Implementation of an Agent-Based Model for Devil Facial Tumor Disease in Tasmanian Devils, and Evaluation of Interventions

A thesis submitted to the
College of Graduate and Postdoctoral Studies
in partial fulfillment of the requirements
for the degree of Master of Science
in the Department of Computer Science
University of Saskatchewan
Saskatoon

By
Lea Lamp

Permission to Use

Disclaimer

Reference in this thesis to any specific commercial products, process, or service by trade name, trademark, manufacturer, or otherwise, does not constitute or imply its endorsement, recommendation, or favoring by the University of Saskatchewan. The views and opinions of the author expressed herein do not state or reflect those of the University of Saskatchewan, and shall not be used for advertising or product endorsement purposes.

Requests for permission to copy or to make other uses of materials in this thesis in whole or part should be addressed to:

Head of the Department of Computer Science
176 Thorvaldson Building, 110 Science Place
University of Saskatchewan
Saskatoon, Saskatchewan S7N 5C9 Canada

OR

Dean
College of Graduate and Postdoctoral Studies
University of Saskatchewan
116 Thorvaldson Building, 110 Science Place
Saskatoon, Saskatchewan S7N 5C9 Canada

Abstract

This thesis presents a geographical agent-based model to investigate different interventions that may be used to combat the spread of devil facial tumour disease (DFTD). DFTD is a clonally transmissible cancer that spreads as an allograft through bite wounds between Tasmanian devils [15]. The population of Tasmanian devils has been reduced by up to 90% since the first documented case of DFTD in 1996, and continued spread of DFTD threatens the survival of the species. The agent-based model presented here uses geographic data to simulate the devil maturation and mating, both spread and progress of DFTD, but also external pressures such as road kill, rodenticide, dog attacks, and generally lower survival in urban settings. Capturing these external pressures addresses a critical gap in current research which can highlight the importance of necessary interventions to preserve the species. Multiple interventions were investigated, including translocation of devils from a disease-free external population, translocation of devils from within Tasmania, use of an injection vaccine, and use of an oral bait vaccine. The injection vaccine increased the devil days lived (DDL) from the baseline of 6.81×10^8 to 7.76×10^8 and decreased the mean daily incidence of DFTD from the baseline of 52.43 to 39.27. Similarly, the oral bait vaccine intervention increased the DDL from 6.81×10^8 to 8.34×10^8, and decreased the mean daily incidence rate from 52.43 to 24.91, using the most aggressive distribution campaign. This oral bait vaccine campaign resulted in eradication of DFTD in the model. As the injection vaccine assumes an intensive trapping effort across the island, which can be very resource intensive, the more promising intervention is the oral bait vaccine due to its significantly lower resource investment and potential for disease eradication.

Acknowledgements

I would like to thank Dr. Nathaniel Osgood, my supervisor, for his invaluable advice and for the learning opportunities that he creates. Dr. Osgood has a wealth of knowledge that he is keen to share, and I am very lucky that he took me on as one of his students. I am indebted to him for his constant support, both academic and personal, during my studies.

I also want to thank Christine Hillis not only for her work in keeping the lab running smoothly, but also for going above and beyond to make me feel welcome. Her ability to solve any issue that may arise is admirable.

I would also like to thank my partner, Eleanor Yates, for helping me through all my work, however difficult. Her support with writing, research, and editing helped to keep me sane during this endeavour. She also helped me to keep a healthy balance between work and life.

Additionally, I would like to thank Dr. Carmel Witte, Dr. David Pemberton, Dr. Samantha Fox, Dr. Billie Lazenby, and Dr. Andrew Flies, for their work on this project, and for their belief in my abilities to complete this project. The discussions I had with them were a key enabler to the success of this project, and this thesis.

Finally, I would like to thank my parents. My mother, Iris Lamp, was a constant source of encouragement throughout my studies. I am also very greatful to my father Ulrich Lamp, and step-mother Margot Lamp, for their constant support, academic, emotional, and financial. My father, Ulrich, was always ready to help me with substantive discussions about my research, and to guide me to think about problems in a new light.

The support from everyone was integral to the completion of this thesis.

Contents

List of Tables

List of Figures

List of Abbreviations

ABM	Agent-based model
GIS	Geographic Information System
SDZWA	San Diego Zoo Wildlife Alliance
STDP	Save the Tasmanian Devil Program
DPIPWE	Department of Primary Industries, Parks, Water and Environment
IUCN	International Union for Conservation of Nature
DFTD	Devil facial tumor disease
MHC-I	Major histocompatibility complex class I
DDL	Devil Days Lived
LOF	List of Figures
LOT	List of Tables
GUI	Graphical User Interface
CSV	Comma Separated Value
TXT	Text
API	Application Programming Interface
PMCMC	Particle Markov Chain Monte Carlo
2D	two-dimensional
SD	Standard Deviation

1 Introduction

Devil Facial Tumour Disease (DFTD) is a clonally transmissible Schwann-cell cancer found in Tasmanian devils. DFTD was first discovered in the Mt. William area of Tasmania in 1996. Since then, it has spread across the majority of the main island of Tasmania. DFTD has caused an 80% reduction in the total Tasmanian devil population, and up to a 95% reduction in some local populations. Since the 1936 extinction of the Thylacine, the Tasmanian devil has been the largest carnivorous marsupial in the world. Preservation of the species is important as Tasmanian devils, a scavenger and predator, play a critical role in the ecosystem. Additionally, the Tasmanian devil is a local symbol of Tasmania. For many reasons, preservation of the species is of utmost importance.

1.1 DFTD

Devil Facial Tumour Disease is transmitted via an allograft — transplantation of cells or tissue to a genetically different recipient — through bite wounds inflicted during regular social interactions between Tasmanian devils. When devils interact during feeding or mating, they can become aggressive towards other devils. During most interactions, the devils will growl and snap at each other, but not cause physical injuries. During a subset of interactions, however, they will bite each other, mainly on the muzzle, neck, and tail. These bites can lead to puncture wounds which then allow the DFTD cells from the biting devil to enter the body of the bitten devil. These cells then start growing as tumours in the bite wound. A devil can be afflicted by multiple tumours from different interactions. Once the tumours within a devil grow too large, it will lead to death, usually by inhibiting that devil's ability to eat, causing starvation.

There are currently several interventions being used in Tasmania to help preserve the species. One of the most potentially promising interventions is the translocation of devils from an external insurance population, such as that on Maria Island. These devils are captured on Maria Island, administered a vaccine via injection, and then released at pre-determined sites on the main island of Tasmania. These interventions are labour intensive and require the maintenance of large external populations from which to pull devils. Other more novel interventions are currently being studied, including the use of an oral bait vaccine. Oral bait vaccines have the potential to provide similar protection from DFTD to that from an injection vaccine, but are administered orally rather than hypodermically. Similar oral vaccination approaches have been used to successfully combat the rabies virus in North America and Europe. Use of oral bait vaccines would allow for easy dispersal of vaccine across the landscape, leading to more wide-ranging vaccination of the devil population.

1.2 The Potential of Simulation Modeling of DTFD

Evaluating multiple different interventions and their impact on the Tasmanian devil population is highly challenging and in some cases not possible in the real world. Agent-based modeling allows a system to be simulated over time, which enables predictions to be made about the current system, as well as potential impacts of interventions. Agent-based models focus on the interaction of individual situated agents, in this case individual Tasmanian devils. Representing individual devils allows the model to readily capture heterogeneity amongst Tasmanian devils, such as sex, age, and their history of infection. Using such a detailed representation of the system allows for targeted interventions to be examined, which can result in better output and predictive capabilities. Agent-based models allow for a rich representation of disease transmission between agents, by characterizing contact between individual agents, as well as exposure to disease via different pathways, such as via water contamination, depending on the disease being studied. Agent-based models are also typically stochastic, which means that they are inherently random. This stochasticity can be utilized to inform uncertain characteristics, such as how often an animal seeks food per day, or how long a vaccine will provide immunity; it can also aid in interpretation of variability in empirical data. Due to this uncertainty integral to the model, many realizations need to be run, in order to ensure confidence in the results. Agent-based models can also be used to simulate geographic features, such as landscape, vegetation, or points of interest to the agent. In this model, geographic data is used to inform mortality rates which are specific to certain geographic areas, such as due to roadkill in areas of high road density, or threat due to humans and dogs in high housing density areas. Combining the geospatial and temporal features of agent-based models allows a highly detailed description of the system. Once built, these agent-based models can then be used to investigate different interventions, such as current interventions that are in use, or proposed interventions that have not yet been tested. These experiments are termed `in silico` experiments because they are run using computers. `In silico` experiments allow for fast iteration to support learning, and can easily be adapted to new scenarios or settings. The additional benefits of using `in silico` experiments opposed to `in situ` experiments are the drastically lower capital and operational costs and human effort required, and low risk extending from the fact that no actual animals and ecosystems are affected. Using such models, it is possible to simulate different types and combinations of interventions, in order to secure a deeper understanding of the issue at hand. This approach poses no direct risk to the system but can provide invaluable insights, that, if used to inform action, can have a profoundly positive impact.

1.3 Goal

This thesis seeks to use agent-based simulation modeling to investigate the behaviour of Tasmanian devils, the spread of DFTD through the population of devils, and to evaluate potential interventions to counteract the spread of DFTD.

The agent-based model designed, constructed and evaluated here is being used to simulate individual animals to account for difference between sex, age, and geographic factors. Four main intervention types are being investigated: Off-island translocation, on-island translocation, injection vaccine, and oral bait vaccine. These interventions have been selected with the help of the research team in Tasmania, notably, Dr. David Pemberton (Tasmania Parks and Wildlife Service), Dr. Samantha Fox (Tasmania Department of Primary Industries, Parks, Water and Environment), and Dr. Billie Lazenby (Tasmania Department of Primary Industries, Parks, Water and Environment). Such stakeholders and scientist Dr. Carmel Witte of the San Diego Zoo Wildlife Alliance provided feedback on the scope, design and results from this model.

Using agent-based modelling, geographic data, and additional mortality risk factors, the thesis investigates the potential for interventions to increase the Tasmanian devil population without increasing the incidence rate of DFTD.

1.4 Contributions

The main contributions of this thesis are as follows:

- **Agent-based model simulating Tasmanian devils affected by Devil Facial Tumour Disease in a geographic setting**

  I implemented an agent-based model to simulate Tasmanian devils in their natural setting of Tasmania, and the impact of DFTD on the population. This model was built using geographic data to inform further mortality rates that impact the Tasmanian devil population, such as roadkill, dog attacks, or poisoning.

- **Developing a geographic grid system to incorporate geographic data into an agent-based model while improving runtime performance**

  The geographic system that was implemented utilized a two dimensional grid of squares to encode the geographic data in an efficient manner. This grid was then used to place all the devil agents in a geographic setting, and inform their movement and contact. Accessing geographic data and searching for contacts in the devil population was both possible in constant time, which allowed for large improvements in runtime performance.

- **Investigation and evaluation of an oral bait vaccine as an intervention to combat and potentially eradicate DFTD**

  A novel intervention that was investigated using this model is the use of an oral bait vaccine. Oral bait vaccines are distributed in the landscape to be taken up by Tasmanian devils naturally, as opposed to through labour-intensive injection of individual animals. Oral bait vaccines are currently in use in North America and Europe to combat Rabies, and such a vaccine is currently being developed to target

DFTD. Characterizing this intervention using this model allowed for the simulation of multiple oral bait vaccine distribution techniques and has the potential to inform potential strategies. In the best distribution configuration, the model predicts that eradication of DFTD is possible using such an oral bait vaccine.

- **Performance optimization**

  Utilizing bit encoding to store agent data during runtime allows for high density storage of data, which then also allowed for flexible on-demand plotting of model output during runtime. This on-demand plotting and data storage has not previously been used with DFTD models, and is a novel way to improve the Stakeholder user experience.

- **Implementing a graphical user interface to control model setup and execution**

  To allow for stakeholder interaction with the model, a GUI was implemented to support the user in performing parameter manipulation before model execution, and then for on-demand plotting of model output during runtime. The implementation of a GUI for an agent-based model is not a novel contribution, but is valuable to stakeholders on account of facilitating interaction with the model.

- **Graphing geographical data produced by the model during runtime**

  To utilize the geographic data produced during model execution and to visualize geographic patterns, a Julia program was implemented which receives data from the model. This data is then processed and individual frames created for each time interval. At the end of model execution, all individual frames are then combined into an animation to show geographic behaviour over time. Multiple different animations are created, each showing a different output from the data stream. For example, these animations show number of infectious devils per cell at a given time, or the amount of oral bait that is present in a cell to monitor intervention rollout and bait decay.

1.5 Thesis Organization

The remaining chapters of this thesis are structured as follows. Chapter 2 provides background on Tasmanian Devil biology, behaviour, and reproduction. Chapter 2 also includes background about DFTD, as well as basic information on agent-based modeling and geographic information systems (GIS). Chapter 3 characterises the agent-based model itself, including a detailed description of the Tasmanian devil agent, the different interventions that were implemented, and all parameters that are used in the model. Chapter 3 also describes how the data are captured in the model during runtime, and how such data are processed and analysed. Chapter 4 reports on findings from model execution. First, the runtime performance for time and memory consumption is presented. The chapter then goes on to characterize outcomes from the different interventions on the Tasmanian devil population. Chapter 5 concludes the thesis by discussing model results, as well as limitations present in the model and potential for future work.

2 Background

2.1 Tasmanian Devils

The Tasmanian devil (*Sarcophilus harrisii* [4]) is a carnivorous marsupial, endemic to Tasmania, Australia. Tasmanian Devils are currently listed as an endangered species on the International Union for Conservation of Nature (IUCN) Red List of Threatened Species [11] with an estimated overall population decline of 80% [29]. Tasmanian devils are the largest living carnivorous marsupials in the world. Evidence has been found to show that Tasmanian devils used to live on mainland Australia, but they died out about 3000 years ago, likely due to the introduction and proliferation of the Dingo, climate change, or human intensification [31, 5, 18]. Human intensification in this context refers to the increase in human activity, usually economic in character, which impacts the Tasmanian devil population. Small insurance populations of Tasmanian devils have been established in multiple locations, most notably on Maria Island, and in New South Wales [32, 3]. Tasmanian devils are mainly scavengers, looking for carcasses of animals, but are also known to hunt smaller prey [2]. Due to their nature of being scavengers, devils are susceptible to being killed by cars and other vehicles as they will feed on roadkill carcasses [2]. Tasmanian devils are also nocturnal, which compounds the issue of roadkill, since they may be hard to spot in the dark due to their mostly black fur, and small build. Tasmanian devils have also been persecuted by humans, especially farmers, due to the belief that they kill lifestock such as sheep [23]. This belief is not true as devils do not hunt animals of this size, but they will feed on the carcasses of deceased lifestock. During the early days of the DFTD outbreak, reduced devil density was noted by farmers due to the fact that carcasses of lifestock would lay in the paddocks instead of being eaten by devils [23]. Tasmanian devils usually live up to six years in the wild, but due to the impacts of DFTD, the mean life span has been shortened to three years [16].

2.1.1 Reproduction

A female Tasmanian devil can have up to three estrous cycles per mating season [14]. The mating season can last from late February to late July, when considering all three cycles [14]. Around 90% female devils get pregnant in the first estrous cycle, and give birth in late March, but some female devils become pregnant later. Female devils will develop a retained fluid roll on their neck in order to endure physical attacks from males [13]. When a female devil is willing to reproduce with a male she will indicate this by being willing to be dragged or escorted to a den by her neck fat, this state has also been described as "Limp-Doggo" (David Pemberton, personal communication, May 21, 2019). Once in the den, the male will copulate with the female several times [13]. The male will also guard the female to prevent her from leaving, and other males from entering the den [27]. This guarding behaviour can last up to 15 days [27], during which the male devil will sometimes use his body to physically block the entrance of the den. As reported by Owen and Pemberton, the female devil will essentially be kept prisoner in the den. This guarding behaviour can even go so far as the male dragging the female to a water source, and back to the den, to keep her under control [23]. A female devil commonly gives birth to between 30 and 40 young, but only up to four will survive because a female devil only has four teats [23]. The number of pouch young follows a bimodal distribution favouring either zero or 4 pouch young [25]. As is common for marsupials, young are born very early, around 21 days of age, and need to finish development in the pouch, where they attach to a teat until they are matured [25]. Pouch young will permanently exit the pouch around 130 days after the median birth date, and will be fully weaned around 278 days [25].

2.1.2 Social Behaviour

Tasmanian devils are usually solitary animals, but they interact when scavenging for food and looking for mating partners. These interactions usually involve vocalization and posturing to ward off another devil, but only some of these interactions result in extensive physical injuries [25]. The most common form of wounds found on devils are puncture wounds to either the muzzle and neck area, or the rump and tail [25]. Even though devils avoid each other in normal circumstances, up to five devils have been shown to feed on a carcass at the same time while tolerating each other [25]. The mating behaviour described above is likely one of the large contributors to the bite wounds incurred by devils as mating involves the most intense interactions.

2.2 DFTD

Devil Facial Tumour Disease (DFTD) is a clonally transmissible schwann-cell cancer, spread through bite wounds as an allograft [22, 24]. DFTD was first detected in 1996 in Mount William National Park [12]. After the discovery, DFTD quickly spread amongst the local population and across large parts of the island, reaching 80% of the main island of Tasmania by 2017 [17]. DFTD cells are a clone from the original female devil which likely started the spread through a genetic mutation in the tumour [6]. Two types of DFTD have been detected which are distinct genetic lines from each other, DFTD1 is the first, detected in 1996, and DFTD2 is the second, first detected in 2014 [26]. While DFTD1 has quickly spread across large parts of Tasmania, DFTD2 has so far been mostly contained to the Channel region of southeast Tasmania. DFTD2 has been traced back to a male devil since it carries a Y chromosome [26]. Since DFTD is transmitted via live cell grafts, it should be detected by the immune system of the host devil; however, this is not the case. DFTD is able to hide, resulting in no immune response being mounted by the infected devil. The mechanism of the immune avoidance is not completely understood yet, but it is likely due to low or no expression of the major histocompatibility complex class I (MHC-I) on the DFTD cells [30]. One way the immune system recognizes foreign cells is through identification of peptides bound by the MHC-I molecule on the cell membrane. The peptides bound by the MHC-I molecule are most commonly produced by degradation of proteins within the cell, and therefore characterizes which proteins are produced within the cell. Once a cell has been identified through the MHC-I molecule as producing foreign proteins, the cell will be destroyed. Due to this low or non-existent expression of MHC-I by the DFTD cells, such cells do not cause an immune reactions and are therefore largely invisible to the immune system of the host devil [7].

2.2.1 Disease Progression

When cancer cells are implanted into the host animal, they start growing into tumours. One devil can, and usually does, have multiple tumours simultaneously, stemming from interactions with different devils. The most common areas are the muzzle, neck and rump, as these are most likely to be targets during aggressive interactions [25]. DFTD is almost always fatal and usually leads to death within one year [9]. Few cases have been documented to show natural regression of DFTD in devils, but these only account for fewer then 20 instances in over 10,000 cases [19]. This natural regression of tumours has lead some to believe that Tasmanian devils are building a natural immunity to DFTD [20]. If Tasmanian devils are developing a natural immunity to DFTD then it does not appear to occur in high enough density to slow or stop the spread of DFTD [7]. The discovery of DFTD2 also shows that other cancers can develop which may contain other immune evasive strategies.

2.2.2 Disease Impact

The impact on the Tasmanian Devil population from DFTD has been extensive. The estimated overall population across the island has exhibited a decline by 80%, and some local populations have declined over 90% [29]. DFTD progression in the population has caused the age structure to collapse, leaving mostly devils up to three years old [16]. An increase in precocial breeding has been observed, but the exact cause is still to be determined, but the leading hypothesis is that higher food availability has led to devils growing faster and reaching the critical mass to reach fertility [16]. The increase in precocial breeding is theorized to have prevented an extinction of Tasmanian devils at this point by boosting the population in low density areas [17]. The lowered density of Tasmanian devils also leaves the population more vulnerable to other population pressures such as roadkill on a local level, or death through wild fires on a large scale. These extra pressures can result in local or regional extinction of Tasmanian devils, increasing the need for human intervention to restore the Tasmanian devil population while decreasing the incidence of DFTD.

2.3 Agent-Based Modeling

Computational models are used in many disciplines, and have many different definitions. In this thesis, I use an agent-based model. An agent-based model uses individual agents, which can interact in an environment with other agents. Agents do not have to represent living mobile individuals such as humans, but can be used to model other entities that have defined behaviour, such as hospitals and factories. Agents can contain smaller sub-models, such as System Dynamics or discrete event simulation models. These models can be used to represent flowcharts or complex processes inside an agent. In this thesis, each Tasmanian devil is represented using one Agent. All devil agents live within the main agent, which is used to contain all model data, functions, and the GIS map. Agents can either live in an abstract space, or in a GIS map if geographic data is used for the model. Due to the complexity of simulating each individual agent, and their entire environment, agent-based models require vast computational resources, and exhibit runtimes that grow with the size of the population of agents. Large agent-based models with a large population (>50,000 agents) or very complex agent behaviour, can easily have a runtime of multiple days unless further optimized. Agent-based models are inherently stochastic because most often rates and probability distributions are used to define an agent and the transition between states. Due to the stochasticity of the model, it needs to be run many times using a Monte-Carlo experiment. This experiment type will run many realizations of the model to capture the different parameter values in the possible parameter space the model operates in. This ensemble of model runs are necessary to account for statistical fluctuations and to be confident in the output that is produced by the model. The more variability that is present in the model output, the more realizations should be run, but usually several hundred to thousand realizations should be used to inform the model output.

2.3.1 Agents

An Agent describes one individual entity in the model. Each agent can contain one or more statecharts, as well as other components such as parameters, variables, and functions. Each statechart is comprised of at least one, but usually multiple states, with each state represents a discrete situation such as being hungry or not. A statechart is the core mechanism of an agent-based model. Statecharts are used to control agent behaviour, trigger certain events, and indicate a status of an agent. Statecharts can consist of three different types of states: simple state, compound state, and final state. A statechart must at least have one simple state to be valid. Compound states group several states together and show a higher level status of the agent. For example the simple states `Juvenile` and `Adult` can be grouped together by a compound state `Alive`. An agent can be in either the `Juvenile` or `Adult` state, but in both cases they will also be contained within the `Alive` state. Final states can only be entered and never exited. In this example, a final state could be `Natural Death`. This final state would be entered by an agent when it has died, and would therefore not be encompassed by the compound state `Alive`. A final state is usually used to remove an agent from the simulation or terminate the execution of a particular statechart. An agent can also connect to one or multiple connections. These connections can either be unidirectional, meaning the source agent can use the connection to contact the destination agent, but not reversed, or it can be bidirectional, in which case both agents can use the connection. Separate connections can be setup for different connection networks, such as general distance based connections, special short duration connections during breeding and more. Using connections allows for quick and easy interaction between agents.

2.3.2 GIS

A Geographic Information System (GIS) is used to capture geographic data. Geographic data includes locations of buildings to calculate housing density, and road lines to calculate road density. Using GIS in an agent-based model provides an easy way of capturing geographic data and make the agent aware of the necessary data in its environment. The use of GIS in agent-based models can enable a richer expressions of results to stakeholders by grounding the model in the physical world. Showing agent behaviour on a map that people can understand — as opposed to an abstract space — can enable more detailed and nuanced discussions as well as a deeper understanding of the model. This visual depiction enables a more in-depth understanding of results and processes which can lead to more substantive discussions and further insights. GIS is a powerful tool for agent-based models, but it needs to be chosen carefully as it can easily lead to large performance penalty if not handled correctly.

3 Methods

3.1 Model

The agent-based model that I developed for this project uses two types of agents. The main agent is used to control all interactions and simulate the environment, and the devil agent simulates each individual devil in the environment. The main agent contains a GIS map to enable the use of geographic data. To circumvent performance issues associated with using the Anylogic GIS map with a large number of agents, I implemented a two-dimensional (2D) grid of cells that is used for all data processing and agent interactions. The GIS map is then only used for visual representations of the devil agents and locations of interest such as trap and release locations.

3.1.1 Collaborative Model Building

The agent-based model was built using a collaborative process with various stakeholders. Initial discussions were used to determine the scope of the model. Determining the scope of the model and which parts of the system to exclude was an important first step, and helped to determine what modeling techniques would be the most effective. During the initial phase of the project, Dr. Witte and I traveled to Tasmania to meet with Dr. Pemberton, Dr. Fox, and Dr. Lazenby. During this five day trip, we were first shown a trapping area near Mount Bethune to learn about how Tasmanian devils are trapped for research, what data is captured, and how it is recorded. This excursion was used to learn about the background of the Tasmanian devil, and its habitat. Following that one day excursion, several meetings were held at the Tasmania Department of Primary Industries, Parks, Water and Environment offices. During these meetings, the example model was presented which was then followed by user requirements gathering to ensure the model built matched the expectations of the stakeholders. These meetings were also used to deepen the modeler's understanding of both the current state of DFTD in Tasmania and nuances of devil biology, so as to further understand their behaviour and necessary components of the model. The GIS component was discussed in great detail to ensure stakeholder understanding of the benefits and drawbacks by using such a system in the model. The GIS map was seen as essential to the model and its value despite its adverse impact on model runtime as well as development time. The meetings in Tasmania were also used to brainstorm an initial set of interventions to investigate with the model. These initial meetings were attended by the following contributors:

- **Dr. Carmel Witte**: Principle investigator, San Diego Zoo Wildlife Alliance

- **Dr. David Pemberton**: Stakeholder, Tasmania Parks and Wildlife Service
- **Dr. Samantha Fox**: Stakeholder, Tasmania Department of Primary Industries, Parks, Water and Environment
- **Dr. Billie Lazenby**: Stakeholder, Tasmania Department of Primary Industries, Parks, Water and Environment

Multiple meetings occurred during the model building phase and were used to further discuss implementation decisions and refine mechanism in the model. At a later date the team was expanded to include **Dr. Andrew Flies** to help with the oral bait vaccine implementation, and parameter review. Meetings employing Zoom teleconferencing software were used to present progress on the model, and to discuss further work. Some exchange and parameter verification occurred via email. Parameter review occurred using a prepared list of parameters with best estimate values already present. These parameters and associated assumed values were then sent to all stakeholders. Feedback from stakeholder included improved parameter estimates and further explanation of systems informing the parameters. All discussions with stakeholders were recorded using electronic notes, and assumptions regarding parameter values were recorded in the parameter Excel file that is utilized in the model.

3.1.2 Parameter Import

To make the model more portable, and easier to use for the stakeholders, I implemented a parameter import system. This system allows for each parameter to be defined in an Excel file prior to being imported into the model on startup. This avoids the needs for stakeholders to open Anylogic to change parameters, or to run the experiment with a GUI when doing so is not needed. Since some parameters are set using probability distributions, these need to be captured in the Excel file properly, and then imported correctly. To allow for the import of distributions, I implemented a custom parameter structure in Anylogic using the Java class element. This class stores parameters using the `ParameterValue` class. This class can store parameters either as an atomic value (`String`, `Boolean`, `Double`), as a Normal Distribution, a Truncated Normal Distribution, or a Triangular Distribution. Each `ParameterValue` is then stored in a `HashTable` using the parameter name as the key. These parameters are used on model startup to initialize all agents in the model. Using this parameter import structure allows for quick iteration of experiments and improves model accessibility for stakeholders by simplifying and streamlining their interactions with the model.

3.1.3 Main Agent

The `Main` agent is the top-level agent of each experiment. It contains the GIS map, and all other elements that are not specific to an individual devil agent. The main agent handles the mating season control for all devils. The mating season is checked each day and controlled by two parameters. The first parameter determines the beginning date of the mating season in days relative to the first day of the current year. The second parameter gives the duration of the mating season. These parameters have been chosen this way opposed to static dates, to allow for the capture of density dependent shifts in mating season start and duration. By checking the mating season daily and sending out messages of the current season to all devils, it allows for devils to reach the age of sexual maturity during a mating season, and then be able to reproduce in the same mating season, without having to wait an extra year. The main agent also controls all interventions, as well as all data output during and after the experiment run.

3.1.4 Tasmanian Devil Agent

The Tasmanian devil agent consists of three statecharts, as well as parameters, variables, and functions. The statecharts define the state of a devil agent with respect to disease, life stage, and reproduction. Each Tasmanian devil agent has a set of parameters which further define the agent and differentiate it from others. Those parameters are `sex`, `birthDate`, and `homeCell`. The `sex` parameter defines if an agent is `Male` or `Female`, which is used in deciding contact between agents and also governs reproductive behaviour. The `birthDate` parameters tracks at which time, relative to the model start, the devil was born. The `birthDate` is then used to calculate the age of the devil and further classify them into an age class. The age classes are defined in table 3.1, and help more easily categorize the devil populations. Since devils are wild animals, an exact age cannot be determined in empirical practice; the use of age brackets is therefore common. The `homeCell` parameter tracks the `ID` of the cell in which the devil lives, and is used to look up geographic data for mortality pressures, as well as to determine contact between devils.

Table 3.1: Age Class definition for Tasmanian Devil agent. Age range is inclusive for the minimum age and exclusive for the maximum age.

Age Class	minimum age (years)	maximum age (years)
Age Class 1	0	1
Age Class 2	1	2
Age Class 3	2	3
Age Class 4	3	4
Age Class 5	4	5
Age Class 6	5	-

Lifestage Statechart

The Lifestage Statechart has four states: Two simple, one complex and one final. Devil agents can only exist in either a simple state or the final state. The `Alive` state is a composite state which indicates that a devil is alive and has not yet died of either DFTD or natural causes. The `Juvenile` state is a simple state and indicates that a devil is juvenile, from zero to a maximum of 1.5 years of age. Devils in the `Juvenile`

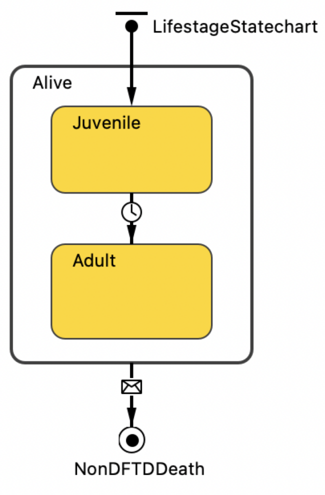


Figure 3.1: Lifestage Statechart

state cannot reproduce. Once a devil becomes sexually mature, they transition to the `Adult` state, where they can reproduce and contact other nearby devils freely. All devils have a monthly chance of death due to three types of non-disease causes: Roadkill, other human threats such as dogs or poison, and other natural causes. These death rates vary by age, and with the geographical location they inhabit, due to changes in housing and road density. If a devil dies of non-DFTD causes, it receives a message, and transitions from the `Alive` state to the `NonDFTDDeath` final state. This final state is used to ensure a devil will be removed from the model gracefully, specifically removing it from the cell it inhabited, removing all connections, and then removing it from the model, which will deregister any outstanding events for this agent.

Disease Statechart

The `DiseaseStatechart` consists of six total states: Five simple states, and one final state. All devil agents initially start in the `Susceptible` state. The `Susceptible` state indicates that it is currently possible for this agent to be infected with DFTD. Once a devil does get infected by another infectious devil, it will transition to the `Exposed` state. In the `Exposed` state, a devil is infected with DFTD, but the tumours have not yet grown large enough to allow for infection of other devils. After a variable time duration between three to 12 months and with a mean of six months, the agent will transition to the `Infectious` state. Once a devil has reached the `Infectious` state, it can infect other devils during contact. Transmission of infection given exposure of a susceptible devil to an infected devil is subject to a transmission probability, because not every interaction between devils leads to bite wounds and transmission of DFTD cells. A small number of devils were recorded to have tumours that were cleared via regression; the post-recovery state of such devils are captured by the `Recovered` state, which a devil can enter from either the `Exposed` or `Infectious` state. A devil in this `Recovered` state is treated as persistently immune to further infection. Recovered devils can return back to susceptibility according to a process of waning immunity, governed by a hazard rate. The

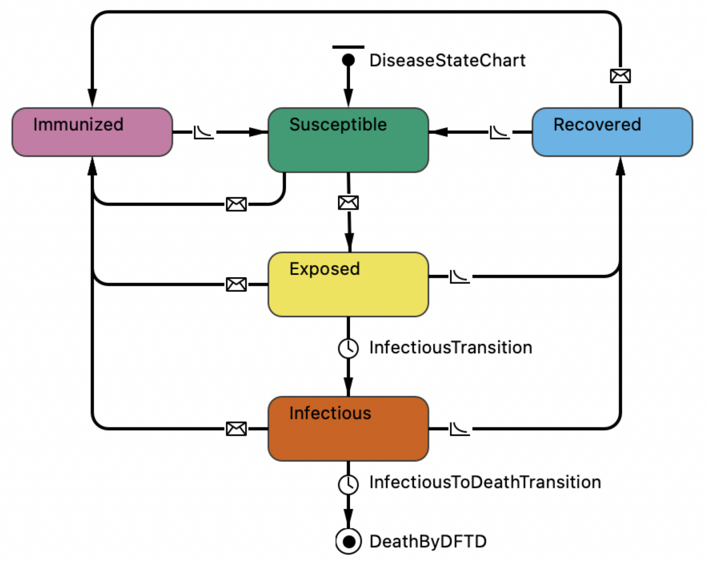


Figure 3.2: Disease Statechart

vaccine interventions are captured through the `Immunized` state. That state can be entered from any other simple state in this statechart, as any devil has the potential to receive the vaccine. Due to the potential of the oral bait vaccine to be used in immunotherapy [7], the transitions from `Exposed` and `Infectious` to the `Immunized` state are captured. The probability for devils moving from either `Susceptible`, `Exposed`, and `Infectious` to `Immunized` can be controlled separately. Once a devil enters the `Immunized` state, they cannot be infected by others and, can also not infect others. Devils return to the `Susceptible` state from the `Immunized` state based on the waning immunity of the vaccine that they were given. Once an agent is in the `Infectious` state for a variable time between three to 12 months, with a mean of six months, they will transition to the `DeathByDFTD` final state. Once the final state is reached, devils will be deregistered from the geographic cell of their residence, and then removed from the model.

Reproduction Statechart

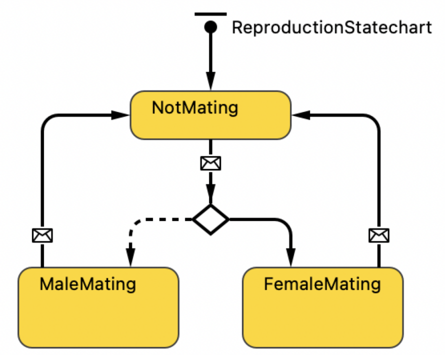


Figure 3.3: Reproduction Statechart

The `Reproduction` statechart is used to characterize the reproductive cycle of devil agents. All devils start in the `NonMating` state. In this state, no reproduction can occur. Once mating season starts, all devils receive a message to trigger the transition to their respective mating states. This transition is guarded to prevent any currently pregnant or juvenile devils from entering the mating states. Adult devils will reach a branch after taken the transition where they will be split by sex. Male devils will enter the `MaleMating` state, and female devils will enter the `FemaleMating` state. Once devils reach the mating states and interact with another devil of the opposite sex, there is a chance for reproduction and pregnancy. When a female devil becomes pregnant, it will immediately take the transition back to the `NonMating` state, in addition, an event will be scheduled for 227 days in the future. When this event triggers, conditional on the fact that the devil has not died, a predetermined number of juvenile devils will be added to the model at the location of the mother. The maximum number of pups is four, since Tasmanian devils only have four teats, and therefore only four pups can survive per litter. The lowest number of pups is zero, as it is possible that the pregnancy is not successful. The probability distribution specifying the count of pups to be born can be controlled using the Parameter spreadsheet, but currently follows a bimodal distribution, with zero and four being the most likely outcomes. During the pregnancy, the female devil may not enter the `FemaleMating` state. All remaining devils will return to the `NonMating` state once the mating season concludes. If a devil reaches sexual maturity, as indicated by transitioning to the `Adult` state, during an active mating season, they will enter the mating state consistent with their sex.

Contact between Tasmanian Devil Agents

Tasmanian devil agents can contact other devil agents located within a given geographical radius. The likelihood of a cell being chosen by a given devil to search for contact devils depends on the distance from the devil to the cell. The closer the cell – including the home cell of the searching devil – the more likely it is to be chosen. This probability follows an exponential distribution with respect to distance. The exponential distribution is used to capture the fact that a devil is much more likely to contact devils within its immediate surrounding, as opposed to devils 20km away. Once a cell for contact is chosen, a suitable devil is selected based on sex and age. If a devil is found, it is returned, and a contact is initiated. If no devil is found, then no contact occurs. The sex and age is determined by the contact rates stored using a Java class in Anylogic, but can be changed using the parameter spreadsheet described in section 3.1.2. Originally, the database functionality in Anylogic was used to store the contact data, but the implementation was altered to instead use a custom Java class due to performance issues with high frequency data requests from the database. Currently all age groups are contacted at the same rate, but contact rates vary by sex and season. Contact rates are given as a mean and SD, and which is then used in a normal distribution to obtain the number of actual contacts per day. These contact rates are used to determine the number of devils that will be contacted in a given day by sex. For each devil being contacted, a new random cell is chosen given the distance from the originating cell. Each contact has the chance of transmitting DFTD if one of the devils is in the `Infectious` state. Each contact can also lead to reproduction if the contact occurs during mating season.

3.1.5 Sensitivity Analysis

To validate the model and investigate the relative impact of parameters on the model output, a sensitivity analysis is critical to perform. One-way sensitivity analysis experiments vary one parameter through a given range, with the output then being examined for relative change against a baseline. Sensitivity analyses differ from Monte Carlo experiments by not only running multiple realizations with the same parameter set, but also by varying the parameter between iterations, while still running multiple replications of each iterations to secure the necessary confidence in the output. In this thesis, the sensitivity analysis will use a baseline parameter ±50%. Using a ±50% change against the baseline value, the relative change of the output can be examined for a large change in the parameter value. The parameters were chosen on the basis of their uncertainty. Certain parameters can be well informed from existing data, or from publications. Other parameters, however, can only be estimated. These estimated parameters are informed by expert opinion and publication, but a sensitivity analysis can show which parameters bear closer examination. For a given change in parameter value, some parameters have a larger impact on the model output than others. For example, a 10% change in the *distanceProbabilityLambda* might show a 20% change in the overall output, or a 2% change. If the change in output is large, then the parameter represents a priority candidate for more careful estimation using calibration or collection of further data points. If the parameter only results in a small change in output, then a larger uncertainty may be acceptable. The parameters listed below were selected to inform best parameter values to use and to investigate their relative impact on the model. Some parameters were not able to be informed by existing data or publication and were included in the sensitivity analysis to inform to what degree of certainty the parameters would have to be estimated for model results to be accurate.

3.1.6 Model Calibration

To calibrate the agent-based model, I used manual calibration by using published values and estimates as starting points. These parameters were then refined by matching the model output to estimated real world data. The biggest issue with wildlife disease is that exact data is typically very difficult to obtain. With the lack of precise and high frequency data, it is often easier to manually calibrate an agent-based model.

First, a baseline without DFTD was calibrated to obtain a stable population over the entire runtime. This ensures that there aren't underlying population issues in the model which will impact the experiment results. Once the disease free baseline was calibrated, then I calibrated the baseline DFTD simulation to match the expected population drop across the island. This also included matching the speed of disease spread over the geographic setting.

Distance Probability Lambda was varied manually from a minimum of 0.05 to 0.8, based on the average

Table 3.2: Parameters for Sensitivity Analysis

Parameter Name	**-50%**	**Baseline**	**+50%**
Distance Probability Lambda	0.15	0.3	0.45
Housing Death Adjustment Factor	0.15	0.3	0.45
Mating Season Duration	20.0	40.0	60.0
Minimum Age For DFTD Infection	0.25	0.5	0.75
Probability Of Moving Away From Home Cell	0.4	0.8	1.2
Road Death Adjustment Factor	0.15	0.3	0.45
Vaccine Probability Of Immunity Per Bait Unit Exposed	0.1	0.2	0.3
Vaccine Probability Of Immunity Per Bait Unit Infectious	0.025	0.05	0.075
Vaccine Probability Of Immunity Per Bait Unit Susceptible	0.25	0.5	0.75
Vaccine Probability Of Immunity Per Injection Exposed	0.1	0.2	0.3
Vaccine Probability Of Immunity Per Injection Infectious	0.025	0.05	0.075
Vaccine Probability Of Immunity Per Injection Susceptible	0.4	0.8	1.2

Table 3.3: Parameters for Model Calibration

Parameter Name	**min**	**max**	**unit**
Distance Probability Lambda	0.05	0.8	-
Housing Death Adjustment Factor	0.01	0.5	-
Road Death Adjustment Factor	0.01	0.5	-
Natural Death Adjustment Factor	0.2	1.5	-
Exposed To Recovered Rate	0.0001	0.1	month^{-1}
Infectious To Recovered Rate	0.0001	0.1	month^{-1}

speed of DFTD spread from $7km * y^{-1}$ to $51km * y^{-1}$[21]. *Housing Death Adjustment Factor* and *Road Death Adjustment Factor* were varied between 0.01 and 0.5, based on their estimated contribution to the Non-DFTD deaths. *Natural Death Adjustment Factor* was varied between 0.2 and 1.5, and was calibrated to produce a stable population over time in conjunction with the other non-DFTD death adjustment factors. *Exposed To Recovered Rate* and *Infectious To Recovered Rate* were varied from a minimum of 0.001 to a maximum of 0.1, being calibrated so as to accord with a published natural recovery rate of 20 recoveries per 10000 cases [19].

3.2 Interventions

3.2.1 Off-Island Translocation

One intervention currently being utilized in Tasmania is the use of off-island translocation. Off-island translocation uses a separate uninfected population of Tasmanian devils that does not live on the main island. One such populations lives on Maria Island. 28 DFTD-free Tasmanian Devils were released in 2013 to build up an insurance population in case of devil extinction on the main island [32]. This population has grown to a size were it is now possible to take healthy devils from Maria Island, and translocate them to an area on the main island. In total 12 individual release locations, across six separate sites, have been utilized too date to release devils (Dr. Samantha Fox, personal communication, 29 October, 2020). All devils chosen for translocation have to fit certain criteria. They all need to be healthy and between two to three years of age. When female devils are translocated, it is preferable to select females with pouch young, to ensure even greater impact and introduction of disease free devils.

3.2.2 On-Island Translocation

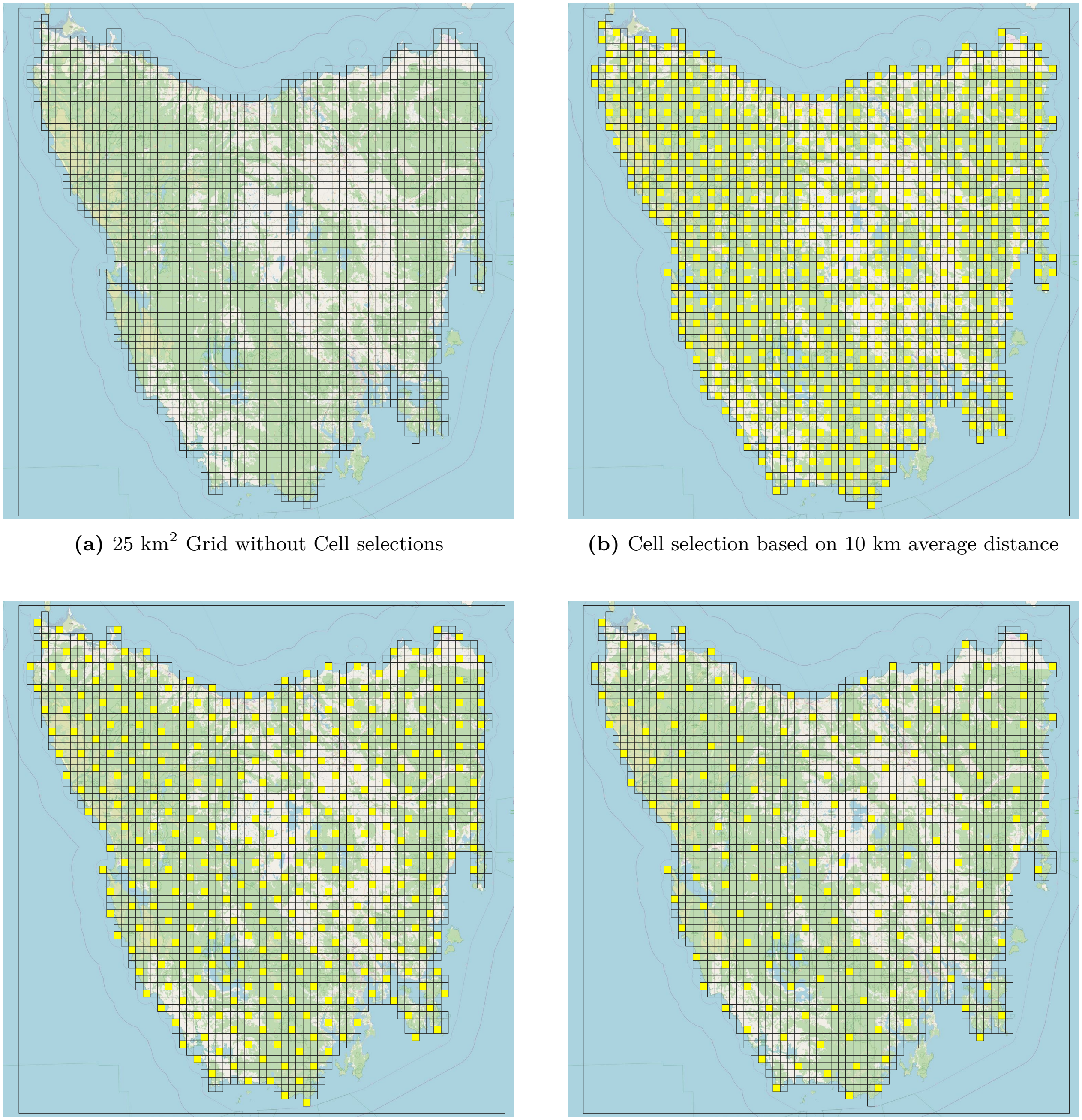
On-Island translocation is similar to off-island translocation, but devils are taken from areas of the main island as opposed to being brought in from external populations. These devils have to fulfil the same criteria as the off-island devils, namely, devils must be healthy, between 2-3 years of age, and if female, it is preferable that they have pouch young. On-island translocation currently uses eight separate trap sites to trap devils for translocation (Dr. Billie Lazenby, personal communication, 16 December, 2020). These sites have been imported into the agent-based model, and are used to trap devils. On-island translocation uses the same 12 release sites as off-island translocation. On-island translocation can be used to combat local extinction and also to increase genetic diversity. As the genetic diversity of Tasmanian devils is very low, this is a very important factor to keep in mind, as increased genetic diversity can help guard against future development of transmissible cancers.

3.2.3 Injection Vaccine

One of the vaccine interventions that is being investigated using this model is the use of an injection based vaccine. An injection vaccine can either be given to devils that are being translocated, or traps can be setup specifically for the purpose of vaccinating devils. An injection vaccine would ideally be limited to a single injection, as recapture of specific animals can be challenging. Injection vaccines are more easily accepted by the general public, but they require vast amounts of resources to administer to the whole population in high enough percentages to result in a decrease in DFTD prevalence and increase the devil population. Trapping devils for vaccination is accompanied by the additional challenge of setting traps in remote locations that might not be easily reached by car. Another issue is that traps need to be checked once daily, and therefore a team of at least two qualified people needs to be stationed nearby and can only cover a small area per day. A large vaccination effort would either require a large number of people and traps, or a very long time period, over which devils vaccinated early on might lose their resulting immunity, decreasing the effectiveness of the vaccine efforts overall.

3.2.4 Oral Vaccine

One novel intervention that this thesis examines is the use of oral bait vaccines. An oral bait vaccine is currently being developed by Andy Flies [7]. These vaccines are packaged as bait to promote animal uptake and to allow for distribution across a large area without the need for trapping individual animals. Bait vaccines require fewer distribution resources as compared to injection vaccines. This includes both financial and personnel requirements. Oral bait vaccines can be distributed during the day and taken up by devils during the night while they are scavenging for food. The distribution possibilities are also more varied as compared to trapping devils, since bait vaccines can be distributed via airplane or helicopter in remote areas. One notable drawback with oral bait vaccine is the decay of the bait. The bait can decay through two means, the first being decay through natural processes such as rain, rotting, or loss of potency. The second decay factor is the uptake of the bait by non-target species such as Quolls, which are another carnivorous marsupial similar to the Tasmanian devil. I investigated three main parameters when examining the use of oral bait vaccine. The first is the density of distribution. Since the agent-based model uses a GIS map to capture geographic data, it allows the placement of bait at different distances and examine the effectiveness of uptake and overall vaccination rate. The second parameter varied the distribution amount. This parameter determines the amount of bait that is dropped at each location. The third parameter is the distribution frequency, which controls the number of months between bait vaccine drops. The effect of three different average distances is shown in Figure 3.4. Using these three parameters allowed me to investigate different distribution strategies and assess their impact on the overall Tasmanian devil population as well as on incidence rate of DFTD.



(a) 25 km^2 Grid without Cell selections

(b) Cell selection based on 10 km average distance

(c) Cell selection based on 15 km average distance

(d) Cell selection based on 20 km average distance

Figure 3.4: Effect of varying average distance between bait drop locations on cell selections made during runtime. (a) shows the empty grid for reference. (b), (c), and (d) show the selected cells in yellow based on the average distance given.

3.3 2D Data Grid

The 2D data grid that I implemented consists of squares that cover the entire main island of Tasmania. The size of these squares can be varied by generating grids of different sizes in QGIS, and then exporting the data including connections. The grid size that was used in the model was 5km by 5km. This grid size was chosen to allow for detailed geographic data to be captured, without excessive computational overhead. All geographic cell data is imported using a CSV file at the beginning of the model. Each cell consists of an ID, a centroid point to define its center, and geographic data such as housing density, road density, and Tasmanian devil density pre-DFTD. The cell objects are defined using a custom Java class in Anylogic (see A.1). A new cell is instantiated for each row in the cell data CSV file. The newly created cells are then added to a hashmap indexed by the cell ID. Use of a hashmap allows for retrieval of the cell object in constant time. The use of a hashmap also allows each devil to store only the ID of the cell they are currently in. Each cell object can store any number of attributes, and it will automatically store all data that is contained within the cell data CSV file that is being loaded in. Each attribute can then be accessed by using the column name of that attribute. This data structure makes the data grid easily expandable to capture more data as necessary. The choice to replace the GIS map with a 2D cell grid to handle all geographic computation allowed for more flexibility in capturing geographic data, and also afforded a large performance improvement making it possible to simulate the entire devil population of the main island.

Coordinate Conversion

Most data that I received from the research group in Tasmania uses the UTM coordinate system. The UTM coordinate system divides the world into a grid, in which each zone is designated by a number and a letter. The number designates the zone, running north to south, and the letter divides the zone into latitude bands. Tasmania is in zone 55G. The coordinates within a given zone is then represented as a Northing and Easting in meters. The conversion from UTM to longitude/latitude can be very complex, and error prone. To avoid any issues with this conversion, I utilize the `CoordinateConversion` library published by IBM [28]. This library allows for easy conversion from UTM to latitude/longitude and back.

Runtime Cell Data Capture

To enable data output per cell over time I added string output functionality to each cell. Using the `summarizeCell` function, each cell was programmed to return a string representation of its ID and statistics regarding devil population in the cell, such as the total number of susceptible, and infectious devils in the cell. This output could then be stored in either a TXT or CSV file for later processing, or sent to the graphing server (see Section 3.10.1).

3.4 Devil Agent Data Capture

A data capturing system was also implemented for devil agent data. At first the Anylogic provided `DataSets` were used, but later replaced with a custom system to allow for highly detailed data capture and on the fly analysis and graphing. Anylogic `DataSets` can be a source of vast memory consumption if used excessively, and does not allow for easy on the fly analysis as required. Due to this, each data output needs to be predetermined and a separate dataset needs to be created and updated. The `DevilDataContainer` implemented here allows high density storage to minimize the memory footprint, while retaining the data resolution necessary to recreate temporal and geospatial data and effects. The addition of the experiment GUI (see Section 3.5) and on demand data aggregation necessitated a custom solution to handling the large amounts of data being generated by the model. Each day data was gathered from each devil agent and stored in the custom Java class `DevilDataContainer` (see A.2). This container stores data for a single time point in the private class `SingleTimeContainer`. The organization of data by timestamp makes it easier to analyse data. Each `SingleTimeContainer` stores all devil data using an array of `short`. Each `short` encodes the devils disease state, ageclass, sex, and lifestage (Juvenile or Adult). The data is compressed into a `short` by using a bitmask for each data point. The bitmask was defined as follows:

Table 3.4: Bitmask used for devil agent data capture

Name	Bit Number	Value
SUSCEPTIBLE	0	1
EXPOSED	1	2
INFECTIOUS	2	4
RECOVERED	3	8
IMMUNIZED	4	16
AGECLASS1	5	32
AGECLASS2	6	64
AGECLASS3	7	128
AGECLASS4	8	256
AGECLASS5	9	512
AGECLASS6	10	1024
MALE	11	2048
FEMALE	12	4096
JUVENILE	13	8192
ADULT	14	16384

Using a bitmask approach to storing data enables highly compressed data storage, as it only needs one `short` for the complete state of each devil agent. This data can then easily be aggregated by a combination of devil states by constructing the appropriate bitmask. For example, to obtain the number of devils that are Exposed, AgeClass2, and Female, the following bitmask can be constructed `001000001000010`, and then a bitwise `AND` is performed. If the result of the `AND` and the mask are the same, then the devil matches all given criteria, and a counter is incremented to represent the sum of devils in those states.

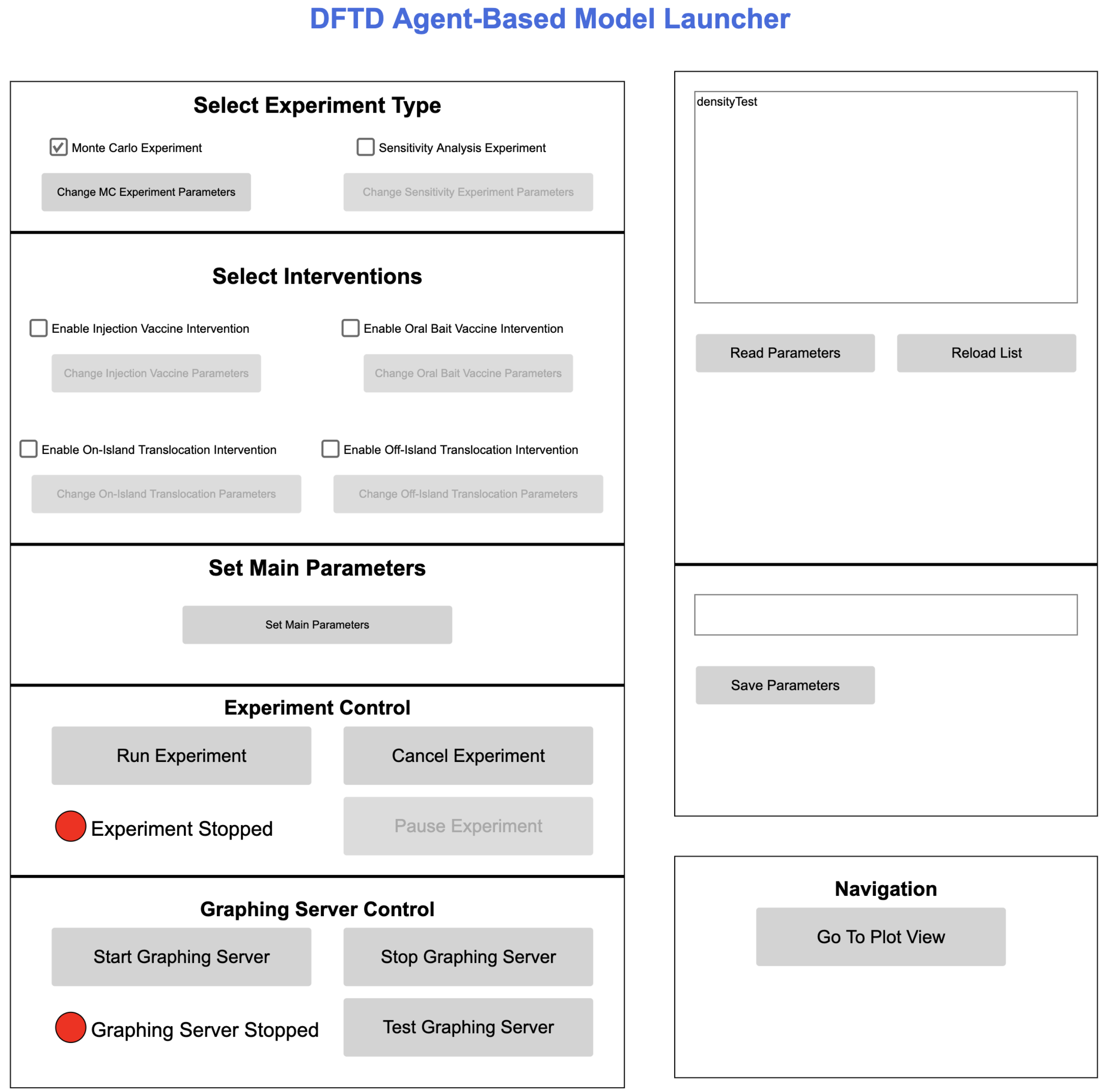
An array of integers is also stored in parallel to the devil data array. The integer array contains the cell ID of each devils home cell. The two arrays are kept synchronous through simultaneous insertion, and by not allowing removal of data. The data stored at the same index in both arrays belongs to the same devil agent at that time point. Consistency between time points is not guaranteed because of the population changes in the model. This data container alone allows for an accurate reconstruction of the model results, not only population dynamics, but also spatial effects, since the cell ID of each devil's home cell is stored.

3.5 Experiment GUI

A custom experiment control GUI was implemented using the Anylogic provided GUI builder. This GUI was necessary because the main experiments were implemented using custom experiments, which have no GUI attached. Implementing the experiment without a GUI allows the experiments to be run on headless servers without the need for a persistent X11 connection. When the GUI is opened a higher level overview is presented. This overview allows the user to decide if they want to run a Monte Carlo experiment, or a Sensitivity Analysis. Next it allows for different intervention types to be enabled or disabled. Each checkbox for intervention and experiment type is accompanied by a button to take the user to a more detailed parameter input view. Each parameter view is unique to the intervention or experiment type that was selected. The high level overview also contains the experiment control section, which allows an experiment to be started, paused, and stopped. It also contains the graphing server control, which starts or stops the Julia graphing server described in section 3.10.1. The high level view also contains controls which allows for saving and loading of parameter preset values. The user can load their previously saved parameters, or parameters from files that were provided to them. When a parameter preset is loaded, and parameters are subsequently changed, a warning will appear which tells the user that the parameters have been changed. This parameter preset loading and saving allows users to easily share a specific parameter set with others in the form of a TXT file. These functionalities make it easy to exchange specific settings without needing to send the entire model. The last button in the high level view is the Navigation section. This section contains a button to take the user to the plot view, where they can view results during runtime.

The Plot view consists of two plots, the first for experiment data, and the second for experiment progress. Six list boxes are presented which allow the user to select a specific iteration, and replication, or even multiple to view at once. The other four list boxes allow the data to be aggregated by disease type, age class, sex, and lifestage. Three checkboxes further control the plot behaviour. The first checkbox enables a moving average which smooths the data over a one year time frame. The second checkbox button shows all data as a percentage of the entire population. The third checkbox sets the Y axis scaling to be fixed at 0 instead of automatic scaling. For example, if the user selects *Susceptible* for disease type, *Age Class 2* for age class, *Female* for sex, and *Juvenile* for lifestage, then the total number of devils that match all those states will be aggregated and displayed.

Plotting data with this level of control allows the user to investigate the data as it is generated and see trends that might not be apparent using more rigid data capture and plotting methods.



☑ Monte Carlo Experiment
□ Sensitivity Analysis Experiment
□ Enable Injection Vaccine Intervention
□ Enable Oral Bait Vaccine Intervention
□ Enable On-Island Translocation Intervention
□ Enable Off-Island Translocation Intervention
● Experiment Stopped
● Graphing Server Stopped
densityTest

Figure 3.5: High Level Model Control GUI

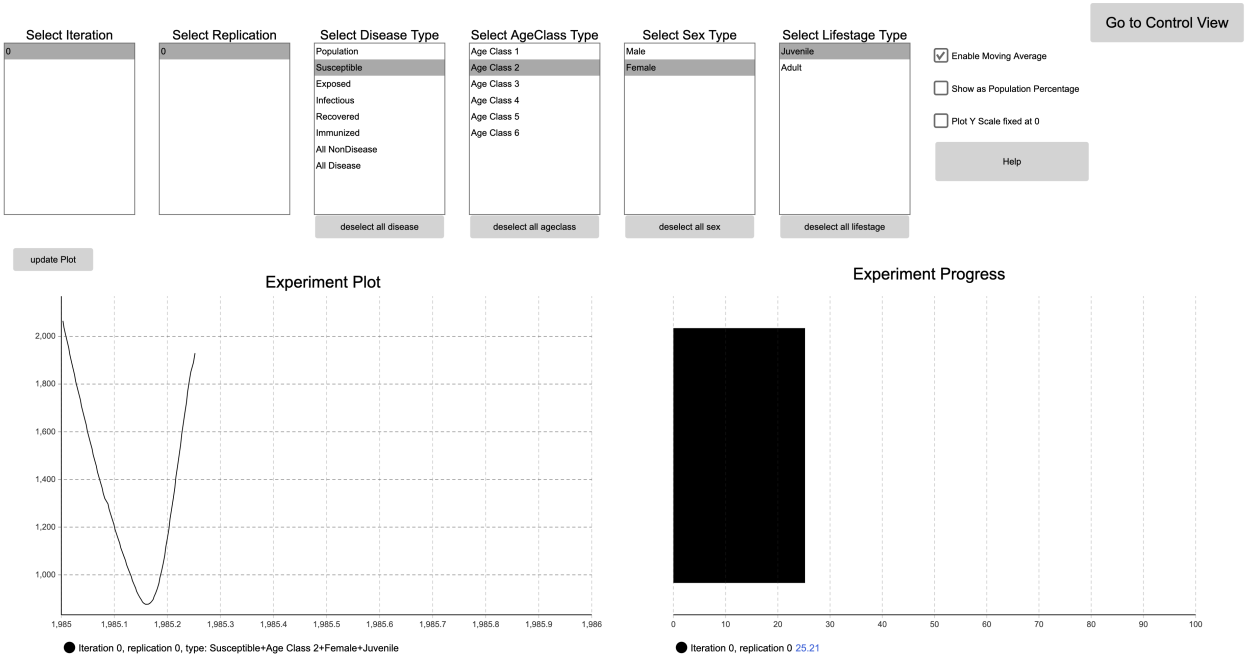


Figure 3.6: Plot View GUI and Example Data Aggregation during runtime

3.6 Scenarios

3.6.1 Experiment Control

All experiments were implemented using the custom experiment in Anylogic. The custom experiment does not generate a GUI and allows full control over all aspects of experiment execution through code. Running the experiments without a GUI allowed me to run them on a research server at the University of Saskatchewan as well as the Compute Canada facilities, without having a persistent connection with a GUI output. If an Anylogic experiment with a GUI loses connection during runtime, then the experiment crashes and all data that is not written to files is lost. Encountering connection issues using X11 over SSH was frequent enough during the multiple day run time, that it was unfeasible. For this reason, I implemented the custom experiments. In addition to not needing a GUI to run, it also gave me more granular control over how many cores to use when running parallel experiments. Each core, or thread in the case of hyperthreading, can run one realization, but at an overall performance penalty. Each experiment keeps track of the status of all realizations as either Ready, Running, or Finished. Each experiment type described below was run with 200 realizations.

3.6.2 Baseline

The baseline experiment examines the behaviour of the Tasmanian devil population in the absence of any interventions. There are two separate runs of the baseline experiment, one is with DFTD, and one is without

DFTD. The non-DFTD experiment was used to examine the model behaviour in the absence of the disease in order to validate the population stability and other death factors. The DFTD baseline experiment was used to ensure the model displays the same characteristics seen in the field and that the disease behaves correctly. The baseline experiment with DFTD was then used as a comparison point for all intervention experiments, providing a metric against which intervention effectiveness could be judged. Each experiment was started in 1985 and stopped in 2035. 1985 was chosen as the start time to provide sufficient time for the model to stabilize the population after start-up and reach a pre-DFDT equilibrium before the introduction of DFTD.

3.6.3 Translocation

The translocation experiments consisted of three separate experiment runs. The first utilized the off-island translocation described in Section 3.2.1. The second experiment used the on-island translocation described in Section 3.2.2. The last experiment type used both off-island translocation and an injection vaccine. In all three experiments, 100 devils were translocated biannually. These devils were released at 12 separate release locations, and in the case of on-island translocation, they were trapped in eight different locations on the main island. All other parameters were kept constant. All interventions were introduced in 2020.

3.6.4 Vaccination

Vaccination intervention experiments used two types of vaccines. The first was an injection vaccine given to trapped devils, and the second was an oral bait vaccine dispersed in the environment (see Section 3.2.4). First an experiment was run using the injection vaccine and trapping devils across the landscape. Then 18 different experiments were run using the oral bait vaccine. Each of the oral bait vaccine experiments varied three parameters to investigate different distribution techniques for the bait vaccine. The first is the average distance between distribution sites, the second is the amount of bait dropped per distribution site, and the last is the number of months between bait drops. These three parameters are used to gain an insight into the optimal parameter space for the oral bait vaccine distribution. The oral bait vaccine experiment also uses a time delayed roll out strategy for distribution of bait. At the beginning of a bait distribution session the cells for bait drops are selected. Then each day a number of cells, in this case 6, receive the bait vaccine. This simulates a real world distribution of vaccine utilizing methods such as airplanes, helicopters and manual distribution. This time delay shows if there are any potential issues with waning immunity and distribution speed, and it can also help to better simulate the real world situation, therefore giving a better insight into potential strategies. All interventions were introduced in 2020.

3.6.5 Inputs

Certain inputs to the model were loaded in from files to setup the model environment. These are:

- **5K_Data.csv.gz** - This file contains all cell specific data. Each row defines one cell and all data associated with it. This file is compressed using gzip to allow for easier and quicker transfer to remote servers.

- **5K_Distance.csv.gz** - This file contains all the connections between different cells that are available and the distance between each connected cell. This file is compressed using gzip to allow for easier and quicker transfer to remote servers.

- **ReleaseLocations.csv** - This file contains all release locations that can be used when translocation is enabled. These release locations will be setup as GIS regions in the model.

- **TrapLocations.csv** - This file contains all the trap locations that can be used when on-island translocation is enabled. Devils can be trapped here to be moved to a different part of the island.

- **Parameters.xlsx** - This file contains all the parameters that are used in the model. Each parameter will be loaded in from this file and set during startup, unless modified in the experiment GUI described in Section 3.5.

- **SensitivityParameters.xlsx** This file contains all parameters that can be toggled for sensitivity analysis. This file is only used when the Sensitivity Analysis experiment is enabled in the experiment GUI described in Section 3.5.

3.6.6 Parameters

Tables 3.5, 3.6, 3.7, 3.8, 3.9, and 3.10 define most parameters used in the model, whether user-defined through the Parameters excel file, internally defined within the model. Beyond the specification of parameter names and values in Table 3.6, user-defined epidemiological parameters are described below. All parameters were discussed with all stakeholders, who also provided estimates for unknown parameters. Where distributions were used for parameters, the following notation will be used:

- **Normal Distribution** - normal(sd, mean)

- **Truncated Normal Distribution** - normal(min, max, mean, sd)

- **Triangular Distribution** - triangular(min, max, mean)

User-Defined Epidemiological Parameters

- **initialPopulationSize** - This parameter controls the count of Tasmanian devil agents at model startup. This value is used to calculate the number of devils per cell based on the density in each cell.

- **beginMatingSeason** - This parameter controls on which day within the year on which the mating season starts, counted from the first day of each year, which in Anylogic is defined as 1.

- **matingSeasonDuration** - This parameter controls how long the mating season lasts. This parameter is specified in days.

- **distanceProbabilityLambda** - This parameter controls the probability that a devil will choose a given cell based on the distance from the devil to that cell. The probability is calculated using an exponential distribution, and this parameter controls the rate of decay of probability with distance in km.

- **housingDeathAdjustmentFactor** - This parameter controls the scaling of the impact of housing density on devil mortality. Housing density mortality is scaled using the relative density of each cell, with the highest housing density resulting in a mortality rate of 1.0 per month, and with the result then being multiplied by this parameter.

- **roadDeathAdjustmentFactor** - This parameter controls the scaling of the impact of road density on devil mortality. Road density mortality is scaled using the relative density of each cell, with the highest road density resulting in a mortality rate of 1.0 per month, and with the result then being multiplied by this parameter.

- **translocationSiteRadius** - This parameter controls the size of each translocation site. For each translocation event, a random point is selected within the site. The cell containing that location is then identified for release of the devil.

- **likelihoodFemale** - This parameter controls the probability that a devil added to the model will be female. The probabilty of being male is 1.0−likelihoodFemale

- **ageOfSexualMaturity** - This parameter controls at which age devils are first able to reproduce.

- **exposedToInfectiousTime** - This parameter specifies how many months a devil will remain in the `Exposed` state before moving to the `Infectious` state. This parameter uses a truncated normal distribution to capture the uncertainty in this parameter.

- **infectiousToDeathTime** - This parameter specifies how many months a devil will remain in the `Infectious` state before dying of DFTD. This parameter uses a truncated normal distribution to capture the uncertainty in this parameter.

- **infectiousProbability** - This parameter controls the probability that a susceptible devil agent will be infected with DFTD given a contact with another infectious devil.

- **exposedToRecoveredRate** - This parameter controls the rate per month at which devils will naturally recover from DFTD given that they are in the `Exposed` state.

- **infectiousToRecoveryRate** - This parameter controls the rate per month at which a devil will naturally recover from DFTD given that they are in the `Infectious` state.

- **recoveredToSusceptibleRate** - This parameter controls the rate per month at which a devil will lose natural immunity against DFTD after naturally recovering from DFTD.

- **probabilityPregnant** - This parameter controls the probability that a female devil will become pregnant as a result of a mating contact with a male devil.

- **probabilityOfMovingAwayFromHomeCell** - This parameter controls the probability that a devil, which has been newly added to the model, will move away from the home cell of its mother.

- **timeUntilPupsAreIndependant** - This parameter controls the count of days until a litter of pups is independent. The value of this counter is dependent on when the female devil became pregnant. Devils will be added to the model once independent.

- **minimumAgeForDFTDInfection** - This parameter controls the minimum age at which a devil can be infected with DFTD; a devil below this age cannot be infected. This parameter uses a truncated normal distribution to capture the uncertainty in this parameter.

- **waningImmunityRate** - This parameter controls the rate at which a devil agent will lose immunity from being vaccinated.

Age Distribution For Initial Devil Generation

The age distribution for initial devil generation — shown in Table 3.7 — is used to assign devil agents to different age classes on model start. These values were generated using the model by running the model until a steady state was reached in all age groups, and then logging the percentage of the population in each age group.

Contact rate parameters

The contact rate parameters, shown in Table 3.9, control the number of contacts each devil has per day for each sex of the devils to be contacted and given the mating season. These values were calculated using the raw devil contact data provided by Hamilton et al.[10]. The data was aggregated by age due to the small sample size in each age group. These contact parameters are used in the model to construct a truncated normal distribution given the *avg* and*sd* values, with a minimum of 0 and a maximum of 60 contacts per day and per sex. The number of devils to contact for each sex in a given day is then calculated by drawing from the distribution; determining the count of contacts on each successive day is achieved by independently draws from that distribution.

Table 3.5: User-Defined Non-Epidemiological Parameters

Parameter	Value	Description
enableDFTD	TRUE	This parameter controls whether DFTD is enabled or not. If it is not enabled, no devils get infected
enableCellDataOutput	TRUE	This parameter controls whether individual cell data is output during the experiment run.

Table 3.6: User-Defined Epidemiological Parameters

Parameter	Value	Unit	Source
initialPopulationSize	90000	devil	-
beginMatingSeason	50	day	[25]
matingSeasonDuration	40	day	[25]
distanceProbabilityLambda	0.4	-	Calibration
housingDeathAdjustmentFactor	0.3	-	Calibration
roadDeathAdjustmentFactor	0.3	-	Calibration
translocationSiteRadius	500	meters	Arbitrary
likelihoodFemale	0.5	-	[25]
ageOfSexualMaturity	triangular(1.2, 2, 1.7)	year	[32]
exposedToInfectiousTime	normal(3, 12, 6, 2)	month	[8]
infectiousToDeathTime	normal(3, 12, 6, 2)	month	[9]
infectiousProbability	0.05	-	Calibration
exposedToRecoveredRate	0.001	month^{-1}	Arbitrary
infectiousToRecoveryRate	0.001	month^{-1}	Arbitrary
recoveredToSusceptibleRate	0.1	month^{-1}	Arbitrary
probabilityPregnant	0.743	-	[25]
probabilityOfMovingAwayFromHomeCell	0.8	-	Arbitrary
timeUntilPupsAreIndependant	227	day	[25]
minimumAgeForDFTDInfection	normal(0.5, 2, 0.5, 0.3)	year	Arbitrary
waningImmunityRate	0.05	month^{-1}	Arbitrary

Table 3.7: Age Distribution for Initial Devil Generation

Age	**Probability**
0-1	0.1299
1-2	0.3102
2-3	0.2295
3-4	0.1641
4-5	0.1227
5+	0.0436

Table 3.8: Number of Pups born per Pregnancy [25]

Number of Pups	**Frequency of Occurrence %**
0	27
1	12
2	13
3	23
4	34

Table 3.9: Contact rate parameters

Season	**Sex**	**Type**	**Male**	**Female**
Mating	Male	AVG	0.267	12.586
		SD	0.255	14.004
	Female	AVG	10.298	1.491
		SD	7.305	2.845
Non-Mating	Male	AVG	0.387	1.765
		SD	0.337	3.359
	Female	AVG	1.471	0.554
		SD	3.045	0.971

Table 3.10: Model Parameters

Parameter	Value	Unit	Description
numberOfDevilsInitialInfection	10	devil	The number of devils that are initially infected when DFTD is introduced
devilDensityPerKm2	2	devil*km^{-2}	The number of devils per km^2, this will be adjusted to match the initialPopulationSize set by the user
lowDensityToRegularDensity	0.25	-	The multiplication factor to determine the devil density for the low density regions
DevilDensityToCarryCapactyMultiplier	1	-	The multiplication factor to determine the carrying capacity of each cell based on its devil density
timeInBetweenCellDataWrites	30	day	The number of days between a full data capture of each GIS cell is performed

3.7 Model execution environment

The agent-based model was executed as a standalone java application using the provided Anylogic export functionality. Exporting the model allows it to be executed on other computers without the need for Anylogic to be installed. Due to the large computational requirement, Compute Canada, a national supercomputing service, was selected to run the model. Compute Canada offers four separate clusters, Cedar, Graham, Béluga, and Niagara. I selected Cedar to run my experiments as it was the most available cluster at the time. Each cluster consists of nodes and each node offers a set of compute cores and memory that are located on the same physical machine within the cluster. Due to the constraints imposed by Anylogic, and for ease of implementation, one experiment could only run on a single node. For this reason, an experiment with more iterations then available cores on a single node were split into smaller blocks and run separately. The experiments consisted of 200 realizations each, and were split into 10 nodes per experiment type. Even though certain nodes can provide up to 40 cores, the amount of memory needed per core is not supported when using 40. Due to this constraint, each node was limited to 20 realizations at once, with a total requested memory use of 180 GB. Each experiment also requested four extra overhead cores to allow for parallel background processing without impacting the 20 running realizations. A script was created to control the scheduling and running of the experiments. This script would schedule each subexperiment and then setup and monitor execution once the node was available.

Earlier model experiments were also run using the research server Skorpio at the University of Saskatchewan. Due to the limited amount of parallel processing power in comparison to the requirement, this was only used for testing purposes and not to generate result data.

Both servers used a headless terminal based system for scheduling and running the experiments. Using a headless approach avoided the X11 and SSH connection instabilities mentioned in section 3.6.1.

3.8 Performance Evaluation

To evaluate the total runtime and memory usage of the agent-based models at different population sizes, and if disease is enabled, a simple time and memory recorder was implemented. Runtime was recorded using the `System.nanotime()` functionality native to Java. This functionality records the time, in nanoseconds, that has passed since a fixed but arbitrary time point. As this time point can be arbitrary, in either the past or the future, it is important that `System.nanotime()` is only used to calculate elapsed time using the difference of two `System.nanotime()` measurements [1]. The maximum memory usage was recorded using the Runtime API native to the Java Virtual Machine. It was not possible to measure the memory consumption of each realization, therefore the maximum memory usage of the entire experiment was measured. Memory usage is evaluated approximately every 30 seconds. To record the runtime for each replication of the model a simple class was implemented. This class captured the ID of the replication, the start time, and stop time. This data was output to a CSV file once the model execution had finished. The start time of a replication was recorded before `engine.run()` was invoked to start the execution of the realization but after initializing the model engines. The stop time of each realization was recorded after the engine has indicated that it had either finished or an error had occurred. The status of each realization was checked approximately every 30 seconds, which lead to an uncertainty of 30 seconds in the stop time. This uncertainty, however, is a minor discrepancy when the multiple hour runtime of the model is taken into consideration. I decided to measure the runtime of each realization instead of the whole model execution because if the number of realizations is set higher then the number of cores available, some realizations have to be run in series, leading to a dramatically increased runtime. The initial setup of the model engine was consistent across all realizations regardless of population size or if DFTD is enabled. The Tasmanian devil population and the geographic grid data are only initialized once `engine.run()` is invoked.

3.9 Statistical Analysis

To analyse the large amount of data produced by each experiment (scenario), multiple Julia scripts were implemented to process the data, test them for statistical significance, and to plot the results. The core script processes the raw data produced by the model. Each realization outputs a CSV file of model data over time. This CSV file is then used to calculate a single DDL and mean daily incidence value for each realization. Mean daily incidence is calculated using the daily incidence output from each model realization. Each new case of DFTD is counted in the model, with the DFTD incidence counter in the model being reset each day. A mean of all daily incidence values is then calculated over all days for each realization. This value is then stored as the mean daily incidence for that realization. Devil Days Lived (DDL) represents the sum of the total population of Tasmanian devils on each day of the model. DDL was used to capture all variation in populations even if they were only temporary. The count of agents at the end of the model execution is sometimes used to measure effectiveness of interventions, but this does not capture fluctuations in population earlier in the model. For this reason, DDL was chosen as the Tasmanian Devil population outcome measure to judge intervention effectiveness. The results of all realizations belonging to a single experiment are then collected and written to a single CSV file. This intermediate storage allows for the time consuming data processing of all realizations to occur only once. The processed results for each experiment are then read into another script which uses the DDL and mean daily incidence values of each realization to check for statistical significance against the baseline experiment. In these results, each experiment produced 200 data points – one for each of 200 realizations – each for DDL and mean daily incidence. The null hypothesis assumed that the intervention used produced no significant difference against the baseline, which utilized no interventions. To test the statistical significance between the baseline and other experiments, two tests were used: The Kolmogorov-Smirnov test, and two-sided Mann-Whitney U test. These tests used the implementation provided by the `HypothesisTests` library for Julia, using the `ApproximateTwoSampleKSTest` and `MannWhitneyUTest`. Since the tests are two tailed, a difference is accepted as significant even if the intervention results in a worsening of DDL or mean daily incidence. Statistical significance was only accepted when the Kolmogorov-Smirnov test and the Mann-Whitney U test both produced a p-value of less then $P < 0.005$. $P < 0.005$ is used as the cutoff for significance, as opposed to the commonly used $P < 0.05$, the marked variability results that can be exhibited between realizations due to stochastics.

3.10 Result Graphing

3.10.1 Graphing output for each cell over time

To capture model output during runtime for each individual cell, I created a Julia program which runs in parallel to the model. This program receives a data stream from each experiment iteration containing data for each cell. The data is formated as a JSON structure to ensure that the data is associated with the appropriate label. This data is then converted into a single frame, shading each cell based on its given value. Once the experiment finishes, it issues a command to the server to indicate that it is done. The server then creates an animation from the individual frames using ffmpeg. Each animation has a fixed length of 30 seconds. Once the animation is created, the individual frames are deleted to free disk space.

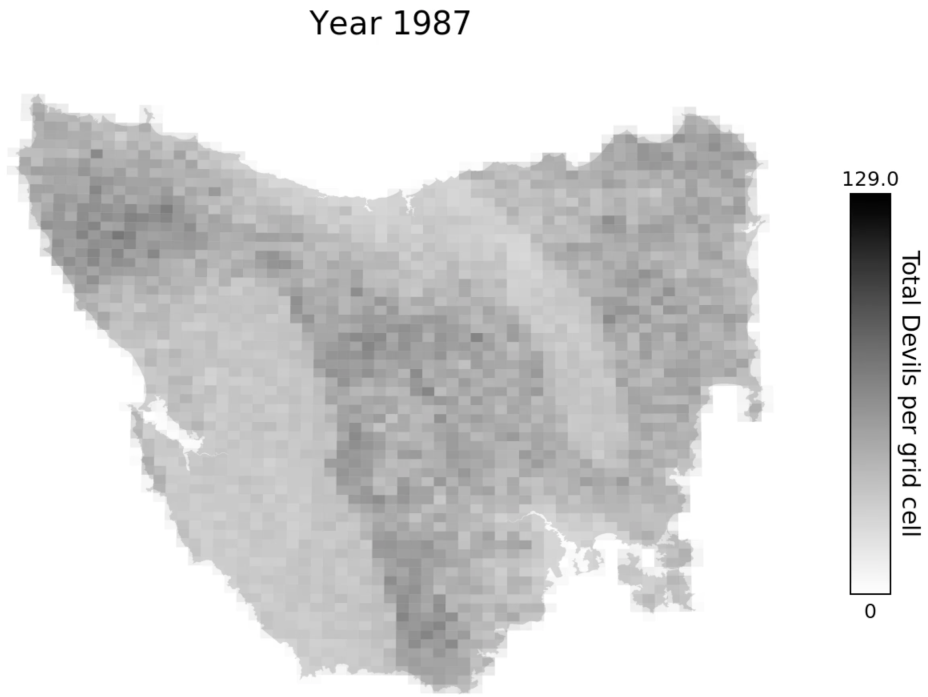


Figure 3.7: Sample output of the total population of Tasmanian devils per cell

4 Results

4.1 Runtime Results

The performance evaluation conducted on the model showed a linear growth in runtime with respect to the number of Tasmanian devil agents, as shown by Figure 4.1. For the experiments where DFTD was disabled, the runtime increased by an average of 54 minutes for every 10,000 devil agents. When DFTD was enabled, the runtime increased by an average of 24 minutes for every 10,000 devil agents. A linear regression produced an R^2 score of 0.9978 when DFTD was disabled, and a R^2 score of 0.9985 when DFTD was enabled.

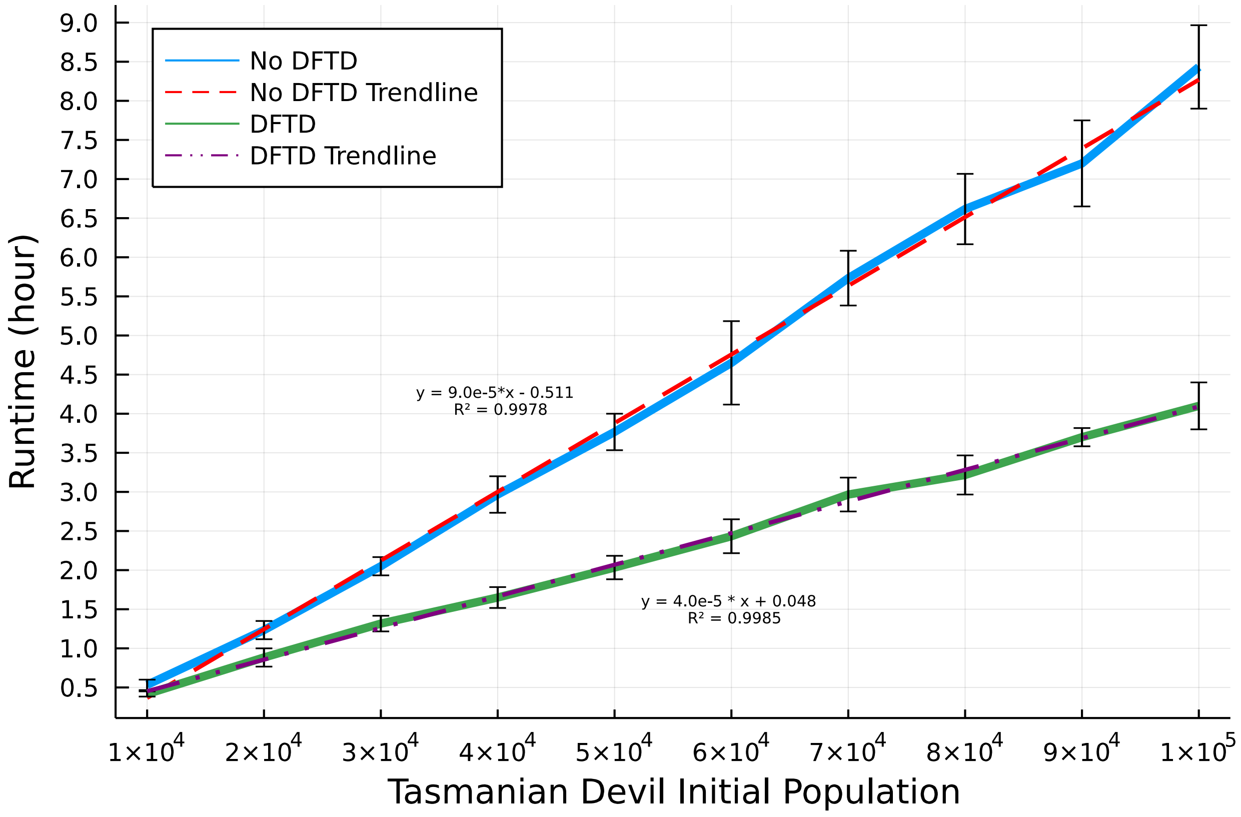


Figure 4.1: Runtime vs. Tasmanian Devil Agent Population

4.2 Memory Results

Maximum memory consumption grew linearly with respect to the number of Tasmanian devil agents, growing by an average of 8.3GB for every 10,000 initial devil agents. The experiments for which DFTD was disabled also showed a linear trend with an average increase of 2.1GB for every 10,000 initial devil agents. Since the slope was very shallow, the linear regression only produced an R^2 score of 0.5844. It is notable that the maximum memory require scales far steeper with the initial population for those model runs positing Devil Facial Tumour Disease; this reflects the greater expansion of that population over the course of the simulation compared to runs in which mortality from DFTD reduces the size of that population.

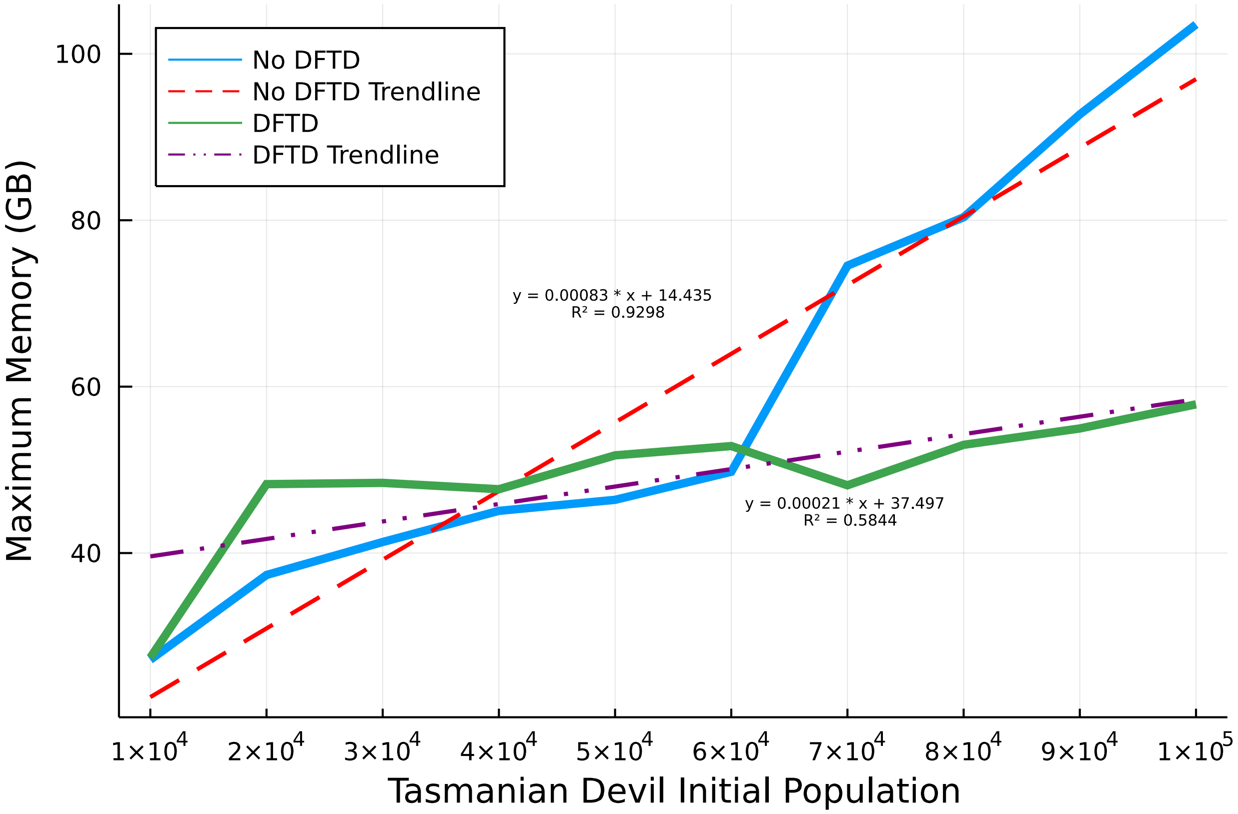


Figure 4.2: Maximum Memory (GB) vs. Tasmanian Devil Agent Population

4.3 Sensitivity Analysis

The sensitivity analysis demonstrated the two parameters with the largest change to impact ratio to be *Distance Probability Lambda* and *Road Death Adjustment Factor*. *Distance Probability Lambda* is used to determine that a cell will be chosen for devil contact given its distance from the current cell of the devil initiating the contact. *Road Death Adjustment Factor* is a multiplier used to determine the probability a devil will die due to roadkill given the road density in the current cell. The impact of these parameters was assessed against two outcomes: Devil days lived (DDL) shown in Tables 4.1 & 4.2 and mean daily incidence shown in Tables 4.3 & 4.4.

In terms of DDL outcomes, *Distance Probability Lambda* produced a change in DDL of -20.7% given a -50% change in the parameter value, and a 30.5% change in DDL for a 50% change over the baseline parameter value. *Road Death Adjustment Factor* produced a 41.4% change in DDL for a -50% change in parameter value, and a -22.9% change in DDL for a 50% change over the baseline parameter value. With respect to mean daily incidence outcomes, *Distance Probability Lambda* exerted a 29.0% change in mean daily incidence given a -50% change over the baseline parameter value, and a -62.4% change in mean daily incidence given a 50% change in the parameter value. *Road Death Adjustment Factor* produced a 19.8% change in mean daily incidence given a -50% change over baseline parameter value, and a -22.7% change in mean daily incidence given a 50% change in the baseline parameter value. All other parameter values showed a less then 10% change for a ±50% change over the baseline parameter value.

Table 4.1: Devil days lived (DDL) for each Sensitivity Analysis experiment given the baseline parameter and ±50%. Values highlighted in Bold represent statistically significant difference compared to the baseline (P<.005)

Parameter Name	**-50%**	**Baseline**	**+50%**
Distance Probability Lambda	$\mathbf{1.27 \times 10^9}$	1.60×10^9	$\mathbf{2.09 \times 10^9}$
Housing Death Adjustment Factor	1.62×10^9	1.61×10^9	1.59×10^9
Mating Season Duration	$\mathbf{1.69 \times 10^9}$	1.60×10^9	$\mathbf{1.55 \times 10^9}$
Minimum Age For DFTD Infection	1.60×10^9	1.60×10^9	1.61×10^9
Probability Of Moving Away From Home Cell	1.61×10^9	1.60×10^9	1.61×10^9
Road Death Adjustment Factor	$\mathbf{2.27 \times 10^9}$	1.61×10^9	$\mathbf{1.24 \times 10^9}$
Vaccine Probability Of Immunity Per Bait Unit Exposed	$\mathbf{1.69 \times 10^9}$	1.71×10^9	1.71×10^9
Vaccine Probability Of Immunity Per Bait Unit Infectious	1.70×10^9	1.71×10^9	1.72×10^9
Vaccine Probability Of Immunity Per Bait Unit Susceptible	$\mathbf{1.68 \times 10^9}$	1.70×10^9	1.72×10^9
Vaccine Probability Of Immunity Per Injection Exposed	1.60×10^9	1.62×10^9	1.61×10^9
Vaccine Probability Of Immunity Per Injection Infectious	1.62×10^9	1.61×10^9	1.62×10^9
Vaccine Probability Of Immunity Per Injection Susceptible	1.61×10^9	1.62×10^9	1.62×10^9

Table 4.2: Devil days lived (DDL) for each Sensitivity Analysis experiment given the baseline parameter and ±50% represented as percent change. Values highlighted in Bold represent statistically significant difference compared to the baseline (P<.005)

Parameter Name	-50%	Baseline	+50%
Distance Probability Lambda	**-20.7%**	-	**30.5%**
Housing Death Adjustment Factor	0.8%	-	-1.1%
Mating Season Duration	**5.5%**	-	**-3.2%**
Minimum Age For DFTD Infection	0.0%	-	0.4%
Probability Of Moving Away From Home Cell	0.4%	-	0.5%
Road Death Adjustment Factor	**41.4%**	-	**-22.9%**
Vaccine Probability Of Immunity Per Bait Unit Exposed	**-1.2%**	-	0.3%
Vaccine Probability Of Immunity Per Bait Unit Infectious	-0.6%	-	0.5%
Vaccine Probability Of Immunity Per Bait Unit Susceptible	**-1.3%**	-	1.1%
Vaccine Probability Of Immunity Per Injection Exposed	-0.9%	-	-0.2%
Vaccine Probability Of Immunity Per Injection Infectious	0.3%	-	0.4%
Vaccine Probability Of Immunity Per Injection Susceptible	-0.1%	-	0.2%

Table 4.3: Mean daily incidence for each Sensitivity Analysis experiment given the baseline parameter and ±50%. Values highlighted in Bold represent statistically significant difference compared to the baseline (P<.005)

Parameter Name	-50%	Baseline	+50%
Distance Probability Lambda	**49.75**	38.58	**14.51**
Housing Death Adjustment Factor	38.81	38.58	38.38
Mating Season Duration	**35.81**	38.52	**39.88**
Minimum Age For DFTD Infection	38.71	38.55	38.41
Probability Of Moving Away From Home Cell	38.49	38.83	38.44
Road Death Adjustment Factor	**46.2**	38.55	**29.8**
Vaccine Probability Of Immunity Per Bait Unit Exposed	**33.94**	33.26	32.58
Vaccine Probability Of Immunity Per Bait Unit Infectious	33.5	33.04	33.1
Vaccine Probability Of Immunity Per Bait Unit Susceptible	**36.33**	33.54	**31.27**
Vaccine Probability Of Immunity Per Injection Exposed	39.42	38.99	39.09
Vaccine Probability Of Immunity Per Injection Infectious	38.93	39.13	39.02
Vaccine Probability Of Immunity Per Injection Susceptible	39.11	39.09	38.99

Table 4.4: Mean daily incidence for each Sensitivity Analysis experiment given the baseline parameter and ±50% represented as percent change. Values highlighted in bold represent statistically significant difference compared to the baseline (P<.005)

Parameter Name	**-50%**	**Baseline**	**+50%**
Distance Probability Lambda	**29.0%**	-	**-62.4%**
Housing Death Adjustment Factor	0.6%	-	-0.5%
Mating Season Duration	**-7.1%**	-	**3.5%**
Minimum Age For DFTD Infection	0.4%	-	-0.4%
Probability Of Moving Away From Home Cell	-0.9%	-	-1.0%
Road Death Adjustment Factor	**19.8%**	-	**-22.7%**
Vaccine Probability Of Immunity Per Bait Unit Exposed	**2.0%**	-	-2.1%
Vaccine Probability Of Immunity Per Bait Unit Infectious	1.4%	-	0.2%
Vaccine Probability Of Immunity Per Bait Unit Susceptible	**8.3%**	-	**-6.8%**
Vaccine Probability Of Immunity Per Injection Exposed	1.1%	-	0.3%
Vaccine Probability Of Immunity Per Injection Infectious	-0.5%	-	-0.3%
Vaccine Probability Of Immunity Per Injection Susceptible	0.0%	-	-0.2%

4.4 Agent-Based Model Results

A total of 23 experiments were run to evaluate different interventions. To serve as a point of reference for comparison, a baseline experiment containing no interventions was initially run. The baseline experiment produced a mean devil days lived (DDL) of 6.81×10^8, and a mean daily incidence rate of 52.43 for all baseline realizations as described in Section 3.9; here and below, the reader should be reminded that this quantity is calculated as the average across incident rates applying on the different days of the study. The first type of intervention was the translocation intervention. This type includes off-island translocation, on-island translocation, and vaccinated off-island translocation. As shown by Tables 4.5 & 4.6 no translocation intervention produced a statistical (P<.005) different result from the baseline. The Injection Vaccine produced a mean DDL of 7.76×10^8 and a mean daily incidence of 39.27. This intervention significantly reduced the burden of DFTD on the population and allowed for the population to recover over time. All oral bait vaccine intervention significantly improved both mean DDL and mean daily incidence over the baseline. The smallest improvement was produced using 20km average distance, 100 bait units per location, and 12 months between bait drops. This combination resulted in a mean DDL of 6.97×10^8 and a mean daily incidence of 51.42. The largest improvement over the baseline resulted from using a 10km average distance, 1000 bait units per location, and six months between bait drops. This parameter set resulted in a mean DDL of 8.34×10^8 and a mean daily incidence of 24.91. Two oral bait vaccine interventions produced a mean prevalence of 0% at the end of the model execution, representing eradication of DFTD as shown in Figure 4.6. Mean prevalence was calculated as the mean of each realization per day. When the mean prevalence equals 0% exactly, then all realizations show eradication of DFTD; given model assumptions, once eradication is achieved, it will be maintained from that point forward. The parameter combination that produced the quickest eradication used 10km average distance between adjacent bait drops, 1000 bait units per location, and six months between bait drops. This scenario required 2.5 years to reach eradication. The other parameter combination used 10km average distance, 500 bait units per location, and six months between bait drops, requiring 4.5 years to eradication. Figure 4.5 shows the same average distance and bait amount per location, but using twelve months between bait drops. This combination does not results in eradication, but using 1000 bait units per location reduced the prevalence to below 0.1% 1.2 years after introduction of the Intervention, with prevalence remaining below 0.1% until the end of model execution.

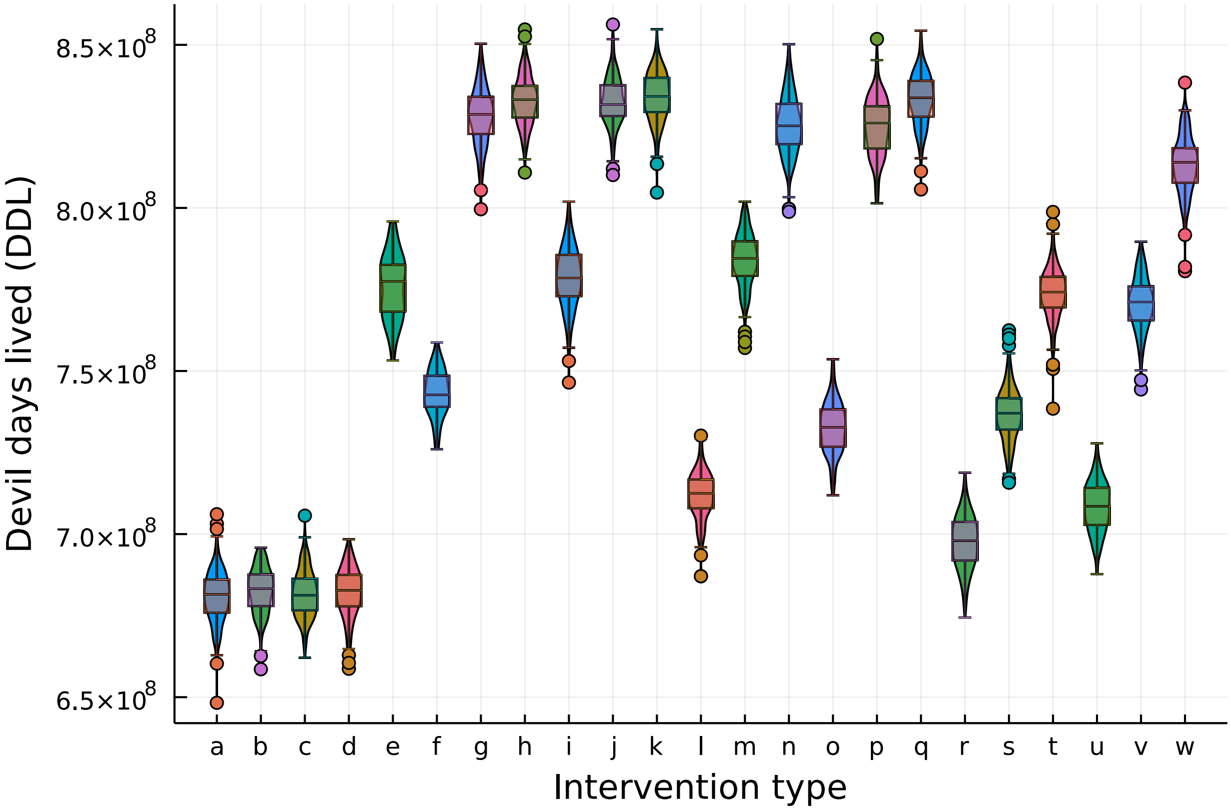


Figure 4.3: Violin and Box plot showing the distribution of DDL for each intervention type. The Oral bait vaccine (OBV) intervention type follows the following notation: X-Y-Z. X denotes distance between drops in kilometers, Y denotes the number of baits per drop, and Z denotes the number of months between drops. (a) No Intervention, (b) Off Island Translocation, (c) On Island Translocaiton, (d) Vaccinated Off-Island Translocation, (e) Injection Vaccine, (f) OBV 10-100-12, (g) OBV 10-500-12, (h) OBV 10-1000-12, (i) OBV 10-100-6, (j) OBV 10-500-6, (k) OBV 10-1000-6, (l) OBV 15-100-12, (m) OBV 15-500-12, (n) OBV 15-1000-12, (o) OBV 15-100-6, (p) OBV 15-500-6, (q) OBV 15-1000-6, (r) OBV 20-100-12, (s) OBV 20-500-12, (t) OBV 20-1000-12, (u) OBV 20-100-6, (v) OBV 20-500-6, (w) OBV 20-1000-6

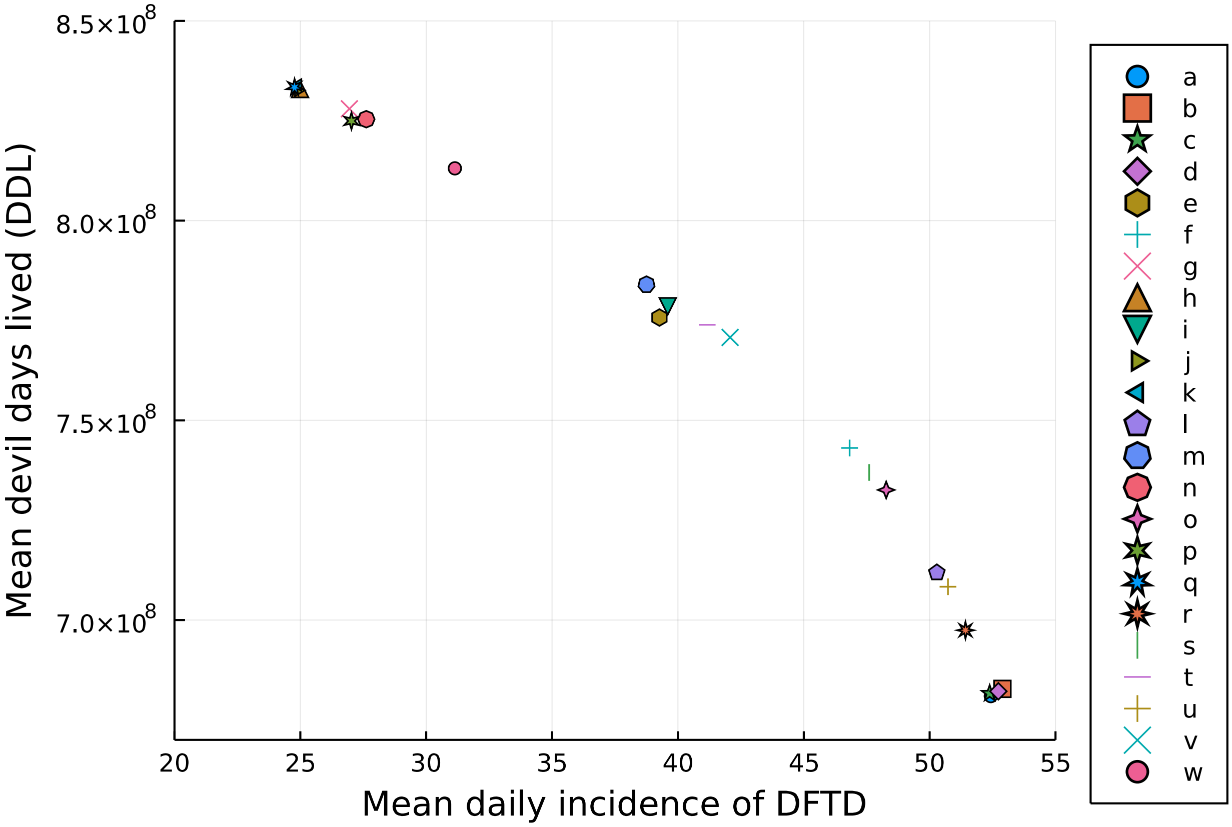


Figure 4.4: Mean DDL vs. mean daily incidence of DFTD of each intervention type. Highlighted in bold are the data points overlapped in the top-left. The Oral bait vaccine (OBV) intervention type follows the following notation: X-Y-Z. X denotes distance between drops in kilometers, Y denotes the number of baits per drop, and Z denotes the number of months between drops. (a) No Intervention, (b) Off Island Translocation, (c) On Island Translocaiton, (d) Vaccinated Off-Island Translocation, (e) Injection Vaccine, (f) OBV 10-100-12, (g) OBV 10-500-12, (h) **OBV 10-1000-12**, (i) OBV 10-100-6, (j) **OBV 10-500-6**, (k) **OBV 10-1000-6**, (l) OBV 15-100-12, (m) OBV 15-500-12, (n) OBV 15-1000-12, (o) OBV 15-100-6, (p) OBV 15-500-6, (q) **OBV 15-1000-6**, (r) OBV 20-100-12, (s) OBV 20-500-12, (t) OBV 20-1000-12, (u) OBV 20-100-6, (v) OBV 20-500-6, (w) OBV 20-1000-6

Table 4.5: Mean DDL for each experiment type. Oral Bait Vaccine intervention (OBV) follows the following notation: X-Y-Z. X denotes distance between drops in kilometers, Y denotes the number of baits per drop, and Z denotes the number of months between drops. Bold values denote significance (P<.005) compared to the no intervention baseline

Experiment type	**mean DDL**	**mean DDL SD**
No Intervention	6.81×10^8	8.03×10^6
Off-Island Translocation	6.83×10^8	7.02×10^6
On-Island Translocation	6.82×10^8	7.25×10^6
Off-Island + Injection Vaccine	6.82×10^8	7.57×10^6
Injection Vaccine	$\mathbf{7.76 \times 10^8}$	1.00×10^7
OBV 10-100-12	$\mathbf{7.43 \times 10^8}$	7.30×10^6
OBV 10-500-12	$\mathbf{8.28 \times 10^8}$	9.08×10^6
OBV 10-1000-12	$\mathbf{8.33 \times 10^8}$	7.54×10^6
OBV 10-100-6	$\mathbf{7.79 \times 10^8}$	9.38×10^6
OBV 10-500-6	$\mathbf{8.33 \times 10^8}$	7.59×10^6
OBV 10-1000-6	$\mathbf{8.34 \times 10^8}$	7.69×10^6
OBV 15-100-12	$\mathbf{7.12 \times 10^8}$	7.34×10^6
OBV 15-500-12	$\mathbf{7.84 \times 10^8}$	8.36×10^6
OBV 15-1000-12	$\mathbf{8.25 \times 10^8}$	9.29×10^6
OBV 15-100-6	$\mathbf{7.33 \times 10^8}$	7.92×10^6
OBV 15-500-6	$\mathbf{8.25 \times 10^8}$	8.94×10^6
OBV 15-1000-6	$\mathbf{8.33 \times 10^8}$	7.85×10^6
OBV 20-100-12	$\mathbf{6.97 \times 10^8}$	8.20×10^6
OBV 20-500-12	$\mathbf{7.37 \times 10^8}$	8.39×10^6
OBV 20-1000-12	$\mathbf{7.74 \times 10^8}$	8.04×10^6
OBV 20-100-6	$\mathbf{7.08 \times 10^8}$	7.77×10^6
OBV 20-500-6	$\mathbf{7.71 \times 10^8}$	8.23×10^6
OBV 20-1000-6	$\mathbf{8.13 \times 10^8}$	9.06×10^6

Table 4.6: Mean daily incidence rate for each experiment type. Oral Bait Vaccine intervention (OBV) follows the following notation: X-Y-Z. X denotes distance between drops in kilometers, Y denotes the number of baits per drop, and Z denotes the number of months between drops. Bold values denote significance (P<.005) compared to the no intervention baseline

Experiment type	**mean daily incidence**	**mean daily incidence SD**
No Intervention	52.43	0.72
Off-Island Translocation	52.89	0.63
On-Island Translocation	52.38	0.62
Off-Island + Injection Vaccine	52.73	0.69
Injection Vaccine	**39.27**	2.46
OBV 10-100-12	**46.82**	0.95
OBV 10-500-12	**26.95**	1.85
OBV 10-1000-12	**24.99**	0.61
OBV 10-100-6	**39.60**	1.83
OBV 10-500-6	**24.92**	0.62
OBV 10-1000-6	**24.91**	0.56
OBV 15-100-12	**50.29**	0.68
OBV 15-500-12	**38.75**	1.83
OBV 15-1000-12	**27.62**	2.25
OBV 15-100-6	**48.27**	0.82
OBV 15-500-6	**27.031**	1.94
OBV 15-1000-6	**24.77**	0.60
OBV 20-100-12	**51.42**	0.71
OBV 20-500-12	**47.60**	0.87
OBV 20-1000-12	**41.16**	1.82
OBV 20-100-6	**50.73**	0.694
OBV 20-500-6	**42.07**	1.31
OBV 20-1000-6	**31.14**	2.01

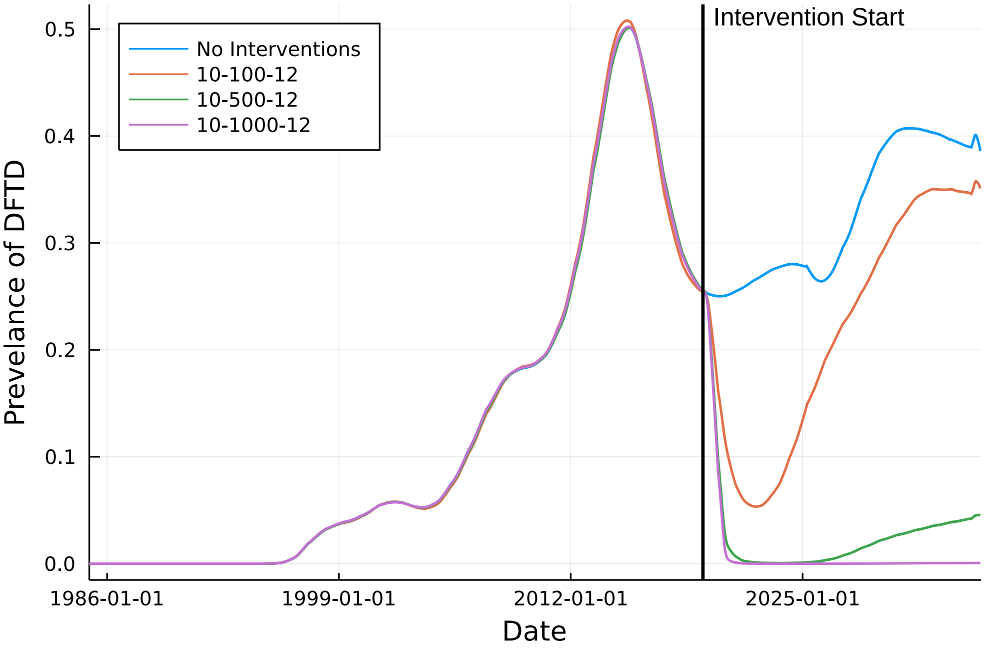


Figure 4.5: Prevalence of DFTD for the Oral Bait Vaccine Intervention with 10km average distance, a varying amount of bait being dropped per location, and distribution every 12 months

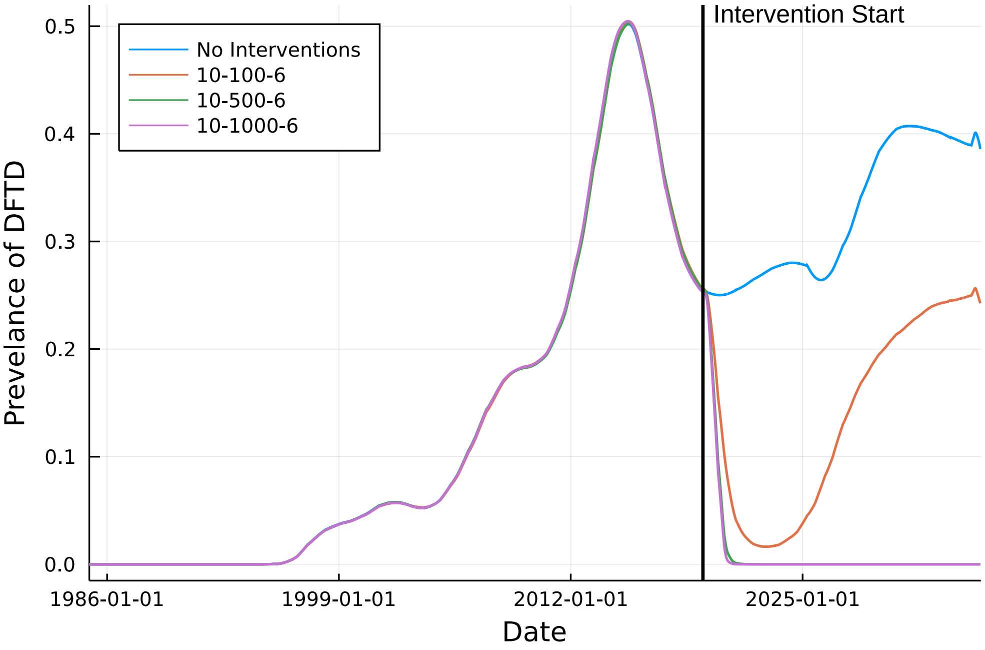


Figure 4.6: Prevalence of DFTD for the Oral Bait Vaccine Intervention with 10km average distance, a varying amount of bait being dropped per location, and distribution every 6 months

5 Conclusion

5.1 Motivation

Devil Facial Tumour Disease has decimated the Tasmanian devil population since its discovery in 1996. When combined with other mortality risks, it is feared that DFTD may lead to extinction of the species, which would be a tragedy for Tasmania, and the ecosystem. To evaluate different interventions, a geospatial agent-based model was constructed to simulate interactions of individual Tasmanian devils. This allows targeted interventions to be evaluated, such as those involving translocation of devils, and different vaccine types. As a novel intervention to combat and potentially eradicate DFTD, a set of scenarios involving use of an oral bait vaccine were also assessed.

5.2 Chapter Summary

Chapter 1 introduced the research that was conducted in this thesis. In this chapter, Devil facial tumour disease and its history was introduced to understand the impact it had on the Tasmanian devil population. Additionally, agent-based modeling was introduced and its benefits described in the setting of epidemiological modeling. Lastly, the goals, scientific contributions and a thesis organization was also presented.

Chapter 2 offered further background on Tasmanian devils, including their behaviour and reproduction, as well as on DFTD and agent-based modeling.

Chapter 3 discussed the model implementation, and specific characteristics of the Tasmanian devil agent. The devil agent description also included discussion of the statecharts utilized within the agent. Chapter 3 additionally presented all parameters used in the model, and the sensitivity analysis and calibration process. The chapter further discussed the different interventions that were implemented in the model and how they were utilized. Lastly, data capture, output, and processing was also described.

Chapter 4 reported the results of the runtime and memory analysis performed, which showed a strong linear trend in both runtime and memory consumption with the count of agents. Sensitivity analysis results were also presented, showing the impact of certain parameters on the output of the model, and highlighting the importance of the parameters impacting death rates, contact rates, and vaccine efficacy. Finally, chapter 4 presented the intervention results in comparison to the baseline. The intervention results highlighted the effectiveness of vaccination, especially using oral bait. The oral bait vaccine was shown to be the most effective, with some configurations even resulting in eradication of DFTD in the agent-based model.

5.3 Deliverables

This project offers three primary deliverables. The first is the geospatial agent-based model with interventions implemented. The second is to set up the model to be runnable on appropriate infrastructure as well as training of all stakeholders on how to run the model and where results are stored. The third is a journal publications regarding the intervention results, aimed at assisting in Tasmanian devil conservation efforts on the island of Tasmania.

5.4 Agent-Based Model Performance

The runtime and maximum memory results presented in sections 4.1 and 4.2 reveal that the model scales linearly in both runtime and maximum memory consumption with the count of agents. This linear — as opposed to super-linear — increase was achieved through extensive optimization of the model. Optimization efforts were especially focused on the contact made between devil agents. The Anylogic implementation of searching for devils within a given distance searches through all agents in the population and calculates a distance. It then keeps a list of all agents within the given distance. This implementation results in a time complexity of $O(n^2)$ since each devil agent will search the entire population for each contact attempt. Given this time complexity, it is infeasible to use this implementation for an agent-based model, with a population of 100,000 or more agents. To achieve a time complexity of $O(n)$ — a linear growth in runtime — the contact search needed to be optimized. This was achieved by tessellating the space into square cells, and storing each devil in a cell. This list of all devils in a cell was then separated by age class and sex. When an index devil searches for another devil, first a cell is selected given its associated probability determined by its distance from the index devil's home cell. Then, given an age class and a sex, a random index is produced for the list of devils in that cell with the given age class and sex. If the list is empty, then no suitable devil was found for contact. Given the ranges of practical population size of devils, this implementation allows for the contact search to occur in essentially constant time for a single devil agent. Given this, a practical linear time complexity is achieved. When running a single realization of an agent-based model on a single core using Anylogic, the best possible time complexity is linear, since each new agent adds to the list of events that need to be processed.

5.5 DFTD Agent-Based Model

The agent-based model was successfully used to assess the viability of several different interventions in increasing the Tasmanian devil population while decreasing the incidence of DFTD. The vaccine interventions demonstrate the best results over the baseline. The oral bait vaccine is the most promising for implementation due to its outstanding performance in decreasing the burden of DFTD and restoring the Tasmanian devil population, and also due to its comparatively simple — if logistically involved — real world implementation.

The use of geospatial data in the agent-based model provided a more in depth understanding of external pressures on the Tasmanian devil population. Incorporation of the geospatial components of the model also enabled the simulation of the entire devil population, including the density distribution of devils across the island.

The off-island translocation was first proposed as the main intervention to investigate because it is currently in use. The results presented here show that it will not have an impact on the devil population due to the low number of devils being translocated per year, even when that rate is increased from current levels. Model results also suggest that it is likely that with enough increased translocation to impact the overall population, incidence of DFTD would increase as well, which would exacerbate the burden of DFTD. Vaccinated off-island translocation did not differ from unvaccinated translocation due to the rapid waning of vaccine-induced immunity — waning that occurs at a rate of 0.1 per month. Translocation can provide relief to local populations, and can be a viable option for preventing local extinction, however, it is unlikely to have a positive impact on the entire Tasmanian devil population.

The injection vaccine intervention showed a promising result by statistically improving mean devil days lived, as well as lowering mean daily incidence of DFTD. However, achieving such benefits requires investment of a large amount of resources and human effort. Traps need to be set out every 10 km across the entire island, even in remote areas. Then, each trap needs to be visited daily to check, vaccinate, and release captured animals, disinfect the trap and reset them. Multiple recaptures of the same animal, or capture of non-target species, is an unavoidable hindrance which would increase the amount of resources required. In the model, this trapping effort occurred for 30 days simultaneously across the entire island every 12 months. The intervention ran for 15 years before the model execution was stopped, but if the intervention is stopped, DFTD prevalence and population levels would likely return to pre-intervention levels. The amount of resources needed, particularly human labour, makes this intervention infeasible. If this intervention were to be implemented in the field, logistical challenges would require it to be modified, specifically, increasing the distance between trap sites and using a time delayed rollout across the island. However, as was seen with the oral bait vaccine, this would drastically lower the effectiveness of the intervention.

The oral bait vaccine intervention was the most impactful and feasible intervention that was tested. Oral bait vaccine is much easier to distribute to Tasmanian devils than injection vaccine, since it does not need to be injected into each devil, but can be dropped into the landscape to be taken up by the devils themselves. Bait vaccines have been shown to be effective at combating communicable diseases amongst wild animals through extensive use in North America and Europe. An oral bait vaccine is currently being used in Europe to combat Rabies, and approximately 665 million units of bait were distributed across 2.5 million km^2 from 1978 to 2014. Rabies bait vaccine efforts in Ontario, Canada, use airplanes and helicopters to distribute bait over large areas over a short time. In urban areas, a hand distribution technique was utilized to more precisely target bait distribution and placement. When considering implementation of a bait vaccine for

DFTD, due to the low density of devils in urban areas, manual distribution of bait would likely be limited to fewer instances then were required when combating rabies in Ontario. In large areas of Tasmania, an aerial distribution technique can be utilized to quickly distribute bait over large uninhabited or sparsely inhabited areas. As seen in Section 4.4, the model suggests that the most aggressive distribution strategies for oral bait in Tasmania may lead to the eradication of DFTD in a few years, allowing the population to rebound to pre-DFTD levels. While the real world outcome may not occur as quickly, or achieve complete eradication of DFTD, it would result in a large reduction of DFTD prevalence in Tasmanian devils. This reduction would allow for a more targeted approach — such as through use of injection vaccine — in local areas as needed, providing a path to eradication. For those intervention experiments where prevalence was merely reduced but DFTD was not eradicated, accruing sustained benefits to the Tasmanian devil population would require a sustained oral bait vaccine campaign. Due to the assumed waning of the oral bait vaccine, over time most devils would lose their immunity and DFTD would again very likely spread amongst the population, reducing it to pre-intervention levels. For the interventions in which DFTD is eradicated, interventions can cease, allowing the Tasmanian devil population to naturally recover to pre-DFDT levels. In the case of eradication, discontinuation of interventions would also offer large savings in resources and costs related to the intervention. All oral bait vaccine interventions that lead to eradication of DFTD used a large amount of bait in a short amount of time, leading to a high percentage of devils being immunized. Once a critical fraction of the devil population achieves immunity, the population achieves herd immunity, making it infeasible for DFTD to effectively spread amongst devils. In the interventions in which prevalence was greatly reduced but eradication did not occur, a critical percentage of immunized devils was not achieved. This then leads to an increase in the Tasmanian devil population, which results in increased transmission due to the limited number of vaccines not being sufficient for that enlarged population to reach herd immunity. This temporary relief then results in increasing prevalence, reaching a new equilibrium supported by the oral bait vaccine – an requiring the costs and logistical effort required to sustain it.

The evaluation of interventions on a geographic level provides invaluable insight into potential intervention implementation strategies and effective prioritization of certain interventions. The time-based roll out of the oral bait vaccine opposed to an instantaneous deployment shows that even when bait is distributed across the island over months rather than simultaneously, the effect is still robust.

The agent based model was successfully implemented to capture geographic data from the island of Tasmania, and to simulate the entire Tasmanian devil population. A GIS map was used in Anylogic to represent the geographic environment being simulated. QGIS was used to compile all geographic data, create the 2D grid, and associate all data with the appropriate cells. The 2D grid was successfully implemented in Anylogic using the `Cell` class shown in A.1. The baseline model was then built up with the feedback from the research group in Tasmania and Dr. Carmel Witte. Interventions were then implemented and the entire model was manually calibrated to match empirical data. All experiment runs can either be controlled through a custom designed GUI implemented using Anylogic, or all parameters can be set through an Excel spreadsheet and the experiment run without a GUI for improved performance. The agent-based model successfully demonstrated the viability of the vaccine interventions to stabilize the Tasmanian devil population as well as to reduce the prevalence of DFTD. The best intervention to increase Tasmanian devil population without increasing DFTD incidence that can be recommended based on this agent-based model is the oral bait vaccine intervention with an aggressive deployment of bait that offers the potential to eradicate DFTD.

5.6 Limitations

- Further calibration with more detailed empirical data should be undertaken to better estimate model intrinsic parameters. Supporting data, however, can be difficult to obtain at the level of quality necessary to calibrate the model successfully.
- A further analysis focused on localized populations should be undertaken to better evaluate the translocation interventions and their impact on the local populations.
- The translocation interventions were limited to the currently used trap and release locations. There are currently no plans to expand to more locations, but these could be simulated to evaluate locations of greater impact for future expansion.
- The injection vaccine intervention should be expanded to simulate different rollout strategies and impacts on intervention effectiveness through trapping of non-target species
- Precocial breeding of a devil within the model does not depend on the food availability or devil density of the area enclosing that devil.

5.7 Scientific Contributions

- The first GIS agent-based model to investigate and evaluate the effectiveness of a DFTD oral bait vaccine as an intervention to combat and potentially eradicate DFTD
- Successful design, construction, refinement, and description of an agent-based model simulating the dynamics of DFTD in Tasmanian devils in a geographic setting
- Findings confirming the high potential effectiveness of DFTD oral bait vaccine in combating and potentially eradicate DFTD
- Findings confirming that such oral bait vaccine exerts high effectiveness even in the context of multi-year vaccine roll-out (e.g., 6 months between bait drops) and at modest geographic densities (bait distributed in a grid with an inter-bait spacing of 20 km)

5.8 Engineering Contributions

- Implementation of the model noted above
- Developing a geographic grid system to incorporate geographic data into an agent-based model while improving runtime performance
- Using bit encoding to store agent data during runtime so as to enhance model efficiency
- Implementing a graphical user interface to control model setup and execution
- Graphing geographical data produced by the model during runtime

5.9 Future Work

Beyond addressing some of the needs suggested in the course of noting the limitations above, great benefits could be secured through use of the model in the context of the Sequential Monte Carlo method of particle filtering. Particle filtering would provide the capability to continually update and reground the model with data as it is gathered. This can lead to more accurate, ongoing and always current short term prediction and would be an important tool once interventions are rolled out to greatly reduce the burden of DFTD, or eradicate DFTD entirely. The use of particle filtering can keep the model current with an up-to-date estimate with the latest data regarding the progress of the intervention and measured evolving estimates of the prevalence of DFTD.

To use particle filtering, the model would need to be implemented in using a custom general modelling framework similar to Anylogic, but one that provides the capability to readily utilize particle filtering with agent-based models. The other option would be to implement the model in an efficient general purpose programming language and to have a model-specific implementation of particle filtering. The latter option would result in a quicker implementation of the particle filter and speedier availability of results.

A general modelling framework which can support hybrid modelling as well as advanced techniques such as particle filtering or PMCMC is needed to advance wildlife models into the realm of real-time decision making. Given the complexity of gathering sufficient data on wild animals, a particle filter or PMCMC model can be utilized to estimate underlying states of the system over time in the context of arriving data, and to use that empirically-informed model to better estimate the current and evolution of the future state of the population. The incorporation of real time data, machine learning, and geospatial agent-based modelling can produce very powerful models to help understand the complex systems involved wildlife diseases and to support insight informed by the latest observed empirical data as to how to best combat such diseases over time.

Further additions to the model could involve characterization of resources and costs associated with each intervention. Such an economic and resource component could inform the implementation of each intervention and highlight both their respective resource costs over time and the potential for savings accrued by pursuing more aggressive strategies to potentially eradicate DFTD. Adding these components would elevate this model by complementing existing epidemiological outcomes with those relevant to implementation science.

References

[1] "Java SE 11 & JDK 11". Java API Reference.

[2] Georgina E. Andersen, Hugh W. McGregor, Christopher N. Johnson, and Menna E. Jones. Activity and social interactions in a wide-ranging specialist scavenger, the Tasmanian devil (Sarcophilus harrisii), revealed by animal-borne video collars. *PLOS ONE*, 15(3):e0230216, March 2020. Publisher: Public Library of Science.

[3] Jason Bittel. Tasmanian devils return to mainland Australia for first time in 3,000 years, October 2020. Section: Animals.

[4] Pierre Boitard and Jules Gabriel Janin. *Le Jardin des plantes : description et mœurs des mammife'res de la Me´nagerie et du Muse´um d'histoire naturelle.* G. Barba,, Paris :, 1841. Pages: 204.

[5] Anna Brüniche–Olsen, Menna E. Jones, Christopher P. Burridge, Elizabeth P. Murchison, Barbara R. Holland, and Jeremy J. Austin. Ancient DNA tracks the mainland extinction and island survival of the Tasmanian devil. *Journal of Biogeography*, 45(5):963–976, 2018. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/jbi.13214.

[6] Janine E. Deakin, Hannah S. Bender, Anne-Maree Pearse, Willem Rens, Patricia C. M. O'Brien, Malcolm A. Ferguson-Smith, Yuanyuan Cheng, Katrina Morris, Robyn Taylor, Andrew Stuart, Katherine Belov, Chris T. Amemiya, Elizabeth P. Murchison, Anthony T. Papenfuss, and Jennifer A. Marshall Graves. Genomic Restructuring in the Tasmanian Devil Facial Tumour: Chromosome Painting and Gene Mapping Provide Clues to Evolution of a Transmissible Tumour. *PLOS Genetics*, 8(2):e1002483, February 2012.

[7] Andrew S. Flies, Emily J. Flies, Samantha Fox, Amy Gilbert, Shylo R. Johnson, Guei-Sheung Liu, A. Bruce Lyons, Amanda L. Patchett, David Pemberton, and Ruth J. Pye. An oral bait vaccination approach for the Tasmanian devil facial tumor diseases. *Expert Review of Vaccines*, 19(1):1–10, January 2020. Publisher: Taylor & Francis _eprint: https://doi.org/10.1080/14760584.2020.1711058.

[8] Rodrigo Hamede, Jim Bashford, Menna Jones, and Hamish McCallum. Simulating devil facial tumour disease outbreaks across empirically derived contact networks. *Journal of Applied Ecology*, 49(2):447–456, 2012. _eprint: https://besjournals.onlinelibrary.wiley.com/doi/pdf/10.1111/j.1365-2664.2011.02103.x.

[9] Rodrigo Hamede, Shelly Lachish, Katherine Belov, Gregory Woods, Alexandre Kreiss, Anne-Maree Pearse, Billie Lazenby, Menna Jones, and Hamish McCallum. Reduced Effect of Tasmanian Devil Facial Tumor Disease at the Disease Front. *Conservation Biology*, 26(1):124–134, 2012. _eprint: https://conbio.onlinelibrary.wiley.com/doi/pdf/10.1111/j.1523-1739.2011.01747.x.

[10] David G. Hamilton, Menna E. Jones, Elissa Z. Cameron, Hamish McCallum, Andrew Storfer, Paul A. Hohenlohe, and Rodrigo K. Hamede. Rate of intersexual interactions affects injury likelihood in Tasmanian devil contact networks. *Behavioral Ecology*, 30(4):1087–1095, July 2019. Publisher: Oxford Academic.

[11] C. E. Hawkins, H McCallum, N Mooney, M Jones, and M Holdsworth. IUCN Red List of Threatened Species: Sarcophilus harrisii, June 2008.

[12] C.E. Hawkins, C. Baars, H. Hesterman, G.J. Hocking, M.E. Jones, B. Lazenby, D. Mann, N. Mooney, D. Pemberton, S. Pyecroft, M. Restani, and J. Wiersma. Emerging disease and population decline of an island endemic, the Tasmanian devil Sarcophilus harrisii. *Biological Conservation*, 131(2):307–324, August 2006.

[13] C.J. Hogg and J. Hockley. Husbandry Guidelines for Tasmanian Devil, Sarcophilus harrisii, 2013.

[14] T. Keeley, J. K. O'Brien, B. G. Fanson, K. Masters, and P. D. McGreevy. The reproductive cycle of the Tasmanian devil (Sarcophilus harrisii) and factors associated with reproductive success in captivity. *General and Comparative Endocrinology*, 176(2):182–191, April 2012.

[15] Alexandre Kreiss, Yuanyuan Cheng, Frank Kimble, Barrie Wells, Shaun Donovan, Katherine Belov, and Gregory M. Woods. Allorecognition in the Tasmanian Devil (Sarcophilus harrisii), an Endangered Marsupial Species with Limited Genetic Diversity. *PLOS ONE*, 6(7):e22402, July 2011. Publisher: Public Library of Science.

[16] Shelly Lachish, Hamish McCallum, and Menna Jones. Demography, Disease and the Devil: Life-History Changes in a Disease-Affected Population of Tasmanian Devils (Sarcophilus harrisii). *Journal of Animal Ecology*, 78(2):427–436, 2009. Publisher: [Wiley, British Ecological Society].

[17] Billie T. Lazenby, Mathias W. Tobler, William E. Brown, Clare E. Hawkins, Greg J. Hocking, Fiona Hume, Stewart Huxtable, Philip Iles, Menna E. Jones, Clare Lawrence, Sam Thalmann, Phil Wise, Howel Williams, Samantha Fox, and David Pemberton. Density trends and demographic signals uncover the long-term impact of transmissible cancer in Tasmanian devils. *Journal of Applied Ecology*, 55(3):1368–1379, 2018. _eprint: https://besjournals.onlinelibrary.wiley.com/doi/pdf/10.1111/1365-2664.13088.

[18] Mike Letnic, Melanie Fillios, and Mathew S. Crowther. Could Direct Killing by Larger Dingoes Have Caused the Extinction of the Thylacine from Mainland Australia? *PLOS ONE*, 7(5):e34877, May 2012. Publisher: Public Library of Science.

[19] Mark J. Margres, Manuel Ruiz-Aravena, Rodrigo Hamede, Kusum Chawla, Austin H. Patton, Matthew F. Lawrance, Alexandra K. Fraik, Amanda R. Stahlke, Brian W. Davis, Elaine A. Ostrander, Menna E. Jones, Hamish McCallum, Patrick J. Paddison, Paul A. Hohenlohe, David Hockenbery, and Andrew Storfer. Spontaneous Tumor Regression in Tasmanian Devils Associated with RASL11A Activation. *Genetics*, 215(4), June 2020. Publisher: Genetics Section: Investigation.

[20] Mark J Margres, Manuel Ruiz-Aravena, Rodrigo Hamede, Menna E Jones, Matthew F Lawrance, Sarah A Hendricks, Austin Patton, Brian W Davis, Elaine A Ostrander, Hamish McCallum, Paul A Hohenlohe, and Andrew Storfer. The Genomic Basis of Tumor Regression in Tasmanian Devils (Sarcophilus harrisii). *Genome Biology and Evolution*, 10(11):3012–3025, November 2018.

[21] Hamish McCallum, Daniel M. Tompkins, Menna Jones, Shelly Lachish, Steve Marvanek, Billie Lazenby, Greg Hocking, Jason Wiersma, and Clare E. Hawkins. Distribution and Impacts of Tasmanian Devil Facial Tumor Disease. *EcoHealth*, 4(3):318–325, October 2007.

[22] Elizabeth P. Murchison, Ole B. Schulz-Trieglaff, Zemin Ning, Ludmil B. Alexandrov, Markus J. Bauer, Beiyuan Fu, Matthew Hims, Zhihao Ding, Sergii Ivakhno, Caitlin Stewart, Bee Ling Ng, Wendy Wong, Bronwen Aken, Simon White, Amber Alsop, Jennifer Becq, Graham R. Bignell, R. Keira Cheetham, William Cheng, Thomas R. Connor, Anthony J. Cox, Zhi-Ping Feng, Yong Gu, Russell J. Grocock, Simon R. Harris, Irina Khrebtukova, Zoya Kingsbury, Mark Kowarsky, Alexandre Kreiss, Shujun Luo, John Marshall, David J. McBride, Lisa Murray, Anne-Maree Pearse, Keiran Raine, Isabelle Rasolonjatovo, Richard Shaw, Philip Tedder, Carolyn Tregidgo, Albert J. Vilella, David C. Wedge, Gregory M. Woods, Niall Gormley, Sean Humphray, Gary Schroth, Geoffrey Smith, Kevin Hall, Stephen M.J. Searle, Nigel P. Carter, Anthony T. Papenfuss, P. Andrew Futreal, Peter J. Campbell, Fengtang Yang, David R. Bentley, Dirk J. Evers, and Michael R. Stratton. Genome Sequencing and Analysis of the Tasmanian Devil and Its Transmissible Cancer. *Cell*, 148(4):780–791, February 2012.

[23] David Owen and David Pemberton. *Tasmanian Devil: A Unique and Threatened Animal.* Allen & Unwin, 2011. OCLC: 746746907.

[24] Austin H. Patton, Matthew F. Lawrance, Mark J. Margres, Christopher P. Kozakiewicz, Rodrigo Hamede, Manuel Ruiz-Aravena, David G. Hamilton, Sebastien Comte, Lauren E. Ricci, Robyn L. Taylor,

Tanja Stadler, Adam Leaché, Hamish McCallum, Menna E. Jones, Paul A. Hohenlohe, and Andrew Storfer. A transmissible cancer shifts from emergence to endemism in Tasmanian devils. *Science*, 370(6522), December 2020. Publisher: American Association for the Advancement of Science Section: Research Article.

[25] D. Pemberton. *Social Organisation and Behaviour of the Tasmanian Devil, Sarcophilus Harrisii.* phd, University of Tasmania, July 1990.

[26] Ruth J. Pye, David Pemberton, Cesar Tovar, Jose M. C. Tubio, Karen A. Dun, Samantha Fox, Jocelyn Darby, Dane Hayes, Graeme W. Knowles, Alexandre Kreiss, Hannah V. T. Siddle, Kate Swift, A. Bruce Lyons, Elizabeth P. Murchison, and Gregory M. Woods. A second transmissible cancer in Tasmanian devils. *Proceedings of the National Academy of Sciences*, 113(2):374–379, January 2016.

[27] Tracey Catherine Russell. *An Investigation into Factors Affecting Breeding Success in the Tasmanian Devil (Sarcophilus Harrisii).* Thesis, The University of Sydney, December 2017. Accepted: 2018-08-06.

[28] Sami Salkosuo. Coordinate conversions made easy, August 2007. Library Catalog: www.ibm.com.

[29] Andrew Storfer, Paul A. Hohenlohe, Mark J. Margres, Austin Patton, Alexandra K. Fraik, Matthew Lawrance, Lauren E. Ricci, Amanda R. Stahlke, Hamish I. McCallum, and Menna E. Jones. The devil is in the details: Genomics of transmissible cancers in Tasmanian devils. *PLOS Pathogens*, 14(8):e1007098, August 2018.

[30] Cesar Tovar, Ruth J. Pye, Alexandre Kreiss, Yuanyuan Cheng, Gabriella K. Brown, Jocelyn Darby, Roslyn C. Malley, Hannah V. T. Siddle, Karsten Skjødt, Jim Kaufman, Anabel Silva, Adriana Baz Morelli, Anthony T. Papenfuss, Lynn M. Corcoran, James M. Murphy, Martin J. Pearse, Katherine Belov, A. Bruce Lyons, and Gregory M. Woods. Regression of devil facial tumour disease following immunotherapy in immunised Tasmanian devils. *Scientific Reports*, 7(1):43827, March 2017. Number: 1 Publisher: Nature Publishing Group.

[31] Lauren C. White, Frédérik Saltré, Corey J. A. Bradshaw, and Jeremy J. Austin. High-quality fossil dates support a synchronous, Late Holocene extinction of devils and thylacines in mainland Australia. *Biology Letters*, 14(1):20170642, January 2018. Publisher: Royal Society.

[32] Gregory M Woods, Samantha Fox, Andrew S Flies, Cesar D Tovar, Menna Jones, Rodrigo Hamede, David Pemberton, A Bruce Lyons, and Silvana S Bettiol. Two Decades of the Impact of Tasmanian Devil Facial Tumor Disease. *Integrative and Comparative Biology*, 58(6):1043–1054, December 2018.

Appendix A

Code Listings

A.1 Cell

```
/**
 * Cell
 */
public class Cell implements Serializable, Comparable<Cell> {

  int id;
  HashMap<String, Double> cellData;
  List<Integer> neighbourCells;
  HashMap<Integer, Double> neighbourDistance;
  HashMap<Integer, Double> neighbourRelativeProbability;
  double carryingCapacity;
  boolean isDFTDInArea;
  double totalNeighbourProbability;
  double baitVaccineAmount;
  boolean isBaitAllowed;
  boolean isInjectionVaccineTrapAllowed;
  List<Trap> traps;
  CustomDistribution neighbourDistribution;

  List<Devil>[][] devils;

    /**
     * Default constructor
     */
    public Cell(int id) {
      this.id = id;
      this.cellData = new HashMap<String, Double>();
      this.neighbourCells = new ArrayList<>();
      this.neighbourDistance = new HashMap<Integer, Double>();
      this.neighbourRelativeProbability = new HashMap<Integer, Double>();
      this.carryingCapacity = 2.0;
      this.isDFTDInArea = false;
      this.totalNeighbourProbability = 0.0;
      this.baitVaccineAmount = 0.0;
      this.isBaitAllowed = false;
      this.isInjectionVaccineTrapAllowed = false;
      this.traps = new ArrayList<>();
      this.neighbourDistribution = null;
      this.devils = new ArrayList[SexList.values().length][6];
      for(int i = 0; i < SexList.values().length; i++) {
        for(int j = 0; j < 6; j++) {
          this.devils[i][j] = new ArrayList<>();
        }
      }
    }

    public Cell(int id, double carryingCapacity) {
      this.id = id;
      this.cellData = new HashMap<String, Double>();
      this.neighbourCells = new ArrayList<>();
      this.neighbourDistance = new HashMap<Integer, Double>();
      this.neighbourRelativeProbability = new HashMap<Integer, Double>();
      this.carryingCapacity = carryingCapacity;
      this.isDFTDInArea = false;
      this.totalNeighbourProbability = 0.0;
      this.baitVaccineAmount = 0.0;
      this.isBaitAllowed = false;
      this.isInjectionVaccineTrapAllowed = false;
      this.traps = new ArrayList<>();
      this.neighbourDistribution = null;
      this.devils = new ArrayList[SexList.values().length][6];
      for(int i = 0; i < SexList.values().length; i++) {
        for(int j = 0; j < 6; j++) {
          this.devils[i][j] = new ArrayList<>();
        }
      }
    }


    /**
     * Get ID of the cell
     * @return cell ID
     */
    public int getID()
    {
      return this.id;
    }


    /*
     * Set of functions to control the amount of bait vaccines in a Grid Cell
     *
     */
    public void setBaitAllowed(boolean baitState) {
      this.isBaitAllowed = baitState;
    }
    public boolean isBaitAllowed() {
      return this.isBaitAllowed;
    }
    public void setBaitVaccineAmount(double amount) {
      this.baitVaccineAmount = max(0.0,amount);
    }
    public void addBaitVaccineAmount(double amount) {
      this.baitVaccineAmount += amount;
    }
    public void subtractBaitVaccineAmount(double amount) {
      this.baitVaccineAmount = max(0.0, this.baitVaccineAmount-amount);
    }
    public double getBaitVaccineAmount() {
      return this.baitVaccineAmount;
    }
    public void updateBaitVaccineAmount(double decayRatePerDay) {
      this.baitVaccineAmount = max(0.0, this.baitVaccineAmount-decayRatePerDay);
    }


    /*
     * Set of functions to control Traps used for Injection Vaccine
     */
    public void setTrapAllowed(boolean trapState) {
      this.isInjectionVaccineTrapAllowed = trapState;
    }
    public boolean isTrapAllowed() {
      return this.isInjectionVaccineTrapAllowed;
    }
    public void setTraps(int numberOfTraps) {
      for(int i = 0; i < numberOfTraps; i++) {
        Trap trap = new Trap(this.id);
        this.traps.add(trap);
      }
    }
    public void removeTraps() {
      for(Trap trap : this.traps) {
        trap.releaseDevil();
      }
```

```java
    this.traps.clear();
  }
  public List<Trap> getTraps() {
    return this.traps;
  }
  public boolean trapDevilIfTrapFree(Devil devil) {
    boolean trapSuccessful = false;

    for(Trap trap : this.traps) {
      if(trap.isTrapFree()) {
        trap.trapDevil(devil);
        trapSuccessful = true;
        break;
      }
    }

    return trapSuccessful;
  }

  /**
   * Stores a new key and value pair in the cell. If the key already exists then the value is overwritten by the new value.
   * @param key the key to enter, or if it exists the key whichs value should be overwritten
   * @param value the value to associate with the key
   */
  public void setValue(String key, Double value)
  {
    this.cellData.put(key, value);
  }


  public void addNeighbour(int id)
  {
    this.neighbourCells.add(id);
  }

  public void addNeighbourDistance(int id, double distance)
  {
    this.neighbourDistance.put(id, distance);
  }

  public void addDevil(Devil devil)
  {
    int ageClass = (int)devil.getAgeClass(devil.getAge());
    this.devils[devil.sex.ordinal()][ageClass-1].add(devil);
  }

  public void removeDevil(Devil devil) {
    this.devils[devil.sex.ordinal()][(int)devil.getAgeClass(devil.getAge())-1].remove(devil);
  }

  public List<Devil> getDevils() {
    List<Devil> devilList = new ArrayList<>();

    for(int i = 0; i < this.devils.length; i++) {
      for(int j = 0; j < this.devils[i].length; j++) {
        devilList.addAll(this.devils[i][j]);
      }
    }

    return devilList;
  }

  public void setCarryingCapacity(double capacity)
  {
    this.carryingCapacity = capacity;
  }
  public void setIsDFTDInArea(boolean newValue)
  {
    this.isDFTDInArea = newValue;
  }




  /**
   * Return the value for the key entered, if the key does not exists returns NULL
   * @param key they key whichs value should be retrieved
   * @return returns the value for the key, or NULL if non-existent
   */
  public Double getValue(String key)
  {
    return this.cellData.get(key);
  }


  public List<Integer> getNeighbourCells()
  {
    return this.neighbourCells;
  }

  public double getNeighbourDistance(int id)
  {
    return this.neighbourDistance.get(id);
  }

  public double getCarryingCapacity()
  {
    return this.carryingCapacity;
  }

  public boolean isDFTDInArea()
  {
    return this.isDFTDInArea;
  }

  public double getDensity()
  {
    double numDevils = this.getNumberOfDevilsInCell();

    return zidz(numDevils,this.carryingCapacity);
  }

  public int getNumberOfDevilsInCell() {
    int numDevils = 0;
    for(int i = 0; i < this.devils.length; i++) {
      for(int j = 0; j < this.devils[i].length; j++) {
        numDevils += this.devils[i][j].size();
      }
    }
    return numDevils;
  }




  public boolean equals(Cell compare)
  {
    return (this.id == compare.id
        && this.cellData == compare.cellData
        && this.neighbourCells == compare.neighbourCells
        && this.neighbourDistance == compare.neighbourDistance
        && this.carryingCapacity == compare.carryingCapacity
        && this.isDFTDInArea == compare.isDFTDInArea);
  }

@Override
public String toString() {
  return "ID: " + String.valueOf(this.id) + "\n"
      + "\t" + "CellData: " + this.cellData.toString() + "\n"
      + "\t" + "Neighbours: " + this.neighbourCells.toString() + "\n"
      + "\t" + "Neighbour Distances: " + this.neighbourDistance.toString() + "\n"
      + "\t" + "Carryin Capacity: " + String.valueOf(this.carryingCapacity) + "\n"
      + "\t" + "Is DFTD In Area: " + String.valueOf(this.isDFTDInArea) + "\n"
      + "\t" + "Bait Vaccine Amount: " + String.valueOf(this.baitVaccineAmount) + "\n";
}

public String summarizeCell() {
  return "ID:" + String.valueOf(this.id) + ";"
      + "DFTDInArea:" + String.valueOf(this.isDFTDInArea) + ";"
      + "TotalDevilPopulation:" + String.valueOf(this.getNumberOfDevilsInCell()) + ";"
      + "TotalDevilsSusceptible:" + String.valueOf(this.devilsSus()) + ";"
      + "TotalDevilsExposed:" + String.valueOf(this.devilsExp()) + ";"
      + "TotalDevilsInfectious:" + String.valueOf(this.devilsInf()) + ";"
      + "TotalDevilsRecovered:" + String.valueOf(this.devilsRec()) + ";"
      + "TotalDevilsImmunized:" + String.valueOf(this.devilsImm()) + ";"
      + "BaitVaccineAmount:" + String.valueOf(this.baitVaccineAmount) + "|"
      ;
}

public SingleTimeCellRecord createCellRecord(double time)
{
  double cellPop = this.getNumberOfDevilsInCell();
  double cellSus = this.devilsSus();
  double cellExp = this.devilsExp();
  double cellInf = this.devilsInf();
  double cellRec = this.devilsRec();
  double cellImm = this.devilsImm();
  boolean isDFTDInArea = this.isDFTDInArea;

  SingleTimeCellRecord cellRecord = new SingleTimeCellRecord(time,  cellPop,  cellSus,
  cellExp,  cellInf,  cellRec,  cellImm, isDFTDInArea, this.baitVaccineAmount);

  return cellRecord;
}

private int devilsSus() {
  int total = 0;
  for(int i = 0; i < this.devils.length; i++) {
      for(int j = 0; j < this.devils[i].length; j++) {
        for(Devil d : this.devils[i][j]) {
          if(d.inState(Devil.Susceptible))
          {
            total++;
          }
        }
      }

    }
  return total;
}

private int devilsExp() {
  int total = 0;
  for(int i = 0; i < this.devils.length; i++) {
      for(int j = 0; j < this.devils[i].length; j++) {
        for(Devil d : this.devils[i][j]) {
          if(d.inState(Devil.Exposed))
          {
            total++;
          }
        }
      }

    }
  return total;
}

private int devilsInf() {
  int total = 0;
  for(int i = 0; i < this.devils.length; i++) {
      for(int j = 0; j < this.devils[i].length; j++) {
        for(Devil d : this.devils[i][j]) {
          if(d.inState(Devil.Infectious))
          {
            total++;
          }
        }
      }

    }
  return total;
}

private int devilsRec() {
  int total = 0;
  for(int i = 0; i < this.devils.length; i++) {
      for(int j = 0; j < this.devils[i].length; j++) {
        for(Devil d : this.devils[i][j]) {
          if(d.inState(Devil.Recovered))
          {
            total++;
          }
        }
      }

    }
  return total;
}

private int devilsImm() {
  int total = 0;
  for(int i = 0; i < this.devils.length; i++) {
      for(int j = 0; j < this.devils[i].length; j++) {
        for(Devil d : this.devils[i][j]) {
          if(d.inState(Devil.Immunized))
          {
            total++;
          }
        }
      }

    }
  return total;
}

@Override
public int compareTo(Cell compareCell) {
  return this.id - compareCell.id;
}

/**
 * This number is here for model snapshot storing purpose<br>
 * It needs to be changed when this class gets changed
 */
private static final long serialVersionUID = 1000001L;
}
```

A.2 Devil Data Container

```java
/**
 * DevilDataContainer
 */
public class DevilDataContainer implements Serializable {
  /**
   * This number is here for model snapshot storing purpose<br>
   * It needs to be changed when this class gets changed
   */
  private static final long serialVersionUID = 2L;

  final static short SUSCEPTIBLE = (1 << 0);      // 1
  final static short EXPOSED =  (1 << 1);       // 2
  final static short INFECTIOUS =  (1 << 2);      // 4
  final static short RECOVERED =  (1 << 3);      // 8
  final static short IMMUNIZED =  (1 << 4);      // 16
  final static short AGECLASS1 =  (1 << 5);   // 32
  final static short AGECLASS2 =  (1 << 6);   // 64
  final static short AGECLASS3 =  (1 << 7);   // 128
  final static short AGECLASS4 =  (1 << 8);   // 256
  final static short AGECLASS5 =  (1 << 9); // 512
  final static short AGECLASS6 =  (1 << 10);  // 1024
  final static short MALE =  (1 << 11);   // 2048
  final static short FEMALE =  (1 << 12); // 4096
  final static short JUVENILE =  (1 << 13); // 8192
  final static short ADULT =  (1 << 14);  // 16384


  enum diseasePredicateEnum {
    SUSCEPTIBLE,
    EXPOSED,
    INFECTIOUS,
    RECOVERED,
    IMMUNIZED,
    ALLDISEASE,
    ALLNONDISEASE,
    NONE
  }
  enum ageClassPredicateEnum {
    AGECLASS1,
    AGECLASS2,
    AGECLASS3,
    AGECLASS4,
    AGECLASS5,
    AGECLASS6,
    NONE
  }
  enum sexPredicateEnum {
    MALE,
    FEMALE,
    NONE
  }

  enum lifestagePredicateEnum {
    JUVENILE,
    ADULT,
    NONE
  }


  protected List<SingleTimeContainer> allData;

    /**
     * Default constructor
     */
    public DevilDataContainer() {
```

```java
    this.allData = new ArrayList<>();
  }
  public DevilDataContainer(List<SingleTimeContainer> allData) {
    this.allData = allData;
  }


  public void addData(double time, Main._devils_Population devils) {
    this.allData.add(new SingleTimeContainer(time, devils));
  }

  public void fillNewData(DevilDataContainer data) {
    double lastTime = 0.0;

    if(this.allData.size() > 0) {
      lastTime = this.allData.get(this.allData.size()-1).time;
    }

    for(SingleTimeContainer d : data.allData) {
      if(d.time > lastTime) {
        this.allData.add(d);
      }
    }
  }

  public void fillAllData(DevilDataContainer data) {
    this.allData.addAll(data.allData);
  }

  public List<Pair<Double, Double>> getDataRaw(diseasePredicateEnum disease, ageClassPredicateEnum age, sexPredicateEnum sex, lifestagePredicateEnum lifestage) {
    List<Pair<Double, Double>> returnData = new ArrayList<>();

    for(SingleTimeContainer singleData : this.allData) {
      double count = this.processSingleTimeData(singleData, disease, age, sex, lifestage);
      returnData.add(new Pair(singleData.time, count));
    }

    return returnData;
  }

  public List<Pair<Double,Double>> getDataPopulationPercentage(diseasePredicateEnum disease, ageClassPredicateEnum age, sexPredicateEnum sex, lifestagePredicateEnum lifestage) {
    List<Pair<Double, Double>> returnData = new ArrayList<>();

    for(SingleTimeContainer singleData : this.allData) {
      int count = this.processSingleTimeData(singleData, disease, age, sex, lifestage);
      double percentage = zidz(count, singleData.devilData.length);
      returnData.add(new Pair(singleData.time, percentage));
    }

    return returnData;
  }

  public double getDataForTime(double time, diseasePredicateEnum disease, ageClassPredicateEnum age, sexPredicateEnum sex, lifestagePredicateEnum lifestage) {
    for(SingleTimeContainer singleData : this.allData) {
      if(singleData.time == time) {
        return this.processSingleTimeData(singleData, disease, age, sex, lifestage);
      }
    }
    return 0.0;
  }

  public double getPercentDataForTime(double time, diseasePredicateEnum disease, ageClassPredicateEnum age, sexPredicateEnum sex, lifestagePredicateEnum lifestage) {
    for(SingleTimeContainer singleData : this.allData) {
      if(singleData.time == time) {
        double count = this.processSingleTimeData(singleData, disease, age, sex, lifestage);
        double percentage = zidz(count, singleData.devilData.length);
        return percentage;
      }
    }
    return 0.0;
  }


  private int processSingleTimeData(SingleTimeContainer singleData, diseasePredicateEnum disease, ageClassPredicateEnum age, sexPredicateEnum sex, lifestagePredicateEnum lifestage) {
    int count = 0;
    if(disease.equals(diseasePredicateEnum.ALLDISEASE)) {
      count += this.processSingleTimeData(singleData, diseasePredicateEnum.EXPOSED, age, sex, lifestage);
      count += this.processSingleTimeData(singleData, diseasePredicateEnum.INFECTIOUS, age, sex, lifestage);
    }
    else if(disease.equals(diseasePredicateEnum.ALLNONDISEASE)) {
      count += this.processSingleTimeData(singleData, diseasePredicateEnum.SUSCEPTIBLE, age, sex, lifestage);
      count += this.processSingleTimeData(singleData, diseasePredicateEnum.RECOVERED, age, sex, lifestage);
      count += this.processSingleTimeData(singleData, diseasePredicateEnum.IMMUNIZED, age, sex, lifestage);
    }
    else {
      short mask = getMask(disease, age, sex, lifestage);

        for(int i = 0; i < singleData.devilData.length; i++) {
          if((singleData.devilData[i] & mask) == mask) {
            count++;
          }
        }
    }


    return count;
  }


  private short getMask(diseasePredicateEnum disease, ageClassPredicateEnum age, sexPredicateEnum sex, lifestagePredicateEnum lifestage) {
    short mask = 0;

  switch(disease) {
    case SUSCEPTIBLE:
      mask |= SUSCEPTIBLE;
      break;
    case EXPOSED:
      mask |= EXPOSED;
      break;
    case INFECTIOUS:
      mask |= INFECTIOUS;
      break;
    case RECOVERED:
      mask |= RECOVERED;
      break;
    case IMMUNIZED:
      mask |= IMMUNIZED;
      break;
    default:
      break;
  }
  switch(age) {
      case AGECLASS1:
        mask |= AGECLASS1;
        break;
      case AGECLASS2:
        mask |= AGECLASS2;
        break;
      case AGECLASS3:
        mask |= AGECLASS3;
        break;
      case AGECLASS4:
        mask |= AGECLASS4;
        break;
      case AGECLASS5:
        mask |= AGECLASS5;
        break;
      case AGECLASS6:
        mask |= AGECLASS6;
        break;
      default:
        break;
  }
  switch(sex) {
    case MALE:
      mask |= MALE;
      break;
    case FEMALE:
      mask |= FEMALE;
      break;
    default:
      break;
  }
  switch(lifestage) {
    case JUVENILE:
      mask |= JUVENILE;
      break;
    case ADULT:
      mask |= ADULT;
      break;
    default:
      break;
  }

  return mask;
  }

  public List<String> toCSV() {
    List<String> outputList = new ArrayList<>();

    for(SingleTimeContainer record : this.allData) {
      outputList.add(record.toCSV());
    }

    return outputList;
  }

@Override
public String toString() {
  String returnStr = "";

  for(SingleTimeContainer data : this.allData) {
    returnStr += data.toString();
    returnStr += "\n";
  }

  return returnStr;
}


```

```java
public boolean equals(DevilDataContainer compare) {
  if(this.allData.size() != compare.allData.size()) {
    traceln("DevilDataContainer equals: sizes different!");
    return false;
  }

  Iterator<SingleTimeContainer> iter1 = this.allData.iterator();
  Iterator<SingleTimeContainer> iter2 = compare.allData.iterator();

  while(iter1.hasNext() && iter2.hasNext()) {
    if(!(iter1.next().equals(iter2.next()))) {
      traceln("DevilDataContainer equals: two elements not equals");
      return false;
    }
  }

  return true;
}



/*
 * SingleTimeContainer captures all data for a single timepoint, including all devil data
 */
private class SingleTimeContainer implements Serializable {
  /**
   * This number is here for model snapshot storing purpose<br>
   * It needs to be changed when this class gets changed
   */
  private static final long serialVersionUID = 3L;

  double time;
  short[] devilData;
  int[] devilCellIDs;

  private SingleTimeContainer() {
    this.time = 0.0;
    this.devilData = new short[0];
    this.devilCellIDs = new int[0];
  }

  private SingleTimeContainer(double time, Main._devils_Population devils) {
    this.time = time;
    this.devilData = new short[devils.size()];
    this.devilCellIDs = new int[devils.size()];

    int idx = 0;
    for(Devil devil : devils) {
      this.devilData[idx] = setData(devil);
      this.devilCellIDs[idx] = devil.homeCell;
      idx++;
    }
  }

  private SingleTimeContainer(double time, short[] data, int[] cellIDs) {
    this.time = time;
    this.devilData = data;
    this.devilCellIDs = cellIDs;
  }

  private short setData(Devil devil) {
    short data = 0;

    switch(devil.DiseaseStateChart.getActiveSimpleState()) {
      case Susceptible:
        data |= SUSCEPTIBLE;
        break;
      case Exposed:
        data |= EXPOSED;
        break;
      case Infectious:
        data |= INFECTIOUS;
        break;
      case Recovered:
        data |= RECOVERED;
        break;
      case Immunized:
        data |= IMMUNIZED;
        break;
      default:
        break;
    }

    switch((int)Math.round(devil.getAgeClass(devil.getAge()))) {
      case 1:
        data |= AGECLASS1;
        break;
      case 2:
        data |= AGECLASS2;
        break;
      case 3:
        data |= AGECLASS3;
        break;
      case 4:
        data |= AGECLASS4;
        break;
      case 5:
        data |= AGECLASS5;
        break;
      case 6:
        data |= AGECLASS6;
        break;
    }

    if(devil.sex.equals(SexList.FEMALE)) {
      data |= FEMALE;
    }
    else {
      data |= MALE;
    }

    if(devil.LifestageStatechart.isStateActive(devil.Juvenile)) {
      data |= JUVENILE;
    }
    else {
      data |= ADULT;
    }

    return data;
  }


  public String toCSV() {
    String output = "";

    output += this.time;
    output += ",";

    for(int i = 0; i < this.devilData.length; i++) {
      output += this.shortToCSV(this.devilData[i]);
      output += this.devilCellIDs[i];

      if(i < this.devilData.length-1) {
        output += "|";
      }
    }
```

```java

    return output;
  }

  private String shortToCSV(short data) {
    String str = "";

    //Decode disease state from bit flag
    if((data & SUSCEPTIBLE) > 0) {
      str += "Susceptible;";
    }
    else if((data & EXPOSED) > 0) {
      str += "Exposed;";
    }
    else if((data & INFECTIOUS) > 0) {
      str += "Infectious;";
    }
    else if((data & RECOVERED) > 0) {
      str += "Recovered;";
    }
    else if((data & IMMUNIZED) > 0) {
      str += "Immunized;";
    }

    //Decode AgeClass from bit flag
    if((data & AGECLASS1) > 0) {
      str += "AgeClass1;";
    }
    else if((data & AGECLASS2) > 0) {
      str += "AgeClass2;";
    }
    else if((data & AGECLASS3) > 0) {
      str += "AgeClass3;";
    }
    else if((data & AGECLASS4) > 0) {
      str += "AgeClass4;";
    }
    else if((data & AGECLASS5) > 0) {
      str += "AgeClass5;";
    }
    else if((data & AGECLASS6) > 0) {
      str += "AgeClass6;";
    }

    //Decode sex from bit flag
    if((data & MALE) > 0) {
      str += "Male;";
    }
    else if((data & FEMALE) > 0) {
      str += "Female;";
    }

    //Decode Lifestage from bit flag
    if((data & JUVENILE) > 0) {
      str += "Juvenile;";
    }
    else if((data & ADULT) > 0) {
      str += "Adult;";
    }

    return str;
  }


  @Override
  public String toString() {
    String returnStr = "";

```

```
      returnStr += String.valueOf(this.time) + ", [" ;

      for(int i = 0; i < this.devilData.length; i++) {
        returnStr += String.valueOf(this.devilData[i]);
        returnStr += ", ";
      }

      returnStr += "]";

      returnStr += ", [";

      for(int i = 0; i < this.devilCellIDs.length; i++) {
        returnStr += String.valueOf(this.devilCellIDs[i]);
        returnStr += ", ";
      }
      returnStr += "]";

      return returnStr;
    }

    public boolean equals(SingleTimeContainer compare) {
      if(this.time == compare.time) {
      }
      else {
        traceln("SingleTimeContainer equals: times different!");
        return false;
      }

      if(Arrays.equals(this.devilData, compare.devilData) && Arrays.equals(this.devilCellIDs
    , compare.devilCellIDs)) {

      }
      else {
        traceln("SingleTimeContainer equals: arrays are different!");
        return false;
      }

      return true;
    }
  }
}
```

A.3 Server Connector

```java
/**
 * GrapherServerConnection
 */
public class GrapherServerConnection implements Serializable {
  String socketname;
  UUID experimentID;
  AtomicInteger numWriters;
  int framenum;

    /**
     * Default constructor
     */
    public GrapherServerConnection(String socketname, UUID experimentID) {
      this.socketname = socketname;
      this.experimentID = experimentID;
      this.numWriters = new AtomicInteger();
      this.framenum = 0;
    }

    public GrapherServerConnection(String socketname) {
      this.socketname = socketname;
      this.experimentID = UUID.randomUUID();
      this.numWriters = new AtomicInteger();
      this.framenum = 0;
    }


    public void shutdownServer() {
      try (AFUNIXSocket socket = AFUNIXSocket.newInstance()) {
        try {
          socket.connect(new AFUNIXSocketAddress(new File(this.socketname)));
        }
        catch (SocketException e) {
          System.out.println("Cannot connect to Graphing Server!");
        }


        OutputStream output = socket.getOutputStream();
        PrintWriter writer = new PrintWriter(output, true);

        writer.println(GraphingServerCommands.shutdown.toString());


      } catch (UnknownHostException ex) {
          return;
      } catch (IOException ex) {
        return;
        }

      //Connect twice to ensure shutdown of server
      try (AFUNIXSocket socket = AFUNIXSocket.newInstance()) {
        try {
          socket.connect(new AFUNIXSocketAddress(new File(this.socketname)));
        }
        catch (SocketException e) {
          System.out.println("Cannot connect to Graphing Server!");
        }

        OutputStream output = socket.getOutputStream();
        PrintWriter writer = new PrintWriter(output, true);

        writer.println(GraphingServerCommands.shutdown.toString());

      } catch (UnknownHostException ex) {
          return;
      } catch (IOException ex) {
        return;
        }
    }


    public boolean createAnimationOnServer(String experimentResultDir) {
      while (this.numWriters.get() > 0) {
        try {
          Thread.sleep(10);
        }
        catch (InterruptedException e) {
          traceln("Can't sleep thread while waiting for all data to be written to server! Creating animation now, some frames might be missing");
          break;
        }
      }

      try (AFUNIXSocket socket = AFUNIXSocket.newInstance()) {
        try {
          socket.connect(new AFUNIXSocketAddress(new File(this.socketname)));
        }
        catch (SocketException e) {
          System.out.println("Cannot connect to Graphing Server!");
        }

        OutputStream output = socket.getOutputStream();
        PrintWriter writer = new PrintWriter(output, true);

        //send the 'createAnimation' command to process all frames into a single animation
        //this deletes all individual frames
        writer.println(GraphingServerCommands.createAnimations.toString());
        writer.println(this.experimentID.toString());
        writer.println(experimentResultDir);

      } catch (UnknownHostException ex) {
          return false;
      } catch (IOException ex) {
        return false;
        }
      return true;
    }

    public void writeDataToServer(String data) {
      new WriteThread(this, this.socketname, data, this.framenum).start();
      this.framenum++;
    }

    public boolean isServerAvailable() {
      try (AFUNIXSocket socket = AFUNIXSocket.newInstance()) {
        try {
          socket.connect(new AFUNIXSocketAddress(new File(this.socketname)));
        }
        catch (SocketException e) {
          return false;
        }

        OutputStream output = socket.getOutputStream();
        PrintWriter writer = new PrintWriter(output, true);

        //send the 'test server' command to check if the server is still alive
        //server will send 'server available' back if available
        writer.println(GraphingServerCommands.testServer.toString());

        InputStream input = socket.getInputStream();

          BufferedReader reader = new BufferedReader(new InputStreamReader(input));

      String line = reader.readLine();

      if(line.equals("server available"))
      {
       return true;
      }

    } catch (UnknownHostException ex) {
      return false;
    } catch (IOException ex) {
     return false;
     }
    return false;
   }



@Override
public String toString() {
 return super.toString();
}

/**
 * This number is here for model snapshot storing purpose<br>
 * It needs to be changed when this class gets changed
 */
private static final long serialVersionUID = 1L;



private class WriteThread extends Thread {
 private String data;
 private String socketname;
 GrapherServerConnection server;
 int framenum;

 public WriteThread(GrapherServerConnection server, String socketname, String data, int framenum) {
  this.server = server;
  this.socketname = socketname;
  this.data = data;
  this.framenum = framenum;
 }

 @Override
 public void run() {
  this.server.numWriters.incrementAndGet();
  try (AFUNIXSocket socket = AFUNIXSocket.newInstance()) {
    try {
     socket.connect(new AFUNIXSocketAddress(new File(this.socketname)));
    }
    catch (SocketException e) {
     System.out.println("Cannot connect to Graphing Server!");
    }

    OutputStream output = socket.getOutputStream();
    PrintWriter writer = new PrintWriter(output, true);

    //send the 'processCellData' command to enable data processing for this connection
    writer.println(GraphingServerCommands.processCellData.toString());
    writer.println(this.framenum);
    writer.println(this.data);

   } catch (UnknownHostException ex) {
     System.out.println("Server not found: " + ex.getMessage());
   } catch (IOException ex) {
      System.out.println("IO excetption: " + ex.getMessage());
    }
```

```
      this.server.numWriters.decrementAndGet();
    }
  }
}
```