AFRL AUTONOMY TRANSCRIPT PART 5

Jim Overholt
Hi! I'm Dr. Jim Overholt, senior scientist for autonomous systems at the Air Force Research Laboratory, and today I'm joined by Ms. Kris Kearns who's the portfolio manager for autonomous system across AFRL, and together we're going to talk to you about the Air Force Research Laboratories autonomy science and technology strategy.

Kris Kearns
This is our final goal that we're going to talk about. What we've done so far I think is we laid out the overall vision and strategy, of what we're talking about how we break those down into goals. Today, we're going to dig down deep into our fourth and final goal. That is very important, but before we do that let's give you an introduction and make sure we're all on the same page as far as what is our vision and what are the goals we've talked about so far. So, our vision is really about intelligent machines seamlessly integrated with people, and we like to say there are a couple of things that are important in that. It's the intelligent machines. So we are talking about machines that have the capability to understand their environment and make decisions about that, and what should they do, but they're seamlessly integrated with people. So we're not talking about taking people out of the mission, but how do we put machines that are intelligent together with people so that we can maximize the performance of the mission in complex contested environments that we the military have to operate in. So when we came up with our vision we said what are the major goals, what are the major challenges we're going to have to address in order to achieve that kind of a vision, and we broke that into four goals. Like I said we talked about the previous three in previous recordings. But just briefly they are, how are we going to demonstrate these highly effective human-machine teams. How are we going to have people and machines working together? The second goal was about how are we going to have coordinated teams and machines. So just like we have people and machines working together, we want machines and machines to be able to work together, coordinate and do work. How do we make sure they can operate in the complex, contested environments that we the military in particular will have to operate in? Which doesn't preclude nonmilitary environments, but it's a lot focused on our military environments. And then finally, the one we're going to talk about today is how do we ensure safe and effective operations. In other words, how are we going to test and evaluate, verify and validate, systems that make decisions and can have the capability to make decisions in environments where their unanticipated in their dynamics so that we don't really know what exactly the environments going to be, and it's going to be changing when we get there.

Jim Overholt
And these are systems that are heavily or intensive software involved with them. So yeah this area is something that's really critical for us to be able to roll these systems out.

Kris Kearns
So just like our other four goals. What we did is we took this one and said, so they're some long term and enduring problems. There are some research challenges that we're going to have to take on. They're not quick fixes and their not things that we can resolve and move on. So what are the enduring problems that we're going to have to investigate, and work towards in this test and evaluation, verification, and validation? So we break those into, actually on this one its four areas. The first area is about making sure that we have the assurance technology so that we can ensure the machine intelligence and the decision making is correct. So, it's about making sure that system is making the right decision. There's a component then of how do we make sure that the architectures are easily verified and validated, so that we cannot spend decades making sure that these things are making the right decisions. So underlying in that, is that whole how do we make sure we have rigorous and verifiable architecture systems. Tied into that though is also, we want to make sure that our systems are making decisions on the

right data. So there are people out in the world who would want to scramble the ones and zeroes of data that a platform is pulling in and making decisions on. We want to make sure when that system takes in that sensor data, that it can verify that this is valid data. Someone isn't sending me a different command that isn't what I'm supposed to be doing. And so we want to make sure they have that underlying security in that framework, and to make sure that it is making decisions on solid data. Previously we've talked about we want a single intelligent machine, but we also want to incorporate machines together, and teams of machines to do things. Once we do that, then were also going to have to have verification and validation strategies on how do we make sure that as teams and machines work together, both as teams, fractionate and then recompose to do things, that they're not displaying any emergent behavior that we wouldn't want them to have. And so that's another one of our enduring problem areas. And finally, because we talk about how do we seamlessly integrate machines and people. We have to be able to test, we have to be able to verify and validate that when we bring people and machines together, that the communication and the interaction is optimized. In other words, we put a person and a machine together, ideally we'd like to see one plus one is greater than two. How are we going to test and verify that actually one plus one is greater than two? And so, these are all the long term challenges under testing evaluating, verifying, and validating that these systems.

Jim Overholt
And those are significant challenges. Every single one of them.

Kris Kearns
Right, so then we take each one of these and we talk about what are the key challenges. What is the key challenge? We haven't really said that testing and verifying, why is this a hard problem?

Jim Overholt
Yeah, and you know when you look at that, so, again we're talking about these extremely highly complex software intensive problems. So, you cannot wait until the very last roll out of the software to do the kind of testing that we need to do. So the fact is that we're taking a multiple number of systems all with their own capabilities, so we call them modules if you want to, or call them subsystems, there is software associated with that, and as we start adding these systems and they start having more and more effect on each other. You can imagine the testing and evaluation could get really complex. And, we can't possibly test for all conditions that these systems are going to have. So you have to come up with some kind of methodology that allows you to test for the, if you want to say like the design of experiments, these kind of things. These are one of the really important features the fact you can't test for everything. And you touched on a real important part too, which really is a fear, is something the community is worried about, this notion of emergent behavior. As these systems get so complex, and mostly from a mathematic standpoint we would call they're nonlinear. They don't have real nice smooth behavior that we can predict necessarily. And people are worried about this notion of emergent behavior.

Will something come out of this system that's unanticipated, and will perform in such a way that it does harm, or it significantly degrades the performance of the autonomous system. So these are some of the kind of things. The cost and the test, and the cost of software, looking at the fact that we can't possibly expose our systems to every possible state or situation it's going to see, and this worry, and this concern, that unanticipated undesirable emergent behavior is going to occur. So these are big-big challenges so to speak, or issues right now, with highly complex autonomous systems that we need to solve in order to roll this technology out to the war fighter, to the Airman.

Kris Kearns
Right, and when we say complex systems, what we are really, I think, talking about are, these are all software systems.

Jim Overholt
They're mostly software with mechanical, obviously electro mechanical systems.

Kris Kearns
And there are, I think you were telling me, there are systems already out there that have hundreds of thousands of lines of code already.

Jim Overholt
Millions, in many cases.

Kris Kearns
So making sure that, that code got all put together, and as it was developed in pieces, how do you make sure that when the pieces come together, that they're communicating and they're sharing data and everything is fluid, and not creating any disturbances in the behavior of the system.

Jim Overholt
Certainly it's not necessarily our background, but there are people obviously involved with the notions of formal software, and being able to roll out this kind of software. So certainly were going to look to that community in order to help us solve these kinds of problems, especially when you have a system potentially making this decision on its own.

Kris Kearns
And what we know is in today's environment the way we the Air Force test our systems is essentially what we do, is say here's the performance envelope, and we test right up to the performance envelope to make sure the systems are going to perform the way they are supposed to. When we start talking about dynamic unanticipated environments, first of all, I can't set that performance envelope.

Jim Overholt
You can't! That's exactly, that's an infinite state.

Kris Kearns
It's unanticipated, right? And even if I could now it's also changing. So how do we test using the current regulations that we have in the Air Force? I think we've kind of come to the conclusion we can't, we're going to have to figure out other ways to, be able to provide the evidence, to provide the confidence, and to instill the right level of trust that these systems are going to behave. They're going to perform the way they're supposed to, and probably most importantly, they're not going to perform in a way that we don't want them to.

Jim Overholt
Absolutely.

Kris Kearns
And so I think, knowing that, we've started to dig into this problem. We've dug into and starting to identify, I think we have five areas that we kind of say these are technology development areas that we think we're going to have to invest in, to be able to eventually be able to field systems that make decisions.

Jim Overholt
And Kris, let me just, before you go into talking about those areas. It's really clear within the DOD right now, that this is an important feature, but we're not necessarily putting the resources in order to tackle this problem. You know, compared to the other things we talked about whether it be, the human machine

teaming, or the machine to machine teaming. Certainty we're putting in the resources. Here at AFRL within the Air Force, we realize how important the problem is to this. So a formal activity and really getting down into the science and technology which is very unusual for looking at things like you said, at testing, evaluation, validation, verification of these highly complex electro-mechanical software systems that make decisions on their own.

Kris Kearns
So I think what we're exploring is, is like I said, that if you look at the way do things today, we test it. Are there other things that we possibly could do, in addition to testing it, to be able to provide the data, the confidence all the evidence that would be needed to say,

Jim Overholt
Simulations.

Kris Kearns
This system is going to behave. And so one of the first things that I think we would identify, is we would say, what we would really like to do is do a better job of accumulating the evidence from the development all the way through what we call developmental test, through the operational test that we do before we certify a platform to go.

Jim Overholt
Great Idea!

Kris Kearns
So why not find ways, especially when we're talking about software, and we're talking code development. Is there a way to cumulatively create the evidence that is required to show that the system is going to behave, it's going to do the things, like I say most importantly, it's not going to do the things we don't want it to do?

Jim Overholt
Right, I agree.

Kris Kearns
So part of that starts with can we also then create the evidence during design.

Jim Overholt
Ah Yes!

Kris Kearns
That says we have designed this system. We have put in the tenants and the hooks to be able to ensure that it's going to behave. We make it testable, as we design it, so that you can start exploring that operational space early.

Jim Overholt
Like test often, right, as you're designing, why not test as you're designing.

Kris Kearns
Right, and put the hooks in there, and make it be able, and the code, that you can just immediately kind of run a test quick and be able to do things like, for instance something as simple as is, if it's a flying platform. If a platform is going to go this way, put a shadow ahead so that you can see what decisions it's

getting ready to make, you can validate in real time that it is going to make the decisions you want. Which kind of leads us to one of our areas, which is how do you do run time monitoring? Can you in real time, as a system is performing doing its thing, can you monitor what's going on? What decisions it's made? What evidence, what sensor data its taking in that led it to those decisions? And in real time monitor its decisions. So that if it's going to do something that you don't want it to, you have the opportunity to interfere and redirect it.

Jim Overholt
That's an important feature, in some ways you can think about it like an outer critic or an outer loop in a way that looks at the decisions being made, and try's to if you want to call it reign in, or correct for things that maybe start going out of bounds of the kind of things that we wanted it to do.

Kris Kearns
In my simple mind it's almost like being a parent, right? When you have a little child, you're pretty much directing them and guiding them along the way as they grow up you give them a little more leeway, but you're still monitoring them. Eventually, they need to be able to work on their own, but there is a time during their development that your kind of, if they are heading off in the wrong direction, you have the ability to re-steer them and say no let's talk about that decision.

Jim Overholt
Not my kids.

Kris Kearns
Why did you make that decision? And maybe we can help you readjust your thinking process; so that now you can make a decision that maybe we think is a little bit more appropriate in that situation.

Jim Overholt
Excellent, that's an excellent description.

Kris Kearns
One of the other areas that we've identified, that we talk about is we need to do better about creating the standards up front. So that will allow us to do the test, and allows us to do all the traceability that we've talked about.

Jim Overholt
It's a great point, I mean standards sometimes and the requirements for these things, they get so complex, and being able to check that the requirements themselves are not conflicting. So trying to work a design where you'd have two different requirements that have come together, that are actually asking the system to do two different things at one time. That's a critical area, and surprisingly not necessarily a lot of work in that area.

Kris Kearns
And then finally, one of the things we need to get better at least from a DOD perspective is be able to reuse the evidence. So if I have created an algorithm and I have created a certified platform and now I want to add in something new do I really need to go back and recertify the whole thing, or is there a way that I can certify the component, do a quicker recertification of the pieces and parts, and be able to get to reuse the evidence that I previously had.

Jim Overholt
The previous evidence right, it almost sounds like you're making a case in a court of law.

Kris Kearns
Exactly, right. So if we can start thinking about our V&V process, our verification and validation process for these things. Similar to, I am a lawyer and I'm trying to defend my autonomous system.

Jim Overholt
Don't look at me okay.

Kris Kearns
What is the evidence? And what do I have to be able to go into court and be able to argue?

Jim Overholt
That's a great point.

Kris Kearns
We've talked about it as very simply as, so we have to be able to verify that the rules existed.

Jim Overholt
YES.

Kris Kearns
We have to know the requirements. Secondly, did we make sure that the system knew the rules? Were they able to play the game, were they able to perform the way they were supposed to. So what evidence during the design did we put into place so that they can, we can validate that there were rules, they knew what the rules were. And then you go through the whole evidence of precedence.

Jim Overholt
Precedence, exactly.

Kris Kearns
So in other instances this is how the court ruled, here's what happened, here's what the decision was, so you can pull in that.

Jim Overholt
That's a fascinating metaphor

Kris Kearns
Reusability and we can start building that evidence case that kind of analogist do to developing a court case to be able to argue that this system is validated to be safe. And we can verify that through all the evidence that we've created in arguing our case.

Jim Overholt
That sounds good. I like that approach. And I, and from both of our experiences, it sounds like it's a very unique and a very fruitful approach to go forward with trying to do this kind of test evaluation validation verification of these very complex autonomous systems.

Kris Kearns
And I think we know we have to do something unique here. Because if you look across the autonomy community this is one of the big challenges and a lot of people through their arms up and say this is a hard problem, but you don't see a whole lot of solutions on how to solve it.

Jim Overholt

Absolutely.

Kris Kearns
So we're hoping that we'll be able to put together a structure and some focus areas that will allow us to at least make progress. If not, substantially come close to being able to solve it so that we can get our systems certified for use.

Jim Overholt
And at least give some people some thought that talking about, it'll give them some pause to think about it and obviously potentially get back to us on some of the ideas and how they, what they think about this notion of this court case metaphor in order to go forward with autonomous systems.

Kris Kearns
And so with that I think.

Jim Overholt
We're done.

Kris Kearns
We've shown you, we've talked to you about what our strategy is. We've given you an overview. We've talked about each of our goals. We will come back and answer questions.

Jim Overholt
And we look at this as somewhat of a living dynamic document certainly as we get more information we'll revisit this every once in a while and come back and change what we need to change and update certainly where we are in terms of the science and technology of autonomous systems. And we welcome feedback on this and your comments, certainly. Thanks Kris, really appreciate it.

Kris Kearns
Thank you, great!

Jim Overholt
Bye!