

Karl Andreas Wik Opheim
Benjamin Alexander Whittaker

Sanking av sau ved bruk av autonome droner

Masteroppgave i Datateknologi
Veileder: Svein-Olaf Hvasshovd
Juli 2023



NTNU

Kunnskap for en bedre verden

Karl Andreas Wik Opheim
Benjamin Alexander Whittaker

Sanking av sau ved bruk av autonome droner

Masteroppgave i Datateknologi
Veileder: Svein-Olaf Hvasshovd
Juli 2023

Norges teknisk-naturvitenskapelige universitet
Fakultet for informasjonsteknologi og elektroteknikk
Institutt for datateknologi og informatikk



Kunnskap for en bedre verden

Sammendrag

På høsten må mange bønder sanke sauene sine inn fra utmark etter en sesong på beite. Dette kan være et stort arbeid som krever mye ressurser og tar lang tid. De siste årene har droner blitt mer og mer brukt i landbruket. Denne masteroppgaven vil se på muligheten for å lage et system der autonome droner sanke sau. Utgangspunktet for oppgaven er et forslag der tre droner utstyrt med høyttaler driver sauer ved hjelp av en fjerde drone, utstyrt med kamera, som lokaliserer sauene og sender kommandoer til de andre dronene.

Det er utviklet algoritmer og modeller som er implementert i en simulator. Simulatoren simulerer saue- og droneatferd og kjører regelbaserte algoritmer som gjør det mulig for dronene å drive sauene. Det er blitt lagd tester for å evaluere metoden som er utviklet.

Selv om simulatoren bestod alle testene som var satt opp, er det vanskelig å evaluere om hvordan metoden hadde fungert i virkeligheten. Dette skyldes blant annet av usikkerheten rundt sauenes atferd, mangelen på data til å evaluere parameterne og algoritmens manglende tilpasningsevne. I tillegg er det feil i metoder og implementasjon som ytterligere øker denne usikkerheten. Siden metoden er regelbasert, er det vanskelig å lage algoritmer og modeller som passer alle situasjoner. Kunstig intelligens kan muligens løse dette problemet.

Som en konklusjon, er den utviklete metoden vanskelig å evaluere, men resultatet antyder at det finnes en mulighet for å bruke droner til å sanke sau. Det krever mer arbeid, og flere løsninger bør vurderes. Kunstig intelligens har allerede vist flere positive egenskaper, og dette bør undersøkes nærmere. I tillegg kan det være av interesse å se på kombinasjon av regelbaserte algoritmer i sammen med kunstig intelligens. Det oppfordres også til å samle store mengder data som kan brukes til å evaluere fremtidig forskning.

Abstract

In autumn, after a season of grazing, many farmers must gather their sheep from the outfield. This might be a demanding task, which requires many resources and is a time-consuming job. In recent years, drones have been more commonly used in agriculture. This master thesis will investigate the possibilities of developing a system, where autonomous drones are able to gather sheep. A basis used for the thesis is a proposition where three drones armed with speakers herd the sheep with aid of a fourth drone armed with a camera, which is able to locate the sheep and send commands to the other drones.

There have been developed models and algorithms which is implemented in a simulator. The simulator simulates sheep and drone behaviour and runs rule-based algorithms which makes it possible for the drones to herd the sheep. There has been designed tests to evaluate the developed method.

The simulator passed all the designed tests, however, evaluating how well the method would perform in real life is difficult. This is partially caused by the uncertainty of the sheep behaviour, the absence of collected data to evaluate the parameters and the algorithms missing ability to adapt. Additionally, several mistakes and errors in the methods and implementation further increases the uncertainty. The method is rule-based, which makes it difficult to develop algorithms and models which fits all situations. Artificial intelligence may solve this problem.

The thesis concludes that the developed method is difficult to evaluate, however, the results indicates that there exists a possibility to use drones to gather sheep. More work is required, and different solutions should be considered. Artificial intelligence has already shown several positive properties and should be investigated further. In addition, exploring the combination of rule-based algorithms and artificial intelligence may be of interest. It's also encouraged to collect large amounts of data which can be used to evaluate future work.

Forord

Vi vil takke veileder Svein-Olaf Hvasshovd for å dele sin kunnskap og erfaring om sauesanking, og alle dere som har lest oppgaven og gitt tilbakemeldinger. En stor takk til Ellen som har reist med kaffe til oss flere ganger mens vi oppholdt oss på biblioteket. En ekstra stor takk til mammaen til Karl Andreas for å ha gitt uvurderlige tilbakemeldinger.

Innhold

Sammendrag	i
Abstract.....	ii
Forord.....	iii
Figurer.....	vii
Tabeller	ix
1 Introduksjon.....	1
1.1 Sauesanking i Norge.....	1
1.2 Effektivisere sauehold	2
1.3 Problemstilling	2
1.4 Utgangspunkt for oppgaven	2
1.5 Ønsket fremtidig resultat.....	3
2 Bakgrunn og tidligere arbeid	4
2.1 Droner i landbruket	4
2.2 Droner til driving av dyr.....	6
2.3 Manuell driving av sau med droner.....	6
2.4 Forskning på autonom driving av sau	7
2.5 Lokalisering av sauer ved hjelp av droner	9
2.6 Simulering av saueflokker.....	10
2.7 Dyrevelferd.....	12
2.8 Lover og regler rundt selvstyrte droner.....	13
3 Metode.....	14
3.1 Valg av metode.....	14
3.2 Avgrensninger	14
3.2.1 Videreutviklet utgangspunkt for oppgaven.....	15
3.2.2 Flatt terreng uten hindringer	17
3.2.3 Perspektiv.....	17

3.2.4	Droneteknologi	18
3.2.5	Kollisjon mellom droner	18
3.2.6	Sau og innhegning.....	18
3.2.7	Automatisk generering av ruter	19
3.3	Utviklingsplattform	19
3.4	Definisjoner på variabler	19
3.5	Sauemodell	20
3.5.1	«Boids»	20
3.5.2	Delgado-Matas utvidelser	21
3.5.3	Ytterlige utvidelser.....	22
3.5.4	Total utregning av sauemodellen	24
3.5.5	Sauemodell radier og synsfeltverdi.....	25
3.6	Dronemodell.....	25
3.6.1	Dronehierarki	26
3.6.2	Motta oppgaver	26
3.6.3	Utføre oppgaver	26
3.7	Sankealgoritme.....	28
3.7.1	Generering av ruter	28
3.7.2	Lokalisere.....	30
3.7.3	Forberede	31
3.7.4	Utføre	32
3.8	Parametere	38
3.9	Testing.....	39
3.9.1	Forberedelse	39
3.9.2	Testscenario 1 – Rett rute	40
3.9.3	Testscenario 2 – Ujevn rute	42
3.9.4	Testscenario 3 – Følg og forlat sti.....	44

3.9.5	Testscenario 4 – Følg riktig sti.....	46
3.9.6	Feilsøking.....	48
4	Resultat og Diskusjon.....	49
4.1	Resultat av tester	49
4.2	Evaluering	49
4.2.1	Evaluering av sauemodellen	49
4.2.2	Evaluering av sankealgoritmen.....	50
4.2.3	Evaluering av parametervalg	50
4.3	Feil og svakheter	51
4.3.1	Sauer som ikke reagerer på dronene	51
4.3.2	Bevegelse til sauer i fravær av droner.....	51
4.3.3	Kalkulering av redselen til sauene	52
4.3.4	Ikke dynamisk sankealgoritme	52
4.3.5	Uregelmessigheter i simulatorkjøring.....	53
4.3.6	Feil i generering av myke svinger.....	54
4.3.7	Feil med diverse forhåndsdefinerte ruter	54
4.3.8	Svakheter i innhegningsimplementasjonen.....	55
4.3.9	Feil i kalkulering av redselsspredning	55
4.4	Regelbaserte Algoritmer vs. Kunstig intelligens.....	56
5	Konklusjon og videre arbeid.....	57
5.1	Konklusjon	57
5.2	Videre arbeid	57
6	Referanser	59

Figurer

Figur 1: Skisse av hva et fullstendig system skal kunne håndtere. Svart stiplet linje er ruten som dronene skal drive sauene på. Mørkere grønnfarge er høydeforskjell. Det blå er vann. Gul stripe er en sti. Brunt er gjerder og innhegning. Grønne flekker er trær/busker.	3
Figur 2: Kameradronen har lokalisert en gruppe med sauer på jordet.	15
Figur 3: Dronene flytter seg i startposisjon uten å fly for nær sauene.	16
Figur 4: Dronene flyr nærmere sauene og driver dem på ruten (stiplet linje).	17
Figur 5: Illustrasjon av hvordan synsfeltet til sauene endrer seg i nærvær av en drone.	23
Figur 6: Sauene som er nær stien blir tiltrukket av den.	24
Figur 7: Illustrasjon av hvordan det sjekkes at dronen skal gå til neste punkt. Rød stiplet linje viser hvordan dronen beveger seg mot neste punkt.	27
Figur 8: Illustrasjon av hvordan dreiepunktet (rød prikk) blir funnet.	29
Figur 9: Illustrasjon av hvordan segmenter av sirkelen blir nye linjestykker til dronen.	29
Figur 10: Illustrasjon av flokkalgoritme. I første iterasjon sjekkes det hvilke sauer som er innenfor radien til hver av de røde sauene. Sauene i hvitt blir lagt til i de respektive flokkene (Rød, Blå, Gul og Oransje sirkler). I neste iterasjon sjekker algoritmen om midtpunktene til flokkene er innenfor radien. Den grønne sirkelen viser to flokker som har blitt slått sammen til en flokk.	30
Figur 11: Illustrasjon av startposisjonene til dronene. Bakdronen blir plassert motsatt i forhold til nærmeste punkt på den forhåndsdefinerte ruten.	31
Figur 12: Illustrasjon av hvordan dronen unngår påvirkningsområdet ved å danne segmenter på kanten av området.	32
Figur 13: Illustrasjon av dronene som beveger seg nærmere flokken.	33
Figur 14: Illustrasjon av generering av fullstendig rute. Først en linje fra sauene til nærmeste punkt (blå), så den forhåndsdefinerte ruten (rød), og til slutt et linjestykke til innhegningen (gul).	34
Figur 15: Illustrasjon av hvordan rutene til styredronene er forskjøvet.	34
Figur 16: Illustrasjon av hva som skjer med en forskjøvet rute i en krapp sving. Svart rute er den forhåndsdefinerte ruten. Rød rute er bakdronens rute. Gul og blå er styredronenes rute. Blå rute får ikke myk sving.	35

Figur 17: Illustrasjon som viser to situasjoner for hvilket punkt på ruten bakdronen skal bevege seg mot. Dette punktet ligger en lengde bak den bakerste sauene.	36
Figur 18: Illustrasjon som viser to situasjoner for hvilket punkt på ruten en styredrone skal bevege seg mot. Punktet blir bestemt fra posisjonen til den valgte sauene.	37
Figur 19: Oppsett av et testscenario i «Godot».	40
Figur 20: Illustrasjon av testscenario 1. Dronene må drive sauene på en rett linje.	41
Figur 21: Skjerm bilde av simulator som kjører variant 1 av testscenario 1	41
Figur 22: Skjerm bilde av simulator som kjører variant 2 av testscenario 1	42
Figur 23: Skjerm bilde av simulator som kjører variant 3 av testscenario 1	42
Figur 24: Illustrasjon av testscenario 2. Dronene må drive sauene på en ujevn rute.	43
Figur 25: Skjerm bilde av simulator som kjører variant 1 av testscenario 2	43
Figur 26: Skjerm bilde av simulator som kjører variant 2 av testscenario 2	44
Figur 27: Skjerm bilde av simulator som kjører variant 3 av testscenario 2	44
Figur 28: Illustrasjon av testscenario 3. Dronene må drive sauene på og av stien.	45
Figur 29: Skjerm bilde av simulator som kjører variant 1 av testscenario 3	45
Figur 30: Skjerm bilde av simulator som kjører variant 2 av testscenario 3	46
Figur 31: Skjerm bilde av simulator som kjører variant 3 av testscenario 3	46
Figur 32: Illustrasjon av testscenario 4. Dronene må drive sauene på riktig sti.	47
Figur 33: Skjerm bilde av simulator som kjører variant 1 av testscenario 4	47
Figur 34: Skjerm bilde av simulator som kjører variant 2 av testscenario 4	48
Figur 35: Skjerm bilde av simulator som kjører variant 3 av testscenario 4	48
Figur 36: Kjøring av tilfelle der en flokk blir drevet mot en annen.	53
Figur 37: Utklipp av testscenario 4 variant 2. Utklippet til venstre har fått en uregelmessighet som forårsaker en linje som gjør at dronen beveger seg lenger inn enn den skal.	53
Figur 38: Et kort linjestykke på ruten, fører til krysninger som forårsaker uønskede ruter.	54
Figur 39: Forhåndsdefinert rute med linjestykker som ligger nært hverandre.	55

Tabeller

Tabell 1: Parameterverdiene i den ferdige simulatoren.	38
---	----

1 Introduksjon

1.1 Sauesanking i Norge

I flere tusen år har sauer som beiter i Norge, hjulpet å rydde land, ved at sauen spiser ned busker og småtrær [1]. I dag slipper fortsatt norske bønder ut sauer på beite for å vedlikeholde utmark, som igjen bidrar til et større biologisk mangfold [2]. Hvert år mot høsten må mange bønder sanke inn sauene sine etter en sesong på beite i utmark. I 2022 ble det sluppet over 1.8 millioner sau og lam på utmarksbeite [3]. Sauene befinner seg ofte på store områder der det kan være ulent terreng som gjør det vanskelig å lete etter sauene for å sanke dem. Dette kan gjøre arbeidet både tids- og ressurskrevende.

Veileder for oppgaven, Svein-Olaf Hvasshovd, har flere års erfaring med sau og sauesanking og er vår muntlige kilde til hvordan sanking av sau foregår. Han har gjennom to semestre delt sin erfaring og kunnskap, og beskriver sauesanking som beskrevet i dette avsnittet. Bonden er ofte avhengig av ekstern hjelp til å sanke sauene. Dette kan være familie, venner eller andre frivillige. Veileder forklarer også at sauesanking ofte foregår i flere faser der man forsøker å lokalisere og føre sauene hjem. De fleste blir da hentet, men dessverre er det ikke alltid alle blir funnet. Da er bonden avhengig av turgåere som varsler hvis de kommer over en sau [4].

Av ulike årsaker blir likevel flere sauer aldri hentet hjem [3]. En sau kan ha en verdi på over 3000 kr, så tap av sau kan ha økonomiske konsekvenser for bonden [5]. I noen tilfeller kan det bli nødvendig å bruke spesialtrente gjeterhunder for å sanke sauene, en slik gjeterhund kan koste opptil flere tusen kroner [6]. Disse gjeterhundene har også et begrenset antall år de kan sanke sauer før de må erstattes av en ny gjeterhund.

1.2 Effektivisere sauehold

Digitalisering av samfunnet har gjort at det har dukket opp flere løsninger på markedet for å effektivisere sauehold. En av disse løsningene er elektroniske bjeller som gjør det mulig å spore saueflokkene. Eksempler på dette er «*Telespor*» og «*Smartbjella*» [7] [8]. En annen løsning er systemet «*Nofence*», som er verdens første virtuelle gjerdesystem til å kontrollere dyr [9].

De siste årene har det også blitt gjennomført en rekke masteroppgaver som har undersøkt mulighetene for å bruke droner til å forenkle og effektivisere sanking av sau, og disse vil bli utdypet ytterligere i kapittel 2.

1.3 Problemstilling

Løsningene beskrevet i forrige delkapittel, og de tidligere masteroppgavene som beskrives senere, har i stor grad hatt fokus på lokalisering av sau og ikke den fysiske flyttingen av sau. Problemstillingen til denne masteroppgaven er derfor å se på muligheten for å bruke droner til å sanke sau ved å simulere dronene og sauene.

1.4 Utgangspunkt for oppgaven

Et forslag til en løsning gitt av veileder er brukt som grunnlag for oppgaven. Forslaget var å bruke fire droner til å drive sauer [4].

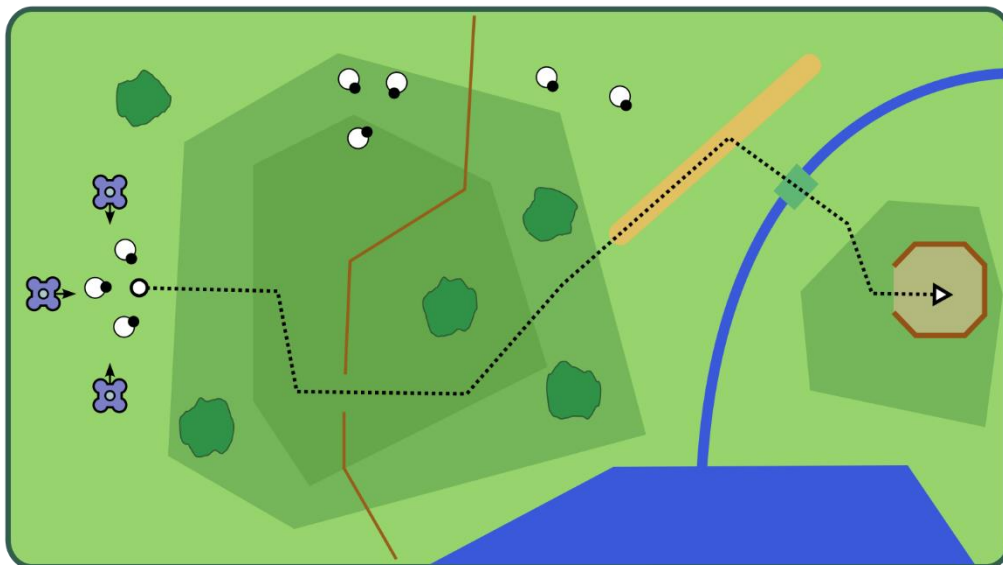
Den ene dronen skal være utstyrt med et kamera og skal fly høyt over flokken og lokalisere sauene. Det er også tenkt at dette er dronen som sender oppgaver til de andre dronene om hvordan de skal bevege seg for å drive sauene [4].

De tre siste dronene skal fly lavt over bakken, bevege seg nær sauene og lage lyd slik at sauene flytter seg. Det er tenkt at en drone er bak og skal drive sauene framover, mens de to siste skal være på hver side av flokken for å sørge at kursen er riktig og at flokken ikke splitter seg [4].

1.5 Ønsket fremtidig resultat

Håpet med oppgaven er at det blir et utgangspunkt for et fullstendig system for sanking av sau. Et slikt system er komplekst og har flere punkt å ta hensyn til for at det skal være et fungerende og ikke minst trygt system å bruke. Figur 1 viser en skisse av hva et slikt system må ta hensyn til, og hvilke problemer som kan oppstå. Sauene kan få panikk og løpe, sette seg fast i gjerder, eller falle ut i vann eller bekker. Dronene må unngå å krasje i trær og ikke fly for lavt når terrenget endrer seg.

Det kan tenkes at lokale bønder slår seg sammen om å investere i et slikt system og at de bytter på å benytte seg av det. Dronene settes da ned på bakken og systemet starter. Det skal ikke være nødvendig for bonden å kontrollere droner eller gjøre noe annet enn å starte systemet. Dronene styrer seg selv til sauene er samlet inn, for å så fly tilbake til bonden.



Figur 1: Skisse av hva et fullstendig system skal kunne håndtere. Svart stiplet linje er ruten som dronene skal drive sauene på. Mørkere grønnfarge er høydeforskjell. Det blå er vann. Gul stripe er en sti. Brunt er gjerder og innhegning. Grønne flekker er trær/busker.

2 Bakgrunn og tidligere arbeid

Oppgavens omfang er stort, og flere fagfelt må utforskes for at en ferdig løsning skal være mulig å realisere. De neste delkapitlene inneholder nyere forskning og informasjon om ulike tema som berører oppgaven. Ikke alle temaene vil være direkte aktuelle i oppgaven, men inkluderes likevel ettersom motivasjonen for oppgaven belyses gjennom diverse tema. Andre tema kan være aktuelle for videre arbeid og nevnes derfor for å danne et bilde av hvordan hele systemet kan fungere i et ferdig produkt. Delkapitler med tema som er direkte aktuelle for oppgaven har en kommentar om dette i sitt siste avsnitt.

2.1 Droner i landbruket

Samtidig som droneteknologien utvikler seg, har landbruket fått flere og flere bruksområder til droner. Under er det beskrevet noen situasjoner fra virkeligheten der droner har blitt benyttet som et verktøy i landbruket.

Innenfor jordbruk er droner allerede i bruk i flere land. En artikkel publisert på «*FNs utviklingsprogram*» sine nettsteder, forteller om et selskap som har fått støtte til å hjelpe bønder i Moldova ved å bruke droner til sprøyting av ugressmiddel på avlingene. Videre står det at dette foreløpig har vist positive resultater, der dronen bruker 20 ganger mindre vann, og opptil 40% mindre ugressmiddel sammenlignet med en traktor. Ifølge artikkelen kan en traktor kjøre ned og ødelegge opptil 10% av avlingen, mens dronene flyr over avlingen [10].

I en artikkel fra det australske kringkastingsselskapet «*ABC News*» blir det fortalt om testing av droner som måler hvor mye vann avokadotrær mister ved hjelp av termiske bilder som viser temperaturen til trærne. Ifølge artikkelen er sluttmålet å hjelpe bøndene med å regulere bruken av vann ved hjelp av informasjonen fra dronene. Dette kan føre til at bonden kan høste inn flere, større frukt av høyere kvalitet [11].

Også i Norge blir droner mer og mer relevant for landbruket. En artikkel fra «NRK» beskriver et prøveprosjekt gjort i Steinkjer i samarbeid mellom «Mære landbruksskole» og gründere i bedriften «Biodrone». Ifølge artikkelen blir åkrene vanligvis sådd med gressfrø ved hjelp av traktor og såmaskin, mens prosjektet skulle teste om droner kunne gjøre dette. Det ble brukt en stor drone som sådde frøene over en åker, og etter tre uker hadde frøene slått rot og spirt. På et annet åkerstykke ble det sådd frø på tradisjonelt vis, og disse ble sammenlignet med frøene som var sådd av dronen. Samarbeidspartnerne var enige om at de tradisjonelt sådde frøene spirte litt bedre enn de som var sådd av dronen. På tross av dette ble prosjektet regnet som en suksess, da prosjektet gikk ut på å finne ut om frøene i det hele tatt ville slå rot [12].

En artikkel fra «UAS Norway» belyser problematikken rundt slåmaskin og rådyrkalver der nestor i viltforvaltning Vidar Holthe estimerer at rundt 10 000 rådyrkalver blir drept hvert år som resultat av å bli kjørt over av slåmaskiner. I artikkelen forteller Holthe at mødrene legger kalvene sine i gresset som skjulested og besøker dem for å die. Dette kan gjøre det vanskelig å oppdage dem [13]. Flere medier og nettsteder i Norge melder om at det benyttes droner med termisk kamera til å lokalisere rådyrkalvene i gresset, for så å flytte dem vekk før slåtten [14] [15] [16]. I følge «UAS Norway» sine nettsteder, ble det reddet omtrent 500 rådyrkalver ved hjelp av droner i 2022 [17] [18]. Flere myndigheter i Norge deler ut flere hundre tusen kroner i støtte til kjøp av droner med termisk kamera [18].

Generelt har droner nyttige egenskaper som bonden kan dra seg nytte av. Droner er ofte utstyrt med kamera som gjør at bonden kan få et større og bedre overblikk over f.eks. en saueflokk. En artikkel fra «NRK» belyser dette, der det blir skrevet at flere norske bønder har begynt å bruke droner til å overvåke saueflokken. En sauebonde som ble intervjuet i artikkelen fortalte at hun bruker dronen til å se etter rovdyr eller kadaver [19]. I tillegg kan droner bevege seg over ulent terreng som bonden ellers kan ha vanskeligheter å befare, hvis bonden ønsker å se etter dyrene sine.

2.2 Droner til driving av dyr

Et av bruksområdene til droner kan være driving av dyr. Ut ifra terreng, mengden dyr, type dyr og mannskap kan det å flytte dyr bli en stor oppgave å utføre. Det har blitt dokumentert flere tilfeller av forsøk på å benytte droner til å forenkle denne oppgaven, der noen av disse blir beskrevet under.

En case-studie fra 2016 beskriver et forsøk på å bruke droner til å drive elefanter. Sammendraget til studien forteller at beskyttede områder, der flere afrikanske savanneelefanter hører til, blir omringet og isolert av landbruk og beboelse. Case-studien beskriver hvordan dette kan føre til konflikter hvis elefanter beveger seg nær for å spise avlinger. Elefanter og menneskers liv settes i fare når ulike taktikker blir benyttet for å drive elefanter vekk. I case-studien ble det utført et eksperiment i Tanzania der droner ble testet til å drive vekk elefanter, og i alle tilfeller som ble testet, reagerte elefanten på dronen og ble drevet vekk uten at man trengte å benytte seg av andre taktikker [20].

En artikkel fra «*Reuters*» i 2020 forteller om en ranch i Israel der bøndene benytter seg av droner for å drive kveg. Artikkelen forteller at dronene blir fjernstyrt og svever nær kyrne slik at de flytter på seg [21].

En rapport i «*NIBIO Rapport*» undersøker nytteverdien av droner i forebyggende- og konfliktdempende tiltak i kategorien åpen kategori [22]. Åpen kategori er en regelkategori for dronebruk uten å kreve særskilt tillatelse. Dette blir ytterligere beskrevet i kapittel 2.8. I rapporten har blant annet flere reineiere i flere reinbeitedistrikt blitt intervjuet om deres bruk av droner. Alle de undersøkte distriktene utenom ett, bruker droner til driften, og disse bruker droner hovedsakelig til å drive og samle dyrene sine [22].

2.3 Manuell driving av sau med droner

Det har også blitt forsøkt å drive sauer ved hjelp av droner. Sauer er som oftest i flokker, og kan bli en stor jobb å håndtere. Det finnes flere tilfeller i verden der enkelte bønder har begynt å bruke droner til å drive sau for å gjøre denne jobben mer effektivt. Noen av disse er beskrevet nedenfor.

Bruk av droner til å drive sau er blant annet dokumentert i Irland og på New Zealand [23] [24]. Dronene som er brukt i disse tilfellene, er manuelt kontrollert av bonden. I artikkelen fra New Zealand forteller en bonde at lyden dronene lager, virker å gjøre sauene redde, noe som gjør at de løper vekk [24].

I en artikkel skrevet på «*Harper Adams University*» sine nettsteder står det skrevet om en bonde har trent opp sauene sine til å assosiere en drone med mat, og etter en tid har han nå klart å få sauene til å følge etter dronen. En professor ved universitet uttaler at han mener dette er en mer etisk måte å flytte sauene på ettersom man ikke bruker frykt [25].

I Norge finnes det få dokumenterte tilfeller på bruk av droner til å drive sau. I en artikkel fra magasinet «*Sau og geit*» kommenterer en bonde at han av og til bruker dronen til å drive enkelte sauer i en retning der man kan overta med manuell driving [26]. I rapporten i fra «*NIBIO Rapport*» er det undersøkt dronebruk til sau ved å intervjuer beitelag. Her er de vanligste bruksområdene omtalt som tilsyn, kadaversøk og sanking. Når det gjelder sanking i form av å drive sauene, melder rapporten at beitelagene oppgir dette som vanskelig da sau i mindre grad enn for eksempel rein lar seg påvirke av drone, noe som motstrider resten av funnene i delkapittelet [22].

2.4 Forskning på autonom driving av sau

Det har blitt skrevet flere artikler som utforsker muligheten til å la autonome kjøretøy drive sauer. Ved hjelp av å bruke regelbaserte algoritmer og/eller kunstig intelligens, kan kjøretøyene potensielt være autonome uten å bli kontrollert av en pilot. Det kan ligge flere motivasjoner bak å gjøre manuell driving av sau automatisk. Slik praksisen er i dag, må bonden selv lære seg å styre dronene. Å skaffe denne ferdigheten kan være krevende. Hvis dronene er autonome, trenger bonden kun å starte systemet, så vil dronene gjøre jobben alene. En annen motivasjon kan være å utforske flere metoder å drive dyr på, slik at forskning kan bidra til å effektivisere landbruket. De neste avsnittene vil se på tre artikler fra de siste årene som har sett på muligheten for driving av sau med autonome kjøretøy, heretter kalt gjeteragenter.

En artikkel fra 2022 presenterer et konsept på et system som kan drive en flokk med et stort antall sau ved hjelp av et nettverk av droner. I likhet med utgangspunktet for vår oppgave består dette systemet av en type drone som er utstyrt med kamera som lokaliserer og følger sauene, og en annen type drone som er utstyrt med høyttaler som spiller av bjeffelyder for å flytte på sauene. Artikkelen presenterer metoden som består av to faser, en samlefase og en drivefase. I samlefasen blir saueflokken omringet av flere droner som beveger seg nærmere, slik at sauene samles tettere sammen. Når alle sauene er innenfor en viss radius fra midtpunktet til flokken, starter dronene å drive sauene mot et mål [27].

En annen artikkel fra 2022 legger mer fokus på å bruke kunstig intelligens i stedet for regelbaserte algoritmer til å drive sauene [28]. Artikkelen begrunner dette ved å bl.a. vise til en tidligere artikkel «*An Investigation of Predator Response in Robotic Herding of Sheep*» av M. Evered, P. Burling og M. Trotter som ifølge [28] har vist at sauers respons på robotgjeterhund endrer seg over tid etter hvert som sauene blir vant til roboten [28]. Dette kan resultere i at en algoritme som fungerer i starten, ikke fungerer etter en tid, ettersom kjøretøyene kanskje må bevege seg nærmere sauene før de reagerer på dem. Artikkelen sin metode går ut på å bruke «*Reinforcement Learning*» (forsterkende læring) til å trene opp virtuelle gjeteragenter som har evnen til å tilpasse seg innenfor et simulert miljø. Resultatet til artikkelen forteller at gjeteragentene klarte å flytte sauer fra et sted til et annet med en suksessrate på neste 96%. Metoden i artikkelen har også vist seg å være tilpasningsdyktig da den har egenskapen å vurdere om den skal drive mindre grupper av sau eller om den skal drive hele flokken mot målet [28].

En artikkel fra 2021 bruker også «*Reinforcement Learning*». Artikkelen begrunner bruken av kunstig intelligens istedenfor regelbaserte algoritmer ettersom det kan være vanskelig å definere regler som håndterer alle mulige situasjoner i regelbaserte algoritmer. Metoden som er utviklet, trener opp virtuelle gjeteragenter til å gjete sauene til et mål i et simulert miljø med flere hindringer. Resultatet viste at gjeteragentene klarte å føre sauene mellom hindringene, men at de ikke klarte å håndtere mer enn 2-4 sauer [29].

Forskerne bak artikkelen fulgte opp med en ny artikkel i 2022 der metoden var videreutviklet og klarte å drive flere sauer til mål [30].

De neste avsnittene vil oppsummere disse artiklene ved å se på styrker og svakheter med metodene som er presentert.

Stryken med den første metoden er at den kan håndtere en stor flokk med sau. Ringen med droner rundt saueflokken hindrer kollisjon mellom dronene [27]. En svakhet med metoden er at det kreves flere droner. Droner kan være dyre, som gjør systemet til en potensiell dyr investering. Hvis saueflokken er liten, kan systemet virke overkomplisert og dyrt for bruken. Metoden tar heller ikke hensyn til terreng, der det å omringe sauene kan være vanskelig å utføre.

En styrke med den andre metoden er at metoden lar gjeteragentene tilpasse seg miljøet i forhold til at sauenes respons kan endre seg over tid. En annen styrke er at det kun er brukt én gjeteragent som kan gjete mange sauer, noe som kan redusere utstyrs kostnader. Metoden viser også egenskaper til å tilpasse seg ulike situasjoner. En svakhet er at man ikke har full kontroll på hvordan roboten beveger seg, noe som kan skape uønskede situasjoner.

Styrker med den tredje metoden er at metoden tar hensyn til hindringer og at det kun er brukt én robot slik som i forrige metode. En svakhet er at den kun fungerer bra på små grupper med 2-4 sau, men dette har blitt forbedret i den nye artikkelen som hevder at den nå klarer opp til 10 sauer [30].

2.5 Lokalisering av sauer ved hjelp av droner

Som nevnt tidligere, er mange droner utstyrt med kamera. I ulike terreng og værforhold kan det være vanskelig å lokalisere sauene på bildene fra dronen, og det kan være store områder som må gjennomføres for å finne alle sauene. Dette har motivert fram metoder som kan gjøre denne jobben mer effektiv. Flere masteroppgaver på «NTNU» har brukt kunstig intelligens som trener på bilder tatt av droner til å gjenkjenne sau i bildene. Noen av disse oppgavene blir beskrevet i de neste avsnittene.

En masteroppgave fra 2019 tok bilder av sauer ved hjelp av drone, og trente en modell som kunne lokalisere sauer. Modellen klarte å gjenkjenne og markere sauene på bildene 99% av gangene. Likevel ble det satt spørsmål rundt

anvendeligheten til modellen ettersom bildene i datasettet hadde lite variasjon [31].

En masteroppgave fra 2020 fortsatte på arbeidet og forsøkte å kombinere fargebilder tatt med vanlig kamera og termiske bilder tatt med infrarødt kamera. Oppgaven kom fram til at den beste måten å gjenkjenne sauene på var å kombinere disse [32]. En masteroppgave fra 2022 konkluderer derimot med at bruk av termiske bilder hadde negativ innflytelse på resultatene.

Masteroppgaven kom fram til at forskjeller i temperatur og vær hadde for stor innflytelse på de termiske bildene, noe som skapte for mye variasjon [33].

Felles for oppgavene var formålet å lokalisere sauer ut fra bilder tatt med droner. Skal bildene brukes til driving av sau, må en modell raskt og nøyaktig gjenkjenne hvor sauene er lokalisert slik at dronene kan få riktig beskjeder.

2.6 Simulering av saueflokker

For å utvikle metoder til å drive sau, er det ønskelig å lage et digitalt simuleringsmiljø før det prøves ut i virkeligheten. Alle metodene for driving av sau som er nevnt i delkapittel 2.4 bruker simuleringsmiljø. Det gjør det lettere å endre på testverdier som antall sauer, antall droner osv. Å teste driving av sau i virkeligheten, kan bli en stor prosess. Droner må unngå å kolliderer med hverandre og med sauene. I tillegg kan det muligens være stressende for sauene å bli drevet flere ganger hvis man ønsker å kjøre flere prøvekjøringer.

For å lage et digitalt simuleringsmiljø, trenger man en måte å simulere en saueflokk på. En anerkjent måte å gjøre dette på, er å bruke Craig W. Reynolds sin modell som beskriver konseptet «*Boids*», noe som de fleste metodene beskrevet i 2.4 bruker. Hans modell fra 1987 gikk ut på ha en flokk med dyr, f.eks. en flokk med fugler eller fiskestim, og at hvert dyr følger tre konsepter. Dyret skal unngå å kolliderer med dyr i nærheten. Dyret skal forsøke å tilpasse farten og retningen til dyr i nærheten. Dyret skal også forsøke å holde seg nær dyr i nærheten. Til sammen danner dette det generelle konseptet «*Boids*» [34].

«Boids» kan altså fungere som et utgangspunkt til simulering av en saueflokk, men i den originale modellen reagerer kun dyrene på hverandre. I tilfelle der en drone skal påvirke sauene, må modellen utvides. Dette kan man gjøre ved å legge til nye konsepter. Carlos Delgado-Mata har skrevet diverse artikler som utvider Reynolds originale modell, der vi velger å fokusere på artikkelen «*On the Use of Virtual Animals with Artificial Fear in Virtual Environments*» fra 2007. I artikkelen blir konseptet «Escape» (Flukt) lagt til «Boids»-konseptene, som da handler om at hvert dyr prøver å flykte fra fare. I tillegg legger Delgado-Mata til et konsept om dyrenes følelser, som i dette tilfellet handler om redsel. Dyrenes redselstilstand påvirker «Escape»- og «Boids»-konseptene slik at de kan vektlegges ut ifra hvor redd dyret er. I artikkelen øker redselen vektleggingen av «Escape»-konseptet og «Boids»-konseptet om å holde seg nært dyrene rundt seg. Dette betyr at om en fare nærmer seg, vil dyrene holde seg nærmere hverandre og bevege seg bort fra faren. I tillegg til dette legger Delgado-Mata til konseptet om at redselen kan spre seg fra et dyr til et annet. Dette baserer artikkelen på virkeligheten der mange dyr skiller ut feromoner for å bl.a. varsle andre dyr om fare [35].

I en artikkel skrevet i 2012, er det gjort et eksperiment der posisjonene til 46 sauer og én gjeterhund ble logget ved hjelp av GPS-utstyr. Deretter ble flokkens midtpunkt regnet ut, og for hvert sekund ble gjeterhunden og hver sau sin avstand til midtpunktet regnet ut. Analyser som ble gjort av disse dataene, viste at sauene viste en stor tiltrekning mot midten av flokken [36]. En bacheloroppgave fra 2019 har brukt denne dataen for å evaluere en sauemodell basert på Delgado-Mata sin artikkel. Ved å loggføre posisjonene til sauene og gjeterhunden i en simulator, ble dataene sammenlignet med dataene fra virkeligheten. Det ble lagd to modeller, en med konseptet om spredning av redsel og en uten. Oppgaven konkluderte med at modellen som hadde spredning av redsel, var litt mer realistisk enn modellen uten [37].

Artiklene beskrevet ovenfor er alle aktuelle for hvordan sauene blir simulert i oppgaven vår, og gir et godt utgangspunkt for implementasjonen.

2.7 Dyrevelferd

Å bruke droner til å drive sauer, bringer opp spørsmålet om hvor mye stress som kan påføres sauene. Det har blitt gjort litt forskning på stress hos sauer, og dette delkapittelet vil gi et lite overblikk på denne forskningen.

I en artikkel fra 2021 ble det gjennomført tester for å måle pulsen til sauer i droners nærvær for å se hvor stresset sauene ble [38]. Motivasjonen bak testene var blant annet at gjeterhunder kan skade sau under driving ved å bite dem, noe som kan forhindres ved å bruke droner til driving. I metoden i artikkelen har droner blitt utstyrt med høyttaler og sauer fått pulsmåler. Deretter ble disse dataene sammenlignet med pulsmålinger fra sau fra en annen artikkel skrevet i 1990 [39], der blant annet pulsdringer til sau ble målt under driving med hund og menneske. [38] konkluderer med at droner kombinert med lyd kan gjøre driving av sau mer trygt, uten å stresse sauene. Samtidig gir artikkelen statistikk på avstander, fart og lydavspilling til dronene og analyser som kan hjelpe å minimere stress hos sau [38].

En artikkel i medlemsbladet til interesseorganisasjonen «*Norsk Sau og Geit*» presenterer resultat til en pilotundersøkelse gjennomført av «*Animalia*» høsten 2008, som undersøkte kjøttkvalitet koblet mot stress hos lam. Her presenteres sankingen som en stor påkjenning for lammene, der det ble tatt blodprøver etter skilling og under slakting, som kan vise tegn på bl.a. fysisk påkjenning. Det ble også målt slutt-pH på kjøttet etter slakting. Høy slutt-pH indikerer dårlig kvalitet på kjøttet. Ifølge artikkelen betyr en slutt-pH på over 6 at det ikke er mer energi igjen i muskelen, og at en verdi på 5,8 er en vanlig grense på kjøtt som må utsorteres. Det ble gjort måling på tre buskaper. Artikkelen forteller at blodprøvene viste at etter hard fysisk påkjenning fra sanking og skilling, så ble energilageret til lammene tømt. I to av buskapene fikk ikke lammene mulighet til å fylle dette før de ble slaktet. Slutt-pH-målingen til en stor andel av lammene i disse buskapene, 53% på den ene og 32% på den andre, var over 5,8. Ingen av dyrene i den tredje buskapen hadde høy slutt-pH. I artikkelen står det at det ikke kan konkluderes noe, men at det indikerer et kvalitetsproblem i produksjon av lam fra utmarksbeite [40]. Artikkelen er flere år gammel, men den indikerer at sanking kan påvirke kjøttkvaliteten hvis lam

ikke får tid til å fylle opp energien etter den fysiske påkjenningen. Hvis den fysiske påkjenningen blir mindre ved å bruke droner, kan det muligens forbedre dyrevelferden og kjøttkvaliteten.

2.8 Lover og regler rundt selvstyrte droner

Bruk av droner er regulert av ulike lover. «Luftfartstilsynet» sine nettsteder forklarer dette og viser til forordninger fra «EU» som har satt sikkerhetsregler som er felles for hele «EU/EØS» og Sveits, og gjelder både privatpersoner og virksomheter. Videre står det at ut fra dronen som brukes og bruksområde, skal dronen opereres innenfor de tre ulike kategorier åpen, spesifikk og sertifisert kategori [41].

I scenarioet av en fremtidig løsning, vil dronene være selvstyrte uten å bli kontrollert av en pilot. I tillegg vil dronene kunne bevege seg utenfor synsrekkevidden og ifølge «Luftfartstilsynet» sine nettsteder vil da bruken av dronene falle i kategorien spesifikk, der det må søkes om potensielt flere autorisasjoner og bl.a. gjennomføre en risikoanalyse [41].

Når det gjelder selvstyrte droner, vil reglene gjelde ut ifra hvordan styringen til dronene er implementert. «EASA», Det europeiske byrå for flysikkerhet, har utarbeidet et dokument med lett tilgjengelige regler om «EU» sin forordning 2019/947, som vi har benyttet i dette avsnittet [42]. Ifølge dette dokumentet faller løsningen enten innfor autonom eller automatisk operasjon. Dokumentet beskriver autonome operasjoner slik: «*'autonomous operation' means an operation during which an unmanned aircraft operates without the remote pilot being able to intervene*» [42, p. 22]. Dokumentet beskriver også forskjellen på autonom og automatisk operasjon slik: «*An autonomous operation should not be confused with an automatic operation, which refers to an operation following pre-programmed instructions that the UAS executes while the remote pilot is able to intervene at any time.*» [42, p. 24]. Det kommer altså an på hvordan løsningen blir til slutt hvorvidt det vil kategoriseres som en autonom eller automatisk operasjon. Dokumentet sier at om løsningen faller innenfor autonom operasjon, kan den ikke operere under reglene i åpen kategori [42, p. 272].

3 Metode

Denne delen vil først gå igjennom hvilke avgrensninger som er tatt for oppgaven. Videre vil det bli forklart teori, algoritmer og implementasjoner av dem. Det blir også beskrevet hvordan det er utviklet en simulator som simulerer et miljø med selvstyrte droner som kan drive sauer.

3.1 Valg av metode

I kapittel 2.4 ble det presentert en rekke artikler om autonom driving av sau. Alle metodene har sine fordeler og ulemper. Metodene er ikke nødvendigvis overførbare til denne masteroppgaven. Oppgaven skal være et utgangspunkt for en metode som skal fungere på norsk sauehold, og det er da viktig at dette tas hensyn til. På grunn av dette er det ikke direkte benyttet elementer fra disse artiklene. Det er derimot valgt å bruke en regelbasert metode i første omgang for å avdekke elementer som må jobbes videre med.

Som skrevet i problemstillingen, er metoden kun utviklet i en simulator og ikke testet i virkeligheten. Dette er både av henhold til lover, oppgaveomfang og oppgavens formål. Før man tester et system i virkeligheten, er det gunstig å ha et godt forarbeid for hvordan metoder og teori kan anvendes til formålet. Innkjøp av droner, planlegging av forsøk og evaluering av disse, bør ha et godt grunnlag ettersom det krever mye ressurser som kan reduseres kraftig med riktig kunnskap og erfaring. En simulering gir et godt utgangspunkt for innhenting av slik informasjon, da det muliggjør hyppig testing av mange ulike algoritmer og situasjoner.

3.2 Avgrensninger

Et fullstendig system for driving av sau ved å bruke autonome droner er en stor oppgave der lover, dyrevelferd og ulik teknologi må gå sammen. Denne prosessen kan ta flere år. Det er gjort relativt lite forskning på området, og forskningen som er gjort, er fortsatt helt i oppstartfasen, ikke utprøvd i virkeligheten og har heller ikke tatt hensyn til norske saueforhold. Målet med denne oppgaven er å utvikle et grunnlag for videre arbeid, der håpet er at det til slutt lages et fullstendig system. Det har derfor blitt gjort flere avgrensninger

for at omfanget ikke skal bli for stort. Før disse blir beskrevet, vil det foreslåtte utgangspunktet for oppgaven bli beskrevet i mer detalj.

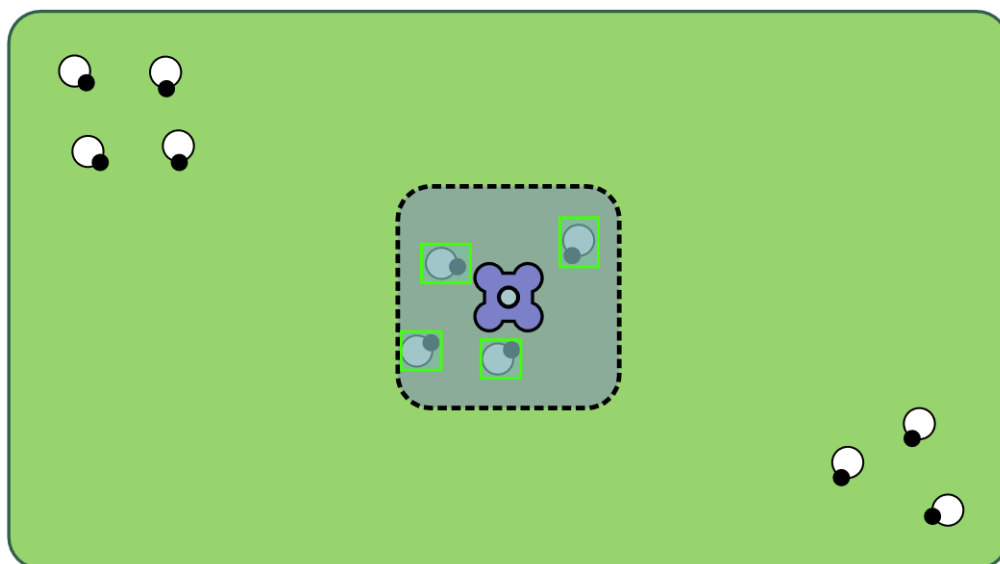
3.2.1 Videreutviklet utgangspunkt for oppgaven

Denne delen vil grundigere illustrere det videreutviklede utgangspunktet for oppgaven, som er utarbeidet i samråd med veileder.

Utgangspunktet går ut på at fire droner flyr ut over et åpent område der målet til dronene er å drive alle sauene inn i en innhegning. Området er stort og helt flatt uten hindringer som trær, bekker og gjerder. På området beiter det flere små grupper med sau. Sauene streifer fritt rundt der veileder erfarer at det er stier som sauene gjerne velger å følge [4]. Oppgavene til dronene kan deles i tre faser: Lokalisere, Forberede, Utføre.

Lokalisere

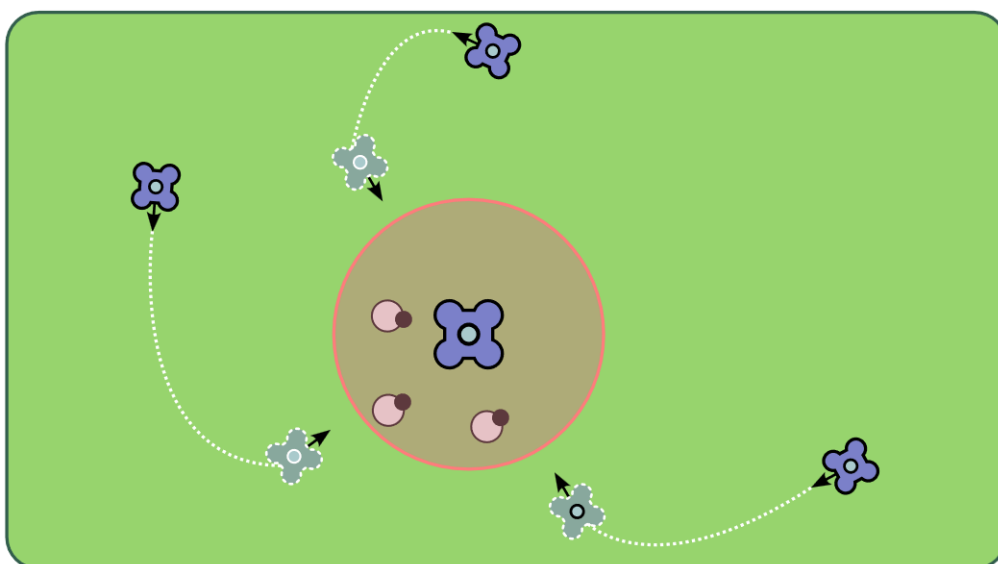
Den ene dronen flyr høyt i luften over bakken og er utstyrt med et kamera som peker nedover, videre kalt kameradronen. Ved å bruke kunstig intelligens, kan kameradronen analysere bildene og gjenkjenne sauene og deres posisjon. Når kameradronen har funnet en gruppe med sau, stiller den seg rett over flokken.



Figur 2: Kameradronen har lokalisert en gruppe med sauer på jordet.

Forberede

De tre andre dronene, videre kalt drivedronene, svever noen få meter over bakken. Deres jobb er å fly så nær sauene at de begynner å bevege seg i riktig retning. Før de skal gjøre det, må drivedronene bevege seg i startposisjonene, orientert bak sauene i motsatt retning av ønsket driveretning. Én drone stiller seg rett bak saueflokken, og de to andre stiller seg på hver side av saueflokken. Når dronene skal bevege seg i startposisjon, må det gjøres på en slik avstand at sauene ikke reagerer på dem, og de må ikke kollidere med hverandre. Når en drivedrone har beveget seg i startposisjonen, gir den beskjed til kameradronen. I denne fasen regner kameradronen ut den mest effektive ruten som sauene kan drives på.

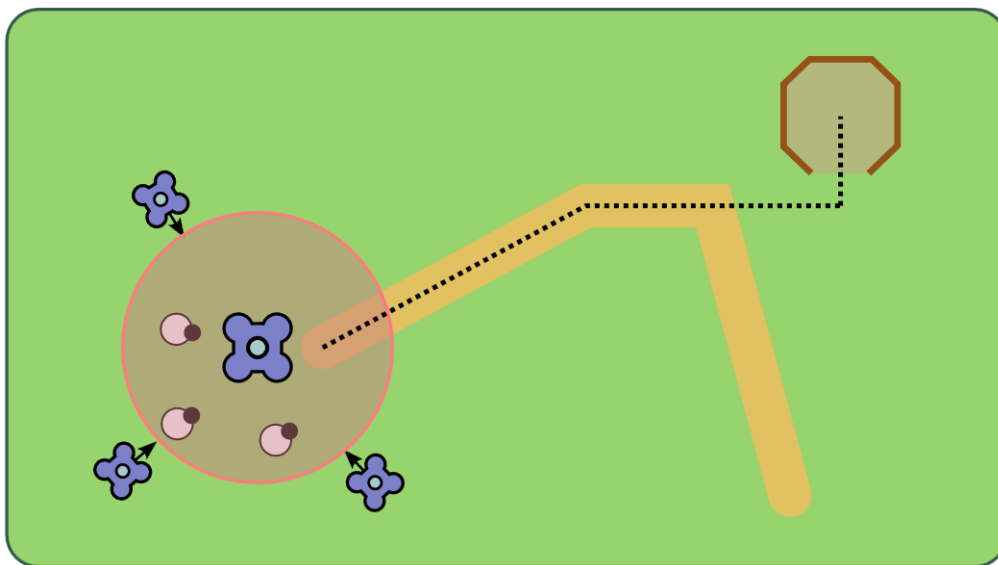


Figur 3: Dronene flytter seg i startposisjon uten å fly for nær sauene

Utføre

Når alle drivedronene er i startposisjon, iverksettes drivingen av sauene. Drivedronene beveger seg nærmere sauene, noe som gjør at sauene begynner å løpe bort fra dem. Drivedronene driver sauene på ruten som kameradronen har regnet ut. Underveis må drivedronene forsøke å unngå at saueflokken deler seg opp eller går i feil retning. Ved en stikryssning må dronene passe på at sauene går på riktig sti. Når gruppen med sau er flyttet til innhegningen, går

kameradronen tilbake til første fase der den leter etter en ny saueflokk. Disse fasen kjøres fra start til slutt til, til alle sauene på jordet er i innhegningen.



Figur 4: Dronene flyr nærmere sauene og driver dem på ruten (stiplet linje)

3.2.2 Flatt terreng uten hindringer

En avgrensning som er gjort for å forenkle metoden, er at sauene beiter på et flatt jorde uten hindringer. Dette valget er gjort selv om det ikke nødvendigvis samsvarer med virkeligheten ettersom mange sauer i Norge er på utmarksbeite i terreng som kan ha varierende hindringer og høyder. Det er mye som må tas hensyn til i natur og terreng med mye detaljnivå som f.eks. at dronene unngår trær, passer på å ikke drive sauene utfor skrenter eller i elver og driving på ulike høydenivå. Av det tidligere arbeidet på området mener vi det ikke er et godt nok grunnlag til å gå videre på dette før man i det hele tatt har et godt system som fungerer på flatt terreng.

3.2.3 Perspektiv

Siden det er valgt et flatt terreng, er det blitt valgt et todimensjonalt fugleperspektiv for simulatoren. I tillegg til det visuelle på skjermen, foregår alt av matematikk og algoritmer i kun to dimensjoner. En konsekvens av dette er at simulatoren ikke nødvendigvis gjenspeiler virkeligheten, og man får ikke et fullstendig bilde av situasjonene som blir testet. Det er for eksempel ikke en

høydeakse, så det er ikke sett på hvordan forskjellige høyder på dronene kan påvirker sauene.

3.2.4 Droneteknologi

Det er ikke tatt utgangspunkt i en spesifikk type drone i simulatoren, men det er forsøkt å etterligne egenskapene til et kvadrokopter¹. En av grunnene til at det ikke er valgt en spesifikk drone, er at oppgaven ikke skal testes i virkeligheten, og kun skal utforske de generelle mulighetene. Det er derfor ikke tatt store hensyn til begrensninger som droner i dag kan ha. Dronene i simulatoren kan bevege seg i alle retninger, rotere og kan sende og motta signal til andre droner uten forsinkelser. Drivedronene vil være utstyrt med høyttalere som sender ut lyd, men dette er ikke simulert. Sauene reagerer kun på avstanden til dronen. Kameradronen er heller ikke simulert, siden saueposisjonen er kjent til enhver tid.

3.2.5 Kollisjon mellom droner

I virkeligheten ønsker man ikke at droner skal kollidere av blant annet økonomiske årsaker. Vårt forslag til å hindre dette er ved at dronene flyr på ulike høyder når rutene deres krysser. Dette er derimot ikke implementert i simulatoren og er kun gjort rede for her. Droner kan gli over hverandre, og det er da tenkt at de flyr på ulike høyder.

3.2.6 Sau og innhegning

Oppgaven har ikke tatt utgangspunkt i noen spesiell type sau. Å simulere sauer og deres atferd er en stor og vanskelig oppgave så den er meget forenklet.

En innhegning har typisk gjerder og en inngangsport. I simulatoren er dette forenklet til et sirkulært område uten gjerder. Som beskrevet tidligere, skal ikke oppgaven ta hensyn til hindringer. Så lenge en sau er innenfor så er den regnet som sanket, og blir slettet fra simulatoren.

¹ <https://no.wikipedia.org/wiki/Kvadrokopter>

3.2.7 Automatisk generering av ruter

Det er ikke implementert automatisk generering av ruter som kameradronen gjør i forberede-fasen. Å lage en algoritme som finner en god rute som benytter seg av stier og som unngår mest mulig krappe svinger er en viktig framtidig oppgave. Dette kan være en egen masteroppgave med tanke på det store omfanget til problemstillingen. Derfor er denne genereringen gjort manuelt på forhånd. Konsekvensen av dette er at alle tester vil bli gjort i ferdige miljø der ruten er satt på forhånd. De eneste rutene som blir generert, skjer nå i utføre-fasen, og består av å lage en rute til hovedruten, og fra ruten til innhegningen.

3.3 Utviklingsplattform

Simulatoren er utviklet i spillmotoren «*Godot*»². Versjonen som er brukt er «*Godot 3.5*». Spillmotoren tilbyr flere nyttige funksjonaliteter som gjør flere elementer enklere å implementere. En av disse er at det er enkelt å legge til brukergrensesnitt som gjør det lettere å legge til knapper og andre komponenter for å kontrollere parametere. Den har også innebygde funksjoner for tegning av elementer, som gjør det mulig å visualisere sauer, droner og andre objekt. I tillegg er matematiske og geometriske funksjoner lett tilgjengelige.

3.4 Definisjoner på variabler

I delkapitlene under vil flere formler bli definert. Noen av variablene som blir brukt i disse formlene blir nå kort forklart. Stor N er sauene innenfor en gitt radius av en gitt sau. Liten s indikerer den gitte sauen som har en posisjon s_p og en hastighetsvektor s_v . Ved iterasjon gjennom N , indikerer variabelen i opphøyd som s^i at det er snakk om en iterert sau sine egenskaper. Retningen til sauen bestemmes av retningen til hastighetsvektoren. Droner følger samme notasjon med bokstaven D for dronene innen en gitt radius og d for en gitt drone.

² <https://godotengine.org/>

3.5 Sauemodell

En stor del av simulatoren er en metode for å simulere en sau sin atferd og interaksjoner i nærvær av andre sauer og farer. Modellen i simulatoren er i utgangspunktet basert på den utvidet versjonen av «*Boids*» beskrevet i kapittel 2.6. I tillegg er den videre modifisert for å tilpasse seg atferden til sau ytterligere. Dette kapitlet vil ta for seg hvordan sauemodellen er implementert.

3.5.1 «*Boids*»

I den originale artikkelen om «*Boids*» presenterer Craig W. Reynolds tre enkle konsepter som gjør det mulig å simulere større flokker med dyr [34]. Dette kapitlet baserer seg på Reynolds originale konsepter, mens implementeringen av algoritmene har utgangspunkt i bacheloroppgaven [37] som er nevnt i kapittel 2.6.

Generelt går «*Boids*» ut på å regne ut forskjellige krefter som gir en totalkraft som virker på hvert dyr i flokken. Den resulterende kraften regnes ved å summere kreftene fra hver av de tre konseptene. Implementasjonene av disse er beskrevet under.

Kollisjonsunngåelse (Seperation)

Det første konseptet til Reynolds handler om at dyrene i en flokk skal unngå å kolliderer med hverandre. Sauemodellen gjør dette ved å summere de inverterte avstandsvektoren til sauene N innen en gitt radius R_{SEP} . Første del av algoritmen regner et forholdstall som blir høyere jo kortere avstanden a mellom sauene er. Dette tallet blir så multipliser med vektoren mellom sauene for å vektlegge de nærmeste sauene. I [37] er siste del av algoritmen notert som $(s_p^i - s_p)$. Dette gir en vektor som peker mot den andre sauene, og skaper derfor en tiltrekningskraft framfor en frastøtningkraft. Vi har derfor notert dette motsatt slik at kraften har en frastøtende evne.

$$s_{SEP} = \sum_{i=0}^N \frac{R_{SEP} - a}{a} * (s_p - s_p^i) \quad (1)$$

Fartstilpasning (Alignment)

Det andre konseptet til Reynolds handler om at dyrene i en flokk skal forsøke å tilpasse farten og retningen til hverandre. Sauemodellen gjør dette ved å regne ut forskjellen mellom egen hastighet og gjennomsnittshastigheten til sauene N innen gitt radius R_{ALI} . I [37] er det notert s_p istedenfor s_v i slutten av algoritmen, men vi antar dette er en skrivefeil.

$$s_{ALI} = \frac{\sum_{i=0}^N s_v^i}{L} - s_v \quad (2)$$

Flokktiltrekning (Cohesion)

Det tredje konseptet til Reynolds handler om at dyrene i en flokk ønsker å bevege seg mot midten av den. Sauemodellen gjør dette ved å regne ut en kraft fra midtpunktet til sauene N innenfor radius R_{COH} og sauens posisjon s_p .

$$s_{COH} = \frac{\sum_{i=0}^N s_p^i}{L} - s_p \quad (3)$$

3.5.2 Delgado-Matas utvidelser

Som beskrevet i 2.6, utvidet Delgado-Mata konseptene til Reynolds ved å legge til konseptene flukt, redsel og spredning av redsel. Disse konseptene har blitt brukt i [37] og implementasjonen av konseptene er beskrevet under.

Flukt (Escape)

Det første utvidete konseptet er flukt, som handler om et dyrs ønske å bevege seg vekk fra fare. I [37] er dette tenkt på som sauenes reaksjon på en gjeterhunds nærvær. Dette har vi overført til sauemodellen som sauers reaksjon på dronene. Flukt er implementert på samme måte som kollisjonsunngåelsen mellom sauene, men den resulterende kraften blir også normalisert. Dette er fordi kraften blir skalert senere når den totale kraften blir utregnet i delkapittel 3.5.4. I [37] er det kun snakk om én fare, mens i vår modell kan det være flere droner som skremmer sauene. Derfor er implementasjonen modifisert slik at den tar hensyn til dette ved å iterere gjennom D .

$$s_{ESC} = norm \left(\sum_{i=0}^D \frac{R_{ESC} - a}{a} * (s_p - d_p^i) \right) \quad (4)$$

Redsel (Fear)

Det andre utvidete konseptet er redsel, som indikerer hvor redd dyret er slik at atferden kan endre seg. I vårt tilfelle får sauene redsel av dronene, og verdien endrer seg ut fra avstanden a mellom dronen og en gitt sau. Ved bruk av en sigmoidfunksjon blir redselen en verdi mellom 0 og 1. Dette tallet blir brukt senere til å vektlegge ulike konsept. I implementasjonen av konseptet redsel hadde [37] kun én gjeterhund. I vår oppgave er det flere droner. Vi løste dette ved at sauen kun får redsel av den nærmeste dronen.

$$S_{FEAR} = \frac{1}{\pi} * \arctan\left(\frac{R_{ESC} - a}{C_{SIG}}\right) + 0.5 \quad (5)$$

Redselsspredning (Contagion)

Det tredje utvidete konseptet er spredning av redsel gjennom en flokk. [37] finner redselsspredningen ved å ta gjennomsnittsredselen til sauene i nærheten, men det er noe uklart om gjennomsnittsredselen skal regnes ved å ta redselen til hver sau eller totalredselen når redsel og redselsspredning regnes i sammen. Vi har valgt å kun ta gjennomsnittet av redselen.

$$S_{CON} = \frac{\sum_{i=0}^N S_{FEAR}^i}{L} \quad (6)$$

For å regne totalredselen til en sau, regnes først redselen til sauen i forhold til den nærmeste dronen, etterfulgt av redselsspredningen fra andre sauer. Til slutt blir dette summert, der hver del er vektlagt av to konstanter.

$$S_{TF} = C_{FEAR} * S_{FEAR} + C_{CON} * S_{CON} \quad (7)$$

3.5.3 Ytterlige utvidelser

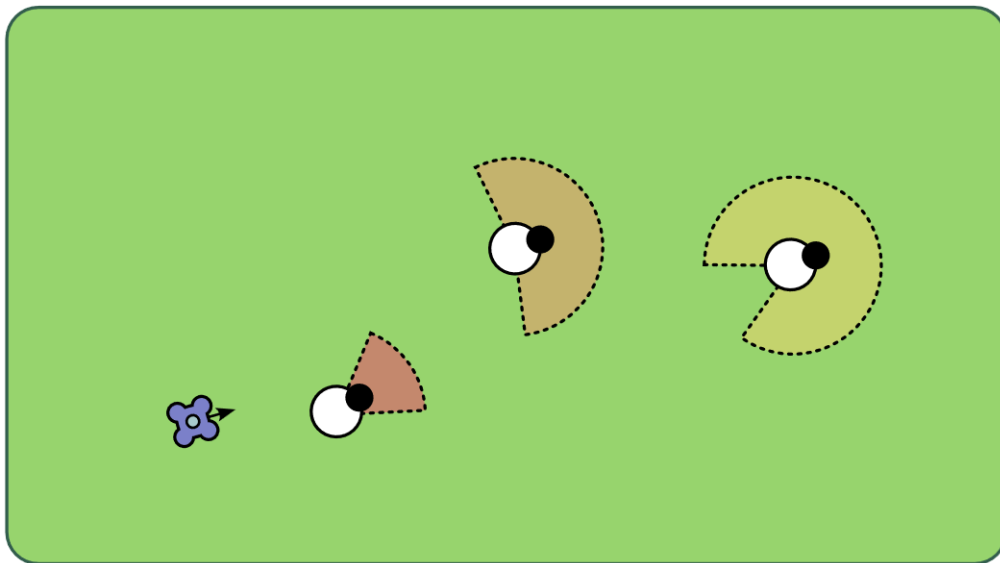
I tillegg til Reynolds' og Delgado-Matas sine konsept, har vi utviklet sauemodellen videre ved å legge til to konsept til, som blir beskrevet under.

Synsfelt

Veileder erfarer at etter hvert som sauene blir drevet, dannes en linjeformasjon, der sauene går på rekke [4]. Dette kan også observeres i en video på YouTube, der en bonde har brukt en drone utstyrt med en høyttaler til

å drive sauer manuelt³. For å simulere denne oppførselen, får saueene smalere synsfelt når de er redde. Dette medfører at saueene blir kun påvirket av saueene som befinner seg innenfor synsfeltet. Dette ble implementert ved å lage et dynamisk synsfelt på hvilke sauer en valgt sau blir påvirket av. Synsfeltet blir interpolert mellom 360 grader og en vinkel w ut ifra den totale redselen til sauen.

$$s_{FOV} = 360 - (360 - w) * s_{TF} \quad (8)$$



Figur 5: Illustrasjon av hvordan synsfeltet til saueene endrer seg i nærvær av en drone.

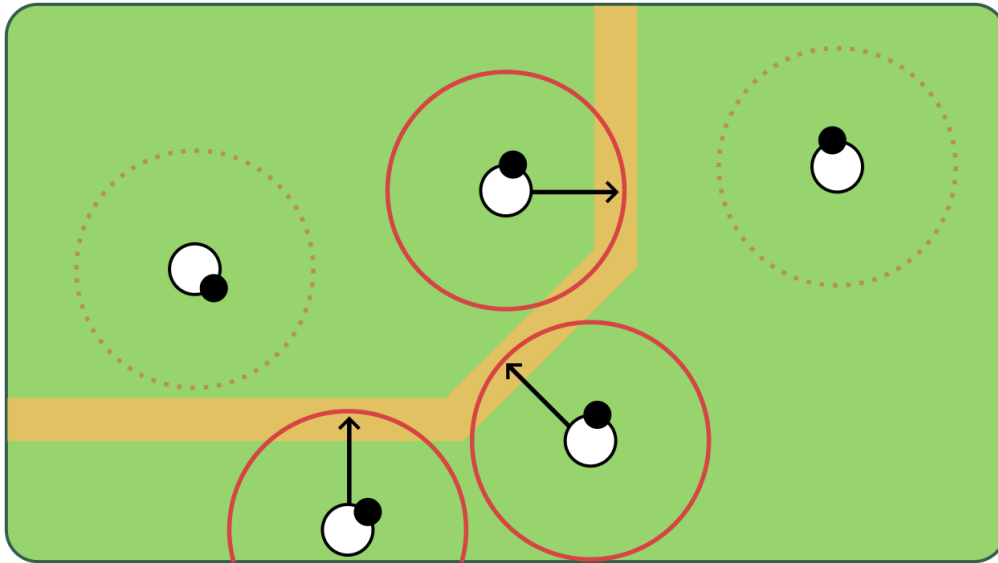
Stifølging

Ifølge veileder går sauer gjerne på stier i terrenget som dem er vant til å gå på [4]. Dette kan simuleres ved å la saueene bli tiltrukket av stiene. Stiene er implementert som en rekke linjestykker i simulatoren. Når en sau nærmer seg en sti innen en radius R_{PATH} så blir sauen tiltrukket av stien. Ved hjelp av funksjoner innebygd i «Godot», finner vi det nærmeste punktet p på stien og regner ut en tiltrekningskraft mot dette punktet.

$$s_{PATH} = norm(p - s_p) \quad (9)$$

³ “Barking drones used on farms instead of sheep dogs”, RNZ

<https://www.youtube.com/watch?v=CTjVjKClpyU>



Figur 6: Sauene som er nær stien blir tiltrukket av den.

3.5.4 Total utregning av sauemodellen

Dette delkapittelet vil gå igjennom hvordan hvert konsept som er beskrevet ovenfor, er koblet i sammen for å danne en fullstendig modell. For hvert tidsintervall blir konseptene regnet ut for hver sau og vektlagt i et totalregnestykket for en kraft. Utregningen av denne kraften er basert på [37] sin implementasjon.

Først regnes totalredselen til sauene. Så blir «Boids» grunnkonsepter og de utvidete konseptene flukt og stier regnet ut. Hvert konsept blir vektlagt med to ulike konstanter, utenom flukt som kun vektlegges med en konstant. De to ulike konstantene gjør det mulig å endre adferden til sauene ut fra redselen. Den første konstanten har alltid en effekt på resultatvektoren, mens den andre konstanten kun har en effekt når sauene har redsel. Grunnen til at flykt kun har den sistnevnte konstanten er at sauene ikke skal flykte når de ikke har redsel.

$$\begin{aligned}
 S_{FORCE} = & C_{COH} * (1 + FC_{COH} * S_{TF}) * S_{COH} + \\
 & C_{SEP} * (1 + FC_{SEP} * S_{TF}) * S_{SEP} + \\
 & C_{ALI} * (1 + FC_{ALI} * S_{TF}) * S_{ALI} + \\
 & C_{PATH} * (1 + FC_{PATH} * S_{TF}) * S_{PATH} + \\
 & FC_{ESC} * S_{TF} * S_{ESC}
 \end{aligned} \tag{10}$$

Den resulterende kraften blir så begrenset til en maks lengde C_{FORCE} , og lagt til sauens hastighetsvektor. For å begrense farten til sauene slik at den ikke blir for stor, gjør vi som i [37] og setter en maks lengde på hastighetsvektoren. Denne lengden blir påvirket av redsel, slik at sauene kan bevege seg raskere når de er redde.

$$S_{SPEED} = S_{TF} * C_{S_SPEED} \quad (11)$$

Hastighetsvektoren blir til slutt lagt til posisjonen til sauene for å få sauens nye posisjon.

3.5.5 Sauemodell radier og synsfeltverdi

Sauemodellen sine konsepter baserer seg som oftest på sauene i nærheten av en gitt sau. Dette er sauer som er innenfor en viss radius fra posisjonen til sauene. Disse radiene har konstant størrelse. I tillegg benytter konseptene kollisjonsunngåelse, fartstilpasning, flokktiltrekning og stifølging seg av konseptet synsfelt som beskrevet tidligere, hvor konstantene FOV_{SEP} , FOV_{ALL} , FOV_{COH} og FOV_{PATH} blir brukt. Dette betyr at ut ifra redselen vil sauene innenfor radien kun bli tatt med i beregningene om sauene er også innenfor synsfeltet.

I sauemodellen er det seks radier som er brukt. Det er tre radier for hvert av «Boids»-konseptene; R_{COH} , R_{SEP} og R_{ALL} . Disse bestemmer hvilke sauer som skal påvirke en gitt sau. Konseptet flukt har også en radius R_{ESC} , som bestemmer avstanden der en sau begynner å flykte. Denne radien blir også brukt i utregningen av redselen til sauene. Sauemodellen bruker radien R_{CON} , som bestemmer hvilke sauer en gitt sau sin redsel påvirkes av. Sauemodellen bruker radien R_{PATH} , som bestemmer hvor nær en sau må være en sti før den blir tiltrukket.

3.6 Dronemodell

I tillegg til sauemodellen er det laget en dronemodell. Dette kapittelet vil forklare hvordan dronene er organisert og hvordan de mottar og utfører oppgaver.

3.6.1 Dronehierarki

Modellen består kun av de tre dronene som skal drive sauene. Kameradronen som flyr over saueflokken og som det er tenkt skal sende kommandoer, har ingen påvirkning på simulatoren som beskrevet i avgrensningene. Oppgavene til kameradronen er simulert gjennom en komponent i programmet som sender ruter til dronene.

De tre dronene har hver sin rolle. Den ene skal være bak sauene og drive dem framover, fra nå av kalt bakdronen. De to siste dronene skal holde seg på hver sin side av flokken og passe på at sauene holder seg på ruten, fra nå av kalt styredronene.

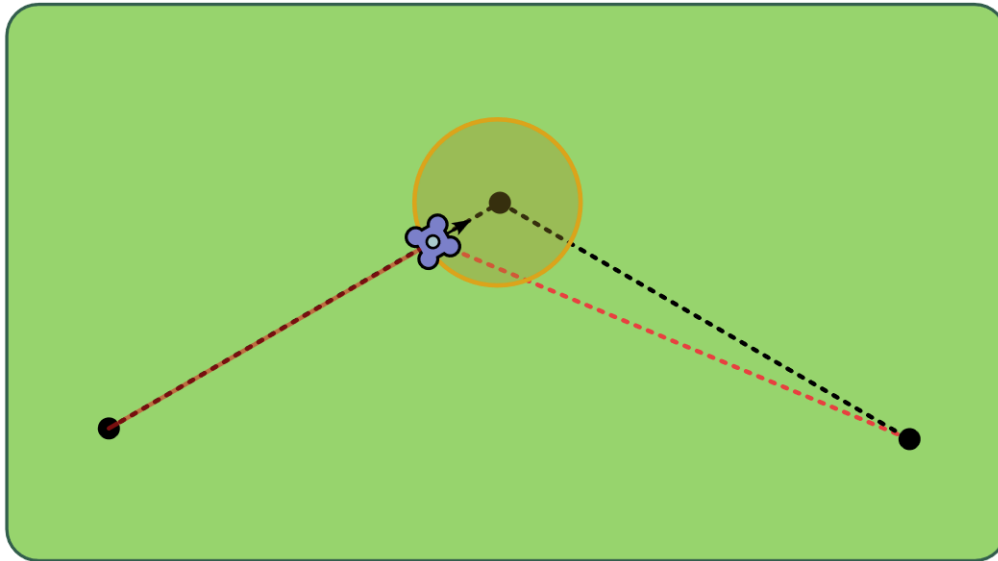
3.6.2 Motta oppgaver

Dronene mottar oppgaver i form av ruter og hastighetsjusteringer. Rutene er i form av en liste med punkt som skaper linjestykker mellom seg. Får dronen en ny rute, blir den tidligere ruten overskrevet. Hver drone får sin egen individuelle rute. Dronene kan også motta kommando for endring av hastigheten som er i form av et reelt tall når dette er nødvendig.

3.6.3 Utføre oppgaver

Når dronene har fått en rute, vil de bevege seg mot neste/forrige punkt fra den nåværende plasseringen på ruten. Dronene kan bevege seg i begge retninger langs ruten, dette blir bestemt ut fra fortegnet til dronens hastighet. Ruten har en bestemt retning som følge av rekkefølgen til punktene i listen. Når dronen har en positiv hastighet, beveger dronen seg mot neste punkt i listen, har den derimot en negativ hastighet, beveger den seg mot det forrige punktet i listen.

For å sjekke at dronen har nådd punktet den beveger seg mot, er det ikke nok å sjekke at posisjonen til dronen er lik punktet den går mot. Dronen får konstant en styringskraft mot punktet, og det er lite sannsynlig at den beveger seg nøyaktig på posisjonen. Derfor er det satt en konstant som beskriver radien til en sirkel, R_{GOAL} . Radien brukes til å sjekke om dronen er innenfor en viss avstand fra punktet den skal til, for å vite når dronen skal begynne å bevege seg mot neste punkt. En liten konsekvens av dette kan være at dronen begynner å bevege seg til neste punkt for tidlig eller for sent. Se Figur 7.



Figur 7: Illustrasjon av hvordan det sjekkes at dronen skal gå til neste punkt. Rød stippet linje viser hvordan dronen beveger seg mot neste punkt.

Dronene beveger seg på samme måte som sauene, hvor det er en vektor som akselererer dronen. Denne vektoren blir utregnet som en normalisert vektor mellom dronen og punktet dronen skal bevege seg mot, for å så skalere den opp med den ønskede hastigheten til dronen. For at dronen ikke skal bevege seg forbi punktet, blir kraften skalert ned med forholdet mellom a , avstanden til punktet og R_{GOAL} . Denne skaleringen skal kun skje når dronen er innenfor R_{GOAL} , derfor blir forholdet aldri større enn 1 ved å begrense det til intervallet mellom 0 og 1. Den første delen av utregningen resulterer i den ønskede vektoren dronen skal ha for å nå punktet, mens funksjonen skal gi dronen en akselerasjonsvektor. Derfor subtraheres dronens hastighet til slutt for å regne ut hva som må legges til for få denne ønskede vektoren.

$$d_{FORCE} = norm(p - d_p) * d_{SPEED} * \min\left(\frac{a}{R_{GOAL}}, 1\right) - d_v \quad (12)$$

Denne kraften blir begrenset til en maks lengde på C_{FORCE} for å så bli lagt til hastighetsvektoren til dronen. Til slutt blir hastighetsvektoren brukt til å oppdatere dronens posisjon ved å legge den til posisjonen til dronen. For å begrense makshastigheten til dronen blir lengden til hastighetsvektoren begrenset til C_{D_SPEED} .

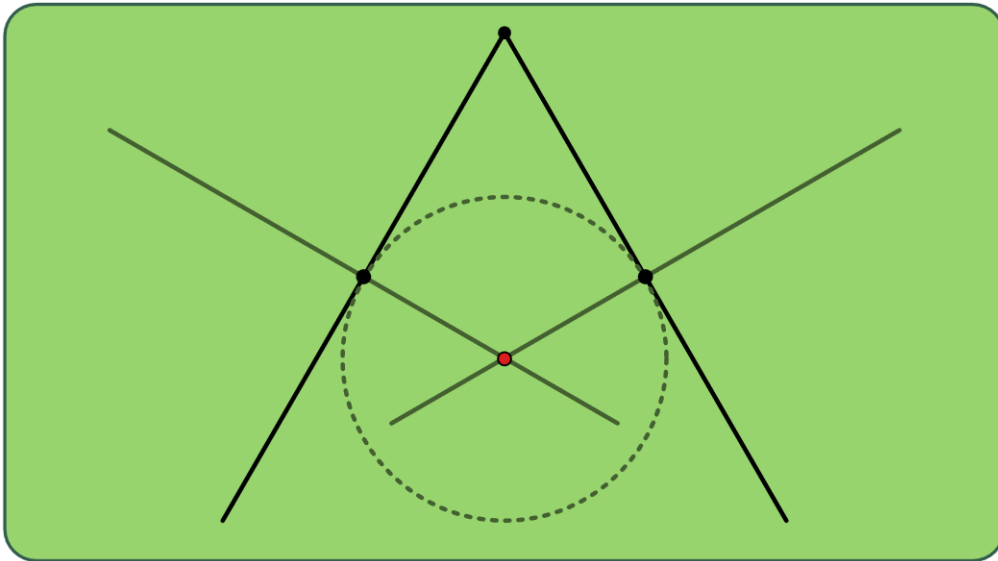
3.7 Sankealgoritme

Det siste som gjenstår, er en regelbasert algoritme som kontrollerer dronene, som igjen kontrollerer sauene. Sauene blir ikke direkte kontrollert, men flytter på seg i nærvær av dronene. Algoritmen er delt opp i tre faser som beskrevet i 3.2.1, lokalisere, forberede og utføre. Hver fase i algoritmen er beskrevet i delkapitlene under.

3.7.1 Generering av ruter

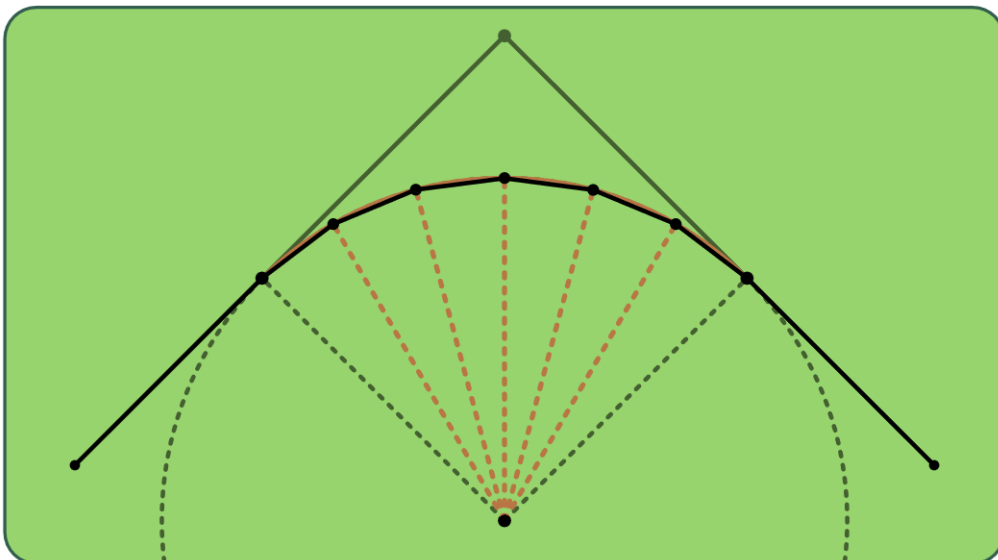
I løpet av sankingen blir det sendt flere ruter til dronene. Som beskrevet tidligere, er dette i form av en liste med punkter som dronen følger etter. Som beskrevet i avgrensninger, blir det manuelt plassert en forhåndsdefinert rute. For å redusere veldig brå bevegelser i overgangene mellom linjestykkene, regner simulatoren en mykere sving med flere punkter før den sender ruter til dronen. Det er lagt to funksjoner for å lage mykere svinger.

Den første funksjonen regner ut dreiepunktet til en tenkt sving ved å finne en sirkel som tangeres av to linjestykker. Først velger man start- og endepunktet til svingen, som er en viss lengde C_{TURN} inn fra hjørne mellom de to linjestykkene. Er halve lengden på det minste linjestykket mindre enn C_{TURN} blir denne brukt istedenfor C_{TURN} . Ved å bruke disse to og hjørne dannes to linjer som tangerer svingens sirkel. Fra tangeringspunktet peker to linjer normalt inn mot senteret av sirkelen til svingen. Krysningspunktet av disse gir dreiepunktet til svingen. Se Figur 8 for illustrasjon.



Figur 8: Illustrasjon av hvordan dreiepunktet (rød prikk) blir funnet.

Den andre funksjonen lager en liste med punkt som definerer svingen ved å splitte deler av sirkelen opp i små segmenter. Funksjonen er avhengig av tre punkt. Start- og endepunktet til svingen og dreiepunktet. Linjestykkene blir regnet ut ved å finne en vektor fra dreiepunktet til startpunktet, og deretter rotere denne med en vinkel C_{ROT_ANGLE} . Dette blir repetert helt til et linjestykke er forbi endepunktet, der siste linjestykke ender opp i endepunktet. Se Figur 9 for illustrasjon.

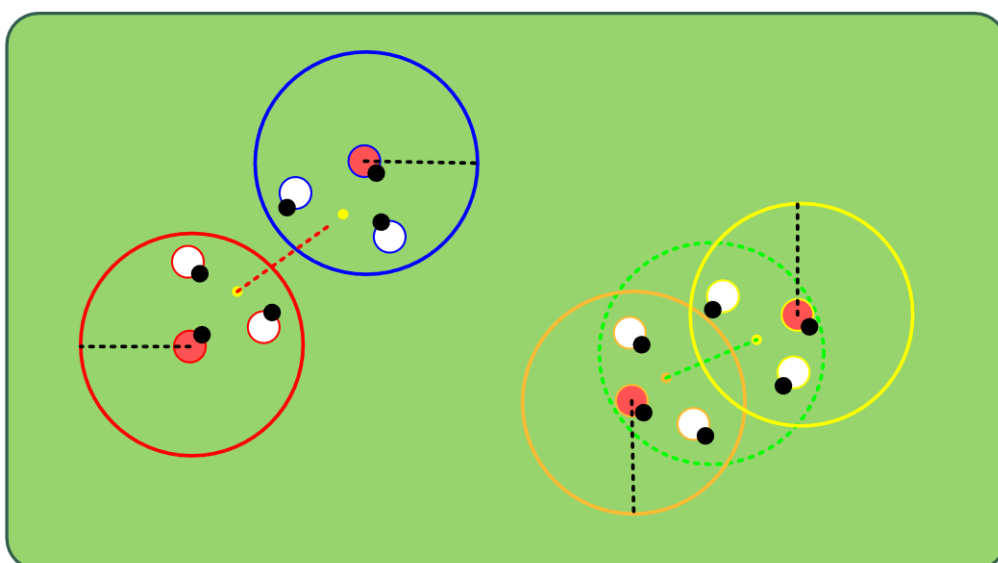


Figur 9: Illustrasjon av hvordan segmenter av sirkelen blir nye linjestykker til dronen.

3.7.2 Lokalisere

I den første fasen skal kameradronene fly over jorden og lete etter saueflokker. Dette har ikke vært en del av oppgaven, ettersom det finnes løsninger på dette fra før, som ikke trengs å lages på nytt. Simulatoren vet til enhver tid hvor alle sauene er. Dette steget er derfor forenklet fra å bruke kunstig intelligens til å analysere flyfoto for å lokalisere hver sau, til å hente posisjonene direkte fra programmet. I steget blir det derimot kjørt en funksjon som deler sauene opp i flokker, slik at dronene kan drive en flokk av gangen.

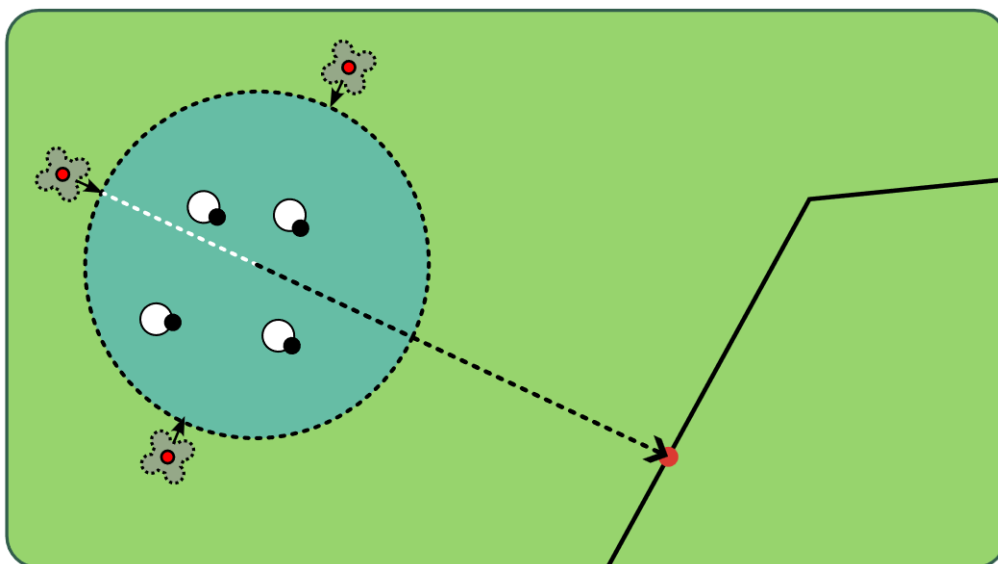
Det finnes flere måter å dele opp individer i grupper. Vi valgte å gjøre det ved å sjekke om avstand mellom sauene er innenfor en radius. Først itereres det gjennom alle sauene, og for hver sau sjekkes avstanden til de andre. Hvis de er innenfor radien R_{HERD} , blir de lagt til i samme flokk, og de vil ikke bli iterert videre. Etter dette, itereres det gjennom alle flokkene og sjekker avstand mellom gjennomsnittspunktet til flokken. Er avstanden mellom flokkene innenfor R_{HERD} blir de om til en flokk.



Figur 10: Illustrasjon av flokkalgoritme. I første iterasjon sjekkes det hvilke sauer som er innenfor radien til hver av de røde sauene. Sauene i hvitt blir lagt til i de respektive flokkene (Rød, Blå, Gul og Oransje sirkler). I neste iterasjon sjekker algoritmen om midtpunktene til flokkene er innenfor radien. Den grønne sirkelen viser to flokker som har blitt slått sammen til en flokk.

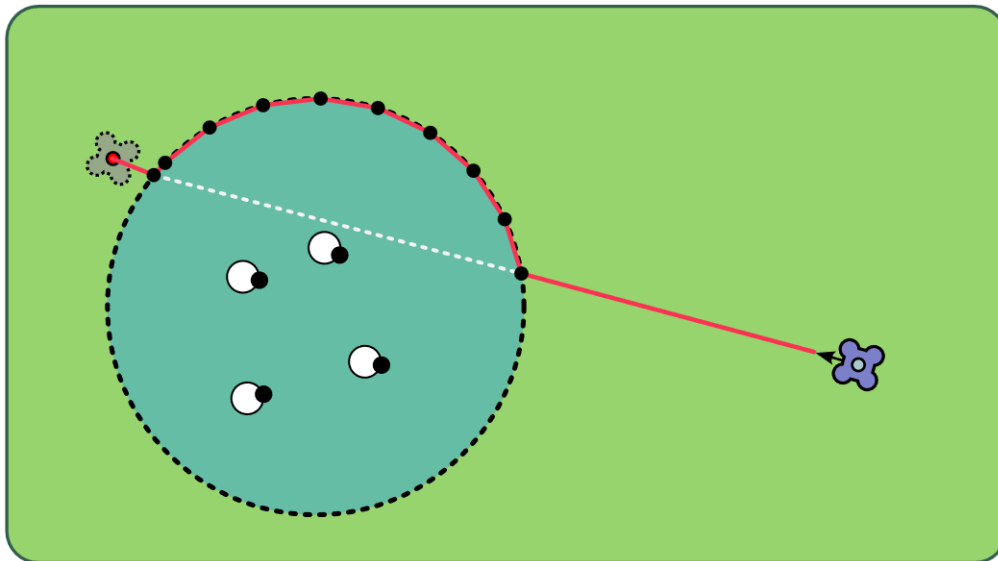
3.7.3 Forberede

I neste fasen flytter dronene seg til flokken som skal drives. Hvis det er flere flokker så velger simulatoren den største flokken. Dronene skal posisjonere seg rundt flokken på en avstand R_{SAFE} som er stor nok til at sauene ikke reagerer på dronene. Hver drone får en posisjon, som er bak og på hver side av flokken. Styredronene er rotert med en vinkel C_{ANGLE} rundt midtpunktet til flokken med avstand R_{SAFE} . Bakdronen er plassert bak flokken slik at flokken blir drevet i riktig startretning når dronene nærmer seg flokken i neste steg. Startretningen er mot det nærmeste punktet til den forhåndsdefinerte ruten. Se Figur 11 for illustrasjon.



Figur 11: Illustrasjon av startposisjonene til dronene. Bakdronen blir plassert motsatt i forhold til nærmeste punkt på den forhåndsdefinerte ruten.

Dronene må unngå å fly innenfor R_{SAFE} der sauene blir påvirket, og dronene må derfor fly utenfor denne radien. Dette løses ved å lage flere linjestykker rundt området som vist i Figur 12. Her benyttes den ene funksjonen fra 3.7.1 som lager flere segmenter, der startpunktet og endepunktet på svingen blir de to skjæringspunktene mellom påvirkningsområdet og linjestykket mellom dronen og startposisjonen, for deretter å danne segment med midtpunktet til flokken som sentrum.



Figur 12: Illustrasjon av hvordan dronen unngår påvirkningsområdet ved å danne segmenter på kanten av området.

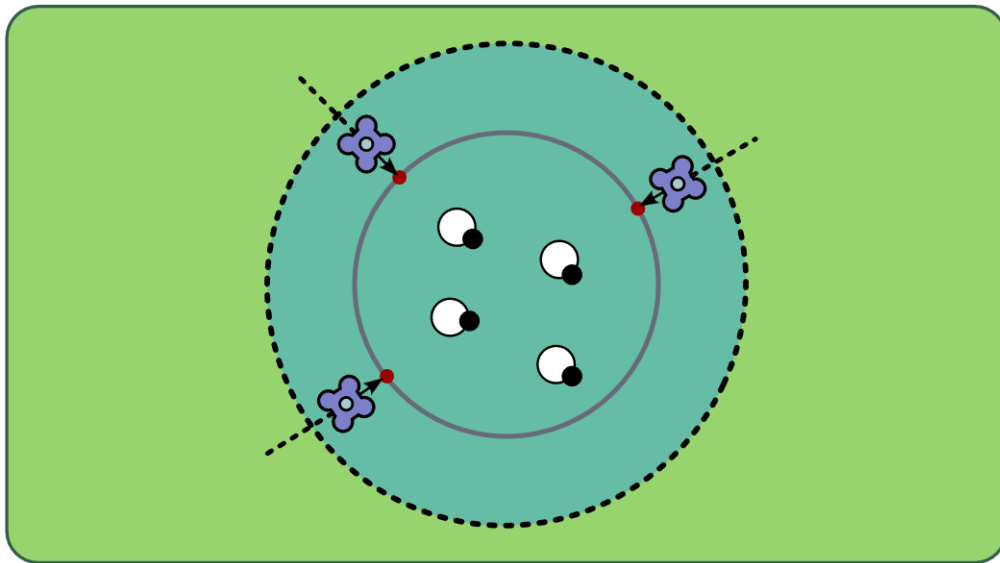
Dronene beveger seg mot startposisjonen på ruten, men får oppdaterte ruter kontinuerlig ettersom flokken kan bevege på seg. Simulatoren sjekker kontinuerlig om alle dronene er innenfor en viss avstand R_{GOAL} fra startposisjonen som initierer neste fase.

3.7.4 Utføre

For å forenkle beskrivelsen av den siste fasen, er den delt opp i tre steg. Initier driving, generer rute og følg rute. Hver av disse stegene vil bli beskrevet under.

Initier driving

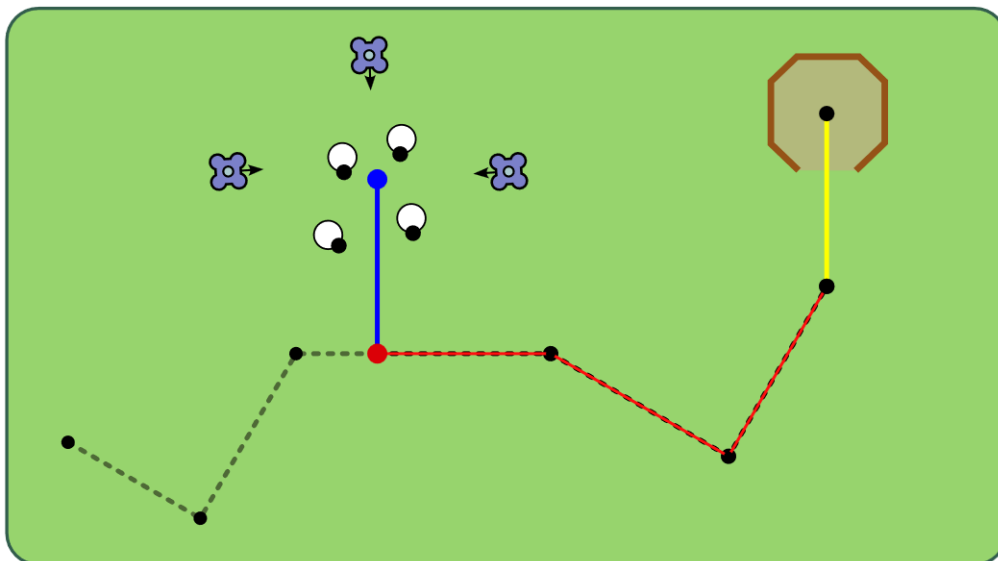
I dette kortet steget beveger dronene seg nærmer flokken slik at sauene begynner å bevege seg i rutens retning. Dronene må ikke bevege seg for langt inn da dette kan medføre at sauene beveger seg ukontrollert. Dronene får en liten rute fra sin egen posisjon til et punkt nærmere sentrum til saueflokk som har avstand R_{DANGER} . Når dronene har fullført denne ruten starter neste steg.



Figur 13: Illustrasjon av dronene som beveger seg nærmere flokken.

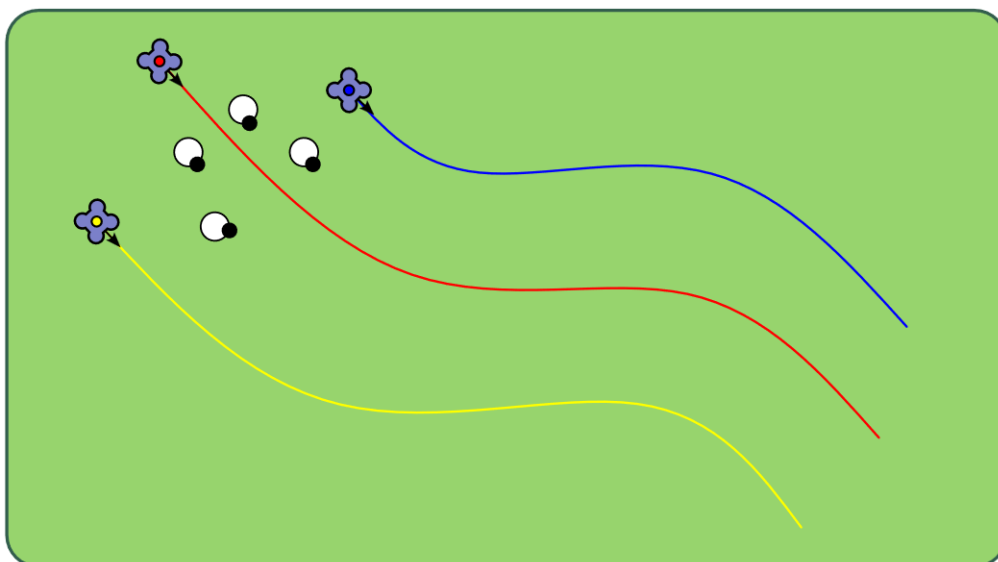
Generer rute

I dette steget generes en fullstendig rute fra saueflokkens posisjon til innhegningen. Som tidligere beskrevet, er det allerede laget en forhåndsdefinert rute som sauene skal drives på. Først må dronene drive sauene til denne ruten ved å lage et linjestykke fra midtpunktet av saueflokken til nærmeste punkt på den forhåndsdefinerte ruten. Dette punktet blir funnet ved hjelp av en innebygd funksjon i «*Godot*». Punktene på ruten som ligger før det nærmeste punktet blir ignorert. Etterpå blir resten av punktene til den forhåndsdefinerte ruten lagt til. Helt til slutt blir det laget et punkt fra siste punkt på den forhåndsdefinerte ruten til innhegningen. Etter hvert som punktene blir lagt til den fullstendige ruten, blir funksjonene som er beskrevet i 3.7.1 benyttet for å gjøre svingene mykere mellom punktene i ruten. Se Figur 14.



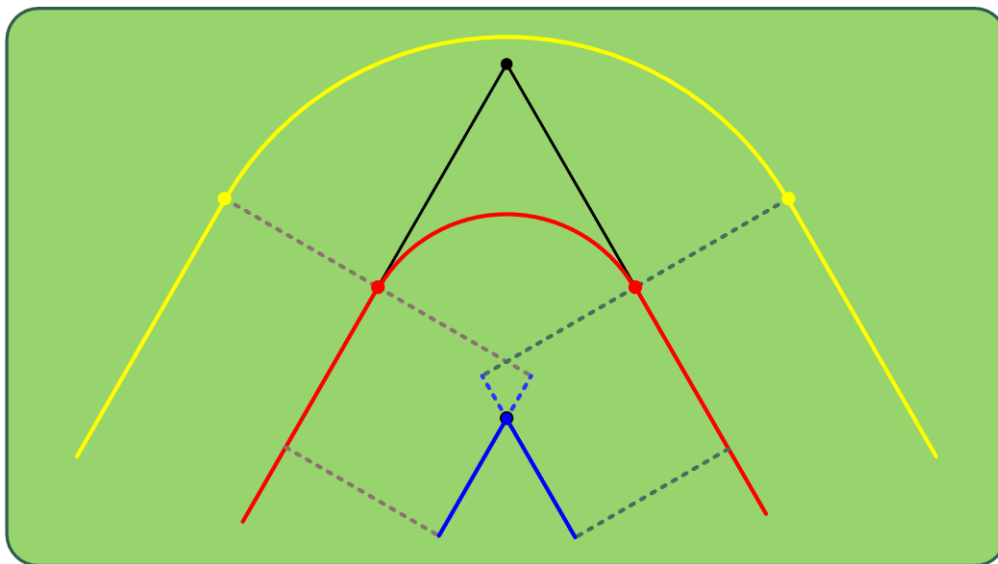
Figur 14: Illustrasjon av generering av fullstendig rute. Først en linje fra sauene til nærmeste punkt (blå), så den forhåndsdefinerte ruten (rød), og til slutt et linjestykke til innhegningen (gul).

Ruten som blir generert først er ruten til bakdrønen. I og med at styredronene er på hver sin side av flokken og skal holde avstanden R_{DANGER} , får styredronene sin egen rute som er ruten til bakdrønen, men forskjøvet en konstant distanse R_{DANGER} , som vist i Figur 15. Dette blir gjort ved å ta punktene fra bakdrønen sin rute og flytte dem utover langs normalen..



Figur 15: Illustrasjon av hvordan rutene til styredronene er forskjøvet

En konsekvens som kan inntreffe hvis en sving er over en viss krapphet, er at linjestykkene som er forskjøvet kan krysse hverandre. Dette problemet er løst ved å sjekke om det oppstår en slik krysning, og i slike tilfeller benyttes dette krysningspunktet som svingen, for å holde riktig avstand til sauene. Se Figur 16. Når rutene til dronene er ferdig generert, starter neste steg.

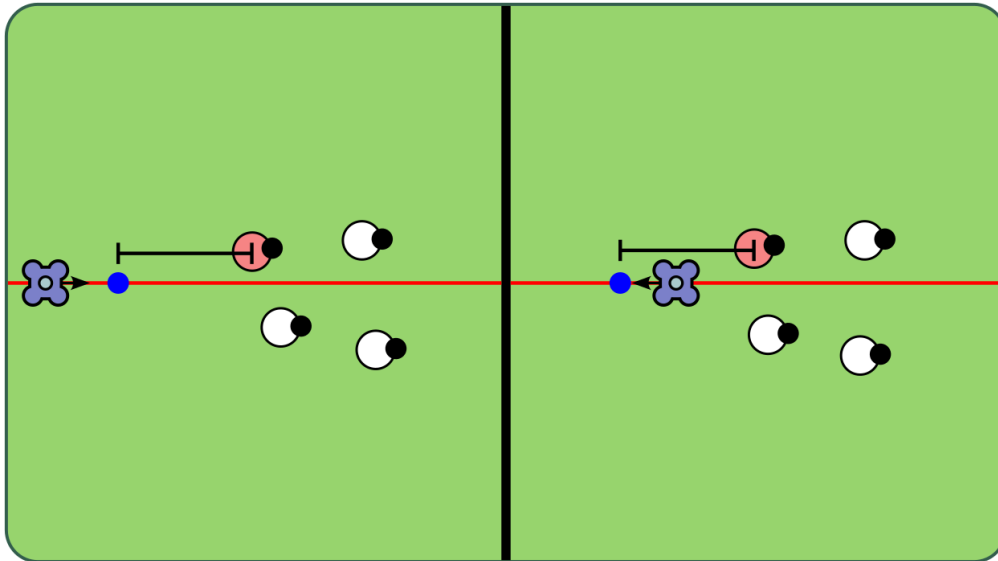


Figur 16: Illustrasjon av hva som skjer med en forskjøvet rute i en krapp sving. Svart rute er den forhåndsdefinerte ruten. Rød rute er bakdronens rute. Gul og blå er styredronenes rute. Blå rute får ikke myk sving.

Følg rute

I det siste steget skal dronene drive sauene til innhegningen ved å følge de genererte rutene. Dronene må alltid være posisjonert slik at sauene beveger seg i riktig retning. Basert på sauenes posisjon, blir det kontinuerlig regnet et punkt på ruten for hver drone som de skal bevege seg mot.

Bakdronen skal forsøke å holde seg på et punkt på ruten, som er en viss lengde, R_{DANGER} , fra den bakerste sauene. Dronen får derfor positivt fortegn på farten hvis den ligger bak dette punktet, og et negativt fortegn dersom den er foran det. Dette skal sørge for at dronen ikke beveger seg for nærme saueflokkene, men samtidig er nært nok til å ha innflytelse på dem.

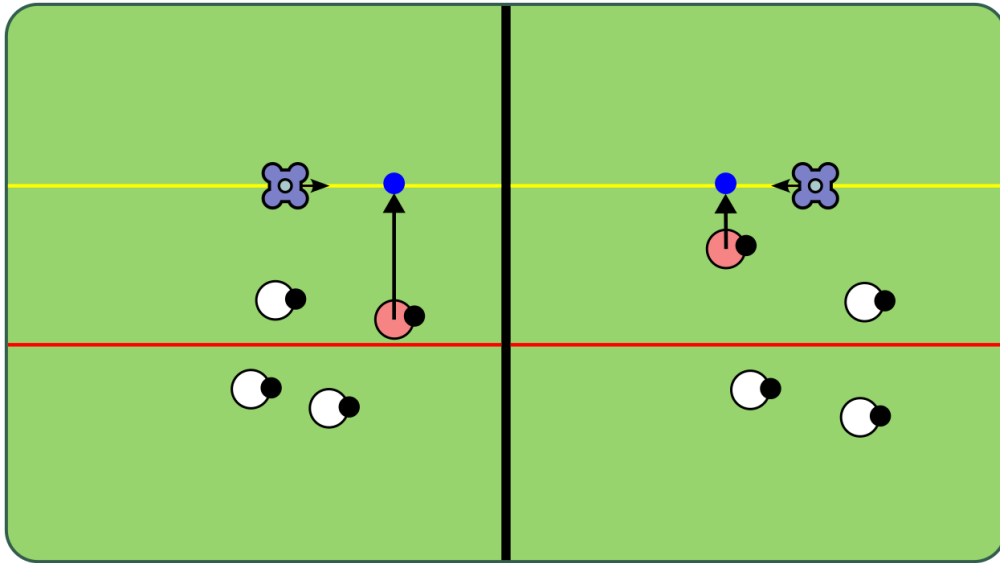


Figur 17: Illustrasjon som viser to situasjoner for hvilket punkt på ruten bakdronen skal bevege seg mot. Dette punktet ligger en lengde bak den bakerste sauen.

Styredronene skal bevege seg mot et punkt på ruten deres som blir regnet ut av en sau sin nærmeste posisjon på ruten. Denne sauen er funnet ved å iterere igjennom flokken og bruke en funksjon til å velge sau. For hver sau gir funksjonen en verdi, ved å bruke a , avstanden sauen har flyttet seg langs bakdronens sin rute, n , som er lengden på normalen mellom sauen og bakdronens rute, og C_{WEIGHT} , som er en konstant som bestemmer hvilke sauer som skal vektlegges mest, sauen på siden eller sauen fremme. C_{WEIGHT} er brukt slik at man kan justere vektleggingen av sauen. Sauen som får høyest resultat, bestemmer punktet dronen skal bevege seg mot. Funksjonen blir brukt på både dronen på høyre og på venstre side, men når n blir regnet ut får den ulikt fortegn ut fra hvilken side av bakdronens rute sauen befinner seg på.

$$s = a * C_{WEIGHT} + n * (1 - w) \quad (13)$$

På samme måte som bakdronen får styredronene negativ eller positiv fart ut ifra hvor de ligger i forhold til punktet de skal bevege seg mot. En illustrasjon av dette er vist i Figur 18.



Figur 18: Illustrasjon som viser to situasjoner for hvilket punkt på ruten en styredrone skal bevege seg mot. Punktet blir bestemt fra posisjonen til den valgte sauene.

Hvis avstanden fra en sau til bakdronens rute er større en R_{DANGER} regnes ikke sauene som en del av flokken lenger og blir fjernet fra den, og blir heller plukket opp senere. Steget er ferdig når alle sauene i flokken er i innhegningen. Algoritmen begynner så lokaliseringsfasen igjen for å lete etter en ny flokk.

3.8 Parametere

Tabell 1 viser de ulike parameterverdiene i den ferdige simulatoren.

Tabell 1: Parameterverdiene i den ferdige simulatoren.

Sauemodell		Dronemodell		Sankealgoritme	
Navn	Verdi	Navn	Verdi	Navn	Verdi
CCOH	2.0	CD_SPEED	10px/s	RSAFE	450px
CSEP	15.0	CFORCE	4.5px/s ²	CANGLE	90°
CALI	5.0	RGOAL	10px	RDANGER	180px
CPATH	4.0			CWEIGHT	0.3
FCESC	21.0			RHERD	500px
FCCOH	2.0			CTURN	300px
FCSEP	0.0			CROT_ANGLE	15°
FCALI	2.0				
FCPATH	2.0				
FOV_{COH}	70°				
FOV_{SEP}	110°				
FOV_{ALI}	130°				
FOV_{PATH}	260°				
RCOH	250px				
RSEP	70px				
RALI	250px				
RESC	350px				
RCON	250px				
RPATH	200px				
CFEAR	0.7				
CCON	0.3				
CS_SPEED	5px/s				
CFORCE	0.3px/s ²				
CSIG	100				

3.9 Testing

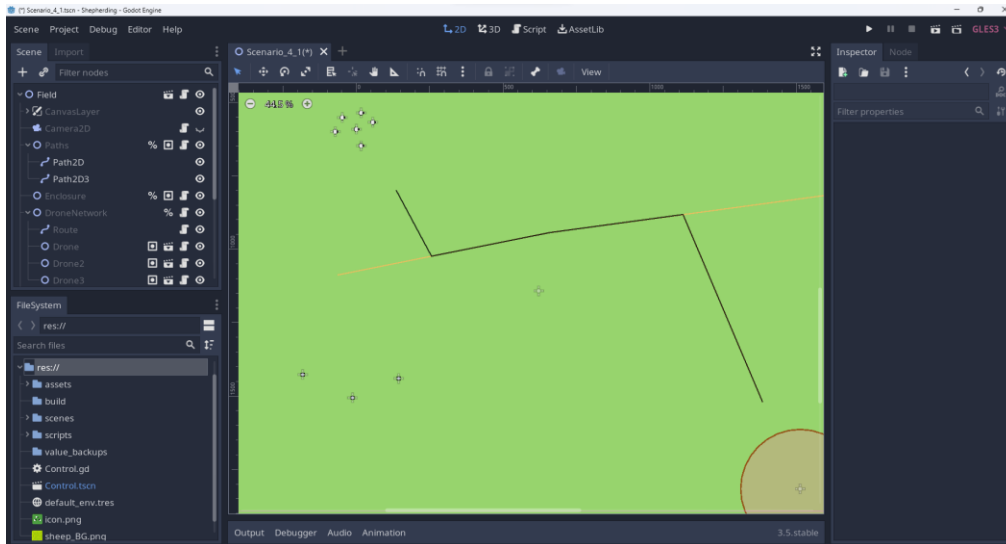
For å teste funksjonaliteten til simulatoren blir det tatt utgangspunkt i fire testscenarier som skal brukes til å evaluere simulatoren. Hvert testscenario er en utvidelse av forrige. Hovedmålet i hvert testscenario er at dronene skal føre sauene til innhegningen. Forskjellen mellom testscenarioene er formen på den forhåndsdefinerte ruten og plassering av stier. Alle testscenarioene blir kjørt med samme parametere i sauemodellen, dronemodellen og sankealgoritmen slik at suksessfaktoren er uavhengig av endring av disse for hvert enkelt testscenario. Testscenarioene blir også kjørt med samme antall sauer. Det ble benyttet seks sau som vi og veileder anser som en god størrelse på en liten flokk. I hvert testscenario blir det kjørt tre ulike varianter av scenarioet.

3.9.1 Forberedelse

Før simulatoren kan kjører et testscenario, må det først gjøres noen forberedelser i form av å sette ut sauer, droner, eventuelle stier og den forhåndsdefinerte ruten.

Det må plasseres ut sauer i simulatoren. Det må være én eller flere sauer i én eller flere flokker. De tre drivedronene må plasseres ut i valgfrie startposisjoner. Det må også plasseres en innhegning der sauene skal sankes. Det er valgfritt om man vil ha stier som plasseres ut som linjestykker i simulatoren eller ikke.

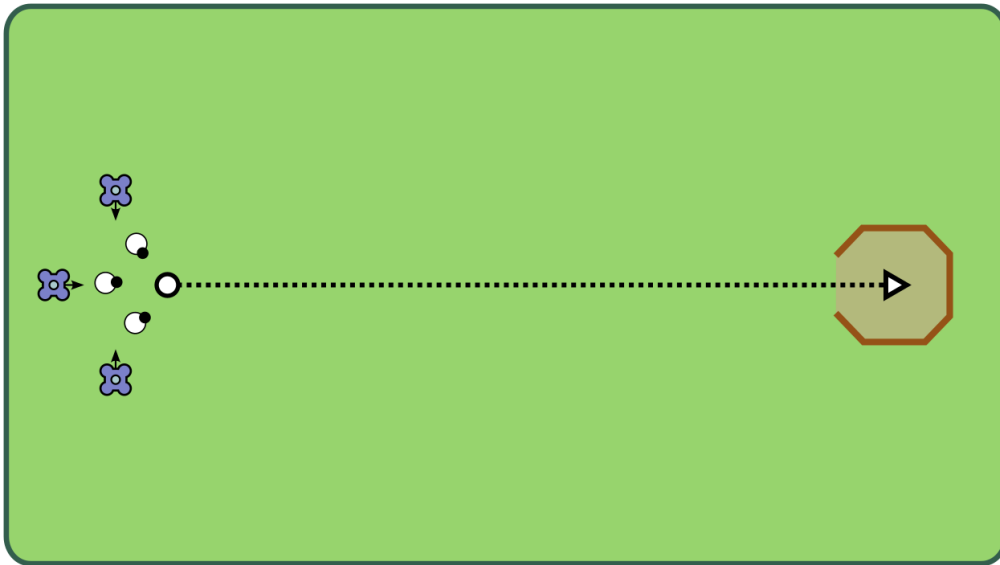
Det kan settes ut en forhåndsdefinert rute som dronene skal drive sauene på. Dette er en rekke linjestykker som kan plasseres hvor som helst, men bør ikke ende opp inne i innhegningen da denne strekning blir lagt til automatisk, og bør heller derfor plasseres nær. Se Figur 19 for et skjermbilde av oppsettet til en test i «*Godot*» før kjøring. De seks prikkene på toppen er sauene og de tre prikkene nede til venstre er dronene. Den svake gule linjen er en sti. Den brune sirkelen er innhegningen, og den svarte streken er den forhåndsdefinerte ruten.



Figur 19: Oppsett av et testscenario i «Godot».

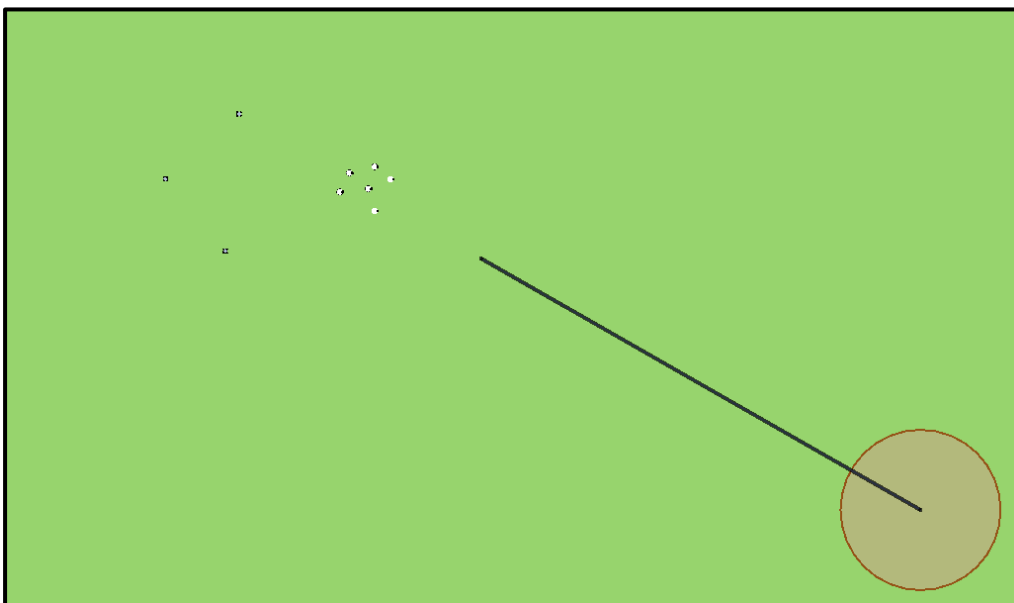
3.9.2 Testscenario 1 – Rett rute

Det første testscenarioet tester om dronene kan føre sauene på en rett rute. Sauene blir frastøtt av dronene og påvirket av hverandre. Konsekvensen av dette kan være at de ikke nødvendigvis beveger seg i en rett linje når de blir drevet. Dronene må derfor klare å holde sauene på ruten. Testscenarioet blir testet ved å sette ute en saueflokk og sette et startpunkt som den forhåndsdefinerte ruten. I kjøretid vil programmet danne en rute fra dette punktet til innhegningen. Flokken plasseres nær dette punktet, men det vil potensielt dannes en liten skrå rute fra saueflokket til startpunktet.

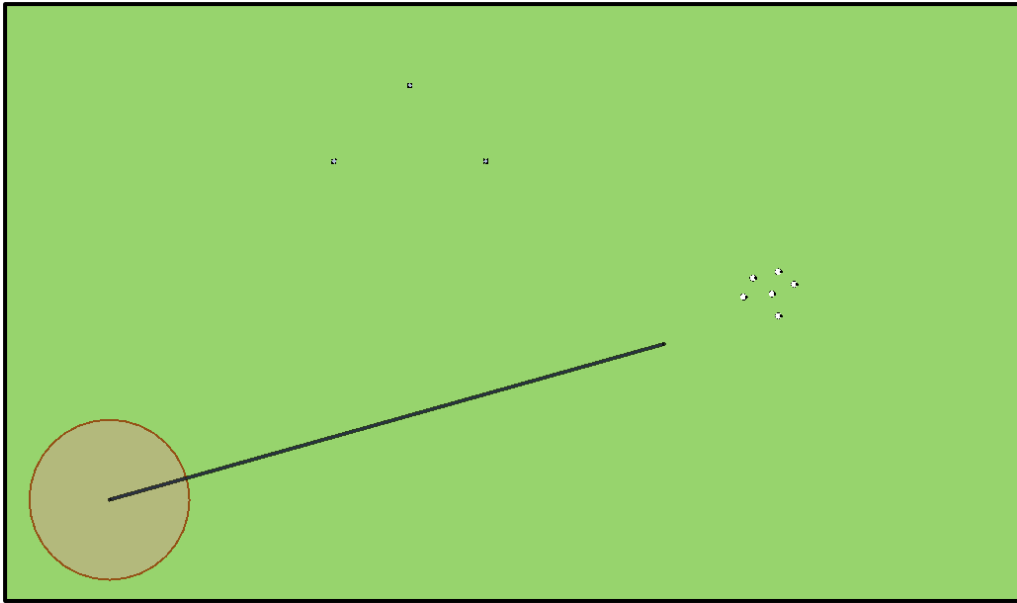


Figur 20: Illustrasjon av testscenario 1. Dronene må drive sauene på en rett linje.

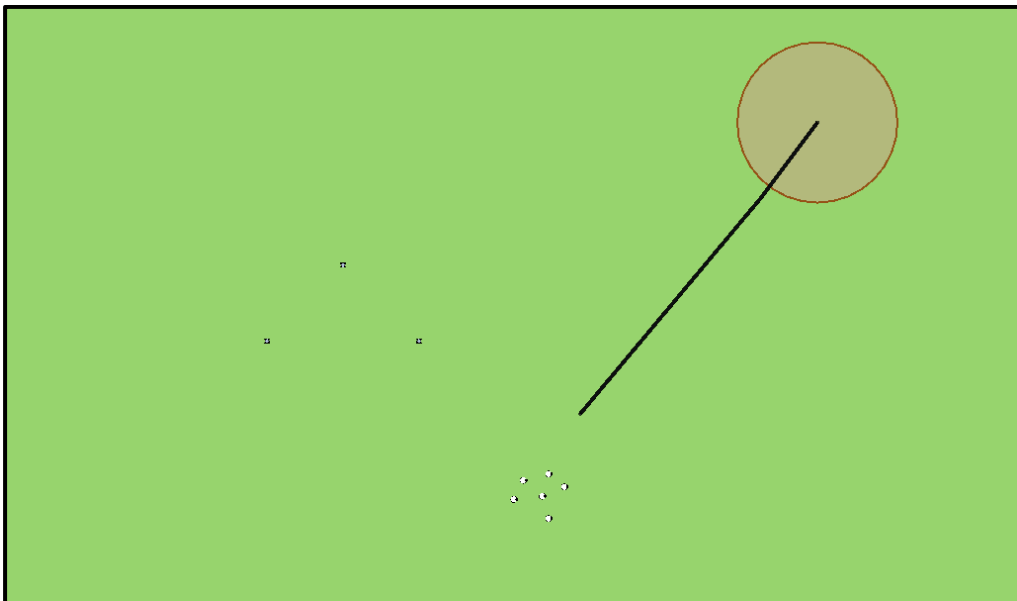
I dette testscenarioet er det lite variasjon ettersom det kun er en rett bane. For å sjekke om verdiene som er satt i simulatoren gjør at dronene kun klarer å drive sauene i en bestemt retning, så har de tre variantene tre ulike innfallsvinkler mot innhegningen. Skjermbildene under viser de tre variantene i kjøretid.



Figur 21: Skjermbilde av simulator som kjører variant 1 av testscenario 1



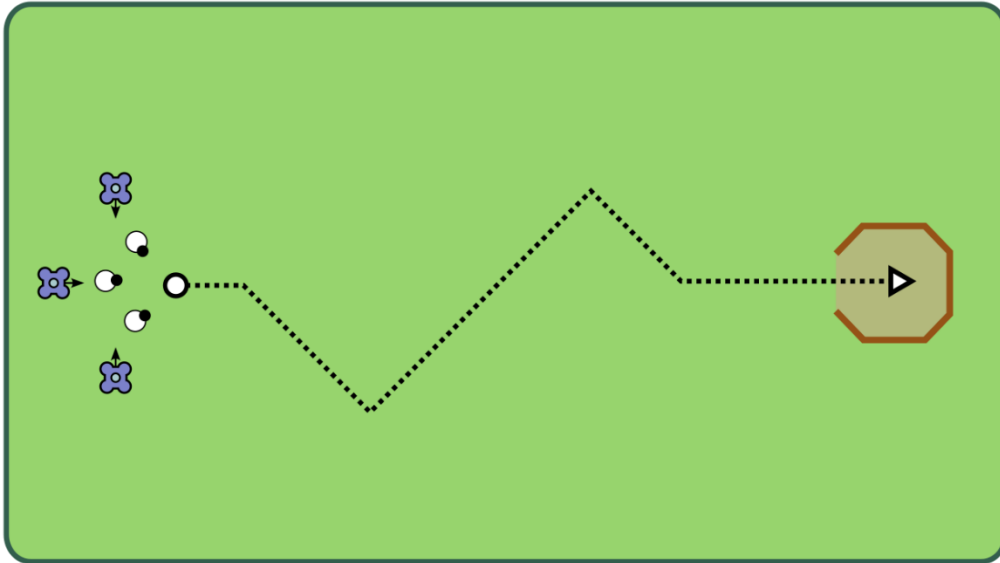
Figur 22: Skjerm bilde av simulator som kjører variant 2 av testscenario 1



Figur 23: Skjerm bilde av simulator som kjører variant 3 av testscenario 1

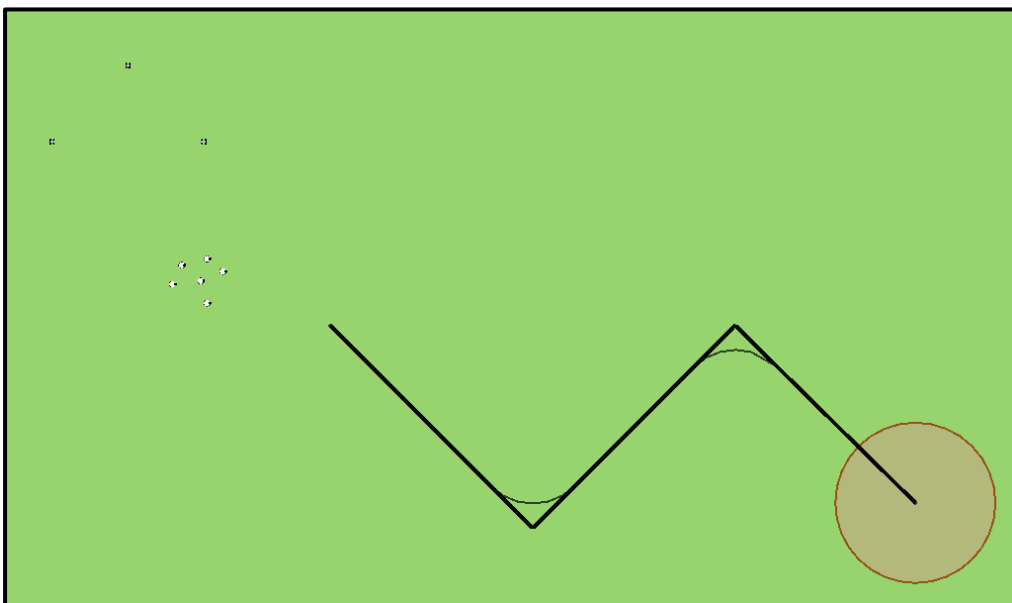
3.9.3 Testscenario 2 – Ujevn rute

Det andre testscenarioet tester om dronene klarer å drive sauene på en ujevn rute. I svingene må dronene plassere seg slik at sauene endrer retning. Testscenarioet testes ved å sette inn flere punkt som lager en ujevn forhåndsdefinert rute.

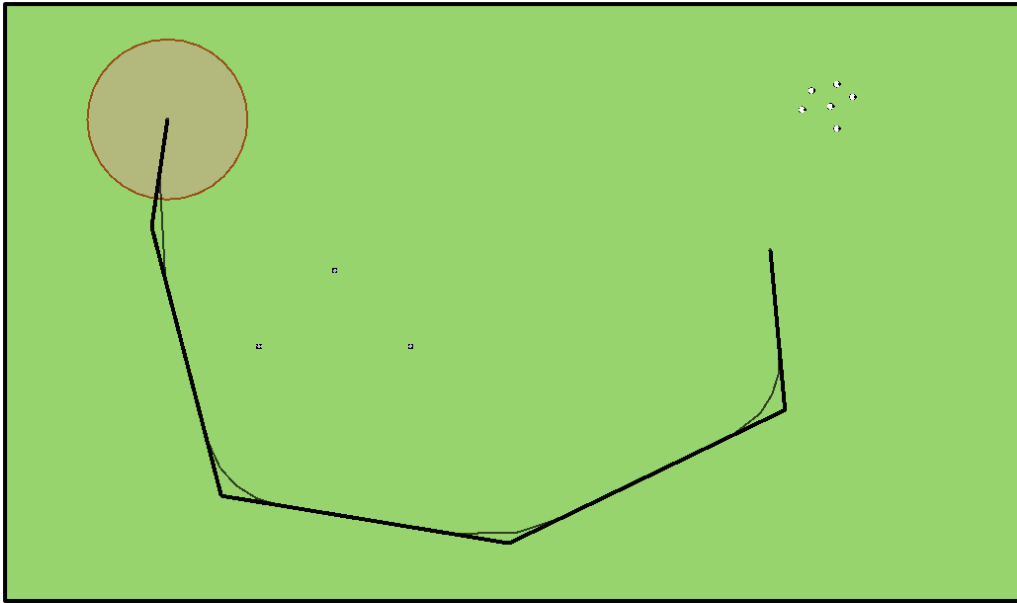


Figur 24: Illustrasjon av testscenario 2. Dronene må drive sauene på en ujevn rute.

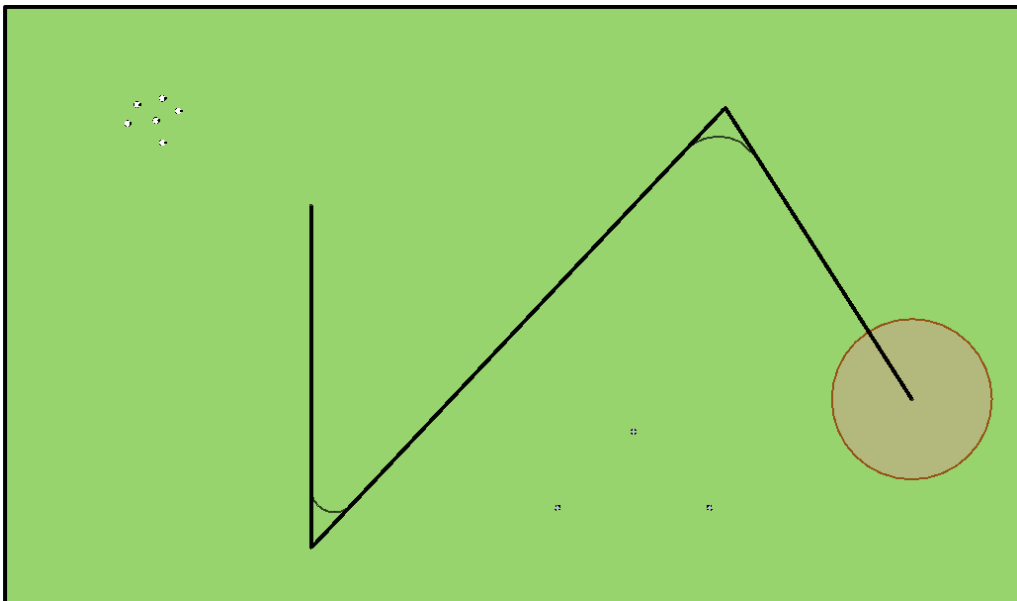
Variantene sjekker om dronene klarer å føre sauene på ulike varianter av en ujevn rute. Noen varianter har krappere svinger enn andre. På skjermbildene i figurene under viser den tjukke streken den forhåndsdefinerte ruten, mens den tynne streken er den myke svingen som algoritmen utregner.



Figur 25: Skjermbilde av simulator som kjører variant 1 av testscenario 2



Figur 26: Skjerm bilde av simulator som kjører variant 2 av testscenario 2



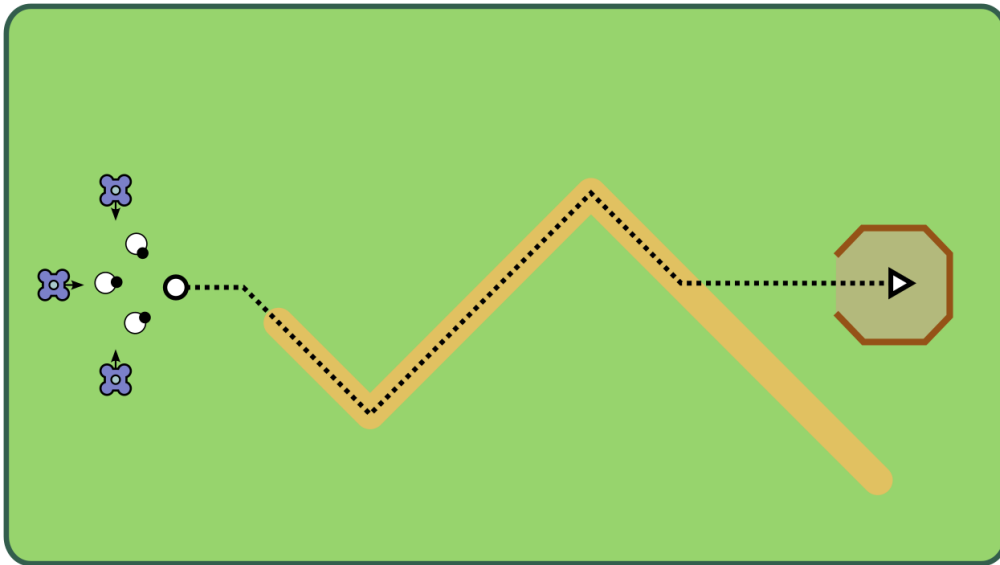
Figur 27: Skjerm bilde av simulator som kjører variant 3 av testscenario 2

3.9.4 Testscenario 3 – Følg og forlat sti

Det tredje testscenarioet tester om dronene klarer å føre sauene på en sti.

Sauene i simulatoren tiltrekkes av stier, men i flere tilfeller er det nødvendig å føre sauene av stien. Testscenarioet tester derfor også om dronene klarer dette.

Testscenarioet utvides ved å legge til stier som sauene må av og på.

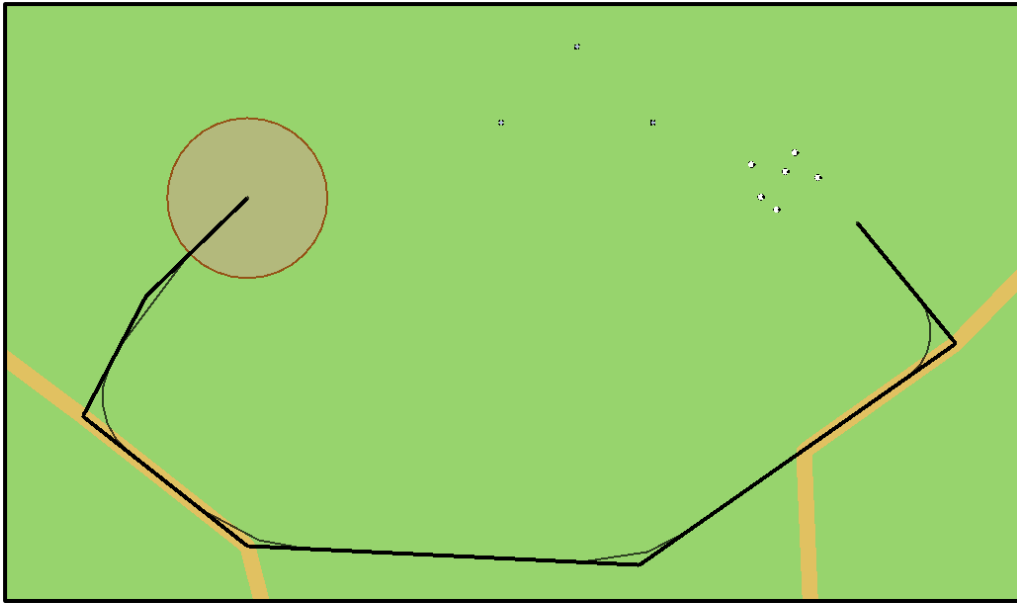


Figur 28: Illustrasjon av testscenario 3. Dronene må drive sauene på og av stien.

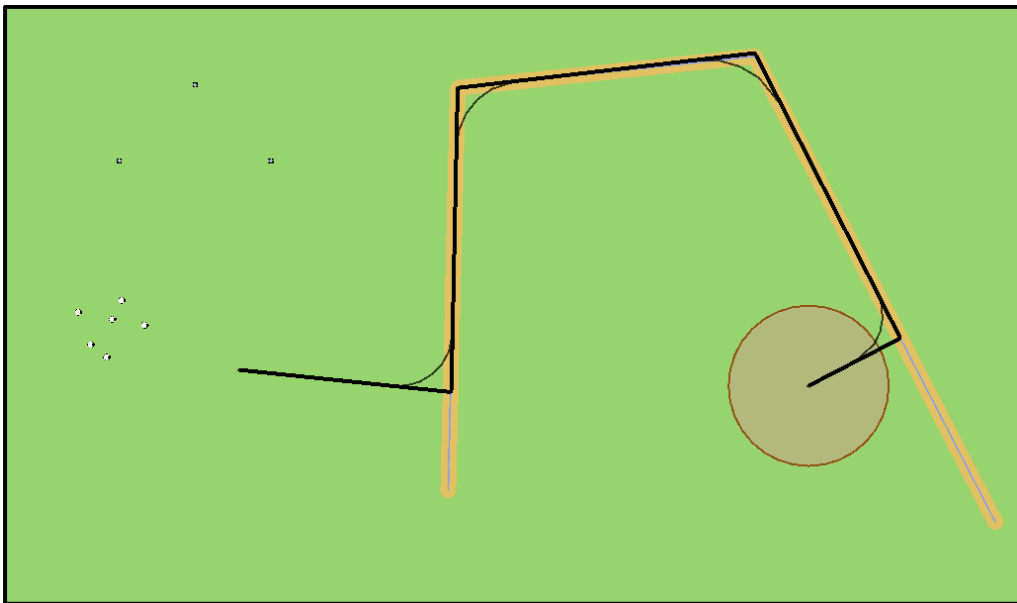
Den første varianten tester om dronene klarer å føre sauene på stien og av stien. Den andre varianten tester om dronene klarer å føre sauene på og av en sti for så å gå på og av en annen sti. Den siste varianten tester om dronene klarer å føre sauene inn i innhegningen hvis den er nær en sti.



Figur 29: Skjerm bilde av simulator som kjører variant 1 av testscenario 3



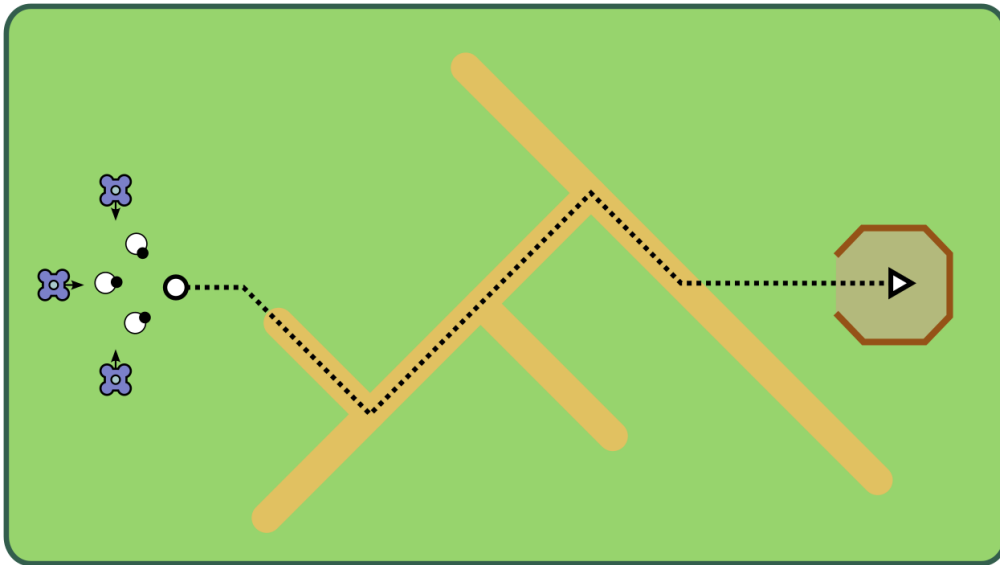
Figur 30: Skjerm bilde av simulator som kjører variant 2 av testscenario 3



Figur 31: Skjerm bilde av simulator som kjører variant 3 av testscenario 3

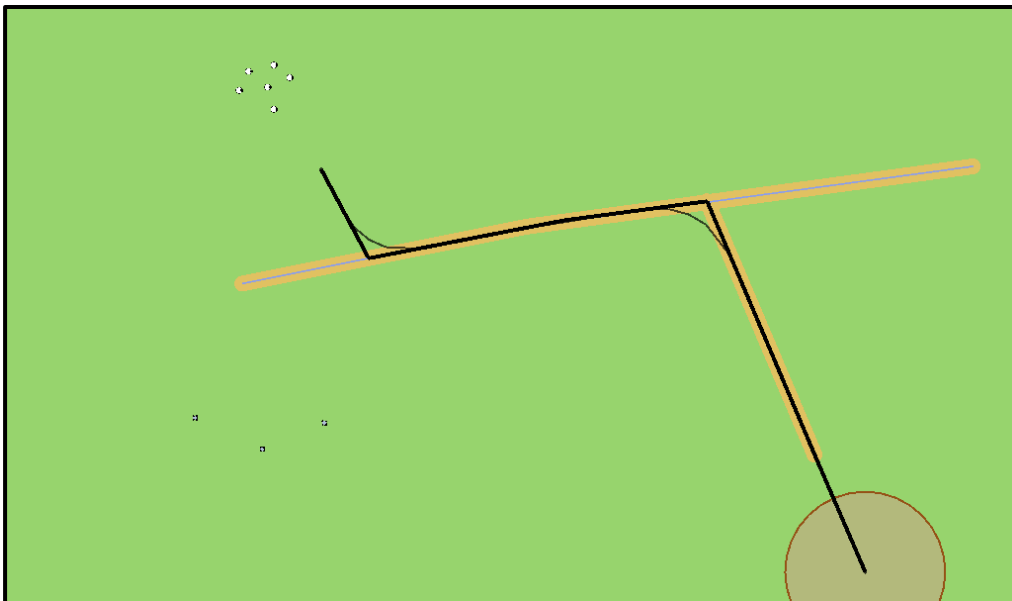
3.9.5 Testscenario 4 – Følg riktig sti

Det fjerde testscenarioet tester om dronen klarer å holde sauene på riktig sti i de tilfellene der en sti forgreiner seg i flere stier. Sauene forsøker å holde seg på stien, og det kan hende at dronene får problemer med å føre dem på riktig sti. Testscenarioet utføres ved å la stier forgreiner seg.

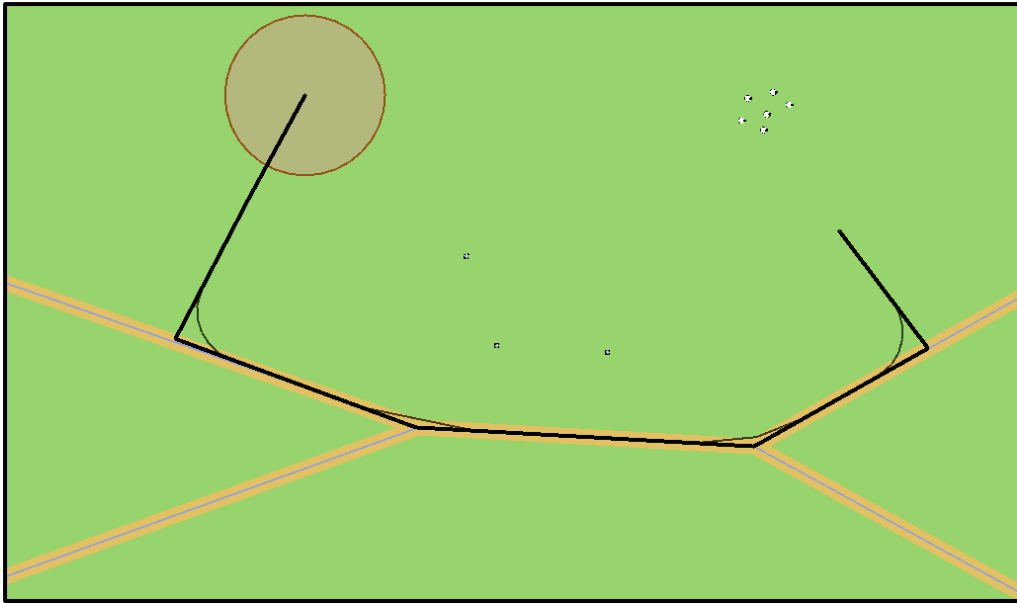


Figur 32: Illustrasjon av testscenario 4. Dronene må drive sauene på riktig sti.

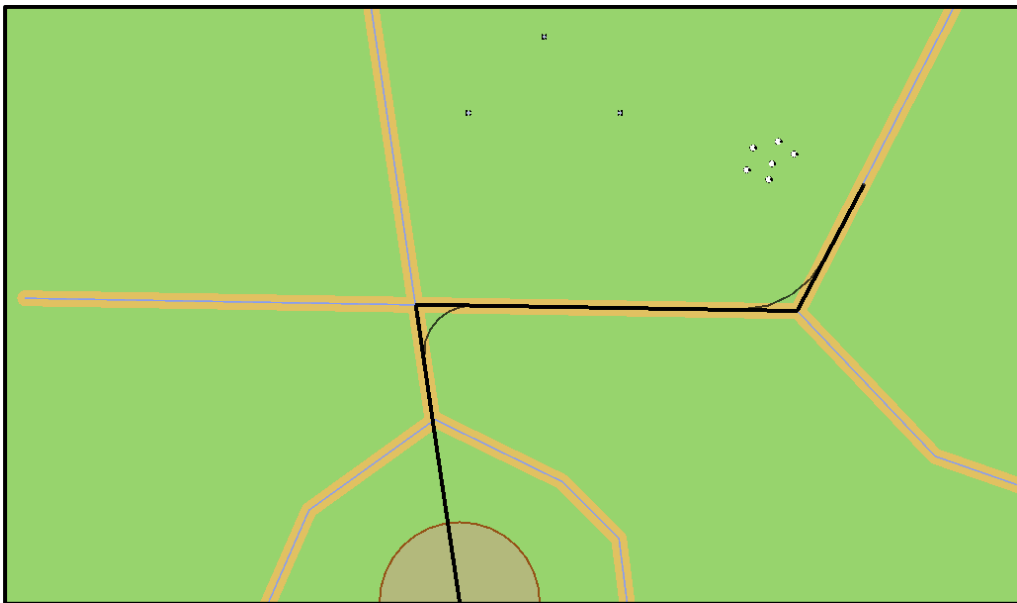
Alle tre varianter tester ulike forgreininger, for å sjekke om dronene klarer å unngå at sauene blir drevet på en annen sti.



Figur 33: Skjerm bilde av simulator som kjører variant 1 av testscenario 4



Figur 34: Skjerm bilde av simulator som kjører variant 2 av testscenario 4



Figur 35: Skjerm bilde av simulator som kjører variant 3 av testscenario 4

3.9.6 Feilsøking

For å avdekke feil og mangler i simulatoren, er det gjennomført feilsøking. Delvis basert på erfaringer gjennom utviklingen av metoden, så er ytterligere forsøk blitt utført, og vil bli nærmere forklart i kapittel 4.3.

4 Resultat og Diskusjon

4.1 Resultat av tester

Ved testkjøringen av testene beskrevet i kapittel 3.8 klarte dronene å drive sauene til innhegningen uten problemer. Dronene unngår sauene under forberedelse-fasen og når de er i posisjon, driver de sauene på den forhåndsdefinerte ruten helt fram til innhegningen. Se fotnote for en lenke til en video som viser kjøring av testene⁴.

4.2 Evaluering

4.2.1 Evaluering av sauemodellen

En god simulering av en saueflokks atferd er en sentral del for å evaluere om sankealgoritmen vil fungere i virkeligheten. Sauemodellen er vanskelig å evaluere, ettersom det mangler et sammenligningsgrunnlag. Utgangspunktet for oppgaven var å simulere driving av små flokker med sauer. Artiklene nevnt i 2.6 som omhandler simulering av saueatferd har i stor grad hatt fokus på større saueflokker. I flere av disse har sauemodellene blitt evaluert mot data fra artikkelen nevnt i kapittel 2.6, der ble det gjort forsøk på en flokk med 46 sauer [36]. Vi anser ikke denne flokkstørrelsen som et godt nok sammenligningsgrunnlag for å evaluere modellen vår. Vi har heller ikke funnet forskning eller data på liknende prosjekt som kan brukes til å evaluere vår modell med små flokker. Hvis det blir gjort GPS-målinger på mindre flokker med sau i nærvær av droner, vil det være lettere å evaluere modellen. Da har man et grunnlag til å justere parameterne slik at modellen samsvarer med virkelig data. Vi kan derfor bare kvalitativt evaluere modellen. Av den grunn er også parameterne satt ved prøving og feiling, og har derfor ingen vitenskapelig tilnærming. I de fleste tilfellene oppfører sauemodellen seg slik vi ønsket. Sauene holder sammen, unngår å kolliderer, danner en

⁴ <https://youtu.be/rjuoPGi0akk> - Oppe per 03.07.2023

NB: Videoen viser ikke testkjøringen som ble brukt i resultatet og det oppstår noen av problemene som er nevnt i diskusjonsdelen.

linjeformasjon når de drives og blir tiltrukket stiene. Likevel er det en del mangler og forbedringspotensialer som vi blir forklart nærmere i kapittel 4.3.

4.2.2 Evaluering av sankealgoritmen

Når det gjelder evalueringen av sankealgoritmen, har vi mange av de samme problemene som med sauemodellen. Det er en stor mangel på data som kan brukes til å evaluere den mot virkeligheten. Selv om resultatene antyder at den fungerer, så er dette kun innenfor simulatorens rammer. Dette gjør det vanskelig å si hvordan sankealgoritmen hadde fungert i virkeligheten. Ser vi bort fra dette, ser sankealgoritmen lovende ut, ettersom den gjennomførte alle testene med gode resultater. Ved feilsøking for å teste grensene til simulatoren, ble det flere tilfeller der noen eller alle sauene ble dyttet vekk fra ruten. Se kapittel 4.3 for utdypning av dette. I et stort flertall av disse tilfellene klarte dronene å omorganisere seg og hente sauene inn igjen uten problem.

4.2.3 Evaluering av parametervalg

Gjennomgående for simulatoren er at det er veldig mange parametere. I tillegg til flere verdier for «*Boids*», konseptene flukt, redsel og redselsspredning, synsfelt, radier for sauene, så er det også flere parametere for dronene som fart, avstand og posisjonsnøyaktighet. I tillegg har sankealgoritmen flere parametere. Parameterne i seg selv fører til tilnærmet uendelige kombinasjoner. I tillegg har flere av disse parameterne innflytelse på hverandre, slik at endring av én, kan påvirke andre. Det er derfor vanskelig å si noe om hvor bra vi har truffet på verdiene vi endte opp med. Da parameterne ble justert for å lage de forskjellige modellene, var det et stort fokus på at algoritmen skulle klare å drive flokken og at sauene skulle oppføre seg slik vi ønsket. Dette satte et stort preg på valg av parameterne og vises godt i situasjoner hvor sauene skal være tiltrukket stier. Ved tiltrekningsverdien som er satt, har stier minimal påvirkning på sankealgoritmen. Dette resulterte i at ingen ytterligere regler måtte lages for å håndtere situasjoner som f.eks. å drive sauene av en sti. Dette medfører at konseptet om stifølging ikke kunne testes på en tilfredsstillende måte.

4.3 Feil og svakheter

Det er en rekke feil og svakheter med simulatoren, og vi vil nå gå igjennom de viktigste.

4.3.1 Sauer som ikke reagerer på dronene

I noen tilfeller er ikke sauene redde nok for dronene. I enkelte situasjoner der de fremste sauene avviker fra ruten, viser resten av flokken ingen antydning til respons på dronene, selv når disse sauene er rett under en drone. Et eksempel på dette er vist i Figur 36, der en flokk blir drevet mot en annen flokk.

Resultatet her er at den fremste sauen ender utenfor ruten, og det virker som resten av sauene følger denne sauen uten å bli påvirket av dronene. Samtidig som dette virker som en usannsynlig oppførsel, så er det vanskelig å si hvordan sauene hadde reagert i en slik situasjon, da vi ikke har funnet noen forskning eller data som kan indikere hvordan sauene hadde reagert på det.

4.3.2 Bevegelse til sauer i fravær av droner

Modellen har ikke tatt hensyn til sauene når de ikke er i nærhet av dronene. Dette medfører at sauene mer eller mindre står i ro og går rundt i sirkler når dronene ikke er til stede. I utgangspunktet så vi ingen problem med dette, ettersom det viktigste i vårt tilfelle var hvordan sauene skulle oppføre seg under driving. I etterkant ser vi derimot at dette medfører at steget der dronene skal gå i sin startposisjon ikke blir ordentlig testet. Ettersom sauene beveger på seg vil posisjonen under første steg i utføre-fasen stadig oppdatere seg, men nå står sauene mer eller mindre i ro og dronene initierer drivingen uten problem. Feilsøking, der sauene blir gitt en bevegelse, viser at dette kan skape et problem ettersom dronene i tilfeller kan forsøke å starte drivingen mot den forhåndsdefinerte ruten, mens sauene beveger seg i motsatt retning. Dette, kombinert med feilen som er beskrevet i forrige avsnitt, medførte ofte at flere av sauene kommer seg unna og må bli hentet igjen senere.

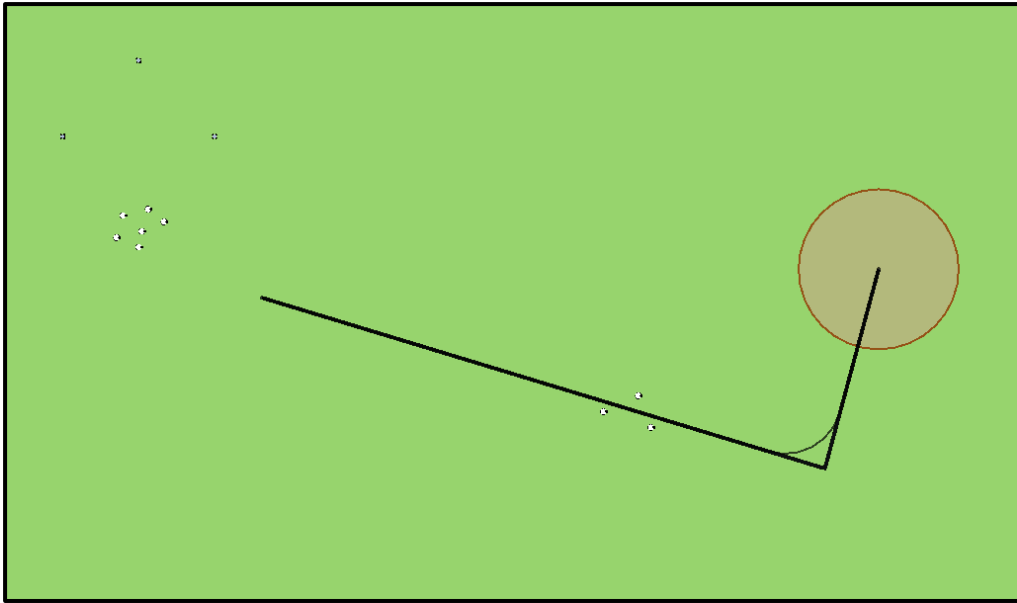
4.3.3 Kalkulering av redselen til sauene

I implementasjonen av redsel ble det gjort slik at en sau kun reagerte på den nærmeste dronen. I ettertid ser vi at det kanskje hadde vært bedre å vekte avstandene til dronene, slik at sauene reagerer på alle dronene istedenfor bare den nærmeste.

4.3.4 Ikke dynamisk sankealgoritme

Et problem med sankealgoritmen er at parameterne som brukes til å generere dronerutene er konstante, og simulasjonsmiljøet har ingen innflytelse på disse. Dette gjør f.eks. at rutene ikke endrer på seg ut ifra hvor mange sau det er eller formen til flokken som skal bli drevet. Dronene som skal bevege seg på sidene for å sørge for at ingen av sauene drives av ruten, holder seg derfor alltid samme avstand fra hovedruten. Dette kan skape problemer for driving av større flokker da flokkens bredde ikke blir tatt hensyn til under genereringen av dronerutene. Dette kan føre til at dronene får en rute som går rett over noen av sauene og splitter flokken.

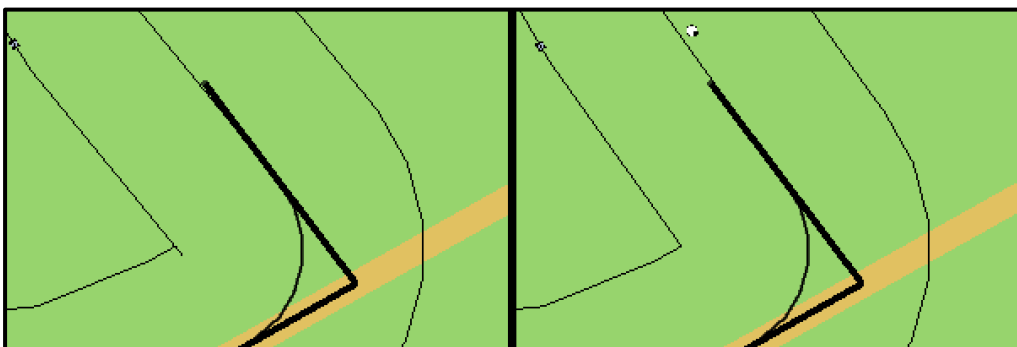
Et annet problem med sankealgoritmen er at den ikke har noen måte å håndtere situasjoner der normal driving blir forhindret. Dette kommer av at hele ruten til dronene blir genererte ved oppstart av utføre-fasen, og sauene som skal drives blir også kun registrert ved oppstarten. Dette kan f.eks. skape problemer i situasjoner hvor flokken beveger seg av ruten dronene driver dem på. Algoritmen har ingen måte å endre rutene til dronene for å unngå dette, og på grunn av problemet beskrevet i delkapittel 4.3.1 mister dronene ofte kontrollen på sauene som gjør at drivingen må startes på nytt. Dette skaper også problemer ved situasjoner der den drevne flokken møter på en annen flokk som vist i Figur 36. Siden sauene kun blir registret ved oppstarten av fasen, blir ikke disse nye sauene håndtert og når dronene skal posisjonere seg ved siden av flokken, blir ikke de nye sauene tatt med i beregningen. Sauene blir påvirket og endrer oppførsel når de møter de nye sauene, mens dronene fortsetter som normalt.



Figur 36: Kjøring av tilfelle der en flokk blir drevet mot en annen.

4.3.5 Uregelmessigheter i simulatorkjøring

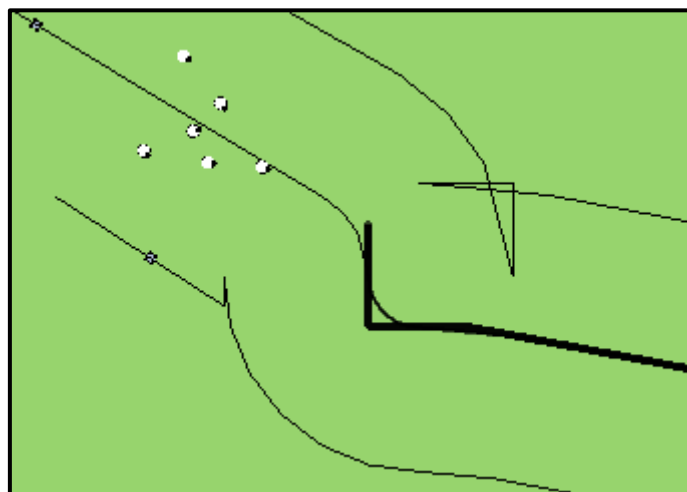
Etter å ha gjennomført testene, ble det oppdaget under andre programkjøringer at det var en uregelmessighet i implementasjonen. Av og til skjer det en feil, som oftest ved første kjøring av programmet, som gjør at rutene til dronene blir feilgenerert, som i enkelte tilfeller fører til at sauene blir dyttet vekk fra ruten. Dronene initierer igjen og klarer å drive sauene til innhegningen likevel i scenarioene som har blitt observert med denne feilen. Testene er ikke kjørt på nytt siden det ble ansett som unødvendig ettersom uregelmessighetene er vurdert som minimale og tilfeldige.



Figur 37: Utklipp av testscenario 4 variant 2. Utklippet til venstre har fått en uregelmessighet som forårsaker en linje som gjør at dronen beveger seg lenger inn enn den skal.

4.3.6 Feil i generering av myke svinger

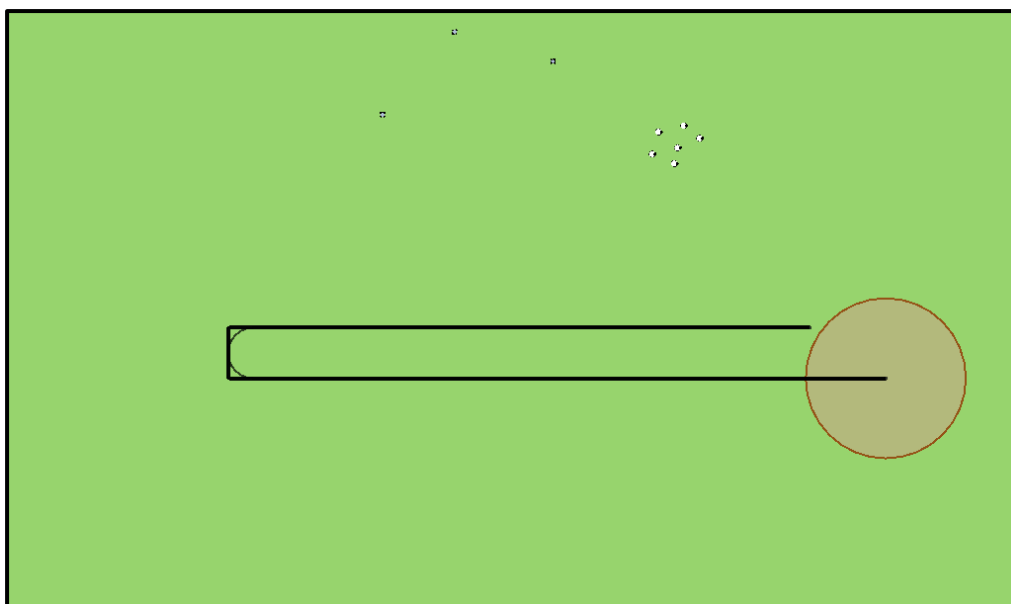
Uregelmessighetene beskrevet i forrige kapittel avdekket en annen feil. I 3.7.4 blir det beskrevet hvordan ruten blir forskjøvet for å generere rutene til styredronene. Det ble oppdaget et problem med at når et linjestykke på ruten er for kort, oppstår det kryssninger i de forskjøvet rutene som ikke blir håndtert i simulatoren. Dette kan føre til at dronene beveger seg for nær sauene eller posisjonere seg feil, noe som gjør at dronene kan mistet kontroll på flokken.



Figur 38: Et kort linjestykke på ruten, fører til kryssninger som forårsaker uønskede ruter.

4.3.7 Feil med diverse forhåndsdefinerte ruter

En viktig del av sankealgoritmen er funksjonen som finner en sau sitt nærmeste punkt på ruten. Dette blir brukt til å regne ut hvilken retning en drone skal bevege seg i langs ruten sin. Blir dette punktet feil, kan det resultere i at en drone kan bevege seg mot feil posisjon som kan gjøre at dronene mister kontroll på flokken. Ved feilsøking av simulatoren, oppdaget vi noen tilfeller som gjør at dette ikke fungerer. I situasjonen vist i Figur 39 ligger to linjestykker i ruten veldig nær hverandre, og ut fra hvilket linjestykke en sau er nærmest, kan dette punktet hoppe store avstander i løpet av kort tid. Det gjør at dronene beveger seg frem og tilbake langs ruten, og de kan dermed miste kontroll på flokken.



Figur 39: Forhåndsdefinert rute med linjestykker som ligger nært hverandre.

4.3.8 Svakheter i innhegningsimplementasjonen

Som beskrevet i 3.2.6 så er innhegning implementert som en sirkel. Når en sau er innenfor denne sirkelen, blir den fjernet. En konsekvens som ble oppdaget ved feilsøking, er at den siste sauen i flokken som skal inn i innhegningen ikke blir påvirket av de andre sauene da de er fjernet fra simulatoren. Derfor får sauen litt unaturlige bevegelser rett før den blir drevet inn. Selv om dette hadde få konsekvenser, kan dette bli unngått hvis man implementerer en annen løsning, der siste sau fortsatt blir påvirket og tiltrukket de andre sauene.

4.3.9 Feil i kalkulering av redselsspredning

Som beskrevet i kapittel 3.5.2, var det usikkerhet knyttet til hvordan redselsspredningen ble kalkulert i bacheloroppgaven vi baserte implementasjonen på. I ettertid, når det gjelder utregningen av redselsspredningen, ser vi at det som kanskje gir mest mening er å bruke totalredselen, ettersom redselen kan spre seg lengre ut enn kun til sauene som er direkte påvirket av dronene. Dette hadde ingen direkte konsekvens i testene, ettersom sauene og dronene er nær hverandre slik at de i praksis oppfører seg likt om man velger å ta totalredselen i utregningen av redselsspredningen eller ei, men vi kan ikke utelukke at det har hatt en minimal effekt.

4.4 Regelbaserte Algoritmer vs. Kunstig intelligens

I kapittel 2 ble det nevnt at tidligere forskning antyder at kunstig intelligens kan brukes til å styre dronene. Et argument for å bruke kunstig intelligens som ble nevnt i kapittel 2, er at det er vist at sauene kan tilvenne seg dronene, og blir mindre redde over tid. Det er vanskelig å si om dette blir et problem for den norske sauene, som blir sanket én gang i året. Et annet argument som ble nevnt, var at det var vanskelig å lage regler som passet for alle situasjoner. Selv om resultatet til oppgaven vår indikerer et fungerende system, så har avgrensningene forenklet problemet. Om man legger til hindringer som trær, elver og gjerder, kan dette øke kompleksiteten betydelig. Sankealgoritmen baserer seg i stor grad på rutene som dronene får tildelt, og spørsmålet om anvendeligheten kan bli utfordret når en drone må bevege seg rundt hindringer for å unngå kollisjon. Når sauene da i tillegg kan tilvenne seg dronene, slik at dronene må flytte seg nærmere eller spiller høyere lyd på høyttalerne, kan systemet bli mer og mer komplekst. Det er derfor viktig å se hvilke egenskaper kunstig intelligens har, som kan benyttes i simulatoren.

Som beskrevet i kapittel 2, er det forsket på å bruke forsterkende læring til å drive droner, og resultatene virker lovende. I artiklene som er nevnt er det kun brukt én drone. Ved flere droner øker dette kompleksiteten, samtidig som det kan føre til bedre resultat. Det kan også oppstå problemer rundt bruk av kunstig intelligens til å drive sauer. En sauemodell og en trent dronemodell kan miste verdien når en endring er nødvendig. Blir det en ny sauemodell, kan det være at en ny dronemodell må trenes opp. Det skaper også mye usikkerhet når uavhengig system arbeider med skjulte verdier. Dette kan blant annet forårsake uønskede hendelser, der droner driver sauer i innsjøer, skremmer dem utfor stup eller at dronene selv blir ødelagt ved å krasje i et tre eller andre hindringer. Det kan tenkes at en kombinasjon av kunstig intelligens og regelbaserte algoritmer kan være en mulighet, ved å bruke forsterkende læring til å drive sauene og samtidig begrense bevegelsene med regelbaserte algoritmer. Algoritmer som hindrer kollisjoner og andre uønskede hendelser, kan danne et komplett system, der algoritmene tar over kontrollen hvis slike uønskede situasjoner oppstår. En slik løsning kan fort bli vanskelig å få til, ettersom det blir flere system som skal samhandle.

5 Konklusjon og videre arbeid

5.1 Konklusjon

Denne masteroppgaven har sett på muligheten for å bruke autonome droner til å sanke sau. Ved å se på tidligere forskning og bruk av droner i landbruket, dannes et bilde av kompleksiteten til oppgaven, som krever arbeid fra flere fagfelt. I tillegg må dyrevelferd og lover tas i betraktning.

Det er utviklet algoritmer og modeller for sauer og droner i tillegg til en regelbasert sankealgoritme. Disse algoritmene og modellene er implementert i et simuleringsmiljø for å evaluere dem ved å kjøre ulike tester.

Simulatoren bestod alle testene som var laget for å evaluere metoden, men mangel på sammenligningsgrunnlag gjør det vanskelig å evaluere om metoden fungerer i virkeligheten. Simulasjonen antyder likevel at muligheten eksisterer, men trenger mye videre arbeid før en løsning kan realiseres. Algoritmene og modellene har mangler og forbedringsmuligheter.

5.2 Videre arbeid

I kapittel 2 ble det gjort klart at det var lite dokumentasjon på bruk av droner til å drive sau i Norge. Dette skaper mye usikkerhet rundt atferden til sau rundt droner, og om det i det hele tatt er mulig å bruke droner til å drive flokker. Det viktigste arbeidet er derfor å innhente og kategorisere store mengder data av norsk sau, både hvordan sauene oppfører seg uten fare, men også oppførselen når sauene blir drevet manuelt ved bruk av droner. Ved hjelp av disse dataene, kan det konkluderes om det er en reell mulighet. Dataene kan også brukes til å utvikle en bedre sauemodell som bruker virkelig data som grunnlag for valg av parameterne.

Når det gjelder sauemodellen som er brukt i oppgaven har den forbedringspotensial. Den største svakheten er parameterne som er valgt for å passe situasjonen til oppgaven, og ikke basert på virkelig atferd. Det bør derfor jobbes for å finne gode parametere på empirisk vis, ved datasamling fra virkelige sauer, men også ved eksperimentering med evaluering av de beste parametere med automatisk prøving og feiling. En annen svakhet med modellen er at sauene står i ro og snurrer rundt når en drone ikke er til stede.

En realistisk modell som simulerer den normale oppførselen til sauer f.eks. under beiting, kan styrke modellen.

Sankealgoritmen har flere områder som kan videreutvikles. For det første så er hovedruten dronene skal følge forhåndsbestemt. Det bør forskes på muligheter for automatisk generering av denne ruten, men det er en stor oppgave, ettersom terreng, hindringer, stier, elver og annet må tas hensyn til når ruten genereres. Dette er likevel en viktig oppgave, ettersom et autonomt system er avhengig av å få korrekte ruter. For det andre så er det ingen funksjoner i metoden som forsøker å innhente sauer med en gang hvis de går av ruten. Det bør sees på metoder for å innhente separerte sauer, ettersom dette kan øke effektiviteten av sankingen.

Denne oppgaven har kun tatt for seg regelbaserte algoritmer. Algoritmene er lite dynamiske, og det ser ut som en krevende oppgave å lage regler som passer alle situasjoner, spesielt når det legges inn hindringer, farer og andre faktorer. Det oppfordres derfor til videre forskning på å bruke kunstig intelligens til kontrollering av dronene. Forskningen som er gjort på dette, har gode resultater, men er enda i oppstartsfasen. I tillegg kan det være interessant å se på kombinasjon av kunstig intelligens og regelbaserte algoritmer.

Det er gjort lite forskning om dyrevelferden og hvordan sauer reagerer på droner, men det lille som er gjort, indikerer at sauer kan være mindre stresset for droner enn gjeterhunder. Det bør derfor forskes videre på dette, da bedre dyrevelferd kan bli en viktig faktor for valg av fremtidige metoder for driving av sau.

6 Referanser

- [1] A. Blix og O. Vangen, «sau – Store norske leksikon,» 11 Januar 2023. [Internett]. Available: <https://snl.no/sau>. [Funnet 19 April 2023].
- [2] A. C. Wangberg, «Fem gode grunner til at vi trenger beitedyr nå og i framtida - Norges Bondelag,» 18 Januar 2021. [Internett]. Available: <https://www.bondelaget.no/nyhetsarkiv/fem-gode-grunner-til-at-vi-trenger-beitedyr-na-og-i-framtida>. [Funnet 19 April 2023].
- [3] Landbruksdirektoratet, «Produksjonstilskudd PT-900,» Landbruksdirektoratet, 12 Juni 2023. [Internett]. Available: https://ldir.statistikdata.no/pt-900_del2_2022_land.html. [Funnet 1 Juli 2023].
- [4] S.-O. Hvasshovd, Interviewee, *Professor*. [Intervju]. Høst 2022 - Vår 2023.
- [5] A. Flatebø, «Verdisatser - Norsk Sau og Geit,» Norsk Sau og Geit, 29 Juli 2018. [Internett]. Available: <https://www.nsg.no/om-nsg/okonomi/verdisatser/>. [Funnet 1 Juli 2023].
- [6] C. Kjelstrup, «Gjeterhund: Valp eller trent hund?,» *Sau og Geit*, vol. 2015, nr. 1, pp. 58-60, 2015.
- [7] Telespor AS, «Telespor AS – Produkter og tjenester for elektronisk overvåkning av husdyr på beite,» Telespor AS, [Internett]. Available: <https://telespor.no/>. [Funnet 3 Juli 2023].
- [8] Smartbjella AS, «Hjem - Smartbjella Sporing,» Smartbjella AS, [Internett]. Available: <https://smartbjella.no/>. [Funnet 3 Juli 2023].
- [9] Nofence AS, «Nofence - Verdens første virtuelle gjerder for beitedyr,» Nofence AS, [Internett]. Available: <https://www.nofence.no/>. [Funnet 3 Juli 2023].

- [10] UNDP, «In Moldova, agricultural drones are used more and more by farmers | United Nations Development Programme,» 12 September 2022. [Internett]. Available: <https://www.undp.org/european-union/stories/moldova-agricultural-drones-are-used-more-and-more-farmers>. [Funnet 5 April 2023].
- [11] K. Buchanan, «Avocado farmers turn to drone tech to measure transpiration, refine irrigation and boost yields - ABC News,» ABC News, 18 Oktober 2022. [Internett]. Available: <https://www.abc.net.au/news/rural/2022-10-19/drones-measure-avocado-farmer-tree-transpiration-orchard-heat/101536322>. [Funnet 6 April 2023].
- [12] S. Flatås og T. Alisubh, «Nå spirer gressfrøene i Norges første dronesådde åker – NRK Trøndelag,» NRK, 8 Juni 2021. [Internett]. Available: <https://www.nrk.no/trondelag/na-spirer-gressfroene-i-norges-forste-dronesadde-aker-1.15523604>. [Funnet 6 April 2023].
- [13] J. Frantzen, «– 10 000 rådyrkalver blir drept av slåmaskiner hvert år - mange kan reddes med droner - UAS Norway,» 18 Juni 2021. [Internett]. Available: <https://www.uasnorway.no/10-000-radyrkalver-blir-drept-av-slamaskiner-hvert-ar-mange-kan-reddes-med-droner/>. [Funnet 6 April 2023].
- [14] Tønsberg kommune, «Førsteslått og rådyrkalver - bruk av drone - Tønsberg kommune,» 30 Mai 2023. [Internett]. Available: <https://www.tonsberg.kommune.no/tjenester/naring-og-landbruk/landbruk/landbruksnyheter/forsteslatt-og-radyrkalver-bruk-av-drone.25054.aspx>. [Funnet 22 Juni 2023].
- [15] Regjeringen.no, «Stadig flere rådyrkillinger reddes ved hjelp av droner - regjeringen.no,» 09 Juni 2022. [Internett]. Available: <https://www.regjeringen.no/no/aktuelt/stadig-flere-radyrkillinger-reddes-ved-hjelp-av-droner/id2917956/>. [Funnet 22 Juni 2023].
- [16] H. Omland, R. Lurås, E. Bjelland og M. G. Svedal, «NOAH ønsker påbud om å bruke varmesøkende droner til å finne rådyrkalver – NRK Vestfold og Telemark – Lokale nyheter, TV og radio,» NRK, 29 Juni 2022. [Internett]. Available: <https://www.nrk.no/vestfoldogtelemark/noah-onsker-pabud-om-a->

- bruke-varmesøkende-droner-til-a-finne-radyrkalver-1.16015261. [Funnet 22 Juni 2023].
- [17] UAS Norway, «Alt du behøver å vite om droner og rådyr - UAS Norway,» [Internett]. Available: <https://www.uasnorway.no/radyr/>. [Funnet 6 April 2023].
- [18] H. O. Torgersen, «Hundretusener deles ut til droner for å redde rådyr - UAS Norway,» 30 Januar 2023. [Internett]. Available: <https://www.uasnorway.no/hundretusener-deles-ut-til-droner-for-a-redde-radyr/>. [Funnet 6 April 2023].
- [19] S. A. Johansen og D. Kessel, «Sauebonde bruker drone for å redde sau fra rovdyr – NRK Innlandet – Lokale nyheter, TV og radio,» NRK, 18 Juni 2019. [Internett]. Available: <https://www.nrk.no/innlandet/sauebonde-bruker-drone-for-a-redde-sau-fra-rovdyr-1.14592101>. [Funnet 5 April 2023].
- [20] N. Hahn, A. Mwakatobe, J. Konuche, N. De Souza, J. Keyyu, M. Goss, A. Chang'a, S. Palminteri, E. Dinerstein og D. Olson, «Unmanned aerial vehicles mitigate human–elephant conflict on the borders of Tanzanian Parks: a case study,» *Oryx*, vol. 51, nr. 3, pp. 513-516, Juli 2017.
- [21] D. Williams, «Droning the drove: Israeli cow-herders turn to flying tech | Reuters,» Reuters, 4 Desember 2020. [Internett]. Available: <https://www.reuters.com/article/us-israel-technology-drone-cattle-change-idUSKBN28E2C8>. [Funnet 3 Juli 2023].
- [22] E. Winje, T.-A. Bjørn, I. Hansen, E. Meisingset, A. Haugen, J. B. Heppelmann, J. N. Myhre og G. Wagner, «Droner som FKT - bruk av droner som forebyggende tiltak i beitenæringen,» *NIBIO Rapport*, vol. 9, nr. 8, p. 48, 10 Februar 2023.
- [23] M. Franco, «Watch this drone shepherd round up its flock on an Irish farm - CNET,» 30 Mars 2015. [Internett]. Available: <https://www.cnet.com/news/privacy/watch-this-drone-shepherd-round-up-a-flock-on-a-farm-in-ireland/>. [Funnet 21 Mars 2023].

- [24] L. Tebbutt, «A slice of Te Kupe: hairy sheep and drone mustering | RNZ,» RNZ, 1 April 2023. [Internett]. Available: <https://www.rnz.co.nz/national/programmes/countrylife/audio/2018884130/a-slice-of-te-kupe-hairy-sheep-and-drone-mustering>. [Funnet 15 April 2023].
- [25] Harper Adams University, «News - Could drones be used to lead sheep? | Harper Adams University,» Harper Adams University, 16 Oktober 2019. [Internett]. Available: <https://www.harper-adams.ac.uk/news/203425/could-drones-be-used-to-lead-sheep>. [Funnet 21 Mars 2023].
- [26] A.-C. Grimstad, «Tilsyn med drone: Rimelig og effektivt,» *Sau og Geit*, nr. 4, pp. 16-20, 2017.
- [27] X. Li, H. Huang, A. Savkin og J. Zhang, «Robotic Herding of Farm Animals Using a Network of Barking Aerial Drones,» *Drones*, vol. 6, nr. 2, Januar 2022.
- [28] A. Hussein, E. Petraki, S. Elsayah og H. A. Abbass, «Autonomous Swarm Shepherding Using Curriculum-Based,» i *Proceedings of the 21st International Conference on Autonomous Agents and Multiagent Systems*, International Foundation for Autonomous Agents and Multiagent Systems, 2022, pp. 633-641.
- [29] J. Zhi og J.-M. Lien, «Learning to Herd Agents Amongst Obstacles: Training Robust Shepherding Behaviors Using Deep Reinforcement Learning,» *IEEE Robotics and Automation Letters*, vol. 6, nr. 2, pp. 4163-4168, April 2021.
- [30] J. Zhi og J.-M. Lien, «Learning to Herd Amongst Obstacles from an Optimized Surrogate,» *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 2954-2961, 2022.
- [31] J. H. Muribø, «Locating Sheep with YOLOv3,» *NTNU*, 2019.
- [32] J. T. Kaarud, M. F. Nordvik og H. R. Paulsen, «Drone-based Detection of Sheep using Thermal and Visual Cameras: A Complete Approach,» *NTNU*, 2020.

- [33] H. Stemshaug, «Impact of Low Resolution IR Images in Drone Based Sheep Detection,» *NTNU*, 2022.
- [34] C. W. Reynolds, «Flocks, Herds, and Schools: A Distributed Behavioral Model,» *Computer Graphics*, vol. 21, nr. 4, pp. 25-34, Juli 1987.
- [35] C. Delgado-Mata, J. Martinez, S. Bee, R. Ruiz-Rodarte og R. Aylett, «On the Use of Virtual Animals with Artificial Fear in Virtual Environments,» *New Gener. Comput.*, vol. 25, pp. 145-169, 2007.
- [36] A. J. King, A. M. Wilson, S. D. Wilshin, J. Lowe, H. Haddadi, S. Hailes og A. J. Morton, «Selfish-herd behaviour of sheep under threat,» *Current Biology*, vol. 22, nr. 14, pp. 561-562, 2012.
- [37] G. Chang og M. Stjernerdal, «Investigating and Modeling the Emergent Flocking Behaviour of Sheep Under Threat with Fear Contagion,» *KTH*, 2019.
- [38] K. Yaxley, K. Joiner og H. Abbass, «Drone approach parameters leading to lower stress sheep flocking and movement: sky shepherding,» *Sci Rep*, vol. 11, nr. 7803, 2021.
- [39] N. Baldock og R. Sibly, «Effects of handling and transportation on the heart rate and behaviour of sheep,» *Applied Animal Behaviour Science*, vol. 28, nr. 1-2, pp. 15-39, 1990.
- [40] I. Midtveit, «Dyrevelferd og kjøtkvalitet: - Effekten av lang sanke- og skiljetid,» *Sau og Geit*, nr. 3, pp. 62-63, 2009.
- [41] Luftfartstilsynet, «Forskrift om luftfart med droner i åpen og i spesifikk kategori,» 30 November 2020. [Internett]. Available: <https://luftfartstilsynet.no/aktorer/regelverk/kommende-endringer/2020/forskrift-om-luftfart-med-droner-i-åpen-og-i-spesifikk-kategori/>. [Funnet 19 Juni 2023].
- [42] EASA, «Easy Access Rules for Unmanned Aircraft Systems (Regulations (EU) 2019/947 and 2019/945) - Revision from September 2022 - Available

in pdf, xml, and online format | EASA,» September 2022. [Internett].
Available: <https://www.easa.europa.eu/en/document-library/easy-access-rules/easy-access-rules-unmanned-aircraft-systems-regulations-eu>. [Funnet
19 Juni 2023].

