The River Engine: A Framework for Simulating Population Dynamics in Rivers Using an Agent-Based Model

Martin Hanzel

Supervised by Dr. Frithjof Lutscher

Submitted in partial fulfillment of the

Honours Biomedical Sciences degree program.

University of Ottawa

Chapter 1

Abstract

Agent-based models (ABMs) have been increasingly used in recent years to simulate and predict real-world phenomena. Increases in computing power have made ABMs widely accessible. In contrast to traditional models that use differential or difference equations, ABMs provide the option to introduce distinct behaviours to each individual. As a result, a rich set of interactions and remarkable behaviours emerge [1]. We present *The River Engine*, a novel agent-based model that simulates fish and resources in a freshwater ecosystem and provides a framework for future investigators who study population dynamics in river networks. Using this model, we were able to observe the effects of various perturbations on a freshwater river ecosystem, such as the effect of nutrient depletion and immigration on fish populations.

Chapter 2

Introduction to Agent-Based Models

Mathematical models are a key tool for predicting and understanding the natural world. Traditionally, mathematicians and biologists have modeled dynamical processes using suites of differential or discrete-time equations, which are often limited to providing a very high-level approximation of some system. These methods become unwieldy when one wants to introduce some degree of stochasticity or heterogeneity to a system, or observe the behaviour of individuals on a small scale [2]. In recent years**,** **Agent-based models** (ABMs) have emerged to address these shortcomings.

Badham and Osborn [3] precisely describe the features of an ABM:

- An ABM models the behaviour of "heterogeneous **agents**," with each agent having its own, possibly unique, set of characteristics and goals.
- An ABM has some representation of an environment in which the agents live. The environment may by dynamic.
- An ABM applies a finite set of interactions to its agents or environment. Interactions are rules that define how agents affect each other, and how they affect the environment around them.

A search on Google Scholar on the keywords "agent-based modeling" reveals that ABMs are applied in a wide variety of fields: sociology [4], biomedicine [5], ecology [6], fire planning [7], and urban planning [8], to name a few. The exponential growth of computing power [9] has enabled ABMs, which were once considered to be too computationally expensive, to become useful tools in modeling real-world systems. In fact, useful ABMs appear in unexpected places, such as video games. In 2005, a bug in the popular multi-player game *World of Warcraft* resulted in the spread of a virtual "disease" among in-game characters. The event, known as the "corrupted blood incident," has been scrutinized by academics and has been presented as a valid model for epidemiology and biological warfare [10,11].

In this thesis, we develop an ABM for studying the dynamics of a fish population in a river network. A particular feature of river networks is their topology. The layout of streams imposes restrictions on the movement of fish, and every fish has a significant degree of autonomy over its own movements. These two characteristics – fish-environment interactions and the independence of fish – are well suited to an ABM.

Tracking Fish in Rivers

Studying wild fish in their natural habitat is difficult. Presently, methods commonly used to track fish rely on radio-frequency or ultrasonic transmissions [12–14] emitted from electronic chips embedded inside or attached to the skin of a fish [A]. These methods have certain limitations. Freshwater fish fitted with tracking devices must pass reasonably close to a receiver, which must be set up beforehand. Larger animals, such as sharks [15] or sea turtles [16], are commonly fitted with GPS transmitters that relay their exact position through a satellite connection, though they work only when the animal surfaces. In any case, these methods cannot provide a fish's position or movement continuously, and so have primarily been used to study migratory fish that periodically pass predictable landmarks. Particularly, much work has been done on tracking migratory salmon [13,17–19].

Tracking systems often have to be set up to make specific observations. For example, if a scientist wanted to observe how fish cross a dam [20,21], he or she would have to capture and mark individuals with radio transmitters and set up radio receivers specifically for that experiment. Understandably, this process is time-consuming and may be expensive, particularly in remote areas. The purpose-built nature of these tracking systems may discourage their use in future work.

Currently, no robust method exists of tracking the day-to-day lives of fish in their ecosystem. This is problematic, as scientists often construct generalized mathematical models that predict how a

[A] These methods, involving chips and fish, should not be confused with **fish and chips**, a tasty pub-fare dish of English origin [40].

population of fish will respond to ecological perturbations. As discussed in the previous section, these models have shortcomings. A strategic model that accounts for the behaviour of individual fish within a population and for inter-fish interactions might be able to answer ecological questions more accurately.

Agent-based models seem to address these concerns perfectly:

- ABMs model a population on an individual level, and so each fish in a population can be considered independently;
- ABMs can simulate the movement of a fish in continuous space, and are able to show how fish move outside the range of a physical receiver; and
- ABMs allow interactions between fish and their environment to be expressed in simple rules, and so can be easily re-used for a different purpose by altering the rules or tweaking parameter values. In some cases, ABMs allow these interactions to be described more accurately [22].

Our objective is to create an ABM that simulates movement, reproduction, and feeding behaviour or fish in an aquatic ecosystem. Such a model will be immensely useful for predicating the fate of fish populations in regions where it is inconvenient or impossible to observe them, for instance in arctic regions, where rivers are frozen for a large part of the year.

The Project

We present a novel software package, called *The River Engine* [B], that provides a framework for ecologists to predict the behaviour and distribution of fish in a river network [C], and to inspect the effects of perturbations on an aquatic ecosystem. *The River Engine* simulates such an ecosystem and its member flora/fauna using an ABM that may easily be tweaked and extended to fit a number of scenarios.

In this work, we discuss the basic mechanics of *The River Engine* and apply it to study the interactions between a single species of fish and their food source. We analyze how fish populations

[B] *The River Machine* was, unfortunately, taken [41] by a similar, but unrelated project that studies the effect of river hydrogeomorphology and fluid dynamics on fish navigation.

[C] One may say that it predicts a *poisson distribution*.

respond to overcrowding, immigration, and nutrient deficit. Of course, this kind of analysis is only the tip of the iceberg of *The River Engine*'s full power. We built *The River Engine* with the goal of being able to easily program new interactions and we hope that future investigators will be able to use it to make new insights into the dynamics of lotic ecosystems.

Chapter 3

Methods

Software

The River Engine was developed in the Java programming language [23] using version 8 of the Oracle Java Development Kit [24]. Third-party software libraries were used to implement certain features of the application: Apache Commons CLI [25], Apache Commons IO [26], Apache Commons Lang [27], Google GSON [28], Ini4J [29], JGraphT [30], JUnit 4 [31]. Source code for *The River Engine* is freely available [32].

The Model

As an agent-based model, *The River Engine* is conceptually divided into three parts:

- A list of agents (fish);
- A representation of a physical environment where the fish reside (the river network); and
- A finite list of rules defining how agents interact with each other and their environment.

We give a detailed description of each of these components in following sections.

The state of the model at any given time is given by the set of states of all fish and the physical environment (collectively, the **river system**).

The River Engine contains additional components, such as a module that runs the simulation and a module that displays a graphical user interface to tweak parameters. A detailed description of these components is more suited to a study of computer science and will not be discussed in this work.

Computer-savvy readers may refer to *The River Engine*'s source code [32] to learn about the technical details of *The River Engine*'s implementation.

Agents

In *The River Engine*, agents represent fish that live in rivers. Fish are free to move upstream and downstream in the river, and possess a set of properties (state variables), such as a measure of stored energy. Each agent has its own independent set of properties (discussed later), and is simulated independently of any other agent. The consequence of this is that each fish can have individual behaviours that set it apart from other fish.

Environment

The River Engine's **environment** consists of a directed graph that represents the topology of a river network. Vertices represent forks in the river network, and edges represent river runs. In this interpretation, each river run is assumed to be a straight channel, without any bends or curves in between forks. All points in the river network have constant and equal cross-sectional area. Spatial resolution is only longitudinal (i.e. upstream-downstream), and not lateral. We also suppose that the river network itself is immutable – it does not change during a simulation. In *The River Engine* jargon, the term **river node** represents a vertex in the directed graph, and **river arc** represents an edge.

The River Engine's environment may also contain data about certain characteristics of the river network, for instance, the availability of nutrients at a certain location. *The River Engine* imposes no limit on the amount of additional data or the size of the river network. Naturally, large amounts of data increase computation time and storage size. This additional information *is* mutable. For example, as fish feed, the local nutrient quantity decreases.

In this work, the only property of a river system's environment that is considered is the quantity of nutrients. Keeping track of the quantity of nutrients at any given point is computationally difficult, and so *The River Engine* makes a discrete-space approximation. It is important to understand this implementation in order to mathematically analyze the nutrient dynamics. The data structure representing a river network contains a number of “feeding points,” situated at each node, and at regular intervals between nodes. Each feeding point is independent of others. *The River* Engine only keeps track of nutrient quantity at these feeding points. When fish feed, they feed from the feeding points closest to them. Any finite river network has a finite number of feeding points, and so the total number of nutrients available in the whole network is equal to the sum of nutrients available at feeding points.

Throughout this work, we use a single topology resembling a complete binary tree with 3 levels. This river network has 391 feeding points spaced at most 5 metres apart. Two arcs at the first level have length 223.607 metres each; four arcs at the second level have length 141.421 metres each; and eight arcs at the third level have length 111.803 metres each, for a total of 1907.33 metres along the entire river network. Figure 1 shows a schematic of this river network.

Interactions

The River Engine updates the state of the river system in discrete time steps. It applies a sequence of rules to the state of the river system at time t, denoted by S_t, to compute the state of the system at time $t + 1$. A “rule” (called an “**influence**” in *The River Engine* jargon) is a function that maps S_t to S_{t+1}. This function includes movement and population dynamics of agents, and possibly other, user-defined processes. This function is stochastic so S_t becomes a stochastic variable.

Influences may depend on a set of parameters. For instance, an influence enabling logistic growth of a population depends on a carrying capacity parameter. These parameters are configurable by the user, and may even be changed while *The River Engine* is running.

Running the simulation

As mentioned above, *The River Engine* updates the state of the river system in discrete time steps. Starting from some initial state, it applies a list of influences in order to arrive at the next state. The program runs until terminated by the user, or until a desired end time is reached. The time-step between iterations plays a crucial role in determining the outcome of the simulation, similar to how, in an ODE solver, the sampling interval determines the accuracy of the solution. Larger time-steps involve less computation time, but are less accurate. As the time-step approaches zero, the model approaches continuous-time behaviour. We shall discuss the importance of the time-step in the context of this model in more detail later. Varying the time-step leads to different **emergent behaviours** in the model, which shall be analyzed further in this work.

The River Engine is stochastic. Random variables are used in many places, for example, to give fish a degree of random movement. Thus, running the simulation under identical conditions twice will produce different results [D].

A Null Model

We constructed a "null model" – a set of influences and parameters that represent a basic ecosystem. This ecosystem is fictitious and does not represent any particular real-world ecosystem. The null model serves as the null hypothesis for future analyses. Throughout this work, we will apply various perturbations to the null model by changing parameter values or altering influences, and observing what effect they have on the simulated ecosystem.

The null model applies the following influences to the state of the river system, in order, at every iteration:

1. *Random Movement***:** Each fish is given a velocity from a normal distribution with a mean of 0 m/s and standard deviation of 0.5 m/s, limited to the interval $[-4, 4]$ [E]. To avoid biasing either tail of the normal distribution, values that fall outside of this range are re-drawn. Negative velocities indicate upstream movement and positive velocities indicate downstream movement. The

[D] Unless the same pseudo-random seed is used in two simulations. *The River Engine* does not support setting the pseudo-random seed directly.

[E] This is the maximum velocity in water of an unladen fish. Velocities of laden fish were not considered, but the reader may refer to the work of Python, M. for a short discussion on the velocities of laden animals [42].

location of each fish in the river network is updated accordingly. Fish that reach an upstream or downstream terminus of the river network do not continue past the terminus. Fish that move into a fork in the river randomly select an arc that is not their own and continue their migration. Note that each fish has a 50/50 chance of moving upstream versus downstream in any iteration, and so the movement of a single fish over time constitutes an unbiased random walk.

2. *Nutrient growth*: Water-borne nutrient growth is simulated based on a logistic growth model with constant addition. The amount of nutrients at a feeding point is denoted by n. It is updated, for every feeding point, from one time step to the next based on a discretization of the logistic growth function $\frac{dn}{dt} = r\, n \left(1 - \frac{n}{k}\right) + s$, where r is the logistic growth rate, k is the logistic carrying capacity, and s is a constant addition rate. The discrete-time analogue is $n_{t+1} = n_t + \Delta t \left(r\, n_t \left(1 - \frac{n_t}{k}\right) + s\right)$, where Δt is the time-step between iterations. In the null model, $r = 1$ units/day, $k = 1$ units, $s = 0.05$ units/day. Note that nutrient amounts are given in arbitrary units. The reader may choose to equate the units to a measure of biomass.
3. *Feeding*: Fish attempt to feed on water-borne nutrients in their immediate neighbourhood. Each fish attempts to consume a total of 1.5 units of nutrients per day and feed within a radius of 10 metres. Each unit of food consumed increments a fish's internal stock of "energy." Once a fish accumulates 6 units of stored energy, it stops feeding. If there are fewer nutrients in a fish's local environment than it can consume, it consumes them all. Fish feed synchronously – a fish appearing earlier in the list of agents feeds first, and may consume all nutrients before other fish feed (though the significance of this effect can be mitigated somewhat with a sufficiently small time-step).
4. *Mortality*: Fish expend 1 unit of energy per day, thus, they must eat at least 1 unit of food per day in order to remain viable. Fish are at risk of dying if their energy falls below 3 units. If a fish has fewer than 3 units of stocked energy, the probability of survival at each iteration is $e_f/3$, where e_f

is the fish's stock of energy. Fish that have zero energy die immediately and are removed from the system.

5. *Reproduction*: At the beginning of every year, each fish produces between zero and ten offspring. The number of offspring is selected from a uniform random variable for each fish. Offspring are placed at their parent's location, but move on their own in the next iteration. Offspring behave the same way as adult fish and may reproduce at the end of their first year.

Selection of Parameter Values for the Null Model

We consulted a number of resources and studies that characterize the swim speed of fish, growth rate of water-borne nutrients, and social dynamics of river fish [33–38]. When selecting parameter values for the null model, we found that it was difficult to adapt the found parameter values to the assumptions and generalizations that we made when building *The River Engine*. Additionally, we found no resource that fully described all the parameters for a single ecosystem, and selecting parameter values from multiple sources introduced certain inconsistencies that made analysis difficult.

We decided instead to select parameter values by selecting sane "defaults" for some parameters and obtaining values for the rest via a Monte Carlo process, so that the parameter values produce a sustainable population over the long term.

We began by considering the swim speed of fish. Various references cite the critical swim speed of freshwater fish as between 0.5 and 4 metres per second (we did not favour a particular species of fish) [34,35,37]. Since fish do not swim at critical velocity all the time, we selected the value 0.5. This value represents the standard deviation or a fish's random velocity, as described in the previous section.

We then arbitrarily set the quantity of food (nutrients) that a fish needs to consume to 1 unit per day. It follows that, if there are n fish in the river, the river system must produce at least n units of food per day to sustain the population. We selected parameters r, s in the logistic growth function that satisfy this constraint, then sampled various values around r, s to see if more stable or more realistic behaviour is produced. We finally settled on $r = 1, s = 0.05$.

The remaining parameters were set in a similar fashion. We selected initial values that satisfy a reasonable steady-state approximation of the population, then tested random values close to the initial value for desired qualitative behaviour.

Chapter 4

Data Collection and Results

Fish-nutrient interactions and effects of nutrient deficit

The null model provides a good opportunity to study the interactions between fish and the nutrients that they eat. Since the movement of each fish is random, an ABM is a useful tool to simulate non-homogenous environments and the effects of overpopulation and overfeeding in certain locales. In particular, because the reproductive behaviour of fish is discrete and happens once a year in the null model, we observed population fluctuations and massive localized extinctions due to overcrowding and a resulting deficit of nutrients in the river system.

We simulated the null model for a range of parameter values for the nutrient logistic growth constant, r. In each simulation, the population of fish grows quickly. By the third generation, the population numbers in the hundreds. At the beginning of each year, after a reproductive event, the fish population spikes briefly, then plummets to a sustained size (generally under 50 fish) for the remainder of the year. Figure 2 shows the population size over time for a single simulation ($r = 1.2$). Figure 3 shows the population size over time for a range of r values.

Notably, Figure 2 shows a sustained two-cycle of post-reproduction population sizes, as well as a two-cycle of population peaks during reproductive events. These oscillations will be explored in the next section.

What determines the **sustainable population size** between reproductive events? Clearly, the population cannot grow indefinitely, since its size is limited by the replenishment of nutrients. Let us find an upper bound of the fish population size, k_f (so named since it represents the carrying capacity of fish – k represents the logistic carrying capacity of nutrients). Each feeding point produces a certain amount of nutrients over time, as governed by the logistic growth equation (see page 11):

$$\frac{dn}{dt} = r\,n\left(1 - \frac{n}{k}\right) + s$$

We shall consider the continuous-time logistic growth in this analysis. This function has a global maximum at $n = k/2$, and so the maximum rate of nutrient growth is:

$$r\frac{k}{2}\left(1 - \frac{k}{2k}\right) + s = \frac{r\,k}{4} + s = 0.3 \text{ where } r = 1, k = 1, s = 0.05$$

for a single feeding point. There are 391 feeding points in our river network, so the total maximum nutrient growth rate is 117.3 units per day for the above parameter values. Since fish must feed on 1 unit of food per day, the habitat can support up to 117 individuals. In general, this reasoning gives us a **maximum population size** $k_f = \left(\frac{r\,k}{4} + s\right) \cdot N_F$ individuals, where N_F is the number of feeding points in the river network.

This formula supposes that fish utilize all resources equally and that fish are evenly spaced throughout the river network, which is unrealistic. It is very likely that, at any given time, some fish will share feeding points, and that some feeding points will be vacant. Thus, the actual upper bound of fish population size is expected to be smaller.

As fish reproduce, they inevitably exceed the maximum population size k_f and consume all available nutrients. Consequently, the population declines. However, we found that the sustainable inter-year population size is *well* below k_f. Plotting the inter-year population size (Figure 4) versus r shows that, while increasing r increases k_f, a very small effect is actually seen on the sustainable population size (the effect is, however, quite highly correlated, with $r^2 = 0.94$ [F]).

[F] There is never any reason to consider the square of the logistic growth rate, so throughout this work, r refers to the logistic growth rate and r^2 to the correlation strength.

Clearly, another factor must be driving the sustainable population size. We ran a set of simulations for a range of parameter values for s, the constant nutrient addition rate. The inter-year population size was found to increase significantly with increasing s.

What causes s to have a more profound effect on the inter-year population? Consider the river network just after a reproductive event – there are far more fish that can be sustained, and they quickly consume the river's stock of nutrients. The quantity of nutrients approaches zero, and some fish, unable to feed, die in a matter of days. At some point, however, the population must cross some critical threshold at which the rate of nutrient replenishment equals the rate of consumption. Until this time, the quantity of nutrients remains close to zero, and so the logistic growth rate is very low. The constant growth rate s still remains. We plotted the mean inter-year population size for a gradient of s alongside another measure, $k_{fs} = 391s$, in Figure 5. k_{fs} represents the upper bound of the fish population size that can be sustained from constant nutrient addition alone (in other words, supposing zero logistic growth). The two series are highly correlated ($r^2 \approx 0.98$) and similar in magnitude, suggesting that it is the nutrient addition rate s that drives the size of the fish population between reproductive events.

This observation has important consequences. It suggests that species of fish that have short and discrete reproductive periods must live in a habitat that has a constant nutrient influx. As large quantities of nutrients flow into the habitat, a large population may be supported immediately after reproductive events when the logistic replenishment of nutrients is low.

Finding a Critical Population Size

We have seen how a sudden influx of fish from a reproductive event triggers a massive drop in the population. This raises the question: does there exist some critical size at which the population becomes unsustainable in the long term?

We modified our model to simulate immigration instead of reproduction. Fish no longer reproduce once a year, but individuals may enter the river network by immigration at a regular rate. We selected an initial immigration rate of one fish every five days and observed the population size over time (Figure 6). Immigrant fish enter the river network at a random point.

The population size stayed well below $k_f = 117.3$, but a fair bit above $k_{fs} = 19.55$ for parameters $r = 1, k = 1, s = 0.05$. Populations larger than 60 individuals have a significant risk of a sudden drop. We think that the gradual introduction of individuals into the river system may be responsible for mitigating the mass mortality effect of a sudden reproductive event.

We also compared the population size and total nutrient quantity over time (Figure 7) and found that the nutrient quantity is governed by the population size – nutrient quantity declines predictably as the population grows, and vice-versa. In particular, we did see evidence of some **critical population size**, between 50 and 60, where the total nutrient quantity declines suddenly and without warning. This decline ultimately triggers a mass mortality event.

Since fish have an internal stock of energy that enables them to survive for at least a few days in conditions of nutrient deficit, the fish population continues to increase even after nutrient levels drop. Thus, the fish population alone may be an over-estimation of the critical population size. To investigate the effect of the initial population size on the sustainable population size in greater detail, we disabled both immigration and reproduction in our model, and manually set the initial population size for several runs. In all cases, the nutrient levels throughout the river network were allowed to recover and stabilize before introducing any new individuals. The simulation was run until the fish population remained stable for at least a week of model time. Figure 8 shows the population size over time.

Populations that are initially very large (above k_f) drop to below k_{fs} rather quickly. Once initial population sizes drop below k_f, sustainable population sizes start to increase, and it takes longer to achieve stability, indicating a lower fish mortality rate. Figure 9 shows the sustainable population size with respect to the initial population size. We found that a population of between 40 and 50 individuals is reasonably stable and unlikely to drop dramatically in the long term.

These findings help us explain the two-cycles found in Figure 2. Reproductive events that yield a relatively small post-reproductive population, such as in the second year, are not followed by such a dramatic drop. The resulting, relatively large, sustainable population enters a reproductive cycle in the following year, producing many offspring. This large population soon drops dramatically, and the cycle repeats.

Evidently, large reproductive events actually contribute to mass mortality within a fish population. A reasonably-sized population is best sustained when the addition of individuals is gradual and continuous – as is the case with immigration.

Importance of the time-step

As mentioned on page 9, the time-step, dt or Δt, plays an important role in discrete-time models. In *The River Engine*, longer time-steps increase the simulation speed at the expense of accuracy, so selecting a time-step is always a compromise between the two. As the time-step approaches zero, the model approaches continuous-time behaviour.

Unless otherwise specified, all the simulations we ran used a time-step of 0.5 seconds. We selected this time-step for two reasons. First, we observed no significant change in population trends with smaller time-steps over the course of several years of model time, so this value seemed to yield adequate precision. Second, a time-step of 0.5 seconds enabled a set of simulations to be run conveniently overnight, producing results in the next morning. Some batches included as many as 12 simulations, and each simulation generally took 6-10 minutes to simulate a year of model time. Shorter time-steps would have increased simulation time

We simulated the null model for different values of dt to see what effect the time-step has on the outcome. Figure 11 shows the population sizes over time for $dt = 0.5$ and $dt = 5$ seconds. The population profile when $dt = 0.5$ s resembles previous results. When $dt = 30$ s, we saw sustained large populations for a short period after reproductive events at years 3, 5, 7, and 9, which shortly dropped to a more sustainable size. We did not see this kind of post-reproduction sustenance in any other simulations where $dt = 0.5$ s.

Figure 12 shows the inter-year sustainable population size for different values of dt. $dt = 0.5$ s clearly shows the two-cycle discussed in the previous section. The other two series are more chaotic and show some irregular oscillations.

One major effect that large time-steps have on this model has to do with fish movement. Fish move in an upstream-downstream random walk, and the distance that they move at every iteration is scaled to the time-step. A larger time-step results in fish being more likely to move longer distances over a period of time. Further, since fish cannot pass the termini of the river network, any fish that reaches a terminus becomes "stuck" at the terminus if it decides to move in the direction of the terminus. With sufficiently large time-steps, fish move distances so great that they are more likely to become "stuck" at termini. As these fish consume nutrients in their local neighbourhood, a nutrient-poor region develops at termini, leading to increased mortality in that locale. Indeed, when inspecting the raw data from a simulation with a large time-step, we observed a disproportionate amount of fish loitering at termini, and populations sometimes disappear completely due to fish unable to escape nutrient-poor regions.

Chapter 5

Conclusion and Future Directions

The River Engine has produced useful data, but since it is a novel work, there are certainly improvements that can be made to the model and methodology highlighted in this thesis.

First, most of the data that we presented were obtained from running a single simulation for each set of parameter values. Since *The River Engine* is stochastic, running more than one simulation would produce different results. It would be more correct to perform multiple simulations and present an aggregate of collected data, e.g. the mean sustainable population size at a given year. Collecting means would also enable us to perform statistical analyses on the results, in particular to determine if changing a parameter has a significant effect with respect to the null model. We chose to use single simulations primarily because we only had access to ordinary desktop computers, which were shared with other users. Performing multiple simulations and analysing results would have taken too much time and inconvenienced other users [G].

Similarly, most simulations were run for 10 years of model-time, to allow the simulation to complete in a reasonable amount of real-world time. In terms of asymptotic dynamics, 10 years is simply not sufficient to infer the long-term fate of a population. Running a series of long simulations (with end times of at least 50 years) is a logical next step in analyzing the viability of this model.

It is ironic that one of the primary reasons that ABMs are gaining ground in scientific computing is the rapid growth of computing power, and yet we cite the unavailability of computing power as a reason

[G] This was a major factor of our decision to run simulations overnight.

for being unable to obtain comprehensive replicates and long-term data. Indeed, we even preached the accessibility of *The River Engine* earlier in this work. While anyone with a reasonably modern computer can download and run *The River Engine*, running hundreds of simulations and aggregating the resulting data is quite a bit more involved. On the other hand, it is incredible that these kinds of models can be run by a layman; ten years ago, computers would have struggled to run *The River Engine* with comparable speed and precision.

We did not use *The River Engine* to model any particular ecosystem, in part due to inconsistent or incomplete data in the literature. A crucial step to evaluate whether *The River Engine* is a valid ecological model is to simulate a known and well-characterized aquatic ecosystem. This would require selecting a site and making specific observations to estimate parameter values, which is a task better suited for field ecologists than computational biologists [H].

Agent-based models are capable of simulating systems where each agent has its own set of "goals." In *The River Engine*, fish do not have unique "goals," and universally follow the same general rules. Fish behaviour may be improved by introducing a Markov chain of states. Fish may switch between different states, such as foraging, roaming, spawning, etc., that influence their immediate behaviour. For example, a foraging fish may me more likely to move more slowly and in a single direction. A roaming fish may move quickly and randomly in search for a nutrient-rich area.

Among the motivations for this project was a study of the effect of GMO species in a wild population of fish. GMO individuals, supposedly consuming more nutrients and growing to a larger size, may out-compete wild fish populations and harm the ecosystem. *The River Engine* is an ideal platform to study a breakout of GMO fish into the wild, as wild and GMO fish may be represented as agents with different properties (such as energy, or hunger). Agents are spatially aware, and so GMO fish affect their

[H] Mathematicians and computational biologists possess an understanding of a variety of spaces, but "outside" is rarely one of them.

immediate neighbourhood differently than far-away reaches. Adapting *The River Engine* to model wild fish as well as GMO fish would open the door to a number of biological questions that can be studied.

As it stands, *The River Engine* has produced insights that are applicable to biology. As agent-based models are further developed by investigators, they will emerge as extremely valuable tools to predict real-world phenomena. We hope that future ecologists will use and extend *The River Engine* for their own work and shed some light on the intimate lives of river fish.

Appendix A

Figures

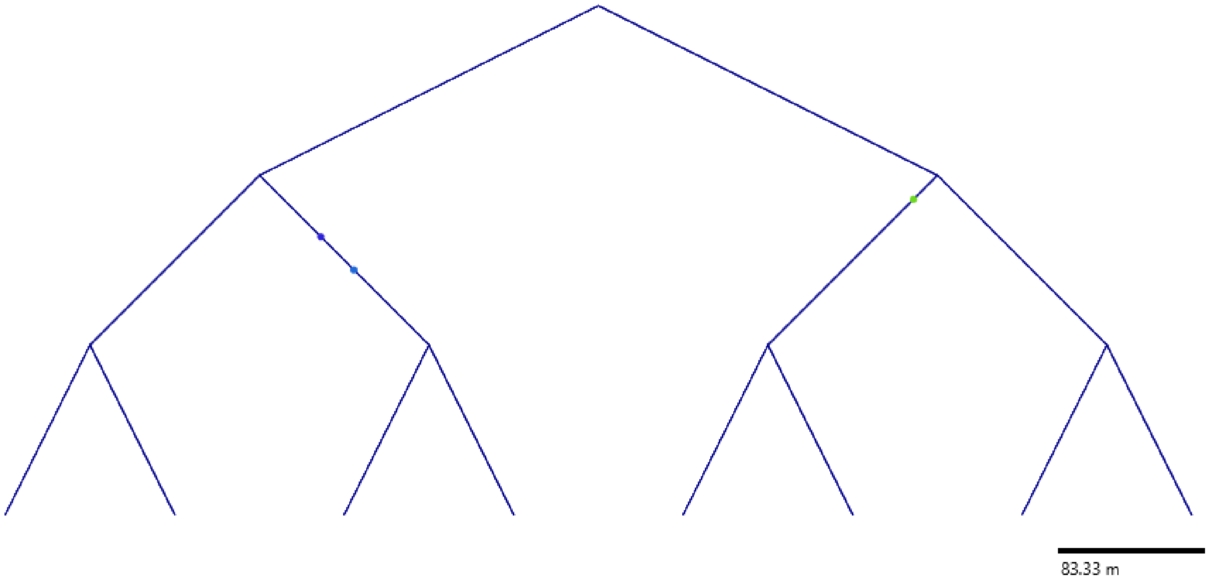

Figure 1: Diagram of the topology of the river network used in our experimental setup. The two blue dots represent fish.

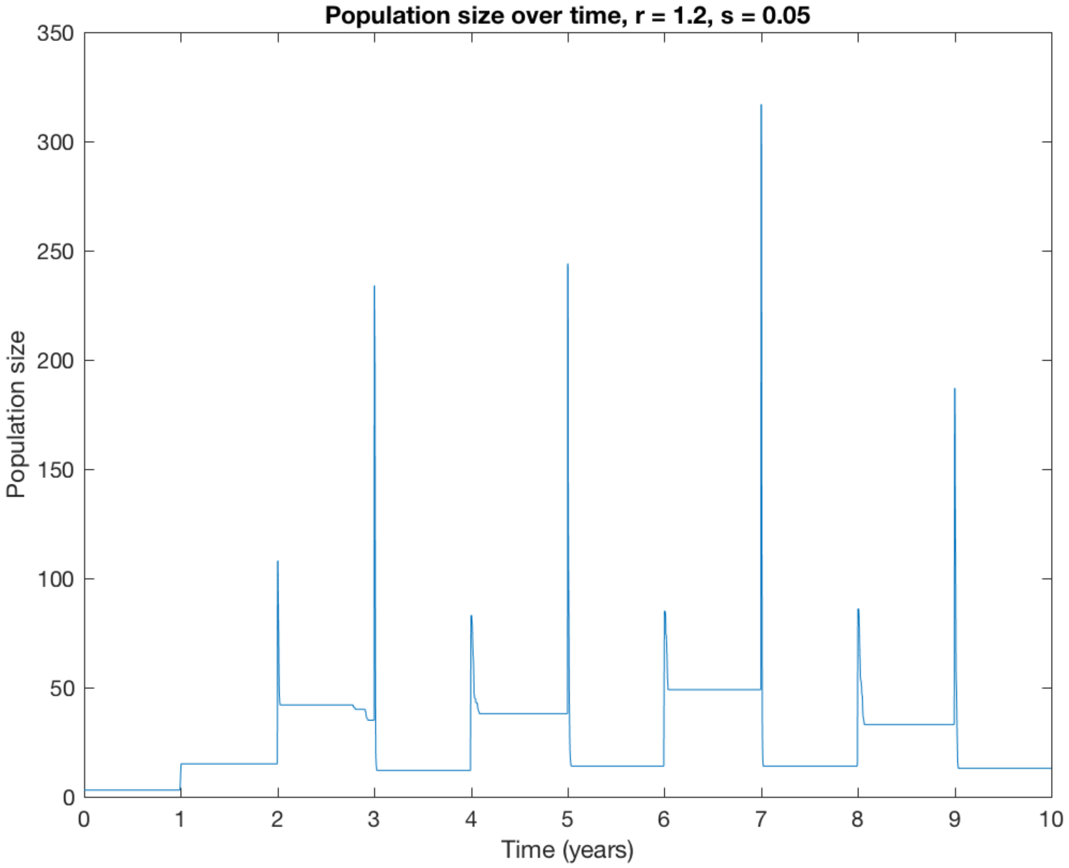

Figure 2: Population of fish over time with nutrient logistic growth rate $r = 1.2$ and nutrient addition rate $s = 0.05$. Peaks represent reproductive events at the beginning of each year, and plateaus represent the year-long, sustained population size. Oscillations of sustained population size from year to year are evident.

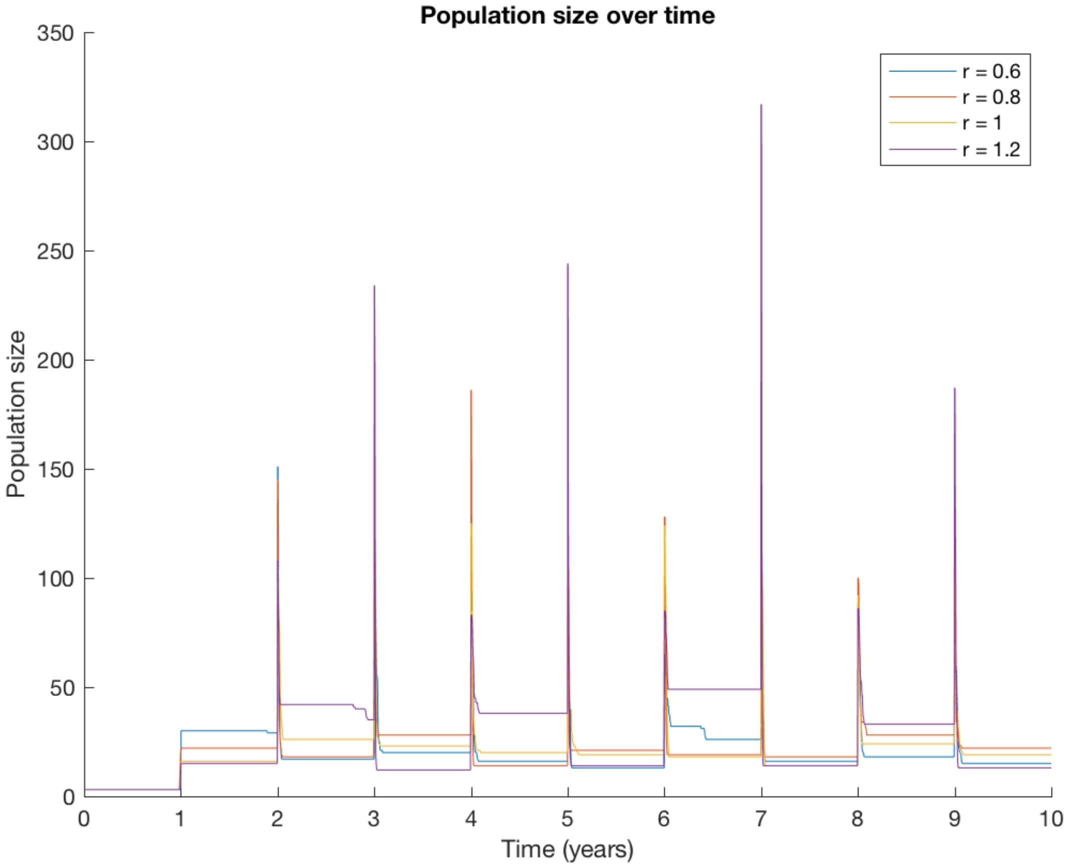

Figure 3: Population of fish over time for a range of nutrient logistic growth rates (r). Inter-year sustainable population sizes are comparable across this gradient.

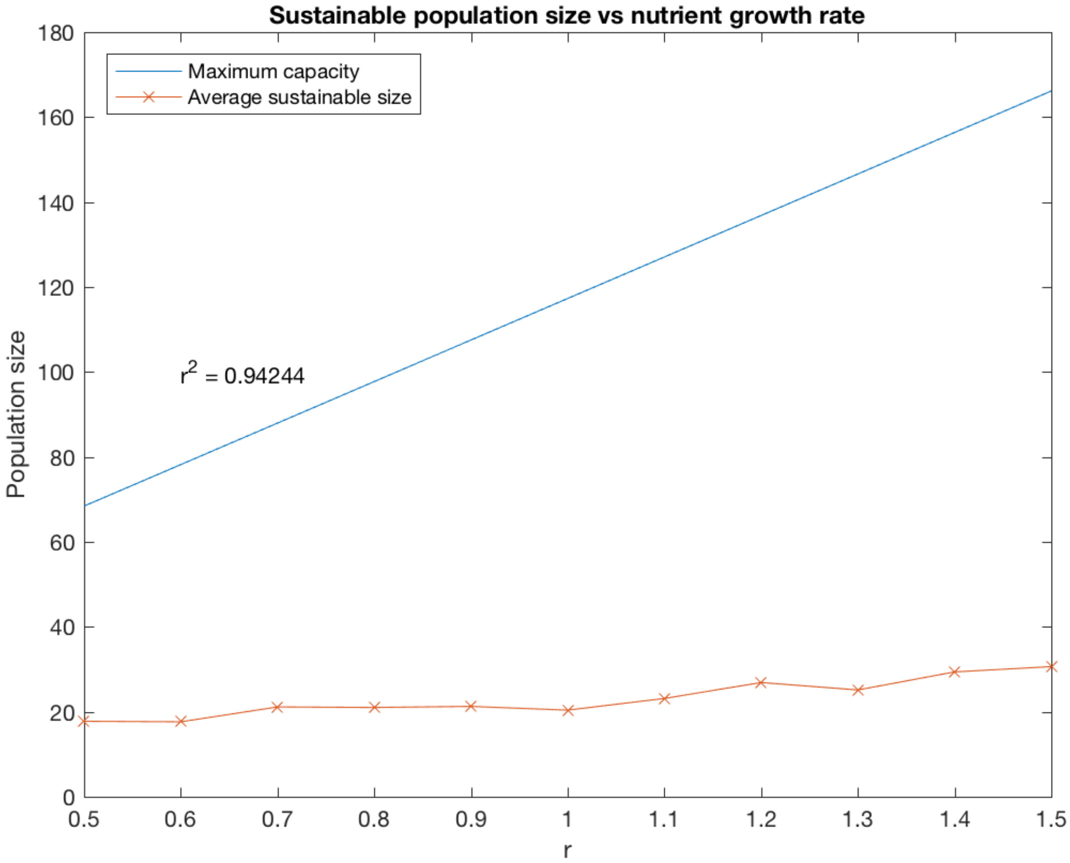

Figure 4: Comparison between average inter-year sustainable population sizes over 10 years and maximum fish capacity k_f versus the nutrient logistic growth rate r. s was fixed at 0.05.

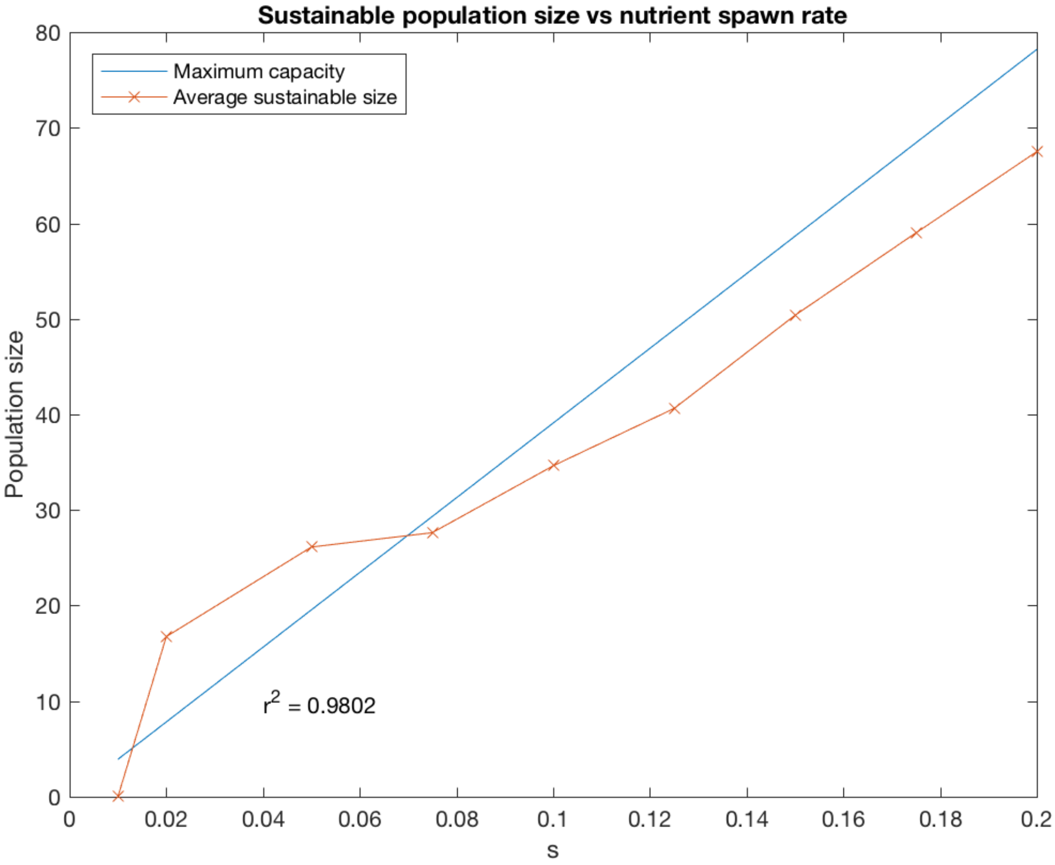

Figure 5: Comparison between average inter-year steady population sizes and maximum fish capacity from nutrient addition versus the nutrient addition rate s. r was fixed at 1.

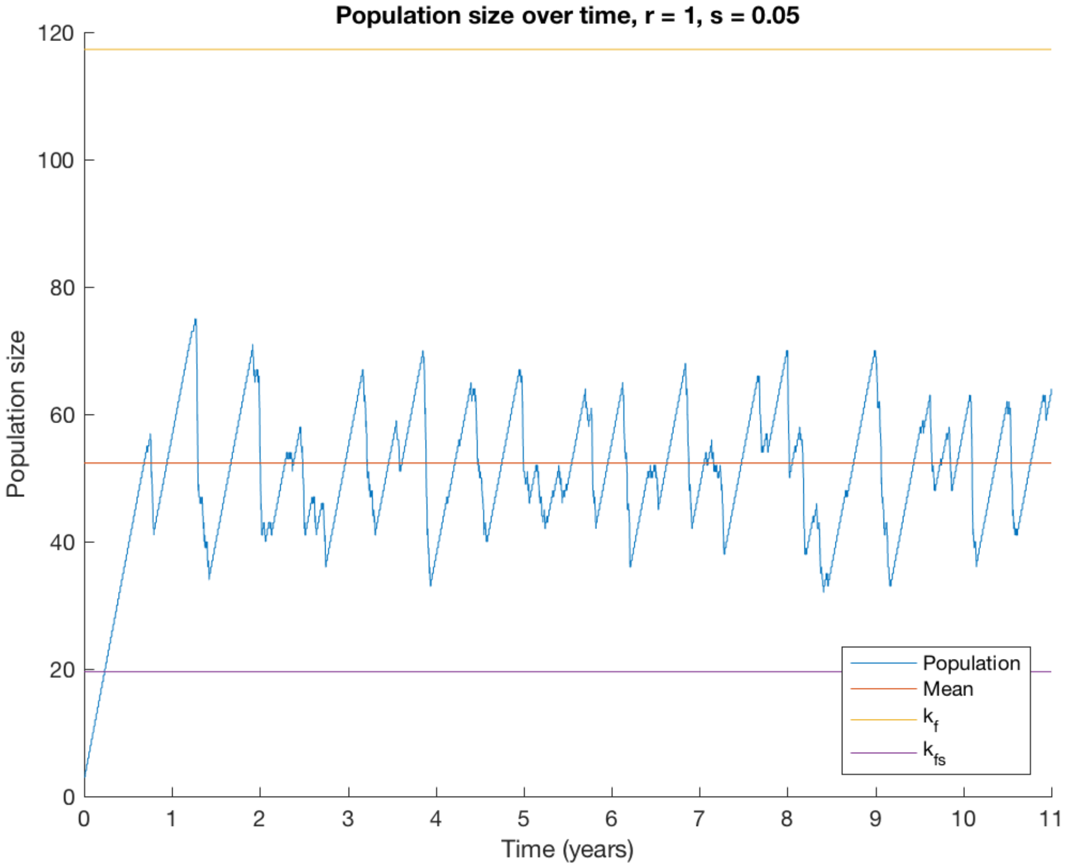

Figure 6: Population size over time with no reproduction and constant immigration – one individual is added every 5 days. $r = 1, s = 0.05$. Reference lines are shown for the population mean, maximum population size k_f, and theoretical sustainable size k_{fs} from nutrient addition.

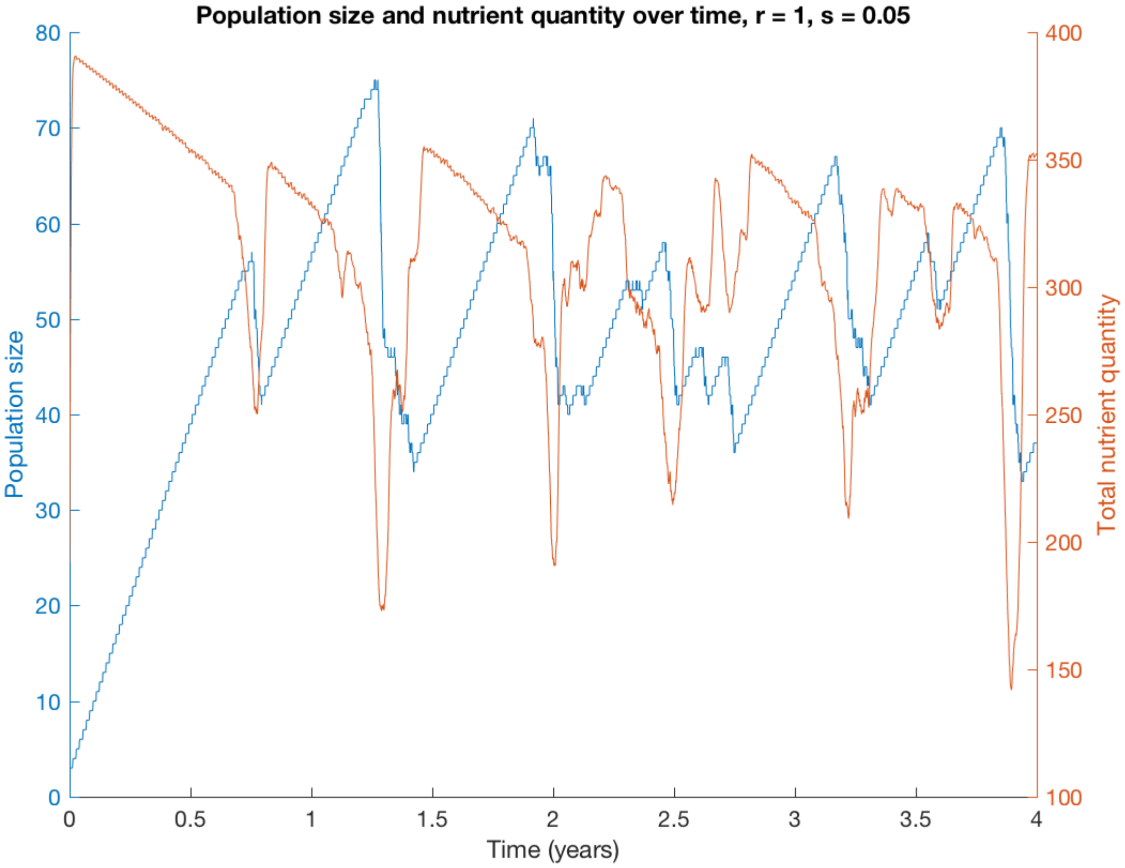

Figure 7: Population size and nutrient quantity over the entire river network over time with no reproduction and constant immigration – one individual is added every 5 days. Only the first 4 years are shown.

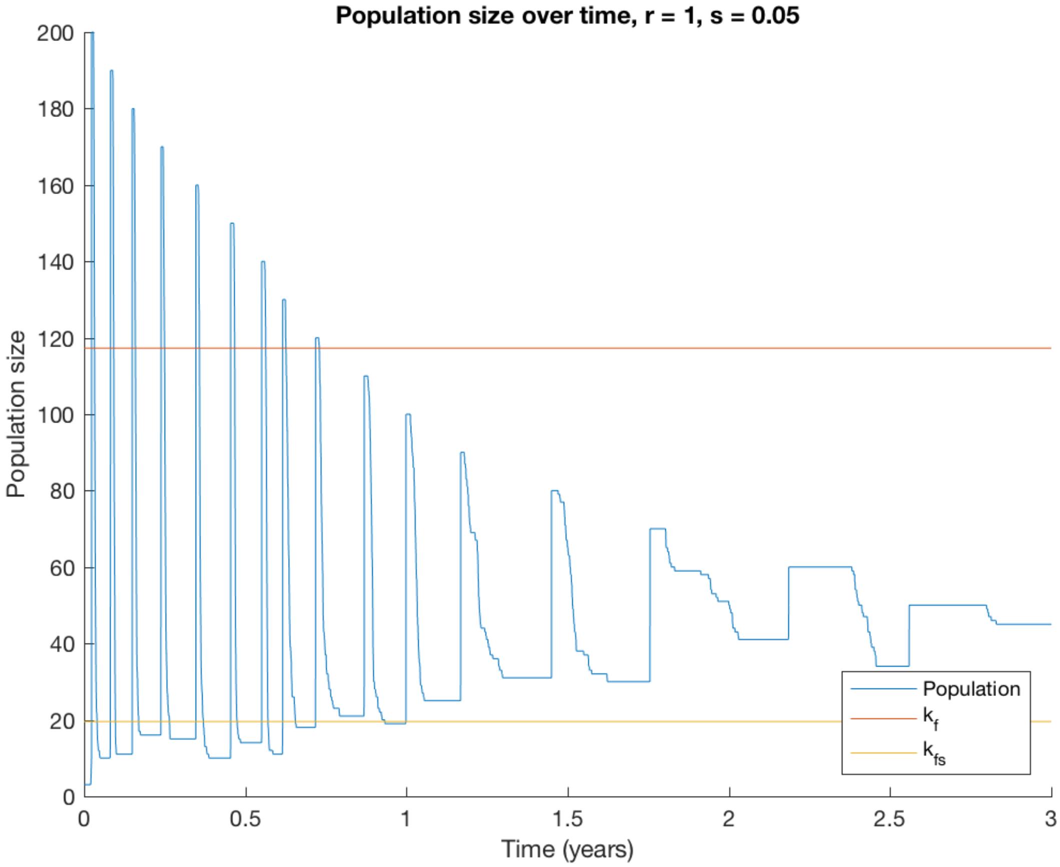

Figure 8: Timeline of population size over time with no reproduction or immigration. Initial population sizes at each peak were set manually and the simulation was run until the fish population and nutrient quantity reached stability. Nutrient logistic growth parameters are $r = 1, s = 0.05$.

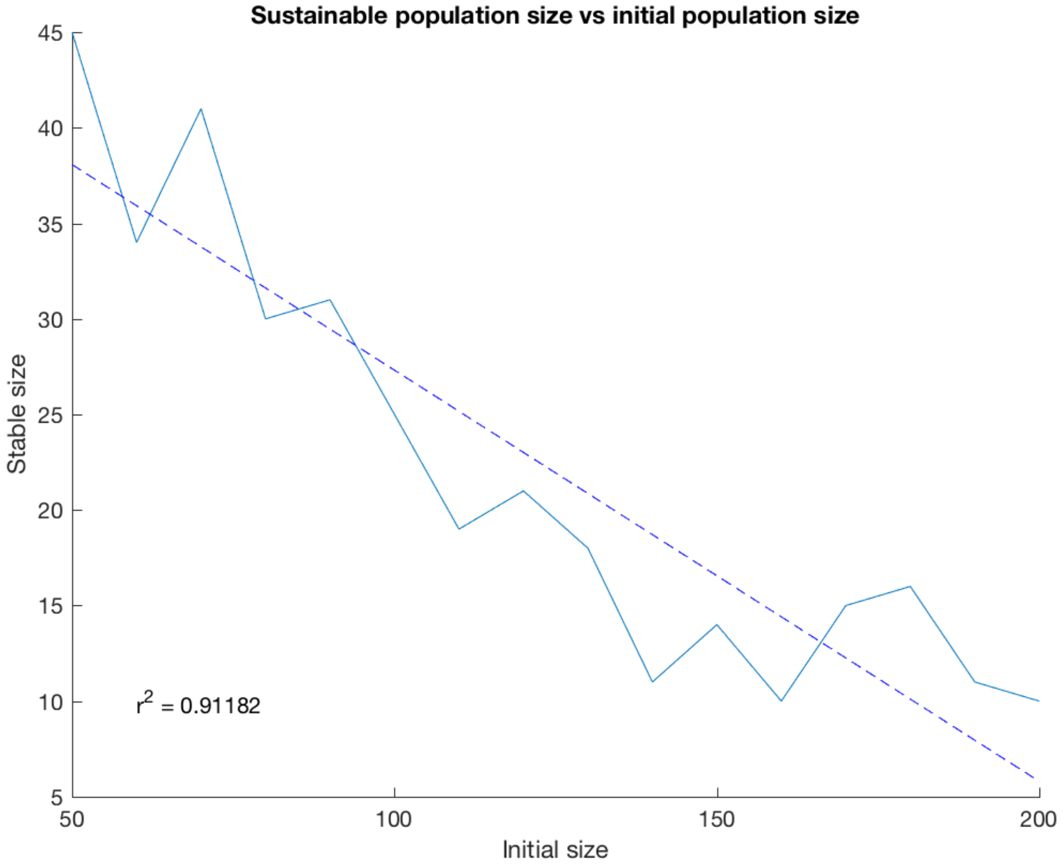

Figure 9: Plot of stable population sizes versus initial population sizes from Figure 7. A line of best fit is shown.

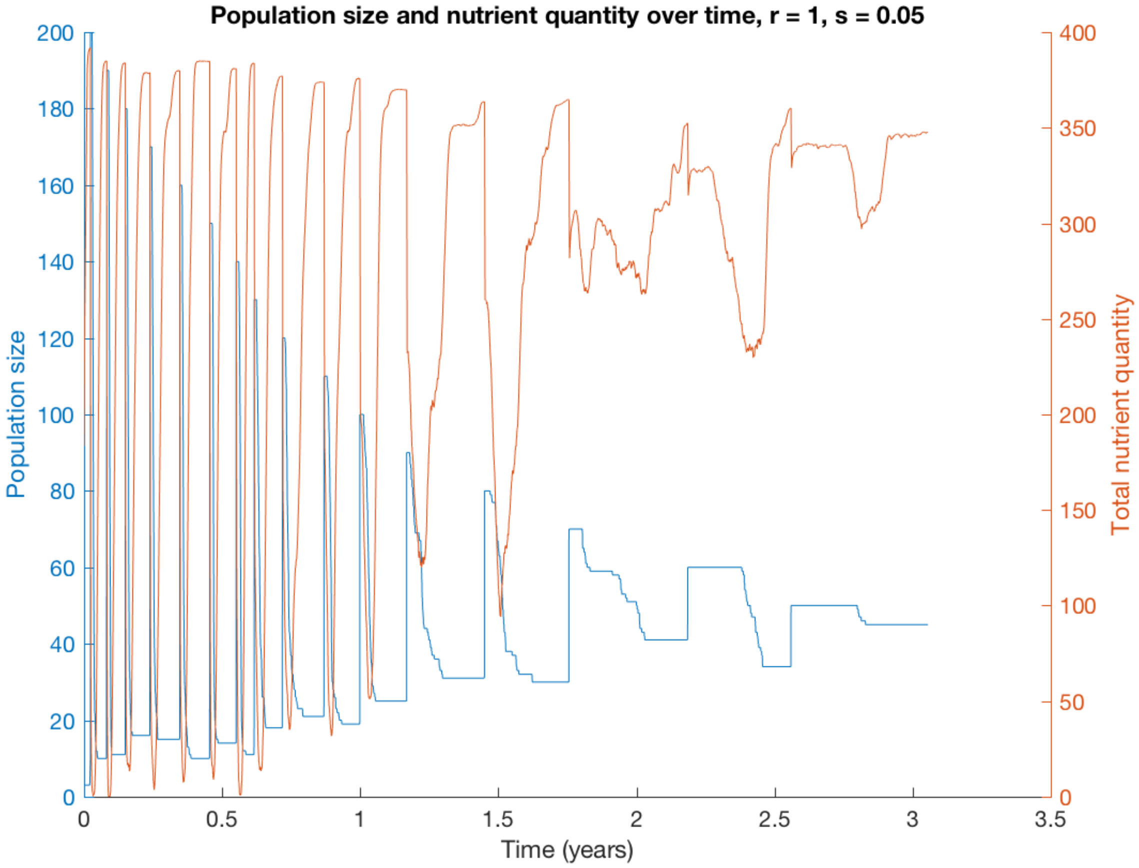

Figure 10: Timeline of population size and nutrient quantity in the entire river network, from the same simulation as Figure 7. Fish immigration and reproduction were disabled.

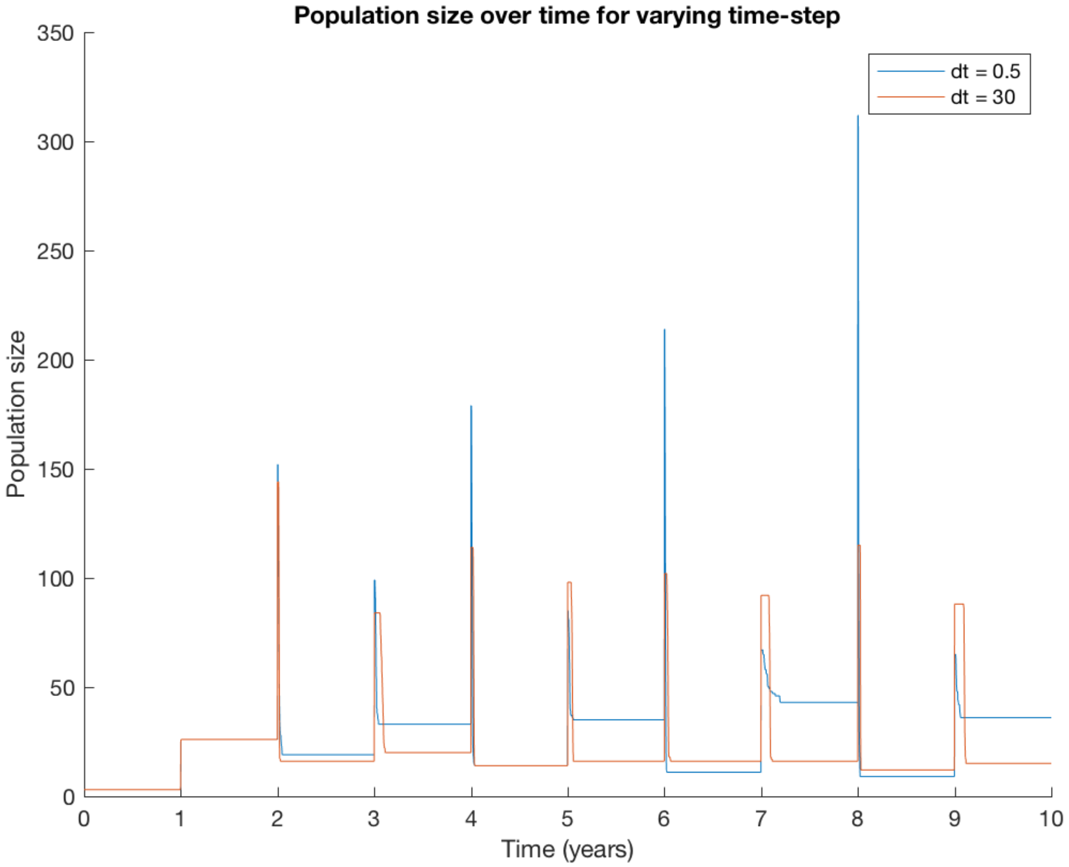

Figure 11: Population size over time for two different time-steps. Time-steps are given in seconds.

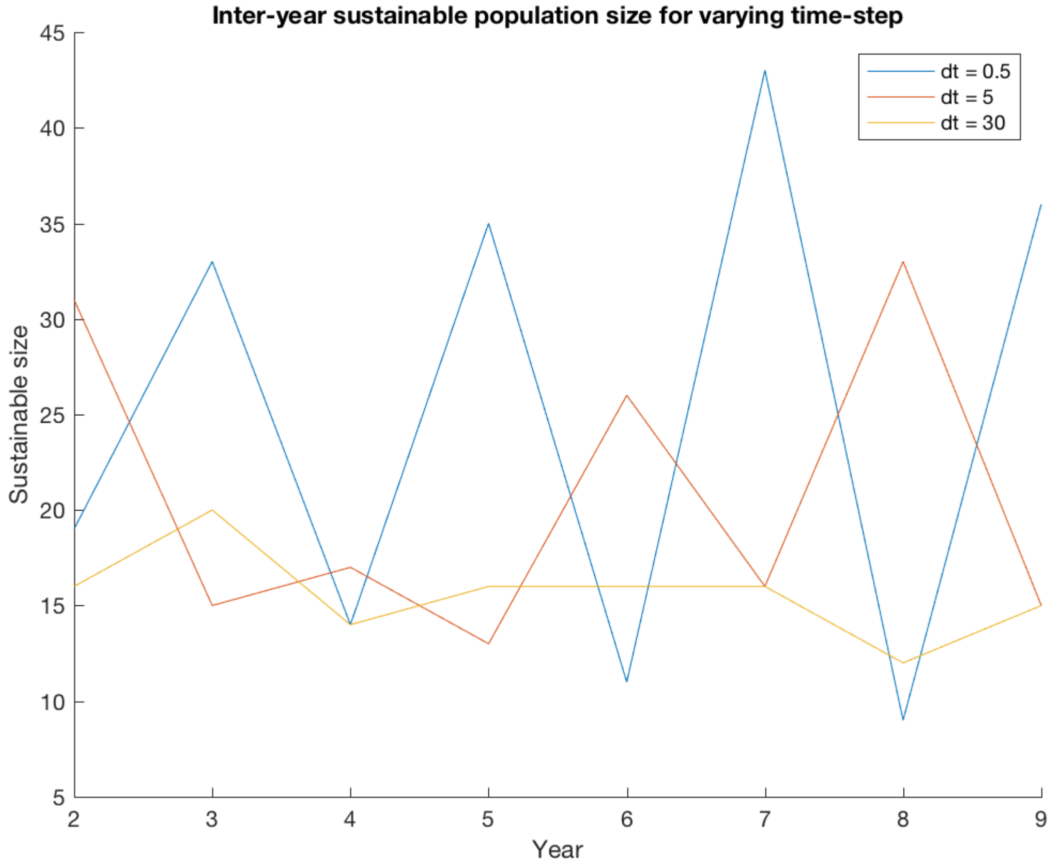

Figure 12: Inter-year sustainable population sizes for various time-steps. Time-steps are given in seconds. A two-cycle is visible in the series for $dt = 0.5$ s. The other two series are more erratic and no clear pattern can be inferred.

Appendix B

Glossary

Agent: In *The River Engine*, an individual fish, with characteristics and goals that may vary from other fish. In ABMs, autonomous entities that possess independent properties, and behaviours.

Agent-based model: A type of model that simulates the actions of autonomous, heterogeneous agents in an environment. Each agent may act independently.

Critical population size: Lowest population size that has a high risk of becoming unsustainable, and that is often followed by a pronounced increase in mortality as the population drops to a more sustainable level.

Environment: In *The River Engine*, the physical environment in which the fish reside; the "locale" of the model. The environment is defined by the river network and any additional data, for instance: availability of nutrients. In ABMs, a representation of the physical environment in which agents reside, and with which agents may interact.

Emergent behaviour: Behaviour of a system that is not explicitly stated as a property, rule, or algorithm. Emergent behaviours arise from interactions between agents and the environment, and are often highly complex and unpredictable. Life itself is an emergent behaviour from a chemical system of interacting molecules [39].

Fish and chips: A popular warm dish of English origin consisting of a breaded fish served with potato fries ("chips"). Not to be confused with common methods of tracking fish, which involve inserting non-potato, electronic tracking chips into a live fish.

Influence/Interaction: A rule that defines how agents and the environment interact or change over time. In *The River Engine*, these rules are called "influences."

Maximum population size: Upper bound of the population size than an environment can support, under ideal conditions, given by k_f. Related to the upper bound of the population size that an environment can support supposing no logistic growth, given by k_{fs}.

River arc: *The River Engine* lingo for a river reach, spanning from one **river node** to another. Represented by an edge in a graph.

River network: The topology of the river network under study, represented as a directed graph.

River node: *The River Engine* lingo for a fork in a river network. Represented by a vertex in a graph.

River system: The state of the model at any given time, given by the combination of the state of all agents and the physical environment.

Sustainable population size: Or **inter-year population size**. Population size that is constantly sustained over the long term between reproductive events.

Works Cited

1. Jennings, N. R. On agent-based software engineering. *Artif. Intell.* **117,** 277–296 (2000).
2. Stephen Wolfram. Cellular Automata: Proceedings of an Interdisciplinary Workshop. in (eds. Stephen Wolfram, D. Farmer & T. Toffoli) (North-Holland, 1983).
3. Badham, J. & Osborn, D. in *Mathematical Modelling of Zombies* (ed. Smith?, R.) 209–232 (University of Ottawa Press, 2014).
4. Macy, M. W. & Willer, R. From Factors to Factors: Computational Sociology and Agent-Based Modeling. *Annu. Rev. Sociol.* **28,** 143–166 (2002).
5. Thorne, B. C., Bailey, A. M. & Peirce, S. M. Combining experiments with multi-cell agent-based modeling to study biological tissue patterning. *Brief. Bioinform.* **8,** 245–57 (2007).
6. Manson, S. M. Agent-based modeling and genetic programming for modeling land change in the Southern Yucatán Peninsular Region of Mexico. *Agric. Ecosyst. Environ.* **111,** 47–62 (2005).
7. Shi, J., Ren, A. & Chen, C. Agent-based evacuation model of large public buildings under fire conditions. *Autom. Constr.* **18,** 338–347 (2009).
8. Benenson, I. in *Spatially Integrated Social Science: Examples in Best Practice* 67–95 (Oxford University Press, 2004).
9. Keyes, R. W. The Impact of Moore's Law. *IEEE Solid-State Circuits Newsl.* **20,** 25–27 (2006).
10. Balicer, R. D. Modeling Infectious Diseases Dissemination Through Online Role-Playing Games. *Epidemiology* **18,** 260–261 (2007).
11. Laura Sydell. 'Virtual' Virus Sheds Light on Real-World Behavior : NPR. *NPR* (2005). Available at: http://www.npr.org/templates/story/story.php?storyId=4946772. (Accessed: 29th March 2017)
12. Banks, M. & Miller, J. *Summary of Fish Tagging and Evaluation Techniques Currently Used in the Columbia River Basin*. *Pacific Northwest Aquatic Monitoring Partnership* (2010).
13. Tytler, P., Thorpe, J. E. & Shearer, W. M. Ultrasonic tracking of the movements of atlantic salmon smolts (Salmo salar L) in the estuaries of two Scottish rivers. *J. Fish Biol.* **12,** 575–586 (1978).
14. Cooke, S. J. *et al.* Tracking animals in freshwater with electronic tags: past, present and future. *Anim. Biotelemetry* **1,** 5 (2013).
15. OSEARCH. OCEARCH.ORG. (2017). Available at: http://www.ocearch.org/. (Accessed: 14th April 2017)
16. Schofield, G. *et al.* Novel GPS tracking of sea turtles as a tool for conservation management. *J. Exp. Mar. Bio. Ecol.* **347,** 58–68 (2007).
17. Baglinière, J.-L., Maisse, G. & Nihouarn, A. Radio-tracking of male adult Atlantic salmon, Salmo salar L., during the last phase of spawning migration in a spawning stream (Brittany, France). *Aquat. Living Resour.* **4,** 161–167 (1991).
18. Thorpe, J. E., Ross, L. G., Struthers, G. & Watts, W. Tracking Atlantic salmon smolts, Salmo salar L., through Loch Voil, Scotland. *J. Fish Biol.* **19,** 519–537 (1981).

19. Welch, D. W., Boehlert, G. W. & Ward, B. R. POST–the Pacific Ocean salmon tracking project. *Oceanol. Acta* **25,** 243–253 (2002).

20. Gowans, A. R. D., Armstrong, J. D. & Priede, I. G. Movements of adult Atlantic salmon in relation to a hydroelectric dam and fish ladder. *J. Fish Biol.* **54,** 713–726 (1999).

21. Cash, K. M. *et al. Three-Dimensional Fish Tracking to Evaluate the Removable Spillway Weir at Lower Granite Dam During 2003*. (2003).

22. Kirby, D. S., Allain, G., Lehodey, P. & Langley, A. Individual/Agent-based Modelling of Fishes, Fishers, and Turtles Introduction to Individual/Agent-based Modelling. (2004).

23. Oracle. Java Software | Oracle. (2017). Available at: https://www.oracle.com/java/index.html. (Accessed: 18th January 2017)

24. Oracle. Java 8 Central. (2017). Available at: http://www.oracle.com/technetwork/java/javase/overview/java8-2100321.html. (Accessed: 18th January 2017)

25. The Apache Software Foundation & Contributors. Commons CLI 1.4. (2017). Available at: https://commons.apache.org/proper/commons-cli/. (Accessed: 29th March 2017)

26. The Apache Software Foundation & Contributors. Commons IO 2.5. (2013). Available at: https://commons.apache.org/proper/commons-io/. (Accessed: 29th March 2017)

27. The Apache Software Foundation & Contributors. Apache Commons Lang 3.4. (2016).

28. Google & Contributors. GSON 2.8.0. (2017).

29. Szkiba, I. Ini4j 0.5.4. (2016).

30. Naveh, B. & Contributors. JGraphT 1.0.0. (2016).

31. JUnit & Contributors. JUnit 4.12. (2016).

32. Hanzel, M. arthanzel/theriverengine. (2017). Available at: http://github.com/arthanzel/theriverengine.

33. Hodder, K. H. *et al.* Techniques for evaluating the spatial behaviour of river fish. *Hydrobiologia* **582,** 257–269 (2007).

34. Hirata, K. Swimming Speeds of Some Common Fish. *National Maritime Research Institute* (1999). Available at: https://www.nmri.go.jp/eng/khirata/fish/general/speed/speede.htm. (Accessed: 1st January 2017)

35. Crook, D. A. Movements associated with home-range establishment by two species of lowland river fish. *Can. J. Fish. Aquat. Sci.* **61,** 2183–2193 (2004).

36. Crook, D. A. Is the home range concept compatible with the movements of two species of lowland river fish? *J. Anim. Ecol.* **73,** 353–366 (2004).

37. Svendsen, M. B. S. *et al.* Maximum swimming speeds of sailfish and three other large marine predatory fish species based on muscle contraction time and stride length: a myth revisited. *Biol. Open* **5,** 1415–1419 (2016).

38. Peake, S. J. Swimming performance and behaviour of fish species endemic to Newfoundland and Labrador: A literature review for the purpose of establishing design and water velocity criteria for fishways and culverts Canadian Manuscript Report of Fisheries and Aquatic Sciences No. 2843. (2008).

39. Luisi, P. L. *The emergence of life : from chemical origins to synthetic biology*. (Cambridge University Press, 2006).

40. James Alexander. BBC News - The unlikely origin of fish and chips. (2009). Available at: http://news.bbc.co.uk/2/hi/8419026.stm. (Accessed: 14th April 2017)

41. Nestler, J. M. *et al.* THE RIVER MACHINE: A TEMPLATE FOR FISH MOVEMENT AND HABITAT, FLUVIAL GEOMORPHOLOGY, FLUID DYNAMICS AND BIOGEOCHEMICAL CYCLING. *River Res. Appl.* **28,** 490–503 (2012).

42. Python, M. Monty Python and the Holy Grail. (1975).