

Botsingsvrije trajectgeneratie voor lasrobots: analyse en verbetering van het Descartes algoritme

Bart Moyaers

Giel Vanvelk

Master IW Energie: Automatisering

Master IW Energie: Automatisering

Omschrijving/Situering

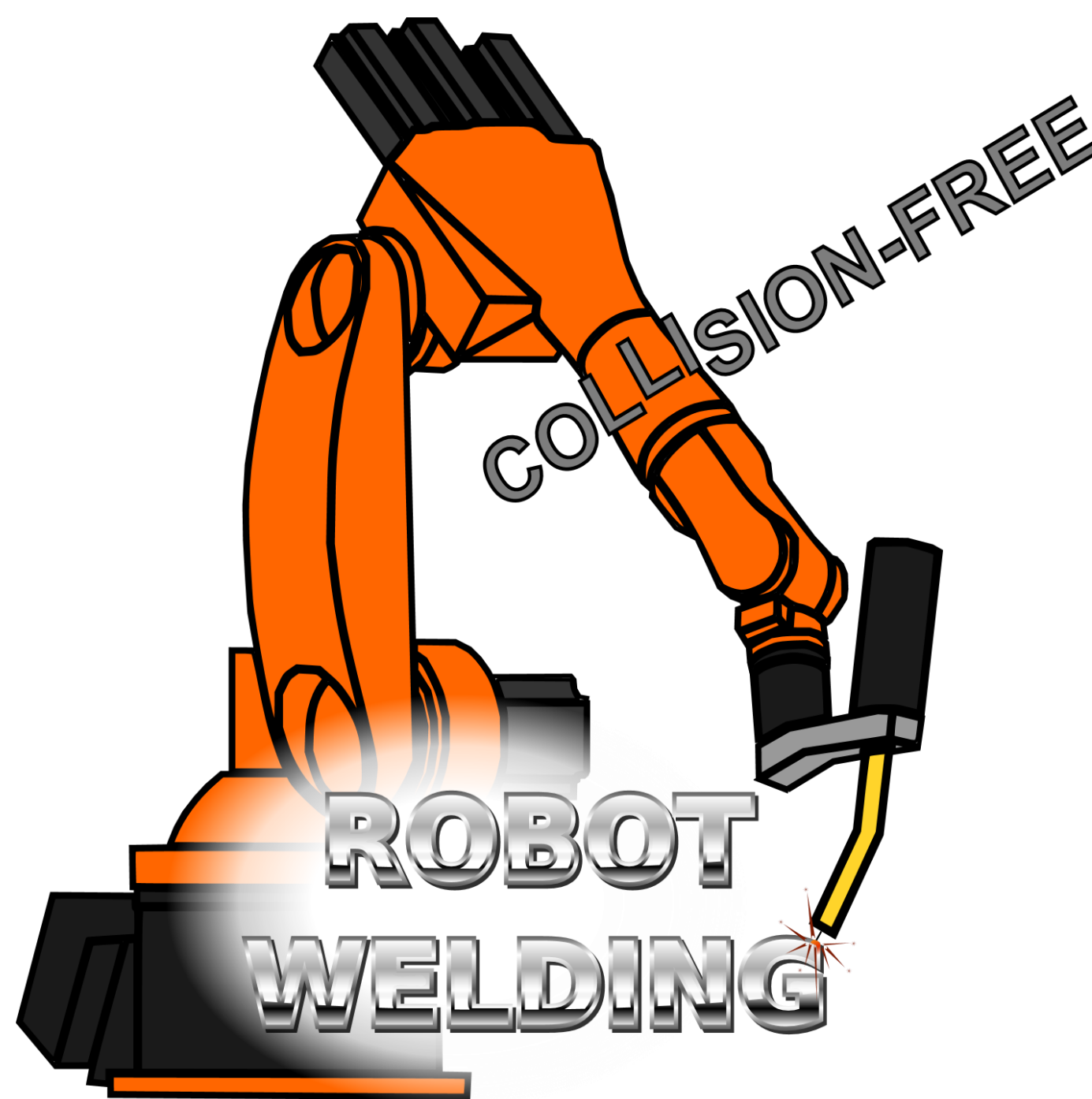
Deze masterproef houdt zich bezig met het automatiseren van lasrobots. Het doel is om voor deze robots automatisch bewegingstrajecten te genereren. Hierbij moet er rekening gehouden worden met mogelijke botsingen van de robot met zichzelf en de omgeving. Om een optimale las na te streven, is het belangrijk om de trajectgeneratie te beheersen. Dit gebeurt onder andere door rekening te houden met bepaalde lasparameters zoals de lassnelheid en verschillende lashoeken.

Doelstellingen

1. Selecteer en evalueer een bestaand open-source softwarepakket voor het genereren van eenvoudige bewegingstrajecten van robots.
2. Breid de software uit zodat:
 - De trajectgeneratie rekeninghoudt met eventuele botsingen
 - Er gebruik gemaakt kan worden van toleranties om niet optimale trajecten te genereren.
3. Maak de software beschikbaar voor derden.

Verwijzingen

Github: https://github.com/Bart123456/lasrobot_ws.git
 ROS: <http://www.ros.org/>
 Descartes-ROS: <http://wiki.ros.org/descartes>



Literatuurstudie

De literatuurstudie vergeleek drie verschillende softwarepakketten: CHOMP, TrajOpt en het Descartes-softwarepakket. Uiteindelijk werd gekozen om verder te werken met het Descartes-softwarepakket.

Evaluatie

In de evaluatie van Descartes kwamen een aantal onvolledigheden aan bod. Deze onvolledigheden werden aangepakt gedurende de uitbreiding

Uitbreiding

In de uitbreiding werd de botsingsdetectie en het gebruik van toleranties met de samenhangende kostenfunctie aangepakt. Het resultaat van deze uitbreidingen kan u hieronder zien.

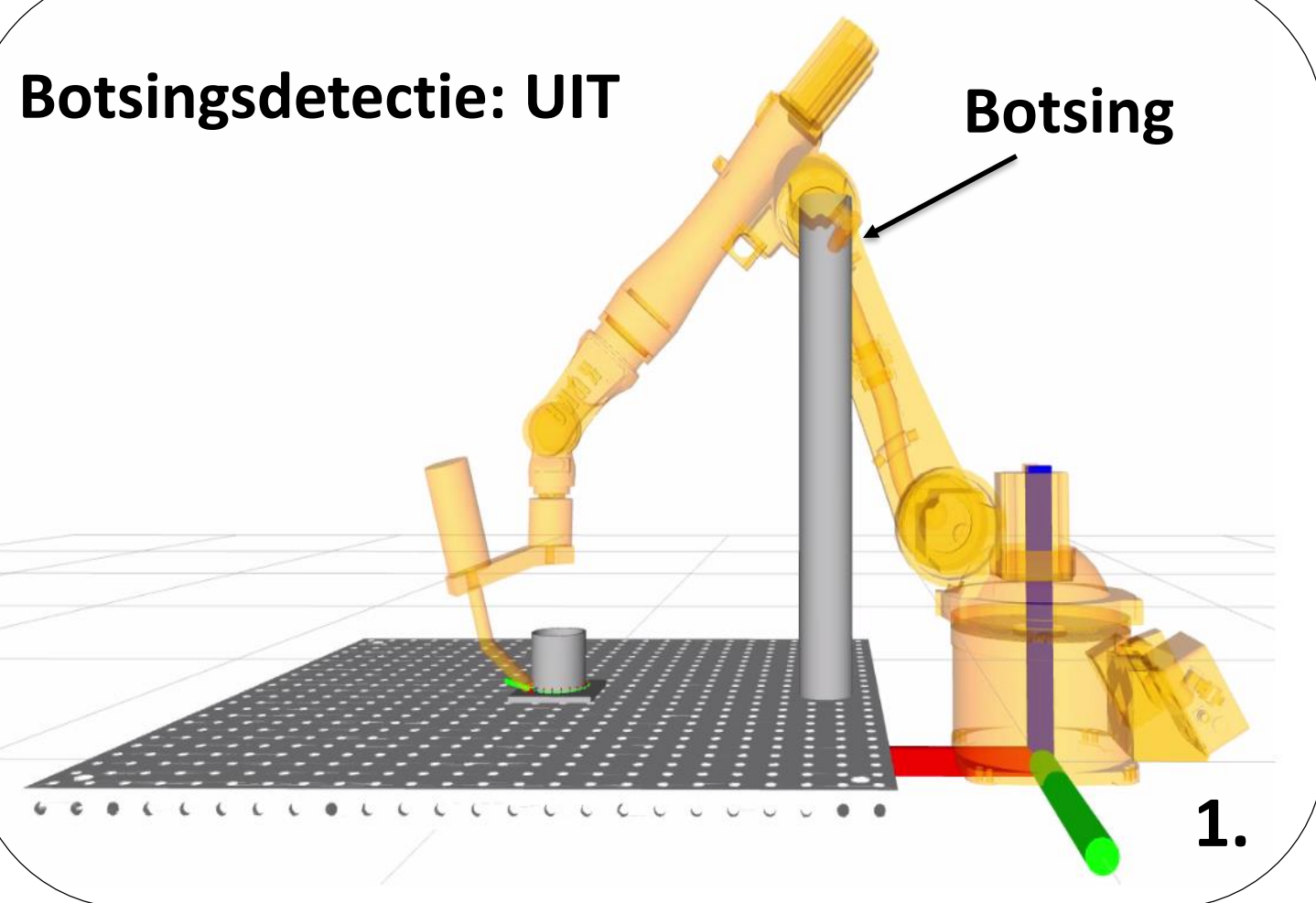
Documenteren

De volledige code van het programma is terug te vinden op Github (zie verwijzingen). Verder zal er nog een tutorial worden geschreven.

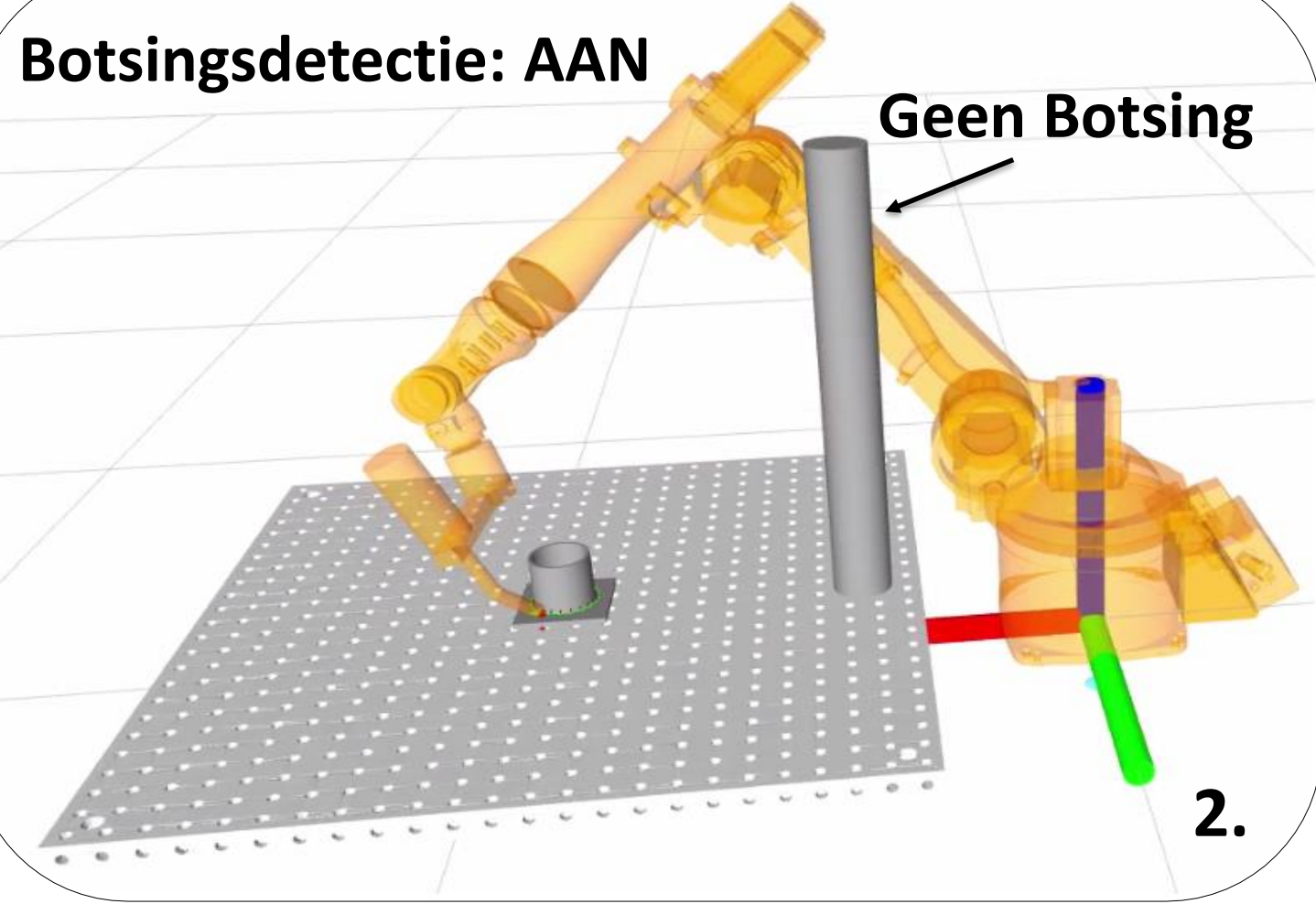
Botsingsdetectie

Descartes gebruikt de bestaande botsingsdetectie uit MoveIt!. Echter detecteerde die enkel botsingen van de robotarm met zichzelf. De code van Descartes is aangepast zodat hij ook controleert op botsingen tussen de robotarm en zijn omgeving, zoals bijvoorbeeld werkstukken.

Botsingsdetectie: UIT



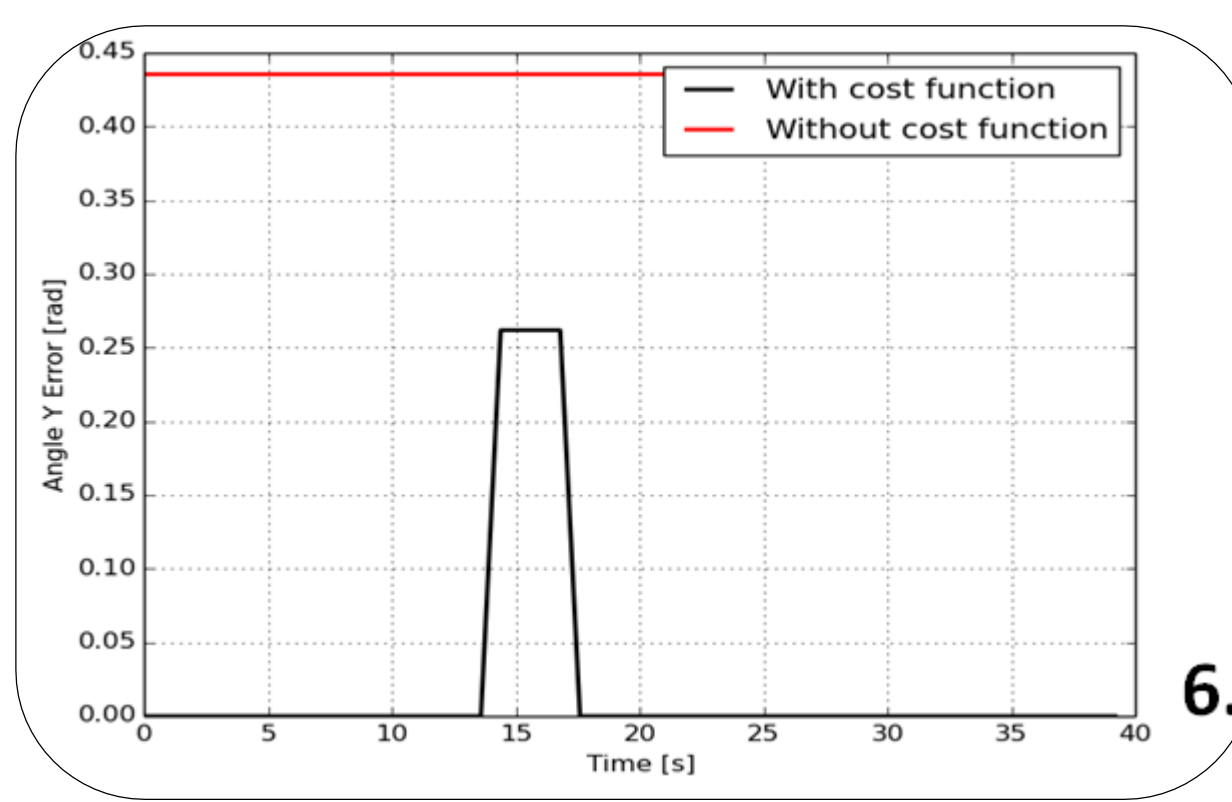
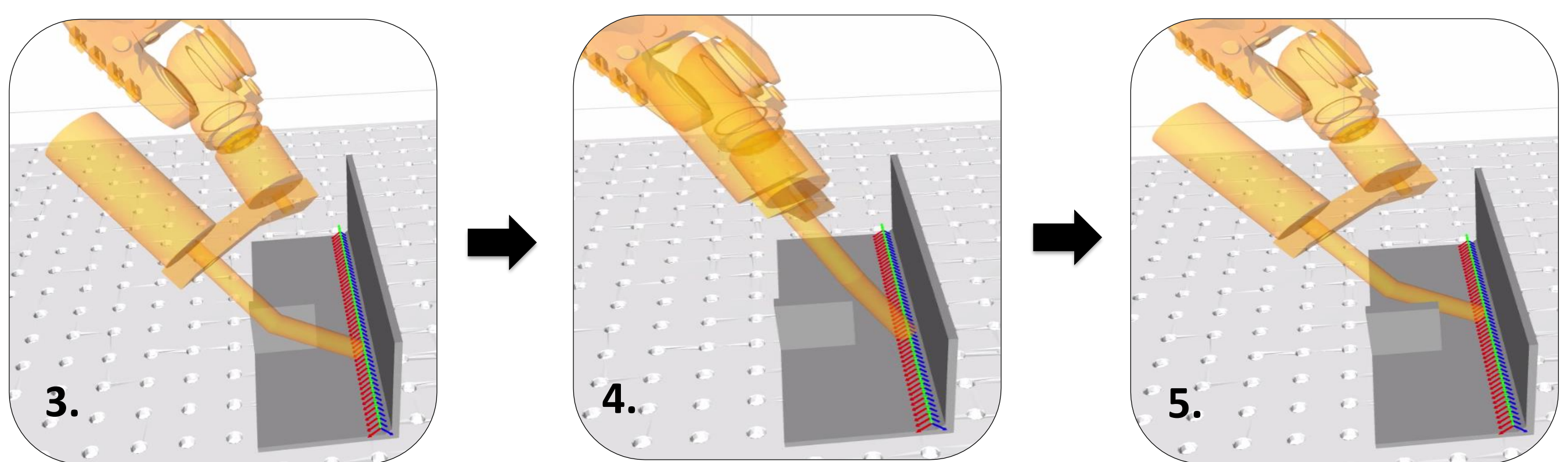
Botsingsdetectie: AAN



Toleranties & Kostenfunctie

Het instellen van toleranties zorgt ervoor dat de robot van zijn nominaal traject mag afwijken. Aan de fout gevolgd door deze afwijking hangt het voordeel vast dat er meer mogelijkheden zijn om een bewegingstraject te genereren. In figuur 3 t.e.m. 5. wordt een werkobject gelast waarbij de lastoorts midden in zijn traject (figuur 4) in botsing komt met een balk. Zonder toleranties zou er voor dit traject onmogelijk een beweging gegenereerd kunnen worden. Door verschillende toleranties toe te staan, mag de lastoorts van het optimale traject afwijken en kan de software toch een beweging genereren. De code van Descartes is aangepast om rotationele toleranties praktischer en intuïtiever te maken.

Om er voor te zorgen dat de software zorgvuldig omspringt met de toleranties en dus niet meteen gebruik maakt van de grootste beschikbare tolerantie (grootste afwijking) is er een kostenfunctie opgesteld die een kost meegeeft aan elke afwijking ten opzichte van het nominale bewegingstraject. Descartes probeert deze kost zo laag mogelijk te houden waardoor een zo laag mogelijke tolerantie gebruik wordt. Figuur 5 en 6 tonen aan dat Descartes terug keert naar zijn nominale pad (kleinste tolerantie) nadat het hiervan noodzakelijk heeft afgeweken. De software streeft zo een optimale las na.



In figuur 6 is de hoekafwijking ten opzichte van het nominale traject uitgezet in functie van de tijd. De groene curve geeft de hoekafwijking weer indien er geen kostenfunctie wordt gebruikt. Bij de blauwe curve is de kostenfunctie wel actief.

Zoals te zien is de totale hoekafwijking (oppervlakte onder de curve) zonder kostenfunctie veel groter dan met kostenfunctie.

Promotoren / Copromotoren: Prof. Dr. Ir. Eric Demeester
 Ing. Maarten Verheyen
 Ir. Jeroen De Maeyer