

Philadelphia PA
1:12:15
m4v

Max Mintz

An interview conducted by
Peter Asaro
with
Selma Sabanovic

May 24 2011

Q: I'm just going to ask you to start with a little bit of your background on where you were born, where you grew up, and how your education started.

Max Mintz: So I'm a child of World War II, and I was born essentially three months after the Battle of Midway, on September 4th, 1942, and it had – the war, of course, had profound implications for the world, but for me, in a very selfish way, putting aside all the geopolitical implications, indeed, the way the war turned out, if it had turned out differently, we wouldn't be here, having this interview. So that itself is a point to ponder. But what was really important for me was the fact that when the war ended, it unleashed on a civilian marketplace an enormous amount of war surplus radio equipment, and as a 10-year-old, okay, growing up outside New York City, I could go down to a place called Radio Row, which was on a street called Cortlandt Street, where they eventually built the World Trade Center, and I could go into this warren of dusty buildings and buy dirt-cheap war surplus radio equipment. Now, it had the following implications: It was all vacuum tube electronics, all right? And that was very important, it turns out, because it made it possible for a 10-year-old to understand the underlying physics, all right, that explained how any of these devices worked. Now, admittedly, it's probably hard to believe, but I used to have hair on the top of my head, and when I had hair on the top of my head, I could take a comb and run it through my hair, as a 10-year-old, and I could charge the comb with a static charge, and I could let water flow out of a faucet, gradually, and cause the water to deflect, due to the static charge on the comb. And this is basically the principle on which electric fields interact with matter, and how one can think about how a vacuum tube works – a vacuum tube amplifier – where you have a weak electric field controlling a strong current, and you get amplification. So, as a 10-year-old, I learned the fundamentals of radioelectronics, and as time went on, I got a radio amateur's license and went on to – got more and more interested in electronics, and went off to Cornell at age 18, and studied electrical engineering. So I was at Cornell for eight years, and I got my bachelor's, master's, and Ph.D. at Cornell, and then I went off to Yale, where I had a postdoc, and then eventually an assistant professorship for a couple of years, where I taught control theory and electrical engineering subjects, and I did the same at the University of Illinois for a couple of years, till I came here in 1974, where I've been ever since. So I'm starting my 38th year here at Penn, and I started out in what is now called the Electrical Systems Engineering Department. It was called something else several years ago. And eventually, in meeting a person who became very pivotal in my life, Ruzena Bajcsy, she enticed me to join the GRASP Lab and turn my interest in control theory and decision making under uncertainty in the direction of robotics. So –

Q: When was that?

Max Mintz: And that would be 1984. So I joined the GRASP Lab as a professor in Electrical – in what was then Systems Engineering, now Electrical and Systems Engineering, and two years later I actually joined the Department of Computer and Information Science, and switched my departmental affiliation. So in 1984 I joined the GRASP Lab, in 1986 I became a professor in

the CIS Department, and I've been doing robotics and related research in decision making under uncertainty – “machine perception in robotics” is the way I refer to it; decision making under uncertainty, I should say – in machine perception in robotics, and that's what I've been doing, essentially, ever since.

Q: What was your Ph.D. work on?

Max Mintz: My Ph.D. work was in, essentially, control theory with stochastic systems. I was doing work in stochastic control theory, which is something we use today in robotics, but we use in a lot of other subjects, as well.

Q: And your postdoc research was...?

Max Mintz: My postdoc was all related to that. In fact, until I got involved in the GRASP Lab, all my stuff was basically doing control theory – the mathematics of control theory.

Q: And so what years were you at the University of Illinois?

Max Mintz: Nineteen seventy-two to nineteen seventy-four. I was with the CSL group in the Department of Electrical Engineering.

Q: Did you ever run into Heinz von Foerster in the Biological Computer Lab?

Max Mintz: I certainly know who he was, but I never did get to meet him. Yes.

Q: I did some work on that on my dissertation. So what would you say was your first robotics project?

Max Mintz: Well, I – to understand my interaction – my – getting involved with projects in robotics, one understands that one – I did it collaboratively. I can sit and do my own thinking about mathematics and such, but when you want to do things with robotics, it's not a bad idea to have some extraordinary graduate students, okay? And the GRASP Lab, in some sense, is the home of extraordinary graduate students, and I'll name a few, okay, who I had the pleasure to interact with, and it was my good fortune to have this interaction. So, first was a guy named Greg Hager [Gregory Hager], who's now a professor in computer science at Johns Hopkins, and Greg got his degree, as I recall, I think it would've been 1988, here at Penn, and I – as I say, I got started in the GRASP Lab in '84, but as I – when I joined the department in '86, I started

supervising students, okay, and Greg was my first Ph.D. student in CIS. Following that, okay, there were students who did work either in robotics, or worked in decision making under uncertainty. So I'll say a little bit about what Greg's work was about, and I say it's "his work." My job in life, as I tell people, is to supply the applause, okay? The students do the brilliant work, and I get to make hopefully helpful suggestions from time to time, but the students are the ones who are really quite pivotal in making all this happen. So Greg – I'll try to summarize it briefly. Greg's work involved seeking information. In other words, we have sensors in the world, and we gather information through sensors, and one really can't do robotics today without thinking about sensors, okay? Sensors are an extraordinarily important part of the story, and the importance of sensors has grown, I would say, exponentially in the time when we started doing a lot of this work, in the mid-eighties, okay, to the present date, where now sensors are everywhere in sight, okay? And not just in robotics; all over the place, all right? So that's a story for perhaps another day. But, in any event, Greg's research idea was to work out how we acquire information in an intelligent fashion by using sensors. So sensors don't want to remain static. You don't want sensors just to remain static in the world. You want sensors to be controlled in such a way that you go out and use current information to get even better information. So it's, "How do we search for information that's helpful for solving a problem?" If you're going to grasp an object, you need to know what the object looks like, and if you just look at the object from one position, you won't get enough information. So you're searching for, in an intelligent, efficient way, good information so you can achieve some kind of task solution, which would be, say, grasping an object, moving an object, or even just identifying an object by looking at it from various points of view. So Greg worked on that problem, and there's a lot of interesting statistical decision theory that goes into this, and he and I had a lot of fun working on this. Other students worked on actuation, where – Greg's – where there was robotic actuation, there was a robot arm with a camera, okay? So the robotics was a robot arm with a camera, okay, as opposed to a mobile robot running around in the GRASP Lab, achieving some kind of other task solution. So in the case of, say, Robbie [Robert] Mandelbaum, who graduated in 1995, he – Robbie's now the Chief Technology Officer at Lockheed Martin. After he graduated, he went to Sarnoff Research in Princeton, and then later left Sarnoff and went to DARPA, okay? And after he left DARPA, he joined Lockheed Martin, where he is presently. So Greg had a couple of postdocs, to go back to Greg Hager. He had a couple of postdocs, went to Yale. After he was – after he left Yale, he went to Johns Hopkins, where he is presently, in the Department of Computer Science. So Robbie would've been the first student I worked with where we actually had mobile robots, and I smile when I think about what mobile robots looked like in the early nineties to 1995, and it's no joke that we had very clunky platforms, okay? They were called TRC LabMates, and they were big, heavy, ponderous objects that could heft a very large payload, but were pretty hard to control. Indeed, the control for what we did with these things – or the students did, really – happened by having an off-board computer connected to the robot, using a serial line, okay? So to understand research then versus research today, the big breakthrough, the big change that has occurred in the nearly 20 years when this whole thing was going on, the fundamental word – two fundamental words: miniaturization and wireless. Okay, now things can be done with very small computers, okay? Powerful calculations are possible. Powerful computations are possible with very small devices, and enormous amounts of

information can be gotten with sensors, and all of this stuff can be done wirelessly, so now we don't have the embarrassment of having a serial line connecting a desktop computer to a mobile robot. And you can only laugh when you hear some of the things that would occur when the serial line ran out of slack, and the line in fact parted from the robot itself. It was, well, amusing in some sense, and not so amusing in other senses.

Q: Do you remember any particular cases?

Max Mintz: Well, if you promise not to publish it too widely, I'll say that I was near one of the robots when it happened, and they've – I had to, shall we say, be nimble to get out of the way of the machine and hit what's called the panic button, or mushroom, okay, to shut the thing down, okay, when the thing basically lost its computer control. So this was – this is something I'm not going to forget. Although I'm not proud of the moment, okay, it's an amusing moment. Now, today, as I say, everything has been miniaturized. If you go into the GRASP Lab, you'll see people working on devices that are very small, okay, that do amazing things that we could not have even dreamed of doing back in 19 – in the interval of 1990 to 1995. So you need to go into the GRASP Lab and see some of the quite wonderful things they're doing with these quadrotors that can fly through hoops, all right, and do really intricate maneuvers, all right? So the control, okay, is exquisite, and it's done wirelessly, and it's done with very intricate sensors, okay, that we couldn't imagine having – at least, *I* didn't imagine having, let me say it that way – in the interval 1990 to 1995. So, in some sense, my research program had been about decision making under uncertainty in machine perception and robotics, and that included how to understand how sensors work, how to understand the uncertainties in sensors, how to use different sensors together, okay, in a fashion so they complement one another, so that I get more information, okay, and it isn't just that I take the sum of the two, but I combine them in an intelligent way to learn more about the world, which is what I need to do, and to do so using the underlying fundamentals of probability theory and statistics in a reasonable way, so that – where we go out and actually measure the real uncertainties about the sensor behavior, as opposed to just making wild guesses. So the students in the lab, in that era, would tell you, "Don't talk to Max about Gaussian. He'll tell you you have to go and establish that the noise really is Gaussian, as opposed to just making a broad assertion." So I have a – there's a joke in the lab – it probably still exists, okay – that I'm the guy who was a pain in the neck, and would raise that point perhaps too often. So the story of decision making under uncertainty is important, but as sensors get better and better and better, okay, now the problem transforms to, "Huge numbers of sensors: How do you organize them in a clever way?" You don't want them all reporting to a single central processor. You want to do things in groups. You want to do things cooperatively. You want to do things independently. So there should be some intelligence embedded in one collection of objects, and intelligence embedded in another. They do their own thing separately, communicate where needed, okay, and the strategy of deciding when to communicate and what to communicate, these are still open research problems, all right? And what's happening is, as you get more and more of these sensors, all right, the problem becomes very, very, very, very – I wouldn't say – well, "complicated" is the right word, but requires a careful analysis of what

one's doing. In other words, one is not beating the problem to death with a hammer, okay? One needs to think cleverly about how to put these things together, and that's what the folks in the lab do today. I'm somewhat backing away from it. Truth be told, I suppose I can admit this to you: Although you're interviewing me about robotics, I've developed, over the last 10-plus years, a new love that has little to do with robotics; it has to do with quantum computation. So today I spend most of my time thinking about quantum computation and how I'm going to teach that to undergraduates, which I've been doing for over 11 years, and how I will continue to do this as it becomes possible, someday, to get quantum – *real* quantum – hardware, as opposed to just talking about mythical machines. So, in a sense, my interest has always been in mathematical physics; at a very simple level, how things work in a robotics environment. And I don't mean trivially simple. Robotics physics is not simple, because you have complex interactions with the world. Just to make a sort of a trivial point, when we first started getting little robots that would roll around the floor, and I was working with a guy who eventually graduated from Penn doing work in machine vision, but as an undergraduate and as a first-year graduate student worked with small robots, okay, we used robots that would roll on rugs in the lab. And the rugs – if you look in the rug in my office here, you'll see it has different behavior. It's not isotropic, and so there's a directionality to the weave. That affects the way the wheels would slip. And so the physics is by no means simple. The physics of the interaction is, in fact, quite sophisticated. One has to work on that to understand what that does to the odometry, because odometry is a place where one can get into trouble if one's trying to figure out where you – what position the robot is in, from merely the rotation of the wheels, where slippage could occur. So the slippage issue is a nontrivial issue, and we learned that through lots of good experiments, and try to understand what can go wrong. I guess, in some sense, my job in the lab were to break other people's experiments, and I mean that in a positive way, to find ways for things that might go wrong, find out where things – what they're sensitive to. The word that we used in those days, which now gets overused today, is the word called "robustness," okay? Robustness means that you want your system to work, in spite of your assumptions, okay, where you've made some assumptions. The assumptions may not be entirely right. You want the thing to gracefully degrade, as opposed to fall off a cliff when the assumptions are no longer valid. So robustness is something that we strive for in a variety of ways, and is still very important today, and becomes – when you put huge numbers of systems together in a complicated fashion, where you're networking systems, but the network doesn't mean a totally connected network, where there's local communication, local communication, more local communication, and then the occasional communication between the local groups, okay, one has to be – one has to understand the robustness issues: loss of signal, for example. I move behind an object where I don't get line of sight, okay, and I might be trusting on the possibility of having line of sight. People are using radio communication now, something we never did in 1990, all right, and trying to understand the nature of how to connect objects together using radio signals.

Q: So you formulated your work in the probability theory of stochastic models. Did you draw on biological models, economic models? You mentioned some physics.

Max Mintz: Okay, so others did. Ruzena Bajcsy always drew in biological models. And so, you ask, economic models, and of course people are interested in – when you talk about teams and such. Team theory, which was – which started in economics, okay, is something that people brought into robotics, okay? So my connection, okay, is somewhat more limited. I'll say how it happened. The decision making under uncertainty, it sometimes pays to use mathematical game theory, because it turns out there's an entire subject of decision making under uncertainty called statistical decision theory, which has game theory built into it. It goes back to the 1940s, with a guy named Wald, who actually pioneered in the subject of statistical decision theory, and using game theory. So he made use of very fundamental ideas in minimax theory. So I, as a young graduate student, got interested in game theory and in control theory, so I got interested in something called differential games, and here – this has nothing to do with robots, at that time. This was 1968, '69, as a postdoc; 1965 to '68, as a graduate student. Got interested in multi-person games, okay, but these were zero-sum games, where one person's gain is another person's loss, and you could think of it loosely as, "How would two aircraft engage in combat so that one aircraft could defeat another aircraft?" Or, "How would an aircraft try to defeat a gun system that was trying to shoot it out of the sky?" Okay, so I worked on problems in statistical decision theory, where I had two dynamic systems, all right, where at least one dynamic system and a gun system – which can be thought of, in some sense, as a dynamic system – where you would develop strategies, okay, where game theory turned out to be a good way to sort out what were good things to do. So this was minimax decision theory in two-person, zero-sum games. And that is work we did, as I said, in the late sixties, early seventies, and then that carried over to things that happen in robotics, but I, truth be told, tended to think more about the more elementary stuff, as opposed to the real team theory stuff, which other people certainly have engaged in. So game theory does play a role in robotics, in – at least, in the theoretical aspects of robotics. And game theory is still a good way of at least getting a qualitative idea, okay, about how to develop good strategies. I think it is still fair to – pretty fair to say that the computational problems associated with serious differential games is still beyond what we can do in real-time robotics. So what one wants to do is figure out, from looking at solutions that are done off – not in real time, but done offline, give you some sense of what are nominally good things to do. Now, the game theory that we also engaged in, which has, again, nothing to do with robotics, okay, dealt with, "How do you deal with things like multiple missile evasion?" And that's work I did in the mid-eighties, and here the problem could be easily stated. It turns out that our experience in – "our experience" meaning the U.S. experience in Vietnam – led the Air Force to understand that a fighter aircraft might be attacked by another fighter aircraft or another – two or more fighter aircraft, okay, where they would launch missiles against our fighter, and they would do it what's called "out of envelope," and that, loosely speaking, means that when you have a missile engagement, an air-to-air missile engagement between one weapon system and another weapon system, there is a region in space where it makes sense to fire a missile. If you're too close to the target, it doesn't work. If you're too far away from the target, it does work, and – crudely. And this is all quite crude. You can think of the target having a donut-shaped – distorted donut-shaped space around it, okay? And I'm just giving the physical space, and not considering velocity issues and other things that have to be considered. So airspeed plays a role in this. It isn't just position, all right? But to simplify the discussion so we can come to a

reasonable point, the fact of the matter is that American aircraft – fighter aircraft – were threatened by multiple missile launches, sometimes out of envelope, meaning not good launches. But you have to understand, what did the pilot have onboard? The pilot had his eyeballs, okay, and could see maybe a flash when the missile left the rail on the enemy aircraft, and that was it, all right? And maybe there was a homing and warning receiver in the American aircraft that would give it a threat warning, okay, that it's under attack, but basically the pilot looks over his shoulder to see if there's a missile bearing down on him, and to evade a missile is a very tricky business, okay? You have to time a maneuver, if you're going to do it, so that you outwit the seeker system in the missile. That means making a high-G maneuver, timed at just the right moment, so that you can make it possible for the missile to miss your aircraft. Now, this gets to be much more difficult when you have two missiles bearing down on you. Even if one of them's not going to be effective, you don't know it, you can't know it – at least, in that era, you couldn't know it – and so you needed means by which you could maneuver your aircraft to blunt the effectiveness, say, of a multiple missile launch. So I worked on heuristic algorithms for doing this. Now, heuristic algorithms are simply algorithms that have a reasonable strategy underlying them, but are not optimal, in some sense. The notion of optimality in control theory, in robotics, okay, is in some sense still a dream. We simply don't have clean, pristine, optimal problems to solve that are large and make interesting problems to work on. Small problems, we may be able to do things optimally, but when you have the big picture, okay, optimality is hard to come by, all right? So optimality is a guidance idea, sort of a thought process, as opposed to a absolute prescription as to what you want to do, because you can't achieve it. You don't even know what the components ought to be, accurately enough, to achieve the end you want. So you have some guidance. You look for things that you're sensitive to, things that might be important. So heuristic algorithms give you a way to do that. And so we worked out heuristic algorithms for maneuvering our aircraft to blunt the effectiveness of a multiple missile threat, and we used game theory to do this. And this was something that we did between 1976 and about 1986. So about 10 years, I worked with the Air Force and with some really good graduate students, who were very interested in this kind of work, and that – it's not a robotics story, okay, but a story about control theory, game theory, and it sort of informed me in thinking about problems in robotics, where you want to avoid collisions, okay, where you're not air-to-air collisions, but ground-based collisions that are our ideas that come from avoiding problems in the air-to-air environment that make sense to talk about, to a degree, on the ground. So we made use of these ideas.

Q: So you mentioned control theory and game theory, but did you find cybernetics influential in your work at all?

Max Mintz: So “cybernetics” is one of those words, okay, that I probably couldn't define if my life depended on it, okay? So cybernetics is due to a guy named Norbert Wiener, as I understand it, and he brought together a whole lot of stuff, okay? I would say that I'm more of a fundamentalist when it comes to how I look at things, so I have a computational engine, I have information, I have probability laws, and I try – and I have some crude ideas about what is a

good cost or payoff function, okay, to try to optimize in a crude sense. So, for instance, if I can make an analogy, a lot of work goes on in decision making, even today – not in robotics, necessarily, but in lots of other places – where mean squared error is an important criteria; that is, an average of a square of an error. Now, not to make a joke of it, but one thinks about it, I need – I have a robotic arm here, and I have another robotic arm here, and I need to put the pen into the cap, or the cap on the pen, if you prefer. Well, if my system – just to make a simple story out of it, if my system is not working well, and I'm missing, here I've missed the cap. I'm not getting this very well, and I miss it over here. It's not at all clear, unless I poke someone with the pen, that my errors over here are any worse than my errors over here, because I've missed the pen in getting into the cap. So what I need to do is get the pen and the cap lined up properly so I can get the pen into the cap, and then let compliance solve the problem. So there is a cross-section area that's important, and in the sense of manipulation, it becomes obvious that you have to get into the zone. And once you're in the zone, it doesn't much matter where you are in the zone. As long as you're in the area of the cap, you'll get it to work right, okay? And that's the idea I want to sort of bring out here. Now, for aircraft and, say, gun engagements, the gun misses the airer – the bullet or round misses the aircraft, and it doesn't much matter if it misses it by five meters or a hundred meters, because it didn't hit the aircraft. So unless this thing has what's called a tracer round, which glows as it passes over the aircraft, and the pilot knows that the gun system on the ground is starting to close in on him, okay, that missed round doesn't do much good unless the round can be visible to the – or at least the effect of the round can be visible to the pilot, and that might make the pilot less aggressive, okay, cause him to back off from attacking the target, okay? The missed round has very little value if it's at five meters versus a hundred meters. The only rounds that are important are those that strike the aircraft. And even in the case of hitting an aircraft, you've got to hit the right part of the aircraft if you're going to destroy it, okay, or make it fail to function properly. You can fly an airplane with a lot of holes in it, and aircraft pilots who engage in combat have done this, and there are plenty of stories to buttress that case. So, again, here it's a, "How close are you?" If you're close enough to actually hit the thing, then you're in the zone. So we talk about what are called zero-one loss functions, where I either get it or I don't get it, and if I get it, I'm good. It doesn't matter where I am in that region. And if I miss it, it didn't matter by how much I missed it. So we made use of these ideas in robotics and in machine perception, and that was one of the things that I thought was important, in terms of trying to understand, "How do we do good decision making?"

Q: So you were here at Penn for quite a long time. Can you tell me about meeting Ruzen Bajcsy for the first time, in the robotics program?

Max Mintz: Okay, so – well, Ruzena will give you the history, and better than I can, but I can tell you that I met her first when I was here as an assistant professor in 1974. She was in computer science, and had joined the Department of Computer Science a year before I got here, to become a member in the – a member of the faculty in Systems Engineering. And so I met her, okay, as part of the then called the Moore School of Electrical Engineering, which had three departments, an Electrical Engineering, Systems Engineering, and Computer and Information

Science. And then Ruzena started the GRASP Lab, and I'm going to say it was 1979, although she may upbraid me and tell me I've got it wrong, okay? But she started her first GRASP Lab work, okay, with a single-finger touch sensor in a basement lab in this building, okay? And she did it on her own, okay, had collaborators elsewhere in the world, but there was no one else at Penn that I can recall, other than her students, who were working with her. And what she did, which was quite enormous, is she took this lab, which was herself and her students, and she drew in other people, okay, from 1979 through the mid-eighties, okay, through the early nineties, before she left for NSF, okay, and she built an interdisciplinary lab, all right, which had a huge number of people working in it, okay, modulo the fact it's part of a larger engineering system – a faculty group. And she drew in Lou Paul. She got Lou to come here. I think it would be fair to say that. And Lou – very talented, wonderful guy. And he and Ruzena worked wonderfully together, and the rest of us benefited in really great ways. When – Ruzena also got other faculty to join the department. Kostas Daniilidis is a person I would cite, okay, C.J. Taylor, okay, and others, okay? And so Ruzena, I think it would be fair to say, was pivotal, in many ways, in bringing robotics to where it is today. And it wasn't just robotics. She was a great believer in vision – machine vision – okay? And yet she did a lot of other things that you'll no doubt hear about from other people. And so I would say she had a very major influence on all of us; not just myself, but on a huge number of people, including not just the faculty, but her students and other colleagues in the department.

Q: And who are some other people that you collaborated with?

Max Mintz: I had the good fortune to have other students I worked with that were not my formal students. One of them was Hugh Durrant-Whyte, who was here in Systems Engineering, who grew into the GRASP Lab as part of his research. He worked with Lou Paul. Lou was his faculty advisor. And Hugh and I would talk about decision making under uncertainty at length, and we had a lot of fun doing that. So I would certainly mention Hugh as a case in point. Other students of mine: Gerda Kamberova would be one, Jeff Agnol [sp?] would be another, Raymond McKendall, and Kevin Atteson are all people who worked either directly in robotics or in decision making under uncertainty in things that were abstractly related to robotics. For example, Dmitry Cherkassky worked on problems in decision making under uncertainty that have a robotics connection, but it was largely minimax decision theory applied to some very interesting distributional problems.

Q: What do you think is the problem that you solved that might have had the biggest impact, a problem you solved with your students that –?

Max Mintz: Well, I don't know that we ever solve any problem completely. I think that would be presumptuous to assume that. We get better at what we do. My students may disagree with me and say, yes, it was solved. I think we learn iteratively. We learn – it's like a progressive

JPEG. We see the shape more clearly as time goes on. As things get sharper, things become clearer. But what happens is that problems don't, in fact, totally get solved because what happens is, is that the technology changes. So what looked like a solved problem today becomes less solved if you bring in the rest of the technology and all the other things that have to happen. So the notion of sensor integration might, in some sense, be solved easily with just two sensors that are static, looking at some object in space, and trying to decide what it is, using two complementary sensors. But when you now have a huge number of sensors, and they're distributed and they're mobile and they have to communicate, and so on and so forth, then that problem, okay, which is still a problem of sensor integration – so if you say, “We've solved the sensor integration problem,” I answer, is, “I don't think so.” We've solved pieces of it at certain scales, okay, and it keeps coming back again and again and again, all right, as the technology changes. So I guess, in some sense, I think the part – rather than saying, “I've solved a problem and it's now settled,” okay, I would say, “I had a lot of fun thinking about how to do statistical decision theory in a wide range of problems, with some extraordinary people who did the real work, and I got to supply the applause.”

Q: What do you think are the big outstanding problems, the things that might really advance robotics, if they were solved?

Max Mintz: It's – I think it's the scale problem. It's the problem of, how do you get all these pieces to work together? How do you integrate huge numbers of disparate, independent, sensor-like systems together to make them work, and do so with a great deal of reliability – how to reason about a very complicated world. And reasoning about the world is everything. In machine vision – and now I'll get shot by others in the GRASP Lab who perhaps see it differently – I claim the vision problem is still a very hard problem, that vision itself is not easy, that vision remains to be settled, in some sense, if it's ever settled, okay? So vision is hard, and I believe it's important that we don't think about cheap solutions where we go out and barcode the world, so now I know where everything is, and I can just scan it and figure out what it is, okay? I want to be able to do it from scratch. I want to be able to do it, or have other people do it, from scratch, and I don't want to have to go out and engineer the world. I want to be able to deal in a world where I haven't been, but I should be able to send sensors into the world on robotic platforms, and learn about the world. So being able to do things from scratch, in a complicated environment where I cannot control the lighting, for example – you put lights here to have this interview, all right? Well, in many situations, I can't control the lighting. Lighting changes, and the students learn this, okay, when they're doing their early machine perception work, that lighting is an important issue. So – and that's just one of gazillions of other issues. So it's the robustness thing coming back: How do you effectively control for the things that are not going to be neat and clean, and something that you can figure out a priori? All right? You have to be able to deal with a range of conditions, and the range of conditions are not necessarily neatly delineated, all right? And the conditions don't interact in a simple way, so there may be a lot of nonlinearities in how these various factors come together. And so that's a – those are issues. So it's the scale of the problem, and the fact that one wants to do things quickly. Real – the word

“real-time” is everywhere, okay? So everyone – everything has to happen quickly, has to happen reliably. Reliability is important. How do you define reliability when you have complex systems, okay – lots of things that have to work? What is – what are the – are there – can you look at the system and figure out, “Is this the piece that, if it fails, everything falls apart, or can we do without this? Can we shift the burden of the decision making or robotics work to another piece of the system? Can we cooperate? How do you define even ‘cooperation’?” Cooperation isn’t easy to cleanly define, all right? And a lot of these problems have been attacked by a lot of – by folks in the GRASP Lab, and have done some really nice stuff. But I would say that trying to sort out these problems and figure out what – that they’re – has one got a clean answer to a problem? Getting clean answers is hard, all right? And I think that’s the problem. We’re – it’s what’s – it is what separates us from mathematics. Mathematics, there may be, if you pick a good problem, a clean answer. I can prove a theorem. Is the theorem useful? Maybe; maybe not. Okay? And does the theorem apply to the world? Well, that depends on whether the assumptions that went into the theorem can be mapped into a physical world, and it’s often the case the physical world is much more complicated than the mathematical model, okay, which represents it actually achieves. So it’s the question of, “How do we build good models?” And as a model builder – and that’s what I do, really, for a living, in some sense – I realize everything I’m going to do on a given day is wrong. All of my work is wrong. Now, I learned to live with that, okay? It’s not as depressing as it sounds. What one has, okay, to understand is, it isn’t that I’m wrong, it’s *how* wrong am I? What are the consequences of my errors? Learning how to deal with errors, okay, and not let them kill you, is an important deal, and that is something that we need to ponder. How do we live with our mistakes and make the system suffice? This idea of optimality is, again, something I don’t really think we achieve. We try to achieve, I think, when we’re practical, something called “sufficing.” We want something to work, okay? That it works perfectly, it doesn’t have to, as long as it works well enough. So it’s the “well enough.” How does one define “well enough”? That’s not easy, and that’s where we have to spend a lot of time, particularly given the complexity of the systems that we have – these huge number of different systems with different kinds of communication, different kinds of information, all right, and so on. So it’s the scale of the problem, okay, that makes this thing very, very, very, very challenging, and I believe that’s where students today, and people working in the future, are going to have to figure – how do you do this distributed problem? I mean, there’s this story about distributed systems in computing, okay? And so it’s not a new story, it’s an old story, but how do you make the thing function?

Q: How do you know what you don’t know?

Max Mintz: Yeah. Well, that gets us to another story, too. Exactly.

Q: Have you also done work in multi-agent systems?

Max Mintz: Yes. The multi-agent story is the game theory, where we did this – we had some early work. This is Ruzena Bajcsy, Vijay Kumar, Lou Paul, myself, and very talented graduated students, and we had some of the – I’m smiling as I say it because I’m recalling the TRC LabMates, which were these clunky robots which we used, and we had multi-agent systems. This is – I hope we’ll be forgiven for saying this in today’s world, but in the 1990s – early 1990s – when we were doing this work, the robotic platforms were very crude, and the sensors, the multi-agents, we had a upper-level agent not just in the hierarchy, but one that was up on the ceiling looking down on the world, talking – in a sense, getting a sense of where the robots were and what was going on, and this was being communicated to the robots. So we had different scales, different –

Q: What were you trying to get the robots to do?

Max Mintz: Something as simple as organize themselves to move through a narrow passage, or to communicate – or to work together to move an object. It was pretty primitive.

Q: But you got it to do them?

Max Mintz: We – well, my colleagues – okay, I will give them the credit – my colleagues did wondrous things with some very crude equipment, okay? And I guess if we had smaller computers, more agile platforms, we could have done more, but we had to learn something, I believe, along the way. If we had, somehow or other, gone from 1985 technology to 2010 technology overnight, I’m not sure we would’ve been able to do it, because I think there’s a learning curve we don’t talk about, all right? And we started out here with some very crude stuff. We learn what doesn’t work. So let me comment on this. This might be useful, and it has to do with a lot of stuff that goes on in research; not just in robotics, but just about everything, including pure mathematics. I had a professor who taught me probability theory many years ago, and he used to say to us, “Gentleman” – there weren’t any women in this particular class – “Gentlemen, if you want to understand this paper, this mathematician’s paper, don’t read the paper. Look at the mathematician’s wastebasket, because what didn’t work is often more illuminating than what did work.” And we never – there’s a whole ethic that we don’t publish our failures, although I believe we should. There should be a journal about failure, because until we learn about failure, we won’t understand success. And there’s no nice way to say that, because the entirety of the community’s ethic is to show how good things are and how fast things can be done and how well things get done, and we never like to talk about what didn’t work, although we should.

Q: Trial and error.

Max Mintz: Yeah.

Q: So who else was influential early in your education, during your graduate work?

Max Mintz: Folks in the Statistics Department, in the Math Department at Cornell, got me interested in minimax decision theory, okay? And Roger Farrell would be one of them. One of them was Jacob Wolfowitz, who was the professor who said, “Look at the wastebasket.” He taught me probability theory. And James Thorp, who was my Ph.D. advisor, who taught me control theory, all right? Much of what I did in my early days dealt with filtering and decision making using Kalman filters and Kalman-like filters, so I spent a lot of time on finite dimensional systems – linear and nonlinear systems – doing filtering. And so that’s how I gradually moved from control theory to robotics, because, in a sense, these decision making modules were important in robotics. So that’s how I, in a sense, got to that.

Q: At the time, were there other applications in mind, or were you working in a pretty abstract level?

Max Mintz: Oh, there were always applications, and the applications were driven largely by the aerospace industry. Just about everything had to do with military systems – military or industrial systems: industrial systems where you’re doing chemical process control, where you have uncertainty – that’s one example; or you’re dealing with aircraft and missiles – that’s another situation. So those were the situations that were typical, where we had examples. Systems that got very complicated early on – that I didn’t have anything to do with, but others did – dealt with things like power systems. You’re probably too young to know this, but in 1965, as I recall, in mid-November – mid to late November – it would probably have been something like November 22nd. We could check the date. I don’t – I’m not absolutely certain it was the 22nd. Twenty-second, twenty-third November 1965, there was a major power failure in the northeast of the United States, and what happened was there was a – I mean, the story goes that it was a squirrel that got into the switchgear in Ontario Hydro. I don’t know if that’s true or not. I may be totally wrong. But in any event, there was a fault in the switchgear at Ontario Hydro that led to a circuit breaker opening, which caused another circuit – caused a transient, which caused another circuit breaker to open. And we were in Ithaca, New York, at the time. My wife and I were in school at Cornell, and we’re sitting in a basement apartment, having dinner, and all of a sudden the lights go out. And it turned out – I wasn’t thinking this way, but it turned out that the lights had gone out over most of the Northeast United States, all the way from Northern New York and Southern Canada, all the way down to perhaps even as far away as Virginia, and certainly up towards Boston, and it was all due to a collapse of a power system. And there you had an example of a complex dynamic system. A power system is a dynamic system. Their flow is of power in the system, so that’s what makes it dynamic. It’s a nonlinear system, because it has the physical components that make it nonlinear, so it’s not easy to analyze. It’s stochastic, meaning there’s a

lot of uncertainty. The stochasticity is not easily modeled, or at least it wasn't in those days. All right? And the system collapsed, quite literally. New York City was without electric power for a goodly period of time, and when you think of shutting down the entire city and having no electric power, okay, it's a probably – it's a breathtaking thing. And it was on a cold November night. This was not a nice summer's evening, okay? And, basically, it affected a whole lot of things. Aircraft flying into then, I guess it was called, Kennedy, okay, at that time, about to land, okay, having just crossed the North Atlantic, okay, as the story goes, okay, the field disappeared. Now, they knew that they were still in the same universe because they could see cars driving on the highways in Queens because they were running their headlights, but the entire – all the strobe lights, all the navigation aids for aircraft to land, okay, disappeared, because they were all making use of this power system that failed. So people caught under the East River in subway cars. Aircraft had to be rerouted, okay? It was – there was – it was a tricky business. The problem is, again, a complex dynamic system: hard to model, hard to understand, and nobody had thought about, "Could this happen?" Okay? This notion that the system was vulnerable to this kind of an event, okay, I don't think had crossed anybody's mind, or at least if it did, they didn't make it out of the – someone wrote a report or a memo, and it didn't go anywhere, okay? Now, you would think, "Well, that settles it. Now the problem will be fixed." And it hasn't been that way at all. We're still seeing occasional blackouts that happen regionally; not just around the university here, but I'm talking states. In... I think it was 2003, there was one that basically started in Ohio, progressed to Michigan, went to Canada, came to New York, and it was just by the grace of God that Philadelphia Electric, through the PJM Interconnection, disconnected from the network before PJM crashed. So it – getting this stuff to work is – complex systems are a problem.

Q: Okay, I want to – <mumbles>. Okay. So, as an engineer, how do you design for all of that complexity and uncertainty? I mean, what do you tell your students? How do you go about designing for that?

Max Mintz: Well, I don't think I know enough about the huge systems to actually address that thing from beginning to end. I would be dealing with parts of it, and the parts of it would be the decision making structures where I have models that give me a range of possibilities for the dynamical system I'm dealing with, and I would train them not to assume linearity, not to assume Gaussianity, but to deal with the real system that is physically reasonable. Cut away the bits and pieces that don't help us get to a good solution: the oddities that increase the dimension of the thing, but don't do so usefully, okay? And sort of come down to a minimal-size system that is realistic enough to handle the problem at hand, or at least the component, or modular component, at hand. So I'm a believer in doing things in a modular way. Not everyone sees it that way. I think there are people today who believe we can do these thing – or to do some of this work that goes on, organically. We build a huge network, and we use machine learning ideas to seek to make the whole thing function. And I'm still not converted to this view, and maybe I'm misstating the view. Maybe no one actually believes what I've just said. But my person view is, is that I like to think – I like to understand component pieces. I'm still a

reductionist. I'm perhaps a little old-fashioned, but I believe in trying to understand component pieces, and making that work.

<crew talk>

Q: So what would you recommend to young people who might want to get into robotics or study robotics in school? What would you suggest the study will be about?

Max Mintz: Yeah. So I'm – I have some strong views on education, and I'd be delighted to recount a few of them – state a few of them. I believe in fundamentals. I believe there's no free lunch. I believe that we don't start with the magic and the high-level stuff, but we get good grounding in physics, good grounding in mathematics, good grounding in engineering science, and that that's what we build from. We can't have one of these huge, razzle-dazzle systems until we understand how the fundamental pieces fit together, and that's – there's no substitute for learning the fundamentals. Physics is absolutely essential. Part of the computer science ethic – and this troubles me a little bit, but I have to show a little bit of appreciation for why it is the way it is – is that computer science, it can be, in some circles, largely divorced from the physical world. Computer science is a science of the mind, where we think algorithmically. We think in terms of how bits are manipulated, and the manipulation of bits doesn't have to happen on a physical platform. We can just think of it as a mathematical abstraction. So, in some sense, we can be divorced from the physical world. I think it's a mistake to divorce computer science entirely from the physical world. I think computer science has to intersect the physical world, surely, in important ways. And to put it – to quote Ruzena Bajcsy, okay, "The bits have to get into the machine somehow," okay? And if you're thinking sensorized systems – sensorized systems – the systems with the sensors bring the bits to the machine. So we have to, in effect, connect to the physical world. So physics is important, all right, whether or not it's robotics or not. It could be something as simple as building a good automobile, okay, where no one would doubt that physics is important. Mathematics is important. What kind of math? Now there's a dichotomy in mathematics that troubles me. In computer science, the mathematics is largely something called "discrete mathematics." We're dealing with discrete structures. And the notion of calculus – the notion of mathematics where we have limits, and we talk about things like integrals and derivatives, and on and on and on, okay – plays no role at all in core computer science. So core computer science is about discrete structures, and has essentially nothing to say about – or has no connections to what I'm going to call core calculus ideas: elementary calculus. Now, elementary calculus is important in robotics, because you're now talking about a physical world, and you need to be able to talk about the robots as physical entities. So dynamical systems requires calculus, and calculus is a serious subject. But if you talk core computer science, as divorced from the application of computer science to robotics, or even graphics, where you're talking about animations, okay, where calculus could play a role, all right, the mathematics of computer science is largely discrete, whereas – and computer scientists work all the time with things called automata, which are primitive models for computation, okay, so

finite-state systems, okay? Well, if you go to mechanical engineering for the – as another – or electrical engineering, not entirely, but it is somewhat true in, say, mechanical engineering that finite-state systems play little or no role in their lives, and the mathematics is largely continuous. So there's a divide between the – between different disciplines, in terms of how they use or don't use mathematics. And I think this is wrong, and that we should get the – we should have appreciation for different kinds of mathematics: where to use them, where not to use them, what one – where the – how things fit together. Mathematics is a fabric, and we can't just say, "It's this piece, it's this piece, it's this piece." It's the entire thing, okay, and we have to see how the pieces fit together. And at one level, I'll need to think discretely; at another level, I'll need to think about continuous systems, okay? And in some cases, I'm going to need to think about both discrete and continuous systems at the same time. I have a colleague, George Pappas, who deals with hybrid systems, and favorite examples of hybrid systems are things like aircraft, where you're using finite-state automata to control the way fuel flows to a jet engine as it functions at different altitudes. So you're going from one control system to another control system as the aircraft moves in space from one altitude to another altitude, as a case in point. So it's both continuous systems and discrete systems playing together, and it's a very serious matter. So this is stuff we need to do better, and there's room for this – totally room for this – in robotics. I mean, it's all over the place in robotics. So this notion of hybrid systems, how to make them work, okay, is a big deal. So teaching people about math, physics, okay? The language of computer science is different from the language of mechanical engineering; very different language. Okay? And we need to be able to learn other people's language, and we need to be comfortable. And there's no free lunch. We need to work hard. We need to dig in. If I want to learn a subject, I need to work on it for a while. And I tell my students, "The best way to know if you've learned a subject or not is, can you teach it to someone else?" And you get up at the board and teach it, okay, without reliance on looking at a set of notes. Can you just walk to the board and talk about a subject? Then, if you can do that, then you know you know something, and if you can't, then maybe you need some more work. So it's how do we learn the bits and pieces before we get the grand ideas that we're going to build huge systems that will have these miraculous properties.

Q: So talking about the transactions between Electrical and mechanical and control theory, and different areas of engineering, sort of historically, what do you remember of the sort of first time people talked about robotics as robotics, or as its own field? How do you see that field emerging from these other fields?

Max Mintz: Well, Ruzena still has this dream, okay, and I don't think it's happened yet, but she stated this dream to me in the early eighties. She said she wanted to someday have a dinner party where there were robotic-like devices that would clean up after the dinner party and totally take care of everything. We have yet to build service industry robots that do that. So my feeling is we still have a long ways to go, in terms of making that happen, and that has to do with the complexities of the environment in which things are. The notion of being able to actually determine what an object is, what state is it in, okay, it's not easy to do. And so there's still

plenty of work to happen. So I'd say Ruzena laid the groundwork for thinking about what I'm going to call service robotics, okay, as opposed to military robotics. I'd say that the idea of being able to combine vision, combine actuation, okay, were the early days, okay? There's a story that I'm told – I don't know if it actually happened – that Marvin Minsky said to a graduate student, I think at MIT, in the – might have been the late fifties or early sixties, one summer day, or whenever, "You should connect that television camera to a computer and see if you can get the computer to see." I mean, and it's laughable today that one would make such a statement, if in fact it was made, okay, that – you know, be so naïve to suppose we will just simply connect A to B, and miraculously we will have something that works terribly, you know, particularly well. And so this began, I'd say, in some sense, a notion of machine vision. So – but it happened at a very crude level, and it took a while. And, again, I think we cycle back. This notion that things are settled and that there's a substrate, and that we never have to go back and look at the bottom of the substrate, is wrong. We come back, we cycle as other dimensions change, and the complexity dimensions of multiple sensors, speed with which things have to happen, okay, accuracy requirements – I need an accuracy requirement that works fine for one thing, and another accuracy requirement quite different for something else. Reliability, for one thing, can – one thing is okay to be this reliable. I can lose this many machines. It doesn't matter, okay? Other machines are very expensive. I don't want to lose them. I don't want something to go wrong. We still have very complex systems, and, I mean, in today's or yesterday's *Wall Street Journal*, and in the *New York Times*, it's alleged – I mean, I have to let the organizations that are responsible for this weigh in on it, but I note with sadness that it may be that that Air France jet that crashed on its way from Brazil to France crashed because of pilot error, okay, because the pilots got distracted, okay, and weren't taking – pay enough attention to their airspeed, okay, and the aircraft basically was lost as a consequence of machine complexity, where there were too many sensors and alarms, okay? Some set of sensors failed, caused the autopilot to be shut off. Alarms were going off in the cockpit, okay? They know this, apparently, because they've now transduced from the black boxes that recorded the information at the last moments of the flight. And so, in a sense, here we have a man-machine system, okay, which was too complicated to control by the people who were in charge of controlling it. And it's not to say that they couldn't have done it, but their training didn't outfit them to handle this particular bad instance. So it's trying to think one's way through bad instances where one has complicated systems to work with. And that's not easy to do, and I'm not going to point fingers at anyone, because I could see it happening again. I mean, again, the power systems fail, okay, different things go wrong, okay? So maybe we solve one problem, but then we don't solve it sufficiently to take care of all future instances where things will be a little different, okay? It's being able to understand what the differences might be. We tend – we've also come to rely on systems in ways that are sort of breathtaking. We have an enormous amount of cockpit automation in aircraft, and it's probably necessary to have that, okay? But, in a sense, when people rely too much on automation, okay, it can be dangerous, and so one needs to always keep alert and keep an eye on what the machine is doing. All right? So, again, this is not robotics, per se, but it's complex automatic systems, okay, that can in fact defeat us if we're – if we rely on them too much.

Q: Well, I know we're sponsored by the Robotics and Automation Society of the IEEE. Do you see a difference between robotics and automation in any significant way?

Max Mintz: I'm one of the outliers in this. I don't know – I wouldn't be able to tell you what everyone thinks, but I see a blend between them. I mean, there may be... I don't want to say "political reasons" for picking – drawing a circle around this piece of turf, and circle around this piece of turf, and saying, "This is X, and this is Y." I'm not a great believer in turf definition. I'm a great believer in trying to understand basic problems, and if part of it is – if I'm doing something with a missile, okay, is a missile a robot? Well, not in the sense of the robots running around in the GRASP Lab, but is there a sensor system? Is the thing making decision based on sensor data? Is there a control system? What differentiates a cruise missile, okay, that has to follow a trajectory at low level, okay, to get in under enemy radar, versus a robot, okay, doing something to move from point A to point B, okay? I think there's some issues that are in common, but there are also things that make them different, and it may be the people who do the work are themselves different – have different outlooks – as opposed to that there is something absolutely separable, intellectually. I doubt the absolute separability of the intellect. I'm very easily convinced that people will want to draw boxes or circles or whatever around domains, because it's easier to define a domain that way. This has to do with organizations, as opposed to dealing – and people – as opposed to dealing with intellect. And that's why we have departments in universities. You could imagine, we're all thrown together in one huge department within the School of Engineering. It would be unmanageable. Okay? It couldn't be done, or at least I'll say it couldn't be done. The dean may tell me I'm wrong, but I don't think it could be done. And it can't be done – or it can't easily be done – because it has to do with the nature of human beings and how we like to think about things, how we like to organize ourselves, and having an inside and an outside in some ways, which isn't always a nice thing to say.

Q: In terms of the institution, when was it first possible at the University of Pennsylvania to get a degree in robotics?

Max Mintz: It happened recently. It's a master's degree, and we have a master's degree in robotics that George Pappas and Vijay Kumar and Kostas Daniilidis are primarily responsible for starting.

Q: And what year was that first time?

Max Mintz: Oh, goodness, it must be three, four years ago. No more than that. We could check for sure, but it's within the last four – three, four years.

Q: So most of the members of the GRASP Lab have been in the Electrical and Systems Engineering?

Max Mintz: No. So the GRASP Lab is broken into – the GRASP Lab is a multidisciplinary laboratory, so it includes – it’s a lab – in a sense, it’s the GRASP Lab of the Department of Computer Science, but that misses the point. It’s Computer – it’s supported by the Department of Computer Science, but it couldn’t function at all without close collaboration, integral collaboration, intertwined like this. You can’t break them apart: Mechanical Engineering, Electrical Engineering – and by “Electrical,” I mean Electrical Systems Engineering – and other divisions. The Psychology Department has people doing work in machine perception – or in perception, anyway – and they work with folks in the GRASP Lab. Okay? We have folks in the GRASP Lab who work with folks in the Statistics Department, okay? So there’s machine learning going on here, which affects work in GRASP. Machine learning interacts with folks in the Statistics Department. I have an appointment in the Graduate Group in Statistics, having more to do with my interest in decision making under uncertainty, okay? But I wouldn’t call it robotics, okay? And I wouldn’t call what the people in the Statistics Department do as having a connection with robotics. They’re interested in statistical theory and application. Okay? So, in a sense, the Ph.D. students came from Computer Science; in no particular order, Computer Science, Electrical Engineering, Mechanical Engineering, and there might even been people who were postdocs who had connections with other things. But let’s talk about just the Ph.D. students. So they came from graduate groups. It has to do with how things are organized. Students get degrees at Penn through something called a graduate group. So there’s a Computer – an Information Science Graduate Group, which is roughly connected with a group of people in the department, but may have some other people as well, from Mathematics, and so forth, okay, and from Philosophy, and so forth, and from other places. And then there’s the Electrical Engineering Graduate Group, or Electrical Systems Engineering Graduate Group, and they may have people from Computer Science as part of their graduate group, but it’s a graduate group that runs the Ph.D. program. Then there’s a Mechanical Engineering Graduate Group. So the students come from those graduate groups, but they work in the GRASP Lab, and they work interactively, so you’ll have mechanical engineers working with electrical engineers working with Computer and Information Science graduate students, okay, to achieve some very interesting research. So I would say it really is a multidisciplinary department, with graduate students who have their home base in, because of the way we’re organized, specific departments. Could we all be one department to do this? I suppose we could. I don’t know how it would be organized, and my instinct is, is it probably wouldn’t work as well, because by making it a little more focused this way, it makes it a little easier to know who’s in charge of what.

Q: Okay. So I think that’s most of my questions. Is there anything else you’d like to add?

Max Mintz: When Selma [Sabanovic] sent me an e-mail, I had to take a deep breath. I hadn’t thought about this, but yeah, I’ve been involved in the GRASP Lab for 27 years. I hadn’t been

counting. So it's been a good time, and been very – I've had a lot of fun, and if I had it to do all over again, I wouldn't change a thing.

Q: Great. Thank you.