



Drone mesh netwerk simulatie

Onderzoeksrapport

HAN Arnhem

561399

MWJ.Berentsen@student.han.nl

Versie 1

Alten Nederland B.V.

Docent: J. Visch, MSc

Assessor: ir. C.G.R. van Uffelen

M.W.J. Berentsen

12 maart 2019

Inhoudsopgave

1	Inleiding	3
2	Hoofd- en deelvragen	4
2.1	Hoofdvraag	4
2.2	Deelvragen	4
2.3	Onderzoeksmethode	5
2.3.1	De gekozen onderzoeksmethodes	5
2.4	Invloed op het project	6
3	Criteria	7
4	Literatuur	9
4.1	Welke simulatie software is geschikt om een drone in na te bootsen?	9
4.1.1	Gazebo of Webots?	10
4.1.2	Conclusie	10
4.2	Wat is de grootte van de payload die een netwerkmodule aangesloten op een drone moet versturen om nuttig te zijn voor een aansturingssysteem?	11
4.2.1	Binaire serialisatie voor het samenstellen van berichten	11
4.2.2	Wat is de grootte van een bericht voor de locatie bepaling van drones?	12
4.2.3	Wat is de grootte van een bericht waarin een netwerkmodule zijn verbinding met anderen aangeeft	14
5	Oplossingsrichtingen	15
6	Experimenten	16
6.1	Wat is minimaal benodigd in een simulatie om abstracte drone te representeren?	16
7	Conclusie	17
	Literatuur	18

Samenvatting

Optioneel een samenvatting van het onderzoek. Hier kunnen anderen snel inzicht krijgen in wat jij hebt onderzocht en wat je conclusie is.

- Kunnen derden snel inzicht krijgen in jouw onderzoek?
 - Staat de conclusie erin vermeld?
-

1 — Inleiding

De inleiding beschrijft:

- Waarvoor het onderzoek gedaan wordt;
- Beschrijf waarom het onderzoek nu wordt uitgevoerd;
- Het doel van het onderzoek.

Het volgende onderzoek rapport wordt uitgevoerd ten behoeve van de het afstudeerproject van Maurice Berentsen, Hogeschool van Arnhem en Nijmegen.

2 — Hoofd- en deelvragen

In dit hoofdstuk worden de hoofd- en deelvragen genoemd en onderbouwd. Er wordt een scope bepaalt met wat er onderzocht wordt en dus ook wat niet. Vervolgens wordt de onderzoeksmethode toegelicht en beargumenteerd. Tenslotte wordt de invloed van het onderzoek op het afstudeerproject beschreven.

2.1 Hoofdvraag

Het doel van dit onderzoeksrapport is het beantwoorden van de volgende opgesteld hoofdvraag:

Hoe kan een netwerkmodule een groep van ongeveer 100 drones voorzien van een onderling meshnetwerk die snel reageert op uitval van netwerkpunten?

Doordat de uitvoerende student van dit onderzoek niet beschikt over een certificaat om met drones te mogen vliegen behoudt dit onderzoek zich tot een simulatie. Van de netwerkmodule wordt wel een fysiek prototype ontwikkeld.

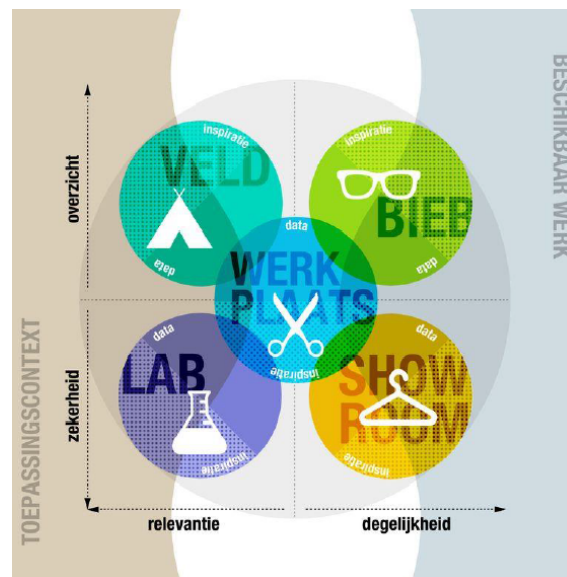
2.2 Deelvragen

Om de hoofdvraag te kunnen beantwoorden zijn er de volgende deelvragen opgesteld:

- Welke simulatie software is geschikt om een netwerk van ongeveer 100 drones in na te bootsen?
- Wat is de grote van de payload die een netwerkmodule aangesloten op een drone moet versturen om nuttig te zijn voor een aansturingssysteem?
- Welke mesh netwerk hardware is geschikt voor het onderling verbinden van drones?
- Welke mesh netwerk library is geschikt voor de te gebruiken hardware?
- Hoe gedraagt een node uit het mesh netwerk zich bij slecht of geen bereik?
- Wat kan er toegevoegd worden aan de communicatie van de netwerkmodule om snel [TODO snel definiëren] achter uitval te komen?
- Wat is minimaal benodigd in een simulatie om abstracte drone te representeren?

2.3 Onderzoeksmethode

De onderzoeksmethodes volgen de structuur van de *ICA methodenkaart* (Van Turnhout et al., 2014). De methodenkaart is een onderzoek framework voor professionals in ICT en media. Het ontwerp van het framework gaat uit van het idee van triangulatie. Triangulatie is het combineren van verschillende theorieën, methoden of databronnen om zo tot betere antwoorden te komen op je onderzoeksvragen (Oates, 2005). Door het definiëren van zo geheten werkplaatsen en hun samenhang geeft het de onderzoeker een set van methoden en technieken. De vijf verschillende werkplaatsen binnen het framework zijn: *lab*, *veld*, *showroom*, *bieb* en *werkplaats*.



Figuur 2.1: De vijf werkplaatsen binnen het framework en hun systematiek. Overgenomen uit *De informatieprofessional 3.0* van Van Turnhout et al. (2014)

2.3.1 De gekozen onderzoeksmethodes

De deelvragen gaan gedeeltelijk over keuzes van gebruik. Welke software en hardware is geschikt, is een vraag die veel naar voren komt. Hieruit valt te concluderen dat de onderzoeker zich nog aan het oriënteren is. "De onderzoeksruimte bieb bevat een verzameling methoden en technieken die dienen tot oriëntatie op beschikbaar werk." (Van Turnhout et al., 2014) Daarom wordt er gekozen om een bieb onderzoek uit te voeren naar de deelvragen die gaan over geschiktheid. Dit zodat de onderzoeker zich nog kan oriënteren naar het beschikbare aanbod en nog op nieuwe deelvragen kan komen.

Lab onderzoek wordt pas later uitgevoerd omdat niet alle hardware voor handen is, maar ook omdat er niet genoeg tijd voor beschikbaar is. Van Turnhout et al. zegt het volgende over lab onderzoek: De onderzoeksruimte lab bevat methoden die geschikt zijn om de oplossing te toetsen aan een aspect van de toepassingscontext. Daarom is het lab onderzoek geschikt voor het moment waarop de hardware beschikbaar is en de onderzoeker het gedrag hiervan exact in kaart wil brengen en wil bewijzen dat de gestelde eisen behaald worden.

Oates (2005) stelt dat een systeem bouwen 'bewijs' is dat een nieuw soort toepassing gebouwd kan worden. Als dit het doel is heeft het onderzoek waarschijnlijk hoofdzakelijk een werkplaats karakter, maar alle andere onderzoeksruimtes kunnen een rol spelen. Gedurende het huidige onderzoek zal er door de onderzoeker software geschreven worden om bewijs te leveren dat een netwerkmodule een groep drones kan voorzien van een onderling netwerk. Daarom past de werkplaats onderzoeksruimte goed bij dit onderzoek.

2.4 Invloed op het project

3 — Criteria

Maak hier een lijst met criteria aan de hand van de hoofd- en deelvragen.

- Zijn de criteria duidelijk opgesteld?
- Waar moet de oplossing aan voldoen, staat dit erin?

In [Tabel 3.1](#) worden codes gebruikt als naam voor de criteria, wat deze betekenen wordt eerst toegelicht:

- ALG: Een algemene eis die in elke onderzoek meegenomen moet worden.
- SK: Een eis die gesteld worden in de keuze naar simulatie software.
- DR: Een eis voor de te gebruiken drone in de simulatie.
- PH: Eisen opgesteld aan het hardware prototype.

Naam	Beschrijving
ALG1	Is voorzien van API documentatie.
ALG2	Ondersteunt aansturing vanuit C++ .
ALG3	Heeft ondersteuning voor het Linux platform Ubuntu 18.04
ALG4	Software is gratis in gebruik voor studenten
SK1	Heeft ondersteuning voor UAV (unmanned air vehicle) ook wel drone genoemd
SK2	Ondersteunt de simulatie van locatie bepaling sensoren zoals GPS.
SK3	Heeft een kant en klare oplossing voor simulatie van externe krachten zoals wind.
SK4	Heeft een ingebouwde pathfinding oplossing.
SK5	Ondersteunt ROS als middleware.
SK6	Ondersteunt de detectie van botsingen.
SK7	Ondersteunt de simulatie 100 drones tegelijk.
DR1	De drone is een quadcopter.
DR2	De drone is is voorzien van een API voor aansturing op basis van coördinaten.
DR3	De API sluit aan op de middleware ROS melodic.
DR4	De drone is bruikbaar binnen de gekozen simulatie software.
DR5	De drone moet een kleine goedkope drone representeren.
PH1	Ondersteunt een bereik van minimaal 100 meter.
PH2	Ondersteunt het gebruik van grid networking om tot een mesh te kunnen komen.
PH3	Maakt gebruik van een openbare bandbreedte.
PH4	De antenne mag niet meer dan 10 euro kosten.
PH5	Moet zuinig in gebruik van stroom zijn. TODO beter definiëren
PS1	Ondersteund een mesh netwerk van minimaal 100 nodes.

Naam	Beschrijving
PS2	Het netwerk is zelf herstellend.
PS3	Ondersteunt het draaien op een raspberry pi of een arduino (UNO of NANO).
PS4	Ondersteunt het versturen van zelfgemaakte berichten.

Tabel 3.1: Opgestelde criteria

4 — Literatuur

4.1 Welke simulatie software is geschikt om een drone in na te bootsen?

Voor het onderzoek wordt gebruik gemaakt van een simulatie omgeving. Welke software wordt gebruik staat vastgesteld in deze paragraaf. De keuze van software wordt bepaald door het uitvoeren van de bieb onderzoeksmethode. In de [Tabel 3.1](#) staan de criteria toegelicht waar de simulatie software aan moet voldoen. Als hoofdcriteria wordt een lijst opgesteld met simulatoren die ondersteuning hebben voor ROS. Vervolgens worden deze aan de hand van een kruistabel onderworpen aan de andere criteria. Deze techniek is gebaseerd op de "Comparison Chart" van (Van Turnhout et al., z. j.) uit de "*CMD Methods Pack*"

De kandidaten zijn:

- **Actin** is een simulatie framework van het bedrijf Energid. Het is in staat om de beweging van verschillende robots gelijktijdig aan te sturen.
- **Gazebo** is een opensource robot simulatie framework bijzonder geschikt voor het simuleren van robotica in outdoor omgevingen door de uitgebreide Physics Engine Support.
- **V-REP (Virtual Robot Experimentation Platform)** is een platform geschikt voor het snel bouwen van robot prototypes.
- **Webots** is een open source-ontwikkelomgeving die wordt gebruikt voor het modelleren, programmeren en simuleren van mobiele robots.
- **OpenRAVE** biedt een ontwikkelomgeving voor het testen van motion planning algoritmes aan de hand van simulaties.

Simulator \ Eis	ALG1	ALG2	ALG3	ALG4	SK1	SK2	SK3	SK4	SK5	SK6
Actin	✓	✓	✓	x	✓	x	✓	x	✓	✓
Gazebo	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
V-REP	✓	x	✓	✓	✓	✓	✓	✓	✓	x
Webots	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
OpenRAVE	✓	✓	✓	✓	x	✓	✓	✓	✓	x

Tabel 4.1: Kruistabel simulatiekeuze

In de [Tabel 4.1](#) komt de simulatie software van Gazebo en Webots naar voren als enige simulatoren die voldoen aan alle gestelde eisen. Het is nu zaak om te uit te zoeken welke van de twee het meest geschikt is voor het uitvoeren van dit onderzoek.

4.1.1 Gazebo of Webots?

In januari 2018 hebben Pitonakova, Giuliani, Pipe en Winfield een publicatie geschreven waarin zij drie simulatoren vergelijken in het aanbod van features en performance. Zij hebben een vergelijking gedaan tussen V-REP, Gazebo en ARGoS. ARGoS is niet meegenomen in de overweging omdat het geen ondersteuning heeft voor ROS.

In de conclusie van de publicatie komt Webots als het meest gebruiksvriendelijk naar voren in het gebruik en is in het bezit van de meeste features. Het grote nadeel van Webots is dat het veel kracht vraagt van de computer waardoor het niet geschikt is voor de simulatie van veel drones tegelijk.

Daarmee komt de keuze uit op Gazebo. De resultaten in de publicatie van Pitonakova et al. (2018) tonen aan dat de software in staat is om veel drones tegelijk aan te kunnen sturen. Pitonakova et al. (2018) geven als nadeel aan Gazebo dat de software niet in staat is om 3d meshes te manipuleren, de user interface niet innovatief is en Gazebo soms problemen met dependencies door de verschillende versies van ROS. Dit laatste brengt een risico met zich mee voor het onderzoek daarom is het ook zaak dat dit als criteria meegenomen word in de deelvraag van [Paragraaf 6.1 Wat is minimaal benodigd in een simulatie om abstracte drone te representeren?](#).

4.1.2 Conclusie

Voor het onderzoek wordt Gazebo gebruikt omdat het voldoet aan de gestelde [Criteria](#). De meest recente versie van Gazebo, die niet meer beta is op 19 februari 2019, is Gazebo 9. Deze versie vereist het gebruik van ROS Melodic volgens haar eigen website ([zie link](#)).

4.2 Wat is de grootte van de payload die een netwerkmodule aangesloten op een drone moet versturen om nuttig te zijn voor een aansturingssysteem?

De payload is het gedeelte van een bericht die vrij is naar de gebruiker om invulling te geven. De maximale grootte van een payload verschilt per protocol. Om een keuze te kunnen maken waar de hardware en het bijhorende protocol aan moet voldoen is het dus zaak om eerst te weten hoe groot deze vrije payload ruimte moet zijn.

Het doel van de berichten is om ze zo kort als mogelijk samen te stellen. Daarom is het belangrijk om de structuur van de berichten zo klein mogelijk te houden. Om deze reden wordt niet gekeken naar een tekstuele serialisatie mogelijkheden zoals XML, YAML, JSON maar naar binaire mogelijkheden omdat deze veel efficiënter zijn (Tauro, Ganesan, Mishra & Bhagwat, 2012).

4.2.1 Binaire serialisatie voor het samenstellen van berichten

"Binary Serialization is converting the object in binary format and being able to store it in a storage medium" (Tauro et al., 2012). Het binair samenstellen van informatie is een efficiënte manier omdat de data met een minimale overhead wordt samengesteld (Castillo, Rosales & Blanco, 2018) Het nadeel die deze wijze met zich meebrengt is dat het resultaat van het bericht niet meer makkelijk leesbaar is voor mensen. Voor dit project ligt daar geen focus op dus is dit nadeel te verwaarlozen.

Castillo et al. (2018) hebben onderzoek gedaan naar een efficiënte manier naar de inzet van binaire berichten te vervanging van tekstuele serialisatie. Hoewel dat in het huidige onderzoek niet het geval is kan er wel geleerd worden van de manier van bericht gebruik. Castillo et al. (2018) tonen in hun onderzoek aan dat berichten vaak een herhalende structuur hebben en dat dit een onnodige overhead creëert in de berichten. Een voorbeeld maken Castillo et al. (2018) zichtbaar in [Figuur 4.1](#) waarover zij het volgende zeggen: "While JSON compared with XML, eliminated parameter name redundancy, it keeps repeating (red box) the definition of each field for each serialized element (green box)".

The figure shows two side-by-side code snippets. The left snippet is a JSON message from 'http://localhost:8080/Json/SyncReply/Contacts'. It contains a list of contact objects, each with 'FirstName', 'LastName', and 'Email' fields. A red box highlights the repeated field names for each contact, and a green box highlights the values. The right snippet is an XML message from 'http://localhost:8080/XML/SyncReply/Contacts'. It uses a more structured approach with a root 'ContactsResponse' element containing a 'Contacts' element, which then contains individual 'Contact' elements. Each 'Contact' element has its fields (Email, FirstName, LastName) defined with explicit XML tags. A red box highlights the repeated field names for each contact, and a green box highlights the values. The XML structure is more verbose than the JSON structure.

```
http://localhost:8080/Json/SyncReply/Contacts
{
  "Contacts": [
    {
      "FirstName": "Dennis",
      "LastName": "Bellot",
      "Email": "dennis.bellot@gmail.com"
    },
    {
      "FirstName": "Steve",
      "LastName": "Jobs",
      "Email": "steve@apple.com"
    },
    {
      "FirstName": "Steve",
      "LastName": "Ballmer",
      "Email": "steve@microsoft.com"
    },
    {
      "FirstName": "Eric",
      "LastName": "Schmidt",
      "Email": "eric@google.com"
    },
    {
      "FirstName": "Larry",
      "LastName": "Ellison",
      "Email": "larry@oracle.com"
    }
  ]
}

http://localhost:8080/XML/SyncReply/Contacts
<ContactsResponse xmlns:i="http://www.w3.org/2003"
  <Contacts>
    <Contact>
      <Email>dennis.bellot@gmail.com</Email>
      <FirstName>Dennis</FirstName>
      <LastName>Bellot</LastName>
    </Contact>
    <Contact>
      <Email>steve@apple.com</Email>
      <FirstName>Steve</FirstName>
      <LastName>Jobs</LastName>
    </Contact>
    <Contact>
      <Email>steve@microsoft.com</Email>
      <FirstName>Steve</FirstName>
      <LastName>Ballmer</LastName>
    </Contact>
    <Contact>
      <Email>eric@google.com</Email>
      <FirstName>Eric</FirstName>
      <LastName>Schmidt</LastName>
    </Contact>
    <Contact>
      <Email>larry@oracle.com</Email>
      <FirstName>Larry</FirstName>
      <LastName>Ellison</LastName>
    </Contact>
  </Contacts>
</ContactsResponse>
```

Figuur 4.1: JSON and XML messages. Overgenomen uit *International Journal of Computer Applications* van Castillo et al. (2018)

Wat zij hebben gedaan om dit op te lossen is het apart versturen van de parameternamen zodat deze in de opvolgende berichten van dat type niet meer nodig is. Het systeem die bedacht is geeft een enorme flexibiliteit omdat er van te voren nog geen bericht types bekend hoeven te zijn. Deze flexibiliteit is alleen niet van toepassing op het doel van dit project, daarom kan de student in zijn software al van te voren berichttypes samenstellen waardoor er de overhead nog kleiner gemaakt wordt. Voor nu wordt er vanuit gegaan dat 1 byte genoeg zal zijn omdat de communicatiesoftware van de netwerkmodule niet meer dan 256 (2^8) berichtentypes hoeft te ondersteunen. Mocht het blijken dat dit toch niet genoeg is kan er een byte toegevoegd worden waardoor er 65536 (2^{16}) types mogelijk zouden zijn. Daarnaast moet een bericht om nuttig te zijn voor het netwerk altijd aan een apparaat gekoppeld kunnen worden. De opdracht gaat over een netwerk van ongeveer 100 drones dit past ruim in een byte.

Hiermee komt het eerste deel van het bericht dus al tot stand het telt nu minimaal 2 bytes.

ID	Berichttype	overig
00 tot FF	00 tot FF	?

Tabel 4.2: Berichtsamenstelling met type aanduiding

4.2.2 Wat is de grootte van een bericht voor de locatie bepaling van drones?

Om duidelijk te maken wat nuttig voor het systeem wordt als eerste de basis informatie vastgesteld. De huidige locatie is essentieel voor de verdeling van de drones. Omdat de drones zich kunnen verplaatsen is naast de locatie ook het tijdstip waarop het bericht verstuurd wordt van belang.

Coördinaat vanuit de informatie van de GPS module

Een positie van een locatie wordt aangeduid door coördinaten te gebruiken. Deze worden bepaald door het gebruik van een GPS module aan boord van het systeem. GPS modules gebruiken triangulatie van satelliet signalen om hun positie te bepalen.

Een coördinaat is opgebouwd door de aarde te verdelen in graden over de assen latitude (horizontaal) en longitude (verticaal). Een coördinaat loopt van -180 tot 180 graden. De aarde heeft een omtrek van 40.07.5161,2 meter gedeeld door 360 graden levert dit per graad een stap op van 111,32 kilometer. Het volgende programma laat zien dat een float gebruikt kan worden in dit geval voor een accuraatheid van 5 cijfers achter het decimaal.

```
#include <stdio.h>

float exampleFloatPositive = 179.1234567890;
float exampleFloatNegative = -179.1234567890;

int main( int argc, const char* argv[] )
{
    printf( "positive float = %f\n", exampleFloatPositive );
    printf( "negative float = %f\n", exampleFloatNegative );
}
```

Door een coördinaat te gebruiken van 5 cijfers achter het decimaal kan een plek op aarde met een precisie van ongeveer 1,10 meter nauwkeurig aangeven worden. Het is gebruik hiervan is realistisch omdat met alleen realtime GPS data de accuraatheid ongeveer 4 meter is (U.S. Government, 2008). Er wordt per as een float gebruikt dit houdt in dat minimaal 8 bytes nodig zijn voor een coördinaat. Dit is 4 bytes per float en exclusief de separator.

Hoogte van de drone

Kulhavy, Hung, Unger en Zhang (2017) hebben onderzoek gedaan naar de accuraatheid van de hoogtebepaling bij drones. Uit dit onderzoek is gebleken dat een continue vliegende drone een accuraatheid heeft van ongeveer twee meter. Hieruit is te halen dat het geen zin heeft om op centimeter niveau de hoogte van een drone aan te geven. Op het moment dat er een signed integer van 16 bits gebruikt zou worden kan er een afstand van 32.767 meter zowel boven als onder zeeniveau verstuurd worden. Het is een realistische waarde om te gebruiken aangezien de grootste hoogte die gebruik kan worden dan 32 kilometer is. Deze hoogte is al ver in de stratosfeer en al boven het niveau de weerballon halen. De keuze om een signed integer te gebruiken is om ondersteuning te bieden aan plekken onder het zeeniveau waarvan Nederland een groot voorbeeld is.

Voor de hoogte moeten 2 bytes gereserveerd worden.

Tijd vanuit de GPS module

De drone maakt al gebruik van de GPS module voor de positiebepaling. In de berichten die de GPS module ontvangt van de satellieten wordt ook de tijd meegestuurd. Deze tijd is volgens (U.S. Government, 2008) in 95 procent van de gevallen accuraat tot 40 Nano seconden. In het geval van het drone netwerk is dit veilig te gebruiken als bron voor tijd. De tijd zal aangegeven worden in het unix time format zodat deze meteen de datum met zich draagt. Unix time maakt gebruik van 32 bits formaat, er is gekozen voor een unsigned variant aangezien er geen interesse is naar data voor 1 januari 1970 ook wel epoch genoemd.

Voor de tijd en datum worden 4 bytes gereserveerd.

Conclusie

De bovenstaande sub paragrafen kunnen samengesteld worden tot een bericht waarin alle informatie zit die nodig is om te bepalen waar een drone zich op een moment bevindt. Een systeem kan met dit bericht in ieder geval elke drone afzonderlijk in kaart brengen.

	ID	Bericht type	latitude	longitude	hoogte	tijd
Aantal Bytes	1	1	4	4	2	4

Tabel 4.3: Locatie bericht in bytes

In 4.3 is het aantal bytes te zien benodigd voor een locatie bericht. Het totaal aantal bytes voor dit bericht is 16 bytes. Later moet er gekeken worden of iets aan een bericht

toegevoegd kan worden om uitval sneller te detecteren. Desondanks het nog niet bekend is wat dit is kan er vanuit gegaan worden dat er niet meer dan 4 bytes nodig zullen zijn.

Concluderend wordt voor dit bericht een grootte van 20 bytes aangehouden waarbij 16 bytes het minimum is.

4.2.3 Wat is de grootte van een bericht waarin een netwerkmodule zijn verbinding met anderen aangeeft

5 — Oplossingsrichtingen

- Beschrijf je hoe jouw oplossingsrichting de hoofdvraag beantwoord?
 - Beschrijft de oplossingsrichting de mogelijkheden?
 - Staat hier "Veld" vermeldt van de methodekaart van de HAN?
-

6 — Experimenten

6.1 Wat is minimaal benodigd in een simulatie om abstracte drone te representeren?

In [Paragraaf 4.1](#) is geconcludeerd dat als simulatie software ROS melodic met Gazebo9 wordt gebruikt. Hierin kwam naar voren dat niet elke drone package gebruikt kan worden doordat ze specifiek gemaakt zijn voor de versie van ROS. Omdat het project de focus heeft op de werking van netwerkmodules voldoet een abstracte versie van een drone. Zo hoeft deze bijvoorbeeld op dit moment nog niet realistisch te vliegen.

- Beschrijft waarom dit experiment van belang is.
 - Voldoet de code van het experiment aan de opgestelde eisen?
 - Voldoet het programma aan de opgestelde QoS?
 - Is gedocumenteerd waarom het programma voldoet aan de opgestelde eisen?
 - Is gedocumenteerd waarom het programma voldoet aan de opgestelde QoS?
-

7 — Conclusie

Conclusie uit resultaten. Herhaal en beantwoord de hoofdvraag.

- Trek een conclusie uit de resultaten van de deelvragen.
 - Wordt de hoofdvraag beantwoord?
 - Wat beveel je aan?
-

Literatuur

- Castillo, D. C., Rosales, J. & Blanco, G. A. T. (2018, 03). Optimizing binary serialization with an independent data definition format. *International Journal of Computer Applications*, 180, 15-18. Op 1 maart 2019 verkregen van https://www.researchgate.net/publication/323893287_Optimizing_Binary_Serialization_with_an_Independent_Data_Definition_Format doi: 10.5120/ijca2018916670
- Kulhavy, D. L., Hung, I.-K., Unger, D. R. & Zhang, Y. (2017, 19 april). *Feature and performance comparison of the v-rep, gazebo and argos robot simulators*. Nacogdoches, Verenigde Staten: Arthur Temple College of Forestry and Agriculture. Op 11 maart 2019 verkregen van <http://lenkaspaces.net/tutorials/programming/robotSimulatorsComparison>
- Oates, B. J. (2005, november). *Researching information systems and computing*. Sage Publications Inc.
- Pitonakova, L., Giuliani, M., Pipe, A. & Winfield, A. (2018, januari). Feature and performance comparison of the v-rep, gazebo and argos robot simulators. *Proceedings of the 19th Towards Autonomous Robotic Systems Conference (TAROS 2018)*. Op 19 februari 2019 verkregen van <http://lenkaspaces.net/tutorials/programming/robotSimulatorsComparison>
- Tauro, C., Ganesan, N., Mishra, S. & Bhagwat, A. (2012, 05). Article: Object serialization: A study of techniques of implementing binary serialization in c++, java and .net. *International Journal of Computer Applications*, 45, 25-29. Op 1 maart 2019 verkregen van https://www.researchgate.net/publication/235719668_Article_Object_Serialization_A_Study_of_Techniques_of_Implementing_Binary_Serialization_in_C_Java_and_NET
- Van Turnhout, K., Craenmehr, S., Holwerda, R., Menijn, M., Zwart, J. & Bakker, R. (2014, september). De informatieprofessional 3.0. In (p. 163-174). Den Haag, Nederland: Academic Service BV. Op 11 februari 2019 verkregen van https://www.researchgate.net/publication/272421605_De_methodenkaart_praktijkonderzoek
- Van Turnhout, K., Mulholland, C., Kamp, I., Jacobs, M., Neumann, A., Rouwhorst, S. & Van der Vlies, L. (z. j.). *Cmd methods pack: Find a combination of research methods that suit your needs*. Nederland: HAN University of Applied Sciences - Amsterdam University of Applied Sciences. Verkregen van <http://cmdmethods.nl>
- U.S. Government. (2008, september). *Global positioning system standard positioning service performance standard* (4e dr.). Arlington, Verenigde Staten: The Office of the Secretary of Defense. Op 11 maart 2019 verkregen van <https://www.gps.gov/technical/ps/2008-SPS-performance-standard.pdf>