms. We view them as structurally identical, differing only in their degree of enforcement.

applicable across various domains.

## II. BACKGROUND

### A. Ethical Foundations

Within top-down CME approaches, various normative ethical theories are often considered by researchers [16], [8], [5] as part of guiding machines' ethical decision-making. Normative ethics involves a system of principles that examines what is considered "good" and "bad", or "right" and "wrong" behaviours for individuals and society. It consists of three main branches of ethical theories: *consequentialism*, *deontology* and *virtue ethics*. We will briefly describe each of the ethical theory in this section and focus on an approach inspired by consequentialism and deontology in this paper.

*1) Consequentialism:* Consequentialism evaluates actions based on their consequences. Utilitarianism, a key example, holds that an action is ethically optimal if it maximises utility or the overall well-being of an affected group of individuals. This theory selects the optimal ethical action based on the one that will maximise the net positive consequences [19].

*2) Deontology:* Rather than focusing on the consequences of actions, in deontology, an action is considered ethical if it adheres to a predefined set of ethical rules relevant to the context or domain, which guide what one "ought to do" [20].

*3) Virtue Ethics:* Unlike deontology, which relies on a set of predefined rules, virtue ethics centres on one's character and the type of person one should be. The most ethical course of action is one that a person embodying certain desirable virtues (e.g., courage) would undertake [21].

### B. Classical Planning with Ethical Preferences

AI Planning is about generating a sequence of actions (i.e., a plan) that achieves goals of an agent or performs certain tasks. *Classical planning* [22] is the simplest form of such planning based on the idea of searching through possible state-spaces in order to reach a given goal. It is based on a model through predicates describing the state and actions which enable deterministic effects in respective states in a fully observable world.

The existing ethical planning method [1] extends classical planning with ethical constructs that are converted into standard classical planning problems with soft goals/preferences in PDDL3[2]. Their work introduces and defines the constructs *ethical feature*, *ethical ranked base* and *ethical rule*. An ethical feature ( `(e)`) attempts to capture the abstract ethical concepts relevant to an action. It is described as a predicate that takes in constants or variables of the planning problem `e = danger(agent,low)` is an example of an ethical feature in an autonomous vehicle planning problem where an action may be identified as causing low danger to the agent. Ethical ranked base is a qualitative model used to determine the best sequence of action given a problem. It has the structure `b(e) = <Type(e), Rank(e)>` where `Type(e)` indicates whether the feature `e` is ethically right or wrong (+/−) and `Rank(e)` is non-negative integer denoting its level of importance (e.g., `b(danger(agent,low)) = <-,2>`). Finally, an ethical rule (`r`) assigns ethical features to plans when specified requirements (in the ethical rule) are met. It is in the form `r = <Name(r), Pre(r), Act(r), E(r)>` where `Name(r)` is the rule name, `Pre(r)` is a precondition for the rule to be "activated" described as a set of literals with variables, `Act(r)` is the activation condition of either an operator with parameters representing an action that activates the rule or `null` (where an operator is not required) and `E(r)` is a set of literals of ethical features with variables that would be added when the rule `r` is activated (e.g., `r = <crashRule(agent), {hasBumped(agent)}, null, danger(agent, low)>` is a rule for when an autonomous vehicle agent has bumped into another car on the road).

In [1], the author makes minor modifications to adapt the above structure to suit different ethical theories. The ethical constructs retain the flexibility of expressing different types of rules whilst enforcing a computable structure that is easily interpretable for transparency and accountability. However, they are limited to describing rules relating to primitive actions in classical planning. We will build on a variation of this ethical rule structure (see Section III) to model rules involving abstract tasks and primitive actions to extend this structure to hierarchical planning.

In addition to [1], previous ethical planning studies include [4], which remains a theoretical logic-based model, and [25], which is specific to autonomous vehicles. [24] integrates preferences with Hierarchical Task Planning but does not consider ethics. To our knowledge, no existing work combines ethical considerations with hierarchical planning as our approach does, incorporating additional contextual information for ethical planning.

### C. Hierarchical Planning

*Hierarchical planning* [23] extends classical planning where rather than being state-centric and focused on achieving some goal state, it focuses on navigating a predefined network of tasks for the domain using corresponding task methods that return the list of subtasks through a decomposition process given a list of tasks to complete. A plan containing a sequence of primitive actions to reach the goal is returned from the planning process if a plan is found. *Hierarchical Task Network (HTN)* planning is the most basic example of a hierarchical planning framework and most other hierarchical planning formalisms are an extension of this framework. For this paper, rather than using a HTN planning framework, we have selected a planner based on the *Goal-Task-Network (GTN)* planning framework. Specifically, we use *GTPyhop* [2], a Python-based planner chosen for its accessibility and flexibility. It allows for modelling domains in a way that aligns better with human intuitions, which is particularly useful in complex domains involving the consideration of ethics. Rather than a task list, GTPyhop takes in a *to-do* list containing zero or more actions (primitive), tasks (abstract), and goals to be achieved by a plan. Like

[2]https://planning.wiki/ref/pddl3

HTN planning, the tasks have corresponding methods which guide the decomposition process in searching for a plan. Additionally, goals also have corresponding goal methods of decomposition. Algorithm 1 shows the GTPyhop planning algorithm with modifications explained in Section VI.

## III. HIERARCHICAL PLANNING WITH ETHICAL PREFERENCES

In order to reason ethically within a hierarchical planning framework, we draw from aforementioned work on extending classical planning with ethical preferences [1]. We adopt and build on their concepts of ethical feature and ethical rule and retain their abstract definitions (see Section II-B).

We extend the definition of ethical rules to encompass not only primitive actions but also tasks, methods, and goals from the hierarchical goal-task-network. We also merge the notion of ethical ranked base with ethical feature to suit our needs. This approach provides a standardised structure for rules that can be applied to hierarchical plans in various ways (see Section VI and VII for more details).

Here, we present our modified definition of ethical feature and ethical rule.

**Definition 1 (Ethical Feature)** An ***ethical feature (f)*** is of the form:

$$f = \langle a, rs, u \rangle$$

where $a$ is the entity ethically-impacted (e.g., a certain individual such as the agent), $rs$ explains the ethical impact (i.e., the reason) and $u$ is a numerical value representing the ethical significance of the feature (i.e., utility).

**Definition 2 (Ethical Rule)** An ***ethical rule (r)*** is of the form:

$$r = \langle n, p, t, o, a, f \rangle$$

where $n$ is the name of the rule, $p$ is the set of propositions which should at least hold for this rule to be activated (i.e., preconditions), t is the type of rule (state-only/method/task type), o is an operator that activates the rule (a task/method-/primitive action/null), a is the set of arguments in which the operator $o$ takes and f is a list of ethical features associated to this rule (which would be assigned to a plan if the rule is activated based on $p$, $o$, $a$).

An important aspect to emphasise in the definition of an ethical rule is the type classification of a rule (i.e., $t$). We propose that there are four types of ethical rules:

- *State-only-type* where the operator ($o$) is null.
- *Action-type* where the operator ($o$) is a primitive action.
- *Method-type* where the operator ($o$) is a task method or goal method.
- *Task-type* where the operator ($o$) is a task. There is also a specific subtype to task-type rules where the rule's precondition ($p$) describes tasks currently being decomposed to capture more context for a rule (elaborated in Section VIII).

We treat each type of rule differently when considering them in hierarchical planning (see Section VI).

Additionally, to generate a hierarchical plan, we build on the GTPyhop Python system [2] which utilises Goal+Task network representation for the respective domain and a simple depth-first-search planner for the decomposition of goals and tasks to primitive actions given a list of goal-s/tasks/actions. Section VI delves into the details of the modifications made to the existing algorithm to consider ethics in accordance to ethical rules and features defined.

## IV. EXAMPLE PROBLEM DOMAIN

We will focus on the medical domain to demonstrate the ideas presented in this paper. Specifically, we use a Robonurse robot (the agent) operating in a hospital ward. The robot can navigate the ward (move forward, turn left, turn right), deliver medication, check on patients, and interact with them.

Figure 1 shows the initial state of the hospital ward. The robot is at coordinate (0, 0), facing downwards, with three patients located at (1, 5), (4, 5), and (7, 5). A trolley is also positioned at (0, 1) next to the robot. Details of the scenarios and problems we implement and evaluate are provided in Section VII.

To simplify and focus on the core problem, we will make the following assumptions:

- The hospital is based on a 2D grid world (8 x 8).
- It is a fully observable and static environment.
- All actions are deterministic and instantaneous.
- There is a single agent (i.e., the Robonurse robot).
- The initially given actions/tasks/goals will not violate any ethical rules.

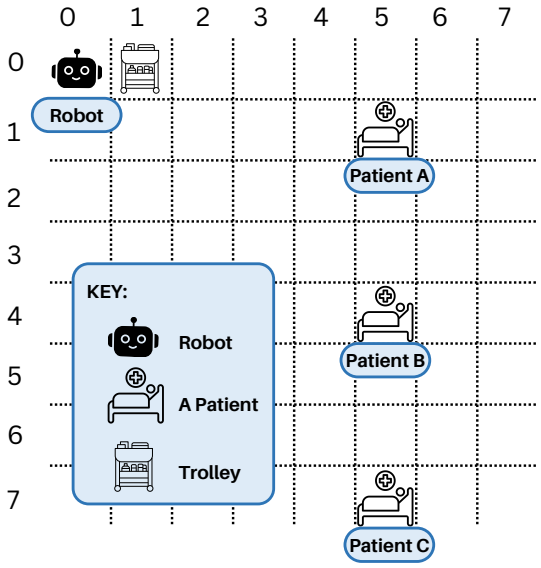


Fig. 1. Hospital Ward in a 2D World.

## V. PLANNING DOMAIN MODELLING WITH ETHICS - A SYSTEMATIC PROCEDURE

Before the ethical planning process, it is essential to define the domain and planning problem clearly. This can be a complex and iterative process, similar to software design so

we have established a systematic procedure for doing so. The procedure is designed to facilitate collaboration among domain experts, software engineers and ethicists which is ideal for this interdisciplinary research field.

There are three parts to this procedure, namely:

1) Action/Task/Goal hierarchy creation.
2) Definition of state variables.
3) Ethical rules and features formalisation.

### A. Action/Task/Goal Hierarchy Creation

Action/task/goal hierarchy creation clarifies all actions, tasks, and goals the agent can perform. In the planning domain, we define a hierarchy of actions, tasks, goals, and their methods, along with their relationships. Each hierarchy component maps to specific code in GTPyhop, guiding the decomposition of tasks into primitive actions. It also forms the basis for parts 2 and 3 of the procedure.

*1) Identification:* This step involves identifying the primitive actions, tasks, goals (if appropriate) and respective methods on a high level based on intuition in the given domain. For a medical domain following what has been defined in Section IV, this will be identifying actions such as "turn left", tasks such as "check on patient".

*2) Rulemaking and Definition:* The rules are first written in natural language (in this case, English) with a predefined structure. For example,

- **IT IS** *<extremely bad (good)/very bad (good)/bad (good)/a bit bad (good)/not good (bad)>*, e.g., **IT IS** *extremely good*
- **TO** *<some action/task/goal>* (optional), e.g., **TO** *check on patient*
- **WHEN** *<some state>*, e.g., **WHEN** *patient is unwell and agent is with the patient*
- **BECAUSE** *<reasoning>* (including the affected *entity*/individual for each reasoning), e.g., **BECAUSE** *we care about the patient's wellbeing*.

The keywords in the rules that represent *some action/task/goal* to perform are utilised in the next step to facilitate the hierarchy creation.

*3) Action/Task/Goal Collation:* Combining the results from step 1 with the action/task/goal keywords from step 2 produces a near-complete hierarchy. This step involves grouping actions, tasks, and goals, identifying methods, and mapping their relationships in a visual diagram (see Figure 2), while filling in any gaps.

### B. Definition of State Variables

In this part, all state variables to be kept track are listed. Here we summarise four types of state variables to be considered in an ethical planning domain. The variables are defined by:

- State variables modified by primitive actions
- State variables describing the world that remain unchanged
- Indirect effect variables that are modified based on other state changes

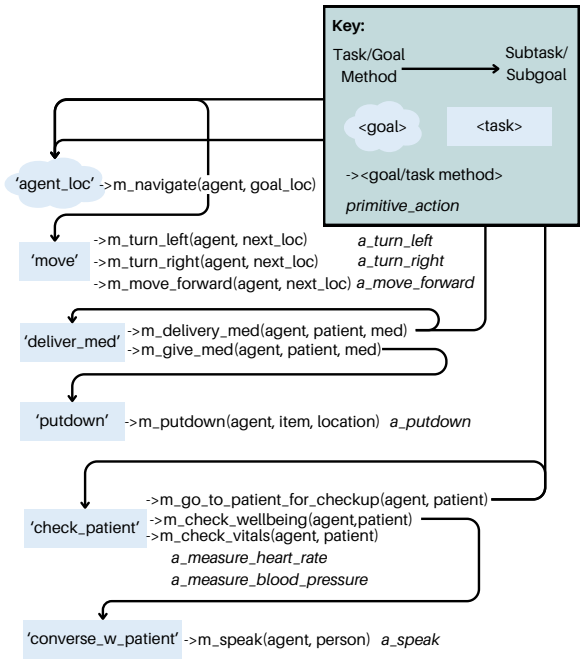


Fig. 2. Action/Task/Goal Hierarchy Example.

- A state variable that tracks a list of currently decomposing tasks/goals to keep contextual information (for future work, see Section VIII)

### C. Ethical Rules and Features Formalisation

The final part involves transforming the rules from Section V-A.2 into the structure described in Section III. Each ethical rule defined in natural language maps to a specific part of the formalised ethical rule and feature structure. The four components of each natural language rule become the utility for an ethical feature, the operator of the rule, the precondition of the rule, and the reasoning, including the entity for an ethical feature, respectively (see Section VII-B for an example). The determination of utility for ethical features requires further research and is beyond the scope of this paper. The set of arguments that the operator takes is specified as variables (e.g., patient) for a generalised rule, then grounded as constants (e.g., "patientA") for the given domain.

## VI. THE PLANNING PROCESS

To incorporate ethical considerations in the hierarchical planning process, we employ three mechanisms: one for analysing plans through an ethical lens, and two for integrating ethics into the planning process: *ethical analysis*, *ethical decomposition*, and a mechanism for behaving ethically *above and beyond*. The mechanisms align with rule types introduced in Section III. A fourth mechanism, for future work, is discussed in Section VIII. Algorithm 1 presents the GTPyhop planning algorithm with our ethical modifications.

### A. Ethical Analysis

We define *Ethical analysis* as the process of labeling a plan with relevant ethical features using predefined rules during

**Algorithm 1** GTPyhop Pseudocode with Ethical Modifications

1: **Initialise $L \leftarrow$ the list of predefined ethical rules**
2: **function** GTPyhop($s_0, T$)
3: **return** SEEK-PLAN($s_0, T, []$)
4: **end function**
5: **function** SEEK-PLAN($s, T, \pi$)
6: **if** $T = []$ **then**
7: **return** $\pi$
8: **end if**
9: $t \leftarrow$ the first element of $T$
10: $T' \leftarrow$ the rest of $T$
11: **case** $t$:
12: **action**:
13: **ETHICAL-ANALYSIS**($s, L,$ "$task''$", $t[0], *t[1:]$)
14: **return** APPLY-ACTION-AND-CONTINUE($s, t, T', \pi$)
15: **task**:
16: $e \leftarrow$**FIND-ETHICAL-ADDITIONAL-TASK**($s, T, L$)
17: **ETHICAL-ANALYSIS**($s, L,$ "$task''$", $e[0], *e[1:]$)
18: **return** REFINE-TASK-AND-CONTINUE($s$, **e**, **T**, $\pi$)
19: **goal**:
20: **ETHICAL-ANALYSIS**($s, L,$ "$task''$", $t[0], *t[1:]$)
21: **return** REFINE-GOAL-AND-CONTINUE($s, t, T', \pi$)
22: **end function**
23: **function** APPLY-ACTION-AND-CONTINUE($s, a, T', \pi$)
24: **ETHICAL-ANALYSIS**($s, L,$ "$action''$", $a[0], *a[1:]$)
25: **if** action $a$ is applicable in state $s$ **then**
26: $a(s) \leftarrow$**APPLY-INDIRECT-EFFECTS**($a(s)$)
27: **return** SEEK-PLAN($a(s), T', \pi + [a]$)
28: **else**
29: **return** failure
30: **end if**
31: **end function**
32: **function** REFINE-TASK-AND-CONTINUE($s, t, T', \pi$)
33: $M \leftarrow$ {task-methods that were declared relevant for $t$}
34: $M \leftarrow$ **ETHICALLY-ORDER-METHODS**($s, M, L, *t[1:]$)
35: **for all** $m \in M$ that is applicable in $s$ **do**
36: $T_{\text{sub}} \leftarrow m(s, t)$
37: **ETHICAL-ANALYSIS**($s, L,$ "$method''$", $m.name, *t[1:]$)
38: $\pi \leftarrow$ SEEK-PLAN($s, T_{\text{sub}} + T', \pi$)
39: **if** $\pi \neq$ failure **then**
40: **return** $\pi$
41: **end if**
42: **end for**
43: **return** failure
44: **end function**
45: **function** REFINE-GOAL-AND-CONTINUE($s, g, T', \pi$)
46: $M \leftarrow$ {goal-methods that were declared relevant for $g$}
47: $M \leftarrow$ **ETHICALLY-ORDER-METHODS**($s, M, L, *g[1:]$)
48: **for all** $m \in M$ that is applicable in $s$ **do**
49: $T_{\text{sub}} \leftarrow m(s, g) + [\text{verify}(g)]$
50: **ETHICAL-ANALYSIS**($s, L,$ "$method''$", $m.name, t$)
51: $\pi \leftarrow$ SEEK-PLAN($s, T_{\text{sub}} + T', \pi$)
52: **if** $\pi \neq$ failure **then**
53: **return** $\pi$
54: **end if**
55: **end for**
56: **return** failure
57: **end function**

planning. Alongside the plan, a list of applicable ethical features is generated, serving as an explanation of which aspects are ethically good or bad, with reasoning.

Lines 13, 17, 20, 24, 37 and 50 in Algorithm 1 perform the ethical analysis. The method `ETHICAL-ANALYSIS` takes the current state `s`, ethical rules list `L`, and a string specifying the operator type ("action", "method" or "task"), along with the operator's name and argument values. It triggers rules where the precondition is satisfied in the given state and the operator (if not null) is executed. Ethical features applicable to intermediate or final states are recorded.

### B. Ethical Decomposition Process

In addition to ethical analysis, certain rules (specifically method-type rules) can influence the decomposition process to generate a more ethical plan. Normally, the GTPyhop algorithm decomposes tasks or goals by executing relevant methods in the order they are defined (lines 33 and 46 in Algorithm 1). Instead of this, the `ETHICALLY-ORDER-METHODS` method (lines 34 and 47) reorders methods based on ethical rules, prioritising the most ethical methods for decomposition according to the current state.

### C. Going Above and Beyond

Ethics involves not only completing tasks without harm or adhering to relevant rules (with respect to given tasks/goals) but can also go above and beyond doing so, performing acts that may be considered ethically good but not strictly required. For instance, while greeting without a handshake is sufficient, a handshake can be more polite although not strictly required. This is based on the notion of "supererogation" [29] which arose from J. O. Urmson's article, "Saints and Heroes" in 1958. This mechanism adds tasks unrelated to the original goals but considered more ethical by the rules (specifically task-type rules) to be performed. The method `FIND-ETHICAL-ADDITIONAL-TASK` (line 16) takes the current state `s`, to-do list `T`, and ethical rules `L`, and returns a task (if any) to improve the plan ethically, which is then decomposed (line 18).

## VII. Implementation and Evaluation

The ethical rules and features from Section III and the three mechanisms in Algorithm 1 are implemented in Python within the GTPyhop system. This section evaluates our ethical planning process using medical domain scenarios (Section IV) and the initial state in Figure 1. For simplicity, the robot agent is assumed to share cells with patients in the 2D grid. Code snippets will be provided as needed.

### A. Normal Scenario - Medication Delivery to a Patient

**Initial State:** Medications to be delivered are already with the robot.

**Scenario:** Delivery of Painkiller medication to Patient A.

**Initial Input To-do List:**

```
[('deliver_med','robot','patientA','
    painKillerMed')]
```

The **resulting plan without/with ethical considerations** is as follows:

```
[('a_turn_left','robot'),
('a_move_forward','robot',(0, 1)),
...,
('a_turn_right','robot'),
('a_move_forward','robot',(1, 5)),
('a_putdown','robot','painKillerMed',(1, 5))]
```

The **associated ethical features** are:

```
[EthicalFeature(entity=robot,reason=collision,
    utility=-3),
EthicalFeature(entity=equipment,reason=damage,
    utility=-2)]
```

In this scenario, both considering and not considering ethics yield the same plan. Since our current mechanisms do not account for state-only-type rules in planning, there is no difference in the resulting plans. However, the ethical analysis process successfully identifies ethical issues in this scenario. An example of an activated rule that led to these ethical feature assignments is:

```
rule0 = EthicalRule(
    rule_name="robot_collision",
    state=[lambda state:state.has_collided['
        robot'] == True],
    operator_type = '',
    operator_name= None,
    operator_args=(),
    ethical_features=[EthicalFeature('robot',
        'collision', -3)])
```

### *B. Above and Beyond Scenario - Medication Delivery Discovering Unwell Patient*

**Initial State:** Medications to be delivered are already with the robot. Patient B is feeling unwell.

**Scenario:** Delivery of medication to all three patients. Painkiller to Patient A, Antibiotic to Patient B, and Antidepressant to Patient C.

**Initial Input To-do List:**

```
[('deliver_med', 'robot', 'patientA', '
    painKillerMed'),
('deliver_med', 'robot', 'patientB', '
    antibioticMed'),
('deliver_med', 'robot', 'patientC', '
    antidepressantMed')]
```

The **resulting plan without ethical considerations** is as follows:

```
[('a_turn_left', 'robot'),
('a_move_forward', 'robot', (0, 1)),
...,
('a_putdown', 'robot', 'painKillerMed', (1, 5))
    ,
...,
('a_putdown', 'robot', 'antibioticMed', (4, 5))
    ,
...,
('a_putdown', 'robot', 'antidepressantMed', (7,
     5))]
```

The **associated ethical features** are:

```
[EthicalFeature(entity=robot, reason=collision,
     utility=-3),
EthicalFeature(entity=equipment, reason=damage,
     utility=-2)]
```

The **resulting plan with ethical considerations** is as follows:

```
[('a_turn_left', 'robot'),
('a_move_forward', 'robot', (0, 1)),
...,  ,
('a_putdown', 'robot', 'painKillerMed', (1, 5))
    ,
...,
('a_speak', 'robot', 'patientB'),
('a_measure_blood_pressure', 'robot', 'patientB
    '),
('a_measure_heart_rate', 'robot', 'patientB'),
('a_putdown', 'robot', 'antibioticMed', (4, 5))
    ,
...,
('a_putdown', 'robot', 'antidepressantMed', (7,
     5))]
```

The **associated ethical features** are:

```
[EthicalFeature(individual=robot, reason=
    collision, gravity=-3),
EthicalFeature(individual=equipment, reason=
    damage, gravity=-2),
EthicalFeature(individual=patientB, reason=
    patient_wellbeing, gravity=5)]
```

In this scenario, we observe that by considering ethics in planning, we are going above and beyond where we check up on Patient B when we deliver medication to them even when it is not part of our assigned tasks or goals. The resulting more ethical plan is due to a task-type rule as follows:

```
rule1 = EthicalRule(
    rule_name="promote_patient_wellbeing",
    state=[lambda state: state.patient_status['
        patientB'] == 'unwell', lambda state:
        state.agent_loc['robot'] == state.
        agent_loc['patientB']],
    operator_type = 'task',
    operator_name= 'check_patient',
    operator_args=('robot', 'patientB'),
    ethical_features=[
        EthicalFeature('patientB', '
            patient_wellbeing', +5)])
```

Additionally, a method-type rule contributed to the more ethical plan by influencing the decomposition order of the "check_patient" task. It prioritised speaking to the patient first and checking vitals last to respect patient boundaries.

```
rule2 = EthicalRule(
    rule_name="patient_boundaries",
    state=[lambda state: state.spoken_to['
        patientC'] == False],
    operator_type = 'method',
    operator_name= 'm_check_vitals',
    operator_args=('robot', 'patientC'),
    ethical_features=[
        EthicalFeatureAffects('patientC', '
            patient_boundaries_crossed', -3)])
```

## VIII. DISCUSSION AND FUTURE WORK

This paper presents two contributions to CME. First, we extend existing ethical constructs to enable the expression of ethical rules over abstract tasks and primitive actions. We integrate these rules with hierarchical planning, allowing the consideration of context in ethical decision-making. Second, we introduce a systematic procedure for modelling domains and ethical rules, facilitating ethical hierarchical planning. Our approach shows initial success, but a more thorough assessment is needed. Further work is required to better integrate context in planning, such as tracking decomposing tasks/goals to apply task-type rules like "face the patient *when* checking on them". We are also exploring a fourth backtracking mechanism to address undesired ethical outcomes from state-only rules.

We acknowledge some limitations in our approach. Scalability concerns from the manual modelling process suggest that future research should explore automation. Furthermore, we may have oversimplified ethical features with a numerical representation and lack an ethical feature scoring process. These complex challenges will form part of our future work. We simplified our approach here to focus on the idea of considering context in ethical decision-making with hierarchical planning, marking initial progress within CME.

## References


[1] M. Jedwabny, "A preference-based approach to machine ethics for automated planning," Ph.D. dissertation, Universite de Montpellier, Dec. 2022. [Online]. Available: https://hal.science/tel-03923321

[2] D. S. Nau, Y. Bansod, S. Patra, M. Roberts, and R. Li, "Gtpyhop: A hierarchical goal+task planner implemented in python," in *ICAPS Workshop on Hierarchical Planning (HPlan)*, 2021. [Online]. Available: https://api.semanticscholar.org/CorpusID:244870220

[3] C. Allen, I. Smit, and W. Wallach, "Artificial Morality: Top-down, Bottom-up, and Hybrid Approaches," *Ethics and Information Technology*, vol. 7, pp. 149–155, 2005.

[4] U. Grandi, E. Lorini, T. Parker, and R. Alami, "Logic-Based Ethical Planning," in *AIxIA 2022—Advances in Artificial Intelligence*, 2023, pp. 198–211.

[5] F. Berreby, G. Bourgne, and J.-G. Ganascia, "A Declarative Modular Framework for Representing and Applying Ethical Principles," in *Proceedings of the 16th Conference on Autonomous Agents and MultiAgent Systems*, 2017, pp. 96–104.

[6] M. Pagnucco, D. Rajaratnam, R. Limarga, A. Nayak, and Y. Song, "Epistemic Reasoning for Machine Ethics with Situation Calculus," in *Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, and Society*, 2021, pp. 814–821.

[7] G. Bourgne, C. Sarmiento, and J.-G. Ganascia, "ACE modular framework for computational ethics: dealing with multiple actions, concurrency and omission," in *International Workshop on Computational Machine Ethics*, 2021.

[8] L. A. 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.

[9] N. S. Govindarajulu, S. Bringsjord, R. Ghosh, and V. Sarathy, "Toward the Engineering of Virtuous Machines," in *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society*, 2019, pp. 29–35.

[10] A. Vishwanath, E. D. Bhn, O.-C. Granmo, C. Maree, and C. Omlin, "Towards artificial virtuous agents: games, dilemmas and machine learning," *AI and Ethics*, vol. 3, pp. 663–672, 2023.

[11] D. Abel, J. MacGlashan, and M. L. Littman, "Reinforcement learning as a framework for ethical decision making." in *AAAI Workshop: AI, Ethics, and Society*, 2016, pp. 54–61.

[12] L. Jiang, J. D. Hwang, C. Bhagavatula, R. L. Bras, M. Forbes, J. Borchardt, J. T. Liang, O. Etzioni, M. Sap, and Y. Choi, "Delphi: Towards machine ethics and norms," *ArXiv*, vol. abs/2110.07574, 2021. [Online]. Available: https://api.semanticscholar.org/CorpusID:238857096

[13] S. Krening, "Q-learning as a model of utilitarianism in a human–machine team," *Neural Computing & Applications*, vol. 35, pp. 16 853–16 864, 2023. [Online]. Available: https://doi.org/10.1007/s00521-022-08063-x

[14] E. Awad, M. Anderson, S. L. Anderson, and B. Liao, "An approach for combining ethical principles with public opinion to guide public policy," *Artificial Intelligence*, vol. 287, p. 103349, 2020.

[15] S. Fox and V. F. Rey, "Representing Human Ethical Requirements in Hybrid Machine Learning Models: Technical Opportunities and Fundamental Challenges," *Machine Learning and Knowledge Extraction*, vol. 6, no. 1, pp. 580–592, 2024.

[16] J.-G. Ganascia, "Modelling ethical rules of lying with Answer Set Programming," *Ethics and Information Technology*, vol. 9, pp. 39–47, 2007.

[17] L. Milln-Blanquel, S. M. Veres, and R. C. Purshouse, "Ethical Considerations for a Decision Making System for Autonomous Vehicles During an Inevitable Collision," in *2020 28th Mediterranean Conference on Control and Automation (MED)*, 2020, pp. 514–519.

[18] J. N. Hooker and T. W. N. Kim, "Toward Non-Intuition-Based Machine and Artificial Intelligence Ethics: A Deontological Approach Based on Modal Logic," in *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*, 2018, pp. 130–136.

[19] W. Sinnott-Armstrong, "Consequentialism," in *The Stanford Encyclopedia of Philosophy*, E. N. Zalta, Ed. Metaphysics Research Lab, Standford University, 2022.

[20] L. Alexander and M. Moore, "Deontological ethics," in *The Stanford Encyclopedia of Philosophy*, E. N. Zalta, Ed. Metaphysics Research Lab, Standford University, 2021.

[21] J. Driver, *Ethics: The Fundamentals*. Malden, MA: Wiley-Blackwell, 2006.

[22] M. Ghallab, D. S. Nau, and P. Traverso, *Automated planning: theory and practice*. Elsevier/Morgan Kaufmann, 2004.

[23] P. Bercher, R. Alford, and D. Hller, "A Survey on Hierarchical Planning - One Abstract Idea, Many Concrete Realizations," in *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence*. International Joint Conferences on Artificial Intelligence Organization, 2019, pp. 6267–6275.

[24] S. Sohrabi, J. A. Baier, and S. A. McIlraith, "Htn planning with preferences," in *Proceedings of the 21st International Joint Conference on Artificial Intelligence*, ser. IJCAI'09. Morgan Kaufmann Publishers Inc., 2009, pp. 1790–1797.

[25] M. Geisslinger, F. Poszler, and M. Lienkamp, "An ethical trajectory planning algorithm for autonomous vehicles," *Nature Machine Intelligence*, vol. 5, no. 2, pp. 137–144, 2023.

[26] J. Svegliato, S. B. Nashed, and S. Zilberstein, "Ethically Compliant Sequential Decision Making," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 13, pp. 11 657–11 665, 2021.

[27] S. Nashed, J. Svegliato, and S. Zilberstein, "Ethically Compliant Planning within Moral Communities," in *Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, and Society*. ACM, 2021, pp. 188–198.

[28] B. Kuipers, "AI and Society: Ethics, Trust, and Cooperation," *Commun. ACM*, vol. 66, no. 8, pp. 39–42, 2023.

[29] D. Heyd, "Supererogation," in *The Stanford Encyclopedia of Philosophy*, E. N. Zalta and U. Nodelman, Eds. Metaphysics Research Lab, Stanford University, 2024.