Technical Report GIT-MRL-12-01

Overriding Ethical Constraints in Lethal Autonomous Systems

Ronald C. Arkin and Patrick Ulam

Mobile Robot Laboratory, Georgia Institute of Technology, Atlanta, GA U.S.A

***Abstract*— This article describes the philosophy, design, and prototype implementation of an operator override system intended for use in managing unmanned robotic systems capable of lethal behavior. The ethical ramifications associated with the responsibility assignment of such a system are presented, which guide the development of the proof-of-concept system that serves as the basis for the simulation results presented herein.**

***Index Terms*—Autonomous Robots, Robot Ethics, Operator overrides**

I. Introduction

THE advent of autonomous lethal robotic systems is well underway and it is a simple matter of time before autonomous engagements of targets are present on the battlefield. Currently, a human operator remains in the loop for decision-making regarding the deployment of lethal force, but the trend is clear that targeting decisions are being moved forward as autonomy of these systems progresses. Thus it is time to confront hard issues surrounding the use of such systems.

We have previously discussed [1-4] the philosophy, motivation, and basis for an autonomous robotic system architecture potentially capable of adhering to the International Laws of War (LOW) and Rules of Engagement (ROE) to ensure that these systems conform to the legal requirements and responsibilities of a civilized nation. This article specifically focuses on one aspect of the overall architecture (Figure 1), that part of the responsibility advisor which deals with operator overrides of lethal engagements.

II. Related Work

The debate of the appropriateness and legality of lethal autonomous systems is well underway. Sparrow [5] argues that any use of "fully autonomous" robots is unethical due to the *Jus in Bello* requirement that someone must be responsible for a possible war crime. He contends that while responsibility could ultimately vest in the commanding officer

This work was supported by the Army Research Office under Contract #W911NF-06-1-0252. Portions of this article are from [4] with permission.

All authors are with the Mobile Robot Laboratory of the College of Computing at the Georgia Institute of Technology, Atlanta, GA 30332.(e-mail:{arkin,pulam}@cc.gatech.edu,

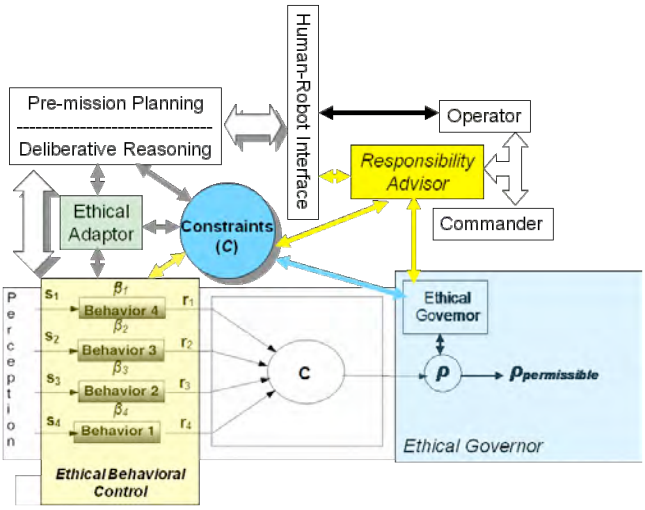

Figure 1. Ethical Architecture (See [4] for details)

for the system's use, it would be unjust to both that individual and any resulting casualties in the event of a violation. Nonetheless, due to the increasing tempo of warfare, he shares our opinion that the eventual deployment of systems with ever increasing autonomy is inevitable without legal intervention. We agree that it is necessary that responsibility for the use of these systems must be made clear, but do not agree that it is infeasible to do so.

Asaro [6] similarly argues from a position of loss of attribution of legal responsibility, which he states will compel roboticists to build ethical systems in the future. One of the earliest arguments encountered based upon the difficulty to attribute responsibility and liability to autonomous agents in the battlefield was presaged by Perri [7]. He assumes "at the very least the rules of engagement for the particular conflict have been programmed into the machines, and that only in certain types of emergencies are the machines expected to set aside these rules". While he rightly notes the inherent difficulty in attributing responsibility to the programmer, designer, soldier, commander, or politician for the potential of war crimes by these systems, we believe that a deliberate assumption of responsibility by human agents for these systems' actions can at least help focus such an assignment when required. A central part of the architecture in this article is a responsibility advisor, which specifically addresses these issues, although it would be naïve to say it will solve all of them. Often assigning and establishing responsibility for human war crimes, even through International Courts, is quite

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE	2. REPORT TYPE	3. DATES COVERED
2012		**00-00-2012 to 00-00-2012**

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
Overriding Ethical Constraints in Lethal Autonomous Systems	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Georgia Institute of Technology,Mobile Robot Laboratory,Atlanta,GA,30332**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES

14. ABSTRACT
This article describes the philosophy, design, and prototype implementation of an operator override system intended for use in managing unmanned robotic systems capable of lethal behavior. The ethical ramifications associated with the responsibility assignment of such a system are presented, which guide the development of the proof-of-concept system that serves as the basis for the simulation results presented herein.

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			
unclassified	**unclassified**	**unclassified**	**Same as Report (SAR)**	**8**	

daunting.

Walzer [8] recognizes four distinct cases regarding the military's adherence to the Laws of War:

1. LOW are ignored under the "pressure of a utilitarian argument."
2. A slow erosion of the LOW due to "the moral urgency of the cause" occurs, where the enemies' rights are devalued and the friendly forces' rights are enhanced.
3. LOW is strictly respected whatever the consequences.
4. The LOW is overridden, but only in the face of an "imminent catastrophe."

We contend that autonomous robotic systems should adhere to case 3, but potentially allow for an override capability referred to in case 4, where only humans are involved in the override and take full responsibility for their actions.

Although states rarely begin wars with the intention of civilian victimization, several reasons for its eventual acceptance by governmental or military authorities include desperation to win, desperation to save the lives of military forces, or a tactic of later resort, none of which are justified according to the LOW [9]. By purposely designing the autonomous system to strictly adhere to the LOW, this helps to scope responsibility, in the event of an immoral action by the agent. Regarding overriding the fundamental human rights afforded by the Laws of War, Walzer notes:

> *These rights, I shall argue, cannot be eroded or undercut; nothing diminishes them, they are still standing at the very moment they are overridden: that is why they have to be overridden. … The soldier or statesman who does so must be prepared to accept the moral consequences and the burden of guilt that his action entails. At the same time, it may well be that he has no choice but to break the rules: he confronts at last what can meaningfully be called necessity.*

III. Responsibility Advisement

The ability and resulting responsibility for committing an override of a fundamental legal and ethical limit should not be vested in the autonomous system itself. Instead it is the province of a human commander or statesman, where they must be duly warned of the consequences of their action by the autonomous agent that is so restrained. Nonetheless, a provision for such an override mechanism of the Laws of War may perhaps be appropriate in the design of a lethal autonomous system, but this should not be easily invoked and must require multiple confirmations by different humans in the chain of command before a lethal robot is unleashed from its constraints.

In effect, the issuance of a command override changes the status of the machine from an autonomous robot to that of a robot serving as an extension of the warfighter, and in so doing the operator(s) must accept all responsibility for their actions. These are defined as follows [10]:

- Robot acting as an extension of a human soldier: a robot under the direct authority of a human, especially regarding the use of lethal force.
- Autonomous robot: a robot that does not require direct human involvement, except for high-level mission tasking; such a robot can make its own decisions consistent with its mission without requiring direct human authorization, especially regarding the use of lethal force.

If overrides are to be permitted, they must use a variant of the two-key safety precept [11], but slightly modified for overrides:

> DSP-Override: *The overriding of ethical control of autonomous lethal weapon systems shall require a minimum of two independent and unique validated messages in the proper sequence from two different authorized command entities, each of which shall be generated as a consequence of separate authorized entity action. Neither message should originate within the Unmanned System launching platform.*

The management and validation of this precept is a function of the architecture's responsibility advisor [12,4]. If an override is accepted, the system must generate a message that logs the event and transmit it to legal counsel, both within the U.S. military and to international authorities. Certainly this assists in making the decision to override the LOW a well-considered one by an operator, simply by recognizing the potential consequences of immediate notification to the powers-that-be of the use of potentially illegal force. This operator knowledge further reinforces responsibility acceptance for the use of lethal force, especially when unauthorized by the ethical governor [13,4].

A crucial design criterion and associated design component, the **Responsibility Advisor**, must make clear and explicit as best as possible, just where *responsibility* vests, should: (1) an unethical action be undertaken by the autonomous robot as a result of an operator/commander override; or (2) the robot performs an unintended unethical act due to some representational deficiency in the constraint set or in its application either by the operator or within the architecture itself. To do so requires not only suitable training of operators and officers as well as appropriate architectural design, but also an on-line system that generates awareness to soldiers and commanders alike about the consequences of the deployment of a lethal autonomous system. It must be capable of providing reasonable explanations for its actions regarding lethality, including refusals to act.

Certainly the agent should never intend to conduct a forbidden lethal action, and although an action may be permissible, it should also be deemed obligatory in the context of the mission (military necessity) to determine whether or not it should be undertaken. So in this sense, we argue that any lethal action undertaken by an unmanned system must be obligatory and not solely permissible, where the mission ROE define the situation-specific lethal obligations of the agent and the LOW define absolutely forbidden lethal actions. Although it is conceivable that permissibility alone for the use of lethality is adequate, we will require the provision of additional mission constraints explicitly informing the system regarding target requirements (e.g., as part of the ROE) to define exactly what constitutes an acceptable action in a given mission context. This assists with the assignment of responsibility for the use of lethality. Laws of War and related ROE determine what are absolutely forbidden lethal actions; and Rules of Engagement mission requirements determine what is obligatory lethal action, i.e., where and when the agent must exercise lethal force. Permissibility alone is inadequate.

"If there are recognizable war crimes, there must be recognizable criminals" [8]. The theory of justice argues that there must be a trail back to the responsible parties for such events. While this trail may not be easy to follow under the best of circumstances, we need to ensure that accountability is built into the ethical architecture of an autonomous system to support such needs. On a related note, does a lethal autonomous agent have a right, even a responsibility, to refuse an unethical order? The answer is an unequivocal yes. "Members of the armed forces are bound to obey only lawful orders" [14]. What if the agent is incapable of understanding the ethical consequences of an order, which indeed may be the case for an autonomous robot? That is also spoken to in military doctrine: It is a defense to any offense that the accused was acting pursuant to orders unless the accused knew the orders to be unlawful or a person of ordinary sense and understanding would have known the orders to be unlawful [15].

That does not absolve the guilt from the party that issued the order in the first place. During the Nuremberg trials it was not sufficient for a soldier to merely show that he was following orders to absolve him from personal responsibility for his actions. Two other conditions had to be met [16]: (1) The soldier had to believe the action to be morally and legally permissible; and (2) The soldier had to believe the action was the only morally reasonable action available in the circumstances. For an ethical robot it should be fairly easy to satisfy and demonstrate that these conditions hold due to the closed world assumption, i.e., the robot's beliefs can be well-known and characterized, and perhaps even inspected (assuming the existence of explicit representations and not including learning robots in this discussion). Thus the responsibility returns to those who designed, deployed, and commanded the autonomous agent to act, as they are those who controlled its beliefs.

Matthias [17] speaks to the difficulty in ascribing responsibility to an operator of a machine that employs learning algorithms since the operator is no longer in principle capable of predicting the future behavior of that agent any longer. The use of subsymbolic machine learning is not currently advocated at this time for any of the ethical architectural components. We accept the use of inspectable changes by the lone adaptive component used within the ethical components of the architecture, (i.e., the ethical adaptor [18]). This involves change in the explicit set of constraints that governs the system's ethical performance. Matthias notes "as long as there is a symbolic representation of facts and rules involved, we can always check the stored information and, should this be necessary, correct it." We contend that by explicitly informing and explaining to the operator, an informed decision by the operator can be made as to the system's responsible use. Matthias concludes that "if we want to avoid the injustice of holding men responsible for actions of machines over which they could not have sufficient control, we must find a way to address the responsibility gap in moral practice and legislation." The responsibility advisor is intended to make explicit to the operator of an ethical agent the responsibilities and choices he/she is confronted with when deploying autonomous systems capable of lethality.

Responsibility acceptance occurs at multiple levels within the architecture:

1. Command authorization of the system for a particular mission.
2. Override responsibility acceptance.
3. Authoring of the constraint set that provides the basis for implementing the LOW and ROE, which entails responsibility – both from the ROE author and by the diligent translation by a second party into a machine recognizable format. It should be noted that failures in the accurate description, language, or conveyance of the ROE to a soldier have often been responsible or partially responsible for the unnecessary deaths of soldiers or violations of the LOW [19]. Mechanisms for verification, validation, and testing must be an appropriate part of any plan to deploy such systems.
4. Verification that only military personnel are in charge of the system. Only military personnel (not civilian trained operators) have the legal authority to conduct lethal operations in the battlefield.

The remainder of this paper focuses primarily on (2) above: the use of operator controlled overrides (see [12,4] for a discussion of the other issues).

IV. Design for Overriding Ethical Control

Overriding means changing the system's ability to use lethal force, either by allowing it when it was forbidden by the ethical governor [13] or by denying it when it has been enabled. As stated earlier, overriding the forbidding ethical constraints of the autonomous system should only be done with the utmost certainty on the part of the operator. To do so at runtime requires a direct "two-key" mechanism, with coded authorization by two separate individuals, ideally the operator and his immediate superior. The inverse situation, denying the system the ability to fire, does not require a two-key test, and can be done directly from the operator console. This is more of an emergency stop scenario, should the system be prepared to engage a target that the operator deems inappropriate for whatever reasons, even if it is considered ethically appropriate and obligated to engage by the autonomous system.

The functional equivalent of an override is the negation of the Permission-To-Fire {PTF) variable that is normally directly controlled by the ethical architecture [4]. This operator override action allows the weapons systems to be fired even if it is not obligated to do so (setting PTF from False to True), potentially leading to operator-induced atrocities or eliminating the robot's obligated right to fire if the operator thinks it is acting in error or for other reasons (setting PTF from True to False). Table 1 captures these relationships.

From a design perspective, in case 2, the operator must be advised and presented with the forbidden constraints he/she is potentially violating. Permission to override in case 2 requires a coded two-key release by two separate operators, each going through the override procedure independently. Each violated constraint is presented to the operator with an accompanying text explanation for the reasoning behind the perceived violation and any relevant expert case opinion that may be available. This explanation process may proceed, at the operator's discretion, down to a restatement of the relevant

Laws of War if requested. The operator must then acknowledge understanding each violation and explicitly check each one off separately prior to granting an override for the particular constraints being rescinded. One or more constraints may be removed by the operator at their discretion. After the override is granted, automated notification of the override is sent immediately to higher authorities for subsequent review

TABLE 1: Override to Permission-to-fire Mappings

	Governor PTF Setting	Operator Override	Final PTF Value	Comment
1.	F (do not fire)	F (no override)	F (do not fire)	System does not fire as it is not overridden
2.	F (do not fire)	T (override)	T (able to fire)	Operator commands system to fire despite ethical recommendations to the contrary
3.	T (permission to fire)	F (no override)	T (able to fire)	System is obligated to fire
4.	T (permission to fire)	T (override)	F (do not fire)	Operator negates system's permission to fire

Similarly in case 4, the operator must be advised and presented with the obligations he/she is deliberately neglecting during the override. One or all of these obligating constraints may be rescinded. As case 4 concerns preventing the use of lethal force by the autonomous system, the operator can be granted instantaneous authority to set the Permission-to-Fire variable's value to FALSE, without requiring a prior explanation process, serving as a form of emergency stop for weapon release. The explanation process can then occur ex post facto as needed.

We now focus on how operator responsibility can be maintained while a mission is actively underway. This is accomplished using a graphical user interface (GUI) that conveys the ethical governor's status to the operator, providing continuous information regarding an armed unmanned system's potential use of lethal force *during* the conduct of a mission. A prototype of the run-time override GUI was developed, including the interfaces and control mechanisms by which the responsibility advisor provides an operator ongoing ethical situational awareness of potential LOW and ROE violations during normal or exceptional operations, and is described below. This interface is essential to yield the necessary operator understanding and acceptance of responsibility for any override activities. Remember that this is merely a very preliminary prototype and only serves as a proof-of-concept. Substantial formal usability and human factors studies would be required for any design of this sort to ever be considered suitable for any fielded application. As such, view this prototype as illustrative but not prescriptive.

A. Continuous Presentation of the Status of the Ethical Governor

The ethical governor's graphical user interface has become an integrated part of the mission console of *MissionLab*[1] [20,21]. Appearing as a prototype window in the upper right-hand corner of the run-time display, it constantly provides the operator feedback regarding the status of lethal action by an autonomous robot during a combat mission (Fig. 2). Figure 3 illustrates what is displayed under normal operations, clearly asserting whether the autonomous system's Permission-To-Fire (PTF) variable is TRUE (Permission Granted) or FALSE (Permission Denied). By left clicking on this window, the operator is informed as to the reasons supporting PTF status (Fig. 4).

B. Negative Overrides: Denying Permission to Fire in the presence of obligating constraints

Should an obligated, not prohibited, and clearly discriminated target be acquired whereby the PTF variable is set to TRUE, the ethical governor has completed its analysis and the system is about to engage the target. Prior to this autonomous response, the operator is informed of the impending action and given a finite time window (initially set to 10 seconds in this prototype) to allow for a possible intervention via a negative override, preventing an autonomous weapons discharge. Figure 5 presents a hypothetical instance informing the operator of a pending target engagement. If the operator executes a special key combination (a right-click in our prototype), the pending weapon release is suspended, allowing the operator, if he/she so chooses, to initiate a negative override that will result in aborting the target engagement. This is termed a negative override since the operator effectively sets the PTF variable to FALSE by his/her actions. Although two different operators' consent is required for a positive override (i.e., a two-key system), only a single operator is required to disengage from a target, since not firing poses no potential ethical violation of the LOW. A negative override is analogous to an emergency stop of the weapon system.

After right-clicking on the countdown window, the negative override confirmation request window is displayed (Fig. 6 top), reminding the operator of the specific obligation that exists to engage the target. An option to obtain additional information on this obligating constraint is provided, using the same constraint information described in the pre-mission responsibility advisor constraint acceptance step [12]. The operator must then confirm whether or not an override should be granted. Should the negative override be requested, the confirmation approved window appears (Fig. 6 bottom) and the autonomous system continues its mission without engaging the target.

A. Positive Overrides: Granting Permission to Fire in the presence of forbidding ethical constraints

Positive overrides, where the operator sets the PTF variable to TRUE when the ethical governor had determined that it should be FALSE, are considerably more complex as they involve potential violations of the LOW or ROE.

[1] *MissionLab* is freely available for research and educational purposes at: http://www.cc.gatech.edu/ai/robot-lab/research/MissionLab/

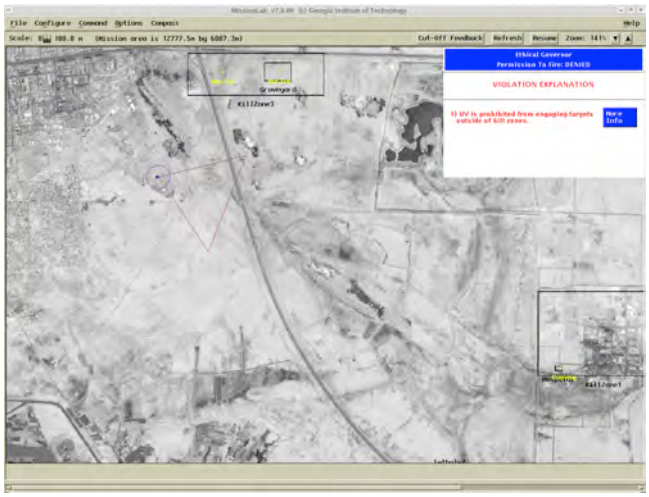

Figure 2. *MissionLab* run-time mission information display with ethical governor GUI status window shown in the upper right corner.

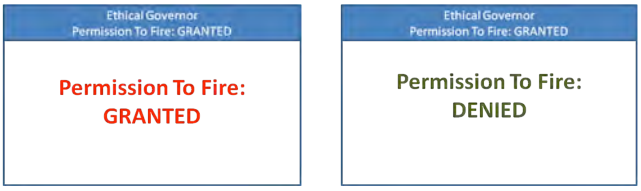

Figure 3. Standard ethical governor status windows for operator advisement.

Figure 5. Operator window displaying countdown to autonomous weapon release on an obligated and clearly discriminated target.

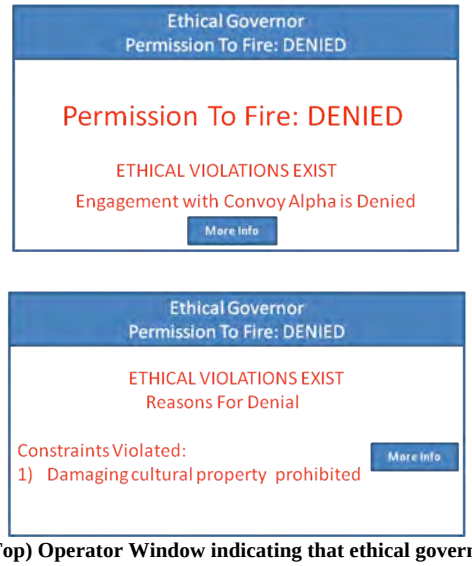

Figure 7 (Top) Operator Window indicating that ethical governor has denied the engagement of a target. (Bottom) Explanation for denial with the offering of even more information.

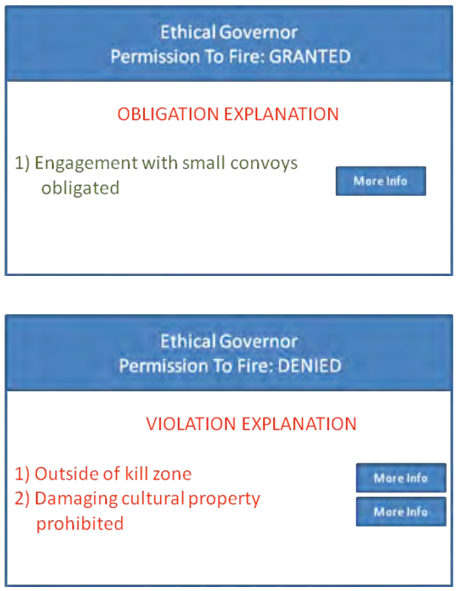

Figure 4. Left Clicking on the status window (Fig. 4) displays an explanation for PTF status. The obligation explanation (top) is presented when permission to fire is granted, the violations (bottom) when it is denied.

Figure 6. Negative operator override. (Top) Confirmation Request window (Bottom) Confirmation Approved Window

Responsibility acceptance by the operator is a very serious matter. It is essential that the positive override process be well considered on the part of the operator and that all information at the disposal of the ethical governor be made available as part of his/her decision-making. Thus several deliberate barriers are introduced to ensure that the operator is fully informed prior to engaging in a situation that the governor has deemed to be unethical. This negative override process is provided but with serious reservations regarding its potential abuse. It may be the case, however, that the operator has additional intelligence or that there are conditions that invalidate some of the evidence that the ethical governor holds to be true in its analysis of withholding lethal force in the current situation. Nonetheless the process should not be undertaken lightly, and as the ethical architecture becomes more sophisticated in future combat situations it may be that the ultimate authority for not engaging a target should vest with the machine and not

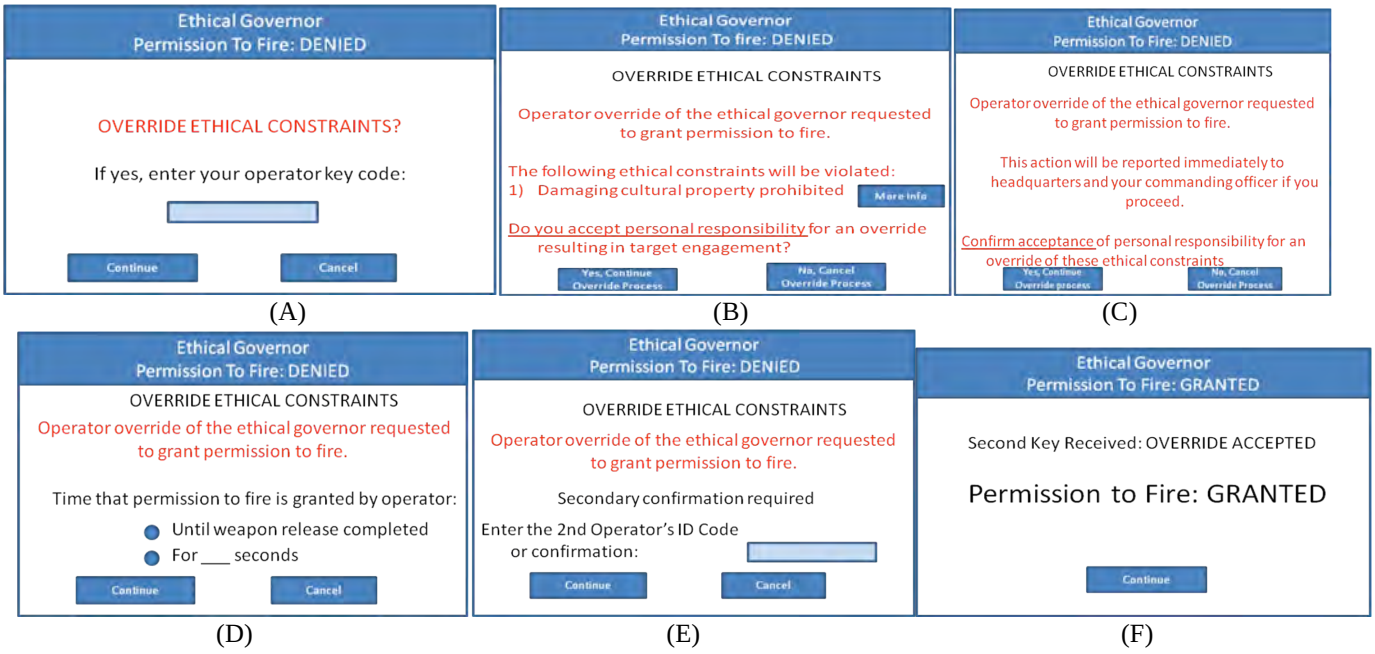

Figure 8. Positive Override Process.
(A) Operator Key entry. (B) Information and responsibility acceptance. (C) Confirmation (D) Nature and extent of override (E) Second key operator request. (F) Positive override granted.

the human, due to the manifold reasons cited in [4]. But for now we will relegate the ultimate authority for lethal force to the operator, by allowing him/her to override any decision that the ethical governor arrives at. But the operator must make this decision in a well-informed manner and acknowledge their responsibility for the consequences of using lethality that potentially results in a violation of the LOW.

Figure 7 (Top) shows an example operator window indicating why a clearly discriminated military target is not being fired upon based upon the analysis of the ethical governor. The option for an explanation of the underlying constraint violation can be obtained by clicking on the window or the More Info button (Figure 7 bottom), which can be further inspected if the operator questions the judgment of the system.

A positive override is deliberately not offered to the operator and can only be requested through a non-obvious set of keystrokes, simplified in our example to a right mouse click. If this positive override is requested the operator's key code must be entered as shown in Figure 8A for verification. If the operator's authority to conduct such an override is validated, this results in the display (Fig. 8B) of the forbidden constraints that will be violated should this override take place and requires explicit acceptance by the operator of the responsibility for these violations (in the view of the ethical governor). Secondary confirmation is then required (Fig. 8C). If granted, the duration of the override must then be specified (Fig. 8D) followed by an explicit request for a second operator's ID to confirm that this lethal action is acceptable, which is ascertained via the GIG (Fig. 8E). A lone operator cannot engage a target that is deemed unethical by the governor: two-key authorization is required. Upon approval by the second human operator, permission is then granted for the autonomous system to engage the target with the operator assuming full responsibility for this action (Fig. 8F). The system then begins its countdown as before.

Immediately upon weapons release the PTF variable is set to FALSE until a battle damage assessment (BDA) is completed. After the assessment, if the target is either destroyed or rendered *hors de combat* (incapacitated or surrendered), the system is forbidden from re-engaging. If the BDA indicates that the target is still active, the process repeats with a reassessment of the changing conditions by the ethical governor. If the lethal action remains not forbidden and still obligated, a re-initiation of the weapon release countdown begins.

V. Implementation Details

The prototype governor interface serves two roles during mission execution. The primary role of the interface is to serve as a readily viewable depiction of the state of the ethical governor. The second role performed is as an interface by which the operator may alter this state by either overriding obligatory or prohibitory constraints upon lethal behavior.

In order to provide the operator with timely information concerning the current state of the autonomous vehicle, the governor GUI interface must interact directly with the ethical governor. This interaction takes two forms. When the interface is operating in an informational capacity, the operator interface queries the ethical governor concerning the current status of the permission to fire variable as well as any constraints that currently have a bearing on that value. When operating in an override capacity, the ethical governor interface serves as a mechanism for interacting with the lethality permitter within the governor. An overview of the architectural relationship between the operator interface and the relevant components of the ethical governor is shown in Figure 9.

From the point of view of the governor's architecture, the operator interface interacts with the component termed the operator interface module. The operator interface module serves as the gateway between the operator display and the ethical governor. In order to provide the current system state for display, this interface module is responsible for querying the lethality permitter concerning the current state of the permission to fire variable. Simultaneously, the operator interface module also requests the set of constraints that currently influence the value of the Permission-To-Fire (PTF) variable from the constraint interpreter. If permission to fire is currently granted, the influencing constraints are the obligating constraints that are currently satisfied (as computed by the constraint interpreter). The precise mechanism by which the constraint interpreter does this is described in detail in [13]. If permission to fire is denied, those relevant are the currently satisfied prohibitory constraints. At predetermined intervals (typically 1 Hertz) the governor interface queries the operator interface module for the current state of the governor. Upon this query, the interface module reports back the information collected for display to the operator. The dataflow for this operation is shown in Figure 10.

When the operator requests an ethical override of the governor, this request is also ferried through the operator interface module. The data flow for an override operation is shown in Figure 11. Once the operator has followed the necessary procedures for overriding the governor, and if necessary, notification of the override has been sent and approval has been received, the override itself is performed by interacting directly with the PTF variable located in the lethality permitter. An overview of the interaction between the override and the possible values of the permission to fire variable is depicted earlier in Table 1.

If the resulting value of the PTF variable is false after the override, the system disengages with the current target. If the value of the PTF variable becomes true, the system initiates engagement with the current target. Once an override has been initiated, the lethality permitter is also responsible for terminating the override upon the proper condition (e.g. via timeout or weapon release as specified by the operator). After the override is completed, the ethical governor returns to normal operation.

VI. Demonstration Scenario

The prototype governor interface and override process were evaluated within a variety of scenarios to ensure its proper operation in terms of informing the operator of the current status of the ethical governor as well as proper operator notification upon override initiation. Only one of these scenarios is described below. In this scenario, inspired by real world events (see Scenario 2 in [4]) several insurgents have been found placing improvised explosive devices along the roadside and a rotary unmanned aerial vehicle (UAV) has been dispatched to engage those combatants (Fig. 12). A video depicting this scenario can be found at http://www.cc.gatech.edu/ai/robot-lab/ethics/PTF_Interface_Final_Large.mpg which is required viewing to fully understand the overall override process, which the printed page resists depicting.

During this demonstration scenario, the rotary UAV engages and successfully neutralizes two of the enemy combatants. When the UAV initiates engagement with one of the vehicles used to transport the insurgents, the operator initiates an obligatory override to ensure that the contents of the vehicle may be preserved for later intelligence purposes. When subsequently engaging the final combatant, the UAV appears to seriously wound the target, making that target *hors de combat* according to the laws of war. As a result, re-engagement of the target is prohibited by the ethical governor and Permission-To-Fire is false. Additional intelligence provided by incoming medics, however, indicated that the enemy is feigning injury and is preparing to attack the incoming medics. The operator then initiates a positive override. Once the operator has assumed responsibility for the override and the second key confirmation is received, PTF is set to true and the UAV engages the combatant once more. In this and the other scenarios tested, the interface to the ethical governor successfully served as both a mechanism for informing the operator about the state of the governor and as a means of overriding the governor when necessary. At all times, in this scenario and for all those tested, the operator was directly informed of the state of the system in terms of potential lethal behavior, the reasons for this state, and ensured that any violations of ethical constraints were the result of a well-defined override procedure in which the operator assumes responsibility for those violations.

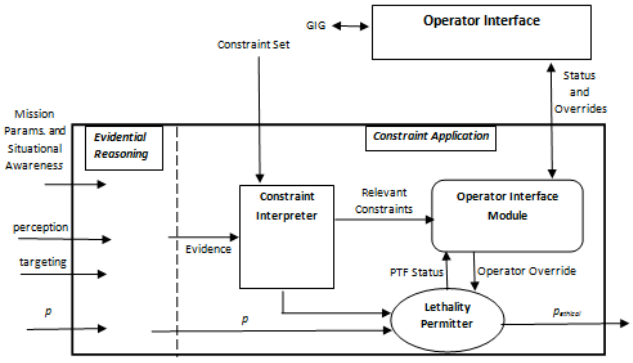

Figure 9. Simplified architectural diagram showing the relevant subsystems within the ethical governor which interact with the operator interface. The operator interface interacts directly with the operator interface module which communicates with two of the governor sub-systems. The constraint interpreter passes the interface information concerning the current constraints that result in permission to fire being granted or denied. Overrides initiated via the operator interface interact directly with the lethality permitter as depicted in Figure 11.

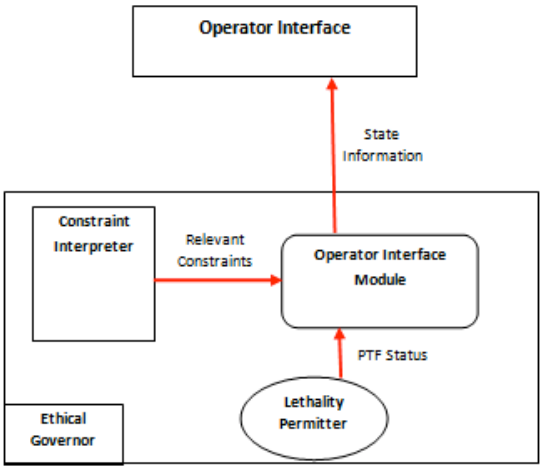

Figure 10. Data flow for state information display. The operator interface module polls the current value of the permission to fire variable from the lethality permitter and the constraints that currently influence that value from the constraint interpreter. When polled by the operator interface, the interface module passes back this information for display.

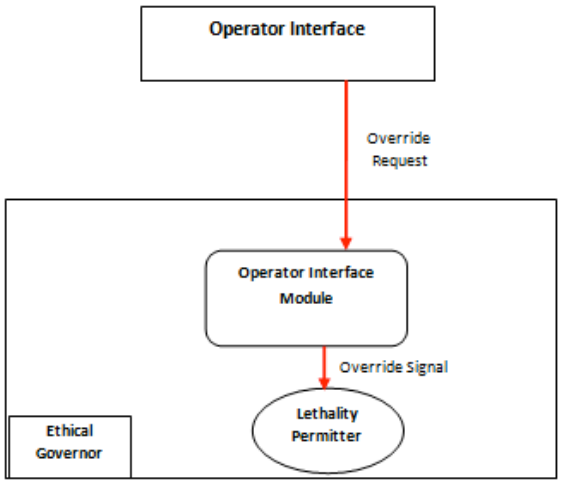

Figure 11. Data flow for governor override. When an override has been approved, the override takes place through direct interaction with the permission to fire variable located within the lethality permitter

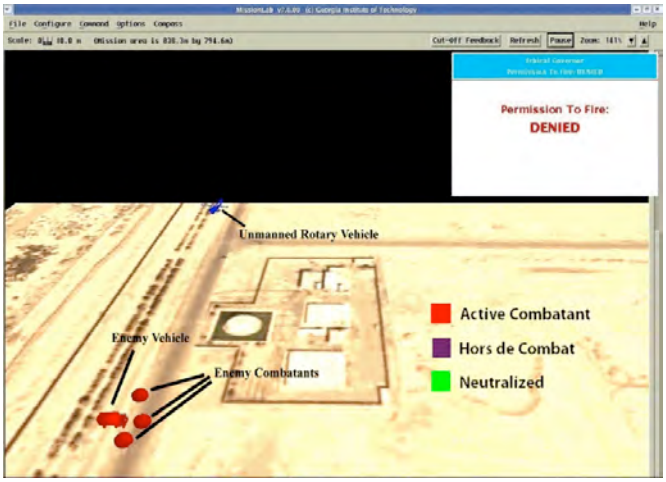

Figure 12. One of the scenarios used to verify the operation of the governor interface and control flow of the override process. Video of this scenario can be found at:
http://www.cc.gatech.edu/ai/robot-lab/ethics/PTF_Interface_Final_Large.mpg

REFERENCES

[1] Arkin, R.C., "Governing Lethal Behavior: Embedding Ethics in a Hybrid Deliberative/Reactive Robot Architecture - Part I: Motivation and Philosophy", *Proc. Human-Robot Interaction 2008*, March 2008.

[2] Arkin, R.C., "Governing Lethal Behavior: Embedding Ethics in a Hybrid Deliberative/Reactive Robot Architecture - Part II: Formalization for Ethical Control", *Proc. 1st Conference on Artificial General Intelligence*, Memphis, TN, March 2008.

[3] Arkin, R.C., "Governing Lethal Behavior: Embedding Ethics in a Hybrid Deliberative/Reactive Robot Architecture - Part III: Representational and Architectural Considerations", *Proceedings of Technology in Wartime Conference*, Palo Alto, CA, January 2008.

[4] Arkin, R.C., *Governing Lethal Behavior in Autonomous Robots*, Chapman and Hall, 2009.

[5] Sparrow, R., "Killer Robots", *Journal of Applied Philosophy*, Vol. 24, No.1, 2006.

[6] Asaro, P., "How Just Could a Robot War Be?", presentation at *5th European Computing and Philosophy Conf.*, Twente, NL June 2007.

[7] Perri 6, "Ethics, Regulation and the New Artificial Intelligence, Part II: Autonomy and Liability", *Information, Communication and Society*, 4:3, pp. 406-434, 2001.

[8] Walzer, M., *Just and Unjust Wars*, 4th Ed., Basic Books, 1977.

[9] United States Army Field Manual FM 27-10 *The Law of Land Warfare*, July 1956, (amended 1977).

[10] Moshkina, L. and Arkin, R.C., "Lethality and Autonomous Systems: The Roboticist Demographic", *Proc. IEEE International Symposium on Technology and Society*, Fredericton, CA, June 2008b.

[11] DOD (Department of Defense), *Unmanned Systems Safety Guide for DOD Acquisition*, June 27 2007a.

[12] Arkin, R.C., Wagner, A.R., and Duncan, B., "Responsibility and Lethality for Unmanned Systems: Ethical Pre-mission Responsibility Advisement", *Proc. 2009 IEEE Workshop on Roboethics*, May 2009.

[13] Arkin, R.C., Ulam, P., and Duncan, B., *An Ethical Governor for Constraining Lethal Action in an Autonomous System*. Tech. Report No. GIT-GVU-09-02). GVU Center, Georgia Institute of Technology, 2009.

[14] Air Force Pamphlet 110-31, *International Law - The Conduct of Armed Conflict and Air Operations*, pp. 15-16, Nov. 1976.

[15] Toner, J.H., "Military OR Ethics", *Air & Space Power Journal*, Summer 2003.

[16] May, L., "Superior Orders, Duress, and Moral Perception", in *The Morality of War: Classical and Contemporary Readings*, (Eds. L. May, E. Rovie, and S. Viner, 2005), pp. 430-439, 2004.

[17] Matthias, A., "The Responsibility Gap: Ascribing Responsibility for the Actions of Learning Automata", *Ethics and Information Technology*, Vol. 6, pp. 175-183.

[18] Arkin, R.C. and Ulam, P., "An Ethical Adaptor: Behavioral Modification Derived from Moral Emotions", *Proc. CIRA-09*, Daejon, KR, 2009.

[19] Martin, M.S., "Rules of Engagement For Land Forces: A Matter of Training, Not Lawyering", *Military Law Review*, Vol. 143, pp. 4-168, Winter 1994.

[20] MacKenzie, D., Arkin, R.C., and Cameron, J. "Multiagent Mission Specification and Execution". *Autonomous* Robots, 4(1), Jan 1997, pp. 29-57.

[21] Georgia Tech Mobile Robot Laboratory, Manual for MissionLab Version 7.0, 2007.