

Software wordt de bottleneck in robotica

Robot Operating System op de fabrieksvloer

Op de Roscon-conferentie afgelopen september heeft Microsoft een experimentele versie van Ros voor Windows gereleased. Traditioneel werkt dat robotbesturingssysteem alleen op Linux, dus hiermee is het toegankelijk geworden voor een grote nieuwe groep gebruikers. Saxion-onderzoeker Wilco Bonestroo vertelt op basis van praktijkervaringen over de mogelijkheden van Ros voor de industrie.

Wilco Bonestroo

Traditioneel worden robots ingezet voor repeterende taken en voor massaproductie. Het programmeren van een robot kost tijd en is kostbaar, maar door de grote serie is die investering acceptabel. Robots zouden echter ook voor kleinere series of voor complexere taken kunnen worden gebruikt. We zien hier twee uitdagingen: de uit te voeren taken en de software worden complexer én de systemen moeten in een kortere tijd worden ontwikkeld. Dat vraagt om een andere engineeringbenadering.

Om de ontwikkeling van software voor robots te versnellen, is het Robot Operating System bedacht. Ros is een platform met functionaliteit voor robotapplicaties. In Mechatronica&Machinebouw werd al in 2013 (!) de potentie van Ros beschreven. Vergeleken bij de stormachtige ontwikkelingen en acceptatie in robotonderzoek en servicerobots verloopt de acceptatie in de industrie een stuk voorzigtiger. Inmiddels heeft een aantal grote bedrijven zoals ABB, Microsoft, Universal Robots en Yaskawa zich aangesloten bij het Ros Industrial Consortium. Andere grote bedrijven zoals Adept, Fanuc en Kuka hebben dat nog niet gedaan, maar er zijn wel Ros-drivers beschikbaar voor hun hardware. Onder eindgebruikers zien we vooral grote bedrijven zoals Airbus, Boeing, BMW en Bosch die zich op Ros richten. Veel kleinere bedrijven zijn zelfs helemaal niet op de hoogte van de mogelijkheden.

Hardwareonafhankelijk

Het lectoraat Mechatronica van hogeschool Saxion werkt samen met bedrijven uit de regio aan vraagstukken uit de industrie. Met Benchmark Electronics uit Almelo en Bronkhorst uit Ruurlo wordt bijvoorbeeld onderzocht hoe cobots kunnen samenwerken met mensen.

In het project werken we met een ABB Yumi, een UR5 en een Aubo-I5. Deze cobots staan niet in een kooi en beschikken over geavanceerde 2d- en 3d-sensoren om hun omgeving te kunnen waarnemen. Cobots moeten met veel complexere situaties kunnen omgaan dan robots in een cel. In het project worden verschillende ontwikkelbenaderingen met elkaar vergeleken.

In een gerelateerd project hebben we gedemonstreerd dat – door taken op een hoger abstractieniveau te beschrijven – er geen afhankelijkheden meer zijn van een specifiek merk robot of van andere hardware. Hiervoor is een taak uitgevoerd op

zowel een UR5 als een Fanuc LR-Mate-robot, zonder hiervoor de software aan te passen. Er zijn slechts drivers geïnstalleerd op de controllers. Ros heeft modellen van de robots en communiceert met de robotcontrollers.

In Ros zijn veel bouwblokken beschikbaar, bijvoorbeeld voor autonome navigatie. Door deze te gebruiken, bouw je verder op bestaande functionaliteit, waaraan vaak jarenlange ontwikkeling is voorafgegaan. Saxion en Riwo Engineering uit Oldenzaal hebben een compact industrieel mobiel platform ontwikkeld om het gebruik van dergelijke bouwblokken te onderzoeken. Met het platform is aangetoond dat door



^ Door taken op een hoger abstractieniveau te beschrijven, wordt de software hardwareonafhankelijk zodat hij kan worden uitgevoerd op zowel een UR5 als een Fanuc LR-Mate-robot.



< Saxion en Riwo hebben een industrieel mobiel platform ontwikkeld om het gebruik van Ros-bouwblokken te onderzoeken.

slechts de hardware te beschrijven en een driver te ontwikkelen veel functionaliteit van Ros beschikbaar komt. De robot kon bijvoorbeeld autonoom navigeren en een kaart maken van de omgeving zonder dat daarvoor een regel code hoefde te worden geschreven en met gebruik van onderdelen van verschillende leveranciers.

Een van de uitdagingen bij het mobiele platform was de computerhardware. Ros draait op een reguliere pc met Linux of Windows. In dit project werden de motoren aangestuurd met een industriële pc met een realtime besturingssysteem. We hebben deze twee systemen gekoppeld. De pc bepaalt op basis van de sensordata en geavanceerde algoritmes de locatie van het platform, plant de route en stuurt opdrachten naar de ipc. Deze stuurt niet alleen de motoren aan, maar controleert ook of de pc opdrachten blijft sturen. Gebeurt dit niet, of duurt het te lang, dan grijpt de ipc in en brengt hij het systeem in een veilige toestand. Op deze wijze hebben we geavanceerde, complexe, niet-realtime software gecombineerd met een betrouwbare en realtime machinebesturing.

Kritische geluiden

De meningen zijn verdeeld over de vraag of Ros geschikt is voor industriële toepassingen. Ros is opensource en een veelgehoorde vraag is hoe betrouwbaar de software is. Deze vraag hebben we eerder gezien bij bijvoorbeeld Linux. Dat besturingssysteem is ook opensource en heeft een grote gebruikerscommunity. Hierdoor kan iedereen fouten vinden en oplossen, en functionaliteit toevoegen. Tegenwoordig wordt Linux gezien als zeer betrouwbaar. Het is de verwachting dat een opensource systeem met een grote community

op den duur beter zal worden dan een afgeschermd product met een veel kleinere community. Ros bestaat nu al enkele jaren en de veelgebruikte componenten zijn inmiddels stabiel en betrouwbaar. Ook is er een variant van Ros die specifiek is gericht op de industrie: Ros Industrial. Hier krijgen zaken als kwaliteit, stabiliteit en betrouwbaarheid extra aandacht.

Daarnaast horen we vaak de vraag of Ros wel realtime machinebesturing kan doen. Wij zien Ros echter niet als vervanging voor een plc, ipc of robotcontroller, maar als een toevoeging waarmee complexe functionaliteit toegankelijk wordt. Bij het ontwerpen van een oplossing is het van belang om te bepalen welke functionaliteit er nodig is en in welk systeem dat dan moet komen. Uitlezen van sensoren en aansturen van actuatoren kan goed op een ipc. Geavanceerde algoritmes zijn juist gemakkelijker op een pc. Ros is dan ook niet altijd nodig. Ervaren plc- of ipc-ontwikkelaars weten wat er mogelijk is met de huidige systemen. Vooral als een systeem gebruikmaakt van 2d of 3d perceptie, machine learning of andere artificial intelligence, en het moet kunnen omgaan met complexe situaties, dan biedt Ros veel bouwblokken om dit snel te realiseren. Heeft het systeem dit niet nodig, dan voegt Ros weinig toe en maakt het het systeem zelfs onnodig complex.

Korte ontwikkeltijd

Ros heeft een tamelijk uitdagende leercurve. Om echt voordeel uit het platform te halen, is er begrip nodig van een groot aantal concepten. Vooral voor technici met ervaring met Ladder of Structured Text is de benadering van Ros fundamenteel anders en ingewikkelder. Ros is dan

ook bedoeld voor een andere orde van problemen. Om alles uit Ros te halen, is het aan te raden om ook software-engineers in het ontwikkelteam op te nemen.

Om Ros te leren, is er op internet bijzonder veel documentatie beschikbaar. Daardoor is het soms lastig om te weten waar te beginnen. Lees een boek of volg een training, want deze zijn meer gericht op begrip en kennisontwikkeling. Zodra de basisbegrippen bekend zijn, is de online documentatie geschikt voor verdere verdieping. In Nederland kunnen tweedaagse trainingen worden gevolgd bij de TU Delft en vanaf 2019 ook bij hogeschool Saxion.

Ros is inmiddels de de facto standaard voor robotonderzoek en voor servicrobots. In de industrie wordt het op kleine schaal toegepast en dan vooral in proefopstellingen. Eerste praktische ervaringen tonen de voordelen aan zoals zeer korte ontwikkeltijden en hardwareonafhankelijke ontwikkeling. Betrouwbaarheid en robuustheid zullen zich de komende tijd moeten bewijzen. Wij hebben de visie dat Ros traditionele plc's, ipc's of robotcontrollers niet zal vervangen, maar dat het ernaast zal worden gebruikt. Ros heeft toegevoegde waarde voor complexe systemen met perceptie waarin geavanceerde taken worden uitgevoerd die afhankelijk zijn van de omgeving.

Wilco Bonestroo is senior onderzoeker bij het lectoraat Mechatronica van hogeschool Saxion in Enschede. Hij heeft ruime ervaring op het gebied van software-engineering en embedded systemen. Momenteel richt hij zich op nieuwe ontwikkelingen op het gebied van artificial intelligence en advanced perception voor industriële toepassingen.