Unmanned Military Systems, the Frame Problem, and Computer Security

Michał Klincewicz[1]
[1] City University of New York, Graduate Center, New York, USA, Philosophy and Cognitive Science
Michal.Klincewicz@gmail.com

Abstract

Unlike human soldiers, autonomous unmanned military systems (UMS) are unaffected by psychological factors and will never act outside the chain of command. This is a compelling moral justification for their development and eventual deployment in war. To achieve this level of sophistication, UMSs will have to first solve the frame problem, which implies that they will be endowed with very complex software. Complex software of this sort will create security risks and will make UMSs critically vulnerable to hacking. The political and tactical consequences of hacked UMSs far outweigh the purported advantages of UMSs not being affected by psychological factors and always following orders. Consequently, one of the moral justifications for the deployment of UMSs is undermined.

1 Introduction

There are at least two sources of ethical problems associated with autonomous unmanned military systems (UMS). The first are the deep problems of artificial intelligence (AI). Can AIs have the kind of free-will that we associate with moral responsibility? Are AIs persons in any sense? Are AIs capable of suffering? At this level of analysis, ethical problems of UMSs fall squarely in the domain of moral philosophy, philosophy of mind, and metaphysics. Solutions to these are probably not coming soon.

The second source of ethical problems for UMSs is more mundane. These are the shallow problems, which result from technological limitations and the use of UMSs. One important shallow problem comes from the current state of computer vision research, which renders it dubious whether UMSs can consistently make visual discriminations (Sharkey, 2010). Friendly human soldiers or civilians can be hurt by UMSs that mistake them for the enemy.

Given the current state of investment into military robotics, which is carried out by more than 40 nations and with a budget of 40 billion in the USA alone (Krishnan, 2009, p. 11), such technological limitations are likely to soon be eliminated and some of the shallow problems with them. We should expect computer vision, for one, to eventually surpass human vision, even in the task of discriminating civilians from enemy combatants. We should also expect UMSs to become ever more dependable and free from malfunction. So goes the curve of progress.

Any *long-term* predictions of the development of UMSs, both positive and negative, are subject to the criticism that they fail to address the imminent tactical danger of an enemy army deploying UMSs first. That leaves the *short-term* considerations of tactical superiority effectively trumping any *long-term* considerations. Given this, a persuasive argument against the deployment of UMSs, should address *short-term* consequences of their development and deployment.

The aim of this paper is to expose a new ethical problem for UMSs that recommends halting their development in the *short-term*, no matter the probability of any *long-term* predictions coming true. This problem is the result of the fact that as the AI that runs UMSs gets more sophisticated, it will simultaneously become more vulnerable to being compromised and hijacked.

This problem falls in neither the shallow nor the deep category. It is not a deep problem since it cannot be addressed by purely philosophical analysis. This problem is also not eliminable with technological advancement, as most shallow problems are. In fact, it is exacerbated by such advancement. This challenges some of the received wisdom about UMSs being a *short-term* panacea to the political, military, or moral ills of war.

2 The Moral Justification for Unmanned Military Systems

Military training stresses following orders and discourages autonomous decision-making, especially in tactical situations. This is generally thought to be a good thing, since an officer's training and distance from battle grants them a better position to assess the tactical situation. As a consequence, moral responsibility for what happens in combat lies largely with officers.

This also exempts individual soldiers from being held morally responsible for much of what happens in theatre of war. Moral responsibility for civilian casualties or collateral damage to friendly units usually will fall on the officers, not on the individual soldiers that are "just following orders." Unfortunately, things do not always work out this way.

Human soldiers are subject to a number of psychological factors that render their behavior unpredictable. They can become emotionally disturbed, suffer from battle fatigue, or simply decide to act outside of the chain of command. This can lead to war crimes, civilian casualties, or friendly fire. If that happens, then the individual soldier bears moral responsibility, not the officer.

Ronald Arkin has argued that autonomous robots running appropriate software are the answer to this kind of moral danger (Arkin, 2010). If an UMS were to have a sort of "ethics module," which implements, say, Immanuel Kant's categorical imperative or the principles of Utilitarianism, then we could expect it act within the bounds of morality. With UMSs in the field, we could expect fewer war crimes, fewer civilian casualties, and fewer friendly fire incidents.

Indeed, UMSs developed along Arkin's suggestion would not be subject to psychological stress or negative emotions, would be completely predictable, and completely unable to act on their own or go outside specified rules of engagement. Hence, Arkin's claim is that eliminating as many soldiers as possible and replacing them with UMSs lessens the moral dangers associated with war. The upshot is that UMSs will lead to *short-term* moral, political and military advantage, which in turn can lead to a better *long-term* outcome.

This is just one of the ways in which using UMSs in war can be thought to be morally superior to using human soldiers. And it is a compelling argument, but it is not without problems. Do we have any reason to reject it?

3 The Frame Problem

To carry out the kind of reasoning that Arkin and others envision, UMSs have to first solve the philosophical frame problem of AI, which has been formulated in a number of ways (Dennett, 1984; Fodor, 1987; Ludwig & Schneider, 2008; Pylyshyn, 1987). The gist of the frame problem is "Hamlet's problem: when to stop thinking" (Fodor, 1987, p. 140). A solution to the frame problem would consist of an engineering or programming solution that enables the AI that runs the UMS to hone in on only relevant information in every context.

Daniel Dennett illustrates the frame problem with an evocative example, which I will rehearse below. Consider the autonomous robot R1, which has the task of extracting a battery from a room in which there is a bomb. R1 has the command PULLOUT(wagon, room, t) that, when executed, makes the robot drag a wagon out of a room at time *t*.

So, R1 enters the room, sees the battery on the wagon and then executes the PULLOUT command. Shortly, the wagon and the battery are out of the room and the task is finished. Then the bomb explodes, destroying the battery and the hapless R1, which failed to consider that the bomb was also on the cart.

This leads engineers to create R1D1, which, in addition to the PULLOUT command, also has a program that models future possibilities. R1R1 is thus:

> made to recognize not just the intended implications of its acts, but also the implications about their side-effects, by deducing these implications from the descriptions it uses in formulating its plans (Dennett, 1984, p. 129).

The engineers' idea is that such software will prevent R1D1 from making the mistake of rolling out the battery together with the bomb because it will deduce the negative side-effects of doing so.

Sadly, this does not work. R1D1 is consumed with deducing all the possible implications of its actions. It deduces, for example, that rolling out the cart will not change the color of the walls and that the path out of the room will cause the sum of the revolutions of its wheels to be greater than the number of wheels it has. Consequently, before R1D1 figures out that it should take the bomb off the cart, the bomb explodes and destroys the battery.

R1D1's problem is that it cannot limit the space of possibilities to just the relevant ones. If that space was well-defined, as it is for, say, a chess program, then deduction via rules of inference would lead to the expected result in a relatively straight-forward way. But the space of possibilities is not antecedently defined at all and R1D1 does not know which inferences are relevant.

This leads to the construction of R2D1, which has the additional ability to identify a space of relevant possibilities. However, faced with the battery and bomb task, R2D1 spends all of its time identifying what is *not* relevant to the task it is about to perform. So, R2D1 is frozen in Hamlet-like anticipation and the bomb explodes.

All of the robots fail for the same reason: they have no way of getting at what is relevantly important without a human explicitly telling them so. This is the gist of the frame problem. This very problem of relevance will arise for UMSs. A fluid and changing battlefield creates indefinitely many moral and tactical possibilities to consider. In such an environment, the UMS endowed with an "ethical module" or not, will sit idly, like R1D1 or R2D1, as it examines all the implications of what it is about to do.

On the other hand, if UMSs are programmed to act without exhaustively deducing consequences, they are much more likely to act rashly like R1 and cause harm to non-combatants or cause collateral damage. The frame problem suggests that UMSs are either useless, because they cannot make the relevant inferences, or add an additional moral danger to the battlefield, because they will be rash. Consequently, Arkin's moral argument for UMSs fails at the crucial premise that they will never act outside the chain of command.

Human soldiers know when to stop thinking. In general, organic intelligence has no problem sorting out what is relevant and limiting search. So, we have reason to reject the argument that UMSs will lessen the moral dangers of war in the *short-term*.

4 Responses to the Frame Problem

Not everyone is convinced that the frame problem is insurmountable. One possible way out of it is to note that it affects classical von Neumann-style computers, which use symbolic representations and rules of inference. But computation is a wide notion (Shagrir, 1997), which includes non-symbolic associative transitions in artificial neural networks (Bechtel & Abrahamsen, 1991) and changes in dynamical systems (Beer, 2000). These alternative models of computation might not be affected by the frame problem in the same way as classical models (Horgan & Tienson, 1994).

A full account of the debate about the virtues and follies of non-classical computation would lead us too far afield from the issues central to this paper. What should be noted, however, is that at present moment cognitively sophisticated AIs that use alternative models of computation are a promissory note (Addyman & French, 2012). It is also at least controversial whether such systems will ever be able to do higher-level cognition, abduction, or scale up in the relevant way to more difficult tasks (Fodor, 2001, pp. 41-53; Morsella, Riddle, & Bargh, 2009).

More promise for UMS AI lies with hybrid systems, which mix bottom-up and top-down processing (Lin, Bekey, & Abney, 2008, pp. 38-41). In cognitive and neural sciences, one popular hybrid model is the global workspace, which was developed to help distinguish conscious and unconscious brain processing (Baars, 2002; Dehaene & Naccache, 2001). On this view, human cognition involves massively parallel unconscious processing that feeds information to a conscious global workspace.

The global workspace model, just like other hybrid models, provides a framework for a solution to the philosophical frame problem (Shanahan & Baars, 2005). The idea here is that distributed networks will first process input in a fast, massively parallel way, sorting information in accordance with its relevance to current goals. Then, the output of that process will be passed on to the workspace, where a symbol-based process will use it to make deductive inferences.

While hybrid architectures might be a route towards a cognitively sophisticated AI, it will not be of much help to UMSs. Just as with non-classical computational models, hybrid architectures, such as CLARION, are not fully developed (Sun & Helie, 2012). UMS development and deployment will likely outrun the pace of development of this kind of software.

Of course, these speculations might be wrong. AI research in military robotics is secret, so the new UMSs might already have solved (at least approximately) the frame problem; perhaps even in one of the ways mentioned above. While it is hard to be certain about what such a solution might be like, we can make some safe predictions.

First, it is highly unlikely that this solution involves a simple algorithm. No such magic algorithm exists. Secondly, what is more likely is that this frame-problem-solving UMS is running an enormously complex AI. Presumably, this program sorts the relevant from the non-relevant pieces of information using a large number of complex search algorithms designed specifically for each of the possible domains of information that an UMS might have to sort during a mission.

So let us assume that such an enormous program comes (or has come) to exist. Let us further assume that it results in a practically viable approximation to a solution of the frame problem. Will UMSs now become (approximately) perfect moral soldiers? Will there be less moral danger on the battlefield than there is with human soldiers, as Arkin suggests?

Sadly, no. The deployment of military frame-problem-solving UMSs creates imminent moral dangers and hence does not carry the typically cited *short-term* political and tactical advantages. I turn to those dangers next.

5 Software Complexity and Computer Security

Most people that have written software in a serious way have heard of Murphy's Laws of Programming, which are tongue-in-cheek observations about the toil of writing lots of code. These laws have the virtue of also being rules of thumb for successful quality control of software. The first and most cited law states that "a working program is one that has only unobserved bugs." In other words, all programs have bugs.

Bugs are not malfunctions. They are errors in the logic of the program itself, which are typically undetectable, except in very specific circumstances and typically only during the execution of the program. Even the best programmers sometimes write buggy programs.

There are many examples of Murphy's first law being vindicated in horrific ways. Among the most famous are the tragic accidents in the Therac-25 radiation therapy device caused by a race condition bug (Leveson & Turner, 1993), the explosion of the Ariane 5 rocket caused by reusing old code (Jazequel & Meyer, 1997), and the floating point truncation bug in the Patriot missile battery that was supposed to protect a Marine barracks (Marshall, 1992). There are many other examples.

Importantly, there is a strong correlation between a program's complexity and the amount of bugs that it has (Khoshgoftaar & Munson, 1990). Complexity can be measured by the size of the program, the sophistication of the algorithms used, and the number of features the program has, all of which can be helpful in predicting the possible number of bugs (Shivaji, Whitehead, Akella, & Kim, 2009). In short, the more complex a piece of software, the more buggy it is.

As mentioned in the previous section of this paper, the program that will make it possible for UMSs to overcome or approximately overcome the philosophical frame problem is likely to be astoundingly complex. We can therefore expect the AI that runs UMSs to have lots of bugs. Some of these bugs might be benign and some might even be caught during testing. But some may put people's lives in danger.

There is the direct way in which bugs in a UMS can cause harm. The UMS is programmed to do X, but, alas, a bug in its software makes it do Y. The UMS might fire its weapon too soon or too late, say, thus causing an accident.

The possibility of an UMS firing when it is not supposed to is disconcerting, but would be rare. With time, such bugs would be discovered and eliminated. Even if UMSs were to have bugs that cause accidents, their rarity would likely have minimal tactical and political consequences.

However, accidents are not the only moral danger of buggy software in UMSs and also not the most serious. In computer security, bugs are typically considered to be vulnerabilities (Krsul, 1998). Such vulnerabilities can be exploited by another program, which can cause an UMS to do something other than what it was designed to do on a regular basis, not just accidentally.

The tactical and political consequences of this are easy to imagine. The most frightening possibility, of course, is that of an UMS being completely hijacked and made to do someone else's bidding. Just imagine the consequences of an army of hacked UMSs controlled by a criminal or terrorist organization.

Such scenarios are made more probable by the fact that the software that runs UMSs will be very complex and hence likely to have a large number of vulnerabilities. This means that UMSs are more likely to have a third-party change their behavior or hijack them than any other military system. The irony of this situation is that UMSs' approximate or actual solution to the philosophical frame problem—no mean feat—exposes them to the potentially lethal problem of being hacked.

6 Can We Make Unmanned Military Systems Safe?

One way of responding to the argument of this paper is to claim that UMSs can be made sufficiently secure. Presumably, the idea here would be to create some kind of hardware or software firewall along the lines of Anti-virus software on PCs. An additional route might be to overhaul computer safety protocols (Bakx & Nyce, 2012).

A corollary of this approach would be to make sure that computer and information security development always outpaces UMS development. There would be a priority of sorts for developing security software and protocols before deploying new UMSs. The optimism of this response hinges on the assumption that complete computer security, whether by technological means or by social construction, is difficult, but not impossible.

The pessimist might demur that this response is naïve (for review: Gollmann, 2010). There are reasons to be very pessimistic, at least in the *short-term*. First of all, almost every part of currently existing military computer infrastructure has been compromised at some point (Lynn III, 2010). That is counting just the attacks that we know about, of course.

Closer to home, Predator drones have been regularly hacked by militants in Iraq, who then recorded the drones' camera feeds (Gorman, Dreazen, & Cole, 2009). It is even possible that the famous downing of an RQ 170 stealth drone in Iran involved direct hacking by Iranian intelligence (Shane & Sanger, 2011). If this is the state of computer security with unmanned aerial vehicles currently in service, then optimism about future improvements for much more sophisticated systems is merely wishful thinking.

Especially troubling in this context is the current militarization of computer hacking and information warfare (Taddeo, 2012). In the near future, we can expect professional hackers in all major intelligence and military establishments. In such a world, no UMS will be safe.

The argument of this paper depends on the claim that as military systems become more autonomous, their software will become more complex, especially if such systems will be expected to solve or approximately solve the philosophical frame problem. That in turn will lead to vulnerabilities and compromised security, which will render them more open to hacking and hijacking. Hacking during war is likely to cause lots of harm. Consequently, the only viable way to prevent *short-term* moral, tactical, and political fallout from compromised UMSs is not building them at all. This is both the moral and prudent thing to do.

References

Addyman, C., & French, R. M. (2012). Computational modeling in cognitive science: A manifesto for change. *Topics in Cognitive Science, 4*(3), 332-341.

Arkin, R. C. (2010). The case for ethical autonomy in unmanned systems. *Journal of Military Ethics, 9*(4), 332-341.

Baars, B. J. (2002). The conscious access hypothesis: origins and recent evidence. *Trends in Cognitive Sciences, 6*(1), 47-52.

Bakx, G. C., & Nyce, J. M. (2012). Social construction of safety in UAS technology in concrete settings: some military cases studied. *International Journal of Safety and Security Engineering, 2*(3), 227-241.

Bechtel, W., & Abrahamsen, A. (1991). *Connectionism and the Mind*: Blackwell Oxford.

Beer, R. D. (2000). Dynamical approaches to cognitive science. *Trends in Cognitive Sciences, 4*(3), 91-99.

Dehaene, S., & Naccache, L. (2001). Towards a cognitive neuroscience of consciousness: basic evidence and a workspace framework. *Cognition, 79*(1–2), 1-37.
Dennett, D. C. (1984). Cognitive wheels: The frame problem of AI. In C. Hookway (Ed.), *Minds, Machines and Evolution*: Cambridge University Press.
Fodor, J. A. (1987). Modules, Frames, Fridgeons, Sleeping Dogs, and the Music of the Spheres. *The Robot's Dilemma: The Frame Problem in Artificial Intelligence*, 139.
Fodor, J. A. (2001). *The mind doesn't work that way: The scope and limits of computational psychology*: The MIT press.
Gollmann, D. (2010). Computer security. *Wiley Interdisciplinary Reviews: Computational Statistics, 2*(5), 544-554.
Gorman, S., Dreazen, Y., & Cole, A. (2009). Insurgents Hack US Drones. The Wall Street Journal.
Horgan, T., & Tienson, J. (1994). A nonclassical framework for cognitive science. *Synthese, 101*(3), 305-345.
Jazequel, J.-M., & Meyer, B. (1997). Design by contract: The lessons of Ariane. *Computer, 30*(1), 129-130.
Khoshgoftaar, T. M., & Munson, J. C. (1990). Predicting software development errors using software complexity metrics. *Selected Areas in Communications, IEEE Journal on, 8*(2), 253-261.
Krishnan, A. (2009). *Killer robots: legality and ethicality of autonomous weapons*: Ashgate Publishing, Ltd.
Krsul, I. V. (1998). *Software vulnerability analysis.* Purdue University.
Leveson, N. G., & Turner, C. S. (1993). An investigation of the Therac-25 accidents. *Computer, 26*(7), 18-41.
Lin, P., Bekey, G., & Abney, K. (2008). *Autonomous military robotics: Risk, ethics, and design*: DTIC Document.
Ludwig, K., & Schneider, S. (2008). Fodor's challenge to the classical computational theory of mind. *Mind & Language, 23*(1), 123-143.
Lynn III, W. F. (2010). Defending a New Domain-The Pentagon's Cyberstrategy. *Foreign Aff., 89*, 97.
Marshall, E. (1992). Fatal error: how patriot overlooked a scud. *Science (New York, NY), 255*(5050), 1347.
Morsella, E., Riddle, T. A., & Bargh, J. A. (2009). Undermining the foundations: Questioning the basic notions of associationism and mental representation. *Behavioral and Brain Sciences, 32*(02), 218-219.
Pylyshyn, Z. W. (1987). *The robot's dilemma: The frame problem in artificial intelligence* (Vol. 4): Ablex Publishing Corporation.
Shagrir, O. (1997). Two dogmas of computationalism. *Minds and Machines, 7*(3), 321-344.
Shanahan, M., & Baars, B. (2005). Applying global workspace theory to the frame problem. *Cognition, 98*(2), 157-176.
Shane, S., & Sanger, D. (2011). Drone crash in Iran reveals secret US surveillance effort. *New York Times, 7*.
Sharkey, N. (2010). Saying 'no!'to lethal autonomous targeting. *Journal of Military Ethics, 9*(4), 369-383.
Shivaji, S., Whitehead, J. E. J., Akella, R., & Kim, S. (2009). *Reducing features to improve bug prediction.* Paper presented at the Proceedings of the 2009 IEEE/ACM International Conference on Automated Software Engineering.
Sun, R., & Helie, S. (2012). Psychologically realistic cognitive agents: taking human cognition seriously. *Journal of Experimental & Theoretical Artificial Intelligence*(ahead-of-print), 1-28.
Taddeo, M. (2012). Information warfare: a philosophical perspective. *Philosophy & Technology, 25*(1), 105-120.