A Novel Approach to Feedback Control with Deep Reinforcement Learning

Kala Agbo Bidi[a], Jean-Michel Coron[a], Amaury Hayat[b], Nathan Lichlté[b,c]

[a]*Sorbonne Université, CNRS, Université Paris-Cité, Laboratoire Jacques-Louis Lions,*
LJLL, INRIA équipe CAGE, Paris, 75005, France
[b]*CERMICS, Ecole des Ponts ParisTech, 6-8 avenue Blaise Pascal,*
amaury.hayat@enpc.fr, Champs-sur-Marne, France
[c]*EECS, UC Berkeley, Berkeley, CA, USA*

Abstract

We present a novel approach to feedback control design by leveraging the power of deep reinforcement learning (RL). We blend the RL methodology with mathematical analysis to extract an explicit feedback control for a dynamical system modeling biological pest control using the Sterile Insect Technique (SIT).

While SIT has traditionally been used in agriculture and is currently a method for combating vector-borne diseases carried by mosquitoes, the practical implementation of feedback controls derived from classical control theory is limited by the need for continuous and often impractical measurements. Finding a feedback control that ensures the global stability of the system with only practical measurements is a complicated mathematical problem. To overcome this, our approach focuses on utilizing deep RL to suggest and construct feedback laws that only depends on these measurements, namely the adult mosquito population, which can be measured using pheromone traps.

Many dynamical systems arising from practical applications are subject to measurement constraints, which render the stabilization problem complex from a mathematical perspective. We believe that this approach could help in finding new solutions to these problems.

Keywords: Asymptotic stability; global stabilisation; Reinforcement Learning; Robust control; Sterile Insect Technique

1. Introduction

Sterile Insect Technique (SIT) is a method of biological pest control that consists of releasing sterilized insects to reduce or eliminate a target population. Initially used in agriculture to control insect pests, it is now employed in the vector-born disease fight against mosquitoes that carry illnesses such as malaria and arboviruses [1, 2] and there is a great interest both in research and in practice to understand which control to use for releasing the sterilized insects [3, 4, 5].

In this paper we introduce an approach that employs deep reinforcement learning (RL) to suggest mathematical control strategies for dynamical systems. We illustrate it on a mathematical model of SIT applied on mosquitoes population, namely (2.1)-(2.4) below.

Several mathematical approaches have already been used in the literature to treat the SIT control problem applied to mosquitoes either for the complete system or for reduced models. Two reduced models have been considered: a two dimensional (2D) model obtained by assuming that the dynamics of males and eggs are fast so that these two populations can be assumed to be at equilibrium (see [5, ($\mathcal{S}_1$), page 231-232] or [6, (2)]); and a three dimensional (3D) model obtained by overlooking the non-adult stages (see [7, (7a)-(7b)-(7c)]). These mathematical approaches have led to the following stabilizing feedback controls:

- Stabilization using impulsive feedback controls for the 3D model: [7, Theorem 6] and [7, Theorem 7] for the case of sparse measurements. The case of vector migration is also considered in [8].

- Stabilization using optimal feedback control for the 2D model: [5, Remark 4].

- Stabilization using the backstepping method: [6] considers the 2D model while [9, Section 3.1] considers the complete model. See, for example, [10, Section I.2.2], [11, Pages 242–246] or [12, Section 12.5] for tutorial presentations of the backstepping method.

- Stabilization using simple linear feedback laws: [9] proves stabilization for positively invariant subsets and conjectures it for the complete model.

These feedback controls above are constructed using classical tools in control theory such as control Lyapunov functions, the LaSalle invariance principle, the maximum principle, monotone dynamical systems, barrier functions, or the backstepping method. Although these feedback controls provide evidence in terms of robustness, their applications requires continuously measuring the different states of the model, which is difficult or impossible in practice. Thank to our approach, we are able to provide a control only based on the most accessible data to be measured: the adult population (adult females and adult males) using pheromone traps, which are already used in practice.

Our approach differs from these classical methods and uses deep reinforcement learning (RL) to construct control feedback laws. Over the past few years, RL has emerged as a powerful approach for control, with its ability to learn near-optimal decision-making strategies through interactions with an environment, and has demonstrated remarkable successes across a wide range of domains and applications with long-term horizons, high-dimensional partially-observable states. In robotics, RL has enabled machines to learn complex tasks such as locomotion, manipulation, and dexterous object handling [13, 14, 15]. In the realm of games, RL algorithms have achieved superhuman performance in challenging domains like Go, Chess, StarCraft [16, 17], as well as in classical Atari games [18]. These remarkable achievements highlight the versatility and potential of RL as a general-purpose approach for solving complex practical control problems in diverse domains.

RL techniques, while powerful for decision-making, inherently provide control mechanisms that are discrete and numerical in nature. However, from a more rigorous mathematical point of view, these mechanisms often don't translate directly into analytical feedback control formulas. Recognizing this limitation, in our work, we blend RL methodologies with mathematical analysis to extract an explicit mathematical control. Importantly, in our approach we apply RL to solve the mathematical problem, rather than solely applying RL to the discretized system, a more conventional application of RL. This allows us to employ deep RL to architect control feedback laws which can be juxtaposed with existing controls that have been derived from more traditional methodologies.

2. Context and problem studied

The SIT model in mosquitoes population is given by the following system of equations:

$$\dot{E} = \beta_E F \left(1 - \frac{E}{K}\right) - \big(\nu_E + \delta_E\big)E, \tag{2.1}$$

$$\dot{M} = (1-\nu)\nu_E E - \delta_M M, \tag{2.2}$$

$$\dot{F} = \nu\nu_E E \frac{M}{M+M_s} - \delta_F F, \tag{2.3}$$

$$\dot{M_s} = u - \delta_s M_s, \tag{2.4}$$

where, at time t, $E(t) \geq 0$ is the mosquito density in aquatic phase, $M(t) \geq 0$ is the wild adult male density, $F(t) \geq 0$ is the density of adult females which have been fertilized, $M_s(t) \geq 0$ is the sterilized adult male density, and $u(t) \geq 0$, the control, is the density of sterilized males released at time t.

In system (2.1)–(2.4) we assume that all females mate as soon as they emerge from the pupal stage. The density of unfertilized females, i.e. the density of females that have mated with sterilized males, is denoted by $F_s(t)$. One has $F_s(t) = F(t)M_s(t)/M(t)$. Besides, we also assume that $\delta_s \geq \delta_M$, which is usually considered to be a biologically relevant assumption [5]. The interpretation of the parameters are given below [5]:

- $\beta_E > 0$ is the oviposition rate,
- $\delta_E, \delta_M, \delta_F > 0$ are the death rates for eggs, wild adult males and fertilized females respectively,
- $\nu_E > 0$ is the hatching rate for eggs,
- $\nu \in (0,1)$ the probability that a pupa gives rise to a female (and $(1-\nu)$ is, therefore, the probability to give rise to a male),
- $\delta_s > 0$ is the death rate of sterilized adults,
- $K > 0$ is the environmental capacity for eggs. It can be interpreted as the maximum density of eggs that females can lay in breeding sites. Since here the larval and pupal compartments are not present, it can be interpreted as E representing all the aquatic compartments and this term K representing a logistic law's carrying capacity for the aquatic phase (that also includes the effects of competition between larvae).

Typical values for these parameters as well as the values used in this work are given in Table .4.

For the parameters given in Table .4, when $u(t) = M_s(t) = 0$ for any $t \geq 0$, the system (2.1)–(2.3) has a unique globally asymptotically stable equilibrium $(E(t), M(t), F(t)) \equiv (E^*, M^*, F^*)$ where E^*, M^* and F^* are large constant values. This corresponds to the situation where mosquitoes reproduce freely. The state $(E(t), M(t), F(t)) \equiv (0, 0, 0)$ is also an equilibrium, albeit an unstable one. The mathematical problem is to find $u(t)$ of the form

$$u(t) = f(M(t) + M_s(t), F(t) + F_s(t)), \tag{2.5}$$

where $f \in L^\infty(\mathbb{R}^2)$, such that the zero equilibrium $(0, 0, 0)$ is globally asymptotically stable and M_s is asymptotically small, meaning there exists $c \in \mathbb{R}_+$ such that

$$\lim_{t\to+\infty} \|u(t)\| = c < U^* \tag{2.6}$$

where

$$U^* := \frac{K\beta_E\nu(1-\nu)\nu_E^2\delta_s}{4(\delta_E + \nu_E)\delta_F\delta_M}\left(1 - \frac{\delta_F(\nu_E + \delta_E)}{\beta_E\nu\nu_E}\right)^2, \tag{2.7}$$

and the equilibrium $(0, 0, 0, c/\delta_s)$ of the system (2.1)–(2.4) is globally asymptotically stable (see Definition 2.1 below, where the notion of solutions of the closed-loop system is understood in the Fillipov sense [9, Section 2.2]). Ideally, one would even like to be able to find, for any $\varepsilon > 0$, a control feedback law f_ε such that

$$\lim_{t\to+\infty} u(t) = \varepsilon. \tag{2.8}$$

Definition 2.1. *The equilibrium* $(0, 0, 0, c/\delta_s)$ *of the system* (2.1)–(2.4) *with the feedback law* (2.5) *is* globally asymptotically stable *if, for any initial condition* $(E_0, M_0, F_0, M_{s,0})$*, the (forward maximal) solutions* (E, M, F, M_s) *to the system* (2.1)–(2.4) *with the feedback law* (2.5) *are defined on* $[0, +\infty)$ *and for any* $\varepsilon > 0$ *there exists* $\delta > 0$ *such that*

$$\begin{gathered}\|(E_0, M_0, F_0, M_{s,0} - c/\delta_s)\| \leq \delta \implies \\ \|(E(t), M(t), F(t), M_s(t) - c/\delta_s)\| \leq \varepsilon,\ \forall t \in [0, +\infty),\end{gathered} \tag{2.9}$$

$$\lim_{t\to+\infty} \|(E(t), M(t), F(t), M_s(t) - c/\delta_s)\| = 0, \tag{2.10}$$

The form constraint (2.5) corresponds to a practical limitation: $M + M_s$ and $F + F_s$ are the total number of males and females which are typically what can be measured in practice (see [1]), although we consider different variants in this work.

Remark 2.1 (Constant control). *The rationale behind the definition of* (2.6) *is that this is the critical value above which a constant control can stabilize the state* $(E^*, M^*, F^*) = (0,0,0)$*. Indeed, for a constant control* $u(t) \equiv \bar{U}$*, if* $\bar{U} > U^*$ *then the equilibrium* $(E^*, M^*, F^*, M_s^*) = (0,0,0,U^*/\delta_s)$ *is globally asymptotically stable (see [5]).*

Remark 2.2 (Optimal decay rate). *Assume that* $E(0) \leq K$*. Then, for any control* $u(t) \geq 0$ *is, for every time* $t \geq 0$*, we have* $E(t) \leq K$ *and*

$$E(t) \geq \tilde{E}(t),\ M(t) \geq \tilde{M}(t),\ F(t) \geq \tilde{F}(t), \tag{2.11}$$

where $(\tilde{E}, \tilde{M}, \tilde{F})$ *is the solution to the Cauchy problem*

$$\dot{\tilde{E}} = \beta_E \tilde{F}\left(1 - \frac{\tilde{E}}{K}\right) - \big(\nu_E + \delta_E\big)\tilde{E}, \tag{2.12}$$

$$\dot{\tilde{M}} = (1-\nu)\nu_E \tilde{E} - \delta_M \tilde{M}, \tag{2.13}$$

$$\dot{\tilde{F}} = -\delta_F \tilde{F}, \tag{2.14}$$

$$(\tilde{E}(0), \tilde{M}(0), \tilde{F}(0)) = (E(0), M(0), F(0)). \tag{2.15}$$

It would be interesting to see if one can get with suitable output feedback laws (vanishing or small at the origin) a decay rate close to the one imposed by (2.11)*, i.e.*

$$E(t) \simeq \tilde{E}(t),\ M(t) \simeq \tilde{M}(t),\ F(t) \simeq \tilde{F}(t). \tag{2.16}$$

(Note that it is possible to get (2.16) *by taking* u *constant and large, depending on* $(E(0), M(0), F(0))$*.) This would be particularly useful in the case where the insect under study reproduces both sexually and asexually: indeed, this would give the best way to reduce the sexual reproduction part as much as possible by output feedback laws (vanishing or small at the origin).*

In [9] a backstepping feedback control was built to stabilize this specific system at the origin. It is defined by

$$u((x^T, M_s)^T) := \max\big(0, G((x^T, M_s)^T)\big)\,. \tag{2.17}$$

where $G : \mathcal{D}' := [0, +\infty)^4 \to \mathbb{R}$, $(x^T, M_s)^T \mapsto G((x^T, M_s)^T)$ is given by

$$\begin{aligned} G((x^T, M_s)^T) := {} & \frac{\psi E(\theta M + M_s)^2}{\alpha(M + M_s)(3\theta M + M_s)} \\ & + \frac{((1-\nu)\nu_E \theta E - \theta \delta_M M)(\theta M + 3M_s)}{3\theta M + M_s} \\ & + \delta_s M_s + \frac{1}{\alpha}(\theta M - M_s) \text{ if } M + M_s \neq 0, \end{aligned} \tag{2.18}$$

$$G((x^T, M_s)^T) := 0 \text{ if } M + M_s = 0. \tag{2.19}$$

$$\psi := \frac{2\beta_E \nu \nu_E}{\delta_F (1 - \mathcal{R}(\theta))(1 + \theta)}, \tag{2.20}$$

and $\theta > 0$ is a regulation constant.

However, this control depends on the three variables (E, M, M_s) and not only on the feasibly observable quantities $M + M_s$ and $F + F_s$. As of now, there is no known control depending only on $M + M_s$ and $F + F_s$.

2.1. Contributions of this paper

In this work we use a deep Reinforcement Learning (RL) approach to construct control feedback laws and compare them with the existing feedback controls that were deduced in previous work.

Our approach works in three steps:

1. We discretize the equations in a numerical scheme,
2. We train an RL model to obtain a numerical control feedback based on this numerical scheme,
3. We recover from the numerical control feedback an explicit mathematical control. We then perform several tests to ensure that the explicit control is efficient.

This is detailed in Section 3. We use this approach to construct three types of control feedback laws:

- A feedback control depending on M only,

- A feedback control depending on $M + M_s$ (total number of males) only,
- A feedback control depending on $M + M_s$ and $F + F_s$ (total number of males and of females).

These different types of controls are studied in Sections 4, 5 and 6 respectively.

3. Reinforcement learning based control

Reinforcement Learning (RL) is a subfield of machine learning concerned with training *agents* to make decisions in an environment to maximize a long-term *reward* function. From a control perspective this can equivalently be seen as finding an optimal control for a cost function over the trajectories. We first define these concepts more formally and discuss the RL algorithm we use in Section 3.1, then explain how we apply this formalism to our specific problem in Section 3.2. After that, we discuss our approach in Section 3.3 and finally go through experiment details in Section 3.4.

3.1. RL Background

We model the environment using the common formalism of a Partially-Observable Markov Decision Process (POMDP) [19] $\mathcal{M} = (\mathcal{S}, \mathcal{A}, T, R, \gamma, \mu, \Omega, \mathcal{O})$ where $\mathcal{S} \subseteq \mathbb{R}^n$ is a set of states, $\mathcal{A} \subseteq \mathbb{R}^m$ a set of actions, $T : \mathcal{S} \times \mathcal{A} \to \Delta\mathcal{S}$ is the state transition function (ie. $T(s'|s, a)$ is the probability of transitioning to state s' given state s and action a), $R : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ is the reward function, $\gamma \in [0, 1)$ is the discount factor, $\mu \in \Delta(\mathcal{S})$ is the initial state distribution, $\Omega \subseteq \mathbb{R}^p$ is a set of observations of the hidden state, and $\mathcal{O} : \mathcal{S} \to \Delta(\Omega)$ is the observation distribution (ie. $\mathcal{O}(o|s)$ is the probability of getting observation o given current state s). Note that given a set X, $\Delta(X)$ denotes the set of probability distributions over X.

The goal for the agent is to learn a policy $\pi_\theta : \Omega \to \Delta(\mathcal{A})$ (stochastic in our case) mapping observations to actions, where θ are the parameters of the policy (typically the weights of a neural network in the case of deep RL), which maximizes the expected discounted sum of rewards

$$J(\pi_\theta) = \mathbb{E}_{\tau \sim (\pi_\theta, \mathcal{M})} \left[\sum_{t=0}^{\infty} \gamma^t r_t \right] \tag{3.1}$$

where the expectation is taken over all trajectories $\tau = (s_t, a_t, r_t)_{t\geq 0}$ generated by the current policy π_θ acting in the POMDP $\mathcal{M}$. Note that maximizing this objective $J(\pi_\theta)$ is analogous to the standard minimizing of the cost function $-J(\pi_\theta)$ in control theory.

One common way to minimize this cost is to use policy gradient methods, which directly optimize the policy parameters by estimating the gradient of the expected cumulative reward. The basic policy gradient algorithm updates the policy parameters in the direction of the estimated gradient to increase the likelihood of actions that lead to higher rewards. To improve stability and convergence, several techniques have been developed, such as Trust Region Policy Optimization (TRPO) [20] and Proximal Policy Optimization (PPO) [21]. TRPO limits the policy update step size by constraining the divergence between the new policy and the old policy. PPO introduces a simple surrogate objective function that includes a clipping mechanism to prevent large policy updates and improve sample efficiency.

3.2. Defining the POMDP

In this section, we define the specific POMDP problem formulation that we consider in this work: states, observations, actions and rewards, as well as the initial state distribution and state transition function that model the system. We present all the variants that we have considered, and the specific settings used for each controller will be specified in the respective feedback law parts in Sections 4, 5 and 6.

State space The states for the POMDP are exactly the states of the SIT model, that is, $E(t)$, $M(t)$, $F(t)$ and $M_s(t)$, all nonnegative and introduced in Section. 2. Thus our state space can be formally written as $\mathcal{S} = \mathbb{R}^4_{\geq 0}$.

Observation space and distribution To account for real-world partial observability constraints, the control does not have access to the full state, but only to some partial observation of it. In this work, we consider three different types of observations which we analyze and compare in their respective sections:

- Section 4 considers an observation consisting of only the number of wild males. Formally, the observation space is $\Omega = \mathbb{R}_{\geq 0}$ and the function mapping states to observations (which in this work is deterministic) is $\mathcal{O}(E(t), M(t), F(t), M_s(t)) = M(t)$.

- Section 5 considers an observation consisting of the total number of males. Formally, $\Omega = \mathbb{R}_{\geq 0}$ and $\mathcal{O}(E(t), M(t), F(t), M_s(t)) = M(t) + M_s(t)$.
- Section 6 considers an observation consisting of the total number of males and the total number of females. Formally, $\Omega = \mathbb{R}^2_{\geq 0}$ and $\mathcal{O}(E(t), M(t), F(t), M_s(t)) = (M(t) + M_s(t), F(t) + F_s(t))$, where $F_s(t) = F(t)M_s(t)/M(t)$ was defined in Section 2.

All of the observations that we consider are quantities that we are able to measure in the real world. This is in contrast to the backstepping control (2.17), which requires measurements of E, M and M_s, which is currently not possible in the real world. Besides, these observations may grow very large and thus can span quite a large range. As such, to enable the neural network to observe large values while still being able to discriminate between smaller values, we input each observation into the control at different orders of magnitude. For instance, $M + M_s$ is inputted at several scales, namely $M + M_s$ becomes $\min(M + M_s, k)$ for k ranging from 5 to $100K$, and similarly for $F + F_s$. We normalize all observations so that they lie within $[0, 1]$. This has proved important to help with convergence during training: normalizing inputs is a common preprocessing technique, and inputting each observation at several scales further helps with training stability without technically adding more inputs to the control. We also consider adding memory of the past observations as an input to the control to enable the neural network to internally build a kind of observer, although it makes it significantly more complex to convert into an explicit feedback.

Action space Our action space $\mathcal{A} = [-1, 1]$ corresponds to a single action $a(t) \in \mathcal{A}$ that is remapped to the range $[0, 10K]$ and then directly inputted into the model equations (2.1)-(2.4) through $u(t) = 5K(a(t) + 1)$. Having the neural network model output a normalized action is a common technique for more robust training, akin to normalization of the inputs (here we output a normalized action, then scale it up).

Initial state distribution We sample the initial state uniformly between 0 and $10K$, which corresponds to physically-realistic values for the states. Namely, $(E(0), M(0), F(0), M_s(0)) \sim \mu = \mathcal{U}([0, 10K]^4)$.

State transition function The state transition function T, that maps a current state and action to a next state, is deterministic in this work and implicitly defined by the ODE system (2.1)–(2.4): we discretize the state $x_t = (E(t), M(t), F(t), M_s(t))$, and at each time step compute the next state through a simple Euler update $x_{t+1} = x_t + \dot{x}_t \, dt$ (with an abuse of notation, since the update is technically done on each of the four individual states), where the action $u_t = u(t)$ comes in the definition of $\dot{M}_s$. The value for dt is indicated in Section 3.4. Thus, the transition function can be written as $T(x_t, u_t) = x_{t+1}$.

Reward function The optimization criterion is usually the most crucial part of the RL learning process. Our reward function, which we aim to maximize over time as per Eq. (3.1), takes the following form at time step t:

$$r_t = c_1 \left(\|E(t)\|_2 + \|M(t)\|_2 + \|F(t)\|_2\right) + c_2(t)\|M_s(t)\|_2 \tag{3.2}$$

with

$$c_2(t) = \begin{cases} c_3 & \text{if } t < 0.9T, \\ c_3 + c_4 & \text{otherwise,} \end{cases} \tag{3.3}$$

where T is the simulation horizon. In a continuous control framework this would correspond to maximizing the functional

$$J(u) = \int_0^T c_1\|E(t), M(t), F(t)\|_2 + c_2(t)\|M_s(t)\|_2 dt. \tag{3.4}$$

The specific values we use for c_1, c_3 and c_4 are indicated in Section 3.4, and are derived empirically and heuristically through hyperparameter tuning. The reason for the specific shape of this reward function Eq. (3.2) is as follows: ideally, we would simply penalize all the states equally and have $r_t = C\left(\|E(t)\|_2 + \|M(t)\|_2 + \|F(t)\|_2 + \|M_s(t)\|_2\right)$ in order to drive all the states to zero. This is what we initially tried; however, for practical reasons, we found this objective to be harder to minimize due to the fact that M_s can typically take much larger values than the 3 other states. Indeed, one of the difficulties of this control problem is that M_s should necessarily take high values to be able to bring M, F and E closer to 0, given (2.1)–(2.4), and this results in a delay between the control action its effect on the mosquito population.

A practical interpretation is that the females have no preferences between sterile males and fertile males, hence M_s should be larger than M to have an influence.

As a result, our choice of $c_2(t)$ corresponds to penalizing M_s with less amplitude than we penalize the other 3 states; nevertheless around the end of the simulation we increase the penalty on M_s to encourage convergence to 0. This allows for a high action at the start to grow M_s without excessive penalty, which in turn makes the other states decrease, then a slowly decreasing action so that M_s converges 0. The main challenge is to not decrease M_s too quickly, or the other states would increase again. We empirically found that this reward design led to increased training stability: it is designed in a way to guide the controller's, which is initially random, to a reasonable behavior more quickly. We note that the rewards are also normalized by K to lie within a reasonable range.

Additionally, in order to artificially reduce the horizon and make training more robust (RL usually suffers from overly long simulations, as it makes optimization a reward over time much more complex), we repeat each action several times, meaning that for each environment step we use the same action to run n_{sims} simulation steps. Finally, the value for γ is indicated in Section 3.4.

3.3. Method

In our work, we aim to train control policies using RL to regulate a dynamic system described by ordinary differential equations. To do that, we design an environment that simulates the behavior of the ODE system, allowing us to interact with it in a controlled manner. Our approach is summarized in Fig. 1. We start by discretizing our system of equations (2.1)-(2.4), and use those dynamics to create a simulation of our model. We use it to create an environment by implementing the observations, actions and rewards described in Section 3.2. Using this code, we train an RL agent that learns to maximize the objective function we assign it through many simulations, using the PPO algorithm mentioned in Section 3.1. Once the policy has converged, we can evaluate it on any simulation, in particular we can query a control u for any current state $[E, M, F, M_s]$. Since we only trained policies with 1 or 2 observations, we can plot the action as a function of the input in 1D or 2D space. This allows us to perform a regression and empirically write a simple explicit control that has the same general shape as the neural network control (see Sections 4 and 6). Finally, we simulate this explicit control to ensure that it still stabilizes the system, and analyze its properties and robustness.

3.4. Experiment details

Here we present the experiment details in greater details. To implement the PPO algorithm we use to train our RL policies, we use Stable Baselines 3 [22] (version 1.6.2 in Python 3.8), a popular RL library that provides a collection of state-of-the-art algorithm implementations, as well as various tools for RL research. The models in Sections 4, 5 and 6 are trained for 10 million environment timesteps (or 7 billion simulation timesteps) on 12 CPUs, which takes about 7 hours. During each iteration, we collect 12288 (1024 per CPU) environment steps, then run 5 epochs of optimization with a batch size of 1024. The agent's policy is a fully-connected neural network with 2 hidden layers of 256 neurons each, with tanh non-linearities between each layer, outputting the mean and standard deviation of a normal distribution that is then used to sample the action. More formally, for a given observation vector o_t, the neural network policy outputs $(\mu_t, \sigma_t) = \pi_\theta(o_t)$ and the action is sampled as $a_t \sim \mathcal{N}(\mu_t, \sigma_t)$. We train with a learning rate of 3×10^{-4}, discount coefficient of $\gamma = 0.99$, and all other hyperparameters are left to their default values.

We run each simulation for $T = 1000$ days, with a timestep $dt = 0.01$ days, and each action is repeated $n_{\text{sims}} = 500$ times, meaning that the environment horizon is 200 steps. For each simulation, the initial condition is uniformly sampled between 0 and $10K$: $E(0), M(0), F(0), M_s(0) \sim \mathcal{U}(0, 10K)$. For our reward function, we use coefficients $c_1 = 0.1$, $c_3 = 0.001$ and $c_4 = 0.01$. For the model, we use the parameters given in Table .4.

4. Feedback RL control using M

In the literature, one of the previous approaches consist in using a linear feedback control that only depends on $M(t)$ [9], that is $u(t) = f(M)$ where f is linear instead of the control law (2.5). They observe that the following linear control feedback law

$$u(t) = \alpha M(t), \tag{4.1}$$

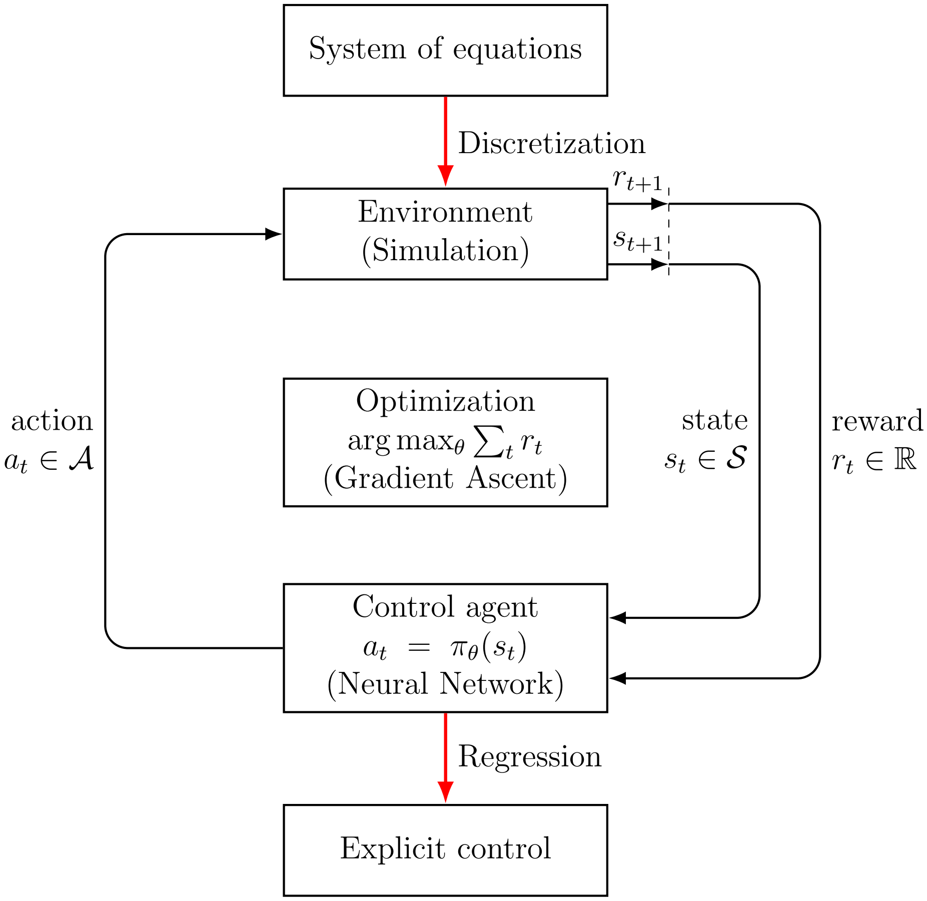

Figure 1: Diagram representing the (simplified) procedure by which we simulate our model in an environment that is used to train an RL agent, whose policy we then convert into an explicit control. The policy π_θ is modeled by a neural network with parameters θ, which takes a state as an input and outputs an action (or a distribution over actions in the stochastic case). The neural network is then optimized to maximize the sum of rewards it obtains over simulations.

seems to stabilize the system as long as

$$\alpha > \frac{(\beta_E \nu \nu_E - (\nu_E + \delta_E)\delta_F)\delta_s}{(\nu_E + \delta_E)\delta_F}. \tag{4.2}$$

In this section, we use our RL procedure to deduce a potentially nonlinear feedback law that similarly only depends on $M(t)$.

We force the feedback control to go to 0 when the number of wild male goes to 0 by searching the control under the form

$$u(t) = \min\left(\min\left(f_{RL}\left(\frac{M(t)}{K}\right), \alpha_M\right) M(t), u_M\right), \tag{4.3}$$

where α_M is a chosen constant, u_M is the maximal value of the control allowed which is dictated by physical constraints, and f_{RL} is the function searched by the RL model. This f_{RL} is searched using the procedure described in Section 3 and using the cost function

$$\begin{aligned} J(u) &= \int_0^T r_t \, dt \\ r_t &= \frac{\|E(t), M(t), F(t)\|}{K} \\ &+ q_1 \left(\frac{M_S(t)}{K} + \max(0, (\frac{M_S(t)}{K} - 30))^2\right) \end{aligned} \tag{4.4}$$

where q_1 is a chosen constant, typically much smaller than 1, and T is a chosen horizon. After training, the RL model converges and the optimal numerical f_{RL} obtained has a relatively simple form, shown in Figure 2, which happens to be exactly piecewise linear. This allows us to deduce the following nonlinear control feedback law for the system

$$u(t) = \min\left(\bar{u}(t), u_M\right) \tag{4.5}$$

with

$$\bar{u}(t) = \min\left[\max\left(\alpha_1 - \alpha_2 \frac{M(t)}{K}, 0\right), \alpha_M\right] M(t), \tag{4.6}$$

where α_1 and α_2 are positive constants. For the values in Table .4 with $\alpha_M = 15$, $\alpha_1 \approx 16$ and $\alpha_2 \approx 12$.

Remark 4.1. *The RL control* (4.5) *tends to be linear when* $K \to +\infty$.

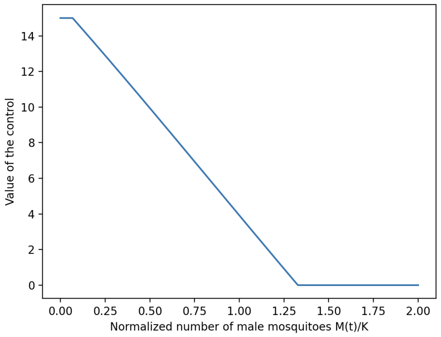

Figure 2: Value of f_{RL} as a function of $M(t)/K$ when $\alpha_M = 15$ and parameters as in Table .4.

	200 days	400 days	800 days	2000 days
average $\|E\|+\|M\|+\|F\|$	$8.7\ 10^4$	$3.3\ \ 10^4$	$1.3\ \ 10^3$	$1.5\ \ 10^{-2}$
variance $\|E\|+\|M\|+\|F\|$	$3.6\ \ 10^7$	$1.4\ \ 10^7$	$3.7\ \ 10^4$	$5.5\ \ 10^{-6}$
average $\|M_s\|$	$1.6\ \ 10^6$	$1.5\ \ 10^6$	$8.4\ \ 10^4$	1.0
variance $\|M_s\|$	$8.2\ \ 10^9$	$1.3\ \ 10^{10}$	$1.6\ \ 10^8$	$2.5\ 10^{-3}$
maximum $\|E\|+\|M\|+\|F\|$	$9.3\ 10^4$	$3.8\ \ 10^4$	$1.5\ \ 10^3$	$1.8\ \ 10^{-2}$

Table 1: Average, variance and maximum of the different components over 100 simulations, with $\alpha_M > \alpha_1 = 13$, $q_1 = 0.004$, $u_M = 10K$ and the parameters of Table .4.

In Figure 3 we present 100 numerical simulations of the closed-loop system when using this explicit control. Each simulation has a different initial condition taken uniformly at random with each state having values in $[0, 10K]$. We can see that all the components of the state of the system converge quickly to 0 after 800 days, while the main components of the system (not sterile males M, females F and eggs E) converge much faster to 0.

In Table 1 we show the average and variance of the states E, M, F and M_s at different times over 10000 numerical simulations as well as the maximal absolute value of each state.

5. RL control using $M + M_s$

In practice, measuring M is a challenge. In a wild population of mosquitoes, synthetic versions of female insect pheromones are released to attract and capture male insects. This allows for measuring $M + M_s$, however there is no easy way to distinguish between the wild male mosquitoes and the released

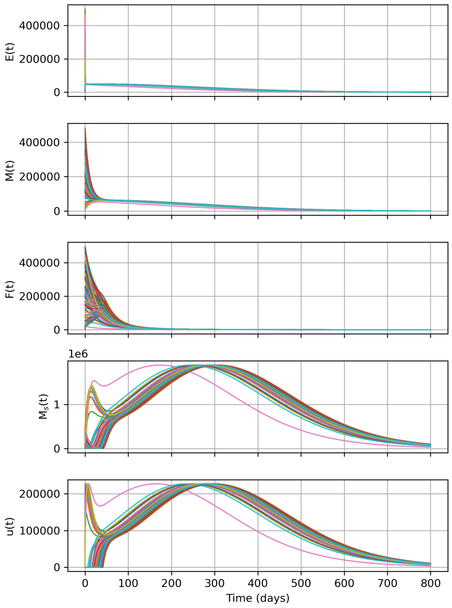

Figure 3: Representation of the state in a 100 simulations, with $\alpha_M = 15$, $q_1 = 0.004$, $T = 2000$ days, $u_M = 10K$, and the parameters of Table .4. Each simulation corresponds to one color.

sterilized male. Another possible measurement can be done by placing simple traps for adult mosquitoes in the wild, then differentiating them based on physical attributes such as their size to separately count total males $M + M_s$ and total females $F + F_s$.

This motivates the search, in practice, for a control that would only depend on $M + M_s$ and $F + F_s$. In this section, we consider using only total males $M + M_s$, since this quantity is more easily measured. In [9], it was conjectured that a linearly dependent feedback of $M + M_s$ stabilizes the dynamics at the origin. However this control lacks robustness (see Figure 4) with respect to the parameters of the model or of the controller. Indeed the control has the linear form

$$u(t) = \beta(M(t) + M_s(t)), \tag{5.1}$$

and can only work if β satisfy

$$\left(\frac{\beta_E \nu \nu_E - (\nu_E + \delta_E)\delta_F}{\beta_E \nu \nu_E} \right) \delta_s \leq \beta < \delta_s. \tag{5.2}$$

With the parameters set in the Table .4, this condition becomes

$$0.118 \leq \beta < 0.12, \tag{5.3}$$

which means that even a tiny imprecision on the model parameters renders the control inefficient.

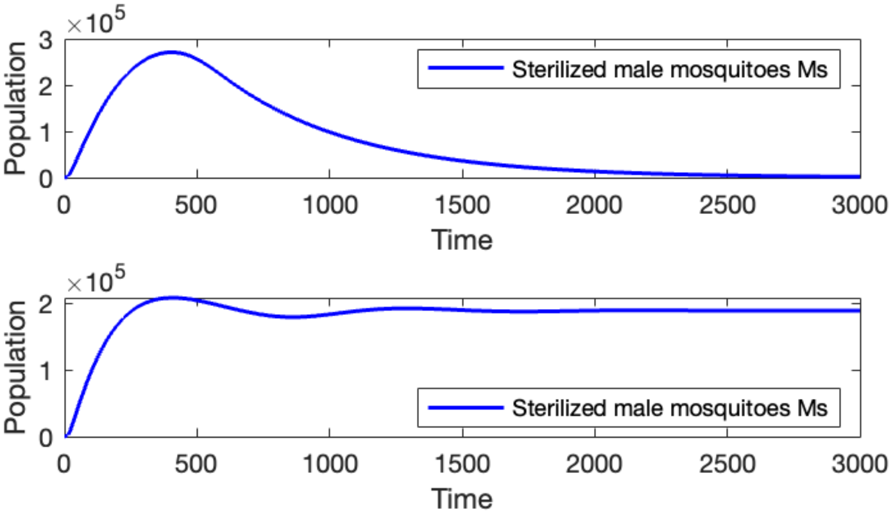

Figure 4: Evolution of the sterilized male population as a function of time when applying the feedback law $u(t) = \beta(M(t)+M_s(t))$ with $\beta = 0.118$ (top) or with $\beta = 0.116$ (bottom), over 3000 days.

We tested our RL control procedure in this framework, having only access to $M + M_s$ at the current time t. Even when allowed a nonlinear control, the model does not converge to an efficient control after 1000 iterations of training, with the same setup used to train the other controls. Figure 5 illustrates the behavior of the trained control, which outputs an approximately constant control $u(t) \approx 200000$ yielding a stabilization of the populations of mosquitoes, except for the released sterilized male mosquitoes M_s which the control does not manage to reduce with the limited information it has about the state of the system. This demonstrates the lack of robustness mentioned earlier.

Nevertheless, when allowing the control to depend not only on $M + M_s$ at the given time t but also on previous times $s \leq t$, that is to say enabling the control with memory of past values of $M + M_s$, the RL policy converges to what seems to be a robust numerical control. The rationale behind giving

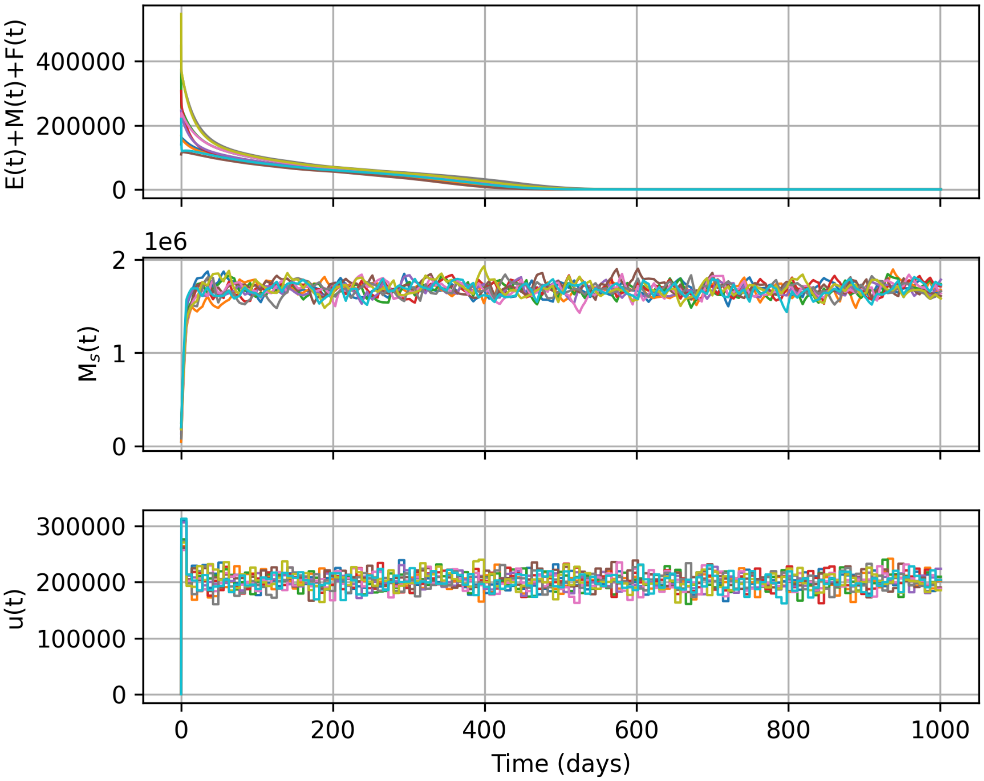

Figure 5: Evolution of the population of mosquitoes, represented as $E(t) + M(t) + E(t)$ and $M_s(t)$, and of the control $u(t)$, as a function of time, over 10 simulations with random initial conditions, using an RL control that only has access to $M + M_s(t)$ at the current time t.

the model access to a memory of the measurements is to allow it to construct internally a kind of observer. We train the model by allowing it to measure the state of the system every 7 days and to keep the memory of the 26 most recent measurements (so 6 months of measurements). This measurement frequency corresponds to what is possible in practice [23]. Besides, the control also takes a single action every 7 days. Since this control has strictly more information than the control with only $M + M_s$ at time t, we expect it to perform better as long as the training procedure is stable. It is however not obvious that adding this memory of past observations would be enough to stabilize the system, but Figure 6 shows numerically that it appears to be sufficient, across 100 numerical simulations of the closed-loop system using this numerical control with initial conditions chosen at random in $[0, 5K]^4$.

6. RL control using $M + M_s$ and $F + F_s$

Using past states can be a challenge to find a mathematical formula from the numerical control. Indeed, when the control depends on the past state, the feedback that is searched is not anymore a function of a finite-dimensional vector but a functional on an infinite dimensional space containing portions of the trajectories (e.g. $(M + M_s(s))_{s\in[t-\tau,t]}$). This makes the symbolic regression a challenge. For this reason we try to find a control using $M+M_S$ and $F+F_S$ only at the current time, that is a control of the form (2.5). These two quantities can be measured in real life, and lead to a much simpler model than the model with memory that has 26 inputs.

A first control. In this framework, the RL model is trained as described in Section 3 and converges to a numerical control that we represent in Figure 7 as a function of $M + M_s$ and $F + F_s$. We see that the plot of the control in linear scale is not really informative (see Fig. 7 left). However in log scale the expression of the control seems clearer (see Fig. 7 right) and clearly has two parts. In each of them the control seems to be close to a bang-bang control with a thin transition. With a simple regression we approximate this numerical control with

$$u_{\text{reg}}(M + M_s, F + F_s) = \begin{cases} u_{\text{reg}}^{\text{left}} & \text{if } M + M_s < 200, \\ u_{\text{reg}}^{\text{right}} & \text{otherwise,} \end{cases} \tag{6.1}$$

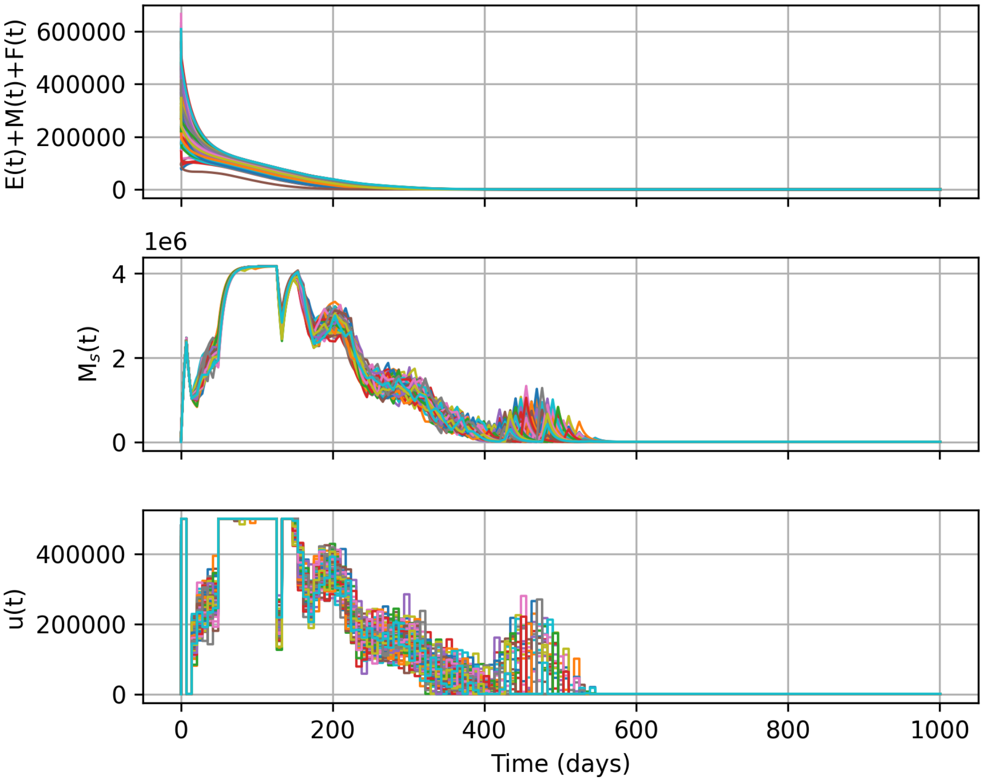

Figure 6: Evolution of the population of mosquitoes, represented as $E(t) + M(t) + E(t)$ and $M_s(t)$, and of the control $u(t)$, as a function of time, over 100 simulations with random initial conditions, using an RL control that only has access to measurements of $M + M_s$ over the past 6 months, and every week obtains a new measurement and take a new action.

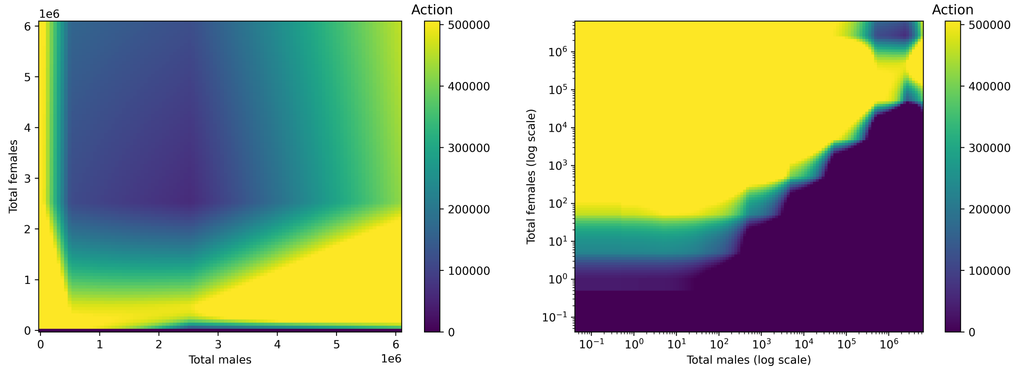

Figure 7: Heatmap of the model's action $u(M + M_S, F + F_s)$ as a function of total males and total females, in linear scale (left) and logarithmic scale (right).

for $M + M_s, F + F_s > 0$, where

$$u_{\text{reg}}^{\text{left}} = \begin{cases} u_{\min} & \text{if } \log \frac{200}{F+F_s} > 4, \\ u_{\max}\left(4 - \log \frac{200}{F+F_s}\right) & \text{if } 4 \geq \log \frac{200}{F+F_s} > 3, \\ u_{\max} & \text{otherwise, and} \end{cases} \tag{6.2}$$

$$u_{\text{reg}}^{\text{right}} = \begin{cases} u_{\min} & \text{if } \log \frac{M+M_s}{F+F_s} > 4, \\ u_{\max}\left(4 - \log \frac{M+M_s}{F+F_s}\right) & \text{if } 4 \geq \log \frac{M+M_s}{F+F_s} > 3, \\ u_{\max} & \text{otherwise.} \end{cases} \tag{6.3}$$

Table 2 shows that this explicit control is still able to quickly stabilize the state over a wide range of initial conditions.

During training and testing, the numerical control feedback law includes by default a small noise. This ensures some robustness of the control and a good exploration. We tested the mathematical control we derived (given in (6.1)) with and without noise. To our surprise, the control with a small noise does seem to ensure the asymptotic stability, whereas the control without any noise does not seem to. Indeed, without noise, the control seems to have a cyclic behavior and never converges (see Figure 8 (left)). When adding a small noise, however, the stability is restored (see Figure 8 (right)). The explication to this apparent paradox is that having exactly $u_{\min} = 0$ in one of the branches of the control given in (6.1) is seemingly too strong to allow the model to converge completely. Replacing this value with $u_{\min} > 0$ for a small $u_{\min}$ (typically $u_{\min} = 5$) allows to stabilize the system without noise. In the

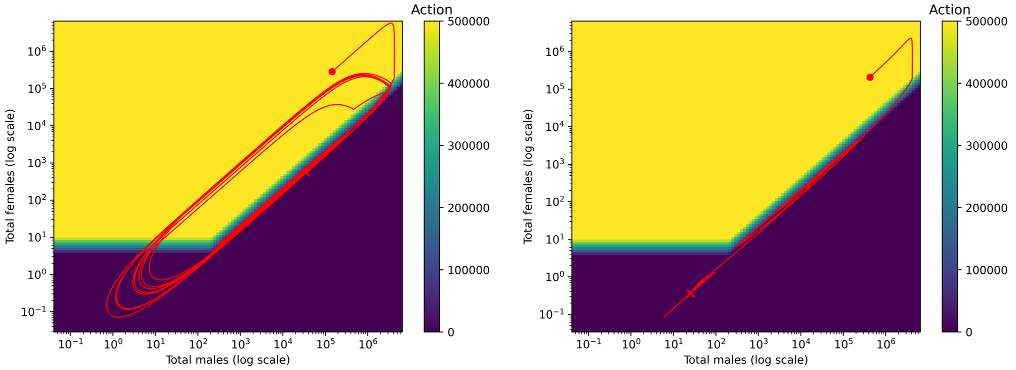

Figure 8: Heatmap of the regression model's action $u_{\text{reg}}(M+M_s, F+F_s)$ as a function of total males and total females. A state-space trajectory is plotted in red, with the dot indicating initial state and the cross final state, comparing the no-noise case (left) with the case when a small noise $\mu \sim \mathcal{N}(0,5)$ is added on top of the action (right).

system with noise, because the control u has to be positive, the noise has the effect of increasing in average the effective value of $u_{\min}$ of the control (6.1), which explains the apparent stabilization. Of course when setting $u_{\min}$ to a small value, the equilibrium that is stabilized is not anymore $(0,0,0,0)$ but $(0,0,0,u_{\min}/\delta_s)$ which is very close to it, remains very acceptable in practice compared to the uncontrolled attraction point (especially as only the density of the sterile male mosquitoes does not converge to 0 and, moreover, this density converges to a small value) and answers the mathematical problem described in Section 2.

A simpler control. We decided to simplify the control found by the RL algorithm. We wanted to see if there is really a need for a different regime when there is only very few mosquitoes. The motivation behind this is that this region does not influence much the cost function that the RL algorithm tries to optimize and the control might be less precise on this part. This leads to the following simplified formula for the feedback control:

$$v_{\text{reg}}(M+M_s, F+F_s) = \begin{cases} u_{\min} & \text{if } \log \frac{M+M_s}{F+F_s} > \alpha_2, \\ u_{\max} & \text{otherwise,} \end{cases} \tag{6.4}$$

for $M+M_s, F+F_s > 0$, where $\alpha_2 = 4$ is a constant found by regression.

In this case again, the control with $u_{\min} = 0$ and no noise does not seem to ever lead to the convergence of the state $(E(t), M(t), F(t), M_s(t))$,

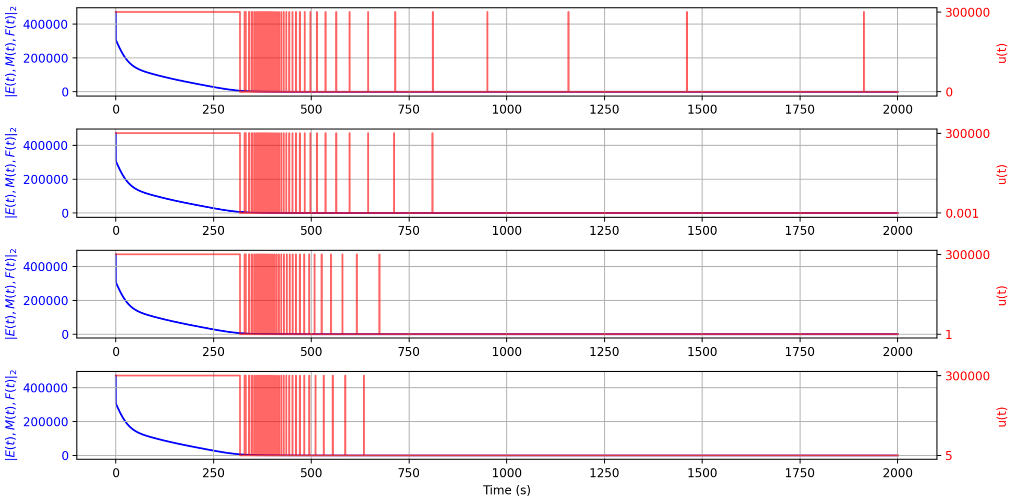

Figure 9: Norm of the states $\|E(t), M(t), F(t)\|_2$ (blue) and control v_{reg} (red) as a function of time for different values of $u_{\min}$ (0, 0.001, 1, and 5 respectively from top to bottom) and $u_{\max} = 300000$, over 2000 days and with the same initial condition. When the minimum control $u_{\min}$ is 0, the control occasionally (more and more rarely) outputs the maximum action (corresponding to the spikes) to prevent the states from going back up. However, when $u_{\min} > 0$, after some time this minimal action is sufficient to stabilize the state around 0 and no more spikes are observed.

with a notable difference however: there is no purely cyclic behavior and $(E(t), M(t), F(t))$ convergences rapidly to $(0, 0, 0)$. The obstacle to the convergence manifests in large peaks that appear in the control feedback and are increasingly spaced in time, as evidenced in Figure 9. When choosing again a small $u_{\min} > 0$ the convergence to the equilibrium is recovered and seems to work for arbitrarily small $u_{\min} > 0$ (see Figure 9).

We numerically demonstrate that this simplified control seems stabilizing over a wide range of initial conditions. Figure 10 shows the evolution of the control and states for randomly sampled initial conditions. The control appears robust, quickly stabilizing the state in every simulation. Table 3 shows corresponding statistics, illustrating that the state rapidly converges to 0 (with the exception of M_s which takes longer to converge to ensure the other states don't regrow) with very small variance. Besides, we can notice by comparing Table 2 and Table 3 that the simplified control leads to quicker convergence of the state than the control defined in (6.1)–(6.3).

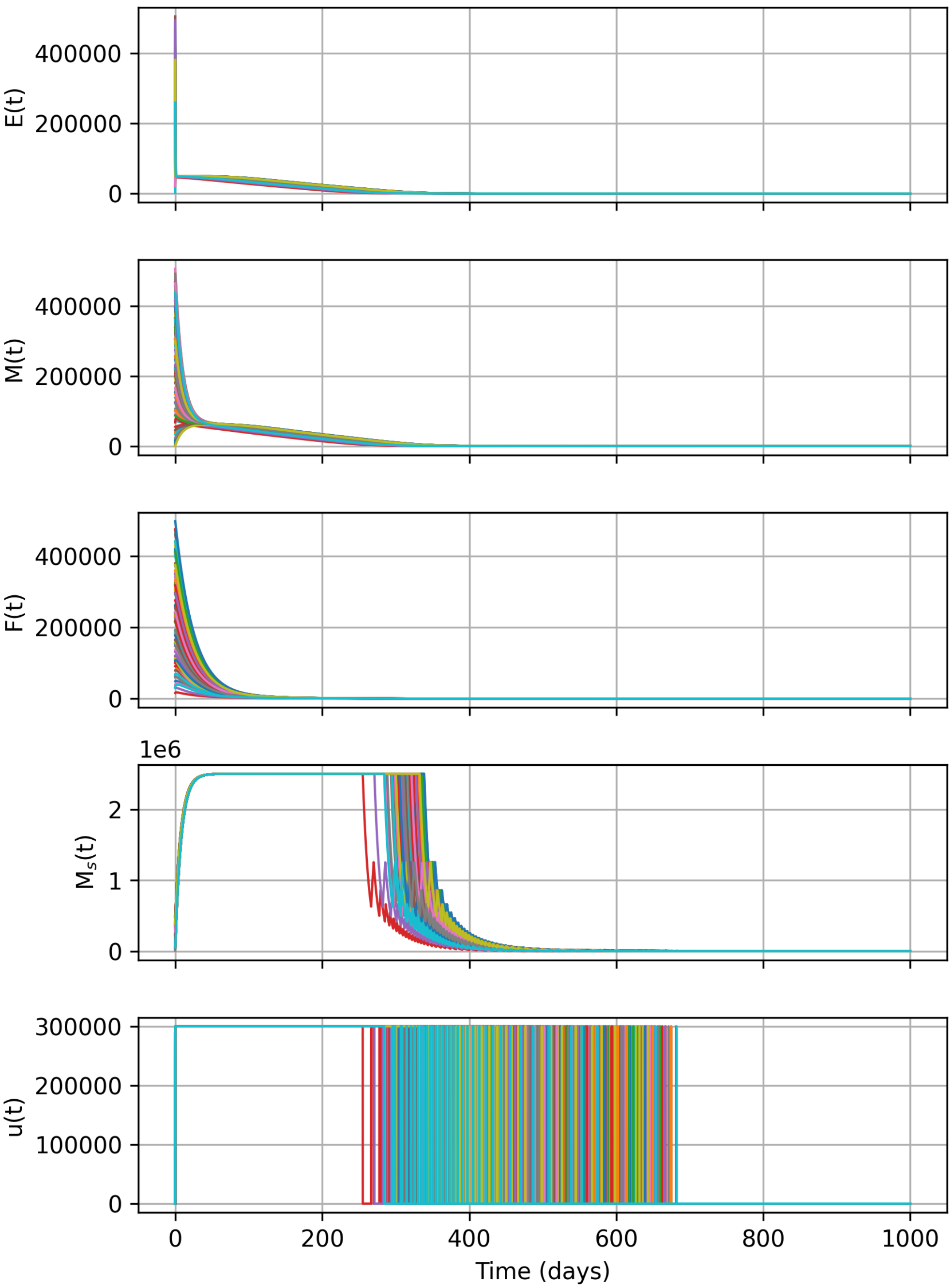

Figure 10: States and control v_{reg} with $u_{\min} = 5$ and $u_{\max} = 300000$ over a duration of 1000 days for 100 simulations with random initial conditions in $[0, 10K]^4$. Each color correspond to a simulation.

	200 days	400 days	600 days	800 days
average $\|E\| + \|M\| + \|F\|$	50,801.64	8,020.47	96.25	0.60
variance $\|E\| + \|M\| + \|F\|$	45,422,159	4,012,119	1,394	0.02
maximum $\|E\| + \|M\| + \|F\|$	59,026.78	10,455.31	149.03	0.80
average $\|M_s\|$	2,207,795.88	693,675.84	16,743.12	50.27
variance $\|M_s\|$	53,101,242,728	13,102,504,436	41,355,608	81.24
maximum $\|M_s\|$	2,473,954.23	822,154.59	25,783.57	70.59

Table 2: Statistics over 100 simulations with random initial conditions in $[0, 10K]^4$ using control u_{reg} (see (6.1)) with $u_{\min} = 5$ and $u_{\max} = 300000$ over a duration of 800 days.

	200 days	400 days	600 days	800 days
average $\|E\| + \|M\| + \|F\|$	48,806.91	688.68	2.47	0.002
variance $\|E\| + \|M\| + \|F\|$	75,826,146	78,547.67	1.31	1.61×10^{-6}
maximum $\|E\| + \|M\| + \|F\|$	59,079.04	1,130.92	4.37	0.006
average $\|M_s\|$	2,500,000	129,308.46	2,204.39	41.67
variance $\|M_s\|$	3.07×10^{-11}	2,949,520,902	5,197,430.54	2.17×10^{-7}
maximum $\|M_s\|$	2,500,000	248,387.02	10,757.19	41.67

Table 3: Statistics over 100 simulations with random initial conditions in $[0, 10K]^4$ using control v_{reg} (see (6.4)) with $u_{\min} = 5$ and $u_{\max} = 300000$ over a duration of 800 days.

7. Conclusion

In this paper, we have explored various feedback control strategies for a population control problem involving mosquitoes. We have investigated the use of nonlinear feedback control strategies that depend only on measurements. We have first considered the case where only the wild male mosquito density is measured, then the case where the male mosquito density is measured, and finally the case where both the male and female mosquito densities are measured.

In particular, we have used reinforcement learning to develop a control strategy that depends only on the total number of male and sterile male mosquitoes, as well as the total number of female and sterile female mosquitoes, at a given time. This control strategy appears to be robust and effective, and has the potential to be applied in practice to control mosquito populations and prevent the spread of diseases.

Our results highlight the usefulness of machine learning and control theory in developing effective control strategies for complex biological systems. Further research in this field could lead to even more powerful techniques for controlling populations of pests and disease vectors.

Acknowledgements

The authors would like to thank Luis Almeida for fruitful discussions about the SIT model. This work was granted access to the HPC resources of IDRIS under the allocation 2023-AD011014527 made by GENCI.

During the preparation of this work the author(s) used an LLM in order to correct spellings in the introduction. After using this tool/service, the author(s) reviewed and edited the content as needed and take(s) full responsibility for the content of the publication.

References

[1] L. Almeida, M. Duprez, Y. Privat, N. Vauchelet, Mosquito population control strategies for fighting against arboviruses, Mathematical Biosciences and Engineering 16 (6) (2019) 6274–6297.

[2] N. Alphey, L. Alphey, M. B. Bonsall, A model framework to estimate impact and cost of genetics-based sterile insect methods for dengue vector control, PLoS One 6 (10) (2011) e25384.

	Parameter name	**Typical interval**	**Value**	**Unit**
β_E	Effective fecundity	[7.46, 14.85]	8	Day^{-1}
ν_E	Hatching parameter	[0.005, 0.25]	0.25	Day^{-1}
δ_E	Aquatic phase death rate	[0.023, 0.046]	0.03	Day^{-1}
δ_F	Female death rate	[0.033, 0.046]	0.04	Day^{-1}
δ_M	Males death rate	[0.077, 0.139]	0.1	Day^{-1}
δ_s	Sterilized male death rate	-	0.12	Day^{-1}
ν	Probability of emergence	-	0.49	
K	Environmental capacity for eggs	-	50000	

Table .4: Parameters for the system (2.1)-(2.4). This includes typical value ranges, which can be found in a population of Aedes polynesiensis in French Polynesia, as well as our chosen values in this work. Further details can be found in [24, 25, 26, 27, 28, 29, 30, 31].

[3] P.-A. Bliman, M. S. Aronna, F. C. Coelho, M. A. da Silva, Ensuring successful introduction of wolbachia in natural populations of aedes aegypti by means of feedback control, Journal of mathematical biology 76 (2018) 1269–1300.

[4] P.-A. Bliman, Feedback control principles for biological control of dengue vectors, in: 2019 18th European Control Conference (ECC), IEEE, 2019, pp. 1659–1664.

[5] L. Almeida, M. Duprez, Y. Privat, N. Vauchelet, Optimal control strategies for the sterile mosquitoes technique, Journal of Differential Equations 311 (2022) 229–266.

[6] A. Cristofaro, L. Rossi, Backstepping control for the sterile mosquitoes technique: stabilization of extinction equilibrium, preprintPreprint (2023).

[7] P.-A. Bliman, D. Cardona-Salgado, Y. Dumont, O. Vasilieva, Implementation of control strategies for sterile insect techniques, Math. Biosci. 314 (2019) 43–60. `doi:10.1016/j.mbs.2019.06.002`.
URL `https://doi.org/10.1016/j.mbs.2019.06.002`

[8] P.-A. Bliman, Y. Dumont, Robust control strategy by the Sterile Insect Technique for reducing epidemiological risk in presence of vector migration, Math. Biosci. 350 (2022) Paper No. 108856, 23. `doi:10.1016/j.mbs.2022.108856`.
URL `https://doi.org/10.1016/j.mbs.2022.108856`

[9] K. Agbo Bidi, L. Almeida, J.-M. Coron, Global stabilization of sterile insect technique model by feedback laws, arXiv, 2307.00846 (2023).

[10] M. Krstic, J. Kanellakopoulos, P. Kokotovic, Nonlinear and adaptive control design, Adaptive and Cognitive Dynamic Systems: Signal Processing, Learning, Communications and Control, John Wiley and sons, 1995.

[11] E. D. Sontag, Mathematical control theory, 2nd Edition, Vol. 6 of Texts in Applied Mathematics, Springer-Verlag, New York, 1998, deterministic finite-dimensional systems. `doi:10.1007/978-1-4612-0577-7`.
URL `https://doi.org/10.1007/978-1-4612-0577-7`

[12] J.-M. Coron, Control and nonlinearity, Vol. 136 of Mathematical Surveys and Monographs, American Mathematical Society, Providence, RI, 2007. doi:10.1090/surv/136.
URL https://doi.org/10.1090/surv/136

[13] S. Levine, C. Finn, T. Darrell, P. Abbeel, End-to-end training of deep visuomotor policies, The Journal of Machine Learning Research 17 (1) (2016) 1334–1373.

[14] S. Gu, E. Holly, T. Lillicrap, S. Levine, Deep reinforcement learning for robotic manipulation with asynchronous off-policy updates, in: 2017 IEEE International Conference on Robotics and Automation (ICRA), 2017, pp. 3389–3396. doi:10.1109/ICRA.2017.7989385.

[15] A. R. Mahmood, D. Korenkevych, G. Vasan, W. Ma, J. Bergstra, Benchmarking reinforcement learning algorithms on real-world robots (2018). arXiv:1809.07731.

[16] D. Silver, T. Hubert, J. Schrittwieser, I. Antonoglou, M. Lai, A. Guez, M. Lanctot, L. Sifre, D. Kumaran, T. Graepel, et al., A general reinforcement learning algorithm that masters chess, shogi, and go through self-play, Science 362 (6419) (2018) 1140–1144.

[17] O. Vinyals, I. Babuschkin, W. M. Czarnecki, M. Mathieu, A. Dudzik, J. Chung, D. H. Choi, R. Powell, T. Ewalds, P. Georgiev, et al., Grandmaster level in starcraft ii using multi-agent reinforcement learning, Nature 575 (7782) (2019) 350–354.

[18] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, M. Riedmiller, Playing atari with deep reinforcement learning, arXiv preprint arXiv:1312.5602 (2013).

[19] M. T. J. Spaan, Partially Observable Markov Decision Processes, in: M. Wiering, M. van Otterlo (Eds.), Reinforcement Learning: State-of-the-Art, Springer, Berlin, Heidelberg, 2012, pp. 387–414. doi:10.1007/978-3-642-27645-3_12.

[20] J. Schulman, S. Levine, P. Moritz, M. I. Jordan, P. Abbeel, Trust region policy optimization (2017). arXiv:1502.05477.

[21] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, O. Klimov, Proximal policy optimization algorithms, arXiv preprint arXiv:1707.06347 (2017).

[22] A. Raffin, A. Hill, A. Gleave, A. Kanervisto, M. Ernestus, N. Dormann, Stable-baselines3: Reliable reinforcement learning implementations, Journal of Machine Learning Research 22 (268) (2021) 1–8.
URL `http://jmlr.org/papers/v22/20-1364.html`

[23] R. Gato, Z. Menéndez, E. Prieto, R. Argilés, M. Rodríguez, W. Baldoquín, Y. Hernández, D. Pérez, J. Anaya, I. Fuentes, et al., Sterile insect technique: successful suppression of an aedes aegypti field population in cuba, Insects 12 (5) (2021) 469.

[24] M. Strugarek, H. Bossin, Y. Dumont, On the use of the sterile insect release technique to reduce or eliminate mosquito populations, Applied Mathematical Modelling 68 (2019) 443–470.

[25] H. Hughes, N. F. Britton, Modelling the use of wolbachia to control dengue fever transmission, Bulletin of mathematical biology 75 (2013) 796–818.

[26] F. Rivière, Ecologie de aedes (stegomyia) polynesiensis, marks, 1951 et transmission de la filariose de bancroft en polynesie, Paris (France): Université Paris-Sud (1988).

[27] T. Suzuki, F. Sone, Breeding habits of vector mosquitoes of filariasis and dengue fever in western samoa, Medical Entomology and Zoology 29 (4) (1978) 279–286.

[28] E. W. Chambers, L. Hapairai, B. A. Peel, H. Bossin, S. L. Dobson, Male mating competitiveness of a wolbachia-introgressed aedes polynesiensis strain under semi-field conditions, PLoS neglected tropical diseases 5 (8) (2011) e1271.

[29] L. K. Hapairai, J. Marie, S. P. Sinkins, H. C. Bossin, Effect of temperature and larval density on aedes polynesiensis (diptera: Culicidae) laboratory rearing productivity and male characteristics, Acta tropica 132 (2014) S108–S115.

[30] L. K. Hapairai, M. A. C. Sang, S. P. Sinkins, H. C. Bossin, Population studies of the filarial vector aedes polynesiensis (diptera: Culicidae) in

two island settings of french polynesia, Journal of medical entomology 50 (5) (2013) 965–976.

[31] L. Hapairai, Studies on aedes polynesiensis introgression and ecology to facilitate lymphatic filariasis control, Ph.D. thesis, Oxford University, UK (2013).