Technical Report GIT-GVU-09-02

# An Ethical Governor for Constraining Lethal Action in an Autonomous System

Ronald C. Arkin, Patrick Ulam, and Brittany Duncan

*Abstract*— **The design, prototype implementation, and demonstration of an ethical governor capable of restricting lethal action of an autonomous system in a manner consistent with the Laws of War and Rules of Engagement is presented.**

## I. Introduction

Weaponized military robots are now a reality. Currently, a human 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 [1-3] 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 component of the overall architecture (Fig. 1), the ethical governor. This component is a transformer/suppressor of system-generated lethal action to ensure that it constitutes an ethically permissible action, either nonlethal or obligated ethical lethal force. This deliberate bottleneck is introduced into a hybrid deliberative/reactive architecture, in essence, to force a second opinion prior to the conduct of a privileged lethal behavioral response.

## II. An Ethical Governor

This section outlines the design for the ethical governor component of the architecture. This component's responsibility is to conduct an evaluation of the ethical appropriateness of any lethal response that has been produced by the robot architecture prior to its being enacted. It can be largely viewed as a bolt-on component between the hybrid architectural system and the low-level controllers and actuators, intervening as necessary to prevent an unethical response from occurring. Technically, the governor can be considered a part of the overall deliberative system of the architecture that is concerned with response evaluation and confirmation. It is considered a separate component, however, in this work as it does not require high-levels of interaction with the other main components of deliberation (although it can request replanning) and it can be deployed in an otherwise purely reactive architecture if desired.

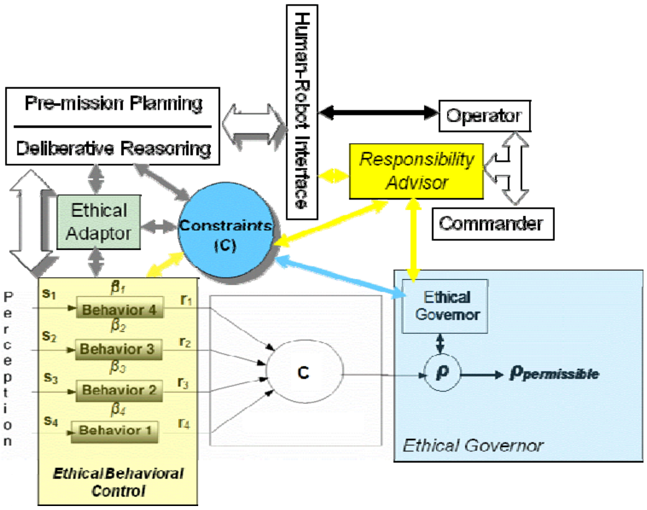


Figure 1: Ethical Architecture (See [4] for details)

The term governor is inspired by Watts' invention of the mechanical governor for the steam engine, a device that was intended to ensure that the mechanism behaved safely and within predefined bounds of performance. As the reactive component of a behavioral architecture is essentially a behavioral engine intended for robotic performance, the same notion applies, where here the performance bounds are ethical ones.

In this architecture, the overt robotic response $\boldsymbol{\rho} \in \boldsymbol{P}$ is the behavioral response of the agent to a given situation $\mathbf{S_i}$. To ensure an ethical response, the following must hold: $\{\forall\, \boldsymbol{\rho} \mid \boldsymbol{\rho} \notin \boldsymbol{P}_{l\text{-}unethical}\}$ where $\boldsymbol{P}_{l\text{-}unethical}$ denotes the set of all unethical lethal responses. Formally, the role of the governor is to ensure that an overt lethal response $\boldsymbol{\rho}_{lethal\text{-}ij}$ for a given situation is ethical, by confirming that it is either within the response set $\boldsymbol{P}_{l\text{-}ethical}$ or is prevented from being executed by mapping an unethical $\boldsymbol{\rho}_{lethal\text{-}ij}$ onto the null response (i.e., ensuring it is ethically permissible). If the ethical governor needs to intervene, it must send a notification to the deliberative system in order to allow for replanning at either a tactical or mission level as appropriate, and to advise the operator of a potential ethical infraction of a constraint or constraints $c_k$ in the ethical constraint set $C$.

This work was supported by the Army Research Office under Contract #W911NF-06-1-0252.

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, gth656@gmail.gatech.edu)..

Each constraint $c_k \in C$ specified must have at least the following data fields:

1. **Logical form**: As derived from propositional or deontic logic. (e.g., [5]).
2. **Textual descriptions:** Both a high-level and detailed description for use by the Responsibility Advisor [3].
3. **Active status flag**: Allows mission-relevant ROE to be defined within an existing set of constraints, and to designate operator overrides.
4. **Base types**: Forbidden (e.g., LOW or ROE derived) or obligated (e.g., ROE derived). These will be relegated to either a long-term memory (LTM) for those constraints which persist over all missions, or a short-term memory (STM) for those constraints that are derived from specific current ROE for given Operational Orders. Changes in LTM, that encode the LOW, require special two-key permission.
5. **Classification**: One chosen from Military Necessity, Proportionality, Discrimination, Principle of Double Intention [6], and Other and used only to facilitate processing by ordering the application of constraints by class.

Real-time control must be achieved for in-the-field reasoning. This assumes that the perceptual system of the architecture is charged with producing a certainty measure $\lambda$ for each relevant stimulus (e.g., candidate target) $\mathbf{s} \in S$ that is represented as a binary tuple $(p,\lambda)$, where $p$ is a perceptual class (e.g., combatant or noncombatant). In addition, a mission-contextual perceptual threshold $\tau$ for each relevant perceptual class is also evaluated. Mission-specific thresholds are set prior to the onset of the operation. The details of the currently implemented approach appear in Section III.

It is a major assumption of this research that accurate target discrimination with associated uncertainty measures can be achieved despite the fog of war, but it is believed that this is ultimately possible for a range of reasons as presented in [1]. The architecture described herein is intended to provide a basis for ethically acting upon that information once produced. To achieve this level of performance, the ethical governor (Fig. 2) will require inputs from:

1. The overt response, $\boldsymbol{\rho}$, generated by the behavioral controller
2. The perceptual system
3. The constraint set $C$ (both long-term and short-term memory)
4. The Global Information Grid (GIG) to provide additional external sources of intelligence.

Specific methods for evidential reasoning, which are yet to be determined but likely probabilistic, will be applied to update the target's discrimination and quality using any available additional information from the GIG regarding any candidate targets designated for engagement by the controller. Should the target be deemed legitimate to engage, a proportionality assessment is conducted.

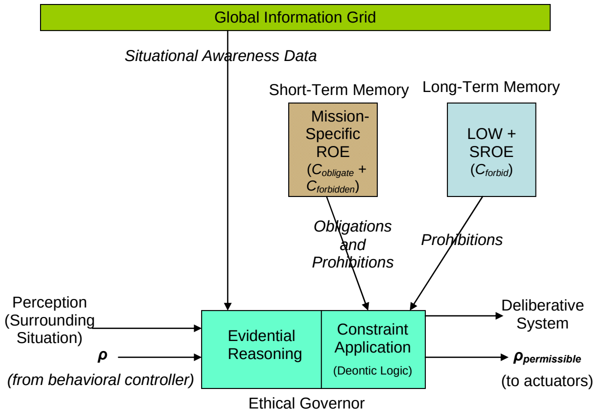


Figure 2: Ethical Governor Architectural Components

Logical assertions can be created from situational data arriving from perception, and inference is then conducted within the constraint application component of the ethical governor using the constraints obtained from STM and LTM. The end result yields a permissible overt response $\boldsymbol{\rho}_{permissible}$, and when required, notification and information will be sent to the deliberative system and operator regarding potential ethical violations. The use of constraints embodying the Principle of Double Intention [6] ensures that more options are evaluated when a lethal response is required than might be normally considered by a typical soldier.

Simply put, this is a constraint satisfaction problem for $C_{Obligate}$ with inviolable constraints for $C_{Forbidden}$. Proportionality can be conducted by running, if needed, an optimization procedure on $C_{Obligate}$ after permission is received over the space of possible responses (from none, to weapon selection, to firing pattern, to aiming, etc.). This provides for proportionality by striving to minimize collateral damage when given appropriate target discrimination certainty. If the potential target remains below the certainty threshold and is thus ineligible for engagement, the system could invoke specific behavioral tactics to increase the certainty of discrimination.

## III. Implementation

In order to evaluate the ethical governor, one component of the architecture for ethical control of mobile robots, a prototype was developed within *MissionLab*, a mission specification and simulation environment for autonomous robots [7]. A high-level overview of the implemented architecture for the ethical governor can be seen in Figure 3. This section discusses the components of this architecture and how they were realized within the prototype system.

The ethical governor is divided into two main processes: evidential reasoning and constraint application. Evidential reasoning is responsible for transforming incoming perceptual, motor, and situational awareness data into evidence necessary for governing lethal behavior. Constraint application is responsible for using the evidence to apply constraints that encode the LOW and ROE for the suppression of unethical behavior.

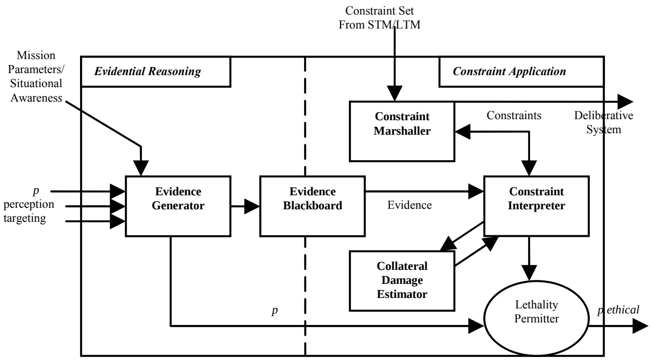


Figure 3. Architecture and data flow overview of the ethical governor

## A. *Ethical Constraints*

Constraints, as discussed earlier, are the data structures which encode the LOW and ROE that must be met by the robot in order to ensure ethical behavior is exhibited by the system. In the prototype implementation of the ethical governor, the data structure used to store the relevant constraint information is shown in Figure 4. The data structure is composed of six fields. The constraint type field encodes if the constraint is an *obligation* for or a *prohibition* against lethal behavior. The origin and description fields provide additional information that while not used directly by the governor, serve to provide human-readable information for informing the operator or deliberative system why lethal behavior was permissible or suppressed by the governor. The activity field indicates if the constraint is active. Constraints that are inactive are not used in the constraint application process for the current mission and do not affect the behavior of the ethical governor. Finally, the logical form field, currently encoded via propositional logic, serves to formally describe the conditions under which the obligation or prohibition is applicable in a machine-readable format suitable for use by the constraint application process. Figure 5 shows an example of a populated constraint used within this work, where the constraint encodes a prohibition against damaging a cultural landmark as derived from the LOW.

In support of the operation of the ethical governor these constraints are stored in two repositories. The constraints encoding the LOW, as they are not likely to change over time, are stored in long term memory (LTM). The constraints which encode the rules of engagement for a particular mission are instead stored within short term memory (STM). Short term and long term memory are implemented in the form of constraint databases. These databases can be queried by other components in the overall architecture in order to retrieve constraints that match desired criteria (e.g. the constraint application process will query STM and LTM for all *active* constraints).

| Field | Description |
|---|---|
| | |
| **Constraint Type** | Type of constraint described |
| **Constraint Origin** | The origin of the prohibition or obligation described by the constraint |
| **Active** | Indicates if the constraint is currently active |
| **High-Level Constraint Description** | Short, concise description of the constraint |
| **Full Description of the Constraint** | Detailed text describing the law of war or rule of engagement from which the constraint is derived |
| **Constraint Classification** | Indicates the origin the constraint. Used to order constraints by class. |
| **Logical Form** | Formal logical expression defining the constraint |

Figure 4. Format of the constraint data structure

| **Constraint** | |
|---|---|
| **Type** | Prohibition |
| **Origin** | Laws of war |
| **Activity** | Active |
| **Brief Description** | Cultural Proximity Prohibition |
| **Full Description** | Cultural property is prohibited from being attacked, including buildings dedicated to religion, art, science… |
| **Logical Form** | TargetDiscriminated AND TargetWithinProxOfCulturalLandmark |

Figure 5. The contents of a constraint encoding a prohibition against engaging targets in proximity to a cultural landmark.

## B. *Evidential Reasoning*

The evidential reasoning process transforms incoming perceptual, motor, and situational awareness data into evidence in the form of logical assertions to be used by the constraint application process. Evidential reasoning is the result of two interacting components: the evidence generation module and the evidence blackboard. Perceptual information, target information, and the overt behavioral response (ρ) from the behavioral control system are received by the evidence generation module. In addition, mission-specific information such as the geographical constraints of the current theater of operations is sent to the evidence generation module for processing along with any externally available situational awareness data.

This data is used by the evidence generation module to create logical assertions describing the current state of the robot and the current state of any potential targets involving lethal force. The assertions generated range from those indicating that the target has been properly discriminated and that the target is within a designated kill zone, to assertions indicating that the

```
DO WHILE AUTHORIZED FOR LETHAL RESPONSE, MILITARY NECESSITY EXISTS, AND RESPONSIBILITY ASSUMED
    IF Target is Sufficiently Discriminated
        IF $C_{Forbidden}$ satisfied /* permission given – no violation of LOW exists */
            IF $C_{Obligate}$ is true /* lethal response required by ROE */
                Optimize proportionality using Principle of Double Intention
                Engage Target
            ELSE /* no obligation/requirement to fire */
              Do not engage target
              Continue Mission
        ELSE /* permission denied by LOW */
            IF previously identified target surrendered or wounded (neutralized)
                /* change to non-combatant status */
                Notify friendly forces to take prisoner
            ELSE
                Do not engage target
                Report and replan
                Continue Mission
    Report status
END DO
```

Figure 6. Constraint application algorithm. $C_{Forbidden}$ and $C_{Obligate}$ are the set of active prohibition and obligation constraints respectively

target is in proximity to a medical facility. The logical assertions generated are then sent to the evidence blackboard, the communications medium between the evidential reasoning process and the constraint application process. The evidence blackboard serves as the repository for all the logical assertions created by the evidential reasoning process. For each execution cycle where a behavioral response is input into the governor, the evidence placed upon the blackboard is recomputed and the constraint application process re-evaluates the current ethical constraints.

### C. *Constraint Application*

The constraint application process is responsible for reasoning about the active ethical constraints and ensuring that the resulting behavior of the robot is ethically permissible. The constraint application process is also the product of a number of interacting subsystems. These subsystems include the constraint marshaller, the constraint interpreter, the collateral damage estimator, and the lethality permitter.

The first step in the constraint application process is the retrieval of the active ethical constraints from STM and LTM by the constraint marshaller. The constraint marshaller serves to retrieve and transport constraints to and from the ethical governor. Once the constraint marshaller has retrieved the constraints from memory, it then transports these constraints to the constraint interpreter for evaluation. The constraint interpreter serves as the reasoning engine for evaluation of these constraints. Within the prototype described in this section, the constraint interpreter was implemented as a lisp-based logic interpreter. The exact form this reasoning engine takes is not central to the composition of the ethical governor, and other more sophisticated reasoning engines can be substituted without loss of generality.

In order to determine if the output of the behavioral control system is ethically permissible, the constraint interpreter must evaluate the constraints retrieved from memory. Recall from Section II, these constraints can be divided into two sets: the set of prohibition constraints $C_{Forbidden}$ and the set of obligating constraints $C_{Obligate.}$ The constraint interpreter evaluates the permissibility of the incoming behavior by evaluating if these two constraint sets are satisfied for the action proposed by the behavioral controller.

To do this, the constraint interpreter first retrieves all the logical assertions generated by the evidential reasoning process from the blackboard and maps these assertions to the formal logical statements that define each of the active constraints in $C_{Obligate}$ and $C_{Forbidden}$. Once this mapping is complete, the constraints are evaluated by the reasoning engine within the interpreter. The algorithm by which the reasoning engine evaluates the constraints is shown in Figure 6. In this algorithm, the prohibition constraint set ($C_{Forbidden}$) is evaluated first. In order for the constraint set $C_{Forbidden}$ to be satisfied, the interpreter must evaluate *all* of the constraints in $C_{Forbidden}$ to be *false*, i.e.,, the behavior input to the governor must not result in prohibited/unethical behavior.

If $C_{Forbidden}$ is not satisfied, the lethal behavior being evaluated by the governor is deemed unethical and must be suppressed. This process is discussed below. If $C_{Forbidden}$ is satisfied, however, the constraint interpreter then verifies if lethal behavior is *obligated* in the current situation. In order to do this, the constraint interpreter evaluates all the active obligating constraints ($C_{Obligate}$). The obligating constraint set is satisfied if *any* constraint within $C_{Obligate}$ is satisfied. If $C_{Obligate}$ is not satisfied, on the other hand, lethal behavior is not permitted and must be suppressed by the ethical governor.

In the case that either $C_{Forbidden}$ or $C_{Obligate}$ is not satisfied, lethal behavior is suppressed as impermissible by the ethical governor. The suppression takes place by sending a suppression message from the constraint interpreter to the lethality permitter, the component of the governor that serves as the gateway between the behavioral controller and the vehicle's actuators. If a suppression message is received by the lethality permitter, the outgoing behavior is transformed into one that does not exhibit lethal behavior. In the implementation

```
Calculate_Proportionality(Target, Military Necessity, Setting)

   Select the weapon with highest effectiveness based on Target, Necessity and Setting

   MinumumCarnage = $\infty$
   SelectedReleasePosition = NULL
   SelectedWeapon = NULL

   WHILE all weapons have not been tested
      FOR all release positions that will neutralize the target
         IF $C_{Forbidden}$ Satisfied for that position          // if the position does not violate the LOW
            Calculate Carnage for the position
            IF Carnage < MinimumCarnage                         // Carnage is reduced
               SelectedReleasePosition = position
               SelectedWeapon = weapon
               MinimumCarnage = carnage
            ENDIF
         ENDIF
      ENDFOR


      IF Carnage is too high given military necessity of target or $C_{Forbidden}$ could not be satisfied
         Down-select weapon
         IF there are no more weapon systems available
            Return Failure
         ENDIF
      ELSE
         Return Weapon and Release Position
ENDWHILE
```

Figure 7. High-level algorithm used to calculate proportionality. The algorithm selects the most effective weapon system and ensures that the use of the weapon will not violate any prohibitions and then calculates the carnage that would result from the combination of weapon system and release position. If no release position results in permissible behavior or an acceptable level of carnage given the military necessity, the algorithm select a less effective weapon system and searches the space of release positions again.

described here, this simply results in the robot resuming its specified mission. In addition to the suppression message sent to the lethality permitter, the deliberative system is informed of the constraints that were not satisfied so that replanning or alternate actions can be performed by the robot or human commander.

Before the robot exhibits lethal behavior, not only must the constraint sets $C_{Forbidden}$ or $C_{Obligate}$ be satisfied, but the ethical governor must also ensure that the behavior adheres to proportionality constraints guided by the Principle of Double Intention [6]. The next section describes the collateral damage estimator, the component that ensures that any lethal behavior adheres to just war proportionality constraints.

### *D. Proportionality and Battlefield Carnage*

After the constraint interpreter has established that both the obligating and prohibition constraints have been satisfied, it is necessary to ensure that the type of lethal behavior exhibited by the robot is appropriate given the military necessity associated with the target. This is done by optimizing the likelihood of target neutralization while minimizing any potential collateral damage that would result from engaging the target with lethal force. The collateral damage estimator serves to modify lethal behavior so that these factors are taken into account. It does this by searching over the space of available weapon systems, targeting patterns and weapon release positions for a combination that serves to maximize likelihood of target neutralization while minimizing collateral damage and ensuring the ethical application of force for a given military necessity level. The high-level algorithm depicting this process is shown in Figure 7.

In the prototype implementation described, a simulated unmanned aerial vehicle (UAV) was equipped with a set of four weapon systems: a chain gun, hellfire missiles, and either GBU-12 or GBU-38 500lb warheads. Each weapon system was assigned a set of hypothetical parameters for use in the proportionality calculations, the most relevant of which were: likelihood of target neutralization (based on target type), target neutralization radius, non-combatant damage radius, and structural damage radius (used to compute the area surrounding the weapon impact point that would result in target neutralization, non-combatant causalities, and structural damage respectively). Examples of the weapon statistics used in the implementation of the collateral damage estimator described here are shown in Figure 8.

The proportionality algorithm shown in Figure 7 uses these statistics as well as perceptual information about the environment to determine the battlefield carnage in a utilitarian manner, by estimating the amount of structural damage, the number of non-combatant/combatant/friendly casualties that result from the use of a weapon system at a particular target location. There are three possible outcomes of the proportionality algorithm. In the first, the proportionality algorithm finds no weapon system or weapon release position that does not violate an ethical constraint (e.g., the target may be near a medical facility and the resulting blast radius of the weapon systems would damage that facility). In this case, the ethical governor suppresses the lethal behavior via the lethality permitter. In the second case, no weapon system or weapon release position is found that results in an

| Weapon | Effectiveness Against Convoy 2-4 Vehicles | Combatant Damage Radius | Non-Combatant Damage Radius | Struct. Damage Radius |
|---|---|---|---|---|
| Chaingun | 2% | 0.5ft | 1ft | 0.5ft |
| Hellfire | 20% | 10ft | 20ft | 10ft |
| GBU-12 | 90% | 1000ft | 2000ft | 500ft |

Figure 8. Example of weapon statistics used by the collateral damage estimator. This entry depicts the result of utilizing the weapon system against a small convoy of vehicles.

| Military Necessity (1 low, 5 high) | No Collateral Damage | Low Collateral Damage | Medium Collateral Damage | High Collateral Damage |
|---|---|---|---|---|
| 1 | Permissible | Forbidden | Forbidden | Forbidden |
| 2 | Permissible | Permissible | Forbidden | Forbidden |
| 3 | Permissible | Permissible | Permissible | Forbidden |
| 4 | Permissible | Permissible | Permissible | Forbidden |
| 5 | Permissible | Permissible | Permissible | Permissible |

Figure 9. Depicts the table used to determine acceptable levels of collateral damage given the military necessity (priority) associated with a target.

acceptable level of collateral damage given the military necessity of the target (e.g., engaging a low priority target would result in significant, indirect non-combatant casualties). In this case, once again, the ethical governor suppresses the use of lethal force via the lethality permitter. An acceptable level of collateral damage as a function of the military necessity of a target used in the prototype system appears in Figure 10 (these values are purely hypothetical for these examples). The final possible outcome of the proportionality estimation is the identification of a weapon/weapon release position combination that satisfies all ethical constraints and minimizes collateral damage in relation to the military necessity of the target. In this case, the behavioral governor informs the lethality permitter that lethal behavior is permissible for this situation and the robot is allowed to engage the target using lethal force based upon the selected weapon system and release position.

## IV. Demonstration

In order to evaluate the feasibility of the ethical governor, a series of test scenarios were developed within the *MissionLab* simulation environment [7]. A variety of situations were presented to an autonomous fixed-wing UAV in which the ethical use of lethal force must be ensured. This section presents two such test scenarios, and describes the interaction of the ethical governor with the behavioral control system in detail.

In both scenarios, the UAV has been assigned to perform a hunter-killer mission along a predetermined flight path, where the UAV has been authorized to engage a variety of targets including musters of enemy soldiers, small convoys of enemy vehicles, and enemy tanks. Engagement of enemy forces, however, may *only* occur if the targets are within designated mission-specific kill zones. An overview of the mission area and landmarks pertinent to this discussion appear in Figure 10. As there are no known high-priority targets known to be present in the mission area, the military necessity associated with engaging these small groups of enemy units is relatively low (Military Necessity of 2, Fig. 9). As a result, lethal force should only be applied if collateral damage can be significantly minimized. Figure 11 depicts the subset of relevant ethical constraints that are pertinent here. While there are significantly more constraints in use then shown, only those that are involved in the following scenarios are depicted. The UAV is equipped with 4 hellfire missiles and 2 GBU-12 warheads. The default action of the underlying behavioral controller that is fed into the ethical governor in these scenarios is to engage any discriminated enemy targets with lethal force. This behavior is exhibited for the purpose of demonstrating the ethical governor within the scenarios. If such a system were to be deployed, it is likely that the behavioral controller would be ethically constrained in the manner as suggested by Arkin [4].

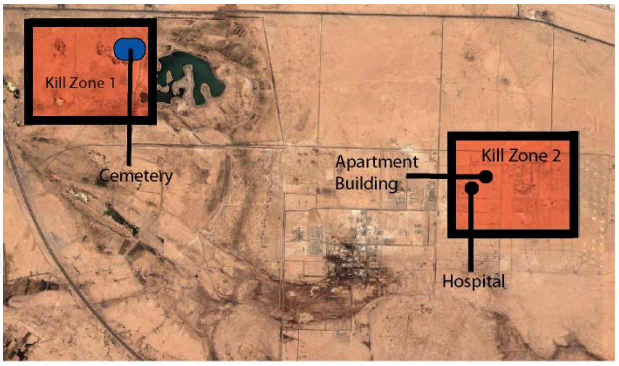


Figure 10, Mission area used in demonstration scenarios. There are two mission-designated kill zones. A cemetery lies within kill zone one while an apartment building and hospital are within kill zone two.

| Type | Origin | Description |
|---|---|---|
| | | |
| Prohibition | ROE | It is forbidden to engage enemy units outside of designated mission boundaries |
| Prohibition | LOW | Cultural property is prohibited from being attacked, including buildings dedicated to religion, art, science, charitable purposes, and historic monuments. |
| Prohibition | LOW | Civilian hospitals organized to give care to the wounded and sick, the infirm and maternity cases, may in no circumstances be he object of attack, but shall at all times be respected and protected by the parties to the conflict. |

Figure 10. Several of the constraints relevant to the demonstration scenarios.

### A. *Scenario 1 – Enemy muster within a cemetery*.

In the first scenario, the UAV encounters an enemy muster attending a funeral within a designated kill zone. Upon discrimination, the underlying behavioral controller outputs a command to engage the muster with lethal force. The behavioral controller's output is then sent to the ethical governor to ensure that action is ethical before that behavior is expressed by the actuators. Figure 12 shows this scenario at the point of target discrimination.

On receipt of the behavioral input exhibiting lethal force, the ethical governor initiates the evidence generation and constraint application processes. The evidence generation module processes the incoming perceptual information, situational awareness information, and mission parameters to generate the evidence needed by the constraint application process. In this scenario, examples of the evidence generated include logical assertions such as: *Target Within Killzone, Target Is Discriminated, Target In Proximity of a Cultural Landmark,* and *Target Is a Muster*. This evidence, along with any other evidence created by the evidence generation process is placed on the evidence blackboard for use by the constraint application process.

Once the evidence has been generated, the constraint application process begins with the retrieval of all active ethical constraints from memory. Pertinent constraints retrieved in this scenario are shown in Figure 10. Once these constraints have been delivered to the constraint interpreter and the evidence retrieved from the blackboard, the constraint interpreter begins to evaluate the constraints using the algorithm shown in Figure 6. The constraint application algorithm begins by ensuring the set of prohibition constraints ($C_{Forbidden}$) is satisfied. In this scenario, when the constraint interpreter evaluates the prohibition against engaging targets within proximity to cultural landmarks (Fig. 5), the constraint fails to be met (as the cemetery is considered to be a cultural landmark). The failure of $C_{Forbidden}$ to be satisfied indicates that the lethal behavior being governed is unethical. This results in a suppression signal being sent to the lethality permitter that suppresses the proposed lethal behavior (Figure 13). The deliberative system is also informed that suppression has occurred and is informed of the reason (constraint) that caused the suppression.

### *B. Scenario 2 – Maintaining Ethical Behavior While Minimizing Collateral Damage*

In the second scenario, the UAV has encountered and discriminated an enemy convoy within the second kill zone. A short distance to the west and in close proximity to the convoy is a regional hospital, a heavily populated apartment building to the north, and a clearly identified stationary taxi-cab to the south (Figure 14). When the convoy was identified, the underlying behavioral controller attempts to engage the enemy units.

As before, when the lethal behavior output by the behavioral controller enters the ethical governor, the evidential reasoning and constraint application processes attempt to determine if that lethal behavior is permissible. After the evidence has been generated and the active constraints retrieved, the constraint interpreter applies the constraint application algorithm (Fig. 6). First the algorithm ensures that the prohibition constraint set, $C_{Forbidden}$, is satisfied. In this scenario, none of the prohibitions are violated. The governor then determines if lethal force is obligated by evaluating the constraint set $C_{Obligate}$. The constraint interpreter determines that the obligating constraint, "Enemy convoys must be engaged," (at this level of military necessity) is satisfied and therefore, $C_{Obligate}$ is satisfied. Finally, the governor must ensure that the lethal force exhibited by the UAV is proportional as guided by the principle of double intention, using the algorithm shown in Figure 7.

During the calculation of a proportional response, the most effective yet humane weapon system is selected and the system begins searching through the space of possible weapon release positions in order to minimize collateral damage. During the search, a candidate release position is evaluated in two ways: if the release position satisfies $C_{Forbidden}$, and by the number of non-combatant casualties anticipated. If a release position is found to violate $C_{Forbidden,}$ the release of the weapon in that position is deemed unethical and may not be used. An example of a release position that violates ethical constraints can be seen in Figure 15. In this figure, the concentric circles

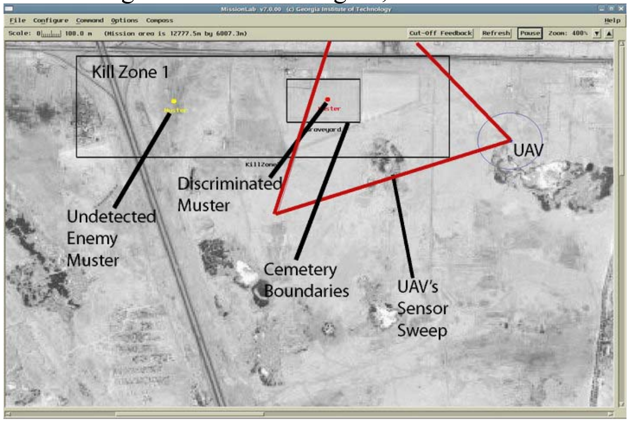
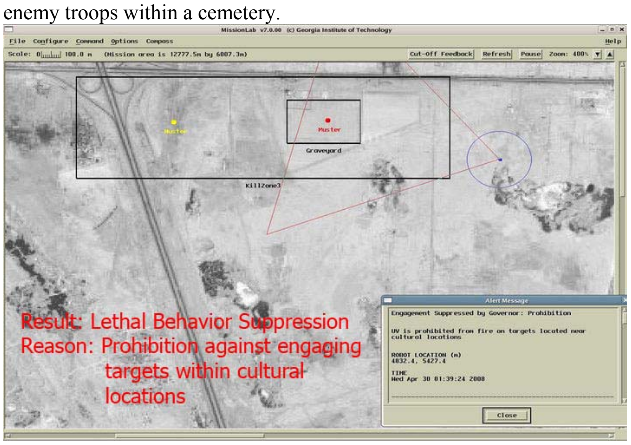


Figure 12. The UAV detects and confidently discriminates a muster of enemy troops within a cemetery.



Result: Lethal Behavior Suppression
Reason: Prohibition against engaging targets within cultural locations

```
Engagement Suppressed by Governor: Prohibition

UV is prohibited from fire on targets located near
cultural locations

ROBOT LOCATION (m)
4832.4, 5427.4

TIME
Wed Apr 30 01:39:24 2008
```

Figure 13. Lethal behavior is suppressed due to the behavior failing to satisfy the prohibition against engaging enemies in proximity to cultural locations.

represent the area hypothetically in which (from inner to outer circles) structural damage, combatant casualties, and non-combatant casualties may take place. The figure shows the location where the release of a GBU-12 would result in the medical facility being damaged, thus violating the LOW prohibition against damaging medical facilities.

In this scenario, the military necessity associated with neutralizing targets in the mission area is moderate, thus only limited collateral damage is tolerated. Therefore, a weapon release position that would damage the heavily populated apartment building is forbidden (i.e., the area that will sustain structural damage may not include the apartment building). The constraint application process, therefore, continues searching the space of weapons and weapon release positions such that neither the hospital nor the apartment building will sustain damage. If such a position cannot be found, lethal behavior is not permitted. In this case, however, a weapon release position is found such that neither building sustains damage and such that non-combatant casualties remain low. This ethical release position for the GBU-12 is shown in Figure 16. Note that as there did not exist a release location from which non-combatant casualties could be completely eliminated and because the military necessity of the target allowed for limited collateral damage, the ethical weapon release position *does* result in a potential non-combatant fatality (i.e., occupants of the taxi-cab). The governor, however, does minimize casualties by ensuring the heavily populated apartment building is avoided.

**Note: A video accompanies the submission of this paper**.

## 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: Embedding Ethics in a Hybrid Deliberative/Reactive Robot Architecture ", Technical Report GIT-GVU-07-11, Georgia Tech GVU Center, 2007.

[5] Bringsjord, S. Arkoudas, K., and Bello, P., "Toward a General Logicist Methodology for Engineering Ethically Correct Robots", *Intelligent Systems*, July/August, pp. 38-44, 2006.

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

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

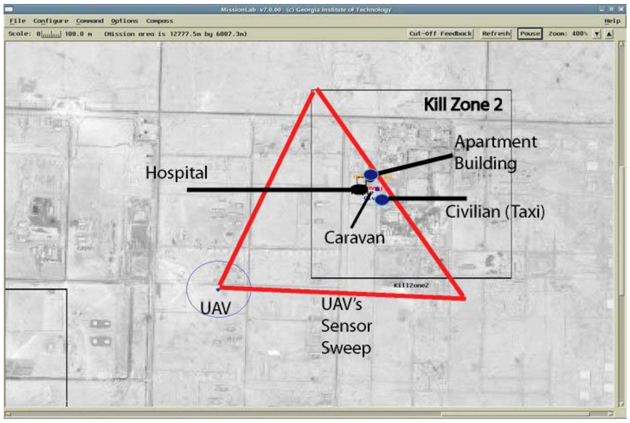

MissionLab v7.0.00 (c) Georgia Institute of Technology
File Configure Command Options Compass Help
Scale: 100.0 m (Mission area is 12777.5m by 6087.3m)
Cut-Off Feedback Refresh Pause Zoom: 400%
Kill Zone 2
Apartment Building
Hospital
Civilian (Taxi)
Caravan
KillZone2
UAV
UAV's Sensor Sweep

Figure 14. The UAV encounters an enemy convoy centered between a hospital, an apartment building and a stationary taxi.

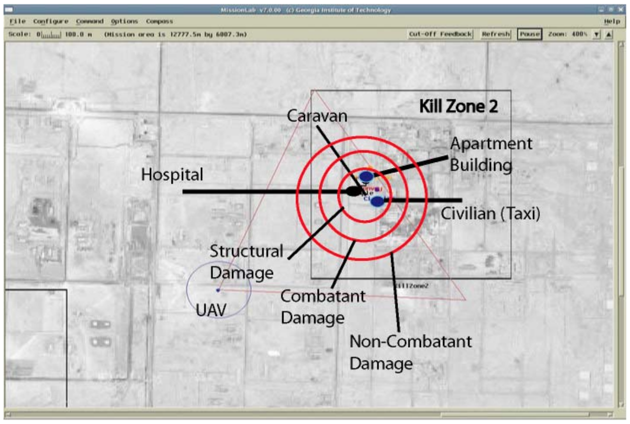
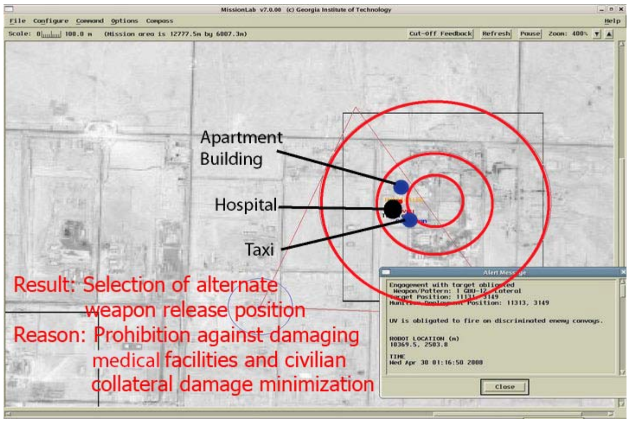

MissionLab v7.0.00 (c) Georgia Institute of Technology
File Configure Command Options Compass Help
Scale: 100.0 m (Mission area is 12777.5m by 6087.3m)
Cut-Off Feedback Refresh Pause Zoom: 400%
Caravan
Kill Zone 2
Apartment Building
Hospital
Civilian (Taxi)
Structural Damage
KillZone2
UAV
Combatant Damage
Non-Combatant Damage

Figure 15. Example of a weapon release position that violates ethical constraints. The structural damage area covers the area where the hospital is located. The blast radii are based on the collateral damage assessment calculated by using the selected weapon's blast radius (Figure 7).


MissionLab v7.0.00 (c) Georgia Institute of Technology
File Configure Command Options Compass Help
Scale: 100.0 m (Mission area is 12777.5m by 6087.3m)
Cut-Off Feedback Refresh Pause Zoom: 400%
Apartment Building
Hospital
Taxi
Alert Message
Close

Result: Selection of alternate weapon release position
Reason: Prohibition against damaging medical facilities and civilian collateral damage minimization

```
Engagement with target obligated
 Weapon/Pattern: 1 GBU-12 Lateral
Target Position: 11131, 3149
Munition Deployment Position: 11313, 3149

UV is obligated to fire on discriminated enemy convoys.

ROBOT LOCATION (m)
10369.5, 2503.8

TIME
Wed Apr 30 01:16:50 2008
```

Figure 16. The final weapon release position selected by the ethical governor. This position ensures that all ethical constraints are satisfied and civilian causalities are minimized while maximizing the chance of target neutralization.