

Pittsburgh PA
1:03:58
m4v

Pradeep Khosla

An interview conducted by
Peter Asaro
with
Selma Sabanovic

November 24 2010

Q: Introduce yourself.

Pradeep Khosla: Oh. So I'm Pradeep Khosla. I'm the dean of engineering at Carnegie Mellon, and to take you back into my life history I was born in India, and I went to high school and undergraduate school in India. So my undergraduate was from Indian Institute of Technology Kharagpur, which is one of the five IITs and the oldest IIT. And I graduated in 1980, and I decided that I was going to go work in India for a while because I was patriotic kind of stupidly. And I was in India working till 1982 in the area of real-time control of power systems, and then I decided it was time to come to the US because I wanted to be a faculty member. I wanted to do research, so '82 I came to Carnegie Mellon.

Q: What was your degree in?

Pradeep Khosla: My undergraduate degree was in electrical engineering, but my honors thesis was in computer science. It was designing a disassembler for TDC316, which was the Indian equivalent of PDP8, I believe.

Q: When did you come to school in the States?

Pradeep Khosla: I came here in 1982 to get a master's and hopefully a Ph.D., and I was fortunate enough that I got both at Carnegie Mellon.

Q: Why here?

Pradeep Khosla: So when I applied to the US I had admissions in three places, and at Carnegie Mellon there was a professor who listed his research interests as computer control of power systems, which is exactly what I was doing, what I wanted to do, so I said "Okay, I'll come to Carnegie Mellon." Plus I had a classmate of mine who had come here in computer science, Bud Mishra, who is now a faculty member at Courant. So I came here.

Q: What was the relation between computer control of power systems and robotics?

Pradeep Khosla: Well, it turns out that when I came here that catalog I was reading was four years old. There was no Web at that time, so people had moved on, and robotics was becoming very hot. In 1980 is when Raj Reddy at Carnegie Mellon started the Robotics Institute, so there was a lot of excitement, a lot of people involved in it, and so I decided to work in that area because it was still computer control of electromechanical systems.

Q: Who was the initial professor you wanted to work with?

Pradeep Khosla: So I came here to work with Chuck Neuman, but that – I did my master's with him, but Chuck was a control systems guy, so I had a co-advisor, Fritz Prinz, who was in mechanical engineering, so my master's thesis was seam-tracking for welding robots, where the idea was that if you're going to use these robots for welding then how would you track seams? And so I developed an algorithm, and then while I was doing my master's I really got very interested in robotics, and for my Ph.D. I worked with Takeo Kanade. He was my advisor for Ph.D.

Q: What was your thesis project?

Pradeep Khosla: So at that point – this is about 1983, late '83 – Takeo Kanade was working with Harry Asada, who is now a professor at MIT, on developing what is called a CMU direct-drive arm one, so that arm was based on new technology, a six-degree of freedom manipulator based on a new technology where there were no gears, so the motors were directly coupled to the lengths using samarium-cobalt motors, high magnetic intensity motors. But that arm hung from the ceiling. It weighed like, I don't know, 600, 700, 800 pounds, so its payload capacity compared to its weight was way too small. And we had this idea of creating what is called a SCARA configuration manipulator, which would be very appropriate and useful for high-speed assembly. So my Ph.D. thesis was CMU direct-drive arm two, and that design was a SCARA design. And, for example, the Adept robot is a SCARA design, and it's a direct-drive robot.

Q: The first SCARA robot was made in Japan, right?

Pradeep Khosla: Yes.

Q: Did you have any connections with that?

Pradeep Khosla: No. So what we wanted to do was – so we knew the advantages of SCARA configuration, and we knew the advantage of direct-drive. Because of lack of gears they were higher-precision, faster. So we combined the two to create a new design for a SCARA configuration robot that was direct-drive.

Q: What were the advantages of the SCARA configuration?

Pradeep Khosla: The SCARA configuration is like – the two lengths are horizontal in a plane, so gravitational forces don't act on these lengths, which means the motors don't have to pick up the weight of the robot, so they could be sized smaller. And if they're sized smaller that means the torque is used for speed, and direct-drive configuration gives you better precision.

Q: What kinds of things had you been using before you had the CMU one and CMU two arms?

Pradeep Khosla: I was not using any system. I was not even in the business, right? But this was actually a very challenging Ph.D. thesis. I may not be exactly right, but I think I'm nearly right. This was also the first experimental thesis where the thesis was the design of this arm, the electromechanical design of this arm. Now, because this arm is higher-speed, higher-precision it required higher sampling, faster sampling rates, and at that time there were no computers fast enough to create – what we wanted was 1,000-hertz sampling, so sample the control system 1,000 hertz. So I built a multi-processor architecture using a special processor that had just come out in the market. It was called Marinko [sp?]. I remember that. It was a 64-bit word length CPU, and I had to write code not in assembly but in bytes. I had to write code in A, C, F, G, because that's all they had. They did not have an assembler at that time. So it was actually rather challenging. So we did that, and we for the first time demonstrated real-time control of a direct-drive arm at 500 hertz. Now, in the process of doing that it turns out that the computation of these dynamical equations is extremely intensive, which is why no computer could compute those dynamical equations and sample at 500 hertz. It took longer than two milliseconds, so that's why we had to build the CPU. But there was also an implicit assumption where the dynamics of the system are known, so there is a very famous scheme called the computer torque scheme, which works extremely well if you know the dynamics or the dynamical parameters of the robot. So the other contribution I had was showing theoretically that even though it's a non-linear system it is linear in the dynamical parameters under some transformation. And then we implemented real-time estimation of dynamical parameters, and then using those parameters we implemented real-time control to show how computer torque really worked very well.

Q: Who were some of the other people that you worked with?

Pradeep Khosla: So my advisor was Takeo Kanade, so it was Takeo and myself. We also had some staff members. Don Schmitz was a staff member, and he was a mechanical designer and an embedded systems person, so he had great contributions to make. I mean, honestly without his creativity in design I don't think we would have built the system as good as we did. There was another electronic technician, very creative guy, Mark De Louis [sp?].

Q: After your Ph.D. what did you do?

Pradeep Khosla: So I finished my Ph.D. in 1986, and at that point I was looking for faculty positions, so I had several offers, and CMU was one of the first, and no better place to do robotics research than CMU. Second, we had a lab already set-up. I had an experimental set-up set-up, and so I decided to just stay here.

Q: What were the other places that you were considering?

Pradeep Khosla: It was Courant Institute, it was University of Maryland College Park, it was University of Rochester. I can't remember the others. Might have been Caltech.

Q: What was the next project you worked on?

Pradeep Khosla: So I did two things. I continued on this direct-drive arm research, and we moved on to demonstrating force control, because that was a big thing. When robots come in contact with the environment there are forces of interaction that are generated. So the idea there is not just to create or to control the position but to control the force of interaction so you can slide around an object, you can pick an object, so you can see this has a big impact on grasping research, for example, designing hands. And the fundamental work in force control was done by Marc Raibert at MIT. I believe it was part of his – wait. Was it Marc Raibert or Matt Mason? One of those two. It was called compliance control. It's a very fundamental piece of work that showed that you could actually – when a robot interacts with a surface or an environment there are certain degrees of freedom where you can control the position. So if you think about a surface and I'm touching it, out here I'm allowed to control the position, but going into my palm I can only control the forces, because I can't control the position. So they had this model of separating the space into where positions could be controlled and where forces could be controlled, and that was called compliance control. And robots to become useful had to be compliant with the environment. So we demonstrated, again for the first time I believe, high sample rate force control strategies. But the most interesting part was – my first Ph.D. student was Richard Volpe, who is now at Jet Propulsion Laboratory, and Richard was a Ph.D. student in the Department of Physics, and I was a professor in electrical and computer engineering. Richard wanted to do his Ph.D. in robotics, so I went and talked to his advisor in physics, and we agreed that I would become the co-advisor. And this is Carnegie Mellon. You know, we allow faculty to advise just about across the whole university. So Richard was the one who did work in force control for his Ph.D., and with a physics background he brought a wealth of knowledge and a style of thinking that typically would not have existed. So he had several contributions, but one of his contributions was showing the equivalence of impedance control and force control. So at that time there were two camps. One was impedance control, which was Neville Hogan, and the other people were doing force control, and there was some impression that these are different strategies. And what I wanted to do was run the equivalent of a Turing test, which is if I implement a force control strategy and a impedance control strategy and I don't tell you which

one is which, can you tell the difference? And we demonstrated that there is actually an equivalence between the two, which was rather interesting.

Q: What kind of tasks did you demonstrate?

Pradeep Khosla: Oh, in this case it was more like following a surface, and there was also a strategy for impact control. When a robot came in contact with the environment at high speed, could you do it in a way that it just touched and never bounced around? So if you imagine two stiff objects, when you hit one against the other there's bouncing. So we wanted to demonstrate what is impact control, which is there was no bouncing, right, and that would then lead automatically to compliant control.

Q: What was the reaction of the community?

Pradeep Khosla: I think when you look at research and there are a lot of young people involved the initial reaction – there's always this notion of trying to manage and maintain your space in the community. But looking back 30 years, 25 years the reaction is more reasonable, more muted and more rational, I believe. So one cannot judge the reaction to research at the instant you publish the paper.

Q: What were some other projects you worked on?

Pradeep Khosla: So after that I had this notion that – well, in the following way. So if you think about computers, they're called general-purpose machines, where the idea is that any function that can be computed can be implemented on any general-purpose computer. Now, people talked about general-purpose six-degree of freedom robots, and intuitively it's obvious that not every task that can be done by a six-degree of freedom robot A can be done by a six-degree of freedom robot B, A and B being two different configurations. So that led to this both scientific and philosophical question in my mind as to why are these called general-purpose robots, because they're both six-degree of freedom robots. I call them general-purpose, but I know that this robot A can do a task and this robot B cannot do that same task, right? So clearly there's a problem. So I started thinking about what would a general-purpose robot look like, and to develop that notion – so what I wanted to really do was build what I think of as a Turing robot just like a Turing machine. So I came up with this notion of a reconfigurable modular manipulator system, which was a project again I started initially with Takao Kinane. So the project was called CMU reconfigurable modular manipulator system, which went the following way, which said that if you give me a finite set of tasks, let's call it set T, and if I create a set of modules, set M – these are like Lego, independent modules – then if I can demonstrate that for every task you pick from this set I can pick a robot from this module set, I can build a robot from

that module set which would do that task, then in some sense you would have defined a general-purpose robot, which would be the box of Lego, vis-à-vis the set of tasks that has been predefined. So the only constraint was the set had to be finite, right, but it didn't put any constraints on the number of tasks in that set. It would be a million, two million, five million. Just could not be infinite. So conceptually this made a lot of sense to me, but it had significant problems, so now we have to think about design methodologies. If you pick a task from here how do I figure out what kinematic configuration works, right? So let's assume I have a method to figure that out, because that was part of our research on designing mapping tasks to robots, and that was Chris Paredis' Ph.D. thesis, who is now a professor at Georgia Tech. And that was some very interesting work that was done. So the next question is – so let's assume that problem is solved. So you give me a set here, a set there. I can pick a task, and I can tell you what kinematic configuration works. So we put it together just like you would put Lego together, so then the task was "How do I design these modules so that I can just put it together?" And, again, Don Schmitz was involved in helping me design this system, and the idea was that whenever you connect two modules together power, data, communications automatically get connected. So when the system is configured the system is totally connected. There's nothing else to do. But I am still left with the problem of programming the system. I have to figure out the kinematics, the forward kinematics, the inverse kinematics, the control strategies, right? So we had a project which would automatically determine kinematic configurations, so as soon as you build the robot every module was a smart module. It knew about itself, but it didn't know about anybody else. And then we had algorithms where as soon as the system was built they would all transmit what they knew about themselves, and the system would figure out what this kinematic configuration looks like, what are the forward and the inverse kinematics. And that was a master's thesis, Laura Kelmar. I have lost track of her, so I don't know where she is. Which was fine too, but I'm still left with the problem of controlling the system now, automatically determining control strategies. And then we realized that this was a non-trivial problem, because to do this right you had to have capabilities in the operating system that would allow me to configure controllers automatically. And there was no operating system at that time that allowed us to do this. So my Ph.D. student, David Stewart, his project was called Chimera, C-H-I-M-E-R-A. It was a real-time modular operating system where this operating system allowed you to create or assemble modular code. So the idea was just like I'm assembling this robot using Lego blocks, if I had Lego block equivalent of software pieces, could I then connect them together automatically and create a controller for this system automatically? So Chimera was that environment that allowed us to do this, and we demonstrated how this would be done on a physical system. So all my life it's been all about theoretical work mapped and supported by experiments. So we demonstrated this as really successful, and if you look today a lot of work in modular robots in different shapes, sizes, forms across the country, across the world, and I believe that the work we did was probably the first or the second work in the world in that area, and that has laid the basis or at least demonstrated you can do it. Modular systems of today look different than what we had, but nonetheless it showed the utility of these systems and how they could be done. So once we had demonstrated this system the next goal was to say "Okay, can a high school dropout program this robot?" because at that point and even today to some extent you literally need a Ph.D. in embedded systems or a Ph.D. in something to program robots. That doesn't make sense if they're

going to be useful. So we developed this iconic programming language called Onika, and that was Matthew Gertz –it was his Ph.D. thesis – where the idea was that somebody with an understanding of the task but no understanding of the robot or the technology could program. So this iconic programming language was puzzle pieces, so each icon was color-coded and shape-coded, so it looked like a puzzle piece, like a 1,000-piece puzzle. So the shape-coding was for colorblind people to be able to connect them together. So when you brought two pieces together that didn't make sense, but if the puzzle pieces didn't fit-in the system would reject it automatically, so it was an aid to help you program too now, because you could not just bring any two pieces together and create arbitrary programs, because they had to follow a certain sequence. So now every one of these puzzle pieces in the background had sophisticated code behind it. So when I brought two puzzle pieces together the code got combined automatically based on this real-time operating system, Chimera, that allowed me to do this. So now I could program a robot at a very high level, and it was all seamless. The code got configured automatically, the controllers got configured automatically, the kinematics got configured automatically, and you just do the task. And we demonstrated that with...

Q: What would one puzzle piece stand for?

Pradeep Khosla: One puzzle piece, for example, might be "move." The other...

Q: So then you would have like "move left."

Pradeep Khosla: Right, whatever. That's right. And then...

Q: How would that compare to the Lego Mindstorms kind of programming environment?

Pradeep Khosla: Yeah. So conceptually it is similar, but, remember, this piece of work is 1990, right? So then after that we got interested in swarm robotics, but it was a little bit – okay. So the idea then was – so '94, '95, '96 I spent three years at DARPA, and there I managed large programs in robotics, in real-time control and in manufacturing, automation and design engineering. And that gave me a broader insight into two things. One was the needs of DoD and how we would have used or we would want to use robots in warfare. And second was capabilities of several universities who were all my contractors, like Stanford, like Berkeley, like U Penn, including companies like Lockheed Martin, Boeing, Northrop Grumman. They were all my contractors. So that was a spectacular three years where using my own background I could influence the direction of the field and at the same time using collective intelligence of the community I could learn a lot more and influence the direction of how robots were going to be used in warfare.

Q: How did you try to shape that direction?

Pradeep Khosla: So the real-time planning and control program, for example, was a more science-based program developing algorithms and so on and so forth. There was another program which was called small-unit operations, where the idea was how would you empower a soldier. So at that time there was this view in DoD which said that conventional techniques of battalions and platoons are not going to work. We need to have small units, like six to eight people, special operations who go do something interesting and useful somewhere. Now, to empower these people because they're in a very dangerous zone you need to be able to deploy forward-looking sensors. You need to be able to collect information. So the idea then was could we create robots small enough that could be thrown who would – then one robot would give you vision information from camera. The other might give you thermal information, infrared, whatever. So that was the program there. So when I came back to Carnegie Mellon, Dick Urban, who was a program manager also at DARPA at that time had another thought where he wanted to combine small form-factor – he wanted to build small form-factor robots. So Dick told me "Pradeep, can you build a robot five centimeters on the side?" Right? Now, sounds easy. There are many robots right now which are that size, but this is, again, 1996, long time ago. So at that point we build these robots about this size, and they were called millibots, right? Now, clearly in a robot this size – they were wheeled robots, so you could not pack all capability that you needed. So we had this vision, which went the following way, where we'd said that no single component of this system has to be able to do everything, but as a system it has to be able to do everything. That means there might be in this system of millibots robots that have IR sensors. There might be robots that have sonar sensors. There might be robots that have camera sensors, right? So think of this system of robots being several blind men feeling an elephant, but they have to be communicating with each other so that the system operator, me, knows that it's an elephant, right? So the vision here was can you have multiple robots behave like a single logical machine. So I want to interact just with one machine, so as soon as I give a command to that machine, says "Tell me what's in this room," the system should automatically figure out what are the capabilities of each one of these subsystems it has, who can do what, create code, dump it on that, let them do it, collect the information and interact with me as a single logical system. So this was now an extension of this modular robot that I was talking about but in a slightly different way where I had several independent systems who were going to behave like a single logical machine. So that's the system we built. It's called millibots, and there was a Scientific American article that we wrote on that, which I think is pretty well-cited, so...

Q: What would you say was the overall role of DARPA in shaping robotics?

Pradeep Khosla: I think if there's one agency that can take credit for – okay. So no agency can take all the credit, but if there's one agency that can take credit for pushing the technology at the cutting edge it is DARPA. DARPA had a very big role in robotics in general, but if you look at, for example, the autonomous land vehicle program, right, without DARPA there would not be

any Grand Challenge one or the Urban Grand Challenge or – what was the other Grand Challenge called? There were two Grand Challenges recently in the last two years. Those Grand Challenges – we were able to demonstrate capabilities because of 20, 30 years of DARPA investment in autonomous land vehicles, for example. Similarly, DARPA made investments in manufacturing and automatic assembly, and DARPA made investments in the next-generation controllers. So DARPA has had more than a 30-year history I believe or a 30-year history of investments in robotics that have brought us to where we are.

Q: In DARPA how is the policy about where robotics is going made? How does the interaction with the military happen?

Pradeep Khosla: Yeah, so there are meetings, but the way the interaction happens is DARPA brings program managers who are domain experts, MEMs, robotics, computer vision, solid-state physics, whatever. Doesn't matter. And DARPA has a mission which says that they want to be doing cutting-edge research but of relevance and application to the military but not necessarily in one year. It could be five, 10 years, right? So as a program manager you know what your passion is. You know what the mission is of the agency, and you have access to just about every other agency in the country, including the three-letter agencies besides the DoD. So what you're expected to do is go talk to various people, figure out what the needs are, understand what the state of the art is, understand how much can it be purged to satisfy this need, and build a program plan which then you pitch to either your office director and to the director of both and so those layers are not that many, just two layers of other smart people who look at it and say yeah that makes sense, but have you thought about this, have you thought about that, so there might be a couple of iterations. Once the program is approved then you go put out the BAA and you run it like a project unlike the National Science Foundation where you are funded, you are funded, I am funded and you're all doing your own work. DARPA we would put together just like my modular robot, a project where you're doing component A, you're doing component B, somebody doing component C, and the program manager would have the overall oversight to make sure these components connect and then they all put together to solve the problem that is of interest and relevance to the country and especially the DoD.

Q: But, the people who are doing the projects aren't necessarily aware of the larger?

Pradeep Khosla: No, absolutely they are because when you build a vision for a program like that you cannot build it top down. You talk to the customers to see what their needs are. You talk to the community to see what the state of the art is and what their passion is, so it doesn't make sense to define a program where nobody in the community wants to work on it and we want the best minds to work on it and the only way you get best minds to work on it is to have a challenging problem and give them freedom and empower them to do great stuff. So, it's a delicate balance between enabling them significantly, i.e. total freedom versus getting your job

done. So, I learned a lot in terms of project management, managing people, and these are the best of the best so.

Q: And so, when you're making the decision of how something that looks much like the very basic science, how do you make a decision of whether it's going to have an actual impact on application in five to ten years?

Pradeep Khosia: So, this is why DARPA wants to hire people who are topnotch domain experts in the area, right. Because these people have a vision, these people have good taste. They have a sense of quality. They know what can be done and how long, so it's intuition, it's experience, that is experience of other people all coming together.

Q: And, how did you before you started working in DARPA for the three years, was your funding previously generally from DARPA?

Pradeep Khosia: Yeah.

Q: Where was it from?

Pradeep Khosia: I had money from both NSF and DARPA, but it was purely accidental when I started working for DARPA where Gary Denman, who was the director, so I joined DARPA in 1994, January, so Gary Denman, I think it might've been '93, June or something, came to visit Carnegie Mellon, he was the director of DARPA, and I remember this clearly. I pitched to him this project Chimera, the next generation operating system in a programming environment and how it was going to change everything we're doing and better systems. And, rightly or wrongly for some bizarre reason he liked it, and he said how much money would you want for it. I forget what the number was, I said like \$10 million or \$5 million dollars, something, it was a big number, way beyond a professor seven years out of Ph.D. would say. His response back to me was why don't you just come work for DARPA, implement this, you won't get 3X the money and the world will solve the problem for you. Just at that time, in 1993, I had just been promoted and I was looking for the next thing to do, and this came at the right time, so I talked with a couple of my mentors and they said hey this is a great thing, go do it. So, I took three years off and I said okay I'm here Gary, tell me what you need to do. <laughs>

Q: So, did you get the funding at that time?

Pradeep Khosia: No, so Chimera was just one project right, so the funding was real-time planning and control. That was a program which involved planning algorithms, control

algorithms, the underlying software base, so it was much broader than just one real-time operating system.

Q: After DARPA when you came back to CMU did you continue to implement that program?

Pradeep Khosia: No, so when I came back to CMU that's when I did this project on Millibots.

Q: Okay.

Pradeep Khosia: Right, which was the next generation of modular systems where you would have multiple physical systems with a single logical machine.

Q: And so, what came after the Millibot?

Pradeep Khosia: So, after the Millibots what happened is I started getting interested, as you can see, so my career involved electrical and computer engineering. It involved software engineering. It involved mechanical design and embedded systems, so towards the end of Millibots I started realizing that once you have many of these systems out there where the code gets generated automatically, the code gets downloaded, the probability that one of these systems could be taken over by an adversary is extremely high. So, this is now '98, '99, so that led me into this whole notion of security in embedded software so how do I make sure that one of these machine subsystems does not turn rogue on me? Right, because that would be a bad thing. From there on, I just got interested in security in embedded systems and in '99 I became department head of Electrical and Computer Engineering and as a new department head I was looking at the department thinking about what's the next big thing we should be doing and I realized this computer security for embedded software is going to be a big thing. So, I created a center, which now is CyLab. It is the largest cyber security research center in the country at a university. I just got interested more in software security.

Q: Who are some of the people that you collaborate with on that?

Pradeep Khosia: Adrian Perrig, who's a professor at Carnegie Mellon. He's one of my main collaborators and the other collaborator is Rohit Negi, but Rohit is an information theory guy, so with him I collaborate – I have two Ph.D.s on sensor networks, so if you think about these little Millibots that I call them and they start running around, you can think of them as deployable, mobile sensor networks, right. So, now the fundamental question is okay so if I have enough of these I know I can sense enough of the environment, but I don't know what the sensing capacity is. So, just like a channel has information capacity, like Shannon's Theorem, so we got

interested in what is the sensing capacity of this system. So, Rohit as an information theory guy brought to bear his knowledge and we have some results on what would this capacity look like? So, we've been doing work in sensor networks.

Q: And, what are some of the special challenges for security in embedded systems versus more traditional computer systems?

Pradeep Khosia: I don't think, so on a very low technical level the problems are different, the challenges are pretty much the same, but in embedded systems, so take power systems, for example, a lot of CPUs that are used for power system management and control are embedded CPUs and the problem there is that they control critical infrastructure like the CPU in your car is an embedded CPU. It's an embedded software system. It controls your car. If that gets taken over and your car starts to do things that you didn't want it to do, it could be bad for everybody, same with power systems. So, it's the implications that are different, the fundamental challenge is still security of software systems.

Q: It seems like in terms of the general application of your work initially you were doing a lot of things that were more kind of industrially oriented, oriented towards industrial application and then after DARPA now you're looking at some other kinds of applications.

Pradeep Khosia: Right, so you can see the transformation, so there's a project I didn't talk about where it was a similar notion of a modular system where people – it was on the Troicabot project where there was a system of three PUMA robots like in a triangular configuration that Westinghouse wanted to get rid of, but I had an interest in them so they gave them to me. And, where people were talking about robotic assembly where all the assembly would happen with robots, but clearly programming these systems was painful, right. So, at that point we had this vision of here's what should work. If I build a CAD model of an assembly, right, and I put it into some system, what should come out the other end is a program to assemble that from components and then I should download this program to the system of Troicabot and demonstrate automatic assembly. We actually did that project where it had geometric reasoning involved in it because now you take a CAD model. You reason on the CAD model to figure out how it's going to be disassembled, then you more often than not we'll reverse sequence which creates the assembly sequence, right, which tells me what motions I should have in the robots. Once I have that I have this iconic programming language that I talked about, which at a high level encapsulated those motions and brings them together, right. The software gets created automatically. I decide which robot the software goes on, which one of these three and then out comes your assembly. So, we did a big project that lasted for like five years in that area. So, my pre-DARPA was more industrial applications, but more thinking about this touring robot and even today, I mean really few people, at least from what I can tell, have been pushing this notion of what a touring robot looks like. We build these robots to solve a task or two, but we have

never really addressed the problem of if there was a touring robot what would it look like and then post DARPA was mobile robots, Millibots, making multiple robots act like a single logical machine, more field/DOD oriented. About 2000 onwards, it's more software security, embedded insecure systems – I'm sorry, security of embedded systems. That's also the time when I became department head, 2004 I became Dean so my interest had expanded to building organizations that could accomplish these big goals and tasks without me having to do all the work on my own.

Q: As the Dean how do you see the position of the Robotics Institute relative to other robotics programs around the world?

Pradeep Khosia: I think the Robotics Institute over the last 30 years has made a significant impact on the whole community. There was a time when the Robotics Institute was a dominant dog, when it started. Then there came a period when MIT, Stanford, Michigan had great robotics programs. Then there came a period after that where many of these programs kind of got diluted, not in the quality sense, but more in the sense of looking at other issues in life in the research area. I think we're at a point right now where there are more "robotics institutes" in the country than we ever had before and many of these are populated by students that graduated from here. The robotics scenario in the country, the research situation, on one hand looks bad when it comes to the amount of funding, but on the other hand if you look at the amount of technology and the impact that that's created is purely amazing because if you think about computer vision as part of robotics a lot of visualization, a lot of HCI type of technology came out of that as the core. So, robotics now is more diffused as an area and it's made a very big impact, I think.

Q: How do you see kind of culturally and socially the trajectory changes in the Robotics Institute? You've been here since almost the very beginning, so...

Pradeep Khosia: Every organization when you start there's a core group; there is a very strong camaraderie. There's only so many people that you have to interact with, that you have to remember, and you know just about everything about nearly everything that somebody else is doing so you're really well-informed, right. So, that was Robotics Institute. Today, the Robotics Institute is about I don't know \$60-\$50 million dollar a year operation. It has got components like the NREC, the National Robotics Center, it has components like social robotics. It has more faculty than even some departments at Carnegie Mellon, so it's bigger, it's more diffuse. Having said that, I don't remember everybody's name, which is unfortunate. I don't know what everybody else does, but I know approximately what's going on. But at the same time, it is high momentum, higher energy, it is more exciting. When I need to work with somebody in modeling graph dynamics, there is somebody, Jessica Hodgins, for example, I will talk to her and it's just more exciting.

Q: And, do you remember when the Ph.D. program started?

Pradeep Khosia: Yeah, actually I was involved in defining the Ph.D. program. Let's see if I can remember, I might have to look at my resume to figure out the date, but that was...

Q: How was it decided to start a Ph.D. program and who were some of the people who were involved in that?

Pradeep Khosia: So, there was committee – typically Carnegie Mellon was a very entrepreneurial place. You set oh getting a Ph.D. in robotics would be good so we put a committee together and I was one of the committee members, but we just defined a Ph.D. program, end of story. Unlike most universities where you'd spend like two years, the reality I think it was less than six months we defined a Ph.D. program.

Q: But, how did you decide that then was the time to have a Ph.D. program in robotics?

Pradeep Khosia: Because at that point, the field – see if you look at a Ph.D. program it has like two components mainly. One is what one would call the coursework component and the other is the thesis component. Now, at a place like Carnegie Mellon you could do the thesis like I did in electrical and computer engineering and you cannot tell the difference. But, we at that point, had the feeling that there were a set of core courses that described the domain and having some confidence in them was necessary, right. So, these were courses in perception, cognition, and manipulation was the three areas initially defined. Just like electrical engineering would define signal processing, solid state, so these were the three areas and so that's why we created the Ph.D. program. It was more to create a common base of knowledge on which you could build your research capability on.

Q: You've been in other, besides DARPA, you've been in other government policymaking, what would you call them?

Pradeep Khosia: I'm not.

Q: Agencies, well you were just talking about being with President McRobbie in ...

Pradeep Khosia: Oh so, DARPA I spent fulltime, nearly three years. Then, I am on the advisory board, for example, CSRIO Australia, NIST in the US. I was on the advisory board ofUCAV program, which was the Unmanned Combat Aerial Vehicle program where starting from

just autonomous flying machines to literally fighter aircraft being unmanned and autonomous. That was a program. So, I've served on advisory boards on several universities, government programs, government agencies, and including companies.

Q: What do you see are some of the future tones in robotics or where is the field going?

Pradeep Khosia: I think when the field started manipulation was the basis, I mean creating a robot that could do humanlike-talent assembly like window washing, whatever. Then it morphed into mobile vehicles, the autonomous vehicles and in doing so computer vision became a big part of robotics, but then at some point computer vision by itself had offshoots like visualization, like human computer interaction, so today I think of computer vision as robotics, but there are some people who think of that as computer vision. They don't think of that as robotics per se, right. So, robotics today is very diffuse, so in my mind – people ask me what's the definition of a robot. I say whatever you want it to be is the definition of a robot. It's like defining pornography, you know, when I see it I'll tell you. So, when I see a robot I'll tell you. When I see something I'll tell you if it's robotics or not.

Q: How do you see the robotics work in the US as compared to robotics, for example, in Europe? Are there particular differences?

Pradeep Khosia: There are differences, but there are also some – so I see the work in the US and in Japan both being extremely exciting. The difference is that in Japan there are companies that are committed to robotics and they developed technology which at a superficial level doesn't look like it has applications, but it has many applications. I'll give you an example. A company that developed a piano playing robot, now if you think about it, it is a nontrivial process to develop a completely articulated system that can play piano pretty much similar to a reasonable level expertise of a pianist, right. A company that developed robots as pets, I thought it was rather futuristic for somebody to think about that instead of having Chia Pets, which are these plants I could have a robot pet, which could have some emotional interaction with me if it's as simple as blinking a light, to smiling, right, pet dogs. You would not see that happen in the US. At least, I do not know of any company that would do it in the US, but there are these Japanese companies that would do it to create markets like the Sony dog robot. I mean that created a market. Sony made good money, so I think that's the difference. So, at the university level it's pretty much a similar type of research, but when it comes to some really creative, far out applications the companies are willing to spend. I also think when it comes to mechanical design; Japan focuses a lot more on the mechanical design of these systems than the US does.

Q: What about Europe or now Korea?

Pradeep Khosia: Korea, right, so they're all in play right now. So, I think the real big impact of robotics is going to be what we at Carnegie Mellon call quality of life where can we improve the quality of life of either disabled people or older people using these technologies. And, some of these technologies don't look like robots anymore. So, for example, a camera sitting in my mother's house in India so I can monitor if she's okay or not sitting in the US you might think it's not robotics, it's a camera, but the technology of computer vision, of processing that information was created in the context of robotics.

Q: Do you maintain your connections with IIT and have you seen the diffusion of robotic technology in India?

Pradeep Khosia: I have not seen as much. There is clearly more robotics in India today than there was 30 years ago, which was nonexistent, but as I look at India today, the focus is not as much on robotics technologies as on web-based systems, some IT systems. It's a whole lot of stuff, but I don't see great mechanical design research in robotics being done in India. I don't see the next generation software research being done in India in robotics right, where it's been done in Japan. It's been done in the US. It's been done in Europe, especially France and Germany. Those are the two major players I think.

Q: Okay, go ahead. Do you want to follow up on that? No, it's not a followup I was going to change to.

Pradeep Khosia: Is this going okay? I mean are you getting.

Q: This is awesome. So, you mentioned that you were on the advisory committee for the Combat Aerial Vehicle.

Pradeep Khosia: I was. The program died, they killed the program.

Q: What do you see are some of the social and ethical issues about arming autonomous robots?

Pradeep Khosia: I don't know if I see ethical issues the way we use robots today because even though we think of them as autonomous systems, there is still a human being behind the decision making process. They're not just making up their own decisions. So, what we have done is removed the human being much farther from the sphere of harm. So, if you think about Afghanistan and the drones that we used to shoot missiles, so instead of a pilot flying the aircraft and have the possibility of being shot at, now we are flying a drone with a missile loaded on it,

but back there somewhere is a human being who's going to make that decision on when to fire and who to fire on, right. It's not clear to me that there are any ethical issues in this context.

Q: But, if you're removing a human from the loop and having a totally autonomous system, do you see any issues there?

Pradeep Khosia: I can see us human beings as becoming nervous. I can see us human beings as thinking here's another society of intelligent inorganic systems that could take over and we have seen that in many Sci-Fi movies, but sometimes I wonder about if I look at the evolution of this world right, it's clear human beings were not the first creatures in the evolution process. There were single-celled creatures, multiple-celled creatures, and so on and so forth. Now, I don't know much about whether these creatures can think or not. I don't know what they felt when human beings came into being and started hunting and destroying these people, so there is a philosophical argument where some of us believe, not necessarily me, that human beings are the highest level of evolution, not clear to me because by that definition evolution would have to stop, so if I buy evolution I don't know what the next generation is going to look like, so I don't know.

Q: As an expert, would you be comfortable with having autonomous vehicles, I don't know a lot of different kinds of autonomous vehicles, a lot of different kinds of autonomous applications out there, both in civilian applications, also in military applications...

Pradeep Khosia: Yeah, actually I would be. So, for example, in civilian applications the DARPA urban challenge, right, so if you look at driving, more people die in road accidents than, for example, die in airplane accidents in this country and across the world. So, dying and the more people are maimed and disfigured and disabled because of road accidents, why would we not want to use technologies that have been demonstrated to be safe and not cause any harm to a third party that would improve our accident ratio, I mean that would reduce the number of accidents to zero. Why would not want to have a goal of zero accidents on the highways if autonomous systems – if we can create autonomous system technology that leads us there I think it's spectacular. You know, we as brothers, sisters, parents would never have to worry about our family member not coming home on time and wondering if he or she was in an accident. I see that to be a great thing. I have no ethical issues.

Q: Chuck was just telling us about one of their early demos of the autonomous land vehicle and they had put his three year old son in front of this thing as it was going. Of course, there was a person inside there, but do you think that there's a possibility of coming to the point where you could have like technology going with people passing through human controller in a sense?

Pradeep Khosia: Absolutely, I can see it.

Q: When do you think that – I know it's really hard?

Pradeep Khosia: So, I think there are two issues. One is does the technology exist to demonstrate that and is there enough of a wherewithal for us to accept that, right. Those are two different barriers. I think the technology exists today. It could be made robust enough in five years that on highway at cruising speeds we can see autonomous vehicles, right. But, for us human beings to have 100 percent confidence that nothing ever is going to go wrong, that bar is higher. So, that bar, I think in about 10 years we would see it on highways where we would have cars. In fact, we already see cars with infrared and radar for obstacle detection in the car, lane changing systems. These are all incremental technologies that are helping us make driving safer and the look on the time, and I think it's less than 10 years when these would be integrated and a car would be autonomous and I would be in the car being driven by the car, reading my email, texting, a lot safer situation than me driving and texting. <laughs>

Q: And, do you think in Japan there's a lot of what I see as kind of deliberate work being done by the government, by various companies to present robots as this thing that we can have in our daily life, do you see that as at all being something that's also being consciously done in the US?

Pradeep Khosia: Right, yes, what about these vacuum cleaner robots? You know, they look like these dust balls that run around collecting dust. It's a very simple technology, right, but just like the example of cars you take somebody does the vacuuming for you. Then the dishwasher is smarter, the refrigerator knows when there's no milk, and you're going to incrementally add it all together and now you have a house where you could be sitting here and on your way home your grocery list could be generated. You would approve it; some third party would deliver it, right. I mean there are companies that deliver groceries and your life; your quality of life will be very different. I think a lot better because now if you're interested in social sciences or in filmmaking you will have more time to spend on that instead of worrying about groceries, worrying about driving, worrying about a whole lot of other stuff.

Q: And, for a young person who's interested in robotics and creating robotics what would you tell them?

Pradeep Khosia: I'd tell them go for it. It is as good as it gets. It is very exciting. It's very open-ended and you get to define your future.

Q: I have one quick question that goes way back, but just cause it stuck with me, but I mean CMU obviously made a huge commitment to robotics that's different potentially from any place that I've seen; do you have a feeling for why they decided to make that commitment?

Pradeep Khosia: That's a really good question. It's actually part of CMU strategy. In the last 40 years, there has been no other university in the country that has come up so far and so fast. Literally from a regional school to an international university ranked in the top 10 in the world right now. This rise has not existed in any other school. And this started with Dick Cyert, who was our president in the 70s. He had this philosophy of comparative advantage. The idea is, we are a small school, we cannot do everything. But whatever we pick and choose to do, we would do it so well that we are in the top few in the world in that area. And this then led us to pick and choose areas and make the best of these areas. Robotics Institute was one of them. Developing or creating a School of Computer Science was the other. In another field, creating a Data Storage Systems Center, which is the premier center in the country doing data storage research. There is hardly a disk drive in the country today that does not have technology from that center. More recently creating CyLab. When we created CyLab we had no professor doing research in security, there was one who left, so we started from scratch and we built an operation that is the largest in the country in less than 5 years. We made a bit bet. And we've been fortunate that we've always won on these bets, we made an impact. One of these days one of these bets will go wrong but that's a risk we're willing to take.

Q: And do you know why robotics?

Pradeep Khosia: Because Raj Reddy had a vision of what this could do, just like I had a vision before 9/11 of cybersecurity and what this could do and nobody was thinking about it. Just like Mike Rider (sp?) had a vision of why storage technology was going to be as important as computing. You kind of bet on people, bet on their vision, and just make it happen.

Are we good? This is great, you just reminded me of my life. Actually I never think of who did what for a project, or how did that go in my career. Wow.