A MODEL OF DEFORESTATION FOR AGRICULTURAL LAND CLEARANCE IN THE BRAZILIAN RAINFOREST*

YIXUAN HE[§] [¶], RAYMOND CHU[†], JOSEPH MCGUIRE[‡], BOHAN CHEN [†], KAIYAN PENG[†], CHRISTIAN PARKINSON[†], ANDREA L. BERTOZZI [†], AND STANLEY J. OSHER[†]

Abstract. Identifying two types of deforestation, agricultural land clearance for farming and illegal logging, we develop a mathematical model for the first phenomenon, deforestation for agricultural land clearance in the Brazilian rainforest. Working from first principles, we build individual agents known as farmers that then illegally clear forest in the area of concern. To account for illegal farming we develop a time series model in which farmers take optimal paths to the location of their crimes, and choose to farm where they can gain highest expected profit. We discuss some of the time series simulation results used in the farming model, compare two different patrol strategies and describe the numerical implementation of the model. We detail the numerical methods for the optimal path and Hamilton-Jacobi equations involved in the model. Finally, we evaluate the performance of our time series simulation model with F1 score, average capture rate, average growing farm rate, average growing time and average profit.

Key words. time series, illegal farming, optimal path, deforestation

AMS subject classifications. 35F21, 00A69, 97M10, 65M99, 37M10

1. Introduction.

Deforestation, and in particular, illegal logging and land clearance have some of the most damaging effects on the world's forest. Modeling and quantifying deforestation has become a recent area of study for ecologists, political scientists, and applied mathematicians. The efficient and effective deployment of law enforcement to the threatened area is the best deterrent for these crimes [1]. Building a model to predict the interactions between the criminals and police is a difficult problem. A crucial first step in this problem is identifying significant parameters to consider.

Rigorous studies have validated the correlation between certain parameters and deforestation in tropical regions such as Brazil [16]. The three dominant categories of parameters are identified by Pfaff and other authors, as accessibility, population, climate, and demand [3, 12]. Accessibility accounts for distance to roads, rivers, and major highways, elevation, the presence of trees or foliage, as well as recent deforestation events in the area. Population density accounts for distance to cities and markets, as well as the presence of farms and other rural settlements. Climate and demand factors refer to the seasonal effect, such as precipitation and the global demand of soy, beef, lumber and other commodities. With these significant parameters identified, the goal is to inform law enforcement agencies as to the best strategies for combating deforestation.

Further analysis of enforcement strategies has been carried out in the state of Roraima, Brazil, using satellite imaging built to detect deforestation events [7, 17]. The

*Submitted to the editors DATE.

Funding: This research was supported by the Department of Defense and the National Science Foundation (DMS-1737770).

[†]Department of Mathematics, University of California, Los Angeles (raychu92@ucla.edu, bhchenyz@g.ucla.edu, kaiyanpeng@gmail.com, chparkinatucla@gmail.com, bertozzi@math.ucla.edu, sjo@math.ucla.edu).

[‡]Department of Mathematics, Sonoma State University

[§]Department of Mathematics, University of Edinburgh(Y.He-48@sms.ed.ac.uk)

[¶]Department of Mathematics, South China University of Technology

[‖]Codes are available at https://github.com/SherylHYX/Deforestation-Brazil strategy on the part of the federal government of Brazil was to monitor deforestation events in each municipality and dispatch federal patrols to the municipalities with the most events. The satellite data employed (DETER), was updated daily and was effective in detecting any deforestation events larger than 25 hectares in size. This particular macro-strategy proved more effective than typical enforcement strategies. However, the issue remains of how to identify effective microscopic patrol strategies for law enforcement.

1.1. Previous Work. A similar question was asked and answered with a game-theoretic model, Protection Assistant for Wildlife Security (PAWS), where an iterative Stackelberg security game was used to describe the interactions between wildlife poachers and rangers [9]. Previously repeated Stackelberg security games were utilized to model the protection of vital infrastructure in the case of attack. PAWS employed a discrete approach to determine effective patrol strategies in an area of interest. Machine learning techniques were used to determine animal density for each cell in the grid, as well as assign an accessibility score to each cell. Cells that were topographically significant, i.e., cells necessary to enter or exit a region, or cells with high accessibility score, were designated key access points. These were then deemed as nodes and edges were drawn to connect the nodes, so that cells with a higher accessibility score were traversed when possible. Other considerations such as time of patrol and the inclusion of base camp from where to start and end were included for realism, as well as an uncertainty of animal density based on the time of last patrol in that cell. Similar algorithms such as INTERcept and SHARP have been developed and deployed around the world with varying success [10, 11].

In a paper by [1], the problem of deforestation was modeled in a spatially continuous setting. The author of this paper considered a circular area of interest, and modeled how deforestation and patrolling against this, might be described. The patrollers are allocated a budget E, meant to represent the resources available. Budget is used to determine a patrol strategy for the defenders. This is done by giving each radius r a probability of detection $\phi(r)$, then the chance of the attackers getting caught as they are moving from distance d_c to d is given by:

$$\Phi(d) = \int_{d_c}^{d} \phi(r)dr$$

The attackers are assumed to move in radially from their starting radius, d_c, and any ground touched by the attackers is now considered unpristine, while the remaining land is termed pristine as seen in figure 1b. Additionally, the profit that the attackers received by moving a distance d into the protected area is given by:

$$P(d) = (1 - \Phi(d))B(d) - C(d)$$

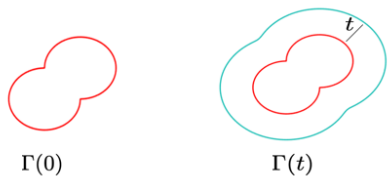
where B is the benefit to the attacker, C is the cost of traveling into depth d and $(1 - \Phi(d))$ represents the probability of not being captured.

A rational attacker will find the d that solves the following optimization problem:

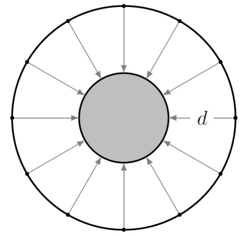
$$\max_{d} (P(d))$$

The defenders then want to minimize the d that solves this problem.

Some issues with this model are the assumption that the area involved is circular, the lack of terrain information, and the uniform behavior of the attackers and



(a) Level sets evolving to a time t



(b) Albers' radial evolution

Fig. 1: Level set method and Albers' curve evolution

defenders. These problems are addressed in a paper from [2], where a similar modeling problem is generalized to any closed, simple curve in $\mathbb{R}^2$. The primary tool employed in this model is the level set method [15]. Starting with a closed, simple curve Γ, a Lipschitz continuous function is defined, such that $\phi_0 : \mathbb{R}^2 \to \mathbb{R}$, where $\phi_0(x, y)$ is positive inside Γ, negative inside of Γ, and zero at the boundary. Note that these requirements on ϕ_0 are satisfied by the signed distance function to the curve Γ, where distance outside of the curve is negative and positive outside. The function $\phi : \mathbb{R}^2 \times [0, \infty) \to \mathbb{R}$ is then defined by the initial value problem:

$$\begin{cases} \phi_t + v(x,y)|\nabla\phi| = 0 \\ \phi(x,y,0) = \phi_0(x,y) \end{cases}$$

where $v(x, y)$ is some non-negative velocity function. Next the authors define the zero level set $\Gamma(t) = \{(x, y) \mid \phi(x, y, t) = 0\}$ and this contour represents the set of points that can be reached from the original contour Γ after traveling time t. In this model cost represents the effort expended by extracting at any point in the protected area, and the velocity is allowed to depend on capture probability and terrain data. The validity of this model hasn't been tested against real-world data, but has been modified and improved by [5].

The purpose of this project is to use the data gathered and categorized by [17] to build a predictive model to describe deforestation events for algricultural land clearance in the state of Roraima. Further, we attempt to determine effective patrol strategies for law enforcement on microscopic scale, and improve upon the model of [2]. First, we will explain some of the analysis and data organization that went into building a foundation for our model. Using this data analysis as justification we then build the theory and implementation of a farming model which describes deforestation for agricultural purposes. We discuss the control-theoretic framework of this optimal path planning problem involved in the model and conclude with the results of the model, and suggest directions for future work.

2. Preparation and Analysis. Our project involves a data set named DETER. The purpose of DETER is to monitor and prevent deforestation in Brazil, specifically Mucajai, a municipality within the state of Roraima, Brazil [17]. In the DETER data set, the space is discretized with polygons and each polygon is assigned a Boolean value to determine whether or not a deforestation event occurred in that polygon in each year and month. Information such as distance to roads and distance to river for

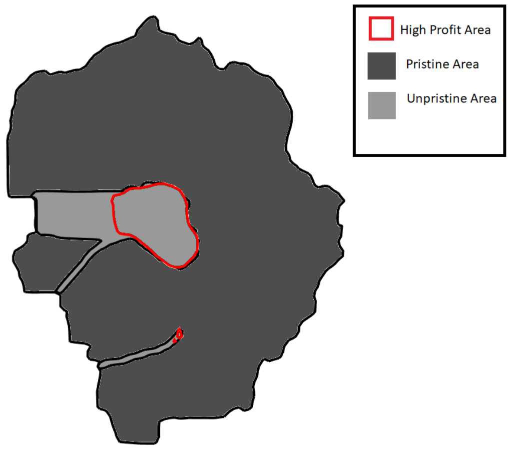


Fig. 2: Yosemite national park, attackers moving in from the boundary

each polygon is also included. This set contained data from 2006 to 2015.

We also utilize another data set, PRODES [14]. This data set is the official data set the Brazilian Government uses to make annual statistics relating to deforestation. The major difference between DETER and PRODES is that the purpose of PRODES is to observe deforestation to make annual statistics. PRODES only includes yearly data from 2001 to 2015. However, the number of data points from PRODES are much more than that of DETER. PRODES also provides us with tree coverage data.

2.1. Data Analysis. Our goal is to use both of these data sets to generate an accurate model of deforestation in Brazil. We first visualize the data to have a better intuition for deforestation in Brazil. In figure 3 we plot the location of deforestation events, a portion of the roads and all the rivers near Mucajai. One of the first observations is that most deforestation events happen near the roads and the events that are farther away tend to be very close to the rivers. In fact, 90 percent of the deforestation events from the PRODES data set lie within 5 km of the roads. This observation makes sense as proximity to the roads or rivers reduces the time it takes to travel to the extraction site and the city, which reduces the cost an extractor associates with going to the extraction site.

In order to gain a probabilistic veiw of the deforestation data, we generate probability density functions (PDFs) from the data set. That is, we are given a set of points $\{\boldsymbol{x}_i\}_{i=1}^N$ where N is very large (for the PRODES data set $N \approx 36,000$) sampled from a PDF ϕ. Now we would like to reconstruct ϕ from these samples. One popular method known as kernal density estimates (KDE) fixes a PDF f with its mass centered at 0, then fixes a parameter $h > 0$ called the bandwidth. Another popular method is maximizing a regularized logarithmic likelihood estimation (MPLE) described in [13]. We follow both methods in our paper.

In figure 4 we plot a heat map of MPLE. The red locations correspond to larger values of the density function. To check the accuracy of MPLE, we generated random events with the MPLE PDF and compared it to real data. We plot the results of this random sampling in figure 5. This figure shows that the predicted event locations

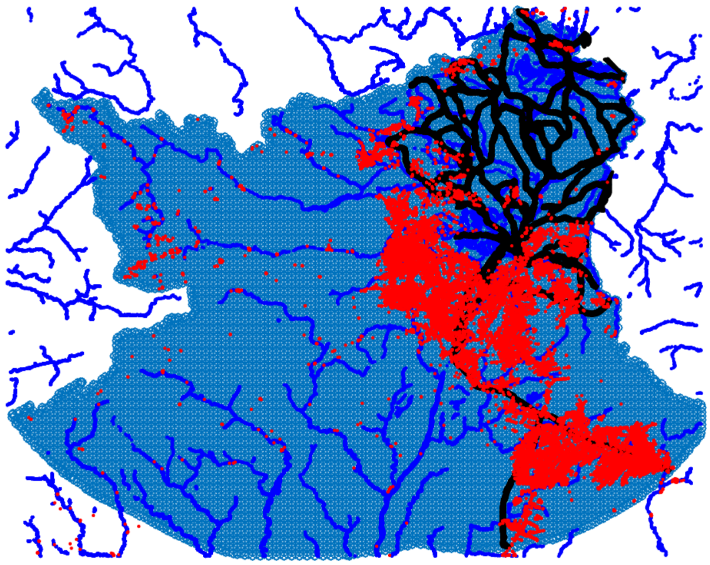


Fig. 3: Dark blue - rivers, black - a subset of roads, light blue - area close to Mucajai, and red dots - deforestation events

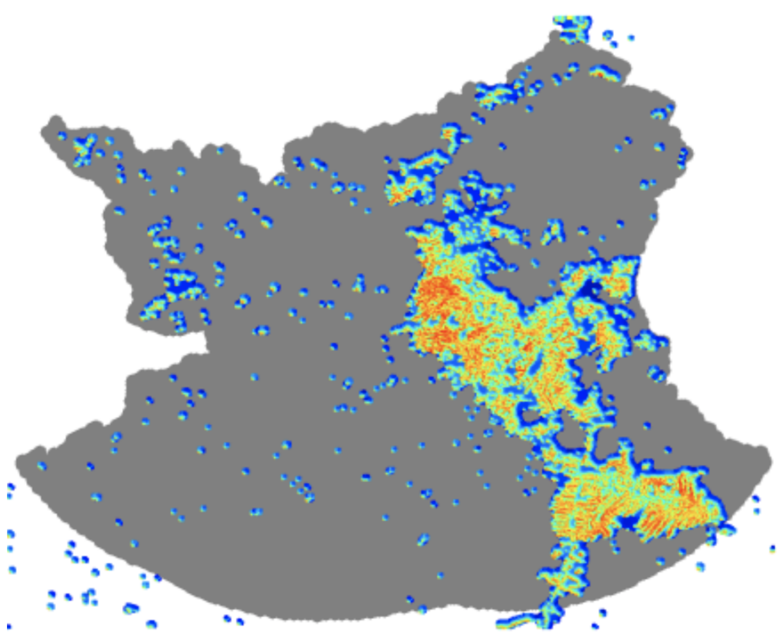
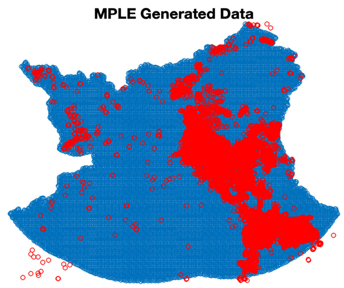
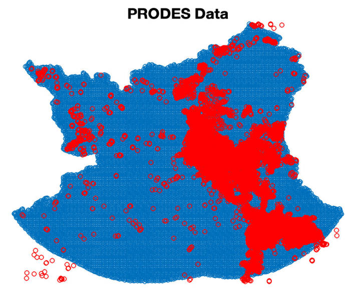


Fig. 4: MPLE PDF on PRODES Data Set

from MPLE are almost identical to the actual data, which leads us to believe MPLE is a good PDF approximation for our purposes.



(a) Deforestation locations predicted by MPLE



(b) Deforestation locations from PRODES

Fig. 5: A comparison of PRODES deforestation data and synthetic data generated from the MPLE density function

We also use the DETER data set to see if there is a seasonal component to deforestation. In figure 6, we plot the average percentage of deforestation that occurs in each month. This bar graph shows that in May to July there is a strong drop in deforestation compared to other months. Those months correspond to the rainy season in Mucajai, confirming our suspicions that the data expresses seasonality.

2.2. Indicator Function of Trees. Since we are trying to model deforestation events, we need to know exactly where the trees are. Therefore, we need to define the indicator function of trees for all grid points in our region of interest. The PRODES data set provides some information about tree coverage locations, so we use the PRODES data with some modifications to obtain our indicator function of trees over the years from 2001 to 2015. The specific algorithm we use to construct the indicator function of trees for each year is as follows:

1. Use the previous year's forest coverage location data for the next year's initial indicator function of trees. For example, we use year 2004's forest coverage location data for the year 2005's initial indicator function of trees. For $n = 2009, \dots, 2015$, we denote the initial indicator function of trees in year n to be $\chi^0_{\text{trees},n}$. We only use such n because PRODES data set only provides forest coverage location data for the years 2008 to 2014 in our region of interest.
2. Get the modified indicator function of trees for the year 2009 by setting the locations where deforestation happened in year 2009 to be 1. That is, the modified indicator function of trees for the year 2009 is

$$\chi^1_{\text{trees},2009} = \min(1, \chi^0_{\text{trees},2009} + \chi_{\text{events},2009}),$$

where $\chi_{\text{events},2009}$ is the indicator matrix of events in the year 2009.
3. For years from 2008 to 2001, we trace backwards. That is, for $n = 2008, 2007, \dots, 2001$, we first let $\chi^0_{\text{trees},n} = \chi^1_{\text{trees},n+1}$. Then we get the modified indicator function of trees for the year n by setting the locations where deforestation

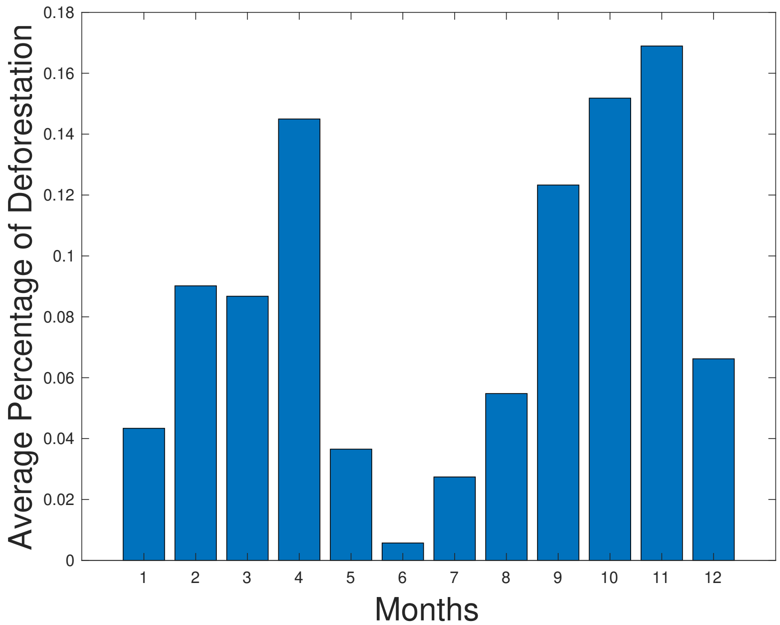


Fig. 6: Seasonality of Deforestation in Mucajai

happened this year to be 1:

$$\chi^1_{\text{trees},n} = \min(1, \chi^0_{\text{trees},n} + \chi_{\text{events},n}).$$

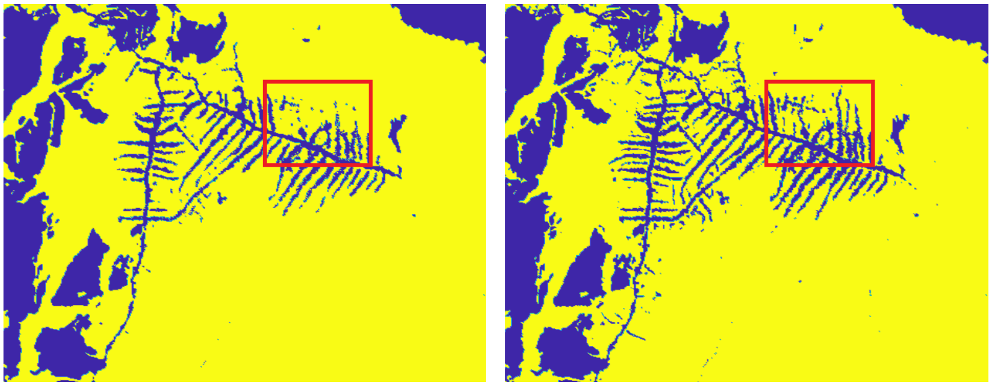
4. For years from 2010 to 2015, we get the modified indicator function of trees for the year n by setting the locations where deforestation happened this year to be 1:

$$\chi^1_{\text{trees},n} = \min(1, \chi^0_{\text{trees},n} + \chi_{\text{events},n}).$$

5. Obtain the final indicator function for trees in the year 2015 as $\chi_{\text{trees},2015} = \chi^1_{\text{trees},2015}$.
6. Make sure that the indicator function will not increase over the years, by taking the maximum of two consecutive years to modify indicator functions of trees backwards. That is, for $n = 2014, 2013, \ldots, 2001$, we trace back to get the final indicator function for trees to be $\chi_{\text{trees},n} = \max(\chi^1_{\text{trees},n}, \chi_{\text{trees},n+1})$.

Figure 7a and 7b show the indicator function of trees for the year 2001 and 2015. The yellow color denotes value 1 and blue denotes value 0 for the function. There is significant difference inside the red box as trees have been added as we trace back to 2001.

3. Farming Model Overview. [17] concluded that there are two main types of deforestation that occur in Roraima. The first type is agricultural deforestation, wherein farmers illegally expand their farms to increase their crop output. In this section, we describe how we model this phenomenon.



(a) Indicator function of trees for 2001 (b) Indicator function of trees for 2015

Fig. 7: Indicator function at the beginning and end of the time period

3.1. Expected Profit. We first assume that the farmers have perfect information regarding patrol. Also, farmers choose locations for farms probabilistically based on expected profit. To define the expected profit, we first have to define the expected cost and expected benefit.

Because most events happen near cities and along the roads the cost has dependence on the distance to cities and distance to roads. In addition, we observe that the deforestation events tend to cluster around one another. Therefore, we define the cost at each point per month in the year n as

$$C(x,y) = \frac{c_1 \cdot d_r(x,y) + c_2 \cdot d_c(x,y) + c_4}{1 + c_3 \cdot A^n(x,y)}, \tag{3.1}$$

where $d_r(x,y)$ denotes the distance to roads at location (x,y), and $d_c(x,y)$ denotes the distance to cities at location (x,y). The distances are calculated from the center of a farm to the nearest road or nearest city. The measurement of fixed costs associated to running a farm is denoted as c_4. Note that $c_1, c_2, c_3, c_4 \geq 0$ can be adjusted to fit the deforestation data. The final calculation of expected profit should be the sum of all costs in related months. This is because there exist ongoing costs when farmers are travelling back and forth. The accessibility at location (x,y) in the year n is denoted as $A^n(x,y)$. For Accessibility, we first define $\{(x_i^{(n)}, y_i^{(n)})\}_{i=1}^{M(n)}$ as the deforestation events which occurred in the year n. Then we fixed a parameter $h > 0$ called the bandwidth and define the accessibility contributed by the year n as

$$A_h^n(x,y) := \sum_{i=1}^{M(n)} \frac{1}{2\pi h} \exp\left(-\frac{\left(x - x_i^{(n)}\right)^2 + \left(y - y_i^{(n)}\right)^2}{h^2} \right) \tag{3.2}$$

Finally we obtain the accessibility for location (x,y) in the year n as

$$A^n(x,y) := \sum_{k=1}^{3} \frac{1}{2^k} A_h^{n-k}(x,y) \tag{3.3}$$

So accessibility is a weighted sum of Gaussian with expected values at the deforestation locations and standard deviation h where the influence of each Gaussian decays over time by a factor of $\frac{1}{2}$ for each year that has passed and we consider only the previous 3 years events. This assumes that farmers develop infrastructure as they plant farms, reducing costs for nearby farms.

Now we move on to define expected benefit. First we define the benefit per square meter per month by:

$$B(x,y) := \xi \chi_{\text{trees}}(x,y) \frac{1}{1 + ||\nabla E(x,y)||^{\gamma}}$$

where $\chi_{\text{trees}}(x,y)$ is the indicator function for the presence of a tree at location (x,y), $E(x,y)$ is the elevation, and ξ represents the conversion of the value of a tree to currency. As a farm grows older, ξ should decline due to lack of nutrition in the soil. We scale this factor by $\frac{1}{1+||\nabla E(x,y)||^{\gamma}}$ because if the area has sharp changes in elevation, then it will be harder to farm, while if it is almost flat then $||\nabla E(x,y)|| \approx 0 \Rightarrow B(x,y) \approx \xi \chi_{\text{trees}}(x,y)$.

Benefit and cost are not the only factors that farmers consider when choosing a farm location. They also account for the probability of getting caught illegally making and expanding their farms in their expected benefit calculation. In any given month, we model the probability of not being captured by

$$1 - \psi(x,y) \cdot S$$

where $\psi(x,y)$ is the capture probability at location (x,y) (a multiple of the patrol density, dependent on patrol budget and patrol strategy, the multiplier measures the patrol budget) and S is the area of the farm.

Assume a farm starts with area S_0, and expands for i months at a rate of dS, then the size of the farm at the end of the i^{th} month will be $S_i = S_0 + i \cdot dS$. As the average area of deforestation in PRODES is about 10^5 square meters per year, we choose $S_0 = 0, dS = \frac{10^5}{12}$. Thus, the probability of not being captured by the end of the i^{th} month at location (x,y), if we assume that the capture activities in different months are independent, is

$$\beta_i(x,y) = \prod_{j=1}^{i} (1 - \psi(x,y) \cdot S_j).$$

By the law of total expectation, the expected benefit a farmer will receive in month i is

$$\mathbb{E}(\text{Benefit}_i(x,y)) = B_i(x,y) \cdot S_i \cdot \beta_i(x,y) - \alpha \cdot B_i(x,y) \cdot S_i \cdot (1 - \beta_i(x,y))$$

Here $\alpha > 0$ is a parameter that measures the strength of punishment. For example, $\alpha = 2.5$ means that if a farmer is caught, then he will be fined with the amount of 2.5 times the benefit he want to gain from that farm in month i. And the expected benefit which a farmer will receive by the end of month i is

$$\sum_{j=1}^{i} \mathbb{E}(\text{Benefit}_j(x,y))$$

As a farm cannot last forever partly due to lack of nutrition in the land, we assumed that a farm can only operate for at most $K_{\max}$ months. For our experiments, we choose $K_{\max} = 36$. Therefore, a farmer will chose a month between $1, .., K_{\max}$ such that he can maximize his expected profit.

Thus, the expected profit function for a farmer is

$$\mathbb{E}(P(x,y)) := \left(\max_{i=1,\ldots,K_{max}} \sum_{j=1}^{i} \left(\mathbb{E}(\text{Benefit}_j(x,y)) - C_j(x,y) \right) \right)_+$$

where $(f)_+ := \max(f, 0)$. So if farmers choose a spot (x, y), then they will stay in that spot until they gain the expected highest profit or until get caught. For expected profit calculation, we assume the both ξ and C decrease linearly from the initial value to 0.

Then we normalize $\mathbb{E}(P(x,y))$ to make it into a probability density function. Farmers will choose where to plant their farms randomly with this density.

The algorithm for the expected profit can be seen in 3.1.

Algorithm 3.1 Expected Profit

Require: $C, B, \psi, S_0, dS, K_{\max}, \alpha$
1: $S \leftarrow S_0 + dS$, $\beta \leftarrow 1 - \psi \cdot S$
2: $P_1 \leftarrow B \cdot S \cdot ((1 + \alpha) \cdot \beta - \alpha) - C$
3: **for** $i = 2$ to $K_{\max}$ **do**
4: $\quad S \leftarrow S + dS$, $\beta \leftarrow \beta \cdot (1 - \psi \cdot S)$
5: $\quad P_i \leftarrow P_{i-1} + (1 - \frac{i}{K_{\max}})(B \cdot S \cdot ((1 + \alpha) \cdot \beta - \alpha) - C)$
6: **end for**
7: $\mathbb{E}P \leftarrow \max_i P_i$ # take the max for every entry
8: $\mathbb{E}P \leftarrow \frac{\mathbb{E}P}{\sum_m \sum_n \mathbb{E}P(m,n)}$
9: **return** $\mathbb{E}P$

3.2. Time Series Simulation. Now we consider a simulation wherein multiple farmers plant their own farms. Then the benefit near their selected farm should decrease as the next farmers cannot farm there easily. Their deforestation events will increase the accessibility of their farms. Besides, by digging into the forest for agricultural land clearance, new roads will be constructed. Patrollers will also update their patrol strategies. Hence, the expected profit should be updated based on updated benefit, accessibility, patrol strategies, and road map. Accordingly, we want to design a time series model with the goal of accurately predicting deforestation events caused by farmers.

We assume that for a fixed year n that we have data about the deforestation events for year $n-1, n-2$, and $n-3$ and a patrol density ψ. We first obtain $\mathbb{E}P$ with expected profit algorithm. Now we randomly Sample N points from $\mathbb{E}P$, where N is based on average proportion of events in month i. These nodes represent the locations of where the first farmers of this year decided to plant a farm. In particular, we assume that the farmer creates a circular farm centered at those selected nodes. We assume that they start a farm at every point chosen with area S_0, profit $P = 0$ and age $t = 1$.. For growing farms, we expect the benefit for them to decrease linearly, so we update benefit (for farmers in growing farms) for every growing farm

$$\bar{B} \leftarrow (1 - \frac{t}{K_{\max}})B.$$

We then calculate marginal expected profit for each farm by $\bar{B} \cdot (S + dS) - C$. If no more marginal expected profit can be gained, farmers will stop growing farms.

For potential farmers, they are less likely to farm at locations that are already occupied. Therefore, benefit for them should decrease in farmed regions. We update Benefit (for potential farmers) for every point in growing farms by

$$B(x, y) \leftarrow (1 - \lambda)B(x, y), \lambda \in (0, 1).$$

In practice, we chose $\lambda = \frac{1}{2}$ because in our data set, multiple events can happen on the same node due to the coarseness of our grid. In addition, as a farmer constructs a farm, we expect this farmer to create infrastructure which makes the surrounding region more accessible for other farmers to construct new farms. Therefore, we update accessibility by the following formula

$$A(x, y) \leftarrow A(x, y) + (\frac{1}{2})^{t_i} \frac{1}{2\pi h} \exp\left(-\frac{(x - x_i)^2 + (y - y_i)^2}{h^2}\right),$$

where t_i is the age of the farm centered at (x_i, y_i). Then we normalized A so that it retains its initial mass.

We also assume that each farmer wants to expand his farm's area by dS by the end of each month, but as this activity is illegal, there is a chance he can get caught while expanding the farm. To simulate this, first we update farm area for every growing farm by $S \leftarrow S + dS$. Then we do a Bernoulli trial with probability $p = \psi(x, y) \cdot S$ where $\psi(x, y)$ is the patrol effect at the farm location (x, y). If the trial is successful, the farmer is caught, so he stops growing his farm and he also receives a fine of $\alpha B(x, y)S$. If the trial is a failure, then the farmer is not caught and will continue growing his farm if he can gain positive marginal expected profit.

Now we update farm age for every growing farm by $t \leftarrow t + 1$. We assume that farmers choose optimal paths to construct farms from roads. Also, they will build new roads on their way to their farms. Therefore, we use optimal path planning as described in Section 4 to obtain all optimal paths between new farms and roads. Then we update the road map and get new distance to road $\bar{d_r}$. For the next month, the distance in the cost calculation will be a combination of the old distance and the new distance, as those man-made new roads may be coarsely constructed. We also assume that patrollers know the expected profit farmers have at this month, so they update ψ to be a multiple of $\mathbb{E}P$. We repeat this procedure for all months we want to simulate.

Details of the time series simulation algorithm are described in 3.2 and 3.3.

Algorithm 3.2 Farm Simulation

Require: $C, B, \psi, S_0, dS, K_{\max}, \alpha, d_r, d_c, N, A$

$\bar{d_r} \leftarrow d_r$ # $\bar{d_r}$ is the new distance to roads
for $i = 1$ to 12 **do**
  $C \leftarrow (c_1 \cdot \frac{d_r + \mu \bar{d_r}}{1+\mu} + c_2 \cdot d_c + c_4)/(1 + c_3 \cdot A)$
  $\mathbb{E}P \leftarrow$ Expected Profit$(C, B, \psi, S_0, dS, K_{\max}, \alpha)$
  $F \leftarrow$ new farms generated from $\mathbb{E}P$
  $f.S \leftarrow S_0, f.t \leftarrow 0, f.g \leftarrow 1, f.P \leftarrow 0$ # g - growing indicator, P - profit $\forall f \in F$
  $F, A, B \leftarrow$ Farm Update$(C, B, \psi, S_0, dS, K_{\max}, \alpha, d_r, d_c, F, A)$
  $\psi \leftarrow w\mathbb{E}P$
  $\bar{d_r} \leftarrow$ Optimal Roads$(\bar{d_r}, F)$
end for
return $\bar{d_r}, F$

Algorithm 3.3 Farm Update

Require: $C, B, \psi, S_0, dS, K_{\max}, \alpha, d_r, d_c, F, A$

for all $f \in F$ **do**
  $\bar{B} \leftarrow (1 - \frac{f.t}{K_{\max}})B, P_m = \bar{B}(f.S + dS) - C$ # P_m - marginal profit
  if $P_m \leq 0$ **or** $f.t \geq K_{\max}$ **then**
    $f.g \leftarrow 0$
  end if
  if $f.g == 1$ **then**
    $f.S \leftarrow f.S + dS, f.P \leftarrow f.P + P_m$
    $B(x, y) \leftarrow (1 - \lambda)B(x, y), \forall (x, y) \in f$
    $A(x, y) \leftarrow A(x, y) + (\frac{1}{2})^{f.t} \frac{1}{2\pi h} \exp\left(-\frac{(x - f.x)^2 + (y - f.y)^2}{h^2}\right), \forall (x, y)$
    # $(f.x, f.y)$ is the center of f
    $f.g \leftarrow 0$ & $f.P \leftarrow f.P - \alpha \cdot \bar{B}$ with prob. $\psi \cdot f.S$
    $f.t \leftarrow f.t + 1$ if $f.g == 1$
  end if
end for
return F, A, B

w in Algorithm 3.2 measures the patrol budget. We may also try different patrol strategies. For example, we can try to update patrol only when farmers are captured, similar to how we update Accessibility, by Algorithm 3.4 and 3.5.

Algorithm 3.4 Farm Simulation2

Require: $C, B, \psi, S_0, dS, K_{\max}, \alpha, d_r, d_c, N, A$

$\bar{d_r} \leftarrow d_r$ # $\bar{d_r}$ is the new distance to roads
for $i = 1$ to 12 **do**
  $C \leftarrow (c_1 \cdot \frac{d_r + \mu \bar{d_r}}{1+\mu} + c_2 \cdot d_c + c_4)/(1 + c_3 \cdot A)$
  $\mathbb{E}P \leftarrow$ Expected Profit$(C, B, \psi, S_0, dS, K_{\max}, \alpha)$
  $F \leftarrow$ new farms generated from $\mathbb{E}P$
  $f.S \leftarrow S_0, f.t \leftarrow 0, f.g \leftarrow 1, f.P \leftarrow 0$ # g - growing indicator, P - profit $\forall f \in F$
  $F, A, B, \psi \leftarrow$ Farm Update2$(C, B, \psi, S_0, dS, K_{\max}, \alpha, d_r, d_c, F, A)$
  $\bar{d_r} \leftarrow$ Optimal Roads$(\bar{d_r}, F)$
end for
return $\bar{d_r}, F$

Algorithm 3.5 Farm Update2

Require: $C, B, \psi, S_0, dS, K_{\max}, \alpha, d_r, d_c, F, A$

$\bar{\psi} \leftarrow 0$
for all $f \in F$ **do**
  $\bar{B} \leftarrow (1 - \frac{f.t}{K_{\max}})B, P_m = \bar{B}(f.S + dS) - C$ # P_m - marginal profit
  if $P_m \leq 0$ **or** $f.t \geq K_{\max}$ **then**
    $f.g \leftarrow 0$
  end if
  if $f.g == 1$ **then**
    $f.S \leftarrow f.S + dS, f.P \leftarrow f.P + P_m$
    $B(x,y) \leftarrow (1-\lambda)B(x,y), \forall (x,y) \in f$
    $A(x,y) \leftarrow A(x,y) + (\frac{1}{2})^{f.t} \frac{1}{2\pi h} \exp\left(-\frac{(x-f.x)^2 + (y-f.y)^2}{h^2}\right), \forall (x,y)$
    # $(f.x, f.y)$ is the center of f
    $f.g \leftarrow 0$ & $f.P \leftarrow f.P - \alpha \cdot \bar{B}$ & $\bar{\psi} \leftarrow \bar{\psi} + G(f)$ with prob. $\psi \cdot f.S$
    # $G(f) = (\frac{1}{2})^{f.t} \frac{1}{2\pi h} \exp\left(-\frac{(x-f.x)^2 + (y-f.y)^2}{h^2}\right), \forall (x,y)$
    if $\sum_x \sum_n \bar{\psi}(m,n) > 0$ **then**
      $\psi \leftarrow (0.95\psi + \frac{\bar{\psi}}{\sum_x \sum_n \bar{\psi}(m,n)})/1.95$
    end if
    $f.t \leftarrow f.t + 1$ if $f.g == 1$
  end if
end for
return F, A, B, ψ

4. Path Planning with Optimal Control Theory. In this section, we will transform the optimal control problem in our model into a Hamilton-Jacobi equation, then describe a method to find the optimal path based on the solution to the Hamilton-Jacobi equation.

When we mention the solution to a Hamilton-Jacobi(-Bellman) equation or eikonal equation below, it always denotes the viscosity solution in the sense of [6].

4.1. Static Hamilton-Jacobi-Bellman Equation. A general formulation of the optimal control we need to solve in our model can be written as

$$\begin{aligned} \min_{\alpha\in\mathcal{A}} \quad & \int_0^\tau r(x(s))ds \\ s.t. \quad & \dot{x} = v(x)\alpha(x) \\ & x(0) = x_0 \\ & \tau = \inf\{t : x(t) = x_{end}\} \end{aligned} \tag{4.1}$$

Here $\alpha(\cdot)$ is the control, taken from set of valid control functions $\mathcal{A}$. The function r is always positive and represents some running cost incurred along the trajectory determined by α. Now we transform this problem into a static eikonal equation. From [4], there are two theorems regarding this transformation.

Theorem 4.1.1: Dynamic Programming Principle (DPP). Suppose $y_x(t, \alpha)$ is the trajectory following the differential equation in (4.1), starting from x and using the control function α. Define the time of reaching the end point as

$$t_x(\alpha) = \begin{cases} \inf\{t : y_x(t,\alpha) = x_{end}\}, \text{ if } \{t : y_x(t,\alpha) = x_{end}\} \neq \emptyset, \\ +\infty, \text{ otherwise.} \end{cases}$$

Define the minimal cost by

$$u(x) := \inf_{\alpha\in\mathcal{A}} \left\{ \int_0^{t_x(\alpha)} r(y_x(s,\alpha))ds \right\}. \tag{4.2}$$

Then $u(x)$ satisfies the Dynamic Programming Principle (DPP). That is, for $t \leq t_x(\alpha)$,

$$u(x) = \inf_{\alpha\in\mathcal{A}} \left\{ \int_0^t rds + u(y_x(t,\alpha)) \right\}, \ \forall t \geq 0, \tag{4.3}$$

Theorem 4.1.2: Static Hamilton-Jacobi-Bellman equation. Suppose $u(x)$ is defined as in **Theorem 4.1.1**, then $u(x)$ is the solution to the static Hamilton-Jacobi-Bellman equation

$$\begin{aligned} &\sup_{\alpha\in\mathcal{A}} \{-v(x)\alpha(x) \cdot \nabla u(x)\} = r(x), \\ &u(x_{\text{end}}) = 0, u(x) \geq 0, \ \forall x \in \mathbb{R}^n \end{aligned} \tag{4.4}$$

Formally, to go from (4.3) to (4.4), we move $u(x)$ to the right hand side of (4.3), divide by t and let $t \to 0$. Using the chain rule and the ODE solved by $y_x(t, \alpha)$, we arrive at (4.4).

4.2. The Eikonal and Hamilton-Jacobi Equations. In this part we will show the connection between eikonal equation and Hamilton-Jacobi equation. For nonnegative function $v(x)$, we consider a Hamilton-Jacobi equation

$$\begin{aligned} &u_t + v(x)|\nabla u| = 0, \ (x,t) \in \mathbb{R}^n \times (0, +\infty), \\ &u(x, 0) = g(x), \ x \in \mathbb{R}^n. \end{aligned} \tag{4.5}$$

From [8], we have these two theorems:

Theorem 4.2.1 For any $y \in \mathbb{R}^n$, the equation

$$\begin{aligned} &v(x)|\nabla d(x;y)| = 1,\ x \in \mathbb{R}^n \setminus \{y\}, \\ &d(y;y) = 0,\ d(x;y) \geq 0,\ \forall x \in \mathbb{R}^n, \end{aligned} \tag{4.6}$$

has a unique solution $d(\cdot\ ;\ y)$.

Theorem 4.2.2 Assume that $g(x) \geq -C(1+|x|)$ for some $C > 0$. Define the function

$$\Phi(x) = \begin{cases} 0 & \text{if } x = 1 \\ +\infty & \text{otherwise} \end{cases}$$

Then the function

$$u(x,t) = \inf_{y \in \mathbb{R}^n} \left\{ g(y) + t\Phi\left(\frac{d(x;y)}{t}\right) \right\}$$

is the unique solution of (4.5) which is bounded below by a function of linear growth and such that

$$\liminf_{(y,t)\to(x,0^+)} u(y,t) = g(x).$$

Consider the initial function $g(x) = |x - z|$, the Euclidean distance to a point z. By **Theorem 4.2.2**, we have that

$$\begin{aligned} u(x,t) &= \inf_{y \in \mathbb{R}^n} \left\{ |y - z| + t\Phi\left(\frac{d(x;y)}{t}\right) \right\} \\ &= \inf_{\substack{y \in \mathbb{R}^n \\ d(x;y)\ =\ t}} \quad |y - z| \end{aligned} \tag{4.7}$$

is a solution to the equation (4.5), where $d(x; y)$ is the solution to the eikonal equation (4.6).

According to (4.7), $u(x,t) = 0$ if and only if $d(x; z) = t$. Thus we deduce the connection between the solution $w(x) = d(x; z)$ to the equation (4.6) where $y = z$ and the solution $u(x,t)$ to the equation (4.5):

$$w(x) = t \text{ if and only if } u(x,t) = 0.$$

In this way, any eikonal equation can be transformed into a time-dependent Hamilton-Jacobi equation, and vice versa.

Note that the equation (4.5) is a level set equation as considered by [15]. If $g(x)$ is the signed distance function to a contour $\Gamma(0)$, we can evolve $\Gamma(0)$ by level set motion when we solve the PDE. If we define $\Gamma(t) = \{x : u(x,t) = 0\}$, then $\Gamma(t)$ represents the set of points which can be reached from the original contour with traveling time t in the velocity field $v(x)$. This is displayed in figure 8. The interpretation of the level sets $\Gamma(t)$ as the reachable sets after traveling for time t from an initial state gives a connection between the level set equation and the optimal control problem for path planning. In view of the equivalence of (4.5) and (4.6), we can obtain the level sets from the solution to either equation; that is, $\Gamma(t) = \{x : u(x,t) = 0\} = \{x : w(x) = t\}$.

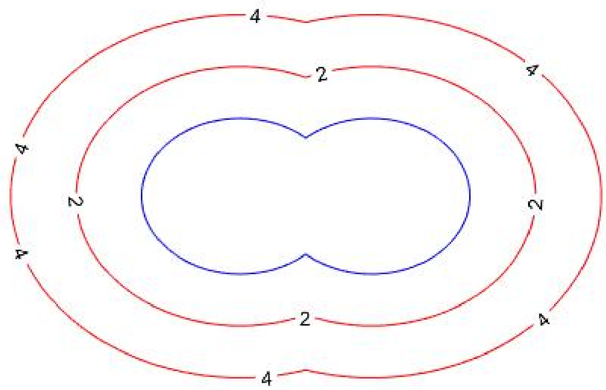


Fig. 8: Level set evolution with velocity $v(x) = 1$, from an initial contour $\Gamma(0)$ (blue). The contours $\Gamma(2)$ and $\Gamma(4)$ (red) represent the contours which could be reached after traveling for $t = 2$ or $t = 4$ respectively. These are zero level contours of the solution $u(x,t)$ to equation (4.5).

4.3. Finding the Optimal Path. In our model, the optimal control problems have the form (4.1) where $\mathcal{A} = \{\alpha : |\alpha(\cdot)| = 1\}$. If we define the optimal cost $u(x)$ as (4.2), then $u(x)$ satisfies

$$\sup_{\alpha \in \mathcal{A}} \{-v(x)\alpha(x) \cdot \nabla u(x)\} = r(x).$$

Since $v(x) \geq 0,\ \forall x \in \mathbb{R}^n$, we can solve for the optimal control $\alpha(x)$ explicitly using the Cauchy-Schwarz inequality:

$$\alpha(x) = -\frac{\nabla u(x)}{|\nabla u(x)|}.$$

Thus $u(x)$ solves the eikonal equation

$$\begin{aligned} &v(x)|\nabla u(x)| = r(x), \\ &u(x_{\text{end}}) = 0, u(x) \geq 0,\ \forall x \in \mathbb{R}^n. \end{aligned}$$

Since the running cost function $r(\cdot)$ is always positive, we can transform this into the corresponding Hamilton-Jacobi equation

$$\begin{aligned} &\phi_t + \frac{v(x)}{r(x)}|\nabla \phi| = 0,\ (x,t) \in \mathbb{R}^n \times (0, +\infty), \\ &\phi(x, 0) = |x - x_{\text{end}}|,\ x \in \mathbb{R}^n. \end{aligned} \tag{4.8}$$

In order to find the best trajectory for the optimal control problem (4.1), we can evolve level sets $\Gamma(t)$ outward from x_{end} according to (4.8). This level sets represent the reachable sets after time t with the velocity field $v(x)/r(x)$. This time t is no longer traveling time, but rather cost associated with the optimal control problem. Hence the level sets $\Gamma(t)$ represent contours of equal traveling cost. To find the optimal path from x_0 to a desired end point x_{end}. We evolve level sets outward from x_{end} until they reach x_0. That is, we evolve (4.8) until the time t^* such that $\phi(x_0, t^*) = 0$. This t^* represents the optimal cost associated with traveling from x_0 to x_{end}.

As we solve (4.8), at each (x, t) we can resolve the optimal control value

$$\alpha(x,t) = -\frac{\nabla\phi(x,t)}{|\nabla\phi(x,t)|}.$$

We can use this to resolve the optimal trajectory, by integrating the ODE in (4.1) backwards from time $t = t^*$ to $t = 0$, so that the optimal trajectory is given by

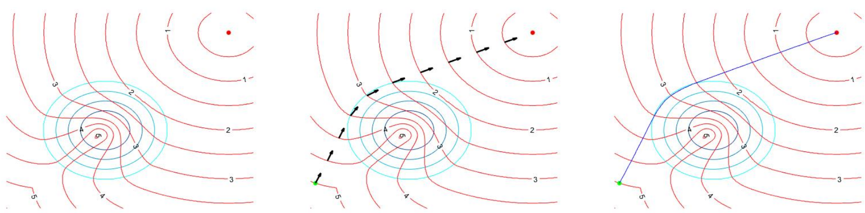
$$\begin{aligned} \dot{x} &= -v(x)\frac{\nabla\phi(x,t)}{|\nabla\phi(x,t)|}, \\ x(t^*) &= x_0. \end{aligned} \tag{4.9}$$

We should solve the ODE backwards because the function $\phi(x, t)$ may not be differentiable at $(x_{\text{end}}, 0)$, and thus we cannot determine the value of the optimal control at that point. Note, the optimal trajectory will always travel parallel to $\nabla\phi$ which is normal to the level sets $\Gamma(t)$.

In summary, we find the optimal trajectory in two steps:

1. **Evolve the level sets**. Solve the equation (4.8) until the time t^* such that $\phi(x_0, t^*) = 0$. Here t^* is the optimal cost in the problem (4.1).
2. **Track back**. Solve the dynamic ODE (4.9) backwards in time, starting from $x(t^*) = x_0$. This ODE gives the optimal path we need.

The process of finding an optimal path is displayed in figure 9.



(a) Level set contours (b) Normal directions (c) Optimal path

Fig. 9: To find the optimal path from the start point (green dot) to the end point (red dot), we first evolve the level sets from the end point until they reach the start point, and then let the point travel along the normal direction of the level sets to find the optimal path.

5. Experimental Setup. As mentioned in section 2, we observe that most deforestation events for agricultural land clearance occur near the roads or cities. We posit that any deforestation events occurring with 10 kilometers of a major road is a farming event. Further, in the southeast portion of Roraima, there is a region with many roads and small cities. We focus on modelling illegal farming in this region, using a 500×500 grid with grid size 300m×300m. The region of interest is pictured in figure 10.

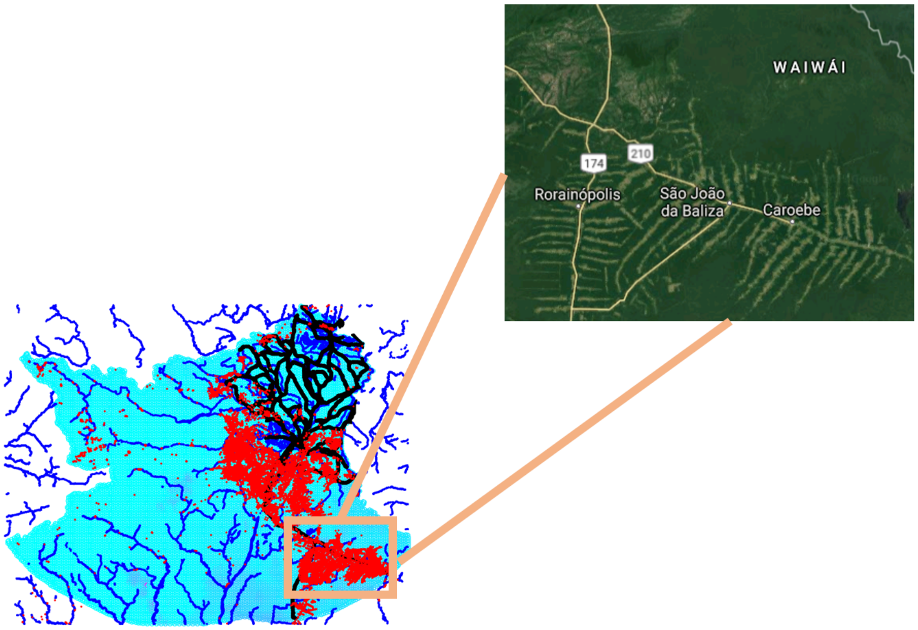


Fig. 10: The southeast region of Roraima where we test our farming model.

6. Evaluation. To evaluate the performance of our simulation model, we compare the results with real events in the PRODES data set and get the F1 score. We also consider how realistic our model is, so we take into account average capture rate, average growing farm rate till the end of a year, and average farm growing time. For patrol strategy evaluation, we use average profit and average capture rate as judging criteria.

6.1. Metric 1. F1 Score. Due to sparsity of events, we give some tolerance for the prediction. Let T_n, R_n denote the predicted and real event matrix for year n respectively. Each entry of the event matrices count the number of events happen in year n that are closest to the location that the entry represents.

Then we classify a real event at location (x, y) as being accurately predicted if the sum of entries in T_n in a neighborhood of (x, y) is no less than $R_n(x, y)$. And we classify a point (x, y) to be false positive only if the sum of entries in R_n in a neighborhood of (x, y) is less than $T_n(x, y)$.

The scores involved in the calculation are recall, precision, and F1 score. That is,

$$\text{Recall} = \frac{|TP|}{\#\text{ of actual events}}, \text{Precision} = \frac{|TP|}{\#\text{ of predicted events}},$$

$$F1 = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}.$$

The higher the F1 score, the more accurate the predictions of the model. The algorithm for F1 score is detailed in 6.1.

Algorithm 6.1 F1 Score

Require: T_n, R_n, N_n

$D \leftarrow \text{pad}(T_n, k)$ # $\text{pad}(R_n, k)$ - R_n with k zero paddings outside
$tmp \leftarrow \text{zeros}(l_x, l_y)$ # T_n has size $l_x \times l_y$
for $i = -k$ to $k, j = -k$ to k **do**
  $tmp \leftarrow tmp + D(1 + k + i : l_x + k + i, 1 + k + j : l_y + k + j)$
end for
$FP \leftarrow \sum_i \sum_j (T_n(i,j) > tmp(i,j))$
$D \leftarrow \text{pad}(R_n, k), tmp \leftarrow \text{zeros}(l_x, l_y)$
for $i = -k$ to $k, j = -k$ to k **do**
  $tmp \leftarrow tmp + D(1 + k + i : l_x + k + i, 1 + k + j : l_y + k + j)$
end for
$FN \leftarrow \sum_i \sum_j (R_n(i,j) > tmp(i,j))$
$TP \leftarrow N_n - FN$ # N_n - number of events
$Pre \leftarrow TP/(TP + FP), Re \leftarrow TP/N_n$ # precision and recall
$F1 \leftarrow 2 \cdot Pre \cdot Re/(Pre + Re)$
return $F1$

6.2. Other Evaluation Metrics. In addition to F1 score, we also consider how realistic our model is. Therefore, we have:

Metric 2. average capture rate:

$$\frac{1}{11} \sum_{n=2005}^{2015} \frac{\#\text{captured}}{N_n}$$

This measures how often illegal farmers get caught. The result is dependent on patrol budget and patrol strategy. A realistic value of average capture rate should be between 1% and 99%. This metric is useful for comparing the patrol strategies given the patrol budget. For a given patrol budget, the lower the average profit, the better the patrol strategy is.

Metric 3. average growing farm rate:

$$\frac{1}{11} \sum_{n=2005}^{2015} \frac{\sum_f f.g}{N_n}$$

This measures how likely illegal farmers continue their expansion of farms towards the next year. The result is dependent on the capture rate and marginal expected profit. A realistic value of average growing farm rate should be between 1% and 99%.

Metric 4. average farm growing time:

$$\frac{1}{11} \sum_{n=2005}^{2015} \frac{\sum_f f.t}{N_n}$$

This measures how long farms are expected to grow within a year. The result is dependent on the capture rate and marginal expected profit. A realistic value of average farm growing time should be between 2 months and 9 months.

Metric 5. average profit:

$$\frac{1}{11} \sum_{n=2005}^{2015} \frac{\sum_f f.P}{N_n}$$

This measures how much profit farms are expected to obtain within a year. The result is dependent on the patrol budget and patrol strategy. The number itself is not useful, but the relative comparison is meaningful. This metric is useful for choosing the patrol strategy given the patrol budget. For a given patrol budget, the lower the average profit, the better the patrol strategy is.

6.3. Parameters & Evaluation Results. To choose the better patrol strategy, we compare the average profit and average capture rate between our patrol strategies. Expected Profit Patrol means the patrol strategy in Algorithm 3.2 and 3.3. Capture Patrol means the other. The comparison result is shown in Table 1.

	average profit	average capture rate
Expected Profit Patrol	0.008	0.238
Capture Patrol	0.012	0.018

Table 1: Comparison between patrol strategies. Expected Profit Patrol (we currently use) has better scores.

Therefore, the better patrol strategy is the Expected Profit Patrol. We then try to find better parametrization for this patrol strategy.

Our optimal set of parameters using the expected profit patrol is:

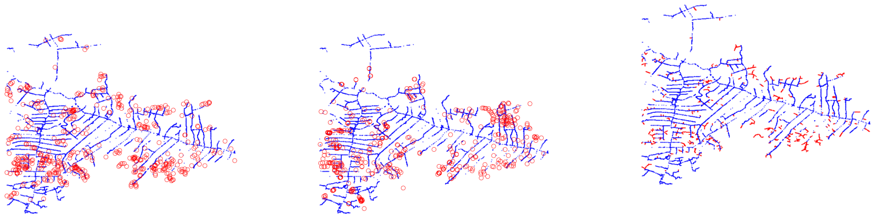
- $c_1 \sim 10^{-1}, c_2 \sim 10^{-2}, c_3 \sim 10^{12}, c_4 \sim 1$
- $\gamma \sim 1, \xi \sim 10^{-7}, h \sim 10^{-3}$
- $\alpha = 2.5, \lambda = 0.5, \mu = 0.2$

This yields an average recall score of 0.64, precision score of 0.57, and F1 score of 0.61. This score is not perfect because of the sparsity of real deforestation events, but we can see clusters along the roads in the result figures.

We have average capture rate of 24%, which is realistic. The average growing farm rate till the end of a year is 49%, which means that about half of the farms starting in a year may continue growing for the next year. The average farm growing time is 5 months within a year.

7. Experimental Results. We conduct the experiments from 2005 to 2015, generating the same number of farms as the real number in each year. In figure 11 we plot the Time Series prediction for centers of the farms created in 2015 compared to the deforestation events that occurred in 2015 according to PRODES data. We also plot the simulated road change in 2015. In figure 12 we plot the simulation result of the change of farms and roads, and how farmers are captured in March 2015. Existing roads are indicated in blue, new roads are in green. The red dots indicate new farms, the black dots indicate existing growing farms.If a farmer gets caught, a pink diamond is shown at the the farm location in the figure. We have 31 farmers getting caught in the simulation for March 2015 and 34 new farms in this month. 194 of the 498 farms are captured in the simulation of the whole year 2015.

8. Conclusion. In our model, we construct the expected profit formula for farmers to decide where to clear land for illegal farming. We also perform time series analysis to simulate the behaviors of farmers and growth of farms, with capture probability built in. We also update the road map based on optimal path planning. For evaluation, we compare our model with real deforestation data and evaluate the results with average F1 score. We also consider how realistic our model is by evaluating

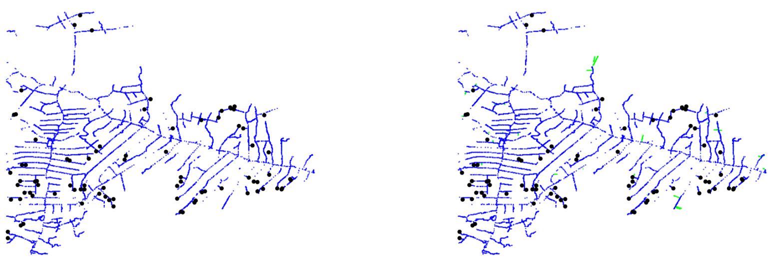


(a) Time Series Model for 2015

(b) Data from PRODES data set for 2015

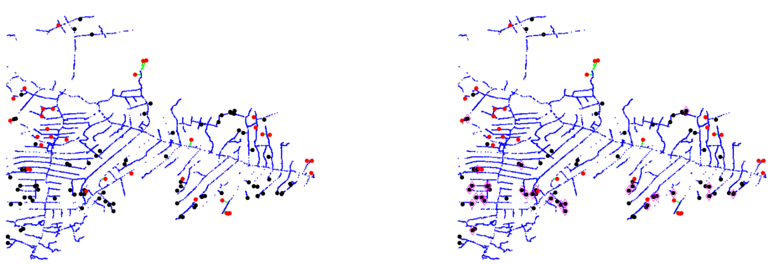
(c) Simulated road change for 2015. Red roads are the new roads.

Fig. 11: Time Series Generated Data compared to Real Data



(a) Existing roads and growing farms.

(b) New roads appear.



(c) New farms are constructed.

(d) Some farms are captured.

Fig. 12: Time Series Simulation for March 2015: blue - existing roads, green - new roads, black - existing farms, red - new farms, pink diamonds - farms captured. In this month, 31 farms are captured, 34 new farms are constructed.

average capture rate, average growing farm rate and average growing time. We compare different patrol strategies by comparing average profit and average capture rate. The strength of our model is that it takes into account many realistic factors such as distance to roads, distance to cities, clustering effect, tree coverage, punishment, capture possibility and maximum operation time for farms. Besides, our model is easy to explain and understand.

In future, we can improve our model by doing some of the following: 1) Further optimize our set of parameters. 2) Consider more realistic factors. 3) Possibly find the optimal patrol strategy using Minimax strategy. 4) Design a better algorithm to model farm expansion.

REFERENCES

[1] H. J. Albers, *Spatial modeling of extraction and enforcement in developing country protected areas*, Resource and Energy Economics, (2010).

[2] D. Arnold, D. Fernandez, R. Jia, C. Parkinson, D. Tonne, Y. Yaniv, A. L. Bertozzi, and S. J. Osher, *Modeling environmental crime in protected areas using the level set method*, SIAM Journal on Applied Mathematics, 79 (2019), pp. 802–821, https://doi.org/10.1137/18M1205339.

[3] J. J. Assunção, C. Gandour, and R. Rocha, *Deforestation slowdown in the Brazilian Amazon: prices or policies?*, 2012.

[4] M. Bardi and I. Capuzzo-Dolcetta, *Optimal control and viscosity solutions of Hamilton-Jacobi-Bellman equations*, Springer Science & Business Media, 2008.

[5] E. Cartee and A. Vladimirsky, *Control-theoretic models of environmental crime - preprint*, 06 2019, https://arxiv.org/abs/1906.09289.

[6] M. G. Crandall and P.-L. Lions, *Two approximations of solutions of Hamilton-Jacobi equations*, Mathematics of Computation, 43 (1984), pp. 1–19.

[7] C. G. Diniz, A. A. d. A. Souza, D. C. Santos, M. C. Dias, N. C. d. Luz, D. R. V. d. Moraes, J. S. Maia, A. R. Gomes, I. d. S. Narvaes, D. M. Valeriano, L. E. P. Maurano, and M. Adami, *DETER-B: The new Amazon near real-time deforestation detection system*, IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 8 (2015), pp. 3619–3628, https://doi.org/10.1109/JSTARS.2015.2437075.

[8] I. C. Dolcetta, *The Hopf-Lax solution for state dependent Hamilton-Jacobi equations (viscosity solutions of differential equations and related topics)*, (2002).

[9] F. Fang, T. H. Nguyen, R. Pickles, W. Y. Lam, G. R. Clements, B. An, A. Singh, B. C. Schwedock, M. Tambe, and A. Lemieux, *PAWS — a deployed game-theoretic application to combat poaching*, AI Magazine, 38 (2017), pp. 23–36, https://doi.org/10.1609/aimag.v38i1.2710, https://www.aaai.org/ojs/index.php/aimagazine/article/view/2710.

[10] D. Kar, F. Fang, F. Delle Fave, N. Sintov, and M. Tambe, *A game of thrones: When human behavior models compete in repeated Stackelberg security games*, in Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems, International Foundation for Autonomous Agents and Multiagent Systems, 2015, pp. 1381–1390.

[11] D. Kar, B. J. Ford, S. Gholami, F. Fang, A. J. Plumptre, M. Tambe, M. Driciru, F. Wanyama, A. Rwetsiba, M. Nsubaga, and J. Mabonga, *Cloudy with a chance of poaching: Adversary behavior modeling and forecasting with real-world poaching data*, in AAMAS, 2017.

[12] W. F. Laurance, A. K. M. Albernaz, G. Schroth, P. M. Fearnside, S. Bergen, E. M. Venticinque, and C. Da Costa, *Predictors of deforestation in the Brazilian Amazon*, Journal of Biogeography, 29 (2002), pp. 737–748, https://doi.org/10.1046/j.1365-2699.2002.00721.x, https://onlinelibrary.wiley.com/doi/abs/10.1046/j.1365-2699.2002.00721.x, https://arxiv.org/abs/https://onlinelibrary.wiley.com/doi/pdf/10.1046/j.1365-2699.2002.00721.x.

[13] G. O. Mohler, A. L. Bertozzi, T. A. Goldstein, and S. J. Osher, *Fast TV regularization for 2D maximum penalized likelihood estimation*, Journal of Computational and Graphical Statistics, 20 (2011), pp. 479–491.

[14] N. I. of Space Research (INPE), *PRODES deforestation. Accessed through Global Forest Watch in 07/2019. www.globalforestwatch.org.*

[15] S. Osher and J. Sethian, *Fronts propagating with curvature dependent speed: Algorithms based on Hamilton-Jacobi formulations*, Journal of Computational Physics, 79 (1988), pp. 12–49.

[16] A. S. P. Pfaff, *What drives deforestation in the Brazilian Amazon?: Evidence from satellite and socioeconomic data*, Journal of Environmental Economics and Management, 37 (1999), pp. 26 – 43, https://doi.org/https://doi.org/10.1006/jeem.1998.1056, http://www.sciencedirect.com/science/article/pii/S0095069698910567.

[17] T. Slough, J. Urpelainen, and J. H. SAIS, *Public policy under limited state capacity: Evidence from deforestation control in the Brazilian Amazon*, tech. report, mimeo, 2018.